CSI-Based Multi-Antenna and Multi-Point Indoor Positioning Using Probability Fusion

Emre Gönültaş, Eric Lei, Jack Langerman, Howard Huang, and Christoph Studer

***Abstract*—Channel state information (CSI)-based fingerprinting via neural networks (NNs) is a promising approach to enable accurate indoor and outdoor positioning of user equipments (UEs), even under challenging propagation conditions. In this paper, we propose a positioning pipeline for wireless LAN MIMO-OFDM systems which uses uplink CSI measurements obtained from one or more unsynchronized access points (APs). For each AP receiver, novel features are first extracted from the CSI that are robust to system impairments arising in real-world transceivers. These features are the inputs to a NN that extracts a probability map indicating the likelihood of a UE being at a given grid point. The NN output is then fused across multiple APs to provide a final position estimate. We provide experimental results with real-world indoor measurements under line-of-sight (LoS) and non-LoS propagation conditions for an 80 MHz bandwidth IEEE 802.11ac system using a two-antenna transmit UE and two AP receivers each with four antennas. Our approach is shown to achieve centimeter-level median distance error, an order of magnitude improvement over a conventional baseline.**

***Index Terms*—Channel-state information (CSI), indoor localization, multi-point fingerprinting, probability fusion.**

I. Introduction

Positioning of mobile user equipment (UE) devices is essential for a broad range of applications, including navigation, virtual reality, asset tracking, advertising, industrial automation, and many more [1]–[5]. While global navigation satellite system (GNSS) technologies enable ubiquitous positioning in outdoor environments with a view of the sky, there is no similarly ubiquitous solution for indoor environments. In addition, some of the indoor positioning applications, such as drone or robot navigation, require centimeter-level precision for executing their missions, which is not achievable using conventional GNSS. One class of solutions for indoor positioning uses infrastructure cameras for visible or infra-red light, viewing an object of interest which is equipped with either an active transmitter or passive reflector [6]–[9]. While such systems can achieve sub-centimeter accuracy, they are expensive, require unobstructed views, and may not work in rooms with bright sunlight.

A. The Challenges of Indoor Positioning using RF Signals

An alternative to camera-based positioning methods is to use radio-frequency (RF) signals [3], [10]–[15]. However, RF-based localization techniques typically incur significant costs in calibrating the infrastructure access points (APs), depending on the required level of accuracy. For example, achieving meter-level accuracy with RF time-difference-of-arrival measurements requires synchronization of the APs with nanosecond accuracy and knowledge of their locations with sub-meter accuracy.

A type of RF localization technique which has the advantage of *not* requiring AP calibration is known as fingerprinting. Instead, these techniques rely on the offline creation of an empirical database that records RF measurements, such as the received signal strength indicator (RSSI) from multiple APs, as a function of the UE location. An algorithm then estimates the location of a UE given its RF measurements and the fingerprinting database. The database could be created with relatively low cost, for example, using an RF receiver mounted on a robot that periodically moves through a space for cleaning.

As an alternative to RSSI measurements, fingerprinting could instead be based on estimates of channel state information (CSI), which is always required for data demodulation. For wideband MIMO-OFDM systems, complex-valued CSI can be estimated for each active subcarrier and each transmit-receive antenna pair, resulting in significantly richer measurement sets compared to RSSI. CSI-based fingerprinting has been shown to enable accurate localization in both indoor [12], [16]–[18] and outdoor [19]–[22] applications. Recently, a range of CSI-based fingerprinting methods that use machine learning—rather than sophisticated geometrical models—has been proposed [16], [17], [20], [21], [23]–[26]. For all of these methods, carefully designed features from CSI turn out to be critical, as real-world channels exhibit small-scale fading and wireless transceivers suffer from a number of hardware impairments. In addition, most of these methods rely on supervised learning to train a neural network (NN) that maps CSI to UE position, which requires dedicated measurement campaigns to acquire CSI and associated ground-truth position. If relative location information is sufficient, self-supervised methods known as channel charting [27]–[29] avoid expensive measurement campaigns. If CSI measurements from multiple APs are available, then the accuracy of channel charting can be improved significantly [30], [31].

E. Gönültaş and E. Lei are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY (e-mail: eg566@cornell.edu; el536@cornell.edu).

J. Langerman and H. Huang are with Nokia Bell-Labs, Murray Hill, NJ (e-mail: jack.langerman@nokia-bell-labs.com; howard.huang@nokia-bell-labs.com).

C. Studer is with the Department of Information Technology and Electrical Engineering at ETH Zürich, Zürich, Switzerland (e-mail: studer@ethz.ch).

The work of EG and CS was supported in part by Xilinx Inc. and by the US NSF under grants CCF-1652065, CNS-1717559, CNS-1955997, and ECCS-1824379. The Quadro P6000 GPU used for this research was donated by the NVIDIA Corporation.

B. Contributions

In this paper, we propose a CSI-based positioning pipeline for multiple-input multiple-output (MIMO) orthogonal frequency-division multiplexing (OFDM) wireless systems, which leverages the availability of CSI acquired at multiple MIMO links. Our positioning system can be built by using off-the-shelf hardware, e.g., using the framework put forward in [32], without knowing the details of the wireless infrastructure. Our main contributions can be summarized as follows:

- We propose a NN-based positioning pipeline that combines so-called *probability maps* from multiple MIMO links (transmit antennas and/or APs). The proposed method does not require accurate synchronization between transmit antennas and/or APs, and minimizes the amount of location information to be transmitted to a centralized processor.
- We evaluate different methods to combine the probability maps, which indicate the likelihood of the UE location on a predefined grid. The proposed methods include probability conflation, Gaussian conflation, and NN-based probability fusion, which improve positioning accuracy over existing NN-based CSI fingerprinting solutions.
- We calculate the probability maps from CSI features using NNs. The proposed method enables probability maps with irregular grid structures. Furthermore, NN training is accomplished by assigning probabilities to multiple grid points using a minimum-variance approach.
- We extract CSI features that are robust to hardware impairments typically arising in IEEE 802.11ac MIMO-OFDM-based systems operating indoors. The proposed CSI features outperform existing features that were designed for massive multiuser MIMO basestations.
- We provide experimental results with real-world indoor channel measurements under line-of-sight (LoS) and non-LoS conditions and for multi-antenna and multi-AP measurements. Our results reveal that centimeter-level accuracy is achievable using our positioning pipeline.

C. Relevant Prior Art

CSI-based positioning has been studied extensively in the past; see, e.g., [12], [15]–[18], [21], [24], [25], [33]–[36] and the references therein. Recent work has focused mainly on neural network (NN)-based approaches [16], [17], [21], [24], [25], [37]–[41], which (i) do not require geometric modeling [24], (ii) avoid storage of potentially very large CSI fingerprint databases [42], and (iii) have the potential to generalize to areas excluded in the training set [43]. In contrast to such NN-based approaches, our focus is on multi-antenna and multi-point positioning in which a NN generates what we call *probability maps*, a probabilistic description of UE location measured at multiple APs and from multiple transmit antennas. These probability maps can then be fused at a centralized processor to improve positioning accuracy.

The references [25] and [24] propose NNs that estimate the likelihood of a UE being in a certain grid point using one-hot encoded vectors trained from a fixed number of uniformly-spaced reference positions. NN learning is then formulated as a classification problem in which the labels correspond to grid points nearest to the UE locations. Instead of one-hot vectors, we propose an improved strategy that enables NN training for general grid structures and for general probability mass functions (PMFs). To compute the PMFs, we propose an optimization-based minimum-variance approach that assigns probabilities to the individual grid points so that the expected location exactly matches the UE position. While the references [44]–[46] also rely on a probabilistic description of UE position, these methods perform positioning using geometrical models instead of NNs.

NN-based positioning from CSI measurements requires carefully-designed features, which are robust to small-scale fading as well as system and hardware impairments. The use of beamspace representations to extract the incident angles has been used in [21], [27], [28], [47]. The conversion of subcarrier CSI into the delay domain to extract relative time-of-flight information has been used in [21], [47]–[49]. The use of cross-correlation in the spatial domain and 1-dimensional autocorrelation in the delay domain has been used in [27], [28] and [20], [29], [50], respectively. Such methods improve resilience to small-scale fading and common hardware impairments, including time synchronization errors as well as residual carrier frequency and sampling rate offsets. To improve robustness of our positioning pipeline to small-scale fading as well as system and hardware impairments that typically arise in IEEE 802.11ac MIMO-OFDM systems, we improve the CSI feature extraction pipeline in [20] specialized for massive MU-MIMO cellular systems by (i) taking the “instantaneous autocorrelation” over the antenna and delay domains (no averaging over multiple measurements), (ii) computing the autocorrelation in the antenna domain (not just the delay domain), (iii) processing both the real and imaginary parts (instead of taking the magnitudes), and (iv) not taking the logarithm on the resulting features.

Virtually all existing NN-based positioning methods rely on single-transmit antenna CSI measurements at a single access point (AP) or basestation (BS) possibly with multiple receive antennas. In contrast to these methods, we generate probability maps for each individual transmit antenna, which can then be fused to improve localization accuracy while reducing the complexity of the neural network and storage requirements. Multi-point localization strategies, where CSI from multiple APs or cellular BSs is combined, have been proposed in [30], [31] for channel charting. While such approaches only enable relative localization, they also require the exchange of CSI features to a centralized processor that performs channel charting. In contrast to these approaches, we propose new methods that fuse probability maps generated at multiple APs (and from multiple transmit antennas), which not only improves (absolute) positioning accuracy but also reduces the amount of information that must be transferred to a centralized processor.

Probability fusion of multiple sensor data such as camera images [51] is a widely studied subject; see, e.g., [52]–[54]. Existing outdoor positioning systems, such as GNSS, triangulation-based methods in cellular systems [55], or indoor positioning systems such as WorldViz [56] or VICON [57], already fuse at least three different data sources to produce a robust position estimate. The methods proposed in this paper combine probabilistic sensor fusion with CSI-based positioning. Concretely, our methods fuse multiple probability maps that indicate the likelihood of a UE being at a given grid point using theoretically principled conflation methods put forward in [58]. To the best of our knowledge, no work has described the use of such conflation methods in a wireless positioning application.

D. Notation

Lowercase boldface letters, uppercase boldface letters, and uppercase calligraphic letters denote column vectors, matrices, and sets, respectively. For a matrix $\mathbf{A}$, we denote its transpose by $\mathbf{A}^T$, its Hermitian transpose by $\mathbf{A}^H$, its ith row and jth column by $A_{i,j}$, and its ith column by $\boldsymbol{a}_i$. For a vector $\mathbf{a}$, the kth entry is denoted by a_k, the ℓ^2 and ℓ^1 norms are $\|\mathbf{a}\|_2 = \sqrt{\sum_k |a_k|^2}$ and $\|\mathbf{a}\|_1 = \sum_k |a_k|$, respectively, and the real and imaginary parts are $\Re(\mathbf{a})$ and $\Im(\mathbf{a})$, respectively.

E. Outline

The remainder of this paper is organized as follows. Section II describes the operation principle of the proposed positioning pipeline and introduces the MIMO-OFDM channel model. Section III details NN-based positioning along with CSI-feature construction. Section IV proposes different probability fusion methods for multi-antenna and multi-point data. Section V shows results for real-world indoor channel measurements. Section VI concludes the paper.

II. Operation Principle and System Model

Our objective is to estimate the position of a UE from CSI measurements acquired at one or multiple APs. We first outline the operation principle of the positioning pipeline and then describe the multi-point MIMO-OFDM system model.

A. Operation Principle

Figure 1 illustrates the basic concept of the proposed multi-antenna multi-point CSI-based positioning pipeline. The uth UE at location $\mathbf{x}^{(u)}$ transmits pilot sequences from one or multiple transmit antennas, which are then used to estimate the wireless channel at one or multiple APs indexed by $b = 1, \ldots, B$ where B is the number of APs.

The APs can have one or many receive antennas. The acquired CSI is then used to generate a probability map $\mathbf{p}^{(u)}[b]$ which contains information on the location of the uth UE as seen from the bth AP. The probability map is generated from measured CSI using neural networks, which have been trained from a dataset containing CSI-location pairs in a dedicated training phase. Details on the positioning pipeline are provided in Section III. Ground-truth location information for neural network training is acquired via a state-of-the-art multi-camera positioning system. Details on the measurement setup are provided in Section V-A. The probability information from all APs is then fused in order to produce an estimate $\hat{\mathbf{x}}^{(u)}$ of the UE location.

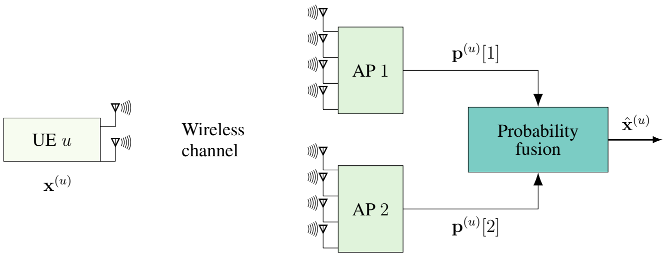


Fig. 1. Overview of the proposed CSI-based multi-antenna and multi-AP positioning system. A user equipment (UE) u at position $\mathbf{x}^{(u)}$ transmits pilots to one or multiple APs, which extract channel state information (CSI). Each AP then uses CSI to generate a probability map indicating the UE's location, which is fused to compute an estimate $\hat{\mathbf{x}}^{(u)}$ of the UE's position.

Remark 1. *A key limitation of supervised positioning methods is the requirement of a dedicated training phase used to acquire a large CSI dataset annotated with ground-truth position. If the cost of creating a CSI dataset is manageable, then it is highly beneficial to use supervised CSI-based fingerprinting methods. Self- or semi-supervised methods that build on channel charting [19], [27], [59], can be used to reduce or completely avoid the acquisition of ground-truth location information at the expense of positioning accuracy. An extension of the proposed positioning pipeline to such self- or semi-supervised methods is part of ongoing work.*

Remark 2. *In order to acquire CSI at multiple APs, CSI sniffing methods that receive all transmitted OFDM packets in the receiver's range can be used [32]. Furthermore, the APs performing localization do not need to be the destination of the payload as CSI can be extracted from any IEEE 802.11ac [60]-compatible receiver.*

B. Multi-Antenna Multi-Point OFDM System Model

We focus on an IEEE 802.11ac-based wireless communication system [60], which is also what we used in Section V to generate our indoor positioning results.[1] We consider a multi-antenna UE with M_T transmit antennas that is located in the range of B multi-antenna APs (or basestations) having M_R antennas each. We consider an OFDM system with W subcarriers and cyclic prefix length C. The set Ω_used contains the indices of subcarriers associated with tones that have been trained (corresponding to data and pilot subcarriers); the set Ω_zero contains the indices associated with unused subcarriers. Consequently, we have $\Omega_\text{used} \cup \Omega_\text{zero} = \{1, \ldots, W\}$. We assume that the uth UE is at position $\mathbf{x}^{(u)} \in \mathbb{R}^D$, where D is typically two or three (representing the spatial location), and is transmitting one pilot per used subcarrier (e.g., during the preamble), which is performed at each AP to compute an estimate of the wireless channel in the frequency domain. The estimated $M_\text{R} \times M_\text{T}$ MIMO channel matrix at subcarrier $w = 1, \ldots, W$ and at AP $b = 1, \ldots, B$ is denoted by

[1] Many of the proposed techniques can be adapted easily to other MIMO-OFDM-based wireless communication systems.

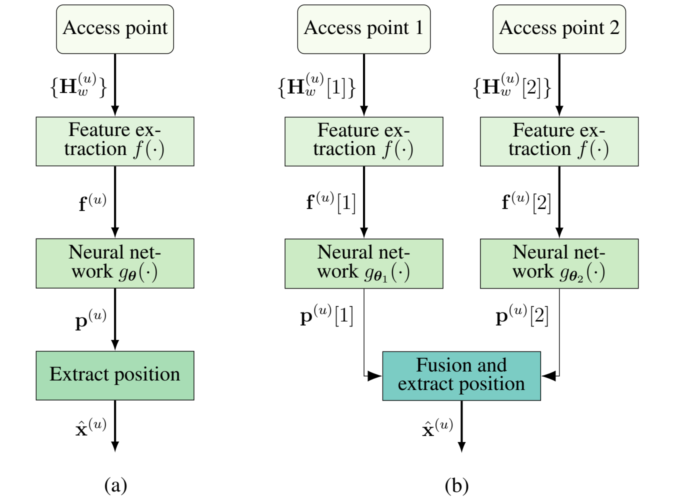

Access point
$\{\mathbf{H}_w^{(u)}\}$
Feature extraction $f(\cdot)$
$\mathbf{f}^{(u)}$
Neural network $g_{\boldsymbol{\theta}}(\cdot)$
$\mathbf{p}^{(u)}$
Extract position
$\hat{\mathbf{x}}^{(u)}$
(a)
Access point 1
Access point 2
$\{\mathbf{H}_w^{(u)}[1]\}$
$\{\mathbf{H}_w^{(u)}[2]\}$
Feature extraction $f(\cdot)$
Feature extraction $f(\cdot)$
$\mathbf{f}^{(u)}[1]$
$\mathbf{f}^{(u)}[2]$
Neural network $g_{\boldsymbol{\theta}_1}(\cdot)$
Neural network $g_{\boldsymbol{\theta}_2}(\cdot)$
$\mathbf{p}^{(u)}[1]$
$\mathbf{p}^{(u)}[2]$
Fusion and extract position
$\hat{\mathbf{x}}^{(u)}$
(b)

Fig. 2. CSI-based positioning pipeline. (a) Positioning with one AP: The AP extracts CSI $\{\mathbf{H}_w^{(u)}\}$ of the uth UE followed by a feature extraction stage that produces $\mathbf{f}^{(u)}$. A neural network then computes a probability map $\mathbf{p}^{(u)}$ that is used to generate an estimate $\hat{\mathbf{x}}^{(u)}$ of the UEs location. (b) Positioning with two APs: A centralized node fuses the two extracted probability maps $\mathbf{p}^{(u)}[1]$ and $\mathbf{p}^{(u)}[2]$ computed from two different neural networks from AP1 and AP2, respectively, to generate an estimate $\hat{\mathbf{x}}^{(u)}$ of the UE location.

$\mathbf{H}_w^{(u)}[b] \in \mathbb{C}^{M_\text{R}\times M_\text{T}}$. We call the collection of these channel matrices

$$\mathbf{H}_w^{(u)}[b], \quad w = 1,\ldots,W, \quad b = 1,\ldots,B, \qquad (1)$$

the CSI associated with UE u; we abbreviate (1) by $\{\mathbf{H}_w^{(u)}[b]\}$. In what follows, we assume that the delay spread of the channel plus the maximum timing offset does not exceed the cyclic prefix length. We furthermore assume $C \leq |\Omega_\text{used}|$ for reasons discussed in Section III-A1. Besides that, we allow the extracted channel estimates to be affected by real-world system and hardware impairments, such as timing errors and residual phase offsets.

Remark 3. *We do not require (i) the B APs to acquire the CSI from the uth UE at the same time instant or (ii) perfect synchronization among APs. The only assumption is that the CSI measured at each APs is for the same UE, which is transmitting from approximately the same location $\mathbf{x}^{(u)}$; this includes the cases of (i) unsynchronized APs or (ii) situations in which the UE is transmitting to multiple APs in a round-robin fashion or scenarios in which one or multiple APs are acquiring CSI without decoding the UEs data.*

III. CSI-Based Fingerprinting via Neural Networks

We now detail the proposed multi-antenna multi-point CSI-based positioning pipeline illustrated in Figure 2. We start by describing the CSI-feature extraction stage followed by discussing the NN that generates probability maps. Means to fuse probability maps to obtain accurate location estimates from multi-antenna and multi-point data are detailed in Section IV.

A. CSI-Feature Extraction

In order to enable CSI-based positioning, it is critical to construct *robust* CSI-features that (i) are unique for a given UE location, (ii) are robust to small-scale fading effects [19], [22], [27], [28], [59], and (iii) are resilient to real-world system and hardware impairments [20], [29]. Clearly, for a given UE location $\mathbf{x}^{(u)}$, the CSI features extracted from the estimated CSI $\{\mathbf{H}_w^{(u)}[b]\}$ should be unique—otherwise, UE location is not uniquely determined. Furthermore, small-scale fading, e.g., caused by moving objects in the surrounding area of the UE and APs, should not affect the CSI features as they are generally difficult to model. Finally, system and hardware impairments, such as varying transmit or receive power as well as timing, carrier frequency, and sampling rate offsets, should not affect the CSI features as they may vary over time and from UE to UE. We emphasize that resilience to system and hardware impairments is crucial in order to achieve accurate positioning from CSI features. We now propose such CSI features (see Figure 2), which address the desired properties (ii) and (iii), as CSI-feature uniqueness in property (i) is mainly determined by the physical channel and difficult to control in practice.

1) Delay-Domain Transform: We start by transforming the frequency (or subcarrier) domain into the delay domain, as OFDM-based systems typically have only a limited delay spread (e.g., no larger than the cyclic prefix length) and only a few taps in the impulse responses are significant. Let us first define the vector $\mathbf{h}_{m,w}^{(u)}[n] = \big[\mathbf{H}_w^{(u)}[b]\big]_m$ as the mth column of $\mathbf{H}_w^{(u)}[b]$, which is the CSI vector corresponding to the mth transmit antenna of the uth UE at subcarrier w. Next, we define the following M-dimensional frequency-domain vector

$$\hat{\mathbf{h}}_{n,m}^{(u)}[b] = \big[\,[\mathbf{h}_{m,1}^{(u)}[b]]_n, [\mathbf{h}_{m,2}^{(u)}[b]]_n, \ldots, [\mathbf{h}_{m,W}^{(u)}[b]]_n\,\big]^T \qquad (2)$$

for a given receive antenna n of AP b and a given UE u transmit antenna m. In words, the vector $\hat{\mathbf{h}}_{n,m}[b]$ contains the channel estimates over all W subcarriers. Ideally, by taking the inverse DFT of the frequency-domain vector $\hat{\mathbf{h}}_{n,m}^{(u)}[b]$, one would obtain the delay-domain description of the frequency-selective channel from the mth UE antenna to the nth receive antenna of AP b. Unfortunately, only the subcarriers indexed by Ω_user are available in practice, whereas the entries pertaining to the subcarriers indexed by Ω_zero are generally unknown (as they were not trained). Fortunately, since we assumed that the cyclic prefix length C is not larger than the number of used subcarriers $|\Omega_\text{used}|$, one can estimate the delay-domain coefficients within the cyclic prefix length. Let Γ_cp be the set of indices associated to the channel taps in the delay-domain so that $C = |\Gamma_\text{cp}|$. Then, the delay-domain coefficients can be estimated by computing

$$\mathbf{t}_{n,m}^{(u)}[b] = (\mathbf{D}_{\Omega_\text{used},\Gamma_\text{cp}})^\dagger \big[\hat{\mathbf{h}}_{n,m}^{(u)}[b]\big]_{\Omega_\text{used}}. \qquad (3)$$

Here, the delay-domain vector $\mathbf{t}_{n,m}^{(u)}[b] \in \mathbb{C}^{|\Gamma_\text{cp}|}$ contains the C dominant taps of the wireless channel between the mth transmit antenna of UE u and the nth receive antenna at AP b, the matrix $\mathbf{D}_{\Omega_\text{used},\Gamma_\text{cp}}$ contains the rows indexed by Ω_used and the columns indexed by Γ_cp of the DFT matrix, $(\cdot)^\dagger$ denotes the left-pseudoinverse, and $[\hat{\mathbf{h}}_{n,m}^{(u)}[b]]_{\Omega_\text{used}}$ is the subset of the frequency-domain vector in (2) corresponding to the used subcarriers indexed by Ω_used. Note that the least-squares estimator in (3) is frequently used to denoise channel vectors in OFDM systems [61].

Remark 4. *We have observed that taking the inverse DFT directly over the frequency-domain vector (2) still works well in practice and requires lower complexity than the approach in (3) as one can simply use an inverse fast Fourier transform (IFFT).*

2) Autocorrelation: In addition to transforming the estimated CSI into the delay domain, we now propose a method that renders the CSI features robust to time-synchronization errors, residual carrier frequency offset (CFO) and sampling rate offset (SRO), and global phase modulations. The method proposed here is inspired by the approach proposed recently in [20] for CSI-based positioning in cellular massive MIMO systems.

Time synchronization errors and residual phase errors can be modeled in the discrete-time domain as $t[k] = y[k-\delta]e^{j\varphi}$, where $t[k]$ is the time-domain signal at sample index k, $y[k-\delta]$ is the true received signal with unknown delay δ caused by synchronization (or frame-start detection) errors, and φ determines the amount of residual phase errors.[2] When computing the "instantaneous[3] autocorrelation" of the signal $t[k]$, we have

$$R_t[\tau] = \sum_{k} t[k]t^*[k+\tau-1] = \sum_{k'} y[k']y^*[k'+\tau-1], \quad (4)$$

for $\tau = 1, 2, \ldots$, which does no longer depend on the time synchronization error δ and the residual phase errors.

To improve robustness of our CSI features, we will follow this approach and compute the instantaneous autocorrelation not only in the delay domain but also in the antenna domain. The latter renders the proposed CSI-features robust to global phase offsets across the antenna array. Let $\left[\mathbf{t}^{(u)}_{n,m}[b]\right]_k$ be the kth delay-domain sample measured at the nth receive antenna of AP b transmitted from the mth antenna from UE u. Then, we compute

$$R_t^{(u)}[m,\tau,\kappa,b] = \sum_{n=1}^{M_\text{R}}\sum_{k=1}^{C} \left[\mathbf{t}^{(u)}_{n,m}[b]\right]_k \left[\mathbf{t}^{(u)}_{n+\kappa-1,m}[b]\right]^*_{k+\tau-1}, \quad (5)$$

where $\tau = 1, 2, \ldots, 2C$ and $\kappa = 1, 2, \ldots, 2M_\text{R}$. For a given AP b and UE u, we vectorize the three-dimensional tensor in (5) so that the vector $\mathbf{r}^{(u)}[b] \in \mathbb{C}^{M_\text{R}M_\text{T}2C}$ contains all the entries of $R_t^{(u)}[m,\tau,\kappa,b]$ for $m = 1, \ldots, M_\text{T}$, $\tau = 1, \ldots, 2C$, and $\kappa = 1, \ldots, 2M_\text{R}$. Finally, to enable the use of off-the-shelf deep-learning software, we convert the complex valued vector $\mathbf{r}^{(u)}[b] \in \mathbb{C}^{M_\text{T}2C2M_\text{R}}$ into a $2M_\text{T}2C2M_\text{R}$-dimensional real-valued CSI-feature vector as follows:

$$\hat{\mathbf{f}}^{(u)}[b] = \left[\Re\{\mathbf{r}^{(u)}[b]\}^T, \Im\{\mathbf{r}^{(u)}[b]\}^T\right]^T. \quad (6)$$

3) CSI Feature Normalization: In practice, the gains of the power amplifier at the UE side and the low-noise amplifier at the AP side can be set independently. While these gain settings are typically kept constant during transmission of one OFDM frame, the AP does, in general, not know the UE's gain settings. Furthermore, the path-loss characteristics in indoor applications depend on the environment. Hence, the receive power is not a reliable indicator for the distance between the UE and the AP, and should be ignored. For these reasons, we normalize the vector in (6) as $\mathbf{f}^{(u)}[b] = \hat{\mathbf{f}}^{(u)}[b]/\|\hat{\mathbf{f}}^{(u)}[b]\|_2$, which ensures that the transmit and receive gains as well as the path loss are ignored, and also improves the convergence of stochastic gradient descent.

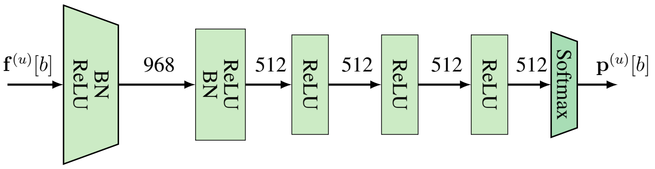


Fig. 3. Neural network (NN) structure at AP b. The NN $g_{\boldsymbol{\theta}_b}$ with weights and biases contained in the vector $\boldsymbol{\theta}_b$ takes in a CSI-feature vector $\mathbf{f}^{(u)}[b]$ and generates a probability map $\mathbf{p}^{(u)}[b]$ that describes the position of UE u.

In what follows, we will also use CSI features extracted separately per AP b and per transmit antenna $m = 1, \ldots, M_\text{T}$, which are obtained by vectorizing $R_t^{(u)}[m,\tau,\kappa,b]$ in only $\tau = 1, \ldots, 2C$ and $\kappa = 1, \ldots, 2M_\text{R}$ and by stacking the real and imaginary parts in the vector $\mathbf{f}^{(u)}_m[b]$.

B. Neural Network Structure

As illustrated in Figure 2, we propose to use one or multiple neural networks (NN) $g_{\boldsymbol{\theta}_b}$ at each AP b with weights and biases from all layers contained in the vector $\boldsymbol{\theta}_b$ that takes in CSI feature vectors $\mathbf{f}^{(u)}[b]$ and generates what we call a *probability map*

$$\mathbf{p}^{(u)}[b] = g_{\boldsymbol{\theta}_b}(\mathbf{f}^{(u)}[b]) \quad (7)$$

that describes the location of UE u; see Section III-C for details on probability maps. This probability map is then fused with probability maps from other APs to extract an estimated position $\hat{\mathbf{x}}^{(u)}$ of UE u. The neural network structure is illustrated in Figure 3. We consider a six-layer neural network with input CSI-feature vector $\mathbf{f}^{(u)}[b]$ and output probability-map vector $\mathbf{p}^{(u)}[b]$. All but the last layers use ReLu activations; the last layer uses a softmax activation for the reasons detailed below. The first and second layer use batch normalization (BN). The number of activations for each layer is shown in Figure 3; the dimensions of the input CSI-feature vector $\mathbf{f}^{(u)}[b]$ and the output probability-map vector $\mathbf{p}^{(u)}[b]$ depend on the feature type and the resolution of the probability map, respectively. Prior to training, We initialize the weights and biases using Glorot [64] with a uniform distribution; during training, we use the binary cross entropy (BCE) loss.

C. Probability Maps

Instead of using NNs that directly produce an estimate $\hat{\mathbf{x}}^{(u)}$ for the true location $\mathbf{x}^{(u)}$ of UE u, which is the de-facto standard approach [16], [17], [20], [21], [23], [24], we propose to use a probabilistic description of UE location. This strategy has the advantage that the NN output contains valuable information for during downstream processing, e.g., (i) to extract reliability information on the estimated UE location and (ii) to

[2] Practical receivers estimate and compensate CFO and SRO, but residual phase errors remain in practice [62], [63].

[3] We are not taking any expectation over the product $t[k]t^*[k+\tau-1]$, which is in contrast to the method proposed in [20].

fuse multiple probability maps to improve positioning accuracy in multi-antenna and/or multi-AP scenarios. In addition, a probabilistic location description can also resolve "conflicts," which may arise if two spatial locations generate similar CSI. Finally, probability maps can also improve NN training for CSI-based positioning.

1) Basics of Probability Maps: We overlay a set of K grid points $\mathbf{g}_k \in \mathbb{R}^D$, $k = 1, \ldots, K$, over the space that will be used to perform positioning. Here, D is typically two or three and represents the number of spatial dimensions used to perform positioning, and the convex hull over all grid points

$$\mathcal{H} = \left\{ \sum_{k=1}^{K} \alpha_k \mathbf{g}_k \mid (\alpha_k \in \mathbb{R}_+, \forall k) \wedge \sum_{k=1}^{K} \alpha_k = 1 \right\} \quad (8)$$

must include the target area in which localization will be performed. The probability map $\mathbf{p}^{(u)}[b] \in [0,1]^K$ represents the probability of UE u being located at each grid point, i.e., we have that $p_k^{(u)}[b] \in [0,1]$ for $k = 1, \ldots, K$ and $\sum_{k=1}^{K} p_k^{(u)}[b] = 1$. Note that the last softmax layer in the proposed NN structure shown in Figure 3 ensures that the generated outputs correspond to probability mass functions (PMFs). Furthermore, if the probabilities contained in $\mathbf{p}^{(u)}[b]$ indeed model the UE's position, we can compute the expected location of UE u as follows:

$$\hat{\mathbf{x}}^{(u)}[b] = \sum_{k=1}^{K} \mathbf{g}_k p_k^{(u)}[b]. \quad (9)$$

By defining the $D \times K$ grid point matrix $\mathbf{G} = [\mathbf{g}_1, \ldots, \mathbf{g}_K]$ the expected location is simply $\hat{\mathbf{x}}^{(u)}[b] = \mathbf{G}\mathbf{p}^{(u)}[b]$. Hence, one could easily augment the NN shown in Figure 3 to directly generate an estimate of the UE location $\hat{\mathbf{x}}^{(u)}[b]$ by adding an additional linear output layer with untrainable weights corresponding to $\mathbf{G}$ and no bias terms.

Remark 5. *The selection of grid points can either form an equispaced rectangular grid for $D = 2$ (or equispaced cubic grid for $D = 3$) that includes the target area or can be chosen arbitrarily. An example of a rectangular grid is shown in Figure 4. Irregular grid points may be useful to only cover locations that are populated or to place grid points at higher density in areas where higher resolution is required.*

2) Training NNs with Probability Maps: In order to ensure that the probability maps $\mathbf{p}^{(u)}[b]$ generated by the proposed NN accurately model the location of UE u being at grid point $\mathbf{g}_k$ with probability $p_k^{(u)}[b]$, the network must be trained accordingly. Assume that we have a training set with CSI feature vectors $\{\mathbf{f}^{(u)}[b]\}_{u=1}^{U'}$ obtained from U' distinct locations $\{\mathbf{x}^{(u)}\}_{u=1}^{U'}$. In order to train the NN shown in Figure 3, we need to compute reference probability maps $\{\mathbf{p}^{(u)}[b]\}_{u=1}^{U'}$ associated with ground-truth positions $\{\mathbf{x}^{(u)}\}_{u=1}^{U'}$. Unfortunately, given a position $\mathbf{x}^{(u)}$ there are, in general, infinitely many probability maps $\mathbf{p}^{(u)}[b]$ for which $\mathbf{x}^{(u)}[b] = \mathbf{G}\mathbf{p}^{(u)}[b]$ holds and $\mathbf{p}^{(u)}[b]$ is a PMF. We propose to select the probability map for which the error variance is minimized—this only activates probabilities associated with grid points that are nearby the ground truth location.

To compute such minimum-variance probability maps, we reiterate that (9) is nothing but the expected location of UE u. The $D \times D$ covariance matrix is then given by

$$\mathbf{C}^u[b] = \sum_{k=1}^{K} p_k^{(u)}[b](\mathbf{g}_k - \hat{\mathbf{x}}^{(u)}[b])(\mathbf{g}_k - \hat{\mathbf{x}}^{(u)}[b])^T. \quad (10)$$

and the combined variance is $\sigma^2[b] = \mathrm{tr}(\mathbf{C}^{(u)}[b])$. By defining the vector $\mathbf{v}^{(u)} \in \mathbb{R}^K$ with entries $v_k^{(u)}[b] = \|\mathbf{g}_k - \hat{\mathbf{x}}^{(u)}[b]\|^2$, $k = 1, \ldots, K$, we have that $\sigma^2[b] = \langle \mathbf{v}^{(u)}[b], \mathbf{p}^{(u)}[b] \rangle$. Hence, we can solve the following convex optimization problem to learn a minimum-variance probability map $\mathbf{p}^{(u)}[b]$ from the ground-truth location $\mathbf{x}^{(u)}$:

$$\begin{cases} \underset{\mathbf{p} \in \mathbb{R}^K}{\text{minimize}} & \mathbf{p}^T \mathbf{v}^{(u)}[b] \\ \text{subject to} & \|\mathbf{G}\mathbf{p} - \mathbf{x}^{(u)}\| \leq \varepsilon, \\ & \sum_{k=1}^{K} p_k = 1, \ p_k \in [0,1], \forall k. \end{cases} \quad (11)$$

Here, $\varepsilon > 0$ can be used to trade-off accuracy vs. variance. Problems of this form can easily be solved using off-the-shelf convex solvers, such as CVX [65], or customized methods that build on Douglas-Rachford splitting [66]. Note that if a ground-truth position is outside the convex hull $\mathcal{H}$ spanned by the grid points as defined in (8), then the optimization problem may no longer be feasible[4]; for positions within the convex hull, the problem in (11) is always feasible. For grid points on an equispaced 2-dimensional ($D = 2$) grid, the problem in (11) reduces to identifying the nearest four grid points (two in x-direction; two in y-direction) to the target location $\mathbf{x}^{(u)}$ and assign the probabilities to these four grid points while minimizing the variance.

After learning the minimum-variance probability maps $\{\mathbf{x}^{(u)}\}_{u=1}^{U'}$ for each ground-truth position $\mathbf{x}^{(u)}$, we can learn the NN parameters $\boldsymbol{\theta}_b$. To this end, we use extracted CSI-features $\{\mathbf{f}^{(u)}[b]\}_{u=1}^{U'}$ and the probability maps $\{\mathbf{p}^{(u)}\}_{u=1}^{U'}$ associated with ground-truth position $\{\mathbf{x}^{(u)}\}_{u=1}^{U'}$, and we train the NN using a conventional binary cross-entropy loss function.

Remark 6. *We have observed that using a cross-entropy loss instead of training the same network augmented with an additional grid point layer $\mathbf{G}$ with a mean-square error loss between estimated and ground truth position resulted in superior positioning accuracy.*

IV. Probability Fusion

As illustrated in Figure 2(b), we are interested in fusing multiple probability maps obtained from different APs and/or transmit antennas to improve positioning accuracy. To explain the advantages of this approach, consider a situation in which two distinct locations yield similar CSI features at one AP, which would result in similar probability maps. While positioning from such probability maps would result in a large positioning error (as the two probability maps should describe two different locations), probability fusion can resolve such ambiguities. As long as the CSI acquired from another AP maps these two locations to *distinct* probability maps, only the probabilities associated with locations where both maps agree would be strong after fusion—all other probabilities would

[4] Uniqueness depends on the choice of the trade-off parameter ε.

be attenuated. We now propose three different methods that implement probability fusion: Probability conflation, Gaussian conflation, and NN-based probability fusion. A comparison of these three fusion approaches is given in Section V.

A. Probability Conflation

Assume that we have multiple neural networks that generate different probability maps for a given UE u, e.g., obtained from different APs $\mathbf{p}^{(u)}[b]$, $b = 1, \ldots, B$, or from different transmit antennas $\mathbf{p}_m^{(u)}[b]$, $m = 1, \ldots, M_\mathrm{T}$, or a combination of both. To simplify notation, assume that B' neural networks generate a collection of B' probability maps denoted by $\{\mathbf{p}^{(u)}[b]\}_{b=1}^{B'}$, irrespective of whether these are obtained from different APs or transmit antennas. Our goal is to fuse this collection of probability maps to a single probability map $\bar{\mathbf{p}}^{(u)}$, which can then be used to produce an improved estimate of the uth UE's position by computing the expected position $\hat{\mathbf{x}}^{(u)} = \mathbf{G}\bar{\mathbf{p}}^{(u)}$.

The idea of combining multiple PMFs into a single PMF that more accurately describes the observed quantity has been widely studied in the literature; see, e.g., [58]. Intuitively, combining PMFs for UE positioning should automatically give more weight to probability maps with smaller variance. Ideally, the fused probability map should provide a more accurate estimate of the UE's position than solely using the most reliable probability map. To achieve this goal, we propose to use *probability conflation* as put forward in [58, Def. 2.7]. The approach is straightforward—simply compute the (unnormalized) sub-PMF via a point-wise Hadamard product as

$$\mu_k^{(u)} = \prod_{b=1}^{B'} p_k^{(u)}[b], \quad k = 1, \ldots, K, \tag{12}$$

followed by normalizing the fused sub-PMF vector $\boldsymbol{\mu}^{(u)}$ to the fused PMF according to

$$\bar{\mathbf{p}}^{(u)} = \frac{\boldsymbol{\mu}^{(u)}}{\|\boldsymbol{\mu}^{(u)}\|_1}. \tag{13}$$

As demonstrated in [58, Sec. 4], probability conflation cannot improve the amount of information contained in all probability maps $\{\mathbf{p}^{(u)}[b]\}_{b=1}^{B'}$, but the resulting conflated PMF can be shown to be optimal (among other properties) in terms of minimizing the loss of Shannon information.[5]

Remark 7. *Probability conflation requires one to pass all probability maps $\{\mathbf{p}^{(u)}[b]\}_{b=1}^{B'}$ with a total number of $K \times B'$ real numbers to a centralized processor, which calculates (12) and (13). The complexity of probability conflation is $K(B'-1)+1+K \times D$ real-valued multiplications.*

B. Gaussian Conflation

While probability conflation requires the transfer of $K \times B'$ real numbers, we can use an alternative conflation approach that reduces the amount of information transfer. This probability fusion approach is inspired by the method used to train probability maps in Section III-C2, where we compute the mean and variance of a probability map. Let us assume that the mean UE position $\hat{\mathbf{x}}^{(u)}$ can be modeled as follows:

$$\hat{\mathbf{x}}^{(u)}[b] = \mathbf{x}^{(u)} + \mathbf{e}^{(u)}[b]. \tag{14}$$

Here, $\mathbf{x}^{(u)}[b]$ is the true position and the error vector $\mathbf{e}^{(u)}[b] \in \mathbb{R}^D$ is assumed to be zero-mean. As shown in (9), given a probability map $\mathbf{p}^{(u)}[b]$, the mean position can be computed as in (9). By furthermore assuming that the entries in the error vector $\mathbf{e}^{(u)}[b]$ are pairwise uncorrelated (meaning that the positioning errors in each spatial dimension are uncorrelated), its covariance matrix $\mathbf{K}^{(u)}[b] = \mathrm{diag}(\mathbf{C}^{(u)}[b])$ corresponds to the main diagonal of the covariance matrix $\mathbf{C}^{(u)}[b]$ defined in (10). With the model in (14), we have B' "noisy" observations of the true location $\mathbf{x}^{(u)}$. By assuming that the entries in the error vector $\mathbf{e}^{(u)}[b]$ are Gaussian and that the error vectors are pairwise independent across observations $b = 1, \ldots, B'$, we can now perform *Gaussian conflation* as analyzed in [58, Thm. 6.1]. The optimal combination of mean positions $\hat{\mathbf{x}}^{(u)}[b]$, $b = 1, \ldots, B'$, in terms of minimizing the post fusion error covariance (or mean-square error) is given by

$$\hat{x}_d^{(u)} = \frac{\sum_{b=1}^{B'} \left[\mathbf{K}^{(u)}[b]\right]_{d,d}^{-1} \hat{x}_d^{(u)}[b]}{\sum_{b=1}^{B'} \left[\mathbf{K}^{(u)}[b]\right]_{d,d}^{-1}}, \quad d = 1, \ldots, D. \tag{15}$$

Intuitively, Gaussian fusion de-weights position estimates with higher variance. The diagonal entries of the error covariance matrix $\mathbf{K}^{(u)}$ of the fused estimate $\hat{\mathbf{x}}_d^{(u)}$ from (15) are given by

$$\left[\mathbf{K}^{(u)}\right]_{d,d} = \sum_{b=1}^{B'} \left[\mathbf{K}^{(u)}[b]\right]_{d,d}^{-1}, \quad d = 1, \ldots, D. \tag{16}$$

Remark 8. *Gaussian conflation requires only B' mean-variance pairs for each dimension D, which requires a transfer of $2 \times D \times B'$ real numbers to a centralized processor, which computes an improved location estimate as in (15). The complexity of Gaussian conflation is $D \times B' \times K + D \times (2B'+1)$ real-valued multiplications.*

C. Neural-Network-Based Probability Fusion

Besides the two conflation methods discussed above, there exist other probability fusion methods. A straightforward approach is to compute simple unweighted average as

$$\hat{\mathbf{x}}^{(u)} = \frac{1}{B'} \sum_{b=1}^{B'} \hat{\mathbf{x}}^{(u)}[b] = \frac{1}{B'} \sum_{b=1}^{B'} \mathbf{G}\mathbf{p}^{(u)}[b], \tag{17}$$

which is a special case of Gaussian conflation in (15) assuming the error variances are equal. While this averaging approach can serve as a baseline probability fusion method, it can be improved by a neural network that uses optimized linear combination weights.

We first train the B' neural networks $g_{\boldsymbol{\theta}_b}$, $b = 1, \ldots, B'$. We then stack the B' neural networks along with their individual probability map outputs $\mathbf{p}^{(u)}[b]$, $b = 1, \ldots, B'$, and add a bias-free linear layer whose weight matrix is initialized with

$$\bar{\mathbf{G}} = \frac{1}{B'} \big[\underbrace{\mathbf{G}, \ldots, \mathbf{G}}_{B' \text{ times}} \big] \tag{18}$$

[5] According to [58, Sec. 4], the Shannon information of an event $\mathcal{A}$ is defined as $S(\mathcal{A}) = -\log_2(P[\mathcal{A}])$, where $P[\mathcal{A}]$ is the probability of $\mathcal{A}$.

to its output. This final linear layer, combined with the stacked probability map vector

$$\bar{\mathbf{p}}^{(u)} = \left[\mathbf{p}^{(u)}[1]^T, \ldots, \mathbf{p}^{(u)}[B']^T\right]^T, \tag{19}$$

computes $\hat{\mathbf{x}}^{(u)} = \bar{\mathbf{G}}\bar{\mathbf{p}}^{(u)}$ as in (17). For the same training set consisting of ground-truth locations $\{\mathbf{x}^{(u)}[b]\}_{u=1}^{U'}$ and associated CSI-feature vectors $\{\mathbf{f}^{(u)}[b]\}_{u=1}^{U'}$, we then continue weight learning of the final layer $\bar{\mathbf{G}}$ by minimizing the mean distance error (MDE) loss defined as

$$L_{\text{MDE}} = \frac{1}{U'}\sum_{u=1}^{U'} \left\|\mathbf{x}^{(u)} - \hat{\mathbf{x}}^{(u)}\right\|_2. \tag{20}$$

Since we continue learning $\bar{\mathbf{G}}$ after initializing it with the matrix in (18), we expect this method to perform no worse than keeping $\bar{\mathbf{G}}$ fixed and performing averaging as in (17).

Remark 9. *NN-based probability fusion requires one to pass all probability map estimates with a total number of $K \times B'$ real numbers to a centralized processor, which multiplies the estimates with the trained weights. The complexity of NN-based probability fusion is $D \times B' \times K$ real-valued multiplications.*

Remark 10. *One could also retrain the weights and biases contained in the network parameters $\boldsymbol{\theta}_b$, $b = 1, \ldots, B$, of the B' networks when learning the weights in the matrix $\bar{\mathbf{G}}$. We have, however, not observed accuracy improvements by doing so.*

V. Results

We now evaluate the performance of the proposed positioning pipeline for a range of indoor CSI measurements. We first describe the system setup, measurement scenarios, and performance metrics. We then show accuracy results for a range of multi-antenna and multi-point probability fusion methods.

A. Measurement Setup

Our measurement setup consists of five components: (i) A portable UE with wireless LAN (WLAN) connectivity; we use a Raspberry Pi Model 4 that is equipped with a two-antenna IEEE 802.11ac transceiver. (ii) A robot equipped with an embedded processor to move the UE; we use iRobot Roomba Create 2 controlled by the Raspberry Pi. (iii) A WLAN AP that enables the extraction of CSI; the AP is equipped with a four-antenna IEEE 802.11ac transceiver operating at 5 GHz with 80 MHz bandwidth and provides an API to access the raw per-subcarrier CSI for each receive and transmit antenna. (iv) A precise positioning system for ground-truth position extraction; we use use two different systems depending on the scenario: A WorldViz PPT-N active point tracking system [56] with sub-millimeter positioning accuracy enabled by triangulation with four infrared (IR) camera readings of active IR transmitters, and Vicon Vero passive point tracking system [57] with sub-millimeter positioning accuracy enabled by triangulation with twelve motion capture camera readings of passive reflective markers. (iv) A host computer that is collecting CSI measurements, running the precision positioning systems, and controlling the robot. Figure 4 illustrates the measurement setup.

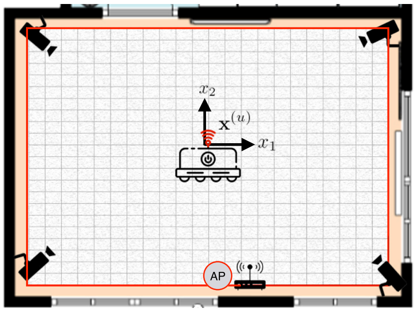


Fig. 4. Measurement setup. We use a robot that carries a WLAN transmitter; one or multiple APs then record CSI measurements. An accurate position estimate of the robot $\mathbf{x}^{(u)}$ is extracted using a multi-camera precision localization system and recorded for NN training and testing.

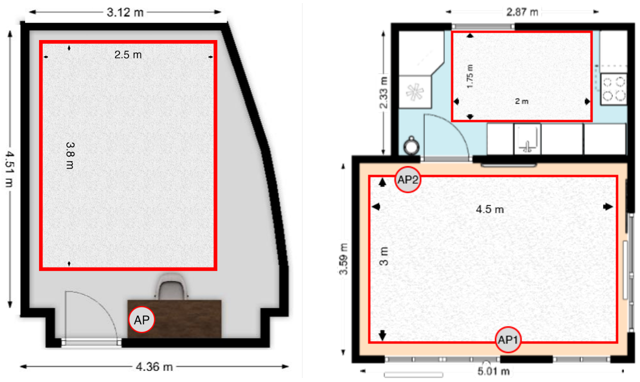
The data collection procedure is as follows: We control the robot's position to follow a predefined path in piecewise linear movements over two dimensions $\mathbf{x} = [x_1, x_2]^T$. The Raspberry Pi continuously transmits high-quality images from an on-board camera to the host PC via the two transmit antennas $M_\text{T} = 2$. At the same time, the AP is receiving data and CSI from the four antennas $M_\text{R} = 4$, and the precision positioning system is extracting ground-truth position information; all the data is stored at the host computer. Since we measure CSI and ground truth positions from two separate sources, we first synchronize the operating-system clocks of the Raspberry Pi and the precision positioning system running at the host PC via the network time protocol (NTP). We then match the measurement times of the CSI and the precision positioning system. For the multi-point measurement, we perform two separate measurement campaigns, i.e., we first measure CSI at AP1 and ground-truth position for the entire area and later we measure CSI at AP2 and ground-truth position while following approximately the same track with the robot. We then match the CSI from both APs so that there is less than 3 cm ground-truth position difference between both measurements.

Remark 11. *There exists a trade-off between the training dataset size and the positioning accuracy. We have observed that larger training sets generally improve accuracy (with diminishing returns), but also increase the time to collect such datasets. The optimal amount of training required in practice depends on the capabilities of the positioning system (in terms of minimum achievable positioning error), the scenario, as well as the time required to acquire the dataset.*

Remark 12. *Since our multi-AP measurements are acquired at two different time instants—more than tens of minutes apart—our results in Section V-E imply that accurate positioning from multi-AP measurements does not need exact synchronization or simultaneous recording of CSI.*

B. Measurement Scenarios

In order to evaluate the proposed multi-antenna and multi-point probability fusion positioning pipeline, we consider



(a)

(b)

Fig. 5. Floor plans used to perform indoor positioning experiments. (a) Lab floor plan with an area of 2.5m × 3.8m. (b) Living room (bottom; area 3m × 4.5m) and kitchen (top; area 1.75m × 2m) floor plans. The access point (AP) locations are marked by circles.

four different scenarios that were measured in three different locations: Lab space, living room, and kitchen. Figure 5 shows the floor plans of these locations as well as the AP positions. For each location, we divided the data acquisition process into two phases: In the first phase, we recorded a training set containing CSI measurements and ground-truth locations. In the second phase, we recorded a separate test set that is used solely for characterizing the position accuracy of our positioning pipeline.[6] During both of these phases, human activity was present in the environment, including people walking around the room, opening and closing doors, turning small appliances off and on, etc. This data collection approach prevents overfitting and demonstrates a certain degree of robustness to dynamic changes in the environment.[7]

a) Single-Point LoS Lab Scenario: We collected CSI and ground-truth position in the lab space shown in Figure 5(a) under line-of-sight (LoS) conditions. For the training set, the robot moved through a pre-defined area in the lab space over two days; Figure 6(a) shows the robot's path. For the test set, the robot followed a "VIP" shaped path (shown with blue color) which was recorded separately from the training data on another day.

b) Single-Point Non-LoS Living Room Scenario: We collected CSI and ground-truth position in the living room shown at the bottom of Figure 5(b) under non-LoS conditions; non-LoS conditions are achieved by placing AP1 behind a TV. For the training set, the robot was moving randomly through a predefined area in the living room; Figure 6(b) shows the robot's path. For the test set, the robot followed a "VIP" shaped path (shown with blue color), which was recorded separately but on the same day.

c) Single-Point Non-LoS Kitchen Scenario: We collected CSI and ground-truth position in the kitchen shown at the top of Figure 5(b) under non-LoS conditions as we used AP1. For the training set, the robot was moving randomly through a predefined area in the kitchen; Figure 6(c) shows the robot's path. For the test set, we used 20% randomly selected measurements from the robot's path (shown with blue color), which were recorded separately but on the same day.

d) Multi-Point LoS Living Room Scenario: We collected CSI and ground-truth position in the living room area shown at the bottom of Figure 5(b) under LoS conditions, where we used both AP1 and AP2. For the training set, we recorded CSI at two distinct times on the same day: one recording for AP1 and one for AP2. For both of these recordings, the robot was following a pre-defined path; Figure 6(d) shows the robot's path. For the test set, the robot followed a rectangular-shaped path (shown with blue color), which was recorded separately but on the same day.

C. Performance Metrics and Positioning Methods

For all experiments, we use the CSI-features detailed in Section III-A and the NN topology discussed in Section III-B. In order to determine the optimal grid resolution, we have conducted experiments with different $K \times K$ rectangular uniform grids with $K \in \{4, 6, 12, 22, 32, 42\}$ for the single-point LoS lab scenario. Figure 7 shows the associated distance errors. We see that a resolution of $K = 22$ (which leads to a total of $K^2 = 484$ grid points) results in the lowest mean and 95th percentile errors, which is what we have selected for all of our experiments.

In order to evaluate the positioning performance of the compared methods, we first obtain position estimates for the test set and then we compute three different metrics: (i) The mean distance error (MDE), i.e., the expression in (20) evaluated over the test set, (ii) the median distance error, and (iii) the 95th percentile error. The median and 95th percentile distance errors were obtained by sorting the distances $\|\mathbf{x}^{(u)} - \hat{\mathbf{x}}^{(u)}\|_2$, $u = 1, \ldots, U'$, of the test set in ascending order and extracting the 50% and 95% distances, respectively.

For the multi-antenna and multi-point scenarios, we use the probability fusion methods discussed in Section IV. Concretely, we compare the performance of CSI-based positioning for the following methods. For single-point and single-antenna experiments, we train a single neural network (NN) for one transmit antenna and one AP; we label these methods as "1 NN, APb TXm," where "1 NN" implies that we use a single NN, $b = 1, \ldots, B$ is the AP number, and $m = 1, \ldots, M_\text{T}$ the transmit (TX) antenna number. For multi-antenna and/or multi-point experiments, we also a single NN that stacks all features extracted from all B APs in a single long CSI-feature vector[8], i.e., $\mathbf{f}_m^{(u)}[b]$ for $b = 1, \ldots, B$ and $m = 1, \ldots, M_\text{T}$. We label this method as "1 NN, stacked features."

In what follows, we will compare this method to probability fusion-based approaches. For fusion-based methods, we train two or four NNs depending on the scenario. For two transmit antennas and one AP (i.e., multi-antenna and single-point) or for two APs and one transmit antenna (i.e., single-antenna and multi-point), we train two NNs and perform probability fusion from two probability maps. We consider the following methods: "2 NN, averaging," where we implement unweighted averaging

[6] We note that CSI-based positioning methods that utilize DNNs do not generalize well to untrained areas; see, e.g., [43].

[7] We note that it is challenging to acquire a dataset in which the dynamic changes in the environment are well-controlled and reproducible.

[8] We note that the same approach has been proposed in [40], which appeared after the submission of our manuscript.

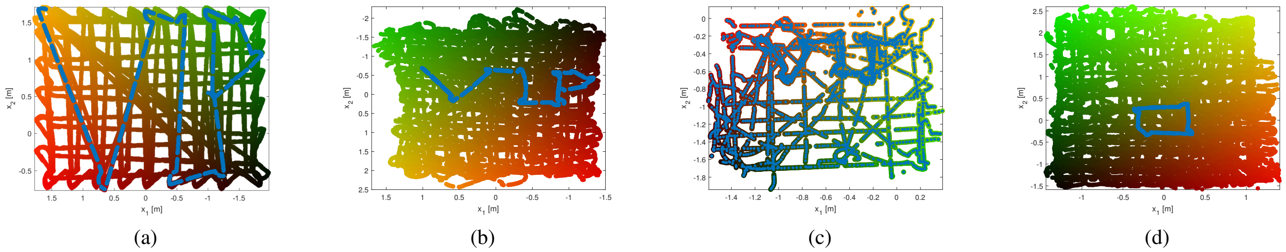


Fig. 6. Ground-truth positions collected for different scenarios: (a) Single-point LoS lab; (b) single-point non-LoS living room; (c) single-point non-LoS kitchen; (d) multi-point LoS living room. The gradient-colored curves represent locations used for training; the blue curves correspond to the test set.

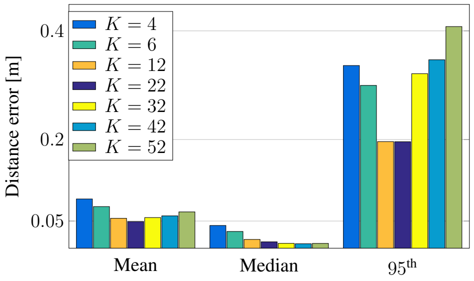


Fig. 7. Impact of grid resolution on distance error. A $K \times K$ grid with $K = 22$ yields the lowest mean and 95th percentile errors. Increasing the resolution further ($K > 22$) does not provide significant improvements in terms of the median distance error.

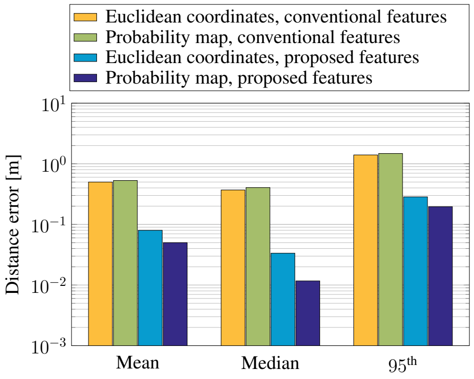


Fig. 8. Bar plot showing mean, median, and 95th percentile distance error evaluated on the test set for the single-point LoS lab scenario detailed in Section V-B. We show a NN trained with the CSI features from [19] using a NN that computes 2D coordinates (orange), and probability maps (green); and with features as in Section III-A using (i) a NN that computes 2D coordinates (light-blue), and (ii) a NN that computes probability maps (dark-blue). Combining our CSI features with probability maps results in significant accuracy improvements.

as in (17); "2 NN, prob. conflation," where we implement probability conflation as detailed in Section IV-A; "2 NN, Gaussian conflation," where we implement Gaussian conflation as detailed in Section IV-B; and "2 NN, NN fusion," where we implement the NN-based fusion as detailed in Section IV-C. For two transmit antennas and two APs (i.e., multi-antenna and multi-point), we train four NNs and perform probability fusion from four probability maps; the associated methods are labeled as "4 NN, averaging," "4 NN, prob. conflation," "4 NN, Gaussian conflation," and "4 NN, NN fusion."

D. Comparison with Single-AP Localization Approaches

Before we discuss the experimental results for multi-AP positioning, we first assess the impact of CSI features and probability maps on positioning accuracy on a single-AP scenario. We consider the single-point LoS scenarios in Section V-B with *conventional* CSI features proposed in [19] which uses magnitudes of beamspace representation of CSI measurements as features. Figure 8 depicts the mean, median, and 95th percentile distance error obtained by (i) training a NN that generates Euclidean coordinates using conventional CSI features [19], [21][9] discussed in Section III-A1 (orange), (ii) training a NN that generates probability maps as in Section III-C using the CSI features as in [19], [21] (green), (iii) training a NN which generates Euclidean coordinates using our feature extraction approach (light-blue) as in Section III-A2, and (iv) training a NN that generates probability maps as in Section III-C using our feature extraction approach (dark-blue) as in Section III-A2. For the NN that generates Euclidean coordinates, we use the same architecture as for the NN that generates probability maps described in Section III-B, and we use a mean-squared-error loss function. We observe significant accuracy improvements with our CSI features for real-world 802.11ac-OFDM measurements over conventional features that have been designed specifically for outdoor multi-user (MU) massive MIMO-OFDM channels. In addition, the use of probability maps not only results in an additional 3 cm MDE improvement, but also enables probability fusion from multiple APs as detailed in Section IV.

Remark 13. *We have conducted* synthetic *experiments with the same positioning pipeline where we used the positions obtained in the single-AP scenario to generate CSI using the QuaDRiGa "WINNER_Indoor_A1_LOS" channel model [67] with the same system parameters. With our positioning pipeline, we have observed a* $0.9\,cm$ *mean distance error, a* $0.8\,cm$ *median distance error, and a* $2.3\,cm$ *95th percentile error. However, such synthetic experiments are not representative as they lack of real-world propagation conditions and hardware impairments.*

E. Positioning Results

Figure 9 shows bar plots evaluated on the test sets for multi-antenna scenarios. Figure 10 shows bar plots evaluated for a

[9] Reference [21] uses convolutional neural networks (CNNs) to estimate location. In our setup, we have observed no improvement in using deep CNNs over simple feedforward networks.

multi-point scenario. Figure 11 shows bar plots for a multi-antenna multi-point scenario. For all three figures, the left bar plot shows the mean distance error, the middle bar plot shows the median distance error, and the right bar plot shows the 95th percentile distance error.

1) Multi-Antenna Results: Figures 9(a), 9(b), and 9(c) show positioning results corresponding to the multi-antenna scenarios a), b), and c) detailed in Section V-B. We observe that when using CSI-features from different transmit antennas but from a single AP, the accuracy can vary significantly across antennas. For the probability fusion methods, we see that simple averaging and NN-based fusion performs equally well. The best performing methods are the use of a single NN with stacked features (labeled by "1 NN, stacked features"), as well as probability conflation (labeled by "2 NN, prob. conflation") and Gaussian conflation (labeled by "2 NN, Gaussian conflation").

2) Multi-Point Results: Figure 10 shows positioning results corresponding to the multi-point scenario d) detailed in Section V-B where we only use one transmit antenna. We observe that the accuracy of using AP1 is superior than that of AP2. Considering the APs have identical hardware, we conclude that the AP position may affect the positioning accuracy. Furthermore, we see that when fusing the probability maps from AP1 and AP2, probability conflation and Gaussian conflation performs equally well as the single NN with stacked features. However, the amount of data to be transferred to a centralized processor for the Gaussian conflation approach is significantly lower than those of the single NN with stacked features and probability conflation.

3) Multi-Antenna Multi-Point Results: Figure 11 shows positioning corresponding to the multi-point scenario detailed in Section V-B where we use both transmit antennas. We see that the accuracy of the first transmit antenna (TX1) of AP1 is superior to the other antenna-AP combinations. We also see that the AP position and the used transmit antennas have an effect on the positioning accuracy. Furthermore, we see that Gaussian conflation outperforms all other fusion approaches for the considered performance metrics.

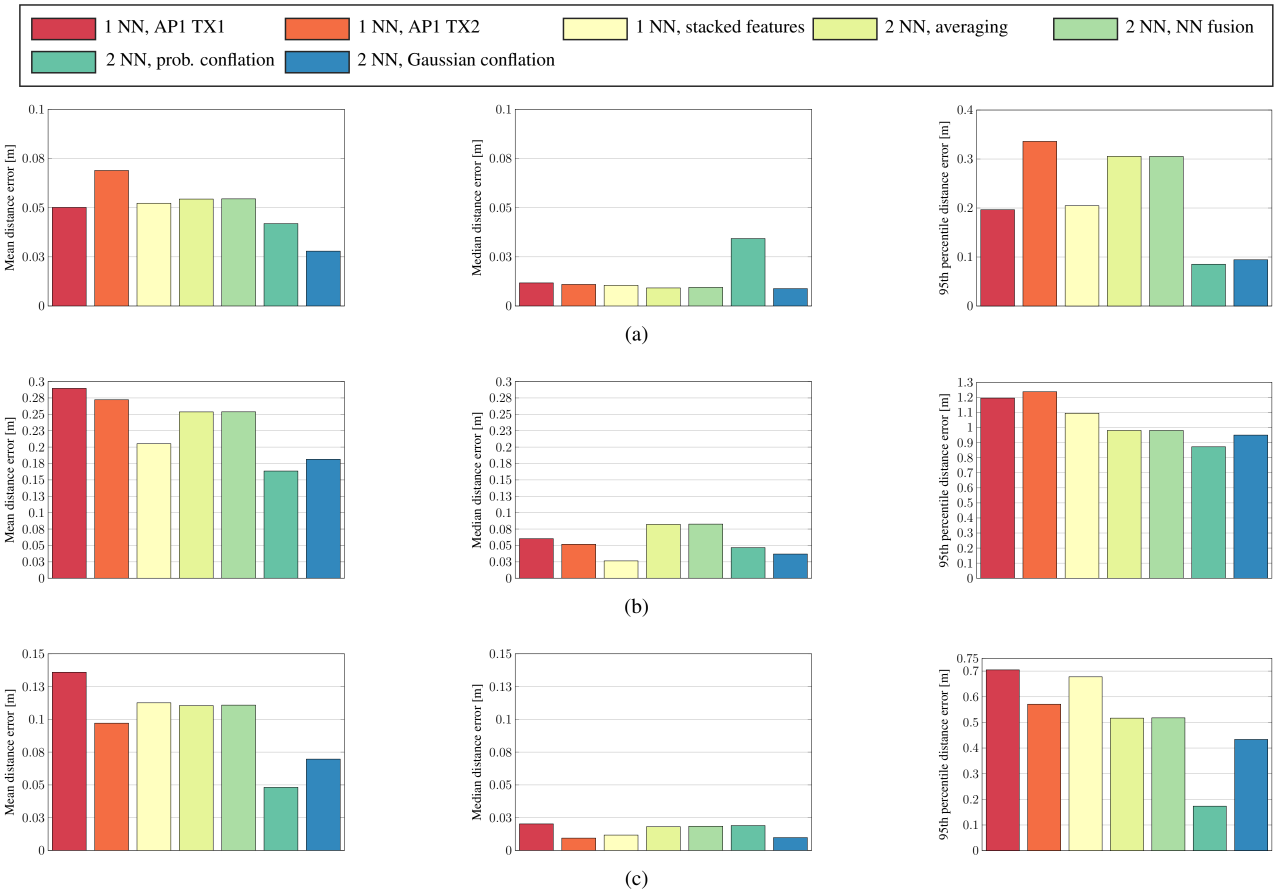
4) Visualization of Probability Fusion: Figure 12 illustrates the efficacy of probability fusion, where we show the ground-truth positions in Figure 12(a) on the test-set for scenario d) in Section V-B and estimated position using the proposed methods. Figure 12(b) shows that the estimated locations at AP2 from the second transmit antenna (TX2) results in quite a few outliers. When performing a single NN with stacked features and multiple NNs with Gaussian conflation in Figure 12(c) in Figure 12(d), respectively, one can see that the position accuracy significantly improves. We observe that probability fusion yields superior accuracy than using only the individual AP estimates. Furthermore, we see that NN-based probability fusion performs on par with simple averaging method. We speculate that much more training data and more complicated neural networks would be required to improve NN-based probability fusion. Clearly, treating the information contained in the probability maps as Gaussians and fusing these mean-variance pairs results in accurate indoor positioning while only requiring a minimum amount of data transmission to a centralized processor that performs fusion. Among the studied probability fusion methods, we note that each method has pros and cons in terms of complexity and data transfer size. For a use-case where the data transfer and/or complexity is not a concern, one could use either probability conflation or NN-based probability fusion. For a use-case where data transfer and/or complexity is severely constrained, Gaussian conflation often outperforms the other probability fusion methods.

Remark 14. *We emphasize that the 1 NN method with a single stacked feature results in excessively large features in the first layer, which substantially increases computational complexity and storage. Furthermore, the stacked feature approach requires centralized processing of all CSI features, which requires a large number of data to be transferred to a centralized processor. In contrast, Gaussian conflation requires the transfer of only mean-variance pairs to the centralized processor, while resulting in similar or often superior accuracy.*

Remark 15. *We have identified six key factors that affect positioning accuracy: (i) line-of-sight connectivity, (ii) the number of transmit and receive antennas, (iii) the number of APs, (iv) the number of subcarriers, (v) the number of training locations, and (vi) the CSI features. We expect to achieve superior accuracy in both LoS and NLoS conditions with more APs and with more receive antennas. Moreover, we have observed that larger training sets reduce positioning accuracy; however, the acquisition of large datasets is not always feasible due to practical limits (e.g., acquisition time or storage). We note that the positioning accuracy is likely to degrade for areas that were not included in the training set [43]. We hope to be able to reduce the amount of training and to improve generalization to untrained areas by combining our pipeline with self-supervised localization methods, such as channel-charting [27].*

VI. Conclusions

We have proposed CSI-based indoor positioning methods that are able to fuse one or multiple probability maps which describe the UEs' positions. We have used a NN-based positioning pipeline that takes in features designed for MIMO-OFDM-based systems and are robust to typical hardware impairments, and generates probability maps. We have proposed three different fusion methods for the computed probability maps, which reduce the amount of data to be transferred to a centralized processor that estimates UE position. To demonstrate the effectiveness of the proposed positioning methods, we have evaluated our methods on four real-world indoor positioning datasets, which include multi transmit-antenna and multi AP scenarios. Our comparison reveals three facts: (i) Indoor position accuracy of a few centimeters is possible from IEEE 802.11ac measurements, (ii) simple probability fusion techniques can significantly improve positioning accuracy while reducing the amount of data to be transported to a centralized processor, and (iii) multi-point probability fusion does not require accurate synchronization between the APs. We believe that the proposed probability fusion approach paves the way for other positioning systems or scenarios in which multiple features from different sensors are available.



(a)

(b)

(c)

Fig. 9. Bar plots showing mean distance error (left), median distance error (middle), and the 95th percentile distance error (right) evaluated on the test set for three multi-antenna scenarios: (a) single-point LoS lab scenario, (b) single-point non-LoS living room scenario, and (c) single-point non-LoS kitchen scenario.

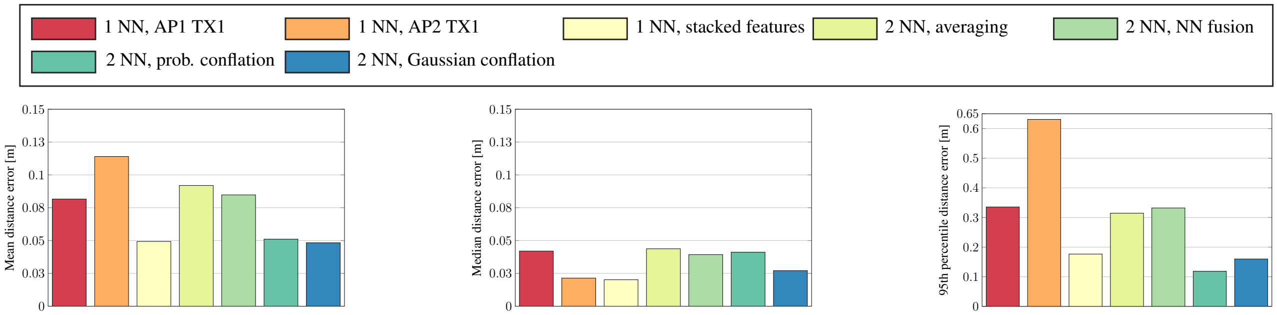


Fig. 10. Bar plots showing mean distance error (left), median distance error (middle), and the 95th percentile distance error (right) evaluated on the test set for the single-antenna multi-point LoS living room scenario detailed in Section V-B.

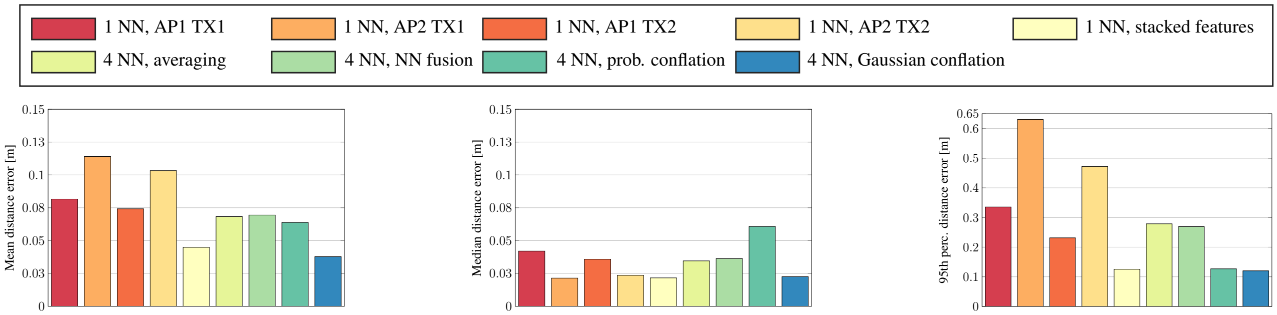
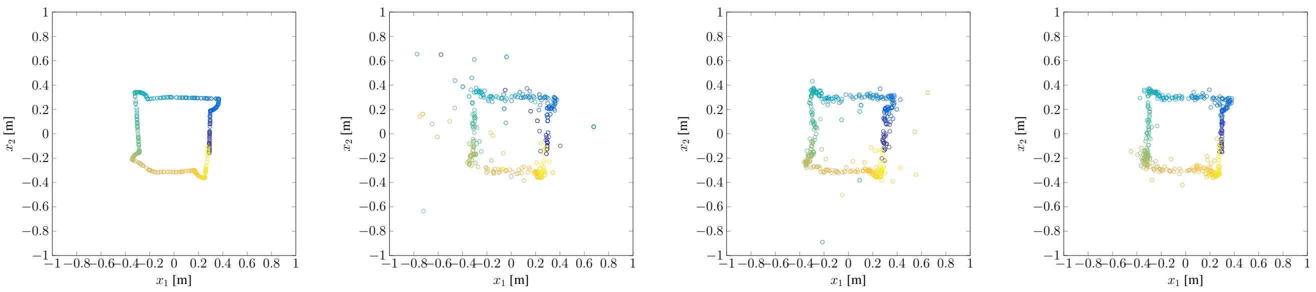


Fig. 11. Bar plots showing mean distance error (left), median distance error (middle), and the 95th percentile distance error (right) evaluated on the test set for the multi-antenna multi-point LoS living room scenario detailed in Section V-B.



(a) (b) (c) (d)

Fig. 12. Ground-truth locations (a) and estimated location (b), (c), and (d) for the multi-point LoS living room scenario detailed in Section V-B. (b) shows the performance of using a single NN from AP2 and TX1 (*MDE* $=$ 11cm); (c) shows the performance of 1 NN with stacked features from 2 APs and 2 TX antennas (*MDE* $=$ 4.9cm); and (d) shows the performance of 4 NNs with Gaussian conflation (*MDE* $=$ 3.7cm).

There are many avenues for future work. First and foremost is the exploration of channel-charting based methods that (i) mitigate the need for dedicated CSI and ground-truth position measurement campaigns and (ii) generalize to the unseen positions. Second is the development of accurate positioning pipelines that fuse multiple sensor modalities (besides CSI) which is part of ongoing work. Third is the evaluation of our approach in larger indoor spaces with multiple rooms and with a larger number of distributed APs. Last is the acquisition of datasets with 3D position information and more (and controlled) dynamic changes in the environment.

VII. Acknowledgments

The authors would like to thank O. Castañeda and B. Rappaport for discussions on CSI-based positioning using neural networks. We thank D. Nonaca for useful comments on a preprint of this paper. We also thank Prof. K. Petersen for allowing us to use the Vicon positioning system [57] and the lab space at Cornell University shown in Figure 5(a).

References

[1] S. Han, Z. Gong, W. Meng, C. Li, and X. Gu, “Future alternative positioning, navigation, and timing techniques: A survey,” *IEEE Wireless Commun.*, vol. 23, no. 6, pp. 154–160, Oct. 2016.

[2] N. Fallah, I. Apostolopoulos, K. Bekris, and E. Folmer, “Indoor human navigation systems: A survey,” *Interacting with Computers*, vol. 25, no. 1, pp. 21–33, Jan. 2013.

[3] F. Wen, H. Wymeersch, B. Peng, W. P. Tay, H. C. So, and D. Yang, “A survey on 5G massive MIMO localization,” *Digital Signal Process.*, vol. 94, pp. 21–28, Nov. 2019.

[4] R. F. Brena, J. P. García-Vázquez, C. E. Galván-Tejada, D. Muñoz-Rodriguez, C. Vargas-Rosales, and J. Fangmeyer, “Evolution of indoor positioning technologies: A survey,” *J. Sensors*, vol. 2017, Mar. 2017.

[5] W. Y. Al-Rashdan and A. Tahat, “A comparative performance evaluation of machine learning algorithms for fingerprinting based localization in DM-MIMO wireless systems relying on big data techniques,” *IEEE Access*, vol. 8, pp. 109 522–109 534, Jun. 2020.

[6] J. Armstrong, Y. A. Sekercioglu, and A. Neild, “Visible light positioning: A roadmap for international standardization,” *IEEE Commun. Mag.*, vol. 51, no. 12, pp. 68–73, Dec. 2013.

[7] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, “Luxapose: Indoor positioning with mobile phones and visible light,” in *Proc. 20th Annual Int. Conf. Mobile Comput. Networking*, Sep. 2014, pp. 447–458.

[8] H. Koyuncu and S. H. Yang, “A survey of indoor positioning and object locating systems,” *Intl. J. Comput. Science Network Security*, vol. 10, no. 5, pp. 121–128, May 2010.

[9] C. Lee, Y. Chang, G. Park, J. Ryu, S.-G. Jeong, S. Park, J. W. Park, H. C. Lee, K. shik Hong, and M. H. Lee, “Indoor positioning system based on incident angles of infrared emitters,” in *Proc. Ann. Conf. of IEEE Ind. Electron. Soc.*, vol. 3, Nov. 2004, pp. 2218–2222.

[10] H. Liu, H. Darabi, P. Banerjee, and J. Liu, “Survey of wireless indoor positioning techniques and systems,” *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[11] W. Liu, Q. Cheng, Z. Deng, H. Chen, X. Fu, X. Zheng, S. Zheng, C. Chen, and S. Wang, “Survey on CSI-based indoor positioning systems and recent advances,” in *Proc. Intl. Conf. Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2019, pp. 1–8.

[12] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. M. Ni, “CSI-based indoor localization,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 7, pp. 1300–1309, Jul. 2013.

[13] F. Gustafsson and F. Gunnarsson, “Mobile positioning using wireless networks: Possibilities and fundamental limitations based on available wireless network measurements,” *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 41–53, Jul. 2005.

[14] Z. Sahinoglu, S. Gezici, and I. Guvenc, “Ultra-wideband positioning systems,” *Cambridge, New York*, 2008.

[15] C. Zhang, Y. Ueng, C. Studer, and A. Burg, “Artificial intelligence for 5G and beyond 5G: Implementations, algorithms, and optimizations,” *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 10, no. 2, pp. 149–163, Jun. 2020.

[16] M. Arnold, J. Hoydis, and S. ten Brink, “Novel massive MIMO channel sounding data applied to deep learning-based indoor positioning,” in *Intl. ITG Conf. Systems, Commun., Coding*, Feb. 2019, pp. 1–6.

[17] X. Wang, L. Gao, S. Mao, and S. Pandey, “CSI-based fingerprinting for indoor localization: A deep learning approach,” *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.

[18] Z. Yang, Z. Zhou, and Y. Liu, “From RSSI to CSI: Indoor localization via channel response,” *ACM Comput. Surveys*, vol. 46, no. 2, pp. 1–32, Nov. 2013.

[19] E. Lei, O. Castañeda, O. Tirkkonen, T. Goldstein, and C. Studer, “Siamese neural networks for wireless positioning and channel charting,” in *Proc. 57th Ann. Allerton Conf. Commun., Control, and Comput.*, Sep. 2019, pp. 200–207.

[20] P. Ferrand, A. Decurninge, and M. Guillaud, “DNN-based localization from channel estimates: Feature design and experimental results,” *ArXiv preprint: 2004.00363*, Apr. 2020.

[21] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, “Deep convolutional neural networks for massive MIMO fingerprint-based positioning,” in *Proc. IEEE Intl. Symp. Personal, Indoor, Mobile Radio Commun.*, Oct. 2017, pp. 1–6.

[22] V. Savic and E. G. Larsson, “Fingerprinting-based positioning in distributed massive MIMO systems,” in *Proc. IEEE 82nd Vehic. Tech. Conf.*, Sep. 2015, pp. 1–5.

[23] A. Zappone, M. Di Renzo, and M. Debbah, “Wireless networks design in the era of deep learning: Model-based, AI-based, or both?” *IEEE Trans. Commun.*, vol. 67, no. 10, pp. 7331–7376, Jun. 2019.

[24] X. Wang, L. Gao, S. Mao, and S. Pandey, “DeepFi: Deep learning for indoor fingerprinting using channel state information,” in *Proc. IEEE Wireless Commun. Netw. Conf.*, Mar. 2015, pp. 1666–1671.

[25] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, “ConFi: Convolutional neural networks based indoor Wi-Fi localization using channel state information,” *IEEE Access*, vol. 5, pp. 18 066–18 074, Sep. 2017.

[26] X. Wang, X. Wang, and S. Mao, “Indoor fingerprinting with bimodal CSI tensors: A deep residual sharing learning approach,” *IEEE Internet Things J.*, Sep. 2020.

[27] C. Studer, S. Medjkouh, E. Gönültaş, T. Goldstein, and O. Tirkkonen, “Channel charting: Locating users within the radio environment using channel state information,” *IEEE Access*, vol. 6, pp. 47 682–47 698, Aug. 2018.

[28] S. Medjkouh, E. Gönültaş, T. Goldstein, O. Tirkkonen, and C. Studer, "Unsupervised charting of wireless channels," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–7.

[29] P. Ferrand, A. Decurninge, L. G. Ordoñez, and M. Guillaud, "Triplet-based wireless channel charting," *ArXiv preprint: 2005.12242*, May 2020.

[30] J. Deng, S. Medjkouh, N. Malm, O. Tirkkonen, and C. Studer, "Multipoint channel charting for wireless networks," in *Proc. IEEE Conf. Rec. Asilomar Conf. Signals, Sys., and Comp.*, Feb. 2018, pp. 286–290.

[31] C. Geng, H. Huang, and J. Langerman, "Multipoint channel charting with multiple-input multiple-output convolutional autoencoder," in *Proc. IEEE/ION Position, Location Navigation Symp. (PLANS)*, Apr. 2020, pp. 1022–1028.

[32] F. Gringoli, M. Schulz, J. Link, and M. Hollick, "Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets," in *Proc. 13th Int. W. Wireless Network Testbeds Experimental Evaluation Characterization*, Oct. 2019, pp. 21–28.

[33] S. He and S. . G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 466–490, Aug. 2016.

[34] Y. Ma, G. Zhou, and S. Wang, "WiFi sensing with channel state information: A survey," *ACM Comput. Surv.*, vol. 52, no. 3, Jun. 2019.

[35] W. Liu, Q. Cheng, Z. Deng, H. Chen, X. Fu, X. Zheng, S. Zheng, C. Chen, and S. Wang, "Survey on CSI-based indoor positioning systems and recent advances," in *Proc. Int. Conf. on Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2019, pp. 1–8.

[36] Y. Chapre, A. Ignjatovic, A. Seneviratne, and S. Jha, "CSI-MIMO: Indoor Wi-Fi fingerprinting system," in *Proc. 39th Ann. IEEE Conf. Local Computer Networks*, Sep. 2014, pp. 202–209.

[37] X. Wang, L. Gao, and S. Mao, "BiLoc: Bi-modal deep learning for indoor localization with commodity 5GHz wifi," *IEEE Access*, vol. 5, pp. 4209–4220, Mar. 2017.

[38] B. Berruet, O. Baala, A. Caminada, and V. Guillet, "DelFin: A deep learning based CSI fingerprinting indoor localization in IoT context," in *Proc. Int. Conf. on Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2018, pp. 1–8.

[39] S. D. Bast, A. P. Guevara, and S. Pollin, "CSI-based positioning in massive MIMO systems using convolutional neural networks," in *Proc. IEEE Veh. Technol. Conf.*, May 2020, pp. 1–5.

[40] P. Li, H. Cui, A. Khan, U. Raza, R. Piechocki, A. Doufexi, and T. Farnham, "Wireless localisation in WiFi using novel deep architectures," *ArXiv preprint: 2010.08658*, Oct. 2020.

[41] A. Foliadis, M. H. Garcia, A. S.-G. A., and T. S. Reiner, "CSI-based localization with CNNs exploiting phase information," in *IEEE Wireless Commun. Networking Conf.*, Apr. 2021, pp. 1–6.

[42] L. Tang, R. Ghods, and C. Studer, "Reducing the complexity of fingerprinting-based positioning using locality-sensitive hashing," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Nov. 2019, pp. 1086–1090.

[43] M. Arnold, S. Dorner, S. Cammerer, and S. Ten Brink, "On deep learning-based massive MIMO indoor user localization," in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Jun. 2018, pp. 1–5.

[44] S. Fang, T. Lin, and K. Lee, "A novel algorithm for multipath fingerprinting in indoor WLAN environments," *IEEE Trans. Wireless Commun.*, vol. 7, no. 9, pp. 3579–3588, Sep. 2008.

[45] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. 3rd Intl. Conf. Mobile sys., Applications, Services*, Jun. 2005, pp. 205–218.

[46] J. Xiao, K. Wu, Y. Yi, and L. M. Ni, "FIFS: Fine-grained indoor fingerprinting system," in *Proc. 21st Intl. Conf. Computer Commun. Networks (ICCCN)*, Aug. 2012.

[47] X. Sun, C. Wu, X. Gao, and G. Y. Li, "Fingerprint-based localization for massive MIMO-OFDM system with deep convolutional neural networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 10 846–10 857, Sep. 2019.

[48] D. Vasisht, S. Kumar, and D. Katabi, "Sub-nanosecond time of flight on commercial Wi-Fi cards," in *Proc. ACM Conf. on Special Interest Group Data Commun.*, vol. 45, no. 4. Association for Computing Machinery, Aug. 2015, pp. 121–122.

[49] Y. Liu, W. Xiong, Z. Zhu, and S. Li, "CSI based high accuracy device free passive localization system," in *Proc. IEEE Veh. Technol. Conf.*, Aug. 2018, pp. 1–5.

[50] P. Agostini, Z. Utkovski, and S. Stańczak, "Channel charting: An Euclidean distance matrix completion perspective," in *Proc. IEEE Intl. Conf. Acoustics, Speech and Signal Proces.*, May 2020, pp. 5010–5014.

[51] S. Acharya and M. Kam, "Evidence combination for hard and soft sensor data fusion," in *Proc. Intl. Conf. Inf. Fusion*, Jul. 2011, pp. 1–8.

[52] R. Krzysztofowicz and D. Long, "Fusion of detection probabilities and comparison of multisensor systems," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, no. 3, pp. 665–677, May-June 1990.

[53] M. Aeberhard, S. Paul, N. Kaempchen, and T. Bertram, "Object existence probability fusion using dempster-shafer theory in a high-level sensor data fusion architecture," in *Proc. IEEE Intelligent Veh. Symp.*, Jun. 2011, pp. 770–775.

[54] G. Wen, Z. Hou, H. Li, D. Li, L. Jiang, and E. Xun, "Ensemble of deep neural networks with probability-based fusion for facial expression recognition," *Cognitive Computation*, vol. 9, no. 5, pp. 597–610, 2017.

[55] N. Garcia, H. Wymeersch, E. G. Larsson, A. M. Haimovich, and M. Coulon, "Direct localization for massive MIMO," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2475–2487, Feb. 2017.

[56] WorldViz. Precision position tracking (PPT). [Online]. Available: https://www.worldviz.com/virtual-reality-motion-tracking

[57] VICON. Vero family motion capture system. [Online]. Available: https://www.vicon.com/hardware/cameras/vero/

[58] T. P. Hill, "Conflations of probability distributions," *Trans. Amer Math. Soc.*, vol. 363, no. 6, pp. 3351–3372, Jun. 2011.

[59] P. Huang, O. Castañeda, E. Gönültaş, S. Medjkouh, O. Tirkkonen, T. Goldstein, and C. Studer, "Improving channel charting with representation-constrained autoencoders," in *Proc. IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Aug. 2019, pp. 1–5.

[60] IEEE 802.11ac-2013, "Draft standard for information technology — telecommunications and information exchange between systems — local and metropolitan area networks — specific requirements — part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," IEEE, Tech. Rep., 2013.

[61] S. Haene, A. Burg, P. Luethi, N. Felber, and W. Fichtner, "FFT processor for OFDM channel estimation," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2007, pp. 1417–1420.

[62] T. Schenk, *RF imperfections in high-rate wireless systems: Impact and digital compensation.* Springer Science & Business Media, 2008.

[63] C. Studer, M. Wenk, and A. Burg, "MIMO transmission with residual transmit-RF impairments," in *Proc. Int. ITG Workshop Smart Antennas (WSA)*, Feb. 2010, pp. 189–196.

[64] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Thirteenth Int. Conf. Artificial Intelligence Stat.*, vol. 9, May 2010, pp. 249–256.

[65] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," 2014.

[66] J. Douglas and H. H. Rachford, "On the numerical solution of heat conduction problems in two and three space variables," *Trans. Amer Math. Soc.*, vol. 82, no. 2, pp. 421–439, Jul. 1956.

[67] S. Jaeckel, L. Raschkowski, K. Börner, and L. Thiele, "QuaDRiGa: A 3-D multi-cell channel model with time evolution for enabling virtual field trials," *IEEE Trans. Antennas Propag.*, vol. 62, no. 6, pp. 3242–3256, Mar. 2014.