Cross-Domain WiFi Sensing with Channel State Information: A Survey

CHEN CHEN, School of Data Science and Engineering, South China Normal University
GANG ZHOU, Computer Science Department, William & Mary
YOUFANG LIN, School of Computer and Information Technology, Beijing Jiaotong University

The past years have witnessed the rapid conceptualization and development of wireless sensing based on **Channel State Information (CSI)** with commodity WiFi devices. Recent studies have demonstrated the vast potential of WiFi sensing in detection, recognition, and estimation applications. However, the widespread deployment of WiFi sensing systems still faces a significant challenge: how to ensure the sensing performance when exposing a pre-trained sensing system to new domains, such as new environments, different configurations, and unseen users, without data collection and system retraining. This survey provides a comprehensive review of recent research efforts on cross-domain WiFi Sensing. We first introduce the mathematical model of CSI and explore the impact of different domains on CSI. Then we present a general workflow of cross-domain WiFi sensing systems, which consists of signal processing and cross-domain sensing. Five cross-domain sensing algorithms, including domain-invariant feature extraction, virtual sample generation, transfer learning, few-shot learning and big data solution, are summarized to show how they achieve high sensing accuracy when encountering new domains. The advantages and limitations of each algorithm are also summarized and the performance comparison is made based on different applications. Finally, we discuss the remaining challenges to further promote the practical usability of cross-domain WiFi sensing systems.

CCS Concepts: • **General and reference** → **Surveys and overviews;**

Additional Key Words and Phrases: Cross-domain Wi-Fi sensing, channel state information, gesture recognition, activity recognition, motion detection, user identification, breathing rate estimation, human localization, human tracking

ACM Reference format:
Chen Chen, Gang Zhou, and Youfang Lin. 2023. Cross-Domain WiFi Sensing with Channel State Information: A Survey. *ACM Comput. Surv.* 55, 11, Article 231 (February 2023), 37 pages.
https://doi.org/10.1145/3570325

1 INTRODUCTION

WiFi sensing has recently drawn considerable attention due to the ubiquitous availability of WiFi and the advancement of wireless communication technologies, including **Multiple-Input**

Authors' addresses: C. Chen, School of Data Science and Engineering, South China Normal University, No. 55 West of Zhongshan Avenue, Tianhe District, Guangzhou, China, 510631; email: 17112088@bjtu.edu.cn; G. Zhou, Computer Science Department, William & Mary, 251 Jamestown Rd. Williamsburg, VA, USA, 23187-8795; email: gzhou@wm.edu; Y. Lin (corresponding author), School of Computer and Information Technology, Beijing Jiaotong University, No. 3 Shangyuancun Haidian District, Beijing, China, 100044; email: yflin@bjtu.edu.cn.

0360-0300/2023/02-ART231 $15.00
https://doi.org/10.1145/3570325

Multiple-Output (MIMO) and **Orthogonal Frequency-Division Multiplexing (OFDM)**. The employment of MIMO and OFDM techniques enables **commercial-off-the-shelf (COTS)** WiFi devices to obtain **Channel State Information (CSI)** from each transmit antenna to each receive antenna at each carrier frequency along multiple paths. The key idea behind WiFi sensing is that the pattern of CSI can be affected when a user (i.e., humans or other objects) moves in the wireless environment. The patterns of change exhibit distinct characteristics due to different movements, which can be used for a set of emerging detection, recognition, and estimation applications [46, 51, 95].

Recent progress in detection with WiFi has explored the detection of human events, such as falls [82, 92] and motion [18, 37], object detection [102], and human presence detection [3, 98]. The above works aim to solve binary classification problems using threshold-based detection methods or simple learning-based algorithms, e.g., one-class **support vector machine (SVM)**. On the other hand, for most recognition applications, such as activity recognition [88, 91], gesture recognition [1, 47], and human/object identification [8, 9], the number of classes that need to be recognized ranges from two to several hundred. Thus, to perform multi-class classification, almost all the recognition applications first extract features in the spatial/time/frequency domain and then apply machine learning algorithms, such as **k-nearest neighbors (kNN)**, **convolutional neural network (CNN)**, **long short-term memory (LSTM)**, and **gated recurrent neural network (GRU)**. Unlike detection/recognition applications aiming for classification problems, most estimation applications focus on human/object localization and tracking [55, 85], as well as breathing/heart rate estimation [41, 43], which only require the calculation of the quantity values of angle, distance, duration, and so on.

WiFi sensing is advantageous, as it reuses existing WiFi infrastructures, removes the need for wearable devices, and is not sensitive to lighting conditions. However, the widespread development and deployment of WiFi-based sensing systems still face three key challenges:

- WiFi signals and their features, e.g., amplitude, phase, and **Doppler Frequency Shift (DFS)**, are highly sensitive to the real-world environments, the configurations (i.e., locations and orientations with respect to the WiFi transmitter and receiver) of the performer, and the users varying in ages, genders, shapes, and habits. For brevity, we use the term "**domain**" to summarize the aforementioned factors. As a result, changing any one of the domains will inevitably cause various fluctuations in WiFi signals.
- When a new domain appears, extra efforts are required for data collection and ground truth labeling. The reason is that multipath wireless signals usually carry information specific to the domain. Then the sensing performance is not reliable if the system is not updated with data in the new domain. However, it is unfeasible to collect such data from innumerable domains.
- Current sensing algorithms, such as predefined models and machine learning algorithms, are trained in a controlled domain, termed as "**training domain**". As a result, they will typically not work effectively when applied to predict other users' movements in new environments and configurations. It is necessary to restart the training process with new data when encountering a new domain, which is termed as "**testing domain**".

The above challenges motivate one promising but challenging research area: cross-domain WiFi sensing. To avoid the labor-intensive and time-consuming process of collecting training data and retraining the pre-trained models or networks for cross-domain WiFi sensing, we further address the following four research questions in this paper:

- How do different environments, configurations, and users impact the wireless channel metrics mathematically?

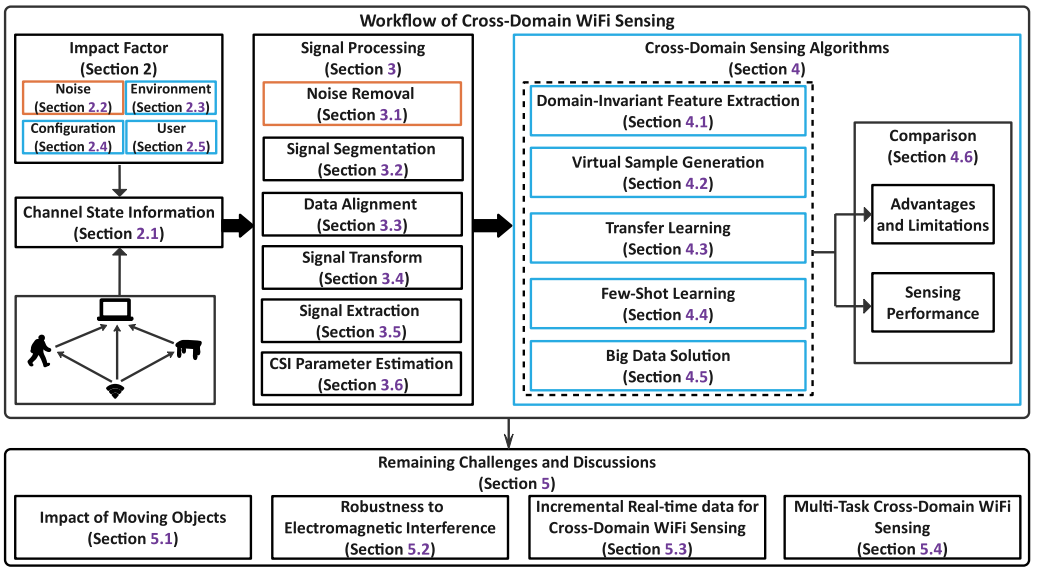

Fig. 1. Workflow of cross-domain WiFi sensing and paper organization.

- What solutions have been proposed to tackle the challenges mentioned above?
- How well do these solutions address the problem of cross-domain WiFi sensing?
- What future trends and open issues still exist in cross-domain WiFi sensing?

To answer the above questions, we first analyze how different domains affect the CSI parameters of the movements to be sensed, which is termed as "**target movement**" for brevity. Because the widely used signal processing approaches cannot restrain such effects, we further summarize existing cross-domain WiFi sensing algorithms from five perspectives: high-level domain-invariant feature extraction removing domain-specific components and preserving the details only relevant to the target movements, virtual sample generation reducing the efforts of data collection in testing domains, transfer learning alleviating the struggles of retraining pre-trained models or networks, few-shot learning making learning-based algorithms adaptable to new domains with very few labeled training data, and big data solution providing richer information from multiple dimensions. The pros and cons of the proposed five algorithms are also summarized and the sensing performance comparison is made based on different categories of WiFi sensing applications. Finally, this survey presents four remaining challenges, i.e., the impact of moving objects, robustness to electromagnetic interference, incremental real-time data for cross-domain WiFi sensing, and multi-task cross-domain WiFi sensing.

The overview of the survey is shown in Figure 1. We introduce the mathematical model of CSI and present four potential impact factors in Section 2. Typical signal processing approaches are listed in Section 3. Section 4 summarizes and compares the state-of-the-art algorithms for cross-domain WiFi sensing. We also discuss the remaining challenges in Section 5. Finally, existing WiFi sensing techniques and existing surveys are summarized in Section 6, and conclusions are drawn in Section 7. In summary, we make the following contributions:

- We explore the impact of environments, configurations, and users on the CSI parameters, including amplitude, phase, **time-of-flight (ToF)**, **angle-of-departure (AoD)**, **angle-of-arrival (AoA)**, and **Doppler frequency shift (DFS)**.
- We provide a comprehensive overview of the advancements made regarding cross-domain WiFi sensing, including removing domain-dependent components and extracting

Table 1. Impact Factors for Different Applications

Impact Factors	WiFi Sensing Applications
Slight Environmental Changes	Gesture Recognition [49, 70, 76, 114, 120]; Activity Recognition [4, 24, 80, 115]; Motion Detection [21]; User Identification [61, 118, 119]; Human Localization [6, 7, 31, 122, 126]; Human Tracking [55]; Object Identification [63]
Different Environments	Gesture Recognition [15, 20, 25, 30, 44, 57, 107, 120, 125, 128]; Activity Recognition [4, 23, 24, 60, 64–66, 79, 80, 115, 121]; Motion Detection [35, 69, 98, 112]; Fall Detection [50, 53, 93]; User Identification [27, 59, 99, 118]; Breathing Rate Estimation [38, 89, 90, 113]; Human Localization [117]; Human Tracking [34, 56, 100, 109, 127]; Object Identification [63]
Different Locations	Gesture Recognition [15, 20, 25, 30, 42, 52, 76, 114, 120, 125]; Activity Recognition [12, 13, 45, 121]; Motion Detection [112]; User Identification [61, 99]; Breathing Rate Estimation [38, 90]
Different Orientations	Gesture Recognition [15, 25, 30, 52, 76, 121, 125]; Motion Detection [112]; User Identification [59, 99, 118]; Breathing Rate Estimation [38, 90]; Human Tracking [34]
Different Users	Gesture Recognition [15, 20, 25, 33, 44, 70, 107, 114, 125]; Activity Recognition [4, 23, 24, 45, 64, 66, 79, 80, 103, 116, 121]; Motion Detection [35, 112]; Fall Detection [93]; User Identification [27, 59, 78]; Breathing Rate Estimation [113]; Human Localization [5, 6]; Human Tracking [34, 55, 56, 100, 109, 127]; Object Identification [63]

domain-invariant features, generating virtual samples for new testing domains instead of collecting new data, training transferrable networks rather than restarting the training process in new domains, learning similarity evaluation ability with few samples instead of calculating similarity directly, and improving the spatial diversity with big data solution.

- We summarize the advantages and limitations of the proposed five cross-domain WiFi sensing algorithms and further compare the sensing performance of the five algorithms based on nine categories of WiFi sensing applications: gesture recognition, activity recognition, motion detection, fall detection, user identification, breathing rate estimation, human localization, human tracking, and object identification.
- We present the remaining challenges and future trends to take cross-domain WiFi sensing to its next level of evolution and to bring it a step closer to real-world deployment.

2 IMPACT FACTORS

In this section, we introduce the basic concept of CSI and demonstrate how the noise and different domains impact the CSI parameters. Table 1 gives a summary of recent cross-domain WiFi sensing applications that attempt to handle various impact factors.

2.1 CSI Primer

CSI portrayed by COTS WiFi devices describes the multipath effects when the wireless signals propagate from the transmitter to the receiver through surrounding objects and humans. COTS WiFi devices typically consist of multiple antennae and support MIMO mode. A MIMO channel between each transmit-receive antenna pair comprises N_s OFDM subcarriers. Assuming the transmitter and the receiver are equipped with M and N antennae, respectively, we can obtain an $N_s * M * N$ matrix of CSI values, representing the amplitude attenuation and phase shift of multipath channels.

Mathematically, the CSI from the mth transmit antenna to the nth receive antenna on the ith subcarrier f_i at time t can be expressed as a superposition of signals from L paths:

$$\begin{aligned} H(f_i, m, n, t) &= |H(f_i, m, n, t)| e^{-j\angle H(f_i, m, n, t)} \\ &= \sum_{l=1}^{L} A_l(f_i, m, n, t) e^{-j2\pi \frac{d_l(t)}{\lambda_i}} e^{-j2\pi \frac{(m-1)\triangle_t \cos \psi_l(t)}{\lambda_i}} e^{-j2\pi \frac{(n-1)\triangle_r \cos \varphi_l(t)}{\lambda_i}}, \end{aligned} \tag{1}$$

where $A_l(f_i, m, n, t)$ is the attenuation factor, $d_l(t)$ is the distance of the lth path from the transmitter to the receiver at time t, and λ_i is the carrier wavelength corresponding to the subcarrier f_i. Then the ToF of the lth path at time t can be represented as $\tau_l(t) = \frac{d_l(t)}{\lambda_i}$. It is also well known that the attenuation factor $A_l(f_i, m, n, t)$ is inversely proportional to the square of the distance $d_l(t)$. In addition, the transmit antenna array and receive antenna array add the terms $2\pi\frac{(m-1)\triangle_t \cos\psi_l(t)}{\lambda_i}$ and $2\pi\frac{(n-1)\triangle_r \cos\varphi_l(t)}{\lambda_i}$ to the CSI phase shift, where $\triangle_t$ and $\triangle_r$ are the transmit and receive antenna separation distances, respectively, $\psi_l(t)$ is the AoD with respect to the transmit antenna array, and $\varphi_l(t)$ is the AoA with respect to the receive antenna array.

When a user is present in a static environment, multiple paths can be divided into static and dynamic paths. Static paths include the paths reflected from stationary objects, such as walls and furniture. The CSI corresponding to the static paths can be represented as a time-invariant constant, i.e., $H_s(f_i, m, n) = \sum_{l=1}^{L_s} A_l^s(f_i, m, n)e^{-j2\pi(f_i\tau_l + \frac{(m-1)\triangle_t \cos\psi_l}{\lambda_i} + \frac{(n-1)\triangle_r \cos\varphi_l}{\lambda_i})}$, where L_s is the number of static paths. Since the static paths do not vary with the user's movements, the ToF τ_l, the AoD ψ_l, and the AoA φ_l are time-invariant. The paths reflected by the user are time-variant and defined as a dynamic component $H_d(f_i, m, n, t)$, which could be further expressed as $\sum_{l=1}^{L_d} A_l^d(f_i, m, n, t)e^{-j2\pi(f_i\tau_l(t) + \frac{(m-1)\triangle_t \cos\psi_l(t)}{\lambda_i} + \frac{(n-1)\triangle_r \cos\varphi_l(t)}{\lambda_i})}$, where L_d is the number of dynamic paths. Movements of the users cause changes in the length of the dynamic paths, resulting in frequency shifts, which are called DFS. In summary, the aggregate CSI $H(f_i, m, n, t)$ (1) can be rewritten as $H_s(f_i, m, n) + H_d(f_i, m, n, t)$.

Note that Amplitude, phase, AoA, AoD, ToF, and DFS are widely used measurements for WiFi sensing. An overview of these WiFi measurements is provided in Section 2 of online supplementary material.

Line-of-sight (LoS) paths are regarded as static paths most of the time [34, 87, 90, 110]. In this case, the magnitude of the static component $|H_s(f_i, m, n)|$ is larger than that of the dynamic component $|H_d(f_i, m, n, t)|$. But there are also some research studies that exploit the LoS path for human sensing when the human body blocks the LoS path and attenuate the signal power [23, 62, 91, 115].

2.2 Impact of Noise

In practice, the CSI measurements usually suffer from amplitude noise, and three types of phase offset [105], including **Carrier Frequency Offset (CFO)**, **Sampling Time Offset (SFO)**, and **Packet Detection Delay (PDD)**. The impacts of amplitude noise and phase offsets are introduced in Section 3 of online supplementary material.

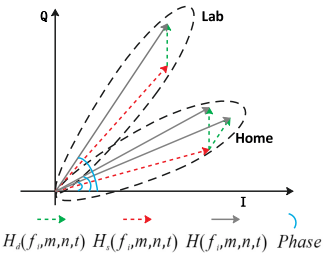

Fig. 2. CSI in complex plane

2.3 Impact of Environments

Figure 2 illustrates the relationship of the received CSI $H(f_i, m, n, t)$, static components $H_s(f_i, m, n)$ and dynamic components $H_d(f_i, m, n, t)$. It can be seen from Figure 2 that whether changing $H_s(f_i, m, n)$ or $H_d(f_i, m, n, t)$, the amplitude and phase of $H(f_i, m, n, t)$ will suffer from noticeable change. Consider a wireless signal propagating in a static environment, consisting of a set of reflectors, such as walls, doors, furniture, and so on. The collected CSI measurements usually carry the environmental information $H_s(f_i, m, n)$ unrelated to the status of the users. Thus, changing the layout of the static environment (i.e., slight environmental changes) or performing

sensing in a new environment (i.e., significant environmental changes) will affect the number of multiple paths L_s and the path lengths from the transmitter to the receiver through surrounding static reflectors. As shown in Figure 2, although such environmental changes do not contribute to the dynamic components of the CSI measurements, they still modify the amplitude and phase shift of the static components. Accordingly, the CSI features extracted from the superimposed WiFi signals, such as amplitude, phase, AOA, AOD, TOF, and DFS, also have significant variations.

In order to clearly show the impact of the static environments, we consider two evaluation environments: lab and home. The two environments are of different sizes and layouts. Thus, they have different multipath effects on the received WiFi signals. There are three instances of one gesture performed by the same user in the lab and home. It can be readily checked from Figure 3 that the amplitude patterns of the two instances are similar in the same environments, while they have a dramatic change in different settings.

2.4 Impact of Configurations

In the same environment, the amplitude and phase changes of the CSI measurements are determined by the dynamic path length changes, which depend on the location and orientation of the user with respect to the transmitter and receiver. When the user moves from one location to another, the lengths of the dynamic paths $d_l(t)$ caused by the movement will change, and the AoA and AoD also need to be reevaluated. Such variations modify the patterns of change observed in CSI amplitude. In addition, if the user turns his body to another orientation, the body may be an obstacle to the dynamic paths and further result in the amplitude plunge. Figure 4 illustrates such amplitude changes caused by configuration transformation.

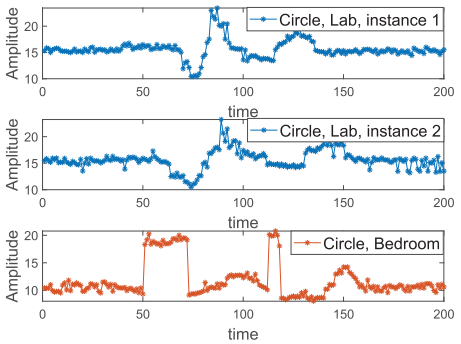

Fig. 3. Impact of environments

We also explore the impact of configurations on the phase shift. As shown in Figure 5, we assume that the original location of the user is in the perpendicular bisector of the transceivers with a LoS length of 2m. If the user moves 0.5m to the right, the reflection path length change is 8.94cm, which incurs a phase shift of 3.14π between two locations at the carrier frequency of 5.24GHz ($\lambda = 5.7$cm). Such phase shift makes the CSI phase in new configurations appear random.

2.5 Impact of Users

The shape of users, e.g., human height, is a vital impact factor for WiFi sensing. Because the user shape may affect the location of the target movements, even a few centimeters may lead to a change in CSI measurements. To illustrate this, as shown in Figure 6, we assume that one gesture is performed at the heights of 1m and 1.1m, respectively. Then the dynamic path length can be calculated by following simple geometry as 2.83m and 2.97m, respectively. Such dynamic path length difference leads to a phase shift of 4.91π at the carrier frequency of 5.24GHz ($\lambda = 5.7$cm).

Besides, to find out the common features of different users, the time series of CSI collected from different users are required to be one-to-one correspondence, especially for gesture/activity recognition. However, the behavior habits of the users, e.g., the speed of movement, can affect the start and the end of the gesture/activity (see Figure 7) and therefore affect the number of practical CSI measurements. For this reason, data segmentation and gesture/activity duration detection are required before feature extraction and system training.

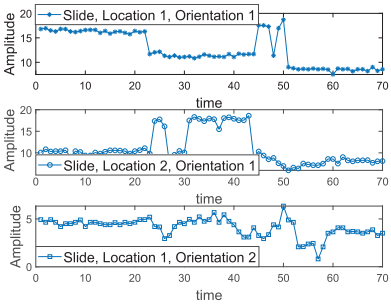

Fig. 4. Impact of configuration.

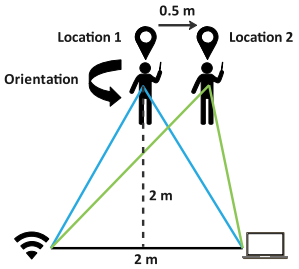

Fig. 5. Example for configuration transformation.

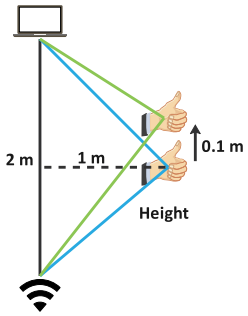

Fig. 6. Example for user transformation.

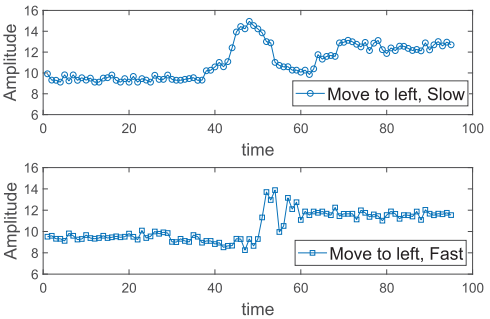

Fig. 7. Impact of speed.

3 SIGNAL PROCESSING

This section presents several signal processing techniques, including noise removal, signal segmentation, data alignment, signal transform, signal extraction, and CSI parameter estimation.

3.1 Noise Removal

Upon receiving the raw CSI, several noise removal approaches are proposed to suppress the impact of amplitude noise and phase offset. The amplitude noise and phase offset removal approaches are summarized in Sections 4.1 and 4.2 of online supplementary material, respectively.

3.2 Signal Segmentation

As Figure 7 shows, one never knows when the target movement will start and when it will be at the end. Signal segmentation is necessary to extract valid fragments corresponding to the target movement from raw CSI measurements. An overview of signal segmentation approaches is provided in Section 5 of online supplementary material.

3.3 Data Alignment

Data alignment techniques are used to unify different CSI series into uniform-length series to ensure the accurate time index of each instance. An overview of data alignment techniques is provided in Section 6 of online supplementary material.

3.4 Signal Transform

The frequency component is a good feature representation since different movements usually have different dominant frequencies. However, raw CSI measurements only show the amplitude and

phase changes over time without exhibiting the frequency components. Thus, signal transform methods are required for the time-frequency analysis of a time series of CSI measurements.

Fast Fourier Transform (FFT) is the most commonly used method to convert the CSI measurements from the time domain to their corresponding frequency-domain counterpart. FFT can also help to obtain the **Power Spectral Density (PSD)**, which has been used in respiration/heart rate estimation [38]. However, FFT loses the information of the time domain.

Short-time Fourier Transform (STFT) and **Discrete Wavelet Transform (DWT)** can capture features in both the time and frequency domains. STFT slides a window over a time series of CSI measurements. At each sliding step, it applies FFT to the CSI values covered by the window. Thus, the window size determines the tradeoff between frequency and time resolution of STFT. With a larger window size, STFT has a higher frequency resolution but lower time resolution, and vice versa. DWT is based on multi-resolution analysis, which provides high time resolution for movements with high frequencies and high-frequency resolution for movements with low frequencies. STFT and DWT have been widely used in activity/gesture recognition [4, 24, 30, 53, 59, 70, 76, 80, 118, 125].

Essentially, signal transform is a way to extract spectrum features that involve abundant target movement information. We can adapt the captured features as the input of sensing algorithms. Nevertheless, there still contains some uninterested components related to domains, which should be removed to guarantee the consistent performance of the sensing system in different domains.

3.5 Signal Extraction

Signal extraction is to remove redundant signals from raw CSI measurements and reserve efficient signals. It can be realized by using dimension reduction techniques such as PCA and signal decomposition algorithms, or selecting subcarriers with high sensitivity to target movements, or applying correlation methods over time or subcarriers. An overview of signal extraction techniques is provided in Section 7 of online supplementary material.

3.6 CSI Parameter Estimation

WiFi sensing could also be facilitated by the estimation of AoA, ToF, and DFS. AoA could determine the human direction because incident signals from different directions introduce different phase differences across an array of antennae. The accurate AoA estimations can be obtained by the **Multiple Signal Classification (MUSIC)** algorithm [5, 34], which computes the eigenvalue decomposition of the covariance matrix from CSI and calculates AoAs based on the steering vectors orthogonal to the eigenvectors. ToF is a measurement that could help estimate the distance or range of a person from the receiver. It is usually calculated with AoA jointly by applying MUSIC algorithm [28, 117].

AoA and ToF are two widely used measurements for CSI-based localization and tracking. While DFS is a popular measurement to detect the movements of different human body parts, because it describes the effect of human mobility on the observed frequency of WiFi signals. Specifically, the observed frequency will increase if the human moves towards the receiver. Otherwise, it will decrease. The amount of frequency increase or decrease Δf, i.e., DFS, can be expressed as $\pm f v/c$, where f is the transmitted frequency, v is the moving velocity, and c is the speed light. Most CSI-based sensing systems estimate DFS by performing time-frequency analysis with STFT for CSI series [4, 30, 70, 125].

CSI parameters, e.g., AoA, ToF, and DFS, could be adopted as the input of localization and tracking applications. However, one more step is required to remove the domain-dependent components, such as the static environmental components, before estimating the aforementioned CSI parameters for cross-domain localization and tracking applications.

4 CROSS-DOMAIN SENSING ALGORITHMS

The aforementioned signal processing techniques cannot solve the problem of cross-domain sensing because the features involved in the processed signals, e.g., denoised amplitude and phase, DFS, and estimated parameters, are still domain-dependent. Thus, when using these processed signal features as the input of the pre-trained sensing algorithms, especially the learning-based networks, one needs a lot of effort to collect training data in testing domains to retrain the networks. To address this problem, new algorithms are required to extract features without regard to domain-specific properties or to get rid of collecting new data and retraining the networks in testing domains. This section presents a summary and comparison of the five types of existing cross-domain sensing algorithms.

4.1 Domain-Invariant Feature Extraction

This subsection introduces how to remove domain-dependent components and screen the domain-invariant features out.

4.1.1 Domain-Dependent Component Removal. The most intuitive way to deal with the domain-dependent components is to remove them directly, including the reflections from the static objects in the surrounding environment and the signal components through longer multipath propagation. Afterward, only dynamic components of the target are retrieved.

There are several static component removal methods: (1) **Local Extreme Value Detection algorithm (LEVD)**. It first detects the local extreme values of CSI real and imaginary parts and then calculates the values of static components as the mean value of local maximum and minimum values. By integrating the updated real part and imaginary part, the CSI of static components is constructed. It has been used in 2D hand tracking [109, 127] and sign language recognition [120]; (2) Recursive algorithm leveraging EWMA. It estimates the static components $\hat{H}_s(t)$ and the dynamic components $\hat{H}_d(t)$ as $\gamma H(t) + (1-\gamma)\hat{H}_s(t-1)$ and $H(t) - \hat{H}_s(t)$, respectively. Because the static components retain constant, such an estimation method could preserve dynamic components while eliminating the static components [65, 90]. (3) LRSD. It could separate the original CSI measurements into low-rank parts indicating the environmental information and the sparse part containing the behavior information [42]. (4) AOA-TOF profile. Dynamic components always change while static components keep stable in the two-dimensional profile. (5) Conjugate multiplication of CSI measurements between two antennae. According to Widar 2.0 [56], the static components of two antennae could be removed by subtracting the average value from the conjugate multiplication. (6) Frequency-based filter. AFEE-MatNet [64] points out that, due to the limited speed and space of movements, most CSI variations caused by target movement are in a relatively low-frequency range, i.e., less than 100Hz. Therefore, the movement-related features could be retrained by discarding raw data in the high-frequency range. Additionally, the zero frequency component also should be removed because it is mainly environment-specific but movement-unrelated.

Slight environmental changes such as moving furniture or walking humans will create additional multipath. With the time-domain power delay profile observation, the created multipath's signal components usually have a longer time delay. The authors in [70] and [120] set a time delay as a baseline to mitigate the multipath effect. From the perspective of CSI change, a breathing rate estimation application [38] uses variance energy of CSI measurements across subcarriers as a metric for determining if the CSI measurements contain sleep events or environment changes because these movements lead to more significant changes in CSI measurements.

4.1.2 Common Feature Extraction. To further extract the domain-invariant feature, the common characteristics of the CSI variation across different domains are explored to model velocity or motion changes during the target movement.

The advantages of the movement velocity are two-fold. On the one hand, the velocity profiles of movements represent unique kinetic characteristics free from the changes in environments and configurations. On the other hand, the velocity varies even when the different users perform the same gestures while staying consistent over time when the same user performs the given gesture. The accurate estimate of the velocity changes during the movement could be computed by simply applying the equation $v = f\lambda/2$ to the frequency-time series derived from FFT or STFT [59, 100]. Following this idea, Widar 3.0 [125] and WiPose [24] further construct a body-coordinate velocity profile for the whole body of users because different body parts, including two hands, two arms, and the torso, move at different velocities when a person performs a gesture or an activity. While the works in [27, 118] directly estimate the velocity of different body parts, including torso contour speed, the torso speed, and the leg speed, with the upper contour torso reflection method [86] and percentile method [73]. Alternatively, the moving speed could be deduced from the path length change rate because the velocity could affect reflecting path lengths when a user moves [55]. However, the above approaches do not consider the multipath effect and thus may not work in NLoS conditions. GaitWay [99] presents a speed estimator for human motions in a rich-scattering indoor environment based on the relationship between the autocorrelation function of the received signals and the speed of the motions. Using such a speed estimator, DeFall [21] further derives the acceleration patterns from depicting different types of falls. Except for the movement speed, the path change speed v_{path} is depicted by the phase difference between the i^{th} CSI sample and the first sample $e^{-2\pi f v_{path}\Delta t_i/c}$ could be used to trace dynamic human reflected path [34, 35, 35]. If the path change speed v is non-zero, the phase difference change across time. On the contrary, zero speed means that there is no CSI change.

Regardless of moving speed, the independent characteristic of the target movement is worth exploring. Note that the movement itself could be divided into several atomic motions, each with different moving directions and time of duration. For example, drawing a rectangle contains four lines towards four different directions and two kinds of durations. Existing works [16, 30, 52, 114] have demonstrated that the switches and relative moving direction changes between two adjacent atomic motions are user-independent and configuration-independent. Based on another observation that the hand follows the same path just before and after a "turn" in a typical gesture, the number of "turn" events represents the number of repeated gesture segments, which is independent of location and environment [57]. Furthermore, the authors in [45] use a state machine to learn the transition probability between two adjacent activities.

The works above propose to use target movement-level domain-independent features to characterize the target movements. The authors in DPSense [15] offer a fresh perspective that the above features cannot entirely remove the domain dependency since the signal qualities inside each target are different and also depend on the domain. To this end, DPSense segment the CSI series and quantify the sensing quality of each signal segment. By maximizing the contribution of the high-quality signal segments and minimizing the impact of low-quality signal segments, the performance of gesture recognition across different environments, locations, orientations, and users is improved. Similar to DPSense, the authors in [90] select a group of sensitive signals based on signal energy detection and movement detection to mitigate the effect of anomalous respiration signals. WiHGR [49] adopts a sparse recovery method to find the main propagation paths disturbed by a human gesture.

There are also several statistical features for reference. DeMan [98] utilizes maximum eigenvalues of correlation matrices as an environment-independent feature. TensorBeat [89] extracts desired breathing signals with canonical polyadic decomposition. The authors in [50, 53, 112, 113] perform rigorous feature selection procedures from the signal energy viewpoint.

4.1.3 How to Apply the Domain-Invariant Features. With domain-invariant features, it is necessary to adopt extra predictors, including modeling-based and learning-based algorithms, to perform final classification, recognition, and estimation.

Modeling-based prediction algorithms include threshold-based detection, hypothesis testing, and peak detection, which are widely used in binary classification tasks, such as fall detection [21], motion detection [112], and human presence detection [35]. Multi-class classification tasks prefer to learning-based algorithms. SVM is a kind of shallow learning-based prediction algorithm. One-class SVM can distinguish falls and other activities [53, 82], while multi-class SVM helps to develop a walking direction agnostic gait-based user identification system [118]. Gesture and activity recognition tasks usually adopt deep learning-based algorithms. For example, Widar 3.0 [125], WiHF [30] and WiSign [120] use CNN, CNN-GRU, and **Hidden Markov Models (HMM)**, respectively, to perform domain-invariant gesture recognition. These networks could be further combined with state machine and reinforcement learning to achieve location and person-independent activity recognition [45]. In addition, multi-dimensional dynamic time warping is an alternative for matching predefined gestures and incoming gestures [52, 70]. Estimation applications usually separate the target reflection path independent of the environments and subjects by eliminating the static environmental paths. With the pure measurements belonging to the regular sleep events, one could monitor human breathing using peak detection without regard to the environmental changes and sleep postures [38, 90]. Besides, amplifying signals reflected by moving targets enable more precise estimations regardless of the environment/user/speed (e.g., AoA and walking direction) for human localization and tracking [34, 109, 117, 127].

4.2 Virtual Sample Generation

This subsection demonstrates how to reduce the efforts of collecting data in testing domains by producing virtual samples.

4.2.1 Translation Function. Via geometric model, WiAG [76] designs a translation function between training configuration A and testing configuration B. The translation function describes how to translate the amplitude difference between two consecutive time instants in configuration A to configuration B. Here it is assumed that the user moves at a constant speed v. Suppose $V(f, t_i, A)$ represents the value of $\|dH(f, m, n, t_i)\|$ when the user is in training configuration A, the amplitude difference in testing configuration B can be expressed as $V(f, t_i, B) = V(f, t_i, A)\frac{k^B D_{i,A}^2 [1+(\frac{v(t_i-t_{i-1})}{D_{i,A}})^2 - s\frac{v(t_i-t_{i-1})}{D_{i,A}} cos(\theta_{i,A}+\Theta_A)]}{k^A D_{i,B}^2 [1+(\frac{v(t_i-t_{i-1})}{D_{i,B}})^2 - s\frac{v(t_i-t_{i-1})}{D_{i,B}} cos(\theta_{i,B}+\Theta_B)]}$, where in configuration X, k^X is the proportionality constant for free-space propagation, $D_{i,X}$ is the travel distance of the wireless signal at time t_i, Θ_X is the initial user orientation, and $\theta_{i,X}$ is the moving direction.

When users change their location and orientation, the translation function automatically generates virtual samples for all gestures in all possible configurations to avoid collecting new training data. However, this method needs to estimate several indispensable parameters before generating virtual data, including the initial location and orientation, the proportionality constant, the moving directions in different time instants, and the moving speed for both training and testing configurations.

The above translation function focuses on CSI data transformation between different configurations, while the authors in [116] overcome activity inconsistency and subject-specific issues using eight CSI translation methods. Specifically, of which two methods attempt to proactively corrupt the training data set with random noise (e.g., dropout or additive Gaussian noise), four methods aim to modify the spectrograms with a set of deformation (e.g., time stretching, spectrum shifting, spectrum scaling, and frequency filtering), and the last two methods create additional training samples by mixing several samples together and altering the intensities of the principal components, respectively.

4.2.2 Generative Adversarial Networks. **Generative adversarial networks (GAN)** is an unsupervised generative deep learning architecture, which contains two sub-networks: a generator network G and a discriminator network D, shown in Figure 8. The generator network attempts to learn the distribution of the authentic samples and produce a virtual sample x_v from noise z generated using *a priori* distribution $p(z)$, i.e., $x_v = G(z)$. While the discriminator network takes the authentic sample x and the virtual sample x_v as inputs and estimates the probability that the input sample is drawn from real-data distribution p_{data}. Essentially, the entire network is a two-player min-max game and the overall GAN objective function can be expressed as $\min_G \max_D E_{x\sim p_{data}}[\log(D(x))] + E_{z\sim p_z}[\log(1 - D(G(z)))]$. The function reveals that the generator network attempts to make virtual samples closer to the authentic ones to confuse the discriminator network. In contrast, the discriminator network D strives to distinguish the authentic samples and the virtual samples.

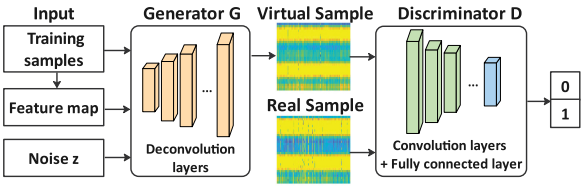

Fig. 8. Architecture of GAN.

During the training process, if the discriminator network D identifies the virtual samples drawn from the generator network, then the generator network will be optimized in the next iteration. Otherwise, the discriminator network will be optimized. The training ends when the two networks reach a dynamic equilibrium. Current WiFi sensing systems [33, 83, 103] usually substitute the random noise with the pre-processed training samples or the feature map generated from the raw training samples so that the training samples can be transferred into virtual samples with the style of the testing domain.

4.2.3 Autoencoder. **Autoencoder (AE)** is another learning-based architecture that could generate virtual samples. It contains two main components called encoder and decoder. As Figure 10 shows, the encoder first takes an input vector $\mathbf{x}$ of length l and then maps it to a hidden representation $\mathbf{h}$ of length l' [74]. $\mathbf{h}$ could be expressed as $f_\theta(\mathbf{x}) = s(\mathbf{W}\mathbf{x} + \mathbf{b})$, where $s(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function, and $\theta = \{\mathbf{W}, \mathbf{b}\}$ is the parameters to be optimized. Note that $\mathbf{h}$ is a low-dimension representation keeping the critical features of the input samples. Finally, the decoder reconstructs the input $\mathbf{z}$ of length l from $\mathbf{h}$, i.e., $\mathbf{z} = g_{\theta'}(\mathbf{h}) = s(\mathbf{W}'\mathbf{h} + \mathbf{b}')$, where $\theta' = \{\mathbf{W}', \mathbf{b}'\}$. The optimization of the parameters θ and θ' could be achieved by minimizing the average reconstruction error, i.e., $\theta*, \theta'* = \arg\min_{\theta,\theta'} \frac{1}{n}\Sigma_{i=1}^{n} L(\mathbf{x}, \mathbf{z}) = \arg\min_{\theta,\theta'} \frac{1}{n}\Sigma_{i=1}^{n} L(\mathbf{x}, g_{\theta'}(f_\theta(\mathbf{x})))$, where $L(\mathbf{x}, \mathbf{z})$ is the loss function such as the mean squared error $\|\mathbf{x} - \mathbf{z}\|^2$.

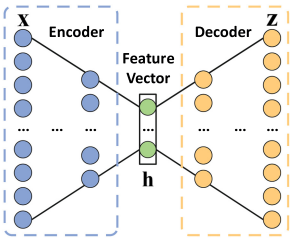

Fig. 9. Architecture of AE.

Cross-domain WiFi sensing benefits from AE in the following two ways. On the one hand, AE is fundamentally a feature-preserving technique that saves and reproduces the critical features related to the training domain of all CSI profiles. Then when the distorted samples pass through the trained AE, the features of distorted samples related to the preserved features are enhanced; meanwhile, the features irrelevant to the preserved features are weakened. For example, SCTRFL [122] trains two AEs to record the features of fingerprints related to the current environment. Thus, the trained AEs could be used to reproduce them in new environments by calibrating input measurements adaptively. In AutoFi [7], the features related to the environmental changes are mitigated or even removed, as these features are not encoded in **h**. In addition, Fido [5] and Fidora [6] use **variational autoencoder (VAE)** instead of AE to generate synthetic fingerprints for unseen users. As shown in Figure **??**, VAE is a more advanced form of AE designed to summarize the probability distribution of the input data from training users and then generate the synthetic samples from the distribution. On the other hand, AE is an unsupervised generative model, as it can self-supervise itself without requiring explicit data labels. It turns out in the aforementioned AE-based systems [5–7, 122], where both the training data and the testing data are unlabeled, eliminating the overhead of labeling data in different domains.

4.2.4 How to Apply the Virtual Samples. With virtual samples generated from the translation function or AE, extra predictors could be trained and then used to perform final classification, recognition, and estimation. For example, WiAG [76] uses kNN to perform gesture recognition and AutoFi [7] uses **Random Forest (RF)** model to perform human localization.

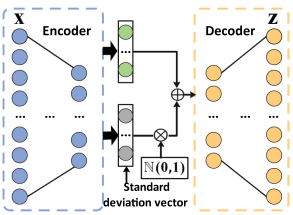

Fig. 10. Architecture of VAE

GANs contain both a discriminator and a generator. After training, the discriminator could be regarded as the predictor to perform the classification on the test domain [103] and the generator could help build up an augmented training dataset, which could be used to train the extra predictors, such as CNN and adaptive network [33, 83].

4.3 Transfer Learning

This subsection summarizes the solution to cross-domain sensing from the perspective of transfer learning [10, 54], which can learn transferable knowledge from a training domain to aid learning in a testing domain, enabling the direct deployment of the trained networks in new scenarios.

In cross-domain WiFi sensing systems, the transfer learning methods could be categorized into parameter transfer and feature-representation transfer. Parameter transfer tends to use very few test samples from the testing domain to fine-tune the network trained in the training domain. Feature-representation transfer aims to map the probability distribution of data in the training and testing domains into a shared space. We discuss the representative architectures in each category.

4.3.1 Parameter Transfer. Inspired by retraining the model trained on ImageNet for a new task, such as action recognition and object detection [26, 67], the authors in [4, 60, 69, 115] propose a similar transfer learning approach used in cross-domain WiFi sensing system that fine-tunes the networks pre-trained in the training domain in a new testing domain instead of retraining the entire network from the beginning. More specifically, the works in [60, 69, 115] first train deep learning networks, such as CNN-RNN and 7-layer **artificial neural network (ANN)**, whereas WiDrive [4] designs a shallow **Hidden Markov Models with Gaussian Mixture emissions Model (HMM-GMM)**. When applying the pre-trained deep learning networks to the new testing domains, the initial parameters of the network of the testing domains are the learned parameters

of the pre-trained networks. Then only particular layers are fine-tuned with a few samples from the new testing domains while other layers remain fixed. The reason is that the first few layers of the networks are likely to focus on abstracting the input WiFi channel metrics and are mainly independent of the model output. Besides, the HMM-GMM parameters could be updated via EM algorithm [11]. In summary, parameter transfer learning can reduce the training cost and speed up learning a new model for a new testing domain.

4.3.2 Feature Representation Transfer. By definition, feature representation transfer means that a "good" feature representation needs to be acquired to reduce the differences between the training domain and testing domain [10, 54]. This can be accomplished by mapping the extracted feature from both the training domain and testing domain to the shared space. **Domain Adaptation Network (DAN)** is a representative network that aims to learn a mapping between two different domains and perform accurate classifications for both domains.

The most widely used DAN in cross-domain WiFi sensing is adversarial-based DAN, i.e., **Domain Adversarial Neural Networks (DANN)** [14]. The basic architecture of DANN is shown in Figure 11. There are three main components: feature extractor G_f, activity predictor G_a, and domain discriminator G_d. Feature extractor G_f learns the feature representations Z for both the labeled samples from the training domain and the unlabeled samples from the testing domain. Based on the learned features, the activity predictor employs two fully connected networks followed by a softmax layer to obtain the predictions $\hat{y}$ on all input samples. Next, the domain discriminator is designed to identify the domain where the activities are conducted. To achieve this goal, the domain discriminator takes the concatenation of the feature representations Z and the predicted labels $\hat{y}$ as input and outputs the domain label distribution S. Like the GANs, DANN plays a min-max game, minimizing the loss of the activity predictor and maximizing the loss of the domain discriminator. During the training process, the feature extractor strives to learn domain-invariant features to cheat the domain discriminator and simultaneously boost the activity predictor's performance. When the training process ends, the samples from the training domain and samples from the testing domain are mapped into a shared space, where their probability distributions are unified. EI [23], GrossGR [33], JADA [128], and AdapLoc [126] employ such DANN to achieve cross-environment activity recognition, cross-target gesture recognition, cross-environment gesture recognition and localization in dynamic environments, respectively. CSIDA [25] and WiCar [80] further extends the single domain discriminator to a set of domain discriminators to avoid the discriminative structures mixing up for different domains. The authors in [61] add a user recognizer that is jointly optimized with the activity predictor. With this user recognizer, the extracted features characterize both identity and activity uniqueness. DANN could also be used in object identification, in which the activity predictor is replaced by an object recognizer [63].

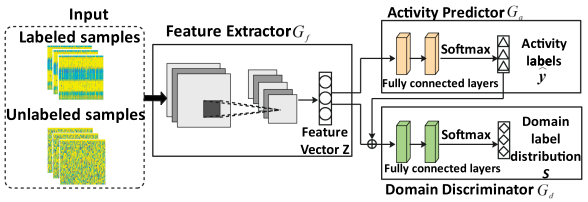

Fig. 11. Architecture of DANN.

Another famous DAN is reconstruction-based DAN using a joint classification-reconstruction structure, in which classification layers predict location labels and reconstruction layers reproduce the input data as similarly as possible. Fidora [6] provides robust Indoor Localization via this kind of DAN.

4.3.3 How to Apply the Transfer Learning Networks. Transfer learning enables deep learning networks to learn the classification ability in the training domain instead of the features of the

training data. Such networks could then employ this ability in new testing domains without any training. Specifically, networks with transferrable parameters could be updated with few labeled data from the testing domains, while networks with transferrable feature representations attempt to obscure the impact of the domain on the final prediction When encountering a new testing domain, the pre-trained deep learning networks could be used as predictors directly.

4.4 Few-Shot Learning

Few-shot learning has received much attention in recent years due to its ability to make learning-based algorithms adaptable to new domains with very few labeled training data. The crucial point of few-shot learning is to compare the similarity of data from different domains and classify them accordingly. There are four kinds of few-shot learning networks used in current cross-domain WiFi sensing tasks.

4.4.1 Deep Similarity Evaluation Network. Inspired by the transferrable knowledge learning ability of humans, a **deep similarity evaluation network (DSEN)** is developed to learn a transferable similarity evaluation ability from the training dataset and apply the learned knowledge to the new testing domain [20, 44, 83]. The proposed trained DSENs could classify samples in the testing set in the new domain by matching them with samples in the supporting dataset. It should be mentioned that the supporting dataset is composed of very few test samples for each class of activities or gestures, such as 1, 3, and 5.

In order to mimic the classification procedure in a new testing domain, episode-based training strategies are adopted by randomly sampling the training dataset to build up a training sample set and a training query set, which are regarded as support set and testing set, respectively (see Figure 12). For a sample x_i in support set and a sample $\hat{x}$ in testing set, DSEN first employs a CNN-based feature extraction network $\mathbf{F}_\xi$ to extract the support feature $\mathbf{F}_\xi(\mathbf{x_i})$ and the testing feature $\mathbf{F}_\xi(\hat{\mathbf{x}})$, where ξ denotes the parameter set. Afterwards, these two features are fed into the feature concatenation module and combined with the operator $\mathbf{C}(\mathbf{F}_\xi(\mathbf{x_i}), \mathbf{F}_\xi(\hat{\mathbf{x}}))$. The CNN-based similarity evaluation network $\mathbf{F}_\zeta$ deals with the concatenated features. It outputs a similarity score $s_{i,j}$ in the range of 0 to 1, where a larger value indicates high similarity between x_i and $\hat{x}$. Mathematically, the similarity score $s_{i,j}$ can be expressed as $\mathbf{F}_\zeta(\mathbf{C}(\mathbf{F}_\xi(\mathbf{x_i}), \mathbf{F}_\xi(\hat{\mathbf{x}})))$. During the training process, the network parameters ξ and ζ are constantly updated until the change is very small.

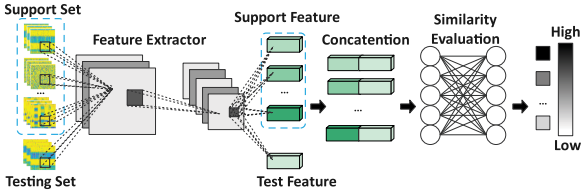

Fig. 12. Architecture of DSEN.

4.4.2 Matching Network. The works in [13, 65] use matching network (MatNet) [75]. As Figure 13 shows, MatNet is similar to DSEN in that they both take the support set and testing set as input and put deep neural networks (which are termed as embedding functions in MatNet) to extract the discriminative features. The difference comes in the similarity evaluation. MatNet introduces attention score in the form of soft-

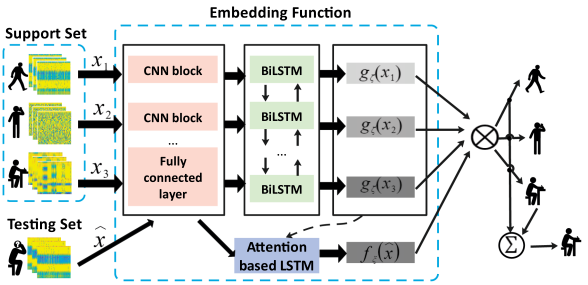

Fig. 13. Architecture of MatNet.

max over the cosine similarity, which is given by $a(\hat{x}, x_i) = \frac{e^{\cos\left(f_{\Lambda_1}(\hat{x}), g_{\Lambda_2}(x_i)\right)}}{\sum_{j=1}^{k} e^{\cos\left(f_{\Lambda_1}(\hat{x}), g_{\Lambda_2}(x_j)\right)}}$, where $\cos(\alpha, \beta)$ denotes the cosine similarity $\frac{\alpha \cdot \beta}{\|\alpha\| \|\beta\|}$, and f_{Λ_1} and g_{Λ_2} denote the embedding functions with parameter sets Λ_1 and Λ_2. Suppose the support set S is a mini-batch of training dataset with k samples, given by $\{(x_i, y_i)\}_{i=1}^{k}$, where (x, y) represents the sample-label pair. For a target sample $\hat{x}$ from the testing set, its label $\hat{y}$ could be estimated as $\sum_{i=1}^{k} a(\hat{x}, x_i) y_i$.

The key issue of the training process is to update the parameter sets with an episode-based training strategy, which is similar to the training process of DSEN. Afterward, the trained MatNet could be used in gesture/activity recognition in a new testing domain with only a few samples for each category of gestures or activities.

4.4.3 Siamese Network. Figure 14 illustrates a standard architecture of the **Siamese network (SiaNet)**, which is composed of two twin networks, that is, the two networks have the same structure and share the same parameters. They accept distinct inputs x_i and x_j collected from the same domain. After the feature extraction network, the feature map $f(x_i)$ and $f(x_j)$ are combined at the top to predict the label for a pair of inputs. More specifically, $y(x_i, x_j) = 1$ if $f(x_i)$ and $f(x_j)$ belong to the same class while $y(x_i, x_j) = 0$ if $f(x_i)$ and $f(x_j)$ belong to different classes. The Siamese network is trained by minimizing the loss function $L_p(x_i, x_j, \theta)$, i.e., $y(x_i, x_j) \max(0, b - m + D(f(x_i), f(x_j), \theta)) + (1 - y(x_i, x_j)) \max(0, b + m - D(f(x_i), f(x_j), \theta))$, where θ is the network parameters, two parameters b and m satisfy $0 < b < m$ and $D(f(x_i), f(x_j), \theta)$ denotes the metric distance between samples. The pairwise loss guarantees that the distance between the same class is restricted within $m - b$ while that from different classes exceeds $m + b$.

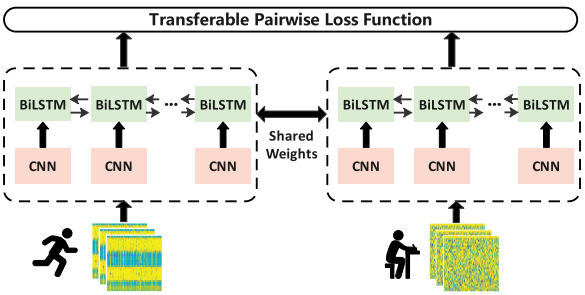

Fig. 14. Architecture of SiaNet.

However, when the two inputs are the gesture samples from two different domains, one needs to eliminate the different domains' impact on the Siamese network. To this end, the authors in [107] introduce transferable pairwise loss by using multiple kernel variants of maximum mean discrepancies (MK-MMD), which bears the unique property for measuring two distributions of samples from different domains [17]. Let X_t and X_s denote the testing domain and training domain, respectively. The pairwise loss function $L_p(x_i, x_j, \theta)$ can be extended as $L = L_p(x_i^s, x_j^T, \theta) + \lambda MMD^2(X_s, X_t)$, where $x_i^s \in X_s, x_j^t \in X_t$, $MMD^2(X_s, X_t)$ indicates the distance between the training domain and testing domain, and λ determines how strongly the model confuses the domains.

The Siamese network with the transferable pairwise loss minimizes the distance between the same class, maximizes the distance between different classes, and confuses the gap between different domains.

4.4.4 Prototypical Network. Prototypical Networks (PN) was first proposed in [68], and its basis is to learn an embedding function to construct an embedding space where the embedding representation of samples from the same class are clustered around a prototype, and those from different classes are separated.

Suppose there is a support set S and a testing set $\hat{S}$ sampled from the training dataset randomly, where $S = \{s_1, s_2, \ldots, s_C\}$ and $s_c = \{(x_i, y_i)\}_{i=1}^{K}$ is a set of labeled samples of the c-th category with class label y_i. As Figure 15 shows, the prototype of each category is the mean values of the

embedding representations of the corresponding support samples, i.e., $\mathbf{P}_c = \frac{1}{|s_c|} \sum_{(x_i, y_i) \in s_c} F_\Theta(x_i)$, where $F_\Theta(x_i)$ is the embedding function with a learnable parameter set Θ that maps input samples into the embedding space. Then given a sample $\hat{x}_i$ from the testing set, PN will compute the probability of $\hat{x}_i$ belonging to class c as $p(\hat{y}_i = c|\hat{x}_i) = \frac{exp(-d(F_\Theta(\hat{x}_i), \mathbf{P}_c))}{\sum_{c=1}^{C} exp(-d(F_\Theta(\hat{x}_i), \mathbf{P}_c))}$, where $d(F_\Theta(\hat{x}_i), \mathbf{P}_c)$ is the distance between $F_\Theta(\hat{x}_i)$ and $\mathbf{P}_c$. PN is trained by minimizing the negative log-probability $-logp(\hat{y}_i = c_t|\hat{x}_i)$, in which c_t is the true label of the sample $\hat{x}_i$.

When using PN in new domains, the prototypes of each class are computed using a small number of labeled samples, and the unlabeled samples are classified by calculating the distance between the unlabeled samples and the prototype. CAUTION [78] utilizes PN to recognize strangers and classify them as illegal intruders using an intruder threshold, while WiGr [121] further extends PN to Dual-path PN to achieve cross-domain gesture recognition.

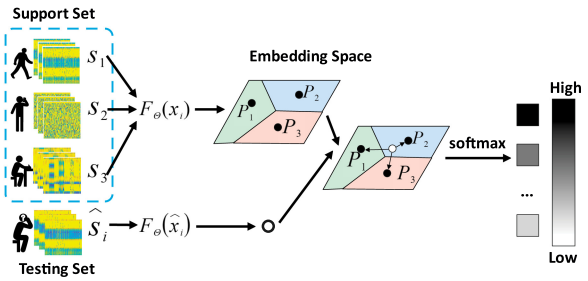

Fig. 15. Architecture of PN.

4.4.5 How to Apply the Few-Shot Learning Networks. Similar to transfer learning networks, the pre-trained few-shot learning networks could be used as predictors directly. The difference comes in the datasets. In the training phase, the embedding function is trained with a collection of labeled data from the source domain, while in the testing phase, the support set contains very few labeled data from the target domain and the testing set consists of unlabeled data from the target domain.

4.5 Big Data Solution

Spatial diversity can be improved by deploying multiple transmitters and receivers or extending the multiple antennae of modern WiFi devices to construct multiple separated links. This method is called big data solution. Big data solution benefits WiFi sensing applications from the following two aspects:

(1) Help extract domain-invariant features: profiling features from multiple spatial dimensions can be more complicated and offer mutual compensating information for robust sensing [32]. For instance, multiple receivers provide both the velocity information perpendicular to the LOS path and the velocity information parallel to LOS, which helps design a direction-independent gait recognition system [118]. With two receivers deploying in the orthogonal direction, the strong dependence between the random walking path and gait information is removed in Wi-PIGR [119]. Widar 3.0 [125] employs at least three links to set up a body-coordinate system, in which the velocity profiles are constructed around the human body rather than the environments and configurations. HandGest [114] utilizes two pairs of WiFi transceivers to build 2D Fresnel zones and further derive two distinct features, dynamic phase vector, and motion rotation variable, without regard to locations and orientations of users. For cross-domain human tracking applications, absolute trajectory independent of environments and configurations could be derived from CSI data received by multiple receivers [34, 109, 127].

(2) Offer more valuable data: it is known that radio signal propagation follows the theory of the Fresnel zones [81, 111], which consists of a series of concentric ellipses with two focal points on the pair of transceivers. The **first Fresnel zone (FFZ)** is the innermost ellipsoid

because most RF energy is transmitted through it. However, when the target moves inside the FFZ (i.e., dead zone), the CSI will experience a significant drop due to the block by the target. Therefore, to avoid the dead zone, WiSDAR [79] constructs multiple separated antenna pairs and discards the measurements of pairs with a dead zone. Besides, Wi-Sleep [40] deploys three pairs of transmitter and receiver to collaboratively monitor the respiration of a person with three different sleeping positions, respectively. Different pairs of transmitter and receiver work when the user sleeps at different sides, thus the respiration rate could be tracked reliably. WiTraj [100] leverages three or more receivers placed at different viewing angles to ensure reliable Doppler speed robust to user posture, orientation, and walking direction by combining the best two views.

It should be mentioned that big data solution cannot solve cross-domain WiFi sensing directly. However, it assists in domain-invariant feature extraction and valuable information selection, performing an indispensable role in the aforementioned cross-domain WiFi sensing applications.

4.6 Comparison of Different Cross-Domain Sensing Algorithms

The proposed five cross-domain WiFi sensing algorithms have their own advantages and limitations. For example, domain-invariant feature extraction avoids data collection and system retraining, but it needs a lot of effort for signal processing and feature engineering, which may require some prerequisites, such as the location and orientation of the users [24, 125]. In addition, extra predictors, including modeling-based and learning-based algorithms, are indispensable to final detection, recognition, and estimation [21, 24, 27, 30, 34, 35, 38, 42, 45, 52, 53, 59, 65, 70, 90, 109, 112, 118, 120, 125, 127]. Virtual sample generation, transfer learning, and few-shot learning do not need domain-invariant features, amplitude, phase, and DFS are enough to serve as the input feature. These two algorithms, therefore, require little or no effort for signal processing and feature engineering. However, they relieve but do not get rid of the pressure of data collection because they still need a few labeled or unlabeled samples in a new testing domain [4, 5, 13, 20, 23, 25, 33, 44, 60, 65, 69, 80, 103, 107, 115, 126, 128]. For final prediction, virtual samples generated from the translation function and AE need to be fed into extra modeling-based and learning-based predictors [5, 7, 76, 122], while the discriminator of GANs, transfer learning and few-shot learning networks could be used as the predictor directly [4, 13, 20, 23, 25, 33, 44, 60, 65, 69, 80, 103, 107, 115, 128]. Compared with the above four algorithms, big data solution plays an auxiliary role because it could help domain invariant feature extraction and provide more accessible information. In summary, we assess the pros and cons of different cross-domain WiFi sensing algorithms from the following five aspects: (1) efforts for feature engineering; (2) efforts for data collection and ground truth labeling; (3) requirements for extra predictors; (4) efforts for predictor retraining; and (5) system prerequisites. Table 2 lists the pros and cons of each algorithm.

We also present a survey of different applications, including gesture recognition, activity recognition, motion detection, fall detection, user identification, breathing rate estimation, human localization, human tracking, and object identification, using different signal processing techniques and cross-domain WiFi sensing algorithms, as shown in Table 3. For convenience, we provide the abbreviations used in Table 3 in the following. For signal processing, **noise removal, signal segmentation, data alignment, signal transform, signal extraction**, and **CSI parameter estimation** are represented by **NR, SS, DA, ST, SE**, and **CSIPE**, respectively. For cross-domain sensing algorithms, **DIFE** indicates **domain-invariant feature extraction**, **DDCR** indicates **domain-dependent component removal**, **CFE** indicates **common feature extraction**, **VSG** indicates **virtual sample generation**, **TL** indicates **transfer learning**, **FSL** indicates **few-shot**

Table 2. Pros and Cons of Different Cross-Domain Sensing Algorithms

Algorithms	Pros	Cons
Domain-Invariant Feature Extraction	(1) Need no training data collection and ground truth labeling in testing domains; (2) avoid retraining the predictors	(1) Need a lot of efforts for feature engineering; (2) need extra predictors; (3) may need some prerequisites, such as the location and orientation of the users
Virtual Sample Generation	(1) Need very little or no feature engineering; (2) reduce the efforts of training data collection and ground truth labeling in testing domains; (3) can use discriminator of GAN as predictor directly	(1) Need extra predictors with samples generated from translation function and AE; (2) need a few samples in testing domains to produce virtual samples; (3) may need some prerequisites, such as the location and orientation of the users
Transfer Learning	(1) Need very little or none feature engineering; (2) reduce the efforts of retraining the predictors; (3) reduce the efforts of training data collection and ground truth labeling in testing domains	(1) Need a few samples in testing domains to fine-tune learning parameters; (2) Need a few samples in testing domains to train the classification networks
Few-shot Learning	(1) Need very little or none feature engineering; (2) reduce the efforts of retraining the predictors; (3) reduce the efforts of training data collection and ground truth labeling in testing domains	Need a few samples in testing domains as support set
Big Data Solution	(1) Help extract domain-invariant features; (2) offer more valuable data	Cannot address the problem of cross-domain WiFi sensing directly unless there is special antenna or receiver placement topology

learning, and **BDS** stands for **big data solution**. **EMP** and **ELP** represent **extra modeling-based predictors** and **extra learning-based predictors**, respectively.

It can be seen from Table 3 that eight works [5–7, 66, 77, 78, 103, 126] leave out signal processing module. The reason is that they take raw amplitude, phase, or CSI as the input of learning-based cross-domain WiFi sensing algorithms, including GAN [77, 103], AE [5–7], transfer learning 686 and few-shot learning networks [66, 78, 126]. Learning-based cross-domain sensing networks are capable of extracting efficient features from noisy data, thus the efforts of signal processing and feature engineering of the above works are greatly reduced. For other works using signal processing techniques, the role of the signal processing module is to provide different types of input (e.g., denoised amplitude [23, 33, 44, 120], relative amplitude [76], phase difference [15, 16, 49, 107, 114, 128], and DFS profiles [52, 70, 79, 80]) for different cross-domain sensing algorithms. Especially for cross-domain WiFi sensing applications with domain-invariant feature extraction, signal processing is important to remove domain-dependent components or to model domain-invariant features, such as velocity [21, 35, 59, 99, 118, 125] and motion changes [30, 52, 55, 112]. Although signal processing and feature engineering introduce overhead, the processed features could give a boost to sensing accuracy in some cases [64, 125].

For cross-domain WiFi sensing algorithms, 38 papers use domain-invariant feature extraction, nine papers use virtual sample generation, 13 papers use transfer learning, nine papers use few-shot learning, and 27 papers use big data solution. Most of the papers using domain-invariant feature extraction and virtual sample generation reduce the efforts of retraining, but they need extra predictors, including modeling-based predictors (e.g., peak detection and threshold-based detection) and learning-based predictors (e.g., SVM, kNN, and CNN). Transfer learning and few-shot learning avoid extra predictors, but they require a few samples from testing domains. Big data solution is usually used in conjunction with other cross-domain WiFi sensing algorithms.

In addition, the comparison of the performance results of these cross-domain WiFi sensing applications is summarized in Table 4. For convenience, we provide the abbreviations used in

Table 3. Cross-Domain WiFi Sensing Applications

Reference	Signal Processing	Input	Cross-Domain Sensing Algorithms	Prerequisite
Gesture Recognition				
WiAG [76]	**NR**: Butterworth Filter; **SS**: CSI Difference-based Detection; **DA**: Extrapolation; **ST**: DWT	First order difference of CSI	**VSG**: Translation Function; **ELP**: kNN	Configuration estimation for both training and testing domains
Widar 3.0 [125]	**NR**: Low-Pass Filter, Conjugate Multiplication; **SS**: Splitting Scheme; **DA**: Normalization; **ST**: STFT	Body Velocity Profile	**DIFE-CFE**: Velocity; **BDS**: 1Tx, 6Rx; **ELP**: CNN	Configuration estimation for both training and testing domains
yang-2019 [107]	**NR**: CSI Ratio; **DA**: Interpolation	Phase difference	**FSL**: SiaNet	20% of data in testing domains
ML-DFGR [44]	**NR**: Low-Pass Filter	Denoised amplitude	**FSL**: DSEN	5 samples for each gesture in testing domains
WiHand [42]	**NR**: Hampel Filter, Butterworth Filter; **SS**: Deviation-based Detection; **DA**: Interpolation; **SE**: Signal Decomposition	Histogram of CSI	**DIFE-DDCR**: LRSD; **ELP**: SVM	N/A
WiHF [30]	**NR**: Band-Pass Filter, Conjugate Multiplication; **SS**: Splitting Scheme; **DA**: Normalization; **ST**: STFT; **SE**: PCA;	Motion change pattern	**DIFE-CFE**: motion change pattern; **BDS**: 1Tx, 2Rx; **ELP**: CNN-GRU	N/A
WiSign [120]	**NR**: Band-Pass Filter, Linear Regression; **SS**: Power-based Detection; **DA**: Interpolation; **SE**: PCA, Correlation Method	Denoised amplitude and phase	**DIFE-DDCR**: LEVD and time delay retention interval; **ELP**: HMM	N/A
WiFinger [70]	**NR**: Wavelet-based Filter, Average Filter, Linear Regression; **ST**: STFT; **SE**: Subcarrier Selection	DFS	**DIFE-DDCR**: time delay retention interval; **EMP**: MD-DTW	N/A
JADA [128]	**NR**: CSI Ratio	Phase difference	**TL**: DANN	Pre-trained target encoder
CrossGR [33]	**NR**: Butterworth Filter; **SE**: Subcarrier Selection	Denoised amplitude	**VSG**: GAN; **TL**: DANN	A small amount of unlabeled samples in testing domains
WiGR [20]	**NR**: Band-Pass Filter; CSI ratio	CSI ratio	**FSL**: DSEN	A small number of samples for each gesture in testing domains
CSIDA [25]	**NR**: Linear Regression; **ST**: STFT	DFS	**TL**: DANN; **BDS**: 1 Tx, 2 Rx	80% of data in testing domains
niu-2021 [52]	**NR**: Savitzky-Golay filter, CSI ratio; **SS**: Frequency-based detection	Gesture fragments and series of direction changes	**DIFE-CFE**: gesture fragments and series of direction changes; **BDS**: 1Tx, 2 Rx; **EMP**: Similarity Evaluation	N/A
WiGRep [57]	**SE**: TRRS	TRRS	**DIFE-CFE**: Number of repeating gesture segments; **EMP**: Peak detection	N/A
HandGest [114]	**NR**: Savitzky-Golay Filter; CSI Ratio	Phase difference	**DIFE-CFE**: Dynamic phase vector, Motion rotation variable; **BDS**: 2 Tx, 2 Rx; **ELP**: Hierarchical sensing framework	2D Fresnel zones with two pairs of WiFi transceivers
WiGesture [16]	**NR**: CSI Ratio; **SS**: CSI Difference-based Detection	Dynamic phase variations	**DIFE-CFE**: Motion Navigation Primitive; **BDS**: 1Tx, 2 Rx; **EMP**: Similarity evaluation	N/A
DPSense [15]	**NR**: Linear Regression; CSI Ratio	Dynamic phase variations	**DIFE-CFE**: Sensing ability; **BDS**: 1Tx, 3 Rx; **EMP**: WiGesture [16]	N/A
WiHGR [49]	**NR**: Phase Measurement Difference	Phase difference	**DIFE-CFE**: Sparse recovery; **ELP**: Modified attention based BiGRU network	N/A

(Continued)

Table 3. Continued

Reference	Signal Processing	Input	Cross-Domain Sensing Algorithms	Prerequisite
Activity Recognition				
CrossSense [115]	**NR**: Conjugate Multiplication; **DA**: Normalization; **ST**: DWT; **SE**: Correlation Method	Statistical, compression, spectrogram and transformed features	**TL**: Parameter Transfer	A small amount of samples in testing domains
EI [23]	**NR**: Hampel Filter; **SS**: Splitting Scheme; **DA**: Normalization; **SE**: Correlation Method	Denoised amplitude	**TL**: DANN	A small amount of samples in testing domains
WiCar [80]	**NR**: Low-Pass Filter, Conjugate Multiplication; **SS**: CSI Difference-based Detection; **ST**: STFT; **SE**: PCA	DFS	**TL**: DANN	A small amount of samples in testing domains
WiSDAR [79]	**NR**: Low-Pass Filter; **SS**: Power-based Detection; **SE**: PCA	DFS	**BDS**: 3 Tx, 3 Rx; **ELP**: CNN-LSTM	Pre-defined Antenna placement topology
CsiGAN [103]	N/A	Amplitude	**VSG**: GAN	A small amount of unlabeled samples in testing domains
sheng-2020 [60]	**NR**: Butterworth Filter; **SS**: CSI Difference-based Detection	Denoised amplitude	**TL**: Parameter Transfer	A small amount of samples in testing domains
WiDrive [4]	**NR**: Low-Pass Filter, Conjugate Multiplication; **SS**: Frequency-based Detection; **ST**: STFT; **SE**: PCA	DFS	**TL**: Parameter Transfer; **BDS**: 1 Tx, 2 Rx	A small amount of samples in testing domains
ma-2020 [45]	**SS**: Deviation-based Detection; **DA**: Normalization	Amplitude	**DIFE-CFE**: CNN+State Machine+Reinforce Learning; **ELP**: CNN+State Machine+Reinforce Learning	N/A
HAR-MN-EF [65]	**SE**: PCA	Denoised amplitude	**DIFE-DDCR**: EWMA; **FSL**: MatNet	A small amount of samples in testing domains
WiLISensing [13]	**NR**: Butterworth Filter	Denoised amplitude	**FSL**: MatNet	A small amount of samples in testing domains
zhang-2021 [116]	**NR**: Butterworth Filter; **SE**: PCA; **ST**:STFT	DFS	**VSG**: Translation function; **BDS**: 1Tx, 2 Rx; **ELP**: Dense-LSTM	N/A
MCBAR [77]	N/A	Amplitude	**VSG**: GAN; **BDS**: 1Tx, 2 Rx; **ELP**: CNN	A small amount of samples in testing domains
WiGr [121]	**NR**: Linear Regression; **DA**: Interpolation	Amplitude and denoised phase	**FSL**: PN	A small amount of samples in testing domains
MatNet-eCSI [66]	N/A	Amplitude and phase	**DIFE-DDCR**: Recursive algorithm leveraging EWMA; **FSL**: MatNet	A small amount of samples in testing domains
AFEE-MatNet [64]	**NR**: Conjugate Multiplication; **ST**: FFT;**SE**: PCA, Correlation Method	Frequency domain feature matrix and correlation feature	**DIFE-DDCR**: Frequency-based filter; **FSL**: MatNet	A small amount of samples in testing domains
Motion Detection				
WiDetect [112]	**SE**: Correlation Method	Motion statistic	**DIFE-CFE**: Motion Statistic; **EMP**: Threshold-based Detection	N/A
M-WiFi [69]	**NR**: CSI Ratio	Covariance matrix of CSI	**TL**: Parameter Transfer	A small amount of samples in testing domains
WiVit [35]	**NR**: Conjugate Multiplication	Path change speed	**DIFE-CFE**: Path change speed; **BDS**: 1 Tx, 4 Rx; **EMP**: Peak Detection	N/A
DeMan [98]	**NR**: Band-Pass Filter, Linear Regression; **SE**: Cross-Correlation	Denoised amplitude and phase	**DIFE-CFE**: Maximum eigenvalues of correlation matrices; **EMP**: Hypothesis test	N/A

(Continued)

Table 3. Continued

Reference	Signal Processing	Input	Cross-Domain Sensing Algorithms	Prerequisite
Fall Detection				
DeFall [21]	**SE**: Correlation Method	Speed pattern and acceleration pattern	**DIFE-CFE**: Speed and Acceleration Patterns; **EMP**: Threshold-based Detection	N/A
FallDefi [53]	**NR**: Wavelet-based Filter; **SS**: Frequency-based Detection; **DA**: Interpolation; **ST**: STFT; **SE**: PCA	Spectral entropy, fractal dimension, and event duration	**DIFE-CFE**: Spectral Entropy, Fractal Dimension and Event Duration; **BDS**: multiple Rx; **ELP**: SVM	N/A
FallViewer [93]	**NR**: Conjugate Multiplication; **ST**: DWT; **SE**: PCA; **SS**: Derivation-based detection	Statistic features from amplitude and phase	**DIFE-DDCR**: Conjugate multiplication; **ELP**: LIBSVM	N/A
nakamura-2022 [50]	**SS**: Splitting Scheme; **DA**: Interpolate, Normalization; **SE**: PCA; **ST**: STFT	Spectrogram	**DIFE-CFE**: Energy occurrence of the spectrogram; **ELP**: CNN	N/A
User Identification				
WiID [59]	**SS**: Splitting Scheme; **DA**: Normalization; **ST**: STFT; **SE**: PCA	Speed-time series	**DIFE-CFE**: Speed; **BDS**: multiple Tx and Rx; **ELP**: SVM	N/A
WiDigr [118]	**NR**: Band-Pass Filter; **SS**: Power-based Detection; **ST**: STFT; **SE**: PCA, Correlation Method	Speed	**DIFE-CFE**: Speed; **BDS**: 1 Tx and 2 Rx; **ELP**: SVM	N/A
korany-2020 [27]	**SS**: Splitting Scheme; **SE**: Correlation Method	10 feature distances	**DIFE-CFE**: 10 Spectrogram Features; **BDS**: multiple Rx; **ELP**: Network consisting of 1 hidden layer with 30 units	N/A
CAUTION [78]	N/A	CSI	**FSL**: PN; **EMP**: Threshold-based detection	A small amount of samples in testing domains
Wi-PIGR [119]	**NR**: Band-Pass Filter, Average Filter; **SS**: Derivation-based Detection, Splitting Scheme; **ST**: STFT; **SE**: PCA	Spectrogram	**BDS**: 2 Rx; **ELP**: CNN+LSTM	Two pairs of transceivers are placed vertically
GaitWay [99]	**SE**: Autocorrelation	Speed	**DIFE-CFE**: Speed calculated from the ACF of CSI ; **BDS**: 2 Rx; **ELP**: SVM	N/A
shi-2020 [61]	**NR**: Conjugate Multiplication; **SS**: Frequency-based Detection; **DA**: Normalization; **ST**: STFT	Denoised amplitude and relative amplitude	**TL**: DANN	Half of unlabeled samples in testing domains
Breathing Rate Estimation				
ResBeat [90]	**NR**: Average Filter, CSI Ratio	Denoised amplitude and phase difference	**DIFE-DDCR**: EWMA-based Filter; **DIFE-CFE**: Sensitive Signal Group; **BDS**: 1 Tx and 2 Rx; **EMP**: Peak Detection	N/A
liu-2015 [38]	**NR**: Hampel Filter, Average Filter; **SE**: Subcarrier Selection	Denoised amplitude	**DIFE-DDCR**: Energy-based Filter; **BDS**: 1 Tx and 2 Rx; **EMP**: Peak Detection	N/A
SMARS [113]	**SE**: Auto correlation	CSI power response	**DIFE-CFE**: Peak prominence, peak width, peak amplitude, motion interference ratio, peak location; **EMP**: Peak Detection	N/A

(Continued)

Table 3. Continued

Reference	Signal Processing	Input	Cross-Domain Sensing Algorithms	Prerequisite
TensorBeat [89]	**NR**: Hampel Filter; **SE**: Autocorrelation	Phase difference	**DIFE-CFE**: Tensor decomposition; **EMP**: Peak detection	N/A
Wi-Sleep [40]	**NR**: Hampel Filter, Wavelet-based Filter; **DA**: Interpolation; **SE**: Subcarrier selection	CSI	**BDS**: Three pairs of Tx-Rx; **EMP**: Peak detection	N/A
Human Localization				
FiDo [5]	N/A	Amplitude	**VSG**: VAE and AE; **ELP**: Pre-trained Classification-Reconstruction Network	A small amount of samples in testing domains
SCTRFL [122]	**NR**: CSI Ratio; **DA**: Normalization	Denoised amplitude and phase	**VSG**: AE; **EMP**: Similarity Evaluation	N/A
AdapLoc [126]	N/A	CSI fingerprints	**TL**: DANN	A small number of CSI fingerprints in testing domains
AutoFi [7]	N/A	Amplitude	**VSG**: AE; **ELP**: Random Forest model	N/A
DeFi [117]	**NR**: Linear Regression; **CSIPE**: AoA-ToF	AOA of the target reflection path	**DIFE-DDCR**: AOA-TOF Estimation; **BDS**: 1 Tx and 2 Rx; **EMP**: Posterior PDF-based Estimation	N/A
WiPose [24]	**NR**: Conjugate Multiplication; **SS**: Splitting Scheme; **ST**: STFT	3D velocity profile	**DIFE-CFE**: 3D Velocity Profile; **ELP**: CNN-LSTM	Input skeletal structure
DAFI [31]	**DA**: Normalization	Amplitude	**TL**: DANN	A small amount of unlabeled samples in testing domains
Fidora [6]	N/A	Amplitude	**VSG**: VAE; **TL**: DAN	N/A
Human Tracking				
IndoTrack [34]	**NR**: CSI Ratio; **CSIPE**: AoA	Doppler velocity and AoA spectrum	**DIFE-CFE**: Speed and AoA; **BDS**: multiple Rx; **EMP**: Joint Trajectory Estimation	N/A
Qgesture [109]	**NR**: Average Filter, CSI Ratio, Linear Regression; **SE**: Subcarrier Selection	CSI	**DIFE-DDCR**: LEVD; **BDS**: 2 Rx; **EMP**: Movement Detection and Distance Estimation	N/A
zhu-2017 [127]	**NR**: Average Filter, CSI Ratio	Dynamic path CSI	**DIFE-DDCR**: LEVD; **BDS**: 2 Rx; **EMP**: Movement Direction Estimation	N/A
WiTraj [100]	**NR**: Savitzky-Golay Filter; CSI Ratio; **SE**: Subcarrier Selection	Phase difference	**DIFE-CFE**: Doppler speed; **BDS**: 3 or more Rxs; **EMP**: 2D displacement reconstruction	Multiple receivers placed at different viewing angles
Widar [55]	**NR**: Butterworth Filter; **SE**: Subcarrier selection, Cross correlation; **ST**: PCA, STFT	Path length change rate	**DIFE-CFE**: Path length change rate; **BDS**: 1Tx, 2 Rx; **EMP**: Successive Tracking	Multiple links
Widar 2.0 [56]	**DIFE-DDCR**: Conjugate Multiplications; High-Pass Filter; **CPE**: AoA, ToF, DFS	AoA, ToF, DFS	**CR**: Conjugate multiplications; **EMP**: Geometric deduction based on AoA and ranges derived from ToF	N/A
Object Identification				
shi-2021 [63]	**NR**: Linear Regression; Conjugate Multiplications	Power ratio and 7 statistic features	**TL**: DANN; **BDS**: 1Tx, 2 Rx	Polarized directional antennas; Half of the data in the target dataset for domain adaptation

Table 4: **LEC, SEC, CL, CO**, and **CU** denote **large environmental changes, slight environmental changes, change in location, change in orientation**, and **change in user**, respectively.

5 REMAINING CHALLENGES AND DISCUSSIONS

Cross-domain WiFi sensing has been regarded as a crucial issue in the area of WiFi sensing [19, 36, 46, 94–96, 108, 129]. This survey gives a comprehensive summary and comparison of five types of cross-domain sensing algorithms, including domain-invariant feature extraction, virtual sample generation, transfer learning, few-shot learning, and big data solution. However, several crucial challenges remain to be handled to facilitate the development of novel applications and further advance this area.

5.1 Impact of Moving Objects

Current cross-environment WiFi sensing applications mainly deal with static environmental changes, including changing the layout of a static environment (i.e., slight environmental changes) and performing sensing in a new environment (i.e., large environmental changes). The static environmental changes could be easily eliminated with the approaches proposed in Section 4.1.2. Nevertheless, there still exists one problem: how to detect target movements when there are moving objects inside the same environment? The moving objects lead to simultaneous and background movements, such as human motion interference. Addressing this problem is challenging because the received signals are a mixture of signal reflections from the target user and intruders, whose motion trajectory is random, leading to difficulty in separating the contribution of target movement from the tangly received signals.

There are two potential solutions to deal with interference from moving objects. The first one is to generate virtual samples with progressed deep learning networks. For example, MCBAR [77] takes advantage of GAN to approximate the CSI data distribution with limited measured CSI data when the interference is caused by surrounding people and SCTRFL [122] adopts AE to calibrate amplitude and phase to localize the user when other people are moving randomly. Multi-user tracking is another feasible solution. MultiTrack [71] can recognize the activity of the target users when there are other users performing unknown background activities. The basic idea of Multitrack is to use multiple links to construct the power delay profile of multiple users and then segment the profile into single-user reflection profiles. We could also leverage the signals received by multiple receivers to detect AoAs of multiple users, then by projecting the total received signal to these AoAs, the tangly received signals could be separated [27].

5.2 Robustness to Electromagnetic Interference

WiFi signals typically operate in the 2.4GHz or 5GHz ISM radio band, which is also occupied by other devices (e.g., Bluetooth, Zigbee, and microwave ovens). These devices introduce **cross-technology interference (CTI)**, which could seriously disrupt the received CSI measurements and degrade the sensing performance [97]. When entering new domains, it is hard to count the interference devices and evaluate CTI in advance. Thus, current cross-domain WiFi sensing systems cannot work well.

The most intuitive approach to avoiding CTI is to find a clean frequency band without CTI. However, with the increase in the number and types of radio devices, it is nearly impossible to find such a frequency band. In particular, when encountering a new environment, one never knows what devices will interfere with the sensing performance. This implies that WiFi sensing systems must be robust to CTI. Work on this topic is rare except for [22, 123, 124]. CTI detection is performed in [123, 124] for further renovating the CSI measurement. On the other hand, the interference-independent phase component is leveraged because this component keeps invariant for CTI and

Table 4. Performance of Different Cross-Domain WiFi Sensing Applications

Reference	Task	Environmental Setup	Performance
		Gesture Recognition	
WiAG [76]	6 Gestures	1 Environment; 5 Locations; 5 Orientations; 10 Participants	**SEC**: 90%; **CO**: minimum accuracy 89.9%; **CL and CO**: minimum accuracy 90.6%
Widar 3.0 [125]	9 Gestures	3 Environments; 5 Locations; 5 Orientations; 16 Participants	**LEC**: average accuracy 92.4%; **CL**: average accuracy 89.7%; **CO**: average accuracy 82.6%; **CU**: average accuracy 88.9%
yang-2019 [107]	6 Gestures	2 Environments; 10 Participants	**LEC**: maximum accuracy 75.3%; **CU**: maximum accuracy 85.7%
ML-DFGR [44]	276 Sign language gestures; 26 Types of letters	Sign language: 2 Environments, 5 Participants; Letter: 2 Environments, 14 Participants	**LEC**: 98% for 50 types of gestures; 90% for 26 types of letters; **CU**: 99% for 50 gestures; 90% for 26 types of letters
WiHand [42]	5 Gestures	1 Environment; 9 Locations; 10 Participants	**CL**: average accuracy 93%
WiHF [30]	6 Gestures	3 Environments; 5 Locations; 5 Orientations; 6 Participants	**LEC**: above 90%; **CL**: above 85%; **CO**: 70%-90%
WiSign [120]	40 ASL words	2 Environements; 30 Participants	**SEC**: 79% for moved furniture and 76% for moving persons; **LEC**: 70%
WiFinger [70]	8 Finger gestures	4 Environments; 6 Participants	**SEC**: over 90%; **CU**: minimum 89%
JADA [128]	6 Gestures	2 Environments	**LEC**: Large room → small room: 87.8%; small room → large room: 90.3%
CrossGR [33]	15 Gestures	3 Environments; 60 Participants	**CU**: 87.4% and 82.6% when the number of gestures is 10 and 15
WiGR [20]	CSIDA [25]; SignFi [47]	CSIDA [25]; SignFi [47]	SignFi dataset: **LEC**: 98%; **CU**: 92%; CSIDA dataset: **LEC**:88%; **CU**::91%; **CL**: 90.8%
CSIDA [25]	6 Gestures	3 Environments; 5 Positions; 5 Orientations; 15 Participants	**CU**: 91.8%; **LEC**: 87.8%; **CL**: 92.5%; **CO**: 87.1%
niu-2021 [52]	8 Gestures	3 Environments, 5 Locations; 5 Orientations; 10 Participants	**CL**: more than 90%; **CO**: more than 90%
WiGRep [57]	50 Gestures	2 Environments, randomly chosen locations	**LEC**: 99.6% and 99.4% with false alarm rate of 5%
HandGest [114]	9 digits	3 Environments, 20 Participants	**LEC**: 95.73%, 95.21% and 96.02% for 3 environments; **SEC**: around 95% for 2 environments; **CL**: around 90%; **drawing speed**: 94.17% for low speed, 90.79% for high speed; **drawing size**: 95.35% for small size, 92.97% for large size
WiGesture [16]	6 basic gestures, 4 conducting gestures, ten digits	3 Environments, 4 Locations, 3 Orientations, 20 Participants	**LEC**: maximum accuracy: 94%, minimum accuracy: 86%; **CL**: maximum accuracy: over 95%, minimum accuracy: over 85%; **CO**: maximum accuracy: over 94%, minimum accuracy: over 88%; **CU**: average accuracy: 92.8%
DPSense [15]	WiGesture dataset [16]	WiGesture dataset [16]	**LEC**: over 95% for 4 environments; **CL**: over 95% for 5 locations; **CO**: over 90% for 3 orientations; **CU**: average accuracy: 96.4% over 12 users
WiHGR [49]	4 Gestures	2 layouts in 1 environment	**SEC**: 95.5%

(Continued)

Table 4. Continued

Reference	Task	Environmental Setup	Performance
		Activity Recognition	
CrossSense [115]	40 Activities	3 Environments; 5 Configurations; 100 Participants	**SEC**: above 85%; **LEC**: 90%;
EI [23]	6 Activities	6 Environments; 21 Participants	**LEC and CU**: maximum accuracy: 75%
WiCar [80]	8 In-car activities	4 Driving conditions; 4 Types of cars; 4 Participants	**Driving condition**: 80%; **Driving condition and car type**: 70%; **Driving condition, car type and user**: 58%
WiSDAR [79]	8 Activities	4 Environments; 6 Participants	**Environment with weak multipath effects**: average accuracy 96%; **Environment with strong multipath effects**: average accuracy around 80%; **User**: average accuracy over 85%
CsiGAN [103]	SignFi Data [47]; FallDeFi Data [53]	SignFi: 2 Environments, 5 Participants; FallDeFi: 4 Environments, 3 Participants	**CU**: SignFi Data: accuracy 84% with 10% unlabeled data; FallDeFi Data: accuracy 84% with 10% unlabeled data
sheng-2020 [60]	4 Activities	3 Environments	**LEC**: over 90% with 6 volunteer data involved fine-tuning
WiDrive [4]	7 In-car activities	2 Types of cars; 8 Participants; 4 Placements of WiFi devices	**WiFi device placement**: maximum accuracy 97%; minimum accuracy 87% ; **Car type**: maximum accuracy 81%; **CU**: maximum accuracy 90%
ma-2020 [45]	ma-2020: 7 activities; FallDeFi [53]; S.Yousefi-2017 [108]	ma-2020: 1 environments; 4 receivers; 7 users; FallDeFi [53]; S.Yousefi-2017 [108]	ma-2020: **CU**: average accuracy 93%; FallDeFi [53]: **CU**: 83%; S.Yousefi-2017 [108] **LEC, CU, CL and CO**: 80%
HAR-MN-EF [65]	6 Activities	3 Environments; 5 Users	**LEC**: 70.1%, 65.2%, and 60.8% for three indoor environments
WiLISensing [13]	4 Activities	2 Environments; 24 Locations; 6 Participants	**CL**: 91.11%
zhang-2021 [116]	10 Activities	5 Participants	**CU**: around 90%
MCBAR [77]	6 Activities	2 Environments, 5 Participants	**SEC**: Lab: 92.32% with 40% unlabeled data; Cubic Office: 91.11% with 40% unlabeled data
WiGr [121]	Widar3.0 dataset [125], ARIL dataset, CSIDA [25] dataset	Widar3.0 dataset [125], ARIL dataset, CSIDA [25] dataset	**LEC**: 95% and 83% in the four-shot testing on Widar3.0 and CSIDA datasets, respectively; **CL**: 92% and 75% in the four-shot testing on ARIL and Widar3.0 datasets, respectively; **CO**: 84% in the four-shot testing on Widar3.0 dataset; **CU**: 89% and over 90% in the four-shot testing on Widar3.0 and CSIDA datasets, respectively
MatNet-eCSI [66]	7 Activities	3 Environments, 5 Participants	**LEC**: over 80% in two environments in one-shot testing; **CU**: over 80% of five participants in one-shot testing
AFEE-MatNet [64]	6 Activities	7 Environments, 5 Participants	**LEC**: over 80% with no fewer than 4 source environments for training; **CU**: over 70% with 2 participants for training
		Motion Detection	
WiDetect [112]	Human motion	2 Environments: one with 10 rooms and another with 6 rooms and 4 areas	**LEC, CL, CO, and CU**: detection rate 99.68% with a zero false rate
M-WiFi [69]	Human presence	7 Environments	**LEC**: around 90% true detection and less than 15% false detection

(Continued)

Table 4. Continued

Reference	Task	Environmental Setup	Performance
WiVit [35]	Still or not still; Walk or other activities	3 Environments; 5 Users	Still or not still: **LEC**: minimum precision 97.8% (FNR1.34%); **CU**: minimum precision 97% (FNR 1.57%); Walk or other activities: **LEC**: minimum precision 94.9% (FNR 4.85%); **CU**: minimum precision 94.4% (FNR 5.21%)
DeMan [98]	Moving or Stationary	5 Environments	**LEC**: TP and TN: over 90% in three environments
Fall Detection			
DeFall [21]	Two kinds of falls	2 Environments	**SEC**: LOS: 97.1%; NLOS: 97.56%
FallDefi [53]	4 Types of falls	4 Environments; 3 Users	**LEC**: maximum accuracy 83%; **CU**:maximum accuracy 81%
FallViewer [93]	3 Types of falls and other non-fall activities	3 Environments; 3 Users	**LEC**: over 90%; **CU**:over 90%
nakamura-2022 [50]	fall and other 9 activities; FallDeFi dataset	2 Environments; 9 Users	**LEC**: maximum accuracy: 92%; maximum accuracy: 86% in FallDeFi dataset
User Identification			
WiID [59]	User identification	4 Enviroments; 15 Users	**3 Training environments and 1 testing environment**: minimum accuracy: 92%; **1 Training environments and 3 testing environments**: minimum accuracy 88.5%; **CU**: maximum accuracy 80%
WiDigr [118]	User identification	3 Environments; 5 Walking directions; 5 Orientations; 60 Users	**SEC**: 81.88%; **CO**: above 75% for both face and back to LOS when group size is 6; above 90% for both face and back to LOS when group size is 3
korany-2020 [27]	Multiple people identification	2 Environments and 13 Participants for training; 4 Environments and 6 Participants for testing	**LEC**: two people: average accuracy: 81%; three people: 83%
CAUTION [78]	Intruder Detection	2 Environments; 20 Participants	**CU**: over 85% with 20 samples of 6 15 legal users in both lab and cubic office; **dressing of users**: over 90% with 20 samples of 6 15 legal users in both lab and cubic office
Wi-PIGR [119]	user identification	3 Environments; 60 Participants	**SEC**: before: 93.49% (TPR), 7.43% (FPR), after: 93.06% (TPR), 7.08% (FPR); **Path Independent**: decrease slightly
GaitWay[99]	user identification	6 Environmental settings; 11 Participants	**LEC, CL, CO**: 90.4% for single user; **gait speed monitoring**: median 0.12 m/s and 90%tile 0.35 m/s error
shi-2020 [61]	user identification	2 Environment; 3 environmental settings for each environment; 4 locations; 15 Participants	**SEC**: 87.3% and 83.6% for 2 environments; **CL**: 91.3%, 84.5%, and 81.2% accuracies given 3 source locations
Breathing Rate Estimation			
ResBeat [90]	Breathing rate estimation	3 Environments; 8 Locations; 6 Orientations	**LEC**: 86.51%-93.04%; **CO**: success rate 87.61%-94.21%; **CL**: 80%-91.34%
liu-2015 [38]	Breathing rate estimation	2 Environments with 3 deployments of AP and WiFi device	**CL**: mean estimation error: lower than 0.4 *bpm*; **LEC**: mean estimation error: lowest error 0.15 *bpm* for both environments
SMARS [113]	Breathing rate estimation	6 Environments, 6 participants	**LEC, CU**: breathing detection ratio for six users: over 80% for both LOS and NLOS condition

(Continued)

Table 4. Continued

Reference	Task	Environmental Setup	Performance
TensorBeat [89]	Breathing rate estimation	3 Environments, 5 participants	**LEC**: maximum estimation error: less than 0.9bpm in through-wall scenario, around 0.7bpm in computer laboratory and long corridor scenarios
Wi-Sleep [40]	Breathing rate estimation	6 sleeping positions	**sleeping position**: Tx1-Rx2 performs well for two sleeping positions, Tx1-Rx1 performs well for the other three sleeping positions
Human Localization			
FiDo [5]	Localization	8 Locations; 9 Participants	**CU**: Average recall, precision and F1 scores: 84.6%, 85%, 0.845; **Height**: 81%-95%; **Weight**: above 80%
SCTRFL [122]	Localization	1 Environment; Sublet environments: 30 locations and moving human: 22 Locations	**SEC**: error percent: 3%; mean error: 1.48cm; **SEC-moving human**: error percent: 18%; mean error: 13.51cm
AdapLoc [126]	Localization	4 Environments	**SEC**: mean distance error: Lab: Rx location: 142cm; door win: 137cm
AutoFi [7]	Localization	1 Environment; 10 Locations	**SEC**: mean localization accuracy: opening windows: 84.9%; opening doors: 71.3%
DeFi [117]	Localization	2 Environments	**LEC**: mean localization error: 0.66m and 0.47m
WiPose [24]	Posture reconstruction	3 Different settings in 1 room; 10 Participants	**LEC**: Average joint localization errors: 61.2mm, 79.1mm and 87.7mm; **CU**: 85.8mm
DAFI [31]	Localization	2 environments, 17 locations, 3 participants	**SEC**: office: maximum average accuracy over all locations: 93.3%, minimum average accuracy: 83%; apartment: maximum average accuracy : 96%, minimum average accuracy: 71.6%;
Fidora [6]	Localization	1 environment with 4 deployments; 7 locations, 9 participants	**SEC**: maximum F1 score: 87.3%, minimum F1 score: 77.2%; **CU**: height: over 87.1% and 90% when height difference is bigger than 20cm and smaller than 5cm, respectively; weight: over 87.9% and 90% when height difference is bigger than 20kg and smaller than 5kg, respectively
Human Tracking			
IndoTrack [34]	Tracking predefined path	2 Environments; 3 Different levels of speeds; 5 Participants	**LEC**: median error: empty room: 33cm; meeting room: 37cm; consistent accuracies across different targets and speeds;
Qgesture [109]	2D hand tracking	3 Environments	**LEC**: distance measurement error: less than 5cm
zhu-2017 [127]	Tracking Hand Movement	3 Environments	**LEC**: Average errors ranged from 5 to 10 degrees for angles and 8 to 16cm for tracking
WiTraj [100]	Tracking 4 walking trajectories	4 Environments, 9 Participants	**LEC**: median tracking errors of 13cm, 36cm, and 44cm for outdoor environment, the corridor and conference room, respectively; **walking speed**: no impact
Widar [55]	Tracking human moving velocity and location	1 Environment with 2 deployments, 5 Participants	**SEC**: median tracking errors of 13cm, 36cm, and 44cm for outdoor environment, the corridor and conference room, respectively; **CU**: tracking error: around 50cm; **walking direction**: speed error: around 20cm/s; **walking distance**: tracking error: around 50cm

(Continued)

Table 4. Continued

Reference	Task	Environmental Setup	Performance
Widar 2.0 [56]	Tracking 3 walking trajectories	3 Environments, 6 Participants	**LEC**: average localization errors of 63cm, 40cm and 50cm for 3 environments; **CU**: average localization error around 50cm for 6 users
Object Identification			
shi-2021 [63]	14 objects of 4 different materials	4 Environments, 6 different furniture settings, 4 different bags	**SEC**: average accuracy: around 80%; **LEC**: average accuracy: around 79.1%; **different bags**: average accuracy: over 83%

contains activity information [22]. How to remove CTI at the receiver remains a considerable problem.

5.3 Incremental Real-time Data for Cross-Domain WiFi Sensing

As Table 3 shows, virtual sample generation and transfer learning have addressed the challenge of cross-domain WiFi sensing by reducing the efforts of training data collection and predictor retraining in new domains. However, they still require a number of new domain data to be available upfront to help generate virtual samples or fine-tune the transfer learning networks. In fact, for real applications, new domain data always arrive sequentially rather than being collected and labeled in advance. Consequently, future cross-domain WiFi sensing needs to be performed continuously and incrementally in real-time with few current observations. It means that the learning network must update its state incrementally based on current observations at short notice. Additionally, it is necessary to forget old information and adapt to new information.

To overcome the challenges above, DeepCount [39] adopts the deep learning model with online learning [48], which, in turn, makes the model evolve over time. The opportunity to take advantage of self-learning [72] and active learning [58] can be further exploited in the near future.

5.4 Multi-Task Cross-Domain WiFi Sensing

Current cross-domain WiFi sensing applications using transfer learning accomplish the sensing tasks in different domains, while the sensing task is always single. For example, EI [23] trains a DANN to classify six various activities, and WiCar [80] trains another DANN to recognize eight different in-car activities across different domains. If they exchange the sensing tasks, the sensing performance will degrade significantly. As a result, we need to train different models corresponding to different sensing tasks, which is obviously unrealistic. It will be more efficient if the learning network is transferrable between different tasks.

At present, a few research works are training general networks for multiple tasks. For example, OneFi [104] leverages both few-shot learning and virtual sample generation to recognize unseen gestures, while CrossSense [115] adopts a mixture-of-experts approach to capture the mapping from diverse WiFi inputs to the desired outputs with different specialized experts. Further exploration needs to be made along with this direction.

6 RELATED WORK

6.1 Existing WiFi Sensing Techniques

In general, the current WiFi sensing procedure consists of signal processing and sensing algorithms [46]. Several signal processing techniques, including noise removal, signal segmentation, data alignment, signal extraction, signal transformation, and CSI parameter estimation, are summarized in Section 3, which are appropriate for both intra-domain and cross-domain WiFi sensing

because they aim to acquire more pure features. However, as Section 3 shows, these processed signals are still domain-variant and cannot meet the demand of cross-domain WiFi sensing.

Given the sanitized signals of interest from the signal processing module, sensing algorithms are required to achieve different sensing goals. Current sensing algorithms include modeling-based and learning-based algorithms. The former is based on the geometric properties to formulate the trilateration [2] and Fresnel zone models [85], or on the empirical measurements and probability functions of wireless channels to formulate a statistical model [106]. Modeling-based algorithms are more suitable for estimation applications, and thus need very accurate measurements along with a lot of signal processing. Furthermore, these models are heavily influenced by domains, especially environments and configurations. Thus, these models are usually not reusable for new domains unless utilizing multiple links or extracting high-level domain-invariant features. Learning-based algorithms try to learn the mapping function using training samples of CSI measurements and the corresponding ground truth labels. However, both shallow learning-based algorithms (e.g., Decision Tree, HMM, kNN, and SVM) and deep learning-based algorithms (e.g., CNN, DNN, and RNN) need a lot of effort for training data collection and ground truth labeling in different domains, and lack effectiveness in new domains without retraining [46]. In summary, existing modeling-based and learning-based algorithms are inapplicable to cross-domain WiFi sensing.

6.2 Existing Surveys for WiFi Sensing

There are some survey articles published in recent years on WiFi sensing, including detection, recognition, and estimation applications. In [95], the authors explore the typical CSI models, such as the AoA model, Fresnel model, and CSI-speed model. This survey provides a brief introduction to modeling-based sensing applications, including localization and tracking, human activity recognition, and human respiration detection. It points out their pros and cons, respectively. Three survey papers [94], [101], and [129] focus on both modeling-based and pattern-based approaches utilized in human sensing systems. These papers attempt to demonstrate the key difference between these two approaches, that is, pattern-based approaches require the unique and consistent relations between CSI variation patterns and certain human activities, while modeling-based approaches need to derive a model that can describe the physical law mathematically between the received CSI and the certain human activities. The other three papers [29, 51, 108] focus on the deep learning-based WiFi sensing approaches with their specific emphasis, including evolution and performance improvement from machine learning to deep learning techniques [108], deep learning techniques used in each module of the architecture of the WiFi sensing system [29], and deep learning-based WiFi sensing applications [51]. The survey conducted by Wang et al. [96] investigates human behavior recognition from three aspects: modeling-based, pattern-based, and deep learning-based approaches. He et al. [19] give a summary of four categories of human activity sensing applications, including intrusion detection and occupancy monitoring, activity and gesture recognition, vital signs monitoring, and user identification and localization. There are also some papers [36, 46, 84] studying the existing WiFi sensing systems from the perspective of their architecture in terms of signal collection, signal processing, feature extraction, algorithm, and so on.

Although the survey papers mentioned above have demonstrated the various techniques and the powerful capability of WiFi-based sensing systems in serving a broad array of applications, none of them provides a review of cross-domain WiFi sensing in this field of research. As summarized in Table 5, most of the above survey papers raise the problem of cross-domain WiFi sensing without solutions when discussing the current challenges and future directions of WiFi-based sensing systems [19, 36, 46, 94–96, 108, 129]. With regard to the solutions to cross-domain issues, The authors in [29, 51] only present several deep learning approaches, e.g., AE and adversarial networks, and

Table 5. Summary of Related Surveys on WiFi Sensing

Reference	Application Scope	Topic Focus	Cross-Domain WiFi Sensing	
			Challenge	Solutions
Wang-2021 [95]	Human localization and tracking, behavior recognition and respiration detection	Modeling-based human sensing approaches: models, applications, and advantages	Mention	No mention
Wu-2017 [101]	Human sensing	Pattern-based and modeling-based approaches	No mention	No mention
Wang-2018 [94]	Human behavior recognition	Pattern-based and modeling-based approaches for signal, action, and activity detection	Mention	No mention
Zou-2017 [129]	Gesture and activity recognition and motion tracking	Data-driven and modeling-based approaches	Mention	No mention
Yousefi-2017 [108]	Human behavior recognition	Evolution and performance improvement from machine learning to deep learning techniques	Mention	No mention
Li-2021 [29]	Detection and recognition, numerical analysis, and image generation	Deep learning used in signal processing, high-level feature and sensing model formulation; comparison between traditional approaches and deep learning approaches	Mention	Deep learning techniques for domain-independent feature extraction
Nirmal-2021 [51]	Human sensing, including gesture, activity, gait, respiration, and localization	A survey of the deep learning-based RF sensing and a review of 20 released public datasets	Mention	Autoencoder and adversarial networks
Wang-2019 [96]	Human behavior recognition	Pattern-based, modeling-based and deep learning-based recognition techniques and applications	Mention	No mention
He-2020 [19]	Human sensing, detection and recognition	A survey of nine different applications related to sensing, recognition, and detection using CSI	Mention	No mention
Ma-2019 [46]	Detection, recognition and estimation applications	Signal processing techniques, modeling-based and learning-based algorithms, applications, performance results, challenges, future trends	Mention	No mention
Wang-2018 [84]	Detection and recognition, numerical analysis, and image generation	Working principle and potential applications of device-free wireless sensing system	No mention	No mention
Liu-2020 [36]	Human activity recognition	The types of wireless signals, theoretical models, signal processing techniques, activity segmentation, feature extraction, classification, and application	Mention	No mention
This survey	Detection, recognition, and estimation applications	Impact factors, signal processing, cross-domain sensing algorithms, performance comparison	Focus	A survey of cross-domain WiFi sensing algorithms

briefly describe how they reduce the effort involved in data collection and model retraining in new domains. Our survey is different from existing ones in that it conducts a comprehensive overview of state-of-the-art solutions to the problem of cross-domain WiFi sensing. To this end, we review more than 60 cross-domain WiFi sensing works to explore the cross-domain issues and summarize five types of approaches: domain-invariant feature extraction, virtual sample generation, transfer learning, few-shot learning, and big data solution.

7 CONCLUSION

This article has conducted a survey of the state-of-the-art cross-domain WiFi sensing applications using CSI information. We introduce the mathematical model of CSI and further analyze how new domains, such as new environments, different configurations, and unseen users, impact the CSI propagations. Then we review the widely used signal processing techniques, which could not deal with the sharp drop in the cross-domain sensing performance. To overcome this drawback, five

types of cross-domain sensing algorithms are summarized based on domain-invariant feature extraction, virtual sample generation, transfer learning, few-shot learning, and big data solution. We further compare existing research works using these five solutions and discuss their respective advantages and limitations. There are still several other challenges that need to be addressed in future work, including impact of moving objects, robustness to electromagnetic interference, incremental real-time data for cross-domain WiFi sensing, and multi-task cross-domain WiFi sensing.

REFERENCES

[1] Heba Abdelnasser, Moustafa Youssef, and Khaled A. Harras. 2015. WiGest: A ubiquitous WiFi-based gesture recognition system. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1472–1480.

[2] Fadel Adib, Zach Kabelac, Dina Katabi, and Robert C. Miller. 2014. 3D tracking via body radio reflections. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI'14)*. 317–329.

[3] Fadel Adib and Dina Katabi. 2013. See through walls with WiFi!. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*. 75–86.

[4] Yunhao Bai, Zejiang Wang, Kuangyu Zheng, Xiaorui Wang, and Junmin Wang. 2019. WiDrive: Adaptive WiFi-based recognition of driver activity for real-time and safe takeover. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 901–911.

[5] Xi Chen, Hang Li, Chenyi Zhou, Xue Liu, Di Wu, and Gregory Dudek. 2020. Fido: Ubiquitous fine-grained WiFi-based localization for unlabelled users via domain adaptation. In *Proceedings of The Web Conference 2020*. 23–33.

[6] Xi Chen, Hang Li, Chenyi Zhou, Xue Liu, Di Wu, and Gregory Dudek. 2022. Fidora: Robust WiFi-based indoor localization via unsupervised domain adaptation. *IEEE Internet of Things Journal* (2022).

[7] Xi Chen, Chen Ma, Michel Allegue, and Xue Liu. 2017. Taming the inconsistency of Wi-Fi fingerprints for device-free passive indoor localization. In *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 1–9.

[8] Yuanying Chen, Wei Dong, Yi Gao, Xue Liu, and Tao Gu. 2017. Rapid: A multimodal and device-free approach using noise estimation for robust person identification. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–27.

[9] Linsong Cheng and Jiliang Wang. 2016. How can I guard my AP? Non-intrusive user identification for mobile devices using WiFi signals. In *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 91–100.

[10] Diane Cook, Kyle D. Feuz, and Narayanan C. Krishnan. 2013. Transfer learning for activity recognition: A survey. *Knowledge and Information Systems* 36, 3 (2013), 537–556.

[11] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)* 39, 1 (1977), 1–22.

[12] Xue Ding, Ting Jiang, Yanan Li, Wenling Xue, and Yi Zhong. 2020. Device-free location-independent human activity recognition using transfer learning based on CNN. In *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 1–6.

[13] Xue Ding, Ting Jiang, Yi Zhong, Yan Huang, and Zhiwei Li. 2021. Wi-Fi-based location-independent human activity recognition via meta learning. *Sensors* 21, 8 (2021), 2654.

[14] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*. PMLR, 1180–1189.

[15] Ruiyang Gao, Wenwei Li, Yaxiong Xie, Enze Yi, Leye Wang, Dan Wu, and Daqing Zhang. 2022. Towards robust gesture recognition by characterizing the sensing quality of WiFi signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 1 (2022), 1–26.

[16] Ruiyang Gao, Mi Zhang, Jie Zhang, Yang Li, Enze Yi, Dan Wu, Leye Wang, and Daqing Zhang. 2021. Towards position-independent sensing for gesture recognition with Wi-Fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 2 (2021), 1–28.

[17] Arthur Gretton, Dino Sejdinovic, Heiko Strathmann, Sivaraman Balakrishnan, Massimiliano Pontil, Kenji Fukumizu, and Bharath K. Sriperumbudur. 2012. Optimal kernel choice for large-scale two-sample tests. In *Advances in Neural Information Processing Systems*. Citeseer, 1205–1213.

[18] Yu Gu, Jinhai Zhan, Yusheng Ji, Jie Li, Fuji Ren, and Shangbing Gao. 2017. MoSense: An RF-based motion detection system via off-the-shelf WiFi devices. *IEEE Internet of Things Journal* 4, 6 (2017), 2326–2341.

[19] Ying He, Yan Chen, Yang Hu, and Bing Zeng. 2020. WiFi vision: Sensing, recognition, and detection with commodity MIMO-OFDM WiFi. *IEEE Internet of Things Journal* 7, 9 (2020), 8296–8317.

[20] Pengli Hu, Chengpei Tang, Kang Yin, and Xie Zhang. 2021. WiGR: A practical Wi-Fi-based gesture recognition system with a lightweight few-shot network. *Applied Sciences* 11, 8 (2021), 3329.

[21] Yuqian Hu, Feng Zhang, Chenshu Wu, Beibei Wang, and K. J. Ray Liu. 2020. A WiFi-based passive fall detection system. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1723–1727.

[22] Jinyang Huang, Bin Liu, Pengfei Liu, Chao Chen, Ning Xiao, Yu Wu, Chi Zhang, and Nenghai Yu. 2020. Towards anti-interference WiFi-based activity recognition system using interference-independent phase component. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 576–585.

[23] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 289–304.

[24] Wenjun Jiang, Hongfei Xue, Chenglin Miao, Shiyang Wang, Sen Lin, Chong Tian, Srinivasan Murali, Haochen Hu, Zhi Sun, and Lu Su. 2020. Towards 3D human pose construction using WiFi. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[25] Hua Kang, Qian Zhang, and Qianyi Huang. 2021. Context-aware wireless based cross domain gesture recognition. *IEEE Internet of Things Journal* (2021).

[26] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1725–1732.

[27] Belal Korany, Hong Cai, and Yasamin Mostofi. 2020. Multiple people identification through walls using off-the-shelf WiFi. *IEEE Internet of Things Journal* (2020).

[28] 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.

[29] Chenning Li, Zhichao Cao, and Yunhao Liu. 2021. Deep AI enabled ubiquitous wireless sensing: A survey. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–35.

[30] Chenning Li, Manni Liu, and Zhichao Cao. 2020. WiHF: Enable user identified gesture recognition with WiFi. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 586–595.

[31] Hang Li, Xi Chen, Ju Wang, Di Wu, and Xue Liu. 2021. DAFI: WiFi-based device-free indoor localization via domain adaptation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 4 (2021), 1–21.

[32] He Li, Kaoru Ota, Mianxiong Dong, and Minyi Guo. 2018. Learning human activities through Wi-Fi channel state information with multiple access points. *IEEE Communications Magazine* 56, 5 (2018), 124–129.

[33] Xinyi Li, Liqiong Chang, Fangfang Song, Ju Wang, Xiaojiang Chen, Zhanyong Tang, and Zheng Wang. 2021. CrossGR: Accurate and low-cost cross-target gesture recognition using Wi-Fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 1 (2021), 1–23.

[34] Xiang Li, Daqing Zhang, Qin Lv, Jie Xiong, Shengjie Li, Yue Zhang, and Hong Mei. 2017. IndoTrack: Device-free indoor human tracking with commodity Wi-Fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–22.

[35] Xiang Li, Daqing Zhang, Jie Xiong, Yue Zhang, Shengjie Li, Yasha Wang, and Hong Mei. 2018. Training-free human vitality monitoring using commodity Wi-Fi devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 1–25.

[36] Jiao Liu, Guanlong Teng, and Feng Hong. 2020. Human activity sensing with wireless signals: A survey. *Sensors* 20, 4 (2020), 1210.

[37] Jialin Liu, Lei Wang, Linlin Guo, Jian Fang, Bingxian Lu, and Wei Zhou. 2017. A research on CSI-based human motion detection in complex scenarios. In *2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, 1–6.

[38] Jian Liu, Yan Wang, Yingying Chen, Jie Yang, Xu Chen, and Jerry Cheng. 2015. Tracking vital signs during sleep leveraging off-the-shelf WiFi. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 267–276.

[39] Shangqing Liu, Yanchao Zhao, Fanggang Xue, Bing Chen, and Xiang Chen. 2019. DeepCount: Crowd counting with WiFi via deep learning. *arXiv preprint arXiv:1903.05316* (2019).

[40] Xuefeng Liu, Jiannong Cao, Shaojie Tang, and Jiaqi Wen. 2014. Wi-Sleep: Contactless sleep monitoring via WiFi signals. In *2014 IEEE Real-Time Systems Symposium*. IEEE, 346–355.

[41] Xuefeng Liu, Jiannong Cao, Shaojie Tang, Jiaqi Wen, and Peng Guo. 2015. Contactless respiration monitoring via off-the-shelf WiFi devices. *IEEE Transactions on Mobile Computing* 15, 10 (2015), 2466–2479.

[42] Yong Lu, Shaohe Lv, and Xiaodong Wang. 2019. Towards location independent gesture recognition with commodity WiFi devices. *Electronics* 8, 10 (2019), 1069.

[43] Junyi Ma, Yuxiang Wang, Hao Wang, Yasha Wang, and Daqing Zhang. 2016. When can we detect human respiration with commodity WiFi devices?. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. 325–328.

[44] Xiaorui Ma, Yunong Zhao, Liang Zhang, Qinghua Gao, Miao Pan, and Jie Wang. 2019. Practical device-free gesture recognition using WiFi signals based on meta-learning. *IEEE Transactions on Industrial Informatics* (2019).

[45] Yongsen Ma, Sheheryar Arshad, Swetha Muniraju, Eric Torkildson, Enrico Rantala, Klaus Doppler, and Gang Zhou. 2021. Location- and person-independent activity recognition with WiFi, deep neural networks, and reinforcement learning. *ACM Transactions on Internet of Things* 2, 1 (2021), 1–25.

[46] Yongsen Ma, Gang Zhou, and Shuangquan Wang. 2019. WiFi sensing with channel state information: A survey. *ACM Computing Surveys (CSUR)* 52, 3 (2019), 1–36.

[47] Yongsen Ma, Gang Zhou, Shuangquan Wang, Hongyang Zhao, and Woosub Jung. 2018. SignFi: Sign language recognition using WiFi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–21.

[48] Alan Mazankiewicz, Klemens Böhm, and Mario Bergés. 2020. Incremental real-time personalization in human activity recognition using domain adaptive batch normalization. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–20.

[49] Wei Meng, Xingcan Chen, Wei Cui, and Jing Guo. 2021. WiHGR: A robust WiFi-based human gesture recognition system via sparse recovery and modified attention-based BGRU. *IEEE Internet of Things Journal* (2021).

[50] Takashi Nakamura, Mondher Bouazizi, Kohei Yamamoto, and Tomoaki Ohtsuki. 2022. Wi-Fi-based fall detection using spectrogram image of channel state information. *IEEE Internet of Things Journal* (2022).

[51] Isura Nirmal, Abdelwahed Khamis, Mahbub Hassan, Wen Hu, and Xiaoqing Zhu. 2021. Deep learning for radio-based human sensing: Recent advances and future directions. *IEEE Communications Surveys & Tutorials* (2021).

[52] Kai Niu, Fusang Zhang, Xuanzhi Wang, Qin Lv, Haitong Luo, and Daqing Zhang. 2021. Understanding WiFi signal frequency features for position-independent gesture sensing. *IEEE Transactions on Mobile Computing* (2021).

[53] Sameera Palipana, David Rojas, Piyush Agrawal, and Dirk Pesch. 2018. FallDeFi: Ubiquitous fall detection using commodity Wi-Fi devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 1–25.

[54] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10 (2009), 1345–1359.

[55] Kun Qian, Chenshu Wu, Zheng Yang, Yunhao Liu, and Kyle Jamieson. 2017. Widar: Decimeter-level passive tracking via velocity monitoring with commodity Wi-Fi. In *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 1–10.

[56] Kun Qian, Chenshu Wu, Yi Zhang, Guidong Zhang, Zheng Yang, and Yunhao Liu. 2018. Widar2.0: Passive human tracking with a single Wi-Fi link. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 350–361.

[57] Sai Deepika Regani, Beibei Wang, Min Wu, and K. J. Ray Liu. 2020. Time reversal based robust gesture recognition using WiFi. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 8309–8313.

[58] Burr Settles. 2009. Active learning literature survey. (2009).

[59] Muhammad Shahzad and Shaohu Zhang. 2018. Augmenting user identification with WiFi based gesture recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 1–27.

[60] Biyun Sheng, Fu Xiao, Letian Sha, and Lijuan Sun. 2020. Deep spatial-temporal model based cross-scene action recognition using commodity WiFi. *IEEE Internet of Things Journal* (2020).

[61] Cong Shi, Jian Liu, Nick Borodinov, Bruno Leao, and Yingying Chen. 2020. Towards environment-independent behavior-based user authentication using WiFi. In *2020 IEEE 17th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 666–674.

[62] Cong Shi, Jian Liu, Hongbo Liu, and Yingying Chen. 2017. Smart user authentication through actuation of daily activities leveraging WiFi-enabled IoT. In *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 1–10.

[63] Cong Shi, Tianming Zhao, Yucheng Xie, Tianfang Zhang, Yan Wang, Xiaonan Guo, and Yingying Chen. 2021. Environment-independent in-baggage object identification using WiFi signals. In *2021 IEEE 18th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*. IEEE, 71–79.

[64] Zhenguo Shi, Qingqing Cheng, J. Andrew Zhang, and Richard Yida Xu. 2022. Environment-robust WiFi-based human activity recognition using enhanced CSI and deep learning. *IEEE Internet of Things Journal* (2022).

[65] Zhenguo Shi, J. Andrew Zhang, Richard Xu, Qingqing Cheng, and Andre Pearce. 2020. Towards environment-independent human activity recognition using deep learning and enhanced CSI. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 1–6.

[66] Zhenguo Shi, Jian Andrew Zhang, Yi Da Richard Xu, and Qingqing Cheng. 2020. Environment-robust device-free human activity recognition with channel-state-information enhancement and one-shot learning. *IEEE Transactions on Mobile Computing* (2020).

[67] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199* (2014).
[68] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems* 30 (2017).
[69] Elahe Soltanaghaei, Rahul Anand Sharma, Zehao Wang, Adarsh Chittilappilly, Anh Luong, Eric Giler, Katie Hall, Steve Elias, and Anthony Rowe. 2020. Robust and practical WiFi human sensing using on-device learning with a domain adaptive model. In *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*. 150–159.
[70] Sheng Tan, Jie Yang, and Yingying Chen. 2020. Enabling fine-grained finger gesture recognition on commodity WiFi devices. *IEEE Transactions on Mobile Computing* (2020).
[71] Sheng Tan, Linghan Zhang, Zi Wang, and Jie Yang. 2019. MultiTrack: Multi-user tracking and activity recognition using commodity WiFi. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
[72] Isaac Triguero, Salvador García, and Francisco Herrera. 2015. Self-labeled techniques for semi-supervised learning: Taxonomy, software and empirical study. *Knowledge and Information Systems* 42, 2 (2015), 245–284.
[73] Ph. Van Dorp and F. C. A. Groen. 2008. Feature-based human motion parameter estimation with radar. *IET Radar, Sonar & Navigation* 2, 2 (2008), 135–145.
[74] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*. 1096–1103.
[75] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. *arXiv preprint arXiv:1606.04080* (2016).
[76] Aditya Virmani and Muhammad Shahzad. 2017. Position and orientation agnostic gesture recognition using WiFi. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. 252–264.
[77] Dazhuo Wang, Jianfei Yang, Wei Cui, Lihua Xie, and Sumei Sun. 2021. Multimodal CSI-based human activity recognition using GANs. *IEEE Internet of Things Journal* 8, 24 (2021), 17345–17355.
[78] Dazhuo Wang, Jianfei Yang, Wei Cui, Lihua Xie, and Sumei Sun. 2022. CAUTION: A robust WiFi-based human authentication system via few-shot open-set gait recognition. *IEEE Internet of Things Journal* (2022).
[79] Fangxin Wang, Wei Gong, and Jiangchuan Liu. 2018. On spatial diversity in WiFi-based human activity recognition: A deep learning-based approach. *IEEE Internet of Things Journal* 6, 2 (2018), 2035–2047.
[80] Fangxin Wang, Jiangchuan Liu, and Wei Gong. 2019. WiCAR: WiFi-based in-car activity recognition with multi-adversarial domain adaptation. In *Proceedings of the International Symposium on Quality of Service*. 1–10.
[81] Hao Wang, Daqing Zhang, Junyi Ma, Yasha Wang, Yuxiang Wang, Dan Wu, Tao Gu, and Bing Xie. 2016. Human respiration detection with commodity WiFi devices: Do user location and body orientation matter?. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 25–36.
[82] Hao Wang, Daqing Zhang, Yasha Wang, Junyi Ma, Yuxiang Wang, and Shengjie Li. 2016. RT-Fall: A real-time and contactless fall detection system with commodity WiFi devices. *IEEE Transactions on Mobile Computing* 16, 2 (2016), 511–526.
[83] Jie Wang, Qinhua Gao, Xiaorui Ma, Yunong Zhao, and Yuguang Fang. 2020. Learning to sense: Deep learning for wireless sensing with less training efforts. *IEEE Wireless Communications* 27, 3 (2020), 156–162.
[84] Jie Wang, Qinhua Gao, Miao Pan, and Yuguang Fang. 2018. Device-free wireless sensing: Challenges, opportunities, and applications. *IEEE Network* 32, 2 (2018), 132–137.
[85] Ju Wang, Hongbo Jiang, Jie Xiong, Kyle Jamieson, Xiaojiang Chen, Dingyi Fang, and Binbin Xie. 2016. LiFS: Low human-effort, device-free localization with fine-grained subcarrier information. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. 243–256.
[86] Wei Wang, Alex X. Liu, and Muhammad Shahzad. 2016. Gait recognition using WiFi signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 363–373.
[87] Wei Wang, Alex X. Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. 2015. Understanding and modeling of WiFi signal based human activity recognition. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 65–76.
[88] Wei Wang, Alex X. Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. 2017. Device-free human activity recognition using commercial WiFi devices. *IEEE Journal on Selected Areas in Communications* 35, 5 (2017), 1118–1131.
[89] Xuyu Wang, Chao Yang, and Shiwen Mao. 2017. TensorBeat: Tensor decomposition for monitoring multiperson breathing beats with commodity WiFi. *ACM Transactions on Intelligent Systems and Technology (TIST)* 9, 1 (2017), 1–27.
[90] Xuyu Wang, Chao Yang, and Shiwen Mao. 2020. Resilient respiration rate monitoring with realtime bimodal CSI data. *IEEE Sensors Journal* 20, 17 (2020), 10187–10198.

[91] Yan Wang, Jian Liu, Yingying Chen, Marco Gruteser, Jie Yang, and Hongbo Liu. 2014. E-eyes: Device-free location-oriented activity identification using fine-grained WiFi signatures. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*. 617–628.

[92] Yuxi Wang, Kaishun Wu, and Lionel M. Ni. 2016. WiFall: Device-free fall detection by wireless networks. *IEEE Transactions on Mobile Computing* 16, 2 (2016), 581–594.

[93] Yongchuan Wang, Song Yang, Fan Li, Yue Wu, and Yu Wang. 2021. FallViewer: A Fine-Grained Indoor Fall Detection System With Ubiquitous Wi-Fi Devices. *IEEE Internet of Things Journal* 8, 15 (2021), 12455–12466.

[94] Zhu Wang, Bin Guo, Zhiwen Yu, and Xingshe Zhou. 2018. Wi-Fi CSI-based behavior recognition: From signals and actions to activities. *IEEE Communications Magazine* 56, 5 (2018), 109–115.

[95] Zhengjie Wang, Zehua Huang, Chengming Zhang, Wenwen Dou, Yinjing Guo, and Da Chen. 2021. CSI-based human sensing using model-based approaches: A survey. *Journal of Computational Design and Engineering* 8, 2 (2021), 510–523.

[96] Zhengjie Wang, Kangkang Jiang, Yushan Hou, Wenwen Dou, Chengming Zhang, Zehua Huang, and Yinjing Guo. 2019. A survey on human behavior recognition using channel state information. *IEEE Access* 7 (2019), 155986–156024.

[97] Bo Wei, Wen Hu, Mingrui Yang, and Chun Tung Chou. 2015. Radio-based device-free activity recognition with radio frequency interference. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*. 154–165.

[98] Chenshu Wu, Zheng Yang, Zimu Zhou, Xuefeng Liu, Yunhao Liu, and Jiannong Cao. 2015. Non-invasive detection of moving and stationary human with WiFi. *IEEE Journal on Selected Areas in Communications* 33, 11 (2015), 2329–2342.

[99] Chenshu Wu, Feng Zhang, Yuqian Hu, and K. J. Ray Liu. 2020. GaitWay: Monitoring and recognizing gait speed through the walls. *IEEE Transactions on Mobile Computing* 20, 6 (2020), 2186–2199.

[100] Dan Wu, Youwei Zeng, Ruiyang Gao, Shengjie Li, Yang Li, Rahul C. Shah, Hong Lu, and Daqing Zhang. 2021. WiTraj: Robust indoor motion tracking with WiFi signals. *IEEE Transactions on Mobile Computing* (2021).

[101] Dan Wu, Daqing Zhang, Chenren Xu, Hao Wang, and Xiang Li. 2017. Device-free WiFi human sensing: From pattern-based to model-based approaches. *IEEE Communications Magazine* 55, 10 (2017), 91–97.

[102] Kaishun Wu. 2016. Wi-metal: Detecting metal by using wireless networks. In *2016 IEEE International Conference on Communications (ICC)*. IEEE, 1–6.

[103] Chunjing Xiao, Daojun Han, Yongsen Ma, and Zhiguang Qin. 2019. CsiGAN: Robust channel state information-based activity recognition with GANs. *IEEE Internet of Things Journal* 6, 6 (2019), 10191–10204.

[104] Rui Xiao, Jianwei Liu, Jinsong Han, and Kui Ren. 2021. OneFi: One-shot recognition for unseen gesture via COTS WiFi. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*. 206–219.

[105] Yaxiong Xie, Zhenjiang Li, and Mo Li. 2018. Precise power delay profiling with commodity Wi-Fi. *IEEE Transactions on Mobile Computing* 18, 6 (2018), 1342–1355.

[106] Yaxiong Xie, Jie Xiong, Mo Li, and Kyle Jamieson. 2019. mD-Track: Leveraging multi-dimensionality for passive indoor Wi-Fi tracking. In *The 25th Annual International Conference on Mobile Computing and Networking*. 1–16.

[107] Jianfei Yang, Han Zou, Yuxun Zhou, and Lihua Xie. 2019. Learning gestures from WiFi: A siamese recurrent convolutional architecture. *IEEE Internet of Things Journal* 6, 6 (2019), 10763–10772.

[108] Siamak Yousefi, Hirokazu Narui, Sankalp Dayal, Stefano Ermon, and Shahrokh Valaee. 2017. A survey on behavior recognition using WiFi channel state information. *IEEE Communications Magazine* 55, 10 (2017), 98–104.

[109] Nan Yu, Wei Wang, Alex X. Liu, and Lingtao Kong. 2018. QGesture: Quantifying gesture distance and direction with WiFi signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–23.

[110] Youwei Zeng, Dan Wu, Jie Xiong, Enze Yi, Ruiyang Gao, and Daqing Zhang. 2019. FarSense: Pushing the range limit of WiFi-based respiration sensing with CSI ratio of two antennas. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–26.

[111] Fusang Zhang, Kai Niu, Jie Xiong, Beihong Jin, Tao Gu, Yuhang Jiang, and Daqing Zhang. 2019. Towards a diffraction-based sensing approach on human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 1 (2019), 1–25.

[112] Feng Zhang, Chenshu Wu, Beibei Wang, Hung-Quoc Lai, Yi Han, and K. J. Ray Liu. 2019. WiDetect: Robust motion detection with a statistical electromagnetic model. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 3 (2019), 1–24.

[113] Feng Zhang, Chenshu Wu, Beibei Wang, Min Wu, Daniel Bugos, Hangfang Zhang, and K. J. Ray Liu. 2019. SMARS: Sleep monitoring via ambient radio signals. *IEEE Transactions on Mobile Computing* 20, 1 (2019), 217–231.

[114] Jie Zhang, Yang Li, Haoyi Xiong, Dejing Dou, Chunyan Miao, and Daqing Zhang. 2022. HandGest: Hierarchical sensing for robust in-the-air handwriting recognition with commodity WiFi devices. *IEEE Internet of Things Journal* (2022).

[115] Jie Zhang, Zhanyong Tang, Meng Li, Dingyi Fang, Petteri Nurmi, and Zheng Wang. 2018. CrossSense: Towards cross-site and large-scale WiFi sensing. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. 305–320.

[116] Jin Zhang, Fuxiang Wu, Bo Wei, Qieshi Zhang, Hui Huang, Syed W. Shah, and Jun Cheng. 2020. Data augmentation and dense-LSTM for human activity recognition using WiFi signal. *IEEE Internet of Things Journal* 8, 6 (2020), 4628–4641.

[117] Lingyan Zhang, Qinghua Gao, Xiaorui Ma, Jie Wang, Tingting Yang, and Hongyu Wang. 2018. DeFi: Robust training-free device-free wireless localization with WiFi. *IEEE Transactions on Vehicular Technology* 67, 9 (2018), 8822–8831.

[118] Lei Zhang, Cong Wang, Maode Ma, and Daqing Zhang. 2019. WiDIGR: Direction-independent gait recognition system using commercial Wi-Fi devices. *IEEE Internet of Things Journal* (2019).

[119] Lei Zhang, Cong Wang, and Daqing Zhang. 2021. Wi-PIGR: Path independent gait recognition with commodity Wi-Fi. *IEEE Transactions on Mobile Computing* (2021).

[120] Lei Zhang, Yixiang Zhang, and Xiaolong Zheng. 2020. WiSign: Ubiquitous American Sign Language recognition using commercial Wi-Fi devices. *ACM Transactions on Intelligent Systems and Technology (TIST)* 11, 3 (2020), 1–24.

[121] Xie Zhang, Chengpei Tang, Kang Yin, and Qingqian Ni. 2021. WiFi-based cross-domain gesture recognition via modified prototypical networks. *IEEE Internet of Things Journal* 9, 11 (2021), 8584–8596.

[122] Lili Zheng, Bin-Jie Hu, Jinguang Qiu, and Manman Cui. 2020. A deep-learning-based self-calibration time-reversal fingerprinting localization approach on Wi-Fi platform. *IEEE Internet of Things Journal* 7, 8 (2020), 7072–7083.

[123] Yue Zheng, Chenshu Wu, Kun Qian, Zheng Yang, and Yunhao Liu. 2017. Detecting radio frequency interference for CSI measurements on COTS WiFi devices. In *2017 IEEE International Conference on Communications (ICC)*. IEEE, 1–6.

[124] Yue Zheng, Zheng Yang, Junjie Yin, Chenshu Wu, Kun Qian, Fu Xiao, and Yunhao Liu. 2018. Combating cross-technology interference for robust wireless sensing with COTS WiFi. In *2018 27th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 1–9.

[125] Yue Zheng, Yi Zhang, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. 2019. Zero-effort cross-domain gesture recognition with Wi-Fi. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. 313–325.

[126] Rui Zhou, Huanhuan Hou, Ziyuan Gong, Zuona Chen, Kai Tang, and Bao Zhou. 2020. Adaptive device-free localization in dynamic environments through adaptive neural networks. *IEEE Sensors Journal* 21, 1 (2020), 548–559.

[127] Jincao Zhu, Youngbin Im, Shivakant Mishra, and Sangtae Ha. 2017. Calibrating time-variant, device-specific phase noise for COTS WiFi devices. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–12.

[128] Han Zou, Jianfei Yang, Yuxun Zhou, and Costas J. Spanos. 2018. Joint adversarial domain adaptation for resilient WiFi-enabled device-free gesture recognition. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 202–207.

[129] Yongpan Zou, Weifeng Liu, Kaishun Wu, and Lionel M. Ni. 2017. Wi-Fi radar: Recognizing human behavior with commodity Wi-Fi. *IEEE Communications Magazine* 55, 10 (2017), 105–111.

Received 23 November 2021; revised 5 October 2022; accepted 17 October 2022