Computer Networks 258 (2025) 111020

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

BeamSense: Rethinking Wireless Sensing with MU-MIMO Wi-Fi Beamforming Feedback

Khandaker Foysal Haque [a,*,1], Milin Zhang [a,1], Francesca Meneghello [b,2], Francesco Restuccia [a,3]

[a] Institute for the Wireless Internet of Things, Northeastern University, United States
[b] Department of Information Engineering, University of Padova, Italy

ARTICLE INFO

Dataset link: https://ieee-dataport.org/documents/dataset-human-activity-classification-mu-mimo-bfi-and-csi

Keywords:
Wi-Fi sensing
IEEE 802.11ac
SU-MIMO
MU-MIMO
Beamforming
Beamforming feedback angles

ABSTRACT

In this paper, we propose BeamSense, a completely novel approach to implement standard-compliant Wi-Fi sensing applications. Existing work leverages the manual extraction of the uncompressed channel state information (CSI) from Wi-Fi chips, which is not supported by the 802.11 standards and hence requires the usage of specialized equipment. On the contrary, BeamSense leverages the standard-compliant compressed beamforming feedback information (BFI) (beamforming feedback angles (BFAs)) to characterize the propagation environment. Conversely from the uncompressed CSI, the compressed BFAs (i) can be recorded without any firmware modification, and (ii) simultaneously captures the channels between the access point and all the stations, thus providing much better sensitivity. BeamSense features a novel cross-domain few-shot learning (FSL) algorithm for human activity recognition to handle unseen environments and subjects with a few additional data samples. We evaluate BeamSense through an extensive data collection campaign with three subjects performing twenty different activities in three different environments. We show that our BFAs-based approach achieves about 10% more accuracy when compared to CSI-based prior work, while our FSL strategy improves accuracy by up to 30% when compared with state-of-the-art cross-domain algorithms. Additionally, to demonstrate its versatility, we apply BeamSense to another smart home application – gesture recognition – achieving over 98% accuracy across various orientations and subjects. We share the collected datasets and BeamSense implementation code for reproducibility – https://github.com/kfoysalhaque/BeamSense.

1. Introduction

Since 1990, Wi-Fi has become the technology of choice for Internet connectivity in indoor environments [1]. Beyond connectivity, Wi-Fi signals can be used as sounding waveforms to perform activity recognition [2], health monitoring [3], and human presence detection [4], among others [5]. The intuition behind Wi-Fi sensing is that humans act as obstacles to the propagation of radio signals in the environment. Specifically, when encountering the human body, the radio waves undergo reflections, diffractions, and scattering that make the signals collected at the Wi-Fi receiver differ from the transmitted ones. Wi-Fi sensing aims at detecting the changes in the Wi-Fi signals and associating them to the way the subject stays/moves in the environment, thus realizing device-free monitoring solutions. To date, the vast majority of Wi-Fi sensing systems – discussed in Section 2 – leverage channel measurements obtained from pilot symbols as sensing primitive. Such measurements are usually referred to as CSI and describe the way the signals propagate in the environment. Despite leading to good performance, CSI-based techniques require extracting and recording the CSI estimated by the Wi-Fi devices involved in the sensing activities, and such operations are currently not supported by the IEEE 802.11 standard. This has led to the introduction of custom-tailored firmware modifications to extract the CSI [6–10], which makes the sensing process not scalable. Such CSI extraction tools only provide support for single-user multiple-input multiple-output (MIMO) sensing as the channel is sounded on the link between the transmitter and the device implementing the extraction tool. Therefore, Wi-Fi sensing approaches relying on CSI extraction tools cannot benefit from the spatial diversity

* Corresponding author.
E-mail addresses: haque.k@northeastern.edu (K.F. Haque), zhang.mil@northeastern.edu (M. Zhang), francesca.meneghello.1@unipd.it (F. Meneghello), frestuc@northeastern.edu (F. Restuccia).
[1] Graduate Student Member, IEEE
[2] Member, IEEE
[3] Senior Member, IEEE

https://doi.org/10.1016/j.comnet.2024.111020
Received 20 May 2024; Received in revised form 18 October 2024; Accepted 25 December 2024
Available online 10 January 2025

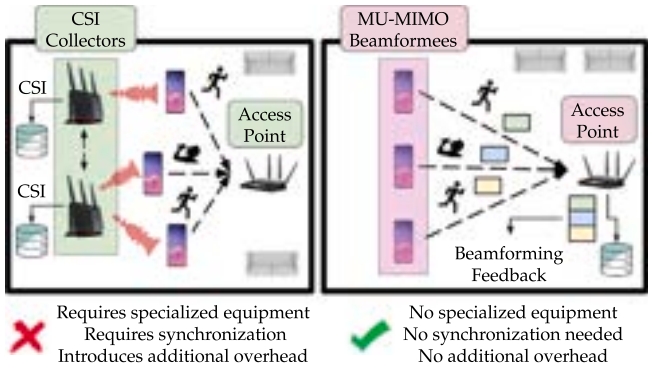

Fig. 1. CSI-based vs. BFI-based Wi-Fi sensing.

that can be gained through multi-user MIMO (MIMO) transmissions. Spatial diversity may be achieved considering multiple CSI collectors but this would increase the computation burden as synchronization among the devices would be needed. Moreover, even if CSI extraction could be supported in the future without the need for custom-tailored firmware modifications, it would require additional processing to extract the data from the chip, thus increasing energy consumption. Therefore, we argue that more suitable approaches to Wi-Fi sensing should be put forward.

In this paper, we propose `BeamSense`, an entirely new approach to Wi-Fi sensing that leverages the MU-MIMO capabilities of Wi-Fi to drastically increase sensing performance while substantially reducing sensing overhead. As shown in Fig. 1, `BeamSense` leverages the compressed BFI (BFAs)– traditionally used to beamform transmissions – to estimate the propagation environment between the access point (AP) and the connected stations (STAs). In stark contrast with CSI-based sensing, `BeamSense` (i) does not need firmware modifications, since any off-the-shelf Wi-Fi device can capture BFI packets, which are sent unencrypted to keep the processing delay below a few milliseconds [11]; and (ii) does not require synchronization among receivers, since a single BFAs report contains the information about all the MIMO channels established between the AP and the STAs. In fact, while devices empowered with CSI extraction tools allow obtaining information on a single MIMO channel, when capturing the BFAs we obtain the channel information associated with all the STAs involved in a MU-MIMO transmission. Thus, multiple spatially diverse channel information is collected with a single capture. For this reason, `BeamSense` exhibits far better performance in challenging environments, as shown in Section 4.

This paper provides the following contributions:

- We propose `BeamSense`, a new approach to Wi-Fi sensing where the standard-compliant BFAs routinely sent in MU-MIMO Wi-Fi networks is used to characterize the propagation environment between the MU-MIMO users and the AP. To the best of our knowledge, this is the first work proposing the utilization of BFAs to perform Wi-Fi sensing;
- We propose a deep learning (DL)-based Fast and Adaptive Micro Reptile Sensing (FAMReS) algorithm to perform activity classification based on BFAs. We chose DL since it has shown remarkable performance in classifying activities in Wi-Fi sensing settings [12]. However, it is well-known that bare-bone DL models may perform poorly when tested in different settings [13]. For this reason, FAMReS leverages FSL to quickly generalize to different subjects and environments with few additional data points;
- We extensively evaluate `BeamSense` through a comprehensive data collection campaign, with three subjects performing twenty different activities in three different environments. For that, we built a reconfigurable IEEE 802.11ac MU-MIMO network with three STAs and one AP. The Wi-Fi network was also synchronized with a camera-based system that records the ground truth for our experiments. A secondary co-located IEEE 802.11ac network empowered with Nexmon CSI [8] concurrently collects the CSI measurements used for comparative analysis. We show that our BFAs based approach combined with a traditional convolutional neural network (CNN) without data pre-processing achieves about 10% more accuracy when compared to state-of-the-art CSI-based techniques with substantial pre-processing. Moreover, FAMReS improves accuracy by up to 30% and 80% when compared with state-of-the-art cross-domain algorithms.
- We demonstrate the versatility of `BeamSense` by applying it to another smart-home application – gesture recognition – achieving over 98% accuracy across varying orientations and subjects. We show that also in this application, FAMReS significantly outperforms state of the art (SOTA) methods like OneFi and WiTransfer, showcasing its robust generalization capabilities. **For reproducibility, we released the entirety of our 800 GB datasets and `BeamSense` implementation code at https://github.com/kfoysalhaque/BeamSense.**

The rest of the article is organized as follows. In Section 2 we review the existing literature in the area. The `BeamSense` Wi-Fi sensing system is illustrated in Section 3 whereas the performance evaluation of the system is presented in Section 4. Section 5 concludes the discussion.

2. Related work

Over the last ten years, a lot of efforts have been made to explore wireless sensing, which is summarized by Liu et al. in [14]. The first Wi-Fi sensing approaches were based on the received signal strength indicator (RSSI) [15–20]. More recently, researchers have focused on the more fine-grained CSI information that describes how the wireless channel modifies signals at different frequencies rather than providing a cumulative metric on the signal attenuation as the RSSI does. Passive Wi-Fi radar (PWR)-based approaches [21–25] have also been proposed in the literature. However, such an approach requires specialized hardware (software defined radio (SDR)) to analyze the collected signal. In the rest of the section, we focus on CSI-based sensing, and summarize the main research on the topic.

Background on CSI-based Sensing. The term CSI can refer both to the time-domain channel impulse response (CIR) or the frequency-domain CFR. Specifically, the CIR encodes the information about the multipath propagation of the transmitted signal: each peak in the CIR represents a propagation path characterized by a specific time delay (linked with the length of the path) and an attenuation. Multipath propagation is a typical phenomenon of indoor environments, where obstacles (objects, people, animals) in the surroundings act as reflectors/diffractors/scatterers for the irradiated wireless signals. In turn, the receiver collected different copies of the transmitted signal each associated with a different propagation, or, equivalently, an obstacle in the environment. The CFR represents the Fourier transform of the CIR and describes how the environment modifies signals transmitted with different carrier frequencies. Specifically, indicating with $\mathbf{x}(f,t)$ and $\mathbf{y}(f,t)$ the frequency domain representation of the transmitted and received signals at time t and frequency f respectively, and with $\mathbf{h}(f,t)$ the CFR, we have that $\mathbf{y}(f,t) = \mathbf{h}(f,t) \times \mathbf{x}(f,t)$ [26]. Considering the $M \times N$ MIMO orthogonal frequency-division multiplexing (OFDM) system, with K sub-channels, and M and N transmitting and receiving antennas respectively, the CFR is a $K \times M \times N$-dimensional matrix providing the amplitude and phase information over each OFDM sub-channel for any given pair of transmitting and receiving antenna.

Existing Research on CSI-based Sensing. Over the last decade, CSI-based sensing has been proposed for a wide variety of applications. Among the most compelling, we mention person detection and identification [27–29], crowd counting [18,30], respiration monitoring [31], baggage tracking [32], smart homes [33,34], human pose tracking [35–38], patient monitoring [39,40], with most of the previous research

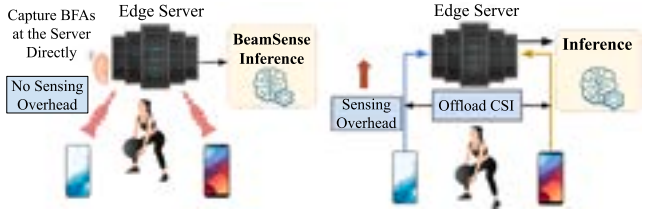

Fig. 2. Advantages of BeamSense over CSI-based approaches in terms of sensing overhead and system latency.

activities focusing on human activity recognition (HAR) and human gesture recognition (HGR) [13,41–45]. ***The above list is definitely not exhaustive.*** For excellent survey papers on the topic, we refer the reader to [2,5,46,47]. In the following, we just summarize the most recent approaches that are most related to the work conducted in this article. Guo et al. presented WiAR [48], a CSI-based system achieving up to 90% accuracy in the recognition of 16 human activities. Similarly, a meta-learning-based approach called RF-Net was presented in [49] based on the usage of recurrent neural networks with long short-term memory (LSTM) cells. However, only six activities were considered in the evaluation. Regarding HGR, [43,44] presented Widar 3.0 and OneFi, respectively considering six and forty gestures. The authors in [43] proposed to use a body velocity profile (BVP) measure which has been shown to improve the generalization capability of the classification algorithm. The authors of [44] used one-shot learning to classify unseen gestures with few labeled samples. The majority of previous work has been evaluated on 802.11n channel data while, to the best of our knowledge, only two works considered HAR in the context of 802.11ac [12,13]. Meneghello et al. proposed to use the Doppler shift estimated through the CSI to obtain an algorithm that generalizes to different environments [13] whereas, Bahadori et al. used few-shot learning approach to achieve environmental robustness [12].

Limitations of CSI-based Sensing. Since the CSI is computed at the physical layer (PHY), it is not readily available with off-the-shelf network interface cards (NICs). Although CSI can be extracted with SDR implementations, which only support up to 40 MHz of bandwidth, being only IEEE 802.11 a/g/p/n compliant [12,50]. Moreover, SDRs are costly specialized hardware that may be unavailable in real-life situations and require expert knowledge to be used. To overcome such limitations, in recent years, researchers have developed some CSI extraction tools that run on commercial Wi-Fi NICs. Two of them, namely Linux CSI [6] and Atheros CSI [7], target IEEE 802.11n compliant NICs (up to 40 MHz bandwidth). The third one, Nexmon CSI [8], allows extracting the CFR from some IEEE 802.11ac compliant devices, supporting bandwidths up to 80 MHz. The most recent one, AX CSI [10] is designed for IEEE 802.11ax devices and provides CFR measurements also on 160 MHz bandwidth channels. These tools, however, need non-trivial firmware modifications of the NICs. Moreover, they do not provide support for estimating the channel on MU-MIMO channels. Both when the CSI extractor tool is implemented on one receiving Wi-Fi device or on another monitor device, only the MIMO links between the transmitter and the CSI collector is monitored, i.e., only SU-MIMO mode is supported. This is a limitation of CSI-based systems as MU-MIMO systems can provide way richer information than SU-MIMO ones as they capture the correlation of the propagated signal from different STAs relative to the sensed subject. As a last consideration, Wang et al. [51] recently pointed out the importance of the placement of the CSI extractor device. Specifically, they showed that accurate placement of the sensing devices can enhance the sensing coverage by mitigating severe interference. Non-calibrated placement of the sensing devices can severely hamper the sensing quality.

Recent BFI-based Sensing Approaches. BFI is gaining momentum in the research community as a proxy to the CSI as it provides spatially diverse rich channel information from commercial Wi-Fi devices without the need for any firmware modification or direct access to the hardware. In this context, while the CSI is nowadays well recognized to be valuable for sensing purposes, some recent research work has also considered BFI for sensing showing its high potentialities.

Jiang et al. investigate the Wi-Fi sensing performance in terms of angle of arrival (AoA), Doppler, and range estimation based on the BFI for two different approaches to obtain the beamforming matrix, i.e., performing eigenvalue decomposition (EVD) (i) on the channel autocorrelation matrix or (ii) on the conjugate transpose of the channel autocorrelation matrix [57]. The results show that the first approach retains only the Doppler and time delay differences of different paths, whereas the second scheme retains absolute Doppler and delay information. Kondo et al. evaluate the impact of uni-directional (DL-MU-MIMO) and bi-directional (DL and UL-MU-MIMO) beamforming on Wi-Fi sensing performance through the BFI reconstructed from BFAs [52]. The results demonstrate that the framework based on bi-directional beamforming achieves better sensing performance in terms of angle of departure (AoD). The same authors leverage the BFI for respiratory rate estimation in [53] achieving an estimation error lower than 3.2 breaths/minute. Finally, Wu et al. proposed a BFI-based wireless sensing system for device localization, passive tracking, and sign language recognition [54]. Their proposed system achieves a localization median error of 0.72 m, passive tracking median error of 0.67–0.95 m, and sign language recognition accuracy of 92.5%–97.14%. We stress that all these sensing systems leverage the BFI matrices reconstructed from the compressed BFAs transmitted over the air. This incurs additional pre-processing stages that increase the system latency and computational burden of the sensing system. On the contrary, BeamSense is based on the compressed BFAs which are directly captured from ongoing transmissions and do not need any pre-processing. Another advantage of using BFAs instead of the BFI is the dimensionality of the data. Being BFAs a compressed version of the BFI, their processing requires neural networks with a smaller input dimensionality and, in turn, with a smaller number of learnable parameters, with respect to processing BFI data. We included some preliminary results about this methodology in [58] where we present **Wi-BFI** [58], an open-source tool to capture BFAs packets from any ongoing Wi-Fi transmissions, decode the BFAs and reconstruct BFI in both real-time and from captured traces. Note that [58] focuses on the BFAs extraction and reconstruction of BFI with only some preliminary results about sensing capabilities with BFAs. In this current work, we instead deeply analyze the use of BFAs for sensing, including a novel FSL-based algorithm that enhances the generalization capabilities of the sensing system.

Table 1 provides a comprehensive comparison between the proposed BFAs-based sensing approach (BeamSense) and other SOTA CSI and BFI-based methods. It provides a comprehensive summary of BeamSense and other SOTA approaches, evaluating them based on the technology utilized, operating bandwidth, sensing primitive considered, firmware modification requirements, sensing applications, number of classes, sensing accuracy, and domain generalization capabilities. The comparison highlights the advantages of BeamSense with respect to other sensing approaches. Specifically, BeamSense does not require firmware modifications and allows achieving high accuracy across multiple tasks (96%–99%), offering superior generalization performance (90%–95%) across different scenarios. These factors make BeamSense the preferred choice for practical applications where ease of integration and robustness to domain shifts are critical.

Advantages of BeamSense over CSI-based sensing approaches. Our approach overcomes the limitations of traditional CSI-based methods by leveraging the MU-MIMO compressed beamforming feedback, which is transmitted as part of the channel sounding procedure standardized in IEEE 802.11. Unlike CSI-based approaches, which require firmware modifications to extract CSI data, our system utilizes

Table 1
Overview of the main characteristics of BeamSense and state-of-the-art approaches.

Model name	Technology considered	Considered bandwidth (MHz)	Sensing primitive	Firmware modification	Sensing applications	No. of classes	Sensing accuracy (%)	Domain generalization accuracy
BeamSense (proposed)	IEEE 802.11 ac	80 MHz	BFAs	No	Activity classification and gesture recognition	20	96–99	90–95
Bi-directional BFM [52]	IEEE 802.11ax	80 MHz	BFI (EVD on autocor-relation matrix)	No	Human localization and AoD estimation	N/A	95–98	92–96
Respiratory rate estimation [53]	IEEE 802.11ax	80 MHz	BFI	No	Respiratory rate estimation	N/A	Error <3.5 breaths/minute	N/A
BFI-based sensing [54]	IEEE 802.11ax	80 MHz	BFI	No	Device localization, passive tracking, and sign language recognition	20	92.5–97.14	Localization error: 0.3–0.72 m, Tracking error: 0.67–0.95 m
SignFi [55]	IEEE 802.11n	40 MHz	CSI	Yes	Sign gesture classification	276	94.81–98.91	86.66
WiAR [48]	IEEE 802.11n	40 MHz	CSI	Yes	Human activity recognition	16	80–95	80-90
OneFi [44]	IEEE 802.11n	40 MHz	CSI	Yes	Human gesture recognition via one-shot learning	40	84.2–98.8	75–91
Wi-Transfer [56]	IEEE 802.11n/ac	80 MHz	CSI	Yes	Transfer learning-based sensing	6	88–99	85–96
Widar 3.0 [43]	IEEE 802.11n	40 MHz	CSI	Yes	Gesture recognition via body velocity profile (BVP)	15	92.7	82.6–92.4
ReWiS [12]	IEEE 802.11ac	80 MHz	CSI	Yes	Activity recognition via multi-receiver CSI learning	4	98–100	90–100
SHARP [13]	IEEE 802.11ac	80 MHz	CSI	Yes	Human activity recognition via micro-Doppler	7	>95	90–95

standard-compliant 802.11 ac/ax devices to collect compressed beamforming feedback packets. This eliminates the need for specialized hardware or infrastructure, making our system more practical for deployment compared to CSI-based strategies. Moreover, BFAs can be captured from anywhere within the network without any direct access to the sensing devices, i.e., the devices estimating the wireless channel. The device collecting the beamforming feedback (monitor device) can remotely obtain channel information of the links between the AP and multiple STAs by simultaneously capturing the beamforming packets transmitted unencrypted over the air at the end of the channel sounding procedure. Hence, as presented in Fig. 2(a), `BeamSense` can be deployed directly at the edge server where the sensing application is deployed, thus reducing the channel airtime overhead for sensing data transmission and, in turn, the overall system latency. Contrarily, traditional CSI-based methods require direct access to the device estimating the channel as the firmware of the devices needs to be modified to enable CSI extraction. Moreover, the extracted CSI needs to be fed back to the edge server as presented in Fig. 2(b), introducing airtime overhead for sensing data transmission. This overhead may lead to a degradation of communication performance.

3. The `BeamSense` Wi-Fi sensing system

Fig. 3 shows a high-level overview of `BeamSense`, which leverages the channel estimation mechanism standardized in IEEE 802.11 to sound the physical environment. The channel estimation is performed on the STAs (beamformees) and is reported to the AP (beamformer) that uses it to properly beamform MU-MIMO transmissions. The report is referred to as the BFI and is transmitted over the air in clear text in the form of BFAs frames. Since the AP continuously triggers the channel estimation procedure on the connected STAs, *the BFAs contains very rich, reliable, and spatially diverse information*. Moreover, the BFAs *from multiple STAs can be collected with a single capture* by the AP or any other Wi-Fi-compliant device, thus reducing the system complexity.

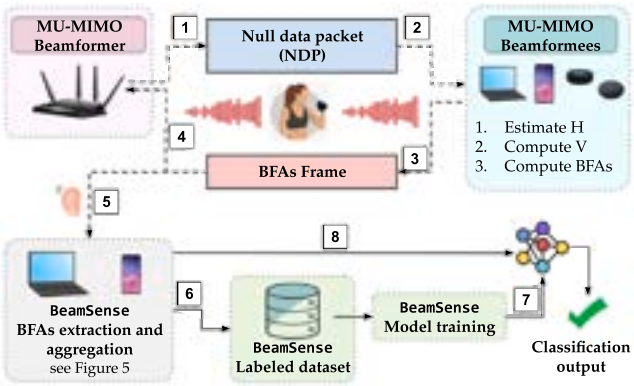

Fig. 3. The `BeamSense` Wi-Fi sensing system.

`BeamSense` Technical Challenges. `BeamSense` is a completely novel way to perform Wi-Fi sensing. While previous work in the literature deal with the well-known CSI data, we instead consider the BFAs as a sensing primitive. We stress that BFAs represents a completely new type of data. While CSI consists of complex I/Q-values, BFAs are expressed in terms of rotational angles of the compressed matrices. In this respect, the first challenge we need to address is the design and implementation of a novel tool to extract the BFAs data embedded within Wi-Fi frames transmitted from the beamformees to the beamformer as part of the channel-sounding procedure. On top

of that, the second challenge concerns the implementation of a new data processing pipeline for the new data type that effectively performs activity classification based on BFAs data and provides environment adaptation features. The third challenge to be addressed is the setup of an extensive experimental testbed to implement and assess the performance of the new Wi-Fi sensing approach in a real-world scenario with commercial Wi-Fi devices.

In the following, we thoroughly detail the `BeamSense` sensing system. We use the superscripts T and $\dagger$ to denote the transpose and the complex conjugate transpose (i.e., the Hermitian). We define with $\angle\mathbf{C}$ the matrix containing the phases of the complex-valued matrix $\mathbf{C}$. Moreover, $\text{diag}(c_1, \dots, c_j)$ indicates the diagonal matrix with elements $(c_1, \dots, c_j)$ on the main diagonal. The (c_1, c_2) entry of matrix $\mathbf{C}$ is defined by $[\mathbf{C}]_{c_1,c_2}$, while $\mathbb{I}_c$ refers to an identity matrix of size $c \times c$ and $\mathbb{I}_{c\times d}$ is a $c \times d$ generalized identity matrix.

3.1. *BeamSense: A walkthrough*

The `BeamSense` sensing system entails eight steps, as depicted in Fig. 3. The process stems from the way beamforming is implemented in IEEE 802.11 networks. Specifically, the beamformer (AP) uses a matrix $\mathbf{W}$ of pre-coding weights – called steering matrix – to linearly combine the signals to be simultaneously transmitted to the different beamformees (STAs). The steering matrix is derived from the CFR matrices $\mathbf{H}$ estimated by each of the beamformee and that describe how the environment modifies the irradiated signals in their path to the receivers. The estimation process is called *channel sounding* and is triggered by the AP which periodically broadcasts a null data packet (NDP) (**step 1** in Fig. 3) that contains sequences of bits – named long training fields (LTFs) – the decoded version of which is known by the beamformees. Since its purpose is to sound the channel, the NDP *is not beamformed* by the AP. *This is particularly advantageous for sensing purposes*, since the resulting CFR estimation will not be affected by inter-stream or inter-user interference. The LTFs are transmitted over the different beamformer antennas in subsequent time slots, thus allowing each beamformee to estimate the CFR of the links between its receiving antennas and the beamformer transmitting antennas. The LTFs are modulated – as the data fields – through OFDM by dividing the signal bandwidth into K partially overlapping and orthogonal sub-channels spaced apart by $1/T$. The input bits are grouped into OFDM symbols, $\mathbf{a} = [a_{-K/2}, \dots, a_{K/2-1}]$, where a_k is named OFDM sample. These K OFDM samples are digitally modulated and transmitted through the K OFDM sub-channels in a parallel fashion thus occupying the channel for T seconds. The transmitted LTF signal is

$$s_{\text{tx}}(t) = e^{j2\pi f_c t} \sum_{k=-K/2}^{K/2-1} a_k e^{j2\pi kt/T}, \tag{1}$$

where f_c is the carrier frequency. The NDP is received and decoded by each STA (**step 2**) to estimate the CFR $\mathbf{H}$. The different LTFs are used to estimate the channel over each pair of transmitting (TX) and receiving (RX) antennas, for every OFDM sub-channel. This generates a $K\times M\times N$ matrix $\mathbf{H}$ for each beamformee, where M and N are respectively the numbers of TX and RX antennas. We refer the reader to Section 2 for additional details about the CFR. Next, the CFR is compressed – to reduce the channel overhead – and fed back to the beamformer. Using $\mathbf{H}_k$ to identify the $M \times N$ sub-matrix of $\mathbf{H}$ containing the CFR samples related to sub-channel k, the *compressed beamforming feedback* is obtained as follows ([59], Chapter 13). First, $\mathbf{H}_k$ is decomposed through singular value decomposition (SVD) as

$$\mathbf{H}_k^T = \mathbf{U}_k \mathbf{S}_k \mathbf{Z}_k^\dagger, \tag{2}$$

where $\mathbf{U}_k$ and $\mathbf{Z}_k$ are, respectively, $N\times N$ and $M\times M$ unitary matrices, while the singular values are collected in the $N\times M$ diagonal matrix $\mathbf{S}_k$. Using this decomposition, the complex-valued beamforming matrix $\mathbf{V}_k$ is defined by collecting the first $N_{\text{SS}} \le N$ columns of $\mathbf{Z}_k$. Such a matrix is used by the beamformer to compute the pre-coding weights for the N_{SS} spatial streams directed to the beamformee. Hence, $\mathbf{V}_k$ is converted into polar coordinates as detailed in Algorithm 1 to avoid transmitting the complete matrix. The output is matrices $\mathbf{D}_{k,i}$ and $\mathbf{G}_{k,\ell,i}$, defined as

Algorithm 1: $\mathbf{V}_k$ matrix decomposition

Require: $\mathbf{V}_k$;
$\tilde{\mathbf{D}}_k = \text{diag}(e^{j\angle[\mathbf{V}_k]_{M,1}}, \dots, e^{j\angle[\mathbf{V}_k]_{M,N_{\text{SS}}}})$;
$\mathbf{\Omega}_k = \mathbf{V}_k\tilde{\mathbf{D}}_k^\dagger$;
for $i \leftarrow 1$ *to* $\min(N_{\text{SS}}, M-1)$ **do**
  $\phi_{k,\ell,i} = \angle\left[\mathbf{\Omega}_k\right]_{\ell,i}$ with $\ell = i, \dots, M-1$;
  compute $\mathbf{D}_{k,i}$ through Eq. (3);
  $\mathbf{\Omega}_k \leftarrow \mathbf{D}_{k,i}^\dagger \mathbf{\Omega}_k$;
  for $\ell \leftarrow i+1$ *to* M **do**
    $\psi_{k,\ell,i} = \arccos\left(\frac{[\mathbf{\Omega}_k]_{i,i}}{\sqrt{[\mathbf{\Omega}_k]_{i,i}^2+[\mathbf{\Omega}_k]_{\ell,i}^2}}\right)$;
    compute $\mathbf{G}_{k,\ell,i}$ through Eq. (4);
    $\mathbf{\Omega}_k \leftarrow \mathbf{G}_{k,\ell,i}\mathbf{\Omega}_k$;

$$\mathbf{D}_{k,i} = \begin{bmatrix} \mathbb{I}_{i-1} & 0 & \dots & 0 \\ 0 & e^{j\phi_{k,i,i}} & 0 & \dots \\ \vdots & 0 & \ddots & 0 \\ & \vdots & 0 & e^{j\phi_{k,M-1,i}} & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}, \tag{3}$$

$$\mathbf{G}_{k,\ell,i} = \begin{bmatrix} \mathbb{I}_{i-1} & 0 & \dots & 0 \\ 0 & \cos\psi_{k,\ell,i} & 0 & \sin\psi_{k,\ell,i} & \vdots \\ \vdots & 0 & \mathbb{I}_{\ell-i-1} & 0 & \\ & -\sin\psi_{k,\ell,i} & 0 & \cos\psi_{k,\ell,i} & 0 \\ 0 & \dots & 0 & \mathbb{I}_{M-\ell} \end{bmatrix}, \tag{4}$$

that allow rewriting $\mathbf{V}_k$ as $\mathbf{V}_k = \tilde{\mathbf{V}}_k\tilde{\mathbf{D}}_k$, with

$$\tilde{\mathbf{V}}_k = \prod_{i=1}^{\min(N_{\text{SS}},M-1)} \left(\mathbf{D}_{k,i} \prod_{l=i+1}^{M} \mathbf{G}_{k,l,i}^T\right) \mathbb{I}_{M\times N_{\text{SS}}}, \tag{5}$$

where the products represent matrix multiplications. In the $\tilde{\mathbf{V}}_k$ matrix, the last row – i.e., the feedback for the Mth transmitting antenna – consists of non-negative real numbers by construction. Using this transformation, the beamformee is only required to transmit the ϕ and ψ angles to the beamformer as they allow reconstructing $\tilde{\mathbf{V}}_k$ precisely. Moreover, it has been proved (see [59], Chapter 13) that the beamforming performance is equivalent at the beamformee when using $\mathbf{V}_k$ or $\tilde{\mathbf{V}}_k$ to construct the steering matrix $\mathbf{W}$. In turn, the feedback for $\tilde{\mathbf{D}}_k$ is not fed back to the beamformer. The angles are quantized using $b_\phi \in \{7, 9\}$ bits for ϕ and $b_\psi = b_\phi - 2$ bits for ψ, to further reduce the channel occupancy. The quantized values – $q_\phi = \{0, \dots, 2^{b_\phi} - 1\}$ and $q_\psi = \{0, \dots, 2^{b_\psi} - 1\}$ – are packed into the compressed beamforming frame (**step 3**) and such *BFAs* are transmitted to the AP (**step 4**) in *clear text*. Each BFAs frame contains A number of angles for each of the K OFDM sub-channels for a total of $(K \cdot A)$ angles each. In Fig. 4, we show an example of how beamforming is conducted in a 3×2 MIMO system.

`BeamSense` captures the BFAs frames (**step 5**), and uses the channel estimation data to perform Wi-Fi sensing. We remark that, since MU-MIMO requires fine-grained channel sounding – every around 10 ms to account for user mobility, according to [60] – it is fundamental to process the BFAs in a fast manner at the AP. For this reason, and since cryptography would lead to excessive delays, the angles are currently sent unencrypted. Therefore, the BFAs frames are exposed to and can be read by any device that can access the wireless channel. Specifically, `BeamSense` relies on the BFAs transmitted by all the beamformees in the environment and captured during a time window of W seconds to reliably estimate the activity being performed by

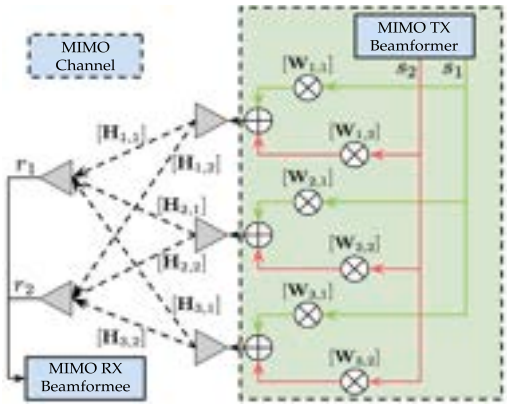

Fig. 4. Example of 3 × 2 MIMO system. s_1, s_2 and r_1, r_2 are respectively the transmitted and received signals. The symbol **W** indicates the steering matrix, while **H** is the CFR.

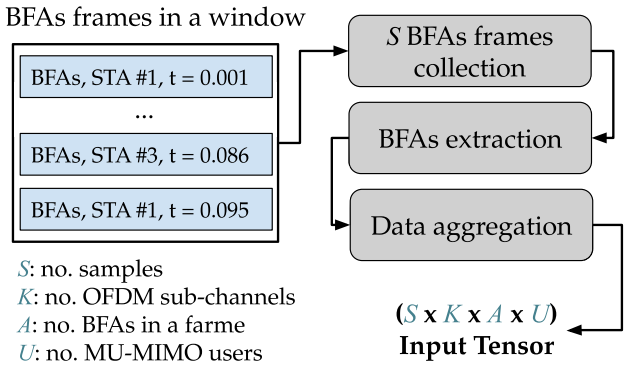

Fig. 5. BFAs data processing. The processing is applied to each observation window of W seconds.

a human moving within the propagation environment. This is done by using the BFAs frames collected within the window as input for a learning-based algorithm (detailed in Section 3.2). Note that, as BeamSense leverages ongoing MU-MIMO transmissions, there is no guarantee that the same number of BFAs frames are collected within a specific time interval of W seconds. This is related to the fact that we have no control over when the beamformer triggers the channel sounding procedure that generates BFAs data. Therefore, as the neural network-based classification algorithm requires the input to be of a fixed dimension, we need to determine a fixed-size input that represents the BFAs frames captured during the time window. The processing is applied just after having collected the data on the wireless channel (gray box in Fig. 3) and is summarized in Fig. 5. Specifically, we consider the average number S of BFAs frames counted (at training time) in each window during an activity recording. Windows having less than S frames are padded with BFAs frames containing zero-valued angles while packets exceeding such threshold are discarded. Hence, the $K \times A$ BFI angles contained in each packet are extracted and the final tensor is obtained by aggregating the $S \times K \times A$ angles for all the U MU-MIMO users for which the BFAs data have been captured in the observation window. Note that even if it would be possible to define learning algorithms that accept input of different sizes, this would lead to an increase in the complexity of the approach, both from the training and inference perspective. Therefore, to keep the model simple for implementation on memory- and battery-constrained devices, we decided to follow a fixed-input approach.

To obtain the training data, the $S \times K \times A \times U$ tensors derived from the BFAs farmes captured during the data collection phase are stored in a dataset, together with their associated activity and/or phenomenon, and a timestamp (**step 6** in Fig. 3). This phase can be performed offline by sensing application vendors without requiring the users' cooperation. The trained model (**step 7**) is then used for online sensing (**step 8**).

The BFAs are transmitted unencrypted in accordance with the IEEE 802.11 standards. Specifically, the standards specify that BFAs should be fed back using *"Not Robust Action Frames"*, which are transmitted unencrypted. This is linked with the need to receive channel feedback with low latency to enable MIMO transmissions. Indeed, given the variability of the wireless channel, BFAs should be transmitted every about 10 ms and promptly used for precoding MIMO transmissions. In this context, encryption would make the procedure less efficient and may lead to a degradation of the communication performance.

Note that BFAs do not contain any sensitive information about users and their data. Indeed, BFAs are a compressed and quantized representation of the CSI estimated for the links between the AP and the STAs in the considered MU-MIMO network. Such angles are used for precoding purposes to enable the simultaneous transmission of multiple data streams to the STAs. Even if the choice to transmit them unencrypted is prone to adversarial attacks [61], exploring secure transmission methods for beamforming feedback, such as encryption or other physical layer security techniques, is outside the scope of this work. Our primary focus is on demonstrating the feasibility and effectiveness of using BFAs for wireless sensing in compliance with existing standards, which require BFAs to be fed back unencrypted.

The timing of the channel sounding procedure, which generates BFAs samples, is a critical factor influencing the performance of sensing systems that rely on this information, such as BeamSense. As mentioned in [60], channel sounding should be performed every 10 ms, resulting in about 100 BFAs frames per second. This rate ensures that the AP keeps the precoding aligned with the channel variations, which are encoded in the BFAs estimated at the STAs and promptly fed back to the AP. This rate is also enough to provide adequate performance in most sensing applications. However, it is important to recognize that increasing the frequency of BFAs frames would result in finer granularity of the CFR. With more frequent updates, the sensing system would have access to more detailed and precise information about the channel characteristics. The increased granularity would enhance the system's sensing capabilities, allowing it to track rapid and subtle variations in the environment more effectively. As a result, higher BFAs frame rates could lead to improved sensing performance, particularly in dynamic environments where the channel conditions change frequently. This important aspect will be addressed in the upcoming IEEE 802.11bf standard that will define proper strategies to integrate communication and sensing services. In particular, new procedures will be defined to enable the collection of channel information even when no data is transmitted over the wireless channel, and, in turn, no channel sounding is performed. This new feature will enable the widespread adoption of BFAs-based sensing techniques such as BeamSense. Importantly, BeamSense will be directly applicable to new Wi-Fi standards, making it a strong candidate for integrated sensing and communications applications.

3.2. The FAMReS classification algorithm

Existing research in CSI-based sensing has exposed that designing classifiers that are robust to changing the subject performing the activity (i.e., different people) and the environment where the activity is performed (i.e., different rooms) is very challenging [12,13,43,44]. On the other hand, it is hardly feasible to collect a large amount of data for all possible scenarios. To address this key issue, we propose a deep learning (DL)-based algorithm for BFI-based activity classification called *Fast and Adaptive Micro Reptile Sensing* (FAMReS), which is a few-shot learning (FSL) algorithm based on Reptile [62] which needs a limited set of new input data to generalize to unseen environments.

FSL is a DL technique that leverages only small amounts of additional data to adapt to classes that are unseen at training time. Specifically, in K-way-N-shot FSL, the model is trained on a set of mini-batches of data sampled from only K different classes (ways) and containing N samples (shots) of each class. The key idea is that by feeding less data, the model is spurred to rapidly adapt to new tasks. This unique property makes FSL a strong candidate to tackle the diversity of environments. The key reason for using FSL for Wi-Fi sensing is that we aim at creating an almost plug-and-play framework for the end-users. In particular, it would be infeasible to account for all the specific end-user scenarios – in terms of activities, people, and environment diversity – during the algorithm design before its release to the public. For these reasons, our `BeamSense` algorithm comes with a base set of 20 different activities on 3 standard environments on which it has been trained and, for generalization, is empowered with few-shot learning capabilities to quickly adapt to new domains (environments/subjects).

FSL can be categorized into embedding learning [63,64], and meta-learning [62,65], among others. Specifically, Reptile is a gradient-based meta-learning algorithm that learns the model parameter initialization for rapid fine-tuning. The key idea is that there are some common features between different tasks that can be learned through meta-learning. Therefore, the model can be fine-tuned on a new task faster with the meta-learned weights instead of training it from the beginning. To find the initialization weights θ^*, Reptile minimizes the expectation of the loss function L_τ with respect to the different tasks τ, i.e.,

$$\theta^* = \min_{\theta} \quad \mathbb{E}_\tau \left\{ L_\tau \left[f\left(x, y|\theta\right) \right] \right\}, \tag{6}$$

where $f(x, y|\theta)$ is the model functional approximation between input data x and output y obtained with parameters θ. This is equivalent to finding the θ^* that satisfies $\mathbb{E}_\tau \left\{ \nabla_\theta \left(L_\tau \left[f\left(x, y|\theta\right) \right] \right) \right\} = 0$ via, e.g., stochastic gradient descent (SGD). SGD finds θ^* through an iterative procedure, by subsequently updating the value of θ with a new value θ' based on the gradient information:

$$\theta' = \theta - \beta \frac{1}{n} \sum_{\tau=1}^{n} \left(\frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \left(L_\tau \left[f\left(x_i, y_i|\theta\right) \right] \right) \right) \tag{7}$$

$$= \theta - \beta \frac{1}{n} \sum_{\tau=1}^{n} \left(\theta - \tilde{\theta} \right), \tag{8}$$

where n and m denote the number of tasks and sampled data points of each task, respectively, β is a scalar denoting the step size, and $\tilde{\theta} = \theta - \alpha \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \left(L_\tau \left[f\left(x_i, y_i|\theta\right) \right] \right)$ are the updated weights using m sampled data from τ, where α denotes the learning rate. $\tilde{\theta}$ can be easily obtained using any deep learning API such as TensorFlow and PyTorch. The meta-learning proceeds through the following steps: (i) sample n new tasks $\{\tau\}$ with m data of each task (for K-way-N-shot, m is the product of K and N); (ii) compute $\tilde{\theta}$; (iii) update θ with Eq. (8); (iv) iterate (ii) and (iii) until the loss function stops decreasing. Fig. 6 shows how FSL is implemented through the Reptile algorithm: once obtained the initialization weights θ^* through meta-learning, the model is fine-tuned on each different task.

3.2.1. FAMReS algorithm

The original purpose of Reptile is to extract meta-features from a large dataset so that it can be quickly fine-turned when a new task is sampled from the given dataset. However, *Reptile requires the inference and meta-learning data to be sampled from the same dataset*. Such a dataset should contain as many classes as possible so that the meta-learner can extract the general characteristics and fine-tune a task with fewer classes. Since this is unfeasible in BFI-based sensing, we find some common ground between meta-learning and general DL. The aim of learning is trying to approach the ground truth between different sampled data, while meta-learning is to find shared features between various tasks. Thus, if we consider each batch of training data as a new task in meta-learning, *the learning problem can be converted into a meta-learning problem*. Formally, we aim to find a set of parameters θ^* that

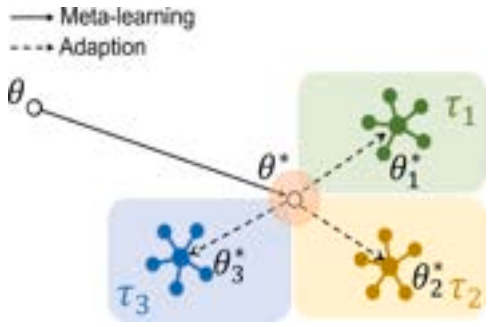

Fig. 6. Example of Few-Shot Learning.

minimize the loss function L on training data x_i and y_i:

$$\theta^* = \min_{\theta} \quad \mathbb{E}_i \left\{ L \left[f\left(x_i, y_i|\theta\right) \right] \right\}. \tag{9}$$

By plugging the derivative $\mathbb{E}_i \left\{ \nabla_\theta \left(L \left[f\left(x_i, y_i|\theta\right) \right] \right) \right\}$ to the SGD optimizer, the optimization problem can be solved as

$$\tilde{\theta} = \theta - \alpha \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \left(L \left[f\left(x_i, y_i|\theta\right) \right] \right). \tag{10}$$

By comparing Eq. (7) with (10), we can easily find that if we set $n = 1$ in Eq. (7), the only difference between these two equations is a constant scalar. Based on this observation, we note that Reptile learns common ground from different mini-batch of data. The meta-learning rate β, which is usually a scalar less than 1, is to adjust the step size of the learning, making it less likely to overfit the mini-batch data. This meta-learning process can be regarded as a warm-up phase before learning, which makes the parameters θ closer to the ground truth in the hyperspace than random initial weights.

Inspired by this idea, FAMReS is divided into two stages: (i) meta-learning stage; and (ii) micro-learning stage. In stage (i), the model utilizes a small portion of data to learn the shared features. In stage (ii), the same micro dataset is used for training. The complete FAMReS workflow is reported in Algorithm 2. **We stress the difference between the original Reptile and FAMReS**: we only use a small portion of data in meta-learning and micro-learning and use other unseen data for testing. On the contrary, Reptile uses the same dataset for both learning and inference. Although we have only done experiments offline in this work, FAMReS is a strong candidate for online learning. The algorithm can run the meta-learning phase while collecting new data. Once there is enough data, it can move on to the next stage. Therefore, we define a time variable δ in experiments to simulate the real-time implementation. We use the data collected within the δ time window for learning and the other for inference. FAMReS is an empirical risk minimizer that can be unstable when using small values for δ, depending on the distribution of training data. Meta-learning on the micro dataset can only bring the initial parameters closer to the ground truth point in the hyperspace, but the final parameters still depend on the training set. Thanks to the high stability of the BFI data, we can always get a reasonable accuracy in the experiments unless δ is extremely small.

Algorithm 2: The FAMReS Algorithm

```
Require: step size β, micro dataset 𝔻;
Initialize: a set of parameters θ;
for iteration = 1, 2, ... do
    sample k points of data from 𝔻 ;          /*stage i*/
    compute θ̃ using the SGD formulation;
    update the parameters: θ ← θ + β(θ̃ − θ);
for epoch = 1, 2, ... do
    update θ running SGD on 𝔻;                /*stage ii*/
```

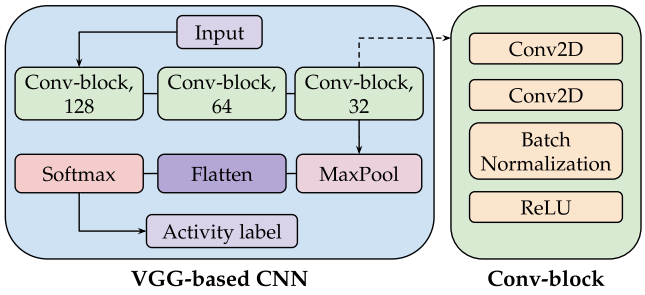

Fig. 7. Learning-based activity classifier.

3.2.2. Learning architecture

In the last decade, CNNs have achieved tremendous success in computer vision [66–68]. The convolution layer, the basis of CNNs, can efficiently extract features by performing convolution operations on the elements of the input data. Given that in this article our aim is to investigate the effectiveness of BFI-based sensing as compared to CSI-based sensing, we propose to use a VGG-based [67] CNN architecture as the human activity classifier. The network is depicted in Fig. 7 and entails stacking three convolutional blocks (`conv-block`) and a max-pooling (`MaxPool`) layer. Softmax is applied to the flattened output to obtain the probability distribution over the activity labels.

The `conv-block` is a stack of two convolution two-dimensional (2D) layers. Following the design of VGG [67], each convolution layer has a kernel size of 3×3 and a step size of 1. To introduce non-linearity in the model, we apply a rectified linear units (ReLU) activation function at the end of each `conv-block`. Batch normalization is also used in `conv-blocks` to avoid gradient explosion or vanishing. Our VGG-based CNN consists of three `conv-blocks` with 128, 64 and 32 filters, respectively. We choose a descending order of filters to reduce the model size since features in lower layers are usually sparser and thus require extracting more activation maps to be properly captured.

4. Performance evaluation

4.1. Experimental setup and data collection

We collected experimental data in three distinct indoor environments: a kitchen, a living room, and a classroom, as depicted in Fig. 8. These environments were selected to capture diverse real-world settings with varying levels of furniture, obstacles, and layout configurations that influence wireless signal propagation. Six human subjects participated in the experiments, performing twenty different activities: *jogging, clapping, push forward, boxing, writing, brushing teeth, rotating, standing, eating, reading a book, waiving, walking, browsing phone, drinking, hands-up-down, phone call, side bends, check the wrist (watch), washing hands, and browsing laptop*. The activities were chosen to represent a broad range of human motions, including both stationary and dynamic actions, to test the system's ability to differentiate between subtle and vigorous movements. Each subject performed the activities independently within a 2 m × 1.5 m rectangular region marked on the floor in each environment to ensure a consistent and controlled area for data collection. The size of the designated region was chosen to allow for free movement while maintaining a practical distance of 2–3 m from the STAs. Both BFAs and CSI data were collected for the same duration of 300 s for each of the twenty activities for every subject in different environments and orientations. This duration was selected to capture sufficient data for analysis while ensuring the subjects could comfortably perform the activities. The continuous data capture for this duration for each activity allowed for the collection of a comprehensive dataset enabling extensive temporal and spatial analyses.

To establish the ground truth, synchronous video streams of the subjects performing each activity were recorded. These video

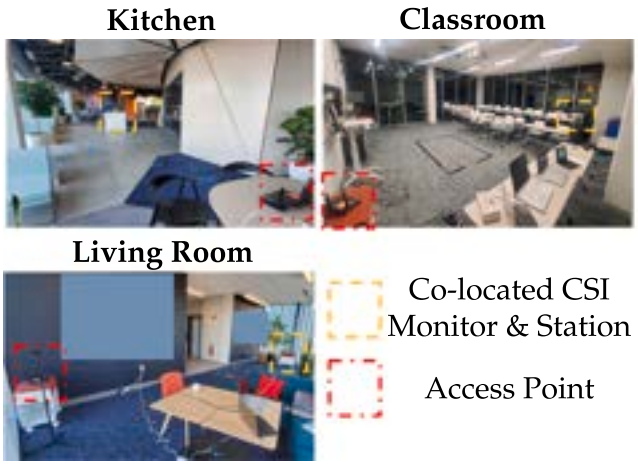

Fig. 8. Sites of experimental data collection.

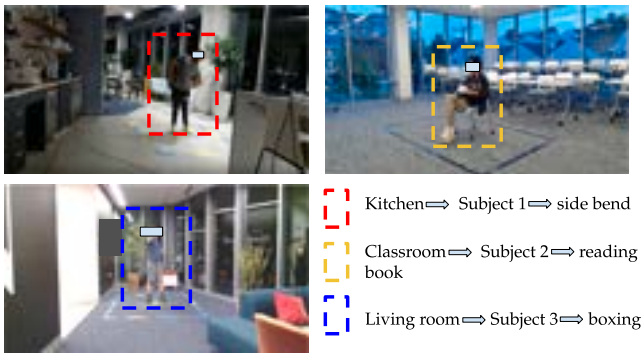

Fig. 9. Sample frames from the video capture.

streams were captured using fixed cameras positioned to cover the entire rectangular region where the subjects performed the activities, ensuring full visibility of the subject's movements. The video cameras were placed at angles that minimized occlusion and ensured clear visibility of the subject's entire body. The video streams were synchronized with the BFI and CSI data using timestamps, ensuring precise alignment between the recorded video streams of the activities and the corresponding BFI and CSI frames. This synchronization ensured that for every captured BFI and CSI frame, the corresponding activity could be accurately identified and labeled, making the dataset reliable for supervised learning models. As an example, three frames from the captured video streams are shown in Fig. 9.

`BeamSense` Network Setup and Equipment. We set up an 802.11ac MU-MIMO network operating on channel 153 with center frequency $f_c = 5.77$ GHz and 80 MHz bandwidth. This allows sounding $K = 234$ sub-channels, i.e., 256 available sub-channels on 80 MHz channels minus 14 control sub-channels and 8 pilots. We use one AP (beamformer) and three STAs (beamformees), as depicted in Fig. 10 in orange. The AP and the STAs are implemented through Netgear Nighthawk X4S AC2600 routers with $M = 3$ and $N = 1$ antennas enabled respectively for the AP and each of the STAs. The three STAs are served with $N_{ss} = 1$ spatial stream each and placed at three different heights and significantly spaced from each other to form a 3×3 MU-MIMO system. According to the IEEE 802.11ac standard, four beamforming feedback angles (two ϕ and two ψ) are needed to represent each of the 3×1 channels between the AP and the STAs. In our setup, the angle quantization process uses $b_\phi = 9$ bits and $b_\psi = 7$ bits for the feedback angles ϕ and ψ respectively. UDP data streams are sent from the AP to the STAs in the downlink direction to trigger

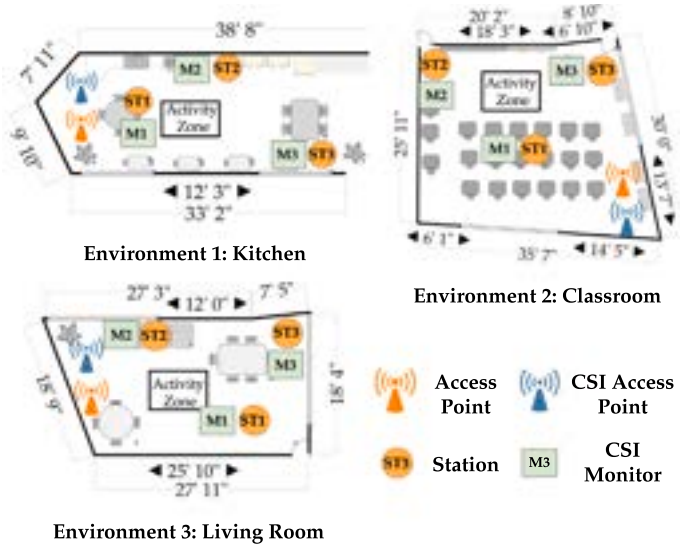

Fig. 10. Experimental setups for data collection.

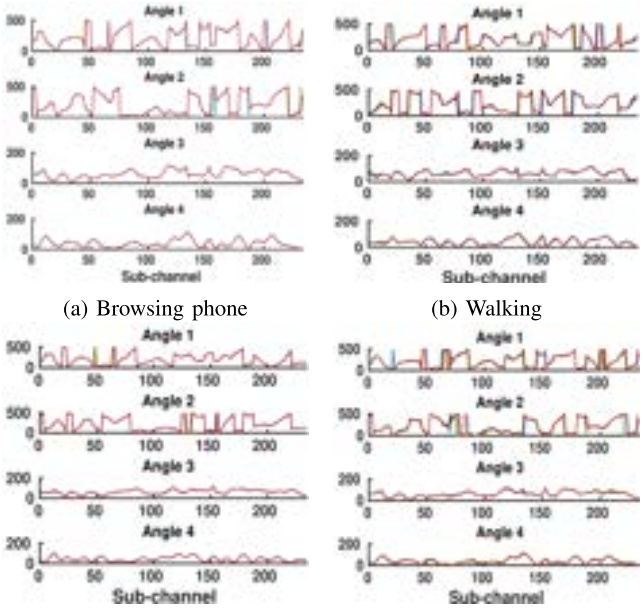

Fig. 11. BFAs for each sub-channel for four activities. Each plot shows the values of 10 different packets (superimposed lines with different colors). The x-axis reports the indices of the sensed sub-channels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the channel sounding. The BFI frames are captured with the Wireshark network protocol analyzer running on an off-the-shelf laptop equipped with an Intel 9560NGW wireless-AC NIC set in monitor mode. However, note that any IEEE 802.11ac-compliant NIC set in monitor mode could be used for this purpose. Moreover, notice that the frame-capturing device does not need any direct link with the AP or the STAs. The only requirement is that the capture is performed on the wireless channel where the Wi-Fi network is operating. From the captured frames, the ϕ and the ψ angles are extracted for each of the STAs and used as input to the BeamSense learning framework (see Section 3.2). Fig. 11 shows a sample taken from our dataset. We plot the magnitude of the four collected beamforming angles for each of the 234 available sub-channels, for ten different packets and four activities. Fig. 11 remarks that the absolute values of the angles change quite significantly among different activities, while do not change significantly among different packets. This indicates that BFI-based sensing is a stable measurement of the channel propagation environment and thus, a strong candidate to be used within Wi-Fi sensing systems.

CSI Network Setup and Equipment. For comparative studies, CSI data has also been collected concurrently with the BFI frame capture. For this purpose, a Wi-Fi network consisting of an AP (referred to as *CSI AP*) and three STAs (referred to as *CSI monitors*) has been co-located with BeamSense network in the same environments, as depicted in Fig. 10. The network operates on the IEEE 802.11ac channel 42, i.e., the center frequency is f_c = 5.21 GHz and the bandwidth is 80 MHz. The AP is implemented with a Netgear Nighthawk X4S AC2600 router, while the CSI client is a PC APU2 board equipped with an Intel 9560NGW wireless-AC NIC. For the CSI extraction, three IEEE 802.11ac-compliant Asus RT-AC86U routers (referred to as *CSI monitors*) equipped with the Nexmon CSI extraction tool [8] have been deployed, as depicted in Fig. 10 in green. To have the same setup as in the MU-MIMO network, the CSI AP is enabled with $M = 3$ antennas whereas the CSI monitors are set up to sense the channel through $N = 1$ antenna over $N_{ss} = 1$ spatial stream each. UDP packets are sent from the CSI AP to the CSI client to trigger the channel estimation on the three CSI monitors.

Real-time Deployment of BeamSense. The BeamSense framework is deployed on a Linux-based workstation, configured to function as an edge server for efficient wireless data processing. The edge server is equipped with an Intel 9560NGW Wireless-AC network NIC, allowing it to directly capture BFAs frames without requiring direct access to the associated STAs or the network infrastructure. The edge server is positioned within the transmission range of the target network, ensuring the passive collection of BFAs data.

For computational efficiency, the server is powered by a high-performance Intel Core i7-12700 processor and an RTX A4000 GPU. These hardware components enable fast preprocessing of the captured BFAs data and facilitate the real-time execution of the BeamSense classification algorithm, ensuring low-latency decision-making. Once the BFAs frames are captured (as detailed in Section 4.1), the edge server utilizes our previously developed open-source tool, Wi-BFI [58], to extract the BFAs samples for all the active STAs. The raw data is then processed through a multi-stage pipeline, outlined in Fig. 5, using Python libraries such as NumPy and Pandas. The preprocessed data is subsequently forwarded through the trained BeamSense classifier for inference. The classification framework consists of two core components: a baseline CNN and our proposed FAMReS algorithm. Both models are implemented using Python's TensorFlow library, which supports the real-time execution required for practical deployment in edge-based environments.

4.2. Comparison between BFA and CSI -based sensing with co-located BFA stations and CSI monitors.

In the following, all the results are obtained with a time window size of 0.1 s with 10 frames/sample with the data of three subjects combined, unless specified otherwise.

4.2.1. Comparison between BFA and CSI-based sensing with co-located BFA stations and CSI monitors

Fig. 12 shows the classification accuracy of BeamSense as compared to the state-of-the-art CSI-based SignFi algorithm [55] in three different environments. For a baseline comparison, we consider M1, M2, & M3, and co-located ST1, ST2 & ST3 as the CSI collection device and BFA STAs respectively. We first evaluate the performance of BFA and CSI-based sensing using the minimalist data processing and the CNN architecture as referenced in Figs. 5 and 7 respectively. The accuracy of BeamSense in the kitchen, living room, and classroom is

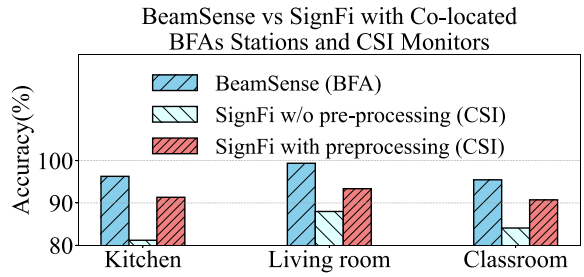

Fig. 12. BeamSense (BFA) vs. SignFi (CSI) performance.

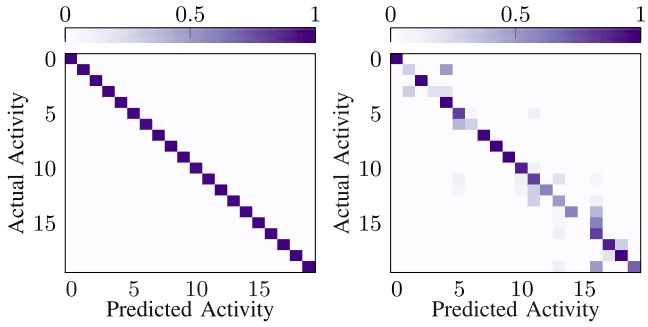

Fig. 13. Conf. matrices for BeamSense and SignFi.

respectively 96%, 99%, and 95.47% whereas SignFi reaches 81.19%, 87.99%, and 84.08% of accuracy respectively, resulting in a 12.6% accuracy decrease on average. We also show the performance of SignFi with the processing pipeline presented in [55], which unwraps the phase of each collected signal and then removes the phase noise by multiple linear regression based on the unwrapped phase across all subcarriers and antennas. The classification accuracy improves to 91.34%, 93%, and 90% in the kitchen, living room, and classroom environments, respectively. **Yet, BeamSense achieves better performance with minimal data preprocessing.**

To shed light on which classes are the hardest to classify with CSI-based sensing, Fig. 13 shows the confusion matrices obtained in the kitchen using BeamSense and SignFi without the custom preprocessing. The bottom five classes are browsing laptop (index 20), phone call (16), hands-up-down (15), clapping (02), and boxing (04), which are indeed among the hardest classes to distinguish.

Fig. 14 shows the performance of BeamSense and SignFi with the pre-processing in [55] evaluated in the kitchen as a function of the CSI and BFAs capture location, and the window size W. We can see that, for all three different locations, the performance of BeamSense and SignFi follow the same trend for $W = 1$, however, when increasing the window size, the performance of SignFi degrades in all the locations in comparison to the BeamSense performance. Specifically, the performance of SignFi drops by 79.25% when we switch from $W = 0.1$ to $W = 0.4$ whereas the BeamSense performance fluctuates only by 2.79%.

It is worth mentioning that, BFAs are affected by phase offsets and Automatic Gain Control (AGC) impairments as these hardware-related impairments percolate from CSI to the BFAs given the processing steps detailed in Section 3.1. However, compensating such offsets would require reconstructing the BFI from the BFAs introducing additional computation and increasing the latency of the system. Given the complexity of performing activity recognition through radio signals and to avoid such offset-removal preprocessing step, we addressed the sensing task through a learning-based algorithm that is effectively able to extract meaningful features from BFAs for activity recognition, reducing the effect of hardware-related offsets on the classification. To further reduce the effect of such offsets, the data from all the beamformees are jointly fed to our learning-based algorithm. As BFAs from different

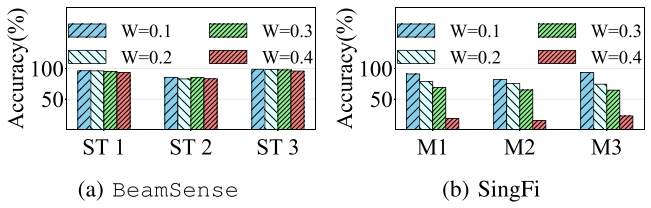

Fig. 14. BeamSense and SignFi performance with the variation of the window size W.

beamformees are affected by different offsets, their combination allows the neural network to effectively extract meaningful features for the sensing task, minimizing the effect of hardware impairments.

4.2.2. Comparison between BeamSense and CSI-based approaches for remote sensing

For evaluating the remote sensing performance, we consider both the BFA and CSI extraction tools do not have any direct access to the sensing location and the STAs of the sensing environment. Thus we place both the BFA and CSI extraction tools outside the sensing environment– beyond the concrete wall, without any direct access to the STAs of the sensing environment as presented in Fig. 15. The comparative performance analysis of BeamSense and SignFi for remote sensing is presented in Fig. 16. Results show that the performance of BeamSense does not hamper at all for any of the environments even if the extraction tool is placed beyond the wall at a remote location. On the contrary, the performance of SignFi with pre-processing decreases by 20.80%, 19.27%, and 19.83% respectively for the kitchen, living room, and classroom. This sudden plunge in SignFi performance is caused by the fact that the CSI tool captures the channel between itself and the AP whereas the BFA tool captures the channel between AP and all the STAs of the network. Thus, for remote sensing, BeamSense achieves better performance in comparison to the CSI based approaches including SignFi.

For evaluating the remote sensing performance, we consider a situation when both the BFAs and the CSI extraction devices do not have any direct access to the sensing location and the STAs placed in the sensing environment. Thus we place both the BFAs and CSI extraction devices outside the sensing environment – i.e., beyond a concrete wall – without any direct access to the STAs deployed in the sensing environment, as presented in Fig. 15. The comparative performance analysis of BeamSense and SignFi for remote sensing is presented in Fig. 16. The results show that the performance of BeamSense does not hamper at all for any of the environments even if the extraction device is placed beyond the wall at 5 m from the AP. On the contrary, the performance of SignFi with pre-processing decreases by 20.80%, 19.27%, and 19.83% respectively for the kitchen, living room, and classroom. This sudden plunge in SignFi performance is caused by the fact that the CSI extraction tool captures the channel between the device where it is installed and the connected AP, whereas the BFAs tool captures the channel between AP and all the STAs of the network, independently on the device where the tool is installed. Thus, for remote sensing, BeamSense achieves higher accuracy in comparison to the CSI based approaches including SignFi.

4.2.3. Performance as a function of the spatial diversity

Fig. 17 presents the performance of BeamSense when trained with data from a single STA and with the combined data. First, we notice that the single STA data is almost always a very stable measurement, with the accuracy remaining high in most of cases. However, we notice that some STAs perform worse than others, especially ST2 in the kitchen, and ST2 and ST3 in the classroom. Indeed, due to the physical location of these STAs, the communication channels between them and the AP might be in deep fade causing BeamSense to perform

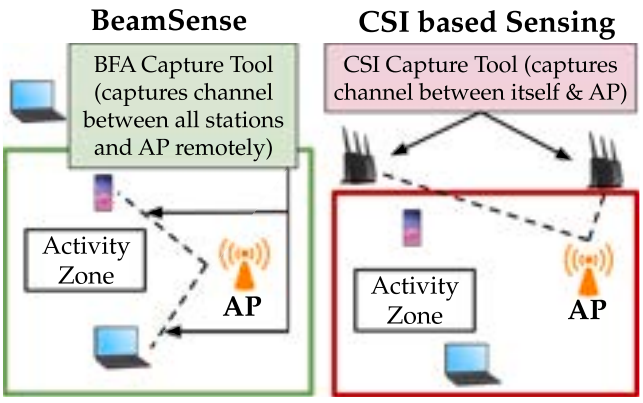

Fig. 15. Experimental setups for remote sensing test.

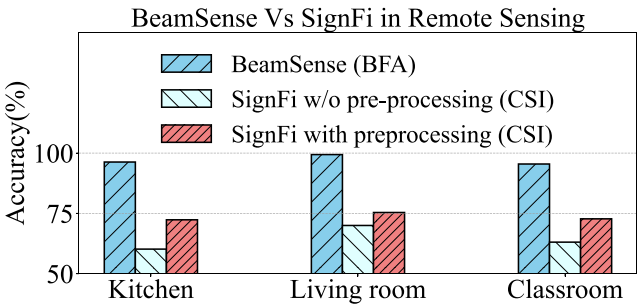

Fig. 16. `BeamSense` (BFA) vs. SignFi (CSI) performance for remote sensing.

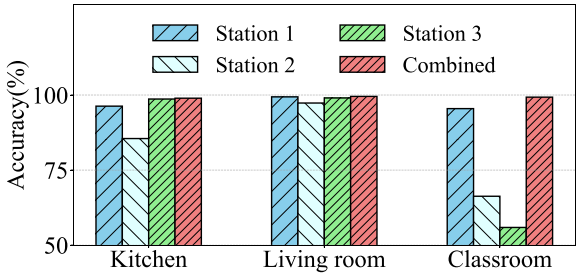

Fig. 17. Impact of the spatial diversity of the beamformees at three different environments.

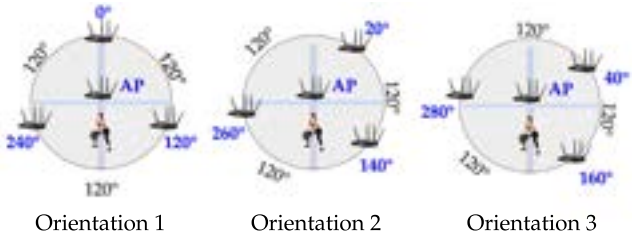

Fig. 18. Different setup/orientation of the STAs.

poorly. However, by aggregating the spatially diverse STA data, **the overall accuracy is improved by up to 43.81%** in the classroom. Given the variability of the Wi-Fi channel, considering different STA locations imply obtaining completely different angles for the same activity, even in the same environment, as shown in Fig. 17. To further investigate the sensing performance as a function of the STA location, we conduct an experiment in the kitchen entailing three different STA locations as depicted in Fig. 18. The first placement is referred to as

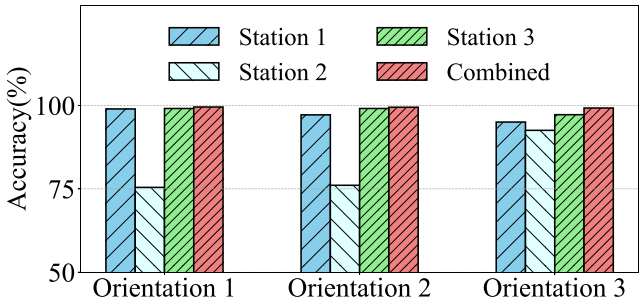

Fig. 19. Impact of different orientations of beamformees in the same environment (Kitchen).

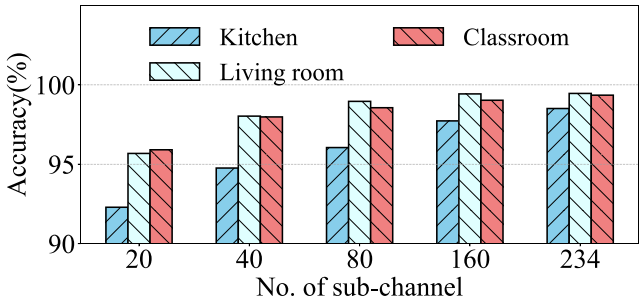

Fig. 20. `BeamSense` accuracy as a function of the number of sensed sub-channels.

'Orientation 1' while 'Orientation 2' and 'Orientation 3' are obtained by physically rotating each STA by 20° clockwise, which corresponds to placing the STA around 2 m away from the previous location. Fig. 19 shows the accuracy of `BeamSense` in the kitchen when using data collected through each of the three setups. We notice some of the STAs individually perform poorly in some orientations due to the physical location of the STA. However, `BeamSense` performs very well when combining all the STAs: the accuracy is 99.53%, 99.46%, and 99.23% respectively in Orientation 1, Orientation 2, and Orientation 3. Therefore, **multi-STA sensing should be preferred over single-STA sensing whenever possible**.

4.2.4. *Evaluation of angle and sub-channel resolution*

It is known that Wi-Fi sensing performs worse when lowering the number of sub-channel considered in the sensing process [31,69]. Extensive feature extraction or higher sampling frequency can be utilized, at the cost of increasing the computational burden and intensifying preprocessing steps, as well as increasing the computational complexity of the learning process. For this reason, we investigate the trade-off between the number of angles and sub-channels considered for sensing and the sensing performance.

Fig. 20 shows the accuracy of `BeamSense` as a function of the number of sub-channels utilized in the learning process. To down-sample the sub-channels, we take the first 20, 40, 80, and 160 sub-channels, to emulate sensing systems with smaller available bandwidths. As expected, the accuracy decreases by 6.31%, 3.80%, and 3.46% respectively for the kitchen, living room, and classroom when we switch from 234 to 20 sub-channels. However, notice that this operation drastically decreases the input tensor dimension from $10 \times 234 \times 12 = 28\,080$ to $10 \times 20 \times 12 = 2400$, implying that sub-channel resolution decreases the computational burden by 10× while maintaining the accuracy above 92% in all the considered scenarios.

Fig. 21 shows `BeamSense` performance as a function of the number of angles considered for sensing. STA1 is considered for angle 1, angle 2, angle 3, angle 4, and the combination of four angles, whereas STA1 and STA2 are considered for the combination of eight angles, and all three stations are considered for the combination of 12 angles. Fig. 21

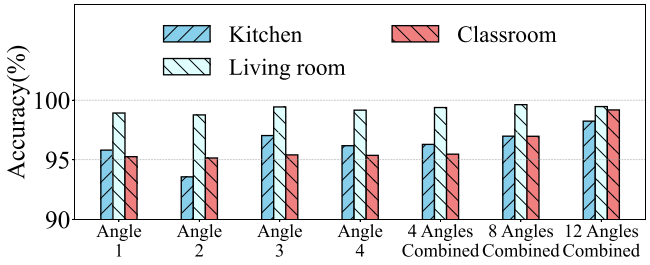

Fig. 21. BeamSense accuracy as a function of the number of the angles considered.

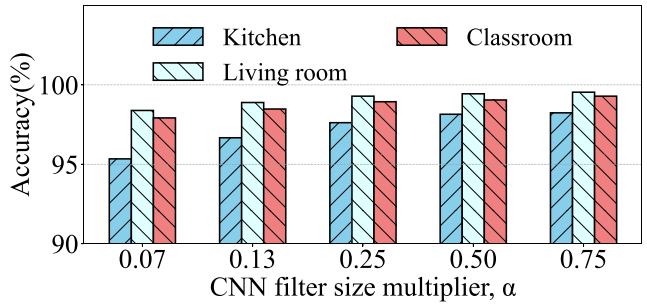

Fig. 22. BeamSense accuracy as a function of the CNN filter sizes.

shows that the accuracy decreases by 1.98%, 0.16%, and 2.22% in the kitchen, living room and classroom respectively when considering a single angle with respect to the combination of 12 angles. Even though the above results show no significant variation in performance even if the angle resolution is decreased from 12 angles combined to any individual angle, we suggest aggregating at least the angles of two spatially diverse STAs to obtain a robust algorithm.

4.2.5. Evaluation of CNN filter size

To further investigate the trade-off between computation complexity and accuracy, we introduce a width multiplier $\alpha \in (0, 1]$ to each layer of the CNN-based classifier. For a given number of input channels C and output channels Z, they become αC and αZ after applying the multiplier. Hence, the computation complexity will be reduced by α^2 roughly. Applying the width multiplier α to BeamSense, the channel size of each conv-block becomes $\alpha \times 128$, $\alpha \times 64$, $\alpha \times 32$, respectively. Fig. 22 shows how the accuracy changes when applying width multiplier $\alpha \in \{0.07, 0.13, 0.25, 0.5, 0.75\}$. BeamSense accuracy, averaged over the three environments, is 97.22%, 98.01%, 98.62%, 98.88%, and 99.02%, respectively. **As the CNN width decreases from 0.75 to 0.07, the accuracy drops marginally by 1.8%**. This observation indicates that BeamSense can adapt to limited computation resources and latency-sensitive cases by sacrificing little accuracy.

4.3. Evaluation of BeamSense with FAMReS algorithm

To address the challenge of generalization to unseen environments and subjects, we have proposed FAMReS in Section 3.2.1. We compare the performance of FAMReS with the state-of-the-art FSL algorithm OneFi [44] and the transfer learning (TL) algorithm presented in WiTransfer [56] for cross-domain WiFi sensing. BeamSense utilizes the FAMReS algorithm to effectively adapt with just 15 s of new data, equivalent to 150 BFAs samples from an unseen environment or subject. This approach achieves an impressive average accuracy of 92.85% when tested in new, unseen environments and 91.87% for previously unseen subjects. This adaptation requires only 36.37 s on average, on a Linux machine with Nvidia A100 GPU, demonstrating its practicality for real-time applications. These results highlight the necessity of FAMReS in enhancing the robustness and versatility of our model, enabling it to maintain high performance across diverse deployment scenarios.

Fig. 23(a) shows that with only 15 s of new data, FAMReS can adapt to new environments with an average accuracy of 94.97%, 90.51%, and 93.09% when trained in the kitchen, living room, and classroom respectively. On the other hand, WiTransfer achieves accuracy of 13.4%, 18.02%, and 16.52% respectively in the three different configurations. The reason relies on the fact that the WiTransfer pre-trained model is optimized for a specific configuration and the adaptation to new configurations through transfer learning requires a considerable amount of data to get rid of the data bias, i.e., 15 s of new data are not enough for WiTransfer to achieve satisfactory accuracy. OneFi achieves an accuracy of 64.72%, 63.36%, and 63.24% respectively in new environments when trained in the kitchen, living room, and classroom. Although the results show that OneFi can generalize to new environments to some extent, FAMReS performs better since it fine-tunes the whole model and learns shared information across different tasks by using meta-learning. On the contrary, OneFi utilizes information from one task and only fine-tunes the last layers of the neural network model that performs the classification. The performance of the algorithms in generalizing over new subjects is presented in Fig. 23(b). The results show a trend similar to the generalization over unseen environments. FAMReS is 73.41% more accurate than WiTransfer and 24.81% more accurate than OneFi on average, confirming the benefit of the few-shot learning approach adopted in this current work. We finally evaluated the performance of FAMReS as a function of different setups as discussed in Section 4.2.3. Fig. 23(c) shows that FAMReS achieves an accuracy of 90.93%, 94.38%, and 93.20% when trained with data collected in setup 1, setup 2, and setup 3 respectively, and tested in the other setups. FAMReS outperforms WiTransfer and OneFi by 74.88% and 27.28% on average when used in the new unseen setups. The generalization performance achieved when using the base CNN model (presented in Fig. 7) is also reported in Fig. 23 for comparison. The results show that the base CNN is unable to adapt to new environments, subjects, and orientations, reaching an average accuracy of 6.21%.

4.4. BeamSense performance as a function of the time variable δ

The time variable δ represents the duration of the period in which FAMReS gathers BFAs samples in a new environment for fine-tuning. Large δ values correspond to more samples used for fine-tuning while small δ corresponds to fast adaptation but may lead to sub-optimal performance. Hence, the quality of a generalization algorithm can be measured by evaluating how the sensing performance varies when changing δ. The fewer samples are needed by an algorithm to generalize effectively over unseen situations, the better that approach is for practical deployments. In this section, we compare the sensing accuracy of BeamSense with the other considered sensing algorithms when generalizing to new environments using different δ values. Fig. 24 illustrates the performance of the different sensing algorithms as a function of δ. The results indicate that as δ decreases from 30 s to 10 s, FAMReS experiences only a modest accuracy drop of 5.30% and 11.13% on average in unseen environments and subjects, respectively. In contrast, WiTransfer's performance deteriorates sharply with a short δ, demonstrating that, without a meta-learning phase, transfer learning demands more data for adaptation. While OneFi remains more stable than WiTransfer, its accuracy drops to 52.26% and 43.92% in unseen environments and subjects, respectively – 39% lower than FAMReS. This confirms the advantage of FAMReS strategy, which fine-tunes the entire network rather than only the classifier, as done by OneFi.

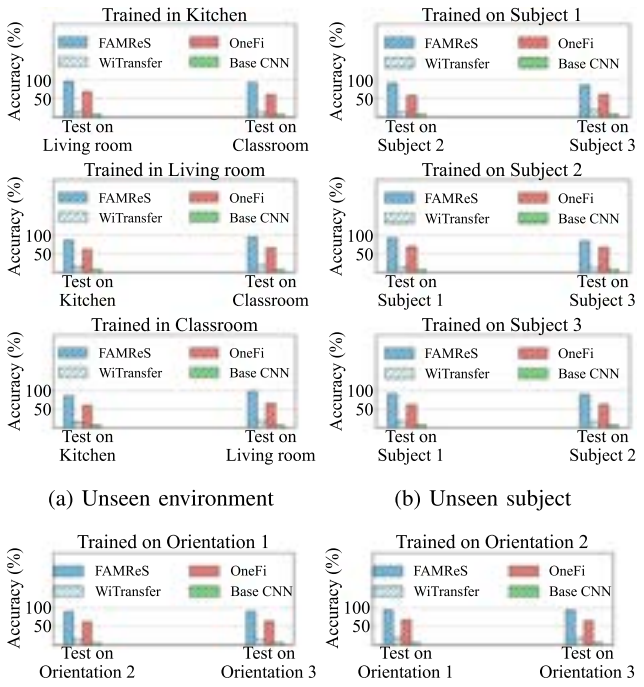

Fig. 23. Comparative analysis of BeamSense in unseen environments, subjects and orientations.

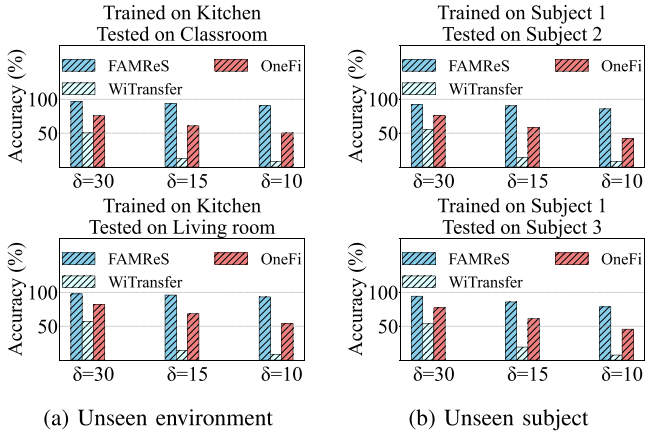

Fig. 24. Comparative analysis of BeamSense as a function of time variable, δ.

4.5. BeamSense performance as a function of the number of subjects in the training dataset

We analyze the performance of BeamSense as a function of the number of subjects considered at training time. Fig. 25(a) shows the classification accuracy of BeamSense with the baseline CNN (i.e., without generalization capabilities) when trained and tested on an increasing number of subjects (1, 3, and 6). Here, data from all the subjects considered at the testing time were included in the training – training and testing datasets contain BFAs from all the subjects but are disjoint in time. The results show that the BeamSense accuracy remains above 95% in all the cases in the different environments revealing that the algorithm effectively learns activity-specific features when trained with data associated with different subjects. In Fig. 25(b) we evaluate the performance of BeamSense as a function of the number of subjects considered at training time when using FAMReS for generalizing over unknown subjects. The results show that the

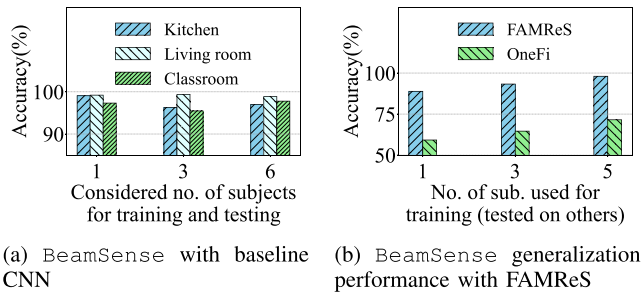

Fig. 25. BeamSense performance as a function of number of subjects.

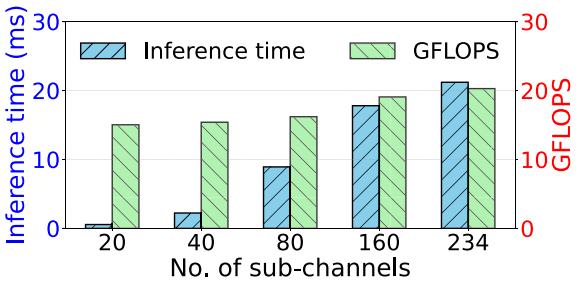

Fig. 26. Inference time and computational complexity of BeamSense with different number of sub-channels.

performance improves by 20.8% and 10.3% with OneFi and FAMReS when the considered number of subjects for training is increased from 1 to 5. This means that the higher the number of subjects in the training set, the more the network is able to focus on subject-independent features that provide generalizability over subjects for whom examples of activity-related traces were not provided to the learning algorithm during training.

4.6. Inference time, computational complexity, and energy efficiency of BeamSense

We analyze BeamSense in terms of inference time, computational resources, and energy efficiency. Fig. 26 presents the inference time and the number of Floating Point Operations Per Seconds (FLOPs) for the execution of BeamSense when using different numbers of OFDM sub-channels for sensing.

The BeamSense model with 234 sub-channels requires 20.28 ms and 21.20 GFLOPs, while the model with 20 sub-channels takes 15.03 ms and 0.559 GFLOPs. Note that the energy consumption of the Wi-Fi devices remains unaffected, as BeamSense runs entirely on the server. Indeed, BeamSense uses BFAs, which are transmitted unencrypted from STAs to the AP in accordance with IEEE 802.11 standards. The BFAs of the multiple STAs are recorded in a single capture at the server without the need to modify the Wi-Fi system. Since BeamSense operates with standard Wi-Fi transmissions, no additional energy is consumed by the Wi-Fi devices themselves.

To study the energy efficiency of BeamSense, we evaluated the computational complexity of executing it. We provide an energy consumption estimation based on GFLOPS, which is independent of the hardware used. For each GFLOP, we estimate an energy consumption of approximately 0.23148 μAh. Therefore, for 234 sub-channels requiring 21.20 GFLOPs, the energy consumption is approximately 0.0049 mAh, and for 20 sub-channels requiring 0.559 GFLOPs, it is about 0.00013 mAh. This analysis demonstrates that BeamSense's energy demands are minimal and confined to the computational server, with no additional burden placed on the Wi-Fi devices themselves.

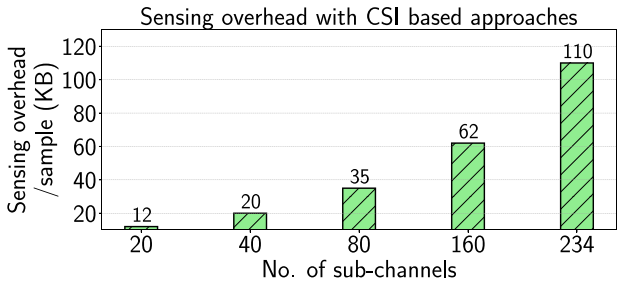

Fig. 27. Sensing overhead per input tensor with CSI based approach for a single monitor for different number of sub-channels.

4.7. BeamSense sensing overhead analysis

Given the limited availability of radio spectrum resources – which should be shared by communications and sensing services – the efficiency of a sensing system is strongly linked with its sensing overhead, i.e., the channel occupancy for sensing data transmission. To minimize this overhead, in our system, the BFAs samples are directly acquired by the BeamSense server given that they are transmitted unencrypted over the air by each STAs in the network to the AP. This eliminates the need for edge offloading thereby minimizing channel occupancy. In turn, BeamSense operates without occupying the wireless channel, regardless of the number of sensing STAs, transmission bandwidth, and MIMO configurations. On the other hand, CSI-based sensing methods, e.g., OneFi and WiTransfer, introduce additional sensing overhead as, unlike BeamSense, sensing data is captured at each STA in the network. This requires direct access to the sensing device and the occupation of spectrum resourced for the offloading of the captured CSI samples to the edge server. This makes the overall channel occupation of the CSI-based approaches depend on the total number of STAs included in the sensing system. Fig. 27 reports the sensing overhead of a CSI input tensor, i.e., 10 CSI samples as depicted in Fig. 5, for a single STA operating on a channel with 80 MHz of bandwidth. The results indicate that even a single input tensor captured within a 0.1-s time window occupies 110 KB when considering 234 OFDM sub-channels, which reduces to 35 KB for 80 sub-channels. Overall, the sensing overhead increases exponentially with CSI-based approaches like OneFi and WiTransfer. Therefore, employing state-of-the-art CSI-based approaches with a high sampling rate or more STAs included in the sensing system inevitably saturates the network by increasing the sensing overhead. In contrast, BeamSense entails zero-redundant channel occupancy, regardless of the sampling rate or the number of sensing STAs.

4.8. Evaluating BeamSense performance in smart home applications: A case study on human gesture recognition

To demonstrate the generalizability of BeamSense across various applications, we further evaluate its performance considering a different smart home application: human gesture recognition. In this task, two subjects perform gestures representing digits 0 through 9 for three minutes per gesture across three different orientations in a conference room, as shown in Fig. 28. The data preprocessing and classification procedures follow the steps summarized in Fig. 5 and detailed in Section 3.2. The results, presented in Fig. 29, illustrate the system's gesture recognition performance across the three orientations. The gesture recognition performance across three orientations demonstrates consistently high accuracy for all stations, with combined results exceeding 98% in every case. In Orientation 1, Station 1 performs best with 96.95% accuracy, while Orientation 2 shows slightly lower but still robust accuracy across all stations, ranging from 91.54% to 93.44%. Orientation 3 yields the highest individual station

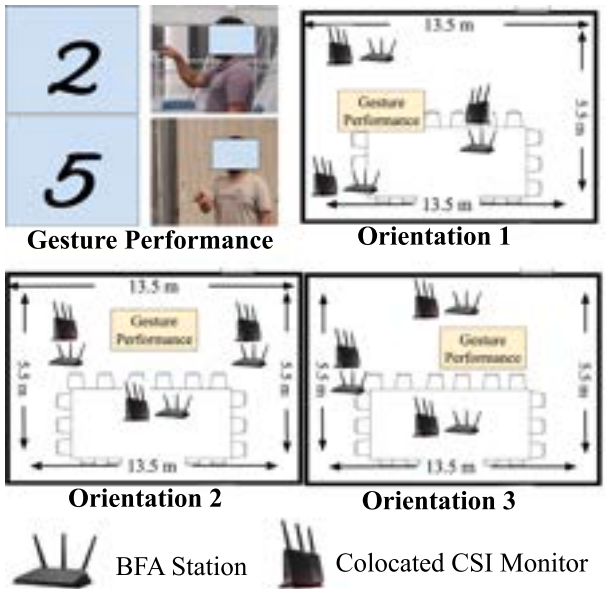

Fig. 28. Experimental setups of human gesture recognition. Two different subjects perform 10 different gestures (digits 0–9) in three different orientations in the same environment.

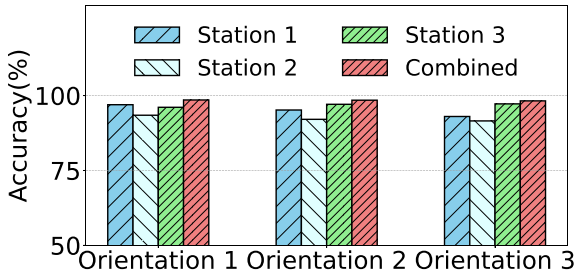

Fig. 29. Performance of human gesture recognition in three different orientations of beamformees.

performances, particularly for Stations 2 and 3, which both exceed 97%. The consistently strong combined results highlight the system's ability to integrate data from multiple stations, ensuring highly accurate gesture recognition across varying orientations and environments. Figs. 30 and 31 presents the orientation and subject generalization performance of FaMReS (learning approach of BeamSense) respectively. FaMReS demonstrates a clear advantage in domain generalization performance compared to both OneFi and WiTransfer. For instance, when trained in Orientation 1 and tested on Orientation 2, FaMReS achieves 90.00% accuracy, which is 23.32% higher than OneFi's 72.68%, and 70.54% higher than WiTransfer's 19.46%. Similarly, when trained in Orientation 3 and tested in Orientation 1, FaMReS achieves 92.28%, outperforming OneFi by 25.04% (OneFi's accuracy being 67.24%) and WiTransfer by 72.62% (WiTransfer's accuracy being 19.66%).

In subject generalization, FaMReS continues to show substantial improvements. When trained with Sub 1 and tested with Sub 2, FaMReS achieves 93.00%, which is 20.32% higher than OneFi's 72.68% and an impressive 63.54% higher than WiTransfer's 29.46%. Similarly, when trained with Sub 2 and tested with Sub 1, FaMReS achieves 94.94%, outperforming OneFi by 29.17% (OneFi's accuracy being 65.77%) and WiTransfer by 57.60% (WiTransfer's accuracy being 37.34%). These results underscore the strong generalization capability of FaMReS, significantly surpassing both competing methods across different orientations and subjects.

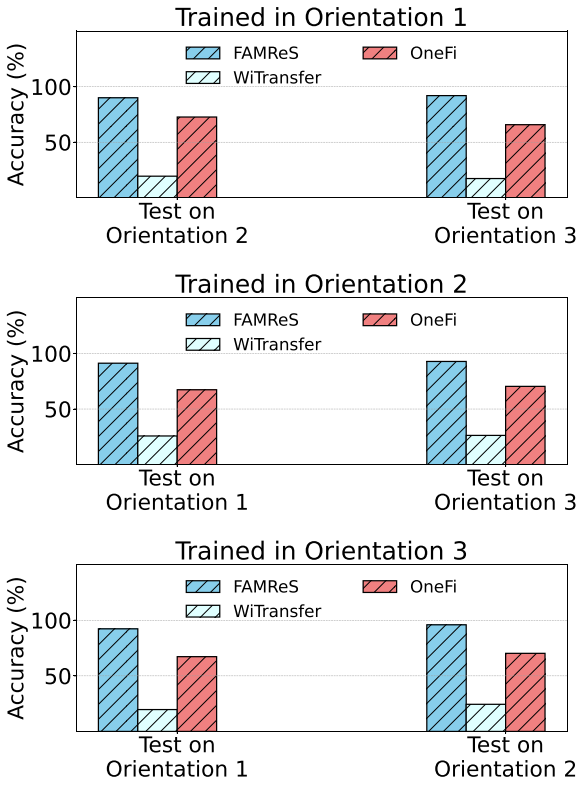

Fig. 30. Comparative analysis of BeamSense performance for gesture recognition when considering new orientations not included in the training dataset.

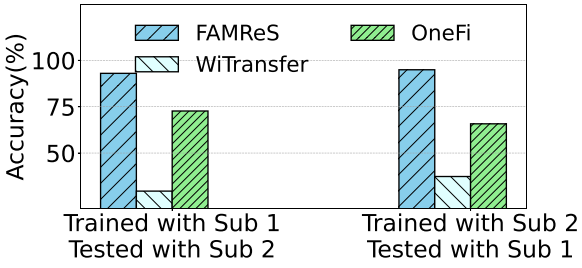

Fig. 31. Comparative analysis of BeamSense performance for gesture recognition when considering new subjects not included in the training dataset.

5. Conclusions and remarks

In this article, we have proposed BeamSense, a novel approach to Wi-Fi sensing based on the usage of MU-MIMO BFAs. Conversely from CSI-based approaches, (i) the BFAs can be easily recorded by off-the-shelf devices without MIMO capabilities and without any firmware modification; (ii) a single frame of the BFAs capture the multiple channels between the AP and the STAs, thus achieving a much better sensing granularity. BeamSense includes a few-shot learning (FSL)-based classification algorithm to adapt to new environments and subjects with few additional data. We have evaluated BeamSense through an extensive data collection campaign involving three subjects performing twenty different activities in three indoor environments. We have compared our approach with traditional CSI-based sensing approaches and show that BeamSense improves the accuracy by 10% on the average, while our FSL-based approach improves accuracy by up to 30% when compared with SOTA domain adaptive sensing models. We hope that this work will pave the way for additional research on BFAs and BFI-based Wi-Fi sensing.

CRediT authorship contribution statement

Khandaker Foysal Haque: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Milin Zhang:** Writing – original draft, Software, Investigation. **Francesca Meneghello:** Writing – review & editing, Validation, Supervision, Project administration. **Francesco Restuccia:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Funding acquisition.

Acknowledgments

This work has been funded in part by the National Science Foundation under grants CNS-2134973, ECCS-2229472; in part by the Air Force Office of Scientific Research under contract number FA9550-23-1-0261 and in part by the Office of Naval Research under award number N00014-23-1-2221. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the author(s) and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of U.S. Air Force, U.S. Navy or the U.S. Government.

Data availability

Dataset is available at – https://ieee-dataport.org/documents/dataset-human-activity-classification-mu-mimo-bfi-and-csi.

References

[1] Wi-Fi Alliance, The Economic Value of Wi-Fi: A Global View (2018 and 2023), 2021, https://tinyurl.com/EconWiFi.
[2] Y. Ma, S. Arshad, S. Muniraju, E. Torkildson, E. Rantala, K. Doppler, G. Zhou, Location- and Person-Independent Activity Recognition with WiFi, Deep Neural Networks, and Reinforcement Learning, ACM Trans. Internet Things 2 (1) (2021).
[3] X. Wang, C. Yang, S. Mao, TensorBeat: Tensor Decomposition for Monitoring Multiperson Breathing Beats with Commodity WiFi, ACM Trans. Intell. Syst. Technol. 9 (1) (2017) 1–27.
[4] H. Zhu, F. Xiao, L. Sun, R. Wang, P. Yang, R-TTWD: Robust Device-Free Through-the-Wall Detection of Moving Human with WiFi, IEEE J. Sel. Areas Commun. 35 (5) (2017) 1090–1103.
[5] Y. Ma, G. Zhou, S. Wang, WiFi Sensing with Channel State Information: A Survey, ACM Comput. Surv. 52 (3) (2019) 1–36.
[6] D. Halperin, W. Hu, A. Sheth, D. Wetherall, Tool Release: Gathering 802.11n Traces with Channel State Information, ACM SIGCOMM Comput. Commun. Rev. 41 (1) (2011) 53.
[7] Y. Xie, Z. Li, M. Li, Precise Power Delay Profiling with Commodity Wi-Fi, in: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, 2015.
[8] F. Gringoli, M. Schulz, J. Link, M. Hollick, Free Your CSI: A Channel State Information Extraction Platform For Modern Wi-Fi Chipsets, in: Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization, Association for Computing Machinery, New York, NY, USA, 2019, pp. 21–28.
[9] Z. Jiang, T.H. Luan, X. Ren, D. Lv, H. Hao, J. Wang, K. Zhao, W. Xi, Y. Xu, R. Li, Eliminating the Barriers: Demystifying Wi-Fi Baseband Design and Introducing the PicoScenes Wi-Fi Sensing Platform, IEEE Internet Things J. 9 (6) (2022) 4476–4496.
[10] F. Gringoli, M. Cominelli, A. Blanco, J. Widmer, AX-CSI: Enabling CSI Extraction on Commercial 802.11ax Wi-Fi Platforms, in: Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental Evaluation & CHaracterization, Association for Computing Machinery, New York, NY, USA, 2022, pp. 46–53.
[11] E. Aryafar, N. Anand, T. Salonidis, E.W. Knightly, Design and Experimental Evaluation of Multi-User Beamforming in Wireless LANs, in: Proc. of the 16th Annual International Conference on Mobile Computing and Networking (MobiCom), New York, NY, USA, 2010.
[12] N. Bahadori, J. Ashdown, F. Restuccia, ReWiS: Reliable Wi-Fi Sensing Through Few-Shot Multi-Antenna Multi-Receiver CSI Learning, in: Proceedings of the IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Los Alamitos, CA, USA, 2022.

[13] F. Meneghello, D. Garlisi, N. Dal Fabbro, I. Tinnirello, M. Rossi, SHARP: Environment and Person Independent Activity Recognition with Commodity IEEE 802.11 Access Points, IEEE Trans. Mob. Comput. (2022) 1–16.

[14] J. Liu, H. Liu, Y. Chen, Y. Wang, C. Wang, Wireless sensing for human activity: A survey, IEEE Commun. Surv. Tutor. 22 (3) (2019) 1629–1645.

[15] C.-F. Hsieh, Y.-C. Chen, C.-Y. Hsieh, M.-L. Ku, Device-free indoor human activity recognition using Wi-Fi RSSI: machine learning approaches, in: 2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan), IEEE, 2020, pp. 1–2.

[16] W. Wang, A.X. Liu, M. Shahzad, K. Ling, S. Lu, Understanding and modeling of wifi signal based human activity recognition, in: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, 2015, pp. 65–76.

[17] M. Zhang, Z. Fan, R. Shibasaki, X. Song, Domain Adversarial Graph Convolutional Network Based on RSSI and Crowdsensing for Indoor Localization, 2022, arXiv preprint arXiv:2204.05184.

[18] S. Depatla, Y. Mostofi, Crowd Counting through Walls Using WiFi, in: Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom), Athens, Greece, 2018.

[19] P. Ssekidde, O. Steven Eyobu, D.S. Han, T.J. Oyana, Augmented CWT features for deep learning-based indoor localization using WiFi RSSI data, Appl. Sci. 11 (4) (2021) 1806.

[20] N. Singh, S. Choe, R. Punmiya, Machine learning based indoor localization using Wi-Fi RSSI fingerprints: an overview, IEEE Access (2021).

[21] W. Li, M.J. Bocus, C. Tang, S. Vishwakarma, R.J. Piechocki, K. Woodbridge, K. Chetty, A Taxonomy of WiFi Sensing: CSI vs passive Wi-Fi Radar, in: 2020 IEEE Globecom Workshops (GC Wkshps, IEEE, 2020, pp. 1–6.

[22] W. Li, R.J. Piechocki, K. Woodbridge, C. Tang, K. Chetty, Passive WiFi Radar for Human Sensing Using a Stand-alone Access Point, IEEE Trans. Geosci. Remote Sens. 59 (3) (2020) 1986–1998.

[23] C. Tang, W. Li, S. Vishwakarma, F. Shi, S. Julier, K. Chetty, People counting using multistatic passive WiFi radar with a multi-input deep convolutional neural network, in: Radar Sensor Technology XXVI, SPIE, 2022.

[24] C. Tang, W. Li, S. Vishwakarma, K. Chetty, S. Julier, K. Woodbridge, Occupancy detection and people counting using WiFi passive radar, in: 2020 IEEE Radar Conference (RadarConf20), IEEE, 2020, pp. 1–6.

[25] B. Huang, G. Mao, Y. Qin, Y. Wei, Pedestrian flow estimation through passive wifi sensing, IEEE Trans. Mob. Comput. 20 (4) (2019) 1529–1542.

[26] Q. Bu, X. Ming, J. Hu, T. Zhang, J. Feng, J. Zhang, TransferSense: towards environment independent and one-shot wifi sensing, Pers. Ubiquitous Comput. 26 (3) (2022) 555–573.

[27] B. Korany, H. Cai, Y. Mostofi, Multiple People Identification Through Walls Using Off-the-Shelf WiFi, IEEE Internet Things J. 8 (8) (2021) 6963–6974.

[28] Y. Zeng, P.H. Pathak, P. Mohapatra, WiWho: WiFi-based Person Identification in Smart Spaces, in: Proceedings of ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN, IEEE, 2016, pp. 1–12.

[29] E. Soltanaghaei, R.A. Sharma, Z. Wang, A. Chittilappilly, A. Luong, E. Giler, K. Hall, S. Elias, A. Rowe, Robust and practical WiFi human sensing using on-device learning with a domain adaptive model, in: Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, 2020, pp. 150–159.

[30] S. Liu, Y. Zhao, F. Xue, B. Chen, X. Chen, DeepCount: Crowd counting with WiFi via deep learning, 2019, arXiv preprint arXiv:1903.05316.

[31] Y. Zeng, D. Wu, J. Xiong, J. Liu, Z. Liu, D. Zhang, MultiSense: Enabling Multi-person Respiration Sensing with Commodity WiFi, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. (IMWUT) 4 (3) (2020) 1–29.

[32] C. Shi, T. Zhao, Y. Xie, T. Zhang, Y. Wang, X. Guo, Y. Chen, Environment-independent In-baggage Object Identification Using WiFi Signals, in: Proceedings of IEEE International Conference on Mobile Ad Hoc and Smart Systems, MASS, IEEE, 2021.

[33] Y. Ren, S. Tan, L. Zhang, Z. Wang, Z. Wang, J. Yang, Liquid level sensing using commodity wifi in a smart home environment, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 4 (1) (2020) 1–30.

[34] Y. He, Y. Chen, Y. Hu, B. Zeng, WiFi vision: Sensing, recognition, and detection with commodity MIMO-OFDM WiFi, IEEE Internet Things J. 7 (9) (2020) 8296–8317.

[35] Y. Ren, Z. Wang, S. Tan, Y. Chen, J. Yang, Winect: 3D Human Pose Tracking for Free-form Activity Using Commodity WiFi, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 5 (4) (2021) 1–29.

[36] Y. Ren, Z. Wang, S. Tan, Y. Chen, J. Yang, Tracking Free-Form Activity Using WiFi Signals, in: Proceedings of the 27th Annual International Conference on Mobile Computing and Networking, 2021, pp. 816–818.

[37] W. Jiang, H. Xue, C. Miao, S. Wang, S. Lin, C. Tian, S. Murali, H. Hu, Z. Sun, L. Su, Towards 3D Human Pose Construction Using Wifi, in: Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, MobiCom '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 1–14.

[38] M. Zhao, T. Li, M. Abu Alsheikh, Y. Tian, H. Zhao, A. Torralba, D. Katabi, Through-wall human pose estimation using radio signals, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7356–7365.

[39] M. Muaaz, A. Chelli, M.W. Gerdes, M. Pätzold, Wi-Sense: A passive human activity recognition system using Wi-Fi and convolutional neural network and its integration in health information systems, Ann. Telecommun. 77 (3) (2022) 163–175.

[40] Y. Ge, A. Taha, S.A. Shah, K. Dashtipour, S. Zhu, J.M. Cooper, Q. Abbasi, M. Imran, Contactless WiFi Sensing and Monitoring for Future Healthcare-Emerging Trends, Challenges and Opportunities, IEEE Rev. Biomed. Eng. (2022).

[41] B. Korany, C.R. Karanam, H. Cai, Y. Mostofi, Teaching RF to Sense without RF Training Measurements, in: Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, IMWUT, Vol. 4, 2020.

[42] B. Wei, W. Hu, M. Yang, C.T. Chou, From Real to Complex: Enhancing Radio-based Activity Recognition Using Complex-Valued CSI, ACM Trans. Sensor Netw. 15 (3) (2019) 1–32.

[43] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, Z. Yang, Zero-Effort Cross-Domain Gesture Recognition with Wi-Fi, in: Proceedings of the ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), 2019.

[44] R. Xiao, J. Liu, J. Han, K. Ren, OneFi: One-Shot Recognition for Unseen Gesture via COTS WiFi, in: Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SenSys), 2021, pp. 206–219.

[45] H.F.T. Ahmed, H. Ahmad, C. Aravind, Device free human gesture recognition using Wi-Fi CSI: A survey, Eng. Appl. Artif. Intell. 87 (2020) 103281.

[46] A. Khalili, A.-H. Soliman, M. Asaduzzaman, A. Griffiths, Wi-Fi sensing: applications and challenges, J. Eng. 2020 (3) (2020) 87–97.

[47] I. Nirmal, A. Khamis, M. Hassan, W. Hu, X. Zhu, Deep Learning for Radio-Based Human Sensing: Recent Advances and Future Directions, IEEE Commun. Surv. Tutor. 23 (2) (2021) 995–1019.

[48] L. Guo, L. Wang, C. Lin, J. Liu, B. Lu, J. Fang, Z. Liu, Z. Shan, J. Yang, S. Guo, Wiar: A Public Dataset for WiFi-based Activity Recognition, IEEE Access 7 (2019) 154935–154945.

[49] S. Ding, Z. Chen, T. Zheng, J. Luo, RF-Net: A Unified Meta-Learning Framework for RF-Enabled One-Shot Human Activity Recognition, in: Proceedings of the 18th Conference on Embedded Networked Sensor Systems (SenSys 2020), Association for Computing Machinery, New York, NY, USA, 2020, pp. 517–530.

[50] B. Bloessl, M. Segata, C. Sommer, F. Dressler, An IEEE 802.11 a/g/p OFDM Receiver for GNU Radio, in: Proceedings of the Second Workshop on Software Radio Implementation Forum, 2013, pp. 9–16.

[51] X. Wang, K. Niu, J. Xiong, B. Qian, Z. Yao, T. Lou, D. Zhang, Placement Matters: Understanding the Effects of Device Placement for WiFi Sensing, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 6 (1) (2022) 1–25.

[52] S. Kondo, S. Itahara, K. Yamashita, K. Yamamoto, Y. Koda, T. Nishio, A. Taya, Bi-Directional Beamforming Feedback-Based Firmware-Agnostic WiFi Sensing: An Empirical Study, IEEE Access 10 (2022) 36924–36934.

[53] T. Kanda, T. Sato, H. Awano, S. Kondo, K. Yamamoto, Respiratory Rate Estimation Based on WiFi Frame Capture, in: 2022 IEEE 19th Annual Consumer Communications & Networking Conference, CCNC, 2022, pp. 881–884.

[54] C. Wu, X. Huang, J. Huang, G. Xing, Enabling ubiquitous WiFi sensing with beamforming reports, in: Proceedings of the ACM SIGCOMM 2023 Conference, in: ACM SIGCOMM '23, Association for Computing Machinery, New York, NY, USA, 2023, pp. 20–32, [Online]. Available: https://doi.org/10.1145/3603269.3604817.

[55] Y. Ma, G. Zhou, S. Wang, H. Zhao, W. Jung, SignFi: Sign language recognition using WiFi, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 2 (1) (2018) 1–21.

[56] Y. Fang, B. Sheng, H. Wang, F. Xiao, WiTransfer: A cross-scene transfer activity recognition system using WiFi, in: Proceedings of the ACM Turing Celebration Conference-China, 2020, pp. 59–63.

[57] Y. Jiang, X. Zhu, R. Du, Y. Lv, T.X. Han, D.X. Yang, Y. Zhang, Y. Li, Y. Gong, On the Design of Beamforming Feedback for Wi-Fi Sensing, IEEE Wirel. Commun. Lett. 11 (10) (2022) 2036–2040.

[58] K.F. Haque, F. Meneghello, F. Restuccia, Wi-BFI: Extracting the IEEE 802.11 beamforming feedback information from commercial Wi-Fi devices, in: Proceedings of the 17th ACM Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization, WiNTECH '23, Association for Computing Machinery, New York, NY, USA, 2023, pp. 104–111.

[59] E. Perahia, R. Stacey, Next Generation Wireless LANs: Throughput, Robustness, and Reliability in 802.11n, Cambridge Univ. Press, 2008.

[60] M.S. Gast, 802.11 ac: A Survival Guide: Wi-Fi at Gigabit and Beyond, " O'Reilly Media, Inc.", 2013.

[61] F. Meneghello, F. Restuccia, M. Rossi, WHACK: Adversarial Beamforming in MU-MIMO Through Compressed Feedback Poisoning, IEEE Trans. Wireless Commun. (2024).

[62] A. Nichol, J. Achiam, J. Schulman, On first-order meta-learning algorithms, 2018, arXiv preprint arXiv:1803.02999.

[63] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, Adv. Neural Inf. Process. Syst. 30 (2017).

[64] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al., Matching networks for one shot learning, Adv. Neural Inf. Process. Syst. 29 (2016).

[65] C. Finn, P. Abbeel, S. Levine, Model-agnostic meta-learning for fast adaptation of deep networks, in: International Conference on Machine Learning, PMLR, 2017, pp. 1126–1135.

[66] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012).
[67] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014, arXiv preprint arXiv:1409.1556.
[68] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[69] S. Shi, Y. Xie, M. Li, A.X. Liu, J. Zhao, Synthesizing wider WiFi bandwidth for respiration rate monitoring in dynamic environments, in: IEEE INFOCOM 2019-IEEE Conference on Computer Communications, IEEE, 2019, pp. 181–189.

Khandaker Foysal Haque (Graduate Student Member, IEEE) is a Ph.D. candidate in the Department of Electrical and Computer Engineering and a member of the Institute for the Wireless Internet of Things at Northeastern University, USA. He received his M.S. in Computer Engineering in 2021 from Central Michigan University, USA, and his B.S. in Electrical and Electronic Engineering in 2016 from the Islamic University of Technology (IUT), Bangladesh. His research interest is in the intersection of wireless networking, embedded systems, and machine learning with a focus on integrated sensing & communication for next-generation wireless networks. He was the recipient of the best paper award in IEEE iSES 2020.

Milin Zhang is a Ph.D student in computer engineering in the Department of Electrical and Computer Engineering and a member of the Institute for the Wireless Internet of Things at Northeastern University. He received his M.S. in electrical engineering from Syracuse University, USA, in 2021. He received B.S. from the University of Electronic Science and Technology of China in 2018. His area of study is the integration of deep learning with emerging wireless technologies.

Francesca Meneghello (Member, IEEE) received the Ph.D. degree in Information Engineering in 2022 from the University of Padova and is currently an Assistant Professor at the Department of Information Engineering at the same university. Her research interests include deep-learning architectures and signal processing with application to remote radio frequency sensing and wireless networks. She received an honorary mention in the 2019 IEEE ComSoc Student Competition. She was a recipient of the Best Student Presentation Award at the IEEE Italy Section SSIE 2019, Best Ph.D. Thesis Award from the Italian Group of Telecommunications in 2022, and the Fulbright-Schuman Fellowship in 2023.

Francesco Restuccia (Senior Member, IEEE) is an Assistant Professor in the Department of Electrical and Computer Engineering, and a member of the Institute for the Wireless Internet of Things and the Roux Institute at Northeastern University. He received his Ph.D. in Computer Science from Missouri University of Science and Technology in 2016, and his B.S. and M.S. in Computer Engineering with highest honors from the University of Pisa, Italy in 2009 and 2011, respectively. His research interests lie in the design and experimental evaluation of next-generation edge-assisted data-driven wireless systems. Prof. Restuccia's research is funded by several grants from the US National Science Foundation and the Department of Defense. He received the Office of Naval Research Young Investigator Award, the Air Force Office of Scientific Research Young Investigator Award and the Mario Gerla Award for Young Investigators in Computer Science, as well as best paper awards at IEEE INFOCOM and IEEE WOWMOM. Prof. Restuccia has published over 60 papers in top-tier venues in computer networking, as well as co-authoring 16 U.S. patents and three book chapters. He regularly serves as a TPC member and reviewer for several ACM and IEEE conferences and journals.