Department of Communications and Networking

Security from Implicit Information

Le Nguyen Ngu Nguyen

Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 128/2020

Security from Implicit Information

Le Nguyen Ngu Nguyen

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Electrical Engineering, Remote connection link https://aalto.zoom.us/j/6549167529, on 18 September 2020 at 17:00 EEST.

Aalto University
School of Electrical Engineering
Department of Communications and Networking
Ambient Intelligence Group

Supervising professor
Professor Stephan Sigg, Aalto University, Finland

Preliminary examiners
Professor Florian Alt, Bundeswehr University Munich, Germany
Professor René Mayrhofer, Johannes Kepler University Linz, Austria

Opponent
Professor Veljko Pejovic, University of Ljubljana, Slovenia

Aalto University publication series
DOCTORAL DISSERTATIONS 128/2020

ISBN 978-952-64-0012-9 (printed)
ISBN 978-952-64-0013-6 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-64-0013-6

Unigrafia Oy
Helsinki 2020

Finland

NORDIC SWAN ECOLABEL
Printed matter
4041-0619

Aalto University

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Abstract

Author
Le Nguyen Ngu Nguyen

Name of the doctoral dissertation
Security from Implicit Information

Publisher School of Electrical Engineering

Unit Department of Communications and Networking

Series Aalto University publication series DOCTORAL DISSERTATIONS 128/2020

Field of research Networking Technology

Manuscript submitted 27 July 2020 **Date of the defence** 18 September 2020

Permission for public defence granted (date) 24 August 2020 **Language** English

☐ **Monograph** ☒ **Article dissertation** ☐ **Essay dissertation**

Abstract

We present novel security mechanisms using implicit information extracted from physiological, behavioural, and ambient data. These mechanisms are implemented with reference to device-to-user and inter-device relationships, including: user authentication with transient image-based passwords, device-to-device secure connection initialization based on vocal commands, collaborative inference over the communication channel, and continuous on-body device pairing.

Authentication methods based on passwords require users to explicitly set their passwords and change them regularly. We introduce a method to generate always-fresh authentication challenges from videos collected by wearable cameras. We implement two password formats that expect users to arrange or select images according to their chronological information.

Radio waves are mainly used for data transmission. We implement function computation over the wireless signals to perform collaborative inference. We encode information into burst sequences in such a way that arithmetic functions can be computed using the interference. Hence, data is hidden inside the wireless signals and implicitly aggregated. Our algorithms allow us to train and deploy a classifier efficiently with the support of minimal backscatter devices.

To initialize a connection between a personal device (e.g. smart-phone) and shared appliances (e.g. smart-screens), users are required to explicitly ask for connection information including device identities and PIN codes. We propose to leverage natural vocal commands to select shared appliance types and generate secure communication keys from the audio implicitly. We perform experiments to verify that device proximity defined by audio fingerprints can restrict the range of device-to-device communication.

PIN codes in device pairing must be manually entered or verified by users. This is inconvenient in scenarios when pairing is performed frequently or devices have limited user interfaces. Our methods generate secure pairing keys for on-body devices continuously from sensor data. Our mechanisms automatically disconnect the devices when they leave the user's body. To cover all human activities, we leverage gait in human ambulatory actions and heartbeat in resting postures.

Keywords security, implicit information, sensor data, signal processing, machine learning

ISBN (printed) 978-952-64-0012-9 **ISBN (pdf)** 978-952-64-0013-6

ISSN (printed) 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2020

Pages 228 **urn** http://urn.fi/URN:ISBN:978-952-64-0013-6

Preface

I really appreciate the guidance and support of Professor Stephan Sigg, my supervisor. I would like to thank my colleagues: Dr Rainhard D. Findling, Jari Lietzen, Dr Kalle Ruttik, Dr Sameera Palipana, Pablo Perez Zarazaga, Professor Tom Bäckström, Viktor Nässi, and others at Aalto University. I would like to thank my collaborators: Dr Dominik Schürmann, Arne Brüsch, Professor Lars Wolf, Nico Jähne-Raden, Dr Ulf Kulau, Dr Maija Poikela, Professor Kenji Leibnitz, Dr Theofilis Konstantinos, and Professor Ferdinand Peper.

We acknowledge the support of funding organizations, including The Academy of Finland, The German Academic Exchange Service, and The European Institute of Innovation & Technology.

Espoo, Finland, September 1, 2020,

Le Nguyen Ngu Nguyen

Contents

Preface 3

Contents 5

List of Publications 9

Author's Contribution 11

List of Figures 15

List of Tables 19

Abbreviations 21

Symbols 23

1. Introduction 25

1.1 Motivation . . . 26
1.2 Research Questions . . . 27
1.3 Contributions . . . 29
1.3.1 Always-fresh Authentication Challenges . . . 29
1.3.2 Collaborative Inference based on Implicit Data Aggregation . . . 31
1.3.3 Proximity-based Secure Communication using Vocal Commands . . . 32
1.3.4 Continuous Secure Device Pairing . . . 33
1.4 Dissertation Structure . . . 34

2. Background 37

2.1 Sources of implicit information . . . 38
2.2 Preprocessing Sensor Data . . . 39
2.3 Extracting Implicit Information from Sensor Data . . . 40
2.4 Applications of Implicit Information . . . 41
2.5 Summary . . . 43

3. Always-fresh Authentication Challenges 45
3.1 User Authentication . . . 45
3.2 Image-based Password Design . . . 47
3.3 Generating Passwords from First-person-view Videos . . . 48
3.3.1 Key Frame Selection . . . 48
3.3.2 Image Features . . . 49
3.3.3 Image Memorability Classification . . . 50
3.3.4 Segmentation . . . 50
3.3.5 Clustering . . . 51
3.4 Case Studies . . . 52
3.4.1 Image-arrangement Study . . . 53
3.4.2 Image-selection Study . . . 55
3.5 Security Analysis . . . 58
3.5.1 Brute Force Attacks . . . 58
3.5.2 Smudge-based Attacks . . . 58
3.5.3 Observation Attacks . . . 59
3.6 Conclusion . . . 60

4. Collaborative Inference based on Implicit Data Aggregation 61
4.1 Training Classifiers across Vertically-Partitioned Data . . . 61
4.2 Computation over Backscattered RF Signals . . . 63
4.3 Security Analysis . . . 67
4.3.1 Model Confidentiality . . . 67
4.3.2 Implicit Data Perturbation . . . 67
4.4 Experiments . . . 68
4.4.1 Decentralized Training . . . 69
4.4.2 Classification Offloading in Backscatter Sensor Networks 70
4.4.3 Effect of Environmental Changes . . . 73
4.5 Conclusion . . . 74

5. Proximity-based Secure Communication using Vocal Commands 77
5.1 Context-based Device Pairing . . . 77
5.2 Audio-based Pairing Protocols . . . 78
5.2.1 Audio-based Pairing Protocols and Application Scenarios 78
5.2.2 Security Analysis . . . 80
5.3 Similarity of Audio Fingerprints in Different Vocal Commands . 82
5.4 Impact of SNR on Audio Pairing . . . 83
5.5 Human and Hardware Eavesdroppers . . . 85
5.5.1 Human Eavesdropper . . . 87
5.5.2 VUI Eavesdropper . . . 87
5.6 Conclusion . . . 88

6. Continuous Secure Device Pairing 89
6.1 Device Pairing . . . 89
6.2 Gait-based Pairing of On-body Devices . . . 91

6.2.1 Body Area Network Device-to-device Authentication using Natural Gait . . . 92
6.2.2 Security Analysis . . . 94
6.3 Representation Learning from Ballistocardiography Data . . . 95
6.3.1 Ballistocardiography . . . 95
6.3.2 Heart-beat Detection . . . 96
6.3.3 Representation Learning Model Architecture . . . 96
6.3.4 Security Analysis . . . 98
6.4 Experiments . . . 99
6.4.1 Evaluation of Gait Fingerprints . . . 99
6.4.2 Video-based Attack on Gait-based Device Pairing . . . 101
6.4.3 Evaluation of Heartbeat Fingerprints . . . 103
6.5 Conclusion . . . 105

7. Conclusions and Future Work 107
7.1 Discussion . . . 107
7.2 Always-fresh Authentication Challenges . . . 109
7.3 Collaborative Inference based on Implicit Data Aggregation . . . 109
7.4 Proximity-based Secure Communication using Vocal Commands 110
7.5 Continuous Secure Device Pairing . . . 110
7.6 Future Work . . . 111

References 113

Errata 125

Publications 127

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I Dominik Schürmann, Arne Brüsch, Ngu Nguyen, Stephan Sigg, and Lars Wolf. Moves like Jagger: Exploiting variations in instantaneous gait for spontaneous device pairing. *Pervasive and Mobile Computing*, Volume 47, Pages 1-12 , July 2018.

II Arne Brüsch, Ngu Nguyen, Dominik Schürmann, Stephan Sigg, and Lars Wolf. Security Properties of Gait for Mobile Device Pairing. *IEEE Transactions on Mobile Computing*, All authors contributed equally to this work and are listed in alphabetical order, February 2019.

III Ngu Nguyen, Rainhard D. Findling, and Stephan Sigg. Always-fresh Authentication Challenges from Videos Captured by a Body-worn Camera. *IEEE Transactions on Mobile Computing*, submitted, 2020.

IV Ngu Nguyen, Stephan Sigg, Jari Lietzen, Rainhard D. Findling, and Kalle Ruttik. Collaborative Inference – Training a Distributed Learner for Smart Environments. *IEEE Transactions on Mobile Computing*, submitted, 2020.

V Ngu Nguyen and Stephan Sigg. PassFrame: Generating image-based passwords from egocentric videos. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, USA, March 2017.

VI Ngu Nguyen and Stephan Sigg. User Authentication based on Personal Image Experiences. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Greece, March 2018.

VII Ngu Nguyen, Nico Jähne-Raden, Ulf Kulau, and Stephan Sigg. Representation Learning for Sensor-based Device Pairing. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Greece, March 2018.

VIII Ngu Nguyen and Stephan Sigg. Secure Context-based Pairing for Unprecedented Devices. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Greece, March 2018.

IX Ngu Nguyen and Stephan Sigg. Learning with Vertically-Partitioned Data, Binary Feedback, and Random Parameter Update. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, France, April 2019.

X Stephan Sigg, Ngu Nguyen, Pablo Perez Zarazaga, and Tom Bäckström. Provable Consent for Voice User Interfaces in Indoor Environments. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, USA, March 2020.

Author's Contribution

Publication I: "Moves like Jagger: Exploiting variations in instantaneous gait for spontaneous device pairing"

Dominik Schürmann and Arne Brüsch proposed and implemented the gait-based pairing protocol. Dominik Schürmann and Arne Brüsch performed the experiments on inertial datasets. Stephan Sigg evaluated the entropy of the gait fingerprints. Ngu Nguyen recorded and analyzed the videos to extract gait information. Ngu Nguyen evaluated gait fingerprints extracted from videos and wearable sensors. All authors wrote the article together.

Publication II: "Security Properties of Gait for Mobile Device Pairing"

Dominik Schürmann and Arne Brüsch implemented and compared the gait-based pairing protocols. Dominik Schürmann, Arne Brüsch, and Stephan Sigg analyzed the security threats of these protocols. Ngu Nguyen conducted the video-based attacking experiment, analyzed the recorded data, and compared the effectiveness of this attack on the pairing protocols. All authors contributed equally to write the article.

Publication III: "Always-fresh Authentication Challenges from Videos Captured by a Body-worn Camera"

Ngu Nguyen proposed the idea of generating transient image-based authentication challenges and implemented the software for experiments. Ngu Nguyen performed the experiments and analysed the results together with Stephan Sigg. Rainhard D. Findling and Ngu Nguyen analyzed the security threats of the approach. All authors contributed equally to write the article.

Publication IV: “Collaborative Inference – Training a Distributed Learner for Smart Environments”

Ngu Nguyen proposed the idea of implementing the classification model on the wireless channel. Ngu Nguyen implemented the algorithms and performed expriments with the backscatter devices. Ngu Nguyen evaluated the power consumption. Ngu Nguyen analyzed the convergence of the algorithms with respect to the amount of data and the power consumption. Stephan Sigg and Ngu Nguyen discussed to design and set-up the experiments. Stephan Sigg designed the computation offloading protocol. Jari Lietzen and Kalle Ruttik designed the backscatter hardware. Jari Lietzen produced and tested the hardware. Rainhard D. Findling contributed to the design of the experiments and the algorithm implementation. All authors contributed equally to write the article.

Publication V: “PassFrame: Generating image-based passwords from egocentric videos”

Ngu Nguyen proposed the initial idea of using images from a wearable camera to generate passwords. Ngu Nguyen and Stephan Sigg discussed to design the password generation algorithm and the password formats. Ngu Nguyen implemented the prototype for experiments. Ngu Nguyen conducted the experiments. All authors contributed equally to analyze the results and write the article.

Publication VI: “User Authentication based on Personal Image Experiences”

Ngu Nguyen proposed, designed and implemented the system. Ngu Nguyen and Stephan Sigg discussed to design the forms of authentication challenges. All authors contributed equally to conduct the experiments, analyze the results, and write the article.

Publication VII: “Representation Learning for Sensor-based Device Pairing”

Ngu Nguyen proposed the idea of using heartbeats for device pairing, implemented the algorithms, and performed experiments. Nico Jähne-Raden and Ulf Kulau provided the dataset and the medical knowledge on BCG. Ngu Nguyen and Stephan Sigg analyzed the experimental results. All authors contributed equally to write the article.

Publication VIII: “Secure Context-based Pairing for Unprecedented Devices”

Ngu Nguyen proposed the idea of using vocal commands to select and pair devices. Ngu Nguyen and Stephan Sigg designed the communication protocol and the experiments. Ngu Nguyen performed the experiments. Both authors contributed equally to analyze the experimental results and write the article.

Publication IX: “Learning with Vertically-Partitioned Data, Binary Feedback, and Random Parameter Update”

Ngu Nguyen proposed, implemented, and evaluated the algorithm. Ngu Nguyen and Stephan Sigg discussed to improve the optimization algorithm and analyze the experimental results. Both authors contributed equally to write the article.

Publication X: “Provable Consent for Voice User Interfaces in Indoor Environments”

Stephan Sigg proposed the idea during the discussion with Pablo Perez Zarazaga and Tom Bäckström. Stephan Sigg designed the experiments. Ngu Nguyen designed the communication protocol. Ngu Nguyen collected data to evaluate audio fingerprint similarity in different loudness levels. Pablo Perez Zarazaga collected audio data in the eavesdropping scenarios while Ngu Nguyen analyzed the audio fingerprints to evaluate the security aspects. All authors contributed equally to write the article.

Confirmation

The language of my dissertation has been checked by the Institute of Language Checks. I have personally examined and accepted/rejected the results of the language check one by one. This has not affected the scientific content of my dissertation.
I hereby confirm the author’s contribution has been approved by other authors.
Author’s signature:

Supervisor’s signature:

List of Figures

1.1 ONE-TO-ONE in Chapter 3: Always-fresh authentication challenges from first-person-view videos 29
1.2 MANY-TO-ONE in Chapter 4: Numerous distributed sensors collaborate to learn a machine learning model across vertically-partitioned data 31
1.3 ONE-TO-MANY in Chapter 5: Audio-based secure connection initialization of new devices using vocal commands 32
1.4 MANY-TO-MANY in Chapter 6: Two application scenarios of our pairing mechanisms, including: 1.4a on-body devices with ambulatory activities and 1.4b on-body devices with resting postures. 33
1.5 Outline of the dissertation 34

2.1 Techniques to extract implicit information from sensing data . . 38

3.1 Potential layouts of our authentication challenges: (a) and (b) are *Image-selection* while (c) and (d) are *Image-arrangement*. Both schemes can be adapted to different screen sizes by varying the number of images. 46
3.2 The process of generating authentication challenges from first-person-view videos 48
3.3 Sample images extracted from our egocentric videos. They vary in activities (e.g. holding objects, playing sport, traveling on public transport, and social interaction), locations (indoor and outdoor), weather conditions (e.g. winter and summer), and the light intensity (e.g. sunny day, having dinner at night, and dark scene). 52
3.4 The devices used to collect video data: Transcend DrivePro^{TM} Body 10 camera (a) and Pupil Labs headset (b) 53

3.5 User effort in terms of the number of attempts and the entry time in the object-interaction scenario with *image-arrangement* authentication challenges. The prototype supports slide-and-swipe interaction to decrease the entry time. 55
3.6 User effort in terms of the number of clicks and the entry time (duration) in the daily condition study with *image-selection* authentication challenges 55
3.7 Number of possible answers (permutation or combination) of our authentication mechanism 58

4.1 Protocols for the distributed gradient descent algorithm between the coordinator and all n distributed devices. The schematic displays an update process with respect to device i on one sample. The process is repeated with new sensing data. Each distributed device i stores its current weight w_i, the previous weight w_i', and the learning rate λ to control the convergence speed. 62
4.2 Computation offloading and aggregation of the weighted feature values $w_i x_i$ during simultaneous superimposed transmission. Weighted feature values are transmitted as the mean $w_i x_i = \mu_i$ of a Poisson-distributed burst sequence. The coordinator estimates the mean $\sum_i \mu_i$ of the convoluted Poisson-distributed sequence to obtain an estimate on $\sum_i w_i x_i$. 63
4.3 Power consumption of the systems implementing our approach and the centralized logistic regression (Active LoRa and Backscatter PLoRa [116]) 71
4.4 Confusion matrices on INDOOR dataset [27] 71
4.5 Confusion matrices on OUTDOOR dataset [45] 71
4.6 Confusion matrices on INTRUSION dataset [85] 71
4.7 Confusion matrices on PHISHING dataset [105] 71
4.8 Layout of our experiments on the operating distance 73
4.9 Signal-to-noise ratio (SNR) with various distances 73
4.10 Layout of devices in the experiments on environmental variation 73
4.11 Evaluation of the burst detection algorithm with various environmental conditions 75

5.1 Protocol for key exchange and management without a central authority 80
5.2 Protocol for device group formation based on context 80
5.3 Protocol for context-based device discovery 81
5.4 Similarity of audio fingerprints in our experiments 84
5.5 Similarity of audio fingerprints in the eavesdropping experiments 86

6.1 BANDANA device pairing protocol based on gait information . . 92

6.2 Conceptual view of the protocol with attack vectors (blue line depicts device boundary, dashed lines indicate communication between the devices) . . . 94

6.3 Similarity of intra-body and inter-body gait fingerprints produced by BANDANA. Each value in the intra-body region is defined by the similarity of fingerprints extracted from two different locations on the same body (all possible combinations within each body). Each value in the inter-body region is defined by the similarity of fingerprints from different bodies at the same location. . . . 99

6.4 Distribution of p-values achieved for fingerprints after 20 runs of the DieHarder set of statistical tests. The tests are: (1) birthdays, (2) operm5, (3) rank32x32, (4) rank6x8, (5) bitstream, (6) opso, (7) oqso, (8) dna, (9) count-1s-str, (10) count-1s-byt, (11) parking, (12) 2D circle, (13) 3D sphere, (14) squeeze, (15) runs, (16) craps, (17) marsaglia, (18) sts monobit, (19) sts runs, (20) sts serial[1-16], (21) rgb bitdist[1-12], (22) rgb minimum distance[2-5], (23) rgb permutations[2-5], (24) rgb lagged sum[0-32], (25) rgb kstest, (26) dab bytedistrib, (27) dab dct 256, (28) dab fill tree, (29) dab fill tree 2, and (30) dab monobit 2. . . . 100

6.5 Experimental setup for video-based attack on gait-based pairing 101

6.6 Gait cycles extracted from shin and waist. . . . 102

6.7 Acceleration signals featuring different noise levels . . . 103

6.8 Attacks using Video-based impersonation: Similarity of gait-fingerprints with different noise levels over four pairing schemes 104

6.9 Siamese auto-encoder to extract fingerprints from heartbeat acceleration data . . . 104

6.10 Similarity of heartbeat fingerprints on the same user and on different users . . . 104

List of Tables

1.1 Contributions of this dissertation (contribution categories: conceptual [$\mathscr{C}$], methodological [$\mathscr{M}$], technical [$\mathscr{T}$], and empirical [$\mathscr{E}$]) 30

3.1 Average entry time in seconds and number of clicks to achieve partial and total correctness in the *image-selection* scheme. The numbers in brackets are standard deviation. 57
3.2 Observation attacks and proposed countermeasure methods . . 59

4.1 Datasets of pervasive systems used to compare decentralized and centralized learning 69
4.2 Benchmark datasets used to compare decentralized and centralized learning 70

5.1 Average similarity (%) of audio fingerprints. Values in brackets are standard deviation. 83

Abbreviations

BCG Ballistocardiography

CNN Convolutional Neural Network

DBSCAN Density-Based Spatial Clustering of Applications with Noise

ECC Error-Correcting Code

FIR Finite Impulse Response

IIR Infinite Impulse Response

IoT Internet-of-Things

KGF Key Generation Function

MAE Mean Absolute Error

MAPE Mean Absolute Percentage Error

PCA Principal Component Analysis

PD Personal Device

PIN Personal Identification Number

OFDMA Orthogonal Frequency-Division Multiple Access

RF Radio Frequency

RFID Radio-Frequency Identification

SA Shared Appliance

SDR Software Defined Radio

SNR Signal-to-Noise Ratio

TIMIT The Texas Instruments/Massachusetts Institute of Technology corpus of read speech

USRP Universal Software Radio Peripheral

VUI Voice User Interface

Symbols

$\mathbf{a}$ coefficient vector

$\mathbf{b}$ bias vector

B blur image

E error function

$\mathbf{f}$ fingerprint (binary vector) characterizing a sequence of audio or inertial data

F image

l negative log likelihood function

$\mathbf{o}$ output vector

$\mathbf{w}$ parameter vector or weight vector

$\mathbf{W}$ parameter matrix or weight matrix

$\mathbf{x}$ feature vector or input vector of a machine learning model

1. Introduction

Globally, in 2020, it is estimated that there are 28.5 billion networked devices such as phones, wearables, and ambient sensors [1]. These devices are able to capture a huge amount of data about users and the environment. For example, smart-phones are typically equipped with multiple sensors to record audio, video, and even users' movements. Smart-watches, which are in contact with the human skin, can sense medical information such as heart-rate and blood pressure. Wearable cameras emerge as instruments to record first-person-view videos, especially in sport and leisure activities. Smart-home systems that comprise a multitude of sensors have been installed to improve user's comfort. Analyzing the sensor data can reveal IMPLICIT INFORMATION of users and the environment. For instance, the characteristics of human gait can be extracted from visual and inertial data [124]. Another example is the motif discovery of users' routines from first-person-view videos captured with a body-cam [169]. Gradually, more and more useful information have been collected by smart devices using their built-in sensors. This provides opportunities to address existing issues of conventional security mechanisms through leveraging IMPLICIT INFORMATION extracted from sensor data. We raise the research question:

> *How to use implicit information to improve security of user-to-device and device-to-device relationships?*

In this dissertation, we design and implement methods that utilize implicit information derived from sensor data to improve the security of smart devices. We categorize the relationships between different smart devices, as well as between these devices and their users, into four types: ONE-TO-ONE, ONE-TO-MANY, MANY-TO-ONE, and MANY-TO MANY. In the context of one device and its user (i.e. ONE-TO-ONE), we introduce a password generation method that personalizes authentication challenges according to each user's activities (see Chapter 3). When there are wirelessly-connected devices to implement a machine learning model (i.e. MANY-TO-ONE), we introduce a distributed training algorithm in

[1] Cisco VNI Forecast Highlights: https://www.cisco.com/c/m/en_us/solutions/service-provider/vni-forecast-highlights.html [Accessed: Jan 18, 2020]

which data and model parameters are scattered across all participant entities (see Chapter 4). Next, we assist users to establish a secure connection between their personal devices and shared appliances in a new environment (i.e. ONE-TO-MANY), using vocal commands (see Chapter 5). Finally, when there are many devices seeking to form a peer-to-peer network (i.e. MANY-TO-MANY), we continuously extract secret keys from inertial data to facilitate continuous secure device pairing (see Chapter 6).

1.1 Motivation

It is essential to protect the data and services of smart devices from unauthorized access [110]. Authentication is the mechanism to ensure the user's identity. There exist three elements used for authentication: something the user knows (such as a password), something the user possesses (such as a physical key or a bank card), and something that characterizes the user (such as fingerprints and other biometric data) [163]. Almost all forms of user authentication require users to *explicitly* input secret information. To strengthen the authentication process, multiple elements can be combined to form multi-factor authentication schemes, such as a bank card with a Personal Identification Number (PIN). Although passwords have a long history of usage, they suffer from shoulder-surfing attacks [49]. Moreover, setting easily-guessed codes and using the same passwords for a long time are the common reasons of making this authentication method ineffective [98]. In Chapter 3, we propose and evaluate novel user authentication schemes that analyse first-person-view videos to generate always-fresh image-based challenges. Our mechanism releases users from fixed passwords that are required to be updated regularly.

Smart sensing devices are not only carried by users but are also integrated into infrastructure such as in smart-home. These systems are comprised of multiple data acquisition units such as thermometers, motion detectors, and cameras. These sensors aim to collect data about users and the environment they inhabit to enhance their comfort and convenience. For example, a occupancy-monitoring system can recognize users' activities to control light intensity and room temperature for balancing utility and energy efficiency. Typically, such a system has one central component that collects and analyzes all data from the distributed sensors to infer situations of the monitored space [6]. This set-up requires the sensors to transmit their collected data over the network, which consumes a significant amount of energy. Backscatter communication has been recently considered as a solution to reduce the resource consumption [73]. However, the existing approaches concentrate on data transmission only. In Chapter 4, we propose algorithms that efficiently train a classification model over data allocated across wirelessly-connected sensors. Our approach employs simple backscatter devices [13] that collaboratively implement a shared inference model through implicit data aggregation on the communication channel.

Nowadays, it is common for a typical user to interact with more than one smart device. For example, one can use a laptop for working, a smart-phone for communication, a smart-watch for health monitoring, and a tablet for entertainment. To exchange data, these devices require a secure communication channel among them; hence, it is necessary for them to authenticate each other mutually. This becomes more challenging whenever users enter a new environment equipped with shared appliances. We require a seamless mechanism to connect and disconnect the user's personal device with shared appliances for exchanging data securely. There are smart objects that are temporarily used by multiple users, such as sensor-equipped shopping carts [161]. A widely-used protocol is to *pair* these devices using a Bluetooth connection [61]. This process asks users to verify generated numeric codes, which is obtrusive from the perspective of users. Some devices are paired via a fixed PIN code and users rely on their identifiers to select the right ones. Furthermore, the emergence of wearable gadgets and electronic textiles with limited user interfaces introduces new challenges for PIN-based pairing. We propose to utilize the implicit information captured by smart devices such as characteristics of ambient audio (see Chapter 5) and behavioural data (see Chapter 6) to address these challenges. Our proposed approaches facilitate the seamless connection of smart devices, the continuous changing of shared secure keys, and the automatic disconnection of these devices.

1.2 Research Questions

The dissertation can be considered from two perspectives: the range of implicit information that smart devices can collected and the paradigm of device relationship. The devices are equipped with multiple sensors to collect environmental and physiological observations, from which we can extract implicit information. We can roughly categorize implicit information into ENVIRONMENT and PERSONAL information. The recordings possess characteristics listed below:

- Transient: implicit information varies constantly over time. Furthermore, we can rarely capture two identical sequences of samples due to hardware and data diversity.
- Implicit: The captured sensor data is always available but uncontrollable from the perspective of smart devices. It contains implicit information that is influenced by users and the environment.
- Diverse: implicit information originates from various sources each of which characterizes an aspect about the users and the environment. Hence, they require different processing techniques.

Analyzing implicit information, we propose novel security mechanisms in four relationships between smart devices and their users as well as between devices: user authentication, collaborative inference, device selection, and device pairing.

ONE-TO-ONE: Before using services or accessing data in a personal device (e.g. a smart-phone), a user is required to prove their identity, through an authentication procedure. Conventional passwords (e.g. alphanumeric or graphical) are selected by users and fixed until being changed by the users, which may not happen regularly. Hence, such passwords suffer from vulnerability to shoulder-surfing [49] and side-channel attacks [12]. This has raised the first research question:

How to authenticate with always-fresh passwords leveraging implicit information?

MANY-TO-ONE: Now consider a network of many or even countless wirelessly-connected smart devices, which can collect a huge amount of data. These devices collaborate to facilitate intelligent services. During their lifetime, they require energy to operate and transmit their data over wireless signals. Their data may contain sensitive information that users do not agree to share widely. These observations have led to the second research question:

How to secure data aggregation in collaborative inference through implicit information convolution?

ONE-TO-MANY: Consider that a user wants to connect a personal device to shared smart-screens or a Bluetooth speaker that supports a voice user interface (VUI) [119]. Normally, within a new environment equipped with public appliances, a user is required to obtain a device identifier and a PIN code, which is inconvenient. For initializing a secure connection between personal devices and new shared appliances, the third research question is:

How to use implicit information in vocal commands to select shared devices naturally and securely?

MANY-TO-MANY: When there are multiple devices sharing data, to securely initiate the device-pairing process (e.g. using Bluetooth [61]), users enter or verify PIN codes, which can be cracked [132]. Mis-binding is another threat when devices are paired to a malicious entity [131]. Moreover, when the sharing ends, these devices often have to be dis-connected manually. Hence, this procedure is obtrusive and not seamlessly. Hence, we raise the research question:

How to continuously generate secret keys for device pairing from implicit information?

In Section 1.3, we summarize the contributions of this dissertation to tackle the aforementioned research questions in the following key areas: user authentication, collaborative inference, device selection, and device pairing. We are aware that there are other types of relationships in each of these areas. For instance, a user can be authenticated to multiple devices or several users can share the same devices (with different user accounts). In this dissertation, we focus on the one-to-one scenario since it is the fundamental case that can be extended to others. Another example is that in the sensor network, a node

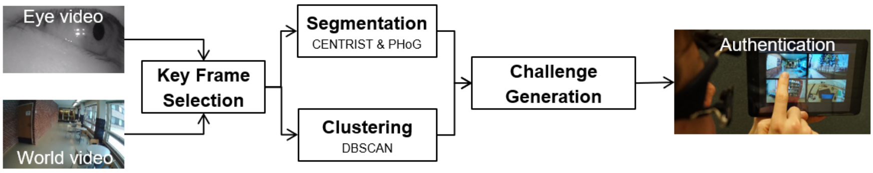


Figure 1.1. ONE-TO-ONE in Chapter 3: Always-fresh authentication challenges from first-person-view videos

can share their information with its neighbours. This requires communication between devices, which consumes much energy.

1.3 Contributions

The dissertation investigates implicit information captured by the built-in sensors of smart objects to implement novel secure collaboration mechanisms. We have investigated a variety of sensors, including: audio, video, acceleration, electroencephalography, and radio signal. After recording the sensed data, we analyzed ever-changing implicit information extracted from body movement, ambient audio, first-person-view videos, behavioural data, and wireless signals to facilitate our security features. Our approaches personalized challenges in user authentication according to users' behaviour and environmental situations. We constantly updated secret information in device-to-device communication without explicit user interaction. We hid sensing data implicitly through signal interference in wireless sensor networks during the optimization of an inference model. Our proposed systems have leveraged the characteristics of implicit information to improve the security, convenience, and efficiency of user authentication, data aggregation, device selection, and device pairing (see Table 1.1 for a summary).

1.3.1 Always-fresh Authentication Challenges

User authentication can be considered as a ONE-TO-ONE relationship between the smart device and its owner. We proposed to utilize implicit information extracted from visual data in password generation (see Chapter 3). Our idea is illustrated in Figure 1.1 with a four-image challenge displayed on the tablet screen. The authentication challenges consist of images captured from the user's point of view. Hence, these images are personalized according to the behaviour of users over time. Due to their origin, authentication challenges are temporary for each log-in session. Our approach resists shoulder-surfing attacks and allows the users to be free from the obligation to change passwords regularly. Our password designs are beneficial from the convenience of graphical pattern passwords [17]. Especially, they rely on visual representation, which is far easier to memorize than alphanumeric characters. We implemented the

Table 1.1. Contributions of this dissertation (contribution categories: conceptual [$\mathscr{C}$], methodological [$\mathscr{M}$], technical [$\mathscr{T}$], and empirical [$\mathscr{E}$])

Chapter	**Contribution**	**Publications**
Chapter 3: ONE-TO-ONE	We propose to utilize first-person-view videos to generate authentication challenges that are always fresh and personalized to individuals. We perform case studies to evaluate the security and usability of our proposed approach. We contributed a concept of authentication challenges comprised of first-person-view images [$\mathscr{C}$], along with a technical implementation to realize the concept using video analysis techniques [$\mathscr{T}$].	Publication III, Publication V, and Publication VI
Chapter 4: MANY-TO-ONE	We introduce a method to efficiently train a machine learning model across data distributed vertically in a sensor network. Our technique implicitly protects the sensing data in burst sequences as a result of wireless signal interference and transfers the ownership of model parameters to distributed devices. We contributed a computation-offloading mechanism of machine learning models [$\mathscr{M}$] as well as empirical experiments with backscatter communication [$\mathscr{E}$].	Publication IV and Publication IX
Chapter 5: ONE-TO-MANY	We proposed a mechanism to connect new devices using natural vocal commands and ambient audio. We analyzed the recorded speech to identify appliance types and derive encryption keys for device-to-device secure communication. We performed experiments with human and hardware attackers to verify the security of our proposed approach. We contributed a technical implementation of our voice-based device-selection mechanism [$\mathscr{T}$] as well as empirical experiments with human and hardware eavesdroppers [$\mathscr{E}$].	Publication VIII and Publication X
Chapter 6: MANY-TO-MANY	We analyzed implicit information extracted from human gait and heart beats to facilitate continuous device pairing in on-body settings. We propose a video-based attack that can infer human gait from high-resolution video recordings. We contributed a technical implementation to continuously generate pairing keys from gait information [$\mathscr{T}$] and an empirical video-based attack of several gait-based pairing protocols [$\mathscr{E}$]. In addition, we proposed a feature-learning method to generate fingerprints of sensor data [$\mathscr{M}$].	Publication I, Publication II, and Publication VII

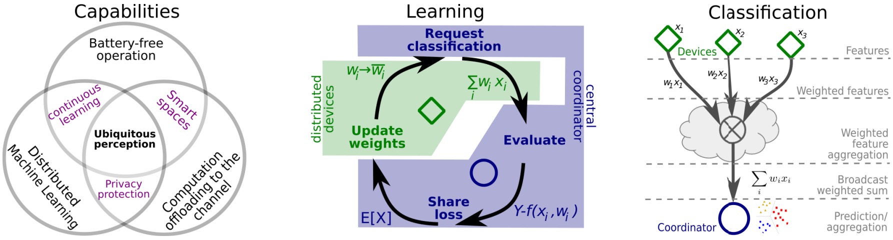


Figure 1.2. MANY-TO-ONE in Chapter 4: Numerous distributed sensors collaborate to learn a machine learning model across vertically-partitioned data

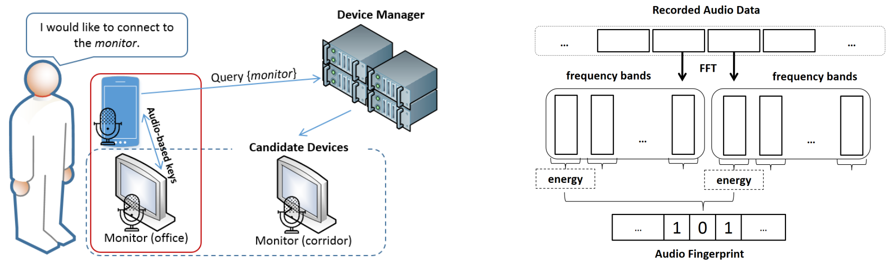
mechanism in a touch-based user interface to evaluate its security and usability.

We realized a procedure to generate transient authentication challenges from a continuous stream of visual information. The data source came from first-person-view videos captured by wearable cameras. We assumed that there existed a secure connection between the data source and the devices that users were authenticating to. After recording the videos, we filtered blurred frames to retain only informative images. We then segmented them into separate scenes depending on the observations and activities of users. After that, authentication challenges were formed into two formats: the first one asked the user to arrange these images in the correct chronological order while the second required the user to select the images that satisfied a temporal condition.

While conventional authentication methods rely on fixed passwords that the user has to change periodically, our authentication challenges are generated according to the personal experience of users. Hence, these challenges are varied in each log-in attempt. Our approach increases the resistance to shoulder-surfing and smudge attacks and releases users from the burden of changing passwords regularly. Furthermore, the images forming our challenges originate from personal observations and activities, which are more memorable than sequences of alphanumeric characters [41].

1.3.2 Collaborative Inference based on Implicit Data Aggregation

A sensor network represents a MANY-TO-ONE relationship in which smart devices collaborate to perform a shared task, such as inferring situations of the monitored environment. We propose a collaborative paradigm to infer the environmental situations through offloading the classification model partially to the wireless channel, and in the meantime, hiding the sensitive information in superimposed signals (see Chapter 4). Figure 1.2 illustrates our approach that implements a logistic regression model distributed across vertically-partitioned data (i.e. each sensor collects one attribute of the observed environment such as temperature or humidity). We aggregate the sensing data implicitly in the signal interference. Based on the proposed method, we develop a model training procedure that can optimize a shared classifier with minimal feedback from a coordinator. Later, the classification model can be used while all of its parame-



(a) Using vocal commands to securely select appliances

(b) Audio fingerprinting

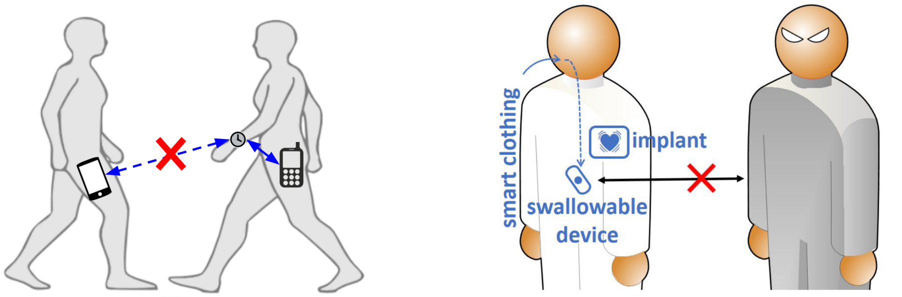
Figure 1.3. ONE-TO-MANY in Chapter 5: Audio-based secure connection initialization of new devices using vocal commands

ters are distributed among sensor nodes. Hence, our approach not only hides the sensitive data but also protects the model parameters.

We realize our algorithms in an energy-efficient sensor network based on backscattering communication. Instead of encoding and transmitting data inside packages, the sensors backscatter a carrier signal to broadcast their sensing information. Each of them encodes its processed data into binary sequences and controls the reflection of carrier signals to transmit information at the physical layer. Since multiple devices are operating simultaneously, the receiver can observe the combination of all transmitted data and use it to aggregate information. Based on this mechanism, we implement an iterative algorithm to train a logistic regression model in which each sensor device optimizes its own parameters using binary feedback from a coordinator. Through extensive experiments with multiple public datasets, we concluded that our technique used less power than that of the traditional sensor network in which all sensors transmit their packaged data to a central server for analysis.

1.3.3 Proximity-based Secure Communication using Vocal Commands

The conceptual system visualized in Figure 1.3 connects a personal device (e.g. a smart-phone) to a public appliance supporting a voice user interface (e.g. a smart-screen) within a specific area using vocal commands. We not only extract the device class from the vocal command but also derive a key for the initialization of communication. After that, the pairing keys are continuously generated and updated using ambient audio. Our proposed architectures and protocols support both peer-to-peer and centralized scenarios. They can be integrated with the available infrastructures that are developed on standardized network protocols. We performed experiments in which users could control their verbal conversation to constrain the device selection range.



(a) GAIT-based secure pairing of on-body devices using human acceleration data during walking

(b) HEARTBEAT-based secure pairing of on-body devices using ballistocardiography data during resting postures

Figure 1.4. MANY-TO-MANY in Chapter 6: Two application scenarios of our pairing mechanisms, including: 1.4a on-body devices with ambulatory activities and 1.4b on-body devices with resting postures.

1.3.4 Continuous Secure Device Pairing

Device-to-device communication represents the MANY-TO-MANY relationship between smart devices. We proposed implicit mechanisms to establish a secure connection between wearable devices within the same body. We leverage behavioural (gait) and physiological (heart-beat) data extracted with sensors to implement secure device pairing. Figure 1.4 visualizes two application scenarios of our pairing schemes for: ambulatory activities and resting postures. Hence, our approach covers all scenarios of unobtrusively yet natural connecting multiple devices in a spontaneous manner. The communication keys are continuously updated and therefore data exchange is prevented whenever a device leaves the context (e.g. taking a smart-watch off the user's wrist).

First, as showed in Figure 1.4a, when devices are located on the same human body, we analyze body movement to form the pairing keys. Our keys are generated from the disparity between an instantaneous gait cycle and the mean cycle within a short time window. We apply error correcting codes in such a way that only data from the same human body can produce identical keys. We update these keys constantly to support automatic grouping and de-grouping devices. Then, we analyze the security threats of the proposed mechanism as well as related techniques. In addition, we implement a video-based attack with the support of a high-resolution camera and a manual tracking set-up.

Second, during resting postures (e.g. standing, sitting, and lying), we process heartbeats extracted from Ballistocardiography (BCG) data to generate the communication keys independently in each device. In this setting, we propose a network model to learn the keys. The architecture of our model is a combination of a Siamese networks and an auto-encoder. With this learning paradigm, we can constrain the network output to issue more similar keys for the same subject than those for different ones.

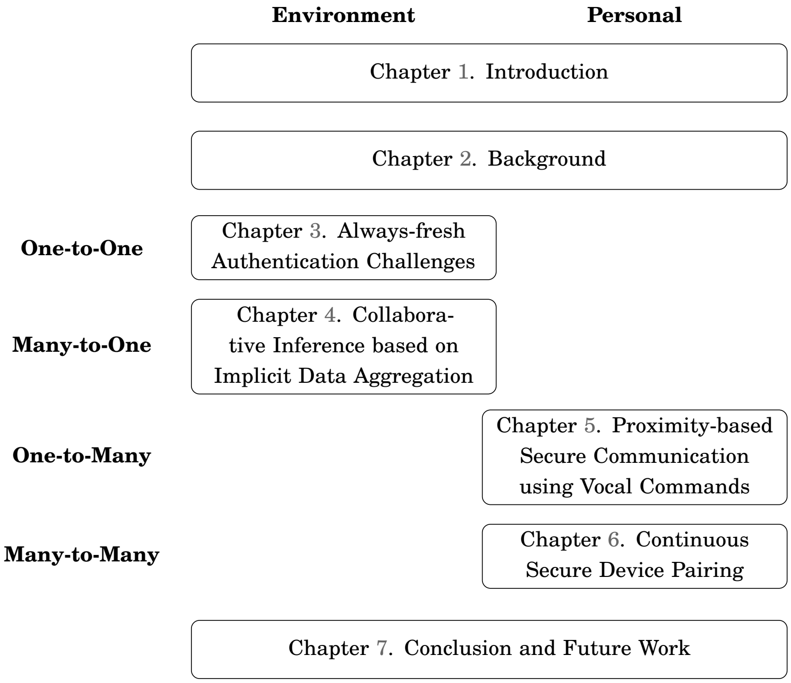

Environment
Personal
Chapter 1. Introduction
Chapter 2. Background
One-to-One
Chapter 3. Always-fresh Authentication Challenges
Many-to-One
Chapter 4. Collaborative Inference based on Implicit Data Aggregation
One-to-Many
Chapter 5. Proximity-based Secure Communication using Vocal Commands
Many-to-Many
Chapter 6. Continuous Secure Device Pairing
Chapter 7. Conclusion and Future Work

Figure 1.5. Outline of the dissertation

1.4 Dissertation Structure

Figure 1.5 illustrates the organization of this dissertation. We go through the background summary, the security mechanism in each device relationship, and conclude the dissertation with future work.

Chapter 2 presents background knowledge and fundamental techniques that are applied to develop our solutions. We define implicit information and its characteristics, as well as introducing specific sensors to collect them. The signal processing algorithms are categorized according to the amount of information extracted from data and its usefulness. We start with pre-processing techniques to reduce redundant information in the raw sensor data. Then, we briefly introduced feature extraction methods that are used to distil implicit information from the pre-processed data. After that, we discuss applications that are closely related to our proposed security mechanisms.

The next chapters discuss thoroughly each security mechanism with respect to the relationships of smart devices. Chapter 3 describes our graphical authentication mechanism that generates always-fresh passwords from visual information. We propose algorithms to filter, extract, and organize video frames into challenges that are personalized to a specific user. We introduce and evaluate different designs of the graphical passwords. Addressing user authentication, we cover the relationship in which one device interacts with its user. Our study then continues to investigate a scenario in which a team of smart devices collaborate towards a common goal: inference of the current situation. We propose a decentralized algorithm that allows devices to share their acquired information over the wireless channel without exposing the raw data. Using our algorithm, they can collaborate to train and use an inference model that is scattered across all entities of the network. This is the many-to-one relationship and is analyzed in Chapter 4. After being identified by personal devices, one may aim to establish a secure connection to new appliances for data exchange (e.g. connecting smartphone to a smart-screen for presentation). Chapter 5 introduces a convenient yet secure mechanism to select an appliance type using vocal commands. Then, the number of smart devices increase in the scenarios considered by Chapter 6. We investigate the secure pairing process within the many-to-many device relationship. In particular, we extract the secret keys that can be use in device-to-device communication from behavioural and physiological data. The secure information is automatically updated over time so that the proposed mechanism supports seamlessly grouping and separating devices.

Finally, we conclude the dissertation in Chapter 7 by summarizing our contributions in the important areas: user authentication, collaborative inference, device selection, and device pairing. Furthermore, we discuss potential extensions of this dissertation: the security mechanisms of backscatter devices with energy-harvesting components.

2. Background

This chapter presents the fundamental concepts and techniques to process data captured by the sensors of smart devices to reveal implicit information. We consider diverse modalities including: inertial data, radio frequency (RF) signals, audio, images, and videos. We focus on signal processing techniques that are required to develop our security mechanisms. Using these methods, we briefly explain how implicit information is uncovered from sensor data to serve our proposed security mechanisms.

We start the chapter by presenting representative sources of external data that can be collected by sensors. We focus on commercial off-the-shelf sensors that are equipped in smart-phones (e.g. cameras, microphones, and inertial measurement units) and smart-home appliances (e.g. hygrometer and thermometer). Then we introduce fundamental techniques to extract useful information from multidimensional sensor readings. These techniques are summarized in Figure 2.1, including: filtering, feature extraction, segmentation, clustering, and classification. Note that the output of one level is the input to the next level of extracting implicit information from sensor data.

Next, we introduce common security applications of smart devices. The first such application occurs in the relationship between a device and its owner: user authentication. We mainly focus on the use of visual information (i.e. images) since it is directly related to our proposed approach. Second, when there are more than one smart device, we focus on methods to securely initialize a communication channel between them, i.e. let them discover each other within a group and derive a pairing key for communication. The key can be used by one device for connecting to other devices and the shared key can be obtained by many appliances in a restricted space. Finally, we investigate scenarios in which a team of smart devices collaborate to achieve a common goal, such as training a machine learning model to monitor the environment. We design and implement a paradigm leveraging backscatter communication to reduce the power consumption of the network. Furthermore, our proposed paradigm implicitly hides the transmitted information inside burst sequences in the wireless signals.

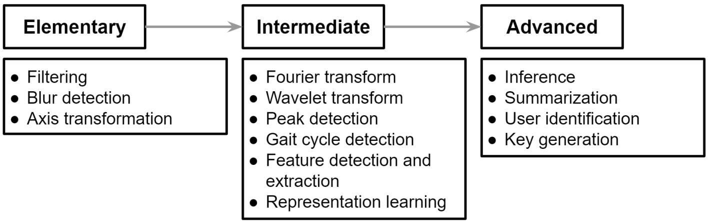


Figure 2.1. Techniques to extract implicit information from sensing data

2.1 Sources of implicit information

Smart devices are equipped with built-in sensors that can capture sensor data from various sources, including data about their users and the environment. In this dissertation, we focus on audio, video, inertial data, and radio frequency signals because they allow us to address issues of existing security mechanisms. The collected data needs to go through processing steps in order to reveal useful yet implicit information. For example, a microphone is the vital component of every phone. It can collect not only the human voice but also ambient sounds. Hence, it has become a rich source of sensor data. Audio data can provide a holistic view on the context surrounding smart devices. Due to the propagation of sound, proximate devices tend to record similar audio data. Based on this observation, audio-based secret keys were utilized to established secure communication between devices in a restricted area [129]. Most smart-phones are integrated with accelerometers and gyroscopes. These sensors collects motion data, including: acceleration, orientation, and angular velocity. The collected inertial data is processed to infer a user's activities [26] or gait [106]. Even subtle body movements caused by heartbeats can be detected through analyzing the recordings captured with these on-body sensors [143].

Although wearable sensors such as accelerometers and gyroscopes have demonstrated their usability, carrying them all the time is obtrusive for the users [26]. In addition, regular charging their batteries that power the sensors is cumbersome, especially for the elderly. In order to overcome these constraints, one rising trend is to employ device-free pervasive systems, for example, through analyzing RF signals in the environment. Due to multi-path propagation, these wireless signals are sensitive to changes in the environment (e.g. human moving) [126, 167]. Hence, wireless signals have been analyzed to deploy device-free activity recognition [136]. Sigg *et al.* [136] implemented a system to recognize human actions using Received Signal Strength Indicator of either ambient FM radio signals or signals generated by an active transmitter (e.g. a software-defined radio device). Another source of wireless signals is Wi-Fi, which have wide coverage, especially in indoor areas. Wang *et al.* [159] collected and processed channel state information to detect gait patterns.

A huge amount of implicit information comes from visual data such as images and videos which capture the world in two or even three-dimensional representation. Analyzing visual information reveals hidden knowledge for such applications as object classification and human activity recognition [57]. Recently, advances in electronics and optics have squeezed cameras to such a compact size that they are able to fit into a mobile phone or are able to be worn by users. These cameras capture the images of a user's daily activities and observations from the first-person point-of-view. The recorded videos have been analyzed to recognized objects or the wearer's actions [16]. These visual cues are then used in other applications such as motif discovery [169] or storyline reconstruction [92]. Egocentric videos can capture motion data that is used to identify the users [72].

The raw sensor data is diverse and contains a significant amount of redundant data. Hence, appropriate techniques such as preprocessing [54] and feature extraction are applied in order to extract implicit and useful information. We visualize some popular techniques to extract and analyze implicit information in Figure 2.1. These techniques can be classified into three categories, based on how many processing stages are required to discover implicit information from raw sensor data.

2.2 Preprocessing Sensor Data

Preprocessing techniques are applied to prepare the data for further analysis, such as feature extraction. These techniques clean the recorded sensor data through removing artifacts caused by a variety of sources (e.g. hardware imperfection, software errors, and electromagnetic interference) [137]. These techniques include data normalization, resampling, and other data transformation algorithms [54]. In general, they transform the sensor data into other formats which are more suitable for specific tasks.

The recorded data in its raw form is the combination of useful and redundant information (e.g. noise [137]). Filtering is applied to reduce the noise while retaining meaningful data for further analysis. For example, Bouten *et al.* [23] experimentally showed that accelerometer data that is sampled at 20 Hz is adequate for activity recognition. In the time domain, filtering is done via convolution $\otimes$ [162, 44]. One example is the moving average filter [137] which smooths the signal. Its convolution filter has the form of $\mathbf{h}[x] = [(\frac{1}{N})_{\times N}]$, where N is the size of the filter. If the elements of the filter are sampled from a Gaussian distribution, it becomes a Gaussian filter [137]. There are two widely-used types of convolution filters: finite impulse response (FIR) and infinite impulse response (IIR) [137]. Equation 2.2 formulates mathematically an FIR filter $\mathbf{h}$ applied on one-dimensional time series data $\mathbf{x}$:$\mathbf{y}[n] = \mathbf{h}[n] \otimes \mathbf{x}[n] = \sum_{k=0}^{N-1} \mathbf{h}[k]\mathbf{x}[n-k] = \sum_{k=0}^{N-1} \mathbf{a}_k \mathbf{x}[n-k]$, where N is the size of the filter $\mathbf{h}$ and $x[n-k] = 0$ if $n-k < 0$. On the other hand, an IIR filter has a infinite number of elements: $\mathbf{y}[n] = \mathbf{h}[n] \otimes \mathbf{x}[n] = \sum_{k=0}^{\infty} \mathbf{h}[k]\mathbf{x}[n-k]$.

The recorded data can be represented in the frequency domain using the z-transform [137]:$\mathbf{X}[z] = \sum_{n=0}^{\infty} \mathbf{x}[n] z^{-n}$ where $z = e^{\frac{i2\pi k}{N}}$ is a complex number. The discrete Fourier transform is a special case of the z-transform when we aim to extract N frequency components from a time series sequence of N values. The k^{th} component $\frac{2\pi k}{N}$, where $0 \leq k < N$, is represented as: $\mathbf{X}[Z_k] = \sum_{n=0}^{N-1} \mathbf{x}[n] \left[e^{i2\pi k}\right]^{-n} = \sum_{n=0}^{N-1} \mathbf{x}[n] e^{\frac{-i2\pi kn}{N}} = \sum_{n=0}^{N-1} \mathbf{x}[n] \cos^{\frac{2\pi kn}{N}} - i\mathbf{x}[n] \sin^{\frac{2\pi kn}{N}}$ Similarly to filters in the time domain, we can formulate filters in the frequency domain as: $\mathbf{Y}[z] = \mathbf{H}[z] * \mathbf{X}[z]$ where $\mathbf{Y}[z]$ and $\mathbf{H}[z]$ are the z-transform of $\mathbf{y}[n]$ and $\mathbf{h}[n]$, respectively. According to the convolution theorem [162, 44], the convolution operator in the time domain is equivalent to the point-wise multiplication in the frequency domain.

In some cases, we design algorithms to remove data whose content is not suitable for our applications. For example, to distinguish between useful and redundant photo frames (i.e. 2D data) in a video stream, we can utilize a *preprocessing* algorithm based on blur detection. We prefer techniques that have low computational complexity, are adaptable to a wide range of visual content, and are independent of reference. The blurriness in first-person-view videos is mainly caused by the relative motion between the camera and the captured scene. A candidate algorithm to filter blurred frames is the no-reference perceptual blur metric proposed by Marziliano *et al* [99]. Their algorithm does not require knowledge of the original image, the content, nor the blurring cause. They measure the blurring effect on the vertical edges detected in the photos. Beyond blurriness, content-based filtering is implemented based on more complicated techniques that extract characteristics or features of images.

2.3 Extracting Implicit Information from Sensor Data

To extract characteristics from preprocessed data, time-series data is often analyzed in separated segments (e.g. a chunk of temporal data points), called time windows, with some percentage of overlapping [26]. These characteristics are represented in a wide range of features. They include statistical values in the time domain such as mean, standard deviation, or kurtosis, which are simple to compute but achieve high accuracy in activity recognition [26]. They can be combined with frequency domain features to improve classification results, such as energy in frequency bands and mel-frequency cepstral coefficients [54]. There are certain features that are specified according to applications. They are defined using expert knowledge on certain domains. For example, gait cycles can be detected with accelerometers as biometrics for identification [2]. One gait cycle includes two steps, each of which is located by minimum values in the preprocessed acceleration data [158]. Another example is the fiducial peak point of ballistocardiography data [143, 133]. In both cases, the core technique is to detect peaks (local minima or maxima) from segmented sensor data [145].

While the above kind of data is comprised of one or more sequence of values (i.e in one dimension), an image can be represented as a matrix (i.e. two-dimensional data). Hence, these two dimensions have a spatial relationship between each other. Human pays attention to the spatial configuration or the scene inside an image to map from visual representations to meaning [66]. They can extract information from an image to identify its semantic properties, such as an image of a street with people walking surrounded by buildings. Oliva and Torralba [112] proposed a computational model to recognize scene categories. They used spectral and coarsely localized information to estimate a set of perceptual dimensions (naturalness, openness, roughness, expansion, ruggedness) representing the dominant spatial structure of an image. Their proposed visual representation, called the GIST descriptor [112], is one of the most well-known global descriptors. On the other hand, there are local descriptors that characterize object appearance and shape in each spatial regions of an image. For example, Histograms of Oriented Gradients (HoG) [38] captures local intensity gradient or edge directions as histograms at each cell of an image. To achieve higher accuracy in image matching, Bosch *et al.* [20] proposed to accumulate HoG descriptors in increasingly finer spatial grids to form a hierarchical histogram-based representation.

Features can be *learned* from the data, for example using the penultimate layer of a neural network [15]. This paradigm has achieved significant advances in various data-intensive problems such as those in computer vision and speech recognition [60]. Many network architectures have been proposed and continuously adapted to a wide range of applications. Deep convolutional neural networks have showed their superior performance in the classification of images [83] and videos [80]. They stacks layers of different types to transform the input data into an output representation. After being trained in a data-driven manner, a network learns filters (i.e. weight vectors) that can detect specific patterns at some spatial position in the input. Another architecture is auto-encoders [156] which are trained to output a representation as close to the original data as possible. Its latent representation at the bottleneck layer is employed as learned features which has a lower number of dimensions than that of the input data. A auto-encoder network improved the accuracy of different accelerometer-based activity recognition tasks [117]. While the two aforementioned architectures tend to model spatial relations, recurrent neural networks [115] aim to capture the dependencies within sequential data. Long short-term memory [70], one of their variants, was applied to learn the representation of biometrics data from electrocardiography [123] and keystroke behaviour [148].

2.4 Applications of Implicit Information

In this section, we introduce representative applications of implicit information, which are implemented on four relationships of users and smart devices. These applications process data collected by embedded sensors to facilitate authentication (in a ONE-TO-ONE relationship, see Chapter 3), data transmission (in a MANY-TO-ONE relationship, see Chapter 4), connection initialization (in a ONE-TO-MANY relationship, see Chapter 5), and peer-to-peer communication (in a MANY-TO-MANY relationship, see Chapter 6). These mechanisms enhance usability while maintaining the security requirements of user-to-device and device-to-device communication.

Autobiographical authentication is developed on the intersection of users' implicit memories and information recorded within personal devices. Das *et al.* [39] proved its effectiveness through two online questionnaires and one field study on mobile phone usage data. Hang *et al.* [64] proposed to generate the questions for fallback authentication from what users have done with their smartphones. The information directly comes from the data inside the smartphone. For example, this system asked the users which photo had been taken or whom they had communicated with. Implicit authentication [76] authenticates users based on data generated from the actions they are carrying out. Human behavioural data captured by sensors is a common modality to implement this authentication paradigm [8]. A similar paradigm is continuous authentication [114], which utilizes sensor data to implicitly identify users and automatically de-authenticate them when they stop using the device. This is a countermeasure to the threat where users either forget to log-out or attackers use the device while users are temporarily absent. For example, Mare *et al.* [97] collect movement data with on-wrist inertial sensors. They then analyzed the correlation between wrist movements and in-device events to continuously identify the legitimate user. Both paradigms require a data-driven training procedure to distinguish legitimate users and adversaries, which is susceptible to evasion attacks [95].

Nowadays, the communication capability of smart devices allows them to form an ad-hoc network with each other to share data and collaborate [63]. The secure connection mostly initiates using a PIN code entered by users [61], which is obtrusive. Using implicit information extracted from physiological and environmental data, we can implement unobtrusive device-pairing protocols that are not only more convenient but also more applicable to limited user interfaces (e.g. e-textiles without a screen). Smart devices within a range of each other (i.e. in proximity) are able to collect similar sensor data characterizing *device context*. Analyzing contextual information can help to detect the co-presence of corresponding devices [153]. The correlation between proximity and context similarity makes the sensor data suitable for unattended proximity-based device pairing. In particular, contextual fingerprints, usually as binary sequences, can be generated separately in each device using such modalities as radio-frequency signals [154, 82] and audio [129]. Then, pairing keys can be derived from these sequences through an error-correcting code whose parameters are configured according to the number of mismatching bits. An identical process can be applied to devices carried by users such as smart-phones and smart-watches. In this setting, human gait is the source to first generate fingerprints, and then pairing keys [166, 127].

Backscattering is the paradigm to implicitly transmit data over a carrier signal without actively generating radio signals [144]. Recently, ambience backscatter has allowed devices in close proximity to communicate by backscattering ambient RF signals (e.g. from a TV tower or cellular network) [91] or enable Orthogonal frequency-division multiple access (OFDMA) in Wi-Fi backscatter [172]. This low-power and low-cost mechanism has been constantly improved to facilitate long-range communication. Talla *et al.* [151] integrated a frequency synthesizer to enable chirp spread spectrum modulation in their backscatter devices. Varshney *et al.* [155] re-designed the computational Radio-frequency identification (RFID) architecture to achieve a long communication range in the 868MHz and 2.4GHz bands. Their design used two oscillators to generate two frequencies for Frequency-Shift Keying on the carrier signal. Recently, in order to implement concurrent transmission of multiple devices, Hessar *et al.* [69] utilized distributed chirp-spread-spectrum coding to organize the backscattered signals. Although backscattering has become a solution for communication with low power consumption, all current backscattering systems focus only on data transmission of individual devices. We extend the state-of-the-art by performing data aggregation over the wireless communication channel. Based on that, we can realize partially the computation of a machine learning model. We hide the transmitted data in backscattered signals (i.e. a *covert* channel to aggregate sensor data). Moreover, our design simplified the prototype through eliminating oscillators to further reduce the power consumption.

2.5 Summary

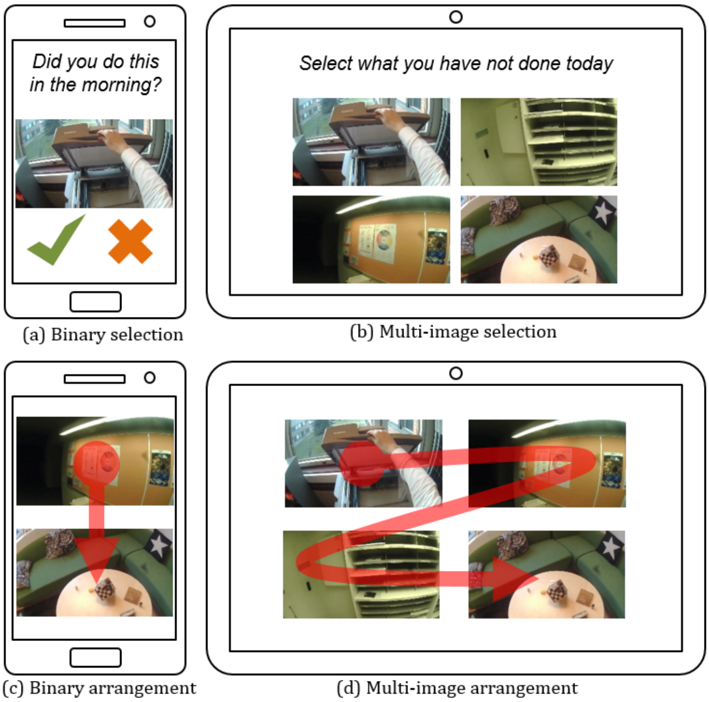
After investigating the existing applications of implicit information, especially in user authentication, device pairing, and secure data transmission, we discovered that implicit information could be further utilized to implement novel security mechanisms or enhance the current ones. In user authentication, we proposed to generate always-fresh passwords from first-person-view videos. The visual data captured by an on-body camera had not been used to authenticate users. Thanks to the ever-changing of first-person-view videos, our image-based passwords could address the issues of shoulder-surfing and smudge-based attacks. In data transmission, especially with such resource-constrained devices as sensors, backscatter communication provided an elegant solution to the power consumption of transmitting data. However, existing work mainly focused on encoding data into network packages to mimic conventional protocols. Based on that, training a machine learning model relied on a central node which aggregated data from sensors before performing the training algorithm. We simplified the data transmission by encoding transmit values into burst sequences so that we could leverage the signal interference as a means to implicitly aggregate data. We derived a variant of the stochastic gradient descent algorithm that could work with vertically-partitioned data and allow sensors to retain their model parameters at their sites. In addition, the superimposition of burst sequences possessed an interesting attribute that hid transmit values. In initiating a communication channel between two devices that have not connected to each other before, we proposed to utilize vocal commands, which were intuitive from the users' point-of-view. We extracted audio fingerprints and used their similarity to establish a secure connection. We then experimented with human and hardware eavesdroppers to evaluate the users' awareness with eavesdroppers. In on-body device pairing, we proposed a novel gait-based pairing protocol that utilized the difference between a instantaneous gait cycle and a mean gait cycle. We implemented and evaluated this protocol as a means to realize continuous on-body device pairing. Later we presented an approach to extend the protocol to resting positions through using heart-beat data captured with accelerometers. We introduced the first-ever attempt that formulate the key extraction problem as a learning model, which is a Siamese auto-encoder. All of these security mechanisms were based on implicit information extracted from sensor data and they addressed existing problems in current mechanisms. In the next chapters, we would introduce in details our proposed mechanisms.

3. Always-fresh Authentication Challenges

A device performs the authentication process to evaluate the truth of a user's identity in order to decide whether it should allow the user to access its data and services or not [35]. To be authenticated, the user is required to recall the secret information considered as passwords [67]. The user authentication process is an example of the ONE-TO-ONE relationship between the device and its owner. In this chapter, we generate transient authentication challenges using first-person-view videos capturing a user's activities and observations.

3.1 User Authentication

We perform the procedure of authentication everyday. For example, we open doors with our physical keys, keycards, or numeric codes. This action is to authenticate ourself with the locations that we intend to enter. Digital devices such as smart-phones and computers offer multiple methods for user authentication, including numeric codes, graphical patterns, and biometrics (e.g. fingerprints). One common authentication method on smart devices is the use of Personal Identification Numbers (PINs) [32], which are sequences of several digits (usually four) and are chosen by users. These PINs, as well as their longer version (e.g. passwords containing alphanumeric and special characters), have widely-known drawbacks, including: selecting weak passwords, using one password for several devices, storing them in unsecured ways (e.g. writing on a piece of paper), using the same password for a long time, and being vulnerable to shoulder-surfing [67]. Graphical passwords have been introduced around 1999 in order to improve memorability and usability while consolidating password strength against guessing attacks [17]. They are nevertheless still vulnerable to such attacks as: shoulder-surfing and smudge attacks [146]. Authentication based on biometrics data [102] is static, limited (e.g. number of fingers), possible to be stolen (e.g. fingerprints from surface), and non-resilient (i.e. if the biometric information is compromised, the user can no longer use it) [88]. Behavioural biometrics [8] such as touching gesture, gait, and keystroke dynamics rely on a machine learning system training on the users' behavioural database and are


Did you do this in the morning?
Select what you have not done today

(a) Binary selection

(b) Multi-image selection

(c) Binary arrangement

(d) Multi-image arrangement

Figure 3.1. Potential layouts of our authentication challenges: (a) and (b) are *Image-selection* while (c) and (d) are *Image-arrangement*. Both schemes can be adapted to different screen sizes by varying the number of images.

vulnerable to evasion attacks [95].

In this chapter, we propose to generate transient authentication challenges from images capturing a user's perspective. Our graphical passwords utilize implicit information on the chronological order of the user's activities and observations. These authentication challenges are always-fresh and personalized for each user. They release users from changing passwords regularly while still offering a memorable representation of log-in information. There are two approaches that are the most related to our proposed mechanism. The first one is called autobiographical authentication [39] [64], which utilizes phone usage events (e.g. sending messages and taking pictures). This approach, however, may not obtain enough information to generate authentication questions for users with little device usage [64]. The second related approach produces passwords from images selected by users [43] [51] [42] or from icons of the installed applications [147]. The images used in these studies came from fixed collections while those in our approach originate from ever-changing stream of personalized videos recorded with wearable cameras.

3.2 Image-based Password Design

We suggest two authentication schemes, as shown in Figure 3.1: *Image-selection* and *Image-arrangement*. These are formed from images in various lengths of timelines so that they can be adapted to different scenarios, from instant log-in to fallback authentication. In addition, we can strengthen the approach by using a sequence of consecutive challenges. In *Image-selection*, the challenge is to identify images which belong to a specific time window. With larger screens, it is also possible to ease the login experience by requiring only a certain percentage of correct choices (e.g. 90%). In *Image-arrangement*, we ask the users to arrange multiple images in the correct chronological order. In all cases, our implementation alters the images to form a new challenge with each wrong trial, so that learning of the right order is difficult in consecutive login attempts. From our experiments (cf. Section 3.4.1), *Image-selection* challenges require only a short completion time. *Image- arrangement*, however, has a higher mental load and therefore requires a longer time to solve the challenges. Hence, the former can be used as an instant login mechanism while the latter is appropriate for fallback authentication.

Image-arrangement

Image-arrangement passwords facilitate the challenge on the chronological order of images. After segmenting and clustering, we group video frames in such a way that images in the same segment belong to the same cluster. Then, photos which are in the same cluster but appear in different segments (i.e. repetitive scenes) are removed from the candidate set. We do that because the user may be confused if passwords contains a image pair that describes the same scene but is interleaved by images belonging to a different scene. Image-arrangement authentication challenges aim to leverage the usability of pattern passwords on the Android platform.

Image-selection

Image-selection passwords utilize the time window when the user executes a certain activity or observes a particular scene. For instance, the authentication question *"What have you done today?"* or *"What have you not done after 11:00am?"* is shown above the images. Hence, the clustering algorithm is supplied with videos both within and exceeding the respective time window. We discard scenes that appear both inside and outside the time window. Repetitive scenes happening within the time window are not removed because they do not cause any confusion to the user.

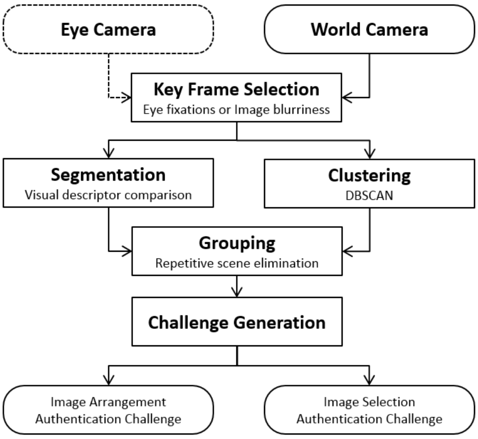

Eye Camera
World Camera
Key Frame Selection
Eye fixations or Image blurriness
Segmentation
Visual descriptor comparison
Clustering
DBSCAN
Grouping
Repetitive scene elimination
Challenge Generation
Image Arrangement Authentication Challenge
Image Selection Authentication Challenge

Figure 3.2. The process of generating authentication challenges from first-person-view videos

3.3 Generating Passwords from First-person-view Videos

In this section, we discuss the technical details behind our image-based password generation. First, we present our algorithm to select candidate video frames which are later used to create image-based authentication challenges. Second, the selected frames are segmented into different scenes according to the user's observations and activities. Next, we introduce a clustering-based technique to remove repetitive scenes which do not support the user's recall process. Finally, we form the authentication challenges from the remaining images, which are clear and memorable. Figure 3.2 summarizes our procedure that transforms first-person-view videos into authentication challenges.

3.3.1 Key Frame Selection

The recorded videos contain a huge number of images or frames with greatly varying quality and content. A significant part of them are blurred due to body and head movements. Hence, we retain only key frames with memorable content. To do that, we calculate the blurriness of each frame using the method proposed by Crete *et al.* [36]. They apply a low-pass filter on a grayscale image, then compare the neighbouring pixel variation between the original and filtered photo. If the difference between the two versions is high, the original one is sharp; otherwise, it is blurred. Let F be an $m \times n$ grayscale image. We apply a vertical and horizontal low-pass filter to obtain the blurred image B:

$$\begin{aligned} B_v &= h_v * F \\ B_h &= h_h * F \end{aligned} \tag{3.1}$$

$$\text{where: } h_v = \frac{1}{9} \times \begin{bmatrix} 111111111 \end{bmatrix} \text{ and } h_h = h_v^\top$$

Then the variation of neighbouring pixels is calculated vertically and horizontally for both images:

$$\begin{aligned}
\Delta_v^F(i,j) &= abs\left(F(i,j) - F(i-1,j)\right) \\
\Delta_h^F(i,j) &= abs\left(F(i,j) - F(i,j-1)\right) \\
\Delta_v^B(i,j) &= abs\left(B_v(i,j) - B_v(i-1,j)\right) \\
\Delta_h^B(i,j) &= abs\left(B_h(i,j) - B_h(i,j-1)\right) \\
&\text{where } i \in [1, m-1] \text{ and } j \in [0, n-1]
\end{aligned} \tag{3.2}$$

Finally, the blurriness of F n_F is:

$$\begin{aligned}
n_F &= max\left(b_v^F, b_h^F\right) \\
b_v^F &= \frac{\sum_{i,j=1}^{m-1,n-1} \Delta_v^F(i,j) - \sum_{i,j=1}^{m-1,n-1} \Delta_v(i,j))}{\sum_{i,j=1}^{m-1,n-1} \Delta_v^F(i,j)} \\
b_h^F &= \frac{\sum_{i,j=1}^{m-1,n-1} \Delta_h^F(i,j) - \sum_{i,j=1}^{m-1,n-1} \Delta_h(i,j))}{\sum_{i,j=1}^{m-1,n-1} \Delta_h^F(i,j)} \\
\Delta_v &= max\left(0, \Delta_v^F(i,j) - \Delta_v^B(i,j)\right. \\
\Delta_h &= max\left(0, \Delta_h^F(i,j) - \Delta_h^B(i,j)\right.
\end{aligned} \tag{3.3}$$

Assume the video consists of k frames v_i whose blurriness is n_i, we remove v_i if $n_i \geqslant median(\{n_i | i \in \mathbb{N}, 1 \leqslant i \leqslant k\})$. The remaining photos tend to describe *memorable* moments that have captured the user's own attention. Alternatively, if the headset is equipped with an eye-tracking camera, we can detect fixation moments to choose important frames. Here, however, we focus on the blurriness-based method because it is more versatile.

3.3.2 Image Features

We aim to extract both global and local characteristics of each video frame. Each feature vector is able to describe the appearance of the whole scene as well as that of the local sub-regions and individual objects. To fulfil this requirement, we combine the Census Transform Histogram (CENTRIST) [164] and the Pyramid of Histograms of Orientation Gradients (PHOG) [20] to form the representation of our first-person-view images. We perform Principle Component Analysis to reduce the number of dimensions. We achieved the best result when each feature vector contains $n = 100$ variables. The 100-dimensional vectors are then used to split the video frames into temporal sequences and cluster images that have similar content.

CENTRIST: [164] generates a holistic representation of a scene by capturing structural characteristics and rough geometry. Census transform compares the intensity value of each pixel with those of its neighbourhood. If the value is less than for one of its neighbours, the corresponding location is assigned 0, otherwise 1. Then, eight bits in the neighbourhood are concatenated and converted to a base-10 number called the census transform value of the central pixel. After all pixels of the image are evaluated, the CENTRIST descriptor is constructed from a histogram of census transform values. A spatial pyramid scheme is applied to obtain a robust global representation. An image is split into (sliding) blocks in different levels. In each block, the CENTRIST descriptor is extracted independently. Finally, the descriptors (i.e. histograms) of all blocks at all levels are aggregated to form the representation of the image.

PHOG: To complement the above global descriptors, we compute the PHOG feature [20]. With this descriptor, Bosch *et al.* [20] aimed to represent an image through its local shape and the spatial relations of the shape. They divide an image into rectangular regions at several resolutions and calculate the distribution (histogram) of the edge orientations in each region. Each bin of the histogram contains the number of edges whose orientations belong to a specified angular range. Then, the concatenation of histograms from all regions becomes the descriptor of the image.

3.3.3 Image Memorability Classification

We evaluate the effectiveness of our blurriness-based image removal technique by selecting 192 confused images and 234 memorable images from the output of our algorithm. These images were then fed to an online evaluation service [1], which was developed from crowdsourced photos, to assess their memorability [74]. The mean memorability score of the former was 0.51 while that of the latter was 0.76 (1 is the most memorable).

This result implies that a classification algorithm can also be developed to discriminate images into two categories. To test this hypothesis, we trained a Support Vector Machine classifier using LibSVM [2] with CENTRIST [164] and PHOG [20] features. We randomly split the above image sets into training and testing subsets. The classifier parameters were optimized with cross-validation. Then, the optimal model was applied on the testing subsets. We repeated this process ten times. The average correct classification rate and F-measure were 92% and 0.93, respectively. This shows that the technique is useful for filtering out unmemorable images.

3.3.4 Segmentation

After the first phase of extracting key frames, the $k' \leqslant k$ selected images are then segmented into scenes, based on the similarity of visual features (cf. Section 3.3.2). If the difference between two frames is below a certain threshold τ, they belong to the same segment. We define τ as the median value of the Eu-

[1] http://memorability.csail.mit.edu/demo.html [Accessed: June 18, 2019]

[2] https://www.csie.ntu.edu.tw/ cjlin/libsvm/ [Accessed: June 18, 2019]

clidean distance between two consecutive (representative) frames in the feature space:

$$median(\{d(f(v_i), f(v_{i+1}))|i \in \mathbb{N}, 1 \leqslant i \leqslant (k'-1)\}), \tag{3.4}$$

where the function f extracts the feature vector from video frames and d is the Euclidean distance function. We choose τ instead of the pairwise distances between every pair of frames for its lower computational complexity ($\mathcal{O}(n)$ instead of $\mathcal{O}(n^2)$). The algorithm is summarized in Algorithm 1. We later refine the segmentation output through discarding short segments resulting from quick head movements.

Algorithm 1: Segmentation

```
Data: An array $F$ of $k'$ images of size $m \times n$
Result: An array $S$ of segment indices
for $i \in [1, k'-1]$ do
    $f_i = PCA((CENTRIST(F_i), PHOG(F_i)))$;
    $f_{i+1} = PCA((CENTRIST(F_{i+1}), PHOG(F_{i+1})))$;
    $d_i = calculate\_distance(f_i, f_{i+1})$;
end
$\tau = median(\{d(i)|i \in [1, k'-1]\})$;
for $i \in [2, k'-1]$ do
    if $d_i < \tau$ then
        $S_i = S_{i-1}$;
    else
        $S_i = S_{i-1} + 1$;
    end
end
```

3.3.5 Clustering

Next, we cluster similar segments into groups of the same scenes. The number of clusters is not determined beforehand because there is no knowledge on which activities are repetitive. We therefore use *Density-based spatial clustering of applications with noise* (DBSCAN) [50], which groups together points that are in a neighbourhood. The algorithm requires a distance threshold to group similar images and the minimum number of images in each cluster. For the distance threshold, we again use the difference of consecutive frames τ in Equation 3.4. The second parameter allows us to discard *noisy* images, which are *non-informative*.

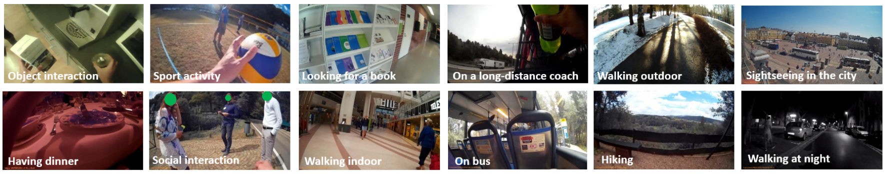


Figure 3.3. Sample images extracted from our egocentric videos. They vary in activities (e.g. holding objects, playing sport, traveling on public transport, and social interaction), locations (indoor and outdoor), weather conditions (e.g. winter and summer), and the light intensity (e.g. sunny day, having dinner at night, and dark scene).

3.4 Case Studies

In this section, we implemented our proposed method and deployed the system to realistic scenarios in order to answer the research question: *How to authenticate with always-fresh passwords leveraging implicit information?* We performed several case studies: (1) experimenting the image-arrangement authentication mechanism which performed during two consecutive days and on an object-interaction setting in an office unfamiliar to the users and (2) experimenting the image-selection authentication mechanism which covered an extreme case of a subject traversing first-time visited locations continuously over a period of three weeks and a basic study spanning two consecutive days. The results achieved indicate that the log-in time required is comparable with other graphical password schemes such as PassApp [147] (7.27 seconds), Passfaces [51] (18.25 seconds), and Déjà Vu [43] (27 - 32 seconds), even though the passwords utilized in our case are, in contrast, not static.

For video recording, we utilized two on-body cameras (cf. Figure 3.4). The Transcend DrivePro$^{\text{TM}}$ Body 10 device (cf. Figure 3.4a) was mounted on the chest while the Pupil Labs headset [81] could be worn near the eyes (cf. Figure 3.4b). Both feature high-definition resolution and a wide-angle lens. A collection of images extracted during these studies is shown in Figure 3.3. Those photos represent diverse settings due to the subjects' activities, body movements, locations, weather, light intensity, and scene appearance. The wearer then solved the passwords on personal devices while the number of attempts and the entry time were collected. We assumed that all involved components (e.g. wearable cameras, personal devices, and processing servers) were securely connected.

(a) Chest-mounted camera

(b) Head-mounted camera

Figure 3.4. The devices used to collect video data: Transcend DrivePro$^{\text{TM}}$ Body 10 camera (a) and Pupil Labs headset (b)

3.4.1 Image-arrangement Study

Experiment on multiple indoor locations

Participants : We recruited five subjects ($\mu_{age} = 30$) and two of them were female. Two of them wore glasses. Their height was from 1.60m to 1.85m ($\mu_{height} = 1.7$m).

Materials : We used a Transcend DrivePro$^{\text{TM}}$ Body 10 device to record first-person-view videos (see Figure 3.4a).

Design : The camera was attached to the subjects' clothes using its clip. We deployed a web application to capture the log-in time and the number of attempts.

Procedure : The camera wearer continuously recorded videos when possible, considering technical, legal, and social regulations. The videos contained data related to the subjects' daily activities and observations.

We conducted two experiments to investigate the performance of the *image arrangement* authentication challenge. For the first study, we recruited five participants (two female). To record the videos, a Transcend DrivePro$^{\text{TM}}$ Body 10 device was worn by each subject on two consecutive days (cf. Figure 3.4a). The camera has a rotational clip to attach easily on clothes or backpack straps. The camera wearer continuously recorded videos when possible, considering technical, legal, and social regulations. The videos contained data related to the subjects' daily activities and observations.

We developed a web application that supports slide-and-swipe gestures with Javascript and HTML. Each challenge, i.e. graphical password, consists of four video frames which are selected from distinguishable temporal segments. In each challenge, the user needs to answer $n = 4$ image-based authentication challenges consecutively. The number of attempts and the entry time are collected for each password. The system changes the challenge (i.e. new images) with every wrong arrangement. We observed a mean entry time of 9.79 seconds and a mean number of attempts of 1.87.

Experiment on object-interacting activities in an indoor scenario

Participants : We recruited seven participants (four females, two with glasses). Their mean age was $\mu_{age} = 32$. Their height was from 1.60m to 1.85m ($\mu_{height} = 1.7$m).

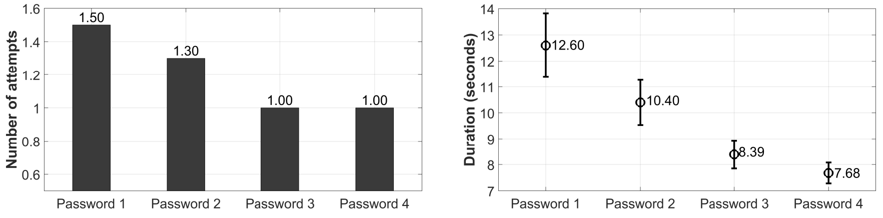
Materials : Each subject wore the Pupil Labs headset [81] (cf. Figure 3.4b). The room in this experiment contained basic furniture and office equipment.

Design : The camera was attached to a headset frame. The activities that the subjects could performed included: reading a book, working on a laptop, writing on paper, writing on a board, viewing a poster, talking to a person, using a smart-phone, unboxing an item, playing a boardgame, and using a paper-cutter. The objects were put on the desks (laptop, paper, boardgame, and paper-cutter), hung on the wall (poster and board), or in the subject's pocket (smart-phone). We utilized the web application to capture the log-in time and the number of attempts.

Procedure : Each subject wore the Pupil Labs headset and performed the activities in any order, before solving authentication challenges generated from the recorded first-person-view videos.

In order to understand the performance of the system in indoor environments with similar background and limited video footage, we prepared an extreme case in which interaction with several objects in a single office room is investigated. The setting was challenging to our system because all activities occurred at the same location, so that object appearance and interaction are the only visual cues that assist our video analysis algorithm. The room featured basic furniture and office equipment and was unknown to all seven participants (four females, two with glasses). Each subject wore the Pupil Labs headset [81] (cf. Figure 3.4b) and performed activities in an arbitrary order, before solving passwords generated from the recorded egocentric videos. The activities included: reading a book, working on a laptop, writing on paper, writing on a board, viewing a poster, talking to a person, using a smart-phone, unboxing an item, playing a boardgame, and using a paper-cutter. The objects were put on the desks (laptop, paper, boardgame, and paper-cutter), hung on the wall (poster and board), or in the subject's pocket (smart-phone).

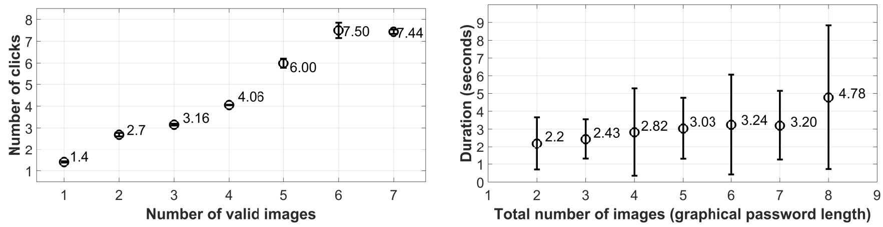
Figure 3.5 shows the number of attempts and entry time. The declining trend in both Figure 3.5a and Figure 3.5b indicates that the wearer is able to learn about the occurrence order due to the limited variation possible in this setting. Participants spent on average 12.62 seconds to answer the first authentication



(a) Average number of attempts to solve each password

(b) Average entry time spent on solving each graphical password

Figure 3.5. User effort in terms of the number of attempts and the entry time in the object-interaction scenario with *image-arrangement* authentication challenges. The prototype supports slide-and-swipe interaction to decrease the entry time.



(a) Number of clicks to solve the passwords

(b) Time duration spent on solving the passwords

Figure 3.6. User effort in terms of the number of clicks and the entry time (duration) in the daily condition study with *image-selection* authentication challenges

challenge. When all four passwords were taken into account, each subject spent 9.77 seconds with 1.21 attempts to solve a password.

3.4.2 Image-selection Study

To evaluate our system in the *Image-selection* mechanism, we conducted two experiments in which the subjects wore the cameras for several days. For this, another web-based prototype was implemented that displays from two to eight photos. Both the total number of images n and the quantity of valid photos $1 \leq k \leq (n-1)$ were varied randomly. The challenge consisted of the images describing events on the previous day but not on the current day. The subjects were able to select and unselect an image multiple times until achieving the correct configuration. We recorded the number of clicks. The entry time was calculated from when the password fully appears until when the proper selection was reached.

Experiment on activities in outdoor scenarios

Participant : We recruited one male subject (30 years old, 1.65m tall, no glasses).

Materials : The subject wore Transcend DrivePro$^{\mathrm{TM}}$ Body 10 camera on the chest.

Design : The experiment aimed to capture scenes of home, workplace, and navigation (both on foot and on vehicle). The subject was authenticated with image-selection challenges generated dynamically. We recorded the number of clicks (on the images) and the log-in time.

Procedure : The camera continuously captured videos in consideration of technical, legal, and social regulations. The videos contained scenes related to the subject' daily activities and observations. The subject randomly performed the authentication with image-selection passwords and the number of clicks and the log-in time were recorded.

In the first experiment, the Transcend DrivePro$^{\mathrm{TM}}$ Body 10 camera was worn by a graduate student over a period of three weeks (cf. Figure 3.4a). Mainly, the videos contain scenes of home, workplace, and navigation (both on foot and on public transport). In particular, the subject also captured videos of a trip to Stockholm (Sweden), Wadern (Germany), and Brussels (Belgium), where the student visited for the first time. The subject performed the log-in actions multiple times during the experiment with dynamically-generated graphical passwords. Depending on password length and the number of valid images (n, k), the time to success varied (cf. Figure 3.6). Apparently, the number of clicks and the entry time increase when the password becomes complex. In the experimental results, 50% of the two-image passwords were solved in about 1 second or less with a single click. Even though passwords with 8 images are more challenging, the subject solved them on average in 4.78 seconds (cf. Figure 3.6a), with 7.44 clicks (cf. Figure 3.6b).

Table 3.1. Average entry time in seconds and number of clicks to achieve partial and total correctness in the *image-selection* scheme. The numbers in brackets are standard deviation.

Correctness	$\geqslant$ **50%**	$\geqslant$ **75%**	**100%**
Entry time (s)	3.77 (2.88)	4.85 (3.08)	5.66 (3.33)
# clicks	2.09 (0.98)	3.14 (1.68)	3.68 (2.08)

Experiment on activities in outdoor scenarios

Participant : We recruited five subjects (μ_{age} = 30) and two of them were female. Two of them wore glasses. Their height was from 1.60m to 1.85m (μ_{height} = 1.7m).

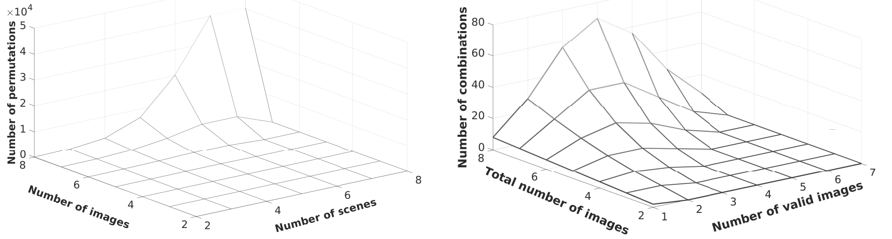
Materials : The subjects wore Transcend DrivePro^{TM} Body 10 camera on their chests.

Design : The experiment aimed to capture scenes of home, workplace, and navigation (both on foot and on vehicle). The subject was authenticated with image-selection challenges generated dynamically. We recorded the number of clicks (on the images) and the log-in time.

Procedure : The camera continuously captured videos in consideration of technical, legal, and social regulations. The videos contained scenes related to the subjects' daily activities and observations. The subject randomly performed the authentication with image-selection passwords and the number of clicks and the log-in time were recorded.

Furthermore, we investigated another case involving five subjects (two females, two with glasses). Each of them wore the camera over two consecutive days. Then, they tried to solve the *image-selection* authentication challenges. For a password length of $2-8$ images with $1-7$ valid images, the mean entry time was 4.67 seconds (standard deviation 3.17) and the mean number of clicks was 2.92 (standard deviation 1.89). For challenges with a bigger number of images, it was more likely that one of the images was selected incorrectly. As a trade-off between usability and security, we therefore also compared the performance until a specific fraction of images was chosen correctly (see Table 3.1 where only 8-image passwords have been considered). When the users answered the challenges, we quantified the correctness of the current solution based on the number and position of the chosen valid photos. The table shows the user effort in terms of the average entry time (seconds) and number of clicks to achieve answers 50%, 75%, and 100% of similarity compared with the correct image-based authentication challenges.

Based on these results, our system can be configured to balance security and



(a) Number of permutations in an image arrangement password

(b) Number of combinations in an image selection password

Figure 3.7. Number of possible answers (permutation or combination) of our authentication mechanism

usability according to individual demand or preference. This approach reduces the effort in terms of entry time and number of clicks.

3.5 Security Analysis

In this section, we analyze four attacks where adversaries try to log in to a user's personal device without obtaining the wearable camera. The case of accessing both devices is not in the scope of this study, though it can be mitigated by using head movements detected by inertial sensors [89] or video analysis [72].

3.5.1 Brute Force Attacks

The bruteforce attack assumes that attackers obtain the personal device but do not have other information such as the user's routine. Hence, the adversaries have to try all possible combinations of images in each authentication challenges. In Figure 3.7, we illustrate the number of potential solutions for one password of fixed images. Note that in our implementation, each image-arrangement password is composed of *fresh* images after each trial. Hence, it would be more challenging for attackers to find the right chronological order.

Previous image-based authentication schemes selected photos from a *fixed* collection, such as in [43, 51, 59]. Hence they suffer from attacks based on probabilistic bias of frequently-selected photos [59] or regions in each image [7]. Our authentication challenges come from ever-changing egocentric videos, which are collected naturally in a personalized manner. This mechanism offers more variety in selected image content while being customized to a specific user.

3.5.2 Smudge-based Attacks

In smudge-based attacks, attackers analyze traces (smudges) on the device screen to reconstruct the user's passwords. Our mechanism forms a new au-

Table 3.2. Observation attacks and proposed countermeasure methods

Attacks	Coverage	Countermeasures
Following users	All	Situational awareness
Insider	All	Situational awareness
Social media	Partial	Privacy settings
Location tracking	Partial	Privacy settings
Video surveillance	Partial	Situational awareness

thentication challenge in terms of image content whenever users initiate the authentication procedure or enter wrong answers. This makes it more challenging to adversaries because our authentication challenges are generated from ever-changing videos. Hence, our approach is resistant to smudge-based attacks.

3.5.3 Observation Attacks

In observation attacks, adversaries can access information sources that contain data to infer the chronological order of images displayed in authentication challenges. For example, adversaries can follow the user or leverage social media, location tracking, and surveillance cameras. We summarize these threats and possible countermeasure methods in Table 3.2. An obvious threat for our authentication schemes is when an adversary following the user, remembering visual context, and is in possession of the locked device. The first two attacking strategies in Table 3.2 are derived from shoulder-surfing. In shoulder-surfing attacks, adversaries observe an authentication process and therefore know at least one right configuration of images. The attacking strategies of following users or obtaining insider's information are more sophisticated to perform since adversaries are required to observe users for a longer time (to collect enough visual context data), compared to stealing alphanumeric passwords. Moreover, a fixed password can be obtained and used later while solving ever-changing challenges requires immediate knowledge of users' activities. While possible, we remark that it is challenging for attackers to remain undetectable, especially for a stranger, if the users are aware of their environment. In addition, users may be in a personal space (e.g. office, car, or home) most of the time. Furthermore, as we show below, even if the adversary obtains knowledge on the routine of the user, it is still challenging to guess the generated passwords without tracking the user suspiciously. To better understand this threat, we have conducted a user study. Using the office setting in Section 3.4.1, we implemented an attack in which two informed adversaries with knowledge on the environment tried to solve the authentication challenges of others. These attackers remembered the furniture layout and the list of activities. They reported that two strategies were leveraged to obtain the correct temporal order of images: (1) employing their knowledge on the suggested activities and the furniture arrangement to form a candidate image order, and (2) fixing an arrangement for every graphical password without paying attention to the images. If they could not determine any answer, they tended to choose a random one. For the first strategy, the mean time spent on a single challenge was 54.91 seconds (standard deviation 67.04) with average 10.72 attempts (standard deviation 10.91). In case of the second strategy, each image-based password took the attacker 64 seconds (standard deviation 56.12) with 22.36 attempts (standard deviation 20.82) on average. Even though occasionally the attackers selected the correct order, their effort was much greater than that of a legitimate user. Hence, thresholds can be set on the user's effort (entry time and number of attempts) to lock the personal device.

We also consider the exploitation of side information to obtain the chronological order of images showed in authentication challenges. Nowadays, users may share their data (e.g. locations or photos) in social media services. An attacker with access to these recordings may infer the user routine and crack the authentication challenges. We advise users to control the privacy settings in social media services as a countermeasure method.

The popularity of surveillance cameras provides another source of information for adversaries to leverage. We notice that these cameras are installed in public places. Hence, they only cover a part of a user's routine. The adversaries must therefore combine multiple sources to solve an authentication challenge.

3.6 Conclusion

In this chapter, we presented a novel user authentication mechanism compatible with touch-based interfaces, which takes advantage of implicit information collected with wearable cameras. Our approach offered always-fresh authentication challenges that were personalized to individual users and were changing constantly in each log-in session. We explained how to select video frames that contain meaningful visual details from the first-person perspective. We realized two image-based password designs: *image-arrangement* and *image-selection*. In the first one, an image sequence must be arranged into the correct chronological order. The second allocates images into time intervals and users are required to identify which scenes have appeared or which activities have happened at a certain moment. We conducted multiple user studies to evaluate the proposed authentication mechanism. The security threats were discussed and another case study was conducted to investigate the vulnerability of our approach towards an active adversary. We conclude that our scheme is robust against shoulder surfing and smudge attacks. It can also mitigate threats arising through active adversaries either following the subject or stealing the hardware with the user's situational awareness or a parametric limit of log-in time. Our image-based authentication mechanism supports the security of the *one-to-one* relationship between a device and its legitimate user.

4. Collaborative Inference based on Implicit Data Aggregation

Ericsson has forecast that by 2022 the number of short-range IoT (Internet-of-Things) devices will reach 16 billion, and that of wide-area IoT devices will be 2.1 billion [1]. The increasing number of connected devices has made their power consumption a main concern [29], especially when they encrypt and transmit data [40]. This motivated us to develop data aggregation algorithms that are secure and efficient. We propose to utilize the characteristics of wireless communication channels to perform confidential data aggregation. The interference of simultaneous transmissions can facilitate the implementation of arithmetic functions on the channel itself. Based on that, we can train a machine learning model by offloading partially the computation over the wireless channel. We employ this mechanism on passive radio devices (i.e. backscatter devices) to introduce a energy-efficient collaborative training procedure of inference models with minimal communication. The connected devices collaborates to implement a shared classification model, which is an instance of the MANY-TO-ONE device relationship.

4.1 Training Classifiers across Vertically-Partitioned Data

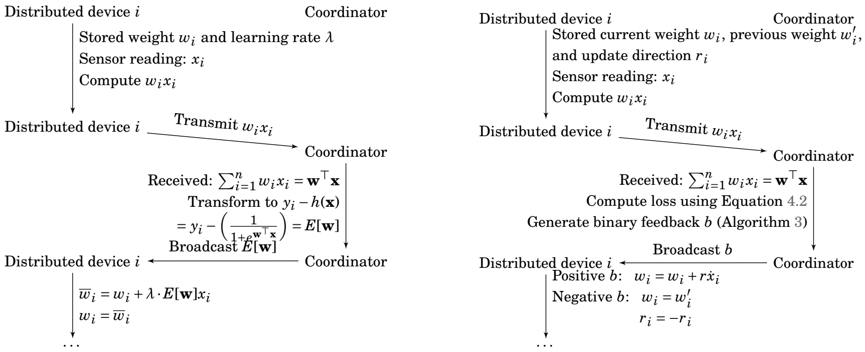
In this section, we describe how weights at individual distributed devices are trained through an interactive protocol. In a nutshell, (1) the coordinator reads the weighted feature values from the channel, (2) evaluates the classification model, (3) shares the loss with the smart devices, that (4) update their weights accordingly. These four steps are repeated until convergence is reached.

In particular, for logistic regression [104], optimal weights can be found by minimizing the error function:

$$E[\mathbf{w}_j] = \begin{cases} -\log(h(\mathbf{x}_j)) & \text{if } y = 1 \\ -\log(1 - h(\mathbf{x}_j)) & \text{else} \end{cases}, \tag{4.1}$$

which is usually formulated as the negative log likelihood (or the loss func-

[1] Internet of Things forecast: https://www.ericsson.com/en/mobility-report/internet-of-things-forecast [Accessed: Jan 18, 2020]


Distributed device i
Coordinator
Stored weight w_i and learning rate λ
Sensor reading: x_i
Compute $w_i x_i$
Distributed device i
Transmit $w_i x_i$
Coordinator
Received: $\sum_{i=1}^{n} w_i x_i = \mathbf{w}^\top \mathbf{x}$
Transform to $y_i - h(\mathbf{x})$
$= y_i - \left(\frac{1}{1+e^{\mathbf{w}^\top \mathbf{x}}}\right) = E[\mathbf{w}]$
Broadcast $E[\mathbf{w}]$
Distributed device i
Coordinator
$\overline{w}_i = w_i + \lambda \cdot E[\mathbf{w}] x_i$
$w_i = \overline{w}_i$
...
Distributed device i
Coordinator
Stored current weight w_i, previous weight w_i',
and update direction r_i
Sensor reading: x_i
Compute $w_i x_i$
Distributed device i
Transmit $w_i x_i$
Coordinator
Received: $\sum_{i=1}^{n} w_i x_i = \mathbf{w}^\top \mathbf{x}$
Compute loss using Equation 4.2
Generate binary feedback b (Algorithm 3)
Broadcast b
Distributed device i
Coordinator
Positive b: $w_i = w_i + r\dot{x}_i$
Negative b: $w_i = w_i'$
$r_i = -r_i$
...

(a) Full feedback

(b) Binary feedback

Figure 4.1. Protocols for the distributed gradient descent algorithm between the coordinator and all n distributed devices. The schematic displays an update process with respect to device i on one sample. The process is repeated with new sensing data. Each distributed device i stores its current weight w_i, the previous weight w_i', and the learning rate λ to control the convergence speed.

tion) [22]:

$$l(\mathbf{w}) = -\log L(\mathbf{w}) = -\sum_{j=1}^{m} y_j \log h(\mathbf{x_j}) + (1 - y_j)\log(1 - h(\mathbf{x}_j)) \tag{4.2}$$

via gradient descent such that for each w_i an improved weight is iteratively found by

$$\overline{w_i} = w_i + \lambda \cdot \frac{\partial}{\partial w_i} E[w_i]. \tag{4.3}$$

In our case, the weights w_i are distributed at the respective devices $i \in \{1, \dots, n\}$. We propose to implement gradient descent across devices spread in the environment (cf. Figure 4.1). In particular, the receiver, which is the coordinator in our scheme, guides the training process for a given class (e.g. an environmental situation that is to be trained). Distributed devices continuously compute and transmit their weighted feature values as Poisson-distributed burst sequences as described in Section 4.2. The coordinator, at receiving $\sum_{i=1}^{n} w_i x_i$, computes the expected loss $E[\mathbf{w}]$ and broadcasts this to the distributed devices, which update their weights accordingly. This process is iterated until convergence is reached. Since this process implements a version of the gradient descent algorithm, the devices eventually approach an optimal configuration of their weights w_i with respect to the trained classes.

Figure 4.1 describes two protocols for training a distributed logistic regression model. Figure 4.1a with full feedback based on the classification error and a modified protocol in Figure 4.1b that achieves gradient descent with minimal binary feedback. In particular, as depicted in Figure 4.1b, instead of transmitting $E[\mathbf{w}]$ to the distributed devices, a binary information that informs whether $E[\mathbf{w}]$ improves or not is broadcast. If the feedback is positive, device i maintains the current direction (increasing or decreasing) in which the weight w_i

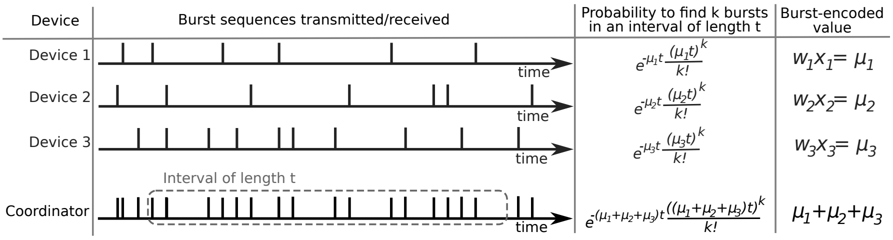

Device
Burst sequences transmitted/received
Probability to find k bursts in an interval of length t
Burst-encoded value
Device 1
Device 2
Device 3
Coordinator
time
time
time
time
Interval of length t
$e^{-\mu_1 t}\frac{(\mu_1 t)^k}{k!}$
$e^{-\mu_2 t}\frac{(\mu_2 t)^k}{k!}$
$e^{-\mu_3 t}\frac{(\mu_3 t)^k}{k!}$
$e^{-(\mu_1+\mu_2+\mu_3)t}\frac{((\mu_1+\mu_2+\mu_3)t)^k}{k!}$
$w_1x_1 = \mu_1$
$w_2x_2 = \mu_2$
$w_3x_3 = \mu_3$
$\mu_1+\mu_2+\mu_3$

Figure 4.2. Computation offloading and aggregation of the weighted feature values $w_i x_i$ during simultaneous superimposed transmission. Weighted feature values are transmitted as the mean $w_i x_i = \mu_i$ of a Poisson-distributed burst sequence. The coordinator estimates the mean $\sum_i \mu_i$ of the convoluted Poisson-distributed sequence to obtain an estimate on $\sum_i w_i x_i$.

is changing. Otherwise it reverses the direction. Regularization terms can be integrated into this scheme if they satisfy an additive factorization form [168]. We implemented mini-batch training for the algorithm [22]. Practically, in order to reduce the variance of updates [22], the weights are modified after computing and transmitting some samples over the channel.

Our update mechanism of the weights is elaborated in Algorithm 2 and the feedback generation procedure is listed in Algorithm 3. Algorithm 2 summarizes the state transition and the corresponding action of each device according to the feedback it receives. Each holds its own set of features for the training dataset. Lines 4 - 6 add or subtract an update that combines a generated quantity r_i and a local feature value x_i to the current weight w_i; then changes the device state to *reversible*. These steps can be customized by putting a probability of updating, which can control the possibility of multiple distributed devices changing their weights at the same time. Lines 8 - 10 reverse the update if it does not improve the model quality, as well as alternating the update direction r_i for future rounds. If the feedback is good (Lines 15 - 20), the devices continue following the current update direction r_i. Algorithm 3 describes how the coordinator generates feedback values when receiving data from the devices over a wireless communication channel. It calculates the loss value at Line 1 before detecting the loss trend (see Equation 4.2 for mathematical formulation). Then, the feedback is generated from Line 3 to Line 7.

4.2 Computation over Backscattered RF Signals

We utilize the superimposition of simultaneously transmitted bursts on the wireless channel to compute part of the distributed classification task. In particular, as demonstrated in [134], it is possible to realize all four basic mathematical operations using Poisson-distributed burst sequences. The scheme is tolerant to weak synchronization of the transmitters and requires only simple operations by participating nodes. The general principle is described in the

```
Algorithm 2: Update algorithm for each distributed device
  Data: A binary feedback value $b \in \{0,1\}$
  Result: Parameter $w_j$ of the distributed device $j$ and its state $s_j$
  if $b = 0$ then
      // Previous update does not improve the model under training
      begin
          switch $s_j$ do
              case 0 do
                  // Possible to update
                  $w_j = w_j + rx_i^j$
                  $s_j = 1$
              end
              case 1 do
                  // Replacing the current parameter with its previous
                     value and alternate the update direction
                  $w_j = \text{Reverse}(j)$
                  $r_i = -r_i$
                  $s_j = 0$ // Ready-to-update state
              end
          end
      end
  else
      if $s_j = 0$ then
          // Possible to update
          $w_j = w_j + rx_i^j$ // Add or substract a value following the current
             update direction
          $s_j = 1$ // Reversible state
      else
          $w_j = w_j + rx_i^j$ // Continue to update following the current
             direction
      end
  end
```

Algorithm 3: Feedback generation at the coordinator

```
Data: A set of values $w_j x_i^j$ from each distributed device $j$ over the sample
      $x_i$
Result: A binary feedback value $b \in \{0,1\}$
// Calculate the loss value
$l(w) = -y_i \log h(x_i) - (1 - y_i)\log(1 - h(x_i))$
$t$ = IsLossDecreasing($l(w)$) // Detecting the trend of loss values
if $t$ = True then
    $b = 1$
else
    $b = 0$
end
```

following paragraph and visualized in Figure 4.2.

For two burst sequences encoding Poisson-distributed variables χ_1 and χ_2 with means μ_1 and μ_2, their combination $\chi_1 + \chi_2$ again yields a Poisson-distributed variable with mean $\mu_1 + \mu_2$ [140]. This is a general property that can be applied similarly to other probability distributions. We leverage Poisson-distributed values due to their representation as a sequence of bursts, which can be combined with superimposition. For the transmission of a value, we divide a burst sequence of length t into κt sub-sequences of length $\frac{1}{\kappa}$ each. Each of these sub-sequences contains with probability p_κ one or more of a finite number of bursts. The Poisson distribution then defines the probability to find k bursts in one such sequence as [52]:

$$p(k;\mu t) = e^{-\frac{\mu}{\kappa}} \frac{\left(\frac{\mu}{\kappa}\right)^k}{k!}. \tag{4.4}$$

The parameter μ determines the density of bursts within the sequence. The larger μ is, the smaller the probability of finding no burst. It is also the mean of the distribution.

We assume that environmental data is recorded by distributed sensors. The resources, especially battery, at such devices are expected to be strictly restricted so that the transmission of sensed values is expensive. Hence, training machine learning models over backscatter sensor networks is a suitable use case for our proposed techniques. Consider n backscatter sensor nodes [13] with corresponding transmit values $v_1, \ldots, v_n$. Each of these devices i defines a Poisson distribution with mean $\mu_i = v_i$. When the antennas of these n backscatter nodes are excited from environmental electromagnetic signals, their transmit sequences are then designed such that each of the κt transmit sub-sequences has the probability $p(k; \frac{\mu_i}{\kappa})$ (cf. Equation (4.4)) that it contains exactly k bursts. Specifically, each respective node switches between its ON-state and OFF-state such that an ON-OFF keying sequence of bursts is generated which establishes exactly the burst occurrence probability $p(k; \frac{\mu}{\kappa})$ for that it contains exactly k

bursts in each transmit sub-sequence. At a receiver, the burst sequences are constructively superimposed. For this received superimposed sequence, note again that the bursts also follow a Poisson distribution, and that the probability to observe exactly k bursts in a sub-sequence of length $\frac{1}{\kappa}$ is then $p(k; \frac{\sum_{i=1}^{n} \mu_i}{\kappa})$ – a convolution of the individual Poisson distributions where the mean of the convolution is the sum of the means μ_i of the individual processes.

We note further that, provided that no collision occurs, for the duration in which the transmit sequences are jointly received from all backscatter nodes, the probability to observe an individual burst in one transmit slot of the received superimposed sequence is identical for all transmit slots. The probability of collisions can be controlled via the length $\frac{1}{\kappa}$ of the sub-sequences and was analyzed in [135]. Equivalently, the probability to observe exactly k bursts within $\frac{1}{\kappa}$ transmit slots of the received burst sequence is identical regardless of which $\frac{1}{\kappa}$ slots are considered. Consequently, when the receiver disregards the beginning and the end of the received burst sequence, transmitting backscatter devices do not require synchronization.

The receiver extracts the combined value $\overline{\mu} = \sum_i \mu_i$, computed during the superimposition of simultaneous transmissions on the wireless channel by estimating the underlying probability distribution. Note that each sub-sequence of length $\frac{1}{\kappa}$ constitutes an individual random experiment so that estimation is possible following the law of large numbers. In particular, the receiver simply counts the number N_i of sub-sequences with exactly i bursts as well as the total number of bursts $T = \sum_{i=1}^{n} i \cdot N_i$. If $N = \sum_{i=1}^{n} N_i$ is large, we expect that $N_i \approx N p(i; \frac{\mu_i}{\kappa})$ [52]. We conclude

$$\begin{aligned} T &\approx N\left(p\left(1;\frac{\overline{\mu}}{\kappa}\right) + 2p\left(2;\frac{\overline{\mu}}{\kappa}\right) + \ldots\right) \\ &= Ne^{-\frac{\overline{\mu}}{\kappa}}\frac{\overline{\mu}}{\kappa}\left(1 + \frac{\frac{\overline{\mu}}{\kappa}}{1} + \frac{\left(\frac{\overline{\mu}}{\kappa}\right)^2}{2!} + \ldots\right) \\ &= N\frac{\overline{\mu}}{\kappa} \end{aligned} \tag{4.5}$$

and consequently

$$\frac{\overline{\mu}}{\kappa} \approx \frac{T}{N}. \tag{4.6}$$

A receiver can therefore extract $\overline{\mu} = \sum_i \mu_i$ from a received superimposed burst sequence transmitted by independent unsynchronized backscatter devices. By utilizing logarithm laws and fractions of values, it is possible to generalize this computation to all four basic mathematical operations (addition, subtraction, multiplication, division) on the wireless communication channel [134].

4.3 Security Analysis

In this section, we discuss two security aspects of the proposed approach. First, the parameters of our trained model are scattered across devices, instead of storing in one central location. Second, our technique of encoding values into burst sequences to transmit simultaneously offers an implicit data perturbation mechanism.

4.3.1 Model Confidentiality

Machine learning models are valuable since training them requires significant effort and a huge amount of data. They are likely to retain sensitive information of training data. Such information could be extracted by attackers once they have access to the model [138]. Storing the classification model in one central location creates a single point of failure, where attackers can concentrate their effort to steal the model parameters.

With our approach, the model is intentionally not shared but instead distributed among devices participating in the training process. Stealing the model in our scheme means to reverse-engineer it from the model prediction, which constitutes a remarkable effort [152]. Since the model is distributed and not known completely to any individual device, it is harder to steal all model parameters and to infer sensitive information of the environment or its inhabitants.

On the other hand, attackers can generate burst sequences to alter aggregation results or even disrupt the model functionality (denial-of-service attacks). Mitigating these threats is considered as future work, e.g. using physical layer security [122] to modify our computation scheme. Note that these threats are also applicable to conventional wireless sensor networks.

4.3.2 Implicit Data Perturbation

We explore an interesting aspect of the proposed approach: implicit data perturbation. As described in Section 4.2, our paradigm partially offloads the computation of a machine learning model to the wireless communication channel. Specifically, each device transforms the transmit values into burst sequences following a Poisson distribution. Since all devices can transmit simultaneously, the receiver can observe the superimposition of all burst sequences. This implicitly hides individual values in the wireless signals, which can be considered as a means of implementing data perturbation. We quantify this property of our computation scheme, using the differential privacy paradigm [46].

Differential privacy [46] is the paradigm to mathematically evaluate the quantity of information leakage from an algorithm applied on sensitive data. We define a dataset $D = (\mathbf{X}, \mathbf{y})$ of which:

- $\mathbf{X} \in \mathbb{R}^{n \times d}$ denotes a matrix in which each row i contains a d-dimensional sample $x_i \in \mathbb{R}^d$ and

- $\mathbf{y} \in \mathbb{R}^n$ denotes a label vector in which each value y_i corresponds to a sample x_i in $\mathbf{X}$.

Consider another dataset D' that differs from D with one sample. D and D' are neighbouring datasets. Dwork and Roth [46] introduced differential privacy to quantify the privacy of a randomized algorithm $\mathscr{M}$ applied on these datasets. Let $\mathscr{O}$ denote the image of $\mathscr{M}$ and $S \subset \mathscr{O}$. $\mathscr{M}$ preserves (ε, δ)-differential privacy if:

$$Pr[\mathscr{M}(D) \in S] \leq Pr[\mathscr{M}(D') \in S] \times e^{\varepsilon} + \delta \tag{4.7}$$

where ε is the privacy budget and δ is the failure probability. If $\delta = 0$, we achieve ε-differential privacy:

$$\ln\left(\frac{Pr[\mathscr{M}(D) \in S]}{Pr[\mathscr{M}(D') \in S]}\right) \leq \varepsilon \tag{4.8}$$

From the definition, low values of ε express high privacy, i.e. less information of the different sample is exposed when performing a function on D and D'. One way to achieve ε-differential privacy is to add noise sampled from a distribution to the outputs of the algorithm $\mathscr{M}$, such as Laplace distribution [46], Gaussian distribution [48], and binomial distribution [47].

Our computation scheme (see Section 4.2) represents feature values from the device i as Poisson-distributed random variables χ_i. Each $\chi_i \sim Poisson(w_i x_i)$ is then encoded into burst sequences in which each burst is generated following a Bernoulli distribution (of a fair coin). Hence, the occurrence of bursts follows a binomial distribution with n trials. This implementation makes the procedure become a randomized algorithm $\mathscr{M}$ with the binomial mechanism. Dwork *et al.* [47] calculated the condition for binomial distributions to achieve (ε, δ)-differential privacy according to the number of trials n as:

$$n \geq \frac{64 \ln(\frac{2}{\delta})}{\varepsilon^2} \tag{4.9}$$

Note that instead of explicitly adding noise to sensitive data, our approach considers the bursts coming from other devices as a mechanism to implement data perturbation for one device. Using Equation 4.9, we can find the minimum number of bursts n in a burst sequence of length t to satisfy the pre-defined (ϵ, δ)-differential privacy policy. Since we allow devices to transmit simultaneously, we can conclude that adding more devices to the network would improve the data privacy.

4.4 Experiments

We study the proposed distributed training of logistic regression with regard to three aspects: the power consumption in comparison to the centralized approach during the training process, the performance of the models trained by our algorithms, and the robustness of detecting burst sequences under changes of the environment. We deploy the computation offloading technique to a backscatter

Table 4.1. Datasets of pervasive systems used to compare decentralized and centralized learning

INDOOR [27]	OUTDOOR [45]
Environmental information in an office over several days. Modalities: temperature, humidity, CO_2 level, and light intensity. The ground-truth was acquired using a surveillance camera. The target of this dataset is to detect occupancy of the office: whether there are people inside the office or not.	DARPA/IXOs Sensor Information Technology experiment: scattered sensors over $900 \times 300\text{m}^2$, separated by at least 20-40m. Modalities: acoustic (microphone), seismic (geophone), and infrared (polarized IR sensor). The data describes vehicles from two classes: tracked and wheeled.

sensor network [13]. In these experiments, we aim to answer the research question: *How to secure data aggregation in collaborative inference through implicit information convolution?*

4.4.1 Decentralized Training

In this section, we compare our distributed training technique to the centralized learning model on multiple datasets (see Table 4.1) in terms of convergence rate and power consumption. With centralized training, all feature values x_i are transmitted to a central device (packet based) [5]. With our decentralized algorithm, the weighted feature values $w_i x_i$ are backscattered via Poisson-distributed burst sequences, and thereby aggregated ($\sum_i w_i x_i$) on the wireless communication channel (see Section 4.2).

To allow repeated execution of the experiments, backscatter devices have been fed with feature samples x_i from existing datasets. For the classification algorithm, it makes no difference whether the values are fed from pre-recorded data, or acquired from a sensor attached to the device. Tables 4.1 and 4.2 summarize the characteristics of the four datasets we used to assess our distributed training scheme. They are diverse in terms of size, number of features, and acquisition sources (i.e. sensors). We presented the results on these datasets to illustrate that our proposed training algorithm were extensible to various application scenarios. In our evaluation, we assume that the number of backscatter devices (i.e. sensors) is equal to the number of features in each dataset. That is, each backscatter device will be fed with values belonging to one particular feature. With OUTDOOR [45], INTRUSION [85], and PHISHING [105], we randomly split 75% for training and 25% for testing. With INDOOR [27], we follow the authors' approach [27] to use six days for training and the remaining days for testing. The data has been evaluated both with a centrally-trained logistic regression (stochastic gradient descent, using scikit-learn 0.20.1 [2]) as well as via our decentralized training approach (see Section 4.1).

To calculate the power consumption of our backscatter prototype hardware, we use the equation $\frac{1}{2}CV^2F$ [171], where C is the capacitance of the diode, V is the

[2] https://scikit-learn.org/stable/ [Accessed: Jan 20, 2019]

Table 4.2. Benchmark datasets used to compare decentralized and centralized learning

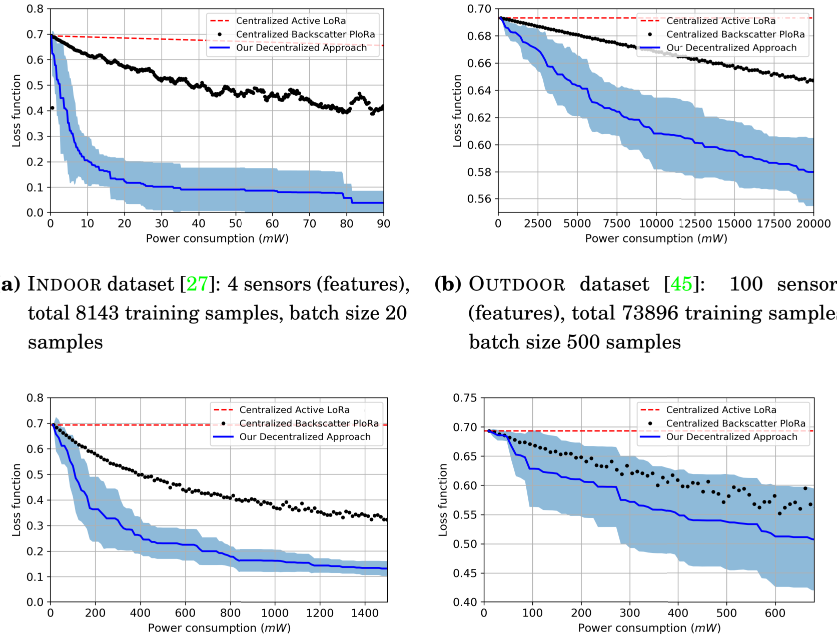
INTRUSION DETECTION [85]	PHISHING [105]
TCP data from 1998 DARPA Intrusion Detection Evaluation [85]: distinguishing "bad" (intrusions) from "good" traffic, using such features as connection duration, used protocol, network service, size of data, connection status, etc.	The dataset is analysed for predicting phishing websites. It characterizes anomaly in potential malicious sites, including description of their links, format, and content.

voltage, and F is the frequency of operating the diode. Based on the datasheet of Infineon Technologies BAR8802VH6327XTSA1 ($C = 0.25\ pF, V = 5\ V, F = 1\ MHz$), the power consumption of our backscatter prototype is $3.125\mu W$ or $0.003125mW$. This amount is lower than that of some backscatter devices reported in the literature. For comparison, that of PLoRa [116] is $0.0035mW$ (which is better than others, including: chirp spread spectrum LoRa backscatter [151] $9.25\mu W$, on-body frequency-shifted backscatter [171] $45\mu W$, and frequency shift keying backscatter [155] $70\mu W$.). The power consumption of an active LoRa node is $4.17mW$ [116]. We repeated the experiments 10 times for each dataset. To show the convergence of our algorithm, we visualized the loss function (minimizing the negative log likelihood function in Equation 4.2) along with their confidence interval. During training, our algorithm uses less power to optimize weights (i.e. parameters) of the logistic regression model, compared to the centralized approach (cf. Figure 4.3).

The prediction accuracy achieved is competitive to that of centralized training (in parentheses): INDOOR 0.92 (0.94), OUTDOOR 0.72 (0.78), INTRUSION 0.98 (0.97), PHISHING 0.75 (0.80). We visualized the confusion matrices on the test samples of four datasets in Figure 4.4, 4.5, 4.6, and 4.7, respectively.

4.4.2 Classification Offloading in Backscatter Sensor Networks

We investigate the classification offloading described in Section 4.2 over a backscatter sensor network [13]. First, we investigate the communication range utilizing omnidirectional antennas (Ettus VERT900 824 to 960 MHz, 1710 to 1990 MHz Quad-band Cellular/PCS and ISM Band omni-directional vertical antenna, at 3dBi gain). As transmitter and receiver, we used the Ettus N200 USRP devices with SBX daughterboards, carrier frequency 868MHz and sample rate at the receiver as 1MHz. The measurements were conducted in the public coffee space of our department. The layout of our experiments is shown in Figure 4.8. The range of a transmission system can be extended into a specific direction by concentrating a larger fraction of the emitted energy in this direction, for instance, by utilizing directional or semi-directional antennas. Then, the semi-directional antenna we use is a microstrip antenna (cf. Figure 4.9). It was designed for the frequency 868MHz and printed in our laboratory. The base material is a fiber glass circuit board (FR4) with copper layers. This has relative



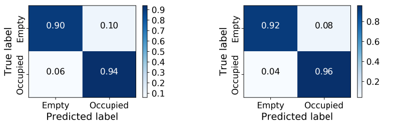
(a) INDOOR dataset [27]: 4 sensors (features), total 8143 training samples, batch size 20 samples

(b) OUTDOOR dataset [45]: 100 sensors (features), total 73896 training samples, batch size 500 samples

(c) INTRUSION [85]: 494020 samples of 40 features, batch size 100

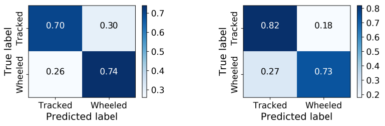
(d) PHISHING [105]: 11055 samples, 30 features, batch size 100

Figure 4.3. Power consumption of the systems implementing our approach and the centralized logistic regression (Active LoRa and Backscatter PLoRa [116])



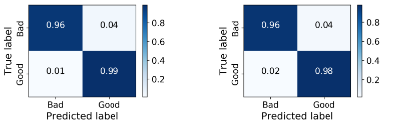
(a) Backscatter **(b)** Centralized

Figure 4.4. Confusion matrices on INDOOR dataset [27]



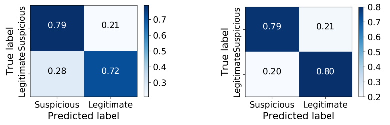
(a) Backscatter **(b)** Centralized

Figure 4.5. Confusion matrices on OUTDOOR dataset [45]



(a) Backscatter **(b)** Centralized

Figure 4.6. Confusion matrices on INTRUSION dataset [85]



(a) Backscatter **(b)** Centralized

Figure 4.7. Confusion matrices on PHISHING dataset [105]

permittivity in the order of 4.5. This antenna has a maximum gain of 4.7 dBi which is directed along the x-axis. It is attached to the transmitter and facing towards the backscatter device (cf Figure 4.8). By having an SMA connector on the backscatter board, we have been free in the choice of antenna, and could therefore also experiment with different frequencies. A drawback of this design choice is the large dimension of the system. For a production-level system, we propose to print a patch antenna, which is cheap to manufacture, on the back of the circuit board. Such an antenna needs a rectangular space of $\frac{1}{4}$ to $\frac{1}{2}$ of the wavelength λ. For instance, a backscatter system with a patch antenna in the 2GHz range could have dimensions smaller than 4cm×4cm. In our measurement (cf. Figure 4.8), the transmitter was fixed 3m from the backscatter device. The distance to the receiver was gradually increased (step size 1m) up to 6m. The measurements were repeated 10 times for each distance and antenna types: semi-directional and omnidirectional. At 6m, the receiver could only capture noise signals. Note that the measurement equipments employed in this scenario was suitable to use in our experiment area. Their compact form limited them to certain transmission capability such as frequencies and transmit power. Using the same backscatter device, Badihi *et al.* [13] conducted the measurement of working distances with a signal generator [3]. In their experiment, they reported a working distance up to 30m in an outdoor environment. Their research aimed to evaluate the capability of this hardware prototype while ours concentrated on assessing the proposed algorithms.

Next, we describe a case study on occupancy monitoring (dataset *Indoors* [27]) in an office with a network of backscatter sensor devices [13]. We employed four devices, which represent sensing components (light, temperature, humidity, and CO_2). They broadcast their weighted sensed values ($w_i x_i$) periodically via Poisson-distributed burst sequences. The dataset comprises three subsets: six-day (for training), two-day (testing), and seven-day (testing). According to the authors [27], all samples were averaged over non-overlapping 60s windows since each sensor had its own sampling rate. The ground-truth of occupancy status was obtained through manual annotation of the videos captured by a surveillance camera. The trained parameters w_i were obtained through the training process described in Section 4.1. Two Ettus N200 USRPs were used as transmitter and receiver. Both were equipped with SBX daughterboards. The transmission frequency was 868MHz and the sampling rate of the radio devices 1MHz. Feeding sensor values from the INDOOR dataset, we achieved accuracy of 0.89 (F_1 score 0.88). For comparison, for logistic regression trained via gradient descent on a desktop computer (Intel Core i5 1.8GHz, 8GB RAM, with scikit-learn library version 0.20.1), we achieved an accuracy of 0.94 (F_1 score 0.93).

[3] SMBV100A Rohde & Schwarz: https://www.rohde-schwarz.com/fi/product/smbv100a-productstartpage_63493-10220.html [Accessed: June 08, 2020]4

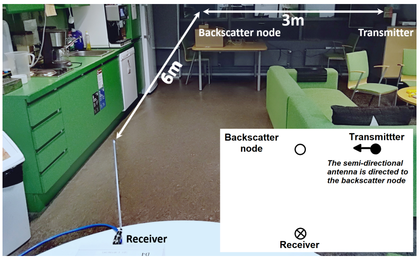


Figure 4.8. Layout of our experiments on the operating distance

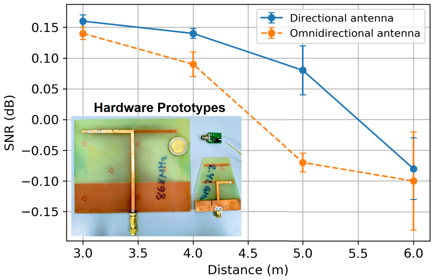
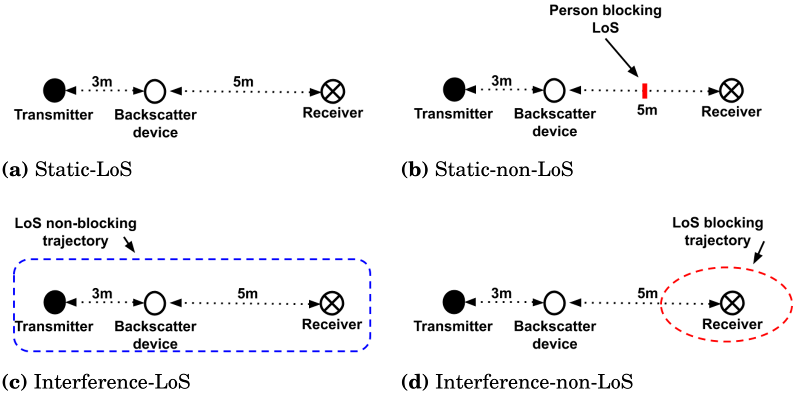


Figure 4.9. Signal-to-noise ratio (SNR) with various distances



(a) Static-LoS

(b) Static-non-LoS

(c) Interference-LoS

(d) Interference-non-LoS

Figure 4.10. Layout of devices in the experiments on environmental variation

4.4.3 Effect of Environmental Changes

Human behaviour inside the monitored space can significantly affect the multipath condition of wireless signals [167]. For example, a person may block the line-of-sight (LoS) between a backscatter device and the receiver (RX) or human movement may distort the backscattered signal. This kind of noise might impair the correct counting of bursts at the receiver. Hence, we investigated the impact of four types of human interference with the layout shown in Figure 4.10.

Experiment on the environmental effect

Participant : We recruited one male subject whose age was 32 years old and height was 1.65m.

Materials : We employed one backscatter device and two USRP devices, one as the transmitter (TX) and one as the receiver (RX).

Design : Three devices were located as in Figure 4.10. The distance between TX and RX was fixed at 8m. The distance between TX and the backscatter device was fixed at 3m. We varied the distance between the backscatter device and TX at the step of 1m.

Procedure :

Static-LoS No person in the room or people sitting still and not blocking the line-of-sight

Static-non-LoS A person blocks the line-of-sight between backscatter device and RX)

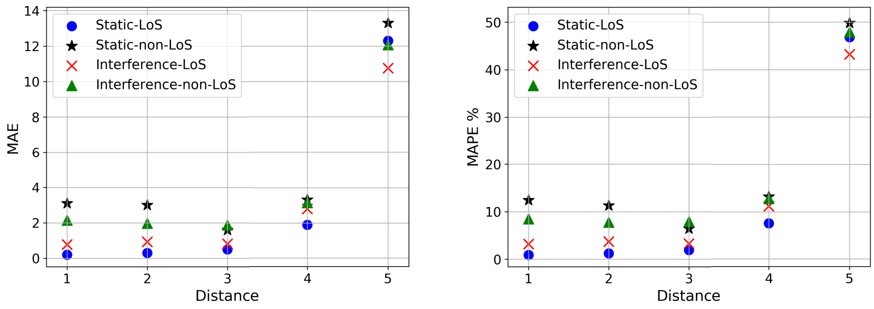
Interference-LoS One person moves around in the room, not blocking the line-of-sight

Interference-non-LoS One person moves around freely in the room, occasionally blocking the line-of-sight between a backscatter device and RX

In all cases, we varied the distance between the backscatter device and the receiver from three to five metres (step size one metre). For each distance, we controlled the backscatter device to transmit 50 distinct values (encoded into burst sequences). These are humidity samples from the INDOOR dataset [27], ranging from 16.7 to 39.1 with mean 25.7 and standard deviation 5.5. We analyzed the burst sequences at the receiver and recovered the transmitted values. Then, we calculated the Mean Absolute Error (MAE) and the Mean Absolute Percentage Error (MAPE), which are shown in Figure 4.11. We observed that the MAE was less than 3.5% when the backscatter device is within a radius of 3 metres from the receiver, even in the case of movement not blocking the line-of-sight. However, the error increased when the backscatter signal was occasionally blocked. This issue can be addressed e.g. by locating backscatter devices at positions higher than human height. When the backscatter device was five metres or farther from the receiver, our algorithm could not reliably detect burst sequences due to the weakness of the backscattered signal which resulted in high error values.

4.5 Conclusion

This chapter introduced a distributed optimization procedure to train machine learning models for vertically-partitioned data from scattered resource-



(a) Mean Absolute Error (MAE) of transmitted and received values

(b) Mean Absolute Percentage Error (MAE) of transmitted and received values

Figure 4.11. Evaluation of the burst detection algorithm with various environmental conditions

constrained devices. Our approach incorporates a computation offloading mechanism implicitly leveraging the interference of backscattered radio signals. It therefore allows battery-free distributed learning, where each backscatter sensor device learns its own set of model parameters, instead of training and storing a model centrally. Each backscatter device senses and processes environmental data, and broadcasts the weighted feature values $w_i x_i$ as Poisson-distributed burst sequences to a coordinator by reflecting a carrier signal. Through superimposition of simultaneously transmitted burst sequences, the burst-encoded weighted feature values $w_i x_i$ are aggregated as their weighted sum $\sum_i w_i x_i$ when received by the coordinator. The coordinator extracted the weighted sum $\sum_i w_i x_i$ to evaluate the models in training. It then issues binary feedback to the sensors to guide the model optimization process. Hence, distributed devices can update their optimal model parameters without sharing sensing information with either their neighbouring sensors or the coordinator. To compare the convergence of our scheme to centralized approaches, we extensively evaluated and compared it to a traditional centralized approach. Our training technique consumed less power than that of the centralized algorithm utilizing radio transceivers. To further prove the practical feasibility of our approach, we have implemented and evaluated it a backscatter sensor network. Our approach extends applications of backscatter communication beyond data transmission: distributed training and offloading a classification model over the wireless communication channel. That is, it protected data transmitted in the *many-to-one relationship* of networked devices in an efficient way. On the other hand, we are aware that our approach is still vulnerable to such attacks as radio jamming, sybil attacks, and compromised hardware. We consider the countermeasures as a future direction of this dissertation.

5. Proximity-based Secure Communication using Vocal Commands

Nowadays, smart devices with voice user interfaces (VUIs) such as Apple Siri and Amazon Alexa can analyze users' vocal commands to perform certain tasks. We aim to leverage them to connect devices in a new environment. For example, imagine a person arriving at a building, where she has not been before. Her smart-phone, a personal device or PD, can connect to the local wireless network. She would like to securely access a printer or connect to a projector she observes in the same room with her. Since the location is new to the user, she does not know the specific device name or identity. She would need to explicitly ask for instruction to pair with local devices via Wi-Fi or Bluetooth. This raises inconvenience from the user's point-of-view. We propose to use contextual information to establish the secure communication channel between the user's device and the share appliances. Specifically, our approach analyzes vocal commands to select a specific device type and initiate the secure connection between a personal device and shared appliances in a new environment, which is an application of the ONE-TO-MANY device relationship.

5.1 Context-based Device Pairing

Sensor modalities suited for context-based device pairing include magnetism [77], RF-signals [154, 82], luminosity [103], and audio [129]. Truong *et al.* [153] investigated the performance of four commonly available sensor modalities (Wi-Fi, Bluetooth, GPS, and audio) for co-presence detection and found that Wi-Fi is better than the rest. Also, they showed that, compared to any single modality, fusing multiple modalities improved resilience while retaining a high level of usability. Miettinen et al. [103] used co-presence and a continuous authentication scheme to pair devices. Their underlying assumption stated that only devices that were worn together or were located nearby would *in the long run* measure the same luminosity or ambient audio. Another method of proximity-based device pairing which required manual user interaction was presented in [9]. Their system used fuzzy cryptography to generate a shared secret on two devices from correlated drawings on the displays of the devices [130].

A conceptional challenge with all context-based authentication approaches is that due to sensing inaccuracies, different hardware and noise, the recorded signals are likely not identical but only similar. Fuzzy cryptography presents a methodology to obtain identical keys from similar patterns [79]. In particular, by mapping the patterns into the codespace of an error-correcting code, mismatches can be mitigated without disclosing the pattern over a potentially insecure channel. These approaches have been applied to various noisy data traces for authentication, such as face biometrics [160].

In our case, we propose to leverage audio rather than other modalities, since it features better room-level recognition due to the longer wavelength and hence less drastically changing environment of the channel. A vocal command is a natural way to generate a pairing key to initiate the connection between devices that support VUIs.

5.2 Audio-based Pairing Protocols

In order to connect a personal device (PD) with one appliance of a certain class (e.g. smart screen with a VUI), a user would express the expectation by speaking out the pairing intention. During this interaction, users are not restricted to any format or convention but they need to mention the expected device class in their request. We then utilize speech recognition in order to extract the device class as the first unique identifier and, in addition, generate an implicit secure key from the same spoken audio command as the second unique identifier. Only the appliances in proximity and belonging to the correct device class match both unique identifiers and are thus identified for secure pairing through a remote device manager.

5.2.1 Audio-based Pairing Protocols and Application Scenarios

Our approach is represented in three protocols which correspond to three application scenarios. Two of them can be implemented without a central authority (e.g. a Device Manager). We describe these protocols as follows, within their respective application scenarios.

- **Scenario 1** (cf. Figure 5.1): *Protocol 1. Key exchange and management without a central authority*. In case there is no central key distribution authority, proximate devices can leverage contextual information to form ad-hoc device groups. The registration and de-registration process (i.e. joining and leaving a group) rely on context-based secret keys only.
- **Scenario 2** (cf. Figure 5.2): *Protocol 2. Device group formation based on context*. Our mechanism adds a fine-grained layer to the conventional group key management framework. For example, the user's smart-phone S is connected with the printer P_1 (in the corridor) and the printer P_2 (at the user's office) in the local wireless network. That means S, P_1, and P_2

share the group key. With our context-based key generation technique, the framework can issue a new secret key only for S and P_1 (based on proximity). S does not need to leave the former group. Furthermore, if attackers compromise P_2, they can not access the new group formed by S and P_1.

- **Scenario 3** (cf. Figure 5.3): *Protocol 3. Context-based device discovery*. A user's device U is in the same group with multiple local devices L_i, which may belong to different contexts (e.g. in different rooms). U wants to access an unprecedented L_i in the same room. It can obtain the device information at a certain location, i.e. L_i. After that, U can connect to the specific L_i.

Our proposed basic protocol (i.e. Protocol 1 in Figure 5.1) does not require a central authority to distribute pairing keys and manage the secure connection. We introduce the following scheme. A set of mobile devices are willing to establish a common secret key extracted from ambient audio data. Each device records a number of audio samples and then independently computes an audio fingerprint [28]. These fingerprints are binary sequences that are designed to fall into the code-space of a Reed-Solomon error correcting code. Audio fingerprints generated from similar ambient audio resemble each other. However, due to noise and inaccuracy in the audio-sampling process (i.e. caused by hardware and software diversity), it is rarely that two fingerprints are identical. The devices therefore utilize the capability of an error-correcting code to map fingerprints to codewords. For two fingerprints whose Hamming distance is within the configurable threshold of the error-correcting code, the codewords are identical and then can be utilized as shared secret keys.

Implementing the aforementioned audio-based approach, we can establish a secure communication session between a user's device and a local device with or without a central authority (which can be referred to as a Device Manager). We assume that both partners are connected to the same wireless network, and the approach increases the connection security with contextual information. Our proposed mechanism ensures that the devices can instantiate a secure session if they are in close proximity to each other (e.g. in the same room). We denote $KGF()$ as an audio-based key generation function which derives one cryptographic key from an audio fingerprint of ambient sounds. Then, the user's device D establishes a secure session with the local shared appliance L according to the following steps (cf. Figure 5.1):

1. D sends the initialization message to L to start the key generation process.
2. D and L start recording a sequence of ambient audio whose length is specified a priori.
3. D and L locally compute the audio fingerprints f_D and f_L, respectively.
4. D transforms f_D to the secret key $K_D = KGF(f_D)$ while L transforms K_L to the secret key $K_L = KGF(f_L)$. Using an error-correcting code (e.g. a

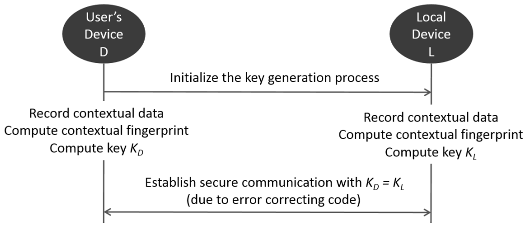

User's Device D
Local Device L
Initialize the key generation process
Record contextual data
Compute contextual fingerprint
Compute key K_D
Record contextual data
Compute contextual fingerprint
Compute key K_L
Establish secure communication with $K_D = K_L$
(due to error correcting code)

Figure 5.1. Protocol for key exchange and management without a central authority

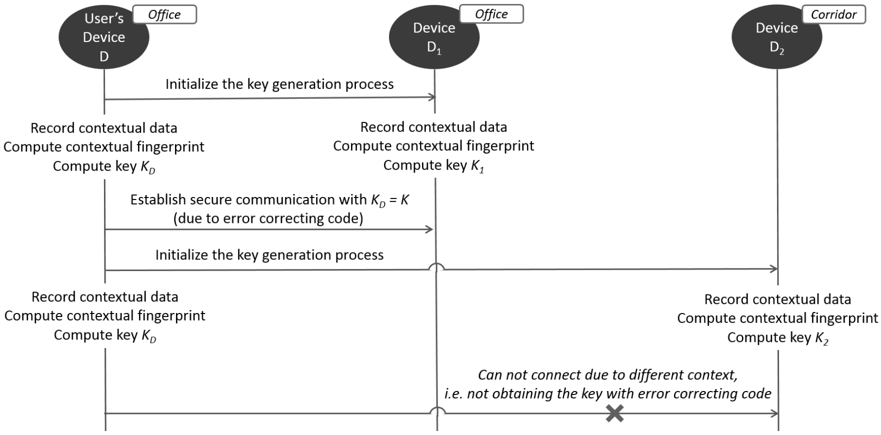

User's Device D
Office
Device D_1
Office
Device D_2
Corridor
Initialize the key generation process
Record contextual data
Compute contextual fingerprint
Compute key K_D
Record contextual data
Compute contextual fingerprint
Compute key K_1
Establish secure communication with $K_D = K$
(due to error correcting code)
Initialize the key generation process
Record contextual data
Compute contextual fingerprint
Compute key K_D
Record contextual data
Compute contextual fingerprint
Compute key K_2
Can not connect due to different context,
i.e. not obtaining the key with error correcting code

Figure 5.2. Protocol for device group formation based on context

Reed-Solomon code [120]), both partners can derive the same secret key K if the Hamming distance between K_D and K_L satisfies a predefined threshold t.

These protocols and scenarios are appropriate for Internet-of-Things devices, including mobile and wearable appliances. They allow the users to freely select local shared devices which are equipped with VUIs. Furthermore, they strengthen the conventional communication channels (e.g. Wi-Fi) in terms of security and usability.

5.2.2 Security Analysis

We introduce a number of attacking strategies and discuss their severity. We include attackers of various technical competences, ranging from guessing attacks to hardware-based attacks. An adversary device in the same context with the user is a valid threat for our protocols. By adapting the loudness of spoken audio, the user is able to implicitly control which devices are the recipients provided that the user can observe where the suspicious eavesdroppers are. We analyzed this attacking strategy in Section 5.5.

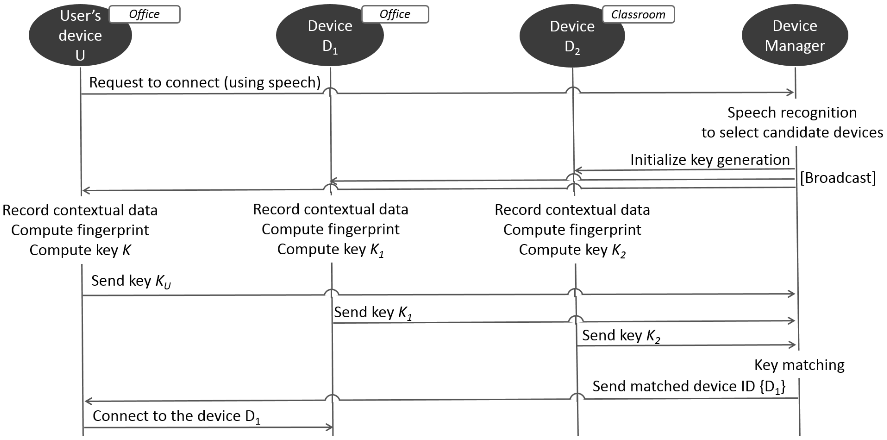


Figure 5.3. Protocol for context-based device discovery

Guessing Attacks
Without any knowledge on the audio context, an attacker can try to guess an audio fingerprint (i.e. a brute force attack) which is sufficiently close to those generated by the user's device and the local shared appliance. This must be done when the user's device and the local shared appliance is in the audio fingerprinting step. Hence, the attacker is in a limited time frame. According to Schürmann and Sigg [129], the success probability of a guessing attack is 1024^{-204} when we employed: 512-bit audio fingerprint, the Reed-Solomon code $RS(2^{10},204,512)$, the set of words $A = \mathbb{F}_{2^{10}}^{204}$, and the set of codewords $C = \mathbb{F}_{2^{10}}^{512}$. In addition, the proposed audio fingerprinting algorithm [129] was analysed for its entropy and potential bias with respect to common weaknesses in pseudo random number generators and no bias could be found after the DieHarder set of statistical tests [25].

Impersonation Attacks
Our protocols are vulnerable to impersonation attacks on device identification information. Without a remote trusted authority, the PD is not able to authenticate device IDs according to the protocols. A possible way to prevent such attacks is to employ a certified authority (e.g. a Device Manager in Figure 5.3). It is unlikely for a device outside of the context (i.e. not in the same room with the user's device PD and the shared appliance SA), to impersonate a device inside the context. This is due to the audio sequences generated by the PD and the attacker's device is not sufficiently similar [129].

Hardware-based Attacks
An attacker can fully or partly control the audio input observed by the user's device and the local shared appliances. It could further be possible to bypass data acquisition and re-use recorded audio data. The adversary could place a device to record audio near the user (e.g. attached to the owner of the PD) and

relay the audio recordings to a remote device. This attack would be successful as long as it satisfies the strict timing requirement controlled by the audio pairing protocols: audio acquisition should start within at most 10ms delay. We conclude that these attacking strategies are sophisticated and require access to either facilities or hardware components of the user's devices The attackers might compromise either the VUIs (the user's device PD and the shared local appliance SA) or the device manager. Our protocols do not guard against such attacking strategies.

5.3 Similarity of Audio Fingerprints in Different Vocal Commands

In order to verify and demonstrate the feasibility of the proposed use case, we implement the protocol specified in Figure 5.3. We aim to answer the research question: *How to use implicit information in vocal commands to select shared devices naturally and securely?* The case study demonstrates that audio-based device selection based on the verbal input of a user is feasible. For this, we developed an Android application which extracted and compared audio fingerprints of proximate devices. In particular, while the subjects specified appliances they want to pair to using unconstrained free speech, the application recorded voice and ambient audio, extracted audio fingerprints [129] and compared the fingerprint similarity. As detailed by Schürmann and Sigg [129], fuzzy cryptography can then be applied in order to correct bit errors in the generated binary fingerprints.

For the experiment, two Android mobile phones running the audio-based adhoc pairing application were placed in the same room at distances of at least one metre. The subject then chose one out of a given five possible device classes (printer, projector, monitor, speaker, TV) and spoke out a request containing the name of the particular device (for instance, the command might be *"I would like to connect to the speaker"* if a user wanted to pair a smart-phone with the Bluetooth speaker in the room) while the Android application was running, recording the ambient audio to generate the secure keys separately on the two devices. The users were asked to issue natural voice commands. Spoken audio sentences were in the order of 2-4 seconds depending on the speaking speed and the length of each sentence. The users were located at two distinct locations on a university campus: a meeting room and an office. The former had more background ambient sounds than the latter. In each environment, the users repeated the experiment 10 times for each device class. Table 5.1 shows the results of the case study. The table depicts the similarity of audio fingerprints generated for the various sentences containing different device classes and conducted in different locations.

Table 5.1. Average similarity (%) of audio fingerprints. Values in brackets are standard deviation.

	Meeting room	Office
Printer	68.2 (14.8)	60.4 (10.5)
Projector	74.6 (5.2)	67.6 (16.5)
Monitor	69.6 (7.6)	74.2 (4.4)
Speaker	75.0 (13.2)	75.4 (6.6)
TV	73.8 (8.8)	63.6 (9.1)

Experiment on command-specific audio

Participant : We recruited two male subjects ($\mu_{age} = 35$), who were fluent at English.

Materials : We implemented an Android application to generate the audio fingerprints and installed it on two phones. We selected five classes of common appliances (printer, projector, monitor, speaker, TV) that are possible to equip VUIs. We emitted a pre-recorded audio file of a meeting as background noise.

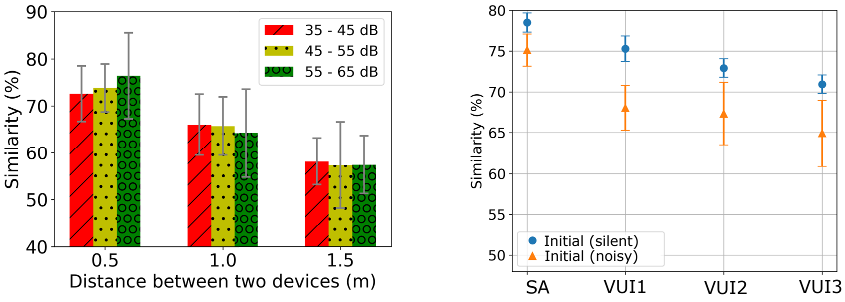
Design : We placed two phones in the same room at distance of one metre. There are two rooms in our experiment: a meeting room and a private office.

Procedure : In each room, the subjects repeated the vocal commands 10 times for each device class. The application captured their voice and background sounds and extract the audio fingerprints.

We observe that the average similarity of audio fingerprints is 70%. Hence, we suggest utilizing fuzzy commitment scheme to derive the secure keys for device selection. For example, an error-correcting code such as the Reed-Solomon code [120] can be used to correct 30% of the bits in the generated audio fingerprints, as employed by Schürmann and Sigg [129].

5.4 Impact of SNR on Audio Pairing

In this experiment, we simulate a scenario in a small room with multiple VUIs. In order to generate a setting that can be repeated and verified, we utilized a phone broadcasting continuously speech recordings. Two phones (the VUIs) were placed in $d_1 = 0.5$ metres and $d_2 = \{1.0, 1.5, 2.0\}$ metres from the audio source. This models a scenario with multiple VUIs scattered across a room. We controlled the sound level of the office (i.e. the audio source) in $35-45$, $45-55$, and $55-65$dB, which correspond with verbal conversation loudness (i.e. an individual raising or lowering her voice to intuitively adapt



(a) Similarity of audio fingerprints with respect to the distance between the personal device (PD) and the shared appliance (SA, VUI-supported)

(b) Similarity of audio fingerprints extracted at the personal devices (PD), the shared appliance (SA), and other VUIs (VUI$_1$, VUI$_2$, VUI$_3$) in quiet and noisy environments

Figure 5.4. Similarity of audio fingerprints in our experiments

to the range of the restricted zone). For all combinations of these settings, audio fingerprints [129] were then extracted according to the steps described in Section 5.2. The similarity of audio fingerprints (based on Hamming distance) is shown in Figure 5.4a.

The similarity of audio fingerprints decreases when the distance increases. Hence, it is possible to configure a fuzzy cryptography scheme [129] to allow only VUIs in a certain proximity (see Section 5.2)). The width of this zone can be implicitly controlled by verbal conversation loudness as depicted in Figure 5.4a. At extended distance, when the emitted sound becomes less audible, the similarity approaches that of random sequences (i.e. 50%). We also performed an experiment to measure the similarity of audio fingerprints collected at the personal device (PD), the shared appliance (SA), and other VUI-supported devices. The noisy environment is simulated using loudspeakers. Figure 5.4b shows that the fingerprint similarity between the PD and SA is higher than that between the PD and other VUIs. More details of this experiment are described in Section 5.5.

5.5 Human and Hardware Eavesdroppers

Experiment on eavesdroppers

Participant : We recruited eight subjects ($\mu_{age} = 35$), who were fluent at English.

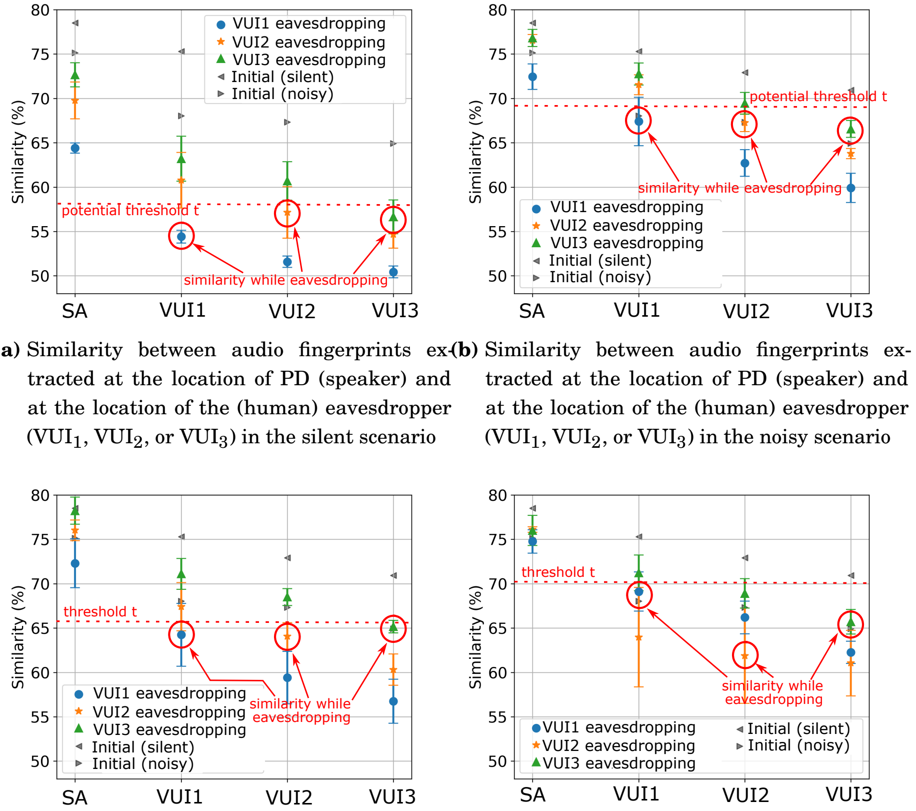
Materials : Five microphones were used to record audio from the subjects and the environment.

Design : Two microphone were located by a distance of one metre. Three microphones are placed in distances of one metres, two metres, and three metres from the speakers but they were not in the direction of speaking (i.e. they were not between the speakers).

Procedure : Each pair of subjects exchange spoken information to each other by reading eight sentences from the Harvard set of phonetically balanced sentences [11].

Our solution relies on the capability of a human subject to control the loudness of her voice accurately with respect to VUIs in proximity. We are interested whether it makes a difference if the eavesdropper is another human or whether it is a VUI. In this section, we describe an experiment conducted in an indoor environment where two subjects exchange spoken information while we place VUIs (represented by microphones) in their proximity. One subject with a personal device (PD) is talking (directionally) to a shared appliance (SA), separated by a distance of 1m) while audio fingerprints are recorded at both locations. In addition, three microphones are placed in distances of 1m, 2m, and 3m from the speaker (but not in the direction of speaking). We repeated these experiment in a quiet environment and in an another where we generated artificial background noise from a crowd of people speaking in the distance. This background noise was played back from a recording to generate approximately the same noise in different repetitions of the experiment for comparison among the recordings. The noise was generated by mixing 32 simultaneous speech sources extracted from the TIMIT database [173]. The loudspeakers were located at one end of the room pointing towards the wall in order to produce diffused noise. To simulate a conversation between the speaker and listener in the scenarios, subjects were asked to read eight sentences from the Harvard set of phonetically balanced sentences [11]. Subjects were asked to adapt their voice loudness level such that an eavesdropper (located at one of the locations of VUI_1, VUI_2, or VUI_3) could not overhear their conversation in each scenario (silent vs. noisy).

In total, we processed 418 audio recordings from eight subjects to evaluate all fingerprints produced by our systems. We expected that the similarity of audio fingerprints to decrease when the distance between the speaker and the eavesdropping VUI (microphone) increased [129]. In the initial recording for both quiet and noisy environments, the speaker was instructed to talk to the



(a) Similarity between audio fingerprints extracted at the location of PD (speaker) and at the location of the (human) eavesdropper (VUI$_1$, VUI$_2$, or VUI$_3$) in the silent scenario

(b) Similarity between audio fingerprints extracted at the location of PD (speaker) and at the location of the (human) eavesdropper (VUI$_1$, VUI$_2$, or VUI$_3$) in the noisy scenario

(c) Similarity between audio fingerprints extracted at the location of PD (speaker) and at the location of the (VUI) eavesdropper (VUI$_1$, VUI$_2$, or VUI$_3$) in the silent scenario

(d) Similarity between audio fingerprints extracted at the location of PD (speaker) and at the location of the (VUI) eavesdropper (VUI$_1$, VUI$_2$, or VUI$_3$) in the noisy scenario

Figure 5.5. Similarity of audio fingerprints in the eavesdropping experiments

second subject in her normal voice. We did not disclose to the speaker that the microphones at 1m, 2m, 3m distance were eavesdropping on the conversation. Figure 5.4b shows the similarity of fingerprints for these initial recordings. As expected and already confirmed in the previous experiments, we observed that the similarity in fingerprints decreases with increasing distance. Hence, it is possible to define an error-correction threshold t in the error-correcting code of our protocol, such that the respective restricted zone separates the two speaking subjects and the VUIs.

5.5.1 Human Eavesdropper

Next, we were interested in the capability of the subjects to adjust the audio loudness of their voice (i.e. volume level) in order to control the range in which their personal and public devices can connect. To investigate this, a human subject was located next to VUI_1, VUI_2, or VUI_3 (3 separate configurations). Only one eavesdropper was present during the experiment and the subject was instructed to adapt the volume of their speech so that the eavesdropper was just unable to overhear the conversation. This setting was repeated for each VUI location (VUI_1, VUI_2, and VUI_3) and for both scenarios (silent and noisy).

Figure 5.5a summarizes the results in the silent scenario. We observe that, as expected, the similarity in audio fingerprints decreases with increasing distance between the PD and the VUIs. The similarity is highest between the PD and SA since the speech audio was directed towards the location of the SA. In addition, notice that the similarity is lower than what we observed in the initial scenario (cf. Figure 5.4b) for both the silent and noisy cases. The figure distinguishes the fingerprint similarity for locations of VUI_1, VUI_2 and VUI_3 while the human eavesdropper was located at either of these locations.

The result shows that, with a proper threshold, it is possible to implement the restricted zone such that a specific eavesdropper can be excluded from this zone by the speaker. For instance, as depicted in Figure 5.5a, the similarity in fingerprints has always been between 55% and 57% for the location where the eavesdropper was located. At the location of the SA, the minimum similarity was higher with roughly 64%. The results for the noisy environment are depicted in Figure 5.5b. In contrast to the silent scenario, the fingerprint similarity rises between the PD and SA but it also rises for VUI_1, VUI_2, and VUI_3. However, a threshold $t = 68\%$ will still keep the eavesdropping VUIs outside the restricted zone. We conclude that a human speaker is able to control the range of a restricted zone with her voice in both silent and noisy environments.

5.5.2 VUI Eavesdropper

The potential threat of a human eavesdropper is common, especially when attackers are familiar with users in their day-to-day situations (e.g. speaking in meetings or public gatherings). A more typical situation for a human subject during speech communication is to lower her voice towards a suspected human eavesdropper. However, VUI-equipped appliances are, due to their size, easily ignored during communication.

We repeated the experiment in a setting where no human subject was present. Figure 5.5c illustrates our results in a silent environment. We observe that, indeed, this scenario is more difficult to control for the human subject (speaker) at the location of the PD, since the similarity of the fingerprint at the PD and VUI_1, VUI_2 and VUI_3 is higher than in the previous section. However, the similarity for the eavesdropping VUI is still beneath 65%, while the similarity of the fingerprints recorded by the PD and SA is exceeding 72% at least, so that a human speaker could again actively exclude a specific eavesdropping VUI.

Likewise, the situation for the noisy scenario is depicted in Figure 5.5d. This is the most difficult scenario since the similarity between the audio fingerprint at the PD and one of the VUIs (namely VUI_1) reaches even 70%. However, despite the increased loudness of the subject's voice, this is still lower than the similarity to the PD, which is roughly 75%.

5.6 Conclusion

Using vocal commands and environmental sounds, we presented three context-based implicit pairing protocols and described how these can be integrated into secure communication of distributed IoT devices. In the protocols, free-form spoken interaction is interpreted by speech recognition to identify the device class to pair with while audio-fingerprints are generated from the same spoken interaction in order to generate secure keys via fuzzy cryptography. Both the device class and the secure key are then utilized as a unique identifier to pair a personal device with a proximate appliance of the requested class. We performed a case study in which users select partner devices through natural vocal commands and two attack settings with human and hardware eavesdroppers. We conclude that our proposed approach can establish a secure communication key between new devices in a restricted zone while preventing attackers from obtaining the key. The scenarios studied in this chapter cover the ONE-TO-MANY relationship between a personal device and shared appliances in a new environment.

6. Continuous Secure Device Pairing

Recent decades have witnessed the proliferation of computers in compact forms that can be carried by users: smart-watches, mobile phones, tablets, laptops, etc. These devices can spontaneously form a network for data exchange, which is a representation of the MANY-TO-MANY relationship. We propose a secure pairing protocol which is based on the surrounding context of partner devices. Device pairing [61] is the procedure to establish a secure connection between devices that have previously never been in contact with each other. However, this protocol relies either on a PIN-code explicitly verified by users or a fixed PIN code, which can be cracked due to its short length [132]. We proposed pairing mechanisms which analyze the surrounding context of partner devices to derive secret keys. Our approaches offer twofold advantages: non-obtrusiveness and continuous pairing (including automatic de-grouping of devices). We introduced key generation algorithms based on gait and heartbeat information in Section 6.2 and Section 6.3, respectively. The former was suitable for ambulatory activities such as walking and running while the latter was applicable in resting postures.

6.1 Device Pairing

The communication capability of mobile devices enables them to interact with each other to exchange data or collaborate in a shared task [63]. Their interaction is often spontaneous [31] with a varying number of connected devices and implicit dis-connection. Users demand convenient yet secure methods in order to determine the trust between these devices. This procedure is often deployed over a wireless communication channel, such as Near Field Communication (NFC) [1], Bluetooth [61], ZigBee [4], Wi-Fi [33], or Wi-Fi Direct [3]. Traditional device-pairing mechanisms (such as Bluetooth [61]) are performed with three assumptions [142, 14, 58]:

- Communication channel: The two devices have access to a common wireless communication channel but there is no assumptions about the security of this channel.

- User interaction: The two devices are both monitored by either a single user or a pair of users who trust each other.
- User interface: Both devices have a means to input or output, e.g. they have at least a keypad or a display. If a device does not have a keypad, then it must at least have an input, e.g. a button, allowing the successful conclusion of a procedure to be indicated to the device. Similarly, if a device lacks a display, then it must at least have an indicator to signal the success or failure of a procedure (e.g. red and green lights or a sound output).

Currently, novel mechanisms have been developed to implement device pairing that leverage built-in sensors to fulfil the following requirements:

- Implicit and continuous device pairing: the protocols do not require explicit user interaction and the secret keys are continuously updated.
- Automatic unpairing (i.e. de-grouping of devices): when a device leave the protected area, it can not decrypt the group's message since the secret key is updated.
- Support devices with limited user interface, such as e-textile: these approaches utilize sensor data to generate the secret keys for pairing instead of using a user's input or verification.

Our proposed approaches address the above requirements through using behavioural (gait) and physiological (heart-beat) data to secure the communication of on-body devices that have not previously been connected.

On-body device pairing has been implemented using acceleration data collected from shaking movements [56] and human gait [108]. The latter is well suited in wearable scenarios since the gait information can be extracted at any body location and it is confined to a individual person [84]. Recently, several authors have considered acceleration or gait for the pairing of devices co-present on the same body [141, 65, 107]. In particular, these approaches exploit correlation in acceleration signals when devices are worn on the same body [87, 34] or shaken together [55, 56]. These device-pairing protocols execute quantization on one or more devices at the same time to generate similar bit sequences. In contrast to user authentication, these sequences are not matched against a template database. Instead, they are used to authenticate a key agreement procedure between all participating parties.

The ShakeUnlock protocol [55, 56] unlocks a mobile device when it is shaken simultaneously with a worn smart-watch. This approach relies on the comparison of acceleration sequences to compute correlation and has to be done within an pre-established secure communication channel. On the other hand, there exist approaches that do not require any already-established secure connection. For authentication based on arbitrary co-aligned sensor data, Mayrhofer [100] proposes the candidate key protocol, whose advanced variant was implemented in SAPHE [62] to solve the issue of low-entropy input data. It uses hashes from acceleration sequences as short secrets and concatenates those with matching hashes to form the key.

Human gait, being used as biometrics for a long time [78, 37], has been applied in device pairing recently. The Inter-Pulse-Interval (IPI) between consecutive steps has been utilized for secret key generation from gait sequences [149]. The protocol uses acceleration data along the z-axis and concatenates IPIs as key sequences. Its security depends on the speed of consecutive steps and the step length. This protocol was verified with gait data captured by devices on the torso of subjects (lower back, upper right arm and right ear). Xu *et al.* [166] presented Walkie-Talkie, which leveraged correlated acceleration sequences from human gait. The authors use individual samples for key bits if they deviate more than a threshold comparing to the mean gait. One extended version is Gait-Key [165] providing a higher bit-rate by applying multiple thresholds. In another scheme by the same authors [121], acceleration data on all three axes is used as a random source to generate a group key. This key is locked in a fuzzy vault using a secret set based on the acceleration along gravity direction. Other wearable devices can unlock the vault if they can obtain a secret set which is sufficiently similar to the original one. Instead of using all three acceleration axes, the BANDANA protocol [128] utilizes acceleration along the z-axis only. It fingerprints human gait from the difference between instantaneous gait and mean gait at each body location. Remaining errors in the binary fingerprints are corrected with fuzzy cryptography exploiting BCH codes [71, 21]. In an extended version [127], the required key length has been reduced from 64-bit to 16-bit through using a Password Authenticated Key Exchange (PAKE) [125].

6.2 Gait-based Pairing of On-body Devices

Smart devices have continued reducing their size to fit in wearable gadgets, such as wristwatch, glass frame, or even e-textile. These compact appliances are equipped with computational, sensing, and communication capability. They can collect contextual data using microphones, accelerometers, gyroscopes, cameras, etc. Some of these devices have limited user interfaces, which cause difficulty to establish the secure connection with other appliances. One common solution is to rely on a fixed PIN code to initiate the communication, which can be cracked [132]. We propose to implement a continuous pairing mechanism of on-body devices by extracting gait information from an accelerometer and a gyroscope. In the next section, we introduce our approach, called BANDANA or *Body Area Network Device-to-device Authentication using Natural gAit*.

1. Collect acceleration readings from the z-axis
2. Correct rotation wrt gravity (gyroscope)
3. Bandpass between 0.5Hz and 12Hz
4. Resampling (40 samples/gait) and gait detection
5. Compute mean gait
6. Difference between mean and instantaneous gait translates to binary sequence
7. Calculate reliability of bits, disregard least reliable
8. Share reliability ordering and create fingerprint
9. Fuzzy cryptography: Get key from fingerprint

Figure 6.1. BANDANA device pairing protocol based on gait information

6.2.1 Body Area Network Device-to-device Authentication using Natural Gait

An on-body device pairing protocol is required to generate keys with high similarity between different locations on the same body (intra-body) and low similarity between different bodies (inter-body). We propose to use the deviation of an instantaneous gait cycle to the mean cycle as implicit information to generate gait-based keys. In our case, the mean gait is accumulated over a few consecutive gait cycles. This can be obtained independently at most positions on the human body, using accelerometers. The comparison between instantaneous and mean gait at a particular body location serves as a normalization procedure. The offset to the mean gait has a better correlation across different body parts than comparing absolute acceleration values. Our proposed scheme covers both upper and lower parts of the body, including extremities, in contrast to related work such as [166], which only considered upper body parts.

We summarized our proposed protocol to extract gait-based secret pairing keys from acceleration data in Figure 6.1. Due to body movement, acceleration data collected at different positions changes their orientations dynamically and independently. Hence, we transformed the data in such a way that one of the three axes follows the opposite direction of gravity through Madgwick's algorithm [96]. We then applied a Type II Chebyshev bandpass filter to retain the frequency between 0.5 Hz and 12 Hz due to the characteristics of human body movement [128]. The result of these two steps is a sequence of n z-axis acceleration values $\mathbf{z} = \{z_1, \ldots, z_n\}$, which is the input for the gait cycle detection algorithm. Next, we detected gait cycles inside the pre-processed data. The un-normalized length of a gait cycle is the time interval between two consecutive steps. To identify repetitions in the input signal, we estimated the discrete auto-correlation at the time lag k:

$$A_{corr}\left(k\right) = \frac{1}{(n-k)\sigma_2}\sum_{t\in\mathbb{Z}} z_{t+k}\bar{z}_t \tag{6.1}$$

where $\bar{z}_t$ is the conjugate of z_t at time $t \in [1,n]$. The output auto-correlation values $\mathbf{a} = \{a_1,\ldots,a_n\}$. Then we found m local maxima in $\mathbf{a}$ as $\zeta = \{\zeta_1,\ldots,\zeta_m\}$. Next, we calculated the mean of difference between consecutive local maxima:

$$\delta_{mean} = \left\lceil \frac{\sum_{i=1}^{m-1}\zeta_{i+1}-\zeta_i}{m-1} \right\rceil \tag{6.2}$$

We then found the indices of local minimum in $\mathbf{z}$ limited to the range of δ_{mean} with a correction factor τ to account for small deviations in gait cycle length:

$$\begin{gathered} \mu = \{\mu_1,\ldots,\mu_{m-1}\} \\ \mu_i = \text{argmin}(z_{\zeta_i-\tau}, z_{\zeta_i-\tau+1},\ldots,z_{\zeta_i+\delta_{mean}+\tau}) \end{gathered} \tag{6.3}$$

The set of indices μ are used to separate $\mathbf{z}$ into gait cycles:

$$\begin{gathered} \mathbf{Z} = \{Z_1,\ldots,Z_q\} \\ Z_i = \{z_{\mu_{\frac{i}{2}}},\ldots,z_{\mu_{\frac{i+1}{2}}-1}\} \end{gathered} \tag{6.4}$$

where $i \in \{2,4,\ldots,q\}$.

After the aforementioned pre-processing and gait cycle detection steps, we generated gait fingerprints as binary sequences and used them to derive secret keys for device pairing. Our quantization algorithm leveraged the deviation between a current and the mean gait sequence. The mean sequence is continuously updated as:

$$\begin{gathered} \mathbf{A} = \{A_1,\ldots,A_p\} \\ A_j = \frac{\sum_{i=1}^{q} Z_{ij}}{q} \end{gathered} \tag{6.5}$$

The mean gait cycle A_j was then overlapped with each instantaneous gait cycle Z_i to derive the binary fingerprint:

$$\begin{gathered} \hat{\mathbf{f}} = \{\hat{f}_{11},\ldots,\hat{f}_{1\frac{p}{q}},\ldots,\hat{f}_{b1},\ldots,\hat{f}_{b\frac{p}{q}}\} \\ \hat{f}_{ij} = \begin{cases} 1, & \text{if } \delta_{ij} > 0 \\ 0, & \text{otherwise} \end{cases} \\ \delta_{ij} = \sum_{k=0}^{\frac{p}{q}} A_{i+k} - Z_{i+k,j} \end{gathered} \tag{6.6}$$

The values of δ were sorted in the descending order of $|\delta_{ij}|$ to form the reliability vector $\mathbf{r} = (r_1,\ldots,r_M)$. Between two devices A and B, the independently-generated vectors $\mathbf{r_A}$ and $\mathbf{r_B}$ were exchanged. The one with the higher hash

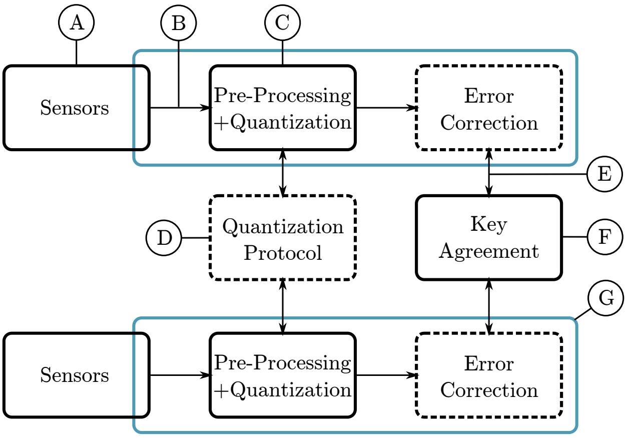


Figure 6.2. Conceptual view of the protocol with attack vectors (blue line depicts device boundary, dashed lines indicate communication between the devices)

value (e.g. SHA-256) was selected on both devices. We then removed the least reliable bits so that the first N bits formed the fingerprint on each device. The remaining mismatches in the fingerprints were corrected with an error-correcting code to produce binary keys k for secure device pairing. We proposed to use BCH code with two parameters (K,N) in which K and N are the length of a fingerprint $\mathbf{f}$ and its corresponding key $\mathbf{k}$, respectively. To derive a binary key k of length K, we transformed a gait fingerprint into the message-space $\mathbf{f} \xrightarrow{Decode} \mathbf{k}$. This decoding function could correct up to $u = \lfloor \frac{N-K}{2} \rfloor$ errors. After the decoding step, two binary keys of a pair of on-body devices should be identical. Based on the required key length in bits and the threshold u for a successful pairing (i.e. devices are on the same body), the required fingerprint length was $N = \frac{K}{2u-1}$. Later, k could be used to derive a share secret s through a key agreement protocol.

6.2.2 Security Analysis

We visualize the attack vectors of the gait-based device-pairing protocols in Figure 6.2. These protocols have the same flow: the devices with built-in sensors collect motion data, perform pre-processing steps, quantize the data to bit sequences, apply error-correcting codes, and agree on a secret key. We discuss the potential attacks according to the attack vectors annotated as A - G labels in Figure 6.2. Especially, we dedicate Section 6.4.2 for the gait-reconstruction attack based on video analysis (G).

One-Shot Guessing Attacks

Without knowledge about the user's gait, an attacker can perform a brute force attack (C) to facilitate a Man-in-the-Middle attack (E) or an impersonation attack (G). Since we continuously generate the communication keys when the user is walking, the brute force attack becomes a one-shot guessing attack. In our proposed protocol (see Section 6.2.1), a gait fingerprint is a 48-bit sequence. In each sequence, 16 bits are discarded to amplify the reliability. Then, using BCH codes, up to eight bits can be corrected, resulting in 16-bit keys. Thus, the success probability of a single randomly-drawn fingerprint is: $\sum_{k=0}^{8}\left(\begin{array}{c}32\\k\end{array}\right)/2^{32} = \frac{\sum_{k=0}^{8}\left(\frac{32!}{(32-k)!\cdot k!}\right)}{2^{32}} \approx 0.0035$.

Gait Mimicry Attacks
Muaaz and Mayrhofer [109] showed that even professional actors could not mimic the observable gait of the users with similar physical characteristics (age, weight, height, shoe size, and upper leg length). However, by walking next to a user, one out of five attackers was able to reach sufficient similarity in the gait acceleration sequences captured by the sensors (A). The assumed reason was that the users instinctively adapted their walking styles to synchronize with those of the attackers.

Hardware-based Attacks
An adversary can force the user to walk in a certain way (e.g through manipulating the walking surface) in order to control the motion data captured by the sensors (A). It is possible to feed the sensors with data from the past or synthetic data (B). However, these attacking strategies are sophisticated and require access to either facilities or hardware components of the user's devices (i.e. compromised devices).

6.3 Representation Learning from Ballistocardiography Data

The aforementioned gait-based algorithms depend on repetitive movements of users to derive keys continuously. We proposed to use heartbeat data collected by highly-sensitive accelerometers to pair on-body devices during resting activities (such as standing, sitting, and lying). We presented a learning model to extract the keys from the acceleration data. Our model combined Siamese networks [24] and de-noising auto-encoders [156]. Although the architecture was applied in other domains such as signature verification and face authentication, ours is the first attempt to employ it for device pairing based on heart-beat data. Our proposed approach is a general paradigm that can handle different types of diverse sensor data.

6.3.1 Ballistocardiography

The heart is a muscular organ which circulates blood throughout the whole body. In a single heartbeat, the ejection of blood into the great vessels produces subtle and repetitive motions. This is physiological information that can be captured by non-invasive devices from the surface of the body. Ballistocardiography (BCG) [143] is the field that measures and analyzes this signal. With the development of sensing technology, BCG can be performed through an accelerometer in contact with human bodies [86], for example the one in a typical smart-phone [68].

The BCG signal has proven its usefulness in user authentication [157] and activity recognition [68]. In the former, Vural *et al.* [157] investigated the identification of individuals with an accelerometer placed on the sternum. In their study, the subjects were instructed to sit stably. A three-axis accelerometer was placed over the chest to record tiny movements of the body caused by heartbeats. For feature extraction, the authors split each heartbeat into two regions, then computed the spectrogram matrix of each region before concatenating them. Fifty bins with the highest relative entropy were selected as features for individual identification. A Gaussian mixture model was trained for each subject and a background model was generated to detect impostors. In the field of activity recognition, Hernandez *et al.* [68] demonstrated that wearable motion sensors could identify wearers and recognize their still body posture (sitting, standing, and lying). They attached commercial off-the-shelf devices (smart-phones and smart-glasses) to two body locations (head and wrist) for movement data collection. From each 10-second windowed data, the features were extracted: raw amplitude, 200-bin histogram of amplitude values, and shape descriptors (angles and distances between five descriptive points of each heartbeat). A linear Support Vector Machine was trained and tested in a cross-subject manner.

6.3.2 Heart-beat Detection

We implemented an algorithm to extract heartbeat regions in acceleration data (see Algorithm 4). Our algorithm analyzes a window of acceleration values to find a potential heartbeat. There are two tunable parameters: window length and heartbeat distance. First, the input sequence is segmented into fixed-length windows and overlapping is possible. Then, the minimum value is located, which can be considered at the centre peak in a heartbeat. After that, a region is expanded to cover the whole beat. This algorithm returns a list of starting and ending locations.

6.3.3 Representation Learning Model Architecture

We would like to combine Siamese networks [24] and de-noising auto-encoders [156] to learn an optimal representation for sensor-based device pairing. The Siamese architecture encourages the discrimination of devices worn by different users while an auto-encoder acts as a fingerprint extraction algorithm. The network architecture is pictured in Figure 6.9, which is called a regularized Siamese network [30].

The Siamese network was presented by Bromley *et al.* [24] to verify handwritten signatures. It includes two neural networks that share identical weight

Algorithm 4: Algorithm of finding heartbeats in acceleration data

```
Input: a sequence of z-axis acceleration data $\mathbf{x}$, window length $l$, and beat
       distance $d$
Output: a list $\mathbf{l}$ of starting and ending positions $(s_i, e_i)$ of found heartbeat
        regions
while not at end of $\mathbf{x}$ do
    extract a window $\mathbf{w}$ at the current position $c$;
    find position $p$ of the minimun value in $\mathbf{w}$;
    if beat found then
        extend around $p$ to obtain $(s_i, e_i)$;
        append $(s_i, e_i)$ to $\mathbf{l}$;
        jump to the next position $c = c + d$;
    else
        slide the window;
    end
end
```

parameters. Given two inputs $\mathbf{x}_1$ and $\mathbf{x}_2$, these networks produce two corresponding outputs $\mathbf{o}_1 = f(\mathbf{x}_1)$ and $\mathbf{o}_2 = f(\mathbf{x}_2)$, respectively. The parameters are trained in such a way that the distance $d(\mathbf{o}_1, \mathbf{o}_2)$ reflects a similarity relation. In each branch of a Siamese network, the ouput is not a vector of posterior probabilities as in classification problems, but it is considered as a feature vector. Let $C = \{C_1, ..., C_K\}$ be the set of K classes in the data, o_1 the output of a reference sample x_1 of C_i, o_2 the ouput of another sample x_2 of the same class, and o_3 the ouput of a sample x_3 from any C_j where $i \neq j$. The goal of training a Siamese network is to maximize the dissimilarity of inter-class samples while minimizing that of intra-class ones. For example, if $\mathbf{x}_1$ and $\mathbf{x}_2$ are signature images of the same person while $\mathbf{x}_3$ is that of an attacker, $d(\mathbf{o}_1, \mathbf{o}_2) < d(\mathbf{o}_1, \mathbf{o}_3)$ and $d(\mathbf{o}_1, \mathbf{o}_2) < d(\mathbf{o}_2, \mathbf{o}_3)$. In the training process, instead of using individual samples, *positive* and *negative* pairs of samples are required. Positive pairs contain two samples of the same class while negative ones include samples of different classes.

An auto-encoder [156] is an unsupervised learning technique in which a feed-forward non-recurrent neural network is trained to reproduce an input. It maps an input vector $\mathbf{x}$ to a hidden representation $\mathbf{o} = f(\mathbf{x}) = s(\mathbf{W}\mathbf{x} + \mathbf{b})$, where $\mathbf{W}$ is the weight matrix, $\mathbf{b}$ is the bias vector, and s is the activation function. Then, the resulting representation $\mathbf{o}$ is reconstructed to $\mathbf{x}^* = g(\mathbf{o}) = s(\mathbf{W}'\mathbf{x} + \mathbf{b}')$. It is possible that $\mathbf{W}' = \mathbf{W}^T$. The parameters $\mathbf{W}$, $\mathbf{W}'$, $\mathbf{b}$, and $\mathbf{b}'$ are optimized through minimizing the reconstructed error: $\mathbf{W}, \mathbf{W}', \mathbf{b}, \mathbf{b}' = \text{argmin} \frac{1}{n} \sum_{i=1}^{n} L(\mathbf{x}^{(i)}, \mathbf{x}^{*(i)})$, where n is the number of training samples and L is the loss function. The principle motivates that auto-encoders can be applied to extract features from sensing data. We trained a standard auto-encoder as described above to encode heartbeat acceleration data. The auto-encoder can reduce noise while amplifying peaks, which is useful for fingerprint generation.

6.3.4 Security Analysis

The pairing protocol based on heart-beat information follows a procedure similar to that of the gait-based pairing protocols (see Section 6.1): the devices with built-in sensors collect motion data, perform pre-processing steps, extract the fingerprints from the Siamese auto-encoder, apply error-correcting codes, and generate secret keys. Hence, it exposes to similar vulnerabilities under guessing attacks and hardware-based attacks (see Section 6.2.2). In our approach, the outputs of the auto-encoder $f(\mathbf{x})$ (where $\mathbf{x}$ is the input acceleration data) is used as the heart-beat fingerprints. They are mapped onto the key-space of an error correcting code, where t bits can be corrected. This process, while mitigating errors in the fingerprints, might boost the success probability for a one-shot guessing attack. Assuming $|c|$-bit fingerprints of which t bits are corrected to result in $|c|-t$-bit keys, the success probability of a single randomly drawn sequence is then only: $\sum_{i=0}^{t} \left(\begin{array}{c} |c| \\ i \end{array} \right) / 2^{|c|} = \frac{\sum_{i=0}^{t} \left(\frac{|c|!}{(|c|-i)! \cdot i!} \right)}{2^{|c|}}$. Although the heart-beat information is less observable than the gait, we are aware that there are sophisticated methods to reconstruct the heart-beat information from afar such as using radio-frequency signals [90]. Lin *et al.* [90] could reconstructed fiducial-based descriptors of heart-beats. Their system was applied in user authentication with the minimum authentication time of 1s, which may be not suitable for continuous device pairing. The timing restriction is one of the different requirements between authentication and pairing. We can exploit it to prevent the heart-beat reconstruction attack. Next, we then analyze two attacking strategies that are specific to our proposed approach.

Attacks on Our Learning Model

Adversaries can exploit the Siamese auto-encoder, which is similar to pattern classifiers used in biometric authentication, to facilitate various attacking strategies such as model evasion during testing (to cause misclassification), data poisoning during training (to control the model behaviour with certain samples), and spoofing attacks [18]. The arm race between attacks and countermeasures has even initiated a new research direction called adversarial machine learning [113]. Years of work in this direction have shown that model evasion and data poisoning attacks could be prevented through building another classification model to detect adversarial samples. On the other hand, the uniqueness of living traits in human cardiac motion can be leverage against spoofing attacks [90].

Heart-beat Synchronization Attacks

Although heart-beat synchronization (or imitation) is less feasible than gait mimicry, there exist special scenarios that make it probable. Ferrer *et al.* [53] showed that the heart rates of a man and woman could be synchronized if they

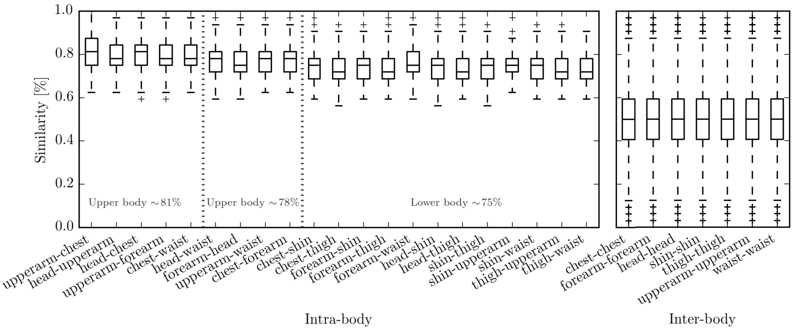


Figure 6.3. Similarity of intra-body and inter-body gait fingerprints produced by BANDANA. Each value in the intra-body region is defined by the similarity of fingerprints extracted from two different locations on the same body (all possible combinations within each body). Each value in the inter-body region is defined by the similarity of fingerprints from different bodies at the same location.

were a couple. However, we have not found any research that investigates the similarity of couples' heart-beat shapes (i.e. fiducial descriptors) as represented in BCG data. Hence, we consider this attacking strategy as a future extension of this dissertation.

6.4 Experiments

In this section, we implemented and evaluated our proposed approaches to answer the research question: *How to continuously generate secret keys for device pairing from implicit information?* We evaluated two techniques: gait fingerprinting in Section 6.4.1 and heart-beat fingerprinting in Section 6.4.3. Section 6.4.2 covered the video-based attack that reconstructed human gait using a surveillance camera.

6.4.1 Evaluation of Gait Fingerprints

This experiment was performed using walking data recorded by Sztyler *et al.* [150] [1]. The dataset includes data captured from 15 subjects (age 31.9±12.4, height 173.1±6.9, weight 74.1±13.8, eight males and seven females). The data acquisition devices were smart-phones with built-in accelerometers and were attached at seven different body locations (sampling rate: 50Hz): chest, forearm, head, shin, thigh, upper arm, and waist.

A gait fingerprinting scheme for secure device pairing has two essential characteristics. First, the generated fingerprints on different body parts of the same subject (intra-body) are more similar than those extracted from different subjects (inter-body). Second, the generated fingerprints are sufficiently-unpredictable

[1] Human Activity Recognition Dataset: http://sensor.informatik.uni-mannheim.de [Accessed: June 08, 2020]

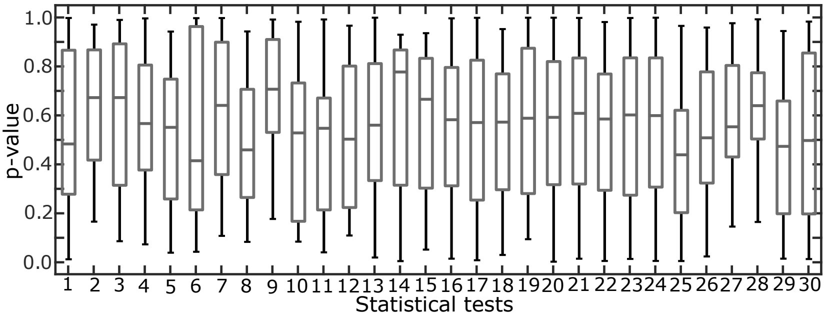


Figure 6.4. Distribution of p-values achieved for fingerprints after 20 runs of the DieHarder set of statistical tests. The tests are: (1) birthdays, (2) operm5, (3) rank32x32, (4) rank6x8, (5) bitstream, (6) opso, (7) oqso, (8) dna, (9) count-1s-str, (10) count-1s-byt, (11) parking, (12) 2D circle, (13) 3D sphere, (14) squeeze, (15) runs, (16) craps, (17) marsaglia, (18) sts monobit, (19) sts runs, (20) sts serial[1-16], (21) rgb bitdist[1-12], (22) rgb minimum distance[2-5], (23) rgb permutations[2-5], (24) rgb lagged sum[0-32], (25) rgb kstest, (26) dab bytedistrib, (27) dab dct 256, (28) dab fill tree, (29) dab fill tree 2, and (30) dab monobit 2.

bit sequences because they are used to derive cryptographic keys. We implemented BANDANA on the aforementioned dataset [150] to evaluate these two characteristics. The similarity between gait fingerprints extracted at all seven body parts is depicted in Figure 6.3. The proposed approach achieves similarity above 75% for all location pairs on the same body (intra-body). It is able to reduce the chance of the attacker (inter-body) to random guess. Hence, we can configure an error-correction code to eliminate the remaining 25% of difference for the intra-body devices. Next, to test randomness of the generated fingerprints, we ran the DieHarder statistical tests [25] on the generated gait fingerprints. Figure 6.4 displays the p-values produced by 20 runs of the DieHarder test suite. BANDANA provides a stable distribution of p-values. A slight bias is associated with the squeeze test (14), which employs a chi-square test for the number of multiplication between 2^{31} and random floats on (0,1) to reduce 2^{31} to 1. These results show that our proposed gait fingerprinting scheme is suitable to generate secret keys for on-body device pairing.

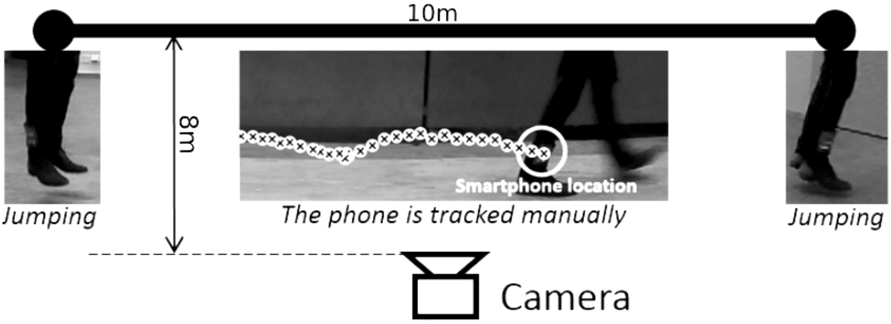


Figure 6.5. Experimental setup for video-based attack on gait-based pairing

6.4.2 Video-based Attack on Gait-based Device Pairing

Experiment to reconstruct gait information from videos

Participants : We recruited five subjects ($\mu_{age} = 31$) and one of them is female. Their height is from 1.63m to 1.95m ($\mu_{height} = 1.76$m).

Materials : We used a smart-phone with built-in accelerometer and gyroscope to capture the inertial data and a high-speed GoPro camera to record the videos.

Design : The smart-phone was attached to the right ankle of each subject. Each subject walked in a straight line of 10m. The camera was located at the distance of 8m perpendicular to the walking route.

Procedure : Each subject was instructed to jump, walk 10m, and jump in order to synchronize the inertial data and the videos. We annotated the videos manually to extract the acceleration data, which was then aligned and compared to the inertial data from the smart-phone.

Video-capturing devices are omnipresent these days, for example surveillance cameras, personal camcorders, or mobile phones. The quality of captured videos is sufficient to discriminate subtle movements. An adversary with camera-support might therefore be able to extract pairing keys from recorded videos. In this section, we investigate the threat of video-based side-channel attacks (G). We investigate how accurate the gait can be estimated by tracking movement of body parts from videos.

For our experiment, we captured the movement of a subject simultaneously with a wearable inertial measurement unit (embedded in a smart-phone) and with a high-speed GoPro camera. The smart-phone was attached to one leg. Five subjects (4 male; height: 1.63-1.95m; $\mu = 1.76$m) walked in a straight line at approximately 8m distance to the camera (1080p resolution; 90fps) mounted on a tripod (cf. Figure 6.5). Acceleration data was sampled at 50Hz. For synchronization between the video and inertial sensor, a single jump both at the

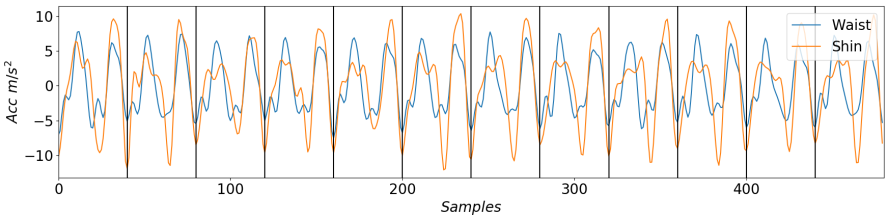


Figure 6.6. Gait cycles extracted from shin and waist.

beginning and at the end framed the walking segment. Each subject conducted the experiment twice. We utilized Tracker[2] to manually track the location of the smartphone on the recorded video. Although human pose estimation [101] is able to estimate leg movements, we manually marked locations of the smart-phone on the video frames.

For gait-based on-body pairing, the attacker is free to estimate gait according to the most easy to attack body location, since the protocols are inherently designed to pair acceleration sequences from arbitrary body location pairs. Spearman's coefficient (1: perfect monotonically increasing relationship; 0: non monotonic relationship; -1: perfect monotonically decreasing relationship) [139] for gait sequences extracted at waist and shin in the dataset [150] is 0.44, which reflects their moderate increasing monotonic association. For instance, the correlation between gaits extracted from these locations can be observed in Figure 6.6.

From the tracked trajectory we estimated the acceleration of the smart-phone. We calculated the velocity in horizontal and vertical directions before computing the acceleration. The obtained result was smoothed by a Gaussian filter to reduce annotation noise. This estimated acceleration sequence was then re-sampled to match the 50Hz sampling rate of the inertial sensor. Note that we estimated the movement orthogonal to the ground since any rotation is implicitly corrected by the pairing scheme (Figure 6.7a).

To estimate the pairing performance and noise from video-extracted acceleration, in the dataset [150], we estimated the mean $\mu_v = 2.0921$[3] and standard deviation $\sigma_v = 6.0210$ of disparity values between optimally synchronized[4] gait acceleration sequences (estimated and recorded) in our experiment. These values were then used as parameters for noise distributions, which we added to the walking data recorded by the dataset in [150]. We generated Gaussian, Laplacian, and uniformly distributed noise[5].

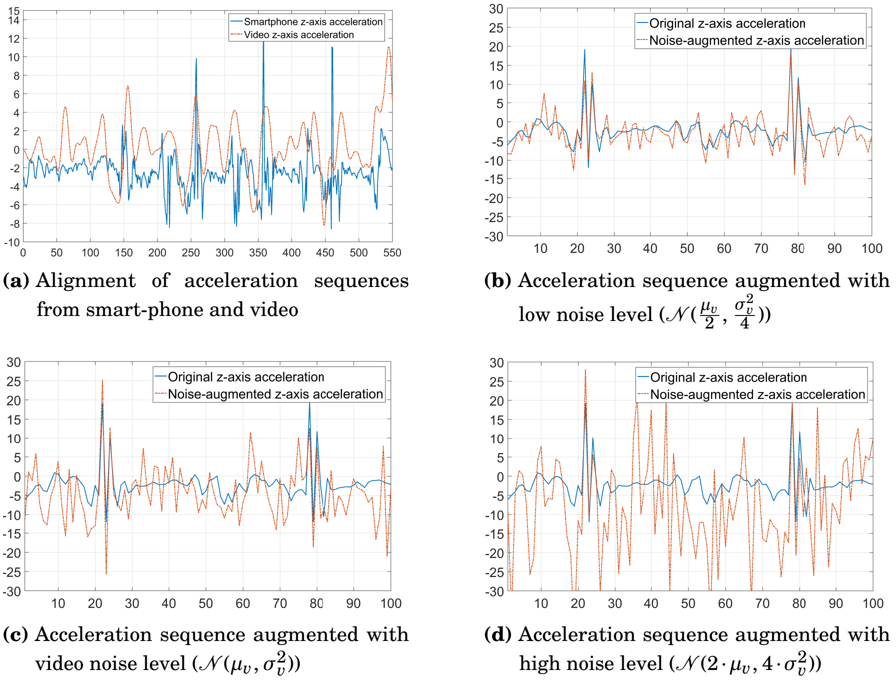
We then generated noisy acceleration signals with $\mathcal{N}(\mu_v, \sigma_v^2)$ (noise observed from video-based acceleration estimation), $\mathcal{N}(\frac{\mu_v}{2}, \frac{\sigma_v^2}{4})$ (low noise) and $\mathcal{N}(2\cdot\mu_v, 4\cdot$

[2] http://physlets.org/tracker/

[3] From the amplitude estimation error due to inaccurate distance measurement between camera and walking subject.

[4] We refined the synchronization between the estimated and recorded acceleration sequences by shifting both sequences until a minimum root mean squared error is achieved

[5] $p_n(n) = \frac{1}{\sqrt{\pi\sigma^2}} \mathrm{e}^{\frac{(n-\mu)^2}{-\sigma^2}}$; $p_n(n) = \frac{1}{\sqrt{2}\sigma} \mathrm{e}^{\frac{\sqrt{2}|n-\mu|}{-\sigma}}$; $p_n(n) = \frac{1}{2\sqrt{3}\sigma}$



(a) Alignment of acceleration sequences from smart-phone and video

(b) Acceleration sequence augmented with low noise level ($\mathcal{N}(\frac{\mu_v}{2}, \frac{\sigma_v^2}{4})$)

(c) Acceleration sequence augmented with video noise level ($\mathcal{N}(\mu_v, \sigma_v^2)$)

(d) Acceleration sequence augmented with high noise level ($\mathcal{N}(2 \cdot \mu_v, 4 \cdot \sigma_v^2)$)

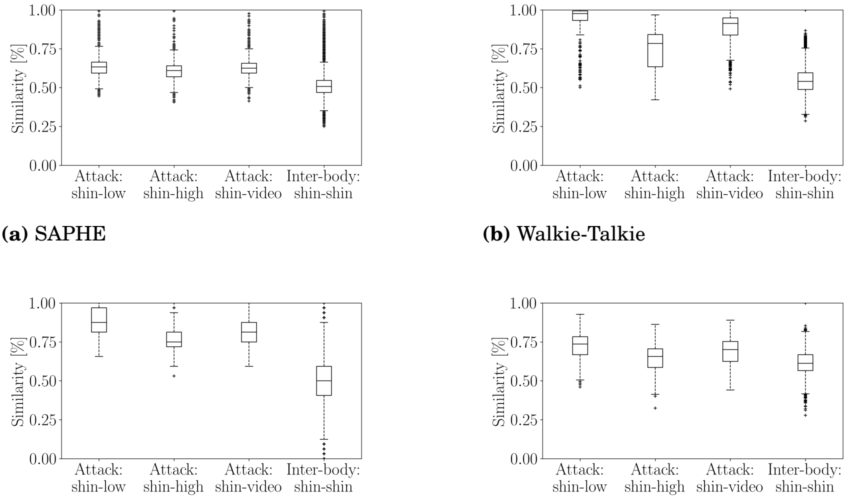
Figure 6.7. Acceleration signals featuring different noise levels

σ_v^2) (high noise) as illustrated in Figure 6.7 for Gaussian additive noise. Other noise models were treated similarly.

Figure 6.8 details the similarity for intra-body, inter-body, and video-based acceleration sequences with three noise levels. We assessed the effectiveness of video-based attacks on the four quantization schemes. We could use gait information reconstructed with videos to generate fingerprints which were sufficiently similar to the ones actually recorded in acceleration sequences. Hence, this video-based attack can break the gait-based pairing protocols for all three noise distributions considered. Walkie-Talkie [166] is the most vulnerable protocol under the video-based attacks. On the other hand, SAPHE [62] is the most secure protocol against video-based attacks (cf. Figure 6.8).

6.4.3 Evaluation of Heartbeat Fingerprints

In this section, we performed the experiment on a 14-subject dataset collected by Jähne-Raden *et al.* [75]. The subjects were reported to not have any cardio-vascular disease. They were students from the medical school. Their age range was from 21 to 34. Five of the subjects were female. Each subject was instructed to lie stably on a bed. Sensor arrays were attached to three body positions: chest, neck (right carotid), and wrist (left radial artery).



(a) SAPHE

(b) Walkie-Talkie

(c) BANDANA

(d) IPI

Figure 6.8. Attacks using Video-based impersonation: Similarity of gait-fingerprints with different noise levels over four pairing schemes

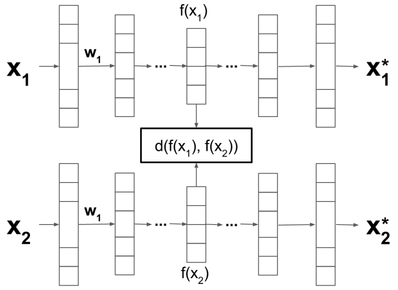


Figure 6.9. Siamese auto-encoder to extract fingerprints from heartbeat acceleration data

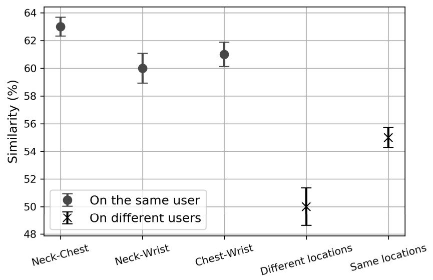


Figure 6.10. Similarity of heartbeat fingerprints on the same user and on different users

Using Keras [6] and Theano [7], we implemented a Siamese network with auto-encoders on heartbeat acceleration samples. In case of the first model, its branches are multilayer perceptrons (32 hidden units and rectified linear unit activation function). We split the pairwise data into training and testing set (75% and 25%, respectively). Other hyperparameters included: Euclidean distance as the distance function, RMSProp as the optimization algorithm, contrastive loss function, batch size 16, and 40 epochs. For the second model, we trained the model on wrist data and then evaluated it using neck data. Its hyperparameters are: 32 hidden units, the sigmoid activation function, the Adam optimization algorithm, the mean squared error loss function, batch size 128, and 1000 epochs.

[6] Keras: keras.io

[7] Theano: deeplearning.net/software/theano/

In the task of identifying whether two sensors are on the same human body or not, our Siamese network achieved the following results: precision of 0.74 (training data) and 0.61 (testing data) and recall of 0.81 (training data) and 0.67 (testing data). We used the trained auto-encoder to extract fingerprints from the heartbeat acceleration data on three body locations. Each fingerprint vector $\mathbf{f}$ was then transformed into a binary sequence $\mathbf{f}_b$. The transformation was: $\mathbf{f}_b(i) = 1$ if $\mathbf{f}(i) > \text{mean}(\mathbf{f})$ and $\mathbf{f}_b(i) = 0$ otherwise. This process could be performed independently for each wearable device. For evaluation, we computed the fingerprint similarity based on the Hamming distance between $\mathbf{f}$ on the same user (three locations) and on different users. Figure 6.10 displays the average similarity in both cases, along with standard errors. From the figure, the heartbeat fingerprints of the same user is more similar than on different users. It shows that an error correcting code technique can be applied to derive the secret key for secure on-body device pairing, for example Reed-Solomon code [120] as employed in [128].

6.5 Conclusion

In this chapter, we investigated two implicit information sources to implement natural and continuous pairing of on-body devices: human gait, and heartbeat data. We studied the use of human gait information to generate pairing keys for on-body device communication. We proposed a scheme that leveraged the deviation of instantaneous and temporary mean gait cycles to derive keys. We discussed the threat of a video-based attack on gait authentication and pairing. We concluded that a sophisticated attacker assisted with high-resolution video capture and real-time gait estimation is able to break the studied gait-based pairing approaches. While the gait-based approach is suitable for ambulatory activities with repetitive movements, our heartbeat-based method is applied in scenarios of resting postures. We presented a model based on Siamese networks and auto-encoders to learn a fingerprinting scheme in device pairing with heartbeat acceleration data. If two devices are on the same subject, the learned fingerprints promotes similarity of collected data. Otherwise, it aims to separate the data extracted from different users. With our two proposed methods, we covered device pairing in on-body settings during stationary and ambulatory activities, using implicit information extracted from contextual data. Hence, they provided solutions for establishing secure communication in the *many-to-many* relationship of devices.

7. Conclusions and Future Work

In this chapter, we summarize our contributions and discuss potential extensions for our proposed security mechanisms. This dissertation investigated methods to extract implicit information from sensor data. Then, based on it, we proposed novel security mechanisms applied at different device-to-user and inter-device relationships: *one-to-one*, *many-to-one*, *one-to-many*, and *many-to-many*. We, first, considered a scenario in which one smart device authenticates its owner. We introduced a log-in mechanism utilizing ever-changing image-based passwords. Our proposed approach personalized authentication challenges for each user and implicitly updated passwords during different log-in attempts (see Chapter 3). Second, a team of networked devices can collaborate to infer situations of an observed area without exchanging sensitive data. We leveraged the interference of wireless signals to hide sensor data while partially offloading computation to the communication channel. We designed an algorithm to train a classification model whose parameters were scattered across networked entities (see Chapter 4). Third, we proposed an audio-based method to establish a secure connection between a personal device and shared appliances in a new environment. Our approach analyzed users' vocal commands to detect appliance types and generate secret keys for device-to-device communication (see Chapter 5). Fourth, we broadened the configuration to more than one devices aiming to establish a secure connection with each other in on-body settings. Our approach processed sensor data to extract secret information for secure device pairing. The proposed methods unobtrusively generated and implicitly updated pairing keys over time (see Chapter 6).

7.1 Discussion

The emergence of smart devices has gradually improved our life in terms of functionality and comfort. From a scientific perspective, these devices become valuable instruments with capability of computation, communication, and data collection. Multi-modal data recorded by these devices gives opportunities to nurture a new research direction: utilizing implicit information in sensing data to enhance security mechanisms. Next, we discuss our contributions based on four types of devices: inertial measurement units, wearable cameras, microphones, and RF devices.

The inertial measurement units have been used in transport and construction, as well as in human sensing for sport and healthcare. Nowadays they are excessive in quantity after being integrated into such devices as smart-phones and smart-watches. Their popularity has made them good candidates to implement a security mechanism that securely connects themselves: motion-based device pairing. When multiple smart devices with built-in inertial sensors are carried by a user, they simultaneously capture two characteristics of the user: gait and heart-beat. We leverage those characteristics to generate shared secret keys: using gait during ambulatory activities and using heart-beat in resting postures.

Wearable cameras have been used to capture recreational activities from their wearers' point-of-view. The first-person-view videos could be analyzed for activity recognition and motif discovery. The motion signature extracted from these videos could be used to identify the users. We saw that the diverse details captured in these videos might benefit the generation of personalized and transient passwords. Hence, we proposed a method that generated image-based authentication challenges using users' memory on the chronological order of images. These challenges mitigated shoulder-surfing and smudge-based attacks.

VUIs appear more and more frequently as virtual assistants integrated in smart-phones and smart-homes. We introduced and evaluated a system that employed natural vocal commands to initiate the secure connection between two smart devices. In particular, we extracted audio fingerprints from both devices independently and used a fuzzy commitment scheme based on an error correcting code to generate a secret key.

All these aforementioned devices are capable of wireless communication, which may cause signal interference and consume much energy. Backscatter communication has emerged as a solution to reduce the energy consumption. We presented a custom stochastic gradient descent algorithm to train a classifier in an online manner across vertically-partitioned sensor data. Our technique encoded transmit values as burst sequences following Poisson distributions. Because multiple devices transmit data simultaneously (via backscattering), the technique becomes an implicit data perturbation method based on the binomial mechanism.

On the other hand, sensor data can expose users to obscure security threats. Especially, video and audio data have been considered to be critical attack vectors. The advance of camera technology has allowed adversaries to capture videos of users at high resolution and high speed. These videos can be used to reconstruct gait information or even heart-beat data. In this dissertation, we showed the effectiveness of video-based attacks on several gait-based device pairing protocols. The popularity of VUI-equipped smart devices has made them potential eavesdroppers. We investigated the awareness of users to these types of eavesdropping and showed that it was more effective than human eavesdroppers because the users are often overlook these devices.

This dissertation has shown that conventional data analysis techniques could be applied on sensor data to reveal implicit information, which in turn was leverage either to serve or attack users. Hence, when collecting and analysing data from one sensor or combining (i.e. fusion) data from multiple sensors to reveal implicit information, one should consider the impact in two aspects: new capabilities and potential vulnerabilities.

7.2 Always-fresh Authentication Challenges

Chapter 3 introduced the use of first-person-view videos to generate ever-changing graphical passwords for user authentication. We explained the process of selecting appropriate images to form authentication challenges in two different formats based on time frame. A combination of video analysis techniques including visual descriptor extraction, segmentation, and clustering selected suitable images to form passwords that conformed a chronological order. A prototype was implemented to evaluate our approach using web and mobile applications. We experimented with the system to quantify the security and usability of both password designs. Using our implementation, we were able to extract implicit information from video data: the chronological order of images. Based on that, we could form authentication challenges that varied over time for different users. Unlike using fixed passwords registered by users, our schemes created graphical authentication challenges from ever-changing videos. We released users from the burden of changing passwords regularly and reduced the success chance of shoulder-surfing attacks. We believe that our approach can be enhanced by designing alternative password formats. For example, picking multiple images at the same time can improve the security and usability of our schemes.

7.3 Collaborative Inference based on Implicit Data Aggregation

Chapter 4 presented our approach to allow a team of networked entities to train a logistic regression model in a collaborative manner. Instead of mitigating the signal interference we could exploit it to implement computation offloading. Based on that, a custom stochastic gradient descent could be realized to train a logistic regression model over vertically-partitioned data. In our setting, sensor data was collected separately in different sites. The networked entities aimed to collaborate in learning a classification model without exposing their sensed information. The model parameter was distributed across networked entities instead of being stored in a single central server. Our implementation facilitated secure sharing and aggregation of data via signal interference in a wireless communication channel. We used only minimal binary feedbacks to guide the optimization of model parameters. Hence, our approach can be deployed within energy-efficient sensor networks, such as those comprised of backscatter devices.

Our work can be extended in terms of hardware implementation that integrates energy harvesting and backscatter communication. It can also be deployed in a network that contains active and passive radio nodes, which increase both data rate and power efficiency.

7.4 Proximity-based Secure Communication using Vocal Commands

In Chapter 5, we introduced a system to allow connecting a personal device and shared appliances using vocal commands and ambient audio. Vocal commands which were widely used in human-computer interaction systems could be leveraged to securely select devices in an intuitive way. The process was initialized through a natural vocal command stating the device class. The proposed system supported both deployments with and without a central authority to manage all partner devices. We performed security evaluation in both scenarios when the adversaries were human and hardware. We observed that users could use directional verbal conversations to securely select devices to establish the connection. In the presence of eavesdroppers, the users tried to control their voice in the existence of eavesdroppers and they acted differently with human and hardware eavesdroppers.

7.5 Continuous Secure Device Pairing

Chapter 6 introduced the extraction of secret keys from data acquired with on-body sensors. Our approach supported continuous key update and automatic device disconnecting in device-to-device communication. When a user was moving, we leveraged gait information to continuously generate secret keys to pair devices on the same body. Our method could work with all sensor positions on the human body, including upper and lower parts. During resting postures, heartbeat information was utilized through a feature-learning model to form the keys. The capability of accelerometers to capture human heart motion in resting positions was a complement to gait-based device pairing protocols, which could only work when the users moved. We found that Siamese auto-encoders, which had been used for such applications as signature verification and face authentication, were useful for device pairing. In addition, we experimentally proved that the video-based reconstruction attack was effective to gait-based pairing protocols.

One potential extension of our approach is sensor fusion. Multiple sensing modalities can be combined to form more secure keys. We can also pro-actively select appropriate sensors based on the context to balance the trade off between security and usability.

7.6 Future Work

In this section, we offer possible extensions that can be developed from the contribution of this dissertation. The future work aims to improve our current systems not only in functionality but also in security and efficiency. For example, we can add recent advances in physical layer security to further protect the information hidden in burst sequences while still ensuring the aggregation of transmitted data. Energy harvesting from ambient sources such as light and RF signal is another component that can be attached to our systems to compensate the energy used for collecting and pre-processing the data.

In Chapter 4, the information is encoded into burst sequences instead of explicitly using encryption algorithms since we leveraged the interference to hide confidential data and enhance privacy. Physical layer security can be an approach to improve the confidentiality of our mechanism, such as injecting Gaussian noise data into the carrier signal to prevent eavesdroppers from extracting confidential data [122]. Then, we can analyze the information-theoretic perspective of our computation scheme through extending the formulation of secure transmission over wireless channels [19]. In future, we can leverage backscatter signals as an implicit information source to establish a secure connection between resource-constrained devices. We aim to generate secret keys for device-to-device communication (see Chapter 5 and 6) from multi-path propagation signatures of IoT [93] and wearable [94] devices.

Energy harvesting from ambient electromagnetic signals can support wireless sensor nodes [118] and on-body devices [170] for sensing, information dissemination, and data reception. Battery-free devices powered from Wi-Fi [111] and GSM [10] bands have also been implemented. Our next step is to integrate energy-harvesting components into our devices to realize long-term autonomous operations. We aim to implement the proposed security mechanisms on battery-free devices.

References

[1] Joint Technical Committee ISO/IEC JTC 1. ISO/IEC 18092:2013 Information technology - Telecommunications and information exchange between systems - Near Field Communication - Interface and Protocol (NFCIP-1). Standard, ISO/IEC JTC 1/SC 6 Telecommunications and information exchange between systems, 2013.

[2] H. J. Ailisto, M. Lindholm, J. Mantyjarvi, E. Vildjiounaite, and S.-M. Makela. Identifying people from gait pattern with accelerometers. In *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, 2005.

[3] Wi-Fi Alliance. Wi-Fi Direct industry white paper. 2010.

[4] ZigBee Alliance. ZigBee Specification. Specification, ZigBee Alliance Document, 2012.

[5] S. Almuairfi, P. Veeraraghavan, and N. Chilamkurti. A novel image-based implicit password authentication system (IPAS) for mobile and non-mobile devices. *Mathematical and Computer Modelling*, 58(1):108 – 116, 2013.

[6] M. A. Alsheikh, S. Lin, D. Niyato, and H. Tan. Machine learning in wireless sensor networks: Algorithms, strategies, and applications. *IEEE Communications Surveys Tutorials*, 16(4):1996–2018, 2014.

[7] F. Alt, S. Schneegass, A. Sahami Shirazi, M. Hassib, and A. Bulling. Graphical passwords in the wild: Understanding how users choose pictures and passwords in image-based authentication schemes. In *International Conference on Human-Computer Interaction with Mobile Devices and Services*, 2015.

[8] A. Alzubaidi and J. Kalita. Authentication of smartphone users using behavioral biometrics. *IEEE Communications Surveys Tutorials*, 18(3):1998–2026, 2016.

[9] M. Antikainen, M. Sethi, S. Matetic, and T. Aura. Commitment-based device-pairing protocol with synchronized drawings and comparison metrics. *Pervasive and Mobile Computing*, 16, 2015.

[10] M. Arrawatia, M. S. Baghini, and G. Kumar. RF energy harvesting system from cell towers in 900MHz band. In *IEEE National Conference on Communications*, 2011.

[11] IEEE Audio and Electroacoustics Group. IEEE Recommended practice for speech quality measurements. *IEEE Transactions on Audio and Electroacoustics*, 17(3), 1969.

[12] A. J. Aviv, K. Gibson, E. Mossop, M. Blaze, and J. M. Smith. Smudge attacks on smartphone touch screens. In *USENIX Conference on Offensive Technologies*, pages 1–7, 2010.

[13] B. Badihi, A. Liljemark, M. U. Sheikh, J. Lietzén, and R. Jäntti. Link budget validation for backscatter-radio system in sub-1GHz. In *IEEE Wireless Communications and Networking Conference*, 2019.

[14] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *Network and Distributed System Security Symposium*, 2002.

[15] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[16] A. Betancourt, P. Morerio, C.S. Regazzoni, and M. Rauterberg. The evolution of first person vision methods: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(5), 2015.

[17] R. Biddle, S. Chiasson, and P.C. Van Oorschot. Graphical passwords: Learning from the first twelve years. *ACM Computing Surveys*, 44(4):19:1–19:41, 2012.

[18] B. Biggio, G. Fumera, and F. Roli. Security evaluation of pattern classifiers under attack. *IEEE Transactions on Knowledge and Data Engineering*, 26(4):984–996, 2013.

[19] M. Bloch, J. Barros, M. R. D. Rodrigues, and S. W. McLaughlin. Wireless information-theoretic security. *IEEE Transactions on Information Theory*, 54(6):2515–2534, 2008.

[20] A. Bosch, A. Zisserman, and X. Munoz. Representing shape with a spatial pyramid kernel. In *ACM International Conference on Image and Video Retrieval*, pages 401–408, 2007.

[21] R. C. Bose and D. K. Ray-Chaudhuri. On a class of error correcting binary group codes. *Information and Control*, 3(1):68–79, 1960.

[22] L. Bottou, F. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[23] C. V. C. Bouten, K. T. M. Koekkoek, M. Verduin, R. Kodde, and J. D. Janssen. A triaxial accelerometer and portable data processing unit for the assessment of daily physical activity. *IEEE Transactions on Biomedical Engineering*, 44(3):136–147, 1997.

[24] J. Bromley, I Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a "siamese" time delay neural network. In *International Conference on Neural Information Processing Systems*, pages 737–744, San Francisco, CA, USA, 1993.

[25] R. G. Brown. DieHarder: A random number test suite. `http://www.phy.duke.edu/~rgb/General/dieharder.php`, 2011.

[26] A. Bulling, U. Blanke, and B. Schiele. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys*, 46(3), 2014.

[27] L. M. Candanedo and V. Feldheim. Accurate occupancy detection of an office room from light, temperature, humidity and CO_2 measurements using statistical learning models. *Energy and Buildings*, 2016.

[28] P. Cano, E. Batlle, T. Kalker, and J. Haitsma. A review of audio fingerprinting. *Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology*, 41(3):271–284, 2005.

[29] X. Cao, L. Liu, Y. Cheng, and X. S. Shen. Towards energy-efficient wireless networking in the big data era: A survey. *IEEE Communications Surveys & Tutorials*, 20(1):303–332, Firstquarter 2018.

[30] K. Chen and A. Salman. Extracting speaker-specific information with a regularized siamese deep network. In *International Conference on Neural Information Processing Systems*, 2011.

[31] Ming Ki Chong, Rene Mayrhofer, and Hans Gellersen. A survey of user interaction for spontaneous device association. *ACM Computing Surveys*, 47(1), 2014.

[32] N. L. Clarke and S. M. Furnell. Authentication of users on mobile telephones – a survey of attitudes and practices. *Computers & Security*, 24(7):519 – 527, 2005.

[33] IEEE 802 LAN/MAN Standards Committee. IEEE Standard for Information technology - Telecommunications and information exchange between systems Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Standard, IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012), 2016.

[34] C. Cornelius and D. Kotz. Recognizing whether sensors are on the same body. In *International Conference on Pervasive Computing*, 2011.

[35] National Research Council. *Who Goes There?: Authentication Through the Lens of Privacy*. The National Academies Press, Washington, DC, 2003.

[36] F. Crete, T. Dolmiere, P. Ladret, and M. Nicolas. The blur effect: perception and estimation with a new no-reference perceptual blur metric. In *Human Vision and Electronic Imaging XII*, 2007.

[37] James E. Cutting and Lynn T. Kozlowski. Recognizing friends by their walk: Gait perception without familiarity cues. *Bulletin of the Psychonomic Society*, 9(5):353–356, 1977.

[38] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, 2005.

[39] S. Das, E. Hayashi, and J. I. Hong. Exploring capturable everyday memory for autobiographical authentication. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2013.

[40] G. de Meulenaer, F. Gosset, F. Standaert, and O. Pereira. On the energy cost of communication and cryptography in wireless sensor networks. In *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2008.

[41] A Defeyter, M, R. Russo, and P. L. McPartlin. The picture superiority effect in recognition memory: A developmental study using the response signal procedure. *Cognitive Development*, 24(3), 2009.

[42] T. Denning, K. Bowers, M. van Dijk, and A. Juels. Exploring implicit memory for painless password recovery. In *CHI Conference on Human Factors in Computing Systems*, 2011.

[43] R. Dhamija and A. Perrig. Déjà vu: A user study using images for authentication. In *USENIX Security Symposium*, 2000.

[44] A. Domínguez. A history of the convolution operation [retrospectroscope]. *IEEE Pulse*, 6(1), 2015.

[45] M. F. Duarte and Y. H. Hu. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7), 2004.

[46] C. Dwork. Differential privacy. In *International Conference on Automata, Languages and Programming*, 2006.

[47] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 2006.

[48] C. Dwork and A. Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 2014.

[49] M. Eiband, M. Khamis, E. von Zezschwitz, H. Hussmann, and F. Alt. Understanding shoulder surfing in the wild: Stories from users and observers. In *CHI Conference on Human Factors in Computing Systems*, pages 4254–4265, New York, NY, USA, 2017.

[50] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1996.

[51] K. M. Everitt, T. Bragin, J. Fogarty, and T. Kohno. A comprehensive study of frequency, interference, and training of multiple graphical passwords. In *CHI Conference on Human Factors in Computing Systems*, 2009.

[52] W. Feller. *An Introduction to Probability Theory and its Applications*. Wiley, 1968.

[53] E. Ferrer and J. L. Helm. Dynamical systems modeling of physiological coregulation in dyadic interactions. *International Journal of Psychophysiology*, 88(3), 2013. Psychophysiology of Relationships.

[54] D. Figo, P. C. Diniz, D. R. Ferreira, and J. M. Cardoso. Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Computing*, 14(7), 2010.

[55] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer. Shakeunlock: Securely unlock mobile devices by shaking them together. In *International Conference on Advances in Mobile Computing and Multimedia*, 2014.

[56] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer. ShakeUnlock: Securely Transfer Authentication States Between Mobile Devices. *IEEE Transactions on Mobile Computing*, (99), 2016.

[57] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.

[58] Christian G. and Kaisa N. Manual authentication for wireless devices. *RSA Cryptobytes*, 7, 2004.

[59] M. Golla, D. Deterring, and M. Durmuth. Emojiauth: Quantifying the security of emoji-based authentication. In *Workshop on Usable Security*, 2017.

[60] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[61] Bluetooth Special Interest Group. Bluetooth Specification Version 5.0. Core specification, Bluetooth SIG, 2016.

[62] B. Groza and R. Mayrhofer. SAPHE: Simple accelerometer based wireless pairing with heuristic trees. In *International Conference on Advances in Mobile Computing & Multimedia*, 2012.

[63] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou. Opportunistic IoT: Exploring the harmonious interaction between human and the internet of things. *Journal of Network and Computer Applications*, 36(6), 2013.

[64] A. Hang, A. De Luca, and H. Hussmann. I know what you did last week! Do you? Dynamic security questions for fallback authentication on smartphones. In *CHI Conference on Human Factors in Computing Systems*, 2015.

[65] E. A. Heinz, K. S. Kunze, S. Sulistyo, H. Junker, P. Lukowicz, and G. Tröster. Experimental evaluation of variations in primary features used for accelerometric context recognition. In *European Symposium on Ambient Intelligence*, 2003.

[66] J. M. Henderson and A. Hollingworth. High-level scene perception. *Annual Review of Psychology*, 50(1), 1999.

[67] C. Herley, P. C. Van Oorschot, and A. S. Patrick. Passwords: If we're so smart, why are we still using them? In *International Conference on Financial Cryptography and Data Security*, 2009.

[68] J. Hernandez, D. J. McDuff, and R. W. Picard. Bioinsights: Extracting personal data from still wearable motion sensors. In *International Conference on Wearable and Implantable Body Sensor Networks*, 2015.

[69] M. Hessar, A. Najafi, and S. Gollakota. NetScatter: Enabling large-scale backscatter networks. In *USENIX Symposium on Networked Systems Design and Implementation*, 2019.

[70] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8), 1997.

[71] A. Hocquenghem. Codes correcteurs d'erreurs. *Chiffres*, 2(2), 1959.

[72] Y. Hoshen and S. Peleg. An egocentric look at video photographer identity. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[73] N. V. Huynh, D. T. Hoang, X. Lu, D. Niyato, P. Wang, and D. I. Kim. Ambient backscatter communications: A contemporary survey. *IEEE Communications Surveys & Tutorials*, 20(4), 2018.

[74] P. Isola, J. Xiao, A. Torralba, and A. Oliva. What makes an image memorable? In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[75] N. Jähne-Raden, T. Märtin, M. Marschollek, K. Heusser, and J. Tank. BCG-mapping of the thorax using different sensors: First experiences and signal quality. In *IEEE SENSORS*, 2016.

[76] M. Jakobsson, E. Shi, P. Golle, and R. Chow. Implicit authentication for mobile devices. In *USENIX Conference on Hot Topics in Security*, 2009.

[77] R. Jin, L. Shi, K. Zeng, A. Pande, and P. Mohapatra. MagPairing: Pairing smartphones in close proximity using magnetometers. *IEEE Transactions on Information Forensics and Security*, 11(6), 2016.

[78] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2), 1973.

[79] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *ACM Conference on Computer and Communications Security*, 1999.

[80] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[81] M. Kassner, W. Patera, and A. Bulling. Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, 2014.

[82] D. A. Knox and T. Kunz. Wireless fingerprints inside a wireless sensor network. *ACM Transactions on Sensor Networks*, 11(2), 2015.

[83] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems*, 2012.

[84] K. Kunze. *Compensating for on-body placement effects in activity recognition*. PhD thesis, Universität Passau, 2011.

[85] W. Lee, S. J. Stolfo, and K. W. Mok. Mining in a data-flow environment: Experience in network intrusion detection. In *SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999.

[86] W. K. Lee, H. Yoon, D. W. Jung, S. H. Hwang, and K. S. Park. Ballistocardiogram of baby during sleep. In *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2015.

[87] J. Lester, B. Hannaford, and G. Borriello. *"Are You with Me?"–Using accelerometers to determine if two devices are carried by the same person*. 2004.

[88] J. Li, K. Fawaz, and Y. Kim. Velody: Nonlinear vibration challenge-response for resilient user authentication. In *Conference on Computer and Communications Security*, 2019.

[89] S. Li, A. Ashok, C. Xu, Y. Zhang, J. Lindqvist, M. Gruteser, and N. Mandayam. Whose move is it anyway? Authenticating smart wearable devices using unique head movement patterns. In *IEEE Conference on Pervasive Computing and Communications*, 2016.

[90] F. Lin, C. Song, Y. Zhuang, W. Xu, C. Li, and K. Ren. Cardiac Scan: A non-contact and continuous heart-based user authentication system. In *International Conference on Mobile Computing and Networking*, 2017.

[91] V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. R. Smith. Ambient backscatter: Wireless communication out of thin air. *Conference of the ACM Special Interest Group on Data Communication on the Applications, Technologies, Architectures, and Protocols for Computer Communication*, 43(4), 2013.

[92] Z. Lu and K. Grauman. Story-driven summarization for egocentric video. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2714–2721, 2013.

[93] Z. Luo, W. Wang, J. Qu, T. Jiang, and Q. Zhang. ShieldScatter: Improving IoT security with backscatter assistance. In *ACM Conference on Embedded Networked Sensor Systems*, 2018.

[94] Z. Luo, W. Wang, J. Xiao, Q. Huang, T. jiang, and Q. Zhang. Authenticating on-body backscatter by exploiting propagation signatures. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2018.

[95] Y. X. M. Tan, A. Iacovazzi, I. Homoliak, Y. Elovici, and A. Binder. Adversarial attacks on remote user authentication using behavioural mouse dynamics. In *International Joint Conference on Neural Networks*, 2019.

[96] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan. Estimation of IMU and MARG orientation using a gradient descent algorithm. In *IEEE International Conference on Rehabilitation Robotics*, 2011.

[97] S. Mare, A. M. Markham, C. Cornelius, R. Peterson, and D. Kotz. ZEBRA: Zero-effort bilateral recurring authentication. In *IEEE Symposium on Security and Privacy*, 2014.

[98] D. Marques, T. Guerreiro, L. Carrico, and Konstantin Beznosov I. Beschastnikh. Vulnerability & blame: Making sense of unauthorized access to smartphones. In *CHI Conference on Human Factors in Computing Systems*, 2019.

[99] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi. A no-reference perceptual blur metric. In *International Conference on Image Processing*, 2002.

[100] R. Mayrhofer. The candidate key protocol for generating secret shared keys from similar sensor data streams. In *European Workshop on Security in Ad-hoc and Sensor Networks*, 2007.

[101] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H-P. Seidel, W. Xu, D. Casas, and C. Theobalt. VNect: Real-time 3D human pose estimation with a single RGB camera. *ACM Transactions on Graphics*, 36(4), 2017.

[102] W. Meng, D. S. Wong, S. Furnell, and J. Zhou. Surveying the development of biometric user authentication on mobile phones. *IEEE Communications Surveys & Tutorials*, 17(3), 2015.

[103] M. Miettinen, N. Asokan, T. D. Nguyen, A.-R. Sadeghi, and M. Sobhani. Context-based zero-interaction pairing and key evolution for advanced personal devices. In *ACM Conference on Computer and Communications Security*, 2014.

[104] T. M. Mitchell. *Machine learning*. McGraw Hill Series in Computer Science. McGraw-Hill, 1997.

[105] R. M. Mohammad, F. Thabtah, and L. McCluskey. An assessment of features related to phishing websites using an automated technique. In *International Conference for Internet Technology and Secured Transactions*, 2012.

[106] S. J. Morris. *A shoe-integrated sensor system for wireless gait analysis and real-time therapeutic feedback*. PhD thesis, Massachusetts Institute of Technology, 2004.

[107] M. Muaaz and R. Mayrhofer. An analysis of different approaches to gait recognition using cell phone based accelerometers. In *International Conference on Advances in Mobile Computing & Multimedia*, 2013.

[108] M. Muaaz and R. Mayrhofer. Smartphone-based gait recognition: From authentication to imitation. *IEEE Transactions on Mobile Computing*, 2017.

[109] M. Muaaz and R. Mayrhofer. Smartphone-based gait recognition: From authentication to imitation. *IEEE Transactions on Mobile Computing*, 2017.

[110] I. Muslukhov, Y. Boshmaf, C. Kuo, J. Lester, and K. Beznosov. Know your enemy: The risk of unauthorized access in smartphones by insiders. In *International Conference on Human-Computer Interaction with Mobile Devices and Services*, 2013.

[111] U. Olgun, C.-C. Chen, and J. L. Volakis. Design of an efficient ambient wifi energy harvesting system. *IET Microwaves, Antennas & Propagation*, 6(11), 2012.

[112] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3), 2001.

[113] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy*, 2016.

[114] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, 33(4):49–61, July 2016.

[115] B. A. Pearlmutter. Learning state space trajectories in recurrent neural networks. *Neural Computation*, 1(2), 1989.

[116] Y. Peng, L. Shangguan, Y. Hu, Y. Qian, X. Lin, X. Chen, D. Fang, and K. Jamieson. PLoRa: A passive long-range data network from ambient LoRa transmissions. In *Conference of the ACM Special Interest Group on Data Communication*, 2018.

[117] T. Plötz, N. Y. Hammerla, and P. Olivier. Feature learning for activity recognition in ubiquitous computing. In *International Joint Conference on Artificial Intelligence*, 2011.

[118] Z. Popović, E. A. Falkenstein, D. Costinett, and R. Zane. Low-power far-field wireless powering for wireless sensors. *Proceedings of the IEEE*, 101(6), 2013.

[119] M. Porcheron, J. E. Fischer, S. Reeves, and S. Sharples. Voice interfaces in everyday life. In *CHI Conference on Human Factors in Computing Systems*, 2018.

[120] I. S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2), 1960.

[121] G. Revadigar, C. Javali, W. Xu, A. V. Vasilakos, W. Hu, and S. Jha. Accelerometer and fuzzy vault-based secure group key generation and sharing protocol for smart wearables. *IEEE Transactions on Information Forensics and Security*, 12(10), 2017.

[122] W. Saad, X. Zhou, Z. Han, and H. V. Poor. On the physical layer security of backscatter wireless systems. *IEEE Transactions on Wireless Communications*, 13(6):3442–3451, June 2014.

[123] R. Salloum and C. . J. Kuo. ECG-based biometrics using recurrent neural networks. In *International Conference on Acoustics, Speech and Signal Processing*, 2017.

[124] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer. The humanID gait challenge problem: data sets, performance, and analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2), 2005.

[125] J. Schmidt. Requirements for password-authenticated key agreement (PAKE) schemes. Technical report, Internet Research Task Force, 2017.

[126] M. Scholz and S. Sigg. SenseWaves: Radiowaves for context recognition. In *International Conference on Pervasive Computing*, 2011.

[127] D. Schürmann, A. Brüsch, N. Nguyen, S. Sigg, and L. Wolf. Moves like Jagger: Exploiting variations in instantaneous gait for spontaneous device pairing. *Pervasive and Mobile Computing*, 47, 2018.

[128] D. Schürmann, A. Brüsch, S. Sigg, and L. Wolf. BANDANA – Body Area Network Device-to-device Authentication using Natural gAit. In *International Conference on Pervasive Computing and Communications*, 2017.

[129] D. Schürmann and S. Sigg. Secure communication based on ambient audio. *IEEE Transactions on Mobile Computing*, 12(2), 2013.

[130] M. Sethi, M. Antikainen, and T. Aura. Commitment-based device pairing with synchronized drawing. In *International Conference on Pervasive Computing and Communications*, 2014.

[131] M. Sethi, A. Peltonen, and T. Aura. Misbinding attacks on secure device pairing and bootstrapping. In *Asia Conference on Computer and Communications Security*, 2019.

[132] Y. Shaked and A. Wool. Cracking the Bluetooth PIN. In *International Conference on Mobile Systems, Applications, and Services*, 2005.

[133] J. H. Shin, B. H. Choi, Y. G. Lim, D. U. Jeong, and K. S. Park. Automatic ballistocardiogram (BCG) beat detection using a template matching approach. In *International Conference of the IEEE Engineering in Medicine and Biology Society*, 2008.

[134] S. Sigg, P. Jakimovski, and M. Beigl. Calculation of functions on the RF-channel for IoT. In *Conference on the Internet of Things*, 2012.

[135] S. Sigg, P. Jakimovski, Y. Ji, and M. Beigl. Utilising an algebra of random functions to realise function calculation via a physical channel. In *International Workshop on Signal Processing Advances in Wireless Communications*, 2013.

[136] S. Sigg, M. Scholz, S. Shi, Y. Ji, and M. Beigl. RF-sensing of activities from non-cooperative subjects in device-free recognition systems using ambient and local signals. *IEEE Transactions on Mobile Computing*, 13(4), 2014.

[137] S. W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing, San Diego, CA, USA, 1997.

[138] C. Song, T. Ristenpart, and V. Shmatikov. Machine learning models that remember too much. In *Conference on Computer and Communications Security*, 2017.

[139] C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 1904.

[140] M. D. Springer. *The algebra of random variables*. Wiley Series in Probability and Mathematical Statistics. Wiley, 1979.

[141] A. Srivastava, J. Gummeson, M. Baker, and K.-H. Kim. Step-by-step detection of personally collocated mobile devices. In *International Workshop on Mobile Computing Systems and Applications*, 2015.

[142] F. Stajano and R. Anderson. The resurrecting duckling: Security issues for ad-hoc wireless networks. In *Security Protocols*, 2000.

[143] I. Starr, A. J. Rawson, H. A. Schroeder, , and N. R. Joseph. Studies on the estimation of cardiac output in man, and of abnormalities in cardiac function, from the hearts recoil and the bloods impacts; the ballistocardiogram. *The American Journal of Physiology*, 127(1), 1939.

[144] H. Stockman. Communication by means of reflected power. *Proceedings of the IRE*, 36(10), 1948.

[145] A. Suliman, C. Carlson, C. J. Ade, S. Warren, and D. E. Thompson. Performance comparison for ballistocardiogram peak detection methods. *IEEE Access*, 7, 2019.

[146] C. Sun, Y. Wang, and J. Zheng. Dissecting pattern unlock: The effect of pattern strength meter on pattern selection. *Journal of Information Security and Applications*, 19(4), 2014.

[147] H. Sun, K. Wang, X. Li, N. Qin, and Z. Chen. PassApp: My app is my password! In *International Conference on Human-Computer Interaction with Mobile Devices and Services*, 2015.

[148] L. Sun, Y. Wang, B. Cao, S. Y. Philip, W. Srisa-An, and A. D. Leow. Sequential keystroke behavioral biometrics for mobile user identification via multi-view deep learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2017.

[149] Y. Sun, C. Wong, G. Yang, and B. Lo. Secure key generation using gait features for body sensor networks. In *International Conference on Wearable and Implantable Body Sensor Networks*, 2017.

[150] T. Sztyler and H. Stuckenschmidt. On-body localization of wearable devices: An investigation of position-aware activity recognition. In *International Conference on Pervasive Computing and Communications*, 2016.

[151] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J. R. Smith, and S. Gollakota. LoRa Backscatter: Enabling the vision of ubiquitous connectivity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3), 2017.

[152] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart. Stealing machine learning models via prediction APIs. In *USENIX Conference on Security Symposium*, 2016.

[153] H. T. T. Truong, X. Gao, B. Shrestha, N. Saxena, N. Asokan, and P. Nurmi. Using contextual co-presence to strengthen zero-interaction authentication: Design, integration and usability. *Pervasive and Mobile Computing*, 16, 2015.

[154] A. Varshavsky, A. Scannell, A. LaMarca, and E. De Lara. *Amigo: Proximity-Based Authentication of Mobile Devices*. 2007.

[155] A. Varshney, C. Pérez-Penichet, C. Rohner, and T. Voigt. LoRea: A backscatter architecture that achieves a long communication range. In *Conference on Embedded Network Sensor Systems*, 2017.

[156] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *International Conference on Machine Learning*, 2008.

[157] E. Vural, S. Simske, and S. Schuckers. Verification of individuals from accelerometer measures of cardiac chest movements. In *International Conference of the Biometrics Special Interest Group*, 2013.

[158] C. Wan, L. Wang, and V. V. Phoha. A survey on gait recognition. *ACM Computing Surveys*, 51(5), 2018.

[159] W. Wang, A. X. Liu, and M. Shahzad. Gait recognition using wifi signals. In *International Joint Conference on Pervasive and Ubiquitous Computing*, 2016.

[160] Y. Wang and K. N. Plataniotis. Fuzzy vault for face based cryptographic key generation. In *Biometrics Symposium*, 2007.

[161] Y. Wang and C. Yang. 3S-cart: A lightweight, interactive sensor-based cart for smart shopping in supermarkets. *IEEE Sensors Journal*, 16(17), 2016.

[162] A. Weil. L'intégration dans les groupes topologiques et ses applications. *Hermann et Cie*, 1940.

[163] H. M. Wood. *The use of passwords for controlled access to computer resources*. National Bureau of Standards Special Publication 500-9 U.S Department of Commerce/NBS, 1977.

[164] J. Wu and J. M. Rehg. CENTRIST: a visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8), 2011.

[165] W. Xu, C. Javali, G. Revadigar, C. Luo, N. Bergmann, and W. Hu. Gait-Key: a gait-based shared secret key generation protocol for wearable devices. *ACM Transactions on Sensor Networks*, 13(1), 2017.

[166] W. Xu, G. Revadigar, C. Luo, N. Bergmann, and W. Hu. Walkie-Talkie: Motion-assisted automatic key generation for secure on-body device communication. In *International Conference on Information Processing in Sensor Networks*, 2016.

[167] Z. Yang, Q. Huang, and Q. Zhang. NICScatter: Backscatter as a covert channel in mobile devices. In *International Conference on Mobile Computing and Networking*, 2017.

[168] B. Ying, K. Yuan, and A. H. Sayed. Supervised learning under distributed features. *IEEE Transactions on Signal Processing*, 67(4), 2019.

[169] R. Yonetani, K. M. Kitani, and Y. Sato. Visual motif discovery via first-person vision. In *European Conference on Computer Vision*, 2016.

[170] F. Zhang, Y. Zhang, J. Silver, Y. Shakhsheer, M. Nagaraju, A. Klinefelter, J. Pandey, J. Boley, E. Carlson, A. Shrivastava, B. Otis, and B. Calhoun. A batteryless 19μW MICS/ISM-band energy harvesting body area sensor node SoC. In *IEEE Solid-State Circuits Conference*, 2012.

[171] P. Zhang, M. Rostami, P. Hu, and D. Ganesan. Enabling practical backscatter communication for on-body sensors. In *Conference of the ACM Special Interest Group on Data Communication*, 2016.

[172] R. Zhao, F. Zhu, Y. Feng, S. Peng, X. Tian, H. Yu, and X. Wang. OFDMA-enabled Wi-Fi backscatter. In *International Conference on Mobile Computing and Networking*, 2019.

[173] V. Zue, S. Seneff, and J. Glass. Speech database development at MIT: TIMIT and beyond. *Speech Communication*, 9(4), 1990.

Aalto-DD 128/2020

ISBN 978-952-64-0012-9 (printed)
ISBN 978-952-64-0013-6 (pdf)
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)

Aalto University
School of Electrical Engineering
Department of Communications and Networking
www.aalto.fi