Multi-Manifold Modeling in Non-Euclidean spaces

Xu Wang
Dept. of Mathematics
University of Minnesota
Minneapolis, MN 55455, USA

Konstantinos Slavakis
Dept. of ECE and DTC
University of Minnesota
Minneapolis, MN 55455, USA

Gilad Lerman
Dept. of Mathematics
University of Minnesota
Minneapolis, MN 55455, USA

Abstract

This paper advocates a novel framework for segmenting a dataset on a Riemannian manifold M into clusters lying around low-dimensional submanifolds of M. Important examples of M, for which the proposed algorithm is computationally efficient, include the sphere, the set of positive definite matrices, and the Grassmannian. The proposed algorithm constructs a data-affinity matrix by thoroughly exploiting the intrinsic geometry and then applies spectral clustering. Local geometry is encoded by sparse coding and directional information of local tangent spaces and geodesics, which is important in resolving intersecting clusters and establishing the theoretical guarantees for a simplified variant of the algorithm. To avoid complication, these guarantees assume that the underlying submanifolds are geodesic. Extensive validation on synthetic and real data demonstrates the resiliency of the proposed method against deviations from the theoretical (geodesic) model as well as its superior performance over state-of-the-art techniques.

1 Introduction

Many modern data sets are of moderate or high dimension, but manifest intrinsically low-dimensional structures. *Multi-manifold modeling* (MMM), or its special case, *hybrid-linear modeling* (HLM), are natural frameworks for studying such data-sets. In MMM, data-sets are modeled as a union of low-dimensional submanifolds (whereas HLM considers the union of affine subspaces). Clustering algorithms in the MMM context aim at partitioning data according to the underlying low-dimensional submanifolds. MMM has been extensively studied and applied to model data-sets embedded in the Euclidean space or the sphere [Arias-Castro et al., 2011, Arias-Castro et al., 2013, Cetingul and Vidal, 2009, Elhamifar and Vidal, 2011, Kushnir et al., 2006, Ho et al., 2013, Lui, 2012, Wang et al., 2011].

There is an overwhelming number of application domains where features extracted from data-sets lie on Riemannian manifolds and the adjacency or similarity between features is better described by non-Euclidean distances. For example, auto-regressive moving average (ARMA) models are utilized to extract low-rank linear subspaces (points on the Grassmannian) for identifying spatio-temporal dynamics in video sequences [Turaga et al., 2011]. Moreover, convolving patches of images by Gabor filters yields covariance matrices (points on the manifold of positive semidefinite matrices) that can capture effectively texture patterns in images [Tou et al., 2009]. Notwithstanding, current MMM strategies are not sufficiently broad for handling data-sets embedded in more general Riemannian manifolds.

The purpose of this paper is to develop theory and algorithms for the MMM problem in general Riemannian manifolds that are relevant to important applications.

Related Work. Recent advances in parsimonious data representations and their important implications in dimensionality reduction techniques have effected the development of non-standard spectral-clustering schemes that result in state-of-the-art results in modern applications [Arias-Castro et al., 2013, Chen and Lerman, 2009a, Elhamifar and Vidal, 2009, Goh and Vidal, 2008, Goldberg et al., 2009, Harandi et al., 2013, Liu et al., 2013, Zhang et al., 2012]. Such schemes rely on the assumption that data exhibit low-dimensional structures, such as unions of low-dimensional linear subspaces or submanifolds, but are restricted to Eu-

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.

clidean distances. Some theoretical guarantees for particular HLM algorithms appear in [Chen and Lerman, 2009b, Lerman and Zhang, 2011, Soltanolkotabi and Candès, 2012, Soltanolkotabi et al., 2014]. There are fewer strategies for the MMM problem, which is also known as manifold clustering. Only higher-order spectral clustering and spectral local PCA are theoretically guaranteed [Arias-Castro et al., 2011, Arias-Castro et al., 2013].

Several algorithms for clustering with non-Euclidean distances are generalizations of well-known schemes developed originally for Euclidean spaces. For example, Gruber and Theis [2006], Tuzel et al. [2005], Subbarao and Meer [2006], Cetingul and Vidal [2009] extended the classical K-means algorithm and mean-shift (MS) algorithm to general analytic manifolds, where non-Euclidean distances on Grassmannians, Stiefel manifolds, and matrix Lie groups, are used. O'Hara et al. [2011] showed promising results by using the geodesic distance of product manifolds in clustering of human expressions, gestures, and actions in videos. Goh and Vidal [2008] extended spectral clustering and nonlinear dimensionality reduction techniques to Riemannian manifolds. These previous works are quite successful when the convex hulls of individual clusters are well-separated, but they often fail when clusters intersect or are closely located.

Contributions. Despite the popularity of manifold learning, the associated literature lacks generic schemes for clustering low-dimensional data embedded in non-Euclidean spaces. Furthermore, even in the Euclidean setting only few algorithms for MMM or HLM are theoretically guaranteed. To this end, this paper aims at filling this gap and provides an MMM approach in non-Euclidean setting with some theoretical guarantees even when the clusters intersect. In order to avoid nontrivial theoretical obstacles, the theory assumes that the underlying submanifolds are geodesic; hence the term *multi-geodesic modeling* (MGM). A more practical and robust variant of the theoretical algorithm is also developed, and its superior performance over state-of-the-art clustering techniques is exhibited by extensive validation on synthetic and real datasets. In this work, we only explored manifolds whose logarithm maps can be computed efficiently but have not exhausted all possible applicable spaces. For example, GCT easily applies to the "space of shapes", which arises in computational anatomy [Pennec, 2009]. There are also works dedicated to learn empirically the logarithm map from a given point cloud [Haro et al., 2006, Hauberg et al., 2012].

Extentions of the theoretical foundations of this work to deal with general submanifolds are possible by leveraging local geodesic submanifolds (in analogy to Arias-Castro et al. [2013]). However, this will significantly increase the complexity of our proofs, which are already not simple. In practice, the proposed method applies directly to the more general setting (without theoretical guarantees). Indeed, numerical tests show that the proposed method works well in real practical scenarios that deviate from the MGM model.

2 Theoretical Preliminaries

This section formulates the generic clustering problem which relates to MGM, and reviews the necessary background from Riemannian geometry.

2.1 Multi-Geodesic Modeling (MGM)

Given a prescribed number K of clusters, MGM assumes that each point in a given data-set $X = \{x_i\}_{i=1}^N$ lies in the tubular neighborhood of some unknown geodesic submanifold S_k, $1 \leq k \leq K$, of a Riemannian manifold, M.[1] The *MGM problem* refers to clustering X into K groups $X_1, \ldots, X_K \subset M$ such that (s.t.) points in X_k are located close to the submanifold S_k. Note that if M is a Euclidean space, geodesic submanifolds are affine subspaces, MGM boils down to HLM, and the MGM problem becomes equivalent to subspace clustering [Elhamifar and Vidal, 2009, Vidal, 2011, Zhang et al., 2012].

For theoretical purposes, the uniform MGM is adopted: Data points are i.i.d. sampled with respect to (w.r.t.) the uniform distribution on a fixed tubular neighborhood of $\cup_{k=1}^K S_k$. The radius τ of the tubular neighborhood is also termed the noise level. Fig. 1 illustrates data generated from uniform MGM with two underlying submanifolds ($K = 2$).

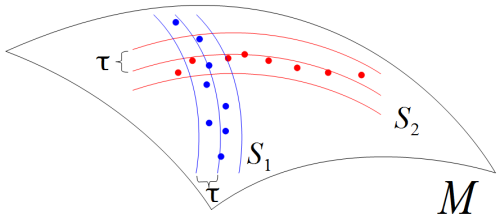


Figure 1: Uniform MGM with $K = 2$.

MGM serves only the theoretical justification of this paper. Numerical tests show that the proposed algorithm works well under general MMM settings, such as not necessarily geodesic submanifolds, non-uniform sampling, as well as different kinds and levels of noise.

[1] The tubular neighborhood with radius $\tau > 0$ of S_k in M (with metric tensor g and induced distance dist_g) is $S_k^\tau = \{x \in M : \text{dist}_g(x, s) < \tau \text{ for some } s \in S_k\}$.

3 Algorithmic Solutions for the MGM (MMM) Problem

This section proposes solutions for the general MMM problem with theoretical guarantees for the MGM case. Section 3.1 defines two key quantities that are instrumental in quantifying directional information: Estimated local tangent subspaces and geodesic angles. Section 3.2 presents two algorithmic solutions for the MMM/MGM problem and discusses their properties.

3.1 Directional Information

Estimated Local Tangent Subspace $T^E_{x_i}S$. Let $\mathbf{x}_j^{(i)} := \log_{x_i}(x_j)$ denote the image of x_j under the logarithm map at x_i. For the sake of illustration, assume a data-set $X = \{x_j\}_{j=1}^N \subset M$ generated by uniform MGM with a single geodesic submanifold S ($K = 1$). The dataset is thus contained in a tubular neighborhood, S^τ, of a d-dimensional geodesic submanifold S. Since S is geodesic, for any $1 \le i \le N$ the set $\{\mathbf{x}_j^{(i)}\}_{j=1}^N$ of images by the logarithm map is contained in a tubular neighborhood of the d-dimensional subspace $T_{x_i}S$ (possibly with a different radius than τ). Since the true tangent subspace $T_{x_i}S$ is unknown, an estimation of it, $T^E_{x_i}S$, is needed. If $B(x_i, r) \subset M$ is the neighborhood centered at x_i with radius $r > 0$, then let

$$J(x, r) := \{j : x_j \in B(x, r) \cap X\}. \tag{1}$$

Moreover, let $\mathbf{C}_{x_j}$ denote the local sample covariance matrix of the dataset $\{\mathbf{x}_j^{(i)}\}_{j\in J(x_i,r)}$ on $T_{x_i}M$, and $\|\mathbf{C}_{x_j}\|$ the spectral norm of $\mathbf{C}_{x_j}$, i.e., its maximum eigenvalue. Since $\{\mathbf{x}_j^{(i)}\}_{j=1}^N$ is in a tubular neighborhood of a d-dimensional subspace, estimates of the intrinsic dimension d of the local tangent subspace, which is also the dimension of S, can be formed by identifying principal eigenvalues of $\mathbf{C}_{x_j}$. We adopt this strategy of dimension estimation and define the estimated local tangent subspace, $T^E_{x_i}S$, as the span in $T_{x_i}M$ of the top eigenvectors of $\mathbf{C}_{x_j}$. In our theoretical inquiries, the number of top eigenvectors is the number of eigenvalues of $\mathbf{C}_{x_i}$ that exceed $\eta\|\mathbf{C}_{x_j}\|$ for some fixed $0 < \eta < 1$ (cf. Theorem 3.1). On the practical side, the top eigenvectors correspond to those eigenvalues of $\mathbf{C}_{x_i}$ that are maximally separated (largest margin) from the rest of them.

Empirical Geodesic Angles. Let $l(x_i, x_j)$ be the shortest geodesic (global length minimizer) connecting x_i and x_j in (M, g). Let $\mathbf{v}_{ij} \in T_{x_i}M$ be the tangent vector of $l(x_i, x_j)$ at x_i, i.e., the "velocity" of $l(x_i, x_j)$ at x_i. Given $X = \{x_j\}_{j=1}^N$, the empirical geodesic angle θ_{ij} is the elevation angle between the vector $\mathbf{v}_{ij}$ and the subspace $T^E_{x_i}S$ in the Euclidean space $T_{x_i}M$ (cf. (9) of Lerman and Whitehouse [2009]).

3.2 Proposed Solutions

In Section 3.2.1, we propose a theoretical solution for data sampled according to uniform MGM. We start with its basic motivation, then describe the proposed algorithm and at last formulate its theoretical guarantees. A more practical solution, tailored to the MMM problem, is detailed in Section 3.2.2.

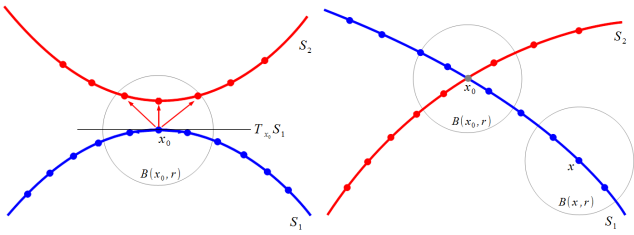
3.2.1 Algorithm 1: Theoretical Geodesic Clustering with Tangent information

The crux of the proposed solution for the MGM problem is the application of spectral clustering with carefully chosen weights for the affinity or adjacency matrix. Specifically, given a data-set, a similarity graph is constructed whose vertices are data points and whose edges represent the similarity between data points. The main challenge is to construct a graph s.t. two points are locally connected only when they are associated with the same cluster. In this way, the application of spectral clustering recovers exactly the underlying clusters.

For illustration, assume only two geodesic submanifolds S_1 and S_2. Assume also that the data were sampled from $S_1 \cup S_2$ according to uniform MGM. Given a point $x_0 \in S_1$ one wishes to associate it with points from the same submanifold within a neighborhood $B(x_0, r)$, for some $r > 0$. It is not realistic to assume that all points in $B(x_0, r)$ stem from the same submanifold. To appreciate these challenges, consider Figs. 2a and 2b where clusters are closely located or intersect in a neighborhood $B(x_0, r)$ of x_0.

Assuming first no cluster intersection at x_0 (Fig. 2a), we use local tangent information at x_0 to identify points in $B(x_0, r)$ from the same submanifold with which x_0 is associated. If $x \in B(x_0, r)$ belongs to S_2, then the geodesic $l(x_0, x)$ has a large angle with the tangent space $T_{x_0}S_1$ at x_0. On the other hand, if such x belongs to S_1, then the geodesic has an angle close to zero. Therefore, thresholding empirical geodesic angles may become beneficial for eliminating neighboring points from a different submanifold.

If x_0 is at or near the intersection, it is hard to estimate correctly the tangent spaces of each submanifold and the geodesic angles may not be reliable. Instead, one may compare the dimensions of estimated local tangent subspaces. The estimated dimensions of local neighborhoods of data points which are close to intersections are larger than the estimated dimensions of local neighborhoods of data points located away from intersections (cf. Fig. 2b). The subsequent algorithm thus connects x_0 to other neighboring points only when their "local dimensions" (linear-algebraic dimension of



(a) Angle Filtering (b) Intersection

Figure 2: Challenges in the construction of data-similarity graphs.

the estimated local tangent) are the same. In this way, points at the intersection of submanifolds will not be connected with other clusters.

The dimension difference criterion, together with the angle filtering procedure, guarantee that there is no false connection between different clusters (rigorous arguments are under the hood of Theorem 3.1). These two simple ideas are leveraged together with spectral clustering to form the following theoretical geodesic clustering with tangent information (TGCT) algorithm.

Algorithm 1 Theoretical geodesic clustering with tangent information (TGCT)

Input: Number of clusters: $K \geq 2$, data-set X of cardinality N, neighborhood radius r, threshold η for estimating tangent subspaces, distance threshold σ_d, and angle threshold σ_a.

Output: Index set $\{\mathrm{Id}_i\}_{i=1}^N \subset \{1,\ldots,K\}$ s.t. Id_i is the cluster label assigned to x_i.

Steps:

• Compute the following geometric quantities per data point:

for $i = 1,\ldots,N$ **do**

∘ For $j \in J(x_i,r)$ [cf. (1)], compute $\mathbf{x}_j^{(i)} = \log_{x_i}(x_j)$.

∘ Compute the sample covariance matrix $\mathbf{C}_{x_i}$ of $\{\mathbf{x}_j^{(i)}\}_{j\in J(x_i,r)}$.

∘ Compute the eigenvectors of $\mathbf{C}_{x_i}$ with eigenvalues exceeding $\eta \cdot \|\mathbf{C}_{x_i}\|$ (their span is $T^E_{x_i}S$).

∘ For all $j = 1,\ldots,N$, compute the empirical geodesic angles θ_{ij} (see Section 3.1).

end for

• Form the following $N \times N$ affinity matrix $\mathbf{W}$:

$$\mathbf{W}_{ij} = \mathbf{1}_{\mathrm{dist}_g(x_i,x_j)<\sigma_d}\mathbf{1}_{\dim(T^E_{x_i}S)=\dim(T^E_{x_j}S)}\mathbf{1}_{(\theta_{ij}+\theta_{ji})<\sigma_a}$$

• Apply spectral clustering to the affinity matrix $\mathbf{W}$ to determine $\{\mathrm{Id}_i\}_{i=1}^N$.

The following theorem asserts that TGCT achieves correct clustering with high probability. Its proof is in [Wang et al., 2014, Sec. 5], where constants $\{C_i\}_{i=0}^6$ and C_0' are clarified and depend only on the underlying geometry of M. For simplicity, the theorem assumes that there are only two geodesic submanifolds and that they are of the same dimension. However, the theorem can be extended to K geodesic submanifolds of different dimensions.

Theorem 3.1 *Consider two smooth compact d-dimensional geodesic submanifolds, S_1 and S_2, of a Riemannian manifold M, and let X be a data-set generated according to uniform MGM w.r.t. $S_1 \cup S_2$ with noise level τ. If the parameters $(r,\sigma_d,\sigma_a,\eta)$ of the TGCT algorithm satisfy the inequalities*

$$\eta < C_2^{-\frac{d+2}{2}}, \sigma_d < C_4^{-\frac{1}{2}},\ \sigma_a < \pi/6,$$
$$\sigma_a < \sin^{-1}(r\sqrt{1 - C_2\eta^{\frac{2}{d+2}}}/(2\sigma_d)) - C_3\eta^{\frac{d}{d+2}} - C_3 r$$
$$\text{and } \tau/C_5 < r < \min(\eta,\sigma_d,\sigma_a)/C_1,$$

then with probability at least $1 - C_0 N \exp[-Nr^{d+2}/C_0']$, the TGCT algorithm can cluster correctly a sufficiently large subset of X, whose relative fraction (over X) has expectation at least $1 - C_6(r+\tau)^{d-\dim(S_1\cap S_2)}$.

3.2.2 Algorithm 2: Geodesic Clustering with Tangent information

A practical version of the TGCT algorithm, referred to as *geodesic clustering with tangent information* (GCT), is described in Algorithm 2. This is the algorithm implemented in Section 4, and the choice of its parameters is clarified in [Wang et al., 2014, Appendix A]. GCT differs from TGCT in three ways. First, hard thresholds in TGCT are replaced by soft ones, which are more flexible. Second, the dimension indicator function is dropped from the affinity matrix W. Indeed, numerical experiments indicate that the algorithm works properly without the dimension indicator function, whenever there is only a small portion of points near the intersection. This numerical observation makes sense since the dimension indicator is only used in theory to avoid connecting intersection points to points not in intersection. At last, pairwise distances are replaced by weights resulting from sparsity-cognizant optimization tasks. Sparse coding takes advantage of the low-dimensional structure, resulting into larger weights for points coming from the same submanifold [Elhamifar and Vidal, 2011].

Algorithm 2 solves a sparse coding task in (2). The penalty used is non-standard since the scalar-valued terms $|\mathbf{S}_{ij}|$ are multiplied by $e^{\|\mathbf{x}_i^{(i)} - \mathbf{x}_j^{(i)}\|_2/\sigma_d}$ (where, e.g., in Cetingul et al. [2014], these latter terms are all

1). These weights were chosen to increase the effect of nearby points (in addition to their sparsity). In particular, it avoids sparse representations via far-away points that are unrelated to the local manifold structure. Similarly to Cetingul et al. [2014], clustering weights in (3) exponentiate the sparse-coding weights.

Algorithm 2 Geodesic clustering with tangent information (GCT)

Input: Number of clusters: $K \geq 2$, dataset X of cardinality N, neighborhood radius r, distance threshold σ_d (default value: $\sigma_d = 1$), and angle threshold σ_a (default value: $\sigma_a = 1$).

Output: Index set $\{\mathrm{Id}_i\}_{i=1}^N \subset \{1, \ldots, K\}$ s.t. Id_i is the cluster label assigned to x_i.

Steps:

for $i = 1, \ldots, N$ **do**

- For $j \in J(x_i, r)$, compute $\mathbf{x}_j^{(i)} = \log_{x_i}(x_j)$.
- Compute weights $\{\mathbf{S}_{ij}\}_{j \in J(x_i,r)}$ that minimize

$$\|\mathbf{x}_i^{(i)} - \sum_{\substack{j \in J(x_i,r) \\ j \neq i}} \mathbf{S}_{ij} \mathbf{x}_j^{(i)}\|_2^2 + \sum_{\substack{j \in J(x_i,r) \\ j \neq i}} e^{\frac{1}{\sigma_d}\|\mathbf{x}_i^{(i)} - \mathbf{x}_j^{(i)}\|_2} |\mathbf{S}_{ij}| \quad (2)$$

among all $\{\mathbf{S}_{ij}\}_{j \in J(x_i,r)}$ s.t. $\mathbf{S}_{ii} = 0$ and $\sum_{j \in J(x_i,r);\, j \neq i} \mathbf{S}_{ij} = 1$. Moreover, set $\mathbf{S}_{ij} = 0$ for $j \notin J(x_i, r)$.

- Compute the sample covariance matrix $\mathbf{C}_{x_i}$ of $\{\mathbf{x}_j^{(i)}\}_{j \in J(x_i,r)}$.
- Find the index m that results into the largest gap between eigenvalues λ_m and λ_{m+1} of $\mathbf{C}_{x_i}$, and identify the top m eigenvectors of $\mathbf{C}_{x_i}$ (their span is $T_{x_i}^E S$).
- For all $j = 1, \ldots, N$, compute the empirical geodesic angles θ_{ij} (cf. Section 3.1).

end for

- Form the following $N \times N$ affinity matrix $\mathbf{W}$:

$$\mathbf{W}_{ij} = e^{|\mathbf{S}_{ij}| + |\mathbf{S}_{ji}|} e^{-\frac{1}{\sigma_a}(\theta_{ij} + \theta_{ji})}. \quad (3)$$

- Apply spectral clustering to the affinity matrix W to determine $\{\mathrm{Id}_i\}_{i=1}^N$.

4 Numerical Tests

To assess performance on both synthetic and real datasets, GCT is compared with the following algorithms: (i) *Sparse manifold clustering* (SMC) [Elhamifar and Vidal, 2011, Cetingul et al., 2014], which is adapted here for clustering on a Riemannian manifold but still referred to as SMC, (ii) *spectral clustering with Riemannian metric* (SCR) of Goh and Vidal [2008], and (iii) *embedded K-means* (EKM). The choices of parameters for all four methods are detailed in [Wang et al., 2014, Appendix A].

Ground truth labels are known in all experiments. To measure the accuracy of each method, assigned labels are first permuted to have the maximal match with the ground truth labels. Clustering rate or accuracy is computed as follows:

$$\frac{\#\text{ of points whose assigned labels equal ground truth}}{\#\text{ of total points}}.$$

4.1 Tests on Synthetic Data-sets

Six datasets were generated. Dataset I and II are from the Grassmannian $\mathrm{G}(6, 2)$, i.e., the set of all rank 2 linear subspaces of the 6-dimensional Euclidean space, datasets III and IV are from 3×3 symmetric positive-definite (PD) matrices, and datasets V and VI are from the sphere $\mathbb{S}^2$. Each dataset contains 260 points generated from two "parallel" or intersecting submanifolds, cropped by white Gaussian noise (cf. [Wang et al., 2014, Sec. 4.1]).

Each one of the six datasets is generated according to the postulated models above, and the experiment is repeated 30 times. Table 1 shows average clustering rate per method. GCT, SMC, and SCR are all based on the spectral clustering scheme. However, when a dataset has low-dimensional structures, GCT's unique procedure of filtering neighboring points yields superior performance over the rest of the methods. This is because both SMC and SCR are sensitive to the local scale σ, and require each neighborhood not to contain points from different groups. This becomes clear by the results on datasets I, IV, and V of non-intersecting submanifolds. SMC only works well in dataset I, where most of the neighborhoods $B(x_0, r)$ contain only points from the same cluster, while neighborhoods $B(x_0, r)$ in datasets IV and V often contain points from different clusters. EKM generally requires that the intrinsic means of different clusters are located far from each other. Its performance is not as good as GCT when different groups have low-dimensional structures.

4.2 Robustness to Noise and Running Time

Section 4.1 illustrated GCT's superior performance over SMC, SCR, and EKM on a variety of manifolds. This section further investigates GCT's robustness to noise and computational cost pertaining to running time. In summary, GCT is shown to be far more robust than SMC against noise at the price of a small increase of running time.

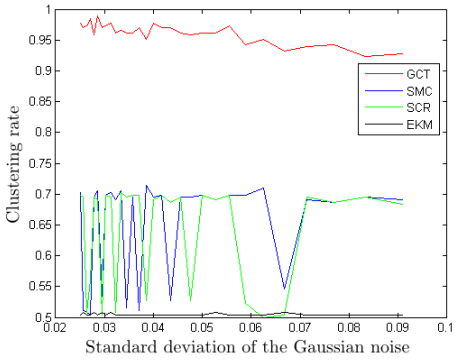
4.2.1 Robustness to Noise

The proposed tangent filtering scheme enables GCT to successfully eliminate neighboring points that originate from different groups. As such, it exhibits robust-

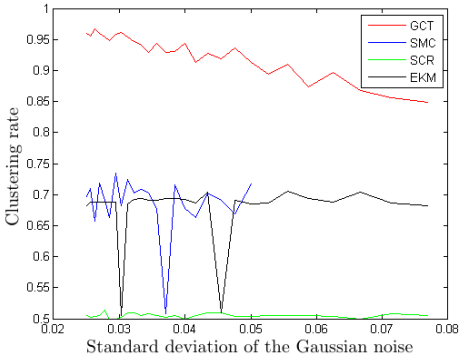
Table 1: Average clustering rates on the synthetic data-sets of Section 4.1.

Methods	Set I	Set II	Set III	Set IV	Set V	Set VI
GCT	**1.00 ±0.00**	**0.98 ±0.01**	**0.98 ±0.00**	**0.95 ±0.01**	**0.98 ±0.01**	**0.96 ±0.01**
SMC	0.97 ±0.04	0.66 ±0.08	0.88 ±0.03	0.80 ±0.02	0.55 ±0.06	0.69 ±0.05
SCR	0.51 ±0.00	0.66 ±0.07	0.84 ±0.00	0.80 ±0.00	0.50 ±0.00	0.53 ±0.07
EKM	0.50 ±0.00	0.50 ±0.00	0.67 ±0.00	0.50 ±0.00	0.50 ±0.00	0.67 ±0.06

ness in the presence of noise and/or whenever different groups are close or even intersecting. On the other hand, SMC appears to be sensitive to noise due to its sole dependence on sparse weights. Figs. 3a and 3b demonstrate the performance of GCT, SMC, SCR, and EKM on the Grassmannian and the sphere for various noise levels (standard deviations of Gaussian noise).



(a) Grassmannian



(b) Sphere

Figure 3: Performance of clustering methods on Riemannian manifolds for various noise levels: (a) the Grassmannian case, (b) the case of the sphere.

Data-sets in Fig. 3a are generated according to the model of dataset II in Section 4.1 but with different noise levels (in Section 4.1 the noise level was 0.025). Both SMC and SCR appear to be volatile over different datasets, with their best clustering rate performance never exceeding 0.75. It is worth noticing that EKM shows poor clustering accuracy. On the contrary, GCT exhibits remarkable robustness to noise, achieving clustering rates above 0.9 even when the standard deviation of the noise approaches 0.1.

GCT's robustness to noise is also demonstrated in Fig. 3b, where datasets are generated on the unit sphere according to the model of the dataset VI, but with different noise levels. SMC appears to be volatile also in this setting; it collapses when the standard deviation of noise exceeds 0.05, since its affinity matrix precludes spectral clustering from identifying eigenvalues with sufficient accuracy.

4.2.2 Running time

This section demonstrates that GCT outperforms SMC at the price of a small increase in computational complexity. Similarly to any other manifold clustering algorithm, computations have to be performed per local neighborhood, where local linear structures are leveraged to increase clustering accuracy. The overall complexity scales quadratically w.r.t. the number of data-points due to the last step of Algorithm 2, which amounts to spectral clustering of the $N \times N$ affinity matrix $\mathbf{W}$. Both the optimization task of (2) and the computation of a few principal eigenvectors of the covariance matrix $\mathbf{C}_{x_i}$ in Algorithm 2 do not contribute much to the complexity since operations are performed on a small number of points in the neighborhood $J(x_i, r)$. The computational complexity of GCT is detailed in [Wang et al., 2014, Appendix C]. We note that GCT can be fully parallelized since computations per neighborhood are independent. Nevertheless, such a route is not followed in this section.

Compared with SMC, GCT has one additional task that entails local calculation of a few principal eigenvectors. Nevertheless, it is shown in [Wang et al., 2014, Appendix C.1] that for k neighbors, this task can be carried out with $\mathcal{O}(D + k^3)$ operations.

The ratios of running times between GCT and SMC for all three types of manifolds are illustrated in Table 2. It can be readily verified that the extra step of

Table 2: Ratio of running times of GCT and SMC for instances of the synthetic data-sets I, IV, and VI.

Running-time ratio	$\mathrm{G}(6,2)$	$PD_{3\times3}$	$\mathbb{S}^2$
GCT/SMC	1.06	1.05	1.11

identifying tangent spaces in GCT increases running time by less than 11% of the one for SMC.

Ratios of running times were also investigated for increasing ambient dimensions. More precisely, dataset VI of Section 4.1, which lies in S^2, was embedded via a random orthonormal matrix into the unit sphere $\mathbb{S}^D$, where D ranged from 100 to 3,000. Fig. 4 shows the ratios of the running time of GCT over that of SMC as a function of D. We observe that the extra cost of computing the eigendecomposition in GCT is mostly less than 20% of SMC, and never exceeds 30%, even when the ambient dimension is as large as 3,000.

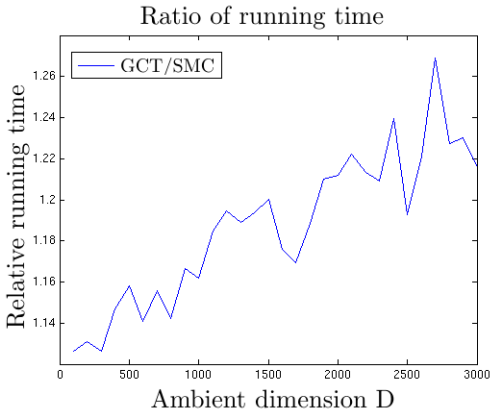


Figure 4: Relative running times of GCT w.r.t. SMC as the ambient dimension increases.

4.2.3 Clustering Dynamic Patterns.

Spatio-temporal data such as dynamic textures and videos of human actions can often be approximated by linear dynamical models [Doretto et al., 2003, Turaga et al., 2011]. In particular, by leveraging the auto-regressive and moving average (ARMA) model, we experiment here with two spatio-temporal databases, Dyntex++ [Ghanem and Ahuja, 2010] and Ballet [Wang and Mori, 2009], in the following two-step procedure: (i) First, the ARMA model is employed to associate local spatio-temporal patches with linear subspaces of the same dimension, and (ii) manifold clustering is applied on the Grassmannian to partition textures and Ballet actions, respectively.

ARMA Model. The premise of ARMA modeling is based on the assumption that the spatio-temporal dataset under study is governed by a small number of latent variables. More specifically, if $\mathbf{f}(t) \in \mathbb{R}^p$ is the observation vector at time t (in our case, it is the vectorized image frame of a video sequence), then

$$\begin{aligned} \mathbf{f}(t) &= \mathbf{C}\mathbf{z}(t) + \boldsymbol{\epsilon}_1(t) \qquad & \boldsymbol{\epsilon}_1(t) &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_1) \\ \mathbf{z}(t+1) &= \mathbf{A}\mathbf{z}(t) + \boldsymbol{\epsilon}_2(t) \qquad & \boldsymbol{\epsilon}_2(t) &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_2) \end{aligned} \tag{4}$$

where $\mathbf{z}(t) \in \mathbb{R}^\ell$, $\ell \leq p$, is the vector of latent variables, $\mathbf{C} \in \mathbb{R}^{p\times\ell}$ is the observation matrix, $\mathbf{A} \in \mathbb{R}^{\ell\times\ell}$ is the transition matrix, and $\boldsymbol{\epsilon}_1(t) \in \mathbb{R}^p$ and $\boldsymbol{\epsilon}_2(t) \in \mathbb{R}^\ell$ are i.i.d. sampled vector-values r.vs. obeying the Gaussian distributions $\mathcal{N}(0, \boldsymbol{\Sigma}_1)$ and $\mathcal{N}(0, \boldsymbol{\Sigma}_2)$, respectively.

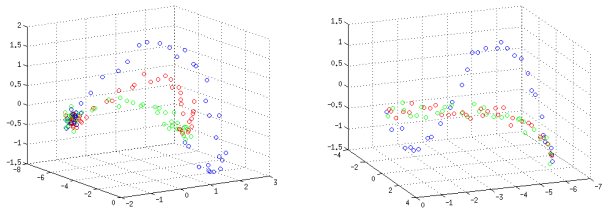
We next explain the idea of Turaga et al. [2011] to associate subspaces with spatio-temporal data. Given data $\{\mathbf{f}(t)\}_{t=\tau_1}^{\tau_2}$, the ARMA parameters $\mathbf{A}$ and $\mathbf{C}$ can be estimated according to the procedure in Turaga et al. [2011]. Moreover, by arbitrarily choosing $\mathbf{z}(0)$, it can be verified that for any $m \in \mathbb{N}$,

$$\mathbb{E}\begin{bmatrix} \mathbf{f}(\tau_1) \\ \mathbf{f}(\tau_1+1) \\ \vdots \\ \mathbf{f}(\tau_1+m-1) \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^{m-1} \end{bmatrix} \mathbf{z}(\tau_1).$$

We then set $\mathbf{V} := [\mathbf{C}^\top, (\mathbf{C}\mathbf{A})^\top, \ldots, (\mathbf{C}\mathbf{A}^{m-1})^\top]^\top \in \mathbb{R}^{mp\times\ell}$, which is known as the mth order observability matrix. If the observability matrix is of full column rank, which was the case in all of the conducted experiments, the column space of $\mathbf{V}$ is a ℓ-dimensional linear subspace of $\mathbb{R}^{pm}$. In other words, the ARMA model estimated from data $\{\mathbf{f}(t)\}_{t=\tau_1}^{\tau_2}$, $\tau_1 \leq \tau_2$, gives rise to a point on the Grassmannian $\mathrm{G}(mp, \ell)$. For a fixed dataset $\{\mathbf{f}(t)\}_{t=1}^{\tau}$, different choices of (τ_1, τ_2), s.t. $\tau_1, \tau_2 \leq \tau$, and several local regions within the image give rise to different estimates of $\mathbf{A}$ and $\mathbf{C}$ and thus to different points in $\mathrm{G}(mp, \ell)$.

Dynamic textures. The Dyntex++ database [Ghanem and Ahuja, 2010] contains 3600 dynamic textures videos of size $50\times50\times50$, which are divided into 36 categories. It is a hard-to-cluster database due to its low resolution. Three videos were randomly chosen, each one from a distinct category.

Per video sequence, 50 patches of size $40\times40\times20$ are randomly chosen. Each frame of the patch is vectorized resulting into patches of size 1600×20. To reduce the size to 30×20, a random projection operator is applied to each patch. As a result, each patch is reduced to the set $\{\mathbf{f}(t)\}_{t=\tau_1}^{\tau_1+20} \subset \mathbb{R}^{30}$. We fix $\ell = 3$ and $m = 3$ and use each such set $\{\mathbf{f}(t)\}_{t=\tau_1}^{\tau_1+20}$ to estimate the underlying ARMA model. Consequently, 150 points on $\mathrm{G}(90, 3)$ are generated, 50 per video category.



(a) Dyntex++ (b) Ballet

Figure 6: Visualization of Dyntex++ and Ballet data-sets via projection onto the space spanned by their three principal components.

Table 3: Average clustering accuracy rates for the Dyntex++ And Ballet data-sets.

Methods	GCT	SMC	SCR	EKM
Dyntex++	**0.85**	0.69	0.77	0.42
Ballet	**0.81**	0.76	0.68	0.47

We expect that points in $\mathrm{G}(90,3)$ of the same cluster lie near a submanifold of $\mathrm{G}(90,3)$. This is due to the repeated pattern of textures in space and time (they often look like a shifted version of each other in space and time). To visualize the submanifold structure, we isometrically embedded $\mathrm{G}(90,3)$ into a Euclidean space [Basri et al., 2011], so that subspaces are mapped to Euclidean points. We then projected the latter points on their top 3 principal components. Fig. 6a demonstrates this projection as well as the submanifold structure within each cluster.

Ballet database. The Ballet database [Wang and Mori, 2009] contains 44 videos of 8 actions from a ballet instruction DVD. The frames of all videos are of size 301×301 and their lengths vary and are larger than 100. Different performers have different attire and speed. Three videos, each one associated with a different action, were randomly chosen.

Figure 5: Two samples of Ballet video sequences: The first and second rows comprise samples from the actions of hopping and leg-swinging, respectively.

Spatio-temporal patches are generated by selecting 10 consecutive frames of size 301×301 from each one of the following overlapping time intervals: $\{1,\ldots,10\}$, $\{4,\ldots,13\}$, $\{7,\ldots,16\}$, $\ldots$, $\{91,\ldots,100\}$. In this way, for each of the three videos, 31 spatio-temporal patches of size $301 \times 301 \times 10$ are generated. As in the case of the Dyntex++ database, video patches are vectorized and downsized to spatio-temporal patches of size 30×10. Following the previous ARMA modeling approach, we set $\ell = 3$ and $m = 3$ and associate each such patch with a subspace in $\mathrm{G}(90,3)$. Consequently, 93 subspaces (31 per cluster) in the Grassmannian $\mathrm{G}(90,3)$ are generated. Fig. 6b visualizes the 3D representation of the subspaces created from three videos. Clusters indeed intersect, and their intersection represents still motion.

The procedure described above (for generating data by randomly choosing 3 videos from the Dyntex++ and Ballet databases and applying clustering methods on $\mathrm{G}(90,3)$) is repeated 30 times. The average clustering accuracy rates are reported in Table 3. GCT achieves the highest rates on both datasets.

5 Conclusions

Aiming at efficiently organizing data embedded in a non-Euclidean space according to low-dimensional structures, the present paper studied the multi-manifold modeling (MMM) problem. The paper advocated the novel geodesic clustering with tangent information (GCT) algorithm to solve the multi-geodesic modeling (MGM) problem; a special case of the MMM task. GCT thoroughly exploits the geometry of the data to build a similarity matrix that can effectively cluster the data (via spectral clustering) even when the underlying submanifolds intersect or have different dimensions. In particular, it leverages the novel idea of exploiting directional information from local tangent spaces to avoid neighboring points of clusters, different than that of the query point, and to establish theoretical guarantees for the theoretical GCT (TGCT) method. Unlike TGCT, GCT combined directional information from local tangent spaces with sparse coding aiming at a two-pronged objective: (i) to improve clustering results by using succinct representations of the underlying low-dimensional structures, and (ii) to enhance robustness against corruption. Geodesic information is only used locally; hence, the proposed framework can be applied not only to MGM but also to the more general MMM context. Validated against state-of-the-art clustering methods for the non-Euclidean setting, GCT exhibited superior performance in clustering accuracy on a wide variety of data-sets and Riemannian manifolds.

References

E. Arias-Castro, G. Chen, and G. Lerman. Spectral clustering based on local linear approximations. *Electron. J. Statist.*, 5:1537–1587, 2011.

E. Arias-Castro, G. Lerman, and T. Zhang. Spectral clustering based on local PCA. *ArXiv e-prints*, 2013.

R. Basri, T. Hassner, and L. Zelnik-Manor. Approximate nearest subspace search. *IEEE Trans. Pattern Analysis Machine Intell.*, 33(2):266–278, 2011.

H. E. Cetingul and R. Vidal. Intrinsic mean shift for clustering on Stiefel and Grassmann manifolds. In *Proc. CVPR*, pages 1896–1902, June 2009.

H. E. Cetingul, M. J. Wright, P. M. Thompson, and R. Vidal. Segmentation of high angular resolution diffusion MRI using sparse Riemannian manifold clustering. *IEEE Trans. Medical Imaging*, 33(2): 301–317, Feb. 2014.

G. Chen and G. Lerman. Spectral curvature clustering (SCC). *Int. J. Comput. Vision*, 81:317–330, 2009a.

G. Chen and G. Lerman. Foundations of a multi-way spectral clustering framework for hybrid linear modeling. *Found. Comput. Math.*, 9(5):517–558, 2009b.

G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto. Dynamic textures. *Intern. J. Computer Vision*, 51(2): 91–109, 2003.

E. Elhamifar and R. Vidal. Sparse subspace clustering. In *Proc. CVPR*, 2009.

E. Elhamifar and R. Vidal. Sparse manifold clustering and embedding. In *Proc. NIPS*, pages 55–63, 2011.

B. Ghanem and N. Ahuja. Maximum margin distance learning for dynamic texture recognition. In *Proc. ECCV (2)*, pages 223–236, 2010.

A. Goh and R. Vidal. Clustering and dimensionality reduction on Riemannian manifolds. In *Proc. CVPR*, pages 1–7, June 2008.

A. Goldberg, X. Zhu, A. Singh, Z. Xu, and R. Nowak. Multi-manifold semi-supervised learning. In *Proc. CVPR*, volume 5, pages 169–176, 2009.

P. Gruber and F. J. Theis. Grassmann clustering. In *Proc. EUSIPCO*, 2006.

M. T. Harandi, C. Sanderson, C. Shen, and B. C. Lovell. Dictionary learning and sparse coding on Grassmann manifolds: An extrinsic solution. In *Proc. ICCV*, 2013.

G. Haro, G. Randall, and G. Sapiro. Stratification learning: Detecting mixed density and dimensionality in high dimensional point clouds. *Neural Information Processing Systems*, 2006.

S. R. Hauberg, O. Freifeld, and M. J. Black. A geometric take on metric learning. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2024–2032. Curran Associates, Inc., 2012.

J. Ho, Y. Xie, and B. Vemuri. On a nonlinear generalization of sparse coding and dictionary learning. In *Proc. ICML*, volume 28, pages 1480–1488, 2013.

D. Kushnir, M. Galun, and A. Brandt. Fast multiscale clustering and manifold identification. *Pattern Recognition*, 39(10):1876–1891, 2006.

G. Lerman and J. T. Whitehouse. On d-dimensional d-semimetrics and simplex-type inequalities for high-dimensional sine functions. *J. Approx. Theory*, 156 (1):52–81, 2009.

G. Lerman and T. Zhang. Robust recovery of multiple subspaces by geometric l_p minimization. *Annals Statist.*, 39(5):2686–2715, 2011.

G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Analysis Machine Intell.*, 35, 2013.

Y. M. Lui. Advances in matrix manifolds for computer vision. *Image Vision Comput.*, 30(6-7):380–388, 2012.

F. Mémoli and G. Sapiro. Distance functions and geodesics on submanifolds of rd and point clouds. *SIAM Journal of Applied Mathematics*, 65(4):1227–1260, 2005.

S. O'Hara, Y. M. Lui, and B. A. Draper. Unsupervised learning of human expressions, gestures, and actions. In *Proc. Automatic Face Gesture Recognition and Workshops*, pages 1–8, March 2011.

X. Pennec. Emerging trends in visual computing. chapter Statistical Computing on Manifolds: From Riemannian Geometry to Computational Anatomy, pages 347–386. Springer-Verlag, Berlin, Heidelberg, 2009. ISBN 978-3-642-00825-2. doi: 10.1007/978-3-642-00826-9_16.

M. Soltanolkotabi and E. J. Candès. A geometric analysis of subspace clustering with outliers. *Ann. Stat.*, 40(4):2195–2238, 2012.

M. Soltanolkotabi, E. Elhamifar, and E. J. Candès. Robust subspace clustering. *Ann. Statist.*, 42(2): 669–699, 2014.

R. Subbarao and P. Meer. Nonlinear mean shift for clustering over analytic manifolds. In *Proc. CVPR*, volume 1, pages 1168–1175, June 2006.

J. Y. Tou, Y. H. Tay, and P. Y. Lau. Gabor filters as feature images for covariance matrix on texture classification problem. In *Advances in Neuro-Information Processing*, volume 5507 of *Lecture Notes in Computer Science*, pages 745–751. Springer Berlin Heidelberg, 2009.

P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa. Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition. *IEEE Trans. Pattern Analysis Machine Intell.*, 33(11):2273–2286, 2011.

O. Tuzel, R. Subbarao, and P. Meer. Simultaneous multiple 3D motion estimation via mode finding on Lie groups. In *Proc. ICCV*, volume 1, pages 18–25, Oct. 2005.

R. Vidal. Subspace clustering. *SPM*, 28:52 –68, 2011.

X. Wang, K. Slavakis, and G. Lerman. Riemannian multi-manifold modeling. *ArXiv e-prints*, 2014.

Y. Wang and G. Mori. Human action recognition by semilatent topic models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(10):1762–1774, 2009.

Y. Wang, Y. Jiang, Y. Wu, and Z.-H. Zhou. Spectral clustering on multiple manifolds. *IEEE Trans. Neural Networks*, 22(7):1149–1161, 2011.

T. Zhang, A. Szlam, Y. Wang, and G. Lerman. Hybrid linear modeling via local best-fit flats. *Intern. J. Computer Vision*, 100:217–240, 2012.