Widar3.0: Zero-Effort Cross-Domain Gesture Recognition with Wi-Fi

Yi Zhang, *Student Member, IEEE,* Yue Zheng, *Student Member, IEEE,* Kun Qian, *Student Member, IEEE,* Guidong Zhang, *Student Member, IEEE,* Yunhao Liu, *Fellow, IEEE,* Chenshu Wu, *Member, IEEE,* Zheng Yang*, *Member, IEEE*

Abstract—With the development of signal processing technology, the ubiquitous Wi-Fi devices open an unprecedented opportunity to solve the challenging human gesture recognition problem by learning motion representations from wireless signals. Wi-Fi-based gesture recognition systems, although yield good performance on specific data domains, are still practically difficult to be used without explicit adaptation efforts to new domains. Various pioneering approaches have been proposed to resolve this contradiction but extra training efforts are still necessary for either data collection or model re-training when new data domains appear. To advance cross-domain recognition and achieve fully zero-effort recognition, we propose Widar3.0, a Wi-Fi-based zero-effort cross-domain gesture recognition system. The key insight of Widar3.0 is to derive and extract domain-independent features of human gestures at the lower signal level, which represent unique kinetic characteristics of gestures and are irrespective of domains. On this basis, we develop a one-fits-all general model that requires only one-time training but can adapt to different data domains. Experiments on various domain factors (i.e. environments, locations, and orientations of persons) demonstrate the accuracy of 92.7% for in-domain recognition and 82.6%-92.4% for cross-domain recognition without model re-training, outperforming the state-of-the-art solutions.

Index Terms—Gesture Recognition, Feature Extraction, Wireless Sensing, COTS WiFi

1 Introduction

Human gesture recognition is the core enabler for a wide range of applications such as smart homes, security surveillance, and virtual reality. Traditional approaches use cameras [1], [2], [3], wearable devices and phones [4], [5], [6] or sonar [7], [8], [9] as the sensing module. While promising, these approaches pose inconvenience due to their respective drawbacks including leakage of privacy, the requirement of on-body sensors, and the limit of sensing range. The need for a secure, device-free, and ubiquitous gesture recognition interface has triggered extensive research on sensing solutions based on the wireless signals extracted from commodity Wi-Fi. Pioneer attempts such as E-eyes [10], CARM [11], WiGest [12] and WIMU [13] have been proposed. In principle, early wireless sensing solutions extract either statistical features (e.g., histograms of signal amplitudes [10]) or physical features (e.g., power profiles of Doppler frequency shifts [11]) from Wi-Fi signals and map them to human gestures. However, these primitive signal features usually carry adverse environment information unrelated to gestures. Specifically, due to lack of spatial resolution, wireless signals, and their features as well, are highly specific to *environment* where the gesture is performed, and the *location* and *orientation* of the performer, as Fig. 1 shows. For brevity, we unitedly term these factors irrelevant to gestures as *domain*. As a result, the classifiers trained with primitive signal features in one domain usually undergo drastically drop in accuracy with another domain.

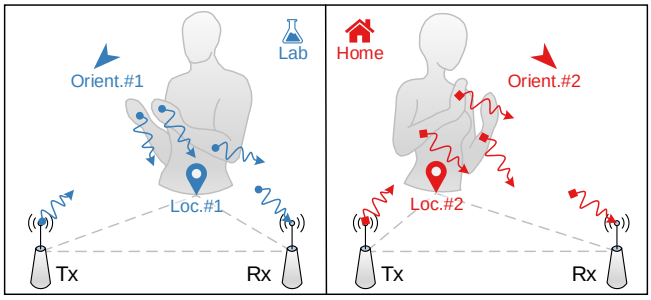


Fig. 1. Cross-domain gesture recognition, where persons may be at different locations and orientations relative to Wi-Fi links, and environments (e.g., lab, home, etc.). In this example, one male and one female are performing clapping gestures in two domains.

Recent innovations in gesture recognition with Wi-Fi have explored the cross-domain generalization ability of recognition models. For example, recent works [14], [15] borrow the ideas from recent advances in deep learning, such as transfer learning and adversarial learning, and apply advanced learning methodologies to improve cross-domain recognition performance. Another solution,

- *A preliminary version of this article appeared in the 17th International Conference on Mobile Systems, Applications, and Services (MobiSys 2019).*
- ** Y. Zhang and Y. Zheng are co-primary authors and Z. Yang is the corresponding author.*
- *Y. Zhang, Y. Zheng, G. Zhang, Y. Liu, Z. Yang are with the School of Software, Tsinghua University, Beijing, PR China. E-mail: {zhangyithss, cczhengy, zhanggd18, yunhaoliu, hmilyyz}@gmail.com*
- *K. Qian is with the Department of Electrical and Computer Engineering, University of California San Diego, California, USA. E-mail: qiank10@gmail.com*
- *Y. Liu is also with the Department of Computer Science and Engineering, Michigan State University, Michigan, USA.*
- *C. Wu is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, Maryland, USA. E-mail: wucs32@gmail.com*

WiAG [16], derives a translation function to generate signal features of the target domain for model re-training. While to some extent achieving cross-domain recognition, all existing works require extra training efforts in either data collection or model re-training at each time a new target domain is added into the recognition model. Even worse, correlated with the continuous location and orientation of a person, Wi-Fi signals have an infinite number of domains, making cross-domain training approaches practically prohibitive.

A more promising but challenging solution is a *"one-fits-all"* model that is able to *train once, use anywhere*. Such an ideal model, trained in one domain, can be directly used in new domains without extra efforts, such as data collection, generation, or re-training. Different from all existing approaches, our key idea is to move generalization ability downward at the lower signal level, rather than the upper model level. Specifically, we extract domain-independent features reflecting only gesture itself from raw domain-dependent signals. On this basis, we aim to build an explainable cross-domain recognition model that can be applied in new scenarios with zero effort and high accuracy.

However, we face three major technical challenges to achieve a *one-fits-all* model. First, previously used signal features (e.g., amplitude, phase, Doppler Frequency Shift (DFS)), as well as their statistics (e.g., max, min, mean, distribution parameter), are domain-dependent, meaning that their values vary with different locations, orientations, and environments even for the same gesture. Second, it is difficult, for radio signals from only several links, to describe human gestures and actions. For example, the kinetic profile of a single gesture still has hundreds of variables, posing the estimation of kinetic profile as a highly under-determined problem. Third, cross-domain generalization often requires sophisticated learning models (e.g., deeper networks, a larger number of parameters, a more complex network structure, and more complicated loss functions), which slow down or even obstruct training, over-consume training data, and make the model less explainable.

To overcome these challenges, we propose Widar3.0, a Wi-Fi-based gesture recognition system. Widar3.0 uses channel state information (CSI) portrayed by commodity Wi-Fi devices. Our prior efforts, Widar [17] and Widar2.0 [18] track coarse human motion status, e.g., torso location and velocity, by extracting the features from the dominant reflected signal off body torso. Widar3.0, however, aims at recognizing complex gestures that involve multiple body parts. The key component of Widar3.0 is our novel theoretically domain-independent feature *body-coordinate velocity profile* (BVP) that describes power distribution over different velocities, at which body parts are involved in the gesture movements. Our observation is that each type of gesture has its unique velocity profile in the body coordinate system (e.g., the coordinates where the orientation of the person is the positive x axis) no matter in which domain is the gesture performed. To estimate BVP, we approximate BVP from several prominent velocity components and further employ compressed sensing techniques to derive accurate estimates. On this basis, we devise a general learning model to capture spatial-temporal characteristics of BVP and finally classify gestures. Through the downward movement of model generalization techniques closer to the raw signals, Widar3.0 enables zero-effort cross-domain human gesture recognition with many expected properties simultaneously, including explainable features, high and reliable accuracy, strong generalization ability, reduced amounts of training data. We implement Widar3.0 on commodity Wi-Fi devices and conduct experiments on our released dataset (16 users, 15 gestures, 15 locations, and 5 orientations in 3 environments). Especially, the results demonstrate that Widar3.0 significantly improves the accuracy of gesture recognition to 92.4% in cross-environment cases, while the recognition accuracy with raw CSI and DFS profiles are 40.2% and 77.8% only. Across different types of domain factors including user's location, orientation, environment, and user diversity, Widar3.0 achieves an average accuracy of 89.7%, 82.6%, 92.4%, and 88.9%, respectively.

In a nutshell, our core contributions are three-fold. First, we present a novel domain-independent feature that captures body-coordinate velocity profiles of human gestures at the lower signal level. BVP is theoretically irrespective of any domain information in raw Wi-Fi signals and thus acts as a unique indicator for human gestures. Second, we develop a *one-fits-all* model on the basis of domain-independent BVP and a learning method that fully exploits spatial-temporal characteristics of BVP. The model enables cross-domain gesture recognition without any extra effort of data collection or model re-training. Third, though trained only once, Widar3.0 achieves an average of 89.7%, 82.6%, and 92.4% recognition accuracy across locations, orientations, and environments, respectively, which outperforms the state-of-the-art solutions that require re-training in new target domains. Such consistently high performance demonstrates its strong ability of cross-domain generalization. To the best of our knowledge, Widar3.0 is the first zero-effort cross-domain gesture recognition via Wi-Fi, a fundamental step towards ubiquitous sensing.

A preliminary version of Widar3.0 has been presented in [19]. We extend it in the following aspects:

- We design and implement a novel outlier detection algorithm to determine whether the gestures belong to the predefined gesture set and evaluation results exhibit conspicuous robustness to the unknown gestures compared to the previous conference version.
- We design and implement a novel dynamic link selection algorithm to mitigate the effect of gesture-irrelevant human motion, and extensive experiments demonstrate a significant improvement in system performance compared to the previous conference version.
- We provide details in the system implementation and conduct additional experiments to evaluate the robustness and agility of Widar3.0 under various settings.

2 Related Work

Our work is highly related to wireless human sensing techniques, which are roughly categorized into model-based and learning-based ones, targeting localization and activity recognition, respectively.

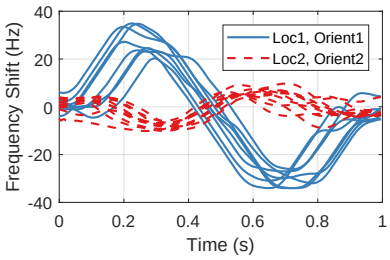


Fig. 2. Dominant DFS of gesture differs with person orientations and locations.

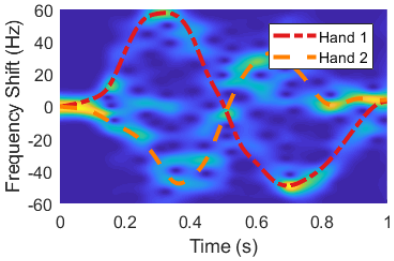


Fig. 3. Complex gestures cause multiple DFS components.

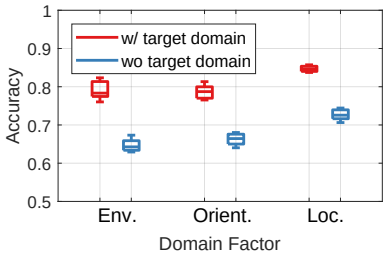


Fig. 4. Accuracy of adversarial learning drops without target domain data.

Model-based wireless localization. Model-based human sensing explicitly builds a physical link between wireless signals and human movements. On the signal side, existing approaches extract various parameters of signals reflected by human or emitted by portable devices, including DFS [11], [17], [20], ToF [21], [22], [23], [24], AoA/AoD [23], [24], [25], [26], ACF [27], and attenuation [28], [29]. Based on the types of devices used, parameters with different extent of accuracy and resolution can be obtained. WiTrack [21], [22] develops FMCW radar with wide bandwidth to accurately estimate ToFs of reflected signals. WiDeo [23] customizes full-duplex Wi-Fi to jointly estimate ToFs and AoAs of major reflectors. In contrast, though limited by the bandwidth and antenna number, Widar2.0 [18] improves resolution by jointly estimating ToF, AoA and DFS.

On the human side, existing model-based works only tracks coarse human motion status, such as location [21], [28], velocity [17], [20], gait [30], [31] and figure [24], [32]. Though not detailed enough, they provide coarse human movement information, which can further help Widar3.0 and other learning-based activity recognition works to remove domain dependencies of input signal features.

Learning-based wireless activity recognition. Due to complexity of human activity, existing approaches extract signal features, either statistical [10], [31], [33], [34], [35], [36], [37] or physical [11], [13], [16], [38], [39], [40], [41], [42], [43] ones, and map them to discrete activities. The statistical methods treat the wireless signal as time-series data, extract its waveforms and distributions in both time and frequency domain as fingerprints. E-eyes [10] is a pioneer work to use strength distribution of commercial Wi-Fi signals and KNN to recognize human activities. Niu et al. [37] uses signal waveforms for fine-grained gesture recognition. The physical methods take a step further to extract features with clear physical meanings. CARM [11] calculates the power distribution of DFS components as learning features of HMM model. WIMU [13] further segments DFS power profile for multi-person activity recognition. However, due to fundamental limits of domain dependencies of wireless signals, directly using either statistical or physical features is infeasible to generalize to different domains.

Tempts to adapt recognition schemes in various domains fall into two categories: virtually generating features for target domains [15], [16], [44], [45] and developing domain-independent features [14], [46], [47]. In the former type, WiAG [16] derives translation functions between CSIs from different domains and generates virtual training data accordingly. CrossSense [15] adopts the idea of transfer learning and proposes a roaming model to translate signal features between domains. However, features generated by these types of works are still domain-dependent, which require training of classifier for each individual domain, leading to a waste of training efforts. In contrast, with the help of passive localization, Widar3.0 directly uses domain-independent BVPs as features and trains the classifier only once.

In the latter type, the idea of adversarial learning is usually adopted to shift the task of separating gesture-related features from domain-related ones. EI [14] incorporates an adversarial network to obtain domain-independent features from CSI. However, cross-domain learning methodologies require extra data samples from the target domain, increasing data collection and training efforts. Moreover, features generated by learning models are semantically uninterpretable. In contrast, Widar3.0 explicitly extracts domain-independent BVPs, and only needs a simply designed learning model without the capability of cross-domain learning.

3 Motivation

Widar3.0 addresses the problem of cross-domain gesture recognition with Wi-Fi signals. Due to the lack of spatial resolution, wireless signals are highly formatted by domain characteristics. Either or not to some extent enabling cross-domain sensing, existing wireless sensing solutions have significant drawbacks in their feature usage. The three main types of features are listed as follows:

Primitive features without cross-domain capability. Most state-of-the-art activity recognition works extract primitive statistical (e.g., power distribution, waveform) or physical features (e.g., DFS, AoA, ToF) from CSI [48]. However, due to different locations and orientations of the person and multipath environments, features of the same gesture may vary significantly and fail to serve successful recognition. As a brief example, a person is asked to push his right-hand multiple times with two different orientations relative to the wireless link. The spectrograms are calculated as in [11], and dominant DFS caused by the movement of the hand is extracted. As shown in Fig. 2, while dominant DFS series of gestures with the same domain form compact clusters, they differ greatly in trends and amplitudes between two domains, and thus fail to indicate the same gesture.

Cross-domain motion features for coarse tracking. Device-free tracking approaches [18], [20] build quantitative relations between physical features of signal and the motion status of the person and enable location and velocity

measurement across environments. However, these works regard a person as a single point, which is infeasible for recognizing complex gestures that involve multiple body parts. Fig. 3 illustrates the spectrogram of a simple hand clap, which contains two major DFS components caused by two hands and a few secondary components.

Latent features from cross-domain learning methods. Cross-domain learning methods such as transfer learning [15] and adversarial learning [14] latently generate features of data samples in the target domain, either by translating samples from the source domain or learning domain-independent features. However, these works require extra efforts of collecting data samples from the target domain and retraining the classifier each time new target domains are added. As an example, we evaluate the performance of an adversarial learning based model, EI [14] over different domain factors (e.g., environment, location and orientation of the person). Specifically, the classifier is trained with and without data samples in every type of target domains. For the absence of target domain, the experiment settings are the same as that in Sec. 7.3. For the involvement of the target domain, the data samples for each domain are equally split into train and test. In Fig. 4, the central mark indicates the median, and the bottom and top edges of the box indicate the 25th and 75th percentiles. The whiskers correspond to the highest and lowest observations. As is shown, the system accuracy significantly drops without the knowledge of the target domains, demonstrating the need for extra data collection and training efforts in these learning methodologies.

Lessons learned. The deficiency of existing cross-dom -ain learning solutions asks for a new type of domain-independent feature. Should it be achieved, a *one-size-fits-all* model could be built upon it to save much data collection and training efforts. Widar3.0 is designed to develop and exploit body-coordinate velocity profile (BVP) to address the issue.

4 Overview of Widar3.0

Widar3.0 is a cross-domain gesture recognition system using off-the-shelf Wi-Fi devices. As shown in Fig. 5, multiple wireless links are deployed around the monitoring area. Wireless signals, as distorted by the user in the monitoring area, are acquired at receivers and their CSI measurements are logged and preprocessed to remove amplitude noises and phase offsets.

The major parts of Widar3.0 are two modules, the *BVP generation module* and the *gesture recognition module*.

Upon receiving sanitized CSI series, Widar3.0 divides CSI series into small segments and generates BVP for each CSI segment via the BVP generation module. Widar3.0 first prepares three intermediate results: DFS profiles, the orientation and location information of the person. DFS profiles are estimated by applying time-frequency analysis to the CSI series. The orientation and location information of the person is calculated via motion tracking approaches. Thereafter, Widar3.0 applies the proposed compressed-sensing-based optimization approach to estimate the BVP of each CSI segment. The BVP series is then output for following gesture recognition.

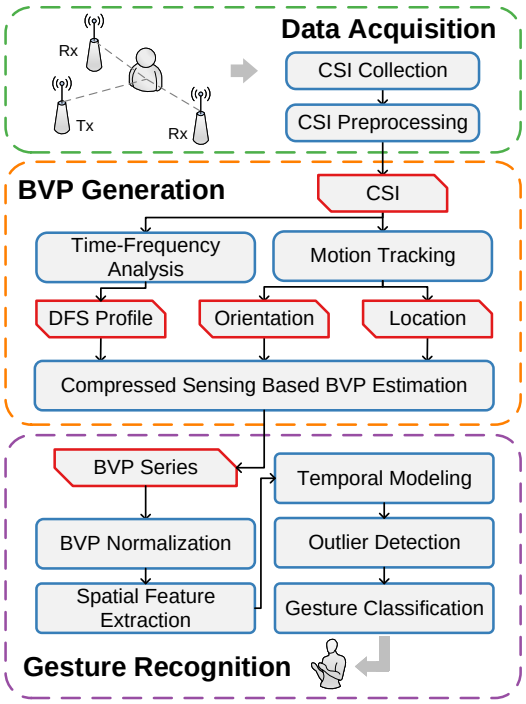


Fig. 5. System overview.

The gesture recognition module implements a deep learning neural network (DNN) for gesture recognition. With the BVP series as input, Widar3.0 performs normalization on each BVP and across the whole series, in order to remove the irrelevant variations of instances and persons. Afterward, the normalized BVP series is input into a spatial-temporal DNN, which has two main functions. First, the DNN extracts high-level spatial features within each BVP using convolutional layers. Then, recurrent layers are adopted to perform temporal modeling of inter-characteristics between BVPs. Before the gesture classification module, a novel outlier detection algorithm is applied on the output of the DNN to testify the legality of the gesture performed. Any gestures out of the predefined gesture set will be reported as illegal and will not be classified. Finally, the output of the DNN is used to indicate the type of gesture performed by the user. In principle, Widar3.0 achieves zero-effort cross-domain gesture recognition, which requires only one-time training of the DNN network but can be directly adapted to as many as new domains.

5 Body-coordinate Velocity Profile

Intuitively, human activities have unique velocity distributions across all body parts involved, which can be used as activity indicators. Among all parameters (i.e. ToF, AoA, DFS, and attenuation) of the signal reflected by the person, DFS embodies most information of velocity distribution. Unfortunately, DFS is also highly correlated with the location and orientation of the person, circumventing direct cross-domain activity recognition with DFS profiles.

In this section, we tempt to derive the distribution of signal power over velocity components in the body coordi-

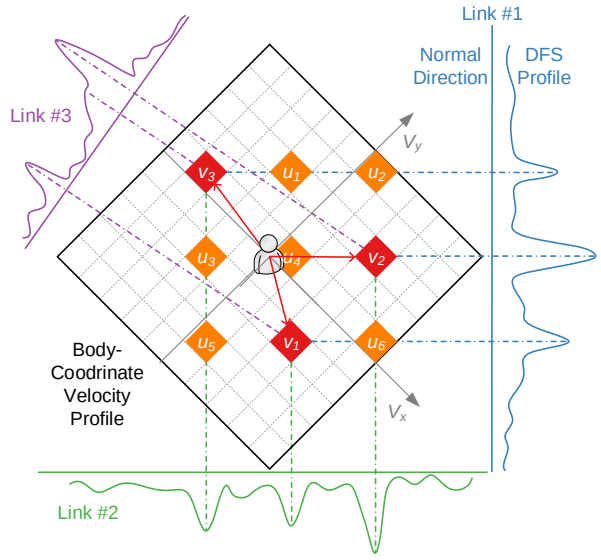


Fig. 6. Relationship between the BVP and DFS profiles. Each velocity component in BVP is projected onto the normal direction of a link, and contributes to the power of the corresponding radial velocity component in the DFS profile.

nate system, i.e. BVP, which uniquely indicates the type of activities. Preliminary of the CSI model is first introduced (§ 5.1), followed by the formulation and calculation of BVP (§ 5.2 and § 5.3).

5.1 Doppler Representation of CSI

CSI portrayed by off-the-shelf Wi-Fi devices describes multipath effects in the indoor environment at arrival time t of packets and frequency f of subcarriers:

$$In\hat{H}(f,t) = \Big(\sum_{l=1}^{L} \alpha_l(f,t) e^{-j2\pi f \tau_l(f,t)}\Big) e^{j\epsilon(f,t)}, \quad (1)$$

where L is the number of paths, α_l and τ_l are the complex attenuation and propagation delay of the l-th path, and $\epsilon(f,t)$ is the phase error caused by timing alignment offset, sampling frequency offset and carrier frequency offset.

By representing phases of multipath signals with the corresponding DFS, CSI can be transformed as [17]:

$$\hat{H}(f,t) = \Big(H_s(f) + \sum_{l \in P_d} \alpha_l(t) e^{j2\pi \int_{-\infty}^{t} f_{D_l}(u)\mathrm{d}u}\Big) e^{j\epsilon(f,t)}, \quad (2)$$

where the constant H_s is the sum of all static signals with zero DFS (e.g., LoS signal), and P_d is the set of dynamic signals with non-zero DFS (e.g., signals reflected by the target).

With conjugate multiplication of CSI of two antennas on the same Wi-Fi NIC calculated, and out-band noises and quasi-static offsets filtered out, random offsets can be removed and only prominent multipath components with non-zero DFS are retained [20]. As subcarriers of CSI are correlated with each other and each of them embodies different center frequency and DFS that is related to the wavelength of the signal, we adopt PCA-based algorithm proposed in CARM [11] to extract principal components of CSI streams. Further applying short-term Fourier transform yields power distribution over the time and Doppler frequency domains. One example of the spectrogram of a single link is shown in Fig. 3. We denote each time snapshot in spectrograms as a DFS profile. Specifically, a DFS profile D is a matrix with dimension as $F \times M$, where F is the number of sampling points in the frequency domain, and M is the number of transceiver links. Based on DFS profile from multiple links, we then derive domain-independent BVP.

5.2 From DFS to BVP

When a person performs a gesture, his body parts (e.g., two hands, two arms and the torso) move at different velocities. As a result, signals reflected by these body parts experience various DFS, which are superimposed at the receiver and form the corresponding DFS profile. As discussed in § 3, while DFS profile contains the information of the gesture, it is also highly specific to the domain. In contrast, the power distribution over physical velocity in the body coordinate system of the person, is only related to the characteristics of the gesture. Thus, in order to remove the impact of domain, BVP is derived out of DFS profiles.

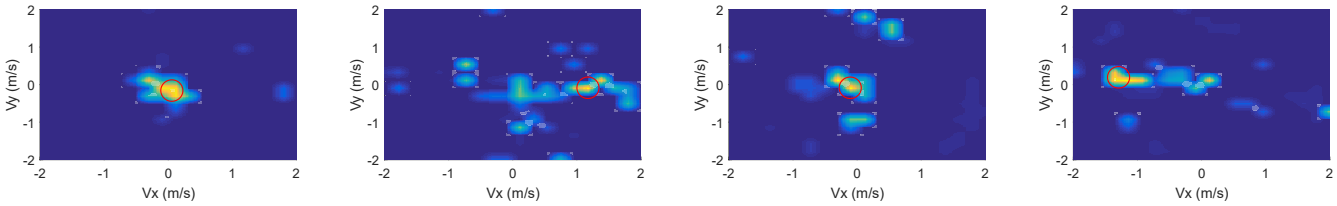
The basic idea of BVP is shown in Fig. 6. For practicality, a BVP V is quantized as a discrete matrix with dimension as $N \times N$, where N is the number of possible values of velocity components decomposed along each axis of the body coordinates. For convenience, we establish the local body coordinates whose origin is the location of the person and positive x-axis aligns with the orientation of the person. Currently, it is assumed that the global location and orientation of the person are available. Then the known global locations of wireless transceivers can be transformed into the local body coordinates. Thus, for better clarity, all locations and orientations used in the following derivation are in the local body coordinates. Suppose the locations of the transmitter and the receiver of the i-th link are $\vec{l}_t^{(i)} = (x_t^{(i)}, y_t^{(i)})$, $\vec{l}_r^{(i)} = (x_r^{(i)}, y_r^{(i)})$, respectively, then any velocity components $\vec{v}^{(b)} = (v_x^{(b)}, v_y^{(b)})$ in the human body coordinate will contribute its signal power to some frequency component, denoted as $f^{(i)}(\vec{v})$, in the DFS profile of the i-th link [17]:

$$f^{(i)}(\vec{v}^{(b)}) = a_x^{(i)} v_x^{(g)} + a_y^{(i)} v_y^{(g)}, \quad (3)$$

where $\vec{v}^{(g)} = (v_x^{(g)}, v_y^{(g)})$ is the velocity in global coordinate and is rotated from $\vec{v}^{(b)}$ with the human orientation. $a_x^{(i)}$ and $a_y^{(i)}$ are coefficients determined by locations of the transmitter and the receiver:

$$\begin{aligned} a_x^{(i)} &= \frac{1}{\lambda}\Big(\frac{x_t^{(i)}}{\|\vec{l}_t^{(i)}\|_2} + \frac{x_r^{(i)}}{\|\vec{l}_r^{(i)}\|_2}\Big), \\ a_y^{(i)} &= \frac{1}{\lambda}\Big(\frac{y_t^{(i)}}{\|\vec{l}_t^{(i)}\|_2} + \frac{y_r^{(i)}}{\|\vec{l}_r^{(i)}\|_2}\Big), \end{aligned} \quad (4)$$

where λ is the wavelength of Wi-Fi signal. As static components with zero DFS (e.g., the line of sight signals and dominant reflections from static objects) are filtered out before DFS profiles are calculated, only signals reflected by the person are retained. Besides, when the person is close to the Wi-Fi link, only signals with one time reflection have prominent magnitudes [18] as Fig. 3 shows. Thus, Equation 3 holds valid for the gesture recognition scenario.



(a) Stage 1: Start (b) Stage 2: Pushing (c) Stage 3: Stop (d) Stage 4: Pulling

Fig. 7. The BVP series of a pushing and pulling gesture. The main velocity component corresponding to the person's hand is highlighted with red circles in all snapshots.

From the geometric view, Equation 3 means that the 2-D velocity vector $\vec{v}$ is projected on a line whose direction vector is $d^{(i)} = (-a_y^{(i)}, a_x^{(i)})$. Suppose the person is on an ellipse curve whose foci are the transmitter and the receiver of the i-th link, then $d^{(i)}$ is indeed the normal direction of the ellipse at the person's location. Fig. 6 shows an example where the person generates three velocity components $\vec{v}_j, j = 1, 2, 3$, and projection of the velocity components on the DFS profiles of three links.

Since coefficients $a_x^{(i)}$ and $a_y^{(i)}$ only depend on the location of the i-th link, the relation of projection of the BVP on the i-th link is fixed. Specifically, an assignment matrix $A^{(i)}_{F\times N^2}$ can be defined:

$$A^{(i)}_{j,k} = \begin{cases} 1 & f_j = f^{(i)}(\vec{v}_k) \\ 0 & \text{else} \end{cases}, \tag{5}$$

where f_j is the j-th frequency sampling point in the DFS profile, and $\vec{v}_k$ is velocity component corresponding to the k-th element of the vectorized BVP V. Thus, the relation between DFS profile of the i-th link and the BVP can be modeled as:

$$D^{(i)} = c^{(i)} A^{(i)} V \tag{6}$$

where $c^{(i)}$ is the scaling factor due to propagation loss of the reflected signal.

5.3 BVP Estimation

How to recover BVP from DFS profiles of only several wireless links is another main challenge because the kinetic profile of a single gesture has hundreds of variables, posing the BVP estimation from DFS profiles as a severely underdetermined problem with only a limited number of constraints provided by several wireless links. Specifically, in practice, we estimate one BVP from DFS profiles calculated from 100 ms CSI data. Due to the uncertainty principle, the frequency resolution of DFS profiles is only about 10 Hz. Given that the range of human-induced DFS is within $\pm$ 60 Hz [11], the DFS profile of one link can only provide about 12 constraints. In contrast, we moderately set the range and the resolution of velocities along two axes of the body coordinates as $\pm$ 2 m/s and 0.2 m/s, respectively, leading to as much as 400 variables! Fortunately, when a person performs a gesture, only a few dominant distinct velocity components exist, due to the limited number of major reflecting multipath signals. Thus, there is an opportunity to correctly recover the BVP from DFS profiles of only several links.

Before a proper solution of BVP developed, it is necessary to understand the minimum number of links required to uniquely recover the BVP. Fig. 6 shows an intuitive example with three velocity components $v_j, j = 1, 2, 3$. With only the first two links (blue and green), the three velocity components create three power peaks in each DFS profile. However, when we recover the BVP, there are 9 candidates of velocity components, i.e. $v_j, j = 1, 2, 3$ and $u_k, k = 1, \cdots, 6$. And one can easily find an alternate solution, i.e. $\{u_1, u_3, u_6\}$, meaning that two links are insufficient.

By adding the third link (purple), it is able to resolve the ambiguity with high probability no matter how many velocity components exist, if no overlap of projections happens in the third DFS profile. When projections overlap, however, it is possible that adding the third or even more links cannot resolve the ambiguity. For example, suppose the third link in the Fig. 6 is in parallel with the y-axis, and there are three overlaps of projections (i.e. $\{u_1, v_2\}$, $\{v_3, u_4, u_6\}$ and $\{u_3, v_1\}$), then the ambiguous solution $\{u_1, u_3, u_6\}$ is still not resolvable. However, such ambiguity can hardly happen due to its stringent requirement on the distribution of velocity components as well as the orientation of the links. Moreover, we can further reduce the probability of the ambiguity by adding more links. We evaluate the impact of the number of links used by Widar3.0 on system performance in Section 7.5.

With observing of the sparsity of BVP and validating the feasibility of recovering BVP from multiple links, we adopt the idea of compressed sensing [49] and formulate the estimation of BVP as an l_0 optimization problem:

$$\min_V \sum_{i=1}^{M} |\text{EMD}(A^{(i)}V, D^i)| + \eta \|V\|_0, \tag{7}$$

where M is the number of Wi-Fi links. The sparsity of the number of the velocity components is coerced by the term $\eta\|V\|_0$, where η represents the sparsity coefficients and $\|\cdot\|_0$ is the number of non-zero velocity components.

$\text{EMD}(\cdot,\cdot)$ is the Earth Mover's Distance [50] between two distributions. The selection of EMD rather than Euclidean distance is mainly due to two reasons. First, the quantization of BVP introduces approximation error, i.e. projection of velocity components to the DFS bin might be adjacent to the true one. Such quantization error can be relieved by EMD, which takes the distance between bins into consideration. Second, there are unknown scaling factors between the BVP and DFS profiles, making the Euclidean distance inapplicable.

Fig. 7 shows an example of solved BVP series of a pushing and pulling gesture. The dominant velocity component from the hand and the coupling ones from the arm can be clearly observed.

5.4 Dynamic Link Selection

For the above BVP estimation algorithm, it is assumed that the reflected signals from each part of user's arms can be received by all the Wi-Fi receivers. However, this assumption may not hold valid due to the blockage of human torso. For example, when a user faces the transmitter and performs clap gesture, the receivers behind his body could hardly capture the reflected signal from his arms. Therefore, the signals received by these shadowed receivers embodies mostly the perturbations rather than motion features and should be dismissed for BVP estimation.

To deal with this problem, we leverage the prior-knowledge on devices deployment. Specifically, when a user intends to use the gesture recognition system, he or she would approach the monitoring area and halt to perform gestures. Widar3.0 leverages the antecedent movement of the person to estimates his location and orientation, which are the location and moving direction of the person at the end of the approaching track. Since existing works [18], [20], [28] have pushed the limit of Wi-Fi-based passive tracking application into decimeter level, Widar3.0 can exploit these approaches for location and orientation estimation. Based on the acquired location and orientation, Widar3.0 prunes those Wi-Fi receivers that may potentially be blocked by human torso and uses the rest devices for BVP estimation. Fig. 8 illustrates the diagram of the proposed dynamic link selection algorithm. With the facing direction as origin and clockwise as positive direction, we regard the sector ranging from -90° to $+90^\circ$ as visible area. Those receivers within this area are considered to be valid for BVP estimation and the others are considered to be shadowed. If a user performs gestures on the side of his body, it is deemed as another type of gesture. In this case, the dynamic link selection algorithm would be invalid.

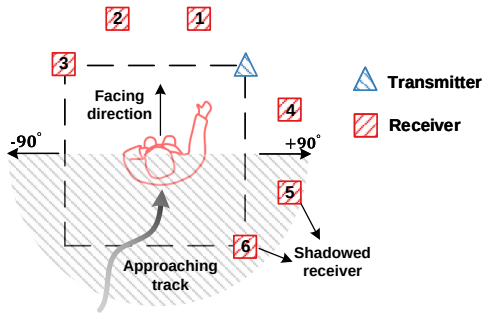


Fig. 8. Diagram of dynamic link selection algorithm.

6 Recognition Mechanism

In Widar3.0, we design a DNN learning model to mining the spatial-temporal characteristics of the BVP series. Fig. 9 illustrates the overall structure of the proposed learning model. Specifically, the BVP series is first normalized to remove irrelevant variations caused by instances, persons and hardware settings (§ 6.1). The normalized output is then inputted into a hybrid deep learning model, which from bottom to top consists of a convolutional neural network (CNN) for spatial feature extraction (§ 6.2) and a recurrent neural network (RNN) for temporal modeling (§ 6.3). The output feature vector of RNN is firstly fed into a KNN-based outlier detection algorithm described in § 6.4, which reports whether the captured motion is from the predefined legitimate gesture set. Those motions which pass the legality testification will be input into a Dense layer cascaded with a Softmax layer for classification.

The designed model is a result of the effectiveness of the domain-independent feature BVP. With BVP as input, the hybrid CNN-RNN model can achieve accurate cross-domain gesture recognition although the learning model itself does not possess generalization capabilities. We will verify that the CNN-RNN model is a simple but effective method in Section 7.4.

6.1 BVP Normalization

While BVP is theoretically only related to gestures, two practical factors may affect its stability as the gesture indicator. First, the overall power of BVP may vary due to the adjustment of transmission power. Second, in practice, instances of the same type of gesture performed by different persons may have different time length and moving velocities. Moreover, even instances performed by the same person may slightly vary. Thus, it is necessary to remove these irrelevant factors to retain the simplicity of the learning model.

For signal power variation, Widar3.0 normalizes the element values in each single BVP by adjusting the sum of all elements in BVP to 1. For instance variation, Widar3.0 normalizes the BVP series along the time domain. Specifically, Widar3.0 first sets the standard time length of gestures, denoted as t_0. Then, for a gesture with time length as t, Widar3.0 scales its BVP series to t_0. The assumption behind the scaling operation is that the total distance moved by each body part remains fixed. Thus, to change the time length of the BVP series, Widar3.0 first scales coordinates of all velocity components in the BVP by a factor of $\frac{t}{t_0}$, and then resamples the series to the sampling rate of the original BVP series. After normalization, the output becomes related to gestures only, and is input to the deep learning model.

6.2 Spatial Feature Extraction

The input of the learning model, BVP data, is similar to a sequence of images. Each single BVP describes the power distribution over physical velocity during a sufficiently short time interval. And the continuous BVP series illustrates how the distribution varies corresponding to a certain kind of action. Therefore, to fully understand the derived BVP data, it is intuitive to extract spatial features from each single BVP first and then model the temporal dependencies of the whole series.

CNN is a useful technique to extract spatial features and compress data [51], [52], and it is especially suitable for handling the single BVP, which is highly sparse but preserves spatial locality, as a velocity component usually corresponds to the same body part as its neighbors with

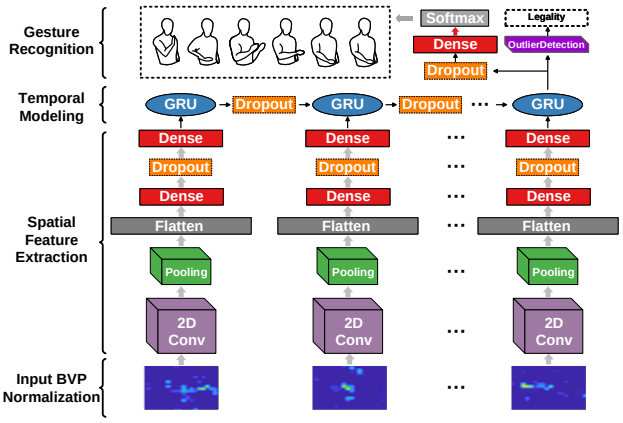


Fig. 9. Structure of gesture recognition model.

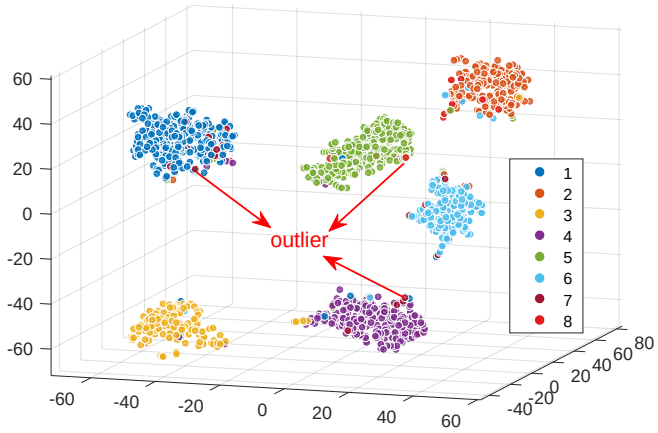


Fig. 10. t-SNE visualization of outlier and resident samples (#7 and #8 are outliers).

similar velocities. Specifically, the input BVP series, denoted as V, is a tensor with dimension as $N \times N \times T$, where T is the number of BVP snapshots. For the t-th sampling BVP, the matrix $V_{..t}$ is fed into the CNN. Within the CNN, 16 2-D filters are first applied to $V_{..t}$ to obtain local patterns in the velocity domain, which form the output $V^{(1)}_{..t}$. Then, max pooling is applied to $V^{(1)}_{..t}$ to down-sample the features and the output is denoted as $V^{(2)}_{..t}$. With $V^{(2)}_{..t}$ flattened into the vector $\vec{v}^{(2)}_{..t}$, two 64-unit dense layers with ReLU as activation functions are used to further extract features in a higher level. Note that one extra dropout layer is added between two dense layers to reduce overfitting. The final output $\vec{v}_{..t}$ characterizes the t-th sampling BVP. And the output series is used as the input of following recurrent layers for temporal modeling.

6.3 Temporal Modeling

Besides local spatial features within each BVP, BVP series also contains temporal dynamics of the gesture. Recurrent neural networks (RNN) are appealing in that they can model complex temporal dynamics of sequences. There are different types of RNN units, e.g., SimpleRNN, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) [53]. Compared with original RNNs, LSTMs and GRUs are more capable of learning long-term dependencies, and we choose GRUs because GRU achieves performance comparable to that of LSTM on sequence modeling, but involves fewer parameters and is easier to train with less data [53].

Specifically, Widar3.0 chooses single-layer GRUs to model the temporal relationships. Inputs $\{\vec{v}_{..t},\ t = 1, \cdots, T\}$ output from CNN are fed into GRUs and a 128-dimensional vector $\vec{v}_{..r}$ is generated. Furthermore, a dropout layer is added for regularization, and a softmax classifier with cross-entropy loss for category prediction is utilized. Note that for recognition systems which involve more sophisticated activities with longer durations, the GRU-based models can be transformed into more complex versions [51], [54]. In § 7.4, we will verify that single-layer GRUs are sufficient for capturing temporal dependencies for short-time human gestures.

6.4 Outlier Detection

The learning pipeline used in Widar3.0 is designed to perform recognition within the predefined gestures. However, when deploying the system for real-world applications, there is often little control over the testing gesture types, which may be unseen during training. Those novel gestures may lead to erroneous and confident predictions. This behavior can deteriorate user experience or even have serious consequences in medical or industrial scenes. Therefore, being able to accurately detect unseen examples can be practically important for gesture recognition tasks. From now on, we define the gestures in the predefined gesture set as *Resident Gestures* and those unseen gestures as *Outlier Gestures*.

It is documented [55] that resident samples have high density in the distribution of training datasets for neural networks and outlier samples have low density. A popular and intuitive strategy for detecting outliers is to train a generative model [56], [57] to approximate the density distribution of training datasets. However, this is not a favorable solution for Widar3.0 as we intend to build an agile system without extra training procedures. We hereby resort to the existing recognition pipeline in Fig. 9 and fully exploit its potential for outlier detection. Specifically, we would like to find a low-dimensional latent representation of the input BVP series that have analogous properties of density distribution to that of the high-dimensional BVP series (i.e., high density for residents and low density for outliers). In our implementation, the output of the RNN layer is a 128-dimensional tensor, which embodies sufficient gesture discriminations after spatial and temporal refining with CNN and RNN layers. We visualize the output tensors from 8 different gestures in Fig. 10, of which gesture #1 to #6 are resident gestures and gestures #7 and #8 are outlier gestures. Apparently, the condensed features of resident gestures are tightly clustered and the samples from outlier gestures occur mostly at the edge zone of resident clusters, which have a lower density than those in the kernel zone. This demonstrates the feasibility of density-based outlier detection with these latent features.

Essentially, our outlier detection algorithm is to identify the local density of the test sample and compare it with

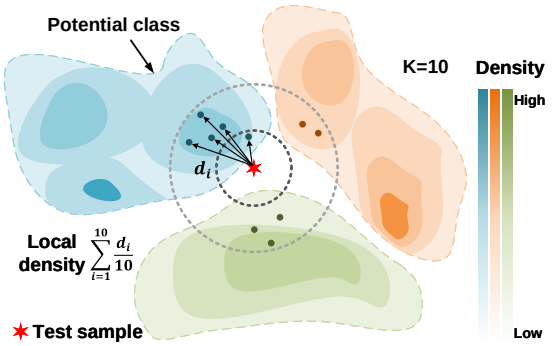


Fig. 11. Illustration of KNN-based outlier detection algorithm (K=10).

the density of the resident samples to determine its identity. We design a KNN-based method to achieve this purpose. Fig. 11 takes $K = 10$ as an example and illustrates the sketch of this algorithm. For each test sample, Widar3.0 first has to determine its *potential class* in the predefined gesture set by the KNN method. Specifically, Widar3.0 identifies the K-nearest neighbors of the test sample in the training set, among which the most prevalent gesture class is deemed as the *potential class*. This step is crucial as different classes in the resident dataset may have different density distributions. The following procedures only consider the resident samples in *potential class*. Then, Widar3.0 identifies the K-nearest neighbours of the test sample in the *potential class*, denoted as s_i, where $i = 1, ..., K$. Their distances to s_i are averaged to indicate the local density of the test sample, denoted as LD_K. Afterwards, Widar3.0 calculates the average distance of s_i to its K-nearest neighbours in *potential class* as d_i and obtains the **expected local density** of test sample as $ELD_K = \sum_{i=1}^{K} d_i/K$. Finally, Widar3.0 detects outliers by comparing the ratio $\rho = LD_K/ELD_K$ with a predefined threshold τ. All the distances are measured by Euclidean distance. To futher clarify the detection process, we present the pseudocode in Algorithm.1.

Algorithm 1 KNN-based outlier detection algorithm

Input: resident samples $s_i \in S$, test sample α, labels of resident samples $l_i \in L$, parameters K and τ.
Output: identity of α (resident or outlier)
(Euclidean distance is employed)

```
Determine potential class of α by performing KNN classification in S.
Reduce S into S̃ by removing the samples not belong to potential class.
Find K-nearest neighbours of α in S̃: s_i ∈ S̃, i = 1...K.
Get local density of α: LD_K = Σ_{i=1}^{K} ||α − s_i||_2 / K.
Calculate the average distance of s_i to its K-nearest neighbours in S̃: d_i, i = 1...K.
Get expected local density of α: ELD_K = Σ_{i=1}^{K} d_i / K.
Obtain outlier indicator: ρ = LD_K / ELD_K.
if ρ > τ then
    α is outlier
else
    α is resident
end if
```

We further investigate the impact of parameters K and τ on Algorithm.1 in § 7.5.

7 Evaluation

This section presents the implementation and detailed performance of Widar3.0.

7.1 Experiment Methodology

Implementation. Widar3.0 consists of one transmitter and at least three receivers. All transceivers are off-the-shelf mini-desktops (physical size 170mm × 170mm) equipped with an Intel 5300 wireless NIC. The cost for each NIC is approximately 5$ and the cost for each laptop is approximately 100$. We would like to emphasize that our system can be deployed on any ubiquitous wireless devices like personal laptop, desktop or even mobile phone, as long as the Wi-Fi NIC on them can support CSI logging. Linux CSI Tool [58] is installed on devices to log CSI measurements. Devices are set to work in the monitor mode, on channel 165 at 5.825 GHz where there are few interfering radios as interference does pose severe impacts on the collected CSI measurements [59]. The transmitter activates one antenna and broadcasts Wi-Fi packets at a rate of 1,000 packets per second. The receiver activates all three antennas which are placed in a line. We implement Widar3.0 in MATLAB and Keras [60].

Evaluation setup. To fully explore the performance of Widar3.0, we conduct extensive experiments on gesture recognition in 3 indoor environments: an empty classroom furnished with desks and chairs (Room 1), a spacious hall (Room 2) and an office room with furniture like sofa and tables (Room 3). Fig. 12 illustrates the general environmental features and the sensing area in different rooms. The size of classroom is 4.5m × 5.5m, the size of office is 2.5m × 4m and the size of hall room is 4.5m × 2.5m. Fig. 13 shows a typical example of the deployment of devices and domain configurations in the sensing area, which is a 2m × 2m square. Note that the 2m × 2m square is a typical setting to perform interactive gestures for recognition and response, especially in the scenario of smart home, with more Wi-Fi nodes incorporated into smart devices (e.g., smart TV, Xbox Kinect, home gateways, smart camera) to help. We assume that only the gesture performer is in the sensing area as moving entities introduce noisy reflection signals and further result in less accurate DFS profiles of the target gestures. Except for the two receivers and one transmitter placed at the corner of the sensing area, the remaining four receivers can be deployed at random locations outside two sides of the sensing area. As Section 5.3 has mentioned, the deployment of devices hardly pose impacts on Widar3.0 theoretically. All devices are held up at the height of 110 cm, where users with different heights can perform gestures comfortably. In total, 16 volunteers (12 males and 4 females) with different heights (varying from 185 cm to 155 cm) and different weights (varying from 44 kg to 89 kg) and somatotypes participate in experiments. The ages of the volunteers vary from 22 to 28.

Dataset. We collect gesture data from 5 locations and 5 orientations in each sensing area, as illustrated in Fig. 13. All

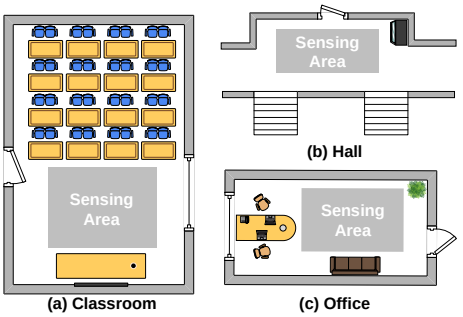


Fig. 12. Layouts of three evaluation environments.

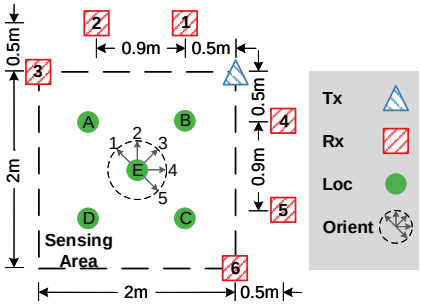


Fig. 13. A typical setup of devices and domains in one environment.

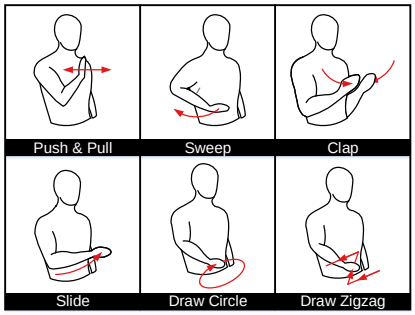


Fig. 14. Sketches of gestures evaluated in the experiment.

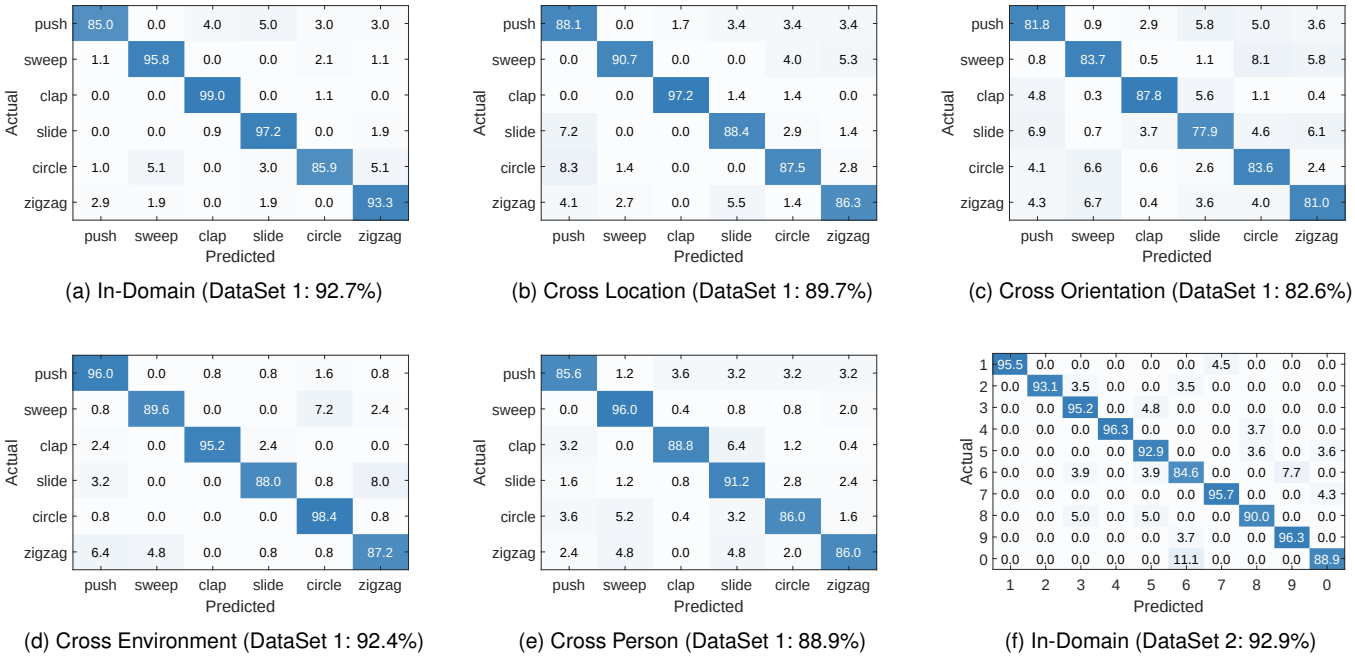


Fig. 15. Confusion matrices of different settings with two gesture datasets.

experiments are approved by our IRB. Two types of datasets are collected. Specifically, the first dataset (Dataset 1) consists of common hand gestures used in human-computer interaction, including pushing and pulling, sweeping, clapping, sliding, drawing circle and drawing zigzag. The sketches of the six gestures are plotted in Fig. 14. This dataset contains gesture samples of 16 users $\times$ 5 positions $\times$ 5 orientations $\times$ 6 gestures $\times$ 5 instances. Eight users contribute to 6,000 data samples collected in the classroom, five users contribute to 3,750 data samples collected in the hall and four users contribute to 3,000 data samples collected in the office. Among them, one user has data in both hall and office and the other users only have data in one room. The second dataset (Dataset 2) is collected for a case study of more complex and semantic gestures. Two volunteers (one male and one female) draw number $0 \sim 9$ in the horizontal plane, and a total of 5,000 samples (2 users $\times$ 5 positions $\times$ 5 orientations $\times$ 10 gestures $\times$ 10 instances) are collected. Before collecting the datasets, we ask volunteers to watch the example video of each gesture. The datasets and the example videos are available at website[1].

Prerequisites acquisition. The position and orientation of the user are prerequisites for the calculation of BVP. In general, the last estimation of location and the last estimation of moving direction can be provided by tracking systems [18], [20], [28], as the location and orientation of the user in Widar3.0. Note that the function of Widar3.0 is independent of that of the motion tracking system. To fully understand how Widar3.0 works, we record the ground truth of location and orientation of the user in most experiments, and explicitly introduce location and orientation error in the parameter study (Section 7.5) to evaluate the relation between recognition accuracy and location and orientation errors.

Model setting. The input shape of the DNN model is $20 \times 20 \times 22$, where the first two dimensions represent the numbers of velocity components along the x and the y axis respectively, and the third dimension represents the frames of snapshots over time. For the 2D convolutional layer, the

1. http://tns.thss.tsinghua.edu.cn/widar3.0/index.html

number of filters is set to 16, the kernel size is set to 5, and the activation function is set to ReLU. For the max-pooling layer, the kernel size is set to 2×2. The two fully connected layers all have 64 units and are activated by ReLU. All the dropout ratios are set to 0.5. For the GRU layer, the hidden units are set to 128 and only the last output is passed to the following layer. The RMSprop optimizer is adopted and the learning rate is set to 0.001. We train the model with a batch size of 32.

7.2 Overall Accuracy

Taking all domain factors into consideration, Widar3.0 achieves an overall accuracy of 92.7%, with 90 and 10 percentage data collected in Room 1 used for training and testing, respectively. Fig. 15a shows the confusion matrix of 6 gestures in dataset 1, and Widar3.0 achieves consistently high accuracy of over 85% for all gestures.

Fig. 15b, 15c, 15d and 15e further show confusion matrices considering each specific domain factors. For each domain factor, we calculate average accuracy of cases where one out of all domain instances are used for testing, while the rest domain instances are for training. The average accuracy over all gestures are provided as well, and it can be seen that Widar3.0 achieves consistent high performance across different domains, demonstrating its capability of cross-domain recognition.

We observe that for in-domain cases, the gestures "pushing and pulling", "drawing circle" and "drawing zigzag" usually correspond to a relatively lower accuracy. While the "pushing and pulling" gesture is the simplest one among all gestures, it is performed just in front of the user torso, and is more likely to be blocked from the perspectives of certain links, which results in less accurate BVP estimation as shown in the following experiments (Section 7.5). When users perform the gesture "drawing circle" or "drawing zigzag", the trajectory has significant changes in vertical direction. However, Widar3.0 is designed to extract BVP only in the horizontal plane, leading to information loss for the two gestures, and decrease in recognition accuracy. Similar gesture-specific performance can be observed from cross-domain results.

Case study. We now examine if Widar3.0 still works well for more complex gesture recognition tasks. In this case study, volunteers draw number 0$\sim$9 in the horizontal plane and 5,000 samples are collected in total. We divide the dataset into training and testing randomly with the ratio 9:1. As shown in Fig. 15f, Widar3.0 achieves satisfying results of over 90% for 8 gestures and the average accuracy is 92.9%.

7.3 Cross-Domain Evaluation

We now evaluate the overall performance of Widar3.0 on different domain factors, including environment, person diversity and location and orientation of the person. For evaluation on each domain factor, we keep the other domain factors unchanged, and perform leave-one-out cross-validation on the datasets. We also evaluate the performance when multiple domain factors change simultaneously, which is important for real-world deployment of Widar3.0. Besides, we evaluate the performance when user is in different dresses on different dates.

Location independence. The model is trained on the BVPs of random 4 locations, all 5 orientations and 8 people in Room 1. And the data collected at the last location in the same room is used for testing. As shown in Fig. 16, the average accuracies for all locations uninvolved in training are all above 85%. Widar3.0 achieves a best performance of 92.3% with location e, which is at the center of the sensing area, as the target domain. The accuracy descends to 85.3% when testing dataset is collected at location d, as wireless signal reflected by human-body becomes weaker after a longer distance of propagation, which leads to less accurate BVPs. In addition, BVP is modeled from signals reflected by the person. If the person happens to pass his arm through the line-of-sight path of any links, the accuracy will slightly drop, as proved by the result of location b.

Orientation sensitivity. In this experiment, we select each orientation as the target domain and other 4 orientations as the source domain. The red component in Fig. 17 shows that the accuracy remains above 80% for orientation 2, 3, 4 with all the 6 links involved. Compared with best target orientation 3, whose accuracy is around 90%, the performance at orientation 1&5 declines by over 10%. The reason is that gestures might be shadowed by human body in these two orientations and the number of effective wireless links for BVP generation decreases. To overcome this phenomenon, we adopt a dynamic link selection algorithm described in Section 5.4. The blue component in Fig. 17 shows a significant improvement when we select partial links for BVP estimation under different user orientations. We attribute this improvement to the relief of torso movement influences after pruning the shadowed Wi-Fi links.

Environment diversity. The accuracy across different environments is another significant criterion for performance of cross-domain recognition. In this experiment, gesture samples collected in room 1 are used as the training dataset, and three groups of gesture samples collected in three rooms are used as testing datasets. As Fig. 18 depicts, while the accuracy for different rooms slightly drops, the average accuracy preserves over 87% even if the environment changes totally. In a nutshell, Widar3.0 is robust to different environments.

Cross multiple domain factors. In this experiment, we evaluate the system performance when multiple domain factors change simultaneously. There are four combinations of the three domain factors $\{R/L, R/O, L/O, R/L/O\}$, in which the R, L and O represent Room, Location and Orientation respectively. From the previous experiments, we observe that Orientation factor has a more significant impact than the others. Hence, the redults for $\{R/O, L/O, R/L/O\}$ cases are presented with the respective orientations. For the $\{R/L\}$ case, we select each location in one room as the target domain and the other 4 locations in another room as the source domain. The accuracies over five locations are averaged to obtain an overall performance. For $\{R/O, L/O\}$ cases, we select each orientation (or location) in one room as the target domain and the other 4 orientations (or locations) in another room as the source domain. The accuracy over each orientation is reported separately. For the $\{R/L/O\}$ case, we select each orientation and location in one room as the target domain and the other 4 orientations and 4 locations in another room as the source domain. The accuracies over

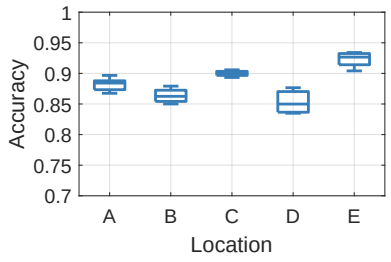


Fig. 16. Accuracy across different locations.

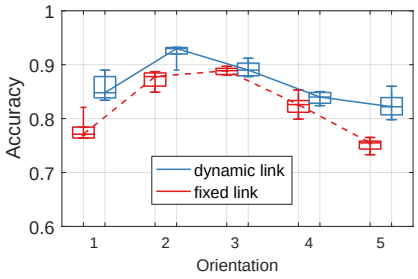


Fig. 17. Accuracy across different orientations.

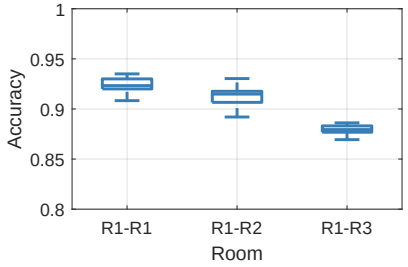


Fig. 18. Accuracy across different environments.

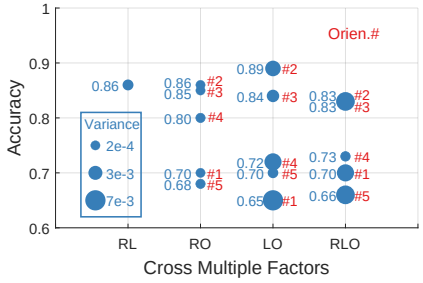


Fig. 19. Accuracy across multiple factors.

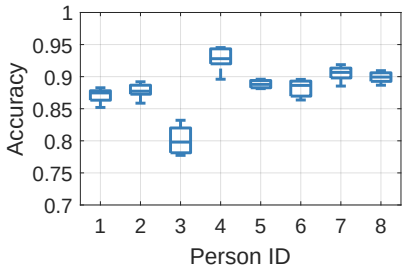


Fig. 20. Accuracy across different persons.

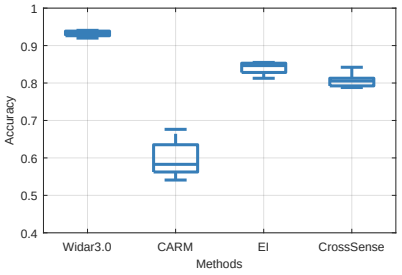


Fig. 21. Comparison of recognition approaches.

five locations are averaged and each orientation is reported separately. The results are shown in Fig. 19. In this figure, the latter three cases each has five markers to represent the performance over five orientations. When two factors change simultaneously, the performance decreases by about 2∼3% compared to one-factor experiments. When three factors change simultaneously, the performance further decreases by about 3∼4% compared to two-factor experiments. Even though some edge orientations impose a strong impact on system performance, this could be trivial in real-world deployment scenarios because the users are accustomed to perform gestures when facing the devices.

Person variety. Data collected from different persons may have discrepancy due to their various behavior patterns. Widar3.0 incorporates BVP normalization to alleviate this problem. To evaluate the performance of Widar3.0 on different users, we train the model on a dataset from every combination of 7 persons, and then test with the data of the resting person. Fig. 20 shows that the accuracy remains over 85% across 7 persons. The impact of the number of persons used in training the recognition model is further investigated in Section 7.5.

Performance on different dates.

Data collected from the same person with different dresses may be different due to the signal reflection on human clothes. In this experiment, we evaluate the system performance when users wear different dresses on different dates. Gesture samples collected from the classroom are used for training and samples collected in the hall from a user on date 1 as well as his samples collected in the office on date 2 are used for testing separately. Results shown in Table. 1 reveals a consistent performance regarding this factor. Essentially, BVP portrays the velocity distribution over human arms and the motion of clothes are synchronous to that of arms. Hence, Widar3.0 is resilient to date factor.

TABLE 1
Performance on different dates.

Accuracy	Date-1	Date-2
Mean	0.912	0.910
Variance	1.7e-4	1.5e-4

7.4 Method Comparison

This section compares the capability of cross-domain recognition with different methods, learning features and structures of learning networks. Some outlier detection methods are evaluated in comparison with our proposed KNN-based method. In the experiment, training and testing datasets are collected separately in Room 1 and 2.

Comparison with the state-of-the-arts works. We compare Widar3.0 against several alternative state-of-the-arts methodologies, CARM [11], EI [14] and CrossSense [15], where the latter two are feasible for cross-domain recognition. Specifically, CARM uses DFS profiles as learning features and adopts HMM model. EI incorporates an adversarial network and specializes the training loss to additionally exploit characteristics of unlabeled data in target domains. CrossSense proposes an ANN-based roaming model to translate signal features from source domains to target domains, and employs multiple expert models for gesture recognition. Fig. 21 shows the system performance of the four approaches. Widar3.0 achieves better performance with the state-of-the-art cross-domain learning methodologies, EI and CrossSense, and it does not require extra data from a new domain or model re-training. In contrast, both feature and learning model of CARM do not have cross-domain capability, which is the main reason for its significantly lower recognition accuracy.

Comparison of input features. We compare three types

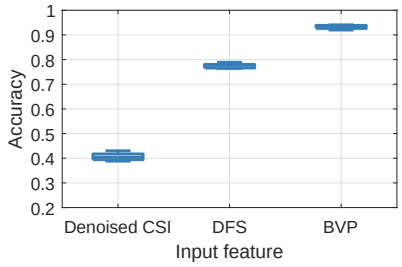


Fig. 22. Comparison of input features.

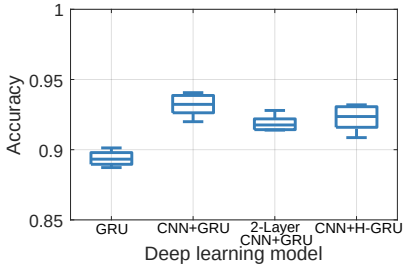


Fig. 23. Comparison of DNNs.

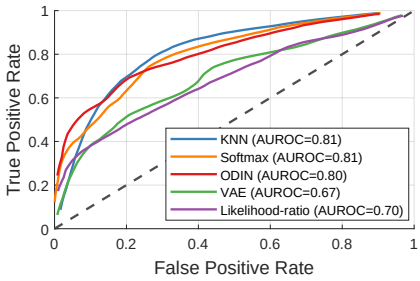


Fig. 24. Outlier detection performance.

of features with different levels of abstraction from raw CSI measurements, i.e. denoised CSI, DFS profiles and BVP, by feeding them into the CNN-GRU hybrid deep learning model, similar to that in Widar3.0. Specifically, the size of denoised CSI is 18 (the number of antennas of 6 receivers) $\times$ 30 (the number of subcarriers) $\times$ T (the number of time samples), and the DFS profile has the shape as 6 (the number of receivers) $\times$ F (the number of Doppler frequency samples) $\times$ T (the number of time samples). As shown in Fig. 22, BVP outperforms both denoised CSI and DFS, with an increase of accuracy by 52% and 15%, respectively. The performance improvement of BVP attributes its immunity to changes of layouts of transceivers, which however may significantly influence the other two types of features.

Comparison of learning model structures. Different deep learning models are further compared and the system performance is demonstrated in Fig. 23. Specifically, the CNN-GRU hybrid model increases the accuracy by around 5% compared with the simple GRU model which merely captures temporal dependencies. The former model benefits from representative high-level spatial features within each BVP snapshot. In addition, we also feed BVP into a two-convolutional-layer CNN-GRU hybrid model and a CNN-Hierarchical-GRU model [54]. It is shown that a more complex deep learning model does not promote the performance, demonstrating that BVP of different gestures are distinct enough to be discriminated by a simple but effective classifier.

Comparison of outlier detection methods. We compare the proposed KNN-based outlier detection algorithm with four other popular methods. 1) The Softmax [61] method recognizes the samples that have low prediction confidence on the Softmax layer as outliers. 2) The ODIN [62] method uses temperature scaling and input preprocessing technique to detect outliers based on the prediction confidence of Softmax layer. 3) The Variational Autoencoder (VAE) method [63] trains a VAE model on resident samples and employs the reconstruction loss during testing as outlier indicator. 4) The Likelihood-ratio method [64] contrasts the likelihood of a generative model against a background model and employs the likelihood ratio between the two models as outlier indicator. Methods 1) and 2) require no modification to the classification model (Fig.9). For methods 3) and 4), we build a VAE model [65] with three fully-connected layers as Encoder and another three fully-connected layers as Decoder, which have sizes of {1024, 512, 128} and {128, 512, 1024} respectively. The latent dimension is set to 16. For method 4), we perturb the training set by randomly rotating and scaling BVP frames along its central point to mimic the variations in gesture directions and speeds, which corrupts the semantic structure in the training data. The perturbed dataset is used to train the background model. The evaluation results are shown in Fig. 24 and Table. 2, in which the metrics have been introduced in Section 6.4. Numbers in front and inside of the brackets are mean and variance respectively based on 10 independent runs with random initialization of network parameters and random shuffling of training inputs. As can be seen, while the Likelihood-ratio method performs better than the VAE method, they are both worse than the other methods. We believe this is due to the intrinsic trait of Wi-Fi signals. Wi-Fi signal is a kind of radio frequency (RF) signal that has a wavelength larger than the roughness of human body, which causes the specular reflection effect when reflected [24]. Hence, the signals reflected from each part of human body would be tightly clustered around a specific direction. Consequently, partial BVP elements would be blanked due to the absence of the received signal on that direction. The distribution of blanked elements deviate significantly even when human performs the same gesture with subtle differences. Therefore, reconstruction loss would be significant. However, for the other three methods, they are based on the latent features extracted by CNN and RNN layers, which mitigate the spatial and temporal mutations caused by specular reflections. The results shows that generative model-based outlier detection methods are not favorable choices for wireless signal-based recognition tasks. For KNN, Softmax and ODIN methods, their performance are very close. Basically, our proposed KNN method exploits the feasibility of using latent features for outlier detection, which requires no extra training process or data collection compared to VAE and Likelihood-ratio methods. Besides, KNN method provides more information on the density distribution of input samples than the Softmax and ODIN methods and could be valuable to gain insight into the properties of datasets.

TABLE 2
Outlier detection performance.

Methods	AUROC↑	AUPRC↑	FPR80↓
KNN	0.81(3e-4)	0.79(7e-4)	0.31(2e-3)
Softmax	0.81(3e-4)	0.82(4e-4)	0.32(2e-3)
ODIN	0.80(7e-4)	0.83(5e-4)	0.40(6e-3)
VAE	0.67(3e-4)	0.69(4e-4)	0.60(3e-3)
Likelihood-ratio	0.70(5e-4)	0.74(3e-4)	0.57(2e-3)

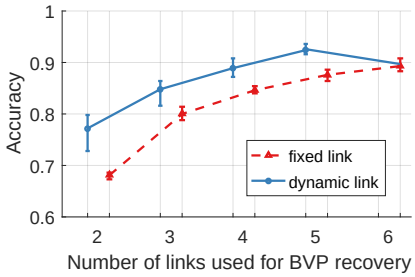


Fig. 25. Impact of link numbers.

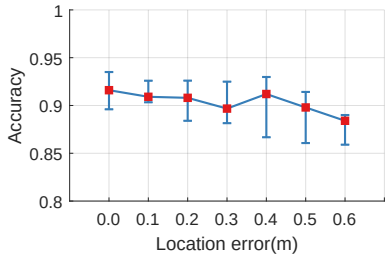


Fig. 26. Impact of location error.

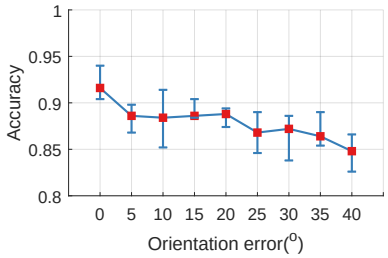


Fig. 27. Impact of orientation error.

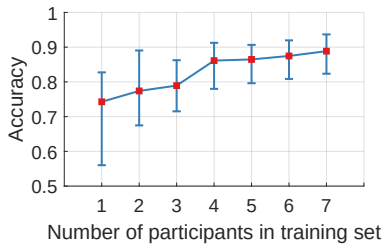


Fig. 28. Impact of training diversity.

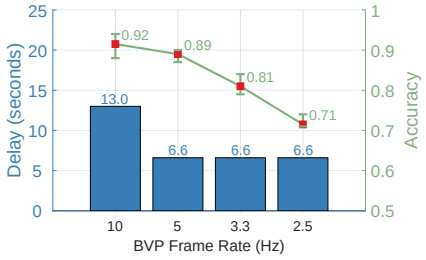


Fig. 29. System time overhead.

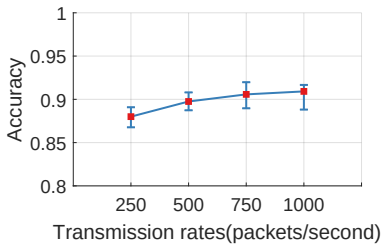


Fig. 30. Impact of CSI rate.

7.5 Parameter Study

Impact of link numbers. In the above experiments, 6 links are deployed for more accurate estimation of BVP. This section studies the impact of the number of links on system performance. In Fig. 25, the red dashed line indicates the performance with fixed link numbers and positions and the blue line indicates the performance with the dynamically selected partial links for different users' orientations as is described in Section 7.1. With fixed link selection, the accuracy gradually decreases as the number of links reduces from 6 to 3, but experiences a more significant drop when only two links are used. The main reason is that some BVPs cannot be correctly recovered with only 2 links considering the ambiguity mentioned in Section 5.3, and gestures at certain locations or orientations cannot be fully captured due to blockage. With dynamic link selection, accuracy grows significantly compared to that with the same number of fixed links. Moreover, even fewer links can outperform that with fixed links selection. For example, when dynamically picking 5 links, the averaged accuracy is 0.917, but when consistently using all the 6 links, the averaged accuracy is 0.895. This phenomenon indicates that the captured gesture-irrelevant information from the shadowed links deteriorates the system performance to some extent.

Impact of location and orientation estimation error. Localizations and orientations provided by Wi-Fi-based motion tracking systems usually have errors of about several decimeters and 20 degrees, respectively. Thus, it is necessary to understand how these errors impact the performance of Widar3.0. Specifically, we record ground truth of location and orientation, and calculate errors where gestures are performed. On one hand, as shown in Fig. 26, the overall accuracy remains over 90% when the location error is within 40 cm, but then drops as the error further increases. The significant decrease in accuracy at 0.3m is potentially due to the line-of-sight blockage at that circumstance. On the other hand, Fig. 27 shows that the overall accuracy gradually drops with more deviation of orientation. While the tracking errors negatively impact the performance of Widar3.0, taking practical location and orientation errors into consideration, we believe existing motion tracking works can still provide location and orientation results with acceptable accuracy.

Impact of training set diversity. This experiment studies how the number of volunteers in training dataset impacts the performance. Specifically, a varying number of volunteers from 1 to 7 participate in collecting the training dataset, and data from another new person is used to test Widar3.0. Fig. 28 shows that the average gesture recognition accuracy decreases from 89% to 74% when the number of people for training varies from 7 to 1. The reasons come from two folds. First, with the training dataset contributed by fewer volunteers, the deep learning model will be less thoroughly trained. Second, the behavior difference between testing persons and training persons will be amplified even if we have adopted BVP normalization. In general, Widar3.0 promises an accuracy of over 85% with more than 4 people in the training set.

System time overhead. In this experiment, we evaluate the system time overhead. Basically, the calculation process of Widar3.0 consists of three major parts: BVP extraction, outlier detection and gesture inference. We noticed that the outlier detection and gesture inference are both performed within several milliseconds, which could be dismissed for practical use. For BVP extraction, the designed algorithm is a compressed sensing-based estimation process and BVP frames for each gesture are estimated sequentially, which would incur significant time complexity. To evaluate the time overhead, we take a gesture instance that lasts for 1.5 seconds as an example. We run BVP extraction algorithm (implemented in MATLAB) on a workstation with 6 CPU cores (12 virtual cores) working at 2.3 GHz frequency and

with CentOS system installed. We use the parallel computing toolbox provided by MATLAB to create 12 (maximum for hardware) parallel processes, each of which calculates one frame of BVP. Fig. 29 demonstrates the results. When frame rate of BVP is set to 10 Hz, a total of 15 frames are to be calculated for this gesture and the time consumption is 13 seconds. When frame rate is less than 5 Hz, the parallel processes only need one round processing to obtain the BVP frames for this gesture, which happens within 6.6 seconds. For accuracy evaluation, the in-domain recognition accuracy is evaluated with different sampling rates of BVP. As is shown, accuracy slightly decreases from 92% to 89% when frame rate reduces to 5 Hz and tumbles to 71% when frame rate reduces to 2.5 Hz. In general, a 6.6 seconds delay for a 1.5 seconds gesture is applicable for various application scenarios. Besides, GPU computing technique could be applied to further reduce the system time consumption.

Impact of CSI rate. In this experiment, we evaluate the performance of Widar3.0 with regard to different CSI transmission rates. We collect CSI measurements at the initial transmission rate of 1,000 packets per second, and down-sample the CSI series to 750 Hz, 500 Hz, 250 Hz. Note that this is different from BVP frame rate evaluation discussed in **System time overhead**. When changing CSI rate, the CSI samples used to estimate each BVP frame is changed accordingly. For example, for 1,000 Hz CSI and 10 Hz BVP, every 100 samples of CSI snapshots are used to estimate one BVP frame and for 500 Hz CSI and 10 Hz BVP, every 50 samples of CSI snapshots are used to estimate one BVP frame. Fig. 30 shows that the accuracy degrades slightly by around 4% when the sampling rate drops to 250Hz, and remains over 85% for all cases. In addition, Widar3.0 can further reduce the impacts on communication with shorter packets used as only CSI measurements are useful for the recognition tasks.

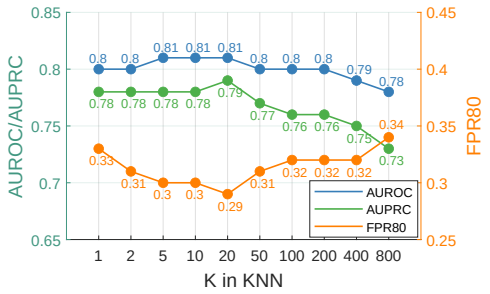


Fig. 31. Outlier detection performance for different K values.

Impact of parameters to outlier detection algorithm. In Algorithm.1, parameters K and τ should be determined beforehand. We select the most proper parameters by performing evaluations on this algorithm. We adopt dataset 1 (described in Section. 7.1) to be resident set, and collect extra 400 instances of outlier samples when a user performs arbitrary gestures. Three metrics are used to depict the outlier detection performance [64], which are the area under the ROC curve (AUROC), the area under the precision-recall curve (AUPRC), and the false positive rate at 80% true positive rate (FPR80). Fig. 31 presents the results. As is shown, when K increases from 1 to 20, AUROC and AUPRC grow slightly. This is because a very small K value could hardly capture a stable local density and is susceptible to variance. When K increase from 20 to 800, the performance reduces significantly, which is due to the uneven distribution of the density distributions and a very large K would capture a global density rather than a local density. In Widar3.0, we adopt a consistent $K = 20$ and $\tau = 1.188$, which corresponds to a False Positive Rate of 15%. Given that different resident classes have different density distributions, we suggest to rerun the parameter determination precess whenever the training dataset changes.

8 Discussions

User height. Since transceivers are placed at the same height, CSI measurements mainly capture the horizontal velocity components. Thus, different user heights may impact the recognition performance of Widar3.0, as the devices may observe different groups of velocity components intercepted at this height. However, Widar3.0 still has the capability of recognizing gestures in 3-D space, as common gestures remain their uniqueness even within the fixed height. As shown in the experiments, Widar3.0 is able to recognize gestures "draw circle" and "draw zigzag", which both contain vertical velocity components due to the fixed length of arms. By regarding the person as on an ellipsoid whose foci are the transceivers of a link, the BVP can be further generalized to 3-D space. Further work includes optimizing the deployment of Wi-Fi links to enable calculation of 3-D BVP and revising the learning model with 3-D BVPs as input.

Number of Wi-Fi links for gesture recognition. Although three wireless links are sufficient to resolve the ambiguity with a high probability for BVP generation, six receivers in total are deployed in the experiments. The reasons are two folds. First, compared with macro activities, the reflected signal of micro gestures is much weaker, since the effective area of hand and arm is much smaller than that of torso and leg, resulting in less prominent DFS profiles. Second, gestures with hands and arms may be opportunistically shadowed by other body parts when the user faces away from the link. For macro activities such as walking, running, jumping, and falling, it is believed that the number of Wi-Fi links required for recognition can be reduced. It is worth noting that Widar3.0 does not require the fixed deployment of Wi-Fi devices in the environment, as BVP is the power distribution over absolute velocities.

Applications beyond gesture recognition. While Widar3.0 is a Wi-Fi-based gesture system, the feature used in Widar3.0, BVP, can theoretically capture movements over the whole body of the person, and thus is envisioned to be used in other device-free sensing scenarios, such as macro activity recognition, gait analysis, and user identification. In these scenarios where users are likely to continuously change their locations and orientations, BVP calculation and motion tracking approaches can be intermittently invoked to obtain BVPs along the whole trace, which then may serve as a unique indicator for the user's activity or identity.

9 Limitations and Future Work

Widar3.0 takes a significant step towards Wi-Fi-based environment-independent gesture recognition with zero hu-

man effort, and there is room for continued research in various perspectives.

Requirement for Line-of-Sight (LOS). Widar3.0 relies on the existence of propagation paths going directly from Wi-Fi transmitter to human and from human to Wi-Fi receivers (i.e., the LOS between Wi-Fi devices and humans). This is because Widar3.0 requires the LOS paths to geometrically establish the relationship between signal propagation and the moving speed of reflectors. Even though the LOS condition is common when performing interaction with smart devices in smart homes, Widar3.0 will fail to work when LOS is not guaranteed in some other scenarios. To push Widar3.0 into a broader application prospect, future works can devise speed estimation algorithms that are resilient to multipath effect and signal occlusions. For example, recent works [66], [67] build statistical electromagnetic signal models that bridge the gap between speed and CSI measurements for rich-scattering and occluded environments. Similar ideas can be borrowed to extend Widar3.0 to NLOS cases.

Constrained user's presentation. To obtain a distinct Doppler Spectrum for BVP estimation, Widar3.0 requires the users to stand within a predefined zone with some flexibility on locations and facing orientations. This experimental setup is typical for some application scenarios where the users stand in front of the devices and perform gestures for interaction. Even though Widar3.0 has the potential for more generalized applications, there still exists a gap between current evaluation and real-world applicability. Future works include investigating the capability of BVP for more ubiquitous sensing tasks. For example, the maximum sensing range and valid facing orientations that can be represented by BVP without ambiguity are interesting research topics.

Loss of vertical resolution. Widar3.0 deploys Wi-Fi devices at the same height during experiments. However, this setting will abandon the resolution on the vertical direction. Concretely, the BVP only captures the velocity components projected in the horizontal plane. Without a vertical resolution, users with different heights will create bias in the BVP representation and eventually deteriorate system performance. Besides, human gestures are performed in three-dimensional space and will lose their uniqueness without vertical resolution. Even though we evaluated the system performance with 16 users and two distinct sets of gestures, future works are needed to investigate the representation capability of BVP to accommodate a more relaxed experimental regime. More importantly, the two-dimensional BVP can be extended to a three-dimensional BVP by reformulating the BVP model and repositioning the Wi-Fi devices.

10 Conclusion

In this paper, we propose a Wi-Fi-based zero-effort cross-domain gesture recognition system. First, we model the quantitative relation between complex gestures and CSI dynamics, and extract velocity profiles of gestures in body coordinates, which are domain-independent and act as unique indicators of gestures. Then, we develop a one-fits-all deep learning model to fully exploit spatial-temporal characteristics of BVP for gesture recognition. We implement Widar3.0 on COTS Wi-Fi devices and evaluate it in real environments. Experimental results show that Widar3.0 achieves high recognition accuracy across different domain factors, specifically, 89.7%, 82.6%, 92.4% and 88.9% for user's location, orientation, environment and user diversity, respectively. Future work focuses on applying Widar3.0 to fortify various sensing applications.

11 Acknowledgement

This work is supported in part by the NSFC under grant 61832010, 61972131.

References

[1] G. Gkioxari, R. Girshick, P. Dollár, and K. He, "Detecting and recognizing human-object interactions," in *Proceedings of IEEE CVPR*, Salt Lake City, UT, USA, 2018.
[2] M. Wang, B. Ni, and X. Yang, "Recurrent modeling of interaction context for collective activity recognition," in *Proceedings of IEEE CVPR*, Honolulu, HI, USA, 2017.
[3] T. Li, Q. Liu, and X. Zhou, "Practical human sensing in the light," in *Proceedings of ACM MobiSys*, Singapore, Singapore, 2016.
[4] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, vol. 46, no. 3, pp. 33:1–33:33, January 2014.
[5] S. Shen, H. Wang, and R. Roy Choudhury, "I am a smartwatch and i can track my user's arm," in *Proceedings of ACM MobiSys*, Singapore, Singapore, 2016.
[6] Y. Guan and T. Plötz, "Ensembles of deep lstm learners for activity recognition using wearables," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 2, pp. 11:1–11:28, June 2017.
[7] R. Nandakumar, A. Takakuwa, T. Kohno, and S. Gollakota, "Covertband: Activity information leakage using music," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 87:1–87:24, September 2017.
[8] K. Kalgaonkar and B. Raj, "One-handed gesture recognition using ultrasonic doppler sonar," in *Proceedings of IEEE ICASSP*, Taipei, Taiwan, 2009.
[9] K. Yatani and K. N. Truong, "Bodyscope: A wearable acoustic sensor for activity recognition," in *Proceedings of ACM UbiComp*, Pittsburgh, PA, USA, 2012.
[10] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: Device-free location-oriented activity identification using fine-grained wifi signatures," in *Proceedings of ACM MobiCom*, Maui, HI, USA, 2014.
[11] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Device-free human activity recognition using commercial wifi devices," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1118–1131, May 2017.
[12] H. Abdelnasser, M. Youssef, and K. A. Harras, "Wigest: A ubiquitous wifi-based gesture recognition system," in *Proceedings of IEEE INFOCOM*, Kowloon, Hong Kong, 2015.
[13] R. H. Venkatnarayan, G. Page, and M. Shahzad, "Multi-user gesture recognition using wifi," in *Proceedings of ACM MobiSys*, Munich, Germany, 2018.
[14] W. Jiang, C. Miao, F. Ma, S. Yao, Y. Wang, Y. Yuan, H. Xue, C. Song, X. Ma, D. Koutsonikolas, W. Xu, and L. Su, "Towards environment independent device free human activity recognition," in *Proceedings of ACM MobiCom*, New Delhi, India, 2018.
[15] J. Zhang, Z. Tang, M. Li, D. Fang, P. T. Nurmi, and Z. Wang, "Crosssense: Towards cross-site and large-scale wifi sensing," in *Proceedings of ACM MobiCom*, New Delhi, India, 2018.
[16] A. Virmani and M. Shahzad, "Position and orientation agnostic gesture recognition using wifi," in *Proceedings of ACM MobiSys*, Niagara Falls, NY, USA, 2017.
[17] K. Qian, C. Wu, Z. Yang, Y. Liu, and K. Jamieson, "Widar: Decimeter-level passive tracking via velocity monitoring with commodity wi-fi," in *Proceedings of ACM MobiHoc*, Chennai, India, 2017.
[18] K. Qian, C. Wu, Y. Zhang, G. Zhang, Z. Yang, and Y. Liu, "Widar2.0: Passive human tracking with a single wi-fi link," in *Proceedings of ACM MobiSys*, Munich, Germany, 2018.

[19] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Zero-effort cross-domain gesture recognition with wi-fi," in *Proceedings of MobiSys*, New York, NY, USA, 2019.

[20] X. Li, D. Zhang, Q. Lv, J. Xiong, S. Li, Y. Zhang, and H. Mei, "Indotrack: Device-free indoor human tracking with commodity wi-fi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, pp. 72:1–72:22, September 2017.

[21] F. Adib, Z. Kabelac, D. Katabi, and R. C. Miller, "3d tracking via body radio reflections," in *Proceedings of USENIX NSDI*, Seattle, WA, USA, 2014.

[22] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via rf body reflections," in *Proceedings of USENIX NSDI*, Oakland, CA, USA, 2015.

[23] K. Joshi, D. Bharadia, M. Kotaru, and S. Katti, "Wideo: Fine-grained device-free motion tracing using rf backscatter," in *Proceedings of USENIX NSDI*, Oakland, CA, USA, 2015.

[24] F. Adib, C.-Y. Hsu, H. Mao, D. Katabi, and F. Durand, "Capturing the human figure through a wall," *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 219:1–219:13, November 2015.

[25] F. Adib and D. Katabi, "See through walls with wi-fi!" in *Proceedings of ACM SIGCOMM*, Hong Kong, China, 2013.

[26] X. Li, S. Li, D. Zhang, J. Xiong, Y. Wang, and H. Mei, "Dynamic-music: Accurate device-free indoor localization," in *Proceedings of ACM UbiComp*, Heidelberg, Germany, 2016.

[27] C. Wu, F. Zhang, Y. Fan, and K. J. R. Liu, "Rf-based inertial measurement," in *ACM SIGCOMM*, 2019.

[28] J. Wang, H. Jiang, J. Xiong, K. Jamieson, X. Chen, D. Fang, and B. Xie, "Lifs: Low human-effort, device-free localization with fine-grained subcarrier information," in *Proceedings of ACM MobiCom*, New York City, NY, USA, 2016.

[29] M. Bocca, O. Kaltiokallio, N. Patwari, and S. Venkatasubramanian, "Multiple target tracking with rf sensor networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1787–1800, August 2014.

[30] W. Wang, A. X. Liu, and M. Shahzad, "Gait recognition using wifi signals," in *Proceedings of ACM UbiComp*, Heidelberg, Germany, 2016.

[31] Y. Zeng, P. H. Pathak, and P. Mohapatra, "Wiwho: Wifi-based person identification in smart spaces," in *Proceedings of ACM/IEEE IPSN*, Vienna, Austria, 2016.

[32] D. Huang, R. Nandakumar, and S. Gollakota, "Feasibility and limits of wi-fi imaging," in *Proceedings of ACM MobiSys*, Bretton Woods, NH, USA, 2014.

[33] B. Fang, N. D. Lane, M. Zhang, A. Boran, and F. Kawsar, "Bodyscan: A wearable device for contact-less radio-based sensing of body-related activities," in *Proceedings of ACM MobiSys*, Singapore, Singapore, 2016.

[34] B. Fang, N. D. Lane, M. Zhang, and F. Kawsar, "Headscan: A wearable system for radio-based sensing of head and mouth-related activities," in *Proceedings of ACM/IEEE IPSN*, Vienna, Austria, 2016.

[35] H. Li, W. Yang, J. Wang, Y. Xu, and L. Huang, "Wifinger: Talk to your smart devices with finger-grained gesture," in *Proceedings of ACM UbiComp*, Heidelberg, Germany, 2016.

[36] Y. Ma, G. Zhou, S. Wang, H. Zhao, and W. Jung, "Signfi: Sign language recognition using wifi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 1, pp. 23:1–23:21, March 2018.

[37] K. Niu, F. Zhang, J. Xiong, X. Li, E. Yi, and D. Zhang, "Boosting fine-grained activity sensing by embracing wireless multipath effects," in *Proceedings of ACM CoNEXT*, Heraklion/Crete, Greece, 2018.

[38] M. Zhao, Y. Tian, H. Zhao, M. A. Alsheikh, T. Li, R. Hristov, Z. Kabelac, D. Katabi, and A. Torralba, "Rf-based 3d skeletons," in *Proceedings of ACM SIGCOMM*, Budapest, Hungary, 2018.

[39] M. Zhao, T. Li, M. Abu Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi, "Through-wall human pose estimation using radio signals," in *Proceedings of IEEE CVPR*, Salt Lake City, UT, USA, 2018.

[40] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Recognizing keystrokes using wifi devices," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1175–1190, May 2017.

[41] K. Qian, C. Wu, Z. Zhou, Y. Zheng, Z. Yang, and Y. Liu, "Inferring motion direction using commodity wi-fi for interactive exergames," in *Proceedings of ACM CHI*, Denver, CO, USA, 2017.

[42] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proceedings of ACM MobiCom*, Miami, FL, USA, 2013.

[43] Y. Zhang, Y. Zheng, G. Zhang, K. Qian, C. Qian, and Z. Yang, "Gaitid: Robust wi-fi based gait recognition," in *Proceedings of Springer WASA*, 2020.

[44] J. Wang, Y. Chen, L. Hu, X. Peng, and P. S. Yu, "Stratified transfer learning for cross-domain activity recognition," in *Proceedings of IEEE PerCom*, Big Island, HI, USA, 2017.

[45] Z. Zhao, Y. Chen, J. Liu, Z. Shen, and M. Liu, "Cross-people mobile-phone based activity recognition," in *Proceedings of IJCAI*, Barcelona, Spain, 2011.

[46] K. Chen, L. Yao, D. Zhang, X. Chang, G. Long, and S. Wang, "Distributionally robust semi-supervised learning for people-centric sensing," in *Proceedings of AAAI*, New Orleans, LA, USA, 2018.

[47] R. Shu, H. H. Bui, H. Narui, and S. Ermon, "A dirt-t approach to unsupervised domain adaptation," in *Proceedings of ICLR*, Vancouver, Canada, 2018.

[48] Z. Yang, Z. Zhou, and Y. Liu, "From rssi to csi: Indoor localization via channel response," *ACM Comput. Surv.*, vol. 46, no. 2, pp. 25:1–25:32, November 2013.

[49] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[50] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, November 2000.

[51] S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher, "Deepsense: A unified deep learning framework for time-series mobile sensing data processing," in *Proceedings of ACM WWW*, Perth, Australia, 2017.

[52] C. Liu, L. Zhang, Z. Liu, K. Liu, X. Li, and Y. Liu, "Lasagna: Towards deep hierarchical understanding and searching over mobile sensing data," in *Proceedings of ACM MobiCom*, New York City, NY, USA, 2016.

[53] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *CoRR*, vol. abs/1412.3555, 2014.

[54] J. Chung, S. Ahn, and Y. Bengio, "Hierarchical multiscale recurrent neural networks," *CoRR*, vol. abs/1609.01704, 2016.

[55] C. M. Bishop, "Novelty detection and neural network validation," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 141, no. 4, pp. 217–222, 1994.

[56] H. Choi, E. Jang, and A. A. Alemi, "Waic, but why? generative ensembles for robust anomaly detection," 2018.

[57] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan, "Hybrid models with deep and invertible features," 2019.

[58] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 1, pp. 53–53, January 2011.

[59] Y. Zheng, C. Wu, K. Qian, Z. Yang, and Y. Liu, "Detecting radio frequency interference for csi measurements on cots wifi devices," in *Proceedings of IEEE ICC*, Paris, France, 2017.

[60] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[61] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," 2016.

[62] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," 2017.

[63] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *SNU Data Mining Center, Tech. Rep*, pp. 1–18, 2015.

[64] J. Ren, P. J. Liu, E. Fertig, J. Snoek, R. Poplin, M. A. DePristo, J. V. Dillon, and B. Lakshminarayanan, "Likelihood ratios for out-of-distribution detection," 2019.

[65] Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, and L. Carin, "Variational autoencoder for deep learning of images, labels and captions," in *Proceedings of NIPS*, 2016.

[66] F. Zhang, C. Chen, B. Wang, and K. J. R. Liu, "Wispeed: A statistical electromagnetic approach for device-free indoor speed estimation," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2163–2177, 2018.

[67] C. Wu, F. Zhang, Y. Hu, and K. J. R. Liu, "Gaitway: Monitoring and recognizing gait speed through the walls," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2020.

Yi Zhang received the BE degree from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, in 2017. He is currently working toward the PhD degree in the School of Software, Tsinghua University. He is a member of the Beijing National Research Center for Information Science and Technology. His research interests include wireless sensing, mobile computing and artificial intelligence. He is a student member of the IEEE.

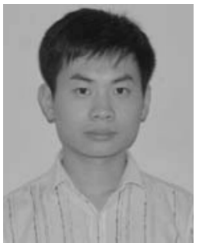
Yue Zheng received the BE degree from Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2015. She is currently a PhD student in Department of Electronic Engineering and School of Software at Tsinghua University. She is a member of the Beijing National Research Center for Information Science and Technology. Her research interests include wireless networks and mobile computing.

Kun Qian received the BE degree from the School of Software, Tsinghua University, in 2014. He is currently working toward the PhD degree in the School of Software, Tsinghua University. He is a member of the Beijing National Research Center for Information Science and Technology. His research interests include wireless networks and mobile computing. He is a student member of the IEEE.

Guidong Zhang received the BE degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China, in 2018. He is currently working toward the PhD degree in the School of Software, Tsinghua University. His research interests include wireless sensing and mobile computing. He is a student member of IEEE and ACM.

Yunhao Liu received the B.S. degree from Automation Department, Tsinghua University, and the M.S. and Ph.D. degrees in computer science and engineering from Michigan State University. He is currently an MSU Foundation Professor and the Chairperson of the Department of Computer Science and Engineering, Michigan State University. His research interests include sensor network and the IoT, localization, RFID, distributed systems, and cloud computing. He is a Fellow of the ACM.

Chenshu Wu is currently an Assistant Professor in the Department of Computer Science, The University of Hong Kong. He is also the Chief Scientist at Origin Wireless Inc. He received his B.E. degree in the School of Software in 2010 and Ph.D. degree in Computer Science in 2015, both from Tsinghua University, Beijing, China. His research focuses on wireless AIoT systems at the intersection of wireless sensing, ubiquitous computing, digital health, and the Internet of Things. He is a Senior Member of the IEEE and a member of the ACM.



Zheng Yang received the BE degree in computer science from Tsinghua University, in 2006 and the PhD degree in computer science from the Hong Kong University of Science and Technology, in 2010. He is currently an associate professor with Tsinghua University. His main research interests include wireless ad-hoc/sensor networks, and mobile computing. He is a member of the IEEE and the ACM.