Non-intrusive Anomaly Detection of Industrial Robot Operations by Exploiting Nonlinear Effect

ZHIQING LUO, Huazhong University of Science and Technology, China
MINGXUAN YAN, Huazhong University of Science and Technology, China
WEI WANG*, Huazhong University of Science and Technology, China
QIAN ZHANG, Hong Kong University of Science and Technology, China

With the development of Internet of Robotic Things concept, low-cost radio technologies open up many opportunities to facilitate the monitoring system of industrial robots, while the openness of wireless medium exposes robots to replay and man-in-the-middle attackers, who send pre-recorded movement data to mislead the system. Recent advances advocate the use of high-resolution sensors to monitor robot operations, which however require invasive retrofit to the robots. To overcome this predicament, we present RobotScatter, a non-intrusive system that exploits the nonlinear effect of RF circuits to fuse the propagation of backscatter tags attached to the robot to defend against active attacks. Specifically, the backscatter propagation interacted by the tags significantly depends on various movement operations, which can be captured with the nonlinearity at the receiver to uniquely determine its identity and the spatial movement trajectory. RobotScatter then profiles the robot movements to verify whether the received movement information matches the backscatter signatures, and thus detects the threat. We implement RobotScatter on two common robotic platforms, Universal Robot and iRobot Create, with over 1,500 operation cycles. The experiment results show that RobotScatter detects up to 94% of anomalies against small movement deviations of 10mm/s in velocity, and 2.6cm in distance.

CCS Concepts: • **Security and privacy → Security services**; **Mobile and wireless security**; • **Human-centered computing → Ubiquitous and mobile computing**.

Additional Key Words and Phrases: industrial robot, physical layer security, nonlinear effect, backscatter

ACM Reference Format:
Zhiqing Luo, Mingxuan Yan, Wei Wang, and Qian Zhang. 2022. Non-intrusive Anomaly Detection of Industrial Robot Operations by Exploiting Nonlinear Effect. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 175 (December 2022), 27 pages. https://doi.org/0000001.0000001

1 INTRODUCTION

With the development of Industry 4.0 and Internet of Robotic Things concept, smart and wireless devices have been widely deployed in the industrial robot area [32, 45], and the progress in information technology and intelligence further makes robotic tasks ubiquitous in the factory and everyday life, such as manufacturing, assembling, cleaning and other domestic tasks [25, 48]. Due to the advantages of low-cost and anti-interference,

*This is the corresponding author

Authors' addresses: Zhiqing Luo, zhiqing_luo@hust.edu.cn, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan, Hubei, China; Mingxuan Yan, mingxuanyan@hust.edu.cn, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan, Hubei, China; Wei Wang, weiwangw@hust.edu.cn, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan, Hubei, China; Qian Zhang, qianzh@cse.ust.hk, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, Hong Kong, China.

2474-9567/2022/12-ART175 $15.00
https://doi.org/0000001.0000001

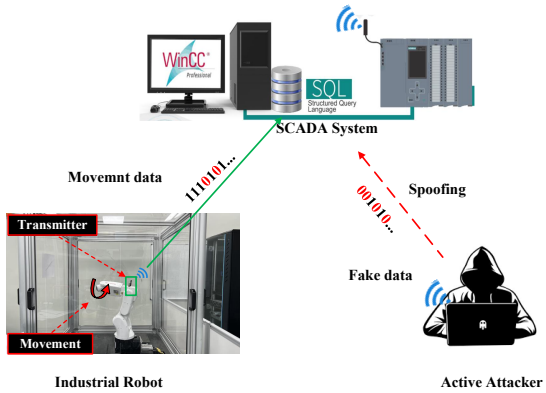


Fig. 1. An illustration of active attack in industrial robot SCADA system.

LoRa communication has been considered a promising technology in the industrial supervisory control and data acquisition (SCADA) system [18, 30]. As shown in Fig.1, when a robot is cyclically operating a manufacturing or assembling task with a pre-programmed procedure, it records the operation information and then sends it to the gateway using LoRa signals. As a result, the SCADA system analyzes the operation data and detects potential anomalies to protect the industrial robot.

However, industrial robots are reportedly suffering from serious active attacks due to cyber vulnerabilities, and cyber attacks remain a major risk to the security system and a hindrance to the market growth [13, 36]. Recent attacks, such as Stuxnet [14, 35], attack the robot by invading industrial control systems and modifying normal movement data to spoof monitoring systems. As illustrated in Fig.1, an active attacker performs a man-in-the-middle attack or a replay attack to impersonate the legitimate robot and inject the fake movement data into the SCADA system [7, 23, 24, 40], resulting in the system failing to capture the real operation state, and thus decreasing the production efficiency. Additionally, because of the vulnerability of LoRa, an attacker can also forge the certificate of the transmitter to spoof the system [6]. Moreover, if the data collection system is malfunctioning and loses the abnormal movement data, the robot would fail to detect the robot state. Therefore, a reliable scheme would significantly help the operators to better understand and monitor the operation states of the robot, and improve the working efficiency and interaction between users and robots.

Building a robust and effective scheme to detect anomalies and secure the wireless monitoring system for industrial robots needs to meet the requirements along three fronts:

- **Integrity.** The monitoring system should guarantee that the movement data received is truly from the target robot and the data has not been modified.
- **Adaptability.** Since the robot will operate in various environments, the security scheme is required to be well-adaptive in different environments.
- **Noninvasiveness.** To enable wide deployment in different robot monitoring systems, it is expected that the security scheme will not require any intrusive modifications.

Unfortunately, no system exists today can achieve all three of these goals simultaneously. Recently, learning the joint angles and torques to detect anomalies has been widely explored, which however cannot guarantee the integrity of the movement data received [20]. To overcome this dilemma, high-resolution cameras have been deployed to monitor movements in the SCADA system [33]. Nevertheless, the visual details are easily shielded by environmental obstacles or the robot itself, and also suffer from strong illumination in real industrial scenarios. Alternatively, installing high-precision gyroscopes to match motion has also been employed to protect industrial

robots [4, 8], while these approaches require intrusive modifications to the robot, resulting in the destruction of the robot structure. RF-based localization and tracking, such as WiFi [26, 43] and mmWave [3, 57], have also gained much attention in recent years, which however are less accurate for fine-grained operation monitoring and fail to be well-adaptive to the robot's mobility when the robot needs to move and complete tasks at the same time. Therefore, a reliable, adaptive, and non-invasive scheme is urgently needed for current wireless industrial robot monitoring systems.

Towards this end, we present RobotScatter, a non-intrusive system that intentionally attaches several tiny and low-cost backscatter tags to the robot's arm to defend again active attacks, such as man-in-the-middle attacks and replay attacks. The key insight of RobotScatter is that the propagation signatures of the tags significantly rely on the movement of the robot's arm, which then are captured and fused by the nonlinear effect of RF circuits at the receiver, and thus can be extracted to detect the robot movement. Specifically, the propagation signatures reflected by the tags are extremely affected by the tag circuit, antenna polarization, as well as propagation distances during the robot's operation of cruise or rotation. As a result, all these incoming backscatter features are fused into harmonic combinations by the nonlinear effect at the receiver. Then, sensitive propagation features can be extracted from the harmonic combinations to characterize the motions, which highly depend on the tag reflection, spatial trajectory, and the receiver's nonlinear characteristics. Finally, we construct the profile from the features to uniquely determine the robot identity and spatial trajectory information, which accordingly help to detect anomalies and defend against active attacks.

However, to realize the above idea, we need to tackle the following challenges.

(1) How to design the backscatter signal and construct sensitive propagation profiles to secure industrial robots? The foundation of RobotScatter is to extract propagation features from the fused propagation caused by the nonlinear effect of the RF circuit. As a result, we will receive many frequency combinations with various harmonics, which prevent the receiver from capturing the desired signals in the current band. To handle this predicament, we observe that there are three commonly used frequencies in the LoRa band, including 433MHz, 868MHz, and 915MHz. Inspired by this, we first set the transmitter to launch the signal at 433MHz and tune the backscatter tags to shift the signal at 434MHz to avoid interference on the emitter. Then, we focus on the second-order harmonic which is the combination of any two tags, and we can capture the backscatter signals at 868MHz which is still located at the LoRa band. Consequently, we extract unique amplitude and phase signatures from the second-order harmonic to construct robot movement profiles to defend against attacks.

(2) How to extract backscatter features to construct motion profiles even though backscatter signals are affected by the tag noise and robot reflections? Imperfect tag circuit design and robot reflections would introduce a lot of noise into the propagation signatures. To capture reliable features, our observation is that the backscatter propagation varies dramatically when the robot is operating in different movement states. Therefore, we divide the feature extraction into the scale-variation and frequency-variation stages based on the movement states, and capture the features in both time and frequency domains. To eliminate the effects caused by robot reflections and dynamic environments, we further observe that the signal reflections on the robot are dynamic when the robot is in operating condition. In contrast, the backscatter signatures caused by the movement remain stable if without any noise. Motivated by this, we design a Triplet network to combine different times and motions to filter the noise and reconstruct more reliable features. Finally, we build robust profiles of each movement based on the reconstructed features, from which we can detect anomalies and secure the industrial robot.

Summary of results. We implement RobotScatter on two common robotic platforms, Universal Robot and iRobot Create, using commercial LoRa chips SX1276, a diode nonlinear circuit, and several customized backscatter tags, and evaluate our system for over 1,500 operating cycles in a variety of environments, including the cluttered laboratory, empty meeting room, and narrow corridor. The experimental results show that RobotScatter achieves a high overall accuracy of 97.7%, and detects up to 94% of anomalies against small movement deviations of 10mm/s in velocity, and 2.6cm in distance.

(a) Manipulator (b) Control cabinet (c) Teach pendant

Fig. 2. The basic structure of Universal Robot.

Contributions. We summarize the contributions of our system as follows.

- First, we are the first to exploit the nonlinear effect of RF circuits to improve robot security without any intrusive sensors, from which we realize a reliable, adaptive, non-intrusive, low-cost, and ubiquitous security system for industrial robots.
- Second, we extract representative features based on the robot's movement stages and reconstruct robust profiles using a Triplet network to detect anomalies and defend against attacks, which efficiently eliminate the effect of environmental noise.
- Finally, we have also validated performance using customized backscatter tags on real robot platforms that are ubiquitous in the factory and daily life, including Universal Robot and iRobot Create, to show the robustness of our system.

The reminders of this paper are organized as follows. In Section 2, we first discuss security threats to the industrial robot monitoring system, and then explore the signal propagation signatures that motivate our design by exploiting the nonlinear effect at the receiver. Section 3 elaborates each step in our system. Implementation and evaluation are presented in Section 4 and 5, followed by some limitations discussed in Section 6 and the literature review in Section 7. Finally, we conclude this work in Section 8.

2 MOTIVATION

We first investigate industrial robots and the potential threats to the SCADA system, and then we characterize the backscatter signatures with the nonlinear effect at the receiver to show that the nonlinearity can be exploited to secure the robots. Finally, we prototype a simple platform to demonstrate the feasibility.

2.1 Industrial Robots and Threat Model

2.1.1 Industrial Robots. As shown in Fig. 2, an industrial robot, such as Universal Robot, consists of a manipulator, a control cabinet, and a teach pendant. In particular, the manipulator of Universal Robot consists of 6 motor-driven rotary joints, where Joint-1 is near the base of the manipulator and Joint-6 is close to the gripper. All joints are connected by the links to form a robot arm, by a combination of which the arm can perform various operations. The control cabinet controls the manipulator and also communicates with the SCADA system to monitor the robot. As for the teach pendant, an operator can program it to perform movements. Since the operator is usually limited access to the dangerous workshop, the robot downloads the code and cyclically performs the specific operations. At the same time, the robot records the movement data and then sends it to the gateway. Based on the priori knowledge of the operations, the operator can analyze whether the robot is moving according to the pre-determined operations.

2.1.2 Robot Movements. As presented in[37], operators can control the robot to move between the pre-defined waypoints, and one can select one of three types of movements to drive the robot, including MoveJ, MoveL, and MoveP. In particular, MoveJ drives the joints to move at the same time and helps to move fast between waypoints, resulting in a curved path to the Tool Center Point (TCP). MoveL performs a complicated combination of the joints to move the TCP linearly between waypoints. MoveP controls the TCP to move linearly with constant velocity and circular blends between the waypoints, where the blend radius is also pre-defined by the operators.

2.1.3 Attack Model. Recent efforts have reported that industrial robots are vulnerable to active attacks, such as man-in-the-middle attacks and replay attacks. In this system, we consider the following typical threats in wireless industrial robot monitoring systems. First, an attacker can break the system and forge the certificate, and then impersonate the legitimate robot to perform a man-in-the-middle attack and send the pre-record movement data to fake the system. Second, we also consider the scenario where an attacker has all the priori knowledge of the robot movements, including the operations, speed, and duration. Then, it replays the movement data that is similar but has some deviations to spoof the system, from which the monitoring system struggles to capture the real state. Finally, we have also considered the state when the robot's data collection system is malfunctioning and some movement data is missing so that the system cannot detect anomalies and fails to protect the robot.

2.2 Propagation Signature with Nonlinear Effect

In order to remove the above threats, we build a reliable and non-intrusive security scheme, where we exploit the nonlinear effect to fuse the backscatter propagation signatures to uniquely determine the robot movements. For simplicity, as described in Fig. 2, we employ only two tags attached to two separate robot links that share the same joint. Accordingly, these two tags will create two propagation paths by reflecting signals launched from the robot's transmitter. According to polarization theory in [38], when the tag reflects the excitation, the strength $R_{b,i}$ reflected by the tag i can be written as

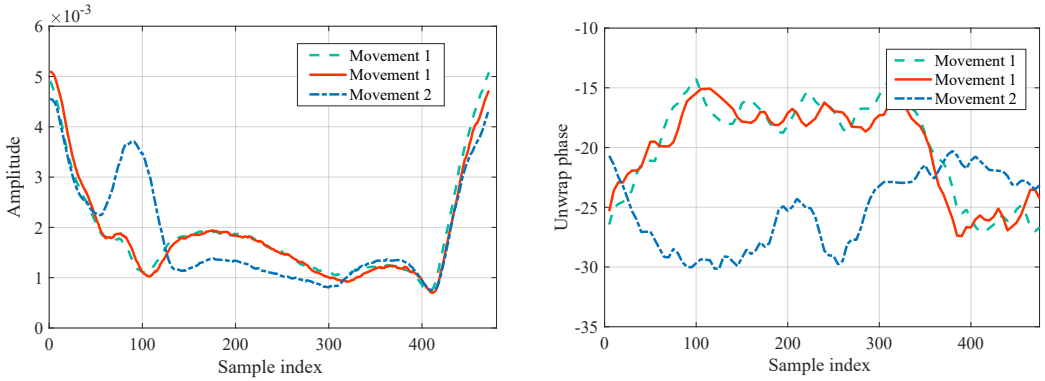
$$\begin{aligned}\mathcal{R}_{b,i} &= \sqrt{\frac{\lambda^2 \mathcal{G}_b(\arcsin(\langle \mathbf{K}, \mathbf{P}\rangle), \gamma_b)}{4\pi Z_0 \cos^2(\arcsin(\langle \mathbf{K}, \mathbf{P}\rangle))}}\, \langle \mathbf{K} \otimes \mathbf{P} \otimes \mathbf{K}, \mathbf{E}(\mathbf{K}, D_b)\rangle \\ &= \sqrt{\frac{\lambda^2 \mathcal{G}_b(\theta_b, \gamma_b)}{4\pi Z_0}} \cos(\phi_b)\, \|\mathbf{E}\|\end{aligned} \tag{1}$$

where λ is the wavelength, $\mathbf{K}$ denotes the normal of equiphase plane, and $\mathbf{P}$ is the antenna's direction. $\mathbf{E}$ is the waveform energy density (J/m^2), which is a function of communication distance D_b between the transmitter and the tag, $\mathcal{G}_b$ is the received gain of the tag, which is constrained by the antenna direction and the reflection coefficient γ_b. Due to imperfect circuit manufacturing, various tags will have different reflection coefficients. θ_b and ϕ_b indicate the elevation angle and polarization angle of the incident signal. According to Eq. (1), when we drive the joint to rotate by an angle, the elevation angle, polarization angle as well as propagation distance change with the rotation, leading to the variations of the signal strength at the tag. Thus, it is obvious that the received signal strength highly depends on the space trajectory of the robot movement.

The reflection of Tag_1 is assumed to be $\mathcal{R}_{b,1} \sin(2\pi f t)$, where f is carrier frequency. Compared to Tag_1, the propagation distance varies with the space trajectory due to the robot's movement, and a phase shift of $\psi(t)$ is introduced in Tag_2 accordingly. Thus, we achieve the signal at Tag_2 is $\mathcal{R}_{b,2} \sin(2\pi f t + \psi(t))$. Based on the nonlinear effect of the receiver's amplifier and circuit [44], all the backscatter propagation paths can be fused as

$$\mathcal{Y}(t) = \alpha_0 \mathcal{X}_{in} + \alpha_1 \mathcal{X}_{in}^2 + \alpha_2 \mathcal{X}_{in}^3 + \ldots + \alpha_i \mathcal{X}_{in}^i \tag{2}$$

where $\mathcal{X}_{in}$ is the combination of two backscatter paths, α_0 denotes the amplifier performance, and α_i indicates the nonlinear coefficient, which significantly relies on the hardware performance of the receiver. We extract the



(a) Amplitude signatures.

(b) Phase signatures.

Fig. 3. When the robot performs repetitive motions, we collect similar backscatter propagation. Whereas, the signatures will be significantly difference if it perform different movements.

second-order harmonic and rewrite it as

$$\begin{aligned}\mathcal{Z}(t) &= \alpha_1[\mathcal{R}_{b,1}\sin(2\pi ft) + \mathcal{R}_{b,2}\sin(2\pi ft + \psi(t))]^2 \\ &= -\frac{\alpha_1}{2}[-(\mathcal{R}_{b,1}^2 + \mathcal{R}_{b,2}^2) - 2\mathcal{R}_{b,1}\mathcal{R}_{b,2}\cos\psi(t) \\ &+ \mathcal{R}_{b,1}^2\cos(4\pi ft) + 2\mathcal{R}_{b,1}\mathcal{R}_{b,2}\cos(4\pi ft + \psi(t)) + \mathcal{R}_{b,2}^2\cos(4\pi ft + 2\psi(t))]\end{aligned} \tag{3}$$

The output signal at second-order harmonic is a combination of two input frequencies $2f = f + f$ and $0 = f - f$. Then, we focus only the $2f$ terms and further simplify Eq. (3) as

$$\mathcal{Z}'(t) = \alpha_1\mathcal{A}(\mathcal{R}_{b,1}, \mathcal{R}_{b,2}, \psi(t))\sin(4\pi ft + \Phi(\mathcal{R}_{b,1}, \mathcal{R}_{b,2}, \psi(t))). \tag{4}$$

Accordingly, both two backscatter paths are fused under the nonlinear effect at second-order harmonic, which are highly constrained by the receiver's hardware characteristic, backscatter performance, as well as the amplitude and phase variations caused by the robot movement. Therefore, we can simultaneously extract the signal strength and phase to uniquely determine the robot's identity and spatial trajectory.

2.3 Feasibility Study

To present a better intuition, we employ low-cost backscatter tags and USRP B210s on a Universal Robot platform to verify the above idea. In particular, we first place the antenna of one USRP on the robot to act as the transmitter and attach two tags to different robot links that connect to one joint, where the backscatter tag is customized following [27, 55]. Then we download the code and drive the joint to perform specific operations cyclically. At the same time, we control these two backscatter tags to reflect the signals launched from the transmitter. Although the nonlinearity of the commercial devices is ubiquitous, the capabilities are diverse and unstable. In order to achieve reliable nonlinear performance, similar to [22, 44], we exploit a diode to design a battery-free nonlinear circuit chip to the antenna port of the receiver. Finally, we receive the fused backscatter propagation at the second harmonic and extract the amplitude and phase features to show the robot movement states.

As presented in Fig. 3, it is obvious that the features vary dramatically with the robot's movement. In addition, the changes in amplitude and phase are highly similar when the robot arm performs the same operation. Whereas, if the robot conducts different operations, the backscatter signatures would change accordingly. The reason is

that the different polarization angles and backscatter distances lead to the diversity in amplitudes and phases when the robot drives to move, which are captured at the second harmonic. This observation verifies that the nonlinear effect at the second harmonic can fuse the backscatter propagation signatures that are sensitive to the backscatter performance and movement spacial trajectory. Therefore, we can extract them to construct a unique movement profile to improve the security of wireless monitoring systems for industrial robots.

3 SYSTEM DESIGN

In this section, we first present an overview of RobotScatter where we divide our system into two stages. Then, we elaborate each stage in the following sections.

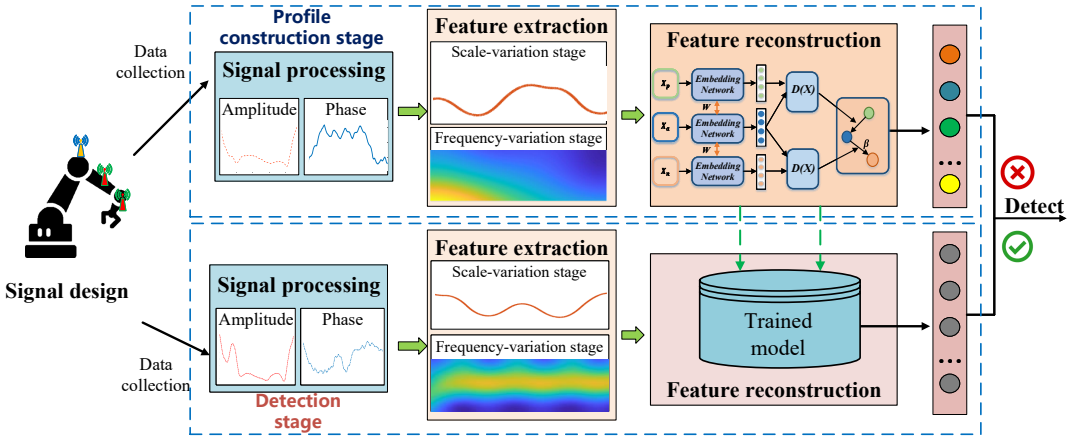
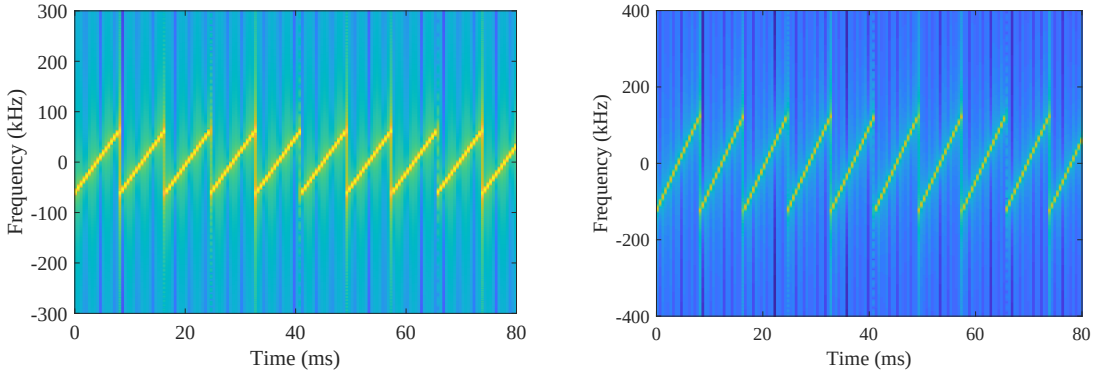


Fig. 4. An overview of our system. RobotScatter is divided into two stage: Profile construction stage and Detection stage, which consists of five components, including Signal design, Signal processing, Feature extraction, Feature reconstruction, and Attacker and anomaly detection.

3.1 System Overview

As illustrated in Fig. 4, we divide our system into to profile construction stage and detection stage, which consist of the following four steps in the profile construction stage and a one-step detection stage:

- **Signal design.** The key insight of RobotScatter is to fuse the backscatter signatures with the nonlinear effect of the device to secure the LoRa-based wireless monitoring system for industrial robots. Therefore, our first step is to design the backscatter scheme to avoid interference while guaranteeing that the gateway can still receive the signal.
- **Signal processing.** After collecting the signals, we further calibrate starting points of the robot movement and segment the backscatter propagation to acquire the amplitude and phase signatures that are sensitive to the robot movement.
- **Feature extraction.** According to the backscatter signatures, we extract reliable features in both the time and frequency domains based on the robot movement state.
- **Feature reconstruction.** This component further removes the interference of the dynamic environment where we employ a Triplet network to reconstruct the features and build profiles of the movements.
- **Attacker and anomaly detection.** When a suspicious message is received, we reconstruct the features from the backscatter signals by the well-trained model and match it with the movement profiles to detect active attacks and anomalies.



(a) LoRa chirp at the 433MHz band.

(b) LoRa chirp at the 868MHz band.

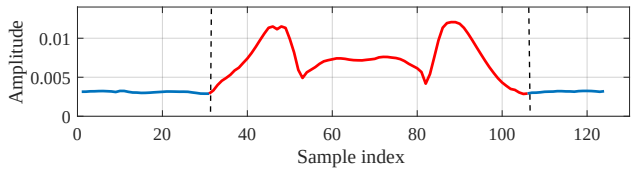
Fig. 5. When launching LoRa signals at 433MHz, the receiver can collect the signals at 868MHz after the nonlinear effect.

In this paper, we still employ a two-tag model to demonstrate our system. As illustrated in Section 2.3 and Fig. 2(a), tags are attached to two links that share the same joint, and then we drive this joint to cyclically move.

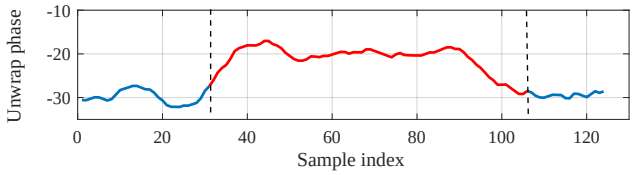
3.2 Signal Design

In a LoRa-based wireless monitoring system, industrial robots report movement data to the gateway in the SCADA systems via LoRa modules. Meanwhile, RobotScatter takes advantage of the tags to reflect the LoRa signal and exploits the nonlinear effect of the receiver to fuse the sensitive backscatter signatures to defend against attacks. However, the backscatter paths on the same channel will introduce interference to the LoRa transmission. To avoid this issue, an intuitive approach is to shift the backscatter signal into an idle channel. Nevertheless, the received signal is a combination of various frequency harmonics after nonlinear fusion at the receiver. Therefore, we need to guarantee that the fused propagation is still located at the LoRa band so that the gateway can capture the backscatter propagation signatures.

To the end, we first investigate there are three widely used frequencies in the LoRa protocol, including 433MHz, 868MHz, and 915MHz. Besides, many countries license both 433MHz and 868MHz bands at the same time. Inspired by this, we first control the LoRa transmitter on the robot to launch signals at the 433MHz band when the robot starts to move. At the same time, to avoid interference, we toggle the switch of these two backscatter tags to reflect the excitation signal at 1 MHz frequency. As a result, RobotScatter can achieve the backscatter signals at 434MHz after tag reflections. Then, the fused backscatter propagation will contain the $2f$ component at the second harmonic due to the nonlinearity shown in Eq. (4). Therefore, RobotScatter can successfully receive the chirp signal at the 868MHz band at the second harmonic, which is still located in the LoRa band. In addition, since LoRa modulates the signal with a spread spectrum and demodulates the signal by detecting the starting frequency of each chirp, a slight shift in phase will not change the LoRa chirp after nonlinearity. Accordingly, we can still extract the features in each LoRa chirp at the gateway to secure the robot. To prove it, we control the LoRa transmitter to launch the radio at 433MHz and attach two tags on the robot to shift the signals to 434MHz. As shown in Fig. 5, we can successfully receive the LoRa chirps at the 868MHz band.



(a) Backscatter amplitude.



(b) Backscatter relative phase.

Fig. 6. RobotScatter calibrates the starting point by detecting the signal strength variance.

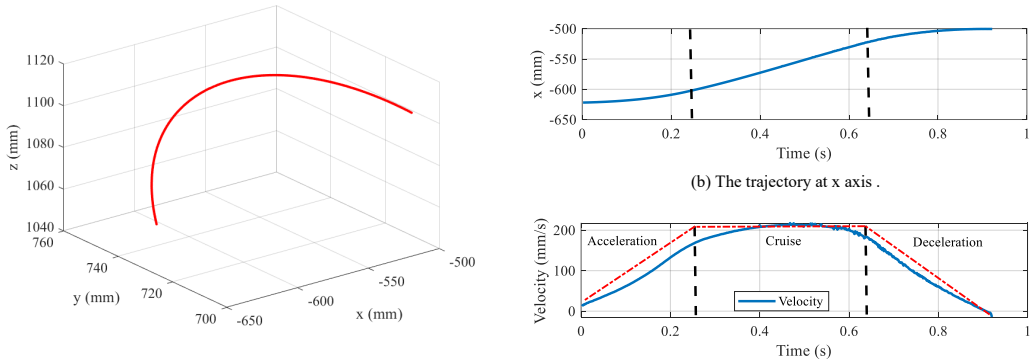
3.3 Signal Processing

After collecting the fused backscatter signals, the first step of RobotScatter is to determine the starting and ending points of the motion in order to accurately capture the features in the movement chunk. Toward this end, RobotScatter first controls the LoRa transmitter to send the packets to inform the movement initiation time. Then, RobotScatter can successfully capture the features of a specific motion based on the priori knowledge of the motion duration. However, the imperfect hardware and initiation delay may introduce some deviations from the starting point. Thus, we turn to further calibrate it based on amplitude changes.

In order to extract the backscatter amplitudes, we employ a two-folded filter to remove the ambient noise. In particular, we first employ a widely-used moving average approach to smooth the backscatter signal, where we set the smooth period as the length of each LoRa chirp. After that, we segment the smooth data into each chunk with the duration of the chirp. Then, we calculate the average signal strength of the chunk and design an interpolation filter to further remove the noise. Accordingly, this two-folded filter operation enables RobotScatter to collect clear amplitude signatures. Based on the amplitude, we can further calibrate the starting points of the robot's movement. Inspired by the fact that robot's movement will lead to a dramatic fluctuation in the backscatter amplitude, we leverage the amplitude variation scale to search for the starting point. In particular, we have the joint rotation duration t_m stored in the data set and also the raw initiation time t_s sent from the robot. Since the starting point should be located nearby t_s, and we thus narrow the searching range in $(t_s - \frac{t_m}{2}, t_s + \frac{t_m}{2})$. Then, we calibrate the starting point by solving the following optimization problem as

$$\begin{aligned} \tau_s^* &= \arg\max_{\tau_s} \frac{\sum_{i=F_s\tau_s}^{F_s(\tau_s+t_m)} (\mathcal{A}_i - \hat{\mathcal{A}})^2}{F_s t_m} \\ \text{s.t.,}\ \ \tau_s &\in (t_s - \frac{t_m}{2}, t_s + \frac{t_m}{2}) \end{aligned} \tag{5}$$

where τ_s is the starting point that we need to search. $\mathcal{A}_i$ denotes the amplitude of sample i , $\hat{\mathcal{A}}$ the amplitude average in a movement segment duration $(F_s\tau_s, F_s(\tau_s + t_m))$ with the sample rate F_s. As shown in Fig. 6, we can well calibrate and segment the backscatter signals.



(a) Spatial trajectory of the robot movement.

(b) The trajectory at x axis .

(c) Velocity profile for the movement.

Fig. 7. Motion state of the robot movement. The motion state of the movement is divided into three stages: acceleration, cruise and deceleration.

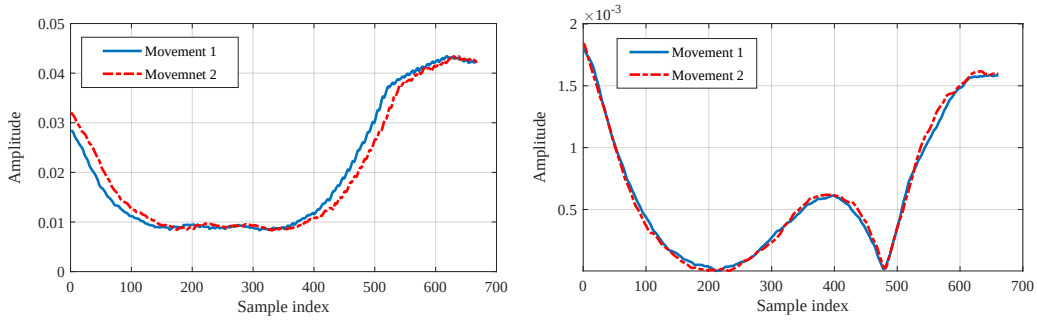
Subsequently, we extract phase signatures in the motion chunks. According to the analysis in Eq. (4), the phase shift depends on variations of the spatial trajectory of the robot movement, and phase changes for repetitive motions remain stable. Therefore, to capture the phase signatures, we fetch the relative phase in the motion chunks instead of extracting each sample's phase directly. Specifically, to avoid the deviation caused by the phase discontinuity, we first calculate the unwrap phases of the signal samples. After that, we divide the unwrap phase into different segments with a length of the LoRa chirp to construct the phase matrix as

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1M} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N1} & \phi_{N2} & \cdots & \phi_{NM} \end{bmatrix} \tag{6}$$

where N is the sample number of each LoRa chirp, and M denotes the number of the segments. Then, we calculate the phase average of each segment as $\Phi = [\hat{\phi}_1, \hat{\phi}_2, ...\hat{\phi}_i...\hat{\phi}_M]$. Consequently, we have the relative phase changes among the chirps as $\Delta\hat{\phi}_i = \hat{\phi}_{i+1} - \hat{\phi}_i$ and accordingly we can capture the relative phase changes as $\Delta\Phi = [\Delta\hat{\phi}_1, \Delta\hat{\phi}_2, ...\Delta\hat{\phi}_i...\Delta\hat{\phi}_{M-1}]$. As presented in Fig. 6, we can achieve a stable phase variation signatures when we repeat the same operation. In addition, it is obvious that the variations in amplitude and relative phase are well matched.

3.4 Feature Extraction

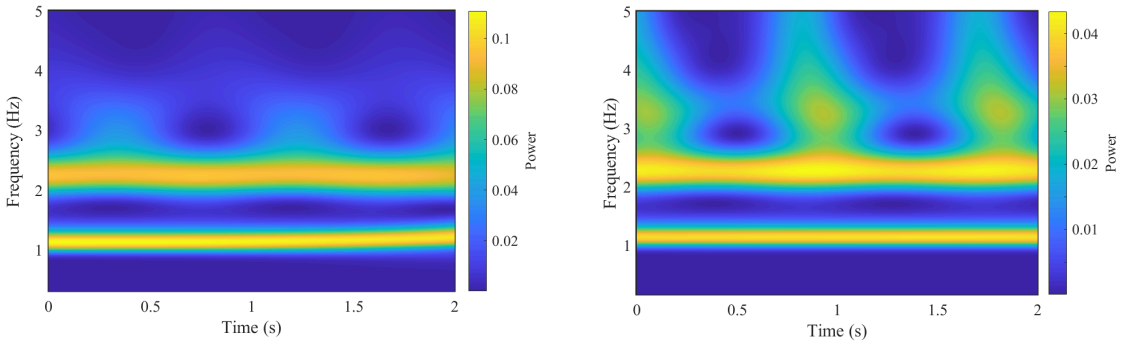
Based on the above processing, RobotScatter collects amplitudes and phases, whose fluctuations represent changes of the robot movements. However, the signatures are not completely identical even though the motion is repeated in a short duration. One of the reasons is that the backscatter signals are easily affected by the imperfect tag circuit and environmental noise. Therefore, we refine the fused backscatter propagation signatures and extract representative features to characterize robot movements. As shown in Fig. 7, our key observation is that when the robot starts to operate, its motion state can be divided into three stages: acceleration, cruise and deceleration. As the red line presented in Fig. 7(c), the robot remains at a relatively constant velocity in the cruise state, and thus the signatures change with a stable state. In contrast, if the robot tends to start, stop, or rotate its direction, the movement will turn to the acceleration or deceleration state, where signatures may contain many velocity



(a) Signal amplitudes.

(b) Features by waveform length extraction approch.

Fig. 8. We employ the waveform length extraction approach to capture the features in time domain, where we can well calibrate the features though there is a slight deviation in amplitudes.



(a) Time-frequency spectrum of the robot trajectory.

(b) Time-frequency spectrum of the backscatter signatures.

Fig. 9. Time-frequency spectrum of the robot trajectory and the backscatter signatures.

variations and high-frequency components. Inspired by this character, we divide the feature extraction into two stages: the scale-variation stage and the frequency-variation stage.

Scale-variation stage. When the joints of the robot move circularly with a constant angular velocity, polarization between the transmitter and tag has a limited effect on the signal strength, leading to a stable scale variation of the backscatter features. While if the joint tends to move away from the transmitter or rotate its direction, the signal would change dramatically. Therefore, the signature scale variation represent the movement of the robot. Additionally, an operation cycle usually combines multiple motions of several joints at different time periods. As a result, an operation may contain multiple operation cycles caused by some of the joints, which are noteworthy to characterize the movement. Motivated by the widely used waveform length extraction approach in EMG signal [42] that can efficiently evaluate the scale change and the complexity of the signal, we design a cycle-based waveform length extraction approach to extract the scale feature in the time domain, where we first extract the cycle features and then calculate the scale change and the complexity. We define the cycle-based

waveform length as

$$\mathcal{W}(i) = \frac{1}{M} \sum_{k=i}^{M+i-1} \left| \sum_{n=1}^{N} x_n x_{n-k} - \sum_{n=1}^{N} x_n x_{n-k+1} \right| \tag{7}$$

where x_k is the input sample and M is the waveform length window size. As shown in Fig. 8, when the signal has large scale variations, the waveform length changes followed the scale. While if the signal is in a stable state, we can efficiently suppress the small fluctuation of the signal caused by the noise. In addition, the sudden and sharp variation can be amplified and captured accurately with the waveform length approach. As illustrated in Fig. 8, even though the rotation and the starting point estimation error lead to some differences in the backscatter amplitude, we can still well calibrate them and achieve a similar feature with this extraction mechanism.

Frequency-variation stage. When the robot drives the joints to start, stop or rotate the direction, its motion turns to the acceleration or deceleration state. As a result, this operation brings some frequency variations to the movement trajectory. As presented in Fig 9(a), the high-frequency components of the movement trajectory fluctuate followed by the movements. Since the backscatter propagation is highly sensitive to the movement, the variations of the backscatter amplitudes and phases will not remain stable, and introduce multiple frequency components related to these variations accordingly. Therefore, the frequency variation of the signatures is unique for the robot movement and can be extracted to characterize the robot's operations. Since the feature variation is dynamic and time-related, instead of a fast Fourier transformation (FFT)-based approach, we employ a continuous wavelet transform (CWT) to extract the stable and high-resolution features in the frequency domain. In particular, we define the CWT as

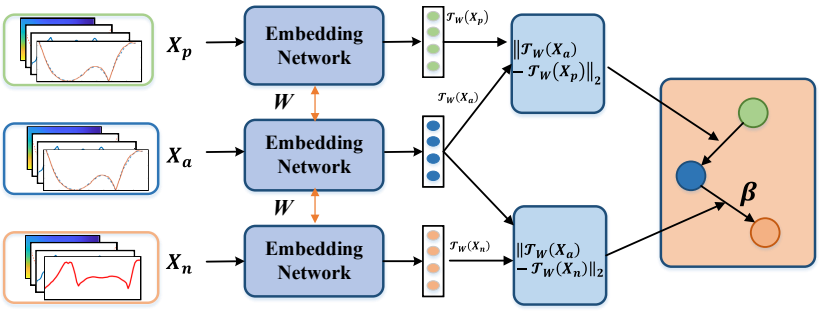
$$\mathcal{F}_{cwt}(a, b) = \int_t x(t) \frac{1}{\sqrt{a}} \Psi(\frac{t-b}{a}) dt. \tag{8}$$

where $\Psi(a, b)$ is the wavelet base function, which significantly the time-frequency result. In our system, we employ a commonly used complex *Morlet* function as the wavelet base function. As described in Fig 9(b), similar to the trajectory variations in Fig 9(a), the time-frequency spectrum of the backscatter signatures varies followed by the robot movement as well. Besides, we also observe that the spectrum energy is dominated by the low-frequency component (1-1.2Hz in the figure), which is caused by the coarse-grained movement of the robot. Conversely, the frequency changes caused by the rotation concentrate in a higher band (2-4Hz), which is affected by the number of the rotation joints, rotation duration as well as acceleration. To capture these frequency variations, instead of employing the whole spectrogram, we only capture the spectrum at a higher band at 2-5Hz based on the parameter settings of the acceleration and the number of joints.

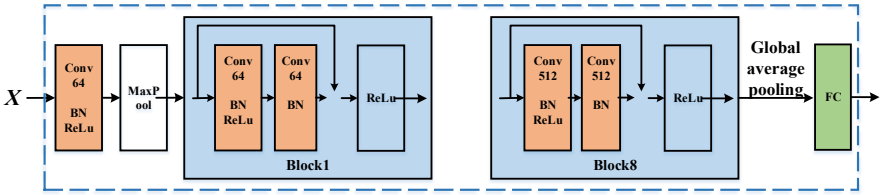
3.5 Feature Reconstruction

In addition to the tag noise, the reflections introduced by robots and environmental obstacles have great impacts on the backscatter signatures. Thus, we need to further eliminate the interference and construct a reliable backscatter profile for each movement. Intuitively, the reflected medium will change following the robot movement, resulting in the signal reflection being dynamic all the time. In contrast, if the same motion is repeated without any noise, backscatter signatures would be identical, which motivates our design of reconstructing reliable profiles from the repeated movements. Inspired by character and [51], we adopt a Triplet network [19] to reconstruct the features, which provides high discrimination power while using both in-class and inter-class relations of various robot movements.

As shown in Fig. 10(a), a Triplet network consists of three embedding networks where they are the same feedforward neural network and share the same weight. The three inputs to the embedding network are defined as positive, anchor, and negative features. Given an anchor input X_a from any movement class, the positive input X_p should be the same movement class as the anchor, while features from the other different classes X_n are sent to the negative block. Then, the Triple network outputs two intermediate L_2 distances between



(a) Triplet Network



(b) Embedding Network

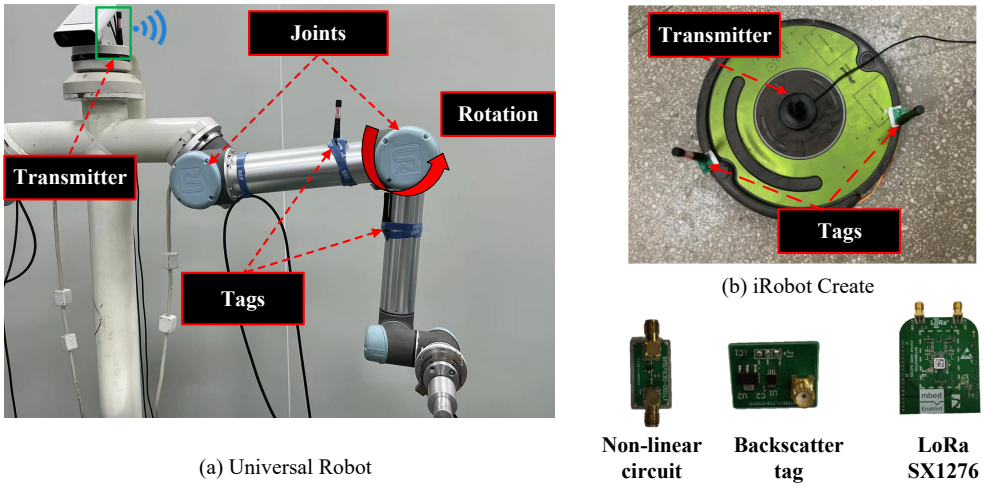
Fig. 10. The structures of the Triplet network and the Embedding network.

two embedded representations, which are captured by the embedding networks. The key insight of the Triplet network is to distinguish the similarity by measuring these two distances. Specifically, when features are from the same movement class, the network favors a small distance between these two embedded representations. While features come from different classes, the embedding network increases their distances. Consequently, the effect of noise caused by environmental reflections on the same movement will be neglected, while the occasional similarity of different movement classes can be removed, and thus we can achieve robust feature profiles for each movement. In our design, to measure the metric distance, we define the loss function of the Triplet network as

$$\begin{aligned}\mathcal{L}_{Triplet}(W) = \max(0, \|\mathcal{T}_W(X_a) - \mathcal{T}_W(X_p)\|_2 \\ - \|\mathcal{T}_W(X_a) - \mathcal{T}_W(X_n)\|_2 + \beta)\end{aligned} \tag{9}$$

where $\mathcal{T}_W(X_a)$, $\mathcal{T}_W(X_p)$ and $\mathcal{T}_W(X_n)$ are the embedded representations for anchor, positive and negative inputs, respectively. With the same weight W, the Triplet network can update the model by decreasing the distance of embedded representations from the same robot movement and increasing that of different movements.

To capture deep features of the movement, we leverage a series of convolution neural networks (CNNs) and connect a fully connected layer to construct the embedding network. However, simply adopting multiple-layer connected CNN makes the learning structure fail to extract any valid features in the deep layer caused by the vanishing gradient effect [15]. Thus, we build the embedding network by exploiting a residual network (ResNet) instead [15]. Concretely, as shown in Fig. 10(b), we employ a 17-layer CNN and add the shortcut link to avoid the vanishing gradient problem, and also add a 512-unit fully connected layer at the end to regress a reconstructed feature vector (i.e., embedded representation), which then can be used to construct profiles for robot movements.



(a) Universal Robot

(b) iRobot Create

Fig. 11. We implement our system in Universal Robot and iRobot Create.

3.6 Attacker and Anomaly Detection

With the reconstructed feature vector, we build profiles for the movement of the industrial robot as $\mathbf{P}_N = [\mathbf{P}_1, \mathbf{P}_2 ... \mathbf{P}_i ... \mathbf{P}_N]$, where $\mathbf{P}_i$ represents the feature vector of movement class i. Our final step is to authenticate the robot's movement and detect any abnormal operation. Specifically, if any suspicious motion data is received, we have also collected backscatter signatures, which are then fed into the well-trained model to reconstruct a feature as $\mathbf{Q}$. Moreover, since the system has a priori knowledge of the robot movement, it should know the movement profile of the motion data, which is marked as $\mathbf{P}_i$. Accordingly, we compare the similarity by calculating the distance

$$\mathcal{D}_i = \sqrt{\|\mathbf{P}_i - \mathbf{Q}\|_2}. \tag{10}$$

We empirically set a threshold as η to compare the similarity of the features. If the distance between the suspicious movement and class i is smaller than η, the system would be recognized as being normally working at movement i. Whereas, the system is attacked by an active attacker or experiences an abnormal movement.

4 IMPLEMENTATION

As shown in Fig. 11, we implement RobotScatter on commercial robots with LoRa transmitter, GNURadio USRP, and backscatter tags. Specifically, we employ the commercial Universal Robot and program to drive its joints to perform different operation cycles. In our system, we deploy the transmitter on the robot to launch the LoRa signal at 433MHz, where we select three kinds of transmitters for comparison, including USRP B210, X310, and commercial LoRa chip SX1276. As for the backscatter tags, we customize them using off-the-shelf low-cost circuit components following the design in [27, 55], and toggle the RF transistor in the tag at 1MHz to avoid interference. Then we attach the tags to the robot links to reflect the signals. At the receiving end, as shown in Fig. 13, we design a battery-free nonlinear circuit as [22, 44] to achieve stable nonlinear performance, where we employ only a capacitor and a diode mounted on a printed circuit board. Finally, we connect the nonlinear circuit to the antenna port of a USRP B210 and turn it to receive the backscatter signal at 868MHz. Besides, we control another USRP and PC to act as an active attacker, who monitors signal strength variation as well as the robot movements to launch the fake movement data to the receiver.

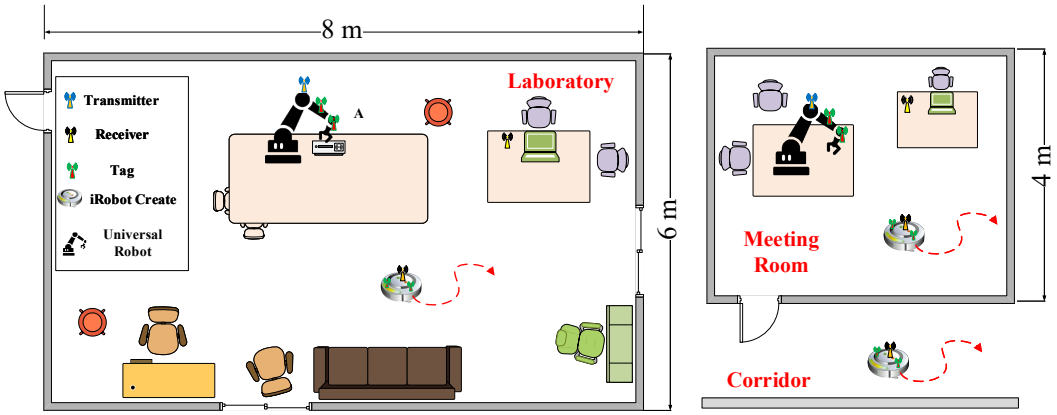


Fig. 12. We evaluate our system in various environments, including the meeting room, laboratory and corridor.

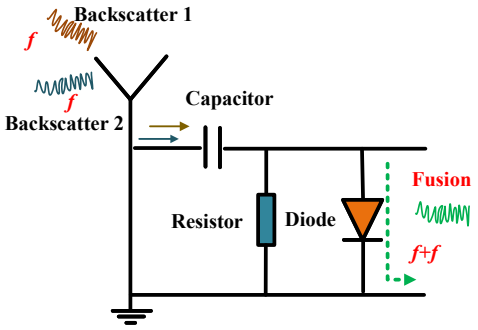


Fig. 13. We exploit a diode to design a nonlinear circuit.

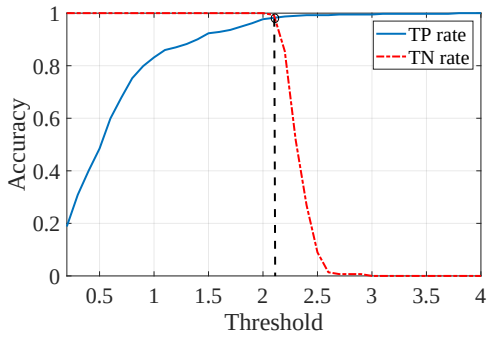


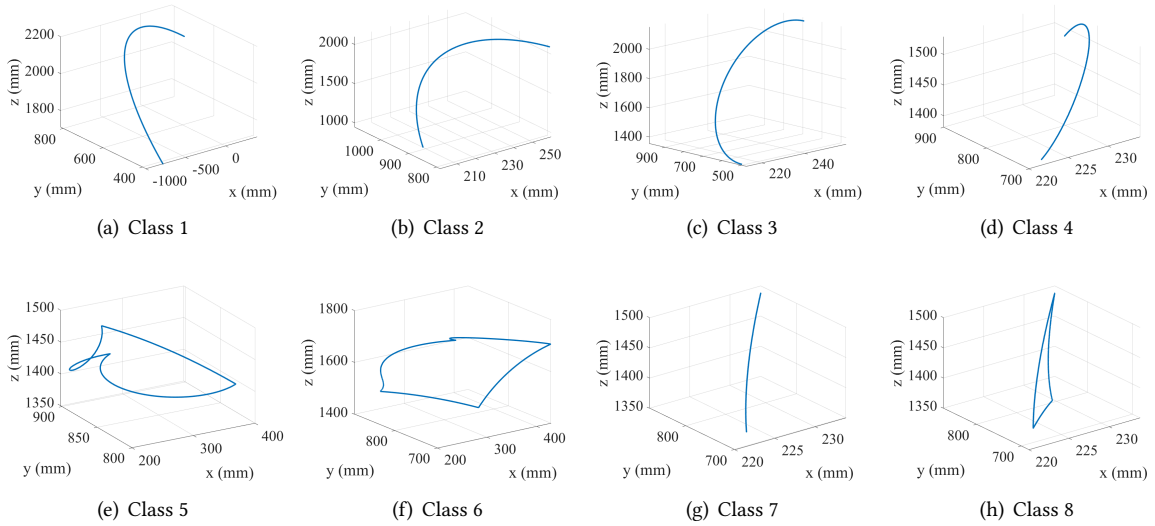
Fig. 14. The performance with different threshold η.

5 EVALUATION

5.1 Experimental Methodology

As presented in Fig. 12, we evaluate the performance of RobotScatter in various environments, including a laboratory with the size of 8m×6m, a meeting room with the area of 4m×4m, and also a corridor with a width of 1.5m. In our experiment, a large number of obstacles, such as cabinets and some furniture, are deployed in the laboratory to simulate a cluttered workshop. While in the meeting room, we remove all the obstacles to act as an independent lathe shop. Besides, we program to drive the robot to play warehousing or delivery tasks in the narrow corridor.

As described in Fig. 2(a), the manipulator of a Universal Robot consists of 6 motor-driven rotary joints, from which the robot arm can perform the desired movements with a complicated combination of the joints [35, 46]. Therefore, the potential movements of these 6 joints are the foundation of the robot operation and thus we select the possible rotation involving all 6 joints as the representative tasks and the basic legitimate movement classes in the profile construction stage. As shown in Fig. 15(a)-(h), Class 1-4 are single-joint operations where we drive only Joint-1 to Joint-4 to sweep half of the possible angles with the same velocity of 60°/s. In contrast, Class 5-8



(a) Class 1 (b) Class 2 (c) Class 3 (d) Class 4

(e) Class 5 (f) Class 6 (g) Class 7 (h) Class 8

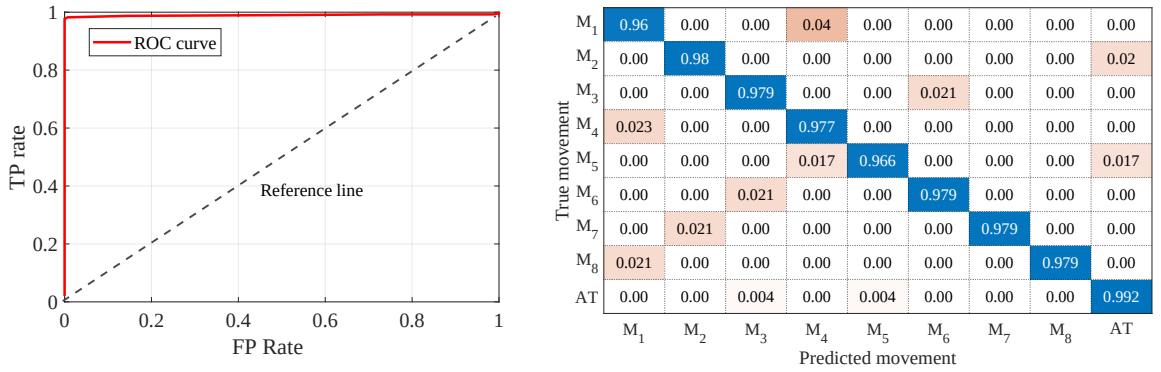
Fig. 15. We define 8 basic movement classes.

are multi-joint movements, consisting of the combinations of Joint-1 and Joint-5; Joint-1, Joint-2 and Joint-5; Joint-3 and Joint-6; Joint-3, Joint-4 and Joint-6, respectively, where Joint 5 and Joint 6 rotate half of the potential angles as well. Then, as presented in Fig. 11(a), we deploy the transmitter close to the base and backscatter tags on the moving links, and capture the features for movements with over 1,500 operation cycles in above these three environments. Finally, we employ a total of 800 movement features to train the Triplet network and construct the profiles for each movement class.

In the detection stage, we first require the robot to repeat these 8-class operations and reconstruct the features in the well-trained model to identify these basic legitimate movements. As for the attacker, we define two types of attacks to evaluate the performance. In particular, the first type of attacker employs a USRP connected to a PC to monitor the robot operations and send a fake movement class that is different from the recent basic movement, to spoof the system. Then, the SCADA system detects attacks by comparing the corresponding reconstructed features between the fake movement class and the recent one. The second type of attacker is required to repeat above the basic operation cycles but rotate to an opposite direction and sweep the other half of the possible angles. Finally, we also evaluate the performance of anomaly detection where we control small deviations to the robot movements.

Evaluation metrics. In our experiment, we define the following three metrics to evaluate the performance of RobotScatter.

- **Accuracy.** Accuracy is the ratio of the movement samples that are correctly recognized to the total number of movement samples
- **TP rate.** TP rate is the ratio of the number of the movement samples collected from these 8 classes that are correctly authenticated to the total number of these 8 movement classes.
- **FP rate.** FP rate is the ratio of the number of the attack movements that are falsely recognized as legitimate samples to the total number of the attack samples.



(a) ROC curve. (b) Overall accuracy.

Fig. 16. System performance.

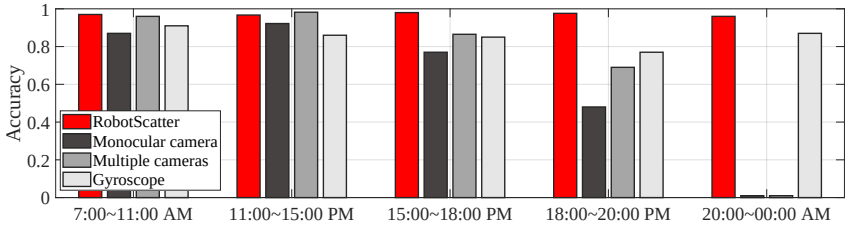


Fig. 17. Comparison with sensor-based approaches.

5.2 Threshold Determination.

As described in Section 3.6, RobotScatter detects anomalies and defends against attacks by comparing the similarity between the suspicious movement and constructed profile with a threshold. Therefore, the first step of RobotScatter is to determine this threshold η. To this end, we traverse η from 0.1 to 20 and then compare the trend of TP rate and TN rate, where TN rate is the ratio of attack samples that are successfully recognized as the attack operations to all the attack samples. As shown in Fig. 14, TP rate increases follow by η, while TN rate decreases. The reason is that a large distance will bring more attack samples into the legitimate bound, at the same time, the legitimate samples far from the class can also be included. To make a trade off, we select the intersection point of these two curves for our system, where we achieve the threshold of 2.1 and accuracy of 0.986.

5.3 Attack Detection

5.3.1 Overall Accuracy. We first present the ROC curve and overall accuracy to show the efficiency of our system, where we mark 8 basic movement classes $M_1, M_2 ... M_8$, and active attackers as AT. As described in Fig. 16, the ROC curve increases closely following the TP rate and then reach by 1 sharply, where we have the area under the ROC curve as 0.99 close to an ideal case of 1. besides, we also have an authentication accuracy of all these 8 basic classes large than 97.6%, which demonstrates that our system can robustly profile each robot movement. As for the active attacker, we have successfully detected up to 99% of attack attempts. We count the overall accuracy greater than 97.7%, which validates the robustness of RobotScatter in distinguishing legitimate movement and attacker.

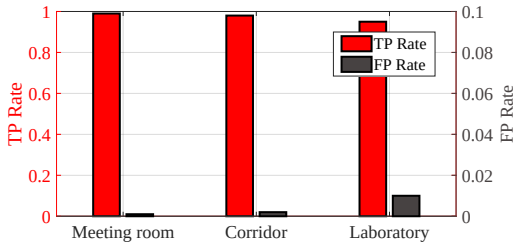


Fig. 18. TP and FP rate under various environments.

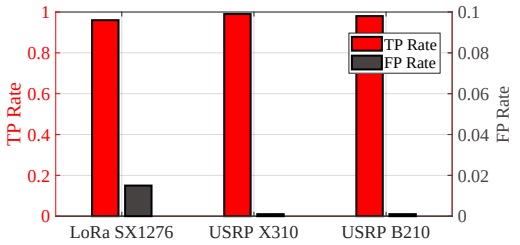


Fig. 19. TP and FP rate with respect to different devices.

5.3.2 Comparison with Sensor-based Approaches. Then, we compared RobotScatter with the typical sensor-based approaches, including high-resolution cameras and gyroscopes. In the camera-based system, we deploy a monocular camera to monitor the robot's movement and detect abnormal motions. To improve the detection rate, we also cooperate 3 monocular cameras around the robot arm to monitor the operations. As for the gyroscope-based solution [10], we install a gyroscope on the gripper and collect the vibration signals to detect movements and anomalies. In our experiment, we evaluate the performance of illumination changes at different time periods, where we test motions as [33], including sudden start, sudden stop and movement recognition. As presented in Fig. 17, the camera-based approach is significantly restricted by poor illumination, while the gyroscope-based system requires invasive mounting and suffers from large drift. In contrast, RobotScatter achieves comparable results to the sensor-based system and is robust to the illumination conditions, which would effectively improve the monitoring ability and security of the SCADA system.

5.3.3 Impact of Experiment Environments. After that, we validate the performance when the robot is operating in various environments, including the meeting room, corridor, and laboratory. As shown in Fig. 18, we have the TP rates in the meeting room and corridor as 99% and 98%, which are higher than the result in the laboratory of 95%. Besides, RobotScatter achieves a higher accuracy of detecting almost 99.7% of attack attempts. The reason is that the obstacles in the laboratory introduce rich multipath due to the scatter and reflection. Even though we have filtered the signals, the signatures are still affected by the ambient noise. However, we have still successfully authenticated 98% of the legitimate movement and mitigated up to 97% active attackers, which shows the efficiency of our feature extraction approaches and Triplet network.

5.3.4 Impact of Different Devices. We also compare the results with different transmitters, including commercial LoRa chip and GNURadio USRP, to launch the LoRa signal on the industrial robot. In order to collect the in-phase and quadrature (IQ) data, we exploit a USRP B210 to receive the LoRa signal. As presented in Fig. 19, our system can accurately authenticate 96% legitimate movements and mitigate nearly 98% active attacks with the LoRa SX1276 chip. In contrast, we achieve the TP rate of 98% and FP rate of 99% by using USRP B210, which is higher than that with LoRa chip. The reason is that the commercial device introduces noise by its circuit design, which brings interference to the backscatter signatures. However, combined with the above results, our system can achieve great accuracy with both Soft Defined Radio (SDR) and commodity devices.

5.3.5 Evaluation Using The Mobile Robot. In this section, we implement our system on iRobot Create, a mobile robot platform designed for cleaning tasks as well as robotics development. In our experiment, we select the representative motions for the floor-mopping task, including moving straight, backward, and rotating in place. Specifically, we first control the robot to move straight along the wall as well as from other different angles. Then, we also make iRobot Create drive backward with the same trajectories. To perform the common anomalies of deviating from the route, we also drive the robot to change its navigation directions. As for the rotation test, we

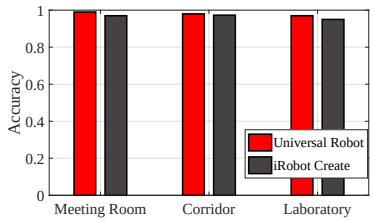


Fig. 20. Accuracy using various robots.

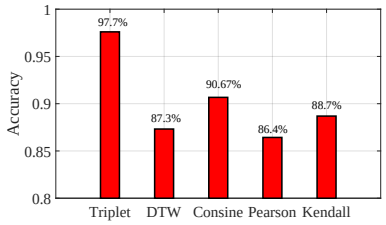


Fig. 21. Accuracy of different distance metrics.

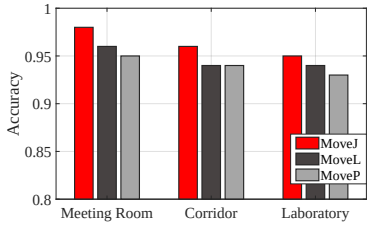
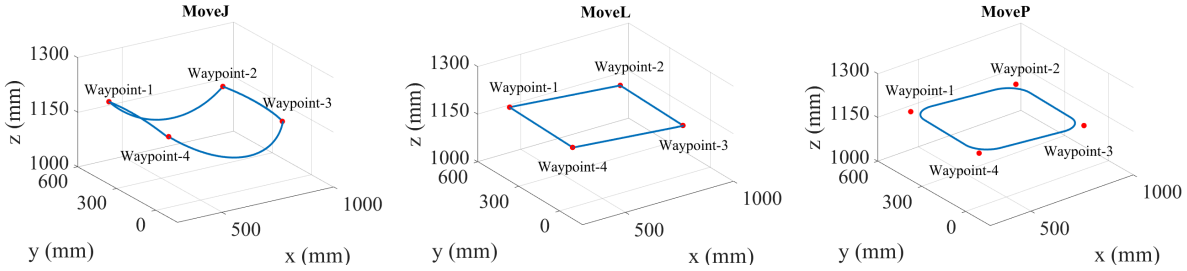


Fig. 22. Accuracy of complex movements.



(a) MoveJ (b) MoveL (c) MoveP

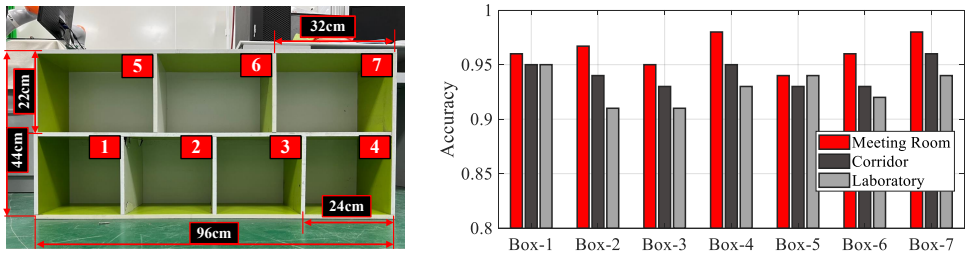
Fig. 23. Trajectories of complex movements, including MoveJ, MoveL and MoveP.

fix the robot to the same position and drive the wheels to rotate from two opposite directions for the same angles. As presented in Fig. 20, we have an overall accuracy of 96% on iRobot Create which is lower than the result of Universal Robot. The reason is that the trajectory of the iRobot Create is hard to accurately control on the floor, resulting in some deviations to the signatures. However, the accuracy still secures many applications that do not require super precision, such as delivery tasks.

5.3.6 Performance of The Triplet Network. We also evaluate the performance of the Triplet network, Specifically, we leverage traditional metrics, including *DTW*, *Consine*, *Pearson*, and *Kendall tau* correlations, to directly compare the similarity among the extracted features without exploiting the Triplet network to filter the noise and reconstruct the features. As shown in Fig. 21, we have the accuracy of all these four metrics lower than 91%, which is much worse than the result by the Triplet network. This means that the ambient noise and robot reflection bring much interference to the backscatter propagation. However, the Triplet network can effectively remove much of the noise to accurately authenticate legitimate movements and defend against attacks.

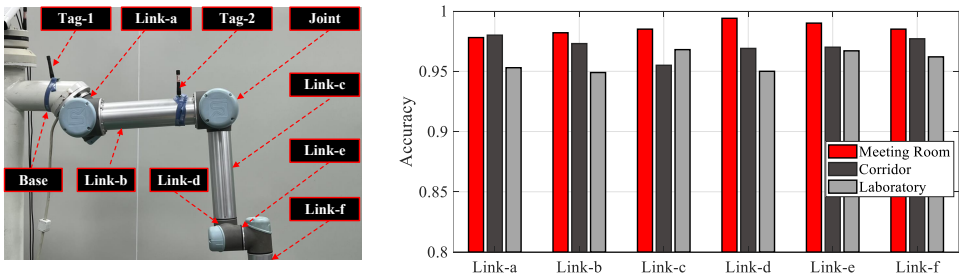
5.3.7 Performance of Complex Motions. In this section, we evaluate the performance with complicated combinations of multiple joints at the same time. As presented in Section 2.1, operators can select one of three types of movements, including MoveJ, MoveL, and MoveP, to drive the robot arm to move between the pre-defined waypoints. Therefore, as illustrated in Fig. 23(a)-(c), we first define 4 waypoints in the platform, and control the gripper to move between the waypoints using these three movement types in turn. Finally, we capture the fused backscatter features to evaluate the performance. As presented in Fig. 22, we can successfully identify all these three movements with an accuracy of 93%.

Moreover, we have also perform a real warehousing task to validate the performance of our system. As shown in Fig. 24(a), the warehouse is divided into 7 different areas, and the size of each box has been marked in the



(a) Warehouse (b) Accuracy

Fig. 24. Performance of warehousing tasks



(a) Tag positions (b) Accuracy at different positions

Fig. 25. Impact of tag positions where we fix one tag on the base and place another tag to the moving link in turn.

Figure. In our experiment, we control the gripper to place a cube to these 7 boxes from the same position. As a result, complex motions, such as grasping, linearly moving and rotating, can be included at the same time. Fig. 24(b) shows that RobotScatter can successfully identify all these three movements with an accuracy higher than 92%, which indicate that our system can successfully distinguish the movements with small deviations.

5.4 Impact of Backscatter Tags

In this section, we evaluate the impact of backscatter tags, including tag positions and the number of tags.

5.4.1 Impact of Tag Positions. We first evaluate the impact when tags are placed in different positions of the robot arm. In particular, as presented in Fig. 25(a), we fix one tag on the base of the robot and attach another tag to the moving links in turn. At the same time, we perform single-joint operations, where the six joints are driven to sweep all the possible angles. After that, we capture the fused backscatter features to evaluate the performance of each joint in three different environments. The results as described in Fig. 25(b), we can successfully distinguish all the operations of the robot, where we achieve an average accuracy larger than 97%. Additionally, we have a performance large than 95%, even though the system is deployed in a clustered environment.

Besides single-joint movements, we also explore the performance of complex motions where tags are placed on different links as well. Specifically, as described in Fig. 23, we control the robot arm to perform complex and similar movements including MoveJ, MoveL and MoveP. Then, we fix one tag on the base and place another tag on the links from Link-a to link-f. Moreover, we also compare the performance where 6 tags are deployed on all the links. As presented in Fig. 26, experimental results show that RobotScatter achieves limited accuracy when the tag is located close to the base. The reason is that the links close to the base experience relatively small

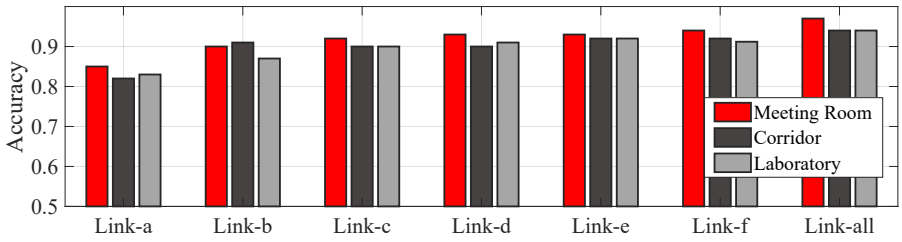
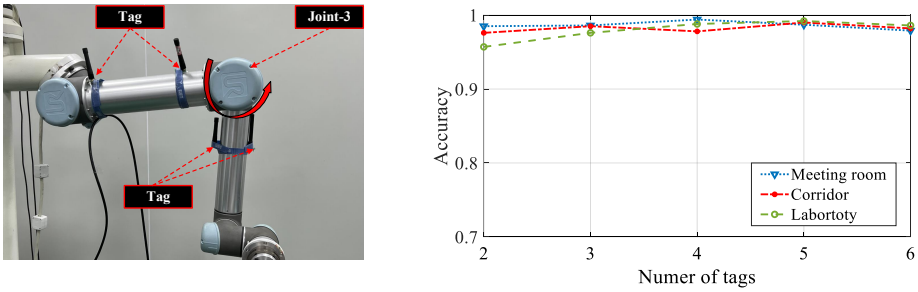


Fig. 26. Performance of complex and similar movements, when tags are placed different links.



(a) Tag deployment

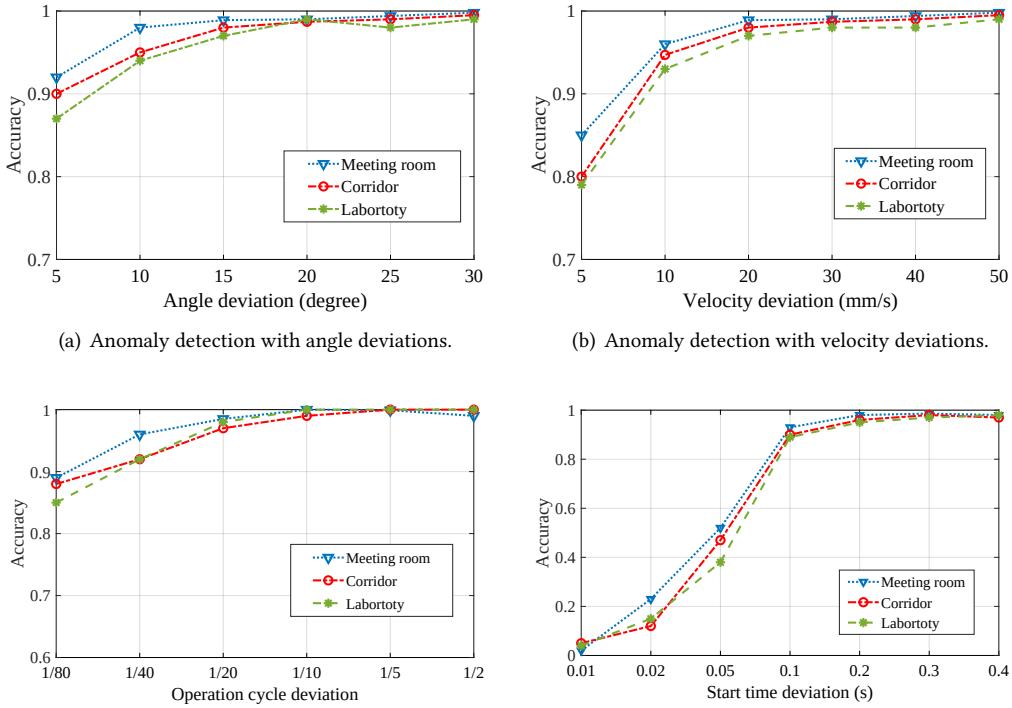
(b) Accuracy with different number of tags

Fig. 27. Impact of the number of tags.

rotating or similar angles, especially to the similar movements, which brings many difficulties in distinguishing these motions. However, deviations of these three movements increase when the tag is placed far from the base, and achieve higher a detection rate about 93% at Link-d to Link-f. In addition, when deploying multiple tags to all the moving links, RobotScatter can capture more details of each link and achieve higher accuracy.

According to the tag placement approaches descried above, RobotScatter can successfully identify the movements of the joints when the tags are deployed on moving links, especially away from the base. Therefore, we can place one tag on the base and another tag close to the gripper, since the first few joint movements will lead to the position changes to the gripper. Alternatively, if we have the priori knowledge of the robot movement and has enough tags, we can place the tags to the links that are going to move, from which the system can captures more details of each moving link.

5.4.2 Impact of The Number of Tags. We also evaluate the performance when we exploit the various number of backscatter tags attached to the robot. As shown in Fig. 27(a), we first attach one tag to a static link. Then, in the two-tag experiment, we deploy another tag to the link connected to Joint 3 and drive this joint to operate movement cycles from -180°-0°. As for the three-tag to six-tag experiments, we increase the number of the tags and attach them to different positions of the links. According to the analysis in Section 2.2, any two backscatter propagation signatures can be fused by the nonlinear effect, and thus we can still capture the features at second harmonics to secure the robot. As shown in Fig. 27(b), we can successfully identify the movements with different tag combinations.



(a) Anomaly detection with angle deviations.

(b) Anomaly detection with velocity deviations.

(c) Anomaly detection with operation cycle deviations.

(d) Anomaly detection with start time deviations.

Fig. 28. Performance of anomaly detection.

5.5 Anomaly Detection

We also evaluate the capability of anomaly detection of our system, where we consider the following typical anomalies including the movement angle, velocity, operation cycle and start time deviations.

5.5.1 Impact of Movement Angle Deviation. We first evaluate the performance when the robot suffers from a movement angle deviation. In particular, we first program only one joint to control the gripper to cyclically perform a rotation movement from -60° to 50°, where the radius of the gripper is 15cm. Then, we control it to deviate the normal movement from various degrees. As presented in Fig. 28(a), we have a detection rate less than 90%, when the angle deviation is by 5° (about 1.3cm). If we further increase the deviations, RobotScatter detect nearly 96% anomalies with a deviation of 10° (about 2.6cm). Compared with all the results, we successfully detect small angle deviations in all these three environments, which shows the robustness of our system and enables to help monitor many applications in the workshop.

5.5.2 Impact of Movement Velocity Deviation. We also verify the efficiency when the robot experiences small deviations of velocity. Specifically, we command the robot to perform movements with a velocity of 100mm/s. Then we control the robot to deviate the basic velocity from 5mm/s to 50mm/s, where we also conduct the robot to repeat the operation in different environments. As described in Fig. 28(b), we have an overall detection rate in all these three environments of 80% if the deviation is only 5mm/s. However, when we increase the velocity

deviation by 10mm/s, we can achieve an overall accuracy up to 94%. We further enlarge the deviation to 20mm/s, and successfully detect almost 99% anomalies. Therefore, our system presents the efficiency in velocity deviations.

5.5.3 Impact of Operation Cycle Deviation. We then evaluate the performance when the robot experiences different operation cycle deviations. In particular, we first attach two tags to the links, and drive only Joint-5 to perform the cyclic movements from -180° to 180°. As for operation cycle deviations, we conduct the joint to deviate the movement from 1/80-1/2 cycles, and then we extract the features to match the original movement. As described in Fig. 28(c), when the operation deviates 1/40 movement cycles, RobotScatter can achieve an average detection rate less than 90% for all these three environments. In contrast, when we increase the deviation by 1/20 movement cycles, RobotScatter can detect cycle deviations larger than 95%. The results show that RobotScatter can efficiently remove the anomaly of operation cycle deviations.

5.5.4 Impact of Start Time Deviation. Finally, we validate the performance when the robot operations have various start time offsets. In particular, similar to before, we also attach two tags to the links shared Joint-3, and then we control the joint to operate the movement from -60° to 50° with a velocity of 60° /s. For the start time offsets, we require the robot to repeat the operations but delay the movement from 0.01s- 0.4s, and then we extract features to detect anomalies. The results as shown in Fig. 28(d), we can detect less than 90% anomalies when the start time deviation is smaller than 0.1s. However, while increasing the offset higher than 0.2s, RobotScatter can correctly distinguish 96% anomalies, which shows the efficiency in detecting start time deviations.

6 DISCUSSION

As with many other schemes, the approach proposed here would fail to identify motions in some cases, and it still has some limitations that need to be further addressed.

First, the obvious limitation is the ability to detect small deviations from normal movements, where the ability to detect the deviations decreases as the similarity to the normal operations increases. However, RobotScatter can still help to secure many operations of warehousing, discard, and assembling tasks [35, 46], which require only limited operational precision. Besides, recent advances have also exploited a large number of cameras to provide high precision of 1cm [2]. In the future, we explore the fusion of backscatter propagation signatures and multiple spatially distributed cameras to achieve higher accuracy as well as adapt to illumination changes.

Second, another limitation could be the tag deployment and model training with some human force. In particular, the key insight of the RobotScatter is to exploit the nonlinear effect to fuse signatures caused by the robot movement. Therefore, one of the tags requires to be attached to the moving link of the robot arm so as to capture the robot's movement. It indicates that the operator requires to have the priori knowledge of the robot movements, so that she can label the movements to train the model and avoid attaching all the tags on the static links. However, since the robot operations are usually pre-defined, the operator can easily avoid this issue.

Finally, when the operator implements the proposed approach on the robot, some cost of the backscatter tag manufacture is needed. However, compared with the multiple camera-based system that costs about 10k dollars [2], RobotScatter requires the purchase of less than 5 dollars for each tag, which is much cheaper and affordable for the robot manufacturers and for personal use.

7 RELATED WORK

7.1 Robot Security

Anomaly detection has attracted much attention in recent years. Many types of research explore to analyze the joint angle to match the real data [1, 31]. However, these approaches cannot guarantee whether the received data is truly from the legitimate robot. To improve security, recent efforts introduce high-resolution sensors to monitor the movements, such cameras [33] and gyroscopes [4, 8]. While the visual information is easily affected by the

obstacles and light conditions, and the gyroscope requires to invade the system to install the sensor, Researchers also employ the power fingerprint in the robot's wire to improve the security [35]. However, it is designed for the wired network system but is not appropriate for a wireless and mobile robot. Therefore, building a reliable and non-invasive scheme to secure wireless robots is significantly important.

7.2 Wireless Tracking

RF-based mechanisms provide a complementary sensing capability to human and material sensing [16, 21, 47, 49, 52, 54] as well as robotic localization and tracking [12, 39, 53]. Traditional approaches to localize the target mainly rely on multiple APs or large antenna array assistance and achieve limited accuracy, which however is not appropriate for fine-grained robotic tasks [56]. Alternatively, a backscatter-aided approach has been explored to provide a low-cost tracking capability [50], while it supports only 2D localization, and is not applicable to an industrial robot. RFID-based approaches have been developed to improve 3D localization for fine-grained robotic tasks with high accuracy [29]. Unfortunately, a dedicated reader and signal design are required to deploy to implement this system. High-resolution mechanisms, such as mmWave [3, 57], have attracted much attention in indoor localization, but they cannot be well-adaptive to the robot's mobility, and needs to be equipped with expensive transceivers as well as super wide-band signals. To overcome this limitation, IMU and mmWave fusion [28] has been applied in robotic ego-motion estimation, whereas it requires invasive sensor installation. Furthermore, mechanical radar [34] also achieves high resolution while it is too heavy and less cost-effective to deploy on personal small-size robots like Roomba and iRobot Create. In contrast, RobotScatter secures robot anomalies with only several low-cost backscatter tags, and realize a reliable, adaptive, non-intrusive, low-cost, and ubiquitous industrial robot monitoring system.

7.3 Backscatter Communication

With the low-power performance, backscatter has been considered a promising technology in industrial area [17]. Different from traditional RFID communication, backscatter has well compatibility with various protocols with a dedicated circuit design, such as WiFi [9], LTE [11] and LoRa [41] signals. In our system, we design a low-cost tag similar as [27] to reflect the LoRa signal launched from the robot to secure the monitoring system.

7.4 Nonlinear Effect

Nonlinear effect caused by the amplifier and circuit is usually treated as being harmful to the communication performance. However, it also brings much attention to designing many applications. For instance, In [5], recent advances employ the nonlinear effect to design a cross-frequency communication for RFID, and some other studies also exploit it to convert wideband radar into a LoRa signal [22], or use it to mitigate interference for in-body backscatter [44]. In our study, we leverage the nonlinear effect to fuse the backscatter signatures and secure the industrial robot.

8 CONCLUSION

We present RobotScatter, a non-intrusive system that leverages the nonlinear effect of the circuit to fuse the backscatter propagation of tags that are attached to the robot to defend against active attacks and detect abnormal movements. RobotScatter secures industrial robots without using any expensive hardware or mounting sensors with invasive retrofit to the robots. We implement RobotScatter on commercial Universal Robot and iRobot Create with LoRa chip SX1276, USRP, and customized backscatter tags, and evaluate the performance in various environments. The experiment results achieve an overall accuracy of 97.7%, Besides, Robotscatter can detect up to 94% of anomalies against small movement deviations of 10mm/s in velocity, and 2.6cm in distance.

ACKNOWLEDGMENTS

This work was supported in part by the National Key R&D Program of China under Grant 2020YFB1806600, National Science Foundation of China with Grant 62071194, the Key R&D Program of Hubei Province of China under Grant No. 2021EHB002, Knowledge Innovation Program of Wuhan-Shuguang, RGC under Contract CERG 16203719, Contract 16204820, and Contract R8015.

REFERENCES

[1] Hamid Abdi and Saeid Nahavandi. 2010. Joint velocity redistribution for fault tolerant manipulators. In *2010 IEEE Conference on Robotics, Automation and Mechatronics*. IEEE, 492–497.

[2] Shaima Al Habsi, Maha Shehada, Marwah Abdoon, Ahmed Mashood, and Hassan Noura. 2015. Integration of a Vicon camera system for indoor flight of a Parrot AR Drone. In *2015 10th International Symposium on Mechatronics and its Applications (ISMA)*. IEEE, 1–6.

[3] Mohammed Aladsani, Ahmed Alkhateeb, and Georgios C Trichopoulos. 2019. Leveraging mmWave imaging and communications for simultaneous localization and mapping. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4539–4543.

[4] Michele Albano, Luis Lino Ferreira, Giovanni Di Orio, Pedro Maló, Godfried Webers, Erkki Jantunen, Iosu Gabilondo, Mikel Viguera, and Gregor Papa. 2020. Advanced sensor-based maintenance in real-world exemplary cases. *Automatika* 61, 4 (2020), 537–553.

[5] Zhenlin An, Qiongzheng Lin, and Lei Yang. 2018. Cross-frequency communication: Near-field identification of uhf rfids with wifi!. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 623–638.

[6] Emekcan Aras, Gowri Sankar Ramachandran, Piers Lawrence, and Danny Hughes. 2017. Exploring the security vulnerabilities of LoRa. In *2017 3rd IEEE International Conference on Cybernetics (CYBCONF)*. IEEE, 1–6.

[7] Tom Bartman and Kevin Carson. 2016. Securing communications for SCADA and critical industrial systems. In *2016 69th annual conference for protective relay engineers (CPRE)*. IEEE, 1–10.

[8] Christian Bayens, Tuan Le, Luis Garcia, Raheem Beyah, Mehdi Javanmard, and Saman Zonouz. 2017. See no evil, hear no evil, feel no evil, print no evil? malicious fill patterns detection in additive manufacturing. In *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 1181–1198.

[9] Dinesh Bharadia, Kiran Raj Joshi, Manikanta Kotaru, and Sachin Katti. 2015. Backfi: High throughput wifi backscatter. *ACM SIGCOMM Computer Communication Review* 45, 4 (2015), 283–296.

[10] Rita Chattopadhyay, Mruthunjaya Jay Chetty, Eric XiaozhongJi, Stephanie Cope, and Jeffrey E Davis. 2017. Real time remote monitoring and anomaly detection in industrial robots based on vibration signals, enabling large scale deployment of condition based maintenance. *International Journal of Nanotechnology & Nanomedicine* 2, 1 (2017), 1–4.

[11] Zicheng Chi, Xin Liu, Wei Wang, Yao Yao, and Ting Zhu. 2020. Leveraging ambient lte traffic for ubiquitous passive communication. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 172–185.

[12] Byoung-Suk Choi and Ju-Jang Lee. 2009. Mobile robot localization in indoor environment using RFID and sonar fusion system. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2039–2044.

[13] Keywhan Chung, Xiao Li, Peicheng Tang, Zeran Zhu, Zbigniew T Kalbarczyk, Ravishankar K Iyer, and Thenkurussi Kesavadas. 2019. Smart malware that uses leaked control data of robotic applications: The case of Raven-II surgical robots. In *22nd International Symposium on Research in Attacks, Intrusions and Defenses ({RAID} 2019)*. 337–351.

[14] Nicolas Falliere, Liam O Murchu, and Eric Chien. 2011. W32. stuxnet dossier. *White paper, symantec corp., security response* 5, 6 (2011), 29.

[15] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. 2019. Deep learning for time series classification: a review. *Data mining and knowledge discovery* 33, 4 (2019), 917–963.

[16] Chao Feng, Jie Xiong, Liqiong Chang, Fuwei Wang, Ju Wang, and Dingyi Fang. 2021. RF-Identity: Non-Intrusive Person Identification Based on Commodity RFID Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–23.

[17] Xiuzhen Guo, Longfei Shangguan, Yuan He, Jia Zhang, Haotian Jiang, Awais Ahmad Siddiqi, and Yunhao Liu. 2020. Aloba: Rethinking ON-OFF keying modulation for ambient lora backscatter. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 192–204.

[18] Nijat Hasanov and Ali Pourmohammad. 2020. Applications of LoRaWAN in SCADA Systems. In *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*. IEEE, 1–6.

[19] Elad Hoffer and Nir Ailon. 2015. Deep metric learning using triplet network. In *International workshop on similarity-based pattern recognition*. Springer, 84–92.

[20] Rachel Hornung, Holger Urbanek, Julian Klodmann, Christian Osendorfer, and Patrick Van Der Smagt. 2014. Model-free robot anomaly detection. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3676–3683.
[21] Anna Huang, Dong Wang, Run Zhao, and Qian Zhang. 2019. Au-id: Automatic user identification and authentication through the motions captured from sequential human activities using rfid. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 2 (2019), 1–26.
[22] Qianyi Huang, Zhiqing Luo, Jin Zhang, Wei Wang, and Qian Zhang. 2020. LoRadar: Enabling Concurrent Radar Sensing and LoRa Communication. *IEEE Transactions on Mobile Computing* (2020).
[23] Fábio Januário, Carolina Carvalho, Alberto Cardoso, and Paulo Gil. 2016. Security challenges in SCADA systems over Wireless Sensor and Actuator Networks. In *2016 8th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*. IEEE, 363–368.
[24] Xingbin Jiang, Michele Lora, and Sudipta Chattopadhyay. 2020. An experimental analysis of security vulnerabilities in industrial IoT devices. *ACM Transactions on Internet Technology (TOIT)* 20, 2 (2020), 1–24.
[25] Lawrence H Kim and Sean Follmer. 2017. Ubiswarm: Ubiquitous robotic interfaces and investigation of abstract motion as a display. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–20.
[26] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. 2015. Spotfi: Decimeter level localization using wifi. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*. 269–282.
[27] Vincent Liu, Aaron Parks, Vamsi Talla, Shyamnath Gollakota, David Wetherall, and Joshua R Smith. 2013. Ambient backscatter: Wireless communication out of thin air. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 39–50.
[28] Chris Xiaoxuan Lu, Muhamad Risqi U Saputra, Peijun Zhao, Yasin Almalioglu, Pedro PB de Gusmao, Changhao Chen, Ke Sun, Niki Trigoni, and Andrew Markham. 2020. milliEgo: single-chip mmWave radar aided egomotion estimation via deep sensor fusion. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 109–122.
[29] Zhihong Luo, Qiping Zhang, Yunfei Ma, Manish Singh, and Fadel Adib. 2019. 3D Backscatter Localization for {Fine-Grained} Robotics. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. 765–782.
[30] Melvin P Manuel and Kevin Daimi. 2021. Implementing cryptography in LoRa based communication devices for unmanned ground vehicle applications. *SN Applied Sciences* 3, 4 (2021), 1–14.
[31] Michael L McIntyre, Warren E Dixon, Darren M Dawson, and Ian D Walker. 2004. Fault detection and identification for robot manipulators. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, Vol. 5. IEEE, 4981–4986.
[32] Nader Mohamed, Jameela Al-Jaroodi, and Imad Jawhar. 2009. A review of middleware for networked robots. *International Journal of Computer Science and Network Security* 9, 5 (2009), 139–148.
[33] Asim Munawar, Phongtharin Vinayavekhin, and Giovanni De Magistris. 2017. Spatio-temporal anomaly detection for industrial robots through prediction in unsupervised feature space. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1017–1025.
[34] Yeong Sang Park, Young-Sik Shin, and Ayoung Kim. 2020. Pharao: Direct radar odometry using phase correlation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2617–2623.
[35] Hongyi Pu, Liang He, Chengcheng Zhao, David KY Yau, Peng Cheng, and Jiming Chen. 2020. Detecting replay attacks against industrial robots via power fingerprinting. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. 285–297.
[36] Transparency Market Research. 2021. Industrial Control Systems Security Solutions Market. https://www.transparencymarketresearch.com/industrial-control-systems-security-solutions-market.html. (2021).
[37] Universal Robots. 2009. Universal Robots e-Series User Manual. http://help.universal-robots.com/SW_5_11/UR5e/Content/_resources/PDFs/UR5e_User_Manual_en_Global.pdf. (2009).
[38] Jac Romme, Johan HC van den Heuvel, Guido Dolmans, G Selimis, Kathleen Philips, and Harmke de Groot. 2013. On remote RF-based orientation detection. In *2013 IEEE International Conference on Communications (ICC)*. IEEE, 4797–4801.
[39] Longfei Shangguan and Kyle Jamieson. 2016. The design and implementation of a mobile RFID tag sorting robot. In *Proceedings of the 14th annual international conference on mobile systems, applications, and services*. 31–42.
[40] Keith Stouffer, Joe Falco, Karen Scarfone, et al. 2011. Guide to industrial control systems (ICS) security. *NIST special publication* 800, 82 (2011), 16–16.
[41] Vamsi Talla, Mehrdad Hessar, Bryce Kellogg, Ali Najafi, Joshua R Smith, and Shyamnath Gollakota. 2017. Lora backscatter: Enabling the vision of ubiquitous connectivity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–24.
[42] Dennis Tkach, He Huang, and Todd A Kuiken. 2010. Study of stability of time-domain features for electromyographic pattern recognition. *Journal of neuroengineering and rehabilitation* 7, 1 (2010), 1–13.
[43] Deepak Vasisht, Swarun Kumar, and Dina Katabi. 2016. {Decimeter-Level} Localization with a Single {WiFi} Access Point. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. 165–178.
[44] Deepak Vasisht, Guo Zhang, Omid Abari, Hsiao-Ming Lu, Jacob Flanz, and Dina Katabi. 2018. In-body backscatter communication and localization. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*. 132–146.

[45] Davide Villa, Xinchao Song, Matthew Heim, and Liangshe Li. 2021. Internet of Robotic Things: Current Technologies, Applications, Challenges and Future Directions. *arXiv preprint arXiv:2101.06256* (2021).
[46] Luigi Villani and Joris De Schutter. 2016. Force control. In *Springer handbook of robotics*. Springer, 195–220.
[47] Xuanzhi Wang, Kai Niu, Jie Xiong, Bochong Qian, Zhiyun Yao, Tairong Lou, and Daqing Zhang. 2022. Placement Matters: Understanding the Effects of Device Placement for WiFi Sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1 (2022), 1–25.
[48] Anna Wojciechowska, Jeremy Frey, Esther Mandelblum, Yair Amichai-Hamburger, and Jessica R Cauchard. 2019. Designing drones: Factors and characteristics influencing the perception of flying robots. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–19.
[49] Chenshu Wu, Feng Zhang, Beibei Wang, and KJ Ray Liu. 2020. mSense: Towards mobile material sensing with a single millimeter-wave radio. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 3 (2020), 1–20.
[50] Ning Xiao, Panlong Yang, Xiang-Yang Li, Yanyong Zhang, Yubo Yan, and Hao Zhou. 2019. MilliBack: Real-time plug-n-play millimeter level tracking using wireless backscattering. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–23.
[51] Xiangyu Xu, Jiadi Yu, Yingying Chen, Qin Hua, Yanmin Zhu, Yi-Chao Chen, and Minglu Li. 2020. TouchPass: towards behavior-irrelevant on-touch user authentication on smartphones leveraging vibrations. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–13.
[52] Meng Xue, Yanjiao Chen, Xueluan Gong, Jian Zhang, and Chunkai Fan. 2022. Wet-Ra: Monitoring Diapers Wetness with Wireless Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 1–26.
[53] Shichao Yue, Hao He, Peng Cao, Kaiwen Zha, Masayuki Koizumi, and Dina Katabi. 2022. CornerRadar: RF-Based Indoor Localization Around Corners. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1 (2022), 1–24.
[54] Shichao Yue, Hao He, Hao Wang, Hariharan Rahul, and Dina Katabi. 2018. Extracting multi-person respiration from entangled RF signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 2 (2018), 1–22.
[55] Pengyu Zhang, Dinesh Bharadia, Kiran Joshi, and Sachin Katti. 2016. Hitchhike: Practical backscatter using commodity wifi. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*. 259–271.
[56] Xianan Zhang, Lieke Chen, Mingjie Feng, and Tao Jiang. 2022. Toward Reliable Non-Line-of-Sight Localization Using Multipath Reflections. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1 (2022), 1–25.
[57] Bingpeng Zhou, An Liu, and Vincent Lau. 2019. Successive localization and beamforming in 5G mmWave MIMO communication systems. *IEEE Transactions on Signal Processing* 67, 6 (2019), 1620–1635.