MULTIANTENNA CHANNEL MAP PREDICTION WITH MISSING LOCATION INFORMATION USING CONTRASTIVE LEARNING AND GRAPH NEURAL NETWORKS

Wilson Funk *Seung-Jun Kim*

Dept. of CSEE, University of Maryland, Baltimore County, Baltimore, MD, USA
E-mails: {wfunk1,sjkim}@umbc.edu

ABSTRACT

A radio map estimation method for multiantenna statistical channel state information (CSI) is proposed, which can interpolate the CSI measurements across space. A practical challenge is that for some measurements, location information may not be readily available, due to, e.g., poor GPS reception or privacy concerns. To circumvent this difficulty, contrastive learning is employed to learn a metric embedding space, in which both CSI and location measurements can be related. We also incorporate local patches of building maps around sensor locations, when locations are available. Based on the embeddings, the relative proximity of all measurements, with or without location tags, is captured in a graph. A graph neural network (GNN) is then employed to perform map interpolation. The numerical tests verify the effectiveness of the proposed graph construction and map estimation methods.

Index Terms— Contrastive learning, graph construction, graph neural network, multiantenna channel map, radio map estimation.

1. INTRODUCTION

A radio map captures the distribution of key quantities characterizing a radio frequency (RF) propagation environment across a geographic area [1]. Signal strength maps (e.g., interference power maps and power spectral density (PSD) maps) can reveal regions of dense or sparse spectrum usage [2]. Propagation maps (e.g., channel gain maps and multiantenna channel maps) can predict the propagation characteristics between arbitrary points in the deployment area and track their variations over time [3]. Such maps are instrumental for various network management and resource allocation tasks, including power control, dynamic spectrum access, unmanned aerial vehicle (UAV) deployment, and fast beam alignment [4].

Constructing radio maps from physical principles often requires detailed obstacle modeling and entails either high computational cost or crude approximations. A viable alternative is to adopt data-driven methods. Typically they involve interpolation of the RF descriptors for new locations in the area where the measurements are not available, based on those taken in the nearby locations. PSDs and multiantenna channels were interpolated over space in this way using deep convolutional autoencoders in [5, 6]. A recent idea is to correlate obstacle maps or satellite imagery of the area with large-scale measurements (obtained either through measurement campaigns or computationally intensive ray-tracing simulations) leveraging machine learning techniques. Two-dimensional building maps with base station (BS) locations were used as the input to a convolutional neural network (CNN) to predict the channel gains in [7]. Sparse measurements and building maps were both used for signal strength interpolation in [8]. Graph neural networks (GNNs) were employed for interpolation of received signal strengths based on the graphs of sensor locations constructed from building maps [9]. In all of these works, it is critical that the measurements are paired with location information. However, in some cases, location information may not be readily available due to, e.g., poor GPS reception, limited battery power of the sensors, or privacy concerns.

In this work, multiantenna channel maps are constructed via spatial interpolation when some measurements lack location information. A key enabling insight is that multiantenna channel state information (CSI) does contain information about its own location. For instance, in channel charting, low-dimensional embeddings were learned for the multiantenna CSI acquired over a geographical area. When the embeddings were visualized, the trajectory closely matched with the path that the sensor traveled to collect the CSI [10].

Based on such intuition, we first perform metric learning that can provide a common embedding space for both CSI measurements and location tags. In particular, contrastive learning is employed to train a pair of encoders for CSI and locations that can maximize the mutual information between their embeddings [11]. A small patch of the building map around the sensor location is also utilized when the location is available. Then, a graph is constructed using the learned metric for all measurements including the ones without location information. The graph intuitively connects the measurements in similar propagation conditions. Finally, a GNN is employed to interpolate the CSI measurements across the embedding space. Our numerical tests show that the method makes effective use of all available measurements.

The rest of the paper is organized as follows. The map construction problem is formulated in Sec. 2. Contrastive learning of the embedding space is discussed in Sec. 3. The GNN-based map estimation procedure is explained in Sec. 4. The results from numerical tests are presented in Sec. 5. Conclusions are provided in Sec. 6.

2. PROBLEM STATEMENT

Consider a set of geographical regions $\mathcal{R}_j, j = 1, 2, \ldots, J$, where each region contains a single base station (BS) at location $\mathbf{x}_j^{\text{BS}}$ and a set of user equipments (UEs) at $\mathcal{X}_j := \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{N_j}\} \subset \mathbb{R}^2$. The BS is equipped with N antennas and the UEs are assumed to have single antennas for simplicity. Our goal is to reconstruct the statistical CSI for multiantenna transmission from the BS to all UE locations within each region, based on a set of CSI measurements collected at a subset of UE locations—some of which lack the associated location information.

Let $\mathbf{h}_\mathbf{x} \in \mathbb{C}^N$ denote the instantaneous CSI at location $\mathbf{x} \in \mathcal{X}_j$.

This work was supported in part by the US National Science Foundation grant 2242412.

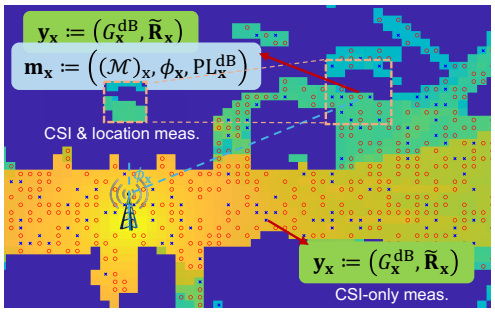


Fig. 1. Illustration of feature construction.

Then, at a set of sensor locations $\mathcal{S}_j \subset \mathcal{X}_j$, measurements of the channel covariances are made.

$$\mathbf{R}_\mathbf{x} = \mathbb{E}\{\mathbf{h}_\mathbf{x}\mathbf{h}_\mathbf{x}^H\} = G_\mathbf{x}\tilde{\mathbf{R}}_\mathbf{x} \in \mathbb{C}^{N\times N},\ \mathbf{x} \in \mathcal{S}_j \tag{1}$$

where $G_\mathbf{x} := \frac{1}{N}\mathrm{tr}\{\mathbf{R}_\mathbf{x}\}$ is the channel gain and $\tilde{\mathbf{R}}_\mathbf{x} := \mathbf{R}_\mathbf{x}/G_\mathbf{x}$ is the normalized covariance. From the combined path-loss and shadowing model [12], the dB-scale gain $G_\mathbf{x}^{\mathrm{dB}} := 10\log_{10} G_\mathbf{x}$ can be expressed as

$$G_\mathbf{x}^{\mathrm{dB}} = G_{0,j}^{\mathrm{dB}} - 10\gamma_j \log_{10}\frac{d_\mathbf{x}}{d_0} - S_\mathbf{x}^{\mathrm{dB}} = -\mathrm{PL}_\mathbf{x}^{\mathrm{dB}} - S_\mathbf{x}^{\mathrm{dB}} \tag{2}$$

where $\mathrm{PL}_\mathbf{x}^{\mathrm{dB}} := -G_{0,j}^{\mathrm{dB}} + 10\gamma_j \log_{10}(d_\mathbf{x}/d_0)$ is the path-loss with constant $G_{0,j}^{\mathrm{dB}}$ and path-loss exponent γ_j for region $\mathcal{R}_j$, $d_\mathbf{x} := \|\mathbf{x} - \mathbf{x}_j^{\mathrm{BS}}\|_2$ is the propagation distance from the BS, d_0 is a reference distance for the antenna far field, and $S_\mathbf{x}^{\mathrm{dB}}$ is a shadow fading variable. The CSI measurement vector $\mathbf{y}_\mathbf{x}$ at location $\mathbf{x}$ is defined to contain $G_\mathbf{x}^{\mathrm{dB}}$ and the strictly upper-triangular part of $\tilde{\mathbf{R}}_\mathbf{x}$ (due to the symmetry of covariance matrices). Thus, with real entries, $\mathbf{y}_\mathbf{x} \in \mathbb{R}^{N^2-N+1}$.

The CSI measurements are often paired with the true locations $\mathbf{x}$. However, in some cases, the location information may not be available. Thus, for $\mathbf{x} \in \mathcal{S}_j^{\mathrm{full}} \subset \mathcal{S}_j$, the measurements are tuples of the form $(\mathbf{x}, \mathbf{y}_\mathbf{x})$, while for $\mathbf{x} \in \mathcal{S}_j^{\mathrm{CSI}} := \mathcal{S}_j \backslash \mathcal{S}_j^{\mathrm{full}}$, only the CSI measurements $\mathbf{y}_\mathbf{x}$ are available, with $\mathbf{x}$ missing.

We also assume that for each region $\mathcal{R}_j$, there is a building map $\mathcal{M}_j$ available. For instance, the building maps can be binary masks indicating the building locations. In addition to providing the configuration of obstacles for RF propagation, the maps are useful for maintaining a common frame of reference regarding the location coordinates across different regions $\mathcal{R}_j$. Specifically, instead of directly using the coordinate $\mathbf{x}$, the local patch $(\mathcal{M}_j)_\mathbf{x}$ of $\mathcal{M}_j$ centered around $\mathbf{x} \in \mathcal{X}_j$ is used, paired with the orientation $\phi_\mathbf{x}$ toward the BS relative to the antenna array normal [13]. Furthermore, the path-loss value $\mathrm{PL}_\mathbf{x}^{\mathrm{dB}}$ is employed as the input feature. Thus, the location-based feature vector $\mathbf{m}_\mathbf{x}$ is defined as $\mathbf{m}_\mathbf{x} := ((\mathcal{M}_j)_\mathbf{x}, \phi_\mathbf{x}, \mathrm{PL}_\mathbf{x}^{\mathrm{dB}})$. The feature construction is illustrated in Fig. 1.

Our goal is, given the datasets $\mathcal{D}_j := \mathcal{D}_j^{\mathrm{full}} \cup \mathcal{D}_j^{\mathrm{CSI}}$, where $\mathcal{D}_j^{\mathrm{full}} := \{(\mathbf{m}_\mathbf{x}, \mathbf{y}_\mathbf{x})\}_{\mathbf{x}\in\mathcal{S}_j^{\mathrm{full}}}$ and $\mathcal{D}_j^{\mathrm{CSI}} := \{\mathbf{y}_\mathbf{x}\}_{\mathbf{x}\in\mathcal{S}_j^{\mathrm{CSI}}}$ for $j = 1, 2, \ldots, J$, to construct multiantenna channel maps, i.e., to predict $G_\mathbf{x}^{\mathrm{dB}}$ and $\tilde{\mathbf{R}}_\mathbf{x}$ for arbitrary UE locations $\mathbf{x} \in \mathcal{X}_j$, $j = 1, \ldots, J$.

The critical challenge arises from the coexistence of two types of measurements: those with location information and those without. As a result, it is not straightforward to jointly process them using a single frame of reference or metric, such as location or distance, which is essential for interpolation based on CNNs [6] or GNNs [9]. Our first task is therefore to learn a common embedding space in which both types of measurements can be related. A graph can then be constructed based on the distance metric derived from the embeddings, and a GNN can be employed for channel map prediction.

3. CONTRASTIVE LEARNING OF EMBEDDING SPACE

Contrastive learning is a self-supervised representation learning strategy, where the embeddings of the positive samples (often obtained by augmentation of the inputs) are encouraged to be highly correlated and those of the negative samples to be maximally separated [14, 15]. More specifically, it has been argued that the popular information noise-contrastive estimation (InfoNCE) cost function maximizes the mutual information or the alignment in the positive samples while preserving the information content in the samples [11, 16]. The so learned embedding space has often showed superior performance in downstream tasks. Extensions to multimodal inputs have been proposed in various contexts [17, 18].

We want to construct a metric space where the three types of measurements can be related so as to construct a graph, namely, the full measurements of the form $(\mathbf{x}, \mathbf{y}_\mathbf{x})$, the CSI-only measurements $\mathbf{y}_\mathbf{x}$, and the location coordinates $\mathbf{x}$, where the last type is needed for inferring the CSI at a new location.

This can be achieved by contrastive learning with the pairs $(\mathbf{m}_\mathbf{x}, \mathbf{y}_\mathbf{x}) \in \mathcal{D}_j^{\mathrm{full}}$ treated as positive samples, and $(\mathbf{m}_{\mathbf{x}_1}|_{\mathbf{x}_1\in\mathcal{X}_j}, \mathbf{y}_{\mathbf{x}_2} \in \mathcal{D}_j)$ as negative samples. Specifically, two encoders $f_1(\mathbf{m}) \in \mathbb{S}^{D-1}$ and $f_2(\mathbf{y}) \in \mathbb{S}^{D-1}$ are trained, where $\mathbb{S}^{D-1}$ is a unit hypersphere in $\mathbb{R}^D$. Given

$$h(\mathbf{m}, \mathbf{y}) := \exp\left(\frac{1}{\tau} f_1(\mathbf{m})^T f_2(\mathbf{y})\right) \tag{3}$$

where ^{T} denotes transposition and $\tau > 0$ is a tunable temperature parameter, the InfoNCE loss functions can be defined as [17]

$$\mathcal{L}_{1,2} := \mathop{\mathbb{E}}_{\substack{(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})\in\cup_j\mathcal{D}_j^{\mathrm{full}} \\ \mathbf{y}_{\mathbf{x}_i}\in\cup_j\mathcal{D}_j, i=1,\ldots,I}} \left[-\log\frac{h(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})}{h(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})+\sum_k h(\mathbf{m}_\mathbf{x},\mathbf{y}_{\mathbf{x}_i})}\right] \tag{4}$$

$$\mathcal{L}_{2,1} := \mathop{\mathbb{E}}_{\substack{(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})\in\cup_j\mathcal{D}_j^{\mathrm{full}} \\ \mathbf{x}_i\in\cup_j\mathcal{X}_j, i=1,\ldots,I}} \left[-\log\frac{h(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})}{h(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})+\sum_k h(\mathbf{m}_{\mathbf{x}_i},\mathbf{y}_\mathbf{x})}\right]. \tag{5}$$

The overall loss function for contrastive learning combines the two:

$$\mathcal{L}^{\mathrm{contrastive}} = \mathcal{L}_{1,2} + \mathcal{L}_{2,1}. \tag{6}$$

Once mappings f_1 and f_2 are learned, they can be used for downstream tasks. For example, one can predict the CSI at a new location $\mathbf{x}$ by a decoder d_1 such that

$$\mathbf{y}_\mathbf{x} \approx d_1(f_1(\mathbf{m}_\mathbf{x})) \tag{7}$$

which can be trained through the cost function

$$\mathcal{L}_1^{\mathrm{dec}}(d_1) := \mathbb{E}_{(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})\in\cup_j\mathcal{D}_j^{\mathrm{full}}}\|\mathbf{y}_\mathbf{x} - d_1(f_1(\mathbf{m}_\mathbf{x}))\|_2^2. \tag{8}$$

One can also denoise the CSI measurement using a decoder d_2 or $d_{1,2}$, which can be trained by minimizing

$$\mathcal{L}_2^{\mathrm{dec}}(d_2) := \mathbb{E}_{\mathbf{y}_\mathbf{x}\in\cup_j\mathcal{D}_j}\|\mathbf{y}_\mathbf{x} - d_2(f_2(\mathbf{y}_\mathbf{x}))\|_2^2 \tag{9}$$

$$\mathcal{L}_{1,2}^{\mathrm{dec}}(d_{1,2}) := \mathop{\mathbb{E}}_{(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x})\in\cup_j\mathcal{D}_j^{\mathrm{full}}}\|\mathbf{y}_\mathbf{x} - d_{1,2}(f^{\mathrm{fus}}(\mathbf{m}_\mathbf{x},\mathbf{y}_\mathbf{x}))\|_2^2 \tag{10}$$

respectively, where the fused embedding can be defined as

$$f^{\mathrm{fus}}(\mathbf{m}_\mathbf{x}, \mathbf{y}_\mathbf{x}) := \mathcal{P}_{\mathbb{S}^{D-1}}\left[\zeta f_1(\mathbf{m}_\mathbf{x}) + (1-\zeta) f_2(\mathbf{y}_\mathbf{x})\right] \tag{11}$$

with $\zeta \in [0,1]$ determined from the dataset to maximize the denoising performance, and projection $\mathcal{P}_{\mathbb{S}^{D-1}}$ ensuring that the fused embedding is normalized to unit norm. Note that in (7)–(11), f_1 and f_2 are fixed and not jointly trained.

Although the CSI prediction using (7) is expected to provide reasonable performance, this approach ignores the fact that the samples in $\mathcal{D}_j$ for a given region $\mathcal{R}_j$ are actually correlated. By capitalizing on the spatial correlations, it is possible to achieve better prediction performance. Furthermore, it is noted that the contrastive learning embeddings are not trained for the prediction tasks per se. Through the information preservation property of the InfoNCE cost [16], it is conceivable that the embeddings contain richer information than required for simply discriminating the positive versus negative examples. Our goal in the next section is to tap into such information present in the neighboring samples to enhance CSI prediction.

4. GNN-BASED MAP CONSTRUCTION

4.1. Graph Construction

For a given region $\mathcal{R}_j$, our approach is to employ a GNN to process the data in $\mathcal{D}_j$ to predict the CSI at a new location $\mathbf{x} \in \mathcal{X}_j \backslash \mathcal{S}_j$. For that, a graph $\mathcal{G}_j = (\mathcal{V}_j, \mathcal{E}_j)$ must first be constructed. The vertex set $\mathcal{V}_j$ is equal to $\mathcal{X}_j$, the set of all locations. Define

$$\tilde{\mathbf{v}}_{\mathbf{x}} := \begin{cases} f_1(\mathbf{m}_{\mathbf{x}}) & \text{if } \mathbf{x} \in \mathcal{X}_j \backslash \mathcal{S}_j \\ f_2(\mathbf{y}_{\mathbf{x}}) & \text{if } \mathbf{x} \in \mathcal{S}_j^{\text{CSI}} \\ f^{\text{fus}}(\mathbf{m}_{\mathbf{x}}, \mathbf{y}_{\mathbf{x}}) & \text{if } \mathbf{x} \in \mathcal{S}_j^{\text{full}} \end{cases} \tag{12}$$

Then, the distance between two vertexes $\mathbf{x}_1$ and $\mathbf{x}_2$ can be evaluated via the Euclidean distance $\|\tilde{\mathbf{v}}_{\mathbf{x}_1} - \tilde{\mathbf{v}}_{\mathbf{x}_2}\|_2 = \sqrt{2(1 - \tilde{\mathbf{v}}_{\mathbf{x}_1}^T \tilde{\mathbf{v}}_{\mathbf{x}_2})}$. Let $e_{(\mathbf{x}_1,\mathbf{x}_2)} := \tilde{\mathbf{v}}_{\mathbf{x}_1}^T \tilde{\mathbf{v}}_{\mathbf{x}_2}$. Then edge set $\mathcal{E}_j$ is constructed from

$$(\mathbf{x}_1, \mathbf{x}_2) \in \mathcal{E}_j \text{ if and only if } e_{(\mathbf{x}_1,\mathbf{x}_2)} \geq \theta \tag{13}$$

where $\theta \in [-1, 1]$ is a pre-specified threshold. Note that the edges are naturally bidirectional and self-loops are included, i.e., $(\mathbf{x}, \mathbf{x}) \in \mathcal{E}_j$. The edges $(\mathbf{x}_1, \mathbf{x}_2)$ are also endowed with edge attributes $e_{(\mathbf{x}_1,\mathbf{x}_2)}$.

4.2. GNN Inference and Training

Once graph $\mathcal{G}_j$ is constructed for region $\mathcal{R}_j$, a GNN can be employed to interpolate CSI over the entire set of locations $\mathcal{X}_j$. Given $\mathcal{E}_j$, the set of neighbors of a vertex $\mathbf{x} \in \mathcal{V}_j$ is defined as $\mathcal{N}_j(\mathbf{x}) := \{\mathbf{x}' : (\mathbf{x}', \mathbf{x}) \in \mathcal{E}_j\}$.

The GNN takes vertex and edge attributes as input. One could use $\tilde{\mathbf{v}}_{\mathbf{x}}$ directly as the vertex attributes. However, this fails to incorporate the origin of the embeddings, which may be important as it can help capture additional details in the measurements not exploited in contrastive learning and account for noise levels that depend on the measurement type. To maximally leverage available information, the vertex attributes are constructed as the concatenation of all feature embeddings as

$$\mathbf{v}_{\mathbf{x}} := \begin{cases} [f_1(\mathbf{m}_{\mathbf{x}}) \parallel \mathbf{0} \parallel \mathbf{0}] & \text{if } \mathbf{x} \in \mathcal{X}_j \backslash \mathcal{S}_j \\ [\mathbf{0} \parallel f_2(\mathbf{y}_{\mathbf{x}}) \parallel \mathbf{0}] & \text{if } \mathbf{x} \in \mathcal{S}_j^{\text{CSI}} \\ [f_1(\mathbf{m}_{\mathbf{x}}) \parallel f_2(\mathbf{y}_{\mathbf{x}}) \parallel f^{\text{fus}}(\mathbf{m}_{\mathbf{x}}, \mathbf{y}_{\mathbf{x}})] & \text{if } \mathbf{x} \in \mathcal{S}_j^{\text{full}} \end{cases} \tag{14}$$

where $\parallel$ denotes concatenation.

We adopt the multi-headed graph attention network v2 (GATv2) [19], which processes the input vertex attributes $\mathbf{z}_{\mathbf{x}}^{(l)}$ from the previous layer l and outputs new attributes $\mathbf{z}_{\mathbf{x}}^{(l+1)}$, for all $\mathbf{x} \in \mathcal{V}_j$. The first layer input $\mathbf{z}_{\mathbf{x}}^{(0)}$ is set to $\mathbf{v}_{\mathbf{x}}$. Specifically, for head $k \in \{1, 2, ..., K\}$ of layer $(l+1)$ with $l \in \{0, 1, ..., L-1\}$, the edge scores for attention are first evaluated as

$$s^{(l,k)}(\mathbf{x}, \mathbf{x}') := \left(\mathbf{a}_{\text{att}}^{(l,k)}\right)^T \text{LeakyReLU}\left(\mathbf{W}_{\text{att}}^{(l,k)}[\mathbf{z}_{\mathbf{x}}^{(l)} \| \mathbf{z}_{\mathbf{x}'}^{(l)} \| e_{\mathbf{x},\mathbf{x}'}]\right) \tag{15}$$

where $\mathbf{W}_{\text{att}}^{(l,k)} \in \mathbb{R}^{D_{\text{att}} \times (2D+1)}$ and $\mathbf{a}_{\text{att}}^{(l,k)} \in \mathbb{R}^{D_{\text{att}}}$ are learned parameters. Next, compute the attention weights as

$$\alpha_{\mathbf{x},\mathbf{x}'}^{(l,k)} := \frac{\exp(s^{(l,k)}(\mathbf{x}, \mathbf{x}'))}{\sum_{\mathbf{x}'' \in \mathcal{N}_j(\mathbf{x})} \exp(s^{(l,k)}(\mathbf{x}, \mathbf{x}''))}. \tag{16}$$

Then, the k-th head outputs the updated vertex attributes as

$$\tilde{\mathbf{z}}_{\mathbf{x}}^{(l,k)} = \sum_{\mathbf{x}' \in \mathcal{N}_j(\mathbf{x})} \alpha_{\mathbf{x},\mathbf{x}'}^{(l,k)} \mathbf{W}^{(l,k)} \mathbf{z}_{\mathbf{x}'}^{(l,k)} \tag{17}$$

where $\mathbf{W}^{(l,k)} \in \mathbb{R}^{\tilde{D} \times D}$ is a learnable linear transformation. Using the attention head outputs, the layer $(l+1)$ output embedding is obtained by concatenating and re-projecting the multihead outputs together with a residual connection through [20]

$$\mathbf{z}_{\mathbf{x}}^{(l+1)} = \sigma\left(\mathbf{W}_0^{(l)} \sigma\left(\mathbf{W}_{res}^{(l)} \mathbf{z}_{\mathbf{x}}^{(l)} + [\tilde{\mathbf{z}}_{\mathbf{x}}^{(l,1)} \| ... \| \tilde{\mathbf{z}}_{\mathbf{x}}^{(l,K)}]\right)\right) \tag{18}$$

where σ is an element-wise nonlinearity, $\mathbf{W}_{res}^{(l)} \in \mathbb{R}^{\tilde{D}K \times D}$ is a transformation applied to the residual connection, and $\mathbf{W}_0^{(l)} \in \mathbb{R}^{D \times \tilde{D}K}$ is another learnable linear transformation. The final layer output $\mathbf{z}_{\mathbf{x}}^{(L)}$ can then be decoded for CSI prediction at location $\mathbf{x}$ by using a decoder d as

$$\hat{\mathbf{y}}_{\mathbf{x}} = d(\mathbf{z}_{\mathbf{x}}^{(L)}). \tag{19}$$

For the training of the GNN including the decoder d, the training set $\mathcal{D}_j$ is first subsampled to construct the set of sensor measurements and the set of target CSI. That is, a fraction δ of $\mathcal{S}_j^{\text{full}}$ are subsampled to construct $\bar{\mathcal{S}}_j^{\text{full}}$. Likewise, a δ fraction of $\mathcal{S}_j^{\text{CSI}}$ are randomly sampled to construct $\bar{\mathcal{S}}_j^{\text{CSI}}$. Then, $\bar{\mathcal{S}}_j := \bar{\mathcal{S}}_j^{\text{full}} \cup \bar{\mathcal{S}}_j^{\text{CSI}}$ contains the sensor locations with (either full or CSI-only) measurements. The remaining $(1-\delta)$ fraction of $\mathcal{S}_j^{\text{full}}$ serves as the locations with the targets for training. Construct the vertex attributes $\mathbf{v}_{\mathbf{x}}$ for all $\mathbf{x} \in \bar{\mathcal{X}}_j := (\mathcal{X}_j \backslash \mathcal{S}_j) \cup \mathcal{S}_j^{\text{full}} \cup \bar{\mathcal{S}}_j^{\text{CSI}} = \mathcal{X}_j \backslash (\mathcal{S}_j^{\text{CSI}} \backslash \bar{\mathcal{S}}_j^{\text{CSI}})$ via (14) with $\mathcal{X}_j$, $\mathcal{S}_j$, $\mathcal{S}_j^{\text{CSI}}$ and $\mathcal{S}_j^{\text{full}}$ replaced with $\bar{\mathcal{X}}_j$, $\bar{\mathcal{S}}_j$, $\bar{\mathcal{S}}_j^{\text{CSI}}$ and $\bar{\mathcal{S}}_j^{\text{full}}$, respectively. (Prediction cannot be done for $\mathcal{S}_j^{\text{CSI}} \backslash \bar{\mathcal{S}}_j^{\text{CSI}}$ since the locations are not actually known.) Then, the GNN-based predictions $\{\hat{\mathbf{y}}_{\mathbf{x}}\}_{\mathbf{x} \in \bar{\mathcal{X}}_j}$ are obtained by (19). Note that for $\mathbf{x} \in \mathcal{S}_j^{\text{full}} \backslash \bar{\mathcal{S}}_j^{\text{full}}$, $\hat{\mathbf{y}}_{\mathbf{x}}$ is the prediction of unknown CSI, while for $\mathbf{x} \in \bar{\mathcal{S}}_j$, $\hat{\mathbf{y}}_{\mathbf{x}}$ amounts to a (denoised) reconstruction of the measured CSI. Overall, the training loss is given as

$$\mathcal{L} = \mathbb{E}_j \mathbb{E}_{\bar{\mathcal{S}}_j} \left[\frac{1}{|\mathcal{S}_j^{\text{full}} \backslash \bar{\mathcal{S}}_j^{\text{full}}|} \sum_{\mathbf{x} \in \mathcal{S}_j^{\text{full}} \backslash \bar{\mathcal{S}}_j^{\text{full}}} \|\mathbf{y}_{\mathbf{x}} - \hat{\mathbf{y}}_{\mathbf{x}}\|^2 + \frac{\lambda}{|\bar{\mathcal{S}}_j|} \sum_{\mathbf{x} \in \bar{\mathcal{S}}_j} \|\mathbf{y}_{\mathbf{x}} - \hat{\mathbf{y}}_{\mathbf{x}}\|^2 \right] \tag{20}$$

where $\lambda \geq 0$ is a hyperparameter balancing the two terms inside the bracket.

5. NUMERICAL EXPERIMENTS

5.1. Experiment Setup

The dataset for the experiment was constructed from the DeepMIMO v3 urban dataset [21] containing 20 cities with 3 BSs per city, yielding 60 CSI maps. Each BS is configured with a 2-by-4 uniform planar antenna array transmitting at 3.5 GHz with a 50 MHz bandwidth. The training set used a fraction α of all locations included in the dataset and a fraction β of the training set was assumed to be without location information. The values of α and β were varied from 10% to 50% in the experiment.

For the GNN structure, we used $L = 3$ layers each with $K = 4$ heads and an embedding dimension of $D = 48$, excluding the first GAT input, which has a corresponding dimension of $D = 144$ due to the concatenation of embeddings. The encoders and decoders were constructed using 5-layer MLPs. The Adam optimizer was used for both contrastive learning and GNN training [22]. For GNN training, warm restarts were used to avoid local minima [23]. We used $\tau = 0.25$, $\theta = 0.5$, $\lambda = 1.0$, $\delta = 0.5$, and $\zeta = 0.5$. The local map patch size was around 100 m $\times$100 m.

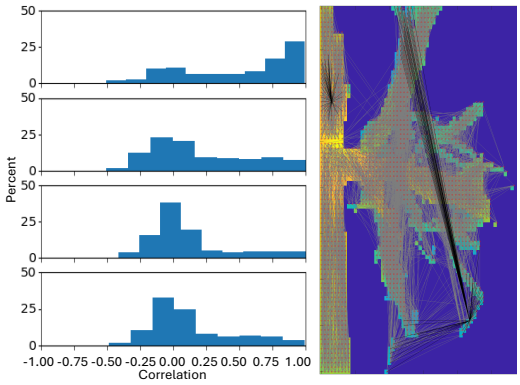
5.2. Contrastive Learning Performance

To see the performance of contrastive learning of the embedding space, the histograms of the correlations of the learned embeddings for a region are shown in Fig. 2(a). The training was done using $\alpha = 50\%$ of the training samples, $\beta = 10\%$ of which were assumed to be the CSI-only measurements. The top histogram corresponds to the distribution of the positive pairs $f_1(\mathbf{m}_\mathbf{x})^T f_2(\mathbf{y}_\mathbf{x})$, whereas the second to the bottom ones are of the negative pairs $f_1(\mathbf{m}_\mathbf{x})^T f_1(\mathbf{m}_{\mathbf{x}'})$, $f_2(\mathbf{y}_\mathbf{x})^T f_2(\mathbf{y}_{\mathbf{x}'})$, and $f_1(\mathbf{m}_\mathbf{x})^T f_2(\mathbf{y}_{\mathbf{x}'})$ for $\mathbf{x} \neq \mathbf{x}'$, respectively. It can be seen that the positive pairs show strong positive correlations while the negative pairs have correlations concentrated around 0, signifying that the two input modalities are well aligned in the embedding space. The heavier tail in the histogram of $f_1(\mathbf{m}_\mathbf{x})^T f_1(\mathbf{m}_{\mathbf{x}'})$ may be due to similar embeddings obtained from nearby locations.

Fig. 2(b) depicts the graph constructed in the region, superimposed on the ground truth gain map $G_\mathbf{x}^{\text{dB}}$ (yellow for high gains and blue for low). The red circles represent the locations in $\mathcal{X}_j$ sampled outside buildings. Two vertexes are highlighted to show more clearly their connections. It can be seen that the vertex near the BS connects to nearby vertexes. Interestingly, the vertex in the non-line-of-sight (NLOS) region from the BS makes distant connections to vertexes in similar NLOS conditions in addition to nearby connections. This is useful for the subsequent GNN-based CSI inference, which essentially interpolates smoothly over the graph.

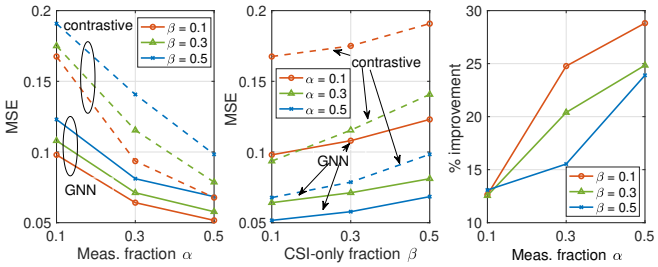
5.3. Map Estimation Performance

Fig. 3 shows the map estimation performance when α (determining the training set size) and β are varied. Fig. 3(a) depicts the mean squared error (MSE) of the normalized channel covariance prediction with α, $\beta \in \{0.1, 0.3, 0.5\}$. (The performance for the gain map prediction showed similar trends.) The solid lines represent the GNN-based predictions, whereas the dashed lines correspond to predictions using (7) without a GNN. It can be seen that the GNN significantly improves the performance compared to contrastive learning-based predictions, especially when the training set is small. This is due to the ability of the GNN to fuse the information spatially. Fig. 3(b) depicts the same results now using β as the x-axis. It can be seen that as the fraction of the CSI-only measurements increases,



(a) (b)

Fig. 2. Histograms of embedding correlations and constructed graph.



(a) (b) (c)

Fig. 3. Map estimation performance.

the performance degrades. However, the curves for GNN-based results tend to have gentler slopes than the ones from the contrastive learning only.

We also compared with the graph construction method in [9], which uses the location information and the building map of the region. Basically, two locations are connected in the graph if they are close enough geographically, but if there is an obstacle between the locations in the building map, the connection is not established. Clearly, this method cannot include the measurements without location tags in the GNN processing, but we still provided the same encoder/decoder structure as in our method. Fig. 3(c) shows the percent improvement in the MSE of our method relative to the method using the graph construction in [9]. It can be observed that our method generally improves the performance over the benchmark, verifying our graph construction method effectively captures the unlocalized measurements.

6. CONCLUSION

A radio map estimation method for multiantenna CSI has been developed, which performs spatial interpolation of CSI measurements to new locations. We tackled the practical challenge of the lack of associated location information for some measurements, which makes it difficult to apply the conventional methods, which require locations for all measurements. A metric learning method was developed so as to relate CSI and location information in the same embedding space. Local building map patches were incorporated when location information was available. Based on the learned metric, a graph of the entire set of measurements could be constructed, which the GNN employed for map inference. Numerical tests showed that the method significantly improves the CSI prediction performance, exploiting spatial dependency, while effectively leveraging measurements without location tags.

7. REFERENCES

[1] D. Romero and S.-J. Kim, "Radio map estimation: A data-driven approach to spectrum cartography," *IEEE Sig. Process. Mag.*, vol. 39, no. 6, pp. 53–72, 2022.

[2] D. Romero, S.-J. Kim, G. B. Giannakis, and R. Lopez-Valcarce, "Learning power spectrum maps from quantized power measurements," *IEEE Trans. Wireless Commun.*, vol. 65, no. 10, pp. 2547–2560, May 2017.

[3] S.-J. Kim, E. Dall'Anese, and G. B. Giannakis, "Cooperative spectrum sensing for cognitive radios using Kriged Kalman filtering," *IEEE J. Sel. Topics Sig. Process.*, vol. 5, no. 1, pp. 24–36, Feb. 2011.

[4] Y. Zeng and X. Xu, "Toward environment-aware 6G communications via channel knowledge map," *IEEE Wireless Commun.*, vol. 28, no. 3, pp. 84–91, 2021.

[5] Y. Teganya and D. Romero, "Deep completion autoencoders for radio map estimation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 3, pp. 1710–1724, 2022.

[6] G. Kim, R. Jin, W. Funk, S.-J. Kim, and H. Lim, "Multi-antenna channel map estimation using deep spatial interpolation," in *Proc. of the IEEE Int. Conf. Acoust. Speech Sig. Process. Workshops (ICASSPW)*, Seoul, South Korea, Aug. 2024, pp. 790–794.

[7] R. Levie, Ç. Yapar, G. Kutyniok, and G. Caire, "RadioUNet: Fast radio map estimation with convolutional neural networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 4001–4015, 2021.

[8] E. Krijestorac, S. Hanna, and D. Cabric, "Spatial signal strength prediction using 3D maps and deep learning," in *Proc. of the IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[9] G. Chen, Y. Liu, T. Zhang, J. Zhang, X. Guo, and J. Yang, "A graph neural network based radio map construction method for urban environment," *IEEE Commun. Lett.*, vol. 27, no. 5, pp. 1327–1331, May 2023.

[10] C. Studer, S. Medjkouh, E. Gönültas, T. Goldstein, and O. Tirkkonen, "Channel charting: Locating users within the radio environment using channel state information," *IEEE Access*, vol. 6, pp. 47682–47698, 2018.

[11] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[12] A. Goldsmith, *Wireless Communications*, Cambridge University Press, New York, NY, 2005.

[13] H. Yu, Z. Hou, Y. Gu, P. Cheng, W. Ouyang, Y. Li, and B. Vucetic, "Distributed signal strength prediction using satellite map empowered by deep vision transformer," in *Proc. of the IEEE Globecom Workshops*, Madrid, Spain, Dec. 2021.

[14] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. of the Int. Conf. Mach. Learn. (ICML)*, 2020.

[15] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, et al., "Bootstrap your own latent-a new approach to self-supervised learning," in *Advances in Neural Info. Process. Syst. (NeurIPS)*, 2020, vol. 33, pp. 21271–21284.

[16] T. Wang and P. Isola, "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," in *Proc. of the Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 9929–9939.

[17] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Proc. of Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 776–794.

[18] Y. Zhang, H. Jiang, Y. Miura, C. D. Manning, and C. P. Langlotz, "Contrastive learning of medical visual representations from paired images and text," in *Proc. of the Machine Learning for Healthcare Conf. (MLHC)*, Durham, NC, Aug. 2022.

[19] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?," in *Proc. of the Int. Conf. Learn. Represent.*, virtual, Apr. 2022.

[20] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Info. Process. Syst. (NeurIPS)*, 2017.

[21] A. Alkhateeb, "DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications," *arXiv preprint arXiv:1902.06435*, 2019.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2017.

[23] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2017.