Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

Intersecting Manifolds: Detection, Segmentation, and Labeling

Shay Deutsch and Gérard Medioni
University of Southern California
Los Angeles, USA
{shaydeut, medioni}@usc.edu

Abstract

Solving multi-manifolds clustering problems that include delineating and resolving multiple intersections is a very challenging problem. In this paper we propose a novel procedure for clustering intersecting multi-manifolds and delineating junctions in high dimensional spaces. We propose to explicitly and directly resolve ambiguities near the intersections by using 2 properties: One is the position of the data points in the vicinity of the detected intersection; the other is the reliable estimation of the tangent spaces away from the intersections. We experiment with our method on a wide range of geometrically complex settings of convoluted intersecting manifolds, on which we demonstrate higher clustering performance than the state of the art. This includes tackling challenging geometric structures such as when the tangent spaces at the intersections points are not orthogonal.

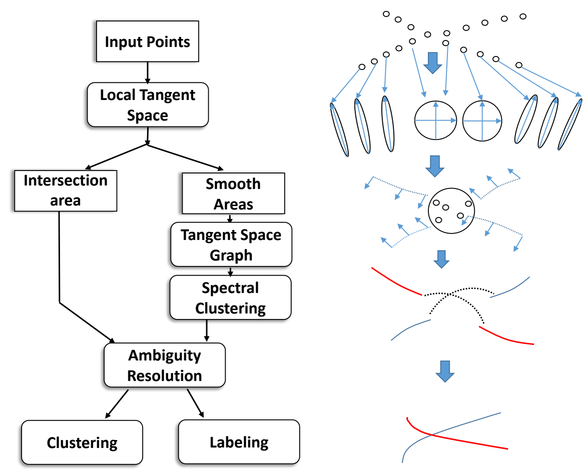

Figure 1: Flow chart of the proposed method

1 Introduction

Resolving intersections in high dimensional spaces is essential in multi-manifold clustering problems that arise in many applications, such as motion segmentation in computer vision [Elhamifar and Vidal, 2009]. Recently, a number of multi-manifold clustering algorithms were proposed, in which a multi-way affinity measure between data points was suggested to capture complex structure in the data. Typically, such methods [Wang *et al.*, 2011] [Gong *et al.*, 2012] [Goldberg *et al.*, 2009], [E.Arias-Castro and Zhang, 2013] construct affinity which is based on local tangent space distance, in addition to their Euclidean distance. However, despite important progress made by this research, they only provide satisfactory results when the angle between the tangent planes is large (typically $> \pi/4$). Moreover, recent work shows that even though relatively few points may be located near the intersections, their contribution to the global structure can be very disruptive to the global manifold structure estimation [Belkin *et al.*, 2012]. Our framework to handle intersecting manifolds or manifolds with singularities is different from previous research in clustering multi-manifolds [Wang *et al.*, 2011] [Gong *et al.*, 2012] [Goldberg *et al.*, 2009], [E.Arias-Castro and Zhang, 2013], as we explicitly and directly resolve the ambiguities near the intersections. In particular, we argue that the position of the points on the manifolds near the intersections contain valuable information that is necessary to achieve high-performance clustering. To resolve complex geometric structures, we suggest decomposing the problem into three main stages (see Figure 1 for an illustration of our overall approach): Given a set of unlabeled points with unknown geometric structure, we first employ a data driven approach, Tensor Voting [Mordohai and Medioni, 2006] - which uses the direct communication between data points to inform whether such intersections occurred, and, most importantly, provide a reliable estimation of the local support of the intersections. Using the smooth manifolds parts, we construct a graph in which the affinities between the data points are based on a local tangent space distance. The smooth manifold parts are then extracted using Spectral Clustering. The next stage performs ambiguity resolution algorithm in the local singularity area, using the classified smooth manifolds and the positions of the points near the singularities.

We show the advantage of our explicit and direct approach in resolving manifolds intersections for a wide range of complex geometric settings, which outperforms the state of the art methods in multi-manifold clustering. We summarize our contributions:

- A general approach to the intersecting multi-manifold clustering problem: although the local intersection area constitute a small part of the manifolds, we show that it contain critical information which is necessary to achieve good clustering performance.
- Handling large amounts of outliers: our suggested method can perform with high accuracy even in the presence of large amounts of outliers.
- Validation on complex dataset: we demonstrate that by using designed tools and algorithms to learn both the local and global structures of the manifolds, higher clustering performance result can be achieved, even when the tangent spaces at the intersections points are not orthogonal.

The paper is organized as follows: Section 2 provides an overview of previous work in multi-manifold clustering methods. Section 3 includes a brief introduction to Tensor Voting [Mordohai and Medioni, 2006] and the Tensor Voting Graph(TVG) [Deutsch and Medioni, 2015], which serves as core tools in our framework. Section 4 details the proposed approach which consist of two steps, intersection delineation, and ambiguity resolution, as illustrated in Figure 1. Section 5 demonstrates the experimental results of our method and compares it to the state of the art algorithms in clustering intersecting multi-manifolds. Section 6 concludes the paper and proposes future work.

2 Related work

The multi-manifold case addresses a general setting where the clusters are low dimensional manifolds that may intersect or overlap. Many situations exist where the data is formed by a number of manifolds. The complexity of the multi-manifold class of distributions is ruled by the minimum of the manifold curvatures, branch separations, and the overlap between distinct manifolds [Goldberg *et al.*, 2009]. Early methods in multi-manifold clustering such as [Zelnik-manor and Perona, 2004] assumed that the manifolds are well separated. Generalized PCA [Vidal *et al.*, 2003] and Sparse Subspace Clustering [Elhamifar and Vidal, 2009] were suggested to address clustering of intersecting linear multi-manifolds. Recently, a number of methods were suggested to address the challenging problem of non-linear intersecting multi-manifolds. [Goldberg *et al.*, 2009] developed a spectral clustering method within a semi-supervised learning framework. As a complementary approach, Robust Multiple Manifold Structure Learning [Gong *et al.*, 2012], Spectral clustering on multiple manifolds [Wang *et al.*, 2011] and Spectral Clustering using local PCA [E.Arias-Castro and Zhang, 2013] are unsupervised learning methods which propose similar approaches for clustering intersecting manifolds. Spectral Clustering using local PCA also provides a deep and elegant theoretical analysis for multi-manifold learning in the context of resolving intersections. Note however that the algorithms suggested in [Goldberg *et al.*, 2009] use a coarsening step, which can hinder a careful treatment of the intersections. The Tensor Voting Graph (TVG) [Deutsch and Medioni, 2015], was suggested to address the limitation of the local Tensor Voting method and can perform global operations such as estimating geodesic distance or clustering on single or multi-manifolds which are intersecting. However, similar to other multi-manifold learning algorithms, the TVG does not address intersections explicitly.

3 Tools for Geometric Structure Estimation

Estimation of the local geometric structure, which includes the local tangent space estimation and the identification of the local intersection area can be performed using local PCA [Zhang and Zha, 2005] or Tensor Voting. We have evaluated both Tensor Voting and Local PCA on a number of synthetic datasets, and found that the local tangent space estimation accuracy was higher using Tensor Voting than local PCA. Thus we use Tensor Voting to estimate the local geometric structure. In the following section we provide a brief introduction to Tensor Voting and the Tensor Voting Graph [Deutsch and Medioni, 2015], which provides an efficient tool to learn the global geometric structure. We refer to [Mordohai and Medioni, 2010],[Mordohai and Medioni, 2006],[Mordohai and Medioni, 2005] for a detailed treatment on Tensor Voting.

3.1 Tensor Voting

The Tensor Voting methodology consists of three important aspects [Mordohai and Medioni, 2010]:

1.**Tensor for representation**: each point is encoded as a second order, positive semi definite symmetric tensor, which is equivalent to an $N \times N$ matrix, and an ellipsoid in N-D space. In the Tensor Voting framework, a tensor represents the structure of a manifold going through the point by encoding the normals to the manifold as eigenvectors of the tensor that correspond to non-zero eigenvalues, and the tangents as eigenvectors that correspond to zero eigenvalues. The tensors can be formed by the summation of the direct products of the eigenvectors that span the normal space of the manifold. The tensor T at a point on a manifold of dimensionality d and with $\hat{n}_i$ corresponding to the unit vectors that span the normal space is expressed as $T = \sum_{i=1}^{d} \hat{n}_i \hat{n}_i{}^t$.

2. **Voting for communication**: The core of the Tensor Voting framework is the way information is propagated from point to point. Given a tensor at O and a tensor at P, the vote the point at O (the voter) casts to P (the receiver) has the orientation the receiver would have, if both the voter and receiver belong to the same structure. The stick tensor voting is the fundamental voting element from which all other voting types and voting in higher dimensions can be derived. The following equation defines the stick tensor voting:

$$S_{vote} = DF(s,k,\sigma) \begin{bmatrix} -sin(2\theta) \\ cos(2\theta) \end{bmatrix} \begin{bmatrix} -sin(2\theta) & cos(2\theta) \end{bmatrix} \tag{1}$$

$$DF(s,k,\sigma) = e^{-\frac{s^2+ck^2}{\sigma^2}}$$

$$\theta = \arcsin(\frac{\vec{v}\hat{e_1}}{\|\vec{v}\|}), s = \frac{\theta||\vec{v}||}{\|\text{sin}\theta\|}, \kappa = \frac{\|2\text{sin}\theta\|}{||\vec{v}||}$$

In the above equation, s is the length of the arc between the voter and receiver (OP), $\vec{v}$ is the vector connecting O and P, e_i

is the normal vector at the voter, κ is its curvature which can be computed from the radius of the osculating circle, σ is the scale of voting, which controls the degree of decay with curvature, and c is a constant defined in [Mordohai and Medioni, 2010]. The magnitude of the vote is a function of proximity and smooth continuation, and is called the saliency decay function. No votes are cast if the receiver is at an angle larger than $45°$ with respect to the tangent of the osculating circle at the voter, in order to limit votes that are due to high curvature or from unrelated points.

3. **Voting analysis**: given an $N \times N$ second order, symmetric, non-negative definite matrix, the type of structure encoded in it can be inferred by examining its eigensystem. Any such tensor can be decomposed as in the following equation:

$$\begin{aligned} T = \sum \lambda_i \hat{e_i}\hat{e_i}^t = (\lambda_1 - \lambda_2)\hat{e_1}\hat{e_1}^t + \\ +(\lambda_2 - \lambda_3)(\hat{e_1}\hat{e_1}^t + \hat{e_2}\hat{e_2}^t) + \ldots \\ \lambda_N(\hat{e_1}\hat{e_1}^t + \hat{e_2}\hat{e_2}^t + \ldots\hat{e_N}\hat{e_N}^t) \end{aligned}$$

where λ_i are the eigenvalues in descending order of magnitude and e_i are the corresponding eigenvectors. Based on the tensor spectral decomposition, the normal and tangent spaces, structure type, dimensionality and outliers are derived. The term $(\hat{e_1}\hat{e_1}^t + \hat{e_2}\hat{e_2}^t + \ldots\hat{e_N}\hat{e_N}^t)$ is called the ball component and is typically used to identify intersection areas, which correspond to peaks of the eigenvalue λ_N.

The main limitation of the Tensor Voting framework is that it is a strictly local method, and performing global operations such as estimating geodesic distances and clustering are not reliable. For example, to estimate geodesics distances on manifolds, previous methods using TV resort to an iterative, non-linear interpolation methods [Mordohai and Medioni, 2010] that marches on the manifold by projecting the desired direction from the starting point. As pointed out in [Mordohai and Medioni, 2010], this process is very slow and unreliable, and also diverges in configurations where points on the path are in deep concavities.

3.2 The Tensor Voting Graph (TVG)

The TVG [Deutsch and Medioni, 2015] employs a graph construction in which the affinity between points on the graph corresponds to the contribution that was made to the tangent space point estimation by the neighboring points that participated in the voting process. Thus in TVG the affinity between two points X_i and X_j summarizes the contribution made to the normal space estimation of X_i by the votes emitted from X_j at X_i using Tensor Voting. Formally, given the normal space of X_i, $N(X_i) = \{\hat{n_1}, ..\hat{n_d}\}$ and the subspace $\tilde{N_j}(X_i) = \left\{\hat{\hat{n_1}}, ..\hat{\hat{n_d}}\right\}$ representing the vectors votes emitted from X_j at X_i ; the affinity value w_{ij} between X_i and X_j is given by:

$$w_{ij} = \begin{cases} |\langle \hat{n}^{max}, \hat{\hat{n}}^{max} \rangle|, & \text{if } \arccos|\langle \hat{n}^{max}, \hat{\hat{n}}^{max} \rangle| < 45° \\ 0, & \text{else} \end{cases} \quad (2)$$

Where X_j is in $k_{nn}(X_i)$ - the k nearest neighbors of X_i, and $\hat{n}^{max}, \hat{\hat{n}}^{max}$ are the vectors corresponding to the maximal principal angle between the subspaces, $N(X_i)$, $\tilde{N_j}(X_i)$, respectively. We note that thanks to the duality between the tangent and normal spaces, we can use the angle between normal spaces and tangent spaces interchangeably. From now on we shall use the term angle between tangent spaces since it is more intuitive and commonly used in the literature. Similar to other multi-manifold learning algorithms, the TVG does not address intersections explicitly, and therefore suffers from similar shortcomings - the failure to handle manifolds intersecting at small principal angles, and distortion around the local intersection area, which we now address.

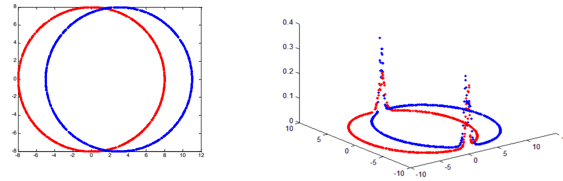

Figure 2: The graph of the ball eigenvalue λ_N (shown on the right hand side), as a function of the positions of the points which correspond to two intersecting circles(shown on the left hand side)

4 Intersecting Manifolds

We suggest a process that directly untangles the ambiguities in the local intersection area by aggregating support from the smooth manifolds parts. Our framework has three main processing steps, which we detail in the following sections:

4.1 Intersection Delineation

The first step in our process is to estimate the dimensionality, tangent space and normal space at every point using Tensor Voting. Given a set of unlabeled points, $X = \{X_i\}_{i=1}^n, X_i \in \mathbb{R}^{\mathbb{N}}$, which are lying on K smooth intersecting manifolds $M_1, ..M_K$. Let $X_J = \{X_j \in M_i \cap M_j, \ |M_i \cap M_j \neq \emptyset\}$ denote the set of intersection points. The set of points which correspond to the manifolds intersections support will be referred as the decision set points, and are defined as

$$\tilde{X}_J = \{X_j \in k_{nn}(X_i) | X_i \in X_J\} \quad (3)$$

To delineate intersections and their local support, we analyze the Tensor at each point X_i. Votes are inconsistent only in the area of intersection, which is characterized by sharp transitions of eigenvalues in non-smooth parts; There are two alternatives to identify the local intersection area. Eigenvalue λ_N is adequate to identify local intersections area in any latent dimension, since normal votes are received in the local intersection area at different angles and directions from points lying on a different manifold (see Figure 2 for illustration of the ball component eigenvalue as a function of the position of the two intersecting circles). The second alternative is to use the eigenvalue $\lambda_d + 1$, where d correspond to the normal dimension of the manifold, to identify the local intersection area. In the smooth parts, the eigenvalue $\lambda_d + 1$ is very small, while in the local intersection area the dimensionality of the normal space is increased by 1 and hence the corresponding eigenvalue $\lambda_d + 1$ will be significantly larger than

Algorithm 1 Ambiguity Resolution Algorithm

Input: Labeled manifolds $\{M_r, T_{M_r}\}_{r=1}^{K}$, unlabeled intersection area points $\tilde{X}_J$, nearest neighbor parameter k .

1 : Set $X_J^{new} = \tilde{X}_J$.

while $X_J^{new} \neq \emptyset$ **do**

2: Set $i = 1, X_J^{new} = \tilde{X}_J$, find $\hat{X}_{j*} = min||X_i - \hat{X}_j||_2$, $X_i \in M_i$, $\hat{X}_j \in \tilde{X}_J$.

3: Extract the sub-manifolds $\tilde{M}_r = \{X_r \in k_{nn}(X_{j*}) \,|\, X_r \in M_r\}$ for all $r = 1...K$

4: Estimate $T_{\tilde{M}_r}(\hat{X}_{j*})$, for all $r = 1...K$.

5: Compute $\phi_r(\hat{X}_{j*}) = \sum_{j=1}^{k} arccos(|\langle \hat{n}^{max}_{\tilde{M}_r r}(X_j), \hat{n}^{max}_{\tilde{M}_r}(\hat{X}_{j*})\rangle|)$, for all $r = 1...K$

6: Add $\hat{X}_{j*} \in M_j$, s.t. $\phi_j(\hat{X}_{j*}$ =min $\left\{\phi_r(\hat{X}_{j*}), r = 1..K\right\}$.

7: Update: $X_J^{new} = X_J^{new} \backslash \hat{X}_{j*}$

8: Process steps (3-7) with $i = i + 1$ if $i < K$, else $i = 1$ if $i = K$

end

Output: Labeled local intersections area points $\left\{\hat{X}_{ji}\right\}_{i=1}^{K} \in M_i$, and their corresponding tangent spaces $\left\{T_{M_i}(\hat{X}_{ji})\right\}_{i=1}^{K}$

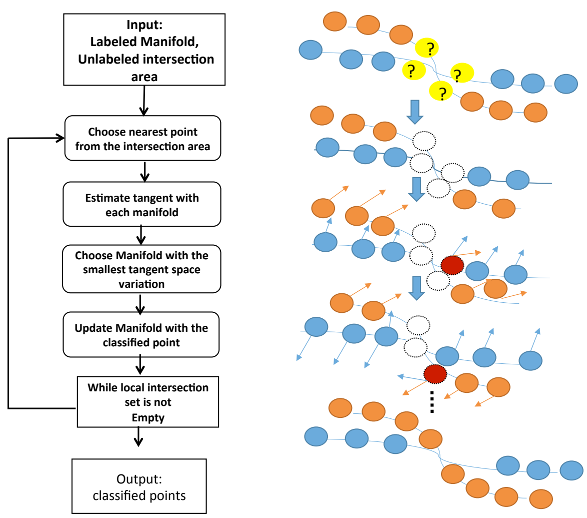

Figure 3: Flow chart of the proposed Ambiguity Resolution Algorithm

in the smooth parts. Note that in Figure 2 these two cases coincide since $\lambda_d + 1$ equals the dimensionality of the ball eigenvalue.

To estimate which points correspond to the local intersection area, we compute the standard deviation σ of the ball eigenvalue λ_N which correspond to all points:

$$\sigma = (\frac{1}{n}\sum_{i=1}^{n}(\lambda_N(X_i) - \bar{\lambda}_N)^2)^{\frac{1}{2}} \tag{4}$$

where $\lambda_N(X_i)$ correspond to the ball eigenvalue of point X_i and $\bar{\lambda}_N$ correspond to the mean of the ball eigenvalues. We identify points X_i that belong to the local intersection area if $\lambda_N(X_i) > 2\sigma$, and all such points are removed for further processing, since their geometric structure information is not reliable. Note also that this threshold is not critical, since the transitions are sharp and distinctive, and only the local intersection area points are characterized by high values of the eigenvalue λ_N.

4.2 Global representation of the smooth manifolds parts

The second stage is to infer the global structure of the smooth manifolds parts, from which the intersections areas were removed. The TVG (X_C, W_C) is constructed for points corresponding to the smooth manifold parts X_C, $(X_C = X \setminus \tilde{X}_J)$ such that the local intersection area points $\tilde{X}_J$ are removed from X. Finally spectral clustering is applied to the affinity matrix W_C to classify points to manifold labels.

4.3 Ambiguity Resolution

We elaborate on iterative algorithm that can be cast as a semi-supervised learning algorithm that is incrementally aggregating support from the labeled smooth manifolds parts, to determine the labels and geometric structure of the local intersection area. Based on the manifolds smoothness properties, in the local intersection area the local tangent space variation is smaller among pairwise points which belong to the same manifold. A through theoretical analysis of the sufficient smoothness conditions will be provided in a forthcoming technical report.

Formally, our objective is to reconstruct the labels of the decision set points $\tilde{X}_J$ and their corresponding tangent spaces $T(\tilde{X}_J)$ such that manifold smoothness is maximized in the local intersection area. This task can be performed by minimizing the total variation of the tangent spaces. For each point at the local intersection area, we estimate its local tangent space independently by using each of the nearby manifolds (which are known at this stage) and assign it to the manifold for which the total tangent space variation was minimal.

Algorithm Description We describe the algorithm for reconstructing the decision set points (the flow chart of the algorithm is illustrated in Figure 3).

Let X_C, be the labeled manifolds data , and let $\{T_{M_i}(X_C)\}$ be their corresponding tangent spaces. $G_C = (X_C, W_C)$ is the Tensor Voting Graph, with W_C corresponding to the affinity matrix between the labeled manifolds. $\tilde{X}_J$ is the set of unlabeled points which correspond to the local

intersection area. $G_C = (X_C, W_C)$ together with the positions of the local intersection area $\tilde{X}_J$ serve as an input to the ambiguity resolution algorithm. The goal is to find the true labels of the points in the local intersection area, and obtain a reliable estimation of their tangent spaces. We begin with selecting a point X^* from the local intersection area which is the nearest neighbor to one of the manifolds: $X^* = min||X_C - \hat{X}_j||_2, \hat{X}_j \in \hat{X}_J$, and compute its tangent spaces $T_{M_1}(X^*), T_{M_2}(X^*), ...T_{M_K}(X^*)$ induced by its k nearest neighbors in each one of the manifolds $M_1, M_2, ..M_K$. We then classify X^* to belong to the manifold M^* for which the tangent space variation $\phi(X^*) = \sum_{j=1}^{k} arccos(|\langle \hat{n}_{M^*}^{max}(X_j), \hat{n}_{M^*}^{max}(\hat{X}^*)\rangle|)$ was minimal. We add X^* and $T_{M^*}(X^*)$ to the corresponding manifold M^* and remove X^* from the decision set $\tilde{X}_J^{new} = \tilde{X}_J \setminus X^*$. In a similar way we process all the remaining decision set points $\tilde{X}_J^{new}$ until the procedure is exhausted. The output is the labels of the entire decision set points and their corresponding tangent spaces.
In the suggested greedy algorithm, computational complexity amounts to estimating the tangent space using Tensor Voting for all the local intersection area points, which requires only $O(jNklogk)$, where N is the dimension of the ambient space , k corresponds to the number of k nearest neighbors, and j is the number of local intersection area points, which typically constitutes a small portion of the total number of points n. Also note that the complexity is $O(Nnlogn)$ for the Tensor Voting computation [Mordohai and Medioni, 2010] and $O(n^2N^2d)$ for computing the affinity between the local tangent spaces, where d corresponds to the normal space dimensionality.

5 Experimental Results

We experimented with synthetic and real data sets of various challenging geometric configurations, such as when the maximal principal angle between the tangent spaces at the intersections points is smaller than 40 degrees.

5.1 Experimental Results without outliers

For comparison and evaluation with the state of the art, we experimented with the following datasets: (1) two circles intersecting at 18 degrees, (2) two planes intersecting at 40 degrees, (3) two Mobius bands, (4) two intersecting spheres, and (5) a Swiss roll intersecting with a plane. The manifolds were uniformly sampled with n=1000 points for each plane, circle and spheres, n=2000 points for the Mobius bands, n=2000 points for the Swiss roll. Each simulation was repeated 10 times. We also compared our method to state of the art algorithms in clustering multiple manifolds, Spectral clustering on multiple manifolds (SMMC) [Wang *et al.*, 2011], and SSC [Elhamifar and Vidal, 2009], which is a state of the art method in clustering linear intersecting manifolds, For the choice of parameters we tested the k nearest neighborhood size $k \in \{10, 20, 30, 40, 50, 60, 70, 80\}$. For the second parameter in SMMC and SSC we tested in $\{10, 20, 30, 40, 50, 60, 70, 80\}$ and $\{0.001, 0.001, 0.1, 1\}$, respectively.The results are reported for the best choice of parameters for each method.

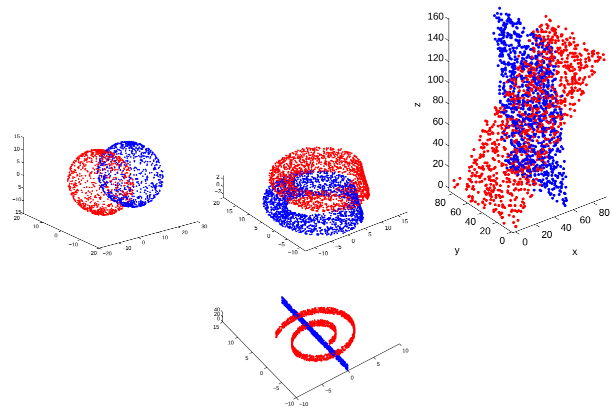

Figure 4: Manifolds dataset used in table 2

Note that Sparse Subspace Clustering [Elhamifar and Vidal, 2009] was only compared to the case of the intersecting planes since it is only adequate to handle linear manifolds. In our method, we chose a scale σ such that the average number of votes from each point in the Tensor Voting iteration equals to $n/20$, and the number of k nearest neighbors on the Tensor Voting was tested in $\{n/40, n/40+5, n/40+10\}$. We report the classification accuracy percent in each dataset both for the set of points which correspond to the area near the intersection in addition to the rest of the points. Note that the most relevant statistics is the clustering accuracy in the area near the intersections. The comparison results in table 1 show that our method consistently outperforms the state of the art both near the intersection areas and in the smooth areas, and in particular for the challenging geometric setting where the principal angle at the intersection point is smaller than $\pi/8$ (such as in the case of the intersecting planes or two circles).

Finally, we highlight both quantitative and qualitative differences between TVG and our new approach. Table 4 shows the tangent space average angular error for two intersecting planes and the two circles using Tensor Voting and the new proposed method. Even though the average error obtained using the standard TV seems relatively marginal, the clustering performance using TVG deteriorates (which is also the case for all the other existing methods) as the principal angles becomes smaller. Using the new approach, the error of the tangent space is reduced and the clustering results are significantly improved . We also note that the choice of parameters is not critical, and is robust against a wide range of parameter selection for the k nearest neighbors on the graph and the scale of Tensor Voting. However, truly automatic parameter selection remains an open problem for future research, which is also the case in all existing intersecting manifolds algorithms [Goldberg *et al.*, 2009] and [E.Arias-Castro and Zhang, 2013].

Dataset / Method	SSC[Elhamifar and Vidal, 2009]		SMMC[Wang *et al.*, 2011]		TVG+Ambiguity Resolution	
	outside intersection area	Intersection area	outside intersection area	Intersection area	outside intersection area	Intersection area
Two circles	-	-	69.47%	59.09%	**100%**	**99.22%**
Two Mobius bands	-	-	95.14%	75.3%	**99.98%**	**98.44%**
Two spheres	-	-	96.79%	80.33%	**100%**	**98.58%**
Two planes	71%	59.58%	72.22%	59.58%	**99.9%**	**96.07%**
Swiss Roll and a plane	-	-	96.5%	95.57%	**99.95%**	**95.9%**

Table 1: comparison with state of the art

Dataset / Method	SMMC [Wang *et al.*, 2011]		TVG + Ambiguity Resolution	
	outside intersection area	Intersection area	outside intersection area	Intersection area
Two circles	65.91%	59.72%	**99.75%**	**94.5%**
Two Mobius bands	87.09%	62.31%	**99.94%**	**98.09 %**
Two spheres	54.9%	53.9%	**99.61%**	**90.05 %**
Two planes	60%	58%	**94.16%**	**99.94 %**
Swiss Roll and a planes	59.03%	52.37%	**98.34%**	**97.29 %**

Table 2: Comparison results in the presence of outliers

Dataset / Method in high dimensional space	Spectral Clustering on multiple manifolds [Wang *et al.*, 2011]		TVG + Ambiguity Resolution	
	outside intersection area	Intersection area	outside intersection area	Intersection area
2D Sphere embedded in 50D	95.56%	94.04%	**97.64%**	**99.3%**
3D Hyper Sphere embedded in 50D	85.5%	62.16%	**98.71%**	**94.2%**

Table 3: Manifolds in high dimensional space

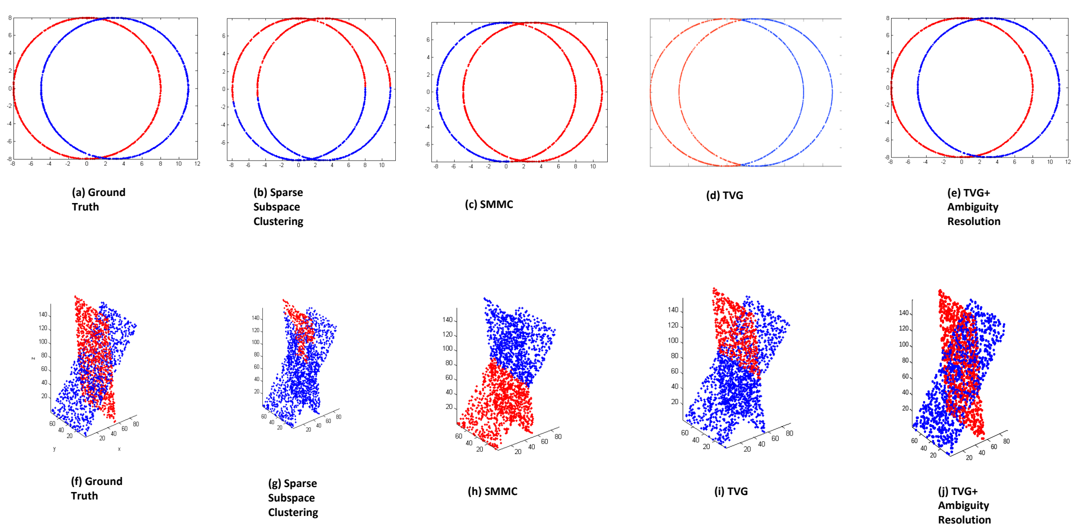

Figure 6: Evaluation on challenging dataset of manifolds with small maximal principal angle reveal the degradation in performance of both linear and non-linear multi-manifold clustering methods

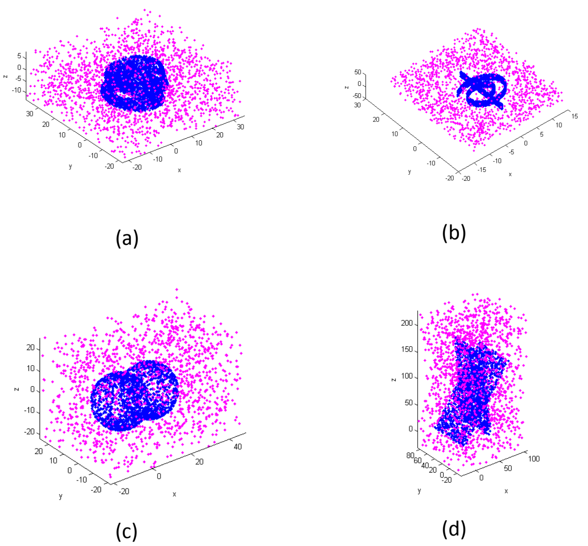

Figure 5: Figures of manifolds with outliers used in the experiments :(a) Intersecting Mobius bands (b) Intersecting Swiss Roll with a plane (c) Intersecting Spheres (d) Intersecting planes

5.2 Experiments with outliers

We also apply our method in the presence of a large amount of outliers. Robustness to outlier noise is a critical issue in manifold clustering and current methods are very sensitive in their presence. This shortcoming was pointed out in previous works [Wang *et al.*, 2011], [Gong *et al.*, 2012] as a challenging open problem and was partially addressed in [Gong *et al.*, 2012], however using only a number of outliers which is 10% of the number of inliers. The Tensor Voting framework, on the other hand, is robust to outliers and since it serves as an integral part of our method, removing the outliers by examining the eigenvalues of the Tensors obtained after the first TV iteration is straightforward to incorporate in our scheme. We experiment our method with outlier noise using the same intersecting manifolds experimented in the previous section. The two circles, Mobius bands, spheres, two planes and Swiss roll intersecting with a plane manifolds are contaminated with 1000 and 2000, 1500, 1500, and 1500 outliers, respectively. To remove outliers, the eigenvalues which correspond to the eigenvalue λ_1 of the tensor at each point are sorted. We remove the points which correspond to the smallest sorted eigenvalues. The experimental results, shown in table 2 demonstrate that our method is robust against outliers while it severely affects the results of the other methods.

5.3 Experiments with manifolds embedded in high dimensional spaces

We also apply our method on manifolds which reside in a high dimensional space for two cases including 1) intersecting spheres corresponding to 2D manifolds and 2) intersecting hyper spheres corresponding to 3D manifolds. In each case, these manifolds were generated using uniform sampling with 2,000 samples for each manifold in 3D and 4D, which were then embedded in 50-D by using a random orthonormal matrix. Experimental results demonstrated in table 3, show that our method remains robust when applied in the high dimensional space, both in the area near the intersections and in the smooth parts.

5.4 Experiments with Real Data sets

For experiments with real data-sets, we tested our method on the problems of human action classification and two view motion segmentation problems.

Motion Capture using the CMU Motion capture data set Classification of human motion sequences as a prepossessing step is important for many tasks in video annotation. The CMU motion capture data-set is a popular and widely used real data set for motion capture. In order to perform evaluation in a strictly unsupervised framework, we remove the temporal information from the data, thus the data provided correspond to static information. In this case, the problem can be considered as clustering multiple manifolds with edge singularity type, which correspond to abrupt change due to a transition of a human action to a different motion activity.

We choose five mixed sequences from subject 86, which includes mixed activities such as walking, turning around, sitting, running, jumping, squats, and stretching. We extract approximately 500 frames per each sequence since it correspond to two or three distinct motion activities. Each point correspond to a human pose, which is represented by 62 dimensional feature vector. The experimental results comparisons in table 5 show that our method outperforms the state of the art. The errors obtained using our framework correspond to the frames which occur during transitions between different motion activities, which are difficult also for humans to evaluate.

Motion segmentation using 155 motion segmentation benchmark Next we show evaluation on the problem of motion segmentation from two-views, using the 155 motion segmentation data-set benchmark, which is a well known dataset for motion segmentation. Trajectory based motion segmentation is a classic and fundamental problem in computer vision which is important for understanding dynamic scenes. We evaluate our method on two image sequences with perspective effects [Li *et al.*, 2013], and compare them to SSC [Elhamifar and Vidal, 2009], which showed state of the art results in the case of motion segmentation based on feature trajectory. The problem of segmenting motions using only 2-views is a challenging task since the feature trajectories lie on quadratic surfaces of dimension at most 3 in $\mathbb{R}^4$ [Arias-Castro, 2011] which may be overlapping or intersecting. Applying our method for motion segmentation achieves an average classification error of 10.8% outperforming SSC which obtained 20.43% classification error.

Method/sequence	two circles	two planes
Tensor Voting	**0.4%**	**1.8%**
New Method	**0.01%**	**0.17%**

Table 4: Tangent space average angular error results for the two intersecting planes and intersecting circles data(Figure 6)

Data/Method	SMMC	TVG + Ambiguity resolution
CMU MoCap	87.06%	**96.01%**

Table 5: Classification results of human activities on Motion Capture data

6 Discussion and Future Work

We have presented a novel method for unsupervised clustering of intersecting multi-manifolds. Our framework extends previous research by explicitly addressing and resolving the ambiguities near the intersections, in convoluted geometric situations such as when the principal angle between the tangent spaces at the intersection is small, and also in the presence of large amount of outliers. Experimental results demonstrate that our method performs clustering with high accuracy in all of these situations, and significantly outperforms the state of the art. The main limitation of the current framework is robustness to inlier noise, where the method may fail is the presence of large amounts of noise in the intersection area itself. Future work includes testing our approach on additional applications such as molecular structure analysis in biology [W.Brown, 2008], using our model for offline training in multivariate time series for human motion [Gong and Medioni, 2011], and extending our framework to handle inlier noise in high dimensional spaces.

Acknowledgments

This work was supported in part by grant DE-NA0001683 from the U.S. Department of Energy.

References

[Anandan and Avidan, 2000] P. Anandan and Shai Avidan. Integrating local affine into global projective images in the joint image space. In *ECCV (1)'00*, pages 907–921, 2000.

[Arias-Castro, 2011] Chen G. Lerman G Arias-Castro, E. Spectral clustering based on local linear approximations. electron. In *Electron. J. Statist*, page 15371587, 2011.

[Belkin *et al.*, 2012] Mikhail Belkin, Qichao Que, Yusu Wang, and Xueyuan Zhou. Graph laplacians on singular manifolds: Toward understanding complex spaces: graph laplacians on manifolds with singularities and boundaries. *CoRR*, abs/1211.6727, 2012.

[Deutsch and Medioni, 2015] Shay Deutsch and Gerard Medioni. Unsupervised learning using the tensor voting graph. *SSVM 2015: Fifth International Conference on Scale Space and Variational Methods in Computer Vision*, 2015.

[E.Arias-Castro and Zhang, 2013] G.Lerman E.Arias-Castro and T. Zhang. Spectral clustering based on local pca. *ArXiv*, 2013.

[Elhamifar and Vidal, 2009] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *CVPR*, pages 2790–2797, 2009.

[Goldberg *et al.*, 2009] Andrew B. Goldberg, Xiaojin Zhu, Aarti Singh, Zhiting Xu, and Robert Nowak. Multi-manifold semi-supervised learning. In *AISTATS*, pages 169–176, 2009.

[Gong and Medioni, 2011] Dian Gong and Gerard Medioni. Dynamic manifold warping for view invariant action recognition. *Computer Vision, IEEE International Conference on*, 0:571–578, 2011.

[Gong *et al.*, 2012] D. Gong, X. Zhao, and G. Medioni. Robust multiple manifold structure learning. In *ICML*, 2012.

[Li *et al.*, 2013] Zhuwen Li, Jiaming Guo, Loong-Fah Cheong, and Steven Zhiying Zhou. Perspective motion segmentation via collaborative clustering. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 1369–1376, 2013.

[Mordohai and Medioni, 2005] Philippos Mordohai and Gérard G. Medioni. Unsupervised dimensionality estimation and manifold learning in high-dimensional spaces by tensor voting. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 798–803, 2005.

[Mordohai and Medioni, 2006] P. Mordohai and G. Medioni. *Tensor Voting: A Perceptual Organization Approach to Computer Vision and Machine Learning*. Morgan & Claypool Publishers, 2006.

[Mordohai and Medioni, 2010] P. Mordohai and G. Medioni. Dimensionality estimation, manifold learning and function approximation using tensor voting. *Journal of Machine Learning Research*, 11:411–450, 2010.

[Vidal *et al.*, 2003] R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca), 2003.

[Wang *et al.*, 2011] Y. Wang, Y. Jiang, Y. Wu, and Z. Zhou. Spectral clustering on multiple manifolds. *IEEE Transactions on Neural Networks*, 22(7):1149–1161, 2011.

[W.Brown, 2008] S.Pollock E.Coutsias JP.Watson W.Brown, S. Martin. Algorithmic dimensionality reduction for molecular structure analysis. *Journal of Chemical Physics*, 2008.

[Zelnik-manor and Perona, 2004] L. Zelnik-manor and P. Perona. Self-tuning spectral clustering. In *Advances in Neural Information Processing Systems*, pages 1601–1608, 2004.

[Zhang and Zha, 2005] Z. Zhang and H. Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal on Scientific Computing*, 26(1):313–338, 2005.