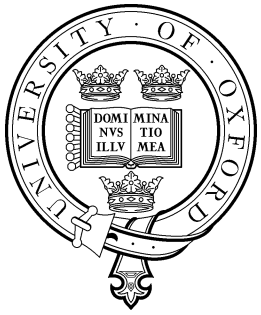
Robustly Inferring Identity across Digital and Physical Worlds



Xiaoxuan Lu

Keble College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2018

Acknowledgements

When I sent an email to Niki and Andrew seeking a PhD position, I did not imagine the wonderful 4 years I would spend with them in Oxford. I thank them more than anyone else. They have influenced my outlook on research, and made me understand the importance of high-quality research with wider impact. Most of all, I value their candid and honest opinions, their calmness and clarity of advice amidst difficult times, and their patience and understanding over the past years. Niki and Andrew provide an environment where researchers can thrive and where the degree to which you succeed is ultimately in your own hands. At the same time, they gave me resources, guidance and support whenever needed during the period that led up to this thesis. I shall always remain indebted to them and hope that I can do the same thing for my students and inspire them as they have done for me.

I am likewise massively indebted to Hongkai Wen, without whom much of this research would not have been possible. He invariably provided fascinating insights in directing my research to solve real-world problems. This is my privilege to have him as the "shadow advisor" and gave me a lot of help in my PhD research.

I feel fortunate to have had the opportunity to collaborate with other faculty members. Jack Stankovic taught me how to frame novel research ideas. Ivan Martinovic's enthusiasm in solving real-world privacy problems is absolutely contagious, and I thoroughly enjoyed our discussions on bridging IoT and privacy problems. I also learned much from Rui Tan through working closely with him, and gained many experiences in coping with measurement studies.

Much credit for the research in this thesis goes to my phenomenal student collaborators. Xuan, Bowen and Amber worked incessantly with me on developing and deploying our systems in different countries with numerous participants. Peijun and Yang stayed up with me many sleepless nights in the lab as we worked on "crazy ideas". Changhao day-to-day shared with me the

state-of-the-art machine learning algorithms and was a reliable collaborator as well.

I could not have wished for a more supportive and fun group of colleagues than my labmates at the Cyber Physical Systems group: Linhai, Shuyu, Zhihua, Bo, Wei, Stefano, Pedro, Savvas, Ronnie, Risqi, Johan, Javier, Milad, Catherine and Joe. I am indebted to them for unwaveringly volunteering in hundreds upon hundreds of experiments and for providing me with feedback on numerous iterations of my papers and talks.

Finally, yet most importantly, words cannot express my gratitude towards my parents, for their endless support. They have always stood by me amidst difficult times. I also feel fortunate to meet my girl friend Dongge in Oxford, who made my PhD life an enjoyable journey.

Abstract

A long-term vision within the realm of ubiquitous computing is the creation of smart, digital environments that provide seamless human-computer interaction, allowing computation to recede into the background of everyday life. Key to realizing this vision is the ability for machines to recognize people, so that spaces can become truly personalized. However, the unpredictability of real-world environments impacts robust recognition, limiting usability. In real conditions, human identification systems have to handle issues such as out-of-set subjects and domain deviations, where conventional supervised learning approaches for training and inference are poorly suited. The inability of supervised methods to cope with this inherent diversity could be overcome if equivalently diverse training data were readily available. Unfortunately, obtaining such comprehensive training datasets would incur huge enrolment effort and would be costly to stage.

With the rapid development of Internet of Things (IoT), we advocate a new labelling method in this thesis that exploits signals of opportunity hidden in heterogeneous IoT data. The key insight is that *one sensor modality can leverage the signals measured by other co-located sensor modalities to improve its own labelling performance*. If identity associations between heterogeneous sensor data can be discovered, it is possible to automatically label data, leading to more robust human recognition, without manual labelling or enrolment. We believe that many currently unsolved identification problems could be addressed through our advocated concept.

Specifically, this thesis demonstrates that leveraging the signals of opportunity in physical and digital observations of subjects can overcome many obstacles surrounding robust human identification, and we comprehensively tackle this in a number of research threads. Firstly, we propose `SCAN`, a general algorithm for cross-modality association, designed to automatically label biometric data sensed in the wild. Secondly, in order to mitigate the errors in the automatically labelled data, we further present `AutoTune`, a generic framework that iteratively adapts the biometric model and updates sensor observations. Lastly, we

comprehensively investigate the privacy implication of our advocated concept, with an application on smartwatch password inference and countermeasures. We demonstrate `Snoopy`, a password inference framework which is able to accurately intercept passwords entered on the touchscreens of smartwatches of out-of-set victims, just by eavesdropping on motion sensors. To mitigate this attack, we propose a countermeasure `DeepAuth`, which is a second-factor authentication system on smartwatches based on behavioural signatures. We prove that the co-located secondary sensor not only can be maliciously used as a leakage channel, but can be effectively employed as a defence channel as well. All the proposed approaches are comprehensively evaluated through large-scale experiments and the results demonstrate their potential impact in a broad spectrum of identification scenarios.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

A long-term vision within the realm of ubiquitous computing is the creation of smart, digital environments that provide seamless interaction, allowing computation to recede into the background of everyday life. The key to realizing this vision is the ability for machines to recognize people, so that spaces can become truly personalized. From a macro perspective, knowledge of person's identity could enable personalized heating and cooling, entertainment, task assistance, and behavioural analysis in smart spaces, such as intelligent buildings and homes [5]. From a micro view, knowing a user's identity could help manage privacy and security on smart devices, such as smartphones and smartwatches [6]. Generally, identifying a user can be categorized into the following two types:

- What you know: **digital attributes**, such as a username, Personal Identification Number (PIN), Social Security number (SSN) and International Mobile Equipment Identity (IMEI) of personal devices.
- Who you are: **physical attributes**, including strong and weak biometrics, such as face, voice, gait and body shape.

A digital attribute is essentially a discrete piece of digital information that can possibly be altered by users. A physical attribute, on the other hand, is derived from behavioural and biological features that cannot be easily modified. Typically, an enrolment or registration step is required to map a certain attribute to a person. For example, signing up for Amazon prime requires the user to create an ID and Password before usage. Similarly, using facial or vocal recognition in workspaces requires registering one's face or voice information.

Using digital attributes for person identification is straightforward and simple as these attributes are static once registered. Nevertheless, it is significantly more complicated to

use physical attributes for identification because of their intrinsic complexity. For example, human voice is a popular biometric, but is sensitive to the specific content of the speech. To account for the complexity in the observations of physical attributes, a supervised statistical model is needed. A robust supervised model that is able to generalize to different dynamics in the wild usually requires a sufficiently diverse training dataset, which is costly to obtain. Although we are witnessing a revolution in machine learning, existing alternative methods, e.g., semi-supervised or unsupervised techniques, are still ill equipped in terms of identity inference. Thus, the availability of training data will remain pivotal for human identification.

The problem explored in this thesis is *how to infer links between physical and digital attributes and leverage this information to robustly infer identities in the wild with minimal human effort*. In particular, we comprehensively study both the advantages and risks of being able to infer identity in spite of the limited training label information.

1.2 Challenges

We have identified the following main challenges in addressing the above problem of real-world human identification:

- **Unpredictable environments in the wild:** - Due to the unpredictability of the real-world environments, many human identification systems have to cope with data that greatly deviate from the samples in the enrolment data. These deviations are mainly caused by two factors: (a) out-of-set labels; (b) out-of-domain observations. With respect to the former, existing recognition models for human identification are supervised and fail to generalize to extreme deviations. For instance, many recognition modules in biometric systems extract biometric features from the acquired data and compare the features with the templates in the enrolment database to infer identities. However, when operating in the real world, these systems face the issue of out-of-set labels, e.g., subjects outside labels in the training set. For example, a face recognition system may encounter strangers outside the user group in the registration database. Secondly, even for the same subject, the observations of her biometrics are altered by domain changes. For example, the voice of a registered user may deviate, if the content in her speech is different or a low-quality microphone is used. Similarly, side profiles of a user cannot be easily recognised by a system trained only with front

facial images. Failures of identification systems often happen when these deviations are significant.

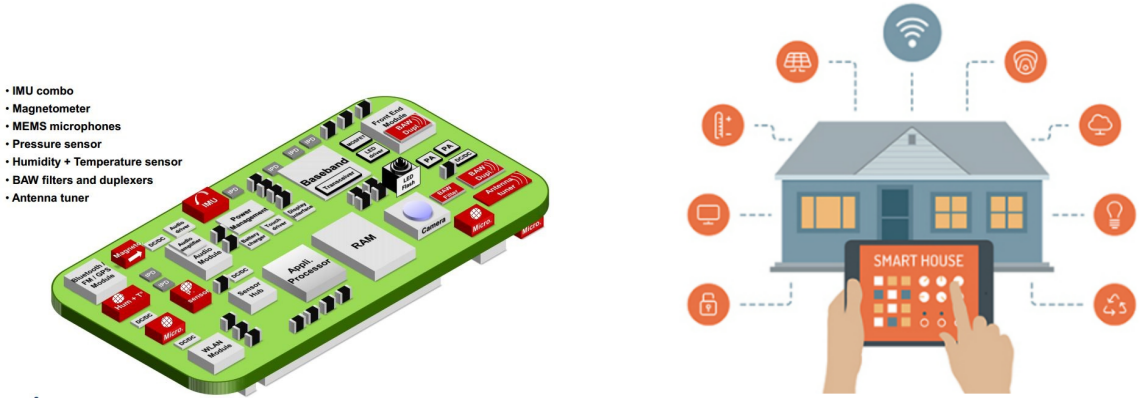
- **Huge enrolment effort:** - The inability of supervised recognition models to cope with diversity can be overcome if additional training data could be readily collected in the database to provide samples in various conditions. Unfortunately, acquiring such a comprehensive training dataset is difficult due to several reasons. First, asking a large number of users to register sufficient data would incur a huge enrolment effort and would be costly to stage. Second, conventional crowd-sourcing tools [7], e.g., Amazon Mechanical Turk, are not applicable in this context. These crowdsourcing tools generally assign data-to-labels on public platforms which would result in serious privacy concerns as many identity related data are sensitive. As such, there is an urgent need for a framework that automatically and securely labels identity data.

- **Uncertain sensor observations:** - In many realistic scenarios, sensor observations are *noisy* and they may not necessarily reliably reveal their uncertainty [8]. Identity sensing is of course no exception. In fact, both digital and physical attributes suffer from uncertain observations. Such uncertainty comes from various sources and is impossible to prevent in the wild. For example, poor deployment of surveillance cameras may result in limited field of view and would miscapture faces. People sometimes forget to carry their smartphone and therefore the sniffed smartphone MAC addresses may not necessarily reflect user presence. Notably, observation uncertainty here is distinct from domain alternation. It points to the fact that sensor observations, i.e., inputs/labels of the recognition model, can be inconsistent with the true status of a person. And in these cases, even if an identity recognition model operates perfectly, its predictions are error-prone as the input itself is corrupted.

- **Privacy concerns:** - Our study relates to various types of sensitive data. It hence requires an investigation on whether the concept or techniques proposed in this thesis may incur privacy breaches when they are used maliciously. If there are privacy implications, timely and effective countermeasures are needed for early prevention.

1.3 Opportunities

Despite the above challenges, we also have great opportunities brought by the advent of the Internet of Things (IoT). Our physical world is now richly and invisibly interwoven with



(a) Smart Wearable (b) Smart Home

Figure 1.1: Opportunities: A number of sensors co-exist in the same device or environment. The correlation between them can be utilized to replace human effort for data labelling.

sensors, actuators, displays and computational elements, embedded seamlessly in the everyday objects of our lives [9]. It has become common to have many heterogeneous sensors co-existing in the same environment or device. For example, the conventional desktop machine is now transforming into smart buildings [10], instrumented with various sensors to observe light levels, temperature, sound and vibration. Beyond the typical requirement of simply being a tool for vocal communication, traditional 2G-feature phones have evolved into smartphones, equipped with sensors that interact and monitor the outside world, such as accelerometers, gyroscopes, proximity sensors and capacitive touchscreens. Fig. 1.1 shows some common sensors in these environments.

As a result, we have massive volumes of information collected by these sensors in the real world, including both digital and physical attributes for human identification. *From the perspective of a single sensor modality, the information captured by other co-located sensor modalities are signals of opportunity, which complement its sensing view and knowledge base.* For example, utilizing device MAC addresses observed by a WiFi sniffer to autonomously develop a speaker recognition system for a co-located microphone in the smart building. Another example, this time adversarial, is a cyber attack in which a co-located motion sensor is used to automatically infer the keystroked PIN on a smartwatch's touchscreen. Although the above two examples are in different physical scenarios, their common theme is that one sensor modality leverages the signals collected by the other co-located sensor modality to improve its own capability of identity inference. Fig. 1.2 illustrates examples of such signals of opportunity.

The underpinning principle of this thesis is that heterogeneous **co-located** sensor modalities sometimes provide sufficient cues to build up a knowledge base for identity inference.

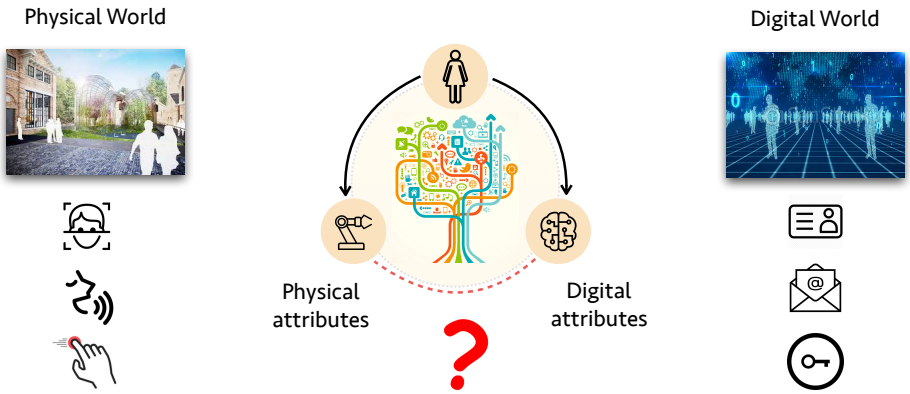


Figure 1.2: Physical and digital attributes of a person. Observations of the person by co-located sensors are naturally *signals of opportunity* for one anther.

More importantly, we advocate that such ambient cues can be exploited autonomously with minimal human intervention.

1.4 Contributions

This thesis focuses on robustly inferring identity in the wild with little human effort. To this end, it presents a class of novel approaches for identity inference and evaluates them in a variety of scenarios. Concretely, the technical contributions of this thesis are as follows:

1. Person identification is the key enabler of context-aware applications. Developing an identification system that robustly operates in the wild usually requires a comprehensive training set. Unfortunately, such labelled datasets are costly to acquire and require considerable human effort. This thesis proposes the idea of identifying signals of opportunity hidden in heterogeneous sensor data and leveraging them to robustly infer identity and automatically label biometric samples. Our work piggybacks rapid development and increasing adoption of IoT technologies, and we believe the proposed approaches can empower a number of areas, including machine learning, ubiquitous computing, security and privacy.

2. In order to autonomously develop the training set for identification, we propose using physical and digital attributes observed by heterogeneous sensors co-located in the same environment. The key idea is that digital attributes are generally static and can serve as an identity proxy for biometric observations. Labelling biometric observations in this way requires finding the fine-grained association between

physical and digital observations, which is far from trivial because the relationship between heterogeneous data is very unstructured. We therefore propose SCAN, a general framework for cross-modality association, allowing us to automatically discover fine-grained associations between heterogeneous data. By simultaneously cluster biometric observations and associate them to their corresponding digital attributes, we can significantly improve the robustness of the labelling process.

3. Due to the inevitable uncertainty in sensing systems, observations of biometrics and digital identities may be inconsistent. As a result, the training set created by cross-modality association is noisy. We therefore propose AutoTune, a cross-modality learning framework which automatically mitigates this inconsistency. Inspired by the expectation-maximization algorithm, AutoTune iteratively adapts the biometric representation model and updates the observations of digital attributes, until the two kinds of observations become sufficiently consistent. In particular, a probabilistic formulation is proposed to tolerate label uncertainty and enable iterative learning. Conventional loss functions that guide the learning of the biometric model are also tailored to operate with the new probabilistic labels. Extensive real-world experiments on different biometrics and in two different countries show that AutoTune is generic and can reliably adapt a biometric recognition model to a new environment with new subjects.

4. We conducted a comprehensive study in order to investigate the privacy implications of our proposed approach to cross-modality learning. This time, we move our physical context from smart environments to smartwatches. The rapid adoption of smartwatches makes them more attractive and vulnerable to malicious attacks, which to date have been largely overlooked. Based on the concept of cross-modality inference, we demonstrate Snoopy, a password inference framework which is able to accurately intercept passwords entered on the touchscreens of smartwatches of out-of-set victims, just by eavesdropping on motion sensors. A deep sequence learning method is adopted in Snoopy that is able to infer PINs and Android pattern locks outside the training set. To mitigate this attack, we further proposed a countermeasure named, DeepAuth, which is an authentication system on smartwatches based on behaviour signatures. The insight behind DeepAuth is that the motion sensor not only can maliciously act as a leakage channel, but it can be also effectively employed as a defence channel. Importantly, such motion behaviour is always available when a user keystrokes passwords, and therefore gives rise to a free and simultaneous two-factor authentication mechanism. Extensive experimental results on more than 300

participants showed that `Snoopy` raises significant risk concerns in practice, which can be effectively mitigated by `DeepAuth`.

1.5 Publications

The main contributions of this thesis have already been published at the following international conferences:

1. **Xiaoxuan Lu**, Hongkai Wen, Sen Wang, Andrew Markham and Niki Trigoni. "SCAN: Learning Speaker Identity From Noisy Sets of Sensor Data" In *International Conference on Information Processing in Sensor Networks (IPSN)*, 2017,
   This paper proposes a generic cross-modality association technique to automatically label biometric data in the wild with digital attributes. I am the first author and main contributor to the ideas and experiments in this paper. The work in this paper is presented in Chapter 3 of the thesis.

2. **Xiaoxuan Lu**, Hongkai Wen, Han Zou, Hao Jiang, Lihua Xie and Niki Trigoni. "Robust Occupancy Inference with Commodity WiFi." In *International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2016.
   This paper develops a channel-hopping WiFi sniffing system to collect sensor data. I am the first author and main contributor to the ideas and experiments in this paper. Part of this paper is presented in Chapter 3 of the thesis.

3. **Xiaoxuan Lu**, Xuan Kan, Bowen Du, Changhao Chen, Hongkai Wen, Andrew Markham, Niki Trigoni and John A. Stankovic. "Autonomous Learning for Face Recognition in the Wild via Ambient Wireless Cues." In *The Web Conference (WWW)*, 2019.
   This paper proposes an iterative adaptation framework to improve cross-modality association. I am the first author and main contributor to the ideas and experiments in this paper. Part of this paper is presented in Chapter 4 of the thesis.

4. **Xiaoxuan Lu**, Bowen Du, Hongkai Wen, Sen Wang, Andrew Markham, Ivan Marinovic, Yiran Shen and Niki Trigoni. "Snoopy: Sniffing Your Smartwatch Passwords via Deep Sequence Learning." In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, 2018.
   This paper demonstrates an attack framework that is able to robustly intercept passwords keystroked on smartwatches, via the co-located motion sensors. I am the first

author and main contributor to the ideas and experiments in this paper. The work of this paper is presented in Chapter 5 of the thesis.

5. **Xiaoxuan Lu**, Bowen Du, Peijun Zhao, Hongkai Wen, Yiran Shen and Niki Trigoni. "DeepAuth: In-situ Authentication for Smartwatches via Deeply Learned Behavioural Biometrics." In *International Symposium on Wearable Computers (ISWC)*, 2018.
   This paper presents a novel behaviour based authentication system, which is a countermeasure to mitigate the above side-channel attack. I am the first author and main contributor to the ideas and experiments in this paper. The work of this paper is presented in Chapter 5 of the thesis.

The work in this thesis also contributes to the following published papers:

1. **Xiaoxuan Lu**, Yang Li, Peijun Zhao, Changhao Chen, Linhai Xie, Hongkai Wen, Rui Tan and Niki Trigoni. "Simultaneous Localization and Mapping with Power Network Electromagnetic Field." In *ACM International Conference on Mobile Computing and Networking (MobiCom)*, 2018.
   This paper is motivated by the advocated concept in this thesis, i.e., signals of opportunity, and proposes a new multi-modal SLAM systems. A new localization channel of powerline electromagnetic radiation is identified and this paper opportunistically uses this ambient signal to address the drifting issue in inertial odometry.

2. Stefano Rosa, Andrea Patanè, **Xiaoxuan Lu**, Niki Trigoni, "Semantic Place Understanding for Human-Robot Cooperation - Towards Intelligent Workplaces", In *IEEE Transactions on Human-Machine Systems*, 2018.
   Similarly, this paper is motivated by the advocated concept of signals of opportunity and leverages a pair of robot and mobile user to opportunistically improve their semantic understanding in their co-located environments.

3. Changhao Chen, **Xiaoxuan Lu**, Andrew Markham and Niki Trigoni. "IONet: Learning to Cure the Curse of Drift in Inertial Odometry." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.
   This paper presents a deep sequence learning based inertial odometry, which significantly outperforms traditional methods. The deep sequence learning method used in this paper is modified from the proposed `Snoopy` system in this thesis.

4. Changhao Chen, Yishu Miao, **Xiaoxuan Lu**, Linhai Xie, Phil Blunsom, Andrew Markham and Niki Trigoni. "MotionTransformer: Transferring Neural Inertial Tracking Between Domains." In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

This paper presents a novel framework that extracts domain-invariant features of raw sequences from arbitrary domains, and transforms to new domains without any paired data. The paper shares the same motivation of reducing human effort of re-collecting labelled data in a new environment.

Other publications during the DPhil study are listed as follows:

1. Changhao Chen, Stefano Rosa, Yishu Miao, **Xiaoxuan Lu**, Wei Wu, Andrew Markham and Niki Trigoni. "Selective Sensor Fusion for Neural Visual Inertial Odometry". In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

2. Hongkai Wen, Ronald Clark, Sen Wang, **Xiaoxuan Lu**, Bowen Du, Wen Hu and Niki Trigoni. "Efficient Indoor Positioning with Visual Experiences via Lifelong Learning.", IEEE Transactions on Mobile Computing, 2018.

3. Bowen Du*, **Xiaoxuan Lu***, Xuan Kan, Man Luo et al. "HydraDoctor: Real-time Liquids Intake Monitoring by Collaborative Sensing." In *International Conference on Distributed Computing and Networking (ICDCN)*, 2019.

4. **Xiaoxuan Lu**, Bowen Du, Xuan Kan, Hongkai Wen, Andrew Markham and Niki Trigoni. "VeriNet: User Verification on Smartwatches via Behavior Biometrics." In *Workshop on Mobile Crowdsensing Systems and Applications (CrowdSense@SenSys)*, ACM, 2017.

5. Stefano Rosa, **Xiaoxuan Lu**, Niki Trigoni. "CommonSense: Collaborative learning of scene semantics by robots and humans." In *International Workshop on Internet of People, Assistive Robots and ThingS (IoPARTS@MobiSys)*, ACM, 2018.

6. **Xiaoxuan Lu**, Peijun Zhao, Bowen Du, Hongkai Wen, Andrew Markham, Stefano Rosa and Niki Trigoni "Demo: Automatic Face Recognition Adaptation via Ambient Wireless Identifiers." In *International Conference on Embedded Networked Sensor Systems (SenSys)*, ACM, 2018.

7. **Xiaoxuan Lu**, Xuan Kan, Stefano Rosa, Bowen Du, Hongkai Wen, Andrew Markham and Niki Trigoni "Poster: Towards Self-supervised Face Labeling via Cross-modality Association." In *International Conference on Embedded Networked Sensor Systems (SenSys)*, ACM, 2017.

1.6 Thesis Structure

The rest of this thesis is organised as follows. Chapter 2 provides an overview of related work. The following three chapters present our proposed approaches. Chapter 3 presents a general cross-modality association framework to automatically label biometric data in the wild. Chapter 4 proposes an iterative adaptation framework that gradually improves the labelling performance. Chapter 5 studies the privacy implication of our advocated concept, demonstrates a cross-modality inference attack on smartwatches and presents an effective countermeasure. Finally, Chapter 6 concludes this thesis and outlines areas for future work.

Chapter 2

Background

In this chapter, we aim to provide readers with an overview of human identification. We begin with the history of human identification in Sec. 2.1, from the early fingerprint method used in second century B.C. to the modern identification approaches in our daily lives. The taxonomy of human identification will also be introduced in this section. In Sec. 2.2, we will first introduce the working mechanism of conventional identification systems. Then we will discuss the common physiological and behavioural biometric measures, followed by a detailed discussion of biometric recognition methods. Next, in Sec. 2.3, we will look at the challenges faced by these identification methods in the wild, and briefly describe existing approaches towards effortless enrolment for human identification and their respective limitations. Finally, we will discuss some important security and privacy implications related to this work in Sec. 2.4.

2.1 Introduction

In this section, we start with a brief overview of the historical context in human identification, followed by the introduction of its basic taxonomy.

2.1.1 Historical Context

In philosophy, the matter of identity deals with such questions as, "What makes it true that a person at one time is the same thing as a person at another time?" [1]. Human identification addresses this long-standing concern irrefutably by making use of what makes one distinctive. Generally, the construction of identity is complex, multidimensional, sometimes passive, sometimes active, relational and above all body-mediated.

Figure 2.1: Frontispiece from Bertillon's Identification anthropométrique (1893). One of the most important books in the history of human identification [1].

Despite the complication of identity, mankind already had a feeling that certain characteristics were sufficient to identify a person as far back as prehistoric times [11]. In the second century B.C., the Chinese emperor Qin Shihuang was authenticating certain seals with a fingerprint [12]. Contemporarily in the Western world, the Roman military reportedly used passwords (also known as watchwords) as a way to distinguish friend from foe. An inflection point in the science of human identification happened in the 19th century, when Alphonse Bertillon took the first step in scientific policing [13]. Based on the measurements taken from certain anatomical characteristics, he was able to identify reoffending criminals. Nearly a century after Bertillon's initiative, identification methods based on physiological and behaviour features have become a *de-factor* tool in forensic investigation and military operations. For instance, the Metropolitan Police Station in the UK started the use of biometrics for identification in 1901. The USA and France immediately followed this action in the coming year and adopted biometrics as a central component in their investigation [14]. In the military field, allied forces used the behaviour patterns in Morse code transmission to authenticate the identity of Telegraph operators and received messages during World War II.

Entering the 21st century, equipments for biometric analysis are becoming much smaller

than the time of Bertillon. In the meantime, the proliferation of various cyber gadgets also opens the door for many digital identification scenarios. Nowadays, human identification is no longer only confined to forensic or military usage, but is deeply integrated in our daily lives. For example, human identification is an important building block for providing personalized services and security control in modern smart buildings [15]. We are also seeing an upsurge of human identification in law enforcement, public safety, healthcare, voter registration and privacy management [16]. Along with such a wide range of applications, a plethora of identification approaches have emerged to fulfill different use cases.

2.1.2 Taxonomy of Human Identification

At a high level, these approaches can be categorized into two types: the first one is based on digital attributes, such as passwords, ID cards, driving license number etc; The second one is based on physical attributes (i.e., biometrics), such as face, fingerprint, voice etc. Fig. 2.2 shows the classification of attributes for human identification.

In general, digital identification methods [17] can be further subdivided into two categories: 1) ownership-factor based approaches and 2) knowledge-factor based approaches. Ownership-factor based approaches are through means of "what you have", such as ID cards, badges, documents etc. In contrast, knowledge-factor based approaches are through means of "what you know", such as email addresses, driving licence number, passwords etc. As both factors are simple and static, digital identification is easy to use and absolute. On the downside, both factors can be borrowed, copied, stolen or separated with subjects, which limits their applicability to situations with high security demands.

Compared with digital identification, biometric identification is notably more secure and inseparable from subjects because biometrics are significantly more difficult to copy or forge. Depending on the criteria, biometric identification systems can be categorized into [18]: a) physiological versus behavioural; b) cooperative versus non-cooperative; c) monomodal versus multi-modal biometric systems; d) contact versus touch-less versus remote technology. The design space is highly dependent on the application scenario. For instance, a biometric based surveillance technology may operate with facial attributes, using non-cooperative user interaction and remote sensors.

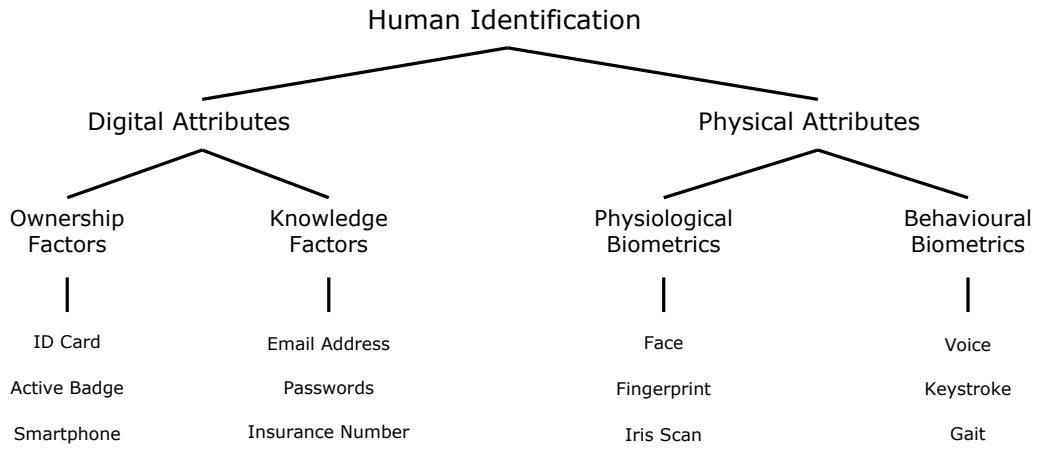


Figure 2.2: An identity attribute based taxonomy of human identification. For each line of attributes, several representative examples are listed.

2.2 Identification Systems

2.2.1 Working Mechanism

Both digital and biometric identification conceptually follow a very similar working mechanism that comprises two basic steps: *enrolment* and *matching*. Enrolment is the initial step where identity templates of registered users are collected by sensors and saved to the database. For example, the voiceprints of users are templates for speaker identification and passwords are templates for smartphone screen lock. Successful enrolment is key to identification, and significantly affects the matching step. In the matching step, test samples are compared with the enrolled templates to determine the identity.

However, digital and biometric identification have a notable difference in feature extraction. Recall that digital attributes are static once registered. Consequently, the collected digital samples can be directly used for developing enrolment templates as well as online matching. On the other side, using physical attributes or biometrics for person identification is much more complicated due to the intrinsic complexity and dynamics in physical attributes. Therefore, in order to robustly recognize users, feature extraction is necessary for both enrolment and matching. This feature extractor is absent in the pipeline of digital identification.

Fig. 2.3 shows the general workflow of a human identification system. As digital identification is very straightforward and simple, in what follows, we will focus on the discussion around biometric identification and review its two important components: biometric measures and recognition methods.

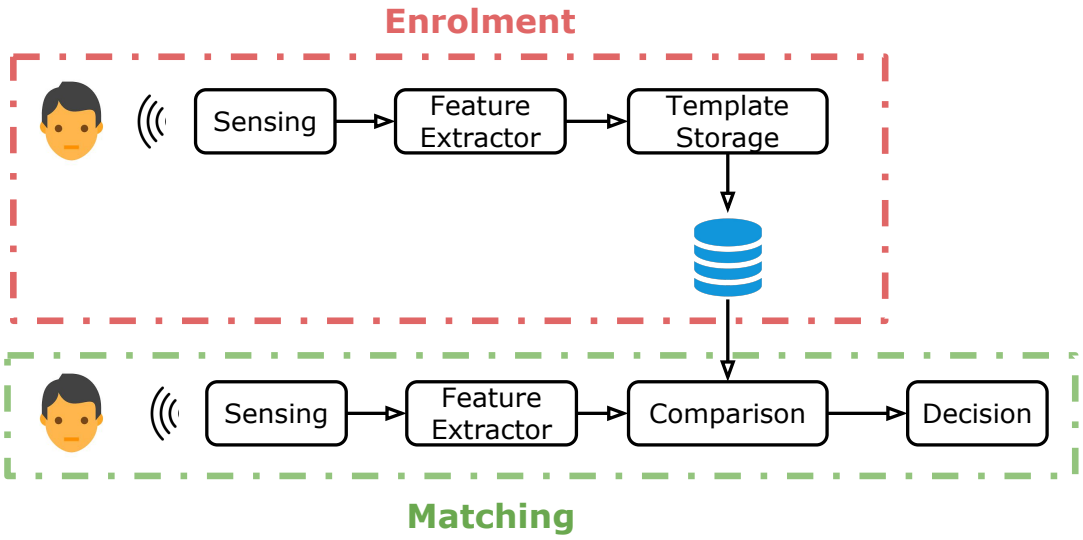


Figure 2.3: The workflow of a human identification system that has two basic step: enrolment and matching. Note that the feature extractor is *only* used for biometric recognition.

2.2.2 Biometric Measures

In 1879, the first personal identification system was introduced by Bertillon, in which he used eyes, hair and skin colours to identify different subjects. Subsequently, a number of new biometric measures have been proposed and adopted in daily identification systems. Biometric measures can be classified into two categories: (1) Physiological biometrics and (2) Behaviour biometrics. Although many biometrics fall into the above two categories, two important characteristics determine whether a specific biometric can be practically usable in our case, i.e., the wild condition:

1. **Availability.** This is the ability of specific biometric measures to be applied to a whole population of users. Availability is decided by both the universality of the biometrics themselves as well as the pervasiveness of the respective sensors or infrastructure that are able to observe them. For instance, a deaf-mute can not be enrolled with a speaker recognition system. DNA is an extremely accurate biometric technology but the respective sensing and analysing devices are very costly and lack ubiquity.

2. **Uniqueness.** The ability to successfully discriminate people. The biometric measures must be as distinct as possible from one individual to another. This characteristic is important to the matching step in an identification pipeline, as it is based on the disambiguity of personal traits.

	Biometric Measure	Sensing Modality	Pros	Cons	User Cooperation
Physiological	Iris Scan [19, 20]	Camera	Potential for high accuracy; Long term stability	Intrusive High cost	Y
	Retinal Scan [21, 22]	Infrared camera	High accuracy; Long-term stability	Intrusive Poor usability	Y
	Fingerprint [23, 24]	Live-scan device	High accuracy; Long-term stability	Affected by skin condition; Sensor may get dirty	Y
	Hand Geometry [25, 26]	Hand reader	Not affected by environment; Non-intrusive	High cost; Low accuracy	Y
	Face [27, 28]	Camera IR camera	Non-intrusive Low cost	Affected by environment; High false non-match rates	N
Behaviour	Voice [16, 29]	Microphone	Non-intrusive Low cost	Variability of the voice; Affected by background noise	N
	Keystroke [30, 31]	Touchscreen Motion sensor	Non-intrusive No additional hardware required	Low accuracy; Narrow range of applications	N
	Gait [32, 33]	Camera Geophone	Recognition at a distance; Great availability	Relatively low accuracy Difficult to scale	N
	Mouse dynamics [34, 35]	Mouse	Non-intrusive Low cost	Relatively low accuracy Difficult to scale	N

Table 2.1: A summary of popular biometric measures.

We now discuss popular biometric measures that possess the above characteristics and have been widely adopted by commercial identification systems. We start with commonly-used physiological biometrics, followed by popular behaviour biometrics. Table 2.1 summarizes popular biometrics.

2.2.2.1 Physiological Biometrics

Physiological biometrics deal with the direct measurements from a human body. As such, biometrics remain steady over a relatively large time interval. In particular, commonly-used physiological biometrics includes the following measures:

- Iris Scan
- Retinal Scan
- Fingerprint
- Hand Geometry
- Face

The above physiological biometric measures have, more or less, reached maturity. However, the adoption of a specific biometric measure greatly depends on the application scenarios.

Iris Scan. Iris recognition is the most reliable type of biometric identification developed to date. It is considered to be the ideal biometric in terms of uniqueness and stability (iris features remain extremely steady over a very long time), which leads to massive deployment for large-scale systems that proved to be very effective [20]. The iris is the coloured portion of the eye surrounding the pupil. An iris identification system searches for its specific intricate patterns composed of many furrows and ridges. Its basic steps are: image acquisition, iris localization using landmark features and segmentation, biometric template generation and biometric template matching. The acquisition factors are resolution, signal/noise ratio, contrast and illumination wavelength. Once the iris is segmented, it may suffer a pseudo-polar coordinate transformation operation to cope with the variations in pupil size. In terms of iris capture, iris recognition systems often require a very short focal length, increasing the intrusiveness of this approach. While for short focal length the image resolution is sufficient for recognition tasks, this becomes very challenging when increasing distance beyond 1 meter, leading to a significant drop in accuracy [19]. Last but not least, this approach is not applicable when the user is wearing contact lenses.

Retinal Scan. Another related biometric measure is retinal scan [21, 22]. In order to obtain retinal images, an infrared camera is often used to capture the unique pattern of veins located at the back of the eye. Similar to iris recognition, retinal measure also suffers from the problem of usability. Users need to carefully look into a camera at a very close proximity. Furthermore, retinal recognition requires complex and expensive dedicated hardware, making itself limited to applications with very high security demands [36]. On the positive side, unlike face, iris or fingerprint biometrics, retina based patterns are very difficult to spoof.

Fingerprint. Fingerprints [23, 24] are perhaps the most widely used biometric. Fingerprint based identification systems are implemented on various platforms and devices, including laptops, mobile phones, or personal digital assistants. Conventional fingerprint systems belong to touch-based sensing technology and require touching or rolling a finger onto a rigid surface with a live-scan device. Modern fingerprint systems have already been integrated in mobile and wearable devices. Despite their high accuracy in general, this system requires active cooperation of users and its performance sometimes suffers from finger placement, gloves or skin conditions (dirt, sweat, moisture) [37].

Hand Geometry. Inspired by fingerprint identification, hand geometry based recognition systems consider the measurement of length, width, thickness, and surface area of the fin-

gers and hand [25, 26]. This biometric offers low levels of security level because it is not scalable, i.e. measurements do not tend to be unique for large-scale identification systems [38]. Another downside of hand geometry that such biometric systems require complex hardware and a large device to capture the hand image and may not be appropriate for commercial or daily application scenarios.

Face. Face identification dates back to 1960's when the computer vision community started to look into the problem [27, 28]. Facial biometrics can be captured by a variety of cameras, including thermal, stereo and RGB cameras. While the performance of face identification is somewhat inferior to some strong biometric measures (such as iris or retinal identification), it is one of the most acceptable biometrics and is adopted in a wide range of applications, such as surveillance monitoring, fast payment, personalized recommendation and so on. In fact, it is the *only* physiological biometric that can be reliably measured at a distance and the identification process can be performed explicit user cooperation. However, the performance of face recognition systems varies considerably depending upon the sensing environment and various factors. Additionally, facial features suffer from long-term and short-term changes. Long-term changes refer to aging where wrinkles may appear upon the face and permanently change the facial texture. In this case, periodic enrolment is necessary to update the biometric template. On the other side, short-term changes refer to weight loss or gain. Other factors affecting the system's accuracy are partial occlusions (growing a beard or moustache, glasses, hat, scarf) or different shooting conditions, e.g., distance from camera, varying lighting conditions, camera view, motion blur, etc. As a result, a reliable face identification system working in the wild needs comprehensive enrolment and timely adaptation.

2.2.2.2 Behavioural Biometrics

Unlike physiological biometrics, behavioural biometrics typically measure human actions over a certain period of time. This type of biometric has significant availability and usually does not explicitly ask users to cooperate. As a consequence, physiological biometrics are more user-friendly, less intrusive and more convenient than their physiological counterparts. On the downside, behavioural biometrics can suffer from relatively low uniqueness and stability. Therefore, use cases of behavioural biometrics are generally unsuitable to situations with high security demands. Popular behaviour biometrics in commercial identification systems include:

- Voice

- Keystroke
- Gait
- Mouse dynamics

Voice. Speaker recognition based on voice is undoubtedly one of the most established behavioural biometrics. Although utterances involve the physical aspects of the mouth, nose and throat, this biometric is considered as a behavioural type because the pronunciation and the manner of speech is intrinsically a behaviour. The specific voice features refer to various statistics such as amplitude spectrum, localization of spectral peaks, and pitch striations. A notable feature of speaker recognition is that it can be performed either in static (text-dependent) or dynamic mode (text-independent) mode [39]. In the text-dependent mode, users are asked to repeat the same phrase as the one in enrolment. In contrast, in the case of the text-independent mode, users can freely speak any phrase. Because of the dynamics in different texts, the identification accuracy of dynamic mode is inferior to the static mode. In addition to text dynamics, human voice can be perturbed by various factors such as illness, emotional or mental state or even age, resulting in inaccurate results.

Keystroke. Keystroke analysis based identification recognizes an individual from her typing characteristics. Similar to speaker identification, keystroke identification can be performed either in static (text-dependent) or dynamic (text-independent) mode. The rationale behind keystroke identification is that when typing, different users tend to have different keystroke habits, such as inter-stroke latency, time duration between the keystrokes, dwell times (i.e. the time a key is pressed down), overall typing speed, frequency of errors (use of undo) etc. For a large scale application, these characteristics are not unique amongst many users. Therefore this analysis cannot be reliably used as a discriminative feature, but can be suitable for verification systems. Aside from relative low reliability, the enrolment procedure is another drawback of such systems. To generate representative biometric templates the user might be asked to repeat the enroll procedure by providing a username, password or a specific text many times.

Gait. Analysing the way an individual walks is the key idea behind gait identification systems. The most notable advantage of this biometric lies in fact that enrolment and identification can operate at long distance with low resolution. The fine details are not crucial, instead the temporal motion patterns are considered. Moreover, gait identification possesses wide availability and can be observed by a variety of sensors, such as radio, geophones and different types of cameras. On the downside, gait identification is error-prone

	Methods	Successful Biometric Application	Description
Unsupervised Learning	Expectation-Maximization [41]	Hand Geometry [42] Voice [43]	Used for source separation as a denoising step
	Hebbian Learning [44]	Keystroke [45] Face [46]	Used for event detection to localise biometric information
	Gaussian Mixture Models [47]	Voice [48] Face [49] Retinal scan [50]	Used for representing a large class of sample distributions; Image segmentation
Supervised Learning	Deep Neural Networks [51]	Face [52], Voice [53], Iris Scan [54], Gait [55], Keystroke [31]	Simultaneously learn features and classifiers
	Probabilistic Linear Discriminant Analysis [56]	Face [57], Voice [58]	Finding best separable subspaces by linear projection
	Random Forests [59]	Face [60], Gait [61]	Mitigating the limitation of a single decision tree, and give robust classifiers
	Support Vector Machine [62]	Face [63], Voice [64], Fingerprint [65] Gait [55], Keystroke [66], Mouse dynamics [67]	Learn an optimal hyper plane, so that samples belonging to different classes can be best separated

Table 2.2: A summary of typical learning methods in human identification systems.

when encountering external factors such as footwear walking, surface or clothing. Gait recognition systems can be subdivided into model based and appearance based approaches [40]. Model based approaches fit a model representing time pattern of the human anatomy against video data then extract and analyse its parameters. Appearance model based approaches analyse the silhouette shape and motion of an individual and the way this varies in time. For both models, the identification accuracy is highly dependent on the camera viewpoint. A change in the walking direction can significant degrade their performance.

Mouse Dynamics. Lastly, a behavioural profile can be also constructed by using mouse actions performed by an user. Common features of mouse dynamics include general movement, drag and drop, stillness, point and click (single or double) actions [34, 35]. Notably, its biometric template is built using data continuously captured in one's daily computer interactions and it can support continuous identification. However, this biometric is not very stable and suffers from behavioural dynamics, which limits its practical usage. Moreover, its identification performance is also dependent on the specific mouse's sensitivity.

2.2.3 Recognition Methods

As identifying people from digital attributes is simple and straightforward, digital identification systems can directly operate on the raw observations and give reliable predictions. However, it becomes much more challenging for the case of biometrics, due to its intrinsic

complexity and dynamics. For example, human voice is a popular biometric, but is sensitive to the specific content of speech. To account for the complexity in observing physical attributes, a pattern recognition model is often required through machine learning methods. This model extracts biometric features for both template samples in enrolment as well as the testing samples in matching. After feature extraction, the model further identifies subjects on the basis of feature similarity between testing samples and templates. Obviously, the biometric feature extractor plays a key role here and meaningful and discriminative *features* are a necessity for reliable biometric identification.

Moreover, as discussed in the above section, different biometrics may have distinct properties and need different sensors for data capture. As such, an appropriate recognition model for identity matching depends on the specific tasks. For example, convolution neural networks are known for dealing with visual data well (e.g., facial images), while recurrent neural networks are more capable of coping with sequential data (e.g., voices).

In this section, we will overview typical recognition methods in the context of biometric identification. In particular, we focus on unsupervised and supervised learning methods, both of which are established pattern recognition methods but play different roles in identification systems. Tab. 2.2 summarizes related recognition methods and their use cases.

2.2.3.1 Unsupervised Learning Methods

Unsupervised learning methods deal with the identification problems where identity labels are absent. Unsupervised learning is an established field proposed decades ago. In the context of human identification, common approaches include:

- Expectation-Maximization
- Hebbian Learning
- Gaussian Mixture Models

A common theme of these approaches is that they do not directly deal with the biometric recognition, but serve in preliminary stages of data preprocessing. For example, Expectation-Maximization (EM) is is an iterative method to find maximum likelihood or maximum a posteriori estimates of parameters in statistical models, where the model depends on unobserved latent variables [41]. EM has been widely used as denoising step when the collection biometric measurements are not clean, e.g., source separation in co-utterance speeches. Similarly, as one of the oldest learning algorithms, Hebbian learning [44] is adopted as unsupervised detection tool to localize biometric events for subsequent

recognition stages [45]. Gaussian Mixture Models (GMM) [47] are the ones closest to the recognition step. For instance, one of the most famous speaker identification systems is the one developed by MIT Lincoln Laboratory, in which an universal background model with 2048 diagonal-covariance Gaussian components was employed [48]. However, the reason for this adoption is because GMMs can represent a large class of sample distributions rather than its ability for unsupervised learning. Similarly, GMMs have also been used as an important preprocessing step for biometric image segmentation, such as retinal scans [50].

In summary, unsupervised learning is considered a good approach only for preliminary stages in an identification pipeline. Its use cases are rather limited to event detection, feature fusion, segmentation etc. As a result, these methods are not as prominent as supervised learning methods, which directly address the vital problem of matching in identification systems.

2.2.3.2 Supervised Learning Methods

Compared with unsupervised learning, the biggest difference of supervised learning is that it further requires *labels* in training. By learning with these identity labels, more reliable and accurate matching results in human identification can be achieved. Established supervised learning methods in this context include:

- Deep Neural Networks
- Probabilistic Linear Discriminant Analysis
- Random Forests
- Support Vector Machine

In what follows, we will discuss their differences and highlight their advantages and disadvantages accordingly.

Deep Neural Networks. Originating in 1960's, deep neural networks (DNNs) have achieved great successes in recent years, thanks to the availability of vast computational power and enormous storehouses of data. Mainly due to their extremely superior property of feature learning, DNNs have been adopted in a number of biometric recognition tasks. Broadly, these DNNs include multi-layer perceptrons (MLPs), convolutional neural networks (CNNs) and recurrent neural networks (RNNs) [51]. The adoption of a specific DNN can be dependent on the data type. For example, CNNs are known for dealing with image data and have been widely used in biometric systems involving facial images [52],

retinal scan and iris scan [54]. On the side, RNNs are good at coping with sequential data, and suit themselves in keystroke and speaker recognition systems [31, 53]. MLPs are one of oldest DNNs and have a wide range of adoption in different biometric systems [55]. Despite the huge advantage in feature learning, all DNNs suffer from the issues of overfitting. They can easily reach high accuracy in training but cannot generalize well to unseen cases when the training data is small. Consequently, a sufficiently comprehensive set of labelled biometric data is crucial for DNN based identification systems, which may be a huge enrolment burden.

Probabilistic Linear Discriminant Analysis. As an extension of linear discriminant analysis, probabilistic linear discriminant analysis (PLDA) is a natural fit for biometric recognition. In order to attain discriminative features, it models the intra-class and inter-class biometric variance as multi-dimensional Gaussian distributions [56]. Through linear discriminants selection, it then finds a subspace that minimizes the intra-class biometric variance caused by observation dynamics and maximizes the inter-class biometric variance between different subjects. Due to its superiority in finding a good feature subspace, PLDA has been implemented in many biometric identification systems, such as face and speaker recognition [57, 58]. However, its performance can be very unstable when the biometric data do not follow a multivariate Gaussian distribution. Moreover, it is reported that PLDA based identification systems are also very sensitive to outliers [3]. Both issues above often happen in the real world and significantly degrade the identification performance in the wild.

Random Forests. Unlike DNNs and PLDA, random forests [59] are an ensemble learning method that operate by constructing a multitude of decision trees in training and outputting the label that is the mode of the classes of the individual decision trees. In random forests, a decision tree [68], i.e. CART (classification and regression trees), is often used as a weak learner. As random forests are a collection of trees, they are able to mitigate the limitation of a single weak learner, whose predictor is rather sensitive to small perturbations in learning and generalizes poorly. Because of this advantage, random forests have been used for a variety of identification tasks, including identification of DNA-binding proteins [69], segmentation of video objects [70] and gait recognition [61]. On the downside, a large number of trees can make the algorithm to slow and ineffective for real-time identification and their performance also depends on the input features. For many biometric measures, e.g., voices, their underlying representations are rather sophisticated that cannot be interpreted well through hand-crafted features. As a result, many random forest based identification

systems have been overtaken by DNNs, which learn biometric representation from the data itself.

Support Vector Machine. Invented by Vladimir N. Vapnik, support vector machines (SVMs) are perhaps the most prevalent learning method in the 1990's. The main idea behind a SVM is the construction of an optimal hyper plane, so that samples belonging to different classes can be best separated [71]. SVMs have been heavily used in identification systems involving a various set of biometrics, including face [63], voice [64], gait [55], keystroke [66], mouse dynamics [67] and so on. In spite of its wide adoption, SVMs are known for the inability to scale, e.g., their performance does not improve with an increasing amount of label data. Unlike other classification tasks such as activity recognition, we always aim for very accurate results in human identification [26] to accommodate downstream tasks, e.g., security management. In this context, SVMs have also been overtaken by DNNs, which are more capable in dealing with "big data" to reach very high accuracy [72].

In summary, with the recent advert of DNNs, traditional supervised learning methods such as PLDA, random forests and SVMs are becoming less used for human identification. However, DNNs require a large amount of labelled data to train a reliable human identification classifier. How to acquire such comprehensive training set remains as an open question.

2.3 Towards Effortless Enrolment

As discussed in the previous section, many biometric observations are dynamic and may deviate from the template due to operation within an unpredictable environment. Failures of identification systems in the wild happen quite often because supervised learning methods, e.g., DNNs, can overcome this diversity only if a comprehensive training set is given. Ideally, this training set needs to contain samples in conditions that are close to the real-world scenarios. Unfortunately, acquiring such a comprehensive training dataset is very challenging for several reasons. First, asking a large number of users to register sufficient data would incur a huge enrolment effort and would be costly to stage. Second, conventional crowd-sourcing tools [7], e.g., Amazon Mechanical Turk, are not applicable in this context. These crowdsourcing tools generally assign data-to-labels on public platforms which would result in serious privacy concerns as many identity related data are sensitive. As a

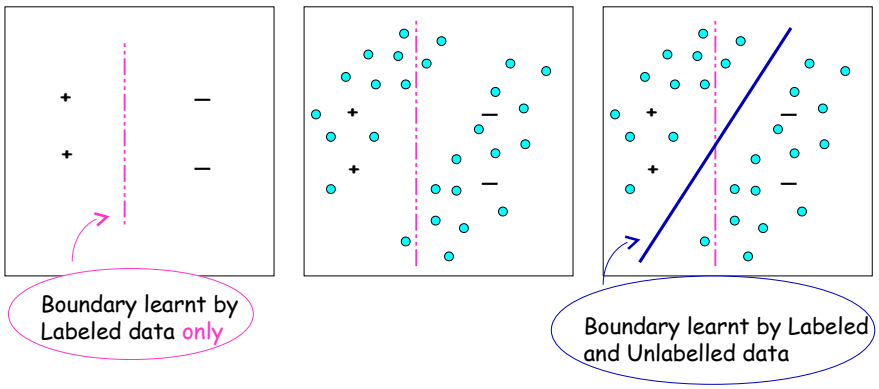


Figure 2.4: Intuition of semi-supervised Learning. It uses the statistical property of a large quantity of unlabelled data to improve learning with sparse labelled data. [2]

consequence, there is an urgent need for reducing the enrolment effort while maintaining the high accuracy of biometric identification.

Existing work towards solving this lack of labelled data can be mainly classified into two categories: (1) semi-supervised learning based approaches and (2) data association based approaches. We are now in position to review them.

2.3.1 Semi-supervised Learning

Semi-supervised learning [2] has attracted an increasing amount of interest as it requires less effort for labelling data. The key rationale behind semi-supervised learning is that unlabelled data is usually easy to acquire in large quantities; such data can be used to either modify or re-prioritize hypotheses obtained from labelled data alone [73]. Fig. 2.4 illustrates the intuition behind semi-supervised learning. Broadly, semi-supervised learning methods can be subdivided into two types: i) iteration-based methods and ii) representation-based methods [74]. Iteration-based methods start from initial identification models and *iteratively* enhance them. For example, using expectation-maximization (EM) methods to optimize an identification model by maximizing the log-likelihood of labelled and unlabelled biometric data [75]. In contrast, representation-based methods aim to discover the inherent feature representation for a certain type of biometric data, and exploit it to find a good identification model. Examples of them include manifold regularization [76, 77], harmonic mixtures [74] and information regularization [78].

Both categories of semi-supervised learning methods have been used in biometric identification, such as face recognition [79], speaker identification [80], gait recognition [81] etc. However, they still require a certain amount labelled data for bootstrapping a model.

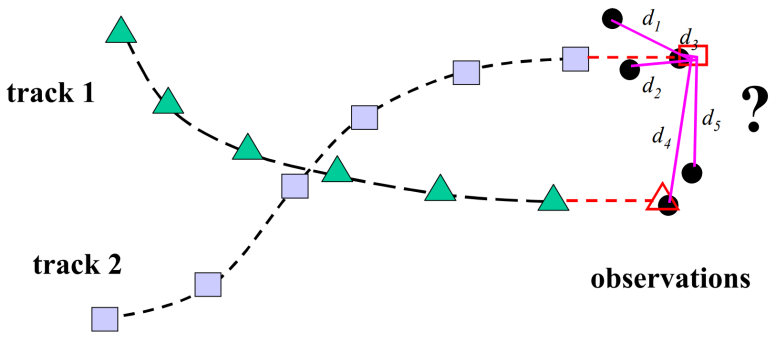


Figure 2.5: Data association. Essentially data association tries to address the assignment problem of *which observations belong to which tracks*. Data association is based on the similarity between (sensor) observations and the model predictions of tracks.

Moreover, these methods can be very error-prone when facing out-of-set subjects in unlabelled data, which is very common for the identification systems in the wild. Therefore, existing semi-supervised learning methods cannot fundamentally reduce the enrolment effort for a robust identification system in the wild.

2.3.2 Data Association

Unlike semi-supervised learning, data-association based methods directly assign labels to unknown data. Generally, data association is used in problem settings with a state space and history of observations. Fig. 2.5 illustrates a general problem setting of data association. In this sense, data association techniques essentially seek to find *which observations belong to which states* based on temporal consistency. A cost or likelihood function is often used to evaluate all possible observation-to-track combinations which are then used as input to the well-known assignment problem.

In the past decades, a variety of data association methods have been proposed, such as forced alignment [82], nearest-neighbour data association filter [83], joint probabilistic data association filter [84] and multiple hypothesis tracking [85]. All of them follow the assumption of a given state space model, e.g., a motion model for object tracking or an acoustic model for speech segmentation. The choice of a specific method is mainly dependent on the application. For example, forced alignment is usually used in speaker identification as an important pre-processing step for utterance segmentation. Joint probabilistic data association filter and nearest-neighbour data association filter are widely adopted in face recognition tasks to disambiguate the sequences of facial images. Multiple hypothesis tracking is a *de-facto* method to retrieve visual tracks for gait recognition in surveillance videos.

Data association seems a plausible method to reduce enrolment effort. Unfortunately, the assumption of an available temporal evolving state-based model for both modalities is difficult to fulfill in the real world. For example, in our problem setting of cross-modality labelling, detecting a MAC address does not imply that someone will be speaking at that exact instant. Therefore, existing data association methods cannot address the cross-modality labelling issue for human identification.

2.4 Security and Privacy Implication

Although effortless enrolment can enable robust human identification, it also has repercussions for security and privacy. When being maliciously used by attackers, cross-modality inference can expose users to significant security and privacy risks, because private data might be covertly labelled and harvested while users are unaware of it. In this section, we present three types of attacks related to this issue and discuss existing solutions to mitigate these attacks.

2.4.1 Spoofing Attack

A spoofing attack occurs when a person tries to masquerade as someone else by falsifying data and thereby gaining illegitimate access and advantages [86]. This class of attacks threatens both digital and physical attributes and poses unique challenges to human identification. For digital identification, commonly-used methods include IP address spoofing [87], MAC address spoofing [88], caller ID spoofing [89] and E-mail spoofing [90] etc. For biometric identification, spoofing attacks have been observed on a wide range of biometrics, such as fingerprint [91], face [92] and voices [93]. Compared with digital spoofing, biometric spoofing is more costly as forging or impersonating physical attributes often requires more biometric observations. As we can imagine, spoofing attack will be exacerbated when cross-modality labelling is maliciously used to automatically provide labelled biometric observations to attackers.

In order to mitigate such attack, a variety of anti-spoofing methods have been proposed. Anomaly detection is the most widely adopted countermeasure against digital spoofing attacks. Anomaly detection leverages learning algorithms (e.g., SVMs) to inspect and certify data before it is transmitted and block data that appears to be spoofed [94]. However, anomaly detection is sometimes error-prone and gives false-negative predictions under abnormal actions of legitimate users [95]. Two-factor authentication [96] is an effective coun-

termeasure against both digital and biometric spoofing attacks. It confirms users' claimed identities by using a combination of two different factors, picked from knowledge factors, ownership factors, or biometrics. The rationale behind this countermeasure is simple as spoofing two or more identity attributes is significantly more difficult than spoofing one. However, as users are required to cooperatively provide an extra factor, the complexity of two-factor authentication is still an issue.

2.4.2 Side Channel Attack

Side channel attacks are a class of attacks proven to be very powerful in practice. By measuring side channel information, the attacker is able to recover very sensitive information [97]. The observed side channel attacks in literature include timing attacks [98], power analysis attacks [99], electromagnetic analysis attacks [100], acoustic attacks [101]and traffic analysis attacks [102]. Conceptually, cross-modality labelling can exacerbate side channel attacks and cause even more damages. By taking one modality as the source of sensitive information (e.g., passwords) and the other modality as the side channel, attackers can use learning methods to increase the success rate of attacks.

Since side-channel attacks rely on the relationship between information emitted (leaked) through a side channel and the secret data, their countermeasures fall into two main categories: (1) *access control* and (2) *decorrelation*. Access control eliminates or reduces the release of emitted information [103]. For example, displays with special shielding to lessen electromagnetic emissions can reduce susceptibility to TEMPEST attacks [104]. On the downside, access control sometimes degrade the quality of data utility for the original services, especially in critical applications. In contrast, decorrelation [105] works by eliminating the relationship between the leaked information and the secret data, that is, make the leaked information unrelated to the secret data, typically through some form of randomization of the ciphertext. Example countermeasures of decorrelation include *blinding* [106] and *masking* [107].

2.4.3 Linkage Attack

Linkage attacks are a recent class of attacks that break k-anonymity and reidentify user [108]. In such an attack, adversaries collect auxiliary information about a certain individual from multiple data sources and then combine that data to form a whole picture about their target, which is often an individual's personally identifiable information. Linkage attacks have a long history and can be dated back to the famous de-anonymization of a Massachusetts hospital discharge database by joining it with a public voter database [109].

Recent studies demonstrated that linkage attacks are able to infer privacy across different scenarios, including movie ratings forums [110, 111], crowdsourced sensor data [112, 113] and social networks [114, 115]. Sparsity in high dimension attributes is the key property that underpins the above de-anonymization approaches. In an environment with heterogeneous sensing modalities, one modality can opportunistically play the role of secondary linkage for the other co-located modality.

Developing solutions to mitigate linkage attack is non-trivial. Conventional solutions, such as perturbing the data, might diminish the utility gained from data [116]. To date, it remains an open question to find countermeasures against linkage attacks that balance the tradeoff of data utility and information safety.

2.5 Summary

As discussed in this chapter, the maturity of biometric sensing and advent of recognition methods have solved the problem of human identification in a variety of scenarios and environments. However, we also observed three key limitations of the prior art that motivate this thesis. Specifically, they are summarized as follows:

1. Supervised learning methods are undoubtedly the most prevalent methods used to identify people. However, in order to robustly operate in the wild, these methods require a comprehensive training set. Unfortunately, such datasets are costly to obtain as we advocated in Sec. 1.2.

2. Data association methods are effective in associating cross-modality data. A straightforward idea would be using data association methods to automatically label data and thereby to reduce the enrolment effort of biometric recognition. However, most existing data association approaches are based on the assumption that there is a temporal evolving model underlying both modalities. This assumption cannot apply to the identification problem as heterogeneous identity data are usually unaligned. For instance, knowing the cast of characters of a television series does not imply who speaks when in the program. Together with the first limitation, we will address both of them in Chapter 3 and Chapter 4.

3. People are largely aware of the privacy implications when they share their sensitive data, such as face and voice. However, when it comes to less well understood yet equally sensitive data, such as motion data for a fitness tracker, privacy implications

are usually ignored. The privacy leakage problems of innocuous sensor data are overlooked, especially on emerging wearable gadgets. Investigating side channel attacks based on cross-modality inference and providing timely countermeasures are urgently needed. Chapter 5 in this thesis is dedicated to addressing this problem.

Chapter 3

Cross-Modality Association

3.1 Introduction

People identification is a key component of smart spaces, e.g., offices and buildings for determining who is where. Biometric features allow a person to be identified and authenticated based on a set of recognizable and verifiable data, which are unique and specific to them. As they are able to irrefutably prove one's identity, biometric features have quickly established themselves as the most pertinent means of identifying individuals in many applications. For example, face recognition [117] has been used for fast payment [118]; voice recognition is now supporting many personalized customer services [119] and gait recognition is widely used in forensic analysis [120].

However, biometrics, such as voice or gait, is challenging to be comprehensively qualified and have complex manifestations of variability in the wild. Biometric identification systems usually contain a *pattern recognition* module that operates by extracting a feature set from the acquired data and comparing this feature set against the template set in the database [121]. A vast amount of research over the past decades has gone into designing tailored pattern recognition models for specific biometrics; and with the advent of deep learning, progress has accelerated. Due to their superior ability in representation learning [122], deep neural networks are widely adopted as the feature extractor for biometric data and achieved remarkable performance when a comprehensive training set is given [117, 3]. However, failures of biometric recognition often happen when the training dataset is small. For example, UK police has been struggling to use the face recognition technology in forensics due to lack of equivalently diverse training images in the wild [123].

A comprehensive training set is crucial for robust human identification. However, obtaining such a dataset incurs huge enrolment effort and is costly to stage. In this chapter, we propose `SCAN`, a general cross-modality association framework that automatically labels

biometric samples via co-located digital observations. SCAN exploits the fact that biometric observations and digital observations are usually, though not always, *co-located*. For example, a person is usually holding or wearing their mobile devices, e.g., smartphones and fitness monitors. To provide ubiquitous connectivity, these devices have some form of wireless interface, e.g., BLE, WiFi, or cellular. These provide a unique identifier, ranging from the hardware level (e.g., IMEI or MAC addresses) to the network authentication level (e.g., usernames). When people are attending meetings, we can simultaneously collect MAC addresses of their devices (digital identifiers) and biometric observations (e.g., face images, voice segments, etc.). Television programs, e.g., Friends[1], are another example in which actors' names (digital identifier) and their faces (biometric observations) are co-located in different episodes. Based on this intuition, SCAN aims to use a set of such digital observations as ID proxies to label a set of biometric observations.

Finding the right associations is not trivial as the relationship between heterogeneous sensor data is generally non-aligned. For example, detecting the WiFi identifier of a device does not imply that the user of that device will be speaking at that exact instant. Further, the biometric observations are noisy and sometimes contain data of non-people-of-interest (non-POI). For instance, the facial images in television programs are not necessarily all drawn from actors, but may also contain additional people whose names are unlisted. Robustly ranging biometric observations under such disturbances poses significant challenges.

SCAN hence develops a novel algorithm that simultaneously clusters and names biometric observations, yielding accurate, effortless biometric labelling. In summary, the contributions of this chapter are as follows:

- We observe that the cross-modality information about the likely attendance of subjects in a session provides valuable, albeit noisy, clues about the person's identity.
- We propose a novel algorithm which simultaneously handles clustering and association, and highlight the benefits of the algorithm compared to handling these problems in a sequential manner.
- We compare SCAN against competing approaches using two case studies, one based on sensor data that we collected, and one based on a real world online sensor dataset and show more than 20% improvements in performance, especially in noisy environments.

[1] https://en.wikipedia.org/wiki/Friends

The rest of this chapter is organised as follows. Sec. 3.2 formulates the problem considered in this chapter and Sec. 3.3 explains how the baseline approach tackles this problem. In Sec. 3.4 we present the proposed Simultaneous Cluster and Naming (SCAN) method and Sec. 3.5 provides the implementation details of SCAN. Sec. 3.6 evaluates the SCAN algorithm and compares its performance with competing approaches. Sec. 3.7 surveys the related work, while Sec. 3.8 concludes this chapter.

3.2 Problem Definition

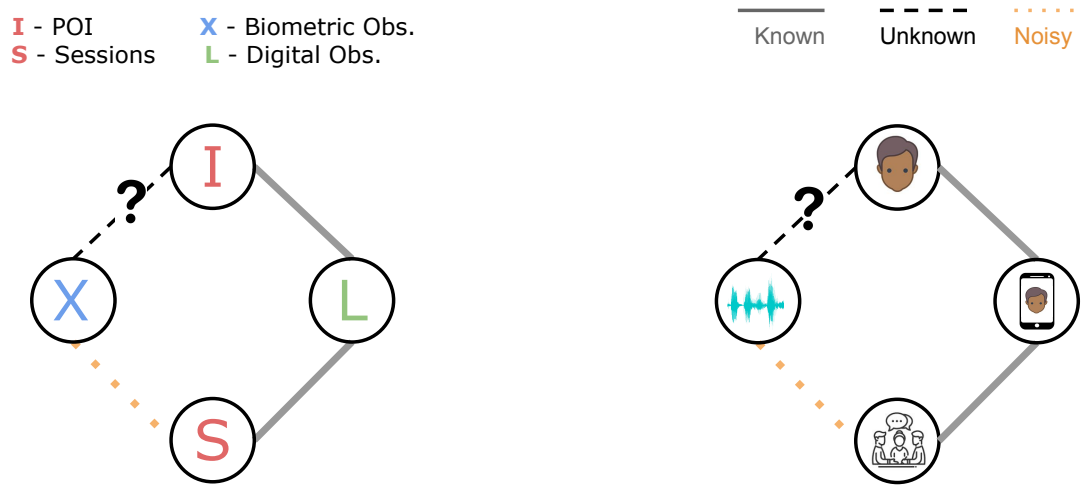
In this section, we explain the key terms in this thesis and define the core problem of cross-modality association.

Sessions: We use the term sessions $\mathcal{S} = \{s_j | j = 1, 2, \ldots, g\}$ to broadly refer to settings in which users interact with entities in an environment, e.g., a physical visit, a meeting in a particular room, a television program or a teleconference.

People-of-Interest (POI): We refer POI to the set of subjects whose mapping to digital attributes are known, denoted as $\mathcal{I} = \{i_j | j = 1, 2, \ldots, m\}$. Examples of POI include regular workers in a smart building, whose smartphone MAC addresses are bonded in the database with their accounts. POI can also be the television actors whose names are in the cast of characters. By contrast, non-POI broadly refers to those people whose mapping to digital attributes are unknown and that we are not interested in profiling, e.g., a short-term visitor or audiences in the television program.

Biometric Observations: Biometric observations are denoted as $\mathcal{X} = \{x_j | j = 1, 2, \ldots, n\}$, which are sensor measurements of a biometric attribute, for instance, voices or facial images of a person. In our context, biometric observations of interest are passive observations of people collected by infrastructure sensors without explicit user cooperation. In order to accurately reflect the real-world complexity, we assume the collection of biometric observations in a session may contain samples of non-POI. For example, the surveillance footages of an office environment usually contain facial images of short-term visitors.

Digital Observations: Digital observations are measurements of digital attributes of a person and are denoted as $\mathcal{L} = \{l_j | j = 1, 2, \ldots, m\}$. Depending on their manifestation types, digital observations can be observed by physical sensors (e.g., a WiFi sniffer) or obtained from the meta information (e.g., the cast of characters). Digital observations are discrete and usually have the same manifestation as the attributes themselves. For example, an observation of a WiFi identifier is the identifier itself. Therefore in this chapter, we



(a) Abstraction (b) Instance

Figure 3.1: Relationship between biometric and digital observations. (a) Abstraction. (b) Instance. Given the *noisy biometric observations* and co-located digital observations, SCAN aims to accurately label biometric observations through the digital observations. The link between biometric observations and sessions are uncertain due to factors such as the disturbance from non-POI.

assume that digital observations are digital attributes, and *accurately* reflect the session attendance of subjects.

In the absence of in-domain training data, it is challenging to immediately associate biometric observations to individual POI. Therefore, the problem to be addressed in this thesis is: given noisy heterogeneous observations in multiple sessions, find the correct association between the noisy biometric observations $\mathcal{X}$ and digital observations $\mathcal{L}$ of POI. Since digital attributes are simple and their mapping to POI $\mathcal{I}$ is known, deriving the above association can replace manual labelling. Fig. 3.1 provides a simple schematic illustration and an example of the problem setting.

3.3 Baseline

3.3.1 Two-step Approach

In order to address the above problem, a naive approach is to leverage the diverse participatory information in multiple sessions and use a two-step procedure: a) in the Clustering Step, biometric observations $\mathcal{X}$ are firstly grouped into clusters across all sessions, each of

which represents the biometric samples of a single person; and then b) in the Data Association Step, the clusters are assigned with identities based on digital observations $\mathcal{L}$.

3.3.1.1 Clustering Step

Given a set of sessions, biometric observations are first transformed into feature vectors $\mathcal{Z}$, through a biometric representation model f_θ pre-trained on out-of-domain public datasets. In practice, such pre-training is generally supervised by metric losses [117], e.g., a triplet loss so that the learned features are suitable for clustering [124]. Based on the extracted features, these biometric observations are then merged into disjoint, non-overlapping clusters. Without loss of generality, we denote the set of derived clusters by $\mathcal{C} = \{c_i | i = 1, 2, \ldots, h\}$. In order to make valid assignments in the subsequent association step, the number of clusters h must be equal to or greater than the number of people of interest (POI) m.

3.3.1.2 Data Association Step

Based on the similarity of session attendance, biometric clusters can be mapped to digital observations by data association. Let $\mathbf{r}_{c_k} = (r^1_{c_k}, r^2_{c_k}, \ldots, r^g_{c_k})$ be the context vector of the k-th biometric cluster c_k, where g is the total number of sessions. $r^j_{c_k}$ is set to 1 only if c_k contains biometric observations from session s_j. At the same time, a POI's digital observation (attribute) l_i is also linked with a context vector $\mathbf{r}_{l_i}$, and $r^j_{l_i}$ is set to 1 only if l_j is detected in session s_j. An edge can be created between a cluster c_k and a digital observation l_j, with the edge weight determined by the similarity in terms of context vector. Intuitively, a higher similarity score means that there are more shared session attendances and such pairs of biometric clusters and digital attributes are more likely to belong to the same identity. Then associating identities with clusters is equivalent to solving the combinatorial optimization problem on the weighted bipartite graph, e.g. using the Hungarian algorithm [125]. Finally, through a mapping table between digital attributes and POI, biometric observations in the same clusters are all labelled with the same user identity.

3.3.2 Limitation of Baseline

The above method addresses the identification problem in two sequential steps: biometric observations are firstly clustered and then matched to digital observations by minimizing the combinatorial mismatch. Although this approach is simple and easy to implement, it is not robust when biometric observations are noisy. For example, as showed in Fig. 3.2, a speaker's voices may vary considerably across sessions due to illness or emotional influences [126], confusing the clustering step and causing unrecoverable knock-on effects on

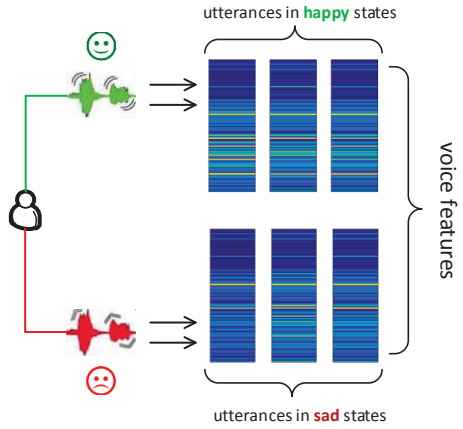


Figure 3.2: Deviations of voices due to the different emotion states of the speaker.

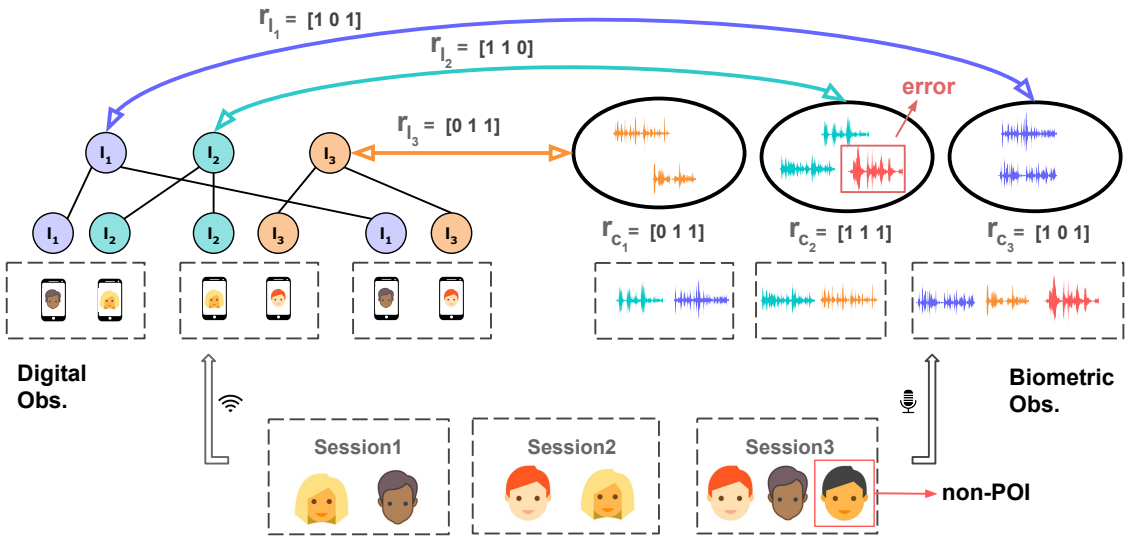
the ensuing association step. Secondly and more importantly, errors can also occur due to non-POI disturbances. For example, a session may contain faces or voices of non-POI's, and their respective digital attributes are unknown to us. Due to the disturbances incurred by non-POI, the number of clusters h is difficult to know. A misleading clustering result could further degrade the quality of data association. Fig. 3.3a shows an example of the erroneous cluster caused by non-POI.

3.4 SCAN: Simultaneously Clustering And Naming

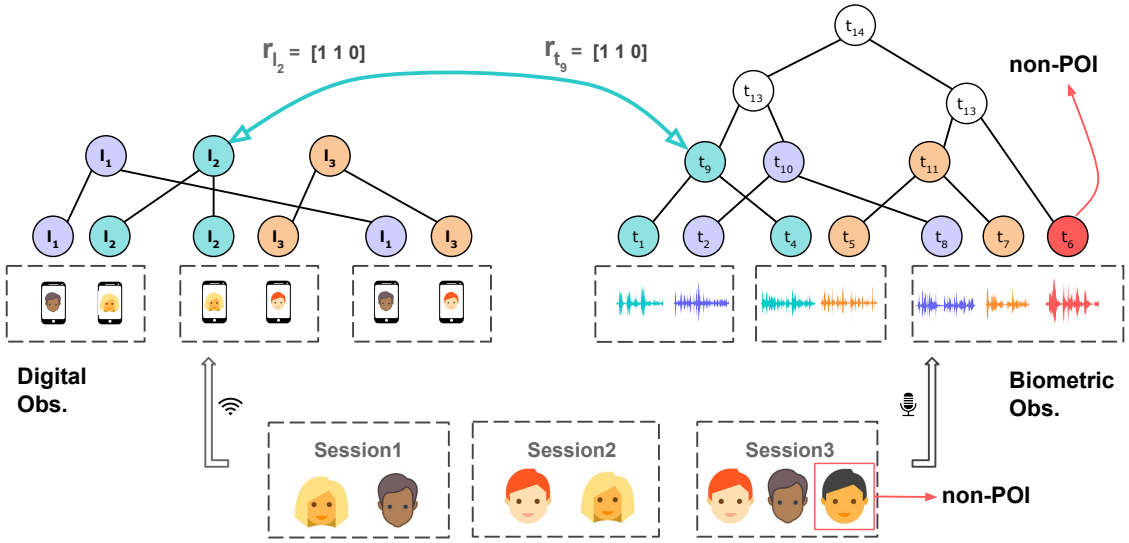
In this section, we introduce how to mitigate the above limitations. The key insight of our solution is that the clustering of biometric observations should not be finalized independently of and in advance of data association, but both tasks should progress in tandem. The proposed **S**imultaneous **C**lustering **A**nd **N**aming (SCAN) algorithm works as follows. It firstly compiles biometric observations as an augmented linkage tree, which succinctly encodes the hierarchical clustering plans of biometric observations across different sessions. More importantly, every node in the linkage tree also encodes the association potential through an extra score term. It then finds the best clustering and data association plan by solving a constraint optimization problem on the constructed linkage tree.

3.4.1 Linkage Tree Construction

Based on the pairwise similarity between biometric features, our proposed algorithm compiles these features into a linkage tree T. The leaf nodes T_{leaf} are biometric samples, while



(a) Baseline: Two-step Approach



(b) SCAN: *Simultaneously* Clustering and Naming

Figure 3.3: Comparison between baseline and SCAN, with an example of co-located microphone and WiFi sniffer. (a) Two-step Approach. It firstly clustered biometric observations based on their feature similarity and then associates these clusters to digital attributes based on the similarity of context vectors. (b) SCAN. It simultaneously performs clustering and association, by directly examining the fitness between a digital attribute and a node in the tree, in terms of context vector similarity. Intuitively, SCAN tolerates disturbances of non-POI as their biometric samples are unselected nodes on the tree.

a branch node represents the cluster of all its descendant leaf nodes. Essentially T represents the hierarchical clustering of all biometric observations in different sessions, and selecting a combination of nodes from the tree will give a specific clustering plan. For

example in Fig. 3.3b, selecting nodes t_9 means that leaf nodes t_1 and t_4 should be grouped together (and thus belong to the same individual). Each node t_i in T is associated with a *linkage score* q_{f_i}, describing the feature similarity or compatibility between the data within the cluster it represents.

Recall that given a linkage tree T, the clustering process of the baseline approach is equivalent to finding the set of nodes in T that maximises the total linkage score. However as discussed in the previous section, this is not reliable due to noisy biometric observations. The proposed `SCAN` algorithm thus augments the linkage tree by introducing additional data association scores to each of its nodes t_i, which represent the fitness of assigning an identity label to t_i given digital attribute observations $\mathcal{L}$. Concretely, let $\mathbf{r}_{t_i}$ be the context vector of a node t_i, where $r^j_{t_i} = 1$ if t_i contains biometric samples collected from session s_j. Similarly, a POI's digital attribute l_k is also linked with a context vector $\mathbf{r}_{l_j}$, and $\mathbf{r}^j_{l_j}$ is set to 1 only if l_j is detected in session s_j. Intuitively, for a node t_i and a digital attribute l_j, if $\mathbf{r}_{t_i}$ and $\mathbf{r}_{l_j}$ are similar enough, it is very likely that biometric observations under node t_i are actually the biometric data of the person who owns digital attribute l_j, since they appear in similar series of sessions and match with each other well.

Formally, for a node t_i, we define its data association scores with respect to the digital observations as a vector $\mathbf{q}_{a_i} = (q^1_{a_i}, q^2_{a_i}, \ldots, q^m_{a_i})$, where the j-th score $q^j_{a_i}$ is the Euclidean distance between the node context vector $\mathbf{r}_{t_i}$ and the digital context vector $\mathbf{r}_{l_j}$. Together with the feature score, the final score to assign node t_i to digital attribute l_j is a composite score function:

$$q^j_i = (1 - \omega) * q_{f_i} + \omega * q^j_{a_i} \tag{3.1}$$

where the parameter ω governs how much we trust the digital attribute observations and to what extent we want them to impact the result of clustering.

3.4.2 Optimization Program

With the previously introduced terms and notations, we formulate the following optimization problem:

$$\max_{\mathbf{A}} \quad \sum_{i=1}^{n}\sum_{j=1}^{m} q_i^j * a_{i,j} \tag{3.2}$$

$$s.t. \quad \sum_{j=1}^{m} a_{i,j} \leq 1, \forall i \in 1, \ldots, n \tag{3.3}$$

$$\sum_{i=1}^{n} a_{i,j} = 1, \forall j \in 1, \ldots, m \tag{3.4}$$

$$\sum_{i\in\Pi_k}\sum_{j=1}^{m} a_{i,j} \leq 1, \forall k \in T_{leaf} \tag{3.5}$$

$$a_{i,j} \in \{0,1\}, \forall i \in \{1, \ldots, n\}, \forall j \in \{1, \ldots, m\} \tag{3.6}$$

where $\mathbf{A} = (a_{i,j})_{n\times m}$ is the decision variable and q_i^j is the composite score determined by Eq. (3.1). T_{leaf} represents the set of all leaf nodes in the linkage tree. The objective function aims to maximize the total scores when selecting m nodes in the linkage tree T with size of n. Intuitively, the selected m nodes are the optimal clusters out of these n biometric observations. The inequality in Eq. (3.3) simply means a node can be assigned to at most one digital attribute. The constraints in Eq. (3.4) are used to ensure each digital attribute is associated with a single node. As a clustering tree, a node cannot be selected with its ancestors or descendants at the same time since they contain duplicate data. In order to compile this tree structure in optimization, the constraint Eq. (3.5) is enforced to guarantee that on any path leading to a leaf node, at most one node is assigned to a digital attribute. Finally, the constraint Eq. (3.6) is there to make sure that decision variable $a_{i,j}$ can take on the integer value 0 and 1 only. The above optimization formulation is essentially an integer linear programming (ILP) problem and can be readily solved by either exact or approximate algorithms [127, 128].

This finishes our SCAN algorithm. Notably, it bypasses the requirement of knowing the exact number of subjects, but only depends on the number of POI to associate. Furthermore, joint clustering and association prevents associating impure clusters. As illustrated in Fig. 3.3b, SCAN early selects pure clusters before they merge with wrong samples that contaminate their context vectors for association.

3.5 Implementation

We are now in position to introduce how we pre-process biometric observations from sensor data. Particularly, we choose two very pervasive biometrics in our experiment, e.g., facial

Algorithm 1: SCAN

Input: pre-trained biometric feature extractor f_θ, digital observations $\mathcal{L}$, vocal observations $\mathcal{X}$, Sessions $\mathcal{S}$, number of SOI m, threshold ϵ and mapping table $\mathcal{L} \Rightarrow \mathcal{I}$

Output: Labelled biometric database $\mathcal{X} \Rightarrow \mathcal{I}$

$\mathcal{Z} = f_\theta(\mathcal{X})$

$\mathbf{r}_{l_{1,\dots,m}} = \texttt{digital\_context\_vector}(\mathcal{L}, \mathcal{S})$

$T = \texttt{linkage\_tree}(\mathcal{Z})$

$\mathbf{A} = \texttt{SCAN}(m, \mathbf{r}_{l_{1,\dots,m}}, T)$

$\mathcal{X} \Rightarrow \mathcal{I} \leftarrow \texttt{table\_mapping}(\mathbf{A}, \mathcal{X}, \mathcal{L} \Rightarrow \mathcal{I})$

images and speaker voices. The rest of this section firstly introduces how to acquire vocal features from audio clips and then discusses the case of facial features.

3.5.1 Vocal SCAN

A conversation audio clip usually lasts the entire session; however, SCAN operates at the instance-level. Speaker diarization hence comes prior to any further steps. Ideally, with utterances of each speaker, conversation processing moves on to feature vector, or voice embedding extraction. The extracted features are then used for linkage tree construction.

3.5.1.1 Utterance Segmentation

Utterance segmentation is the product of speaker diarization [129]. We adopted the speaker diarization pipeline implemented in Kaldi toolkit[2]. The underlying speaker diarization system operates as a means of intra-context merging of overlapping sliding windows on the session-wise audio clips. Sliding windows are firstly processed through MFCC feature extraction, and remove non-informative components, such as silent gaps, background and high-frequency noise. Cepstral mean and variance normalization (CMVN) is then used to account for the intra-conversation variability of the audio data, followed by speaker feature extraction (x-vector in our case). Intuitively, given two consecutive sliding windows, if the latter one contains a changing point while the former window is integral, their speaker feature vectors should be significantly dissimilar. In practice, this similarity is measured by a scoring function (PLDA scoring in our case), and is compared with a predefined threshold (e.g., 0 as instructed by Kaldi) to determine whether a changing point should be placed. Fig. 3.4 shows the pipeline of utterance segmentation.

[2] http://kaldi-asr.org/

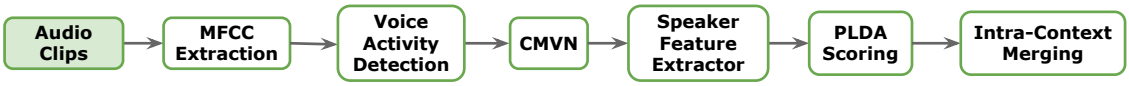


Figure 3.4: Steps of utterance segmentation. MFCC: Mel-frequency cepstral coefficients; CMVN: cepstral mean and variance normalization; PLDA: probabilistic linear discriminant analysis. The speaker feature extractor used in this implementation is x-vector [3]. Segmented utterances are then used as the input (biometric samples) to SCAN.

3.5.1.2 Voice Embedding

Voice embedding is critical for speaker recognition and aims to extract the underlying biometric representations of different speakers. The state-of-the-art x-vector architecture proposed in [3] was adopted in our experiment, which is publicly available in Kaldi [130]. The system uses 24 MFCC banks as input features for a time-delayed deep neural network. After five time-delay layers, a stats pooling layer is used to aggregate frame-level knowledge into segment-level features. The aggregated vector is then passed through several fully-connected layers to generate a high-level speaker embedding. This feature extractor is trained with a softmax cross entropy loss function, and a PLDA backend is adopted to encourage discriminative features. In our implementation, we used the x-vector feature extractor pre-trained on the augmented VoxCeleb corpus [53] (augmented with MUSAN[131]), and the PLDA backend initially parametrized on VoxCeleb. Notably, although x-vector is adopted in both diarization and recognition, the input to it is different. For diarization, x-vector is extracted on a sliding-window (or frame) basis, while x-vector used in recognition is extracted from utterances that comprises several frames.

3.5.2 Facial SCAN

We also implemented a preprocessing module to retrieve facial images in videos before feeding them into *Facial SCAN*. Two key steps in preprocessing are detailed in the following section.

3.5.2.1 Face Detection

Face detection is the first step prior to any further steps. A cascaded convolution network MTCNN [132], was adopted in this implementation to detect facial images in videos. It is cascaded by three sub-networks for different detection stages, a proposal network, a refine network and an output network. In the first stage, a fully convolution network, called Proposal Network (P-Net), is exploited to obtain the candidate facial windows and their bounding box regression vectors. Then candidates are calibrated based on the estimated

Figure 3.5: Steps of face detection. NMS: non-maximum suppression. BBR: bounding box regression. It includes three sub-networks (P-Net, R-Net and O-Net) in different detection stages. The detected facial images by O-Net are then used as the input (biometric samples) to SCAN.

bounding box regression vectors. After that, P-Net employs non-maximum suppression (NMS) to merge highly overlapped candidates. In the second stage, all candidates are fed to another CNN, called Refine Network (R-Net), which further rejects a large number of false candidates, performs calibration with bounding box regression and conducts NMS again. Lastly, in order to identify face regions with more supervision, the output network (O-Net) returns five facial landmarks' positions of a certain image. Each of the three networks uses the bounding boxes of potential faces and the corresponding detection probabilities, i.e., confidences. Face detection with small confidence will be discarded soon and not sent to the subsequent sub-network. In our implementation, the confidence threshold was set to 0.7, 0.7 and 0.9 for three sub-networks respectively. Following the original setting in [132], we set the minimal face size in detection to 40×40 pixels. Fig. 3.5 illustrates this pipeline of face detection using MTCNN.

3.5.2.2 Face Embedding

In order to extract discriminative features from detected images, we adopted the popular FaceNet architecture [117], which is the state-of-the-art method that achieves an accuracy of 99.65% on the LFW face verification task[3]. It uses a deep convolution network trained to directly optimize the embedding itself, rather than an intermediate bottleneck layer as in previous deep learning approaches. The training of FaceNet is supervised by triplet loss [52], which summarizes the matching / non-matching face patches and encourages discriminative feature representation. In our implementation, we used Inception-ResNet-v1 [133] as the backbone of FaceNet and its weights were pre-trained on the VGGFace training set [134]. The pre-training protocols, e.g, parameter settings, can be found in [117].

[3] http://vis-www.cs.umass.edu/lfw/

3.6 Evaluation

In this section we evaluate the effectiveness and sensitivity of SCAN. Our experiments show that SCAN is consistently superior to baseline techniques and is robust under a variety of conditions.

3.6.1 Datasets

To comprehensively evaluate SCAN, we use the datasets of two tasks with different biometrics. The first task is automatic *voice labelling*, which contains the 50 POI and 20 disturbed non-POI. The conversations between different sets of speakers are downloaded from VoxCeleb2 [135]. After diarization, each speaker has over 300 utterances. We used these conversations to mimic the scenarios of radio programs and synthesized 100 sessions. On average, there are ~ 11 POI in each session while each session comprises ~ 97 utterances. On top of the widely used test set of VGGFace2 [134], we similarly developed our second dataset for automatic *face labelling*, which also contains the same number of POI, non-POI and sessions as *vocal labelling*. This time, there are 18 POI and 113 facial images on average per session. Note that, we use two different biometrics in evaluation to demonstrate that SCAN is biometrics-agnostic and can be useful to various scenarios.

3.6.2 Evaluation Methodology

3.6.2.1 Metrics

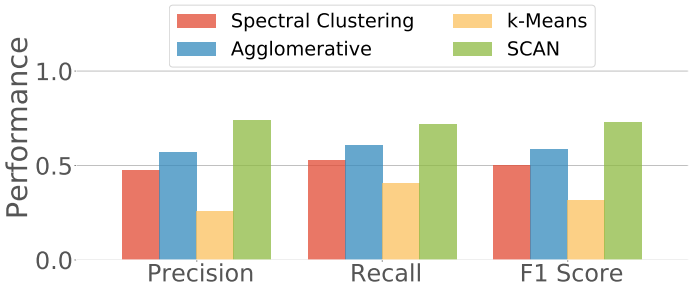
Following the convention of other automatic labelling work [7], we evaluate the performance of labelling in terms of the following metrics:

$$\begin{aligned} Precision &= \frac{TP}{TP + FP} \\ Recall &= \frac{TP}{TP + FN} \\ F_1 &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \end{aligned} \tag{3.7}$$

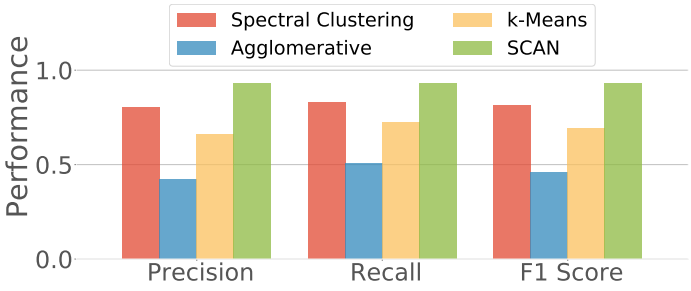
where TP, TN, FP, FN are true positive, true negative, false positive and false negative respectively. Each metric captures a different aspect of data association [136].

3.6.2.2 Competing Approaches

After defining the evaluation metrics, we compare SCAN against a number of competing approaches. They follow a similar two-step pipeline as discussed in Sec. 3.3. For the association step, they all use the Hungarian algorithm, but for the clustering step we consider



(a) Voice Labelling



(b) Face Labelling

Figure 3.6: Overall performance of SCAN on two labelling tasks.

three clustering algorithms namely: i) spectral clustering, ii) agglomerative clustering and iii) k-means. Each of them is a representative algorithm of a different category. Spectral clustering is a popular algorithm based on the spectral graph theory, the basic idea of it is to regard the biometric observations as the vertex and the feature similarity among objects as the weighted edge in order to transform the clustering problem into a graph partition problem [137]. Agglomerative clustering is widely used and is based on hierarchy. The key idea of agglomerative clustering is to construct the hierarchical relationship among biometric observations in order to attain clusters [138]. Lastly, k-means is one of the most established method based on partition. The core idea of k-means is to update the centre of biometric cluster which is represented by the centre of data points, by iterative computation and the iterative process will be continued until some criteria for convergence is met [139]. In the following graphs, for brevity, we refer to baselines by the name of the clustering algorithm that they use, but remind the reader that they also include the association step.

3.6.3 Results

We now proceed to the performance analysis which comprises two parts. The overall performance of SCAN on two datasets is reported in Sec. 3.6.3.1. We then study the robustness of SCAN under different conditions in Sec. 3.6.3.2.

3.6.3.1 Overall Performance

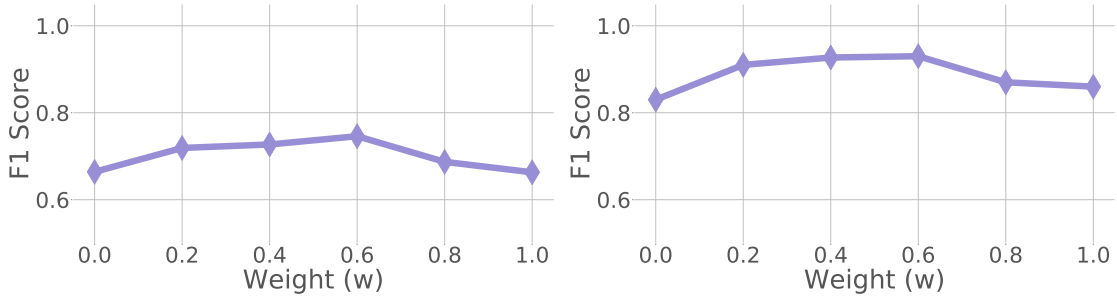
As shown in Fig. 3.6, SCAN is consistently more accurate than the competing approaches in all metrics. On average, SCAN has a F_1 score of 0.73 and 0.93 on the voice and face labelling tasks respectively. Considering that the biometric representation models are trained with out-of-domain data (i.e., different subjects in different conditions), this performance is remarkable. SCAN is found to balance well between precision and recall, which is very useful to different use cases. On the other hand, all two-step approaches are inferior to SCAN, regardless of which clustering algorithms are adopted. For *voice labelling*, the best two-step approach is agglomerative clustering which only achieves the F_1 score of 0.59. This result is $\sim 20\%$ worse than SCAN. At the same time, the best two-step approach on *face labelling* is spectral clustering and its F_1 score is 0.81, which is 13% lower than SCAN. As explained in Sec. 3.3.2, deviated biometric features and non-POI disturbances jeopardise the accuracy of the clustering step, which results in erroneous mapping between biometric clusters and digital attributes (IDs). SCAN fully exploits the context information as constraints to form clusters and tackle clustering and association at the same time, resulting in a significant increase in robustness.

Interestingly, compared to the face labelling, voice labelling is a much more challenging task. Under the similar experiment setups, we observed that SCAN is $\sim 21\%$ more effective in the experiment of *face labelling*. One reason is that, due to the dynamic nature of voices, e.g., different tones in various speeches, the pre-trained vocal features deviate more and the performance of SCAN gets significantly affected. Additionally, we found that the utterance segmentation module in *vocal SCAN* is not as effective as the face detection model in *facial SCAN*, which gives more noisy features and further challenges the merging of biometric samples. In fact, this observation implicitly coincides with our motivation that clustering and association should be performed in tandem to jointly combat the feature deviation.

3.6.3.2 Sensitivity Analysis

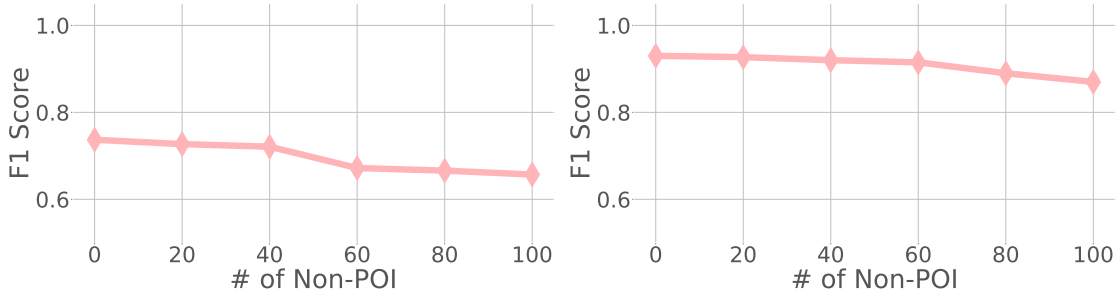
Now we focus on the sensitivity of SCAN to different parameter settings and conditions. The key performance metric here is the F_1 *Score*, which weights recall and precision equally.

Impact of Weights Between Linkage and Association Score. The first experiment is designed to explore the impact of the level of trust that we place on the digital observations. Recall that digital observations provide a prior on the attendance of participants in a session. In SCAN, such information is jointly optimised with the biometric clustering as



(a) Voice Labelling (b) Face Labelling

Figure 3.7: Impact of choice of ω.

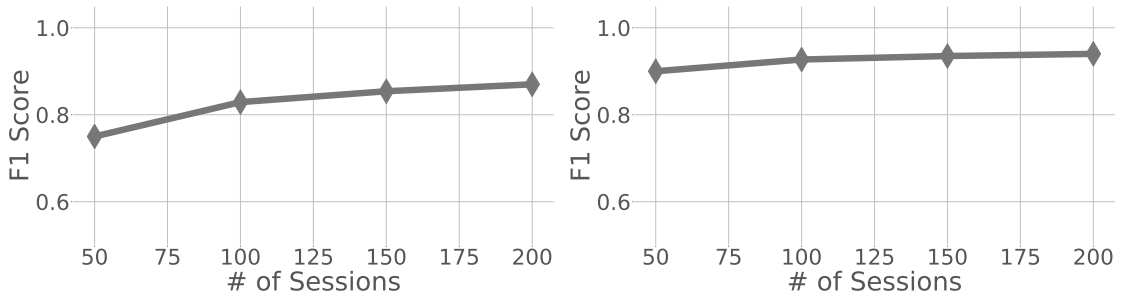


(a) Voice Labelling (b) Face Labelling

Figure 3.8: Impact of number of non-POI.

discussed in Eq. (3.1). The parameter ω in Eq. (3.1) indicates the interplay between the linkage score (derived from the biometric observations $\mathcal{X}$) and data association score (derived from the digital observations $\mathcal{L}$). Intuitively, when ω is set to a small value, the digital observations have little impact on clustering and SCAN mostly relies on the similarity of biometric features. This will of course have negative impact on the performance, since the valuable information encoded in digital observations is largely ignored. For instance, as shown in Fig. 3.7, when we set $\omega = 0$, SCAN only achieves very low F_1 score of 0.66 on the voice labelling task and 0.83 on the face labelling task. On the other hand, large ω tends to over-trust the digital observations, which can be sometimes ambiguous and leading to suboptimal F_1 score of 0.67 and 0.86 on two datasets respectively. Empirically we observe that a slightly skewed mix between linkage score and association score works well in practice for both datasets, with suitable values of ω lying in the range of $[0.4, 0.6]$. The exact optimal value is slightly higher for the face labelling task. This is because voice observations suffer from more feature deviation, and it therefore makes sense to trust the linkage score less.

Impact of number of non-POI. The next experiment aims to evaluate the impact of the



(a) Voice Labelling (b) Face Labelling

Figure 3.9: Impact of number of Sessions.

number of non-POI on the performance of SCAN. With the same set of POI, we gradually mix in more non-POI samples in both tasks. As shown in Fig. 3.8, SCAN is robust to the disturbances of non-POI, and is able to maintain comparable F_1 scores even facing an increasing number of non-POI. In particular, we found that when there is only a small number of non-POI (e.g., 40), SCAN achieves very similar performance to the no-disturbance cases. Clearly, the joint optimization in SCAN helped ignore the disturbing samples from non-POI in these cases. On the other hand, the performance on both tasks drops more quickly when the number of non-POI increases to 80. However, the degradation is graceful and the F_1 scores decrease less than 10% even there are 100 non-POI in both tasks. An explanation for this is that when the number of distinct subjects increases, the chance that two individuals have similar biometric observations goes up. As discussed in Sec. 3.4, developing a linkage tree of biometric observations is based on similarity comparison. The branches of the linkage tree will be incorrect if some of subjects have similar samples, which deteriorates the performance of SCAN.

Impact of number of Sessions. SCAN uses the context vectors of biometric clusters to associate them with digital observations. Its effectiveness is hence affected by the session-attendance diversity in subjects. The last experiment is designed to look into the impact of session attendance. To this end, we keep the number of POI and non-POI unchanged (50 and 20 respectively) and vary the number of sessions participated by these groups of speakers, simulating the growing number of sessions. As depicted in Fig. 3.9, the increase of sessions benefits the labelling results on both tasks. Compared with the initial case where only 50 sessions are given, the F_1 scores were improved by $\sim 12\%$ and $\sim 5\%$ when four times of the sessions are provided. The improvement in performance is easy to comprehend. As the number of sessions increases, the subjects become more distinguishable since more diversity is introduced into context vectors. On the other hand, this improve-

ment becomes marginal when too many sessions are given. It implies that using more sessions in SCAN is helpful but with limited impact. The fundamental issue still lies in the out-of-domain feature deviations, which will be addressed in next chapter.

3.7 Related Work

Cross-modality Matching: Cross-modal matching has received considerable attention in different research areas. Methods have been developed to establish mappings from images [140, 141, 142] and videos [143] to textual descriptions (e.g., captioning), developing image representation from sounds [144, 145], and generating visual models from text [146]. In cross-modality matching between images and radio signals, however, related work is very limited and all dedicated to trajectory tracking of humans [147, 148, 149]. The field of recognizing speaker identities from wireless signals is still in its infancy.

Data Association: Our proposed cross-modal labelling approach is also related to data association methods. Given a track of sensor readings, data association aims to figure out inter-frame correspondences between them. Data association is widely used in radar systems, when tracking blips on a radar screen [150], as well as object monitoring of surveillance systems [151]. To find inter-frame correspondences, various Bayesian filtering approaches have been developed, including Nearest-Neighbour Data Association Filter [83], Probabilistic Data Association Filter [152], Joint Probabilistic Data Association Filter [84] and Multiple Hypothesis Tracking [85]. Unlike SCAN, these approaches rely on state-based models, where both sensors are observing temporally evolving systems. For instance, in our approach, detecting a MAC address does not imply that someone will be speaking at that exact instant.

Truth Discovery: Finally, this work aims to discover knowledge from noisy sensor data, which shares the similar idea with the truth discovery in social sensing [153, 154] and accuracy estimation [8] techniques. Those approaches typically assume that sensor data is homogeneous but comes from multiple sources, and consider the Expectation-Maximization (EM) framework to jointly estimate the reliability/accuracy of the sources and sensor measurements at the same time. However, SCAN focuses on using heterogeneous sensor data (i.e., biometric and digital observations) from different sensing modalities to learn their associations. A promising direction is to incorporate the truth discovery/accuracy estimation step on top of SCAN, and use the learned trustworthiness to adjust the behaviour of SCAN accordingly.

3.8 Summary

As it has been shown in this chapter, it is possible to automatically label biometric data from noisy heterogeneous data without manual labelling or enrolment. We demonstrated that the baseline, a two-stage technique, is very sensitive to errors both in terms of deviated out-domain features and disturbances of non-POI. By coupling the clustering and association process, SCAN is able to robustly reject noise, with up to over 20% improvement in the face of noisy data.

Despite its potential to make cross-modality association robust to perturbation, the performance of SCAN is sometimes insufficiently accurate, e.g., in the case of vocal SCAN. Recall that in SCAN, biometric features are extracted by a model pre-trained on out-of-domain labelled datasets. Although these pre-training datasets are publicly available and easy to obtain, their samples may significantly deviate from our biometric observations due to domain differences. On the downside, SCAN is able to alleviate the impact of feature deviation in clustering but it cannot eradicate these domain differences. As a consequence, the labelled biometric samples by SCAN can be noisy. Obliviously, it will be risky to directly use these noisy labels as supervision signals when training a recognition model for identifying (in-domain) local subjects. The proposed approaches in Chapter 4 aim at addressing this issue.

Chapter 4

Iterative Adaptation

4.1 Introduction

Chapter 3 describes a novel cross-modality association algorithm that automatically labels biometric features with ambient digital observations. An assumption we implicitly made in the previous chapter is that observations of digital attributes truthfully reveal the presence of subjects (see Sec. 3.2). This assumption is largely valid when digital observations are directly retrieved from the meta information of sessions, e.g., a cast of characters in a television program. However, in many realistic scenarios, digital observations are collected through physical sensors and they may not reflect the true events [8]. For example, people sometimes forget to carry their smartphone, and the lack of detected wireless identifiers in this case may incorrectly indicate absence whereas their biometric observations such as facial images are still captured by biometric sensors. Additionally, when using the device ID (e.g., a wireless identifier) of subjects as a digital attribute, the presence or absence of the device is determined by the received signal strength (RSS). However, these collocations are not necessarily accurate. For instance, the presence of a smartphone in the room is determined by a threshold on the RSS value, which effectively defines a geofence, i.e., an observation model. Due to device heterogeneity, this observation model is defined on a case-by-case basis and largely depends on the manufacturers. Fig. 4.1 illustrates the impact of device heterogeneity on RSS. As we cannot exhaustively obtain a suitable geofence model for each device, the detected collocations are uncertain and may generate attendance inconsistency between biometric and digital observations. Unfortunately, the association part proposed in `SCAN` is sensitive to this inconsistency. Last but not least, as discussed in Sec. 3.8, biometric features are extracted by a model pre-trained on out-of-domain labelled datasets. Although these pre-training datasets are publicly available and easy to obtain, their samples may significantly deviate from our biometric observations due to domain differences. The performance of `SCAN` is limited by this noisy (pre-trained) feature similarity.

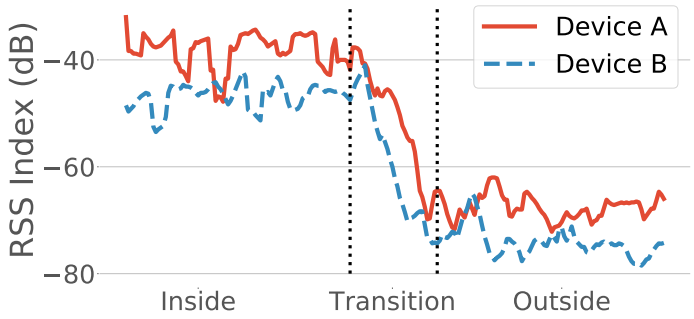


Figure 4.1: The impact of device heterogeneity on RSS index. The difference of RSS index of two different devices can be as large as 8dB. A universal room-specific geofence value will be inherently noisy and inaccurate.

In order to enhance labelling robustness, we observed that updating the models of biometric representations and digital observations is crucial. We therefore propose `AutoTune`, a cross-modality adaptation framework that iteratively labels heterogeneous data and updates observations. The key idea behind `AutoTune` is that correctly labelled biometric data can be used to feedback into the adaptation of identity recognition models. Through this iterative adaptation, the inconsistencies between biometric and digital observations are then gradually reduced.

However, this adaptation is non-trivial as we also face the issue of imperfect labels. This is because the labels are obtained through cross-modality association and they inevitably contain mislabelled samples. When the amount of mislabelled data is non-negligible, class "pollution" will undermine the adaptation. If such label noise accumulates to the point of counteracting the benefits of in-domain labels, knock-on effects of iterative learning might worsen the association performance of `SCAN`. `AutoTune` hence adopts a probabilistic framework to diminish the risks caused by imperfect labels. It leverages the probabilistic labels generated from soft voting on possible association results and then adapts models with these labels. In summary, the contributions in this chapter are:

- We create `AutoTune`, a novel cross-modality learning pipeline to simultaneously label biometric observations in the wild and adapt the biometric recognition and digital observation models to new environments. The key idea is to repeat the label association and model update in tandem.

- To handle imperfect labels and out-of-domain extracted features, we propose a novel probabilistic framework in `AutoTune`. Concretely, this probabilistic consideration leads to two different adaptation strategies, depending on whether the digital observation model is modifiable or not: i) to adapt biometric recognition under uncertainty,

we design a new stochastic center loss to enhance the robustness of fine-tuning the representation model; ii) to adapt the observation model of digital attributes, we design a soft geofence model that is able to tolerate uncertain labels.

- We deployed AutoTune in three real-world environments with two different biometric modalities. Experimental results demonstrate that AutoTune is able to significantly outperform the best competing approach in all metrics. Using the adapted models by AutoTune also ports reliable online identification and localization performance.

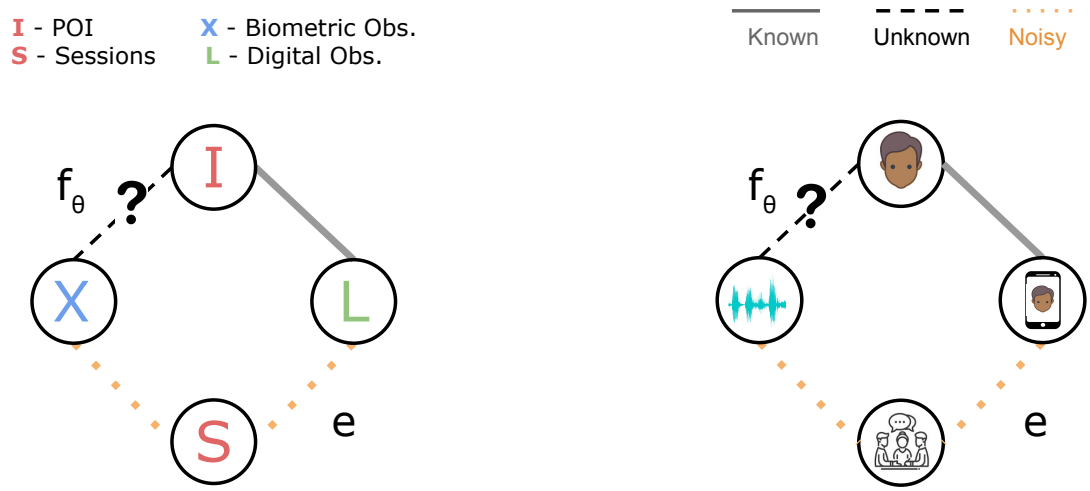
The rest of this chapter is organized as follows. Sec. 4.2 overviews the generic framework of AutoTune. In Sec. 4.3, we introduce cross-modality labelling, based on the SCAN solution proposed in the previous chapter. In Sec. 4.4, a generic adaptation framework is proposed to iteratively enhance the consistency between digital and biometric observations. Sec. 4.5 provides the implementation details. Sec. 4.6 evaluates AutoTune, and compares its performance with competing approaches. Sec. 4.7 surveys the related work, while Sec. 4.8 concludes this chapter.

4.2 Overview

4.2.1 Problem Definition

We firstly go through the same notations as introduced in SCAN. In particular, we specify *sessions* $\mathcal{S} = \{s_j | j = 1, 2, \ldots, g\}$, *POI* $\mathcal{I} = \{i_j | j = 1, 2, \ldots, m\}$, *digital observations* $\mathcal{L} = \{l_j | j = 1, 2, \ldots, m\}$, and *biometric observations* $\mathcal{X} = \{x_j | j = 1, 2, \ldots, n\}$ which have been defined in Sec. 3.2. AutoTune also considers a deep representation model f_θ for biometric feature extraction. This representation model is pre-trained on public datasets that contain no POI, but is suitable for clustering as it is supervised by metric losses, e.g., triplet loss in learning [117, 3].

Different from the problem setting in SCAN, the digital observations considered in AutoTune are uncertain and loosely linked with sessions. This chapter is dedicated to addressing this issue. We introduce a new term: *digital observation model* e, which broadly refers to a model that interprets sensor measurement to digital observations. For instance, when employing WiFi identifiers as the digital attribute, a WiFi sniffer is usually needed to detect users' smartphones. The sniffer then uses a geofence model to determine the device's attendance status in a session based on the received signal strength (RSS) in the



(a) Abstraction (b) Instance

Figure 4.2: Relationship between biometric and digital observations. (a) Abstraction. (b) Instance. Given the noisy biometric observations and *uncertain digital observations*, `AutoTune` aims to accurately label biometric observations and reduce the inconsistency between two types of observations. f_θ and e are biometric representation model and digital observation model respectively. Compared with `SCAN` (see Fig. 3.1), both links between observations and sessions in `AutoTune` are uncertain.

sniffed WiFi packets. The geofence model in the above example is a digital observation model e as it infers the relationship between a digital attribute (wireless identifier) and sessions based on the measurement (RSS) in the physical world. Notably, the digital observation e model is not always a geofence model and is sometimes non-modifiable. For example, when employing the cast of characters as the digital attribute. However, we can still update digital observations by directly using their associated biometric observations as will be discussed in Sec. 4.4.2.1. In summary, the *problem* in this chapter is how to improve the robustness of cross-modality labelling when both biometric and digital observations are noisy. Fig. 4.2 provides a simple schematic illustration and an example of this problem.

4.2.2 `AutoTune` Workflow

As illustrated in Fig. 4.3, `AutoTune` consists of three main modules:

- *Heterogeneous Sensing*. This module contains two heterogeneous sensors that collect biometric and digital observations respectively. After pre-processing, e.g., biometric segmentation or face detection (see Sec. 3.5), biometric observations are acquired and fed into the next module. Opportunistically, a digital observation model is used to interpret digital attributes from the sensor measurements.

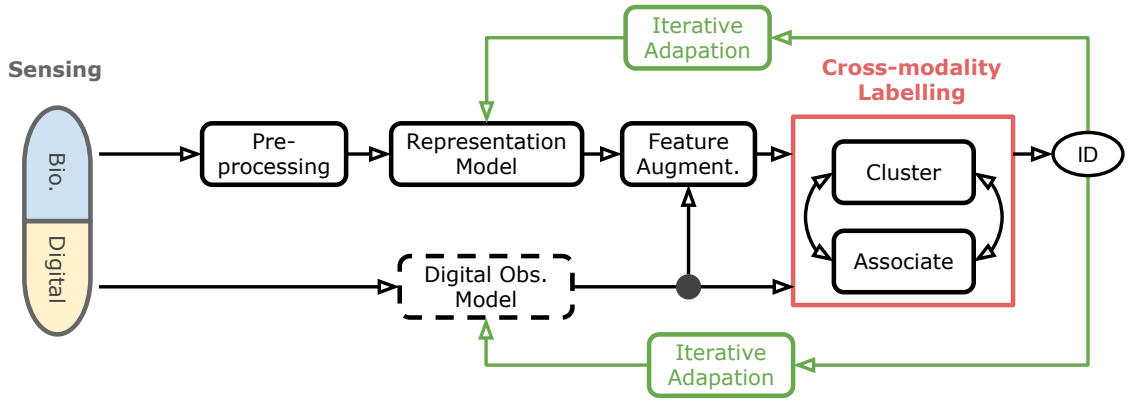


Figure 4.3: Workflow of `AutoTune`. The adaptation module reuses correctly labelled biometric data to feedback into the adaptation of models to improve the labelling performance in subsequent iterations. Notably, the adaptation of the digital observation model e is only performed if possible.

- *Cross-modality Labelling*. Initialized with a pre-trained representation model, this module extracts features from the pre-processed data and automatically labels the biometric samples. For example, Using MAC addresses detected via a collocated WiFi sniffer to label the voices recorded by a microphone. With a known table, MAC addresses can be mapped to user identities, i.e., labels. It is important to notice that this module extends `SCAN` to a probabilistic version that can assign confidence scores for associated labels.

- *Iterative Adaptation*. The adaptation module is the key to `AutoTune`. The idea is that correctly labelled biometric data can be used to improve the labelling performance in subsequent iterations. Concretely, this module uses the probabilistic labels given by the labelling module and iteratively fine-tunes the biometric representation model f_θ. The adapted model f_θ is able to better recognize in-domain subjects. At the same time, this module also iteratively updates the digital observation model e to enhance the consistency between biometric and digital observations. A side product of this update is a set of personalized models that can robustly interpret individual's presence from noisy sensor measurements.

As the sensing module is solely an implementation, we defer to give its details until entering Sec. 4.5. The subsequent two sections (Sec. 4.3 and Sec. 4.4) will introduce the labelling and iterative adaptation modules in `AutoTune`.

4.3 Cross-modality Labelling

The cross-modality labeling module is inherited from the approaches proposed in Chapter 3. It includes biometric clustering across multiple sessions and uses the context vector as a linkage to associate biometric observation with digital observation. However, `AutoTune` modifies `SCAN` by introducing *feature augmentation* and *probabilistic labelling*, which allows us to mitigate noisy labels.

4.3.1 Feature Augmentation

The augmented feature of a biometric observation is hybrid and comprises both biometric representation and session attendance. Biometric representation is obtained through a feature extractor, as in Sec. 3.3.2. On the other hand, session attendance describes the present subjects based on the digital observations. We will discuss how to construct a hybrid feature in Sec. 4.3.1.1, and then introduce the similarity measure for this new feature.

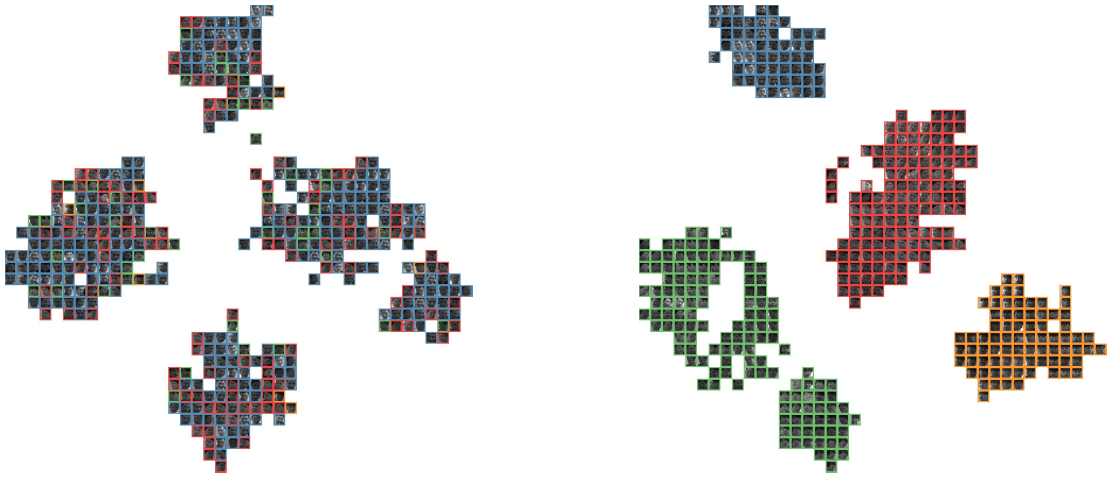
4.3.1.1 Hybrid Features

Biometric features extracted by a pre-trained representation model are less discriminative than the model trained on in-domain data. Taking facial images as an example, Fig. 4.4 shows that when using a pre-trained FaceNet as the feature extractor, facial features in a new domain are not discriminative. As a result, samples of the same subject will scatter in different clusters and "pollute" the association step. In order to minimize feature deviations for similar images, we propose to augment features with attendance information when comparing biometric samples. Formally, we denote the digital attendance vector as $\mathbf{u}_k = (u_k^1, u_k^2, \ldots, u_k^m)$, where $u_k^j = 1$ if a digital attribute l_j is detected in session s_k. In this way, we are able to construct the following hybrid feature $\widetilde{\mathbf{z}}_i$ for a biometric sample x_i collected in the session s_k:

$$\widetilde{\mathbf{z}}_i = [\mathbf{z}_i, \mathbf{u}_k] \tag{4.1}$$

where $\mathbf{z}_i$ is the extracted feature from the biometric observation x_i, through a biometric representation model f_θ.

The key intuition behind the above augmentation is that session attendance information of digital attributes can bootstrap the merging of biometrics into cohesive clusters. Recall that digital observations already reveal subjects' attendance in sessions, and the collected biometric observations probably contain the biometric samples of these subjects. The intersection of subjects in distinct sessions can be used as a prior that guides the merging of biometric data. For example, if there are no shared MAC addresses sniffed in two sessions,



(a) Pre-trained FaceNet on VGG datasets (b) Fine-tuned Model via `AutoTune`

Figure 4.4: Similarity comparison of the biometric observations of four subjects in a new environment based on their features extracted by VGG-pre-trained FaceNet (left) and adapted model by `AutoTune` (right) respectively. Features are projected to 2d plane via t-SNE [4]. Images belonging to the same subject identities are framed with the *same* colors. It can be seen that face features provided by the pre-trained model are much less discriminative than the adapted model of `AutoTune`, with significant color overlap.

then it is very likely that the collected biometric samples in these two sessions should lie in different clusters.

4.3.1.2 Feature Similarity

After deriving the hybrid features, we introduce a new measure to compare biometric samples collected in different sessions. Given a biometric observation x_i captured in session s_k (i.e., $\widetilde{\mathbf{z}}_i = [\mathbf{z}_i, \mathbf{u}_k]$) and an observation x_j captured in session s_p (i.e., $\widetilde{\mathbf{z}}_j = [\mathbf{z}_j, \mathbf{u}_p]$), the likelihood that two cross-session biometric observations belong to the same subject depends on two factors: i) the similarity of their feature representation between $\mathbf{z}_i$ and $\mathbf{z}_j$; and ii) the overlap ratio between the attendance observations in their corresponding sessions $\mathbf{u}_k$ and $\mathbf{u}_p$ respectively. The resulting joint similarity is a log likelihood $log(\mathbf{Pr}(x_i = x_j))$ defined as follows:

$$\begin{aligned} \mathbf{Pr}(x_i = x_j) &\propto \frac{exp(\beta * |\mathbf{u}_k \otimes \mathbf{u}_p|)}{exp(\beta * |\mathbf{u}_k \oplus \mathbf{u}_p|)} * exp(-D(\mathbf{z}_i, \mathbf{z}_j)) \\ log(\mathbf{Pr}(x_i = x_j)) &\propto \beta * \underbrace{(|\mathbf{u}_k \otimes \mathbf{u}_p| - |\mathbf{u}_k \oplus \mathbf{u}_p|)}_{\text{attendance similarity}} - \underbrace{D(\mathbf{z}_i, \mathbf{z}_j)}_{\text{feature distance}} \end{aligned} \tag{4.2}$$

where $\otimes$ and $\oplus$ are element-wise AND and OR, and $|\cdot|$ here is the L^1-norm. z represent the features transformed by the biometric representation model and D is a distance measure

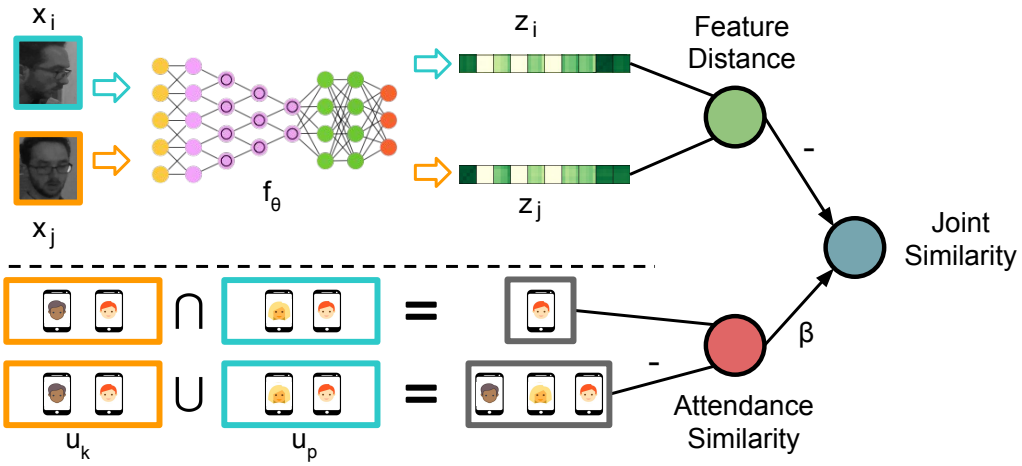


Figure 4.5: Feature similarity. The final similarity is jointly determined by attendance similarity derived from session attendance vectors u and distance of biometric features $\mathbf{z}$, which are transformed by f_θ. x_i and x_j are two biometric observations drawn from session s_k and s_p respectively. u_k and u_p are the attendance vectors of these sessions. Detailed explanations can be found in Sec. 4.3.1.2.

(e.g., Euclidean distance) between biometric features. β, analogous to the regularization parameter in composite loss functions, is a hyper-parameter that controls the contributions of the attendance assistance and feature similarity. The above derivation is inspired by the principle of Jaccard coefficients[155], with the difference lying in the log function. The rationale behind the term $|\mathbf{u}_k \oplus \mathbf{u}_p|$ is that the greater the number of different subjects attending a set of sessions, the more uncertain it is that any two observations drawn from these sessions will point to the same subject. In contrast, when the intersection $|\mathbf{u}_k \otimes \mathbf{u}_p|$ is large, the chance that these two observations point to the same subject will also become high. This joint similarity can also be explained from a Bayesian perspective. The attendance similarity of two sessions can serve as a prior that two cross-session observations belong to the same subject and the feature similarity can be seen as the likelihood. Together they determine the posterior probability that two cross-session observations fall into the same cluster. Note that such hybrid features do not affect the grouping of two biometric samples observed within the same session. Fig. 4.5 illustrates the above similarity measure.

4.3.2 Probabilistic Labelling via Soft Voting

Although feature augmentation alleviates the issue of non-discriminative features in merging biometric observations, the associated labels are deterministic and can still be imperfect. Inspired by ensemble methods [68] that combine predictions of several base estimators, we consider using multiple cross-modality labellers and average their results to

mitigate the impact of noisy labels. Recall that the parameter β in Eq. (4.2) is important for labelling as it weighs biometric similarity and attendance similarity. Intuitively, the choice of β is case dependent, yet a suitable β is difficult to be determined a priori. We hence vary β to create different labelling settings and then vote on top of several association results. Notably, deterministic labels after voting cannot comprehensively summarize the uncertainty in labels. We instead adopt soft voting, whereby to attain the needed probabilistic labels. Concretely, we set β, from 10^{-3} to 10^{-1} and run the labelling block for multiple rounds. The number of rounds depends on the step interval of increasing β. Then, for every sample, its assigned digital attribute is finalized by *soft voting* on individual association results computed with different β. Fig. 4.6 illustrates the process of soft voting. Formally, we introduce a probability vector $\mathbf{y}_i = (y_{i,1}, y_{i,2}, \ldots, y_{i,m})$. For instance, $y_{i,j} = 0.4$ indicates 40% of associations assigning the digital attribute l_j to the biometric sample x_i.

4.4 Iterative Adaptation

We are now in a position to introduce the iterative adaptation module. At every iteration, `AutoTune` updates the previous biometric representation model f_θ^τ to $f_\theta^{\tau+1}$. At the same time, the digital observation model e^τ also gets updated to $e^{\tau+1}$ that diminishes the session-attendance inconsistency between heterogeneous observations.

4.4.1 Adaptation of Biometric Representation

Based on the new similarity measure defined in Eq. (4.2), `AutoTune` uses cross-modality labelling methods to assign biometric observations with their respective digital attributes. Due to its superior performance and robustness (see Sec. 3.6), `SCAN` is a natural choice for `AutoTune` in labelling. However, even `SCAN` cannot give perfect labels and using noisy labels for iterative adaptation is risky because most DNNs of biometric representation learning are ill-equipped to deal with uncertain labels. We therefore consider a probabilistic learning framework in `AutoTune`.

4.4.1.1 Discriminative Biometric Representation Learning

We start by briefly introducing the general framework of biometric representation learning. Biometric representation learning optimizes a representation loss $\mathcal{L}_R$ to enforce the learnt features to be as discriminative as possible. Strong discrimination bears two properties: inter-class dispersion and intra-class compactness. Inter-class dispersion pushes biometric

	l_1	l_2	l_3
x_1	0	1	0
x_2	0	1	0
x_3	1	0	0
x_4	1	0	0

(a) $\beta = 10^{-3}$

	l_1	l_2	l_3
x_1	0	1	0
x_2	0	0	1
x_3	1	0	0
x_4	0	1	0

(b) $\beta = 10^{-2}$

	l_1	l_2	l_3
x_1	0	1	0
x_2	0	0	1
x_3	1	0	0
x_4	1	0	0

(c) $\beta = 10^{-1}$

	l_1	l_2	l_3
x_1	0	1	0
x_2	0	0.33	0.67
x_3	1	0	0
x_4	0.67	0.33	0

(d) Probabilistic labels after soft voting

Figure 4.6: An example of soft voting on the labelling results with 3 different β. x - biometric samples;l - digital attribute.

samples of different subjects away from one another and the intra-class compactness pulls the biometric samples of the same subject together. Both of the properties are critical to biometric recognition. At iteration τ, given the current labels y_i^τ for the ith biometric sample x_i and the transformed features $\mathbf{z}_i^\tau = f_\theta^\tau(x_i)$, the representation loss $\mathcal{L}_R$ is determined by a composition of the softmax loss and center loss:

$$\begin{aligned}\mathcal{L}_R &= \mathcal{L}_{softmax} + \mathcal{L}_{center} \\ &= \underbrace{\sum_i -log(\frac{e^{W_{y_i^\tau}^T \mathbf{z}_i^\tau + \mathbf{b}_{y_i^\tau}}}{\sum_{j=1}^m e^{W_{l_j}^T \mathbf{z}_i^\tau + \mathbf{b}_{l_j}}})}_{\text{softmax loss}} + \underbrace{\sum_i \frac{\lambda}{2}||\mathbf{z}_i^\tau - \mathbf{o}_{y_i^\tau}||^2}_{\text{center loss}}\end{aligned} \tag{4.3}$$

where W and b are the weights and bias terms in the last fully connected layer of the pre-trained model. $\mathbf{o}_{y_i^\tau}$ denotes a centroid feature vector by averaging all feature vectors with the same identity label y_i. The center loss $\mathcal{L}_{center}$ explicitly enhances the intra-class compactness while the inter-class dispersion is implicitly strengthened by the softmax loss $\mathcal{L}_{softmax}$ [156]. λ is a hyper-parameter that balances the above two losses.

4.4.1.2 Stochastic Center Loss

The center loss $\mathcal{L}_{center}$ in Eq. (4.3) is shown to be helpful to enhance the intra-class compactness [156]. However, we cannot directly adopt it for fine-tuning as computing the centers requires explicit labels (see Eq. (4.3)) of observations, but the association steps above only provide probabilistic ones through soft labels. To solve this, we propose a new loss termed *stochastic center loss* $\mathcal{L}_{stoc}$ to replace the center loss. Similar to the core idea of fuzzy sets [157], we allow each biometric observation to belong to more than one subject. The membership grades indicating the degree to which an observation belongs to each subject can be directly retrieved from the soft labels and the stochastic center $\mathbf{o}_k^\tau$ for the k-th identity is given as:

$$\mathbf{o}_k^\tau = \frac{\sum_i^n \mathbf{z}_i^\tau * y_{i,k}^\tau}{\sum_i^n y_{i,k}^\tau} \tag{4.4}$$

This gives the stochastic center loss as follows:

$$\mathcal{L}_{stoc} = \sum_i \sum_k y_{i,k}^\tau * ||\mathbf{z}_i^\tau - \mathbf{o}_k^\tau||^2 \tag{4.5}$$

We leave the softmax loss $\mathcal{L}_{softmax}$ unchanged as in Eq. (4.3), because the soft labels are compatible with the computation of cross-entropy. Then the new representation loss to minimize is:

$$\mathcal{L}_{new} = \mathcal{L}_{softmax} + \mathcal{L}_{stoc} \tag{4.6}$$

`AutoTune` updates the model parameters θ^τ to $\theta^{\tau+1}$ based on the gradients of $\nabla_\theta \mathcal{L}_{new}$, which are calculated via back prorogation of errors. Algorithm 2 summarizes the workflow of this module.

4.4.2 Update of Digital Observation

Session-attendance information observed by digital sensors can be error-prone and inconsistent with the one inferred by the biometric observations. For instance, the device presence observations made by WiFi sniffing are noisy because the WiFi signal of a device is opportunistic and people do not carry/use their devices all the time. Based on the results

Algorithm 2: Cross-modality Labelling at τ-th iteration.

Input: Biometric representation model $f_\theta^{(\tau)}$, digital observations $\mathcal{L}$, biometric observations $\mathcal{X}$, number of sessions g, number of POI m, context vector $\mathbf{r}_{l_{1,\ldots,m}}^{(\tau)}$, digital session-attendance vector $\mathbf{u}_{1,\ldots,g}^{(\tau)}$

Output: Soft Labels $\mathcal{Y}$

// See Sec. 4.3.1

$\mathcal{Z}^{(\tau)} = f_\theta^{(\tau)}(\mathcal{X})$

$\widetilde{\mathcal{Z}}^{(\tau)} = \texttt{feature\_augmentation}(\mathcal{Z}^{(\tau)}, \mathcal{U}^{(\tau)})$

$T^{(\tau)} = \texttt{linkage\_tree}(\widetilde{\mathcal{Z}}^{(\tau)})$

// See Sec. 4.3.2

for $\beta \leftarrow 10^{-3}$ *to* 10^{-1} **do**

  $\mathbf{A}_\beta^{(\tau)} = \texttt{SCAN}(m, \beta, \mathbf{r}_{l_{1,\ldots,m}}^{(\tau)}, T^{(\tau)})$ // $\mathbf{A}_\beta^{(\tau)}$ are hard labels given a β

end

$\mathcal{Y}^{(\tau)} \leftarrow \texttt{soft\_voting}(\mathbf{A}_\beta^{(\tau)}, \mathcal{L})$

of the cross-modality labelling, we now have the opportunity to update our belief on which users attended each session and mitigate inconsistency. We consider two cases of updating digital observations, depending on whether the digital observation model e is modifiable or not. In particular, a geofence model is studied as the modifiable digital observation model. Geofence is a simple localization model that determines inside and outside events. We note that our technique could be applied to other modifiable observation model as well.

4.4.2.1 Inconsistency Mitigation

We start with the most general case where the digital observation model e is fixed. After the soft voting step (see Sec. 4.3.2), each biometric sample is associated with a probability vector, whose elements denote the probability that this biometric sample corresponds to a particular digital attribute. By averaging the probability vectors of all biometric samples that have been drawn from the same session s_k, and normalizing the result, we can estimate the digital attendance of this session. The elements of the resulting digital attendance vector $\widehat{\mathbf{u}}_k^\tau$ denote the probabilities of different digital attribute appearing in session s_k.

We can now use $\widehat{\mathbf{u}}_k^\tau$ as a reference signal to update our previous digital attendance vector $\mathbf{u}_k^\tau$ as follows:

$$\mathbf{u}_k^{\tau+1} = \mathbf{u}_k^\tau - \gamma \cdot (\mathbf{u}_k^\tau - \widehat{\mathbf{u}}_k^\tau) \tag{4.7}$$

where γ is a pre-defined parameter that controls the update rate of digital attendance. In principle, a large update rate will speed up the convergence rate, at the risk of missing the

optima. AutoTune sequentially repeats the steps of cross-modality labelling and inconsistency mitigation, until the changes in the digital attendance observations are negligible (≤ 0.1 in our case). Fig. 4.7 illustrates an example of inconsistency mitigation when we apply AutoTune to the observations of facial images and WiFi identifiers.

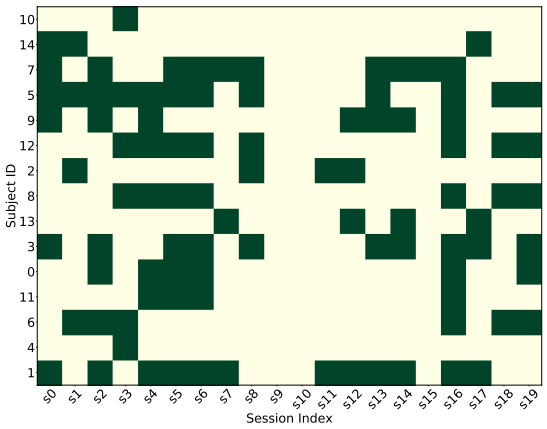
4.4.2.2 Observation Model Customization

Now we consider another case, where we update a modifiable digital model e. In particular, we customize a geofence model to address the issue caused by device heterogeneity. As shown in Fig. 4.8, the RSS distribution of two different devices can be very distinct even if they are physically co-located. Due to such device heterogeneity, a universal geofence model is usually ineffective when there are wireless identifiers of many different devices. In this section, we propose an iterative customization method for individual geofence model to improve the reliability of digital observation. Notably, though we focus on the case of wireless identifier, it is intuitive that our method can generalize to other types of digital attributes such as smartphone IMEI, where the observation model is the cellular geofence.

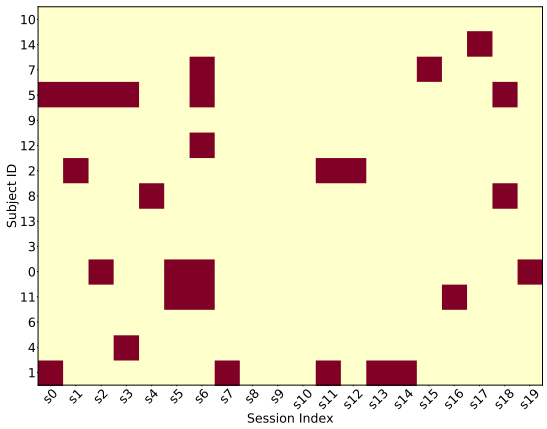
Geofence Initialization. We start by initializing a common geofence value, ζ_{init} for all devices. Then in any session, the digital sensing module of AutoTune sniffs WiFi packets and group RSS readings together if they share the same source MAC addresses, namely from same devices. However, due to environmental dynamics, RSS values can vary a lot even if the device is physically stationary. In order to reduce the impact of RSS variations [158], we use the median RSS index v_k^j to summarize all RSS readings of a device (i.e., the digital attribute) l_j collected in the session s_k. Finally, by comparing v_k^j to the initialized geofence value ζ_{init}, we can distinguish whether device l_j is presented at a particular session s_k and construct its context vector $\mathbf{r}_{l_j}$ and digital attendance vector $\mathbf{u}_k$ for cross-modality labelling, as introduced in Sec. 3.4.

Iterative Model Customization. After the first iteration of labelling, AutoTune is able to customize the geofence model with labelled biometric samples. The rationale is that the geofencing of biometric data is much easier to determine, e.g., human voices can be well insulated if a door is closed, and leads to a more reliable spatial relationship between the biometric observations and sessions. Therefore, once a device is associated with a biometric cluster, we use the biometric context vector to update the attendance information of associated devices.

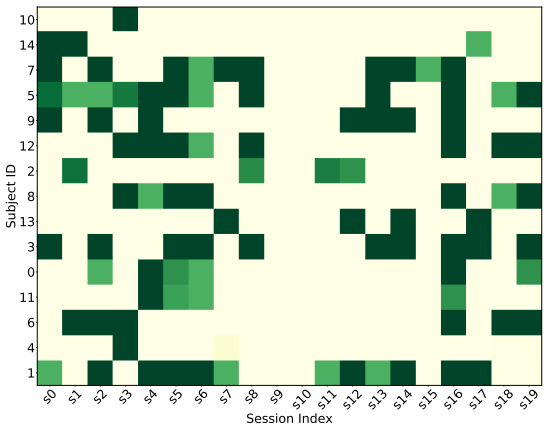
Given an associated pair of device l_j and biometric cluster t_i, we re-group the RSS readings of a wireless ID l_j based on all participating sessions inferred from the cluster context vector $\mathbf{r}_{t_i}$. But recall that biometric clusters can be impure, especially in early iterations due to lack of sufficient representation adaptation. Consequently, the context vector



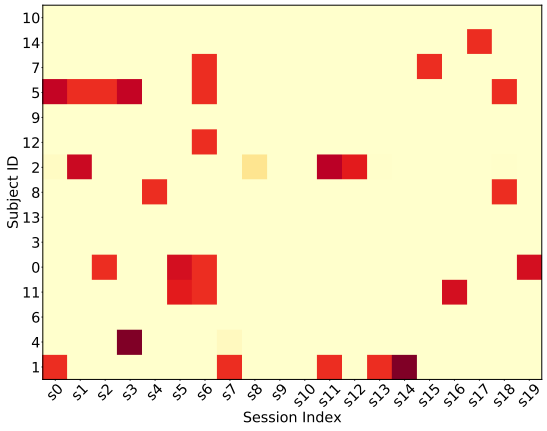
(a) Initial digital observations on session attendance.



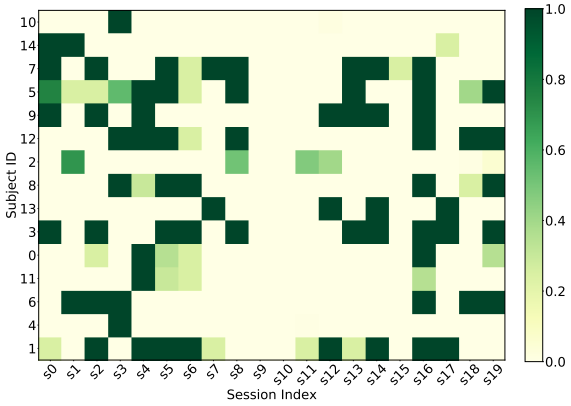
(b) Initial inconsistency on session attendance.



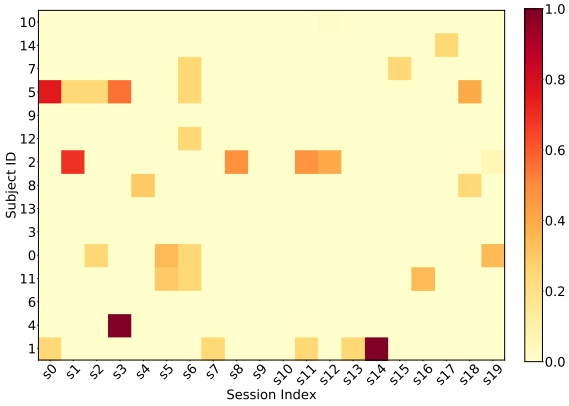
(c) Digital observations after 1st iter.



(d) Attendance inconsistency after 1st iter.

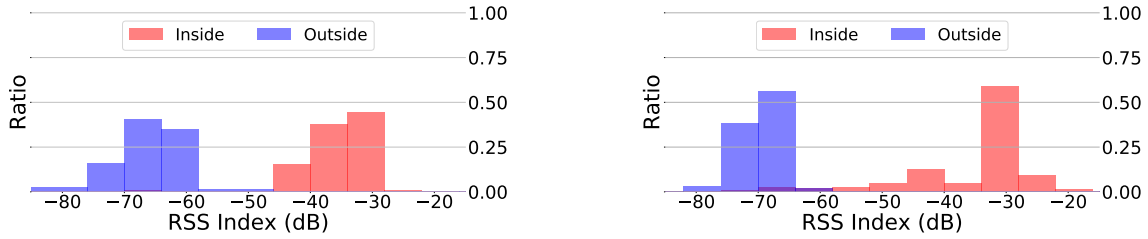


(e) Digital observations after 3rd iter.



(f) Attendance inconsistency after 3rd iter.

Figure 4.7: An example of observation update in `AutoTune` (Sec. 4.4.2), with a 15-subject subset of 20 sessions. **Left**: Digital observations on session attendance of different subjects; **Right**: Inconsistency of attendance observations in different sessions. It can be seen that the attendance inconsistency between biometric and digital observations is iteratively reduced.



(a) Device A

(b) Device B

Figure 4.8: Distributions of RSS when two different devices are placed inside and outside the target environment. Device heterogeneity influences the ability to detect a device's presence in a target environment.

of associated clusters might be inaccurate, which may significantly affect a deterministic update of geofence. To reduce the risks incurred by impure biometric clusters, we propose to use a probabilistic geofence, under the Gaussian noise model. We then examine the associated biometric context vector to group the RSS readings of a device l_j across all participated sessions and fit these in-room RSS readings to a normal distribution, denoted by $e^j_{in} \sim \mathcal{N}(\mu_{in}, \sigma^2_{in})$. Additionally, we fit another normal distribution of the RSS readings sniffed in absent sessions inferred from the biometric context vector. We denote this distribution by $e^j_{out} \sim \mathcal{N}(\mu_{out}, \sigma^2_{out})$. Then, the presence of a device l_j in session s_k is geofenced by a normalized probability:

$$p^j_k = \frac{p(v^j_k|e^j_{in})}{p(v^j_k|e^j_{in}) + p(v^j_k|e^j_{out})} \tag{4.8}$$

The new context vector of device l_j is a probabilistic vector, where each element represents the presence probability in a particular session:

$$\mathbf{r}^\tau_{l_j} = (p^j_1, p^j_2, \ldots, p^j_g) \tag{4.9}$$

Similarly, a new digital attendance vector for session s_k is updated as follows:

$$\breve{\mathbf{u}}^\tau_k = (p^1_k, p^2_k, \ldots, p^m_k) \tag{4.10}$$

Then the digital attendance information in the next iteration can be re-estimated by integrating the predictions from customization observations and biometric clusters as follows:

$$\mathbf{u}^{\tau+1}_k = \mathbf{u}^\tau_k - \gamma \cdot (\breve{\mathbf{u}}^\tau_k - \widehat{\mathbf{u}}^\tau_k) \tag{4.11}$$

where $\widehat{\mathbf{u}}^\tau_k$ is the attendance prediction by biometric clusters as in Eq. (4.7).

Algorithm 3 provides the overall workflow of iterative adaptation.

Algorithm 3: Iterative Adaptation at τ-th iteration

Input: Biometric representation model $f_\theta^{(\tau)}$, digital observations $\mathcal{L}$, biometric observations $\mathcal{X}$, number of sessions g, number of POI m, digital session-attendance vector $\mathbf{u}_{1,\ldots,g}^{(\tau)}$, soft labels $\mathcal{Y}^{(\tau)}$, stopping threshold ϵ

Output: adapted model f_θ^*; optional: customized geofence model e^*

if $\sqrt{\frac{1}{|g|}\sum_{k=1}^{g}||\mathbf{u}_k^{(\tau)} - \mathbf{u}_k^{(\tau-1)}||^2} > \epsilon$ **then**
  $\mathbf{o}^{(\tau)} \leftarrow$ `stochastic_center`$(\mathcal{Y}^{(\tau)}, \mathcal{Z}^{(\tau)})$
  $f_\theta^{\tau+1} \leftarrow$ `representation_model_update`$(\mathcal{X}, \mathcal{Y}^{\tau}, \mathbf{o}^{\tau}, f_\theta^{\tau})$
  /* Customization of e. See Sec. 4.4.2.2 */
  if e *is modifiable* **then**
    for $j \leftarrow 1$ *to* m **do**
      $e_{in}^{(\tau),j}, e_{out}^{(\tau),j} \leftarrow$ `Gaussian_model`$(\mathcal{Y}^{(\tau)}, l_j)$;
    end
  /* Digital observation update. See Sec. 4.4.2.1 */
  for $k \leftarrow 1$ *to* g **do**
    $\widehat{\mathbf{u}}_k^{(\tau)} \leftarrow$ `biometric_attendance`$(\mathcal{Y}^{(\tau)})$
    if e *is modifiable* **then**
      $\breve{\mathbf{u}}_k^{(\tau)} \leftarrow$ `digital_attendance`$(e_{in}^{(\tau),j}, e_{out}^{(\tau),j}, s_k, \mathcal{L})$
      $\mathbf{u}_k^{(\tau+1)} = \mathbf{u}_k^{(\tau)} - \gamma \cdot (\breve{\mathbf{u}}_k^{(\tau)} - \widehat{\mathbf{u}}_k^{(\tau)})$
    else
      $\mathbf{u}_k^{(\tau+1)} = \mathbf{u}_k^{(\tau)} - \gamma \cdot (\mathbf{u}_k^{(\tau)} - \widehat{\mathbf{u}}_k^{(\tau)})$
    end
  end
  $\tau \leftarrow \tau + 1$
else
  return f_θ^*, e^*
end

4.5 Implementation

4.5.1 Heterogeneous Sensing

There are two sensing modules in `AutoTune`, a biometric sensing module to collect vocal and visual data, and a digital sensing module for collecting ambient WiFi IDs[1]. We now discuss these specific implementations.

4.5.1.1 Biometric Sensing

We implemented a surveillance camera system which has a front-end remote camera and a back-end computation server. Specifically, the remote camera in our experiment is a *GoPro*

[1]The study has received ethical approval R50950/RE002

Hero 4 [2]. The camera is able to communicate and transfer data to the back-end through a wireless network. To avoid capturing excess data without people in it, we implement a motion detection module in our system with a circular buffer. The system works by continuously taking low-resolution images, and comparing them to one another for changes caused by something moving in the camera's field of view. When a change is detected, the camera takes a higher-resolution video for 5 seconds and goes back to look for changes. All the collected videos are sent back to the backend at midnight. On the backend of facial `AutoTune` we remove videos if no faces are detected in their frames by MTCNN [132]. We then feed remaining videos to the pre-processing pipeline as described in Sec. 3.5.2 and proceed to subsequent modules. Note that this implementation can be easily replaced with an off-the-shelf IP-based smart camera. For the case of voices, speech data is recorded via the embedded microphone on commercial smartphones, with a sampling rate of 16KHz. Note that, the positions of smartphones were different in various sessions which reflects the real-world complexity. On the backend of vocal `AutoTune`, `AutoTune` uses speaker activity detection [159] and diarization (see Sec. 3.5.1) to first extract biometrics and then feed it to feature extractor for subsequent processing.

4.5.1.2 Digital Sensing

This module is realized on a WiFi-enabled laptop running Ubuntu 14.04. Our sniffer uses Aircrack-ng [3] and tshark [4] to opportunistically capture the WiFi packets in its surrounding. The captured packet has unencrypted information such as transmission time, source MAC address and the Received Signal Strengths (RSS). As `AutoTune` aims to label biometric observations for POI, our WiFi sniffer only records the packets containing POI's device MAC addresses and discards them otherwise, so as not to harvest addresses from people who have not given consent. A channel hop mechanism is used in the sniffing module to cope with cases where the POI's device(s) may connect to different WiFi networks, namely, on different wireless channels. The channel hop mechanism forces the sniffing channel to change by every second and monitor the active channels periodically (1 second) in the environment. The RSS value in the packet implies how far away the sniffed device is from the sniffer [160]. By putting the sniffer near the camera, we can use a threshold to filter out those devices with small RSS values, e.g., less than -55 dBm in this work, as they are empirically unlikely to be within the camera's field of view.

[2] https://shop.gopro.com/cameras
[3] https://www.aircrack-ng.org/
[4] https://www.wireshark.org/docs/man-pages/tshark.html

4.5.2 AutoTune Configuration

4.5.2.1 Fine-tuning

AutoTune fine-tunes biometric models in every model update iteration. The fine-tuning process is detailed in Sec. 4.4.1, and we specify the fine-tuning protocol here. After running SCAN, the labelled data is split into a training set and a validation set, with a ratio of $8:2$ respectively. The pre-trained representation model is then fine-tuned on the training set, and the model that achieves the best performance on the validation set is saved. Note that the fine-tuning process in AutoTune does not involve the test set.

The mis-match between the small training data and the complex model architecture will result in overfitting. We therefore use mixed data in each iteration of model fine-tuning. More specifically, we use a set of data sampled from public labelled datasets and our cross-modality labelled samples to create a training set for a specific iteration τ. Additionally, we adopt the dropout mechanism (with a ratio of 0.2) in training, which is widely adopted to avoid overfitting [51]. It is worth noting that the data used for fine-tuning is uncorrelated with the test set for online identification. As we will discuss in the Sec. 4.6.1.2, the online testing is performed on a held-out set that is collected on different days.

Lastly, the backbone feature extractors used in AutoTune are FaceNet [117] and x-vector [3] for face and speaker recognition respectively. Both of them are state-of-the-art in their own fields. AutoTune is flexible and compatible with them and can automatically adapt them to identify people in a new domain.

4.5.2.2 Threshold of Convergence

The convergence of AutoTune depends on when the residual of ID predictions in two iterations are below a certain threshold ϵ. Apparently, ϵ affects the convergence speed, however, we fix it to 0.1 to control the variables in the following evaluations. In practice, we observe that AutoTune can work comparably for a relatively wide range (e.g., $[0.05, 0.12]$) of ϵ.

4.6 Evaluation

We are now in a position to evaluate AutoTune.

4.6.1 Evaluation Methodology

4.6.1.1 Datasets

We evaluate the effectiveness of AutoTune on three different real-world datasets: *Face(3 RMs)*, *Face(CommonRM)* and *Voice(Meeting)*. The first two datasets are collected in two different countries (UK and China) and we deployed surveillance cameras and WiFi sniffers in two testbeds. The third dataset is collected with a microphone and a WiFi sniffer in a meeting room. We now describe them in detail.

1. **Face(3 RMs):** This first dataset is collected in a commercial building in UK. We deploy the heterogeneous sensing front-ends, including surveillance cameras and WiFi sniffers, on a floor with three different types of rooms: office, workshop and kitchen. 24 long-term occupants work inside and can freely transit across these rooms. These occupants are naturally chosen as people of interest (POI). For the office, face images are captured with a surveillance camera that faces the entrance. The presence logs of occupants' WiFi MAC addresses are collected by a sniffer that is situated in the center of the room for the same time period. Besides the POI's faces, these images also contain the faces of 11 short-terms visitors (non-POI) who came to this floor during the experiments. We put different cameras in different rooms to examine the performance of AutoTune under camera heterogeneity. In particular, we put a Mi camera and a Pi camera in the kitchen and the workshop alternatively for a half of our experiment period. To further examine the resilience of AutoTune, we put cameras in adversarial positions. In kitchen, we deploy cameras with bad views near the entrance so that they can only capture subjects above $1.7m$ in height. While in the workshop room with two entrances, only the primary entrance is equipped with cameras. As a result, the digital observations (WiFi sniffing) and biometric observations (surveillance videos) are sometimes inconsistent. Tab. 4.1 summarizes this data collection.

2. **Face(CommonRM):** We collect another dataset in a common room of a university in China. There are no long-term occupants in this site and all undergraduates can enter. Of the 37 people that appeared during the three week period, 12 subjects are selected as the POI, and their WiFi MAC address presence is continuously recorded by the sniffer. Other settings remain the same as the office testbed in the UK. The challenge in this dataset lies in that the captured face images, both for POI and non-POI, are all of Asian people, while the initial face representation model is trained primarily on

Table 4.1: Key Metrics of Two Collected *Facial* Datasets.

Site	# of Rooms	# of POI	# of non-POI	# of Images	# of Sessions	Session Length	People /Session	Camera(s) in rooms	Experiment Note
3 RMs	3	24	11	15, 286	83	3h	9.14	Office: GoPro Hero 4 Kitchen: Pi Cam & Mi Cam Meeting: Pi Cam & Mi Cam	GoPro: 1080p, 90FPS PiCam: 720p, 90FPS MiCam: 720p, 15FPS
CommonRM	1	12	25	7, 495	102	2h	3.36	CommonRM: Gopro Hero 4	All Faces are from Asian

Table 4.2: Key Metrics of the Collected *Vocal* Dataset.

# of Rooms	# of POI	# of non-POI	# of utterances	# of Sessions	Session Length	Experiment Note
1	21	9	3,555	49	23.5min	Serious device-heterogeneity problem

Caucasians. As such, significant domain deviations are found in this dataset. Details of this dataset are given in Tab. 4.1.

3. **Voice(Meeting)**: The last dataset we collected is in a meeting room where a microphone and a WiFi sniffer were located inside. A microphone in the room recorded the conversations of 21 SOIs and 9 non-SOIs. On average, there are 3.22 people in each meeting. A WiFi sniffer was deployed inside the same meeting room to continuously scan the ambient WiFi identifiers in the vicinity. A notable feature of the last dataset is that we found a serious issue device heterogeneity. Not only does it have 16 different smartphones from 6 manufactures, but the meeting room is close to three other rooms and needs an accurate geofence model to infer occupancy. We therefore adopt the geofence model for *Voice(Meeting)*. We did not turn this option on for the other two datasets as inside/outside sessions are easy to determine in these two testbeds. The details of the this dataset are given in Tab. 4.2.

4.6.1.2 Metrics

Main task: Following the evaluation method in Sec. 3.6, we evaluate the performance of `AutoTune` in terms of the following metrics:

$$\begin{aligned} Precision &= \frac{TP}{TP+FP} \\ Recall &= \frac{TP}{TP+FN} \\ F_1 &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \end{aligned} \tag{4.12}$$

where TP, TN, FP, FN are true positive, true negative, false positive and false negative respectively. Each metric captures a different aspect of data association [136].

Downstream Tasks: Additionally, the adapted representation models by AutoTune are evaluated through online person identification tasks. We follow [161, 162] to use the Cumulative Match Characteristic (CMC) for identification evaluation. Lastly, we use binary detection accuracy to evaluate the customized geofence model on the third dataset. Note that, all the experimental data for downstream tasks were collected on different days from the adaptation period and do not overlap with the training data.

4.6.1.3 Competing Approaches

We compare the performance of AutoTune with 2 competing approaches:

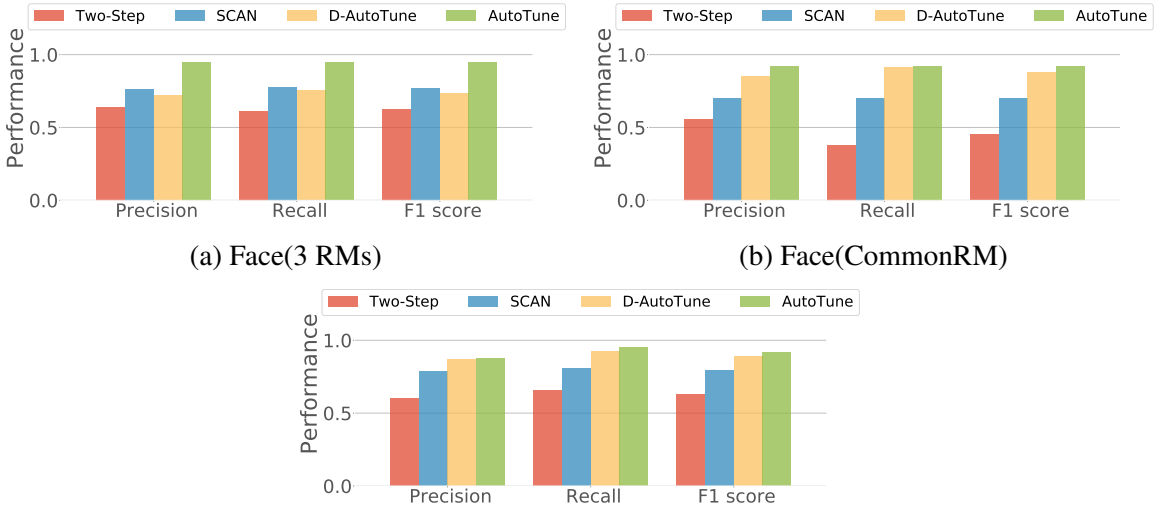
- **Two-Step** is a baseline method that sequentially clusters and labels biometric observation as introduced in Sec. 3.3.
- **SCAN** uses one-off associations to directly label the biometric clusters without adaptation, which is as stated in Sec. 3.4.
- **Deterministic AutoTune (D-AutoTune)** is the deterministic version of AutoTune. D-AutoTune directly uses the hard labels after association but performs iterative adaptation as in AutoTune.

4.6.2 Results

The analysis of results comprises three parts. The overall performance of AutoTune on three real-world datasets is reported in Sec. 4.6.2.1. We then analyse the robustness of AutoTune under different conditions in Sec. 4.6.2.2. Lastly, Sec. 4.6.2.3 provides the results of downstream tasks.

4.6.2.1 Overall Performance

Fig. 4.9 shows the performance comparison on three datasets. On the *Face(3 RMs)* dataset, AutoTune achieves an F_1 score of 0.95 and outperforms SCAN by ~ 0.18 in all metrics. Considering the different camera setups in this experiment, this result implies that AutoTune is able to cope with the data from heterogeneous surveillance cameras and supports many real-world use cases. The advantage of AutoTune is more significant in the *Face(CommonRM)* experiment where its F_1 score is as high as 0.92 and beats SCAN by ~ 0.22. An explanation is that the subjects of *Face(CommonRM)* dataset are Asian faces and requires more representation adaptation as the pre-trained model is learnt on Caucasian faces. As expected, the two-step approach struggles in both experiments. It is

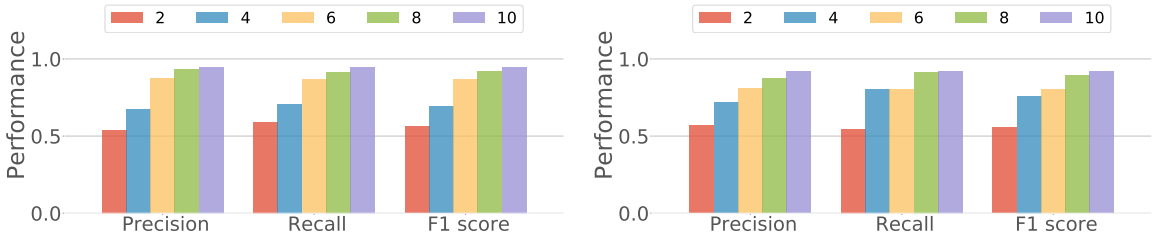


(a) Face(3 RMs)

(b) Face(CommonRM)

(c) Voice(Meeting)

Figure 4.9: Overall performance comparison on three different real-world datasets.



(a) Face(3 RMs)

(b) Face(CommonRM)

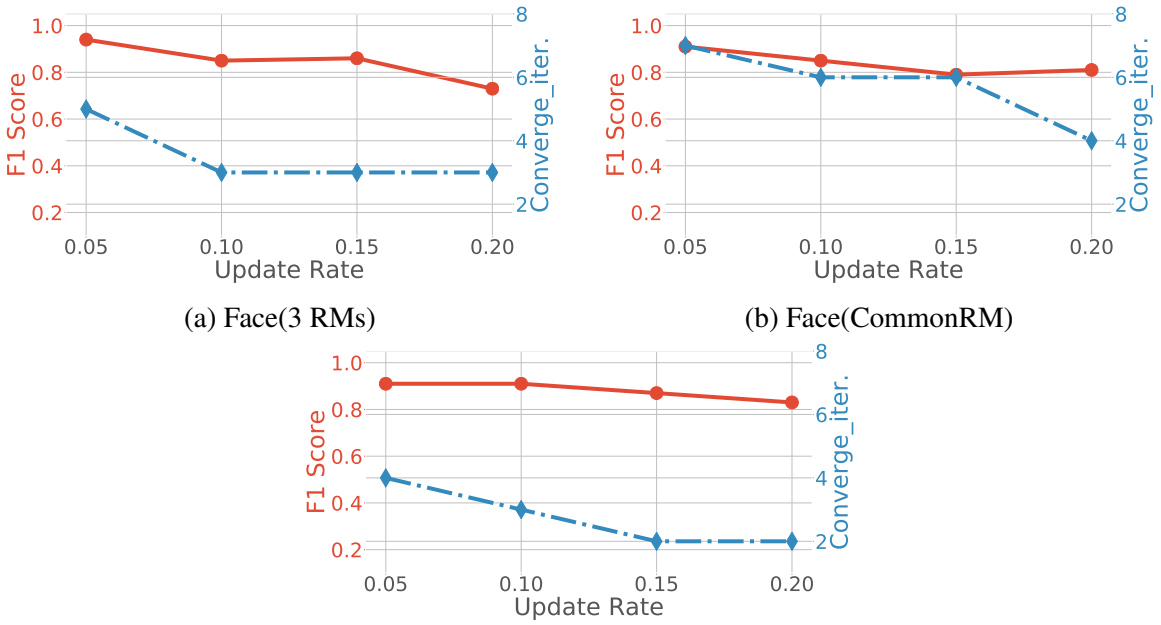
Figure 4.10: Performance vs. Lifespan on two face datasets.

$>$ 40% worse than AutoTune on the *Face(CommonRM)* dataset because of more disturbances of non-POI in this experiment. As shown in Fig. 4.9c, similar observations are drawn on the *Voice(Meeting)* dataset, though the gaps between AutoTune and the competing approaches are relatively small.

Lastly, we found that adopting probabilistic labels (see Sec. 4.3.2) is important to AutoTune. When the probabilistic labels are absent, the impact of noisy labels can counteract the benefits of iterative adaptation. For example, as shown in Fig. 4.9a, compared to the non-iterative method such as SCAN, we notice a performance drop of D-AutoTune on the *Face(3 RMs)* dataset due to the bad starting point. However, the probabilistic labels used in AutoTune are able to combat this issue and give significant performance gain.

4.6.2.2 Sensitivity Analysis

After discussing the overall performance, we can now analyse the robustness of AutoTune.



(a) Face(3 RMs)

(b) Face(CommonRM)

(c) Voice(Meeting)

Figure 4.11: Impact of update rate γ on three different real-world datasets.

Performance vs. Lifespan. In this experiment, we investigate the impact of the collection span on the performance of labelling. Longer collection days give more sessions, but an overly-long collection requirement is not ideal. We investigate its impact by feeding `AutoTune` with data collected on different number of days, and compare them with all days on two face datasets which are continuously collected. As *Voice(Meeting)* was conducted in several disjoint periods, we skip this analysis on it. Fig. 4.10 shows that `AutoTune` performs better with increasing number of days on two face datasets. The gap of F_1 score between the case with all days (10 days) and case with the least amount of days (2 days) can be as large as > 0.4 on both datasets. As discussed in the previous chapter (see Sec. 3.4), cross-modality labelling needs sufficiently diverse sessions to create discriminative enough context vectors. Otherwise, there will be faces or devices with the same session vectors that hinders `AutoTune`'s ability to disambiguate their mutual association. However, we also observe that when collection span is more than 8 days, the performance improvement of `AutoTune` becomes marginal. Overall, `AutoTune` needs a week of data to converge which is relatively short and acceptable.

Impact of update rate. This section investigates the impact of γ, which is the update rate of digital observations as introduced in Sec. 4.4.2.1. It uses the adapted face representation model to update the digital observations, e.g., wireless identifiers. A large update rate

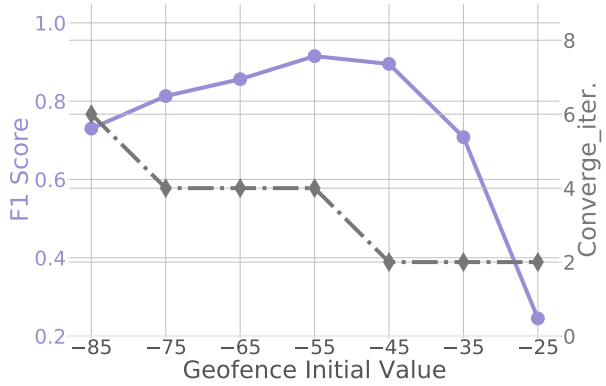
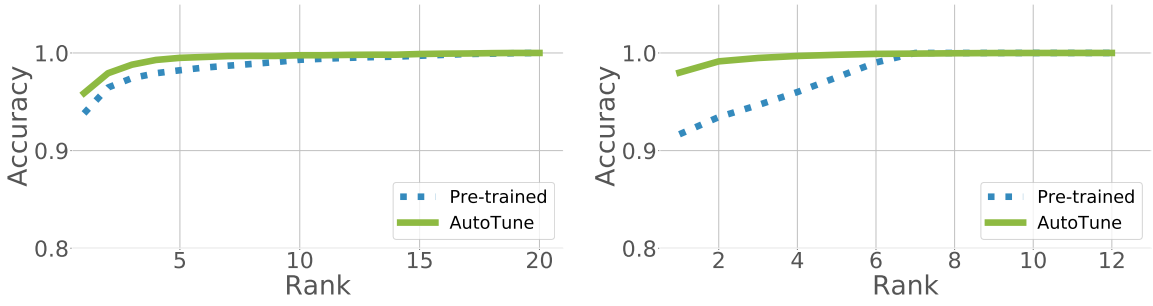


Figure 4.12: Impact of Geofence Initialization on *Voice(Meeting)* dataset.

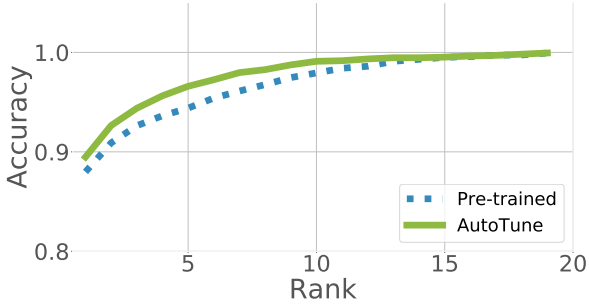
forces digital observations to quickly become consistent with the biometric observations. However, a large update rate also runs the risk of missing the optima. We vary the update rate γ from 0.05 to 0.20 at a step length of 0.05. Fig. 4.11a demonstrates that `AutoTune` achieves the best performance on the *Face(3 RMs)* when the update rate is set to 0.05. The performance declines by 10% when the rate rises to 0.2. This is because the updated digital observations quickly become consistent with the biometric observations but the representation models are not quite correct yet. When it comes to the *Face(CommonRM)* dataset (see Fig. 4.11b), a similar trend of F_1 score change can be seen. Although overall, the convergence becomes faster when the update rate increases, we observe that there is a fluctuation point at the update rate of 0.15, where `AutoTune` takes 6 iterations to converge. By inspecting the optimization process, we found that, under this parameter setting, `AutoTune` oscillated because the large update step makes it jump around in the vicinity of the optima but it is unable to approach it furthermore. Fig. 4.11c further shows the result on the *Voice(Meeting)* dataset, where `AutoTune` updates both digital observations and their observation model, i.e., the geofence model. Similarly, there is a clear tradeoff between effectiveness and convergence effort. Notably, we observed that `AutoTune` converged faster on this dataset because both observation update and model update would mitigate its inconsistency with the true biometric observations, i.e., voices. In practice, we suggest that users of `AutoTune` should select their update rate from a relatively safe region between 0.05 to 0.1.

Impact of Geofence Initialization. In the last part of the sensitivity analysis, we vary geofence initialization in `AutoTune` by using different RSS thresholds. This experiment was examined on the *Voice(Meeting)* dataset. Fig. 4.12 indicates that when the initial ge-



(a) Face(3 RMs)

(b) Face(CommonRM)



(c) Voice(Meeting)

Figure 4.13: Online Identification Performance on three held-out datasets.

ofence is defined too conservative, e.g., setting RSS threshold to -25dB, the labelling performance drastically declines. Under such conservative initialization, multiple devices that were present in the meeting room are falsely ruled out due to their weak signal reception ability. As a result, a large portion of RSS statistics that should be used to update the inside-geofence model is excluded whereas undesirably included for outside-geofence model update. Similarly, using a very small initial value wrongly included absent devices. We also found that the number of iterations required for convergence decreases when the initial value becomes larger. This behaviour is the natural consequence of the decreased number of POIs' devices present in sessions incurred by overly-conservative initialization. The above empirical results suggest that `AutoTune` is relatively insensitive to a sensible initial geofence initialization, and is able to operate well when initial values lie in the range of $[-70, -45]$dB.

4.6.2.3 Downstream Tasks

We now move to two downstream tasks. The first task is online identification, a multi-class classification task that aims to identify an unknown person in a dataset of POI. We use this task to examine the effectiveness of biometric representation in `AutoTune`. Without any overlapping with the samples used for adaptation, the test sets for online POI identification

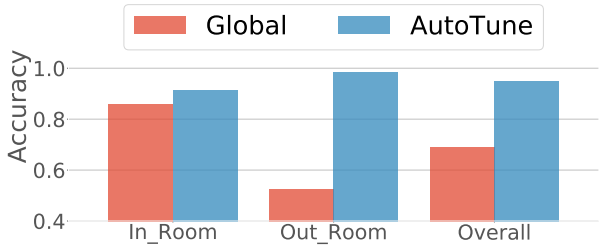


Figure 4.14: Effectiveness of Geofence Customization in the *Voice(Meeting)* Experiment.

are collected in different days. In total, the test set of *Face(3 RMs)* contains $5,580$ face images and the test set of *Face(CommonRM)* has $2,840$ face images collected in the same environments. We similarly acquire the test set of *Voice(Meeting)*, which contains 991 utterances from POI. The second downstream task is online geofence, in which we use the adapted geofence model in the *Voice(Meeting)* experiment for online localization. To this end, 10 POIs were asked to stay inside and then outside the same meeting room. A WiFi sniffer continuously recorded their device RSS values and we use these data to test the geofence model.

Online Identification. Through iterative cross-modality labelling, a biometric database of POI is developed and an online recognition system can be trained on it. In particular, we use the adapted representation model by `AutoTune` to extract biometric features and then employ a classifier, e.g., a linear SVM as in [163], to recognize subjects online. We study the effectiveness of the model adaptation by comparing it with the pre-trained representation model. It is worth noting that the training set for the classifier is through the labels associated by `AutoTune` itself. Fig. 4.13 shows that the adapted model by `AutoTune` is consistently better than the pre-trained model on all three datasets. The `AutoTune`-based recognizer is able to reach > 0.98 accuracy within three trials on both face datasets. Although its performance drops a little bit on the voice dataset, the `AutoTune`-based recognizer can still provide ~ 0.95 accuracy in three trials which is $\sim 3\%$ better than the pre-trained model. The largest gap is found in the *Face(CommonRM)* experiment, which is the dataset that bears the most domain differences. Recall that the face representation model we used is pre-trained on the Caucasian face database but all subjects in *Face(CommonRM)* are Asian. This experiment proves that the representation model can be enhanced by `AutoTune` for a new target environment or domain.

Online Geofence. In the last experiment, we evaluate the effectiveness of geofence customization as introduced in Sec. 4.4.2.2. Specifically, we compare the customized geofence model against a global threshold based method on a test set collected in the *Voice(Meeting)*

experiment. The global RSS threshold was calibrated by a single smartphone in the meeting room, without considering the device heterogeneity issue. As shown in Fig. 4.14, `AutoTune` is able to attain the best overall accuracy ($\sim 95\%$) for this online localization task. In contrast, the global threshold is biased to the calibrated smartphone and generalizes poorly to other devices. As a result, it only achieves an overall accuracy of 0.75. This downstream task implies that `AutoTune` not only can mitigate the labelling effort for biometric recognition, but also automatically customizes localization model for individuals.

4.7 Related Work

The adaptation module in `AutoTune` fine-tunes the biometric representation model to improve the robustness for in-domain data. In this chapter, we discussed two popular biometrics: facial images and human voices. We now briefly introduce some related work on facial and voice recognition in this section.

Face Recognition: Face recognition is arguably one of the most active research areas in the past few years, with a vast corpus of face verification and recognition work [52, 164, 165]. With the advent of deep learning, progress has accelerated significantly. Here we briefly overview state-of-the art work in Deep Face Recognition (DFR). Taigman et al. pioneered this research area and proposed DeepFace [72]. It uses CNNs supervised by softmax loss, which essentially solves a multi-class classification problem. When introduced, DeepFace achieved the best performance on the labeled face in the wild (LFW) [166] benchmark. Since then, many DFR systems have been proposed. In a series of papers [167, 168], Sun et al. extended on DeepFace incrementally but steadily increased the recognition performance. A critical point in DFR happened in 2015, when researchers from Google [117] used a massive dataset of 200 million face identities and 800 million image face pairs to train a CNN called Facenet, which largely outperformed prior art on the LFW benchmark when introduced. A point of difference is in their use of a "triplet-based" loss [124], that guides the network to learn both inter-class dispersion and inner-class compactness. The architecture of Facenet has since become the mainstream architecture in DFR, and research has incrementally improved the recognition performance by introducing different training losses, e.g., center loss [156], Large-margin softmax loss [169], angular softmax [161] and so forth.

Speaker Recognition: A standard speaker recognition system can be distilled down to two tasks: speaker feature extraction and feature scoring. i-vector [170] is a unsupervised feature extractor based on a linear Gaussian model. It has dominated speaker recognition tasks

and has inspired the design of DNN-based systems in this field. On the other hand, DNN-based embedding extractors are supervised models. Speaker features can be extracted from the last layers of a DNN-based feature extractor. A typical example is the x-vector system proposed in [3]. The difference in model structures makes x-vector more discriminative and superior in short-biometrics compared with i-vector [171]. Different scoring backends have also been applied to speaker features. One of the most widely used techniques is PLDA due to its discriminative ability. One of the main purposes of PLDA is also its use for domain shift ,i.e., adapting to a new domain. Commonly, to train a feature extractor, a massive amount of data is required ,e.g., more than $7,000$ speakers for VoxCeleb2 [135]. Since the in-domain dataset is usually insufficient, the easily available out-of-domain data is used for feature extractor training, whilst the scoring backend is adapted with in-domain data.

4.8 Summary

In this chapter, we proposed an iterative adaptation framework to improve cross-modality labelling and make it robust to noisy real-world scenarios. In particular, we address two challenges that limit the performance of `SCAN`. The first challenge is that digital observations interpreted from sensor measurements might be incorrect. For example, people sometimes forget to carry their smartphones, and the lack of detected wireless identifiers in this case may incorrectly indicate absence whereas their biometric observations such as facial images are still captured by biometric sensors. The second challenge is that `SCAN` suffers from out-of-domain biometric representation models. For instance, the face representation model pre-trained on public celebrity faces cannot generalize well to the local scenarios due to subject differences.

We observed that updates of biometric representations and digital observations are crucial for enhancing labelling robustness. By iterative learning and refining the noisy and weak association between biometric and digital observations, `AutoTune` is able to fine-tune a deep biometric representation model to tailor it to the environment, users, and conditions of a particular sensing systems. To handle imperfect labels and out-of-domain features, we propose a probabilistic framework in AutoTune that handles these types of noise. Lastly, `AutoTune` can adapt the modifiable digital observation model of interpreting sensor measurements. For example, in this chapter we studied the geofence model which translates wireless signals into user presence.

`AutoTune` is a generic framework that accounts for different biometric and digital attributes. Extensive real-world experiments in different applications demonstrated its effectiveness and robustness.

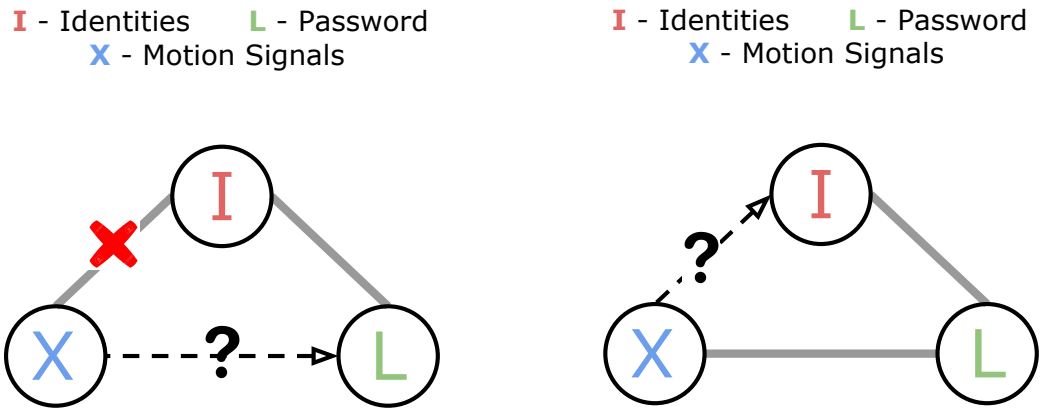
Chapter 5

Password Inference and Countermeasures

Chapter 3 and Chapter 4 proposed cross-modality approaches to automatically label biometric data, and further demonstrated that these labelled data can be used for online identification in the wild. The key concept behind automatic labelling is that one sensor modality is able to exploit shared *signals of opportunity* by other co-located sensor modalities to augment its own knowledge. In this chapter, we consider the other side of the coin, and try to understand the potential privacy concerns when shared signals of opportunity are maliciously utilized. Specifically, we investigate the feasibility of intercepting password information entered on smartwatches by sensing resulting motion data. The digital attribute here is the smartwatch passwords while the physical attribute is the motion signals from password entry. This study coincides with the theme of our thesis in that the touchscreen used for sensing password keystrokes is *co-located* with the motion sensors on a smartwatch. However, rather than inferring the identity of a person, the imposters need to develop a user-agnostic inference model where the goal is to correctly infer the password rather than the person entering it. In contrast, when we use the motion signal as an authentication source, the goal is the opposite as we want to build a user-specific inference model so that it can robustly authenticate users via behavioural biometrics. Fig. 5.1 describes this dual problem.

5.1 Introduction

Smartwatches are becoming increasingly ubiquitous: it is expected that the global smartwatch market has a potential to reach $32.9 billion by 2020 [172]. They are now deeply



(a) Attack Problem - `Snoopy` (b) Countermeasure Problem - `DeepAuth`

Figure 5.1: Schematic illustration of the attack and mitigation problems studied in this chapter. In the *attack* problem, the imposters need to develop a user-agnostic inference model where the goal is to correctly infer the password rather than the person entering it. Whereas for the *countermeasure* problem, the goal is to build a user-specific inference model so that it can robustly authenticate users via behavioural biometrics.

embedded in our daily lives, and over time can accumulate a variety of sensitive and important information such as emails, contacts and payment details. Due to their current role as an extension to the smartphones, the security and privacy of smartwatches have been delegated to the paired phones. However, driven by the major players such as Google and Apple, smartwatches are becoming more independent and are becoming prominent in the mobile ecosystem: they are no long just secondary displays, but are able to offer all basic functionalities without the presence of smartphones. For instance, it is already possible to pay via a smartwatch without needing to carry a smartphone [173, 174, 175], and many recent apps on smartwatches such as fitness tracking, well-being monitoring, and messaging (email/text) apps can work independently of phone usage.

These increased functionalities make smartwatches more useful, but also attract malicious attacks which traditionally target smartphone devices only. Smartwatches are typically secured through passwords, which is a sequence of digits. These are used not only to unlock the watch, but also to authenticate payments. In practice, the consequences of such an attack can be more serious than just security breach of the smartwatch screen lock [176]: as shown in our user study (discussed in Sec. 5.2), over 80% of 745 anonymous participants have a frequent habit of reusing the same passwords across services (e.g. PayPal, card payments ATM PIN codes) or even physical security (e.g. home alarm systems). Therefore compromising a smartwatch password could lead to a series of cyber and physical attacks.

There has been a solid body of work on the similar problem of attacking passwords on

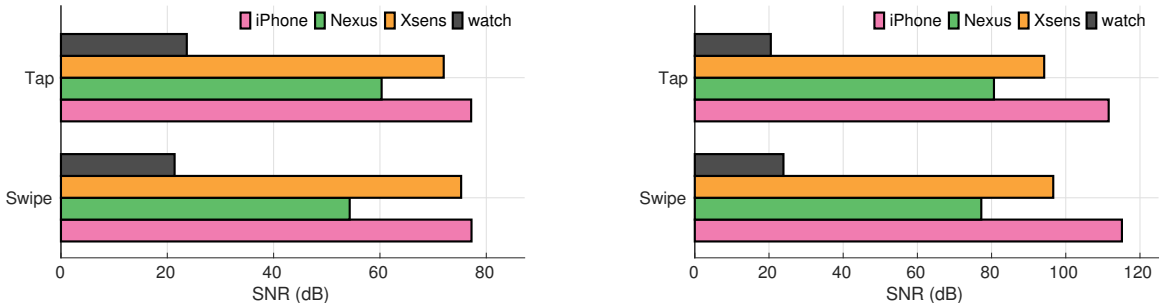


Figure 5.2: Signal-to-noise ratio (SNR) of motion sensors on smartphones, high-end IMUs and smartwatches. Left: Accelerometers; Right: Gyroscopes

smartphones, including analysing oily residues on the screen [177], video footage [178], radio signal perturbations [179] and motion sensor data [180]. In particular, attacking passwords by eavesdropping motion data is popular, since motion sensors are commonly sampled by a wide variety of applications, e.g. those designed for positioning, fitness tracking and activity recognition. In addition, giving access to on-board motion sensors appears to be innocuous, and many users (76% according to our user study) would grant permission instantly. Motion sensors leak information about the location of entry events such as taps through small changes in orientation and impacts. Existing techniques for cracking PINs on smartphones typically segment digits by extracting tap events and then use the extracted features to determine which digit has been pressed. As such, existing techniques rely on significantly handcrafted features and ad hoc approaches for digit segmentation.

Smartwatches with their smaller form factor compound the password classification problem, making it far more challenging than the smartphone case. Fig. 5.2 highlights this by considering the signal-to-noise (SNR) ratio of motion sensors on different devices. We see that motion signals on smartwatches are far noisier, and can be 20-40dB worse than that of smartphones or high-end IMUs. In the presence of low SNR, existing techniques designed for smartphones [180] typically fail to work. This is due to the reliance on hand-engineered features, which are not robust to variability across users and devices, particularly given the much weaker and noisier motion signals on smartwatches.

Although there are a number of papers which look at cracking PINs on smartphones, to date, only one paper has considered the issue of revealing swiped password. This is because swiped password usually can have an arbitrary length and hence have significantly more possible combinations. In this chapter, we provide a *universal* data driven technique for inferring both swiped and tapped passwords on smartwatches. This requires no handcrafted feature extraction or digit segmentation and is able to generalize well to the problem of arbitrary length passwords, even when faced with the extremely low SNRs found on smart-

watches. Our novel deep learning approach, based on recurrent neural networks (RNNs), exhibits a 3-4 fold increase in accuracy compared with the state-of-the-art. We propose two different architectures. The first exploits the skewed distribution of passwords to perform complete code inference. This technique shows superior results on popular passwords. The second is capable of digit level inference, i.e., it can generalize to any password that may or may not be present in a training database. For mitigation, a lightweight RNN based countermeasure is proposed, which employs the motion behavioural signals from password entry as a second authentication source.

In summary, the contributions of this chapter are:

- We present `Snoopy`, the first system that demonstrates the feasibility of intercepting passwords entered on smartwatches just by eavesdropping motion sensors. We propose a *universal* password inference mechanism based on deep recurrent neural networks that is able to attack tapped and swiped password. We present two variants, one which cracks popular passwords and another one which infers arbitrary passwords. Our system does not require any handcrafted features, only a crowdsourced training dataset.

- We propose a novel countermeasure `DeepAuth`, which exploits motion signatures when users enter passwords on smartwatches as behavioural biometrics, to provide a natural way of authentication in addition to the traditional passwords. It uses a deep neural network to model the input motion data, and considers a novel loss function to learn the optimal feature representations which are robust to noise, and can reliably reject unseen attackers with limited training data. `DeepAuth` offers in-situ real-time authentication on off-the-shelf smartwatches, by slimming the proposed deep network and paralleling inference.

- We have conducted a user study and collected over $1,000$ answered questionnaires, which shows that the affected population of smartwatch users is nonnegligible and the majority of users are not aware of the potential password leaks on smartwatches via motion data and its consequences.

- We have extensively evaluated the proposed systems, using data from over 360 distinct participants and $>$ 60K password entries on both Android and Apple devices. Our results show that the `Snoopy` achieves significant performance gains compared to competing techniques.

The remainder of the paper is organised as follows. Sec. 5.2 reports the results of our user study. Sec. 5.3 covers the necessary background. Sec. 5.4 presents a password inference approach, followed by the evaluation of it in Sec. 5.5. Sec. 5.6 describes an in-situ countermeasure for user authentication on smartwatches. The evaluation of the authentication system is given in Sec. 5.7. Finally, Sec. 5.8 presents an overview of related work, and Sec. 5.9 concludes this chapter.

5.2 Survey

In this section, we report the survey results of a user study that aims to understand the users' awareness of the potential password leak on smartwatches via motion data and its consequences. We distributed questionnaires on social media in the UK and China, asking anonymous participants to provide basic demographic information such as gender, age, occupation and smartwatch platforms used. The questionnaire consists of the following yes-or-no questions:

- Q1: Have you used the same passwords in different accounts/platforms, e.g., same PIN for PayPal[1] and your device screen lock?
- Q2: Did you know (before this survey) your passwords on smart devices could be leaked through motion sensors?
- Q3: Have you allowed or will allow third-party apps on your smart wearable devices to access your motion data, e.g., allowing WeChat+[2] to record the number of steps you've walked?
- Q4: Have you disabled or considered disabling third-party apps monitoring your motion data when entering passwords on your smart device?
- Q5: Do you often type on smartwatches (> 3 times a day), e.g., sending instant messages[3] or editing emails[4]?
- Q6: Are you a smartwatch owner?
- Q7: Have you set up unlock passwords on your smartwatches?

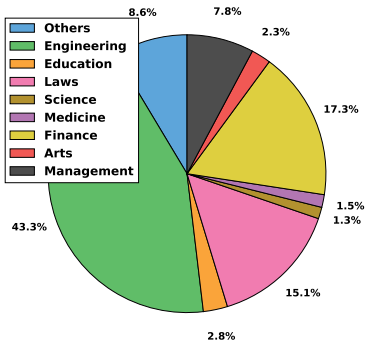


Figure 5.3: Background distribution.

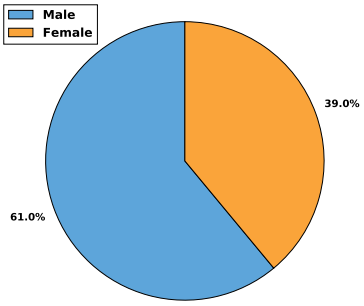


Figure 5.4: Gender distribution.

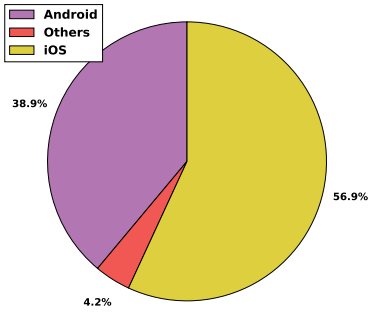


Figure 5.5: Platform distribution.

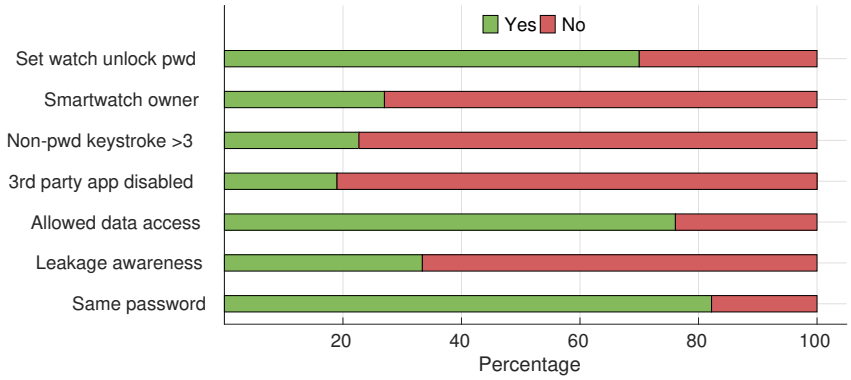


Figure 5.6: Survey results. They were asked 8 questions about smartwacth usage and password settings.

We received 745 anonymous responses for the first 5 questions and 301 anonymous responses for the last two questions respectively. Among them we have users from nine different occupation categories (Fig. 5.3), with slightly more male than female (61% vs. 39% as in Fig. 5.4), and an average age of 32.3 (σ=12.1, Mdn=28, ranging from 18 to 63). This is expected since in general, young male users are more willing to try out new gadgets such as smartwatches. We also observe that Android and iOS dominate the smartwatch market. As shown in Fig. 5.5, 39% and 57% of our participants wear Android or Apple watches, while only a tiny percentage (4%) of them were using other platforms.

Fig 5.6 summarizes the distribution of answers. First of all, we see that about 25% of our participants are smartwatch owners, and this number is expected to grow rapidly in the near future according to [172]. Among those smartwatch owners, we find that the majority of them (73%) would use asswords to unlock their devices. This indicates that

[1] https://www.paypal.com/gb/home
[2] https://www.wechat.com/en/
[3] https://hangouts.google.com/
[4] https://sparkmailapp.com/

smartwatches are truly becoming pervasive, and users tend to rely on the built-in screen passwords (swiped and tapped password) to protect their devices. Another key finding is that over 80% of the participants tend to use the same password across different applications. This means that the consequences of leaking smartwatch passwords can be significant: what if the smartwatch password is the same PIN used for online banking or Paypal? We also observe that a large number of users ($>$60%) did not know that the motion sensors on smart devices may leak sensitive information. In fact, 76% of the participants would allow, or had already allowed, third-party apps to access their motion data, and less than 20% of them would consider disabling motion sensors when entering sensitive information on their devices. This shows although motion sensor attacks have been extensively studied in academia, most users are still not aware of this. In addition, we see that unlike smartphones, users seldom perform complex interactions on smartwatch screens such as typing, since the size of them is much smaller than phones. This means in practice it is easier to detect password input events from other tapping/swiping on smartwatches, which makes this a more vulnerable attack surface.

5.3 Preliminary

5.3.1 Tapped vs. Swiped Passwords

There are two predominant types of password input mechanism on smartwatches (also on smartphones): tapped and swiped passwords [181].

For iOS platforms, the default password type is four digit password (referred as PIN hereafter), where the users *tap* their passwords on the screen when prompted. A four digit PIN has $10,000$ possible combinations. It is possible to use longer passwords, but in this work, we only consider the four-digit PIN.

For the Android platform, users have the option to use a tapped password or a swiped Android Pattern Lock (referred as APL hereafter), where the users *swipe* a pattern over a three-by-three matrix of dots (see Fig.5.7 for an example). Unlike the numerical passwords where the users can choose freely from ten possible digits at each tap, the smartwatch operating systems typically have certain constraints over the trajectories of the swiped patterns. For instance as shown in Fig.5.7, starting from the top left dot, it is only possible to swipe towards four reachable neighbours: the immediate right and the three dots in the second row. Therefore, the size of the search space for swiped passwords is restricted to a maximum of $389,112$ [177].

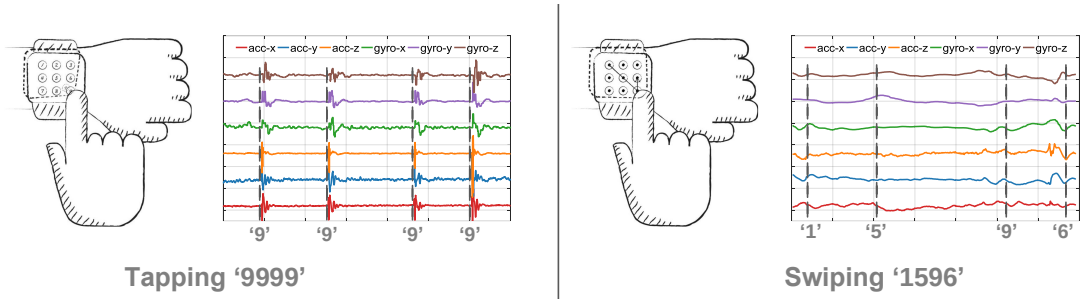


Figure 5.7: An example of motion sensor data changes induced by swiping a pattern-lock on a smartwatch. Tapping or swiping passwords does not follow uniform motion and is very challenging to distinguish individual digits, let alone reveal the entire code.

In practice, for both tapped and swiped passwords, if one fails to input the correct credential three times the smartwatch will forbid any further attempts for a few minutes. When the number of failed attempts reaches a threshold, the smartwatch can enter 'lost' mode, e.g. erase all data. As such, attackers need develop effective inference model that can guess password within few shots.

5.3.2 Motion Induced by Password Input

Intuitively, entering both tapped and swiped passwords will induce force and orientation changes on the smartwatch [182]. Since human skin has a certain level of elasticity, tapping on the smartwatch screen will cause minor displacement at the contact point along the vertical direction, i.e. the watch body will rotate about a small angle. Tapping causes an underdamped impulsive wave to develop, which causes small oscillations, shown in Fig. 5.7. On the other hand, when swiping passwords, the pressing and friction force between the user's finger and touch screen will "drag" the smartwatch to move along both vertical and horizontal directions. This gives rise to small slip-pulse waves which have a longer duration than impulse taps, as shown in Fig. 5.7.

In practice, induced motion can be picked up by the Inertial Measurement Units (IMUs) embedded on most of the commercial smartwatches. IMU sensors have been widely used in many mobile sensing scenarios, since they are able to capture displacement and rotation of the devices in 3-D space, and become increasingly cheap and power efficient. Concretely in this work we consider both accelerometers and gyroscopes, which capture the linear acceleration and angular velocity (roll, pitch and yaw) with respect to the three axis. By default the IMU sensors on most smartwatches are set to be always-on, continuously sensing motion for various applications such as gesture recognition, localisation, and fitness monitoring. This can lead to involuntary information leakage, which may be leveraged

by malicious parties to infer private and valuable data such as passwords.

5.4 Snoopy: Password Inference via Motion Sensors

5.4.1 Overview

5.4.1.1 Attack Assumptions

We assume that the user installs Snoopy, a Trojan app that can be easily disguised as a fitness or gaming app [183]. Snoopy requires access to the motion sensors, which does not require explicit permission in Android or via CMMotionManager in iOS. Snoopy logs and periodically sends candidate extracted password events via the network. In Android this is given by the *INTERNET* permission which is classed as a normal permission, not a dangerous permission. In iOS, this is done via a normal system API and thus no additional permissions are required. The amount of data that needs to be sent is also small - a 10s batch of candidate password data is only 3 kByte, so Snoopy is unlikely to trigger any network level monitors. Note that throughout the attack, Snoopy only needs to eavesdrop motion data, without having access to any other sensing modality, such as monitoring the touch screen [178, 184]. We assume that the attacker has physical access at some point in the future to either the watch or a related system (e.g., bank card or alarm system) where the password may be reused.

5.4.1.2 Attack Goals

The attack goal is to be able to infer both tapped and swiped passwords. Once comprised, an attacker can unlock the physical device, accessing all stored information. Alternatively as we observe in our user study (Sec. 5.2), many users would use the same passwords across different accounts and platforms, where compromising one password can be harmful to many services.

5.4.1.3 System Architecture

In this section we present the high-level architecture of *Snoopy*, a system for inferring PINs and APLs on smartwatches. Snoopy contains a client front-end which runs locally on a victim's smartwatch and periodically sends motion data to a password inference back-end that resides on the cloud. Fig. 5.8 shows the architecture of the proposed system.

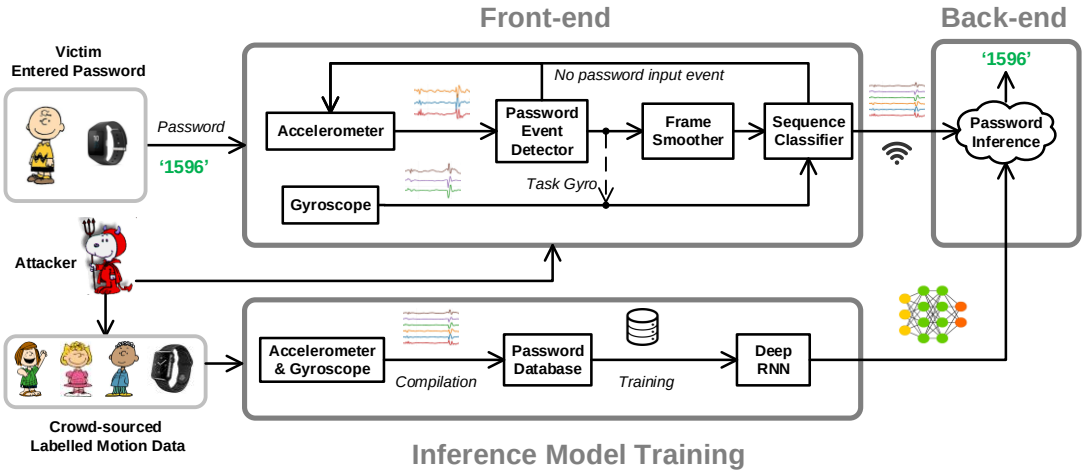


Figure 5.8: System overview of Snoopy. The attacker builds a deep RNN classifier using crowd-sourced data. On a victim's smartwatch, a trojan app uses an adaptive sampling scheme to record and identify a victim's motion data. Candidate password sequences are uploaded to the server. The back-end server runs the trained deep RNN classifier to infer possible passwords. Note, training is only required by the attacker; no training is needed by the victim.

Front-end Password Input Extraction: The front-end of Snoopy disguises itself as a harmless app, such as fitness app, and runs in the background continuously once installed. It listens to the IMU sensors and tries to detect when users are tapping or swiping passwords on their watches. To avoid being flagged as malicious by the host OS, the front-end of Snoopy uses an adaptive motion sensing strategy. It continuously samples the accelerometers at low rates to detect potential password input events. This conserves power, as accelerometers are typically one to two orders of magnitude more power efficient than gyroscopes. Once a candidate event has been detected, it enables the gyroscope and increases the sampling rate of both sensors, logging motion data until the user finishes entering passwords. Then the segment of data is smoothed and passed through a lightweight classifier, to determine retrospectively if it corresponds to a true password input event, or other user interactions such as swiping down to check notifications. In the latter case the data segment is simply discarded, while the data of true password input events is transmitted to the back-end for further analysis.

Back-end Password Inference: Given extracted segments of motion data, the back-end of Snoopy aims to infer user entered passwords. Instead of relying on bespoke signal processing algorithms which require hand-crafted features and tuning, Snoopy considers an end-to-end deep learning approach, which takes the raw motion data as the input, and computes the most likely password that the users has entered. To achieve this, Snoopy

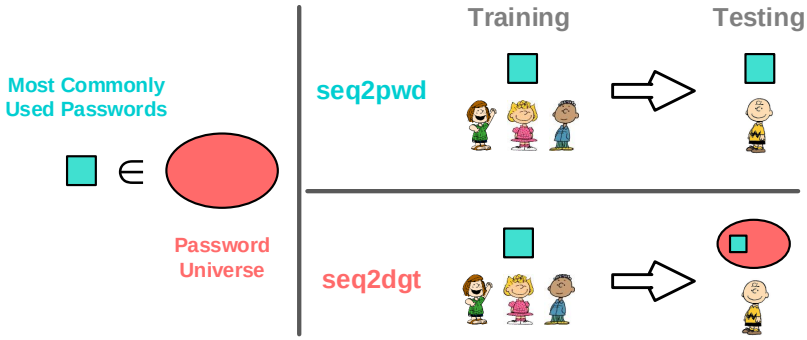


Figure 5.9: Comparison of seq2pwd and seq2dgt models in Snoopy. Both models are able to attack users outside the training cohorts. In terms of password coverage, seq2pwd model can infer passwords seen before, while seq2dgt model is able to infer any password including those not encountered before.

extends standard deep Recurrent Neural Networks (RNNs) to capture the unique characteristics of device motion induced by tapped PINs and swiped APLs. For PINs, it uses a hierarchical RNN with two layers to filter out the motion gaps (i.e. when the user lifts her finger off the touchscreen in between two taps) before inference, while for APLs it considers a bidirectional RNN to model the long continuous motion caused by swiping patterns. Now we are in a position to present the proposed Snoopy system in more detail.

The front-end is based on a standard motion extraction approach, implementation details of which are provided in the appendix (see Appendix A). The extracted data segments corresponding to those passwords are then transmitted to the cloud, where the back-end of the Snoopy system tries to infer the content of the passwords. Snoopy has two inference models: sequence2password (seq2pwd) and sequence2digits (seq2dgt). Both models adopt a novel deep learning based password inference approach, which does not rely on accurate keystroke segmentation or handcrafted features, and is able to infer passwords reliably across different users and devices. The rest of this section will focus on the model design in the back-end. Sec. 5.4.2 first explains how we cast the problem of password inference into a classification problem. Sec. 5.4.3 and 5.4.4 describe the design of two novel deep RNN models to infer the passwords from the captured motion data.

5.4.2 Password Inference via Classification

We consider the task of password inference as a *classification* problem, where the category labels are a set of passwords P, i.e. four digits PINs or APLs on 3×3 grid. Then given the segment of motion data, the problem of inferring the password that the user has just input becomes that of finding a label within the database P, which can best explain the

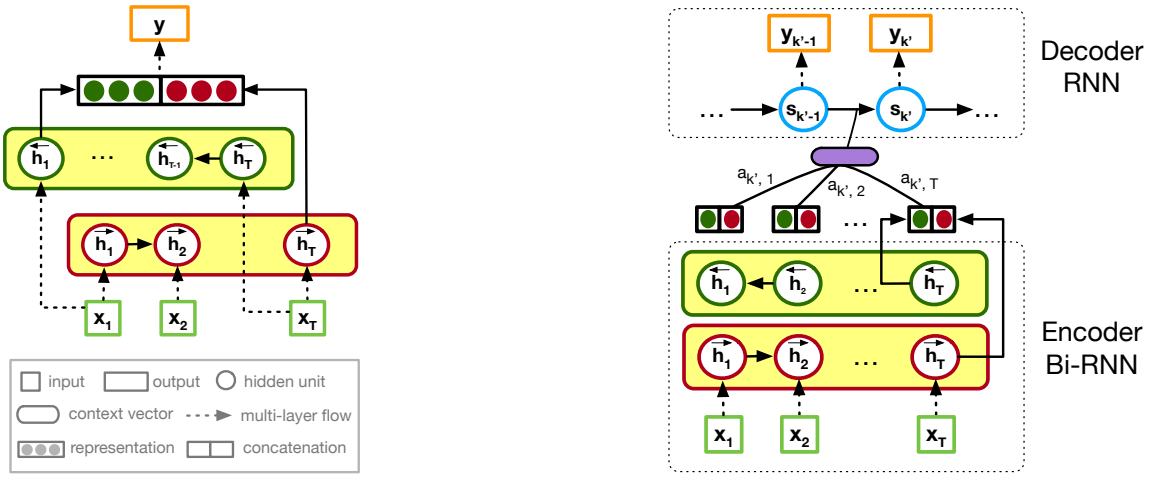
observed motion data. The size of the database P determines the inference model. Though the universe of all APLs and PINs is very large, the distribution of the adoption of them in real-world is skewed. For instance, according to statistical studies [185, 186], certain APLs tend to be more popular than others, and people only use a small set of passwords due to their bias. This means that one can utilize the skewed distribution and develop their database P targeting the most commonly used passwords, which is more efficient and more cost-effective. The seq2pwd model in `Snoopy` is designed in this context. As in [187], by taking inputs as the motion data, the proposed seq2pwd model classifies a sequence to the mostly likely passwords in P, without any digit segmentation.

However, despite the high likelihood of a password existing in the most commonly used password database, the expressive power of password inference is somewhat limited as seq2pwd loses its effectiveness when encountering unseen passwords ($\notin P$). And this problem gets serious when it comes to PIN inference, as the statistics of the most commonly used PINs are not as strong as the one of APLs. To solve this problem, a seq2dgt model is also proposed in `Snoopy` that takes inputs as motion data but predicts the password digit-by-digit. That is to say, we could train a model by a subset of the password universe but the learnt model is able to infer any member in the universe, as long as the constituted digits are seen by the model. Fortunately, there are only 10 possibilities of the digits in APLs or PINs, which is easy to meet. Therefore, the seq2dgt is essentially a digit classifier that outputs multiple predictions at each time, where the length of predictions is decided by the password length. Notably, unlike existing work, the proposed seq2dgt model does not rely on pre-processed keystroke segmentation [188] and known password lengths [187]. It automatically learns to align the chunks of motion data to the corresponding digits and learns to predict digits without knowing in advance how many there are. This is particularly useful in the case of APLs, where keystroke segmentation is not applicable [187], as the motion data of swiping APLs gives little information for digit segmentation (Fig. 5.7).

In the rest of this section, we introduce the seq2pwd model designed for APL inference as the distribution of popular adopted PINs is not as centered as that of APLs (see the survey in Sec.5.5). The seq2dgt model is proposed for both APL and PIN inference as they cover the whole universe. Fig. 5.9 illustrates the inference coverage of the two models.

5.4.3 Sequence-to-Password (seq2pwd) Model for Most Commonly Used Password Inference

Swiping passwords on the touchscreen of smartwatches, the user's finger causes the device to shift around slightly, creating slip-pulse waves in the acceleration and gyroscope data (as shown in Fig. 5.7). This means the motion signals induced by swiping are more continuous



(a) seq2pwd (b) seq2dgt

Figure 5.10: The architectures of two inference models in Snoopy. (a) seq2pwd model for commonly used password inference. (b) seq2dgt model for universal password inference.

than those of tapping, and typically without any gaps in between. Therefore, in this case the keystroke based inference approaches [189, 190] won't work well since it is impossible to segment the motion data without clear boundaries between different keystrokes.

On the other hand, for APLs the temporal correlations within the swiped pattern are much stronger, which can significantly reduce the search space and help the inference process. As discussed in Sec.5.3, given the current finger position, the smartwatch OS poses certain constraints on the possible directions of swipe [185], it is only possible to swipe towards three to four reachable neighbours when starting from the top left dot. Note that although the standard RNNs with LSTMs are able to capture these correlations to a certain extent, they have certain limitations. The most important is that the network only generates output from the last hidden state (i.e. $\mathbf{h_T}$). Standard problems solved by LSTMs in NLP are with the input sequences of at most 100 samples. However, the input sequence of APLs tends to be longer, and it is more difficult for the information encoded at the beginning of the input sequence to propagate through and impact the inference results.

To address this, Snoopy proposes a Bidirectional RNN (B-RNN) to model the rich temporal correlations within the input motion data. Concretely, at each timestamp k the proposed B-RNN keeps two hidden states $\overleftarrow{\mathbf{h}}_k$ and $\overrightarrow{\mathbf{h}}_k$, which incorporate the future ($k+1$, ..., T) and past (1, ..., $k-1$) information in the input sequence respectively, as shown in Fig. 5.10a. Then B-RNN uses the same machinery to update those states from both directions. Unlike the standard network, it has two output nodes: one $\overrightarrow{\mathbf{h}}_T$ at the end and the other $\overleftarrow{\mathbf{h}}_1$ at the beginning. Therefore in B-RNN, information flows from both the start and the end of the input sequence, and the output of the network is generated from

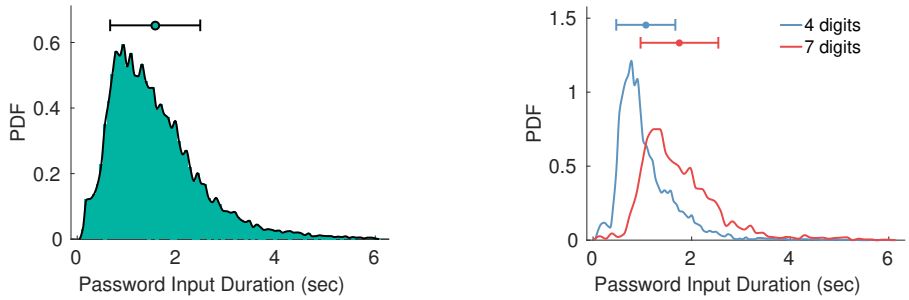


Figure 5.11: PDF of APL input duration. Left: duration distribution of swiping APLs. Right: Differentiating a 4-digit APL and 7-digit APL is difficult based on their duration distribution.

the concatenation of the two output variables $\overrightarrow{\mathbf{h}}_T$ and $\overleftarrow{\mathbf{h}}_1$. As shown in the next section, by using the bidirectional network architecture, `Snoopy` is able to preserve the long-term dependencies in the motion signals caused by swiping passwords, and thus infers passwords at much higher accuracy compared to competing approaches.

5.4.4 Sequence-to-Digits (seq2dgt) Model for Universal Password Inference

By formulating password inference as a sequence labeling problem, seq2pwd based RNN can guess passwords with a very high accuracy. However, it is difficult to adapt the seq2pwd framework to infer universal passwords. For instance, there are 389,112 possibilities for APLs and 10,000 for PINs. Using this large amount of labels for training classifiers requires huge amount of samples which is intractable in practice. We therefore propose the seq2dgt model to transform a sequence classification problem to a series of digit classification problems, where the current predicted digit conditions on the last prediction.

Despite their flexibility and power, standard RNNs can only be applied to problems whose inputs and targets share the same dimensions. It is a significant limitation in our context for two reasons. First, the lengths of APLs vary from 4 to 9, which implies the classifier needs to predict the length of digits implicitly. Fig. 5.11 (right) shows the duration distribution of 4 digit APLs and 7-digit APLs. As we can see, though there are 3 digits difference, the overlap of their distributions is above 40%. Second, an input IMU reading can be as long as several hundred samples, while the readings corresponding to a certain digit are only centered in a chunk of samples. Even if the number of digits is given, associating chunks to digits is difficult, as entering PINs or APLs on smartwatches *does not necessarily* occur with a uniform motion (see Fig. 5.7).

To solve the first problem, we leverage the encoder-decoder RNN architecture. It firstly uses an RNN as the encoder to map an input sequence to a context vector $\mathbf{c}$, and then stacks another RNN on it to decode the target sequence from the context vector. The decoder is often trained to predict the next sample $y_{k'}$, given the previous prediction $\{y_1, \ldots, y_{k'-1}\}$. Formally, the probability of output sequence Y with the length of T' (a few number of digits in our context) is defined as:

$$p(\mathbf{Y}) = \sum_{k'=1}^{T'} p(y_{k'}|\{y_1, y_2, \cdots, y_{k'-1}\}, \mathbf{X}) \tag{5.1}$$

With the decoder RNN, each conditional probability is modeled as:

$$p(y_{k'}|\{y_1, y_2, \cdots, y_{k'-1}\}, \mathbf{X}) = g(y_{k'-1}, s_{k'}, \mathbf{c}) \tag{5.2}$$

where k denotes a timestep in inputs ($1 < k < T$) and k' is a timestep in outputs ($1 < k' < T'$). g is a nonlinear, potentially multi-layered function that outputs the probability of $y_{k'}$; $s_{k'}$ is the hidden state of the decoder RNN and $\mathbf{c}$ is the encoded context vector. Fig. 5.10b illustrates the above RNN architecture used in our seq2dgt model.

By introducing a dummy digit symbol $< EOS >$, standing for end of output sequences, the unknown lengths of APLs can be implicitly determined. In this way, we have 10 candidate 'digits' for each digit in APLs that our model needs to predict, i.e., $\{1, 2, \ldots, 9, < EOS >\}$. The digit length of an APL is decided when the seq2dgt models gives the first $< EOS >$ symbol. For instances, a prediction $'1, 2, 3, 6, < EOS >, 9, < EOS >'$ indicates the length of target APL is 4. All predicted digits after the first $< EOS >$ symbol e.g., $'9', < EOS >$ in the example, are not counted. $< EOS >$ usage is widely adopted in the field of NLP [191]. An analogous instance of ours is machine translation, where a source English sentence may not have the same number of words as its Chinese translation. The dummy $< EOS >$ symbol can prevent the model generating an infinite number of words. Note that, this step is only for APLs; a PIN's length is fixed to 4 in most scenarios.

Originally, the context vector $\mathbf{c}$ is computed by encoding all inputs. However, the second problem remains as the IMU readings are sampled at 200Hz, whose lengths are dramatically longer than a few digits but only a part of them contribute at one decoding timestep. Here we introduce the attention mechanism in our seq2dgt model. Formally, the conditional probability in this attention seq2dgt is defined as:

$$p(y_{k'}|\{y_1, y_2, \cdots, y_{k'-1}\}, \mathbf{X}) = g(y_{k'-1}, s_{k'}, \mathbf{c}_{k'}) \tag{5.3}$$

Unlike the conditional probability in Eq. (5.2), here the probability is conditioned on a distinct context vector $\mathbf{c}_{k'}$ for each output digit $y_{k'}$. The new context vector depends on

a sequence of hidden sates $(h_1, \ldots, h_T)$ to which an encoder maps the input sequence $\mathbf{X}$, where we adopt a bidirectional RNN, i.e., $h_k = [\overleftarrow{\mathbf{h}}_k, \overrightarrow{\mathbf{h}}_k]$. Formally,

$$\mathbf{c}_{k'} = \sum_{k=1}^{T} a_{k'k} h_k \tag{5.4}$$

where $a_{k'k}$ are the weights determing the contribution of h_k in encoding $\mathbf{c}_{k'}$ for the k'-th digit and it can be determined through backpropagation in an end-to-end optimization. The attention mechanism is widely adopted in the scenario where input sequences are very long, and a single context vector is too compressed to decode outputs. For example, Hermann et al. [192] have achieved impressive results in document summarization by introducing the attention mechanism in their models, which solves the problem that the number of words of in the input documents are much larger than the ones in the output summaries. As shown in the next section, the seq2dgt model benefits from this attention mechanism and it is able to adaptively focus on specific chunks of input (with high attention weights) when generating digits in different positions of the passwords.

5.5 Evaluation of Password Inference

We now evaluate the performance of `Snoopy`. In the following, Sec. 5.5.1 describes the collected dataset. Sec. A.2 focuses on evaluating the performance of the front-end password extraction capability of the proposed `Snoopy` system; Sec. 5.5.2 presents the performance of `Snoopy` in inferring APLs.

5.5.1 Data Collection

We evaluate the proposed `Snoopy` system extensively on large-scale real world datasets collected in three different sites: *Oxford*, *Shanghai* and *Harbin*. In total, our experiments collected preferred/generated passwords from **420** anonymous participants, recruited a separate group of **362** volunteers to contribute their motion data when entering passwords on smartwatches (worn on their left hands), and accumulated over **60**k samples of motion data during password entries[5].

[5]The study has received ethical approval *R50768* from the University of Oxford.

5.5.2 Performance of Swiped Passwords Inference

We are now in a position to turn our attention to how the back-end password inference component of Snoopy performs. In this section we firstly discuss the performance of APL inference, while the PIN inference will be covered in Sec. 5.5.3.

5.5.2.1 Experiment Setup

APL Database Construction: As discussed in Sec. 5.4, to infer the user entered APLs, both the seq2pwd and seq2dgt models considered in Snoopy require a good password database P for training, which can cover as many common passwords as possible. To construct such a database P, we consider the publicly available APL data reported in [186] and also collected our own dataset. The APL dataset in [186] contains $\sim 4,000$ APLs entries collected from the anonymous users (with duplications). From this dataset, we rank the distinct APLs according to their frequencies, and select the most popular 113 APLs that can cover half of all the APL entries (2000 out of 4000). This ensures that the selected APLs can achieve a good coverage of the most commonly used APLs, while leaving out those APLs that are seldom used.

We also recruited 112 anonymous participants to survey their preferred passwords (both PIN and APL) when using mobile devices. The purpose of collecting our own password dataset is to obtain an independent dataset in addition to the publicly available data, which would make the constructed password database P more diverse. During the data collection process, we have made sure that every step complied with data privacy policies, and there is no link between the collected data and any individual participant. In particular, we have first obtained the participants' consent that their data will be used in a scientific study to evaluate password security on smartwatches. If a participant agreed to proceed, she was then given an Android watch, and we asked her to wear the watch on her left wrist. Then the participant is provided with an instruction sheet, which asks her to set a password in a survey app on the smartwatch. The survey app only records the entered passwords by monitoring touches on the touchscreen. When the participant finished entering the password, it asks if she is aware of the purpose of the study and would like to contribute this password. If so, the password is assigned with a unique random ID, and written into a random line of a local text file on the smartwatch. Otherwise there is no information saved. Note that during this process, the participants were asked to input passwords in private and take their time. The watches and instruction sheet were passed directly to the next participant without our intervention. After the survey process, we obtained 112 APL entries from all participants, among which we have extracted 64 distinct APLs. Finally, we fuse those 64

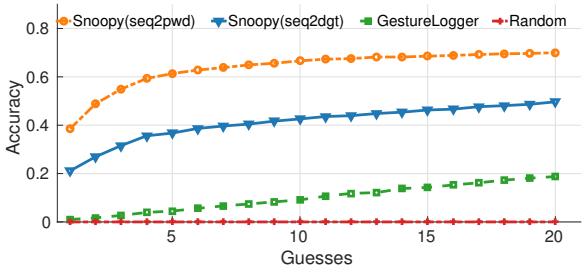


Figure 5.12: APL inference accuracy of competing approach and two proposed models in Snoopy.

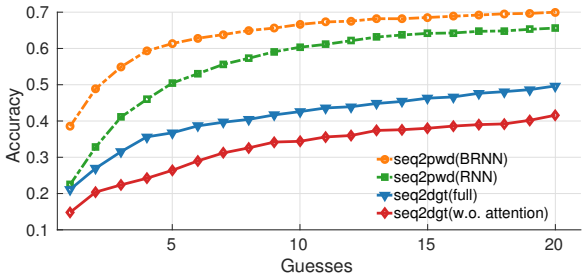


Figure 5.13: Impact of network architectures on the inference accuracy.

surveyed APLs and the 113 APLs extract from the existing dataset [186], and construct a password database P with 147 *distinct* APLs.

APL Input Motion Data Collection: Given the above constructed APL database P, we recruited a total number of 322 participants across three experiment sites to collect the motion data when they are entering APLs on their smartwatches. Each participant was randomly given 6 APLs selected from P. The participants were asked to wear the smartwaches on their left wrist, and then enter each password in our data collection app 20 times. The app logs the ground truth by monitoring tap/swipe on the smartwatch screen, and saves the motion data at the same time. In total, we have collected $36,569$ valid samples, each of which contains an APL and the motion data when it was entered. This set of data is used to train our models in Snoopy.

Competing Approaches: We implement both the seq2pwd and seq2dgt models considered in Snoopy using Keras [193], and train them on NVIDIA K80 GPUs with the Adam optimiser [194]. To the best of our knowledge, Snoopy is the first work to study the problem of inferring smartwatch APLs, and there is no existing work that can infer APLs without knowing the exact segmentation of digits within APLs (as discussed in 5.4.4). Therefore, here we only consider one of the best APL inference approach designed for smartphones, **GestureLogger** [187], which bears some resemblance to the proposed seq2pwd model in Snoopy.

5.5.2.2 Experiment Results

Field Test APLs vs. Constructed APL Database: The first experiment verifies the representativeness of the constructed APL database. We recruited an independent cohort of 308 volunteers (115 female and 193 male, mean age 39.8 with σ = 11.3, Mdn = 40, ranging from 18 to 63), and made sure that none of them were involved in building the APL database. Then we asked them to conduct an anonymous online survey to provide their preferred APLs. This survey also complies with data privacy policies and there is no link between the collected APLs and any individual participants. After the survey process, we obtained 308 APL entries from all participants. We found that among the 308 APL entries, 223 (72.4%) fall into the constructed APL database. This confirms that the constructed P covers a good variety of commonly used APLs, and it is possible to use P to accurately infer the user entered APLs.

APL Inference Accuracy in Field Test: This experiment evaluates the performance of APL inference of the proposed Snoopy system in the field test. As discussed above, Snoopy uses the constructed password database P and the associated motion data to train its models. To evaluate its true capability of inferring APLs in real-world scenarios, we consider the field test APLs which are independent with the APL database P. Concretely, we consider a similar approach as in [178], and recruited another 20 volunteers (13 males and 7 females), who had not contributed any password or motion data, to input the 308 APLs obtained from the filed test. The motion data associated with APL entries was collected using the same watch app, and on average each volunteer swiped about 120 APLs. We obtained $2,368$ valid samples using three different types of watches (Sony SW3, Samsung Gear Live and Moto 360), and this data is the used to assess the accuracy of APL inference.

We consider the successful rate at different number of attempts [185, 195] as the metric inference accuracy, which has been widely used to quantify the threat level of a malicious app [196]. As in GestureLogger [187], we set the maximum possible number of attempts to 20. Both proposed and competing methods take as input a motion signal sequence, and return scores for different candidate passwords. We then select the top 20 passwords, which are the most likely passwords according to the technique used. The first guess selects the top password, the second guess the next most likely, and so on.

Fig. 5.12 shows inference accuracy of APLs, where we include random guess as the naive baseline. We see that both of the proposed models (seq2pwd and seq2dgt) in Snoopy consistently outperform GestureLogger, achieving up to 3-4 fold improvement in inference accuracy. In particular, if only allowed to guess once, seq2dgt model can get 21% accuracy, i.e. one in five times it is able to guess the correct APL, while seq2pwd can achieve an even higher accuracy of 39%. We found that although seq2pwd model can only predict APLs

within the constructed database P ($|P| = 147$), its inference accuracy is 'worryingly' good: if 10 guesses are allowed, its accuracy can be 65% and increases up to to 68% for 20 guesses. Note that here the inference is performed on the field test data which is completely independent from the data used to construct P. This means that the APL database P constructed in our experiments is very representative, and thus in practice, it is possible to infer most of the popular APLs with such a database P. In addition, although GestureLogger also infers APLs from P, its accuracy is very limited and only able to reach 19% after 20 attempts (more than 3 fold lower than seq2pwd).

On the other hand, seq2dgt is not limited to the size of database P, and can predict any APLs within all the $389,112$ possibilities. We see that although the search space now is ~ 2700 times bigger, seq2dgt can still achieve good inference accuracy: about 43% after 10 attempts and up to 50% with 20 guesses. This indicates that the proposed seq2dgt model can indeed learn the underlying mechanism of user entering APLs, and make informed predictions when applicable. Note that although seq2dgt solves a much more challenging problem, i.e. no prior knowledge on popular APLs or perfect segmentation between digits, its accuracy is still way superior than the state-of-the-art GestureLogger: within 20 guesses, seq2dgt is 250% more likely to hit the correct password than GestureLogger.

Impact of Network Architecture: This experiment investigates the inference performance of Snoopy when using different deep network architectures. For seq2pwd model, we compare the inference accuracy of the proposed bi-directional RNN (B-RNN) and standard RNN. As shown in Fig. 5.13, B-RNN is about 15% superior to standard RNN at the first attempt, and is $\sim 8\%$ more accurate on average within 20 attempts. This means that the temporal correlations within APLs are difficult to be captured by standard RNNs, and by allowing gradient flow from both directions, B-RNN is able to capture richer information especially in long APL sequences. On the other hand, for the seq2dgt model, we see that the proposed attention mechanism can improve about 10% of inference accuracy over the standard architecture. This is because the attention mechanism helps the network to focus more on the chunks of informative sensor readings, i.e. when finger tips slide through digits, while the standard network only decodes APLs based on fixed context vectors.

Inference Accuracy vs. Device Heterogeneity: In previous experiments, although we always consider cross-user inference, i.e. our system is trained on data collected from one group of users, but tested against data generated from the others, we assume that the same model of device are used in training and testing. For instance, to infer passwords entered on a Sony SW3 watch, we assume that Snoopy can be trained with data collected on Sony SW3 watches (not necessarily the same one). In this experiment, we further push the limit, and see how Snoopy performs in the presence of device heterogeneity. This is very challenging,

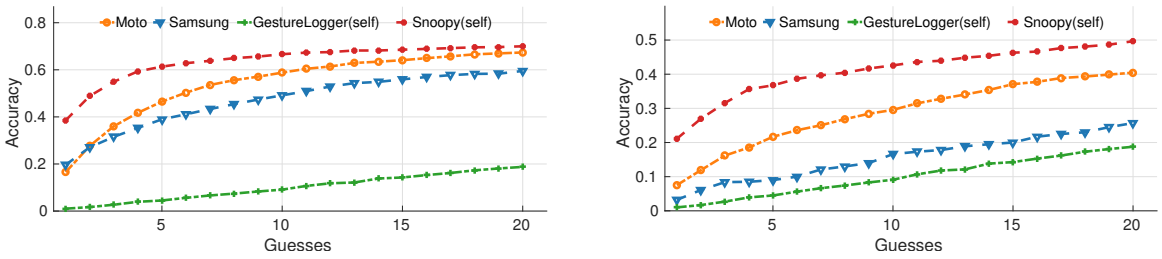


Figure 5.14: Cross-device APL inference accuracy. Left: seq2pwd model; Right: seq2dgt model.

since the watches used for testing may have different sensors, dimensions or shapes (round vs. square) with those used for training. To demonstrate this, for APLs we train Snoopy with data from Sony SW3 watches, and test it on the other two models, Samsung Gear Live and Moto 360 Sports respectively. Fig. 5.14 shows the inference accuracy with the same device model, and across difference models. Note that here we put the performance of GestureLogger (trained and tested on the *same* device model) as the baseline. As we can see, for seq2pwd, when tested on different devices, its performance drops elegantly. For Moto 360 which has round shape (the Sony watches used for training are square shaped), the performance only decreases by $\sim$13% on average. This means heterogeneity in the shape of smartwatches won't affect password inference performance significantly. On the other hand, the performance on Samsung Gear Live (square shaped) drops by about 20%. Note that even for this worst case, the inference accuracy of seq2pwd can still reach $\sim$50% after 10 attempts, while the best competing approach GestureLoger is less than 10%. On the other hand, from the right of Fig. 5.14, we see that seq2dgt model is slightly more sensitive to device heterogeneity. On Moto watches the accuracy decreases $\sim 18\%$ while about 25% on the Samsung watches. This is also expected since seq2dgt works against a massive search space ($389,112$ possibilities), where a small perturbation in sensor readings might lead to very different predictions. However even in this challenging case, the inference accuracy is still consistently higher than that of GeastureLogger, which is trained and tested on the same device models.

5.5.3 Performance of Tapped Password Inference

In this section, we further evaluate the performance of the proposed Snoopy system in inferring PINs entered on smartwatches.

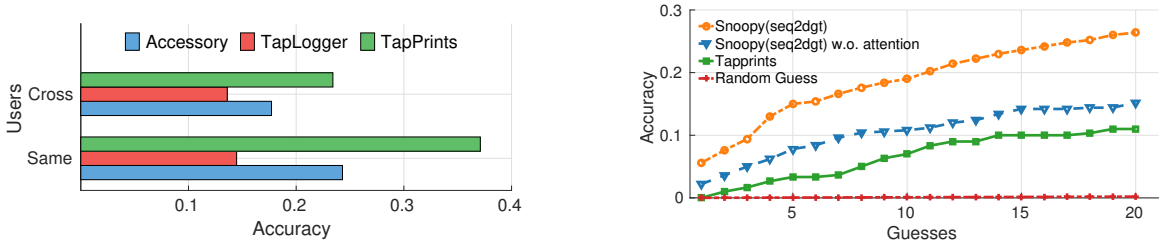


Figure 5.15: Performance of PIN inference. Left: element-wise accuracy of existing approaches; Right: inference accuracy of Snoopy and competing approaches.

5.5.3.1 Experiment Setup

Training Data Collection: Like the previous APL case, we first constructed a PIN database by surveying the same 112 anonymous participants. We follow the same data collection protocol as discussed in the previous section, and obtained a database P containing 79 distinct PINs from the 112 responses. To collect the PIN input motion data, we recruited a group of 156 users, and ask them to input 6 randomly selected PINs from P with smartwatches (iWatches) worn on their left wrists. We use a similar data collection app, and collected $23,144$ valid samples of motion data associated with the PIN input events. As in the APL case, this data is used to train the proposed Snoopy system.

Competing Approaches: To the best of our knowledge, there is no existing work studying PIN inference on smartwatches. Therefore, we compare Snoopy with the state-of-the-art PIN inference approaches designed for smartphones. These approaches usually adopt an element-wise inference: it first identifies each digit of the PINs and then concatenate the identified elements into whole passwords. We implemented three well-known approaches: 1) **Accessory** [190]: which uses a random forest classifier to identify the individual tapped digits from accelerator data; 2) **TapLogger** [189] which is very similar to Accessory but uses a k-NN classifier; and 3) **TapPrints** [188], which considers both acceleration and gyroscope data, and uses an ensemble classifier (SVC, decision tree, logistic regression and random forests as base learners) to detect PIN elements. Details of the competing approaches can be found in Sec. 5.8. Note that all of the three competing approaches require prior knowledge on the accurate segmentation of motion data, while Snoopy is able to perform end-to-end inference.

5.5.3.2 Experiment Results

Field Test PINs vs. Constructed PIN Database: The first experiment is to evaluate to what extent can the constructed PIN database cover the commonly used PINs in the real world. We obtain from [197] a large online surveyed PIN dataset containing $204,508$

PINs, and consider it as the field test PIN data. As in the APL case, we rank those PINs according to their frequencies, and then select the top 642 distinct most popular PINs to cover half of all the PIN entries ($> 100k$). For those 642 distinct PINs, we compare them with those in our PIN database P. We found that unlike the APL case where we observe a significant overlap between the field test passwords and the constructed password database, here the overlap is only about 7%. Unfortunately to the best of our knowledge there is no study so far that can provide a thorough explanation of why this happens. Our intuition is that people often use meaningful numbers to themselves as PINs, such as birthdays or addresses, which are quite unlikely to collide. In addition, clearly there is less constraints when tapping digits on touch screen than that of swiping APLs, and thus people may tend to choose from those easy-to-swipe APLs. Based on this observation, in the following we only consider *seq2dgt* for PIN inference but not seq2pwd, since the latter can only predict PINs from the database P which only covers a small percentage of commonly used PINs.

Element-wise Inference Accuracy: Before evaluating Snoopy, in this experiment, we first evaluate the performance of the existing element-wise inference approaches. We consider two types of password inference. Firstly, the *same-user inference* assumes that the algorithms would infer passwords entered by a user *with* access to the ground truth password-input motion data of this particular user, e.g. they have previously "seen" the user entering passwords (i.e. knowing the password contents), and collected the corresponding motion data. On the other hand, the *same-user inference* assumes the algorithms have to infer a user's passwords *without* access to her previous labelled motion data. As shown in Fig. 5.15 (Left), the performance of element-wise approaches is very limited: the best algorithm can only achieve about 25% accuracy when inferring a single digit for a PIN (one time guess), while the accuracy of random guess is 1/10. Even in the most favorable case where the testing objects are the same users in training, the performance only grows $\sim 10\%$. A possible reason for the low segmentation accuracy is that the SNR of the motion data on smartwatches is much lower than that of smartphones, which limits the performance of prior art designed for smartphones significantly. Overall TapPrints outperforms the other two, and thus in the following experiments, we only include TapPrints in our competing approaches.

PIN Inference Accuracy in Field Test: To evaluate the inference accuracy of the proposed Snoopy and competing approaches, we firstly collected a PIN input motion dataset based on the 642 distinct PINs obtained from the field test. As in the previous section, we recruited an independent group of 20 participants who hadn't contributed any data to enter those PINs on their smartwatches. The mean age of participants is 32.3 (σ = 10.4, Mdn = 31, ranging from 18 to 53), and the data collection process is similar to that in the previous APL case. We compare the performance of the proposed seq2dgt approach in Snoopy

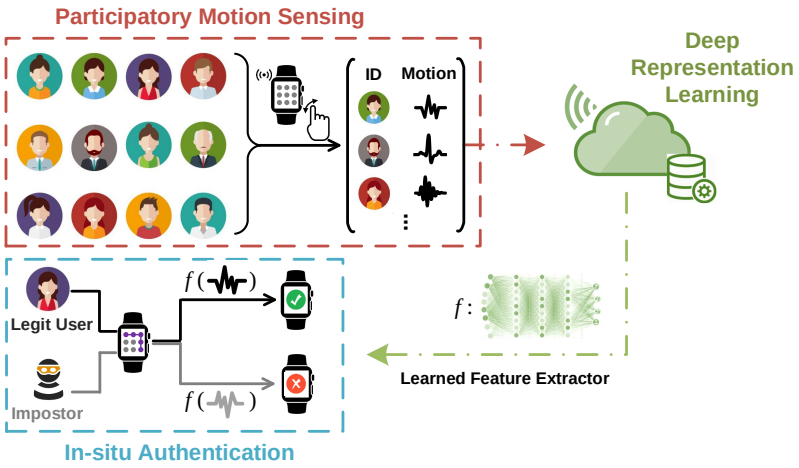


Figure 5.16: DeepAuth consists of three major components: *participatory motion sensing* module is designed for training data collection. *deep representation learning* module is to learn an optimal feature extractor that can best distinguish the password input behaviour of legitimate users from attackers. *in-situ authentication* module runs as a daemon on smartwatches and authenticates the users when they enter passwords.

(with and without the attention mechanism) and the best competing algorithm TapPrints on this dataset, and include the naive random guess as the baseline. As shown in Fig. 5.15 (Right), both variants of Snoopy(seq2dgt) consistently outperform the TapPrints, and is able to achieve $> 2\times$ accuracy improvement. In particular, with a single chance Snoopy is able to achieve 6% success rate, which is much higher than random random guess (0.01%). If more attempts are allowed, Snoopy can achieve up to 18% success rate after 10 guesses and 28% within 20 attempts. The performance of the competing TapPrints is much lower, and can only make to 11% after 20 attempts. In addition, we see that in this case the attention mechanism provides more performance gain (up to about 10%) comparing to that in the previous APL case. This is because in the case of PIN inference, the motion data associated with gaps between two taps is mostly noise, which won't provide any useful information for prediction. Therefore by using the attention mechanism, Snoopy can effectively ignore those gaps by assigning dynamic weights during decoding, i.e. it would put more weights on the data segments associated with real taps.

5.6 DeepAuth: Robust and In-situ User Authentication

In this section, we propose a countermeasure DeepAuth which also exploits the co-located motion sensors. Rather than treating them as a side channel, DeepAuth uses the behaviour signals as a secondary factor supplementing authentication. The key idea

behind DeepAuth is that password entering is a behaviour signature that can be harvested as an implicit defence tool. Furthermore, this secondary factor is implicit as such behaviour sensing is piggybacking on normal password entering hence requiring no extra user cooperation. DeepAuth is a generic authentication approach that is able to mitigate side-channel attacks such as Snoopy, as well as other types of password interception, e.g., dictionary attacks [198] and shoulder surfing [199].

5.6.1 Overview

DeepAuth consists of three major components, as shown in Fig. 5.16. In the *participatory motion sensing* module, smartwatch users voluntarily contribute their motion data to DeepAuth when entering passwords, in exchange for more secure authentication. Such crowd-sourced data, together with the associated user IDs are then used by the *deep representation learning* module, to learn an optimal feature extractor that can best distinguish the password input behaviour of legitimate users from attackers. Finally, the learned feature extractor is used by the *in-situ authentication* module, which runs as a daemon on smartwatches and authenticates the users when they enter passwords.

5.6.2 Participatory Motion Sensing

DeepAuth considers a participatory motion sensing approach, and opportunistically harvests IMU data when participating users are entering passwords. To make sure we obtain data from legitimate users, DeepAuth only initialises data collection when the smartwatches are in *authenticated states*, e.g. paired with the trusted smartphones. In this way, we implicitly leverage the strong authentication capabilities of smartphones, e.g. fingerprint or face recognition, to bootstrap DeepAuth. When in such states, DeepAuth tasks on-board accelerometer to listen for potential password input events. We use a Support Vector Machine (SVM) to analyse acceleration signals. When detecting such an event, we sample both accelerometer and gyroscope to extract motion data segment pertaining to that password input. The data and associated user identity are uploaded to the cloud for further learning. DeepAuth only samples and transmits motion data, and need **not** know the actual passwords. In OS level, e.g., Android, sending motion data from device to cloud without compromising the actual passwords is cheap and safe. Many mobile apps also send users' motion data back to server for data analysis. It is also worth pointing out that this motion sensing is not necessarily a one-off process, but can continuously operate to accumulate more training samples.

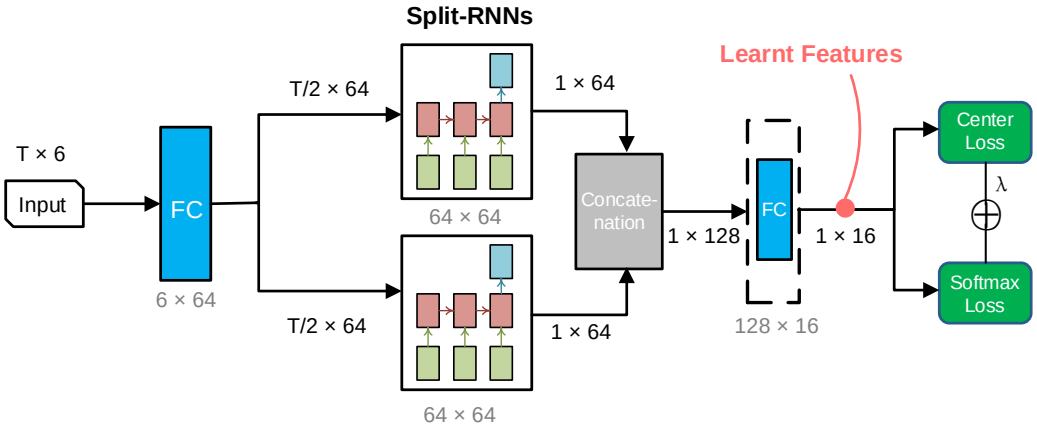


Figure 5.17: Proposed split-RNN layer (bottom). T is a variable denoting the length of an input motion sequence.

5.6.3 Deep Representation Learning with Limited Data

5.6.3.1 Problem Formulation

With motion sensing, `DeepAuth` obtains a set of motion data from different users when they input passwords on smartwatches. Each data point is a pair $(\mathbf{x}, y)$, where $\mathbf{x}$ is the motion data and y is its corresponding identity label. `DeepAuth` aims to learn a feature extractor f, which maps the motion data $\mathbf{x}$ to a lower-dimensional feature representation $\mathbf{f_x} = f(\mathbf{x})$, and further to user identity y_i under a certain model.

Ideally, we would like f to be *password agnostic*, and *robust against unknown imposters*. The former requires that, given new motion data $\mathbf{x}^*$ entered by user y_i, which is generated from an *unseen* password, the extracted feature $\mathbf{f}_{\mathbf{x}^*}$ should still be mapped to identity y_i. More importantly, when a malicious attacker mimicked the legitimate user y_i to input the stolen password (generating motion data $\tilde{\mathbf{x}}$), we require that in the feature space, $\mathbf{f}_{\tilde{\mathbf{x}}}$ shouldn't be mapped to identity y_i.

However, learning such an extractor $\mathbf{f}$ is very challenging, especially given limited training data available: it is not possible to obtain motion data of all password combinations, nor any prior data from the unknown imposters. To address this, `DeepAuth` employs a deep Recurrent Neural Network (RNN) for feature learning, and considers a novel composite loss to enable the network to work with limited training data. In the following, we first briefly explain the RNN used in `DeepAuth`, and then show how we design appropriate loss functions to learn the optimal feature representation.

5.6.3.2 Deep Representation Learning

In DeepAuth, we consider a many-to-one network architecture, where the input motion data is firstly pre-processed by a fully-connected layer, and then fed into a RNN layer. This RNN layer can be implemented in different ways. Fig. 5.17 shows our split-RNN model (discussed later in Sec. 5.6.4). The output of the RNN layer is forwarded to a fully-connected bottleneck layer, and then to the output supervised via loss functions. We extract the activations of the bottleneck layer as the learned features $\mathbf{f}$, because by design it is compact in size, and should encode sufficient information since it is the last fully-connected layer before output. Now we explain how to train the above network to obtain the optimal feature extractor.

5.6.3.3 Composite Loss

To make the learned features password agnostic, DeepAuth prepares the training data by indexing over identity labels, i.e. for a given user y_i, all her motion data is fed into the training process directly, regardless of the password contents. This allows the network to pick up common patterns when a user enters different passwords, and produces password neutral features. More concretely, given the training data of m samples collected from g users ($y_i \in \{1, \ldots, g\}$) , we use a combination of softmax loss and center loss to train a deep recurrent neural network:

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{softmax} + \lambda \mathcal{L}_{center} \\ &= -\sum_{i=1}^{m} log \frac{e^{W_{y_i}^T \mathbf{f}_i} + b_{y_i}}{\sum_{j=1}^{g} e^{W_j^T \mathbf{f}_i} + b_j} + \frac{\lambda}{2} \sum_{i=1}^{m} \|\mathbf{f}_i - \mathbf{c}_{y_i}\|_2^2 \end{aligned} \tag{5.5}$$

$\mathbf{f}_i \in \mathbb{R}^n$ is the extracted feature of the i-th input sample and current network. W and b are weights and bias and they are learn-able by the network. $\mathbf{c}_{y_i} \in \mathbb{R}^n$ be the centre of features for label y_i, which will updated as well during network training. The rationale of the centre loss term is that we found that the $\mathbf{f}$ learned in practice can successfully distinguish between the known users, but is not robust to unknown imposters. As shown in Fig. 5.18(a) and (b), features of different users (points) are separable, but the clusters are not compact enough to reject potential imposters (see Fig. 5.18(b)). This is because by using softmax loss we implicitly train the network only for *classification* within labels $\{y_i\}$, but not extrapolation. To address this, we introduce a centre loss function in the training process, to pull the learned features towards their centres. Centre loss is proved to be effective in clustering face images [156] and enhance intra-class compactness. Therefore in DeepAuth, we propose to use the *composite loss function* in training deep recurrent neural network. The hyper-parameter λ determines the trade-off between softmax and centre loss, and is obtained via

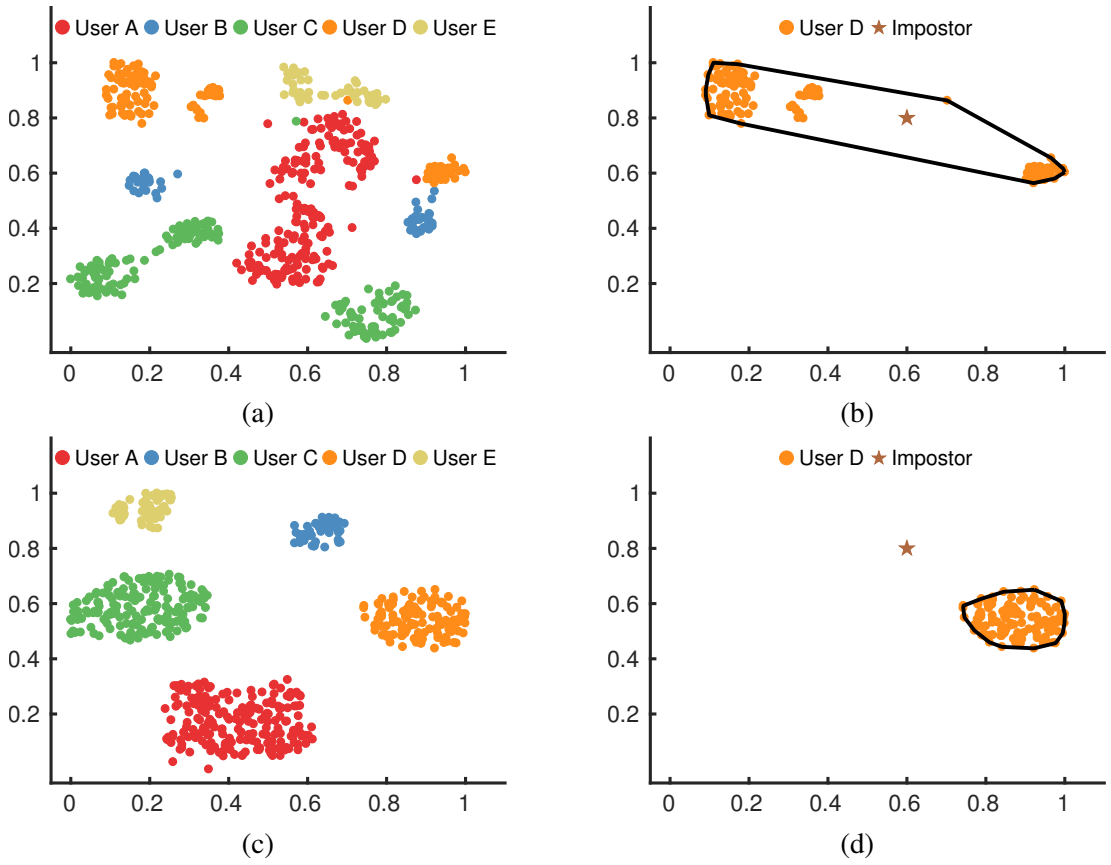


Figure 5.18: t-SNE visualisation of features learned using only softmax loss (a, b), and the proposed composite loss (c, d)

cross validation. Fig.5.18 (c) and (d) show the effect of using composite authentication loss.

5.6.4 In-situ Authentication on Smartwatches

5.6.4.1 Decision Function

`DeepAuth` deploys the learned feature extractor f to the users' smartwatches, which is used to authenticate users from malicious imposters. For a user y (who may not appear in training), `DeepAuth` firstly builds a behavioural model locally, which encodes her unique motion signatures when entering passwords. Concretely, it collects password input motion data in authenticated states, e.g. when the smartwatch is paired with her own smartphones, and fits the extracted features with a multivariate normal (MVN) model: $\mathbf{f}_y \sim \mathcal{N}(\mu, \Sigma)$. When a correct password has been input on her smartwatch, `DeepAuth` extracts the fea-

ture $\mathbf{f_{new}}$, and evaluates its distance from the user's behavioural model:

$$d(\mathbf{f_{new}}, \mathbf{f}_y) = \sum_{j=1}^{n} \frac{\|\mathbf{f_{new}}(j) - \mu(j)\|}{\Sigma(j,j)} \tag{5.6}$$

Here we assume that individual feature elements are independent. If distance $d(\mathbf{f_{new}}, \mathbf{f}_y)$ is below a certain threshold, `DeepAuth` accepts that the password was indeed entered by the legitimate user, and otherwise rejects this attempt.

5.6.4.2 Efficient Inference with split-RNNs:

A key step of the above authentication process is to compute the feature $\mathbf{f_{new}}$ from the observed motion data in real-time on the user's smartwatch. This is particularly challenging since resources (both computation and memory) on smartwatches are much more constrained than other platforms, and standard RNN inference is not feasible as it requires recursive processing of the input sequence.

However, we observe that in our context it is not always necessary to perform inference over the full sequence, because the correlation between the head and tail of the input may not be significant. For instance when entering an APL, the last few digits won't depend much on what was entered at the beginning. Therefore, it is possible to break the long input sequence and parallelise inference. Based on this intuition, `DeepAuth` splits the standard RNN model and distributes the inference task across two *split-RNNs*, as shown in Fig. 5.17.

The benefits are two-fold. Firstly, the RNN model size is reduced significantly, since the weight matrices in split-RNNs are halved, resulting in a much lower memory footprint. More importantly, inference on split-RNNs can be performed in parallel, and is more efficient since nearly half of the computation can be avoided. As shown later in Sec. 5.7.2, by using split-RNNs, `DeepAuth` can achieve real-time authentication on off-the-shelf smartwatches (<0.5s).

5.7 Evaluation of User Authentication

We are now in a position to evaluate the proposed authentication system that mitigates the password interception on smartwatches. In particular, we focus on the APL case because the evaluation in Sec. 5.7 implies more risks of `Snoopy` in this password class and fewer countermeasures are designed.

5.7.1 Experiment Setup

Data Collection: In the same settings as described in Sec. 5.5.2.1, we recruited 155 participants (38% female) from the three sites, with age ranging from 20 to 35. Each participant is given 6 APLs randomly selected from the above 64, and is asked to enter them on a smartwatch worn on left wrist, for multiple times across different days. In total, we have collected $27,145$ valid samples, each of which contains the motion data segment, and anonymous user identity.

Competing Approaches: We compare DeepAuth with three competing approaches: `DeepAuth-Softmax`, which is a naive version of DeepAuth using only softmax loss; `ICNP14` [200]; and `Mobicom13` [201]. The latter two are the state-of-the-art smartphone authentication approaches using shallow learning methods to recognize behavioural biometrics. We port their implementations to smartwatch platforms and trained on our data. Unlike DeepAuth, they use handcrafted features with shallow classifiers such as SVMs.

Metrics and Evaluation Protocol: We evaluate the competing approaches with the following metrics commonly considered in existing work [200, 201]: *accuracy, precision, recall* and *F1 score*. For each subject, we randomly split all her instances of 6 APLs into a training and a test set, at the ratio of 7:3. Then in evaluation, we mix both seen and unseen passwords (outside 6 APLs of the subject) entered by imposters, and examine the authentication performance for every subject. We guarantee that each subject has data in both training and test sets, and 30% data in her test set are labelled as positive and the data entered by impostors are marked as negative.

Implementation: We train DeepAuth in an end-to-end manner with Adam optimizer. The initial learning rate is set to 10^{-3} and the hyper-parameter λ is set to 0.001. The batch size in training is 512. In order to avoid overfitting, we implement dropout in every fully connected layer and the dropout ratio is set to 0.2. All hyper-parameters, including the threshold of MVN model, are determined on a held-out validation set (15%) from the training set.

5.7.2 Results

Overall Authentication Performance: We first evaluate the overall authentication performance of DeepAuth and the competing approaches with respect to different amount of training data. We alternate the ratio between users and imposters in training and testing sets, from 0.2 meaning that only data from 20% of the participants is used for training, while the rest 80% is considered as unknown imposters in the test set, to 0.8 which is the other way around. In Tab. 5.1, we see that overall DeepAuth is able to achieve much

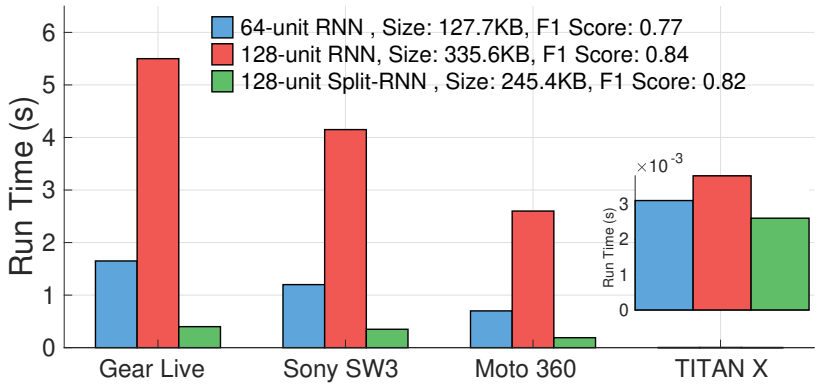


Figure 5.19: Efficiency and model sizes of different RNN layers across three smartwatch platforms and a desktop GPU. Model sizes and F_1 score are on the legend.

Methods	User Ratio	Precision	Recall	F1 Score	Accuracy
DeepAuth	0.2	0.75	0.77	0.76	0.92
	0.4	0.85	0.84	0.84	0.95
	0.6	0.87	0.85	0.86	0.96
	0.8	0.90	0.88	0.89	0.97
DeepAuth-Softmax Loss Only	0.2	0.67	0.84	0.71	0.87
	0.4	0.71	0.86	0.76	0.89
	0.6	0.78	0.87	0.81	0.93
	0.8	0.79	0.88	0.82	0.93
ICNP14	0.4	0.64	0.85	0.65	0.79
	0.8	0.72	0.91	0.77	0.89
MobiCom13	0.4	0.69	0.87	0.73	0.87
	0.8	0.75	0.87	0.80	0.92

Table 5.1: Authentication performance of DeepAuth and competing approaches.

better performance even with limited training data. On the other hand, using only the softmax loss (DeepAuth-Softmax) does not perform well: the F1 score when facing only 20% imposters is inferior to that of DeepAuth with 60% imposters. In addition, we see that approaches using handcrafted features (ICNP14 and MobiCom13) generally perform worse than DeepAuth. When facing 60% imposters, their F1 scores are far inferior to DeepAuth's. DeepAuth outperforms them in facing 60% imposters, even if competing approaches only faces 20% imposters. The significantly lower precision scores make competing approaches impractical for authentication. This means that they struggle in distinguishing legitimate users from imposters.

In-situ Authentication on Smartwatches: Now we evaluate the performance of DeepAuth when executed on off-the-shelf smartwatches. We deploy DeepAuth on three different watch models, including Sony SW3, Samsung Gear Live and Moto 360 Sports. We consider three variants of DeepAuth with different RNN implementations: 64-unit and 128-

unit standard RNN, and the proposed 128-unit split-RNN (see Fig. 5.17). For all variants, the feature extractor is trained with the composite loss, on datasets with 0.4 user/imposter ratio. As shown in Fig. 5.19, we see that although on desktop GPUs (TITAN X) the inference time is comparable, on resource constrained smartwatches, standard RNN may take more than 5s to process one authentication request, which is clearly not practical. On the other hand, the proposed 128-unit split-RNNs can speed up inference up to ten-fold compared to the 128-unit standard RNN, and is even 3-4 times faster than the 64-unit RNN, due to the effective parallelisation and reduced computation. This is because for both 64-unit and 128-unit RNNs, the computation is sequential while with 128-unit split-RNN, we can parallel the computation of two sub sequences halved from the original one. In addition, as shown in lengend of Fig. 5.19 the size of split-RNN is $\sim$30% smaller than the standard 128-unit model, while the F1 score is comparable (only 2% lower). Above results imply that DeepAuth with split-RNNs can reduce both inference time and memory footprint significantly, offering real-time authentication on smartwatches.

5.8 Related Work

Attacking secrets via Side-channel: Leveraging physical sensors as a side to attack secrets has recently received lots of attention. The authors found that the MEMS gyro sensors are able to pick up low-frequency vibrations from ambient sounds. Aviv et al. demonstrated that it is possible to reconstruct a locking pattern by analyzing the oily residues left on the screen [177]. This method has limited application as oily residues can be altered by other on-screen activities after pattern drawing, and also requires an attacker to have physical access to the device. Li et al. proposed a keystroke inference framework using variations in WiFi signals. They observe that keystrokes on mobile devices will lead to different hand positions and finger motion which alter the channel properties, reflected in the channel state information (CSI). A similar idea is proposed in [202], where WiFi CSI is used to infer keystrokes on a physical keyboard. However, these classes of attacks require similar environments and are highly sensitive to nearby moving objects. Vision-based attacks are also well established. Shukla et al. [184] used video footage captured near the victim to decode the PIN entered on the smartphone. Ye et al. [178] recently extend [184] to APL and their method is robust to different lighting conditions. Though video-based side-channel attacks are very efficient in determining passwords, it is difficult for the attackers to access and locate video footage containing password input events. Compared with the above more direct attacks, eavesdropping motion sensor data is robust to environmental dynamics, is

scalable, and can be achieved discreetly by a malicious app. On the other hand, due to their motion tracking capability and pervasiveness, motion sensors are popular side channels for attackers. Gyrophone [203] presents a new type of threat to intercept human speech by using a smartphone gyroscope. Marquardt et al. showed that it is possible to use the accelerometer within an iPhone to recover text entered on a keyboard when the phone is placed nearby on a rigid surface [204].

Inferring Secrets on Smart Devices via Motion Sensors: Researchers have attempted to infer keystrokes on smart wearables via motion sensors [205, 188, 187, 190, 189, 206]. The core idea behind these works is similar to the aim of `Snoopy`: keystrokes on device screen lead to distinct force/attitude patterns. The motion data on smart wearables can thus be used to infer entered secrets. TouchLogger [205, 180] and ACCessory [190] are early works, where ACCessory uses accelerometer only and TouchLogger utilizes both accelerometer and gyroscope to infer PINs. Similarly, TapLogger [189] refines previous techniques and uses a gyroscope to predict PIN-like secrets on smartphones. TapLogger uses a k-means clustering approach to extract the most likely classes (typically top 3). Given substantial observations of the secret (e.g. 32 PIN entry events), this is sufficient to estimate the true secret. Note that TapLogger uses manually extracted statistical features. TapPrints [188] advances the technique and extends inference capability beyond digits to English words. The papers mentioned above require accurate digit-wise classification, which is hard to achieve with smartwatch motion data. In contrast, GestureLogger [187] infers keystrokes in an end-to-end manner rather than individually identifying each tapped digit. To this end, GestureLogger firstly designs a password database of 50 graphical passwords and 50 numerical PINs as possible passwords. It then develops a sequence classifier that infers the most likely match given the motion data sequence. However, GestureLogger uses handcrafted features for inference, which is not robust to the variability of scenarios [51]. For example, as demonstrated in experimental results (Sec. 5.5), the features designed for smartphones in GestureLogger did not work well in the context of smartwatches. Though redesigning new features for smartwatches is possible but the process needs domain knowledge (e.g., motion sensors). Unlike all the above, `Snoopy` is the only one using deep neural networks that is able to learn the best feature representations automatically; its password inference framework can be easily applied to new scenarios without domain knowledge about the functioned sensors.

While TapPrints is specifically tailored to inferring PIN passwords, we provide a uniform approach that can be used to infer both PIN and APL (swiped) passwords. Even if one focuses on PINs only, we demonstrate a 2.5 fold improvement in accuracy compared

to TapPrints. This is because TapPrints decouples the problems of segmentation and classification into two separate steps, whereas our approach handles them more robustly by tackling both tasks using the same Deep Neural Network architecture. When compared to previous work that has focused on APLs (GestureLogger), our work is fundamentally different as it can address not only APLs that exist in the training dataset, but also new previously unseen APLs. This is a significant benefit that increases the impact of the attack. `Snoopy` is the only approach requires little domain knowledge on attackers' side, which significantly lowers the bar for attack deployment. In addition, whereas all prior art has focused on smartphones, we address the problem in the context of smartwatches. This is a far more challenging scenario, due to low SNR, in which previous approaches show very low performance compared to the proposed approach.

Smartwatch Security: As an increasingly ubiquitous device, the smartwatch has triggered new security issues. Wang et al. [207] and Liu et al. [208] pioneered this thread and have demonstrated the feasibility of inferring keystrokes on QWERTY keyboards by smartwatches. Their idea is that the keystroke-induced motions can be read from the motion sensors as long as smartwatches are worn the on victims' wrists. Similar risks are also reported on the ATM machines, where victims' digital PINs are leaked through motion sensors on smartwatches. Maiti et al. [209] proposed a context-aware protection mechanism which identifies sensitive motion events (e.g., typing on the keyboard) and trigger sensor access controller accordingly. The protection mechanism has been proved to work effectively to mitigate smartwatch based side-channel attacks with least interruption for the third-party applications.

Though `Snoopy` is an attack framework based on smartwatch, the goals are fundamentally different. The above works use smartwatches as a side channel to infer secrets entered on external devices, while `Snoopy` infers the inputs entered into the watch through the screen.

User Authentication on Smartwatch: Existing biometrics used for smartphones such as fingerprint scanning or face recognition are not suitable for watch platforms, as it is difficult to squeeze extra hardware into the current design. A recent patent [210] considers vein structure as biometrics, but it relies on particular IR sensors, which may consume more energy and are not widely available in commercial devices. Voiceprint can provide convenient and fast user authentication, but it may not be user-friendly in many scenarios such as during meetings. On the other hand, behavioural biometrics, i.e. the ways users interact with devices, offers a promising option, but it is very challenging to make such authentication practically useful on smartwatches. One major reason is that it is often harder to learn behavioural biometrics such as touch patterns [211] or motion signatures

[201] on smartwatches due to hardware limitation, e.g. smaller screens and noisier sensors. Another fundamental challenge is that behavioural biometrics are typically learned from limited training data, and therefore tend to work well for previously trained users, but often fail when unseen imposters imitate the legitimate users.

Unlike the above authentication methods, our proposed authentication system `DeepAuth` exploits motion signatures when users entering passwords on smartwatches as behavioural biometrics, to provide a natural way of authentication in addition to the traditional APL.

5.9 Summary

In this chapter, we studied the privacy implication of cross-modality inference and presented `Snoopy`, which intercepts passwords on smartwatches via the on-board motion sensors. Although side-channel attacks based on motion data have been widely investigated on smartphones, the problem has so far been overlooked on smartwatch platforms. As a result, users are not fully aware of the risks and potential consequences. To our knowledge, this is the first work that demonstrates the feasibility of attacking arbitrary passwords (PIN and Android Pattern Lock) on smartwatches using motion sensors. The proposed `Snoopy` system can disguise itself as a normal app (for fitness or wellbeing monitoring), and can successfully eavesdrop motion data in the background while passwords are entered. The extracted motion data is uploaded to the cloud, where `Snoopy` infers the contents of passwords using deep neural networks trained with crowd-sourced data. We collected large scale datasets (3 different sites, 362 users and $>$50k password entires), and compared the performance of `Snoopy` with state-of-the-art methods. By lowering the barrier to attackers in terms of engineering effort, the likelihood of being able to successfully compromise smart devices becomes significantly higher, simply by harvesting innocuous motion data from victims.

In light of these risks, we further proposed the countermeasure `DeepAuth`, which is a novel authentication framework for smartwatches based on behavioural biometrics. `DeepAuth` uses an innovative slimmed deep neural network with a composite loss function, to learn robust features from noisy motion data across different users, which can run in real-time on resource constrained smartwatch platforms. Extensive experiments with real-world data confirm that `DeepAuth` is able to provide a natural and user-friendly authentication mechanism on smartwatches in addition to traditional passwords, and can achieve impressive performance against unseen attackers even with limited training data.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Recognizing people underpins much research and development, through making spaces and devices truly personalized. However, the robustness of many recognition methods is threatened by the unpredictable conditions when operating in the wild. In such environments, recognition systems have to face new subjects who are outside the training set and domain deviations due to environment dynamics, where conventional supervised learning approaches to training and inference are poorly suited. The inability of supervised methods to cope with diversity could be overcome if a comprehensive training set is available to provide samples in various conditions. Unfortunately, obtaining such a labelled dataset would incur huge enrolment effort and be costly to acquire.

In this thesis, we observed the shared properties of signals of opportunity brought about by the advent of the Internet of Things (IoT). Based on this observation, the key insight underpinning this thesis is that *one sensor modality can leverage signals measured by other co-located sensor modalities to improve its own knowledge and functionality*. For example, if identity associations between heterogeneous sensor data can be structured and discovered, it is possible to automatically label data, leading to reliable human recognition, without manual labelling or enrolment.

The primary contribution of this dissertation is the development of techniques for intelligently integrating heterogeneous sensor data and providing a generic framework for robust identity inference in the wild. As sensing technology becomes more mature, our spaces are equipped with an ever-increasing volume and diversity of sensors. Many physical and digital attributes central to human identity can be observed by these sensors. However, the heterogeneous sensor data is usually unstructured and cannot be directly linked. For instance, detecting a WiFi MAC address of a speaker's device does not imply that she

is speaking at the exact instant. This unstructured relationship makes cross-modality labelling non-trivial. In Chapter 3, we utilized the similarity of session presence to perform cross-modality labelling. A novel method SCAN, was proposed that simultaneously clusters and associates biometric data and can thus cope with the noise in biometric sensing. To further mitigate the impact of noisy labels caused by sensing inconsistency, we proposed AutoTune in Chapter 4, a general framework that iteratively adapts the biometric representation model and corrects inconsistent digital observations. In order to thoroughly evaluate our proposed approaches, we perfomed experiments with both simulated and real-world datasets. Importantly, we collected 3 real-world data in two different countries that contain two different biometrics, namely, facial images and human voices. Extensive experiment results show the significant robustness improvements achieved by SCAN and AutoTune. At the same time, we found that adapted representation models by AutoTune are able to identify people better for online tasks.

Lastly, we considered the other side of the coin in Chapter 5 and tried to understand the potential concerns when the shared signals of opportunity are maliciously utilized. This time, the focus moved to privacy on smartwatches, and looked into the shared signals between the co-located motion sensor and touchscreen. We presented Snoopy, which intercepts password entries on smartwatches via the on-board motion sensors. Although side-channel attacks based on motion data have been widely investigated on smartphones, the problem has so far been overlooked on smartwatch platforms. As a result, users are not fully aware of the risks and potential consequences. To our knowledge, this is the first work that demonstrates the feasibility of attacking arbitrary passwords on smartwatches using motion sensors. Through cross-modality learning, we found that Snoopy is able to infer arbitrary passwords without segmentation even if they are outside the training set. In light of these risks, we further proposed the countermeasure DeepAuth in the same chapter. DeepAuth is a novel authentication framework for smartwatches based on behavioural biometrics. It uses an innovative slimmed deep neural network with composite loss functions, to learn robust features from noisy motion data across different users, which can run in real-time on resource constrained smartwatch platforms. Experiment results on > 350 participants and $50k$ samples demonstrated that the co-located secondary sensor not only can be maliciously used as a leakage channel, but can be effectively employed as an authentication source as well.

6.2 Future Work

Although we have provided novel directions in this new field, there are still underlying limitations and areas for future investigation. For instance, we have only considered two attributes in our cross-modality association algorithm whereas sometimes there is more than one type of physical or digital attribute observed in the same environment. Extending our proposed solutions to more general multi-modal scenarios has not been tackled in this thesis, but conceptually our framework could be extended to multi-sensor linkage. Another limitation is that the proposed iterative cross-modality adaptation is currently accomplished offline, which may limit its usability in real deployments, particularly for privacy sensitive domain adaptation. How to operate our adaptation approach online in a life-long learning manner is a topic worthwhile to explore in the future. Demand-based adaptation needs to be considered as the trade-off between model robustness and maintenance cost is usually a great challenge. Last but not least, in this thesis we proved that it is feasible to automatically label biometric data with the advocated concept of signals of opportunity. But whether this concept could generalize to other context-aware applications (e.g., localization and activity recognition) remains unknown. Therefore, the possible directions of future work that may address the above limitations include:

Coping with more types of attributes: The first direction of future work concerns the problem of incorporating more types and classes of attributes from multiple co-located sensors. In this thesis, we have only considered dual-attribute cases where each sensor modality observes a physical or a digital attribute. Intuitively, a holistic view provided by more types of attributes can reduce the uncertainty in identity inference. For example, as discussed in Sec. 4.1, if the WiFi sniffer is error-prone it gives erroneous observations of wireless IDs. If another type of digital attribute (e.g., name pass) can also be observed in the same environment, we can leverage it to supplement the sensed wireless identifier. Similarly, when having two different yet co-located biometric sensors, such as a microphone for voice and a camera for face, the subjects' session attendance inferred from biometric observations can be made more reliable. An interesting future direction is therefore to extend the cross-modality association and adaptation approaches proposed in this thesis to a generic and versatile framework able to incorporate multiple attributes. A possible first line of investigation could be using a graph neural network [212] to encode all heterogeneous observations as nodes, and recursively re-fine their associations.

Efficient Lifelong Adaptation: Another direction of future work is to update the biometric representation model online, so that it is able to promptly adapt to environmental changes and perform re-tuning locally. Recall that the current adaptation framework is completely

offline and occurs after heterogeneous sensing. However, the monitored people of interest (POI) in the environment sometimes change. For instance, when a new worker joins a company, the facial recognition system needs to update her profile by `AutoTune` again. Running `AutoTune` from scratch is of course inefficient, and the computational burden would accumulate with more environment changes. In order to customize our proposed framework to perform lifelong learning, an online adaptation strategy is required. Achieving the optimal trade-off between system performance and maintenance cost is an interesting problem. A possible approach is to draw techniques from demand-based protocols [213] to efficiently enable online adaptation of `AutoTune`.

Side-channel attacks exploiting co-located sensors: Another potential direction for future work concerns the investigation of unexplored side-channel threats that come from co-located sensor information. This will be a natural extension of `Snoopy` and `DeepAuth` proposed in Chapter 5. Detecting new types of side-channel attacks could provide insights, which can often be used to (1) turn a vulnerability into a functionality (e.g., using ambient WiFi identifiers to label speaker voices), as there is a fine line between risk and value-added services. It can also be used to (2) design defence mechanisms to protect against such vulnerabilities. Particularly, in Sec. 4.4.2.2) we found that a side product of biometric recognition is customized fingerprints. This customization can significantly improve the localization performance that accounts for heterogeneous devices. This is an overlooked vulnerability of today's voice assistants (e.g., Google Home/Amazon Alexa) in domestic settings. Basically voice assistants could gradually infer users' location precisely through iterative biometric adaptation.

Extension to other context inference problems: Lastly, this thesis focuses on human identification tasks. But our advocated concept of cross-modality learning is obviously a general principle that is able to apply to other context inference problems. Taking the location context as an example for mobile robotics, we could leverage perceived LIDAR data to help augment the location understanding of a co-located millimetre wave radar on the same robot. With such "cross-modality training", the millimetre wave radar could operate independently the next time, without the use of the expensive and power-hungry LIDAR. We believe that with the signals of opportunity becoming more pervasive in IoT environments, the traditional sensor fusion methods based on temporal evolving models will be revolutionized by cross-modality learning. After all, the heterogeneous data in many context inference problems is inherently unstructured and requires sophisticated analytic techniques for information extraction.

Appendix A

Password Extraction

Here we show the implementation details of the password extraction module, which is the front-end of `Snoopy`. From a high level point of view, `Snoopy` infers the users' passwords by collecting and analyzing the motion data generated on their smartwatches. Of course one can task the motion sensors continuously at high sampling rates, and stream the sensor data to the cloud for password inference. However in practice, this will incur significant cost in energy, computation and communication, where the smartwatch operating systems (Android Wear or WatchOS) can easily detect such unusual behaviour and kill `Snoopy` instantly. To make our attack realistic, we would like `Snoopy` to be as "benign" as possible, i.e. it should not ask for excessive battery or bandwidth use most of time, but only become active (processing/transmitting) when the users are actually inputting passwords. In Sec. A.1.1 and A.1.2, we first explain how to adaptively task the motion sensors to detect potential password input events without incurring heavy load on the system. Then, in Sec. A.1.3 we discuss how to extract the precise segments of motion data corresponding to those detected candidate events, and identify if the segments are related to actual password input events.

A.1 Implementation of Password Input Extraction

A.1.1 Adaptive Motion Sensing

`Snoopy` uses the onboard accelerometers to detect potential password input events, since they are very power efficient compared with gyroscopes [214]. Concretely, we consider an adaptive sensing strategy, which switches between three modes: *passive listening*, *password input monitoring*, and *motion data extraction*, depending on different user behaviour.

Most of the time Snoopy stays in the passive listening mode, where it only samples accelerometer data at low rates and runs a gesture detection algorithm. Note that in this mode, Snoopy won't necessarily incur extra load on the sensors, since in practice major smartwatch platforms have their own gesture recognition or fitness services running in the background, which already task the accelerometer continuously. When it detects a user's intention to interact with their device, via detection of a characteristic wrist movement, Snoopy transitions into the password input monitoring mode, where it increases the accelerometer sampling rate to look for potential password input events. It keeps analysing the received acceleration data, seeking to detect when the user will start entering their password. Once such an event is detected, Snoopy immediately turns into the motion data extraction mode, and samples both accelerometer and gyroscope at higher rates until it detects that the user has finished typing/swiping passwords. This segment of motion data is cached locally and passed through a classifier which decides if it corresponds to normal tapping/swiping, e.g. check email notifications, or a true password input event. In the latter case, the cached data is sent to the cloud for password inference. At the end of this process, Snoopy goes back to passive listening. In this way, Snoopy only actively processes and transmits short bursts of password related motion data, and avoids unnecessarily alerting the OS or malware monitoring frameworks.

A.1.2 Password Input Event Detection

As discussed above, when the users try to interact with their watches, Snoopy increases the accelerometer sampling rate and starts to check if any password is entered. Given the raw acceleration stream, Snoopy uses a sliding window of length T and stride S (both expressed in terms of samples) to segment the data into *frames*. Each frame contains T data points and the overlap between adjacent frames is $T - S$ (assuming $T \geq S$). In practice, the optimal T and S depend on the accelerometer sampling rate and can be learned from the data. For instance in our experiments, when the accelerometer rate is set to 40 Hz, the best T and S are 60 and 6 measurements respectively. Then for each frame, we would like to decide whether the user starts to input passwords within that frame. To achieve this, we first extract various features of the data frame, e.g. moments, maximum/minimum, skewness, kurtosis of individual acceleration axis, and different norms (e.g. l_1, l_2, Infinity and Frobenius norms) across all three axes. In the current Snoopy implementation we consider 41 features in total. Based on the extracted feature vector, we consider a Support Vector Machine (SVM) to label if the current frame belongs to a password input event. If so, Snoopy switches to the motion data extraction mode, which samples both the accelerometer and gyroscope at a high rate (e.g. 200Hz) and caches the data locally. This continues

until it observes a sequence of consecutive frames that are not labelled as password input. In this way, Snoopy tends to save the motion data of as many potential password input events as possible. In what follows, we show how to extract the true password input event through sequence alignment and classification.

A.1.3 Frame Smoothing and Password-positive Sequence Identification

Given a cached sequence of data frames, which correspond to a potential password input event, Snoopy needs to decide: a) the accurate starting and ending frames of this event, and b) if this candidate event is a true password input event or not. For the former task, we use a smoother to align the sequence of frames based on labels of nearby frames. The intuition is that labels of adjacent frames should be consistent, i.e. a chunk of frames should either belong to a password input event or not, but not have many interleaving labels. In Snoopy we consider two types of smoothers, one based on a Hidden Markov Model (HMM) to exploit the temporal correlations, and the other based on moving average (essentially majority voting). From the output of the smoother, Snoopy extracts the longest segment of frames whose labels are positive. If the segment length exceeds a minimum threshold, Snoopy considers this segment of motion data to be able to cover the potential password input event precisely. However in practice, the motion data extracted might not always correspond to password input; for instance, it could correspond to users tapping or swiping their smartwatches to preview email, or check upcoming calendar notifications. Therefore, given the extracted motion data, we need to identify whether it is corresponding to a true password input event, or not. Snoopy addresses this by post-hoc feeding the extracted data segment into a binary classifier, which is trained on a pre-collected motion dataset covering various user interactions. In our experiments, we find that this classification step can be efficiently run on the smartwatches in real-time. Therefore, Snoopy is able to locally identify and extract the precise segments of motion data corresponding to password input events, and only send such data to the cloud for further password inference, which will be discussed in the next section.

A.2 Performance of Password Input Extraction

Password extraction is the first and important step on the frontend of Snoopy. We evaluate this module first in different dimensions.

A.2.1 Experimental Setup

Data Collection: To evaluate the performance of password input extraction, we recruited 15 volunteers (10 males and 5 females), and asked them to wear different smartwatches (Android and iWatches) on their left wrists. Throughout our experiments we use four different models of Android watches: Sony SmartWatch3, Samsung Gear Live, Moto 360 Sports, LG Urbane, and two Apple watches: 38mm and 42mm versions of iWatch2. Note that Android smart watches typically use APLs for system level authentication, whereas Apple watches use PINs.

During the experiments, we asked the participants to perform three different types of actions: *password input* where they enter their passwords on their smartwatches; *non-password input* where they tap/swipe on the watches screen to do other tasks (e.g. preview email or check a calendar notification) but not to enter a password; and *no input* when they just perform a series of activities wearing their smartwatches, such as drinking, drawing, eating, walking, going down/upstairs, typing on keyboards and holding hands still. We designed a data collection app on the smartwatches, which samples the motion sensors in the background (100Hz in this case), and instructs the participants to perform certain actions at a given time. In this way we can obtain accurate ground truth as to when the user is performing a certain action.

To collect rich enough data, in one episode we requested a participant to at least perform three actions, where the action in the middle should be password input or non-password input action, e.g. she may first walk, then enter her password, and finally go upstairs. Each participant is requested to contribute multiple episodes, and in total we obtained 455 episodes for Android watches, and 387 for Apple watches.

Competing Approaches: Since the front-end of Snoopy has to run locally, in this series of experiments we only consider lightweight approaches that can run in real-time on the smartwatches. Recall that the task of password extraction has not been attempted on smartwatches before. Previous related work has focused on the extraction of PINs on smartphones only [189, 180, 188]. However, features used to extract keystrokes on smartphones are not suitable for smartwatches due to the limited screen size and low signal to noise ratio (see Fig. 5.2). And even in the case of smartphones, previous work has assumed that any keystroke is part of a password; however in practice, keystrokes could be used in the middle of other tasks not related to password input, for example replying to email. There is no competing approach that currently addresses the entire password extraction task. In what follows, we evaluate a realistic password extraction approach for smartwatches that has three stages. In the first stage, we assess how well we can detect the beginning of a password (see Sec. A.1.2 for details); we compare a number of classifiers including the

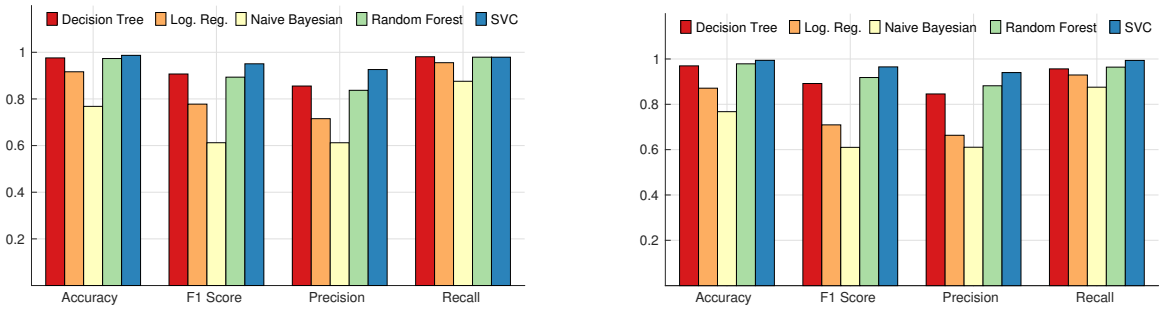


Figure A.1: Performance of detecting potential password input events. Left: PINs; Right: APLs

Support Vector Classifier (**SVC**), **decision trees**, **logistic regression**, **naive Bayesian** and **random forest** classifiers. Once we have detected the beginning of a password, in the second stage, we continue to define the full extent of the potential password, by classifying individual frames, and smoothing classification results (see Sec. A.1.3 for details). Here for smoothing, we compare the Hidden Markov Model (**HMM**) based and the voting based moving average (**moving average**) approaches implemented in Snoopy with the baseline approach without smoothing (**raw**). Once the start and end of a potential password are found, the final stage classifies this sequence as an actual password, or a non-password sequence (as discussed at the end of Sec. A.1.3); here we also compare the performance of several classifiers, including **SVC**, **decision trees**, **logistic regression**, **naive Bayesian** and **random forest**. The collected dataset is split into a training set (data from 10 subjects) and a test set (data from the other 5 subjects), and we consider 5-fold cross-validation.

A.2.2 Experiment Results

Detecting Password Input Events: As discussed in Sec. A.1.2, for a given frame of acceleration data, we would like to decide whether it corresponds to password input or not. As in many other binary classification problems, here we consider the precision, recall, F_1 score and accuracy of the classifiers. Fig. A.1 shows that the SVC outperforms competing classifiers in terms of all evaluation metrics. For both PINs and APLs, it can achieve > 0.95 F_1 scores and > 0.98 accuracy. The random forest and decision tree classifiers can achieve comparable recall with SVC, but their precision is nearly 8% lower than that of SVC. Based on these results, in what follows, we adopt SVC as the default classifier for detecting the beginning of a potential password input event in Snoopy.

Smoothing Detected Sequences: The above SVC approach is iteratively used in ensuing frames to classify them as password-related (‘1’) or not (‘0’). When we start seeing a lot of ‘0’-s this indicates the end of a potential password. The candidate password sequence

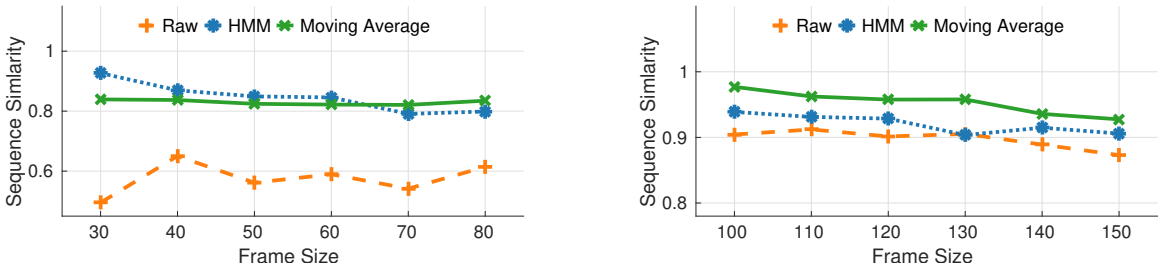


Figure A.2: Performance of sequence smoothing when using different frame sizes. Left: PINs; Right: APLs

that we derive (e.g. '11001111110001101') is then passed through a smoothing process as discussed in Sec.A.1.3. The smoother adjusts the frame labels taking into account those of nearby frames, and further refines the start and end point of the potential password input event. To evaluate the performance of the smoothing process, we consider the similarity of a frame sequence s_r with respect to the ground truth s_t:

$$d(s_r, s_t) = \frac{|s_r \cap s_t|}{(|s_r| + |s_t|)/2} \tag{A.1}$$

where $|\cdot|$ is the cardinality of the positive labels, and $|s_r \cap s_t|$ is the number of frames that have the same labels in both s_r and s_t. Intuitively a sequence s_r with higher d is better because it is closer to the ground truth, and thus tends to contain larger portion of correctly labelled frames. Fig. A.2 shows the average similarity scores of sequences generated by different smoothers (HMM and majority voting moving average vs. no smoothing). As we can see for different frame sizes, sequences without smoothing (raw) consistently have lower d scores. This confirms that the smoothing process is beneficial. For instance, for APLs, the moving average smoother can reach 0.98 in terms of sequence similarity score, while for PINs the HMM smoother can achieve 0.94. Note that for the PINs, the performance gain between not using (raw) and using smoothers (HMM and moving average) can be up to 35%. This is because the motion data generated by PINs contains "gaps" between two adjacent finger touches, where the detection approaches would naturally label frames within the gaps as negative (i.e. no password input). In those cases, the smoothers can correct those errors, and output a sequence with more consistent labels. It is also interesting to see that although the two smoothers considered in Snoopy have comparable performance, HMM tends to work better than moving average for PINs, but can be inferior for APLs. Again this is because HMM is able to mitigate those non-informative gaps within data of the PINs, while for APLs moving average is more robust.

Password-positive vs. Password-negative Sequences: Given a smoothed sequence, the front-end of Snoopy needs to identify if it corresponds to an actual password input, or non-

	F1 Score	Precision	Recall
Decision Tree	0.89	0.92	0.86
Naive Bayesian	0.91	0.90	0.91
Logistic Regression	0.95	0.96	0.93
SVC	0.93	0.92	0.95

Table A.1: **PIN** sequence identification results.

	F1 Score	Precision	Recall
Decision Tree	0.93	0.91	0.93
Naive Bayesian	0.91	0.90	0.91
Logistic Regression	0.96	0.97	0.95
SVC	0.94	0.95	0.94

Table A.2: **APL** sequence identification results.

ave./max (%)	Feature Extraction	SVM
Sony SW3	8.9/23.0	7.2/21.4
Samsung Gear Live	10.9/25.2	9.8/18.5
Moto 360 Sports	10.5/22.7	10.1/25.5

Table A.3: CPU load of running feature extraction and SVM.

password input such as swiping to check notifications. As discussed in Sec.A.1.3, Snoopy addresses this by feeding an entire sequence into a binary classifier. Tables A.1 and A.2 show the identification performance of different classifiers for PINs and APLs respectively. We see that unlike the more expensive random forest and SVC, the simpler classifiers work surprisingly well for this task. Note that for both types of passwords, the simple classifiers can achieve up to $>$0.95 F_1 score, which means they are able to reliably distinguish motion caused by password input from that of other interactions. This is because due to the small size of the smartwatches, the ways to interact with their touchscreens are very limited. Therefore the motion signals of PINs or APLs are very unique compared to others.

Resource Consumption: The final set of experiments analyse the resource consumption of Snoopy front-end on three Android watches with different hardware specifications. Tab. A.4 shows the average/maximum CPU load, delta current consumption and battery usages when the front-end is running the following three tasks: a) password input detection (Sec. A.1.2); b) password-input data smoothing and identification (Sec. A.1.3); and c) uploading extracted data to the back-end. We see that among the three tasks, uploading actually consumes the largest amount of energy, while detection and smoothing are relatively cheap. On the other hand, detection and smoothing tend to occupy the CPU more than uploading. This is expected because it is well known that transmitting over WiFi is

Model	SoC	RAM	Battery Cap.	Task	CPU load	Current Delta.	Battery (h)
Sony SW3	Qualcomm APQ8026 SD 400	512MB	420 mAh	Detection	9.2%/17.5%	8.9 mA	2%
				Smoothing	7.2%/15.2%	3.1 mA	
				Uploading	2.4%/9.9%	18.9 mA	
Samsung Gear Live	Qualcomm MSM8226 SD 400	512MB	300 mAh	Detection	8.1%/19.4%	11.4 mA	3%
				Smoothing	15.6%/29.3%	6.2 mA	
				Uploading	2.1%/19.3%	25.1 mA	
Moto 360 Sports	Qualcomm MSM8926 SD 400	512MB	300 mAh	Detection	8.3%/26.2%	16.1 mA	3%
				Smoothing	7.3%/22.4%	3.8 mA	
				Uploading	2.3%/26.1%	22.3 mA	

Table A.4: Resource consumption (CPU and power) of the Snoopy front-end on smartwatches with different hardware specs.

power-consuming on smartwatches, while detection and smoothing are more computation-intensive as they involve running SVM classifiers. More specifically, as shown in Tab. A.3, on all three watch platforms, running feature extraction and SVM classifiers consume similar level of CPU resources ($\sim$10%), but the former is slightly more expensive since it involves continuous caching operations, e.g. maintaining the sliding windows. In addition, like many other apps [208], to minimize impact on battery lifetime, Snoopy only uploads cached data when the watches are connected to power with WiFi connections available. As shown in Tab. A.4, in general the Snoopy front-end doesn't require excessive resources, and when disguised as an innocent fitness app, it is not likely to have noticeably abnormal energy/computation impact on the smartwatches.

References

[1] J. Glover, "I: The philosophy and psychology of personal identity," 1988.

[2] X. Zhu, "Semi-supervised learning literature survey," *Computer Science, University of Wisconsin-Madison*, vol. 2, no. 3, p. 4, 2006.

[3] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2018.

[4] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[5] Y. Agarwal, B. Balaji, R. Gupta, J. Lyles, M. Wei, and T. Weng, "Occupancy-driven energy management for smart building automation," in *ACM BuildSys*, 2010.

[6] R. Morris and K. Thompson, "Password security: A case history," *Communications of the ACM*, vol. 22, no. 11, pp. 594–597, 1979.

[7] N. D. Lane, *Community-Guided Mobile Phone Sensing Systems*. PhD thesis, Dartmouth College, 2011.

[8] H. Wen, Z. Xiao, A. Markham, and N. Trigoni, "Accuracy estimation for sensor systems," *IEEE Transactions on Mobile Computing*, vol. 14, no. 7, pp. 1330–1343, 2015.

[9] J. A. Stankovic, "Research directions for the internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 3–9, 2014.

[10] J. Lu, T. Sookoor, V. Srinivasan, G. Gao, B. Holben, J. Stankovic, E. Field, and K. Whitehouse, "The smart thermostat: using occupancy sensors to save energy in homes," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pp. 211–224, ACM, 2010.

[11] A. J. Goldstein, L. D. Harmon, and A. B. Lesk, "Identification of human faces," *Proceedings of the IEEE*, vol. 59, no. 5, pp. 748–760, 1971.

[12] J. L. Wayman, A. K. Jain, D. Maltoni, and D. Maio, *Biometric systems: Technology, design and performance evaluation*. Springer Science & Business Media, 2005.

[13] M. Kaluszynski, "Alphonse bertillon et l'anthropométrie," 1987.

[14] E. Higgs, "Fingerprints and citizenship: the british state and the identification of pensioners in the interwar period," in *History Workshop Journal*, vol. 69, pp. 52–67, Oxford University Press, 2010.

[15] X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler, "Design and implementation of a high-fidelity ac metering network," in *Information Processing in Sensor Networks, 2009. IPSN 2009. International Conference on*, pp. 253–264, 2009.

[16] C. X. Lu, H. Wen, S. Wang, A. Markham, and N. Trigoni, "Scan: learning speaker identity from noisy sensor data," in *ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 67–78, IEEE, 2017.

[17] A. Dantcheva, P. Elia, and A. Ross, "What else does your biometric data reveal? a survey on soft biometrics," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 3, pp. 441–467, 2016.

[18] K. Delac and M. Grgic, "A survey of biometric recognition methods," in *Proceedings. Elmar-2004. 46th International Symposium on Electronics in Marine*, pp. 184–193, IEEE, 2004.

[19] R. Y. F. Ng, Y. H. Tay, and K. M. Mok, "A review of iris recognition algorithms," in *2008 International Symposium on Information Technology*, vol. 2, pp. 1–7, IEEE, 2008.

[20] K. W. Bowyer, K. Hollingsworth, and P. J. Flynn, "Image understanding for iris biometrics: A survey," *Computer vision and image understanding*, vol. 110, no. 2, pp. 281–307, 2008.

[21] H. Farzin, H. Abrishami-Moghaddam, and M.-S. Moin, "A novel retinal identification system," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, no. 1, p. 280635, 2008.

[22] A. Hoover and M. Goldbaum, "Locating the optic nerve in a retinal image using the fuzzy convergence of the blood vessels," *IEEE transactions on medical imaging*, vol. 22, no. 8, pp. 951–958, 2003.

[23] W. F. Lane, "Self-authenticating identification card with fingerprint identification," Apr. 22 1997. US Patent 5,623,552.

[24] D. Isenor and S. G. Zaky, "Fingerprint identification using graph matching," *Pattern Recognition*, vol. 19, no. 2, pp. 113–122, 1986.

[25] D. Zhang, W.-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 25, no. 9, pp. 1041–1050, 2003.

[26] R. Sanchez-Reillo, C. Sanchez-Avila, and A. Gonzalez-Marcos, "Biometric identification through hand geometry measurements," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 10, pp. 1168–1171, 2000.

[27] F. S. Samaria and A. C. Harter, "Parameterisation of a stochastic model for human face identification," in *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, pp. 138–142, IEEE, 1994.

[28] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1891–1898, 2014.

[29] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE transactions on speech and audio processing*, vol. 3, no. 1, pp. 72–83, 1995.

[30] F. Monrose and A. Rubin, "Authentication via keystroke dynamics," in *Proceedings of the 4th ACM conference on Computer and communications security*, pp. 48–56, 1997.

[31] C. X. Lu, B. Du, P. Zhao, H. Wen, Y. Shen, A. Markham, and N. Trigoni, "Deepauth: in-situ authentication for smartwatches via deeply learned behavioural biometrics," in *Proceedings of the 2018 ACM International Symposium on Wearable Computers*, pp. 204–207, ACM, 2018.

[32] 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*, vol. 27, no. 2, pp. 162–177, 2005.

[33] P. J. Phillips, S. Sarkar, I. Robledo, P. Grother, and K. Bowyer, "The gait identification challenge problem: Data sets and baseline algorithm," in *Object recognition supported by user interaction for service robots*, vol. 1, pp. 385–388, IEEE, 2002.

[34] C. Shen, Z. Cai, R. A. Maxion, G. Xiang, and X. Guan, "Comparing classification algorithm for mouse dynamics based user identification," in *2012 IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pp. 61–66, IEEE, 2012.

[35] A. A. E. Ahmed and I. Traore, "A new biometric technology based on mouse dynamics," *IEEE Transactions on dependable and secure computing*, vol. 4, no. 3, pp. 165–179, 2007.

[36] R. Das, *Biometric technology: authentication, biocryptography, and cloud-based architecture*. CRC press, 2014.

[37] G. Parziale, E. Diaz-Santana, and R. Hauke, "The surround imager tm: a multicamera touchless device to acquire 3d rolled-equivalent fingerprints," in *International Conference on Biometrics*, pp. 244–250, Springer, 2006.

[38] J. Ashbourn, *Biometrics: Advanced identity verification: The complete guide*. Springer, 2014.

[39] N. Scaringella, G. Zoia, and D. Mlynek, "Automatic genre classification of music content: a survey," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 133–141, 2006.

[40] A. Kale, A. R. Chowdhury, and R. Chellappa, "Towards a view invariant gait recognition algorithm," in *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, 2003.*, pp. 143–150, IEEE, 2003.

[41] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal processing magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[42] P. Varchol and D. Levicky, "Using of hand geometry in biometric security systems," *Radioengineering-Prague-*, vol. 16, no. 4, p. 82, 2007.

[43] M. I. Mandel, R. J. Weiss, and D. P. Ellis, "Model-based expectation-maximization source separation and localization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 382–394, 2010.

[44] S. Song, K. D. Miller, and L. F. Abbott, "Competitive hebbian learning through spike-timing-dependent synaptic plasticity," *Nature neuroscience*, vol. 3, no. 9, p. 919, 2000.

[45] F. W. M. H. Wong, A. S. M. Supian, A. F. Ismail, L. W. Kin, and O. C. Soon, "Enhanced user authentication through typing biometrics with artificial neural networks and k-nearest neighbor algorithm," in *Conference Record of Thirty-Fifth Asilomar Conference on Signals, Systems and Computers (Cat. No. 01CH37256)*, vol. 2, pp. 911–915, IEEE, 2001.

[46] M. Rizk and E. Koosha, "A comparison of principal component analysis and generalized hebbian algorithm for image compression and face recognition," in *2006 International Conference on Computer Engineering and Systems*, pp. 214–219, IEEE, 2006.

[47] D. Reynolds, "Gaussian mixture models," *Encyclopedia of biometrics*, pp. 827–832, 2015.

[48] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[49] S. J. McKenna, S. Gong, and Y. Raja, "Modelling facial colour and identity with gaussian mixtures," *Pattern recognition*, vol. 31, no. 12, pp. 1883–1892, 1998.

[50] J. V. Soares, J. J. Leandro, R. M. Cesar, H. F. Jelinek, and M. J. Cree, "Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification," *IEEE Transactions on medical Imaging*, vol. 25, no. 9, pp. 1214–1222, 2006.

[51] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[52] O. M. Parkhi, A. Vedaldi, A. Zisserman, *et al.*, "Deep face recognition.," in *BMVC*, vol. 1, p. 6, 2015.

[53] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *INTERSPEECH*, 2017.

[54] A. Gangwar and A. Joshi, "Deepirisnet: Deep iris representation with applications in iris recognition and cross-sensor iris recognition," in *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 2301–2305, IEEE, 2016.

[55] J. Hannink, T. Kautz, C. F. Pasluosta, K.-G. Gaßmann, J. Klucken, and B. M. Eskofier, "Sensor-based gait parameter extraction with deep convolutional neural networks," *IEEE journal of biomedical and health informatics*, vol. 21, no. 1, pp. 85–93, 2017.

[56] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8, IEEE, 2007.

[57] B. Moghaddam, W. Wahid, and A. Pentland, "Beyond eigenfaces: Probabilistic matching for face recognition," in *fg*, p. 30, IEEE, 1998.

[58] L. Burget, O. Plchot, S. Cumani, O. Glembek, P. Matějka, and N. Brümmer, "Discriminatively trained probabilistic linear discriminant analysis for speaker verification," in *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4832–4835, IEEE, 2011.

[59] A. Verikas, A. Gelzinis, and M. Bacauskiene, "Mining data with random forests: A survey and results of new tests," *Pattern recognition*, vol. 44, no. 2, pp. 330–349, 2011.

[60] E. Kremic and A. Subasi, "Performance of random forest and svm in face recognition.," *Int. Arab J. Inf. Technol.*, vol. 13, no. 2, pp. 287–293, 2016.

[61] G.-Y. Zhang, C.-X. Zhang, and J.-S. Zhang, "Out-of-bag estimation of the optimal hyperparameter in subbag ensemble method," *Communications in Statistics-Simulation and Computation*, vol. 39, no. 10, pp. 1877–1892, 2010.

[62] B. Scholkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001.

[63] P. J. Phillips, "Support vector machines applied to face recognition," in *Advances in Neural Information Processing Systems*, pp. 803–809, 1999.

[64] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using gmm supervectors for speaker verification," *IEEE signal processing letters*, vol. 13, no. 5, pp. 308–311, 2006.

[65] M. Vatsa, R. Singh, and A. Noore, "Improving biometric recognition accuracy and robustness using dwt and svm watermarking," *IEICE Electronics Express*, vol. 2, no. 12, pp. 362–367, 2005.

[66] E. Yu and S. Cho, "Ga-svm wrapper approach for feature subset selection in keystroke dynamics identity verification," in *Proceedings of the International Joint Conference on Neural Networks, 2003.*, vol. 3, pp. 2253–2257, IEEE, 2003.

[67] C. Shen, Z. Cai, X. Guan, Y. Du, and R. A. Maxion, "User authentication through mouse dynamics," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 16–30, 2013.

[68] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[69] G. Nimrod, A. Szilágyi, C. Leslie, and N. Ben-Tal, "Identification of dna-binding proteins using structural, electrostatic and evolutionary features," *Journal of molecular biology*, vol. 387, no. 4, pp. 1040–1053, 2009.

[70] H.-T. Chen, T.-L. Liu, and C.-S. Fuh, "Segmenting highly articulated video objects with weak-prior random forests," in *European Conference on Computer Vision*, pp. 373–385, 2006.

[71] B. Schölkopf and C. J. Burges, *Advances in kernel methods: support vector learning*. MIT press, 1999.

[72] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1701–1708, 2014.

[73] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pp. 912–919, 2003.

[74] X. Zhu and J. Lafferty, "Harmonic mixtures: combining mixture models and graph-based methods for inductive and scalable semi-supervised learning," in *Proceedings of the 22nd international conference on Machine learning*, pp. 1052–1059, ACM, 2005.

[75] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[76] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of machine learning research*, vol. 7, no. Nov, pp. 2399–2434, 2006.

[77] M. Belkin and P. Niyogi, "Using manifold stucture for partially labeled classification," in *Advances in neural information processing systems*, pp. 953–960, 2003.

[78] A. Corduneanu and T. Jaakkola, "On information regularization," in *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pp. 151–158, 2002.

[79] F. Roli and G. L. Marcialis, "Semi-supervised pca-based face recognition using self-training," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pp. 560–568, Springer, 2006.

[80] M. Yamada, M. Sugiyama, and T. Matsui, "Semi-supervised speaker identification under covariate shift," *Signal Processing*, vol. 90, no. 8, pp. 2353–2361, 2010.

[81] Y. Li, Y. Yin, L. Liu, S. Pang, and Q. Yu, "Semi-supervised gait recognition based on self-training," in *2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, pp. 288–293, IEEE, 2012.

[82] P. J. Moreno, C. Joerg, J.-M. V. Thong, and O. Glickman, "A recursive algorithm for the forced alignment of very long audio segments," in *Fifth International Conference on Spoken Language Processing*, 1998.

[83] X. R. Li and Y. Bar-Shalom, "Tracking in clutter with nearest neighbor filters: analysis and performance," *IEEE transactions on aerospace and electronic systems*, vol. 32, no. 3, pp. 995–1010, 1996.

[84] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "People tracking with mobile robots using sample-based joint probabilistic data association filters," *The International Journal of Robotics Research*, vol. 22, no. 2, pp. 99–116, 2003.

[85] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.

[86] A. Hadid, "Face biometrics under spoofing attacks: Vulnerabilities, countermeasures, open issues, and research directions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 113–118, 2014.

[87] L. T. Heberlein and M. Bishop, "Attack class: Address spoofing," in *Proceedings of the 19th National Information Systems Security Conference*, pp. 371–377, 1996.

[88] F. Guo and T.-c. Chiueh, "Sequence number-based mac address spoof detection," in *International Workshop on Recent Advances in Intrusion Detection*, pp. 309–329, Springer, 2005.

[89] X. Zhao, G. Chen, and K. Dong, "Techniques for protecting telephone users from caller id spoofing attacks," Mar. 13 2012. US Patent 8,135,119.

[90] K. Pandove, A. Jindal, and R. Kumar, "Email spoofing," *International Journal of Computer Applications*, vol. 5, no. 1, pp. 27–30, 2010.

[91] Y. S. Moon, J. Chen, K. Chan, K. So, and K. Woo, "Wavelet based fingerprint liveness detection," *Electronics Letters*, vol. 41, no. 20, pp. 1112–1113, 2005.

[92] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *2011 international joint conference on Biometrics (IJCB)*, pp. 1–7, IEEE, 2011.

[93] S. K. Ergünay, E. Khoury, A. Lazaridis, and S. Marcel, "On the vulnerability of speaker verification to realistic voice spoofing," in *2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1–6, IEEE, 2015.

[94] S. J. Templeton and K. E. Levitt, "Detecting spoofed packets," in *Proceedings DARPA Information Survivability Conference and Exposition*, vol. 1, pp. 164–175, IEEE, 2003.

[95] V. Kumar, J. Srivastava, and A. Lazarevic, *Managing cyber threats: issues, approaches, and challenges*, vol. 5. Springer Science & Business Media, 2006.

[96] A. T. B. Jin, D. N. C. Ling, and A. Goh, "Biohashing: two factor authentication featuring fingerprint data and tokenised random number," *Pattern recognition*, vol. 37, no. 11, pp. 2245–2255, 2004.

[97] G. Joy Persial, M. Prabhu, and R. Shanmugalakshmi, "Side channel attack-survey," *Int J Adva Sci Res Rev*, vol. 1, no. 4, pp. 54–57, 2011.

[98] J.-F. Dhem, F. Koeune, P.-A. Leroux, P. Mestré, J.-J. Quisquater, and J.-L. Willems, "A practical implementation of the timing attack," in *International Conference on Smart Card Research and Advanced Applications*, pp. 167–182, Springer, 1998.

[99] F. Amiel, K. Villegas, B. Feix, and L. Marcel, "Passive and active combined attacks: Combining fault attacks and side channel analysis," in *Fault Diagnosis and Tolerance in Cryptography, 2007. FDTC 2007. Workshop on*, pp. 92–102, IEEE, 2007.

[100] D. Réal, F. Valette, and M. Drissi, "Enhancing correlation electromagnetic attack using planar near-field cartography," in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 628–633, European Design and Automation Association, 2009.

[101] M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, and C. Sporleder, "Acoustic side-channel attacks on printers.," in *USENIX Security symposium*, pp. 307–322, 2010.

[102] S. Kadloor, X. Gong, N. Kiyavash, T. Tezcan, and N. Borisov, "Low-cost side channel remote traffic analysis attack in packet networks," in *Communications (ICC), 2010 IEEE International Conference on*, pp. 1–5, IEEE, 2010.

[103] Y. Zhou and D. Feng, "Side-channel attacks: Ten years after its publication and the impacts on cryptographic module security testing.," *IACR Cryptology ePrint Archive*, vol. 2005, p. 388, 2005.

[104] I. Kubiak and A. Przybysz, "Laser printers and effectiveness of attack type tempest," *Warsaw, Poland: Publisher House of Military University of Technology*, 2016.

[105] S. Guilley, P. Hoogvorst, R. Pacalet, and J. Schmidt, "Improving side-channel attacks by exploiting substitution boxes properties," in *International Conference on Boolean Functions: Cryptography and Applications (BFCA)*, pp. 1–25, 2007.

[106] J. Fan, X. Guo, E. De Mulder, P. Schaumont, B. Preneel, and I. Verbauwhede, "State-of-the-art of secure ecc implementations: a survey on known side-channel attacks and countermeasures," in *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pp. 76–87, IEEE, 2010.

[107] E. Prouff and M. Rivain, "Masking against side-channel attacks: A formal security proof," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 142–159, Springer, 2013.

[108] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[109] L. Sweeney, "Weaving technology and policy together to maintain confidentiality," *The Journal of Law, Medicine & Ethics*, vol. 25, no. 2-3, pp. 98–110, 1997.

[110] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pp. 111–125, IEEE, 2008.

[111] D. Frankowski, D. Cosley, S. Sen, L. Terveen, and J. Riedl, "You are what you say: privacy risks of public mentions," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 565–572, ACM, 2006.

[112] N. D. Lane, J. Xie, T. Moscibroda, and F. Zhao, "On the feasibility of user deanonymization from shared mobile sensor data," in *Proceedings of the Third International Workshop on Sensing Applications on Mobile Phones*, p. 3, ACM, 2012.

[113] R. Jia, F. C. Sangogboye, T. Hong, C. Spanos, and M. B. Kjærgaard, "Pad: Protecting anonymity in publishing building related datasets," in *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments*, p. 4, ACM, 2017.

[114] E. Zheleva and L. Getoor, "To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles," in *Proceedings of the 18th international conference on World wide web*, pp. 531–540, ACM, 2009.

[115] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: inferring user profiles in online social networks," in *Proceedings of the third ACM international conference on Web search and data mining*, pp. 251–260, ACM, 2010.

[116] Y. Shen, F. Wang, and H. Jin, "Defending against user identity linkage attack across multiple online social networks," in *Proceedings of the 23rd International Conference on World Wide Web*, pp. 375–376, ACM, 2014.

[117] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

[118] Mindsight, "Pay with your face — facial recognition software," 2018.

[119] F. T. Council, "12 exciting ways you can use voice-activated technology in the workplace," 2018.

[120] T. E. Witness, "What is forensic gait analysis?," 2015.

[121] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, 2004.

[122] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[123] T. Guardian, "Uk police use of facial recognition technology a failure," 2018.

[124] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based cnn with improved triplet loss function," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1335–1344, 2016.

[125] R. Jonker and T. Volgenant, "Improving the hungarian assignment algorithm," *Operations Research Letters*, vol. 5, no. 4, pp. 171–175, 1986.

[126] K. Wang, N. An, B. N. Li, Y. Zhang, and L. Li, "Speech emotion recognition using Fourier parameters," *IEEE Transactions on Affective Computing*, 2015.

[127] N. Karmarkar, "A new polynomial-time algorithm for linear programming," in *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, pp. 302–311, ACM, 1984.

[128] M. L. Fisher, "The lagrangian relaxation method for solving integer programming problems," *Management science*, vol. 27, no. 1, pp. 1–18, 1981.

[129] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on Audio, Speech, and Language Processing*, 2006.

[130] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 2011.

[131] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," 2015. arXiv:1510.08484v1.

[132] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[133] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning.," in *AAAI*, 2017.

[134] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *IEEE Conference on Automatic Face and Gesture Recognition*, 2018.

[135] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *INTERSPEECH*, 2018.

[136] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," 2011.

[137] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in neural information processing systems*, pp. 849–856, 2002.

[138] D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log," in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 407–416, ACM, 2000.

[139] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[140] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, *et al.*, "Devise: A deep visual-semantic embedding model," in *Advances in neural information processing systems*, pp. 2121–2129, 2013.

[141] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128–3137, 2015.

[142] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pp. 3156–3164, IEEE, 2015.

[143] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," *arXiv preprint arXiv:1412.4729*, 2014.

[144] A. Owens, J. Wu, J. H. McDermott, W. T. Freeman, and A. Torralba, "Ambient sound provides supervision for visual learning," in *European Conference on Computer Vision*, pp. 801–816, Springer, 2016.

[145] A. Nagrani, S. Albanie, and A. Zisserman, "Seeing voices and hearing faces: Cross-modal biometric matching," *arXiv preprint arXiv:1804.00326*, 2018.

[146] C. L. Zitnick, D. Parikh, and L. Vanderwende, "Learning the visual interpretation of sentences," in *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 1681–1688, IEEE, 2013.

[147] J. Teng, B. Zhang, J. Zhu, X. Li, D. Xuan, and Y. F. Zheng, "Ev-loc: integrating electronic and visual signals for accurate localization," *IEEE/ACM Transactions on Networking*, 2014.

[148] A. Alahi, A. Haque, and L. Fei-Fei, "Rgb-w: When vision meets wireless," in *2015 IEEE International Conference on Computer Vision (ICCV)*, no. EPFL-CONF-230282, pp. 3289–3297, IEEE, 2015.

[149] S. Papaioannou, H. Wen, Z. Xiao, A. Markham, and N. Trigoni, "Accurate positioning via cross-modality training," in *ACM SenSys*, 2015.

[150] S. S. Blackman, "Multiple-target tracking with radar applications," *Dedham, MA, Artech House, Inc., 1986, 463 p.*, 1986.

[151] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 34, no. 3, pp. 334–352, 2004.

[152] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Systems*, vol. 29, no. 6, 2009.

[153] D. Wang, T. Abdelzaher, L. Kaplan, and C. Aggarwal, "Recursive fact-finding: A streaming approach to truth estimation in crowdsourcing applications," in *Proc . ICDCS*, 2013.

[154] C. Huang and D. Wang, "Topic-aware social sensing with arbitrary source dependency graphs," in *IEEE IPSN*, 2016.

[155] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

[156] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *ECCV*, 2016.

[157] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 13, no. 8, pp. 841–847, 1991.

[158] P. Tarrio, A. M. Bernardos, and J. R. Casar, "An rss localization method based on parametric channel models," in *2007 International Conference on Sensor Technologies and Applications (SENSORCOMM 2007)*, pp. 265–270, IEEE, 2007.

[159] S. Shum, N. Dehak, E. Chuangsuwanich, D. A. Reynolds, and J. R. Glass, "Exploiting intra-conversation variability for speaker diarization.," in *INTERSPEECH*, 2011.

[160] S. He and S.-H. G. Chan, "Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466–490, 2016.

[161] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[162] W. Liu, Y.-M. Zhang, X. Li, Z. Yu, B. Dai, T. Zhao, and L. Song, "Deep hyperspherical learning," in *Advances in Neural Information Processing Systems*, pp. 3953–3963, 2017.

[163] M. Wang and W. Deng, "Deep face recognition: A survey," *arXiv preprint arXiv:1804.06655*, 2018.

[164] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[165] X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang, "Face recognition from a single image per person: A survey," *Pattern recognition*, vol. 39, no. 9, pp. 1725–1745, 2006.

[166] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," tech. rep., Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[167] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Advances in neural information processing systems*, pp. 1988–1996, 2014.

[168] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1891–1898, 2014.

[169] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks.," in *ICML*, pp. 507–516, 2016.

[170] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[171] L. Li, D. Wang, Z. Zhang, and T. F. Zheng, "Deep speaker vectors for semi text-independent speaker verification," *CoRR*, vol. abs/1505.06427, 2015.

[172] A. M. Research, "Smartwatch market is expected to reach 32.9 billion, globally, by 2020." https://goo.gl/DJjHYs, 2016.

[173] T. Graziani, "smartwatch report." https://walkthechat.com/apple-watch-wechat-are-adding-further-integrated-features/, 2016.

[174] L. Alipay.com Co., "Alipay - makes life easy." https://itunes.apple.com/us/app/alipay-makes-life-easy/id333206289?mt=8, 2017.

[175] Baidu, "Offline Alipay Setup." http://jingyan.baidu.com/article/ce4366492b02263773afd3f0.html, 2016.

[176] E. Stobert and R. Biddle, "The password life cycle: user behaviour in managing passwords," in *USENIX Symposium On Usable Privacy and Security, SOUPS*, 2014.

[177] A. J. Aviv, K. L. Gibson, E. Mossop, M. Blaze, and J. M. Smith, "Smudge attacks on smartphone touch screens.," *USENIX Workshop on Offensive Technologies, Woot*, 2010.

[178] G. Ye, Z. Tang, D. Fang, X. Chen, K. I. Kim, B. Taylor, and Z. Wang, "Cracking android pattern lock in five attempts," 2017.

[179] M. Li, Y. Meng, J. Liu, H. Zhu, X. Liang, Y. Liu, and N. Ruan, "When csi meets public wifi: Inferring your mobile phone password via wifi signals," in *ACM SIGSAC Conference on Computer and Communications Security, CCS*, 2016.

[180] L. Cai and H. Chen, "On the practicality of motion based keystroke inference attack," in *International Conference on Trust and Trustworthy Computing, TRUST*, 2012.

[181] E. Von Zezschwitz, P. Dunphy, and A. De Luca, "Patterns in the wild: a field study of the usability of pattern and pin-based authentication on mobile devices," in *ACM Conference on Human Factors in Computing Systems, CHI*, 2013.

[182] C. J. Turner, B. S. Chaparro, and J. He, "Text input on a smartwatch qwerty keyboard: tap vs. trace," *International Journal of Human–Computer Interaction*, 2017.

[183] X. Liu, Z. Zhou, W. Diao, Z. Li, and K. Zhang, "When good becomes evil: Keystroke inference with smartwatch," in *CCS*, ACM, 2015.

[184] D. Shukla, R. Kumar, A. Serwadda, and V. V. Phoha, "Beware, your hands reveal your secrets!," in *ACM SIGSAC Conference on Computer and Communications Security, CCS*, 2014.

[185] S. Uellenbeck, M. Dürmuth, C. Wolf, and T. Holz, "Quantifying the security of graphical passwords: the case of android unlock patterns," in *ACM SIGSAC conference on Computer & communications security, CCS*, 2013.

[186] M. D. Løge, "Tell me who you are and i will tell you your unlock pattern," Master's thesis, NTNU, 2015.

[187] A. J. Aviv, B. Sapp, M. Blaze, and J. M. Smith, "Practicality of accelerometer side channels on smartphones," in *Proceedings of the 28th Annual Computer Security Applications Conference, ACSAC*, 2012.

[188] E. Miluzzo, A. Varshavsky, S. Balakrishnan, and R. R. Choudhury, "Tapprints: your finger taps have fingerprints," in *International Conference on Mobile Systems, Applications, and Services, MobiSys*, ACM, 2012.

[189] Z. Xu, K. Bai, and S. Zhu, "Taplogger: Inferring user inputs on smartphone touchscreens using on-board motion sensors," in *Proceedings of the 5th ACM conference on Security and Privacy in Wireless and Mobile Networks*, 2012.

[190] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "Accessory: password inference using accelerometers on smartphones," in *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications, HotMobile*, 2012.

[191] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems, NIPS*, 2014.

[192] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, "Teaching machines to read and comprehend," in *Advances in Neural Information Processing Systems, NIPS*, 2015.

[193] F. Chollet *et al.*, "Keras: Deep learning library for theano and tensorflow," *URL: https://keras. io/k*, 2015.

[194] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations, ICLR*, 2014.

[195] J. L. Massey, "Guessing and entropy," in *IEEE International Symposium on Information Theory (ISIT)*, 1994.

[196] M. Harbach, A. De Luca, and S. Egelman, "The anatomy of smartphone unlocking: A field study of android lock screens," in *ACM Conference on Human Factors in Computing Systems, CHI*, 2016.

[197] D. Amitay, "Most Common iPhone Passcodes." http://danielamitay.com/blog/2011/6/13/most-common-iphone-passcodes, 2014.

[198] J. Nam, J. Paik, H.-K. Kang, U. M. Kim, and D. Won, "An off-line dictionary attack on a simple three-party key exchange protocol," *IEEE Communications Letters*, vol. 13, no. 3, pp. 205–207, 2009.

[199] M. Kumar, T. Garfinkel, D. Boneh, and T. Winograd, "Reducing shoulder-surfing by using gaze-based password entry," in *Proceedings of the 3rd symposium on Usable privacy and security*, pp. 13–19, ACM, 2007.

[200] N. Zheng, K. Bai, H. Huang, and H. Wang, "You are how you touch: User verification on smartphones via tapping behaviors," in *Network Protocols (ICNP), 2014 IEEE 22nd International Conference on*, pp. 221–232, IEEE, 2014.

[201] M. Shahzad, A. X. Liu, and A. Samuel, "Secure unlocking of mobile touch screen devices by simple gestures: you can see it but you can not do it," in *Proceedings of the 19th annual international conference on Mobile computing & networking, (MobiCom)*, 2013.

[202] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using wifi signals," in *ACM MobiCom*, 2015.

[203] Y. Michalevsky, D. Boneh, and G. Nakibly, "Gyrophone: Recognizing speech from gyroscope signals," in *USENIX Security*, 2014.

[204] P. Marquardt, A. Verma, H. Carter, and P. Traynor, "(sp) iphone: Decoding vibrations from nearby keyboards using mobile phone accelerometers," in *ACM SIGSAC Conference on Computer and Communications Security, CCS*, 2011.

[205] L. Cai and H. Chen, "Touchlogger: Inferring keystrokes on touch screen from smartphone motion.," in *Proceedings of the 6th USENIX Conference on Hot Topics in Security, HotSec*, USENIX, 2011.

[206] M. Mehrnezhad, E. Toreini, S. F. Shahandashti, and F. Hao, "Stealing pins via mobile sensors: actual risk versus user perception," *International Journal of Information Security*, 2017.

[207] H. Wang, T. T.-T. Lai, and R. Roy Choudhury, "Mole: Motion leaks through smartwatch sensors," in *ACM MobiCom*, 2015.

[208] C. Liu, L. Zhang, Z. Liu, K. Liu, X. Li, and Y. Liu, "Lasagna: towards deep hierarchical understanding and searching over mobile sensing data," in *ACM Conference on Mobile Computing and Networking, MobiCom*, 2016.

[209] A. Maiti, O. Armbruster, M. Jadliwala, and J. He, "Smartwatch-based keystroke inference attacks and context-aware protection mechanisms," in *CM on Asia Conference on Computer and Communications Security AsiaCCS*, 2016.

[210] H. Kim and et al., "Wearable device and method of operating the same," 2017.

[211] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song, "Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication," 2013.

[212] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in Neural Information Processing Systems*, pp. 1024–1034, 2017.

[213] J. Macker, "Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations," 1999.

[214] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Lightweight map matching for indoor localisation using conditional random fields," in *ACM/IEEE IPSN*, 2014.