Millimeter Wave Beamforming Codebook Design via Learning Channel Covariance Matrices Over Riemannian Manifolds

IMTIAZ NASIM, (Student Member, IEEE), AND AHMED S. IBRAHIM, (Member, IEEE)
Department of Electrical and Computer Engineering, Florida International University, Miami, FL 33199, USA
Corresponding author: Imtiaz Nasim (inas001@fiu.edu)

This work was supported in part by the National Science Foundation under Award CNS-2144297.

ABSTRACT Covariance matrices of spatially-correlated wireless channels in millimeter wave (mmWave) vehicular networks can be employed to design environment-aware beamforming codebooks. Such covariance matrices can be represented over non-Euclidean (i.e., curved surfaces) manifolds, thanks to their symmetric positive definite (SPD) structures. In this paper, we propose three learning models for channel covariance estimation over Riemannian manifolds. First, we propose an unsupervised Riemannian K-means clustering approach, where Log-Euclidean metric (LEM) is utilized as a distance metric among channel covariance matrices over the Riemannian manifold. Second, we propose a Riemannian Competitive Learning (RCL) model, which is an online clustering solution that is intended to reduce the learning time of the offline K-means models. Third, we apply a Riemannian Dictionary Learning (RDL) model that leverages the sparsity properties of mmWave channels, while estimating their channel covariance matrices. Simulation results show that the proposed Riemannian K-means, online RCL and RDL models can achieve 41%, 30% and 44% higher data rate, respectively, than their Euclidean-based counterparts. Furthermore, they require few training samples and hence fast construction of codebook design in dynamic environments, which leads to low latency communication.

INDEX TERMS Competitive learning, covariance matrices, dictionary learning, K-means, Riemannian geometry.

I. INTRODUCTION
High data rate and low latency are crucial to support emerging applications, in vehicular networks, such as infotainment, safety information, and online route mapping messages [2], [3], [4]. While vehicular communication over millimeter Wave (mmWave) spectrum bands [5] has large bandwidths [6], which is a key enabler for high data rate, it also experiences large path loss. Dynamic deployment of mobile road side units (RSUs), or generally relays with large antenna arrays, in zones of temporary high data demand can partially compensate for such large path loss.

Equally important, constructing environment-aware beamforming codebooks at such mobile relays, which are matched to the spatial correlation [7] of the mmWave user channels within each zone, is fundamental to enabling faster (i.e., low latency) [8] high-rate communications. Otherwise, having a fixed or pre-defined beamforming codebooks for all relays will result in lower data rates, compared to what can be potentially achieved. Therefore, providing high-rate and low-latency vehicular communication can be accomplished through relay-assisted mmWave communication with environment-aware beamforming codebooks, and this is the motivation of this work.

Beamforming codebook for spatially-correlated channels can be designed to match their second order statistics (i.e., covariance matrices) [9], and hence there is a need to estimate channel covariance matrices. The problem of channel covariance estimation has been discussed in several studies. For example, the authors in [10] developed a low-rank...
matrix structure for covariance estimation in hybrid mmWave systems. The work in [11] considered a maximum-likelihood technique, which is based on pilot allocations, to estimate channel covariance matrices. The work in [12] studied the impact of imperfect channel covariance information at the receiver on the overall system performance. Also, a channel covariance estimation scheme for downlink channels in Massive MIMO was proposed in [13] based on the uplink covariance information. In addition, a compressive sensing-based covariance estimation for sparse mmWave channels was introduced in [14].

In an attempt to avoid assumptions of the aforementioned algorithms on the exact modeling (e.g., sparsity modeling) of the correlated channels, supervised deep learning models were utilized to learn channel covariance matrices in [15] and [16] or to directly learn the best beamforming codebook [17]. Moreover, a situation-aware channel covariance estimation was proposed in [18] for beamforming design in downlink MIMO systems. However, such deep learning models are a) supervised ones that require labeled training samples of the best codebooks, which are hard to acquire in new and dynamic relay-assisted environments and b) multi-layered complex ones that require large training samples, and hence longer latency, before establishing the best codebook. Therefore, we focus in this work on unsupervised (i.e., no labelled training [19]) and low-complexity (e.g., single-layer network) learning models to estimate the channel covariance matrices.

Intuitively, the complexity of any machine learning (ML) model can be reduced if the underlying structure of its dataset is known. Knowing that practical multiple-antenna channels are spatially correlated, they can be utilized to reduce the model complexity. More specifically, the covariance matrices of such channels are symmetric positive semi-definite ones. Applying positive shifts to these semi-definite matrices transform them into symmetric positive definite (SPD) ones, which can be modeled as points over Riemannian manifolds [1], [20], [21], which are particularly conic ones. Representing channel covariance matrices over Riemannian manifolds is the first novelty of this work, and it paves the way to employ simple learning algorithms for estimating such covariance matrices over Riemannian manifolds, which can be denoted as Riemannian-geometric learning models.

Riemannian geometry [22] has been recently considered in designing beamforming vectors [23], [24] and in other wireless problems such as link scheduling [25], [26]. In addition, non-Euclidean methods have been utilized in codebook design such as non-conic Riemannian manifolds (i.e., unitary, fixed rank) as in [27], and [28]. While these studies present novel non-Euclidean geometric concepts of beamforming and codebook designs, they have not been utilized in channel covariance estimation, which is the novel solution approach of this paper.

In this paper, we proposed three Riemannian-geometric learning models for channel covariance estimation as follows. First, we propose an unsupervised Riemannian K-Means model to cluster the instantaneous channel covariance matrices, which are collected at the mobile relay during initial transmission with its associated users in its newly-deployed area, into multiple distinct groups. Unlike the traditional Euclidean-based K-Means clustering model that takes vector inputs only [29], [30], [31], [32], the proposed Riemannian K-Means approach clusters the SPD matrix samples (i.e., instantaneous channel covariance matrices) over the Riemannian manifold. Appropriate metrics, such as Log-Euclidean metric (LEM) [33], is used to calculate the geodesic distance which is the shortest distance among the SPD covariance matrices over the manifold. We show that compared to a Euclidean-based benchmark, the proposed Riemannian K-Means can provide more accurate estimation of the covariance matrices, resulting in a higher achievable data rate.

As the aforementioned K-means clustering model depends on storing the entire needed data samples of instantaneous channel covariance matrices prior to clustering, it may require large latency before constructing its best-learned beamforming codebook. Therefore and secondly, we propose another Riemannian-geometric solution, namely online Riemannian Competitive Learning (RCL), which performs fast real-time clustering of channel covariance matrices over the Riemannian manifold. The online RCL solution is a quantization algorithm [34] that learns on the fly and avoids waiting for large number of channel samples by processing a single channel estimate at a time. Our study reveals that the faster estimation of online RCL comes at the cost of losing some estimation accuracy compared to the offline Riemannian K-Means.

While the above two solutions present novel Riemannian-geometric perspectives, they do not generally consider the sparse scattering nature of mmWave channels [14], [35]. In that regard, we point out that dictionary learning (DL) has been utilized as a powerful tool to represent sparse signals with few non-zero elements in many scientific research [36], [37]. Hence and third, we propose an unsupervised Riemannian Dictionary Learning (RDL) algorithm to leverage the sparsity of mmWave channels while learning their SPD matrices over Riemannian manifolds. Our results show that the RDL scheme is a highly-efficient learning model compared to the other solutions.

While traditional Euclidean-based solutions of the above mentioned approaches exist in the literature [38], [39], we propose their application in channel covariance clustering based on non-Euclidean geometry considering the Riemannian manifold. Overall, the three proposed models attempt to utilize their clustering accuracy to construct best environment-aware beamforming codebooks, which provides higher data rate compared to a fixed or uniform one, while requiring as few learning samples as possible and hence low latency. As was initially stated, together high data rate and low latency are needed in vehicular communication.
We highlight the contributions of this paper as follows:

- We model instantaneous channel covariance matrices, thanks to their SPD structure, as points over Riemannian manifolds, which enables the employment of unsupervised clustering approaches towards finding the statistical channel covariance matrices of a specific environment and hence constructing environment-aware beamforming codebooks for high-rate communication.
- We employ Riemannian K-Means clustering scheme for estimating statistical channel covariance matrices, which can achieve 41% higher data rate compared to its Euclidean K-means counterpart.
- We employ an online Riemannian Competitive Learning clustering, which makes faster estimation of the channel covariance matrices than the offline Riemannian K-Means, and provides 30% more data rate than a Euclidean K-means clustering solution.
- We also employ a Riemannian dictionary learning algorithm that exploits the sparse mmWave channel properties and achieve a data rate 44% higher than the Euclidean K-Means benchmark.

The list of abbreviations used in this paper are listed in Table 1. The rest of the paper is organized as follows. Section II provides some preliminary concepts on Riemannian geometry, Competitive learning and Dictionary learning. Section III introduces the system model with problem formulation. Section IV describes the proposed solutions. Section V explains our simulation setup and the results for the proposed solutions. Finally, we conclude the paper in Section VI.

TABLE 1. List of abbreviations.

| Abbreviation | Full Meaning |
|--------------|--------------|
| CL           | Competitive Learning |
| DL           | Dictionary learning |
| LEM          | Log-Euclidean metric |
| ML           | Machine learning |
| mmWave       | Millimeter-wave |
| MSO          | Multiple-input single-output |
| NR           | New Radios |
| NMSQ         | Normalised mean square error |
| RCL          | Riemannian competitive learning |
| RDL          | Riemannian dictionary learning |
| RSU          | Road side unit |
| SPD          | Symmetric positive definite |
| UE           | User equipment |

II. PRELIMINARIES

This section introduces some preliminary concepts on Riemannian geometry, competitive learning and dictionary learning solutions.

A. RIEMANNIAN MANIFOLDS

Topological manifolds represent non-Euclidean curved surfaces like $\Phi$ as shown in Fig. 1. A tangent space $T_\chi \Phi$ at $\chi$ is the composition of all the tangent vectors $v$ through the point $\chi$. An exponential map at $\chi$ is defined as the function mapping from the tangent space $T_\chi \Phi$ of the manifold $\Phi$ at point $\chi$ to the manifold $\Phi$ itself, i.e., $\exp_\chi : T_\chi \Phi \to \Phi$ [40]. Conversely, $\log_\chi$ is defined as the inverse of the exponential map $\exp_\chi$. Riemannian manifolds [22], [28] are equipped with Riemannian metrics. Riemannian metrics, such as LEM [33], are considered as inner products on the tangent space $T_\chi \Phi$ and measure the length of geodesics between any two points over the manifold. SPD matrices are modeled as points over the interior surface of conic (i.e., Riemannian) manifolds [21]. Leveraging the intrinsic Riemannian-geometric structure of data given by the SPD matrices opens up the possibilities for numerous research studies [40], [41]. In particular, the application of geometric learning solutions using SPD matrices can characterize the data in non-Euclidean domains, which is not possible with traditional learning algorithms that deal with Euclidean-structured data samples (i.e., vectors) [42], [43].

B. COMPETITIVE LEARNING

In a competitive learning (CL) framework, the output units of the algorithm compete over each other to win a specific input vector, and hence, the name Competitive learning. Such competition oriented solutions were studied in many vector classification problems [44], [45], [46]. The work in [44] provided a survey on CL algorithms for optimal vector quantization with its application in numerical probability and data mining. The authors in [45] presented an overview of several CL algorithms and their possible hardware implementation for computer architecture. Ahalt et al. in [46] proposed a CL solution and showed its application to design a vector quantizer for speech database. Also, the work in [47] studied the interpretation of a gradient descent based CL framework and investigated its relationship to fuzzy learning vector quantization.

Considering the unique capability of SPD matrices to represent the underlying non-Euclidean data structure, CL has been recently applied for clustering over Riemannian manifolds. For example, CL for clustering SPD matrices over Riemannian manifolds has been studied in [34] where a neural network is applied to perform the clustering task. Also,
the authors in [48] presented a novel CL based quantization algorithm over the Riemannian manifold for air traffic congestion analysis. However, we note that CL has not been previously considered for channel covariance estimation over Riemannian manifolds, as proposed in this paper.

C. DICTIONARY LEARNING

In the context of DL, a set of overcomplete basis known as a dictionary is used to present the essential information within a sparse signal with few nonzero elements [49]. In particular, DL is an efficient data representation tool to represent sparse data as a combination of basis vectors (or atoms) from a learned dictionary. Towards doing so, the DL solution finds a set of dictionary atoms that can accurately reconstruct the sparse data with minimum errors. In other words, the algorithm searches for the optimal vectors, known as atoms, that can be sparsely combined to represent each input data point of the given dataset. These atoms are obtained by factorizing the training data as a product of a dictionary and a sparse coefficient matrix [50]. The sparser the representation, the better will be the learned dictionary atoms. Consequently, the learned dictionary matrix is used to represent the signal as a linear combination of few nonzero dictionary atoms [51]. DL framework was studied in [36] to solve a pattern classification problem where the locality information of the atoms was preserved by employing a graph Laplacian matrix. In [37], the authors solved the DL problem for matrix factorization using alternating minimization.

Motivated by the huge success of DL algorithms over linear spaced Euclidean data samples [52], such solutions were applied for SPD matrices over Riemannian spaces in [41], [53], [54]. Authors in [41] represented non-Euclidean data in the form of SPD matrices as a sparse combination of SPD atoms by applying DL over Riemannian manifolds. The authors in [53] and [54] provided kernel based DL solutions considering Riemannian geometry for sparse coding in computer vision applications. Nevertheless, DL has not been studied previously for channel covariance matrices estimation, which differentiates our work from the existing literature.

III. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we briefly describe the system model along with the problem formulation for channel covariance estimation in a mmWave vehicular network.

A. SYSTEM DESCRIPTION

We consider a typical vehicular setting where a mobile road side unit (RSU) or a relay moves to new vehicular environments every-time, and gathers the wireless channels towards its dynamic UEs. We assume these channels between the RSU and the existing UEs are collected over $N$ (where $N = 1, \cdots, N$) communication sessions that resemble possible locations of the UEs. The covariance matrices of the newly collected channels are estimated at the RSU applying the clustering solutions. Finally, we apply matrix eigenvalue decomposition to construct an environment-aware beamforming codebook using the estimated covariance matrices.

Note that knowing the channel covariance matrices beforehand aids designing of efficient beamforming codebooks with higher data rates [9]. However, in a high-mobility vehicular scenario with dynamically deployed RSU as shown in Fig. 2, the covariance matrices of spatially-correlated channels are not known in advance because of their varying nature due to the relative position of RSU and its moving UEs. As a result, constructing fixed and predefined codebooks will be inefficient for such scenarios. Therefore, the codebook design needs to be learned based on the user’s correlated channels within each relay’s cluster and hence, requires an adaptive codebook [55].

![Covariance clustering at RSU](image)

**FIGURE 2.** A vehicular scenario with users located at different locations.

To construct an adaptive codebook via channel covariance estimation, we consider the RSU is equipped with $m$ antennas, and a single antenna is placed at the UEs. Let $h_i \in \mathbb{C}^{m \times 1}$ represents an $m \times 1$ wireless channel vector between the RSU and one of its UEs at the $i$-th location spot. The considered channel vectors $h_i$ are originated using the ray tracing model of an outdoor vehicular environment at 60 GHz [56]. No statistical model (e.g., circularly-symmetric complex Gaussian) was used to generate these vehicular channels.

A multiple-input single-output (MISO) channel between a given RSU and its UE is described as a correlated Rayleigh fading channel vector [20]. Calculating the spatially-correlated sample covariance matrices, $h_i h_i^H$ of these multi-antenna vehicular channels result in symmetric positive semi-definite matrices. Applying positive shifts to these matrices transform them into SPD ones according to the following

$$Q_i = \lambda I + h_i h_i^H,$$  \hspace{1cm} (1)

where $I$ is a $m \times m$ identity matrix, $\lambda$ is an arbitrary small scalar, and $H$ denotes matrix hermitian. We consider $\lambda = 0.5$ in this paper. The resulted positively-shifted sample covariance matrices (or simply SPD matrices) in (1) can be modeled as points on the conic Riemannian manifold [21]
over the locations \( i = 1, 2, \ldots, N \). Consequently, \( \Phi = \{Q_1, Q_2, \ldots, Q_N\} \) represents the set of observations for all the given SPD matrices that are to be estimated, where each observation within \( \Phi \) is an \( m \times m \) matrix.

### B. PROBLEM FORMULATION

In order to design an environment-aware beamforming codebook through channel covariance estimation, we aim to partition the collected samples of the shifted covariance matrices into \( K \leq N \) disjoint groups \( R_1, R_2, \ldots, R_K \), based on their distinct spatial correlation characteristics. The set of indices for the location points that belong to any group \( R_k \) is denoted as \( I_k = \{1, 2, \ldots, n_k\} \), where \( k = 1, 2, \ldots, K \). Also, we denote the channel vectors within group \( R_k \) as \( h_{i,k} = h_i, (\forall i \in I_k) \). For estimating the covariance matrices, the collected samples \( \{Q_1, Q_2, \ldots, Q_N\} \) over the \( i = 1, 2, \ldots, N \) location spots are formulated as the input to our non-Euclidean based clustering algorithms. The mean (i.e., centroid) is then computed for each group of samples based on their unique spatial-correlation characteristics. More specifically, for the considered groups of covariance matrices, the proposed clustering solutions seek to find the set of the estimated means given as \( \hat{G} = [\hat{G}_1, \hat{G}_2, \ldots, \hat{G}_K] \) over \( w \) iterations (where \( w = 1, \ldots, W \)). Matrix eigenvalue decomposition is applied over these estimated means of covariance matrices to construct the beamforming codebook.

### IV. LEARNING CHANNEL COVARIANCE MATRICES

In this section, we explain our proposed clustering solutions for estimating channel covariance matrices in a vehicular environment.

#### A. RIEMANNIAN K-MEANS CLUSTERING

We model the observations of instantaneous channel covariance matrices as points over the Riemannian manifold, and consider the LEM Riemannian metric [20], [33] to measure the geodesic distances among them. The proposed Riemannian K-Means clustering algorithm assigns an input sample \( Q_i \in \Phi \) to its nearest cluster \( R_k^* \) by calculating the optimal cluster index as

\[
 k^* = \arg\min_k d^2(Q_i, \hat{G}_k), \tag{2}
\]

where \( \hat{G}_k \) is the mean (i.e., centroid) for cluster \( R_k \) and \( k = 1, 2, \ldots, K \). The term \( d \) in (2) denotes the LEM geodesic distance. The centroids \( \hat{G}_k \) of the channel covariance matrices are given by the Fréchet mean (also known as Riemannian mean [57]) and is defined as the minimizer of sum of the squared distances according to [58]

\[
 \hat{G}_k = \arg\min_{G \in \Phi} \sum_{i=1}^{N} d^2(Q_i, G). \tag{3}
\]

The minimizer in (3) is actually obtained with a gradient descent procedure where from an initial value \( t = 0 \), the optimal centroid for group \( R_k \) is reached by repeatedly applying the following update formula according to [59]

\[
 \hat{G}_{k+1} = \exp_{\hat{G}_k} \left( \frac{1}{n_k} \sum_{i \in I_k} \log_{\hat{G}_k} (Q_i) \right), \quad k = 1, 2, \ldots, K, \tag{4}
\]

where \( n_k \) denotes the number of samples within group \( R_k \). Feeding more samples to the algorithm within a particular region of the RSU’s coverage allows for a more precise identification of that cluster center. Finally, the induced positive shift is subtracted to achieve the desired estimated centroid of the channel covariance matrices for cluster \( R_k \)

\[
 \hat{G}_k = \hat{G}_{k+1} - \lambda I. \tag{5}
\]

Fig. 3 shows an illustration of the proposed Riemannian K-Means solution for channel covariance clustering where the collected SPD covariance matrices are grouped into different clusters over a conic Riemannian manifold. The different shapes of the SPD points in Fig. 3 are used to represent a vehicular environment with distinct characters.

![Riemannian K-Means clustering for distinguishable SPD covariance matrices.](image)

The proposed Riemannian K-Means clustering algorithm is summarized in Algorithm 1.

#### Algorithm 1 Riemannian K-Means Clustering Algorithm

1. Specify the number of desired clusters \( K \).
2. Initialize the \( K \) centroids at random.
3. repeat
4. Assign the points \( Q_i \) (where \( i = 1, 2, \ldots, N \)) to their closest cluster \( R_k \) by calculating \( d \) in (2).
5. Recompute the centroid \( \hat{G}_k \) for cluster \( R_k \) (where \( k = 1, 2, \ldots, K \)).
6. until Centroid positions do not change.

#### B. ONLINE RIEMANNIAN COMPETITIVE LEARNING

The Riemannian K-Means algorithm requires large waiting time prior to conducting the clustering step, as it requires the RSU to collect all needed channel estimates from different locations within its coverage. To reduce such waiting time, we propose to apply online RCL, to cluster the channel covariance matrices over the Riemannian manifold. The online RCL is also an unsupervised clustering approach, but processes one channel estimate at a time (i.e., online)
without requiring to wait for all the channel estimates together [34], [48].

In order to cluster the overall observations \( \{ Q_1, Q_2, \ldots, Q_N \} \), the online RCL algorithm computes the mean geodesic error (MGE) [34] as

\[
MGE = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} d^2(Q_i, \hat{G}_k). \quad (6)
\]

The mean of the proposed online RCL algorithm is calculated using a gradient descent approach, where the update [48] is given by

\[
\hat{G}_k \leftarrow \exp_{G_k} \left( \frac{1}{N} \sum_{i=1}^{N} G_i Q_i \right). \quad (7)
\]

The solution follows the direction given in (7) at each step to update the cluster center \( \hat{G}_k \) for any new input \( Q_i \). Instead of going along straight linear lines, in the Riemannian context, we use the exponential mapping to follow the geodesic distance over the Riemannian manifold.

For making the clustering real-time by processing the SPD inputs one at a time, the online RCL method employs a competition among the cluster centers to win a particular input. The LEM is used to decide the winner which gets updated by advancing towards that input. More specifically, the algorithm calculates the LEM distance from each cluster center to that input, and the cluster center with the shortest LEM distance to that SPD input wins the competition and is updated by approaching towards that SPD point. This differentiates the online RCL clustering scheme from the one explained in Algorithm 1, where the offline Riemannian K-Means considers all the samples together for calculating the centroids.

It is worth noting that at each step, the online RCL algorithm only requires as many distance computations as the number of desired clusters, and solely the winning output is updated by advancing towards the input, while the rest remain unchanged. Finally, the cluster center that wins the competition for a given set of input points is used to represent those input SPD matrices. The algorithm for the online RCL clustering scheme is provided in Algorithm 2.

**Algorithm 2 Online RCL Clustering Algorithm**

**Input:** \( \{ Q_1, Q_2, \ldots, Q_N \} \) collected from \( \Phi \)

**Output:** \( \{ \hat{G}_1, \hat{G}_2, \ldots, \hat{G}_K \} \) for \( K \) clusters

1. \( t \leftarrow 0 \), select \( \{ \hat{G}_1(0), \hat{G}_2(0), \ldots, \hat{G}_K(0) \} \) at random
2. **repeat**
   1. \( k \leftarrow \arg\min_{k} d^2(Q_i(t+1), \hat{G}_k) \) for winner cluster center,
   2. \( \hat{G}_{k_{t+1}} \leftarrow \exp_{\hat{G}_k} (\hat{G}_k Q_i(t+1)) \) for else
   3. \( \hat{G}_{j_{t+1}} \leftarrow \hat{G}_j, \forall j \neq k \)
   4. \( t \leftarrow t + 1 \)
3. **until** cluster centers do not change.

\( \{ R_1, \ldots, R_k \} \leftarrow \text{cluster center} \{ \hat{G}_1, \ldots, \hat{G}_K \} \)

C. DICTIONARY LEARNING FOR SPARSE CHANNELS

Through fewer RF chains than antennas, the mmWave channel estimator observes a reduced dimensional representation of the entire signal. Such sparse nature of the mmWave channels has been indicated in [14]. Consequently, the covariance matrices resulting from these mmWave channels develop sparse SPD matrices. Therefore, we point out the necessity of a learning model that involves sparse mmWave channel properties while estimating their statistical covariance matrices. Our proposed RDL solution clusters the instantaneous covariance matrices by learning a dictionary of SPD atoms in a way that the distinct features of the covariance matrices are preserved within a few atoms. More specifically, the given solution seeks a dictionary \( D = \{ D_1, D_2, \ldots, D_K \} \) of few basis atoms such that each data point \( Q_i \in \Phi \) can be represented as a combination of these atoms. This implies that \( Q_i \sim \sum_{k=1}^{K} D_i \alpha_{i,k} \), where \( \alpha_i = \{ \alpha_{i,1}, \ldots, \alpha_{i,K} \} \in \mathbb{R}^k \). We formulate the objective of the proposed RDL model as

\[
\min_{D \in \Phi, \alpha_{i,k} \in \mathbb{R}} \frac{1}{N} \sum_{i=1}^{N} d^2(Q_i, D_k \alpha_{i,k}). \quad (8)
\]

The obtained dictionary atoms \( D \) by the proposed RDL solution are treated as centroids \( \hat{G} \) of the channel covariance clusters. We summarize the proposed RDL algorithm in Algorithm 3.

**Algorithm 3 RDL Algorithm**

1. Specify required number of dictionary atoms \( K \).
2. Initialize \( D \) by randomly selecting \( K \) samples from \( \{ Q_1, Q_2, \ldots, Q_N \} \) over \( \Phi \), and set \( \alpha_{i,k} \geq 0 \)
3. **repeat**
   1. Read the sparse samples from \( \Phi \)
   2. Calculate \( d \) from \( Q_i \) \{ where \( i = 1, 2, \ldots, N \) \} to dictionary atoms \( \{ D_1, D_2, \ldots, D_K \} \)
   3. For each point \( Q_i \), update the dictionary atoms \( D_k \) \{ where \( k = 1, 2, \ldots, K \) \} to minimize (8).
4. **until** Dictionary atoms do not change.

D. EUCLIDEAN K-MEANS CLUSTERING

In order to show the potential advantages for representing channel covariance matrices over Riemannian manifolds, we also propose an Euclidean-based K-Means clustering approach. The Euclidean K-means algorithm treats the channel vectors \( h_i \) as inputs and uses their linear Euclidean distances to compute the centroids given by the arithmetic
mean [60]. Unlike our proposed solutions that cluster channel covariance matrices over the Riemannian manifold, the Euclidean-based K-Means attempt to make $K$-partitions of the channel vectors by applying the following minimization formula [61]

$$\hat{h}_k = \frac{1}{K} \sum_{i \in k} \| h_i - \hat{h}_k \|^2,$$

where $\hat{h}_k$ denotes the centroid obtained by the Euclidean-based K-Means clustering model for group $R_k$. The algorithm then calculates covariance matrices of those estimated channels for each group $R_k$ as

$$\hat{G}_k = \hat{h}_k \hat{h}_k^H,$$

where $\hat{G}_k$ is the covariance matrix for centroid $\hat{h}_k$. Finally, the sample covariance matrices are estimated from the ones given in (10).

The Euclidean-based K-Means algorithm follows the same methodology provided in Algorithm 1, with the exception that it computes the linear distances between the channel vectors over the Euclidean space.

V. SIMULATION RESULTS

In this section, we introduce the different datasets that are considered for simulation along with the performance evaluation.

A. SIMULATION SETUP

Our work assumes a vehicular network operating at 60 GHz with 0.5 GHz bandwidth. A $4 \times 4$ phased array is considered at the RSU. We adopt three different vehicular datasets from the “O1 – 60 ray tracing scenario” [56] along with their channel models. The purpose of examining with different datasets is to observe the efficacy of the proposed schemes for various vehicular settings.

The area of interest under Dataset-1 considers 5460 user location points from user grid-1 of the “O1 – 60 ray tracing scenario” and is demonstrated in Fig. 4 (a). For Dataset-2, our work assumes a total of 10860 location points from user grid-1 as shown in Fig. 4 (b). The considered area under Dataset-3 is the largest one which consists of a total of 16800 samples from user grid-1, as shown in 4 (c). The distance between any two adjacent points is 20 cm [56] which is constant for all the considered scenarios. The time required for a vehicular UE to cover any of these regions can be determined using simple calculations. For instance, the Dataset-3 with 16800 location spots covers a distance of 3360 meters. A vehicle travelling at a speed of 30 mph (i.e., 48.2 km/h) will require almost 250 seconds or 4.2 minutes to cover this region. On the other hand, a vehicle moving at a speed of 63 mph (or 101.3 km/h) supported by the 5G NR C-V2X [62] will require approximately 120 seconds or 2 minutes to cover the considered area under Dataset-3. Considering such a dynamic vehicular environment makes it difficult to model it using a specific statistical model or with specific parameters (e.g., circularly symmetric zero-mean complex Gaussian with a specific variance.)

B. CODEBOOK DESIGN

The spatial-correlation characteristics of the multiple-antenna channels can be utilized to construct matched beamforming codebooks [9], [63]. Matrix eigenvalue decomposition or spectral decomposition has been used previously for optimal precoder design [64]. Following these principles, the cluster centers $\hat{G}_k$ obtained by our proposed solutions are factorized using their spectral decomposition

$$\hat{G}_k = U_k A_k U_k^H, \quad (k = 1, 2, \ldots K),$$

where $U_k$ and $A_k$ are the $m \times m$ unitary and diagonal matrix, respectively. Then similar to the previous studies in [64] and [65], we simply choose the best eigenvector corresponding to the largest eigenvalue from the unitary matrix $U_k$. More specifically, for a given cluster center $\hat{G}_k$ for group $R_k$, we select the first column of $U_k$, which refers to the strongest vector from the estimated covariance matrices.

C. PERFORMANCE EVALUATION

We use the normalized mean square error (NMSE) as the considered loss function to evaluate the performance of the proposed solutions. For any learning scheme, the NMSE for learning the mmWave channel covariance matrices is calculated as

$$\text{NMSE}_k = \frac{||\hat{G}_k - G_k||^2_F}{||G_k||^2_F}, \quad (k = 1, 2, \ldots, K)$$

where $G_k = \mathbb{E}[h_{i,k} h_{i,k}^H]$ is the true expected value of covariance matrices over the location $i$ for group $R_k$, and $||.||_F$ denotes matrix Frobenius norm operator. It is obvious from (12) that accurate estimation of the cluster centers $\hat{G}_k$ will make it as close to the expected value $G_k$, and hence lower NMSE, which is the goal of our non-Euclidean clustering solutions towards designing the codebook. The average NMSE over all the given clusters is denoted as

$$\text{NMSE} = \frac{1}{K} \sum_{k=1}^{K} \text{NMSE}_k.$$
and RDL scheme with the Euclidean K-Means benchmark. The proposed Riemannian K-Means clustering, which is an offline quantization algorithm for estimating channel covariance matrices, yields slightly higher NMSE than the RDL scheme, but performs better than the online RCL and the Euclidean K-Means benchmark. This is because the online RCL algorithm can only update the cluster centers in a localized manner, but the offline Riemannian K-Means algorithm updates the cluster centers with full knowledge of the dataset in a global manner, resulting in higher accuracy. The result in Fig. 5 is supported by the previous studies where the offline solution outperforms the corresponding online clustering solutions [68], [69], but can only make the estimation after collecting all samples.

Furthermore, Fig. 5 shows that the proposed RDL solution achieves the least NMSE compared to the other solutions as it can efficiently exploit the sparse mmWave channel properties. This is in agreement with [41], which shows that the Riemannian DL solution slightly outperforms the Riemannian K-Means method in terms of accuracy. As shown in Fig. 5, all of the non-Euclidean based Riemannian-geometric strategies outperform the Euclidean K-Means benchmark for any given number of clusters. For example, while the Euclidean K-Means approach produces an average NMSE of $3.5 \times 10^{-3}$ with 4 clusters, the Riemannian K-Means provides a substantially lower NMSE value of $7 \times 10^{-4}$. In other words, the proposed Riemannian K-Means can provide 80% higher accuracy compared to the corresponding Euclidean-based benchmark. For the same number of clusters, our online RCL achieves an NMSE value of $2 \times 10^{-3}$ making it 42% more accurate than the Euclidean-based K-Means strategy. In addition, our proposed RDL solution has an NMSE of $4.8 \times 10^{-4}$ for four dictionary atoms, resulting in 86% higher accuracy than the Euclidean K-Means benchmark for learning channel covariance matrices in a mmWave network.

In comparison with recent literature (e.g., [15]), we consider Dataset-2 along with the scenario of a single cluster (i.e., $K = 1$). Table 2 shows that the Euclidean-based supervised Generative Adversarial Network (GAN) model of [15] (with NMSE of 0.004) has less NMSE compared to the Euclidean-based unsupervised K-Means model (with NMSE of 0.063), which is due to using labeled learning while training the GAN model. Table 2 also shows that all of the proposed non-Euclidean-based unsupervised models (i.e., Riemannian K-means, Online RCL, and RDL) achieve less NMSE than the Euclidean-based GAN. Such comparison shows the advantages of our proposed non-Euclidean and unsupervised clustering approaches, which estimate channel covariance matrices over SPD manifolds as opposed to a Euclidean space.

**TABLE 2.** NMSE comparison.

| Scheme            | NMSE value |
|-------------------|------------|
| Euclidean K-Means | 0.063      |
| GAN [15]          | 0.004      |
| Online RCL        | 0.0039     |
| Riemannian K-Means| 0.0025     |
| RDL               | 0.0023     |

One of the main differences between clustering over Riemannian manifold versus Euclidean spaces is the mathematical expression of the clusters’ centroids. On one hand for the proposed non-Euclidean (i.e., Riemannian) approach, the centroid of each cluster is computed via the Fréchet mean of the instantaneous covariance matrices within it. Such Fréchet mean is in fact the corresponding geometric mean [70] of the instantaneous covariance matrices. On the other hand, the centroid of each cluster in the conventional Euclidean-based clustering approach is the arithmetic mean of the instantaneous channel vectors within it. Generally, the geometric mean of a group of numbers is less impacted by extreme values within such group compared to their arithmetic mean [71]. We believe that such intrinsic advantage,
of having low impact for out-of-range values in the geometric (or Fréchet) mean, contributes towards having better performance (i.e., less clustering error) for Riemannian-based clustering approaches compared to Euclidean-based benchmarks. In other words, Riemannian-based clustering is less susceptible to out-of-range instantaneous channels (i.e., outliers) compared to Euclidean-based clustering. Consequently, Riemannian clustering results in less NMSE thanks to finding centroids (i.e., geometric means), which are well-positioned within close-by instantaneous channels that happen with high frequency.

Fig. 6 shows the convergence of the three proposed clustering algorithms that are used to estimate the channel covariance matrices. As shown, the three clustering models converge to low values of NMSE, as that previously shown in Fig. 5, with a few iterations.

The results comparing the average NMSE between Riemannian K-Means and traditional Euclidean-based K-Means for all the considered Dataset-1, 2 and 3 is demonstrated in Fig. 7. This result deeply analyses the performance of a non-Euclidean based Riemannian-geometric solution to a Euclidean-based benchmark for estimating channel covariance matrices. It can be noted that the average NMSE is less while having a total of 5460 location points for Dataset-1, but it increases with the size of observations for Dataset-2 and 3. This is intuitive because increasing the number of location points will increase the NMSE. However, the proposed Riemannian K-Means clustering approach always performs better compared to its corresponding Euclidean-based strategy for any given data size. These results indicate that clustering channel covariance matrices directly over non-Euclidean Riemannian manifolds is more accurate compared to clustering the channel vectors based on their linear Euclidean distances. Note that both of the simulated cases in Fig. 7 apply unsupervised learning and require no prior labelled training for clustering channel covariance matrices.

Table 3. Data size versus optimal cluster.

| Data Size (samples) | Required cluster |
|---------------------|------------------|
| Dataset-1 (5460)    | 2                |
| Dataset-2 (10860)   | 4                |
| Dataset-3 (16800)   | 6                |

We further analyze the performance of the proposed non-Euclidean based solutions with different group sizes for estimating channel covariance matrices in Fig. 8. Towards doing so, we looked at Dataset-2 and separated it into four disjoint groups. First, we consider 500 location points in each group (i.e., $4 \times 500 = 2000$ samples from Dataset-2), and then gradually increase the group size to 2715 location points in each cluster which is equivalent to considering the entire Dataset-2 with 10860 location points. As shown, reducing the samples from 2715 to 2000, the NMSE value provided by the Riemannian K-Means clustering solution increases to $8 \times 10^{-4}$ from $7 \times 10^{-4}$. Similar behavior is observed for the other two clustering schemes. In particular, considering enough samples from the considered region allows the proposed solutions to make precise estimation of the cluster centers which leads to lower NMSE.

2) ACHIEVABLE THROUGHPUT PERFORMANCE

We compare the achievable throughput of the proposed learning solutions with two benchmarks. First, the uniform
The codebook is the one that picks the best beam for every UE locations from a fixed predefined codebook. Second, the Euclidean K-Means serves as a traditional benchmark of the Euclidean solution that was considered in Fig. 5 and Fig. 7. The transition from estimation accuracy towards codebook performance is described in two folds- (i) first, we show the trade-off between low latency (faster design) and achievable rate, then (ii) we consider fixed latency and compare the achievable throughput for different codewords (i.e., beamforming vectors).

The methodology explained in Section V-B is adopted to construct an adaptive environment-aware beamforming codebook from the estimated cluster centers. Fig. 9 demonstrates the trade-off between number of available samples (i.e., faster codebook design) and achievable data rate considering Dataset-2. We consider 4 codewords in the designed codebook for each of the schemes shown in Fig. 9. As shown in Fig. 9, reducing the data size costs some sacrifice in the achievable rate of the given learning solutions, which is a result of the estimation accuracy in Fig. 8. However, the rate achieved by our three non-Euclidean clustering approaches outperform the Euclidean benchmark for any sample size. In addition, our proposed solutions can achieve a better data rate than the uniform codebook benchmark that uses fixed codewords.

Considering Dataset-2, Fig. 10 compares the possible throughput for each strategy while taking into account different codewords. As was previously demonstrated in Fig. 5, the segmentation of a given dataset into certain particular clusters is what led to the different number of codewords (or simply beams) in the designed codebook. As shown in Fig. 10, the proposed Riemannian-geometric based solutions provide higher throughput than the Euclidean-based K-Means and uniform codebook benchmarks even for different codeword assumption. Also, the proposed RDL scheme always provides the highest throughput among the considered scenarios, thanks to its higher accuracy in estimating channel covariance matrices as shown earlier.

For a detailed comparison about the achievable throughput, we focus on the case with 4 codewords for all the considered scenarios in Fig. 10. For having 4 unique codewords, our RDL, Riemannian K-Means and online RCL clustering solutions yield an achievable rate of 4.62 bps/Hz, 4.53 bps/Hz, and 4.17 bps/Hz respectively. In the case for Euclidean K-Means and uniform codebook benchmark, the maximum data rate is 3.20 bps/Hz and 4.10 bps/Hz, respectively. As a result, our findings reveal that when compared to the Euclidean-based K-Means strategy, our proposed Riemannian K-Means enhances the data rate by more than 41%. While compared to the uniform codebook solution, the Riemannian K-Means increases the throughput by approximately 10%. We see that the online RCL solution can also provide approximately 30% higher data rate than the Euclidean-based K-Means benchmark. Finally, the proposed RDL solution provides 44% higher rate than the Euclidean K-Means and 13% higher rate than the uniform codebook, which is a result of its highest estimation accuracy in Fig. 5. Consequently, accurate estimation of
channel covariance matrices over Riemannian manifold leads to increased throughput in the codebook performance.

3) COMPLEXITY COMPARISON

Fig. 11 depicts the running time of different clustering approaches as it varies with the potential number of clusters. The shown clustering approaches were all implemented using the same Python platform [66], [67] with an Intel(R) i7-4770 CPU, 3.4 GHz and 16 GB RAM configuration. As shown, the online RCL consumes less running time compared to the offline Riemannian K-means. For example, at 4 clusters, the online RCL has 4.6% saving in running time compared to the Riemannian K-Means approach. Fig. 11 also shows that Euclidean K-means requires less running time than Riemannian-based approaches. Contrasting Fig. 11 with Fig. 5, we find that requiring more running time results in lower NMSE (i.e., higher accuracy.)

Table 4 shows the computational complexity of various clustering approaches for each input sample (i.e., assuming 1 channel input). For each of the clustering approaches, $K$ denotes the number of clusters, $W$ is the number of iterations for each clustering approach to find the centroids, and $m$ is the dimension of the channel vectors. For a total of $N$ samples, the complexities for all the clustering models in Table 4 are scaled by the factor $N$. On one hand and as presented in Table 4, the Riemannian K-means has complexity of $O(WKm^2)$, which is mainly due to handling $m \times m$ dimensional SPD matrices [72]. Similarly, the online RCL and RDL have the same computational complexity [75] of $O(WKm^2)$, thanks to clustering $m \times m$ SPD matrices. On the other hand, the Euclidean K-Means has lower computational complexity of $O(Wkm)$ [74], thanks to considering only $m$-dimensional channel vectors. In summary and on one hand, the Riemannian-based clustering approaches (i.e., Riemannian K-Means, Online RCL, and RDL) achieve less NMSE (Fig. 5) and higher data rate (Fig. 9) compared to Euclidean-based K-Means. On other hand, such Riemannian-based clustering approaches require more running time (Fig. 11) and computational complexity (Table 4) compared to Euclidean-based approaches.

4) IMPACT OF ANTENNA PARAMETERS

We further investigate the effect of various antenna parameters on NMSE in Fig. 12. We choose Riemannian K-Means as a candidate of our Riemannian-geometric approach, and study how antenna parameters affect estimation accuracy while considering different antenna elements at the transmitter. As shown in Fig. 12, with a phased array of $2 \times 2$ antenna elements, the proposed Riemannian K-Means solution provides an NMSE of $1 \times 10^{-3}$ for 4 clusters, while the scheme provides an NMSE value of $1 \times 10^{-3}$ with $4 \times 2$ antenna elements, and performs even better while considering $4 \times 4$ phased arrays with an NMSE of $7 \times 10^{-4}$. This implies that the proposed solution achieves approximately 128% higher accuracy when the number of antenna elements at the transmitter is increased from 4 to 16. Therefore, adopting larger phased arrays at the transmitter can help in estimating channel covariance matrices more accurately.

VI. CONCLUSION

In this paper, we have proposed an unsupervised Riemannian K-Means clustering scheme for estimating channel covariance matrices in a mmWave vehicular environment. The proposed solution is novel as it uses the underlying Riemannian-geometric structures of spatially-correlated wireless channels while learning their covariance matrices and does not require any prior labelled data for training. We have shown that the proposed non-Euclidean based Riemannian K-Means approach can provide 80% higher accuracy compared to the corresponding Euclidean-based K-Means benchmark. In addition, the achievable rate by the designed codebook for the Riemannian K-Means solution outperforms the Euclidean-based benchmark by more
than 41%. We have provided another non-Euclidean clustering solution, namely online RCL, where the algorithm makes faster estimation by applying real-time clustering and can provide 30% higher rate compared to a Euclidean-based scheme. Finally, we have shown that our third proposed Riemannian-geometric solution, the RDL, exploits the mmWave sparse channel properties while learning their covariance matrices and can provide 44% higher achievable data rate than the Euclidean-based K-Means solution. Furthermore, our proposed clustering solutions use few training samples for faster codebook construction, which leads to low latency communication, with minimal loss in throughput.

**ACKNOWLEDGMENT**

An earlier version of this paper was presented at the Geo-


dynamics for intelligent transportation systems,” Sensors, vol. 18, no. 4, p. 1212, Apr. 2018.

[9] V. Raghavan, R. W. Heath, and A. M. Sayeed, “Systematic codebook design for quantized beamforming in correlated MIMO channels,” IEEE J. Sel. Areas Commun., vol. 25, no. 7, pp. 1298–1310, Sep. 2007.

[10] R. Xu, J. Zhang, X. Tao, A. Alkhateeb, and S. Ma, “Deep learning based channel covariance matrix estimation with scene images,” in Proc. IEEE/CIC Int. Conf. Commun. China (ICCC), Jul. 2021, pp. 162–166.

[11] Y. Zhang, M. Ablalbeiah, and A. Alkhateeb, “Learning beam codebooks with neural networks: Towards environment-aware mmWave MIMO,” 2020, arXiv:2002.10669.

[12] A. Taha and A. Alkhateeb, “Situation-aware channel covariance prediction for deep learning aided massive MIMO systems,” in Proc. 54th Asilomar Conf. Signals, Syst., Comput., Nov. 2020, pp. 1342–1346.

[13] A. Joby. (2021). Unsupervised Learning: How Machines Learn on Their Own. Accessed: Apr. 15, 2021. [Online]. Available: https://learn2g.com/unsupervised-learning

[14] A. S. Ibrahim, “Rethinking maximum flow problem and beamforming design through brain-inspired geometric lens,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2020, pp. 1–6.

[15] S. Rra and R. Hosseini, “Conic geometric optimization on the manifold of positive definite matrices,” SIAM J. Optim., vol. 25, no. 1, pp. 713–739, Jan. 2015.

[16] S. Gudmundsson, An Introduction to Riemannian Geometry. Lund, Sweden: Lund Univ., 2014.

[17] W. Fan, C. Zhang, and Y. Huang, “Flat beam design for massive MIMO systems via Riemannian optimization,” IEEE Wireless Commun. Lett., vol. 8, no. 1, pp. 301–304, Feb. 2019.

[18] F. Chen, “Low-PAR precoding design for massive multiuser MIMO systems via Riemannian manifold optimization,” IEEE Commun. Lett., vol. 21, no. 4, pp. 945–948, Apr. 2017.

[19] A. S. Ibrahim, “Wireless link scheduling via interference-aware symmetric positive definite connectivity manifolds,” in Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops), Jun. 2021, pp. 1–5.

[20] R. Sheshlin and A. S. Ibrahim, “Geometric machine learning over Rie-

mannian manifolds for wireless link scheduling,” IEEE Access, vol. 10, pp. 22854–22864, 2022.

[21] G. Arvanitidis, S. Hauberg, P. Hennig, and M. Schobinger, “Fast and robust shortest paths on manifolds learned from data,” in Proc. 22nd Int. Conf. Artif. Intell. Statist., vol. 89, 2019, pp. 1506–1515.

[22] X. Yu J., J. Shen, J. Zhang, and K. B. Letaief, “Alternating minimization algorithms for hybrid precoding in millimeter wave MIMO systems,” IEEE J. Sel. Topics Signal Process., vol. 10, no. 3, pp. 485–500, Apr. 2016.

[23] C. Levrad, “Quantization/clustering: When and why does K-means work?” Jan. 2018, arXiv:1801.03742.

[24] N. Wong, D. Kim, Z. Robinson, C. Huang, and I. M. Conboy, “K-means quantization for a web-based open-source flow cytometry analysis platform,” Sci. Rep., vol. 11, no. 1, Dec. 2021.

[25] K. He, F. Wen, and J. Sun, “K-means hashing: An affinity-preserving quantization method for learning binary compact codes,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2013, pp. 2938–2945.

[26] K. P. Sinaga and M.-S. Yang, “Unsupervised K-means clustering algo-

rithm,” IEEE Access, vol. 8, pp. 80716–80727, 2020.

[27] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, “Log-Euclidean metrics for fast and simple calculus on diffusion tensors,” Magn. Reson. Med., vol. 56, no. 2, pp. 411–421, 2006.

[28] L. Zheng, G. Qiu, and J. Huang, “Riemannian competitive learning for symmetric positive definite matrices clustering,” Neurocomputing, vol. 295, pp. 153–164, Jun. 2018.

[29] R. Wang, M. Pajovic, P. V. Orlak, T. Koike-Akino, K. J. Kim, and J. Fang, “Sparse channel estimation in millimeter wave communications: Exploiting joint AoD-AoA angular spread,” in Proc. IEEE Int. Conf. Commun. (ICC), May 2017, pp. 1–6.

[30] H.-F. Yin, X.-J. Wu, and S.-G. Chen, “Locality constraint dictionary learning with support vector for pattern classification,” IEEE Access, vol. 7, pp. 175071–175082, 2019, doi: 10.1109/ACCESS.2019.2957417.

[31] M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, “Dictionary learning for sparse representation: A novel approach,” IEEE Signal Process. Lett., vol. 20, no. 12, pp. 1195–1198, Dec. 2013, doi: 10.1109/LSP.2013.2285218.

[32] R. A. Haraty, M. Dimishkieh, and M. Masud, “An enhanced k-means clustering algorithm for hybrid precoding in millimeter wave MIMO systems,” IEEE Trans. Veh. Technol., vol. 65, no. 1, pp. 106–112, Jan. 2016.

[33] R. A. Haraty, M. Dimishkieh, and M. Masud, “An enhanced k-means clus-
tering algorithm,” IEEE Trans. Veh. Technol., vol. 64, no. 7, pp. 2760–2767, Sep. 2015.

[34] R. A. Haraty, M. Dimishkieh, and M. Masud, “An enhanced k-means clus-
tering algorithm,” IEEE Trans. Veh. Technol., vol. 64, no. 7, pp. 2760–2767, Sep. 2015.

[35] P. Wang, M. Pajovic, P. V. Orlak, T. Koike-Akino, K. J. Kim, and J. Fang, “Sparse channel estimation in millimeter wave communications: Exploiting joint AoD-AoA angular spread,” in Proc. IEEE Int. Conf. Commun. (ICC), May 2017, pp. 1–6.

[36] H.-F. Yin, X.-J. Wu, and S.-G. Chen, “Locality constraint dictionary learning with support vector for pattern classification,” IEEE Access, vol. 7, pp. 175071–175082, 2019, doi: 10.1109/ACCESS.2019.2957417.

[37] M. Sadeghi, M. Babaie-Zadeh, and C. Jutten, “Dictionary learning for sparse representation: A novel approach,” IEEE Signal Process. Lett., vol. 20, no. 12, pp. 1195–1198, Dec. 2013, doi: 10.1109/LSP.2013.2285218.

[38] R. A. Haraty, M. Dimishkieh, and M. Masud, “An enhanced k-means clustering algorithm for hybrid precoding in millimeter wave MIMO systems,” IEEE Trans. Veh. Technol., vol. 65, no. 1, pp. 106–112, Jan. 2016.
I. Nasim, A. S. Ibrahim: Millimeter Wave Beamforming Codebook Design via Learning Channel Covariance Matrices

[41] A. Cherian and S. Sr, “Riemannian dictionary learning and sparse coding for positive definite matrices,” IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 12, pp. 2859–2871, Dec. 2017, doi: 10.1109/TNNLS.2016.2601307.

[42] F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein, “Geometric deep learning on graphs and manifolds using mixture model CNNs,” 2016, arXiv:1611.08492.

[43] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric deep learning: Going beyond Euclidean data,” IEEE Signal Process. Mag., vol. 34, no. 4, pp. 18–42, Jul. 2017.

[44] G. Pagès, “Introduction to vector quantization and its applications for numerics,” EXAII: Proc. Surveys, pp. 29–79, Jan. 2015.

[45] H. C. Card, G. K. Rosendahl, D. K. McNeill, and R. D. McLeod, “Competitive learning algorithms and neurocomputer architecture,” IEEE Trans. Comput., vol. 47, no. 8, pp. 847–858, Aug. 1998.

[46] S. C. Ahalt, A. K. Krishnamurthy, P. Chen, and D. E. Melton, “Competitive learning algorithms for vector quantization,” Neural Netw., vol. 3, no. 3, pp. 277–290, 1990.

[47] N. B. Karayiannis and J. C. Bezdek, “An integrated approach to fuzzy learning vector quantization and fuzzy C-means clustering,” IEEE Trans. Fuzzy Syst., vol. 5, no. 4, pp. 622–628, Nov. 1997.

[48] A. L. Brigant and S. Puechmorel, “Optimal Riemannian quantization with an application to air traffic analysis,” J. Multivariate Anal., vol. 173, pp. 686–703, Sep. 2020.

[49] X. Chen, Z. Du, J. Li, X. Li, and H. Zhang, “Compressed sensing based on dictionary learning for extracting impulse components,” Signal Process., vol. 96, pp. 94–109, Mar. 2014.

[50] A. Mensch, J. Mairal, B. Thirion, and G. Varoquaux, “Dictionary learning based subject-independent classification of motor imagery?” in Proc. 18th Workshop Algorithm Eng. Exp. (ALENEX), Jan. 2016.

[51] T. Guha and R. K. Ward, “Learning sparse representations for human action recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 8, pp. 1576–1588, Aug. 2012, doi: 10.1109/TPAMI.2011.253.

[52] I. Tosi and P. Frossard, “Dictionary learning,” IEEE Signal Process. Mag., vol. 28, no. 2, pp. 27–38, Mar. 2011.

[53] M. T. Harandi, C. Sanderson, R. Hartley, and B. C. Lovell, “Sparse coding and dictionary learning for semantic positive definite matrices: A kernel approach,” 2013, arXiv:1304.4344.

[54] P. Li, Q. Wang, W. Zuo, and L. Zhang, “Log-Euclidean kernels for sparse representation and dictionary learning,” in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2013, pp. 1601–1608, doi: 10.1109/ICCV.2013.202.

[55] B. Clerckx and C. Oestges, space-time Coding With Partial Transmit Channel Knowledge. Oxford, U.K.: Oxford Univ. Press, 2013, ch. 10, pp. 335–384, [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780123850553000109

[56] A. Alkhateeb, “DeepMIMO: A generic deep learning dataset for millimeter wave and massive MIMO applications,” in Proc. Inf. Theory Appl. Workshop (ITA), San Diego, CA, USA, Feb. 2019, pp. 1–8.

[57] Y. Kainolda, B. Abibullaev, R. Sameni, and A. Zollanvari, “Is Riemannian geometry better than Euclidean in averaging covariance matrices for CSP-based subject-independent classification of motor imagery?”, in Proc. 43rd Ann. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC), Nov. 2021, pp. 910–914.

[58] Y. Shimohara, T. Masuko, and M. Akamine, “Covariance clustering on Riemannian manifolds for acoustic model compression,” in Proc. IEEE Int. Conf. Acoust.. Speech Signal Process., Mar. 2010, pp. 4326–4329.

[59] X. Pempec, P. Fillard, and N. Ayache, “A Riemannian framework for tensor computing,” INRIA, France, Tech. Rep. RR-5255, 2004, p. 34.

[60] R. Shang, B. Ara, I. Zada, S. Nazir, Z. Ulah, and S. U. Khan, “Analysis of simple K-mean and parallel K-mean clustering for software products and organizational performance using education sector dataset,” Sci. Program., vol. 2021, pp. 1–20, May 2021.

[61] M. Aalto and N. Verma, “Metric learning on manifolds,” 2019, arXiv:1902.01738.

[62] (2021), C-V2X Technical Performance-Frequently Asked Questions. Qualcomm Tech. Accessed: Oct. 2, 2021. [Online]. Available: https://www.qualcomm.com/media/documents/files/c-v2x-technical-performance-faq.pdf

[63] D. J. Love and R. W. Heath, “Limited feedback diversity techniques for correlated channels,” IEEE Trans. Veh. Technol., vol. 55, no. 2, pp. 718–722, Mar. 2006.

[64] S. Zhou and G. B. Giannakis, “Optimal transmitter Eigen-beamforming and space-time block coding based on channel correlations,” IEEE Trans. Inf. Theory, vol. 49, no. 7, pp. 1673–1690, Jul. 2003, doi: 10.1109/TIT.2003.813565.

[65] M.-O. Pun, R. Portor, P. Orlik, J. Zhang, and T. Kuz, “Codebook-based quantized MIMO feedback for closed-loop transmit precoding,” in Proc. Conf. Rec. 43rd Asilomar Conf. Signals, Syst. Comput., 2009.

[66] M. Niolane, J. Mathe, C. Donnat, M. Jorda, and X. Pennec. (2018). Geomstats: A Python Package for Riemannian Geometry in Machine Learning. [Online]. Available: https://openreview.net/forum?id=rklXaoAcFX

[67] Y. Chauhan. (2021).

[68] E. Liberty, R. Sriharsha, and M. Sviridenko, “An algorithm for online K-means clustering,” in Proc. 18th Workshop Algorithm Eng. Exp. (ALENEX), Jan. 2016.

[69] A. Banerjee and S. Basu, “Topic models over text streams: A study of batch and online unsupervised learning,” in Proc. SIAM Int. Conf. Data Mining, Apr. 2007, pp. 431–436, doi: 10.1137/1097811972771.40.

[70] M. Moakher, “A differential geometric approach to the geometric mean of symmetric positive-definite matrices,” SIAM J. Matrix Anal. Appl., vol. 26, no. 3, pp. 735–747, Jan. 2005.

[71] D. Clark-Carter, Measures of Central Tendency. Amsterdam, The Netherlands: Elsevier, 2010.

[72] S. Zhang, Y. Wang, Y. Zhang, P. Wan, and J. Zhuang, “Riemannian distance-based fast K-medoids clustering algorithm for cooperative spectrum sensing,” IEEE Syst. J., vol. 16, no. 1, pp. 880–890, Mar. 2022.

[73] G. Schnass, “Convergence radius and sample complexity of ITRM algorithms for dictionary learning,” Appl. Comput. Harmon. Anal., vol. 45, no. 1, pp. 22–58, 2018.

[74] M. K. Pakhira, “A linear time-complexity K-means algorithm using cluster shifting,” in Proc. Int. Conf. Comput. Intell. Commun. Netw., Nov. 2014, pp. 1047–1051.

[75] A. Abernathy and M. E. Celebi, “The incremental online k-means clustering algorithm and its application to color quantization,” Expert Syst. Appl., vol. 207, Nov. 2022, Art. no. 117927.

IMTIAZ NASIM (Student Member, IEEE) received the B.S. degree in electrical and electronic engineering from the Khulna University of Engineering and Technology, Bangladesh, in 2015, and the M.S. degree in electrical engineering from Georgia Southern University, Statesboro, GA, USA, in 2019. He is currently pursuing the Ph.D. degree with Florida International University, Miami, FL, USA. His research interests include vehicle-to-everything (V2X) communications, reliability of autonomous vehicles with machine learning, human exposure to electromagnetic field in communications systems, and millimeter wave communications. He is a Reviewer of multiple journals and conferences, including IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE ACCESS, and IEEE Vehicular Technology Conference.

AHMED S. IBRAHIM (Member, IEEE) received the B.S. and M.S. degrees in electronics and electrical communications engineering from Cairo University, Cairo, Egypt, in 2002 and 2004, respectively, and the Ph.D. degree in electrical engineering from the University of Maryland, College Park, MD, USA, in 2009. He was an Assistant Professor at Cairo University, a Wireless Research Scientist at Intel Corporation, and a Senior Engineer at Interdigital Communications Inc. He is currently an Associate Professor with the Electrical and Computer Engineering Department, Florida International University (FIU), Miami, FL, USA. His research interests include wireless systems including drone-assisted millimeter wave communications, vehicular communications, and rethinking wireless networks through the lens of Riemannian geometry. He was a recipient of the NSF CAREER Award, in 2022.

* * *