

Received November 16, 2021, accepted January 15, 2022, date of publication January 31, 2022, date of current version February 8, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3148051

Epidemic Exposure Tracking With Wearables: A Machine Learning Approach to Contact Tracing

PAI CHET NG^{1,2}, (Member, IEEE), PETROS SPACHOS¹, (Senior Member, IEEE),
STEFANO GREGORI¹, (Senior Member, IEEE), AND
KONSTANTINOS N. PLATANIOTIS², (Fellow, IEEE)

¹School of Engineering, University of Guelph, Guelph, ON N1G 2W1, Canada

²Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G8, Canada

Corresponding author: Petros Spachos (petros@uoguelph.ca)

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) through the NSERC Alliance COVID-19 grant #401656.

ABSTRACT The recent pandemic revealed weaknesses in several areas, including the limited capacity of public health systems for efficient case tracking and reporting. In the post-pandemic era, it is essential to be ready and provide not only preventive measures, but also effective digital strategies and solutions to protect our population from future outbreaks. This work presents a contact tracing solution based on wearable devices to track epidemic exposure. Our proximity-based privacy-preserving contact tracing (P3CT) integrates: 1) the Bluetooth Low Energy (BLE) technology for reliable proximity sensing, 2) a machine-learning approach to classify the exposure risk of a user, and 3) an ambient signature protocol for preserving the user's identity. Proximity sensing exploits the signals emitted from a smartwatch to estimate users' interaction, in terms of distance and duration. Supervised learning is then used to train four classification models to identify the exposure risk of a user with respect to a patient diagnosed with an infectious disease. Finally, our proposed P3CT protocol uses ambient signatures to anonymize the infected patient's identity. Extensive experiments demonstrate the feasibility of our proposed solution for real-world contact tracing problems. The large-scale dataset consisting of the signal information collected from the smartwatch is available online. According to experimental results, wearable devices along with machine learning models are a promising approach for epidemic exposure notification and tracking.

INDEX TERMS Bluetooth low energy, contact tracing, disease outbreak, physical distancing, smartwatch, wearables, COVID-19, proximity sensing, pandemic, artificial intelligence, supervised learning, neural network.

I. INTRODUCTION

The global and highly contagious COVID-19 pandemic affected everyday activities in relation to government's lockdowns and restrictions for slowing down and stopping the spread. Even though such measures can effectively contain the pandemic at least for a short-term period, they create adverse effects on long-term economic and social developments [1]. A long-term solution, which can balance our daily life while preventing the further spreading of the virus, would be more practical than restrictive measures. To date, several countries have begun to relax their restriction allowing business reopening and supporting people to return to work.

The associate editor coordinating the review of this manuscript and approving it for publication was Tao Liu¹.

While relaxing the restrictions can help the economic growth, essential preventive measures must be applied to protect workers and customers alike from the next outbreak. Among those measures, such as temperature checking, wearing a face mask, and practicing hand hygiene, contact tracing is deemed essential in monitoring the daily interaction between users and thus providing an immediate alert to all the users when someone is diagnosed with an infectious disease [2]–[4].

While several smartphone-based contact tracing solutions (e.g., Pan European Privacy-Preserving Proximity Tracing (PEPP-PT) [5], COVID-19 Watch [6], Privacy-Preserving Automated Contact Tracing (PACT) [7], etc.) are available nowadays, these solutions might not be effective in a working environment because the user does not necessarily carry with them the smartphone all the time due to the inherent nature of

their work. Furthermore, many people might put their smartphones inside their pocket or backpack, which increases the difficulty in producing proximity sensing with satisfactory performance.

The goal of smartphone-based contact tracing is to classify the risk of a user based on the proximity information extracted from the Bluetooth Low Energy (BLE) signals. According to our previous work for smartphone-based contact tracing [8], only when both users were holding the smartphone in their hands, the performance was high and about 90%. Five classifiers were trained to classify the risk given the received signal strength (RSS) measured from the smartphone over a certain period. Our experimental result shows that the classification performance drops severely when there is non-line-of-sight (NLOS) between any two smartphones, for example, when one of the smartphones is in the pocket and another in the backpack. Since machine learning is a data-driven approach, sufficient data is necessary to train a good classification model. Data-driven approaches have been applied to many COVID-19 problems, as reviewed by [9], we can see that most of these problems utilize the widely available dataset from computed tomography scans, textual data, sound data, and embedded sensor data to either classify the COVID-19 diagnosis result [10], identify the disease infection symptom [11], detect abnormal pattern from radiographical signal [12], construct a predictive model [13], etc. Unlike those widely accessible images and sensor datasets for COVID-19 [14], the available data containing radio signal measurements from mobile and wearable devices is relatively rare, and most of them did not cover every possible aspect of signal measurements. Lastly, the privacy issue is still the top concern for contact tracing applications. Even though one can exploit the encryption method to encrypt the user's information, such an encryption method can be decrypted easily once the encryption key leaks.

An effective and low-cost contact tracing solution that can be adopted by many users without affecting their work routine is deemed necessary to track the exposure risk of users, who need to constantly perform their job in a work environment with limited access to their smartphones. Motivated by the limitation of the smartphone-based approach in facilitating contact tracing in a work environment, this work proposes a wearable contact tracing solution based on a low-cost smartwatch, namely proximity-based privacy-preserving contact tracing (P³CT). Our proposed P³CT addresses the questions we highlighted above by 1) exploiting the BLE signals to monitor the interaction between users, 2) leveraging machine learning to train classification models for classifying the exposure risk, and 3) designing a novel ambient signature protocol to anonymize users' identify. Since the antenna position and form factor of a smartwatch is different from a smartphone, the received signal strength (RSS) measured by the smartwatch might exhibit different signal behaviors, in which it is impossible to implement the existing contact tracing solution developed with the smartphone to the smartwatch platform directly without understanding

the statistical characteristics of RSS values measured by the smartwatch.

In contrast to most contact tracing solutions that identify a high-risk user (i.e., the user who is most likely to contract the virus) based on the proximity information estimated from the given RSS values, our proposed P³CT identifies the high-risk user by jointly considering the interaction range and interaction duration when any two users come into closed proximity. This is inspired by the fact that, according to epidemiologists, the exposure risk is low if the user spent less than 1 s in close proximity to the infected patient, compared to the user who spent more than 1 hr in not so close proximity, yet still relatively near (i.e., the smartwatch still in the broadcasting range), to the infected patient [15]. Given the RSS data containing the interaction range and interaction duration information, we use the machine learning approach to train four classification models to evaluate the performance of our proposed P³CT. Our ambient signature protocol, on the other hand, reuses the same set of RSS measurements to anonymize the user's identity.

The major contributions of our proposed P³CT are summarized as follows:

- Accurate proximity sensing: A comprehensive performance evaluation of RSS-based proximity sensing is provided to verify the feasibility of using RSS from wearable devices. While RSS suffers severe attenuation due to the effect of the human body, our empirical analysis verifies that P³CT achieves satisfactory performance with existing classification methods.
- Risk classification: we jointly consider the interaction range and interaction duration when defining the classification model. Four classification methods are examined while other possible input features are also explored, including the number of samples observed by the smart-watch, the maximum RSS, the minimum RSS, and the range of RSS measurements at a particular interval.
- Real-world dataset: Our experimental results were validated with real-world datasets collected from smartwatches worn on the human wrists. We consolidate the data and organize them into training and testing sets according to the 80%-20% splitting rule. The dataset is publicly available to encourage further research [16].
- Real-time exposure alert: By exploiting a low-cost commercial off-the-shelf smartwatch equipped with BLE technology, smartwatch-based contact tracing can be a cost-effective solution in many workplaces. The implementation of our proposed P³CT into these low-cost smartwatches demonstrate the practicality of our proposed solution for contact tracing, as well as its ability to trigger real-time exposure alert.

The rest of the paper is organized as follows. Section II provides the background related to contact tracing and discusses its current development. Section III presents our proposed P³CT. Section IV describes the method to classify the

risk level. Section V discusses our experimental evaluations. Section VI concludes this work.

II. BACKGROUND AND MOTIVATION

Traditional contact tracing relies on manual human efforts, such as conducting interviews with the patient, tracking down the people who have come in close contact with the patient for the past few days. Undoubtedly, manual contact tracing is not only labor-intensive but also is slow in fighting the rapid virus spreading rate [3], [17]. Recognizing the urgency to have a more effective contact tracing, this section reviews emerging digital-based solutions and then discusses the current development in contact tracing.

A. DIGITAL-BASED CONTACT TRACING

To date, many digital-based contact tracing solutions have been developed to automatically identify a group of users who are more likely at risk, while preserving the private information of each user. These digital-based contact tracing solutions can be categorized into the following two types:

1) SMARTPHONE-BASED CONTACT TRACING

Pervasive smartphones are the popular option for digital-based contact tracing due to their rich sensing features, providing a better estimation of interaction distance and duration. Many works leverage geolocation information [18], [19] and proximity sensing [5] to monitor the interaction between any two users. Besides homogeneous sensing, there are also works exploiting the heterogeneous sensing features to improve the distance estimation [20]. However, these works fail to consider the location of the smartphone when the users are doing grocery or working. While people might carry the smartphone with them for grocery shopping, the smartphone will be either holding on hands or sitting inside the pocket. Such a holding variation might affect the accuracy of distance estimation and thus confuse the contact tracing process. In the work environment, on the other hand, people might not carry their smartphones with them all the time.

2) WEARABLE-BASED CONTACT TRACING

While some works utilize the physiological signals [21] or activity tracker data [22] from the smartwatch to detect the possible symptoms to contract COVID-19, there are not many works utilizing the wireless signal from the smartwatch for contact tracing. Considering the high variability with the smartphone's use cases, many industries start to exploit the wearable solution to contact tracing [23]. The main motivation to exploit the wearable solution is that they can allow their workers to resume the work routine with less distraction. For example, EasyBand [24] presents a wearable-based contact tracing to facilitate safe social distancing practice. The EasyBand uses a centralized server for contact tracing, in which all the users' data is uploaded to the cloud through TCP/IP sockets. Such a centralized approach is not scalable since it relies on cloud computations to identify all the possible contact for all the infected patients. Furthermore, there

is a high possibility of information leaks when the server is compromised.

B. CURRENT DEVELOPMENT IN CONTACT TRACING

Recognizing the importance of contact tracing in resuming the normal lifestyle while preventing the further spread of the contagious virus, industry and academia have devoted efforts to developing a more effective contact tracing solution to fight against COVID-19.

1) NATIONAL-LEVEL EFFORTS

China, South Korea, and Singapore are among the first countries that have implemented the digital-based contact tracing solution. With its country-wide surveillance systems, China government deployed a close contact detector based on QR code [25]. South Korea utilizes the location data (i.e., the GPS data) from the smartphone to detect the location of the infected patient and push a notification containing personal details of the infected patient to the nearby users [18]. Singapore developed a smartphone application, known as TraceTogether, that exploits BLE signals transmitted by the smartphone to detect the proximity between any two users [26]. In general, the digital contact tracing deployed by China and South Korea is more intrusive compared to the TraceTogether developed by Singapore aiming to protect the user's privacy by tracking only the proximity between any two users without explicit location information.

2) ACADEMIA-LEVEL EFFORTS

In contrast to the intrusive approach, academic researchers have initiated several privacy-preserving contact tracing solutions [27], [28]. For example, Pan European Privacy-Preserving Proximity Tracing (PEPP-PT) estimates the proximity based on the broadcast BLE packet containing a full anonymous ID [5]. COVID-19 Watch automatically alerts the user when he/she is suspected to be in contact with the infected patient [6]. The Privacy-Preserving Automated Contact Tracing (PACT) exploits the BLE signals in combination with secure encryption to detect possible contacts while protecting users' privacy [7].

Even though many initiatives exploit the BLE signals for contact tracing purposes, most of the contact tracing solutions simply develop their application assuming perfect proximity sensing scenarios with BLE signals. Unfortunately, BLE signals from the smartphone are highly inconsistent regardless of the smartphone is held steadily and remains stationary in the same location. So far, there is no work examining the BLE signals transmitted by the smartwatch for contact tracing. Considering the form factor of the smartwatch, as well as its processing capability, the BLE signals from the smartwatch might suffer a different attenuation and distortion, in which we cannot simply adopt the existing solution that was developed based on the RSS measurement by the smartphone to the smartwatch without understanding the signal behaviors from the smartwatch.

To bridge the gap, this work presents extensive experiments to validate the feasibility of using BLE signals from the smartwatch for proximity detection, before developing classification models based on the machine learning approach to classify the exposure risk of a user. Since the smartwatch should be always worn on the human's hand, the likelihood of NLOS is relatively low. Most of the time, NLOS happens when the signals are blocked by the human body. This can be observed when two users wear the smartwatch on the same hand while standing side by side. However, signal distortion due to body shadowing always has a certain distortion pattern that can be learned if there are sufficient data unveiling this distortion pattern. Lastly, we manipulate the RSS values measured by the smartwatch to construct an ambient signature for each user rather than having to hard-code a user's identity based on an encryption key. Since the RSS values vary spatially as well as temporally, it is almost impossible for the attacker to duplicate the signature. Note that a smartwatch differs from a smartphone in its form factor, processing capability, antenna position, and available memory. Hence, it is hard, if not infeasible, to adopt the existing solution with smartphones onto smartwatches directly without a thorough understanding of the signal behaviors.

III. PROPOSED PROXIMITY-BASED PRIVACY-PRESERVING CONTACT TRACING

Our proposed P³CT leverages the BLE technology available on the smartwatch for proximity sensing. To achieve privacy-preserving contact tracing, we adopt the same signature protocol proposed by our previous work [8] to define the BLE advertising packet. The main framework describing the contact tracing based on BLE technology is shown in Fig. 1. It has the following two phases.

A. INTERACTING PHASE

The interacting phase keeps track of the daily interaction including the interaction distance and interaction duration. A contact tracing application should be able to detect when any two persons are in proximity with each other at the same time keeping track of the duration they remain in close proximity. An effective contact tracing application should be able to detect the proximity with high accuracy rather than seeking to estimate the exact distance, which is quite expansive considering the dynamic movement of humans.

B. TRACING PHASE

When a person is diagnosed with an infectious disease, tracing down a list of people who have been in close contact with the infected patient is of critical importance because these people are more likely to get affected. If this group of people can get informed almost immediately, it reduces the chances for the virus to continue to spread to others. However, many people are concerned about exposing their identity during the tracing phase. Hence, a privacy-preserving contact tracing should provide these two pieces of information without disclosing one's sensitive information.

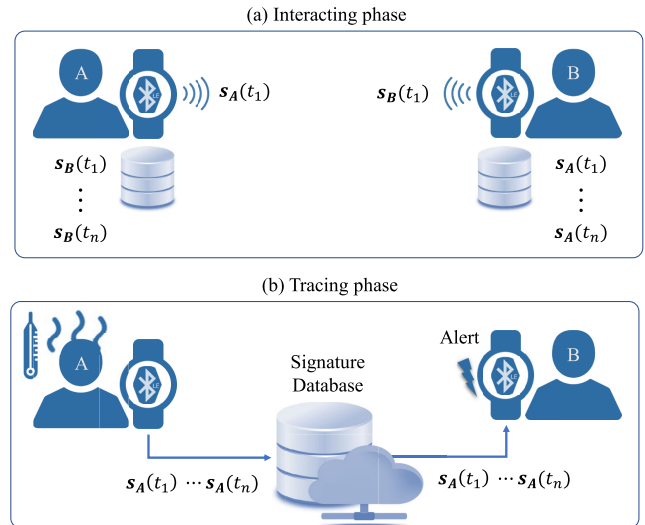


FIGURE 1. When users A and B are in proximity to each other, their smartwatches will log the received BLE packet containing the signature information into their local storage. When user A is diagnosed with an infectious disease, the watch will upload his/her own signatures to the signature database. All the other users can download those signatures and compared them to a list of signatures they have observed in the past 14 days. An alert will be triggered when the downloaded signatures match one of the signatures on the list.

When two users are in proximity to each other, that is, when the smartwatches are within the broadcasting range, they can listen to the incoming packet and measure the RSS. The smartwatches will log the packet including the measured RSS value into its local storage, as shown in Fig. 1(a). The packet contains the ambient signature information observed by the user's smartwatch at a particular timestamp. When a user is diagnosed with an infectious disease, as shown in Fig. 1(b), the smartwatch will upload the user's own signatures generated for the past 14 days to the signature database (the number of days depends on the epidemiological situation and can change dynamically). All the other users will download the infected signatures into their smartwatch for signature matching. In other words, the signature matching process is taken place in the user's smartwatch rather than the cloud server. In this case, there is no way for others to know who has come into close contact with the infected patient. The smartwatch will automatically trigger an alert when it found a matched signature. Based on the alert, the user can take the necessary action, such as self-quarantine and acquire coronavirus testing, to prevent the further spread of this highly contagious disease.

The proposed P³CT has two main parts, the proximity sensing and the signature protocol.

C. PROXIMITY SENSING WITH BLE TECHNOLOGY

Our proposed P³CT exploits the proximity sensing information extracted from the received BLE signals to monitor the interaction between users. As a popular short-range communication over the 2.4 GHz ISM band [29], [30], BLE is readily available in many smart devices including

smartwatches, earbuds, smart thermostats, beacons etc. [31], [32]. BLE communicates through either non-connectable advertising or connectable advertising [33]. The latter advertising mode allows another device to request a secure connection through handshaking. Our proposed P³CT uses the former non-connectable advertising mode, which rejects any incoming connection requests, as the major communication platform for contact tracing. Hence, it is almost impossible for any malicious device to connect to the smartwatch and get access to sensitive information.

Note that our proposed P³CT is a wearable solution based on commercial off-the-shelf smartwatches. Being a low-cost device equipped with essential BLE technology, the smartwatch has become an ideal solution for privacy-preserving contact tracing in workplace environments. The non-connectable advertising mode allows the smartwatch to broadcast a short advertising packet periodically according to the system-defined advertising interval, T_a . Each smartwatch can measure the RSS values upon receiving the advertising packet. RSS is inversely proportional to the square of the distance as according to the inverse square law [34], [35], i.e., $P_r \propto \frac{1}{d^n}$ where P_r indicates the RSS value in the scale of dBm, d is the distance between any two smartwatches, and n is the path loss exponent

While the RSS-distance relationship holds for the signal in the free space, RSS values suffer a great distortion in practical environments owing to the multipath [36] and body shadowing effects [37], [38]. The unexpected distortion causes signal variation even though two smartwatches remain still in the same position. This signal variation can be minimized by applying some signal filtering methods, such as moving average. As shown in Fig. 2, the RSS values at each distance are more distinct and with less variation when a moving average is applied (shown in Fig. 2(a)) as compared to the raw RSS data (shown in Fig. 2(b)). While we can set a cut-off threshold, for example, any value greater than -75 dBm as being in close proximity, such a thresholding approach will result in the high false negative with raw RSS value and high false positive with filtered RSS value. Rather than using a thresholding approach, Section IV presents machine learning methods for high-risk and low-risk classification given the RSS data.

D. PRIVACY-PRESERVING SIGNATURE PROTOCOL

We design a privacy-preserving protocol that encapsulates the BLE packet with an ambient signature packet rather than the user's identity or location-related information. The novelty of the signature protocol is to construct a signature vector that can be fit into the length-constrained advertising packet (i.e., the available payload is only 31 bytes). Specifically, each smartwatch is configured to execute the following functions:

- i. Signature Generation: The smartwatch scans for the ambient environmental features. These features are selectively processed to generate a unique signature that fits into the 31 bytes advertising payload. The signature will be updated every few minutes.

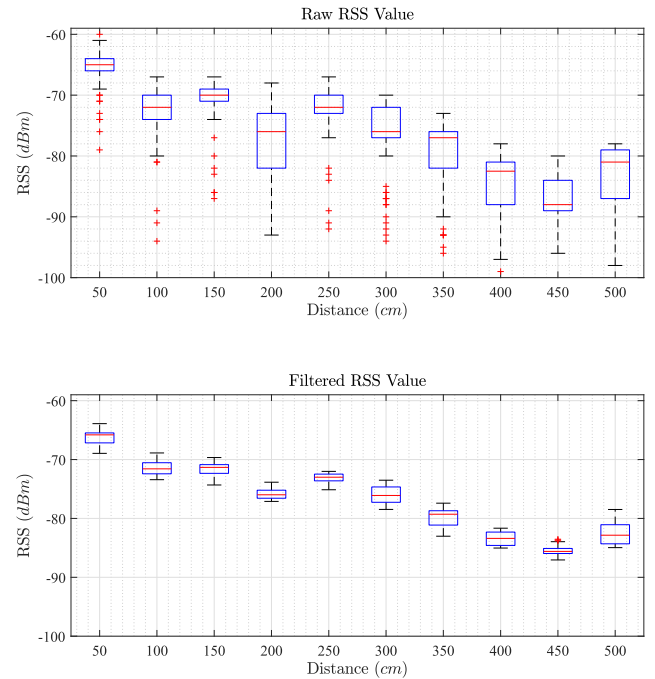


FIGURE 2. Comparing the RSS values at each distance given (a) the raw RSS value, and (b) the filtered RSS value.

- ii. Signature Broadcasting: The smartwatch broadcasts the advertising packet containing the unique signature periodically according to the advertising interval of T_a . The packet is broadcasted through non-connectable advertising channels.
- iii. Signatures Observation: The smartwatch scans the three advertising channels to listen to the advertising packet broadcast by the neighboring smartwatches. The scanning is performed in between the broadcasting event.

The signature is a 31-dimensional transformed vector containing the ambient environmental features. Upon the generation of signature, the smartwatch will encapsulate this signature information into its advertising packet and broadcast the packet through the non-connectable advertising channels. The nearby smartwatches can see the packet when it scans on those advertising channels where the packet is transmitted.

The timing diagram for the advertising, scanning, and signature generation activities, in which each activity is triggered periodically according to their interval, i.e., generation interval T_g , advertising interval T_a , and scanning interval T_s , is shown in Fig. 3. Given T_s , the smartwatch will only stay active to listen for the incoming packet for a duration defined by the scanning window T_w . While it is possible to use continuous scanning (i.e., by setting $T_w = T_s$) to increase the packet receiving rate, such a scanning approach has an adverse effect on energy consumption.

IV. RISK CLASSIFICATION WITH MACHINE LEARNING

Rather than using the measured RSS value for proximity sensing based on the thresholding method, this work leverages

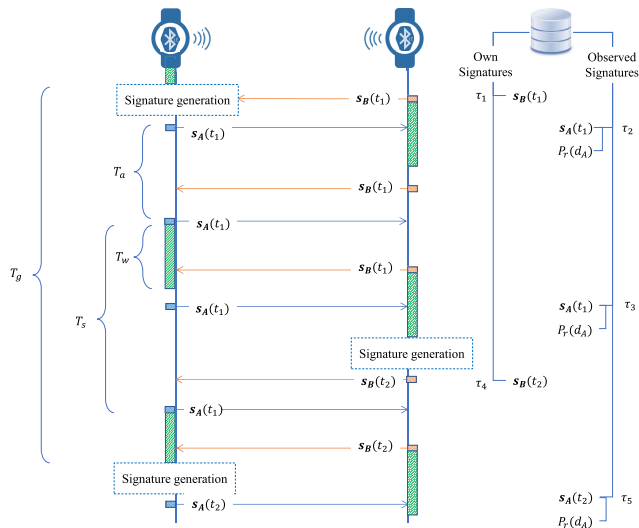


FIGURE 3. The timing diagram for the advertising, scanning, and signature generation activities. All the generated and observed signatures will be logged in the local database, together with a timestamp τ .

machine learning, in particular, supervised learning, to classify the risk of a user with respect to his/her interaction distance and interaction duration with the infected patient. While [39] presents the interaction between a smartphone and a beacon equipped with BLE signals, the work only discusses the interaction distance but did not cover the interaction duration. For a contact tracing application, it is necessary to understand the interaction duration besides the interaction distance. This is because the likelihood for a user to contract a virus does not depend on the interaction distance only, but also on the interaction duration.

This section first discusses the useful features we can obtain from the proximity sensing information, before presenting our hypothesis to risk classification using these features. Next, we describe the four classification models that we will train for our experiments. Our main novelty is on selecting the meaningful features from the proximity sensing for risk classification, whereas designing a classification model is not the main focus of this work. Rather a few general classification models are described to provide an idea on how to adopt our selected features to train a classifier.

A. PROXIMITY SENSING

Proximity sensing has been employed in many scenarios, for example, to identify the user proximity to museum collection [40], to gallery art pieces [41], to other human [42] etc. There are works that study proximity detection in dense environment [43], or proximity accuracy with filtering technique [44]. Most of these works study the proximity detection between a human and an object attached with BLE beacon [45]. In this work, we study the proximity sensing between the devices carried by two human beings. While estimating the distance can help to check if the user participates in a safe physical distancing, an exact distance, such

as 2 meters, should not be a rigid requirement in classifying the risk of a user. Rather, we are more interested to know the proximity between any two users, and how long they remain in proximity.

BLE is the best technology for the above purpose since BLE is a short-range communication that can only be heard when two smartwatches, A and B, are in the communication range of each other. Upon receiving the advertising packet from another smartwatch B, smartwatch A can measure the RSS and thus estimate its proximity to the nearby smartwatch B. We classify the proximity into two classes, i.e., far and close. We define close proximity when the distance between any two smartwatches is less than a predefined threshold, for instance 2 meters, and any distance greater than 2 meters but less than the broadcasting range is considered far. In other words, the two smartwatches are not in proximity if they are outside the broadcasting range of each other.

The RSS distributions for far and close proximity is shown in Fig. 4. It is clear that there will have a lot of errors if we decided the proximity by simply setting an RSS threshold. For example, if we set everything above -80 dBm as close proximity, chances are some values greater than -80 dBm are from the smartwatch located at a distance greater than 2 meters. Hence, it is unreliable to identify the risk of a user simply based on the proximity. At the same time, some users might be in very close proximity when they pass by each other. Hence, we also consider the interaction duration when we want to identify the risk of a user.

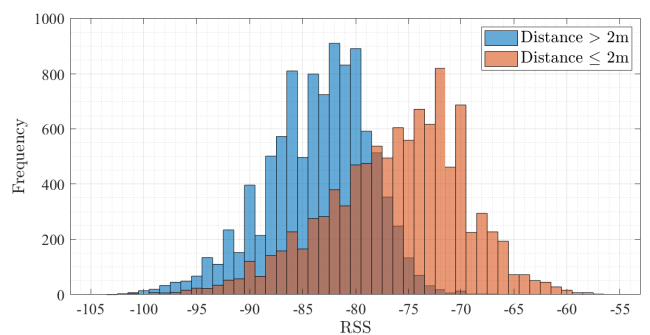


FIGURE 4. RSS distributions for two types of proximity: far (blue color bars) and close (orange color bars).

B. HYPOTHESIS TO RISK CLASSIFICATION

While it is more likely to be infected when the user is in close proximity to the infected patient, the risk of getting infected is relatively low if the user spends less than 1 s in such close proximity. On the other hand, the exposure risk can be high if the user spends a very long time with the infected patient, even if they are keeping a safe physical distance from each other. The possible risk of getting infected with respect to the interaction range and interaction duration between the user and the infected patient is shown in Fig. 5.

The problem of classifying the potential risk of a user can be modeled as a binary hypothesis test. Let \mathbf{x} be an

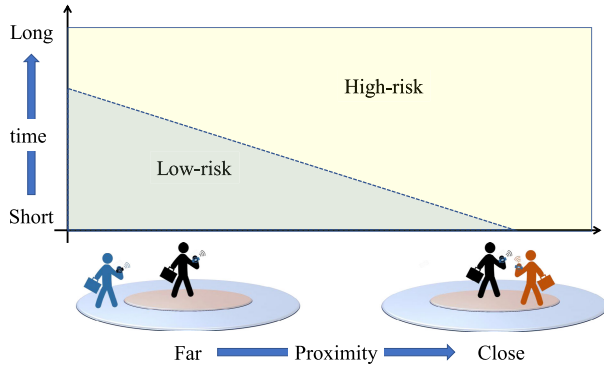


FIGURE 5. The user is considered to have a high risk of being infected when he/she is in close proximity with the infected patient or has spent a very long time with the infected patient within a confined space even if the distance between them is greater than the physical distancing threshold.

m -dimensional feature vector (containing information related to the interaction time and interaction duration) and consider a risk mapping function $R: (\mathbf{x}) \rightarrow \{+1, -1\}$, where $+1$ indicates high-risk and -1 low-risk, then we have the following two hypotheses:

$$\begin{aligned} H_+ : R(\mathbf{x}) &= +1 \\ H_- : R(\mathbf{x}) &= -1 \end{aligned} \quad (1)$$

where H_+ denote the hypothesis that the user belongs to the high-risk ($+1$) group, H_- the hypothesis that the user belongs to the low-risk (-1).

For our problem setting, we only consider the people who received the BLE signals. So, we do not need to consider the null hypothesis H_0 because the null hypothesis only occurs when the user is outside the communication range of the infected patient. Miss detection is undesirable because the user might be at risk but the system considers the user safe. False negative, on the other hand, misclassified the high-risk user to low-risk. This may give a wrong impression to the user that the possibility for them to get infected is low, but in fact, the possibility could be high. While false positive is a bit more conservative by misclassifying the low-risk user to high-risk, it is a relatively safer outcome than miss detection and false negative.

C. CLASSIFICATION MODELS

We apply supervised machine learning methods to train a classification model. The training and testing phases are described in Fig. 6. During the training phase, the data is divided into training and validation set before feeding the data for model learning. The objective is to learn a set of weights that fit the hypothesis function $R(\mathbf{x}, C)$ defined by the corresponding classification model C . 10-fold cross-validation is performed to evaluate the learned model as well as to prevent the model from overfitting. If necessary, model fine-tuning can be performed to improve the classification performance.

Mathematically, the learning process aims to fit the risk mapping function $R: (\mathbf{x}) \rightarrow y$ given a set of n training

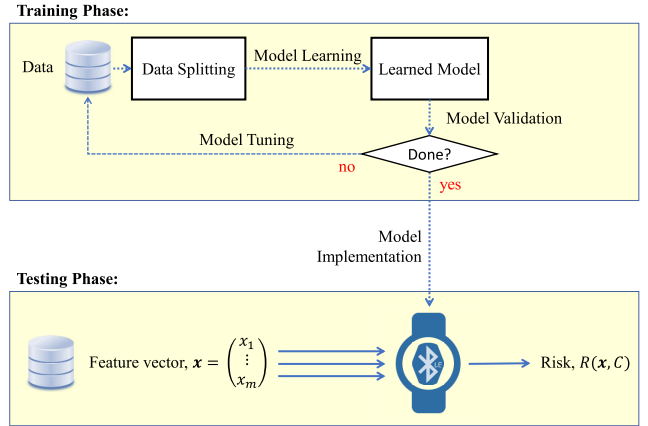


FIGURE 6. A classification model can be learned, validated, and fine-tuned using the collected data. The trained model is loaded to a smartwatch to classify the risk of a user given a set of feature vectors stored in the smartwatch for a number of past days.

samples $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$, where $\mathbf{x} = (x_1, \dots, x_m)^T$ is an m -dimensional feature vector and $y = \{+1, -1\}$ is the classification output indicating the risk of a user.

In this work, we exploit four classification methods: decision tree (DT), linear discriminant analysis (LDA), naive Bayes (NB), and k nearest neighbors (kNN).

1) DECISION TREE (DT)

The top-down approach is a commonly used method to learn a classification tree. More precisely, DT starts by choosing a feature from the feature vector that provides the best splitting in connection to the target risk label and then repeats the same splitting procedures for each separated branch until it reaches a final decision. Let $\theta = (x, \gamma)$ be the splitting rule given feature x and threshold γ , we can split n samples of training data \mathcal{T} into two subsets, i.e.,

$$\begin{aligned} \mathcal{T}_r(\theta) &= (\mathbf{x}, y) | x \leq \gamma \\ \mathcal{T}_l(\theta) &= \mathcal{T} \setminus \mathcal{T}_r(\theta) \end{aligned} \quad (2)$$

where \mathcal{T}_r and \mathcal{T}_l are the resultant subsets representing the data for right and left branches, respectively.

The common measure used to govern the splitting rule is the Gini impurity $G(\cdot)$, which tells how likely the model will produce a misclassification if the model predicts the labels based on the label's distribution from a randomly chosen feature. Mathematically, the Gini impurity can be computed as follows:

$$G(\mathcal{T}, \theta) = \frac{n_l}{n} \mathbb{H}(\mathcal{T}_l(\theta)) + \frac{n_r}{n} \mathbb{H}(\mathcal{T}_r(\theta)) \quad (3)$$

where n_l and n_r are the number of training samples for each subset, and $\mathbb{H}(\cdot)$ is the entropy function, i.e.,

$$\begin{aligned} \mathbb{H}(\mathbf{x}) &= - \sum_{y=\{+1, -1\}} p_y \log(p_y) \\ &= - \sum_{y=\{+1, -1\}} p_y \log(p_y) \end{aligned} \quad (4)$$

and p_y denotes the probability of correct classification. Suppose that $\mathbb{I} = \{1, 0\}$ be the indication function and \tilde{y} be the predicted output, then we have

$$p_y = \frac{1}{n} \sum_{\forall x \in \mathbf{x}} \mathbb{I}(\tilde{y} = y) \quad (5)$$

The objective of DT is to find the parameters that produce the best splitting rule, i.e.,

$$\theta^* = \arg \min G(\mathcal{T}, \theta) \quad (6)$$

2) LINEAR DISCRIMINANT ANALYSIS (LDA)

Assume that the covariance for each class is the same, LDA learns a classifier by fitting a Gaussian density to each class. Let $\mathbb{P}(\mathbf{x}|\tilde{y} = y)$ be the conditional distribution for each class $y = \{+1, -1\}$, by applying Bayes' rule, we obtain:

$$\mathbb{P}(\tilde{y} = y|\mathbf{x}) = \frac{\mathbb{P}(\mathbf{x}|\tilde{y} = y)\mathbb{P}(\tilde{y} = y)}{\sum_{y=\{+1, -1\}} \mathbb{P}(\mathbf{x}|y)\mathbb{P}(y)} \quad (7)$$

Then, the class (i.e., the risk) can be determined by selecting the output with the highest posterior probability.

3) NAIVE BAYES (NB)

Following a naive assumption that each feature is conditional independence, we can apply Bayes' theorem to learn a classification model.

By simplifying $\mathbb{P}(x|y, \forall x \in \mathbf{x})$ to $\mathbb{P}(x|y)$, we have:

$$\mathbb{P}(y|\forall x \in \mathbf{x}) = \frac{\mathbb{P}(y) \prod_{i=1}^m \mathbb{P}(x|y)}{\mathbb{P}(\mathbf{x})}. \quad (8)$$

Since $\mathbb{P}(y|\forall x \in \mathbf{x})$ is proportional to $\mathbb{P}(y) \prod_{i=1}^m \mathbb{P}(x|y)$, then we can use maximum a posteriori (MAP) to estimate the probability for each class $\mathbb{P}(y)$ and the conditional probability for each class given the feature $\mathbb{P}(x|y)$. The output risk can then be predicted based on the following rule:

$$\tilde{y} = \arg \max_{y=\{+1, -1\}} \mathbb{P}(y) \prod_{i=1}^m \mathbb{P}(x|y) \quad (9)$$

4) K NEAREST NEIGHBORS (kNN)

The goal of kNN is to maximize the probability of correct classification. Let p_i indicate the probability that a training sample i is classified correctly, according to the stochastic nearest neighbors rule, we have:

$$p_i = \sum_{j \in \mathcal{T}_i} p_{ij} \quad (10)$$

where \mathcal{T}_i is a subset of data belonging to the same class as the training sample.

Given p_i , the goal of kNN can be defined as follows:

$$\arg \max_{y=\{+1, -1\}} \sum_{i=1}^n p_i \quad (11)$$

These four classifiers can be further extended by assuming different distribution functions. One of the possible future works is to calibrate the classifier based on the prior empirical

distribution knowledge about a certain environment. More precisely, different environments might produce different distributions, and if we can acquire this information, it could help to better calibrate the classifier and thus improve the classification performance.

V. EXPERIMENTS AND EVALUATIONS

Supervised learning, such as classification, requires a set of labeled data, which is not readily available in the context of smartwatches. In contrast to the abundant and open-accessible sources of text-based (e.g., WikiLens, BookCrossing, etc.) or image-based (e.g., MNIST, imageNet, etc.) datasets, there are not many publicly available datasets including the BLE signals received by the smartwatch. We developed an application on the smartwatch to collect the BLE data and store them in a public available dataset [16].

This section first presents the experimental setup with smartwatches and then describes the data we have collected through our smartwatch applications. We consolidated the collected data from both smartwatches before dividing them into training and testing datasets. Lastly, we evaluate the experimental results obtained from different classifiers.

A. EXPERIMENTAL SETUP FOR DATA COLLECTION

For the experiment, we used Fossil Sport, a smartwatch based on Google's Wear OS. The smartwatch is powered by a Qualcomm Snapdragon Wear 3100 processor and has an internal memory of up to 1 GB. The 8 GB internal storage is sufficient to store the generated and observed signatures for at least 14 days. The small form factor (i.e., 1.28in AMOLED screen with 44 mm case size and 12 mm case thickness) makes the smartwatch an ideal candidate for contact tracing in the workplace. As shown in Fig. 7, the smartwatch can trigger the alert automatically when any two smartwatches are in close proximity to each other.

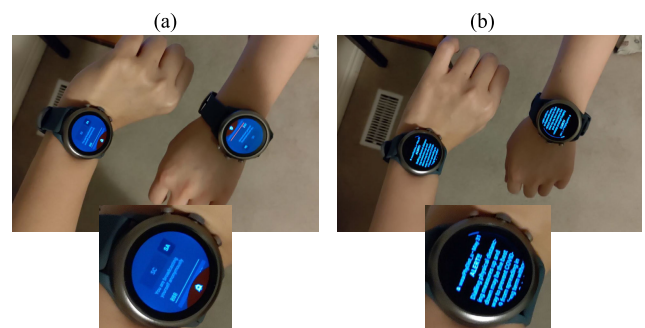


FIGURE 7. When any two persons come close to each other, (a) the smartwatch will vibrate with an alert, and (b) the smartwatch will also trigger an alert notification to remind the users to practice safe physical distancing.

We programmed the smartwatch application to broadcast the advertising packet in the background. For experimental purposes, we also programmed the application to log all the advertising packets it received at every distance. Besides the advertising packet, the smartwatch also logged the following

information: the truth distance, name of the smartwatch, MAC address of BLE chipset, the packet payload, RSS values, time elapsed, and timestamp. The time elapsed indicates the time difference between the previous broadcast packet and the current broadcast packet, whereas the timestamp is the exact time when the smartwatch received the packet.

We performed the experiment by asking two subjects to stand at a certain distance from each other, from 0.5 m up to 5 m. A measuring tape is used as a reference to the ground truth distance. The subjects were asked to wear the smartwatch on different hands and repeat the experiment. Specifically, volunteer A wore the smartwatch on her left hand, and volunteer B on her right hand (i.e., left to right (LR)). After that, the same experiment was repeated with right hand to left hand (RL), left hand to left hand (LL), and right hand to right hand (RR). Since LR and RL constitute a direct view between two smartwatches and LL and RR constitute the crosswise view, we categorize these four hand-combinations into two groups: a) direct, and b) crosswise, as illustrated in Fig. 8.

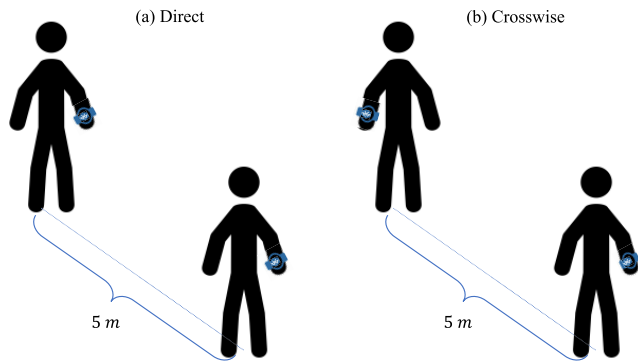


FIGURE 8. Four combinations of smartwatches on hand’s position have been tested, i.e., left hand to right hand (LR), right hand to left hand (RL), left hand to left hand (LL), and right hand to right hand (RR). These four combinations can be classified into two categories: (a) Direct (LR and RL) and (b) Crosswise line (LL and RR).

All the experiments were conducted in indoor environments with a lot of interference from commercial BLE devices, such as tablets, smartphones, earbuds, smart thermostats, etc. Furthermore, the data collection were executed at different times with uncontrolled indoor environmental settings (for example, having people using the microwave, having some people walking around, and also having different furniture arrangements).

The goal is to collect sufficient data capturing the signal distortion subject to the environmental dynamics. Since outdoor environments are less dynamic as compared to indoor environments, the classification models that we have trained should have better, or at least the same performance, as what we have achieved with the indoor setting. All the measurement data is saved into a “comma-separated values” (.csv) file format and exported to Matlab for training and testing.

B. DATA PREPARATION AND PROCESSING

In total, we have collected 37,644 data points from all four combinations, as shown in Table 1. We consolidated the data from RR and LL into a single dataset (i.e., the crosswise dataset). We applied segmentation with 90% overlapping when sampling the data from the 37,644 data points. Such a segmentation results in a total of 17,282 training samples and 4320 testing samples with six training features. These features are computed from the raw RSS data. These six input features include 1) the number of samples observed by the smartwatch, 2) mean RSS, 3) standard deviation RSS, 4) maximum RSS, 5) minimum RSS, and 6) RSS range (i.e., maximum RSS – minimum RSS). The number of samples observed by the smartwatch tells how long the smartwatch are in proximity to each other. To encourage future work, we have included the raw RSS data for people who would like to exploit other features.

TABLE 1. Total data from each combination.

Combination	Total Data Points
RR	8168
LL	7874
RL	13117
LR	8485

Next, an 80%-20% splitting rule was applied to split the data into training and testing sets. Similarly, we applied the same segmentation and splitting rule to the consolidated data from RL and LR (i.e., the direct dataset), resulting in 12,834 training samples and 3208 testing samples. Both direct and crosswise datasets are shared openly in our Github repository, along with example codes to provide a detailed walk-through on reproducing our work [16].

C. EVALUATION METRICS

We used four metrics (i.e., precision (p), recall (r), F1-score (f_1) and accuracy (a)) to evaluate the performance of these classifiers. Let T^+ , T^- , F^+ and F^- denote the true positive, true negative, false positive, and false negative, respectively, then the above four metrics can be computed as follows:

$$p = \frac{T^+}{T^+ + F^+} \tag{12}$$

$$r = \frac{T^+}{T^+ + F^-} \tag{13}$$

$$f_1 = 2 \frac{rp}{r + p} \tag{14}$$

$$a = \frac{T^+ + T^-}{T^+ + T^- + F^+ + F^-} \tag{15}$$

Intuitively, precision tells how many are actually at the high risk out of all the predictions as positive. High precision indicates the capability of a classifier in producing a low false positive, avoiding creating unnecessary tension and anxiety to the people. Recall, on the other hand, tells how many we predicted as high-risk are in fact having high-risk of being

TABLE 2. The performance of each classifier for the direct dataset.

Classifier	Precision		Recall		F1-score		Accuracy	
	Mean	95% CI	Mean	95% CI	Mean	95% CI	Mean	95% CI
DT	0.9670	(0.9653, 0.9686)	0.9330	(0.9305, 0.9353)	0.9497	(0.9483, 0.9511)	0.9416	(0.9400, 0.9431)
LDA	0.8600	(0.8572, 0.8628)	0.8439	(0.8406, 0.8472)	0.8518	(0.8497, 0.8541)	0.8293	(0.8268, 0.8318)
NB	0.9562	(0.9548, 0.9577)	0.9127	(0.9099, 0.9152)	0.9339	(0.9324, 0.9355)	0.9228	(0.9212, 0.9247)
kNN	0.8853	(0.8828, 0.8882)	0.9170	(0.9145, 0.9194)	0.9009	(0.8994, 0.9028)	0.8889	(0.8872, 0.8909)

TABLE 3. The performance of each classifier for the crosswise dataset.

Classifier	Precision		Recall		F1-score		Accuracy	
	Mean	95% CI	Mean	95% CI	Mean	95% CI	Mean	95% CI
DT	0.9176	(0.9140, 0.9208)	0.9133	(0.9104, 0.9157)	0.9154	(0.9133, 0.9175)	0.9059	(0.9037, 0.9083)
LDA	0.7959	(0.7912, 0.8008)	0.8255	(0.8224, 0.8292)	0.8104	(0.8070, 0.8137)	0.7934	(0.7903, 0.7966)
NB	0.9247	(0.9212, 0.9272)	0.8622	(0.8590, 0.8650)	0.8924	(0.8902, 0.8948)	0.8763	(0.8739, 0.8788)
kNN	0.9392	(0.9354, 0.9422)	0.8270	(0.8238, 0.8319)	0.8795	(0.8767, 0.8820)	0.8573	(0.8541, 0.8600)

infected. In contrast to the accuracy that considers the number of correctly classified true positives and true negatives, F1-score considers the balance of precision and recall. The F1-score is a useful metric when false negatives and false positives are important factors in evaluating the classifier performance.

D. EXPERIMENTAL RESULTS AND DISCUSSIONS

We fed the two datasets, i.e., direct and crosswise datasets, to the four different classifiers (i.e., DT, LDA, NB, and kNN) for training. We repeated the experiment 100 times with a different set of testing data. Specifically, we randomly sampled 20% of data from the dataset for testing purposes at every iteration. For each evaluation metric, we show the mean result and its corresponding 95% confidence interval (CI). The overall mean results and 95% CI for both direct and crosswise datasets are shown in Table 2 and Table 3, respectively. An illustration of the F1-score distribution obtained from DT with the 100 testing sets, is shown in Fig. 9. From both tables, we can see that all the classifiers achieve satisfactory performance with high precision and recall. In other words, the classifier did not penalize the recall in order to achieve

high precision. Hence, the F1-scores for both datasets are high.

We also observed that the direct dataset gave a better performance than the crosswise dataset. This can be explained by the possible signal attenuation when the two hands are blocked by the human body. Among all the classifiers, DT achieves the best performance with the highest precision, recall, F1-score, and accuracy. The precision-recall curve for both (a) direct and (b) crosswise, is shown in Fig. 10. The precision-recall curve provides further insight into the trade-off between precision and recall. Both plots indicate that DT achieves superior performance with high precision and recall, whereas other methods tend to trade-off the recall in order to achieve high precision.

1) IMPLICATION OF INPUT FEATURES

Previously, we used all the five input features (i.e., number of samples observed by the smartwatch, mean RSS, maximum RSS, minimum RSS, and RSS range) to train the model. All the four trained classifiers were able to produce satisfactory classification performance, i.e., at least 85% accuracy. Hence, we would like to investigate the implication of input features on classification performance.

We repeated the experiment by using only one feature (i.e., mean RSS), and then two features (i.e., mean RSS and the number of samples), and so on. The classification accuracy achieved by all the four classifiers is shown in Fig. 11. From both bar charts, we can see that kNN suffers severe performance degradation when only one input feature is available. Overall, the performance increases when the number of features increases.

The performance gain of each classifier with respect to the number of features increases, is shown in Fig. 12. Clearly, kNN is benefited when there are more input features. Both LDA and NB did not show improvement after two features. Their performance saturates when the number of features is more than two. The performance of DT also increases when

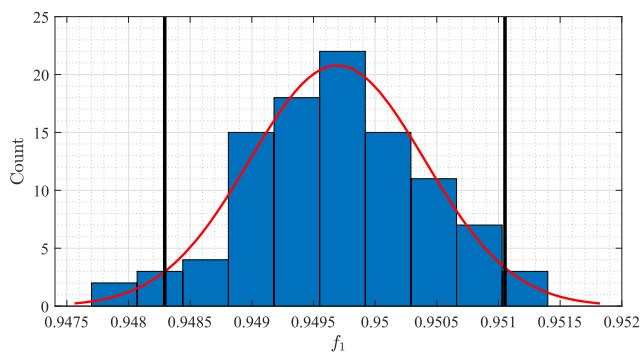


FIGURE 9. The histogram illustrates the F1-score distribution obtained from DT with the 100 different testing sets. The area in between the black lines indicates the 95% confidence interval.

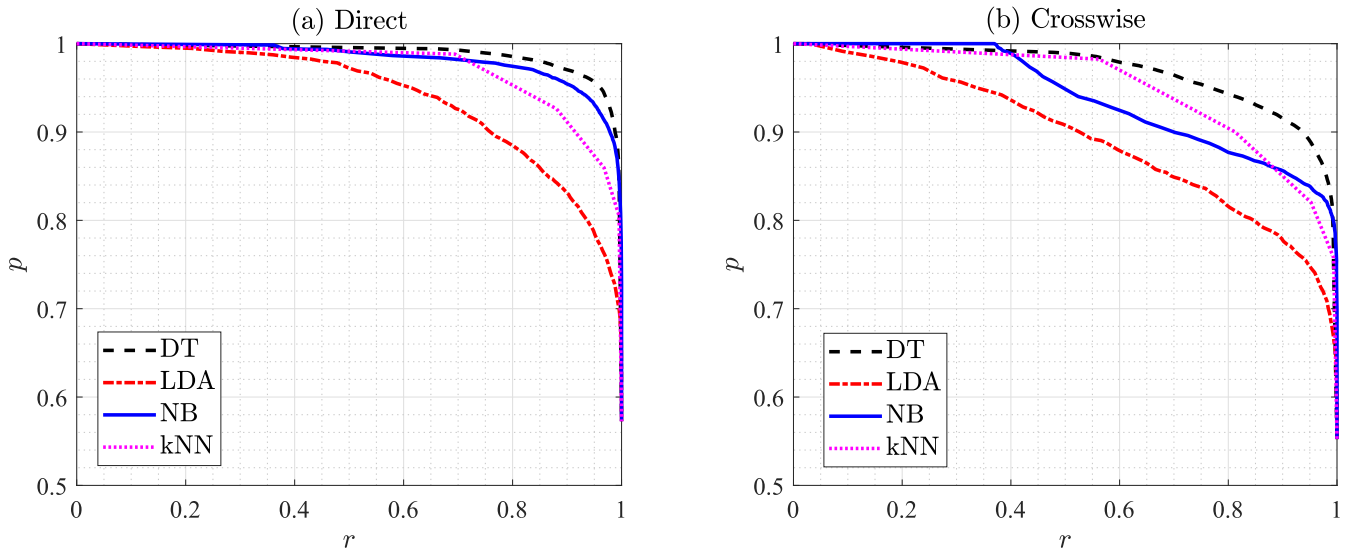


FIGURE 10. Precision-recall curves for: (a) direct dataset, (b) crosswise dataset.

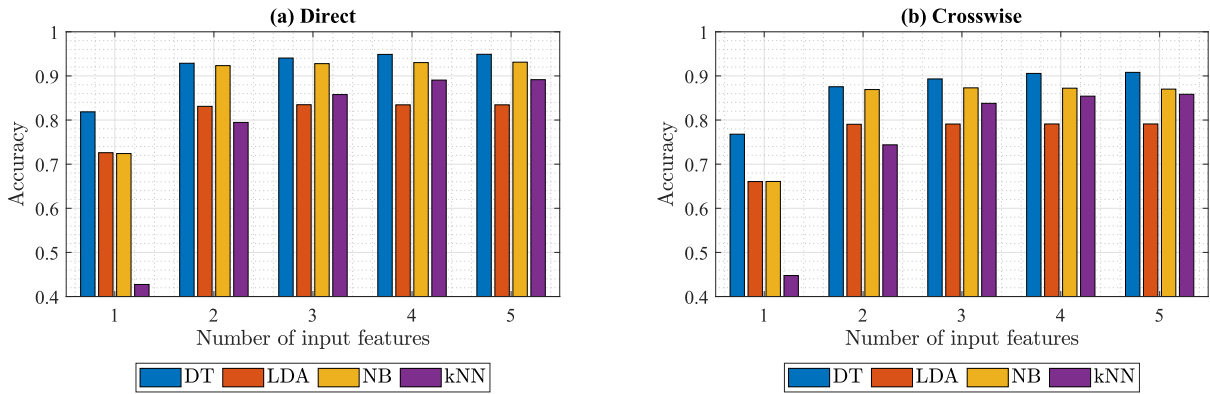


FIGURE 11. The effect of number of input features on the accuracy.

the number of features increases, even though the performance gain is quite minimal.

In summary, some features are indeed useful in training a good model, while some features might be redundant and can be excluded from training. For example, the maximum RSS and minimum RSS might not provide good information to the model training, whereas the RSS range provides more useful information. The RSS range provides an indication of how big the RSS fluctuated during a particular observation period, and this piece of information is indeed helpful to model learning.

2) IMPLICATION OF NUMBER OF SAMPLES

As discussed, the number of samples observed by the smartwatch is a good indication of how long the user has been interacting with each other. We can make a better inference when the number of samples observed by the smartwatch increases.

The effect of the number of samples on the classification accuracy is illustrated in Fig. 13. The accuracy increases when the number of samples increases and then slowly saturates after it obtains a sufficient number of samples. The increase in the number of samples has less effect on accuracy when the system has obtained a sufficient number of samples to make an inference. The results show that the accuracy starts to saturate when the number of samples reaches 100, for both (a) direct and (b) crosswise cases. Hence, we can conclude that most classifiers can produce decent classification output when there are at least 100 samples. If the smartwatch is configured to advertise the packet every 100 ms, we should expect approximately 10 samples per second, which means approximately 10 s is required for each classifier to reach a stable performance.

In practice, this is a reasonable duration considering the interaction duration between users. If the interaction duration is less than 10 s, the risk of getting infected is very low even though the user is very close to the infected patient.

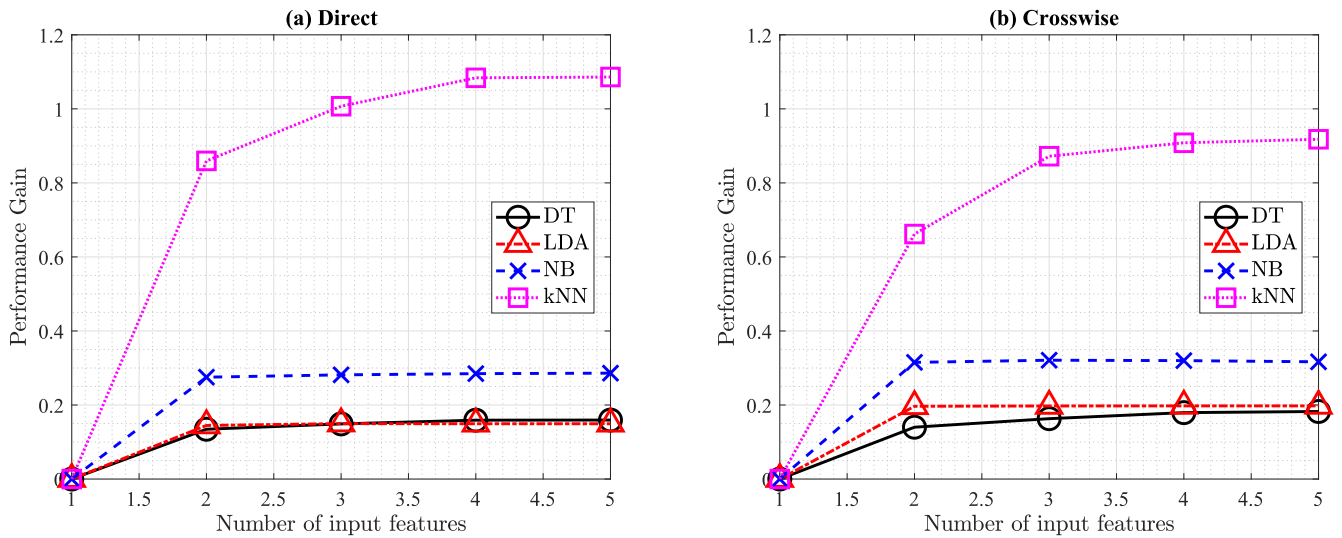


FIGURE 12. The performance gain achieved by each classifier.

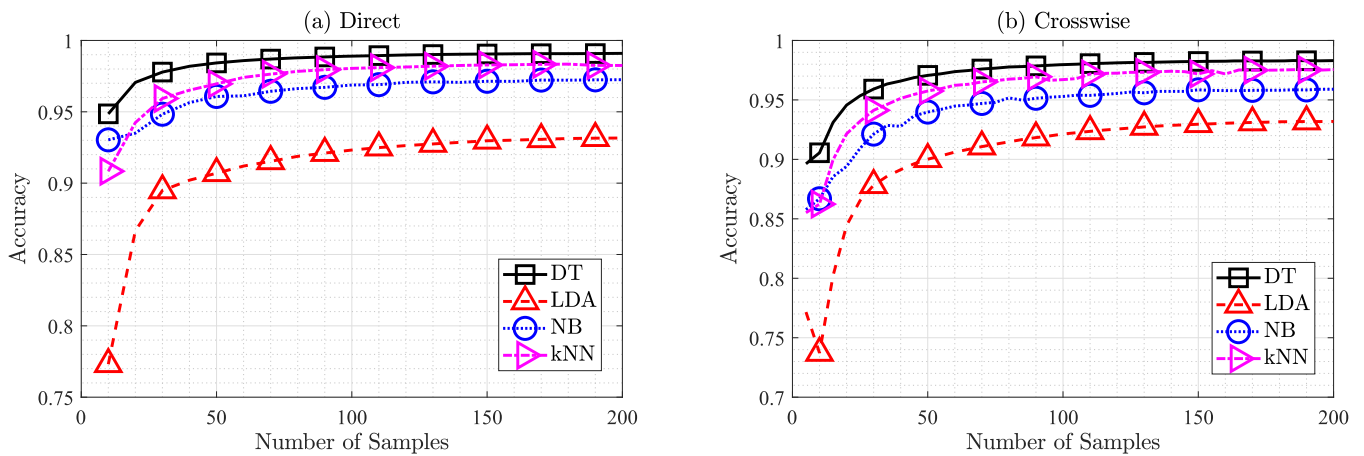


FIGURE 13. The effect of number of input samples on the accuracy.

3) IMPLICATION OF PHYSICAL DISTANCING RULE

While Canada imposed a 2 meters physical distancing rule [46], different countries might have different sets of physical distancing measures. For example, Italy requires its citizens to practice 1 meter physical distancing [47]. Considering the physical distancing differences from country to country, we conducted an experiment to verify our classification approach with different physical distancing thresholds. The classification accuracy with different physical distancing thresholds, is shown in Fig. 14. The results prove the robustness of our classification approach, in which each classifier achieves almost similar accuracy despite the differences in the physical distancing threshold. This means that our proposed approach is practical and can be applied by any country directly by simply updating the physical distancing threshold in correspondence to the set of preventive measures defined by the government.

E. COMPARISON TO SHALLOW NEURAL NETWORK MODEL

To further examine the classification performance, we build a shallow 4-layers feedforward neural network (FNN) model. The number of hidden neurons for each layer is 8, 16, 24, and 36, as shown in Fig. 15. We compared the FNN model to the best classifier we have achieved (i.e., DT) previously. The classification performance of these two methods is summarized in Table 4. The example code for constructing the FNN model is also provided in our Github repository.

While FNN achieves a better performance than DT, the FNN model is more complex in terms of computation than DT which makes the direct implementation on the low-cost and low-end smartwatch almost impossible. Since smartwatch-based contact tracing is targeting industry sectors with hundreds to thousands of employees, a low-end smartwatch is a cost-effective solution.

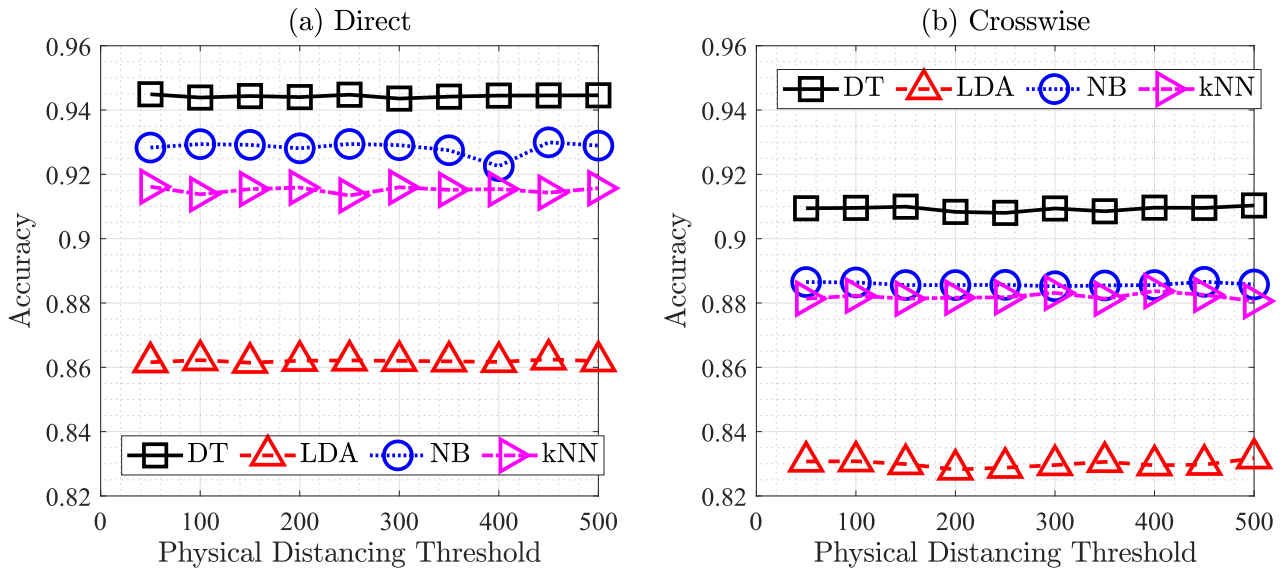


FIGURE 14. The effect of physical distancing threshold on the accuracy.

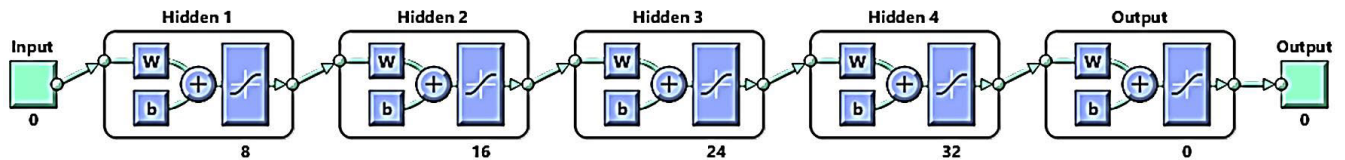


FIGURE 15. The feedforward neural network used in our experiment. The diagram is generated by Matlab Deep Learning Toolbox after defining the network architecture.

TABLE 4. Classification performance comparing DT to Feedforward NN.

Metrics	DT	FNN
Precision	0.9067	0.9534
Recall	0.9330	0.9303
F1-score	0.9497	0.9417
Accuracy	0.9416	0.9331

VI. CONCLUSION

Contact tracing is deemed to be an essential measure in the post-pandemic to prevent and alleviate a future outbreak while slowly reopening the workplace. Even though smartphone-based contact tracing is cost-effective considering the ubiquity of smartphones, it is not convenient to have the user carry with them the smartphone all the time during working. On the other hand, a smart wearable approach provides a more practical solution to contact tracing in the workplace. In this work, we verify the practicality of our proposed P³CT with real-world BLE data collected from the smartwatch. We examine the performance of our approach while using supervised learning and four classifiers. According to experimental results, DT achieves the best performance with the highest precision, recall, F1-score, and accuracy. For future work, we can integrate the embedded sensors within

the watch to monitor users’ activity and thus to better predict their interaction behaviors. The additional knowledge of interaction behaviors, besides the interaction proximity and duration, provide further information to estimate the risk of being infected. Besides that, future work can also consider reframing the risk classification problem as a risk regression problem so that better insights can be obtained regarding the correlation between risk and RSS values measured in indoor and outdoor environments.

REFERENCES

- [1] A. Sharif, C. Aloui, and L. Yarovaya, “COVID-19 pandemic, oil prices, stock market, geopolitical risk and policy uncertainty Nexus in the U.S. economy: Fresh evidence from the wavelet-based approach,” *Int. Rev. Financial Anal.*, vol. 70, Jul. 2020, Art. no. 101496.
- [2] P. C. Ng, P. Spachos, S. Gregori, and K. N. Plataniotis, “Personal devices for contact tracing: Smartphones and wearables to fight COVID-19,” *IEEE Commun. Mag.*, vol. 59, no. 9, pp. 24–29, Sep. 2021.
- [3] L. Ferretti, C. Wymant, M. Kendall, L. Zhao, A. Nurtay, L. Abeler-Dörner, M. Parker, D. Bonsall, and C. Fraser, “Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing,” *Science*, vol. 368, no. 6491, May 2020.
- [4] M. J. M. Chowdhury, M. S. Ferdous, K. Biswas, N. Chowdhury, and V. Muthukkumarasamy, “COVID-19 contact tracing: Challenges and future directions,” *IEEE Access*, vol. 8, pp. 225703–225729, 2020.
- [5] (2020). *Pan-European Privacy-Preserving Proximity Tracing*. [Online]. Available: <https://www.pepp-pt.org/>
- [6] *We Put the Power to Reduce the Spread of COVID-19 in the Palm of Your Hand*. Accessed: Feb. 2, 2022. [Online]. Available: <https://www.covid-watch.org/>

- [7] PACT: Private Automated Contact Tracing. Accessed: Feb. 2, 2022. [Online]. Available: <https://pact.mit.edu/>
- [8] P. C. Ng, P. Spachos, and K. N. Plataniotis, "COVID-19 and your smartphone: BLE-based smart contact tracing," *IEEE Syst. J.*, vol. 15, no. 4, pp. 5367–5378, Dec. 2021.
- [9] S. Latif, M. Usman, S. Manzoor, W. Iqbal, J. Qadir, G. Tyson, I. Castro, A. Razi, M. N. K. Boulos, A. Weller, and J. Crowcroft, "Leveraging data science to combat COVID-19: A comprehensive review," *IEEE Trans. Artif. Intell.*, vol. 1, no. 1, pp. 85–103, Aug. 2020.
- [10] Y. Li, D. Wei, J. Chen, S. Cao, H. Zhou, Y. Zhu, J. Wu, L. Lan, W. Sun, T. Qian, K. Ma, H. Xu, and Y. Zheng, "Efficient and effective training of COVID-19 classification networks with self-supervised dual-track learning to rank," *IEEE J. Biomed. Health Informat.*, vol. 24, no. 10, pp. 2787–2797, Oct. 2020.
- [11] S. Heidarian, P. Afshar, A. Mohammadi, M. Javad Rafiee, A. Oikonomou, K. N. Plataniotis, and F. Naderkhani, "CT-CAPS: Feature extraction-based automated framework for COVID-19 disease identification from chest CT scans using capsule networks," 2020, *arXiv:2010.16043*.
- [12] X. Wang, X. Deng, Q. Fu, Q. Zhou, J. Feng, H. Ma, W. Liu, and C. Zheng, "A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT," *IEEE Trans. Med. Imag.*, vol. 39, no. 8, pp. 2615–2625, May 2020.
- [13] V. Z. Marmarelis, "Predictive modeling of COVID-19 data in the U.S.: Adaptive phase-space approach," *IEEE Open J. Eng. Med. Biol.*, vol. 1, pp. 207–213, 2020.
- [14] Y. Peng, Y. Tang, S. Lee, Y. Zhu, R. M. Summers, and Z. Lu, "COVID-19-CT-CXR: A freely accessible and weakly labeled chest X-ray and CT image collection on COVID-19 from biomedical literature," *IEEE Trans. Big Data*, vol. 7, no. 1, pp. 3–12, Mar. 2021.
- [15] Dr. Anthony Fauci Says Staying Closed for Too Long Could Cause 'Irreparable Damage'. Accessed: Feb. 2, 2022. [Online]. Available: <https://www.cnn.com/2020/05/22/dr-anthony-fauci-says-staying-closed-for-too-long-could-cause-irreparable-damage.html>
- [16] RSS Smartwatch. Accessed: Feb. 2, 2022. [Online]. Available: <https://github.com/pspachos/RSSI-Smartwatch>
- [17] N. Ahmed, R. A. Michelin, W. Xue, S. Ruj, R. Malaney, S. S. Kanhere, A. Seneviratne, W. Hu, H. Janicke, and S. K. Jha, "A survey of COVID-19 contact tracing apps," *IEEE Access*, vol. 8, pp. 134577–134601, 2020.
- [18] Coronavirus Mobile Apps are Surging in Popularity in South Korea. Accessed: Feb. 2, 2022. [Online]. Available: <https://www.cnn.com/2020/02/28/tech/korea-coronavirus-tracking-apps/index.html>
- [19] M. T. Rahman, R. T. Khan, M. R. A. Khandaker, M. Sellathurai, and M. S. A. Salan, "An automated contact tracing approach for controlling COVID-19 spread based on geolocation data from mobile cellular networks," *IEEE Access*, vol. 8, pp. 213554–213565, 2020.
- [20] K. A. Nguyen, Z. Luo, and C. Watkins, "Epidemic contact tracing with smartphone sensors," *J. Location Based Services*, vol. 14, no. 2, pp. 92–128, Apr. 2020.
- [21] T. Mishra et al., "Pre-symptomatic detection of COVID-19 from smartwatch data," *Nature Biomed. Eng.*, vol. 4, pp. 1–13, Nov. 2020.
- [22] G. Quer, J. M. Radin, M. Gadaleta, K. Baca-Motes, L. Ariniello, E. Ramos, V. Khetarpal, E. J. Topol, and S. R. Steinhubl, "Wearable sensor data and self-reported symptoms for COVID-19 detection," *Nature Med.*, vol. 27, pp. 73–77, Jan. 2020.
- [23] O. Amft, L. I. Lopera Gonzalez, P. Lukowicz, S. Bian, P. Burggraf, O. Amft, and K. V. Laerhoven, "Wearables to fight COVID-19: From symptom tracking to contact tracing," *IEEE Pervasive Comput.*, vol. 19, no. 4, pp. 53–60, Oct. 2020.
- [24] A. K. Tripathy, A. G. Mohapatra, S. P. Mohanty, E. Kougianos, A. M. Joshi, and G. Das, "EasyBand: A wearable for safety-aware mobility during pandemic outbreak," *IEEE Consum. Electron. Mag.*, vol. 9, no. 5, pp. 57–61, Sep. 2020.
- [25] China Launches Coronavirus 'Close Contact detector' App. [Online]. Available: <https://www.bbc.com/news/technology-51439401>
- [26] Tracetgether, Safer Together. [Online]. Available: <https://www.tracetgether.gov.sg/>
- [27] M. Shukla, R. S. Lodha, G. Shroff, and R. Raskar, "Privacy guidelines for contact tracing applications," 2020, *arXiv:2004.13328*.
- [28] J. Bell, D. Butler, C. Hicks, and J. Crowcroft, "TraceSecure: Towards privacy preserving contact tracing," 2020, *arXiv:2004.04059*.
- [29] R. Want, B. Schilit, and D. Laskowski, "Bluetooth LE finds its niche," *IEEE Pervasive Comput.*, vol. 12, no. 4, pp. 12–16, Oct. 2013.
- [30] K.-H. Chang, "Bluetooth: A viable solution for IoT? [industry perspectives]," *IEEE Wireless Commun.*, vol. 21, no. 6, pp. 6–7, Dec. 2014.
- [31] K. E. Jeon, J. She, P. Soonsawad, and P. C. Ng, "BLE beacons for Internet of Things applications: Survey, challenges, and opportunities," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 811–828, Apr. 2018.
- [32] M. Radhakrishnan, A. Misra, R. K. Balan, and Y. Lee, "Smartphones and BLE services: Empirical insights," in *Proc. IEEE 12th Int. Conf. Mobile Ad Hoc Sensor Syst.*, Oct. 2015, pp. 226–234.
- [33] S. R. Hussain, S. Mehnaz, S. Nirjon, and E. Bertino, "Secure seamless Bluetooth low energy connection migration for unmodified IoT devices," *IEEE Trans. Mobile Comput.*, vol. 17, no. 4, pp. 927–944, Apr. 2018.
- [34] R. Sari and H. Zayyani, "RSS localization using unknown statistical path loss exponent model," *IEEE Commun. Lett.*, vol. 22, no. 9, pp. 1830–1833, Sep. 2018.
- [35] C. H. Lam, P. C. Ng, and J. She, "Improved distance estimation with BLE beacon using Kalman filter and SVM," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [36] M. Leigsnering, F. Ahmad, M. G. Amin, and A. M. Zoubir, "Compressive sensing-based multipath exploitation for stationary and moving indoor target localization," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 8, pp. 1469–1483, Dec. 2015.
- [37] M. Ayadi and A. B. Zineb, "Body shadowing and furniture effects for accuracy improvement of indoor wave propagation models," *IEEE Trans. Wireless Commun.*, vol. 13, no. 11, pp. 5999–6006, Nov. 2014.
- [38] Q. Tian, K. I.-K. Wang, and Z. Salcic, "Human body shadowing effect on UWB-based ranging system for pedestrian tracking," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 10, pp. 4028–4037, Oct. 2019.
- [39] P. C. Ng, J. She, and R. Ran, "A reliable smart interaction with physical thing attached with BLE beacon," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3650–3662, Apr. 2020.
- [40] P. Spachos and K. N. Plataniotis, "BLE beacons for indoor positioning at an interactive IoT-based smart museum," *IEEE Syst. J.*, vol. 14, no. 3, pp. 3483–3493, Sep. 2020.
- [41] P. C. Ng, J. She, and S. Park, "Notify-and-interact: A beacon-smartphone interaction for user engagement in galleries," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2017, pp. 1069–1074.
- [42] A. Montanari, S. Nawaz, C. Mascolo, and K. Sailer, "A study of Bluetooth low energy performance for human proximity detection in the workplace," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2017, pp. 90–99.
- [43] P. C. Ng, J. She, and S. Park, "High resolution beacon-based proximity detection for dense deployment," *IEEE Trans. Mobile Comput.*, vol. 17, no. 6, pp. 1369–1382, Jun. 2018.
- [44] A. Mackey, P. Spachos, L. Song, and K. N. Plataniotis, "Improving BLE beacon proximity estimation accuracy through Bayesian filtering," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3160–3169, Apr. 2020.
- [45] P. C. Ng, J. She, and R. Ran, "A compressive sensing approach to detect the proximity between smartphones and BLE beacons," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7162–7174, Aug. 2019.
- [46] Physical Distancing: How to Slow the Spread of COVID-19. Accessed: Feb. 2, 2022. [Online]. Available: <https://www.canada.ca/en/public-health/services/publications/diseases-conditions/social-distancing.html>
- [47] Coronavirus: Could Social Distancing of Less Than Two Metres Work?. Accessed: Feb. 2, 2022. [Online]. Available: <https://www.bbc.com/news/science-environment-52522460>



PAI CHET NG (Member, IEEE) received the Ph.D. degree in electronic and computer engineering from The Hong Kong University of Science and Technology. She is currently a Postdoctoral Researcher with the University of Toronto, Canada. Her research interests include proximity-based sensing and networking, physiological signals with mobile and wearable devices, and IoT systems with artificial intelligence.



PETROS SPACHOS (Senior Member, IEEE) received the Diploma degree in electronic and computer engineering from the Technical University of Crete, Chania, Greece, and the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Toronto, ON, Canada. He is currently an Associate Professor with the School of Engineering, University of Guelph, ON, Canada. He is also a Registered Professional Engineer in Ontario. His research interests include experimental wireless networking and mobile computing with a focus on wireless sensor networks, smart cities, and the Internet of Things.



KONSTANTINOS N. PLATANIOTIS (Fellow, IEEE) is currently a Professor and the Bell Canada Chair of Multimedia with the University of Toronto. His research interests include the areas of machine learning and signal processing, and their applications in imaging systems, communications, and knowledge media design systems. He is a fellow of the Engineering Institute of Canada, a fellow of The Canadian Academy of Engineering/L'Académie Canadienne Du Genie, and a Registered Professional Engineer in Ontario. He is the General Co-Chair of the 2027 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2027).

• • •



STEFANO GREGORI (Senior Member, IEEE) received the Laurea and Ph.D. degrees in electrical and computer engineering from the University of Pavia, Pavia, Italy. He was with The University of Texas at Dallas, Dallas, TX, USA, from 2002 to 2004. He is currently a Professor in computer engineering with the University of Guelph, ON, Canada. He is also a Registered Professional Engineer in Ontario. His research interests include the design, analysis, and characterization of integrated power converters, energy harvesters, and integrated circuits with analog and digital applications. He served on the organizing committees for the IEEE Canadian Conference on Electrical and Computer Engineering and the IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks.