Semi-supervised t-SNE for Millimeter-wave Wireless Localization

Junquan Deng[1], Wei Shi, Jian Hu, Xianlong Jiao[2]

[1]Sixty-third Research Institute, National University of Defense Technology, China
[2]College of Computer Science, Chongqing University, China

CHENGDU,CHINA
ICCC

2021-12-11

Motivation

- **Mobile location information acquisition** is fundamental in building smart cites and intelligent transportation systems
- Seamless high-precision localization, especially on the network side, is a challenging problem in **NLOS** and **GPS-denied environments**
- High-resolution sensing functions are envisioned to be integrated in future **6G Radio Access Networks** (RANs), which opens up new opportunities for high-precision positioning and tracking

State of the Art

- Current network-side **5G NR positioning** techniques require rigorous calibration or accurate synchronization among network elements
- Conventional **fingerprinting** methods require a large number of **densely-sampled measurements**, which scales poorly to large areas and renders automation to dynamic environments challenging

Principle of CSI Mapping

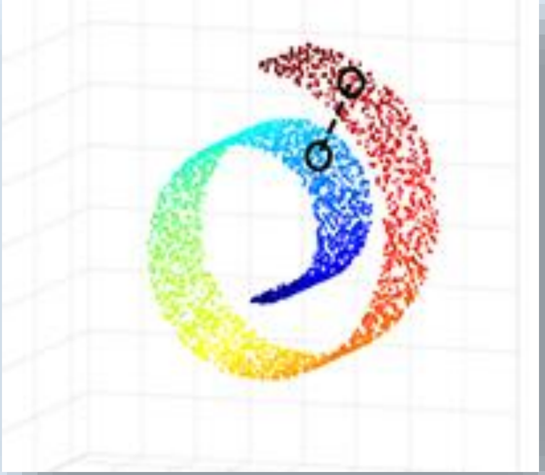
- **High-dimensional** multi-antenna Channel State Information (**CSI**) strongly dependents on UE position, which is **low-dimensional**

- **Manifold learning** methods can be applied to map CSI data to a channel chart where nearby points correspond to nearby locations in geographical space

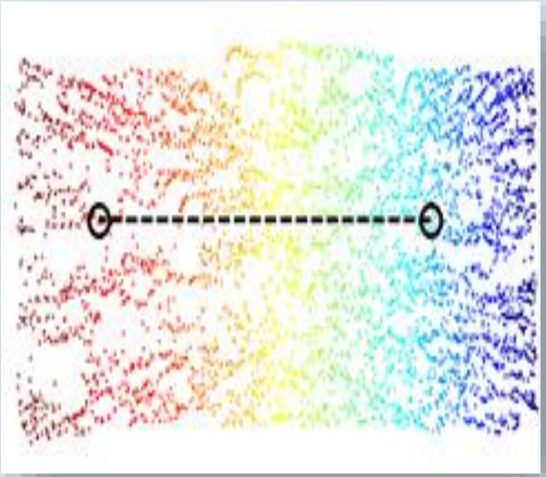
PCA
Sammon Mapping
Laplacian Eigenmaps
Autoencoder
CNN
t-SNE
…

CSI $\longrightarrow$ **x**

High-dimensional



Low-dimensional



Proposed mmWave Localization Framework

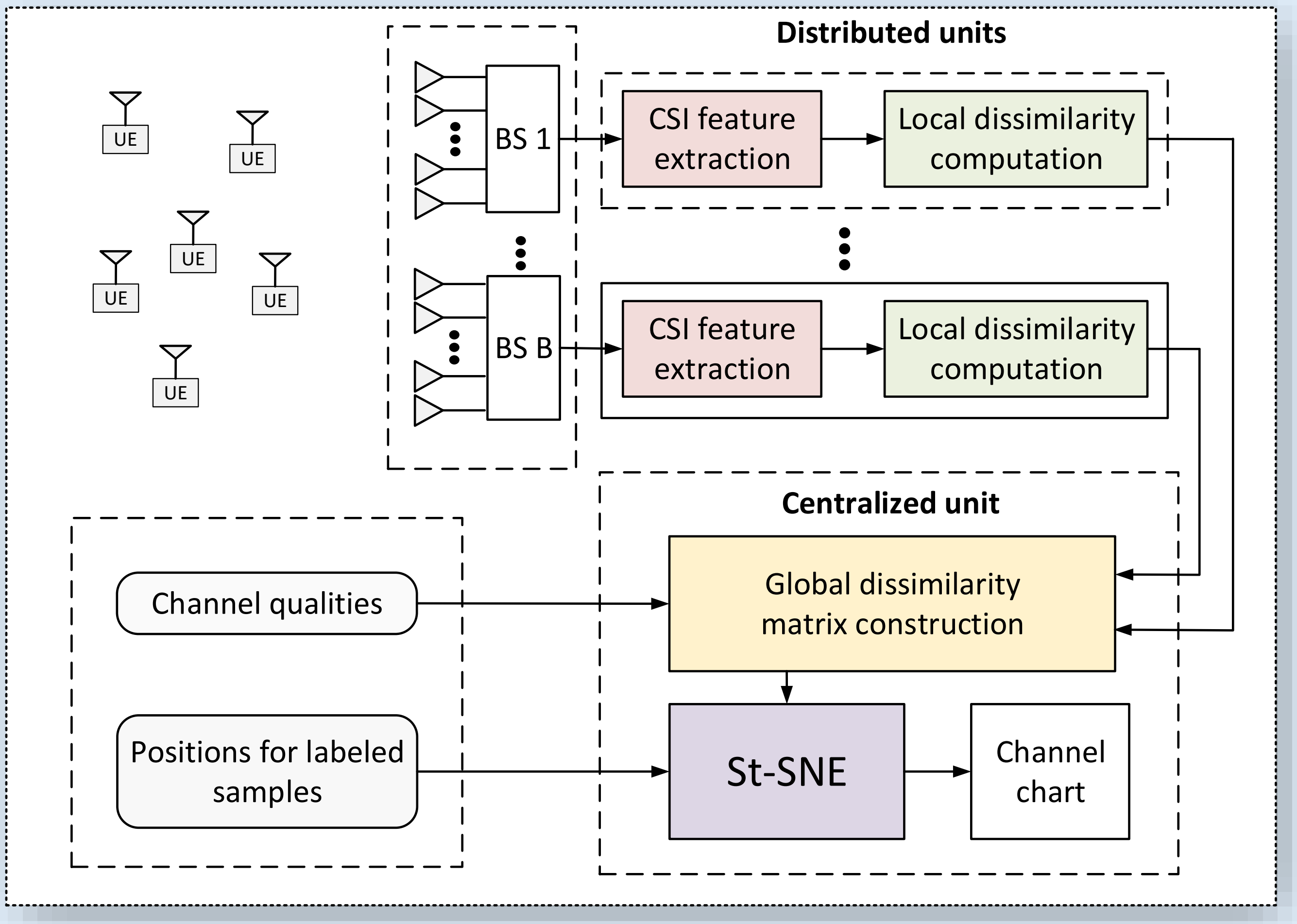
① CSI and Side-information Collection

② CSI Feature Extraction

③ Local CSI Dissimilarity Learning

④ Global CSI Dissimilarity Matrix construction

⑤ Semi-Supervised Manifold Learning



Multi-antenna CSI Sample Collection

One CSI sample:

Multi-path components

$$\mathbf{h}_{t,f} = \sum_{l=1}^{P} \alpha_{t,f}^{(l)} \mathbf{s}(\phi_t^{(l)}) + \mathbf{n}$$

$$\mathbf{C} = \mathbb{E}_f \left[\mathbf{h}_{t,f} \mathbf{h}_{t,f}^{\mathrm{H}} \right]$$

Labeled CSI data set:

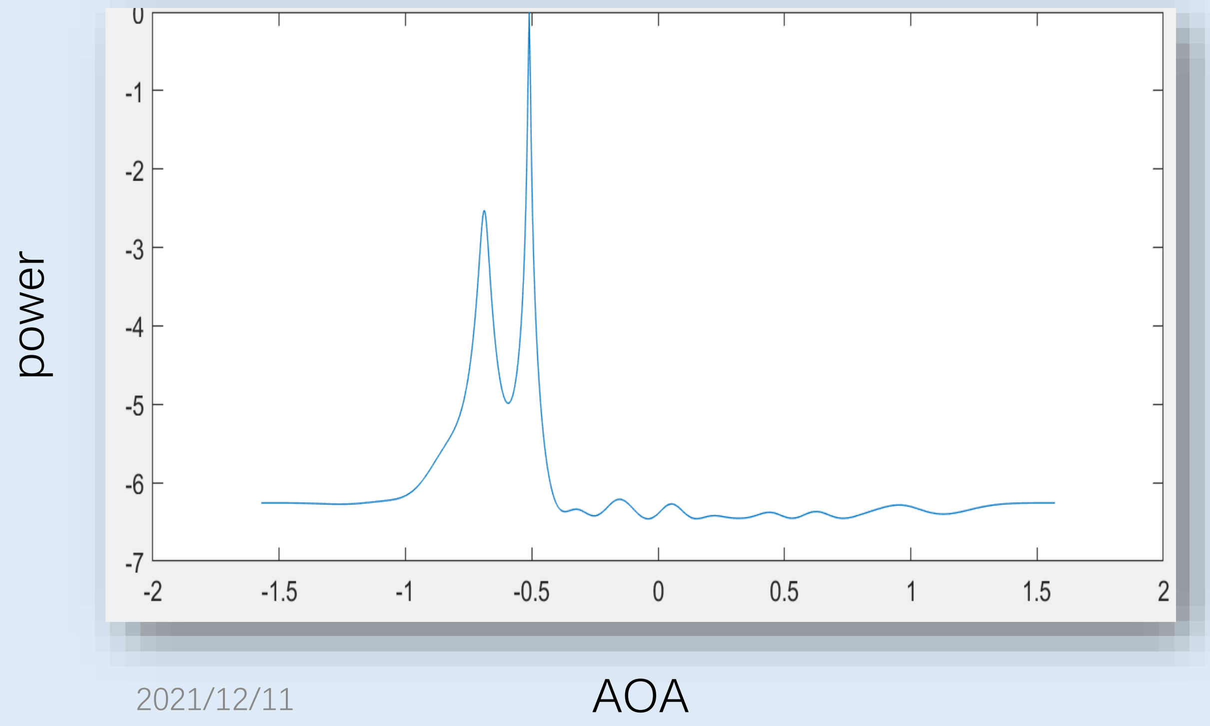
$$\mathcal{L}^{(b)} = \{\mathbf{C}_1^{(b)}, \ldots, \mathbf{C}_L^{(b)}\} \text{ with } \mathbf{P} = [\mathbf{p}_1, \ldots, \mathbf{p}_L]$$

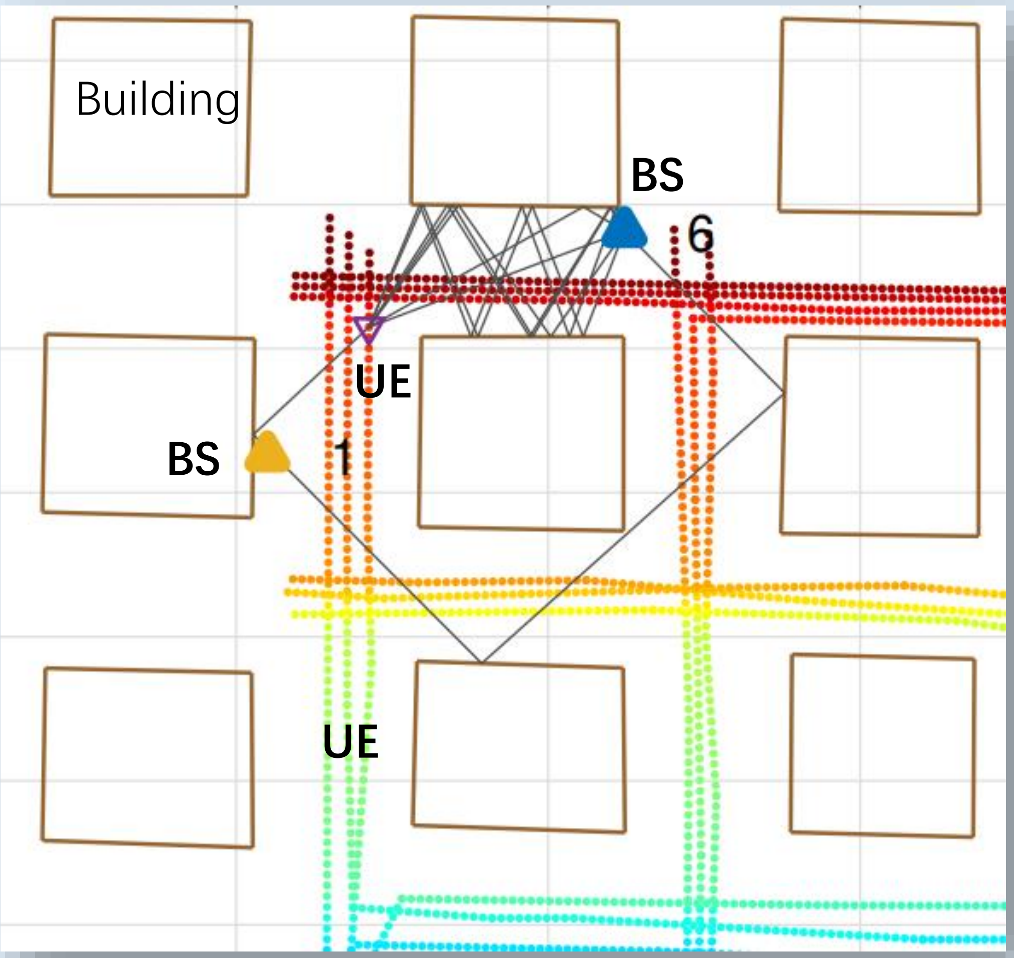
Unlabeled CSI data set:

$$\mathcal{U}^{(b)} = \{\mathbf{C}_{L+1}^{(b)}, \ldots, \mathbf{C}_{L+U}^{(b)}\}$$

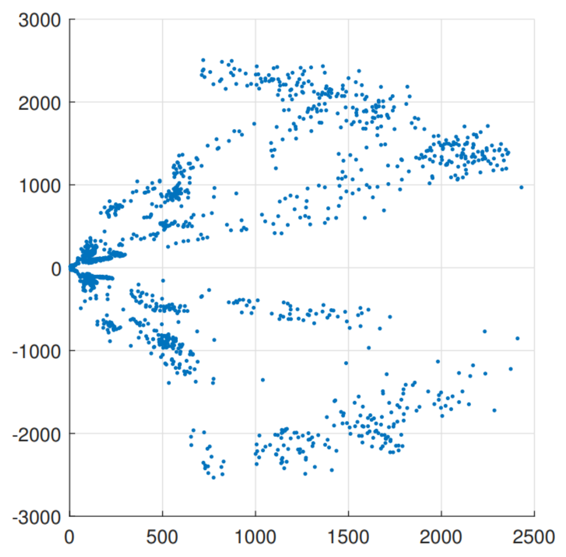
CSI Feature Extraction

- BS uses MUSIC to extract Multipath Parameters
- Power Angular Profile (**PAP**)

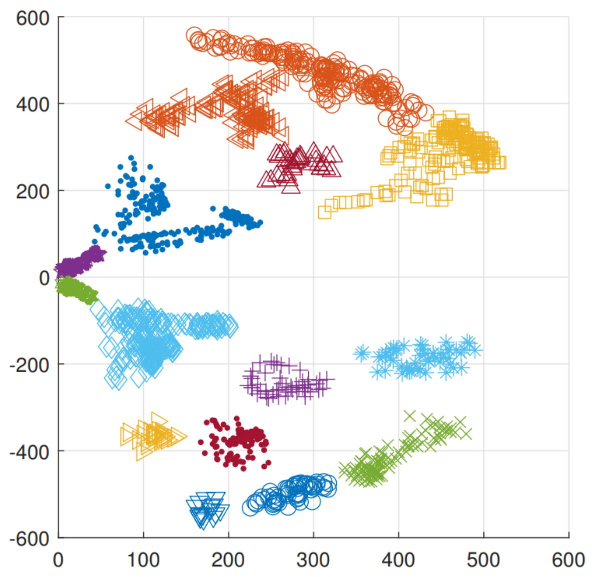




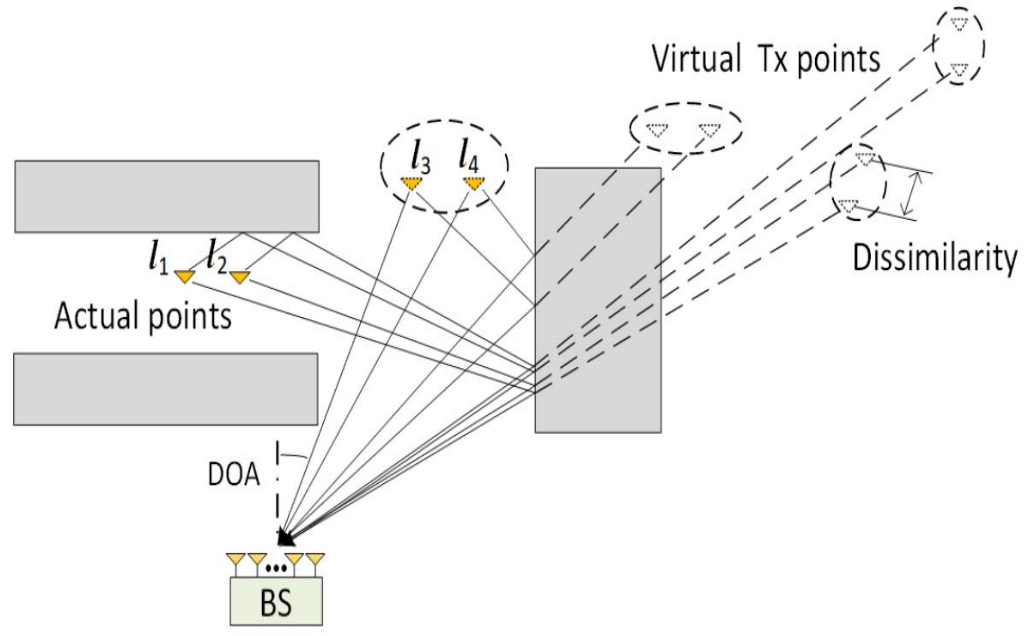
CSI Dissimilarity based on PAP



(a)



(b)



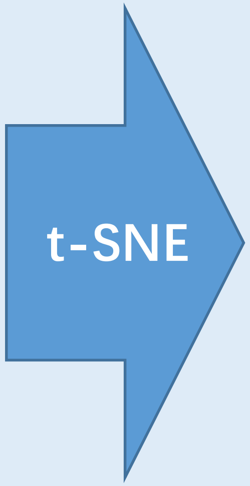
Point clouds of virtual transmitter points :
(a) Without Clustering (b) With Clustering

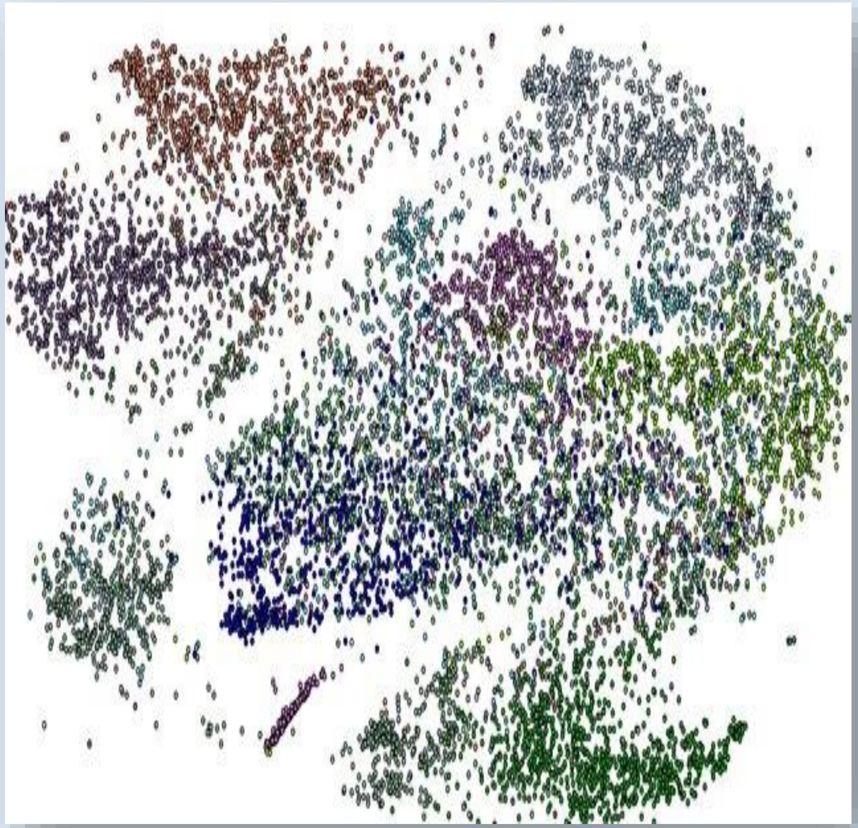
Dissimilarity metric deduced
from a virtual transmitter point cloud

t-SNE (t-distributed stochastic neighbor embedding)

Widely used for visualizing high-dimensional data

unsupervised





t-SNE Basic

N data samples
Dissimilarity matrix ***D***

Mapping

Low-dimensional representation **Z** = { z_1, . . . , z_N }

Similarity probability matrix **P**

$$p_{nm} = \frac{1}{2}(p_{m|n} + p_{n|m}) = \frac{1}{2}\left(\frac{\mathrm{e}^{-d_{mn}^2/2\sigma_n^2}}{\sum_{k\neq n}\mathrm{e}^{-d_{kn}^2/2\sigma_n^2}} + \frac{\mathrm{e}^{-d_{nm}^2/2\sigma_m^2}}{\sum_{k\neq m}\mathrm{e}^{-d_{km}^2/2\sigma_m^2}}\right)$$

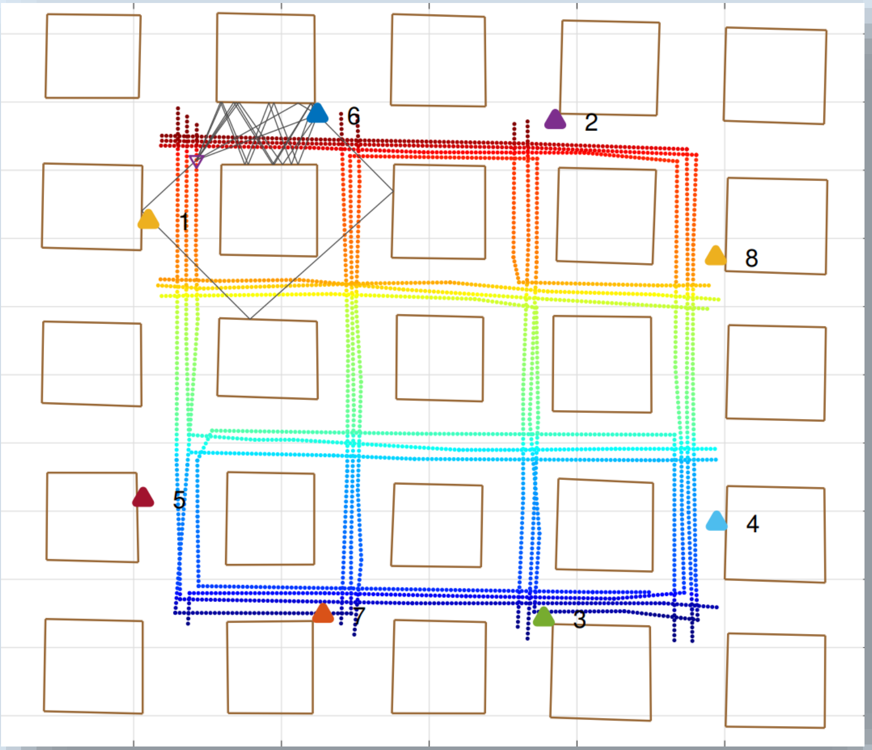
Similarity probability matrix **Q**

$$q_{nm} = \frac{(1+\|\mathbf{z}_n - \mathbf{z}_m\|_2^2)^{-1}}{\sum_{l\neq k}(1+\|\mathbf{z}_l - \mathbf{z}_k\|_2^2)^{-1}}$$

Minimize $f_{t\text{-SNE}}(\mathbf{Z}) = \sum_n \sum_m p_{nm} \log \frac{p_{nm}}{q_{nm}}$ **via gradient descent**

Kullback-Leibler (**KL**) divergence

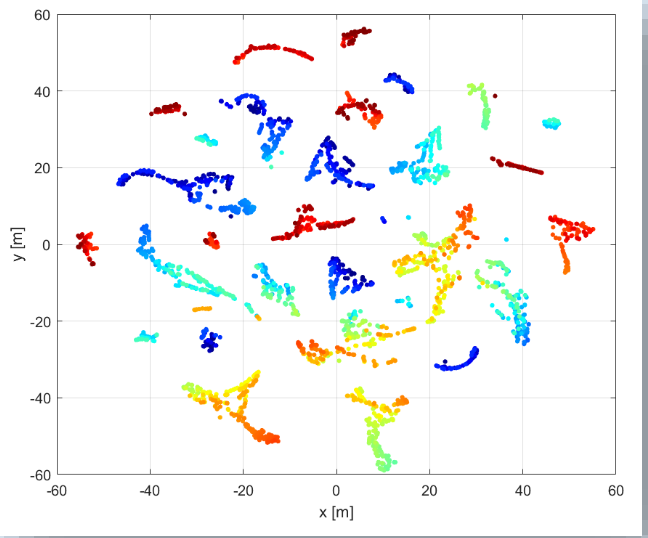
Apply t-SNE to CSI Data with PAP Dissimilarity



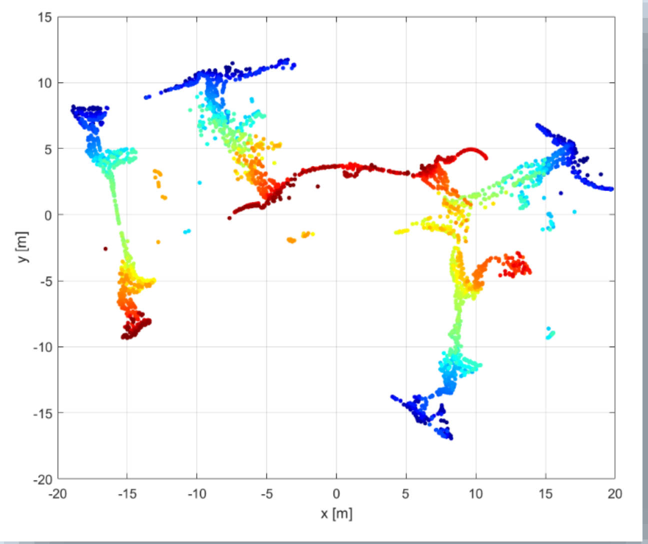
CSI samples collected at colored positions

t-SNE

Can't obtain absolute geographical location info.



Small number of neighbors **(perplexity)**



Large number of neighbors **(perplexity)**

Semi-supervised t-SNE Problem

- To equip t-SNE with localization capability, we need to use some **position labels** to govern the learning process of t-SNE.
- To this end, we formalize the following semi-supervised t-SNE problem:

$$\begin{cases} \underset{\mathbf{Z}}{\text{minimize}} & f_{t\text{-SNE}}(\mathbf{Z}) = \sum_n \sum_m p_{nm} \log \frac{p_{nm}}{q_{nm}}, \\ \text{subject to} & \mathbf{z}_i = \mathbf{y}_i, i \in \mathcal{L}, \mathcal{L} = \{1, \ldots, L\}. \end{cases}$$

Semi-supervised t-SNE Algorithm

- Four adjustable Parameters
- Map the labeled CSI to its fixed position during learning iterations
- The idea is simple, and we will see it is effective!

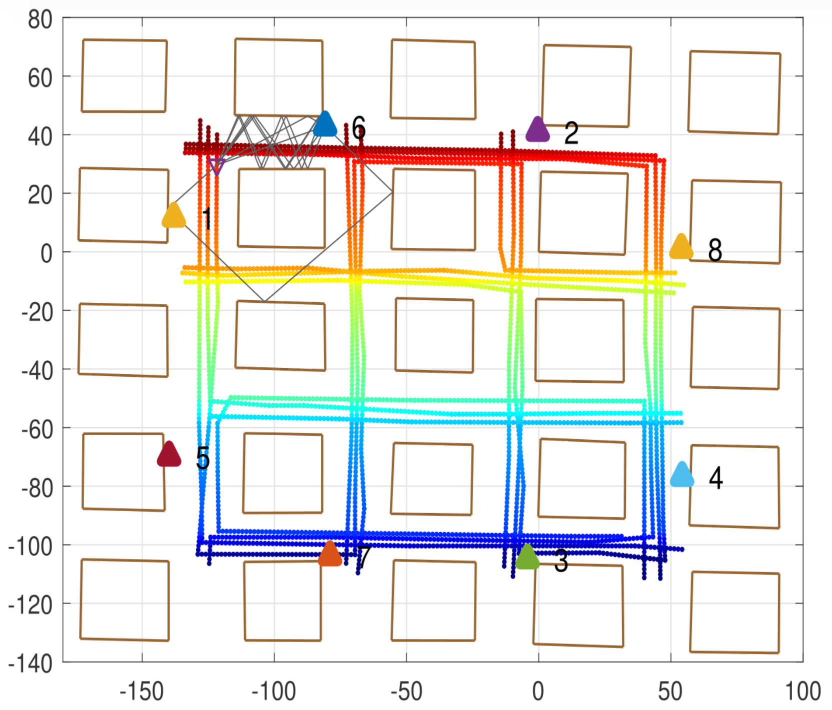
Algorithm 1 The semi-supervised t-SNE algorithm

Inputs: $\mathbf{D} \in \mathbb{R}^{N \times N}$, $\{\mathbf{y}_1, \ldots, \mathbf{y}_L\}$, $\mathcal{L} = \{1, \ldots, L\}$
Cost function parameter: k_t
Optimization parameters: T, η and α
Initialize: $\mathbf{Z}^{(0)} = [\mathbf{z}_1^{(0)}, \ldots, \mathbf{z}_N^{(0)}]$, $\mathbf{Z}^{(-1)} = \mathbf{Z}^{(0)}$, with
    $\mathbf{z}_i^{(0)} = \mathbf{y}_i$, for $i \in \mathcal{L}$,
    $\mathbf{z}_i^{(0)} = \mathbf{y}_n$, for $i \notin \mathcal{L}$ and $d_{n,i}$ is smallest for all $n \neq i$
Use binary search to determine the kernel sizes $\{\delta_n\}_{n=1}^N$, and compute the probability matrix $\mathbf{P}$ using (3)
for $t = 1, \ldots, T$ **do**
    Compute probability matrix $\mathbf{Q}$ using (4)
    Compute $\nabla = [\frac{\mathrm{d} f_{t\text{-SNE}}(\mathbf{Z})}{\mathrm{d}\mathbf{z}_1}, \ldots, \frac{\mathrm{d} f_{t\text{-SNE}}(\mathbf{Z})}{\mathrm{d}\mathbf{z}_N}]$ using (5)
    Update $\mathbf{Z}^{(t)} = \mathbf{Z}^{(t-1)} + \eta\nabla + \alpha(\mathbf{Z}^{(t-1)} - \mathbf{Z}^{(t-2)})$
    Set $\mathbf{z}_i^{(t)} = \mathbf{y}_i$, for $i \in \mathcal{L}$
end for
return: $\mathbf{Z}^{(T)}$

Performance Evaluation- Simulated Scenario

- GIS map data from OpenStreetMap (OSM)
- 8 BSs with ULAs, each has M = 16 elements
- A ray-tracing channel model is used to generate the multi-path channels
- The carrier frequency is 28 GHz, system bandwidth is 256 MHz with 128 OFDM subcarriers
- UEs transmit signals with a fixed power of 23 dBm
- UEs on roads with a speed of 5 meters per second
- We collect 1500 CSI samples from UE traces

Parameter	Value	Parameter	Value
Carrier frequency	28 GHz	UE pilot Tx power	23 dBm
System bandwidth	256 MHz	UE antenna pattern	Omnidirectional
Subcarrier number	128	BS antenna pattern	Cosine response



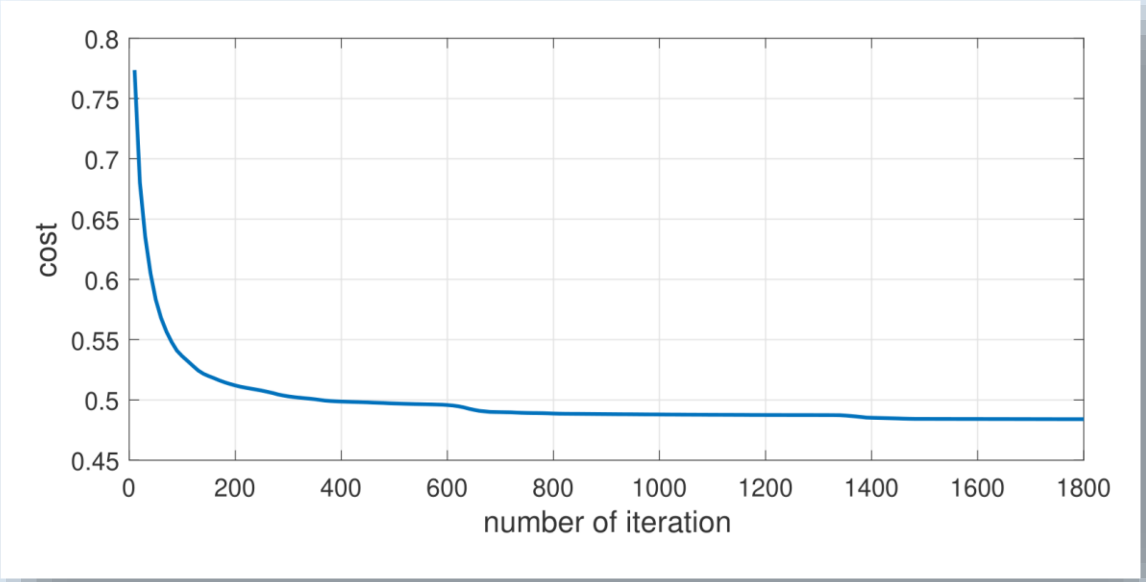
Performance Evaluation - Metric

- Mean localization error (MLE)

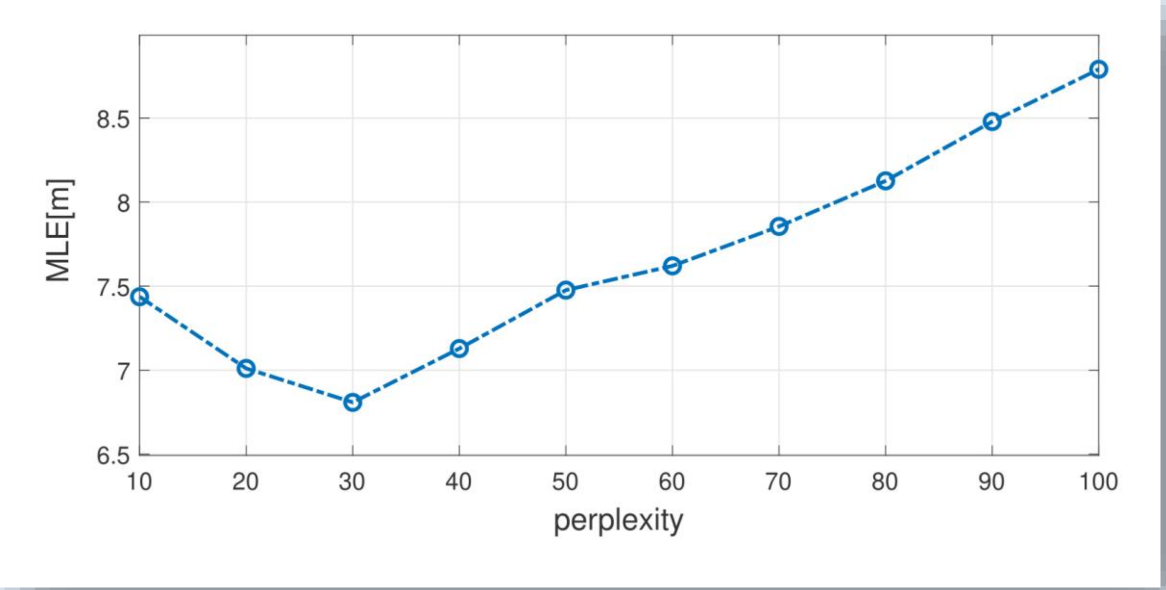
$$MLE = \frac{1}{U} \sum_{n=1}^{U} \|\mathbf{z}_{L+n} - \mathbf{p}_{L+n}\|_2$$

U = 1425 unlabeled samples, L = 75 labeled samples

Performance Evaluation - Learning Process



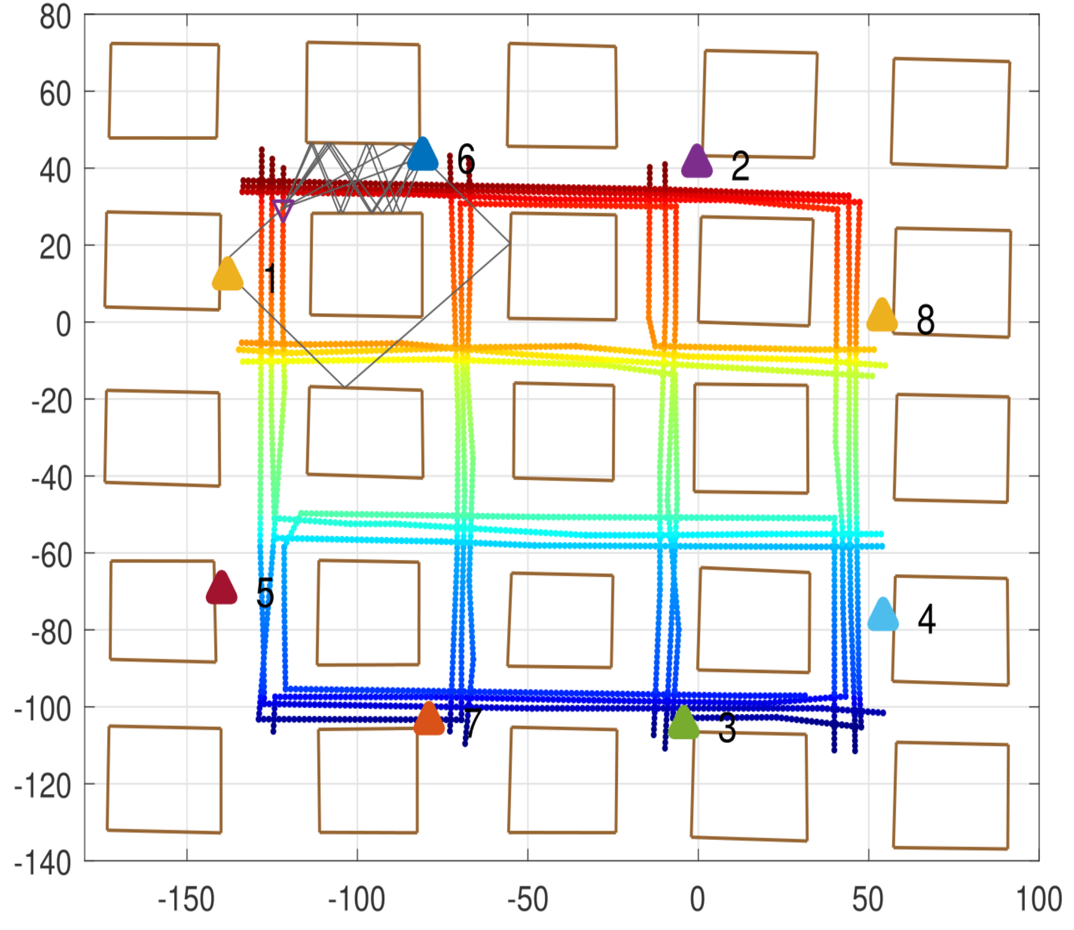
St-SNE iteration process with perplexity k_t = 30, learning rate η = 1000, and momentum α = 0.6

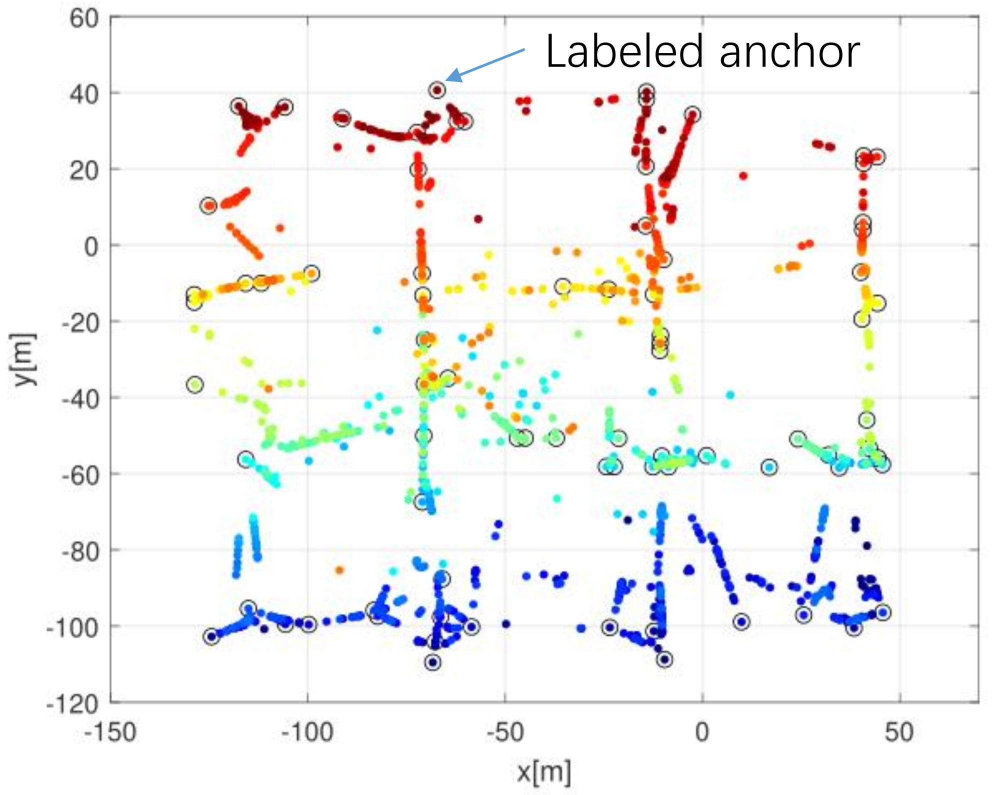


The effects of perplexity k_t in St-SNE on the localization performance

Performance Evaluation - Comparisons

Compared to **kNN** and **semi-supervised Laplacian Eigenmap** (SLE)



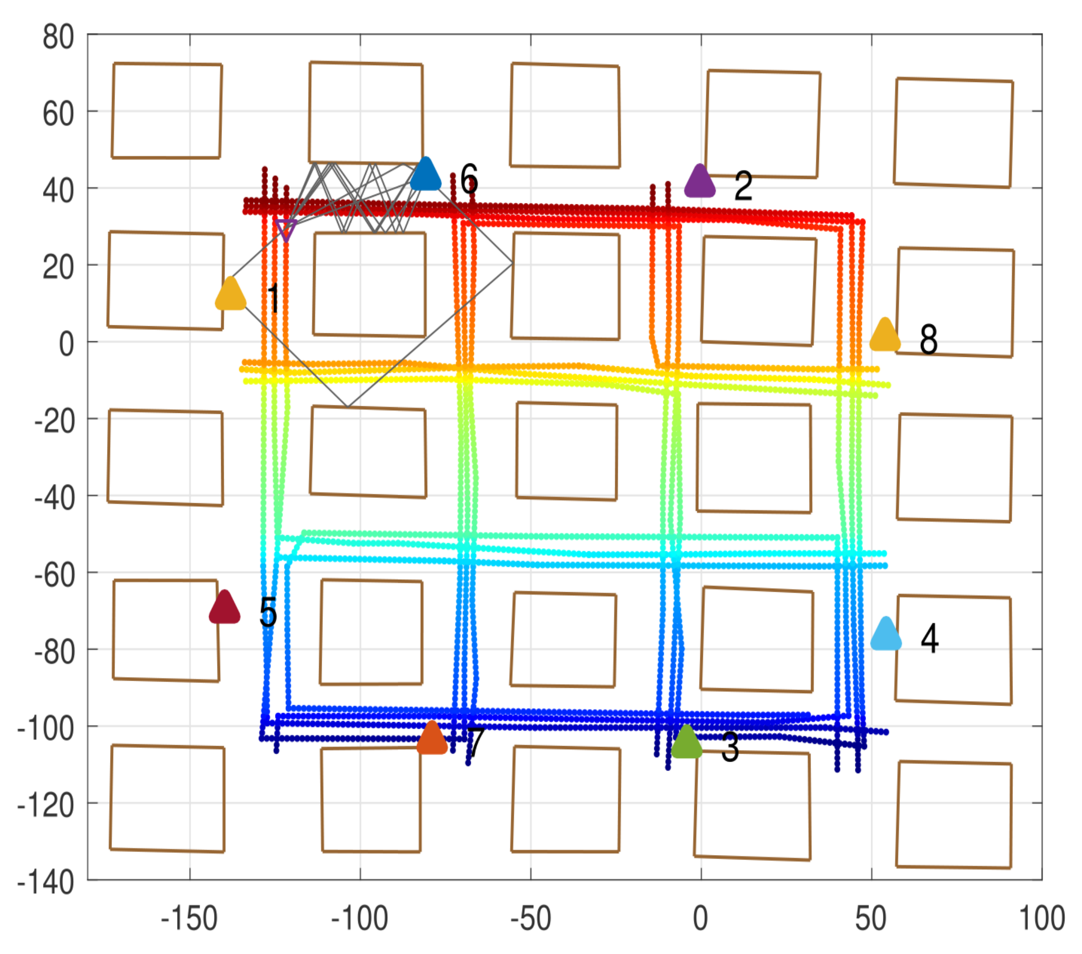


(a) kNN

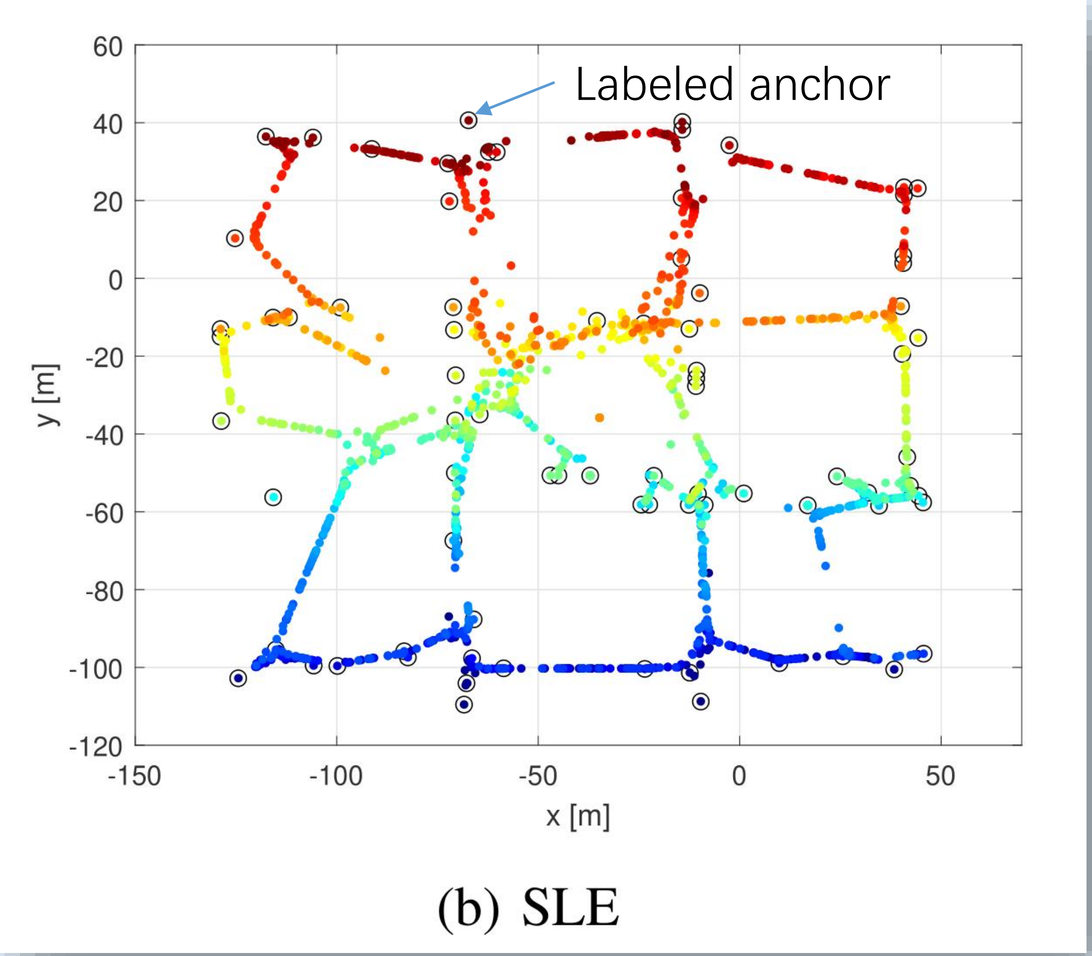
Ground truth **Estimated**

Performance Evaluation - Comparisons

Compared to **kNN** and **semi-supervised Laplacian Eigenmap** (SLE)



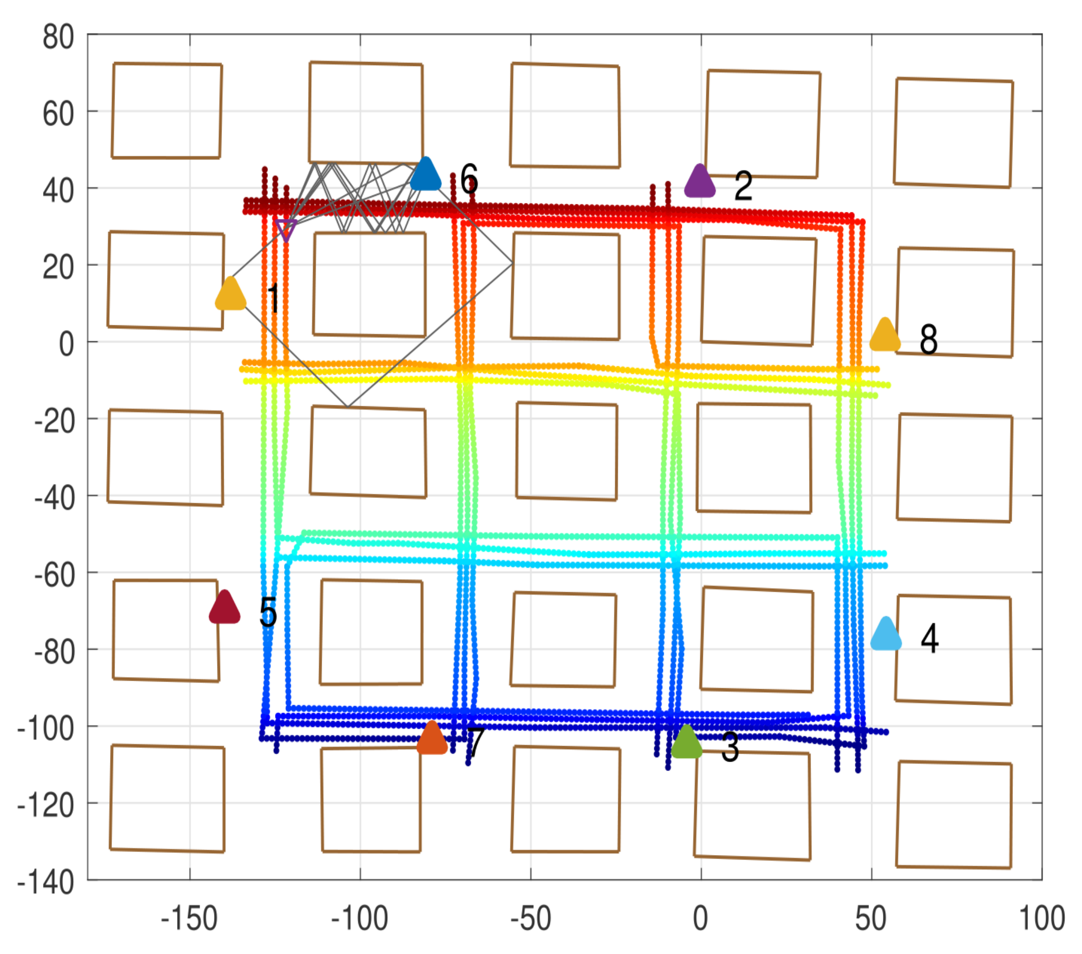
Ground truth



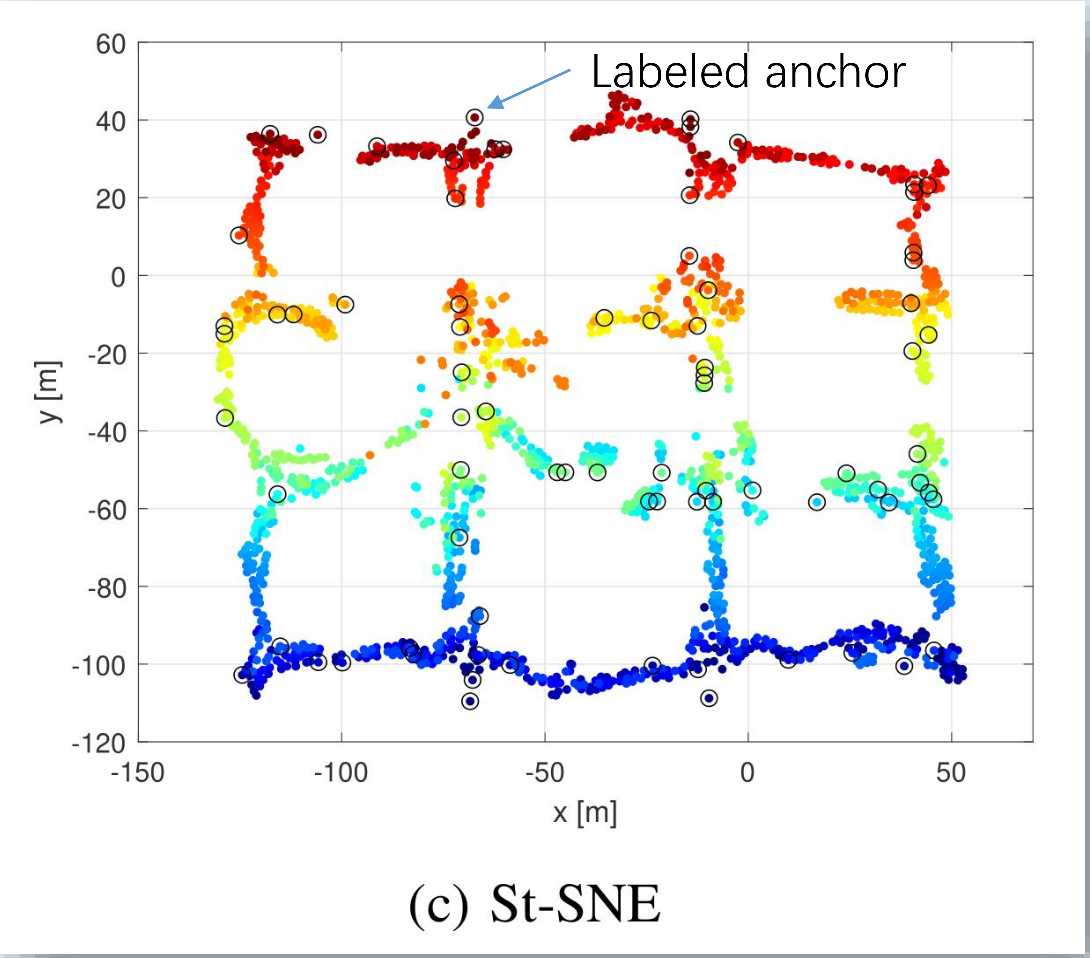
Estimated

Performance Evaluation - Comparisons

Compared to **kNN** and **semi-supervised Laplacian Eigenmap** (SLE)



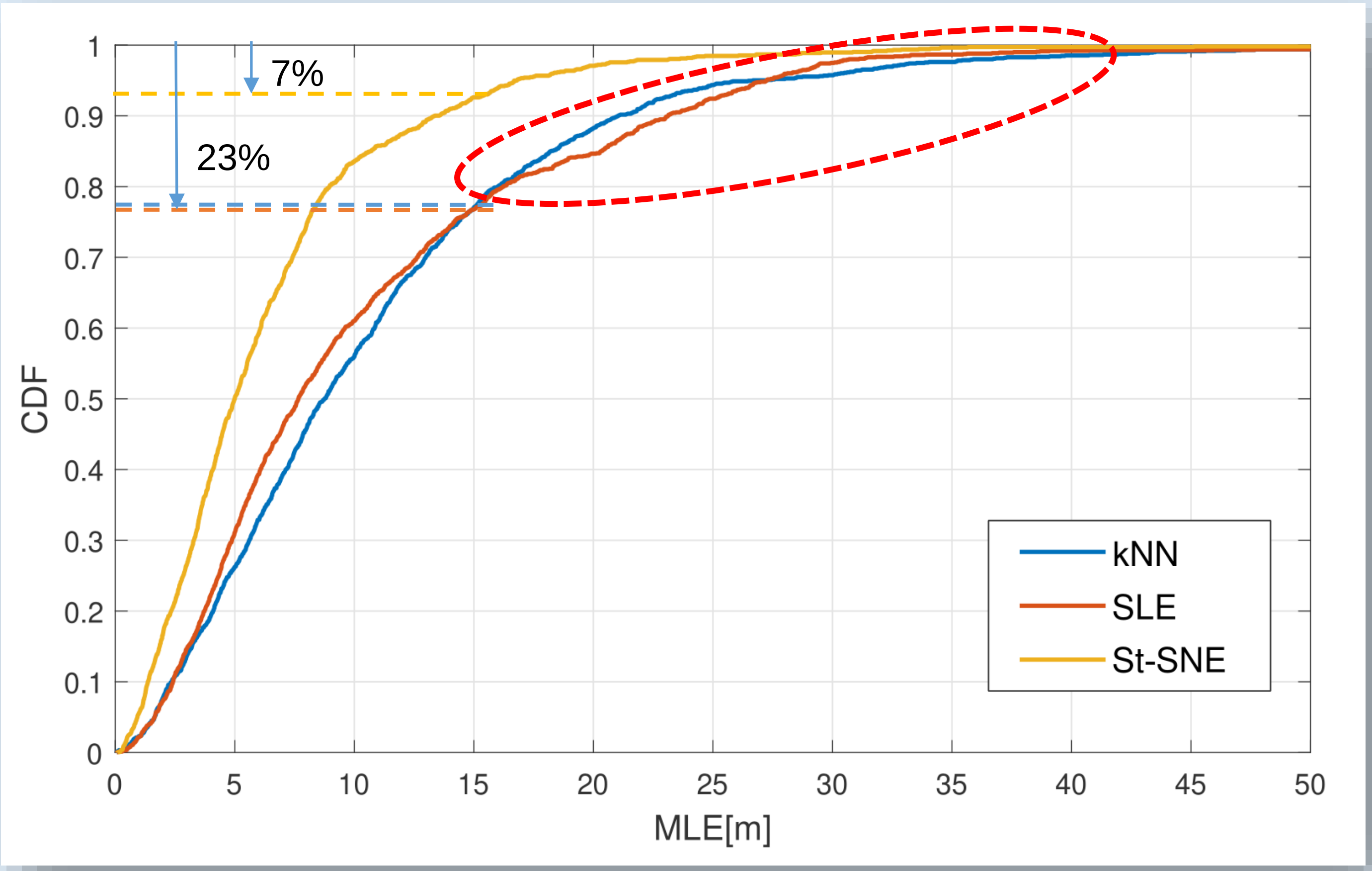
Ground truth



Estimated

Performance Evaluation - Comparisons

- Positions of points far from the labeled anchors cannot be accurately estimated via kNN and SLE
- St-SNE greatly reduces the errors of those points, with only **7%** of unlabeled points having a error **>15 m**



Cumulative Distribution Function (CDF) of localization errors

Conclusion

- We have proposed a machine learning method called St-SNE for **mmWave multi-cell mobile localization**.
- It embeds the high-dimensional multi-antenna CSI into the 2D map by governing the self-learning process of t-SNE with **a few position labels**.
- It is **scalable and automatic** in the sense that it could be implemented for multi-cell networks, with spatially sparse labeled samples, and does not require accurate network synchronization.
- With **relatively higher computation complexity**. A prospective research direction would be to use a graph or tree method to accelerate its computation of the probability matrixes.