Positioning for Distributed Large Intelligent Surfaces using Neural Network with Probabilistic Layer

Jesús Rodríguez Sánchez, Ove Edfors, and Liang Liu

Department of Electrical and Information Technology, Lund University, Sweden
{jesus.rodriguez, ove.edfors, and liang.liu}@eit.lth.se

***Abstract*—Wireless-based positioning with large antenna arrays is a promising enabler of the high accuracy positioning services envisioned for 6G. These systems provide high spatial resolution due to the large number of antennas, while enjoying the benefit of sharing a common infrastructure between communication and positioning. Among the available techniques for wireless-based positioning, channel state information (CSI)-based fingerprinting via neural networks (NNs) offers high accuracy under challenging propagation conditions, without the need of storing and accessing large amounts of measurement data during inference. On the other hand, large antenna systems, such as Large Intelligent Surfaces (LIS), benefits from a distributed architecture and local processing of wireless signals received from nearby antennas, producing intermediate results that can be aggregated, and therefore considerably reducing the demand on interconnection bandwidth. In this work, we propose a method to perform positioning of users based on estimated CSI in a LIS built from panels. Following this method, panels provide a parameterized probability density function for the location of each user, which can be shared conveniently and fused in different panels or a central processing unit (CPU), providing high positioning accuracy using very low interconnection bandwidth.**

I. Introduction

6G is envisioned to enable services based on high positioning accuracy in indoor and outdoor venues [1]. It is well known that large antenna arrays with very high spatial resolution are expected to be part of 6G radio-access systems. In addition to enabling the promised high communication data-rates, they can also provide high positioning accuracy. This facilitates a common infrastructure for communications and positioning, which is quite beneficial from cost and maintenance points of view. Large Intelligent Surfaces (LIS) is a technology providing these two capabilities [2], [3].

However, such large antenna systems, despite their benefits, present formidable implementation challenges, where computational and interconnection resources face critical bottlenecks that need to be overcome in order to realize these systems. In order to alleviate these limitations, a panelized LIS with tree topology has already been proposed for communication purposes [4], where panels contain processing capabilities to perform local baseband processing, with very little or limited cooperation among them. Individual panel results are aggregated before reaching the central processing unit (CPU), reducing considerably the interconnection bandwidth compared to a centralized LIS architecture, where raw baseband samples are shared with the CPU during the uplink detection phase, with the additional high computational burden.

In order to ensure LIS is able to support positioning applications (apart from communication ones), we explore efficient algorithms that can be mapped onto the tree-based panelized topology with distributed processing proposed in [4] without much hardware overhead. Following this idea, in this work we propose a method to perform positioning of users that suits such architecture. In this method, each panel estimates channel state information (CSI) (functionality available already in communication), which is further processed by a local neural network (NN) in order to map CSI to positioning information.

Neural networks have recently been applied to wireless positioning [5]–[15], mainly in the Massive MIMO arena, but also for indoor applications, for example based on WiFi. Most previous work is based on centralized processing providing point estimates of the user location. Recently, [15] proposed a distributed scheme for indoor positioning with probabilistic description and support for fusion of position information from several access points. Models based on probabilistic descriptions are far superior to the ones based on point estimates for one fundamental reason: probabilistic results contain a measure of the uncertainty in the estimate (an estimate with very high uncertainty does not provide much information), which is of great importance as it allows the model to express its uncertainty in the result based on the observations; in addition, uncertainty is the base for fusion of different estimates, which allows them to be properly weighted. Following this reasoning, in our proposed method panels provide a parameterized probability density function for the location of a certain user, which can be conveniently shared and fused in different panels, tree processing nodes, or in the CPU, providing high accuracy using much lower interconnection bandwidth than the centralized architecture, where panels would share their estimated CSI with the CPU, and one NN would process all incoming CSI to deliver a position estimate. As we will see in Section V, this decentralized approach can achieve few hundred times reduction in the interconnection data-rate.

II. System Model

The system under consideration is graphically represented in Fig. 1. We consider a single-antenna user u [1] whose position, denoted by $\mathbf{p}^u = (x_u, y_u, z_u) \in \mathbb{R}^3$, is to be estimated. The user is transmitting a signal which is received by multiple panels forming a LIS. Each panel contains M_p antenna elements, together with radio-frequency, analog and baseband (BB) processing capabilities in order to perform down conversion of the received signal and obtain CSI. Once CSI is available locally at a panel, a machine learning algorithm based on neural networks produces a probabilistic description of the user position, denoted by $p_i(\mathbf{p}^u)$ for panel i. In other words, $p_i(\mathbf{p}^u)$ is the probability density of the user being in position $\mathbf{p}^u$. Multiple probability functions, from different panels, can be fused into a single probability density function, which can be used for further fusion down the pipeline with other panels or sensors, or to obtain a point estimate of the user location.

A. Signal Model

We consider a LIS containing a total of M active antenna elements, and divided into P square-shaped panels, each with M_p elements, such that $M_\mathrm{p} \cdot P = M$. We assume an OFDM-based transmission system, centered at carrier frequency f_c, with a bandwidth BW across which N_sc equally spaced subcarriers contain pilots for channel estimation.

The $M \times 1$ received vector at the LIS for a certain subcarrier is given by

$$\mathbf{y} = \mathbf{h}x + \mathbf{n}, \tag{1}$$

where x is the transmitted pilot signal for which we, without loss of generality, assume $x = 1$, $\mathbf{h}$ is the channel response vector, and $\mathbf{n} \sim \mathcal{CN}(0, \sigma_n^2 \mathbf{I})$ is an $M \times 1$ i.i.d. noise vector.

We model the line-of-sight (LoS) channel between the user at location $\mathbf{p}^u$ and a LIS antenna element at location $\mathbf{p}$ by the complex value [3]

$$h(\mathbf{p}, \mathbf{p}^u) = \frac{1}{d_u} \sqrt{\cos \phi(\mathbf{p}, \mathbf{p}^u)} \exp\left(-j \frac{2\pi d_u}{\lambda}\right), \tag{2}$$

where $\phi(\mathbf{p}, \mathbf{p}^u)$ is the relative orientation angle between the user antenna and the LIS antenna element at $\mathbf{p}$. When $\phi(\mathbf{p}, \mathbf{p}^u) = 0$ the LIS antenna is facing perpendicularly to the incoming wavefront. $d_u = \|\mathbf{p} - \mathbf{p}^u\|$ is the distance between the user and the antenna. λ is the wavelength at the corresponding subcarrier frequency. For our analysis we will consider more realistic channel models, concretely based on multipath propagation caused from specular reflection in walls, where a certain reflection coefficient is assumed, being denoted as α, and $0 \leq \alpha \leq 1$. The channel in this case is modeled as linear combination of individual components with respective reflection coefficients, this is $\sum \alpha_i h_i$, where α_i and h_i are the reflection coefficient and channel associated to the i-th multipath component respectively.

[1] Extension to multiple users is straightforward under the assumption that the channel responses are independently measured.

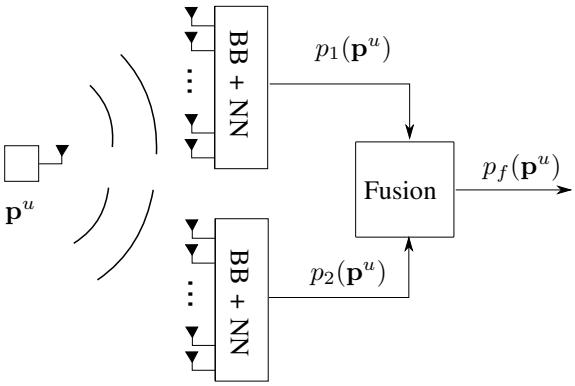


Fig. 1: System model with a single-antenna user transmitting. Two distant panels provide an individual probabilistic description of the user's location, which is fused into a single probability function.

B. User position density model

In this work panels output the inferred probability distribution of the user position, which we model as a multivariate Gaussian distribution, this is $p_i(\mathbf{p}^u) = \mathcal{N}(\mu_i, \mathbf{\Sigma}_i)$ for the i-th panel, where $\mu_i \in \mathbb{R}^3$ is the mean, and $\mathbf{\Sigma}_i \in \mathbb{R}^{3\times3}$ the covariance. The reason to select the distribution as Gaussian is twofold: the distribution is represented exclusively by the tuple $\{\mu_i, \mathbf{\Sigma}_i\}$, requiring only a reduced number of values to be shared with the fusion module, and the fusion process becomes relatively simple, as further described in Section IV.

III. Positioning via Neural Networks

As presented in the Introduction, NN have recently been used for user positioning in wireless systems. It provides a low complexity approach for inference, as an alternative to CSI-based fingerprinting stored in a data-base.

A. Feature extraction

In our analysis we model the CSI estimate as $\mathcal{CN}(\mathbf{h}, \sigma_n^2 \mathbf{I})$, and it is used during training and inference [2]. These user-specific complex-valued CSI obtained at pilot subcarriers are separated into real and imaginary parts and stacked together in a feature vector $\widehat{\mathbf{h}}$.

B. Neural Network with probability

The NN architecture is illustrated in Fig. 2. After the channel is estimated and the feature vector ($\widehat{\mathbf{h}}$) formed, four dense layers are used (three with ReLU activation functions and one with linear outputs). The output of the last dense layer is the probability distribution parameters $\{\mu, \mathbf{\Sigma}\}$. These parameters completely represent the distribution, and can be used during inference for point estimate (i.e. selecting the mean μ) and for fusion (see Section IV).

The last layer (dashed line in the figure) is the probabilistic layer and provides the probability density function using the

[2] We remark that we consider a noisy channel estimate in our analysis during training and inference.

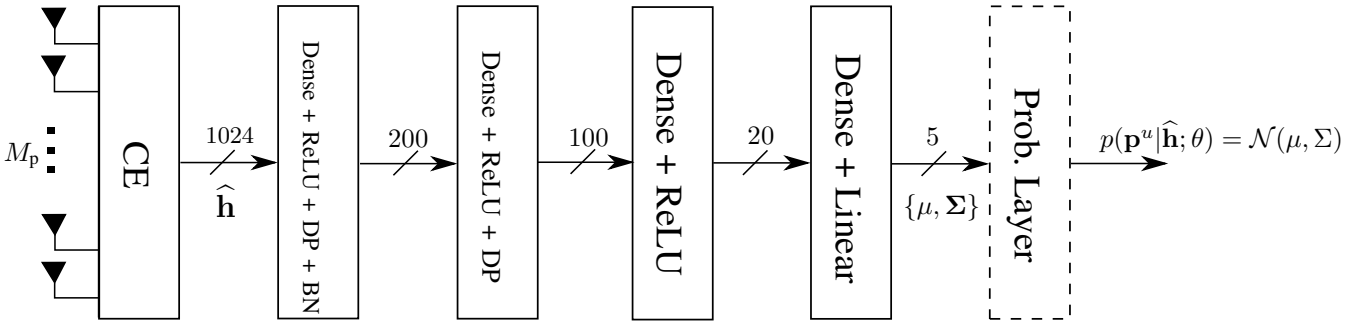


Fig. 2: Processing pipeline for positioning inference in a panel, including channel estimation block and neural network. Acronyms: CE = channel estimation, DP = dropout, BN = batch normalization. Probabilistic layer is only used during training (dashed line). Numerical values represent dimensionality (real numbers) of the data exchanged assuming $M_{\text{p}} = 64$ and $N_{\text{sc}} = 8$. 2D positioning assumed.

input parameters. This is only used during training, and is described in more detail in next subsection.

The numerical values depicted in Fig. 2 correspond to the case of users positioned on a plane (used in our analysis in Section V), therefore a 2D multivariate distribution is generated. In this particular case, the mean μ as a 2×1 vector, and the covariance $\mathbf{\Sigma}$ as a 2×2 matrix which, due to symmetry, only has three quantifying values.

C. Training Neural Networks with probability layers

The goal of training is to learn the NN parameters, here named θ, comprising weights and biases. In this section, and for clarity, we note that the NN outputs are a function of the input channel estimate $\widehat{\mathbf{h}}$ and θ, this is $\mu \equiv \mu(\widehat{\mathbf{h}}, \theta)$, and $\mathbf{\Sigma} \equiv \mathbf{\Sigma}(\widehat{\mathbf{h}}, \theta)$. As commented before, the probabilistic layer is only used during training. Its mission is to provide the corresponding probability density functions, which is required for computing the loss function.

The loss function used in this work is the Negative Log-Likelihood function, defined as

$$\text{NLL}(\theta) = -\sum_{n} \log\{p(\mathbf{p}_n^u|\widehat{\mathbf{h}}_n;\theta)\}, \tag{3}$$

where $p(\mathbf{p}^u|\widehat{\mathbf{h}}_n;\theta) = \mathcal{N}(\mu(\widehat{\mathbf{h}}_n,\theta), \mathbf{\Sigma}(\widehat{\mathbf{h}}_n,\theta))$, and $\{\widehat{\mathbf{h}}_n, \mathbf{p}_n^u\}$ is the training set made with different locations covering the area of service. For simplicity, we assume no error in the estimate of these locations. For a certain training location $\mathbf{p}_n^u$, the channel estimate is obtained $\widehat{\mathbf{h}}_n$, and the corresponding NN output for the current θ is calculated: $\{\mu(\widehat{\mathbf{h}}_n,\theta), \mathbf{\Sigma}(\widehat{\mathbf{h}}_n,\theta)\}$. Given these input parameters, the probabilistic layer provides the full probability density, which is used to compute the likelihood, by evaluating the probability density at training location $\mathbf{p}_n^u$, as $p(\mathbf{p}_n^u|\widehat{\mathbf{h}}_n;\theta)$, as shown in (3). High values of likelihood (or equivalently lower values of NLL) indicate a good fit between the parameterized distribution and the ground truth location. The sum covers usually a subset of the training set (minibatch), and gives a cost value. This cost function is minimized using the Adam optimizer [16], whose outcome is the maximum likelihood (ML) solution (which minimizes the NLL), this is $\theta_{\text{ML}} = \arg\min_\theta \text{NLL}(\theta)$, which is used for inference.

Additionally during training, and to take the effect of noise distribution into account, we take N_{rep} samples per training location in the training set. This means that for each location we sample the random variable $\widehat{\mathbf{h}}$ multiple times. This also imply to augment the size of the training set (even though the number of physical locations remains the same).

IV. Probability Fusion

The goal of probability fusion is to consolidate a finite number of probability distributions into a single one. In our case, we are interested in the fusion of probability densities provided by the panels as shown in Fig 1. Given that the individual density functions are Gaussian, Gaussian conflation represents a convenient fusion method, as it ensures the resulting distribution is also Gaussian, and leads to the classical weighted least squares method, providing the best unbiased and maximum likelihood estimators [17]. Following this method, the fused distribution is proportional to the product of the individual ones.

As mentioned before, the Gaussian conflation of P individual Gaussian distributions is also Gaussian, with covariance and mean represented respectively as

$$\mathbf{\Sigma}_f = \left(\sum_{i=1}^{P} \mathbf{\Sigma}_i^{-1}\right)^{-1}, \tag{4}$$

and

$$\mu_f = \mathbf{\Sigma}_f \left(\sum_{i=1}^{P} \mathbf{\Sigma}_i^{-1}\mu_i\right). \tag{5}$$

V. Simulation Results and Analysis

The scenario considered for our analysis is a volume of size (width×depth×height) = $(10m \times 10m \times 0.4m)$. The volume has four solid reflecting walls covering the sides (we do not consider reflection in floor and roof). Four panels of size $(0.4m \times 0.4m)$ are installed on the walls, occupying the center of each, as shown in Fig. 3a with red lines. The

Panels	0	0, 1	0, 1, 2	0, 1, 2, 3
mean error	31	4.2	2.7	3.0
std error	21	4.1	2.1	2.3

TABLE I: Mean and std error for panel 0 inference and fusion results in experiment where user is in nine locations, depicted in Fig. 3 . Units in cm.

Panels	0	0, 1	0, 1, 2	0, 1, 2, 3
mean error	15	5.7	4.3	3.5
std error	11	3.0	2.3	1.7

TABLE II: Mean and std error for panel 0 inference and fusion results in experiment with square-shaped trajectory, depicted in Fig.4. Units in cm.

users are located on the plane crossing the panels by half, this is $(x, y, 0)$, assuming panels are within $(x, y, -0.2m)$ and $(x, y, 0.2m)$. The panels are trained individually according to the method described in Subsection III-C. The system parameters are: $M_{\text{p}} = 64$, $f_{\text{c}} = 3$GHz ($\lambda = 10$cm), BW = 100MHz, $N_{\text{sc}} = 8$, $N_{\text{rep}} = 10$, and noise variance $\sigma_n^2 = 0.002$. For simplicity we only consider one specular reflection in the walls with $\alpha = 0.1$ [3]. We first consider 9 locations for inference (red dots). In Fig. 3a it is shown the result of inference of panel 0 (bottom) as 2 std ellipses (black) and the respective means (blue dots). We observe that the angular accuracy is quite good for all positions, while the distance accuracy gets worse for points further away from the panel, as they lay outside of the near-field region and less information about the distance is contained in observed, the increasingly planar, wave front. Error values are shown in Table I, where error is measured between the distribution mean (μ) and the ground truth. Figure 3b shows the result of the fusion between panel 0 (bottom) and 1 (left), with an important improvement in accuracy as both panels complement each other [4]. Results of extended fusion process is shown in Fig. 3c and 3d, with incremental improvements in the accuracy.

Fig. 4 shows the result of another experiment, where we analyze 100 locations for a square-shaped trajectory. For some of them, 2 std and mean results of inference from panel 0 are shown in Fig. 4a, while 4b shows only the mean values of all locations. Error results are shown in Table II. Result of fusion between panel 0 and 1 is shown in Fig. 4c and 4d for 2 std and mean, where we observe an important improvement in accuracy, similar to the observed in previous experiment. Results of the fusion of panels 0, 1 and 2 are shown in Fig. 4e and Fig. 4f. The result of fusion of all panels is shown in Fig. 4g and 4h.

From interconnection bandwidth point of view, there is a significant reduction in exchange of data compared to the centralized approach. Each panel shares 5 values (in case of 2D), instead of 1024 required by $\widehat{\mathbf{h}}$, which is a 200x reduction.

VI. Conclusions

In the preset work we have introduced a novel method for wireless positioning in distributed Large Intelligent Surfaces using neural network with probabilistic layer. Each panel forming the LIS provides a probabilistic description of the user location based on the local channel estimate, that can later be fused to a single probability distribution comprising information from more/all panels. By choosing a parameterized probability distribution, as the Gaussian, only the parameters need to be inferred, considerably reducing the interconnection bandwidth with the fusion module or CPU. Our analysis show that by fusion of two panels is enough to achieve fraction of wavelength accuracy level in a scenario with users distributed over a $100\lambda \times 100\lambda$ area.

[3]Extension to more advanced channel models is left for future work.

[4]The uncertainty in the distance shown by panel 0 is compensated by the high accuracy in the angle from panel 1 and vice versa.

VII. Acknowledgment

This work was supported by ELLIIT, the Excellence Center at Linköping-Lund in Information Technology.

References

[1] C. De Lima, D. Belot, R. Berkvens, A. Bourdoux, D. Dardari, M. Guillaud, M. Isomursu, E.-S. Lohan, Y. Miao, A. N. Barreto, M. R. K. Aziz, J. Saloranta, T. Sanguanpuak, H. Sarieddeen, G. Seco-Granados, J. Suutala, T. Svensson, M. Valkama, B. Van Liempd, and H. Wymeersch, "Convergent communication, sensing and localization in 6G systems: An overview of technologies, opportunities and challenges," *IEEE Access*, vol. 9, pp. 26 902–26 925, 2021.

[2] S. Hu, F. Rusek, and O. Edfors, "Beyond Massive MIMO: The Potential of Data Transmission With Large Intelligent Surfaces," *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2746–2758, May 2018.

[3] S. Hu, F. Rusek, and O. Edfors, "Beyond Massive MIMO: The Potential of Positioning With Large Intelligent Surfaces," *IEEE Transactions on Signal Processing*, vol. 66, no. 7, pp. 1761–1774, April 2018.

[4] J. Rodriguez Sanchez, F. Rusek, O. Edfors, and L. Liu, "Distributed and Scalable Uplink Processing for LIS: Algorithm, Architecture, and Design Trade-offs," *arXiv e-prints*, Dec. 2020. [Online]. Available: http://arxiv.org/abs/2012.05296

[5] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, "Deep convolutional neural networks for massive MIMO fingerprint-based positioning," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2017, pp. 1–6.

[6] S. D. Bast, A. P. Guevara, and S. Pollin, "CSI-based positioning in massive mimo systems using convolutional neural networks," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–5.

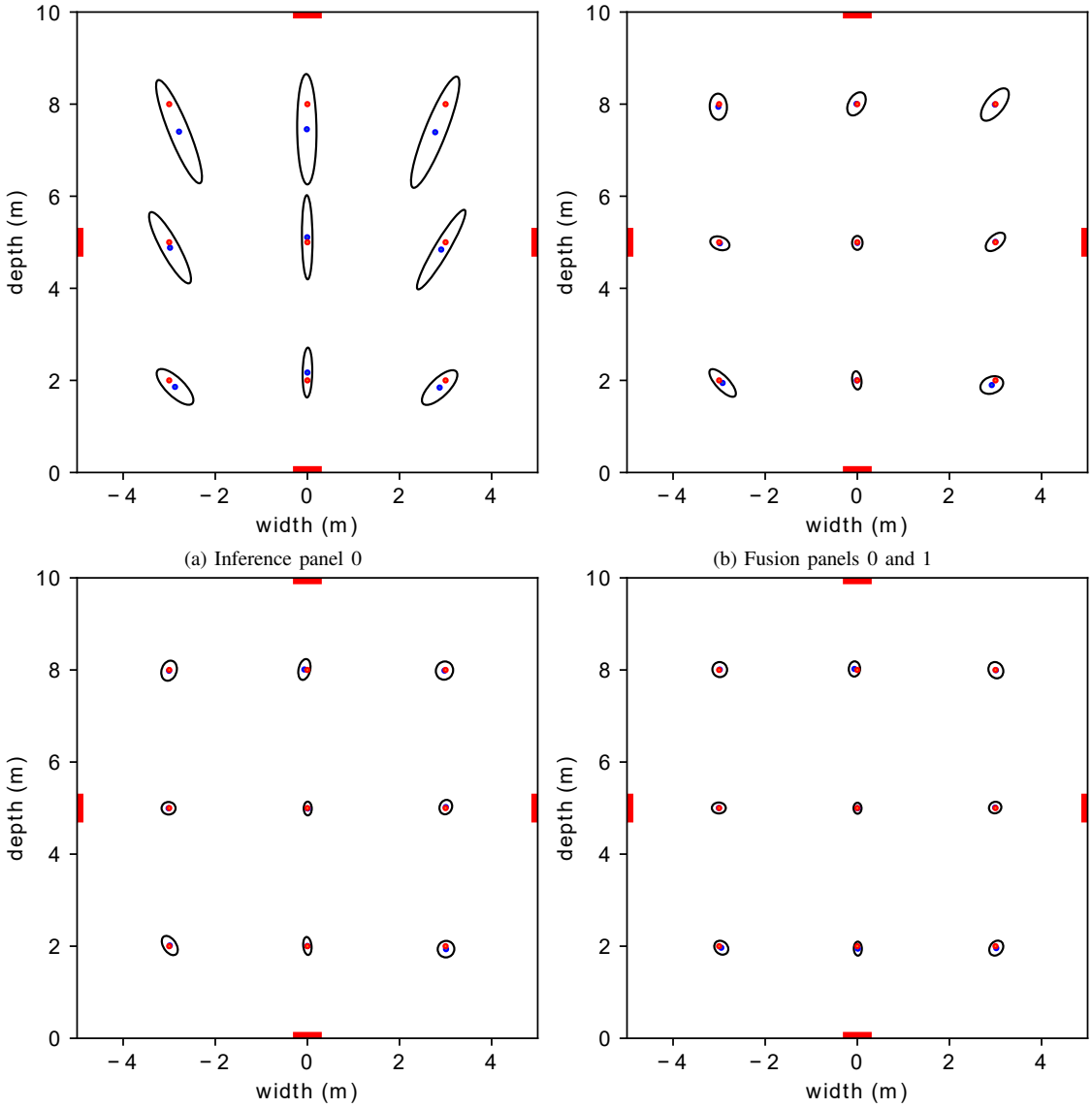
[7] K. N. R. S. V. Prasad, E. Hossain, and V. K. Bhargava, "Machine learning methods for RSS-based user positioning in distributed massive MIMO," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 8402–8417, 2018.

[8] J. Fan, S. Chen, X. Luo, Y. Zhang, and G. Y. Li, "A machine learning approach for hierarchical localization based on multipath MIMO fingerprints," *IEEE Communications Letters*, vol. 23, no. 10, pp. 1765–1768, 2019.

[9] S. De Bast and S. Pollin, "MaMIMO CSI-based positioning using CNNs: Peeking inside the black box," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–6.

[10] M. Arnold, S. Dorner, S. Cammerer, and S. Ten Brink, "On deep learning-based massive MIMO indoor user localization," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2018, pp. 1–5.

[11] G. Cerar, A. vigelj, M. Mohori, C. Fortuna, and T. Javornik, "Improving CSI-based massive MIMO indoor positioning using convolutional neural network," in *2021 Joint European Conference on Networks and Communications 6G Summit (EuCNC/6G Summit)*, 2021, pp. 276–281.



(a) Inference panel 0

(b) Fusion panels 0 and 1

(c) Fusion panels 0, 1, and 2

(d) Fusion panels 0, 1, 2, and 3

Fig. 3: Results of inference on 9 user locations. Top view of the scenario. Red lines represent panels, and red points denote ground truth locations. Blue points denote mean of distribution, and black 2 std ellipse. Fig. 3a represents results from panel 0. Fig. 3b represents the result of the fusion of panels 0 and 1. Fig. 3c represents the result of the fusion of panels 0, 1 and 2, and Fig. 3d shows the result of the fusion of all four panels.

[12] B. Berruet, O. Baala, A. Caminada, and V. Guillet, "Delfin: A deep learning based CSI fingerprinting indoor localization in IoT context," in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2018, pp. 1–8.

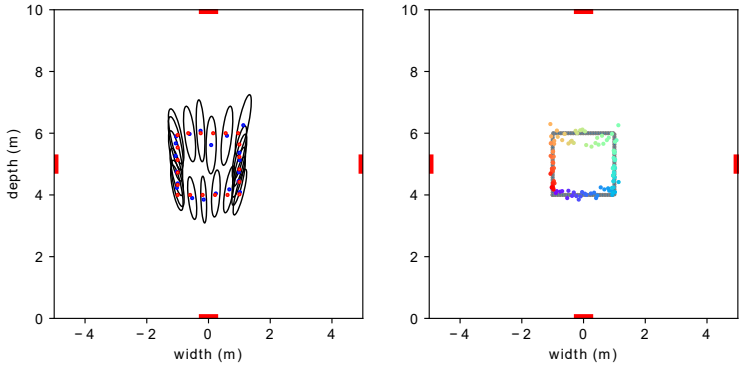
[13] P. Ferrand, A. Decurninge, and M. Guillaud, "DNN-based localization from channel estimates: Feature design and experimental results," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

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

[15] E. Gönültaş, E. Lei, J. Langerman, H. Huang, and C. Studer, "CSI-based multi-antenna and multi-point indoor positioning using probability fusion," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2021.

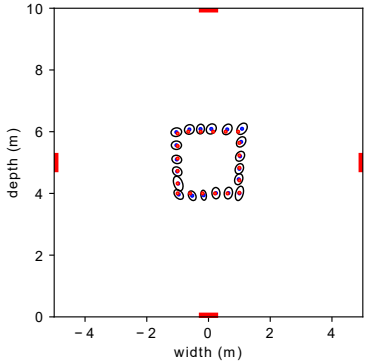
[16] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[17] T. P. Hill, "Conflations of Probability Distributions," *Transactions of the American Mathematical Society*, vol. 363, no. 06, pp. 3351–3351, aug 2008. [Online]. Available: http://arxiv.org/abs/0808.1808

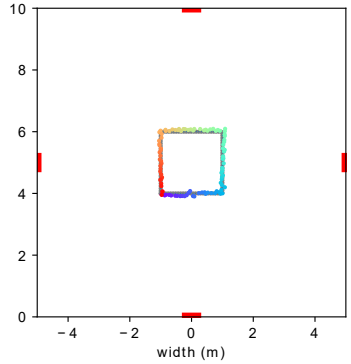


(a) Inference panel 0

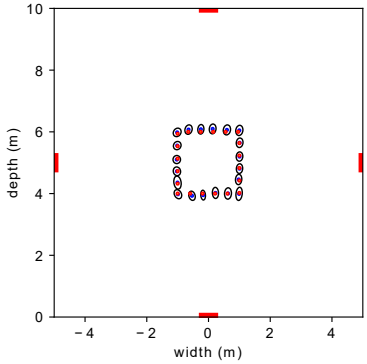
(b) Mean (μ) panel 0



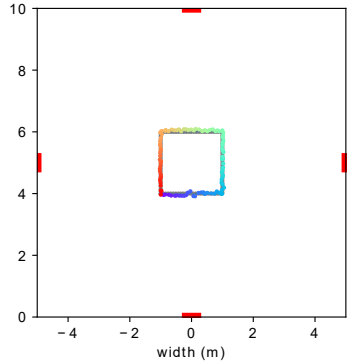
(c) Fusion panels 0 and 1



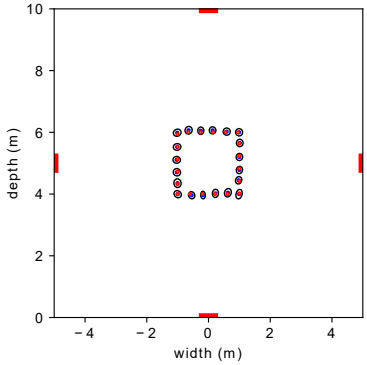
(d) Mean (μ) fusion panels 0 and 1



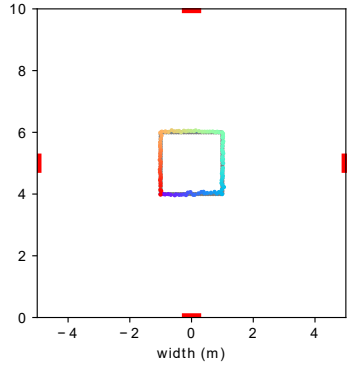
(e) Fusion panels 0, 1 and 2



(f) Mean (μ) fusion panels 0, 1 and 2



(g) Fusion panels 0, 1, 2 and 3



(h) Mean (μ) fusion panels 0, 1, 2 and 3

Fig. 4: Results of inference on 100 user locations forming a square-shaped trajectory. Top view of the scenario. Left column: Red points denote ground truth. Blue points denote mean of inferred distribution, and black 2 std ellipse. Fig. 4a represents results from panel 0 (only 20 out of 100 locations are shown for convenience). Figures 4c, 4e, and 4g represent different results of fusion with different panels. Right column: Figures 4b, 4d, 4f, and 4h show the mean. Colors used are only intended to ease the visual association between estimated and true locations, given the high number of points.