User Tracking with Multipath Assisted Positioning-based Fingerprinting and Deep Learning

Markus Ulmschneider and Christian Gentner
German Aerospace Center (DLR), Institute of Communications and Navigation
Muenchener Str. 20, 82334 Wessling, Germany
{markus.ulmschneider,christian.gentner}@dlr.de

***Abstract*—Multipath assisted positioning schemes allow localizing a user with only a single physical transmitter by treating multipath components (MPCs) as line-of-sight signals from virtual transmitters. The user position and the locations of the physical and virtual transmitters can be estimated jointly with simultaneous localization and mapping (SLAM). While such approaches often show very good positioning performance, they come at the cost of a high computational complexity. To reduce this complexity, multipath assisted positioning schemes based on SLAM may be combined with fingerprinting, where the fingerprints are features of the wireless radio channel. Within this paper, we present such an approach, where a deep neural network (DNN) is trained on data from a multipath assisted positioning scheme to predict the user position and the corresponding uncertainty from channel information. Based on the DNN, a Kalman filter can accurately and efficiently track the user position. We show by simulations that the positioning performance is improved by a factor of 1.5 while the computational complexity is crucially lower than that of multipath assisted positioning-based SLAM.**

***Index Terms*—cooperative Channel-SLAM, deep learning, fingerprinting, localization, simultaneous localization and mapping**

I. Introduction

In multipath assisted positioning, multipath propagation is exploited for localization by regarding multipath components (MPCs) as line-of-sight (LoS) signals from *virtual transmitters*. The locations of the physical transmitter(s) and the virtual transmitters are typically unknown, but can be estimated jointly with the receiver position with simultaneous localization and mapping (SLAM) [1]–[3]. We have previously introduced a multipath assisted positioning scheme called cooperative Channel-SLAM in [4], [5], where users cooperate by sharing information on the transmitter locations. A physical transmitter could be a wireless local area network (WLAN) router or a fifth generation (5G) base station, for example. In the following, the term user may refer to either a user or the radio receiver carried by the user depending on the context. The term transmitter is a general term including both physical and virtual transmitters.

Cooperative Channel-SLAM shows good localization performance, but suffers from a high computational complexity due to complex signal processing methods. While Channel-SLAM is a model-based approach, fingerprinting schemes [6], [7] are data driven. Such schemes consist of two stages.

In the offline stage, fingerprints are taken at known locations and stored in a database together with these locations. In wireless radio positioning, fingerprints are often received signal strength indicator (RSSI) values or channel state information (CSI). In the online stage, fingerprints are taken and matched against the database for localization. Being data driven, a big advantage of fingerprinting methods is their simplicity due to the lack of possibly complex models. However, it suffers from two major disadvantages. On the one hand, a third-party localization system is needed in the offline stage. On the other hand, the database needs to be updated whenever there are changes in the environment.

We have proposed a new fingerprinting-like approach named DNN-CC-SLAM in [8], where fingerprints are time of arrival (ToA) estimates of signal components impinging at a receiver, and cooperative Channel-SLAM is performed in the offline stage to create the fingerprint database. A deep neural network (DNN) is trained with the ToA and position estimates obtained from cooperative Channel-SLAM to predict a position based on channel information. In the online stage, a position estimate is obtained by evaluating the DNN with ToA estimates. Within this paper, by ToA we mean the propagation time of a signal from the transmitter to the receiver.

With cooperative Channel-SLAM being used in the offline stage, no third party reference system is needed, as the fingerprinting database is built up with SLAM. Likewise, the fingerprint database can be updated with cooperative Channel-SLAM to react to changes in the environment. In both cases, no third-party reference system is needed.

Within this paper, we built upon our approach in [8] and specifically work on the architecture of the DNN. The DNN in [8] estimates only a position and does not include any uncertainty measure on the estimated positions. In addition, no temporal correlations regarding the user trajectories are taken into account, leading to a snapshot-based estimator. In this paper, we expand [8] by a DNN architecture that learns to predict not only a single position, but a Gaussian distribution. The output of the neural network is then fed as measurement into a Kalman filter that tracks the user position over time. Since the measurement in the Kalman filter is a position, the filter can be implemented very efficiently. In this way, both temporal correlations and an uncertainty measure are included in the user position estimate. In particular, the positioning performance improves considerably.

The remainder of this paper is organized as follows. Section II introduces cooperative Channel-SLAM and fingerprint-

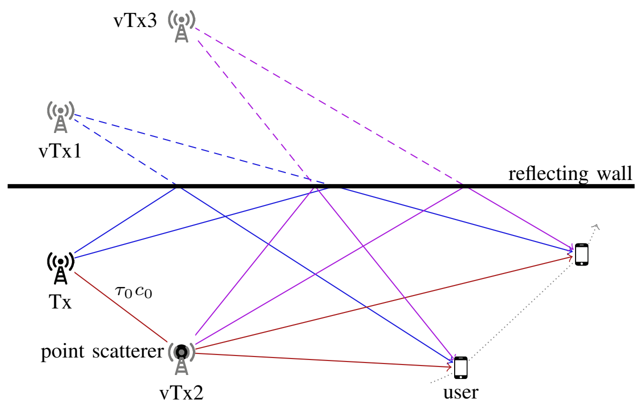

Fig. 1. The physical transmitter Tx emits a radio signal. The idea of multipath assisted positioning is to treat each MPC as a LoS signal from a virtual transmitter. The three different virtual transmitters vTx1, vTx2 and vTx3 are plotted for three different propagation paths.

ing schemes. In Section III, we present DNN-CC-SLAM and our new network architecture. Evaluations based on simulations follow in Section IV, before Section V concludes the paper.

II. Cooperative Channel-SLAM and Fingerprinting

A. Cooperative Channel-SLAM

Fig. 1 illustrates the idea behind multipath assisted positioning. The transmit signal from the physical transmitter Tx arrives at the user via different propagation paths interacting with a planar surface represented by the wall and a point scatterer. Each MPC arriving at the user is regarded as a LoS signal from a virtual transmitter. From Fig. 1, we make the following observations.

First, the location of the virtual transmitter vTx1 corresponding to the reflection of the signal at the wall is at the location of Tx mirrored at the wall. The corresponding propagation path is drawn blue. Tx and vTx1 are perfectly time synchronized. Second, the location of the virtual transmitter vTx2 corresponding to scattering is at the point scatterer. The corresponding propagation path is drawn red. There is a delay offset τ_0 between the Tx and vTx2, corresponding to the Euclidean distance $\tau_0 c_0$ between the two. Such a delay offset can and will be interpreted as a clock offset in the following. Third, single interactions of the transmit signal with planar surfaces and point scatterers can be extended to multiple interactions in a straightforward manner. For example, the propagation path corresponding to vTx3 is drawn purple and involves both scattering at the point scatterer and a reflection at the wall. Fourth, the virtual transmitter locations are independent from the user position.

Channel-SLAM is a multipath assisted positioning approach that considers a static scenario. Within this paper, we restrict ourselves to a single physical transmitter for clarity, while the extension to multiple physical transmitters is straightforward. We assume a linear multipath channel which is time variant due to the movement of the user who is equipped with an antenna array.

The channel is modeled as a linear superposition of signal components arriving at the user. The signal components of the transmit signal $s(t)$ correspond to different propagation paths. The signal component with index j is characterized by a ToA $\tau_j(t)$ and a complex amplitude $a_j(t)$. The received signal at the user is expressed at one antenna element at time t as

$$r(\tau, t) = \sum_j a_j(t) s\left(\tau - \tau_j(t)\right) + n(\tau, t), \qquad (1)$$

where the random process $n(\tau, t)$ incorporates both additive white Gaussian noise (AWGN) and dense multipath components (DMCs) [9].

Channel-SLAM works in two steps. In the first step, a channel estimator tracks the parameters of signal components over time. Within the scope of this paper, we use the Kalman Enhanced Super Resolution Tracking (KEST) estimator for channel estimation. KEST does not only track these parameters with parallel Kalman filters yielding a data association over time, but also estimates the number of signal components, i.e., the model order, at each time step. We assume that the parameter estimates are uncorrelated among the different signal components. If such correlations occur, they affect the estimates only for short time spans but not in the long run.

A subset of the channel parameter estimates, typically ToA and angle of arrival (AoA), are used in the second step of Channel-SLAM, where a Rao-Blackwellized particle filter jointly tracks the user position and velocity and estimates the states of transmitters. We denote this subset at time instant k by $\boldsymbol{z}_k$.

At time instant k, the user position is denoted by $\boldsymbol{p}_{\mathrm{u},k}$ and its velocity by $\boldsymbol{v}_{\mathrm{u},k}$, leading to the user state $\boldsymbol{x}_{\mathrm{u},k} = [\boldsymbol{p}_{\mathrm{u},k}^T \;\; \boldsymbol{v}_{\mathrm{u},k}^T]^T$. The state of the the j^{th} transmitter at time instant k is denoted by $\boldsymbol{x}_{\mathrm{TX},j}^{<k>}$ and comprises the position $\boldsymbol{p}_{\mathrm{TX},j}^{<k>}$ and clock offset $\tau_{0,j}^{<k>}$. The full state vector at time instant k is thus given by

$$\boldsymbol{x}_k = \left[\boldsymbol{x}_{\mathrm{u},k}{}^T \;\; \boldsymbol{x}_{\mathrm{TX},k}{}^T\right]^T = \left[\boldsymbol{x}_{\mathrm{u},k}{}^T \;\; \boldsymbol{x}_{\mathrm{TX},k}^{<1>\,T} \;\; \ldots \;\; \boldsymbol{x}_{\mathrm{TX},k}^{<N_{\mathrm{TX},k}>\,T}\right]^T, \qquad (2)$$

where $N_{\mathrm{TX},k}$ is the number of transmitters at time instant k.

Assuming uncorrelated estimates for the signal components, the single transmitter states can be estimated independently from each other, reducing the estimation problem complexity crucially. Each signal component at one time instant is regarded as the LoS signal from one transmitter. Hence, the model order estimated by KEST corresponds to the number of transmitters. The Kalman filters in KEST inherently yield a data association for the transmitters over time.

The probability density function (PDF) for the overall state from time instants zero to k given the history of measurements

$z_{1:k}$ and control inputs $u_{1:k}$ from time instants one to k is

$$
\begin{aligned}
\mathrm{p}\left(\boldsymbol{x}_{0:k}|\boldsymbol{z}_{1:k},\boldsymbol{u}_{1:k}\right) &= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k},\boldsymbol{x}_{\mathrm{u},0:k}|\boldsymbol{z}_{1:k},\boldsymbol{u}_{1:k}\right)\\
&= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}|\boldsymbol{z}_{1:k},\boldsymbol{u}_{1:k}\right)\\
&\quad\times \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}|\boldsymbol{z}_{1:k},\boldsymbol{x}_{\mathrm{u},0:k}\right)\\
&= \mathrm{p}\left(\boldsymbol{x}_{\mathrm{u},0:k}|\boldsymbol{z}_{1:k},\boldsymbol{u}_{1:k}\right)\\
&\quad\times \prod_{j=1}^{N_{\mathrm{TX},k}} \mathrm{p}\left(\boldsymbol{x}_{\mathrm{TX},0:k}^{<j>}|\boldsymbol{x}_{\mathrm{u},0:k},\boldsymbol{z}_{1:k}\right),
\end{aligned}
\tag{3}
$$

where we exploit our assumption on the uncorrelated transmitters in the last step. Eq. (3) also yields the separation of the user state from the transmitters' states and thus the structure of the Rao-Blackwellized particle filter.

The Channel-SLAM algorithm outlined above is targeted for single users. In scenarios such as malls, stations or public buildings, we expect multiple users on different trajectories in the same area. In such settings, information regarding the states of transmitters can be exchanged among users. In cooperative Channel-SLAM, users exchange and improve *maps* of estimated transmitter states. Such prior information can drastically improve the positioning performance and the computational complexity. In general, we cannot assume that users know the transformation parameters relating the coordinate system of their local map of estimated transmitter states with the coordinate systems of maps of other users. Estimating such transformation parameters, i.e., rotation and translation, relating two maps as well as the correspondences among transmitters in these maps is denoted by the term map matching. Map matching is a crucial element of cooperative Channel-SLAM. A comprehensive overview of cooperative Channel-SLAM can be found in [5] and details in [4].

B. Fingerprinting Schemes for Localization

Fingerprinting schemes suffer from two major drawbacks. First, a precise localization is typically required in the offline stage. To achieve that, third-party positioning systems are often deployed. In addition, fingerprints need to be taken at all locations where a user needs to be located in the online stage with a rather fine grid. In general, this implies a very high effort. Second, changes in the environment reduce the positioning performance drastically. Accordingly, the database needs to be updated very often, making standard fingerprinting schemes practically infeasible.

However, various methods have been proposed to alleviate these disadvantages. For example, the authors of [10] try to update the fingerprints in the online stage to adapt to environmental changes. Other authors apply methods from machine learning, such as k-nearest neighbours (kNN) [11], support vector machine (SVM) and DNNs [12]. Using a DNN, the idea is to train the network in a supervised manner to predict the location of a user based on collected fingerprints in the offline stage. In the online stage, a position estimate is obtained by evaluating the DNN given a fingerprint at the current location. An important strength of DNNs are their good generalization capabilities.

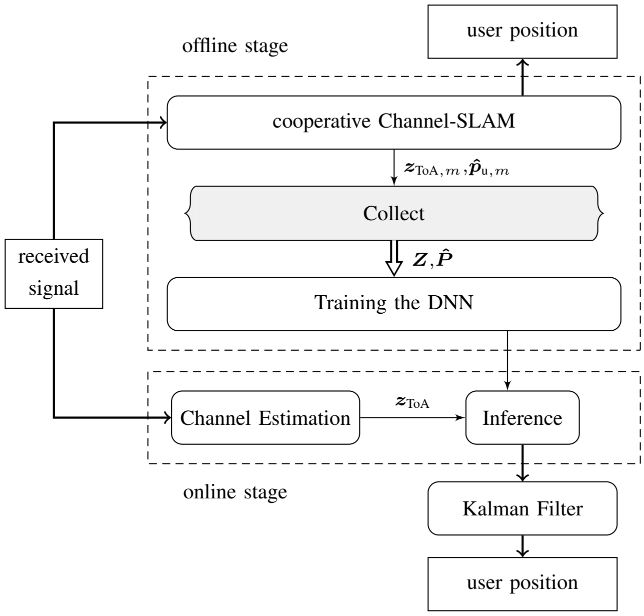

Fig. 2. The flow chart of DNN-CC-SLAM with the offline stage, the online stage and the Kalman Filter.

III. Multipath Assisted Positioning-based Fingerprinting

While cooperative Channel-SLAM shows a good positioning performance even with only a single physical transmitter, its computational complexity is very high due to the channel estimator, the particle filter and map matching. In order to drastically reduce the computational complexity, we have proposed a fingerprinting-like scheme based on cooperative Channel-SLAM named DNN-CC-SLAM. A flowchart DNN-CC-SLAM is illustrated in Fig. 2. In the offline stage, users localize themselves in the scenario with cooperative Channel-SLAM. The ToA and the position estimates for all users at all time instants are collected and stacked in the matrices $\boldsymbol{Z}$ and $\hat{\boldsymbol{P}}$, respectively. Each row of $\boldsymbol{Z}$ contains the ToA estimates for the corresponding position estimate which is in the same row in $\hat{\boldsymbol{P}}$. In Fig. 2, $\boldsymbol{z}_{\mathrm{ToA},m}$ and $\hat{\boldsymbol{p}}_{\mathrm{u},m}$ denote the set of ToAs and position estimates of the m^{th} user. Since the number of detected signal components may be different for different user positions, the rows of $\boldsymbol{Z}$ are padded with zeros.

Once a sufficient amount of fingerprints has been collected, a DNN is trained to learn the user position with $\boldsymbol{Z}$ and $\hat{\boldsymbol{P}}$ as training data. In [8], the output of the DNN is a two-dimensional position without any uncertainty information. Within this paper, we design a new mixture density network (MDN) following [13] which is trained to learn the parameters of a unimodal, bivariate normal distribution instead.

In the online stage, a channel estimator provides again a set $\boldsymbol{z}_{\mathrm{ToA}}$ of ToA estimates at each position. Since no AoA information and no parameter tracking is necessary in the online stage, efficient channel estimators such as superfast line spectral estimation (SFLSE) [14] may be used instead of KEST, leading to a crucially lower complexity. The resulting estimates are propagated through the DNN in the Inference block to obtain the mean and covariance matrix estimates, $\boldsymbol{\mu}_k$ and $\boldsymbol{C}_k$, respectively, for the current position of the user at time instant k. These estimates are then passed to a Kalman, in which the state vector consists of the user position and velocity. At time instant k, the state vector is expressed analogue to the Channel-SLAM case as $\boldsymbol{x}_{\mathrm{u},k} = [\boldsymbol{p}_{\mathrm{u},k}^T \;\; \boldsymbol{v}_{\mathrm{u},k}^T]^T$. The mean $\boldsymbol{\mu}_k$ from the DNN is considered the measurement at time instant k, and $\boldsymbol{C}_k$ the covariance matrix of the measurement noise. With the Kalman filter, temporal correlations regarding the user position are taken into account, leading to a significantly improved positioning performance.

The number of nodes in the input layer of the DNN corresponds to the number of columns of $\boldsymbol{Z}$. The DNN has six fully connected hidden layers which have 1000, 300, 200, 100, 50 and 50 neurons. The output layer consists of five nodes which are related to the user position. These are the position mean $\boldsymbol{\mu}_k$ in two dimensions, the two entries on the diagonal of the 2×2 covariance matrix $\boldsymbol{C}_k$, and the off-diagonal element of $\boldsymbol{C}_k$. The two diagonal entries in $\boldsymbol{C}_k$ are restricted by a rectified linear unit (ReLU) activation function to be non-negative.

The activation function of all other nodes in the DNN is the ReLU function as well. We train the DNN with the Adam optimizer [15]. Following [13], the loss function in the MDN is the negative log-likelihood function of the PDF of the normal distribution $f(\boldsymbol{\mu}, \boldsymbol{C})$, where $\boldsymbol{\mu}$ is the mean, corresponding to the user position, and $\boldsymbol{C}$ the covariance matrix. The negative log-likelihood is expressed for one sample $\boldsymbol{y}_s$ as

$$\begin{aligned} &-\log\left(f(\boldsymbol{y}_s|\boldsymbol{\mu},\boldsymbol{C})\right) = \\ &\quad -\frac{1}{2}\left(\log|\boldsymbol{C}| + \frac{1}{2}(\boldsymbol{y}_s-\boldsymbol{\mu})'\boldsymbol{C}^{-1}(\boldsymbol{y}_s-\boldsymbol{\mu}) + N\log 2\pi\right), \end{aligned} \tag{4}$$

where N is the dimension of $\boldsymbol{\mu}$. With the output of the DNN and the loss function in Eq. (4), the network is trained to learn the user position mean and covariance matrix of a Gaussian distribution from the channel information, specifically from the ToAs of signal components.

Since the DNN directly measures the user position $\boldsymbol{p}_{\mathrm{u},k}$, i.e., a part of the state vector in the Kalman filter, a standard Kalman filter can be applied with a simple linear observation model. The inversion of the innovation matrix in the Kalman filter is performed only on a 2×2 matrix, making the entire filter very efficient. The system model of the user may be implemented as a random walk model.

IV. Simulations

To evaluate our approach, we have performed simulations in an indoor scenario. A top view of the two-dimensional scenario, an indoor mall, is depicted in Fig. 3. There is only one physical transmitter in the scenario, and it is represented by the red triangle labeled Tx. The black lines are walls reflecting the transmit signal in a specular manner, and the black circles represent point scatterers such as pillars or similar structures. The walls and scatterers are the basis for calculating a channel impulse response (CIR) with ray-tracing and accordingly the received signal for the simulations.

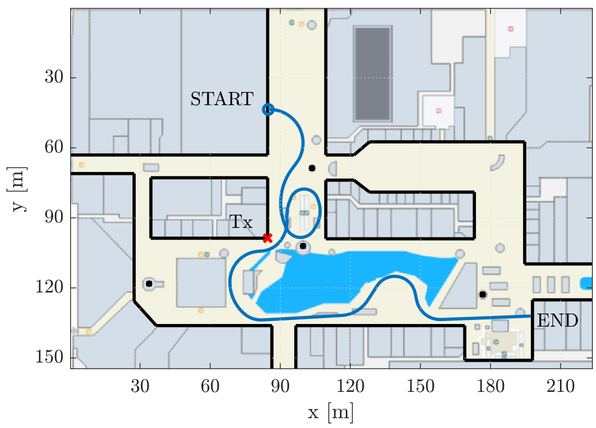

Fig. 3. Top view of the indoor mall serving as simulation scenario. The red cross labeled Tx represents the only physical transmitter in the scenario. The thick black lines represent walls and the black circles point scatterers. The reference track is depicted in blue.

The physical transmitter constantly transmits a signal with a bandwidth of $100\,\mathrm{MHz}$ at a center frequency of $1.9\,\mathrm{GHz}$. The power spectral density is constant over the entire bandwidth. The transmit signal and its parameters are known to all users.

In cooperative Channel-SLAM, i.e., in the offline stage of DNN-CC-SLAM, the user takes a snapshot of the received signal every $100\,\mathrm{ms}$ as input to the channel estimator KEST. Each user is equipped with an antenna array consisting of nine elements, which are arranged in a uniform 3×3 grid. Furthermore, an inertial measurement unit (IMU) that is rigidly mounted to the receiver is used in the particle filter in cooperative Channel-SLAM. However, only turn rates and no acceleration measurements are used from the IMU.

In the online stage of DNN-CC-SLAM, the update rate is chosen to be $10\,\mathrm{Hz}$ as for cooperative Channel-SLAM. However, the receivers are not required to have multiple antennas, and accordingly no AoA information used. In addition, no IMU information is used either.

We define a reference user walking along the track depicted in Fig. 3, and refer to that track as reference track. The length of the reference track is $311.2\,\mathrm{m}$. In the following, we will analyze the positioning performance of the reference user with different positioning methods.

For evaluating cooperative Channel-SLAM, we regard the positioning performance of the reference user who receives a transmitter map to which 21 different users have contributed. The overall traveled distance of these 21 users in the same scenario is $3.8\,\mathrm{km}$. The mean absolute error (MAE) of the reference user with cooperative Channel-SLAM is plotted blue in Fig. 4. It stays in the order of approx. $4\,\mathrm{m}$ for the most part of the track. Only towards the end, it increases to some extent due to an unfavorable geometrical dilution of precision (GDoP) situation. The error averaged over the entire reference track is $4.2\,\mathrm{m}$. This and all other results in Fig. 4 are averaged

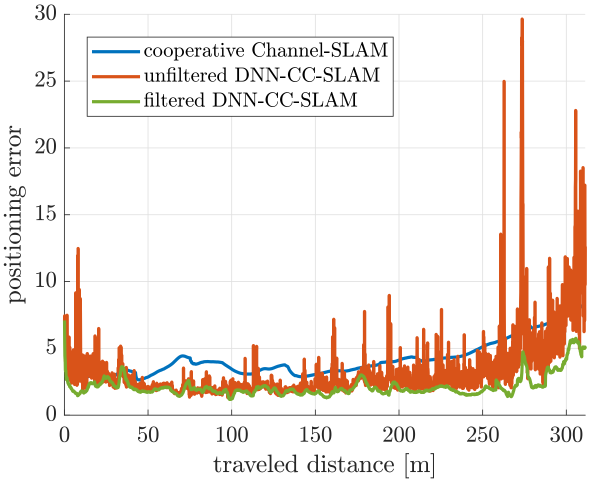

Fig. 4. Positioning error of the reference user with cooperative Channel-SLAM in blue, unfiltered DNN-CC-SLAM in red and DNN-CC-SLAM followed by a Kalman filter in green.

over 100 runs due to the stochastic nature of the particle filter and the optimization method in the DNN.

To evaluate DNN-CC-SLAM, we simulated another $5.6\,\mathrm{km}$ of synthetic trajectories with cooperative Channel-SLAM corresponding to further 23 users. The ToAs and positions of these overall 44 different users with a total traveled distance of $8.9\,\mathrm{km}$ are used as training data as input to the DNN.

We refer to the positioning performance of DNN-CC-SLAM with our new MDN architecture before Kalman filtering as *unfiltered DNN-CC-SLAM*. In that case, we disregard the uncertainty information in the output of the DNN and only consider the mean. For clarity, we refer to the case where we use a Kalman filter in addition to track the user as *filtered DNN-CC-SLAM*.

The red curve shows the positioning error of *unfiltered DNN-CC-SLAM*, which has an average positioning error of $4.1\,\mathrm{m}$. While being a snapshot-based approach, the performance is on average very similar to the performance of cooperative Channel-SLAM. However, approximately twice the training data is needed compared to the cooperative Channel-SLAM case for that similar performance. Finally, the blue curve denotes the positioning error of *filtered DNN-CC-SLAM* with an average error of $2.6\,\mathrm{m}$, which is an improvement by a factor of approx. 1.6 compared to cooperative Channel-SLAM and unfiltered DNN-CC-SLAM.

V. Conclusion

Within this paper, we have presented a fingerprinting approach for localization, where fingerprints and positions are estimated with cooperative Channel-SLAM. Hence, no external positioning system needs to be installed in the offline phase. The fingerprints are ToA estimates from a channel estimator. Updating the fingerprinting database can be achieved with low effort as well by training the DNN with additional data collected by cooperative Channel-SLAM. Compared to a previous paper, the MDN structure of the DNN allows for inference of both mean and covariance of the user position. Thus, the user position can be tracked very efficiently and accurately with a Kalman filter.

Both in the offline and online phase of our approach, only one single physical transmitter is required. Compared to cooperative Channel-SLAM, the computational complexity is crucially smaller while the positioning performance is improved.

References

[1] K. Witrisal, P. Meissner, E. Leitinger, Y. Shen, C. Gustafson, F. Tufvesson, K. Haneda, D. Dardari, A. F. Molisch, A. Conti, and M. Z. Win, “High-Accuracy Localization for Assisted Living - 5G Systems will turn Multipath Channels from Foe to Friend,” *IEEE Signal Process. Mag.*, vol. 33, no. 2, pp. 59–70, Mar. 2016.

[2] H. Wymeersch, N. Garcia, H. Kim, G. Seco-Granados, S. Kim, F. Wen, and M. Fröhle, “5G mm Wave Downlink Vehicular Positioning,” in *IEEE Global Communications Conference (GLOBECOM)*, Dec. 2018, pp. 206–212.

[3] R. Mendrzik, F. Meyer, G. Bauch, and M. Win, “Localization, Mapping, and Synchronization in 5G Millimeter Wave Massive MIMO Systems,” in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2019, pp. 1–5.

[4] M. Ulmschneider, “Cooperative Multipath Assisted Positioning,” Ph.D. dissertation, Hamburg University of Technology, 2021.

[5] M. Ulmschneider, C. Gentner, and A. Dammann, “Cooperative Estimation of Maps of Physical and Virtual Radio Transmitters,” in *Proceedings of the 34th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2021)*, Sep. 2021.

[6] 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, 2016.

[7] A. Khalajmehrabadi, N. Gatsis, and D. Akopian, “Modern WLAN Fingerprinting Indoor Positioning Methods and Deployment Challenges,” *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1974–2002, 2017.

[8] M. Ulmschneider, C. Gentner, and A. Dammann, “Learning-Based Fusion of Multipath Assisted Positioning and Fingerprinting,” in *Proceedings of 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022)*, Sep. 2022.

[9] A. Richter, “Estimation of Radio Channel Parameters: Models and Algorithms,” Ph.D. dissertation, Technische Universität Ilmenau, 2005.

[10] X. Liu, J. Cen, Y. Zhan, and C. Tang, “An Adaptive Fingerprint Database Updating Method for Room Localization,” *IEEE Access*, vol. 7, pp. 42 626–42 638, 2019.

[11] Z. Liu, X. Luo, and T. He, “Indoor Positioning System Based on the Improved W-KNN Algorithm,” in *IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, March 2017, pp. 1355–1359.

[12] L. Xiao, A. Behboodi, and R. Mathar, “A Deep Learning Approach to Fingerprinting Indoor Localization Solutions,” in *27th International Telecommunication Networks and Applications Conference (ITNAC)*, Nov 2017, pp. 1–7.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer, 2006.

[14] T. L. Hansen, B. H. Fleury, and B. D. Rao, “Superfast Line Spectral Estimation,” *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2511–2526, May 2018.

[15] D. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *International Conference on Learning Representations*, 2015.