# Global Scale Self-Supervised Channel Charting with Sensor Fusion

Omid Esrafilian[(1)]*, Mohsen Ahadi[(1)]*, Florian Kaltenberger[(1)(2)], David Gesbert[(1)]
[(1)]Communication Systems Department, EURECOM, Sophia Antipolis, France
[(2)]Institute for the Wireless Internet of Things, Northeastern University, Boston, USA
Emails: {omid.esrafilian, mohsen.ahadi, florian.kaltenberger, david.gesbert}@eurecom.fr

***Abstract*—The accurate sensing and positioning capabilities foreseen in 6G have great potential for technology advancements in various domains, such as future smart cities and industrial use cases. Channel charting has emerged as a promising technology in recent years for radio frequency-based sensing and localization. However, the accuracy of these techniques is yet far behind the numbers envisioned in 6G. To reduce this gap, in this paper, we propose a novel channel charting technique capitalizing on the time of arrival measurements from surrounding antennas along with their locations and leveraging sensor fusion in channel charting by incorporating laser scanner data during the training phase of our algorithm. The proposed algorithm remains self-supervised during training and test phases, requiring no geometrical models or user position ground truth. Simulation results validate the achievement of a sub-meter level localization accuracy using our algorithm 90% of the time, outperforming the state-of-the-art channel charting techniques and the traditional triangulation-based approaches.**

## I. Introduction

Positioning techniques in wireless networks traditionally rely on channel parameter estimation methods like Received Signal Strength Indicator (RSSI), Time of Arrival (ToA), Time Difference of Arrival (TDoA), and Angle of Arrival/Departure (AoA/AoD) [1], [2]. These parameters help triangulate or trilaterate a device's position while Channel State Information (CSI) offers a more comprehensive approach by considering detailed channel properties and environmental effects on the signal. This method is particularly effective in complex indoor environments where dense multipath propagation and NLoS conditions are dominant. Direct CSI positioning methods are divided into two categories, supervised and unsupervised [3]. Supervised learning, particularly fingerprinting (FP) methods, relies on a pre-established database of signal characteristics like CSI or Multi Path Components (MPC) as features collected at known UE locations. This database is used to train a model that can later predict the location of a device based on its observed CSI or features, making it highly effective in environments with stable signal patterns. Unsupervised learning, on the other hand, does not require a pre-labeled dataset. Instead, it estimates a mapping directly from the collected data for an objective function. This approach is valuable in dynamic environments where it is impractical to create and maintain an extensive labeled dataset, offering the flexibility to adapt to changes in the environment over time.
Although CSI provides a comprehensive view of wireless signal propagation from the transmitter to the receiver, including the effects of environmental factors (e.g., scattering, fading, and reflection), this detailed information results in high-dimensional data, making analysis in positioning systems challenging. Channel Charting (CC), an innovative approach in wireless networking, seeks to create a map or chart of the wireless medium using CSI, enabling the precise localization and tracking of devices within complex environments [4]. CC is one of the applications of manifold learning, a non-linear dimensionality reduction technique, that plays a vital role in interpreting CSI data. Manifold learning effectively uncovers the low-dimensional structures hidden within high-dimensional CSI datasets. This refines signal propagation properties, resulting in more precise perceptions of wireless signal interactions with their surroundings. The primary objective in dimensionality reduction is to map data from a high-dimensional space (D) to a lower-dimensional space (d) (where $d << D$), with two key objectives: i) The mapping should maintain the proximal relationships among data points, and ii) It should effectively generalize to new and unseen data. Non-parametric CC techniques such as Multidimensional Scaling (MDS), Isometric Mapping (ISOMAP), and Principal Component Analysis (PCA) perform well in simplifying high-dimensional data while retaining critical structures [5]. MDS focuses on preserving pairwise distances, ISOMAP extends this by maintaining geodesic distances on a manifold, and PCA captures the maximum variance through orthogonal principal components. However, these methods often fail to accurately predict unseen data as they do not learn a general mapping function from the original high-dimensional space to the reduced space. Instead, they are entirely dependent on the specific dataset they are applied to, and any new data requires recalculating the entire model, making them unsuitable for prediction in dynamic environments. Conversely, parametric approaches like deep learning-based methods are adept at learning a mapping function that can be applied to novel, unseen data. This capacity to extend CC to previously unseen CSI data is a significant advantage over traditional strategies.

* Equal contributions.

Part of the work by O. Esrafilian was funded via HUAWEI France supported Chair on Future Wireless Networks at EURECOM. In addition, part of this work has been supported by the "France 2030" investment program through the projects 5G-OPERA and GEO-5G.

Despite the recent advancements in self-supervised CC using deep metric learning in [6]–[18] (Explained in section II), they still struggle to match the precision achieved by supervised or traditional triangulation methods even in line-of-sight (LoS) (See section V). Thus, in this paper, we developed a novel CC algorithm, leveraging neural network and data fusion to accurately localize the user. Specifically, our contributions are:

- A neural network-based CC function to accurately localize users while preserving global geometry.
- We enhanced localization accuracy using data fusion with depth data only during the training phase.
- Our algorithm is self-supervised, utilizing nearby antenna a.k.a Transmission Reception Point (TRP) locations, CSI, and depth data during training without requiring labels.
- Our method achieves a sub-meter localization accuracy using two LoS TRPs in 90% of the time, superior to the state-of-the-art and traditional triangulation methods.

## II. Related work

Channel charting for localization in wireless networks has been used for the first time in [6] from a single base station (BS) with multiple antennas, and in [7], [8] from multiple massive MIMO BSs in space. Since CC relies on dimensionality reduction of the CSI, [9] and [10] used autoencoders to improve this task. A Siamese neural network in [11], [14] is proposed that takes random pairs of CSI to first learn a local channel chart and then transform it to the global form using a subset of labeled data as reference points in a semi-supervised manner. In this method, the Euclidean distances of the Channel Impulse Response (CIR) measurements are used as a dissimilarity metric. To overcome the limitations of the Siamese loss function with a Euclidean distance metric, a triplet-based loss is used in [12], [13] to learn the similarity between triplets of CSI data based on the distance of other side-informations such as the relative recording timestamps. Authors of [15], [16] combined CC with the classical localization approaches, taking ToA and AoA measurements to improve the global channel chart. Although the CSI measurements can contain rich information, none of these CC studies exploiting only CSI data have surpassed the performance of traditional triangulation-based methods, even when LoS conditions are present. In [17], velocity estimation and topological map data are used for the global transformation of the CC. However, the global consistency of this algorithm relies on the length of the trajectory taken by the user. Also, the map-matching algorithm in this study works only if a unique match of the channel chart exists in the map. Finally in [18], by proposing a loss function containing a bilateration loss including multiple BSs with known locations and a triplet loss, a self-supervised CC is made in real-world coordinates. Motivated by this, in this paper, we leverage sensor fusion and the location of the nearby TRPs in the training phase to enhance localization accuracy.

## III. System Model

In this paper, we consider two phases of training and testing. For both phases we assume an indoor scenario where a mobile user, a.k.a UE, transmits Up Link Sounding Reference Signals (UL-SRS) to a base station[1] with $M$ distributed TRPs with fixed and known locations $\mathbf{x}_m \in \mathbb{R}^3, \forall m \in [1, M]$ placed in the environment. The UE follows a trajectory with a duration of $N$ time steps. The UE location at each time step $n$ is assumed unknown and denoted by $\mathbf{u}_n \in \mathbb{R}^3, n \in [1, N]$. The height of the UE is presumed to be fixed during its trajectory. The UE and TRPs are equipped with single antennas operating in Orthogonal Frequency-Division Multiplexing (OFDM) transmission mode with total $C$ sub-carriers. The estimated CSI of the link between the $m$-th TRP and the UE location $\mathbf{u}_n$ (yet assumed unknown) at time step $n$ over all sub-carriers is denoted by $\mathbf{h}_{m,n} \in \mathbb{C}^C$. We obtain the CIR vector $\mathbf{w}_{m,n} \in \mathbb{C}^C$ by applying an inverse discrete Fourier transform on each CSI vector. Consequently, we denote $\mathbf{W}_n \in \mathbb{C}^{M \times C}$ as the CIR matrix which is the concatenation of CIR vectors over all TRPs at time step $n$. During the training phase, the UE is equipped with a 2D laser scanner to collect depth measurements in addition to CSI, while during the test time, the algorithm only uses the CSI measurements, and no laser scanner is required, hence reducing the sensor requirements and the complexity as well as saving the energy in the testing phase. It is worth mentioning that, the algorithm remains self-supervised during both train and test phases and the only difference is the type of measurements available at each phase. The depth measurements collected by the laser scanner at time step $n$ during the training phase are denoted by $\boldsymbol{\ell}_n = \{(r_k, \phi_k), \forall k \in [1, K]\} \in \mathbb{R}^{K \times 2}$, where $K$ is the number of points at each scan of the laser scanner, $r_k$, and $\phi_k$ are the relative distance and the angle of each scanned point, respectively, in the laser scanner body coordinate frame which we assume to be the same as the UE coordinate frame.

### A. Data Preprocessing and Feature Extraction

To increase the robustness of the algorithm, we preprocess and extract certain features from the measured CIR. Since the majority of the received power is usually concentrated in the first few taps, we only consider the first $\bar{C}$ columns of the CIR matrix. The truncated CIR is denoted by $\hat{\mathbf{W}}_n \in \mathbb{C}^{M \times \bar{C}}$. Therefore, the main input of our algorithm is computed as $\mathbf{Y}_n = |\hat{\mathbf{W}}_n| \in \mathbb{R}^{M \times \bar{C}}$, where $|.|$ is the element-wise absolute value operator. We assume that the UE and TRPs are synchronized akin to [19] using Round Trip Time (RTT). We then estimate the ToA of the LoS path between UE and TRPs at time step $n$ by detecting the largest peak among the columns of $\mathbf{Y}_n$. Equivalently, we can write:

$$\tau_{m,n} \propto \arg \max_{c \in [1, \bar{C}]} y_n^{(m)(c)}, \tag{1}$$

where $y_n^{(m)(c)}$ is the element of matrix $\mathbf{Y}_n$ at row $m$ and column $c$, and $\tau_{m,n}$ is the measured ToA corresponding to the LoS path (i.e. the largest pick of the CIR vector) between the UE and the $m$-th TRP at time step $n$. The ToA of all TRPs and the UE at time step $n$ in a vectorized form is given

[1]Our algorithm can be generalized to multiple BSs without a tight synchronization between them, as well as to multiple UE scenarios.

by $\boldsymbol{\tau}_n = (\tau_{1,n}, \cdots, \tau_{M,n}) \in \mathbb{R}^M$. Furthermore, we exploit the additional measurements provided by a 2D laser scanner, which is available only during the training phase, to estimate the UE displacement between two different time steps. To this end, we employ the well-known Iterative Closest Point (ICP) algorithm for processing the laser scanner data. Due to limited space, we omit the details of the ICP algorithm and we refer to [20] for more information. We denote the estimated UE displacement between time steps $n$ and $n'$ using ICP algorithm and utilizing laser scanner data by $\hat{T}_{n,n'}$.

### *B. Deep Channel Charting*

Given the CIR dataset, it is possible to find a mapping function $f_{\boldsymbol{\theta}} : \mathbb{R}^{M \times \bar{C}} \rightarrow \mathbb{R}^D$ that transforms the CIR matrix $\mathbf{Y}_n$ to a lower dimension $D \leq 3$ as a proxy to the locations, a.k.a pseudo-position, of the user as $\tilde{\mathbf{u}}_n = f_{\boldsymbol{\theta}}(\mathbf{Y}_n)$. Deep neural network has proven to be a good candidate for estimating the mapping function $f_{\boldsymbol{\theta}}$, as $f_{\boldsymbol{\theta}}$ is a complicated and non-linear function. Various methods have been introduced in the literature to find the mapping function using deep neural networks. These methods range from supervised to unsupervised [6]–[18]. In this paper, we build a channel chart algorithm upon the bilateration loss function akin to [18] and by capitalizing on ToA measurements and the location of the TRPs. We extend this method further by incorporating laser scanner data to improve the accuracy of localization. Note that, the TRP locations and laser scanner data are only required during the training phase. Moreover, our approach is self-unsupervised and will provide a global scale representation of the user's location in the global coordinate frame very close to the ground truth as opposed to the pseudo-position of the user. In the following section, we elaborate on our approach.

## IV. Channel Charting Using Data Fusion

In this section, we seek to learn a channel chart function $f_{\boldsymbol{\theta}}$ given a training dataset $\mathcal{D}_{\text{tr}} = \{\mathbf{Y}_n, \boldsymbol{\tau}_{\boldsymbol{n}}, \mathbf{x}_m, \boldsymbol{\ell}_n; \forall n, m\}$. We assume that the pilot signal sent by the UE is received at all TRPs. As expressed in [18], from the received CSI, when we compare the relative received powers at TRPs for a given UE, the TRPs closer to the UE tend to receive a signal with higher power under LoS conditions. Let's denote the received power at TRP $m$ from the UE at time step $n$ with $\gamma_{m,n} = 20 \log(\|\mathbf{h}_{n,m}\|_F)$, where $\|.\|_F$ is the Frobenius norm. Therefore, we can write:

$$\gamma_{m,n_c} > \gamma_{m,n_f} + \Gamma, \forall n_c, n_f \in [1, N], \tag{2}$$

where $n_c, n_f$ are time steps chosen such that the UE location at time step $n_c$ is closer to TRP $m$ than when the UE is at time step $n_f$. In other words, $n_c, n_f$ should satisfy the following:

$$\|\mathbf{x}_m - \mathbf{u}_{n_c}\| < \|\mathbf{x}_m - \mathbf{u}_{n_f}\|, \tag{3}$$

where $\|.\|$ is the Euclidean norm. The constant $\Gamma$ imposes that the received power differs at least by $\Gamma$. From (3), we can constitute the following bilateration loss function:

$$\mathcal{L}^b_{m,n_c,n_f} = \max(\|\mathbf{x}_m - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_c})\| - \|\mathbf{x}_m - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_f})\| + d, 0), \tag{4}$$

where $d > 0$ indicates that the UE at estimated location $f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_c})$ is closer to TRP $m$ than the UE estimated location $f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_f})$ by at least $d$ meters. Assuming that the TRPs are LoS to the UE at all time steps, finding a $f_{\boldsymbol{\theta}}$ that minimizes (4) will ideally satisfy inequality (2). However, in a realistic scenario, there is no guarantee that the power-distance relation holds, as also indicated in [18]. Moreover, the loss in (4) might be equal to zero for a vast majority of $(n_c, n_f)$ pairs, depending on the value chosen for $d$, rendering it sample inefficient for learning $f_{\boldsymbol{\theta}}$. To tackle this problem, since we can estimate the ToA at each TRP, therefore we can have an estimate of $d$ per measurement at each time step. Consequently, we can reformulate the loss function in (4) as follows

$$\mathcal{L}^b_{m,n_c,n_f} = (\|\mathbf{x}_m - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_c})\| - \|\mathbf{x}_m - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_f})\| + d_{m,n_c,n_f})^2, \tag{5}$$

with

$$d_{m,n_c,n_f} = |\tau_{m,n_f} - \tau_{m,n_c}|\, \nu, \tag{6}$$

where $\nu$ is the speed of light, and $|.|$ represents the absolute value operator. Please note that, the loss function in (5) is specified per each measurement pair and is not sparse, resulting in a more sample-efficient training process. However, the expression in (5) is a differential loss function with respect to the UE location in two time steps and can bring ambiguity to the UE location estimate, hence not preserving the global geometry features. To solve this issue, we redefine $\mathcal{L}^b_{m,n_c,n_f}$ by splitting it into two additive parts as follows:

$$\mathcal{L}^b_{m,n_c,n_f} \triangleq \mathcal{L}_{m,n_c} + \mathcal{L}_{m,n_f}, \tag{7}$$

where

$$\mathcal{L}_{m,n} = (\|\mathbf{x}_m - f_{\boldsymbol{\theta}}(\mathbf{Y}_n)\| - \tau_{m,n}\, \nu)^2. \tag{8}$$

Furthermore, We can formulate a separate loss function for the UE locations at time steps $n_c, n_f$ as follows:

$$\mathcal{L}^\ell_{n_c,n_f} = (\|f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_c}) - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_f})\| - \|\mathbf{u}_{n_c} - \mathbf{u}_{n_f}\|)^2, \tag{9}$$

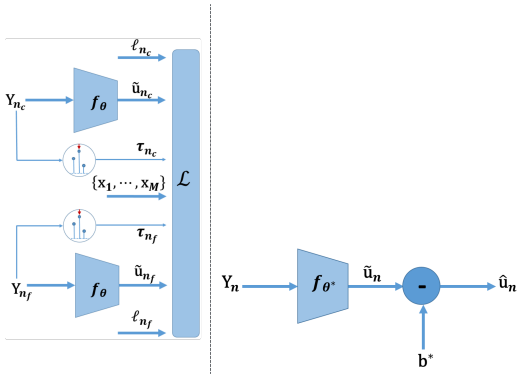
where the first part of this loss function $(\|f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_c}) - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_f})\|)$ represents the displacement between two estimated UE locations at time steps $n_c, n_f$, and the second part $(\|\mathbf{u}_{n_c} - \mathbf{u}_{n_f}\|)$ is equivalent to the true UE displacement. Therefore, minimizing (9) will result in preserving the relative displacement between the estimated UE locations in two different time steps. However, the second part of this loss function is not available. To tackle this problem, we exploit the measurements obtained from the laser scanner. We reformulate the loss function (9) by incorporating the laser scanner data as follows:

$$\mathcal{L}^\ell_{n_c,n_f} = (\|f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_c}) - f_{\boldsymbol{\theta}}(\mathbf{Y}_{n_f})\| - \hat{T}_{n_c,n_f})^2, \tag{10}$$

where $\hat{T}_{n_c,n_f}$ is the estimated UE displacement between two time steps $n_c, n_f$ using laser scanner data and the ICP algorithm [20].

Finally, the total loss function including the CIR radio measurements and the laser scanner data is given by:

$$\mathcal{L}_{m,n_c,n_f} = \mathcal{L}^b_{m,n_c,n_f} + \lambda_{n_c,n_f} \mathcal{L}^\ell_{n_c,n_f}, \tag{11}$$




(a) Training phase (b) Test phase

Fig. 1: Diagram of training and test phases of our algorithm.

where $\lambda_{n_c,n_f}$ is a coefficient that determines the impact of the loss pertaining to the laser scanner data. It is worth mentioning that the ICP algorithm might fail to estimate the displacement between two time steps if the depth measurements in two corresponding scans are very different. Therefore, to mitigate this error, we choose the value for $\lambda_{n_c,n_f}$ to be small when $n_c$ and $n_f$ are far in time.

Finally, the total loss function for all TRPs and overall time steps is given by:

$$\mathcal{L} = \sum_{\substack{n_f,n_c\in[1,N]\\ n_f\neq n_c}} \sum_{m=1}^{M} \mathcal{L}_{m,n_c,n_f}. \tag{12}$$

A neural network can then be trained to obtain a channel chart function $f_{\boldsymbol{\theta}}$ by minimizing $\mathcal{L}$ and using training dataset $\mathcal{D}_{\text{tr}}$. We denote the trained channel chart neural network model by $f_{\boldsymbol{\theta}^*}$, where $\theta^*$ is the optimized parameters of the neural network model after training. Consequently, the trained channel chart function can be used to estimate the UE location in a global scale coordinate frame owing to (7). This approach is self-supervised as it does not require labeling during the training or evaluation/test phases.

### A. Offset Estimation

The UE location estimated using the channel chart neural network model $(f_{\boldsymbol{\theta}^*})$ trained in the last section might deviate from the ground truth by an offset. This might happen when a small set of TRPs (less than 3 TRPs) is available, which can introduce an ambiguity to the UE location estimate and result in getting stuck in a local minima. To tackle this problem, given a trained channel chart model, we formulate the following optimization problem to calculate the offset in the estimation:

$$\mathbf{b}^* := \arg\min_{\mathbf{b}} \sum_{n=1}^{N} \sum_{m=1}^{M} |\hat{\tau}_{m,n}(\mathbf{b}) - \tau_{m,n}|, \tag{13}$$

where $\mathbf{b} \in \mathbb{R}^3$ is a bias vector to be estimated, and $\hat{\tau}_{m,n}(\mathbf{b})$ is the estimated ToA between the $m$-th TRP and the UE location at time step $n$ and taking into account the bias vector $\mathbf{b}$. Using the trained channel chart model, $\hat{\tau}_{m,n}(\mathbf{b})$ is given by:

$$\hat{\tau}_{m,n}(\mathbf{b}) = \frac{\| (f_{\boldsymbol{\theta}^*}(\mathbf{Y}_n) - \mathbf{b}) - \mathbf{x}_m \|}{\nu}. \tag{14}$$

Solving (13) will find a bias vector, minimizing the error between the estimated ToA and the measured ToA, thus improving the localization accuracy. We solve (13) using the training dataset $\mathcal{D}_{\text{tr}}$ and employing the Particle Swarm Optimization (PSO) technique [21]. We denote the optimized bias vector by $\mathbf{b}^*$.

Finally, the UE location estimate at time step $n$ by taking into account the bias vector is given by

$$\hat{\mathbf{u}}_n = f_{\boldsymbol{\theta}^*}(\mathbf{Y}_n) - \mathbf{b}^*. \tag{15}$$

Note that $(\mathbf{Y}_n, \mathbf{b}^*)$ are sufficient to estimate the UE location during the test time. A diagram illustrating the overall procedure of our proposed CC algorithm during the training and test phases is shown in Fig. 1.

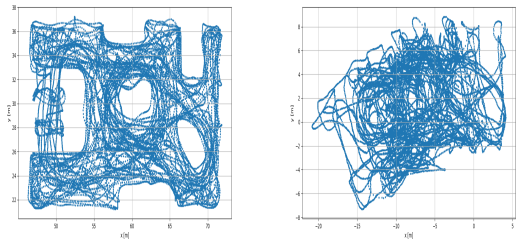
## V. Evaluations and Results

To evaluate our proposed method against existing state-of-the-art approaches, the following benchmarks were employed:

- **Classical PSO**: We compared the results with an extension of the classical TDoA-based localization algorithm in [22], where the PSO technique [21] is instead used to triangulate and estimate the UE location given the TDoA measurements from the surrounding TRPs.
- **Siamese Neural Network** [14]: Uses pairs of CSI measurements and their corresponding Euclidean distances as a dissimilarity metric, preserved in a 2-D encoded latent space. This technique is semi-supervised since the estimated latent space requires a linear transformation to the global coordinate frame using a subset of CSI data labeled with true UE locations.
- **Triplet-based Neural Network** [12]: It encodes triplets of CSI into a 2-D latent space and similar to Siamese, is semi-supervised and utilizes some labeled data.
- **Triplets + Bilat.** [18]: Employs a self-supervised approach using known TRPs locations and their received power in a combined triplet and bilateration loss function to train a model in the global scale coordinate frame.

The performance of the CC models is assessed through both quantitative and qualitative methods. Quantitative key metrics include Continuity (CT) (ensuring spatial relationships are preserved in lower-dimensional space), and Trustworthiness (TW) (ensuring charted similarities reflect true proximities) as detailed in [23]. In addition, the CDF of the localization error at the 90th percentile (CE90), indicates localization accuracy and the spread of errors across locations. Qualitatively, evaluating the charted space through visual inspections against expected wireless environment geometries helps verify if the model accurately identifies meaningful patterns and relationships within the data.

In our simulations, we consider an indoor factory scenario where there is a UE transmitting UL-SRS signals with



(a) Ground truth (b) Laser scanner

Fig. 2: True trajectory (a) vs. the estimated trajectory using only the laser scanner data (b), during the training phase.

100MHz bandwidth and 122.88 MSamples/sec sampling rate in sub-6 GHz frequency to $M = 2$ LoS TRPs with a fixed height of 8 m. For the sake of comparison, the training and testing trajectories are similar to the datasets from [14] and IPIN competition 2023 Track 7, both recorded in the Fraunhofer IIS L.I.N.K. hall, where the measurements are taken in an approximately $20 \times 15\,\mathrm{m}^2$ area. Nevertheless, we reconstructed the CIR data from the above trajectories in Matlab using a 3D map-based Ray-Tracing toolbox to simulate the laser data in the same environment. In Fig. 2a, we depicted the actual trajectory followed during the training phase, and Fig. 2b shows the reconstructed training trajectory using only the laser scanner data incrementally and using the ICP algorithm [20]. It's important to note that, the reconstructed trajectory based on laser scanner data is prone to errors due to measurement drift and coordinate differences as the true user locations are unknown (see Fig. 2). Consequently, this trajectory cannot be used directly, making approaches like fingerprinting infeasible. In Fig. 3a and 3f, we illustrated the true trajectories followed by the mobile UE during the testing phase. We also define the global coordinate frame as the coordinate frame used for recording ground truth trajectories. The test datasets corresponding to the test trajectories shown in Fig. 3a and 3f are denoted by Dataset 1 and Dataset 2, respectively. We choose the first 49 elements of the CIR matrix, therefore the dimension of the input to our CC network is as $\mathbf{Y}_n \in \mathbb{R}^{2\times 49}$. The height of the UE is assumed known and equal to 1.5 m and remained constant during both the training and test phases. Therefore, we estimate the UE location in the 2D space. [2] We simulated a 2D laser scanner during the training phase, which provides depth measurements with a resolution of 0.6 degree in the angular domain, and an accuracy of 5 cm for ranging. Both depth and CIR measurements are collected every 20 ms. The architecture of the encoder-based neural network used for the CC function is similar to [14] and summarized in Table I. In addition, We selected $\lambda_{n_c,n_f} = 5$ when $|n_c - n_f| \leq 500$, otherwise $\lambda_{n_c,n_f} = 0$.

In Fig. 3b to 3e the results of channel charts for test Dataset 1, and in Fig. 3g to 3j, the results related to test Dataset 2 for different benchmarks are shown. For better visualization, plots are color-coded in RGB values normalized between $[0, 1.0]$, whereby points with color values closer to 1.0 indicate more accuracy or smaller error in position estimate compared to the ground truth. It is worth mentioning that the location estimate provided by our proposed algorithm is in the global coordinate frame, very close to the ground truth. This is owing to using the ToA measurements along with the TRPs locations and the laser scanner data during the training phase.

Furthermore, the value metrics introduced in Sec. V for the test datasets and different benchmarks are compared in Table II which clearly shows that our proposed algorithm outperforms the benchmarks. Also, we compared the results with the Classical PSO benchmark which is a classical TDoA-based localization method as described in Sec. V. It is worth mentioning that, to triangulate the UE using such traditional algorithms, at least 3 TRPs are required, as opposed to 2 TRPs used in our algorithm. Despite using 3 TRPs, the localization accuracy of the classical method is inferior to our algorithm.

In Table II, we also present the results related to an additional experiment that we conducted by running our algorithm without utilizing the laser scanner data, which is equivalent to set $\lambda_{n_c,n_f} = 0, \forall n_c, n_f$. The results confirm that the incorporation of laser scanner data can significantly improve the accuracy of localization.

TABLE I: Embedding Model Architecture Summary

| Layer | Output Dimension | Kernel Size | Activation |
|---|---|---|---|
| Conv2D | (8, 2, 49) | (3, 3) | ReLU |
| Conv2D | (8, 2, 49) | (5, 5) | ReLU |
| Conv2D | (8, 2, 49) | (8, 8) | ReLU |
| Conv2D | (16, 2, 49) | (10, 10) | ReLU |
| Flatten | (1, 1568) | - | - |
| Fully Con. | (1, 200) | - | ReLU |
| Fully Con. | (1, 100) | - | - |
| Fully Con. | (1, 2) | - | - |

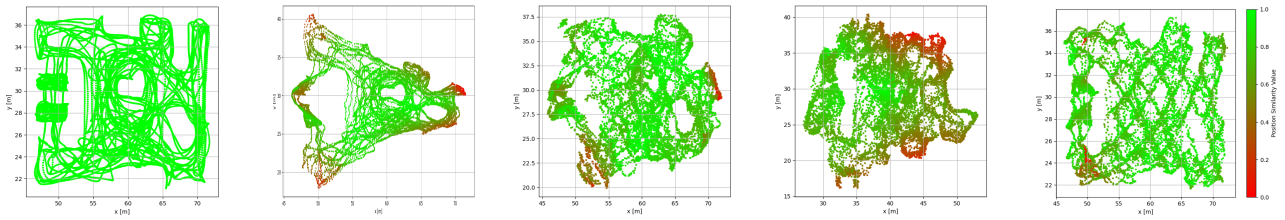
TABLE II: 2 TRPs Comparison of Our Model with State-of-the-Art over Datasets 1 and 2

| | CT | | TW | | CE90 [m] | |
|---|---|---|---|---|---|---|
| Model | 1 | 2 | 1 | 2 | 1 | 2 |
| Classical PSO | 0.987 | 0.978 | 0.986 | 0.984 | 1.59 | 1.45 |
| Siamese [14] | 0.996 | 0.994 | 0.994 | 0.991 | 3.08 | 2.24 |
| Triplets [12] | 0.993 | 0.994 | 0.992 | 0.994 | 2.29 | 1.35 |
| Triplets+Bilat. [18] | 0.991 | 0.980 | 0.990 | 0.964 | 24.14 | 22.16 |
| Ours (no Laser) | 0.995 | 0.992 | 0.995 | 0.991 | 3.98 | 3.81 |
| **Ours** | 0.998 | 0.996 | 0.997 | 0.995 | 0.94 | 0.97 |

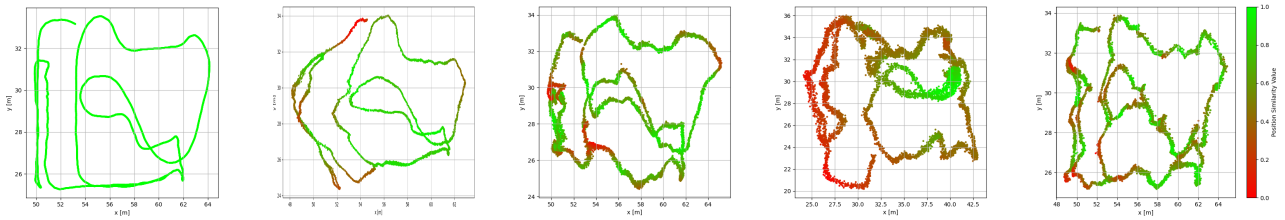
## VI. Conclusions and Future Work

In this paper, we presented a novel channel charting method that uses ToA measurements from nearby TRPs along with their locations. In addition, we leveraged sensor fusion by incorporating laser scanner data during the training phase of the algorithm. Our algorithm is self-supervised during the training

[2] Our algorithm can be generalized to 3D UE localization without limit.



(a) Ground Truth (b) Siamese [14] (c) Triplets [12] (d) Trip.+Bilat. [18] (e) Ours



(f) Ground Truth (g) Siamese [14] (h) Triplets [12] (i) Trip.+Bilat. [18] (j) Ours

Fig. 3: Figures (a) to (e) are the results for test Dataset 1, and figures (f) to (j) are the results for test Dataset 2.

and test phases, requiring no geometrical models or user position ground truth. Simulation results demonstrated that our algorithm achieves sub-meter level localization accuracy 90% of the time, surpassing the state-of-the-art channel charting techniques and the traditional triangulation-based approaches. Channel charting leveraging data fusion is still in its infancy and requires further investigation. For future research, we will extend this work by exploring NLoS scenarios as well as exploiting additional measurements, such as AoA. Moreover, $\lambda_{n_c,n_f}$ can be selected more dynamically.

## REFERENCES


[1] 3GPP, "NG Radio Access Network (NG-RAN); Stage 2 functional specification of User Equipment (UE) positioning in NG-RAN," Technical Specification 3GPP TS 38.305, 2022.

[2] M. Ahadi and F. Kaltenberger, "5GNR Indoor Positioning By Joint DL-TDoA and DL-AoD," in *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, 2023.

[3] A. Nessa, B. Adhikari, F. Hussain, and X. N. Fernando, "A Survey of Machine Learning for Indoor Positioning," *IEEE Access*, vol. 8, 2020.

[4] P. Ferrand, M. Guillaud, C. Studer, and O. Tirkkonen, "Wireless channel charting: Theory, practice, and applications," *IEEE Communications Magazine*, vol. 61, no. 6, pp. 124–130, 2023.

[5] L. Van Der Maaten, E. O. Postma, H. J. van den Herik *et al.*, "Dimensionality reduction: A comparative review," *Journal of Machine Learning Research*, vol. 10, no. 66-71, p. 13, 2009.

[6] C. Studer, S. Medjkouh, E. Gonultaş, T. Goldstein, and O. Tirkkonen, "Channel Charting: Locating Users Within the Radio Environment Using Channel State Information," *IEEE Access*, vol. 6.

[7] J. Deng, S. Medjkouh, N. Malm, O. Tirkkonen, and C. Studer, "Multipoint Channel Charting for Wireless Networks," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*.

[8] 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*.

[9] P. Huang, O. Castañeda, E. Gönültaş, S. Medjkouh, O. Tirkkonen, T. Goldstein, and C. Studer, "Improving Channel Charting with Representation-Constrained Autoencoders," *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*.

[10] P. Agostini, Z. Utkovski, S. Stańczak, A. A. Memon, B. Zafar, and M. Haardt, "Not-Too-Deep Channel Charting (N2D-CC)," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2022.

[11] E. Lei, O. Castañeda, O. Tirkkonen, T. Goldstein, and C. Studer, "Siamese Neural Networks for Wireless Positioning and Channel Charting," in *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*.

[12] P. Ferrand, A. Decurninge, L. G. Ordoñez, and M. Guillaud, "Triplet-Based Wireless Channel Charting: Architecture and Experiments," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, 2021.

[13] F. Euchner, P. Stephan, M. Gauger, S. Dörner, and S. Ten Brink, "Improving Triplet-Based Channel Charting on Distributed Massive MIMO Measurements," in *IEEE 23rd International Workshop on Signal Processing Advances in Wireless Communication (SPAWC)*, 2022.

[14] M. Stahlke, G. Yammine, T. Feigl, B. M. Eskofier, and C. Mutschler, "Indoor Localization With Robust Global Channel Charting: A Time-Distance-Based Approach," *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 1, pp. 3–17, 2023.

[15] A. Aly and E. Ayanoglu, "Estimation of Cellular Wireless User Coordinates via Channel Charting and MUSIC," in *2023 International Conference on Computing, Networking and Communications (ICNC)*.

[16] F. Euchner, P. Stephan, and S. t. Brink, "Augmenting channel charting with classical wireless source localization techniques," in *2023 57th Asilomar Conference on Signals, Systems, and Computers*, 2023.

[17] M. Stahlke, G. Yammine, T. Feigl, B. M. Eskofier, and C. Mutschler, "Velocity-Based Channel Charting with Spatial Distribution Map Matching," 2023.

[18] S. Taner, V. Palhares, and C. Studer, "Channel Charting in Real-World Coordinates," in *2023 IEEE Global Communications Conference*.

[19] R. Mundlamuri, R. Gangula, F. Kaltenberger, and R. Knopp, "Novel round trip time estimation in 5G NR," in *2024 IEEE Global Communications Conference: Wireless Communications (Globecom 2024 WC)*, Cape Town, South Africa, available as arXiv preprint arXiv:2404.19618.

[20] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, 1995.

[22] F. Gustafsson and F. Gunnarsson, "Positioning using time-difference of arrival measurements," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)*.

[23] H. AL–Tous, P. Kazemi, C. Studer, and O. Tirkkonen, "Channel Charting with Angle-Delay-Power-Profile Features and Earth-Mover Distance," in *2022 56th Asilomar Conference on Signals, Systems, and Computers*, 2022.