CLUSTERING OF MANIFOLD-MODELED DATA BASED ON TANGENT SPACE VARIATIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÖKHAN GÖKDOĞAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2017

Approval of the thesis:

CLUSTERING OF MANIFOLD-MODELED DATA BASED ON TANGENT SPACE VARIATIONS

submitted by **GÖKHAN GÖKDOĞAN** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver
Dean, Graduate School of **Natural and Applied Sciences** _______________

Prof. Dr. Tolga Çiloglu
Head of Department, **Electrical and Electronics Engineering** _______________

Assist. Prof. Dr. Elif Vural
Supervisor, **Electrical and Electronics Engineering, METU** _______________

Examining Committee Members:

Prof. Dr. Aydın Alatan
Electrical and Electronics Engineering Department, METU _______________

Assist. Prof. Dr. Elif Vural
Electrical and Electronics Engineering Department, METU _______________

Assoc. Prof. Dr. İlkay Ulusoy
Electrical and Electronics Engineering Department, METU _______________

Assist. Prof. Dr. Sevinç Figen Öktem
Electrical and Electronics Engineering Department, METU _______________

Assist. Prof. Dr. Zafer Arıcan
Electrical and Electronics Engineering Department,
Konya Food and Agriculture University _______________

Date: *06.09.2017*

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: GÖKHAN GÖKDOĞAN

Signature :

ABSTRACT

CLUSTERING OF MANIFOLD-MODELED DATA BASED ON TANGENT SPACE VARIATIONS

GÖKDOĞAN, GÖKHAN
M.S., Department of Electrical and Electronics Engineering
Supervisor : Assist. Prof. Dr. Elif Vural

September 2017, 60 pages

An important research topic of the recent years has been to understand and analyze data collections for clustering and classification applications. In many data analysis problems, the data sets at hand have an intrinsically low-dimensional structure and admit a manifold model. Most state-of-the-art clustering methods developed for data of non-linear and low-dimensional structure are based on local linearity assumptions. However, clustering algorithms based on locally linear representations can tolerate difficult sampling conditions only to some extent, and may fail for scarcely sampled data manifolds or at high-curvature regions. In this thesis, we consider a setting where each cluster is concentrated around a manifold and propose a manifold clustering algorithm that relies on the observation that the variation of the tangent space must be consistent along curves over the same data manifold. We argue that the non-linear geometric structure of manifold-modeled data sets can be better handled by taking into account the global data geometry via the change in the tangent space over the whole manifold. We first theoretically characterize some properties of manifolds of bounded curvature. We then use these observations to develop a geometry-based clustering approach. Finally, we evaluate the performance of the presented method with experiments on synthetic and real data sets and the results show that the proposed method outperforms the manifold clustering algorithms in comparison based on Euclidean distance, geodesic distance and sparse representations in some kind of data sets. Our study suggests that geometry-based dissimilarity measures can provide promising tools for the clustering of intrinsically low-dimensional data sets.

Keywords: Manifold Clustering, Dimensionality Reduction, Tangent Space, Unsupervised Classification

ÖZ

MANİFOLD MODELLİ DATANIN TANJANT UZAYI DEĞİŞİKLİKLERİNE DAYALI KÜMELENMESİ

GÖKDOĞAN, GÖKHAN
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Yrd. Doç. Dr. Elif Vural

Eylül 2017 , 60 sayfa

Son yılların önemli bir araştırma konusu, kümeleme ve sınıflandırma uygulamaları için veri kümelerini anlamak ve analiz etmektir. Birçok veri analizi probleminde, eldeki veri setleri özünde düşük boyutlu bir yapıya sahiptir ve bu yapı manifold modeli olarak kabul edilir. Doğrusal olmayan düşük boyutlu yapılar için geliştirilen en gelişkin kümeleme yöntemlerinden çoğu yerel doğrusallık varsayımlarına dayanır. Ancak, yerel doğrusal gösterimlere dayalı kümeleme algoritmaları, örnekleme koşullarının kötü olduğu durumları sadece bir dereceye kadar tolere edebilir ve az örneklenen manifoldlarda veya yüksek eğimli bölgelerde başarısız olabilir. Bu tezde, her bir kümenin bir manifold etrafında yoğunlaştığı ve tanjant uzayı değişiminin aynı manifolddaki eğriler boyunca tutarlı olması gerektiği gözlemine dayanan bir manifold kümeleme algoritması öneriyoruz. Manifold modelli veri kümelerinin doğrusal olmayan geometrik yapısının, bütün manifold üzerindeki tanjant alan değişimini gözleyerek elde ettiğimiz verinin küresel geometri bilgisini dikkate alarak daha iyi kavranabileceğini savunuyoruz. İlk olarak, sınırlı bir eğime sahip manifoldların bazı özelliklerini teorik olarak karakterize ettik. Daha sonra bu gözlemleri, geometri temelli bir kümeleme yaklaşımı geliştirmek için kullandık. Son olarak, önerdiğimiz yöntemin performansını, gerçek ve sentetik veri setleri ile yapılan deneyler ile değerlendirdik. Sonuçlar, bazı tür veri kümelerinde yöntemimizin Öklit uzaklık, jeodezik uzaklık ve seyrek gösterime dayalı yöntemlerden daha başarılı olduğunu gösterdi. Çalışmamız, geometri tabanlı benzerlik ölçütlerinin, temelinde düşük boyutlu bir yapıya sahip olan veri kümelerinin kümelenmesi için umut verici olduğunu önermektedir.

Anahtar Kelimeler: Manifold Kümeleme, Boyut Düşürme, Tanjant Uzayı, Denetimsiz Sınıflandırma

"Mutability is our tragedy, but it is also our hope."
Boethius

ACKNOWLEDGMENTS

Above all, I would like to thank my supervisor Assist. Prof. Elif Vural for her valuable guidance and tolerance. Her patience and self-sacrifice have made this process more instructive for me. Apart from all these, getting to know her has greatly increased my desire to become an academician.

I want to thank all the researchers, whose study is refereed in this work, for their contribution to accomplish my thesis.

I would like to thank my colleagues for helping me by alleviating my tasks, during writing my thesis.

A huge thanks to Bengisu Yalçınkaya for being the one who encourages me to follow an academic path in METU.

Last but not least, I thank my parents for being with me in all my successes and failures.

TABLE OF CONTENTS

ABSTRACT v

ÖZ vii

ACKNOWLEDGMENTS x

TABLE OF CONTENTS xi

LIST OF TABLES xiv

LIST OF FIGURES xv

LIST OF ABBREVIATIONS xvii

CHAPTERS

1 INTRODUCTION 1

1.1 Motivation 1

1.2 Thesis Outline 5

2 RELATED WORK 7

2.1 A Global Overview of Data Clustering Methods 7

2.2 Methods Taking Euclidean Distance As Similarity Measure . 9

2.2.1 K-Means Algorithm 9

2.2.2 Fuzzy C-Means Algorithm 10

2.3 Graph-Based Methods 12

2.3.1 Spectral Clustering 12

2.3.2 Spectral Clustering on Multiple Manifolds 14

2.4 Methods Based On Linear Representations 16

2.4.1 Sparse Subspace Clustering 17

2.4.2 Low Rank Representations For Subspace Clustering 18

3 PROPOSED MANIFOLD CLUSTERING ALGORITHM 21

3.1 Basic Concepts 21

3.1.1 Subspaces 21

3.1.2 Manifolds 22

3.1.3 Tangent Spaces 23

3.1.4 Principal Component Analysis 24

3.1.5 Geodesic Path and Geodesic Distance 27

3.1.6 Dijkstra's Algorithm 28

3.2 Theoretical Insights For Our Proposed Algorithm 29

3.3 Proposed Manifold Clustering Algorithm 32

3.3.1 Shortest path algorithm based on the variation of the tangent space 34

3.3.2 Merging the pre-clusters 35

3.3.3 Assignment of noisy points to clusters 37

4 EXPERIMENTAL RESULTS 41

4.1 Synthetic Data Set 42

4.2 The COIL-20 Data Set 43

4.3 The VidTIMIT Data Set 45

4.4 Analysis of Algorithm Parameters 46

4.4.1 Sensitivity to the Choice of "K" 46

4.4.2 Sensitivity to the Choice of "d" 47

4.4.3 Sensitivity to the Choice of "τ" 47

4.5 Conclusions Based on Experiments 49

5 CONCLUSION AND FUTURE WORK 51

5.1 Summary and Conclusions 51

5.2 Future Directions 52

REFERENCES 55

LIST OF TABLES

TABLES

Table 2.1 The types of information employed in basic clustering methods and our proposed algorithm. 9

Table 4.1 Clustering error and the NMI obtained on the VidTIMIT data set . . 46

Table 4.2 The variation of the clustering performance with the number of neighbors 47

Table 4.3 The variation of the clustering performance with the intrinsic dimension d 48

Table 4.4 The variation of the clustering performance with the threshold τ on the COIL-20 data set 48

Table 4.5 The variation of the clustering performance with the threshold t on the VidTIMIT data set 49

LIST OF FIGURES

FIGURES

Figure 1.1 Given face images of multiple people (top), the aim is to find images that belong to same person (bottom) [17]. 1

Figure 1.2 Clustering with the k-means algorithm (left), clustering with the spectral clustering method (right) [18]. 2

Figure 1.3 Given the images of an object from the COIL-20 data set obtained from different angles [47]. 3

Figure 1.4 Twenty objects from the COIL-20 database [47]. 4

Figure 1.5 Five objects whose dimension is reduced to 3 by PCA method. . . . 4

Figure 1.6 Given feature points on multiple moving objects tracked in consecutive frames of a video (top), the aim is to separate the feature points according to the moving objects (bottom) [17]. 5

Figure 2.1 Dataset which exhibits complex cluster shapes (left). In the embedded space given by two leading eigenvectors (right) [18]. 13

Figure 3.1 There are three 1-dimensional subspaces and they span the 3-dimensional space (left). There are two 2-dimensional subspaces and they span the 3-dimensional space (right). Adapted from [17],[61]. 22

Figure 3.2 Locally linearity of non-linear manifold. Adopted from [46]. 23

Figure 3.3 The tangent space $T_p\mathcal{M}$ at the point p on the manifold $\mathcal{M} \in \mathbb{R}^N$ [49]. 24

Figure 3.4 The angles between estimated and actual tangent spaces at each point of a noisy two-dimensional data set in $\mathbb{R}^3$. The tangent spaces are estimated (a) from small fixed neighborhood; (b) from large fixed neighborhood. Adopted from [30]. 25

Figure 3.5 A random Gaussian distribution (left), the principal components of the data (right) [71]. 26

Figure 3.6 The data which lie in correlated basis vectors (left), the data which lie in uncorrelated basis vectors (right) [71]. 26

Figure 3.7 The blue solid line illustrates the geodesic distance and the red dashed line illustrates the Euclidean distance [27]. 28

Figure 3.8 (a) Variation of the tangent space between neighboring points on a single noisy manifold (b) First stage of the proposed method. Pre-clusters are formed according to the connected components based on the distance F (c) Second stage of the proposed method: Pre-clusters are merged until the desired number of clusters is obtained 35

Figure 4.1 4 concentric ellipsoids of dimension $d = 2$. 42

Figure 4.2 Clustering error and the NMI obtained on the synthetic data set . . 43

Figure 4.3 Images of three similar objects that are very close to each other according to Euclidean metric in ambient space. 44

Figure 4.4 Clustering error and the NMI obtained on the COIL-20 data set . . 45

Figure 4.5 Given the images of a person obtained from different angles. 45

LIST OF ABBREVIATIONS

FCM	Fuzzy C-Means
LRR	Low Rank Representation
NMI	Normalized Mutual Information
PCA	Principal Component Analysis
PGMC	Progressive Geometric Manifold Clustering
SMCE	Sparse Manifold Clustering and Embedding
SSC	Sparse Subspace Clustering
SC	Spectral Clustering
SMMC	Spectral Multi-Manifold Clustering

CHAPTER 1

INTRODUCTION

1.1 Motivation

A large amount of visual data is captured every day and the analysis of the content of these data is an important research area. With the progress of the technology, there has been a significant increase in the amount of data obtained and the resolution of these data. High resolution means that the dimension of the data is also high. Besides the difficulty of analyzing a large number of data, the high computational cost caused by the high dimension of these data also presents a new problem.

One of the important machine learning approaches is clustering. Classification without the knowledge of priorly assigned class labels is called clustering or unsupervised learning. All clustering methods need a restriction about data distribution to achieve reasonable results. An important problem in clustering is the determination of the similarity between data samples. With this similarity, some intuition about the distribution of the data can be developed and data collections can be separated into groups by suitably processing the information of similarity.

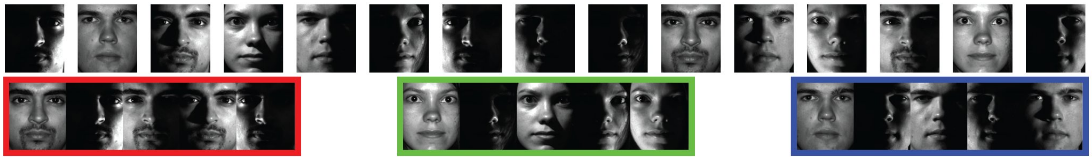


Figure 1.1: Given face images of multiple people (top), the aim is to find images that belong to same person (bottom) [17].

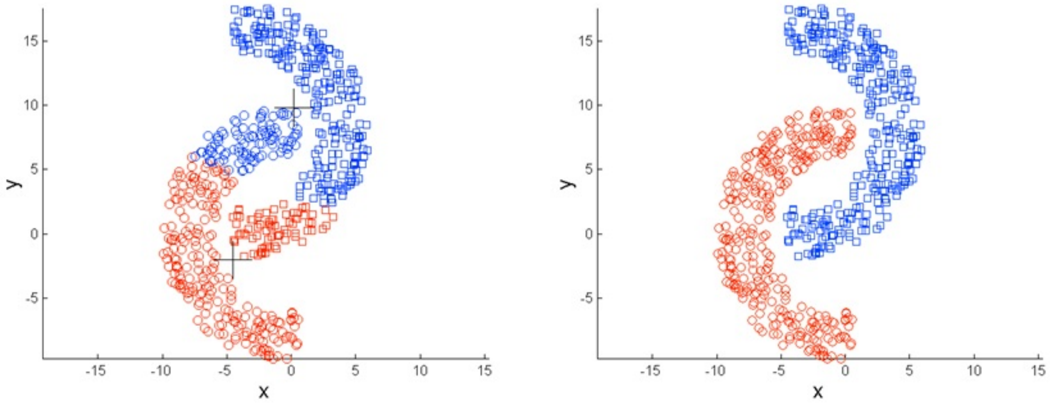


Figure 1.2: Clustering with the k-means algorithm (left), clustering with the spectral clustering method (right) [18].

One of the first developed methods was the k-means algorithm taking the Euclidean distance as a dissimilarity measure. However, the human perception and the Euclidean distance does not always match. This situation led to the development of algorithms such as spectral clustering. In Figure 1.2, the difference between the k-means algorithm and the spectral clustering algorithm can be observed. The k-means algorithm clusters data samples based on the Euclidean distance between them, while the spectral clustering algorithm takes into account the connections between the data samples over a graph. Then, the low-dimensional structure of the data and the connections between the data samples began to be addressed increasingly in clustering research.

It has been observed in the recent years that the data we deal with in many real world problems, such as feature trajectories of a rigidly moving object in a video [64],[41],[21],[69], face images of a person under varying illumination [[2],[28]], and multiple instances of a handwritten digit with different rotations, translations, and thicknesses [26], have a low-dimensional structure although they reside in a high dimensional ambient space. This discovery opens the way for improvements in the assessment of similarities between data points, as well as innovations for overcoming computational cost problems caused by the high-dimensionality of the data which is referred to as the “curse of dimensionality” [4].

We are particularly interested in low-dimensional data collections that are generated with respect to a small number of parameters that capture the main variations in data.

For example in Figure 1.1 , the same face images illuminated by different light intensities from different angles have an intrinsic dimension of 9 [17], which is the number of angle and brightness parameters required to model the data.

For another example in Figure 1.3, there are the images of an object from the COIL-20 data set obtained from different angles that change with a certain continuity. We select 5 objects from this data set, which is shown in Figure 1.4 and reduce their dimensions to 3 by the PCA method in order to have a visualization of the data set. The new coordinates of the data samples after dimensionality reduction are plotted in Figure 1.5, where the images of each object are plotted with a different color. In Figure 1.5, we observe that these object images actually have a low-dimensional structure.

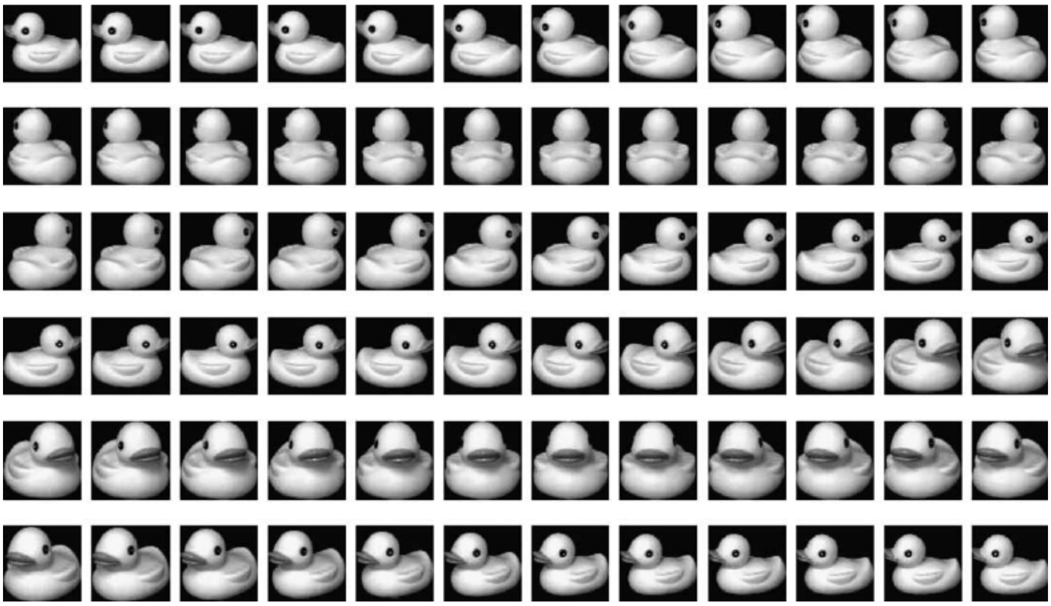


Figure 1.3: Given the images of an object from the COIL-20 data set obtained from different angles [47].

Also in video data, we can observe this low dimensional structure [66]. Figure 1.6 shows some points randomly selected in a video sequence. While some of these points will be on moving objects, others will be in stationary regions. Consider the x and y coordinates of each point along different frames in the video and suppose we align them as a vector to get the value of that point in the high dimensional space. Since the coordinates of the points on the same moving object change in a similar way, these points lie in the same low-dimensional structure in a high-dimensional space.

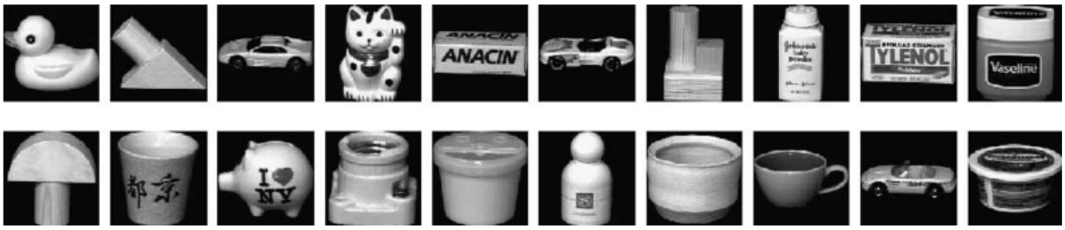


Figure 1.4: Twenty objects from the COIL-20 database [47].

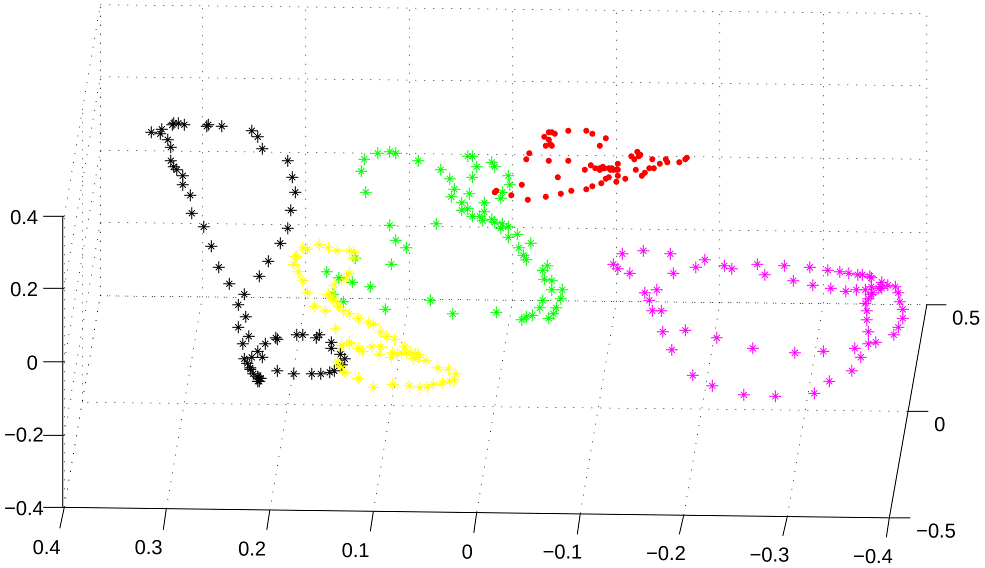


Figure 1.5: Five objects whose dimension is reduced to 3 by PCA method.

The low dimensional structures we learn from these data are made up of non-linear surfaces called “manifolds”. Many works have been devoted to the analysis of the intrinsic structures of data sets, which are called manifold learning methods. The analysis of the low-dimensional structures in data sets has also led to new approaches in clustering.

Most of the state-of-the-art clustering methods developed for data of non-linear and low-dimensional structure are based on local linearity assumptions. However, clustering algorithms based on locally linear representations can tolerate scarce sampling conditions only to some extent, and may fail for scarcely sampled data manifolds or at high-curvature regions.

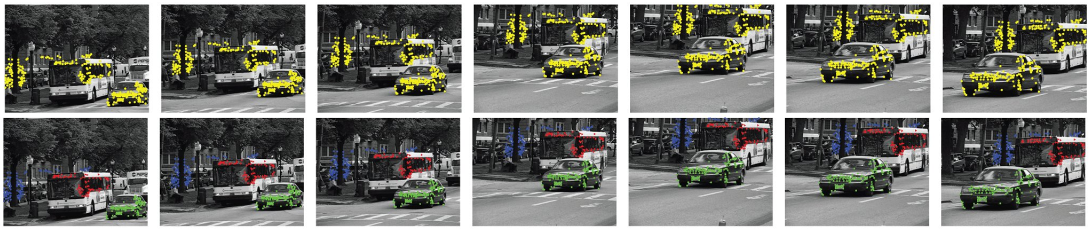


Figure 1.6: Given feature points on multiple moving objects tracked in consecutive frames of a video (top), the aim is to separate the feature points according to the moving objects (bottom) [17].

In this thesis, we consider a setting where each cluster is concentrated around a manifold and propose a manifold clustering algorithm that relies on the observation that the variation of the tangent space must be consistent along curves over the same data manifold. In order to achieve robustness against challenges due to noise, manifold intersections, and high curvature, we propose a progressive clustering approach. Observing the variation of the tangent space, we first detect the non-problematic manifold regions and form pre-clusters with the data samples belonging to such reliable regions. Next, these pre-clusters are merged together to form larger clusters with respect to constraints on both the distance and the tangent space variations. Finally, the samples identified as problematic are also assigned to the computed clusters to finalize the clustering.

1.2 Thesis Outline

The goal of this study is to develop a progressive manifold clustering approach in order to achieve robustness to challenging conditions caused by noise, intersecting manifolds, etc.

For this purpose, we first present a brief overview of the manifold clustering methods. In Chapter 2, we summarize some dissimilarity measures commonly used in clustering and the methods based on these dissimilarity measures.

In Chapter 3, we first overview some basic concepts to provide a better insight of our proposed algorithm. Then we mention some theoretical findings that motivate our work. Finally we describe our proposed algorithm.

Then we present experimental results to evaluate the performance of our method in Chapter 4. We compare our method to some classical and state-of-the-art methods.

Finally in Chapter 5, the thesis is concluded in the light of the experimental findings and the issues to be improved further are discussed.

CHAPTER 2

RELATED WORK

2.1 A Global Overview of Data Clustering Methods

In this chapter, we present a brief overview of the manifold clustering methods. These clustering methods differ from each other according to the dissimilarity measure they use.

Traditional clustering methods such as the k-means algorithm uses the Euclidean distance as a dissimilarity measure [25], [5], [31], so the clusters need to be sufficiently distant from each other in Euclidean sense in order to perform well. Therefore, this approach fails in data clouds that contain clusters having a non-linear structure and high variation.

During the past two decades, spectral clustering methods [48], [44], [52], [59], [75] have aimed to handle such data sets. They make use of an affinity matrix that contains the similarity between pairs of data samples, so that the similarity is essentially assessed based on the connections between the data samples over the data graph. In Figure 1.2, the difference between the k-means algorithm and the spectral clustering algorithm can be observed. However, the performance of spectral clustering highly depends on the construction of the data graph, and may suffer from erroneous connections between different clusters due to noise and insufficient sampling.

After discovering the low-dimensional inner structure of the high-dimensional data, the works in this area tended to analyze the manifold structure of the data. Many works have been devoted to the analysis of the intrinsic structures of data sets, which are called manifold learning methods [63], [53], [13], [3], [73],[57] ,[58]. Most manifold learning methods find a mapping that projects the data to a low-dimensional space, which preserves the intrinsic geometric structure of the data. The analysis of the low-dimensional structures in data sets has also led to new approaches in clustering [68].

Following the k-means and the spectral clustering algorithms, several multi-manifold clustering methods are proposed in [24], [1], [45], [22], [36] that handle data sets that consist of manifolds with different intrinsic dimensions and densities. Some other similar approaches are [10], [6], [9], which, however, assume that the data has a linear structure, and may not perform sufficiently well on data sets with a non-linear structure.

There are also some methods relying on non-linear representations. The k-manifold [61] method does multi-manifold clustering with an iterative approach similar to k-means based on the geodesic distance. However, due to erroneous connections, the geodesic distance as a dissimilarity measure may also fail for intersecting or critically close manifolds. The method in [72] uses the information of the local tangent space in addition to the Euclidean distance with the purpose of improving the performance of spectral clustering at intersecting manifold regions. The algorithm proposed in [29] is also based on computing the variation of the local tangent spaces. This however addresses the different problem of approximating a given manifold as a combination of flat local planes, rather than the manifold clustering problem.

Most of the recent manifold clustering methods attempting to cope with such challenges tend to exploit the self-expressiveness property of the data. Some of these methods are based on locally linear or sparse representations [16], [15], [60], [14], [51],[50], [17], while some others also include low rank assumptions [38], [40], [74], [39], [19]. Such methods yield quite favorable performance on data sets that admit nearly linear representations, e.g., when each cluster can be well approximated with a single subspace. However, the performances of these methods degrade when the data set at hand has a highly non-linear structure due to the high curvature of the underlying data manifolds or when the sampling of the manifold is not sufficiently dense.

Table 2.1: The types of information employed in basic clustering methods and our proposed algorithm.

	Euclidean dist.	Graph	Linear rep.	Geometric
K-means	✓	X	X	X
Spectral cl.	X	✓	X	X
SSC, LLR	X	✓	✓	X
Proposed	✓	✓	X	✓

A summary of the types of information employed in basic clustering methods and our proposed algorithm is given in Table 2.1. Some dissimilarity measures commonly used in clustering are elaborated below.

2.2 Methods Taking Euclidean Distance As Similarity Measure

We now overview some methods that assess the similarity between data samples based on the Euclidean distance. They vary according to the way they use the similarity information.

2.2.1 K-Means Algorithm

The k-means algorithm is a simple method. The input to the algorithm consists of points $X = \{x_1, \dots, x_n\}$ and the number of clusters k.

In the first step, the algorithm randomly chooses the cluster centers μ_j, for $j = 1, \dots, k$. In the second step, all points in the dataset $X = \{x_1, \dots, x_n\}$ are included in the clusters $S = \{x_1, \dots, x_k\}$ according to the closest cluster center μ_j, for $j = 1, \dots, k$, relative to the Euclidean distance.

$$\arg\min_S \sum_{j=1}^{k} \sum_{x \in S_j} ||x - \mu_j||^2$$

In the third step, by using this clustering information we have obtained, we update the vectors μ_j. We update each cluster center as the mean value of the points that belong to that cluster.

$$\mu_j = \frac{1}{n_j} \sum_{x \in S_j} x$$

where n_j is the number of points belonging to the j-th cluster. The second and third steps are iterated as long as the cluster assignments change for any point. Since the method is based on the Euclidean distance as the dissimilarity measure, the k-means algorithm performs well only if the clusters are sufficiently separated from each other. Therefore, such methods fail in data sets of an intricate geometric structure, with highly non-linear variations in a cluster. It is also a handicap that the performance of k-means is quite dependent on the initialization of the cluster centers.

In the following years some methods which are derived from the k-means algorithm were developed. One of these methods is the geodesic k-medoids algorithm. It extends the k-means algorithm to use the geodesic distance and finds the geodesic distances between all sample pairs via the Dijkstra's algorithm. The geodesic k-medoids algorithm uses the concept of "medoid" when initializing and updating the cluster centers. If the cluster center, which is based on the average of the distances to the cluster members, is selected from the cluster members, this point is called "medoids". Although using the geodesic distance as a dissimilarity measure provides an advantage, it may also fail in case of intersecting or critically close manifolds because of erroneous connections between sample pairs belong to different clusters.

2.2.2 Fuzzy C-Means Algorithm

The fuzzy c-means (FCM) algorithm allows points to belong to several clusters. According to the fuzzy logic principle, each point belongs to each one of the clusters with a membership value varying between 0 and 1. The sum of the all membership values of a point must be "1". The FCM algorithm aims to minimize the following objective function,

$$\arg\min_{C} \sum_{i=1}^{n} \sum_{j=1}^{k} w_{ij}^{m} \|x_i - c_j\|^2 \quad .$$

Given a set of vectors $X = \{x_1, \ldots, x_n\}$, w_{ij} indicates the membership value of point x_i to cluster j. The vector c_j, for $j = 1, \ldots, k$, is the center of the cluster j. The parameter m is any real number greater than or equal to 1 and indicates the degree of cluster fuzziness. A large m value causes a large fuzziness , therefore the clusters become blurred. The elements tend to belong to all clusters with the close memberships. On the contrary, if the parameter m is close to 1, the elements are assigned to one cluster and the memberships to other clusters are negligible. So the fuzzy c-means algorithm behaves like the k-means algorithm [65].

With this function, if the point is closer to the any cluster center, then the membership value of that cluster will be larger than the membership value of the other clusters. The algorithm is initialized by assigning membership values w_{ij} randomly. In the second step, the center vectors are calculated according to the following equation,

$$c_j = \frac{\sum_{i=1}^{n} w_{ij}^m x_i}{\sum_{i=1}^{n} w_{ij}^m} \quad .$$

In the third step, according to the calculated cluster centers, the membership values w_{ij} are updated using the following equation,

$$w_{ij} = \frac{1}{\sum_{l=1}^{k} \left(\frac{\|x_i - c_j\|}{\|x_i - c_l\|} \right)^{\frac{2}{m-1}}} \quad .$$

Then it goes back to the second step again and the second and third steps are repeated iteratively. The iterations are terminated when,

$$\max_{ij}(|w_{ij}^{(t+1)} - w_{ij}^{(t)}|) < \epsilon$$

where ϵ is a termination criterion and t is the iteration number. Like the k-means algorithm, the fuzzy c-means algorithm also fails in data which has complex geometric structure and the results depend on the choice of the initial membership values.

2.3 Graph-Based Methods

In graph-based clustering methods, the similarity between the pairs of data samples is established via the connections over the data graph. The differences between these methods are based on the similarity measure that they prefer when constructing the affinity matrix.

2.3.1 Spectral Clustering

The spectral clustering (SC) method approaches the data clustering as a graph partitioning problem. It makes no assumptions about the distribution of data clusters. There are different ways to construct a graph that indicates the similarities between data points. If all the pairs of data samples are connected to each other, it is called a complete graph. In another approach, each point is connected to points falling inside the ball of radius r centered at the point, where r is a real value that must be set to capture the local structure of the data. In the last method, each point is connected to its k-nearest neighbor points. The r-neighborhood graph and the k-nearest neighbor graph can be used together. Another important parameter in constructing a graph is the similarity measure to be used. One of the most commonly used similarity measures is the Gaussian type similarity and we used this type of similarity in our comparative tests. The Gaussian type similarity is defined by

$$w(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2})\right) \tag{2.1}$$

where $\|x_i - x_j\|$ is the Euclidean distance between x_i and x_j. Distances and affinities have an inverse relationship. If there is a low distance between two points, the similarity is high. After constructing the graph and the affinity matrix, we need to form the Laplacian matrix. The unnormalized graph Laplacian matrix , L, is defined as $L = D - W$. The entries of the Laplacian matrix are obtained as

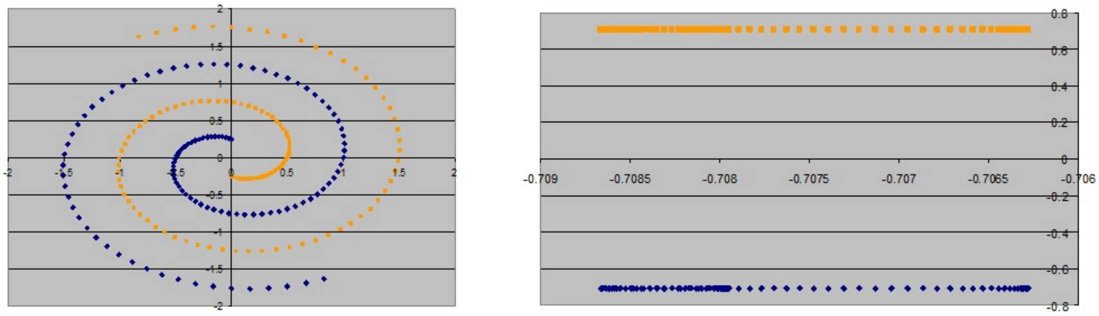


Figure 2.1: Dataset which exhibits complex cluster shapes (left). In the embedded space given by two leading eigenvectors (right) [18].

$$L_{ij} = \begin{cases} D_{ij} & if\ i = j \\ -W_{ij} & if\ i \neq j \end{cases}$$

where W is the affinity matrix whose off-diagonal entries are obtained according to (2.1). The diagonal elements are zero, while the off-diagonal elements represent the similarities of points with each other. D is the diagonal degree matrix with the diagonal elements representing the total affinity value established by the corresponding data point, i.e., the degree of the node. Considering that n is the number of data points, the entries of D are given by

$$D_{ii} = \sum_{j=1}^{n} w_{ij} \quad .$$

The unnormalized graph Laplacian matrix L satisfies the following properties [43], [44];

1$-$) L is symmetric and positive semi-definite.

2$-$) The smallest eigenvalue of L is 0, the corresponding eigenvector is a vector whose elements are all 1.

3$-$) L has n non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2, \ldots, \leq \lambda_n$.

The eigenvalues and the eigenvectors of the L matrix are found in order to complete the clustering. Then, the k eigenvectors $(u_1, \ldots, u_k)$ of L are chosen corresponding to the k smallest eigenvalues. Let $U \in \mathbb{R}^{nxk}$ contain the vectors $u_1, \ldots, u_k$ as columns such that each row corresponds to a data point $y_i \in \mathbb{R}^k$. Taking each y_i as the new coordinates of the data sample x_i, the samples are thus mapped to a new domain via the eigenvectors of the graph Laplacian. Lastly, the points y_i, for $i = 1, \ldots, n$, are clustered into k clusters with the k-means algorithm. See Algorithm 1 for a summary of spectral clustering.

Spectral clustering handles non-linear data sets under certain contidions, which can also be seen in Figure 2.1.

However, for this method to be successful, it is necessary to have sufficiently large gaps between the clusters. In the case of clusters intersecting or critically approaching each other, erroneous connections contaminate the graph structure and lead to failure.

Algorithm 1 Spectral Clustering [70]

```
Input: $n$ point dataset $X = \{x_1, \ldots, x_n\}$ to be segmented into $k$ clusters.
Construct weight matrix $W \in \mathbb{R}^{nxn}$
Compute the Laplacian matrix $L = D - W$
Compute the first $k$ eigenvectors $u_1, \ldots, u_k$ of $L$
Construct $U \in \mathbb{R}^{nxk}$ matrix using $u_1, \ldots, u_k$ as columns
Let $y_i \in \mathbb{R}^k$ be the vector corresponding to the $i$-th row of $U$
Cluster the points $(y_i)_{i=1,\ldots,n}$ into clusters $C_1, \ldots, C_k$ via the k-means algorithm
Assign point $x_i$ to cluster-$j$ if $y_i$ was assigned to $C_j$
```

2.3.2 Spectral Clustering on Multiple Manifolds

The spectral multi-manifold clustering (SMMC) method extends the spectral clustering method to construct a more robust method for intersecting manifolds. The basic idea is the following: The SC method is successful only when the affinity value between the points belonging to different clusters is relatively low. The classical SC method can not be successful at intersection areas or close manifolds as it uses the Euclidean distances between points as a dissimilarity measure. Because, in these regions, the points in different clusters have a high affinity value as the distance between them is small. Since this erroneous information will diffuse across different clusters, SC can not achieve a reasonable result [23]. Therefore, SMMC adapts SC for regions where points from different clusters are close. Even if the data is globally lying on or close to multiple non-linear manifolds, if the manifolds are sampled adequately, each data point and its neighbors locally construct an approximately linear structure [53], [56]. The local tangent space at any point provides the best linear approximation around this point and this approximation gives us information about the geometric structure of the manifold locally [76]. According to this information, at the intersection areas of different manifolds, the points belonging to the same manifold have similar local tangent spaces, while the points belonging to different manifolds have dissimilar tangent spaces. In addition to the Euclidean distance information used by the classical SC, SMMC also uses local geometric information from local tangent space similarities. Therefore, SMMC uses two functions while obtaining the affinity value between two points x_i and x_j. One of these functions (q_{ij}) uses the Euclidean distance between these points, while the other (p_{ij}) uses the local tangent space information of these points. Using the values coming from these two functions, the last affinity value is obtained as

$$w_{ij} = f(q_{ij}, p_{ij}) \ .$$

The tangent space at $(x_i)_{i=1,\ldots,N}$ is Θ_i. Using the local tangent space information between x_i and x_j, the parameter p_{ij} is defined as

$$p_{ij} = P(\Theta_i, \Theta_j) = \left(\prod_{l=1}^{d} \cos(\theta_l) \right)^{o} \quad .$$

Here $o \in \mathbb{N}^{+}$ is an adjustable parameter, d is the dimension of the manifolds and P is a function of the tangent spaces Θ_i and Θ_j related to the principal angles as above. $0 = \theta_1 \leq \theta_2, \ldots, \leq \theta_d \leq \pi/2$ are the principal angles between the two tangent spaces Θ_i and Θ_j, formulated as

$$\cos(\theta_1) = \max_{u_1 \in \Theta_i, \nu_1 \in \Theta_j} (u_1^T \nu_1) \quad subject\ to \quad \|u_1\| = \|\nu_1\| = 1$$

and for $l = 2, \ldots, d$;

$$\cos(\theta_l) = \max_{u_l \in \Theta_i, \nu_l \in \Theta_j} (u_l^T \nu_l) \quad subject\,to \quad \|u_l\| = \|\nu_l\| = 1$$

where $u_l{}^T u_i = 0,\ \nu_l{}^T \nu_i = 0,\ i = 1, \ldots, l-1.$

Using the Euclidean distance information between x_i and x_j, q_{ij} is defined as

$$q_{ij} = \begin{cases} 1 & if \quad x_i \in Knn(x_j) \quad or \quad x_j \in Knn(x_i), \\ 0 & otherwise. \end{cases}$$

Here $Knn(x)$ indicates the k-nearest neighbors of x. Finally, these two functions are multiplied to obtain the following affinity value,

$$w_{ij} = p_{ij} q_{ij} = \begin{cases} \left(\prod_{i=1}^{d} \cos(\theta_l) \right)^o & if \quad x_i \in Knn(x_j) \quad or \quad x_j \in Knn(x_i), \\ 0 & otherwise. \end{cases}$$

Although this method is especially developed for the problem of intersection areas, the tangent space approximations obtained in those regions are often not sufficient. Because the faulty neighborhood selection, which has already caused the problem, also negatively affects the local tangent space approximation. Therefore, even though this approach is more successful than spectral clustering, a small erroneous similarity information can lead to big errors on the graph, as the method is based on spectral clustering.

2.4 Methods Based On Linear Representations

Several clustering methods construct the affinity matrix based on the self-expressiveness property of the data, which argues that each data point can be effectively represented as a linear or affine combination of the other points.

2.4.1 Sparse Subspace Clustering

Sparse subspace clustering (SSC) is a sparsity-based approach that attempts to express the data as sparse linear combinations of the other points as the data representation [17]. The basic assumption here is that in a well-sampled data set, the points will have a locally linear structure.

Consider N data points $(y_i)_{i=1,\ldots,N}$ that lie in the union of n subspaces. The matrix containing all the data points can be shown as follows,

$$Y = [\, y_1 \; \ldots \; y_N \,] \, .$$

The SSC algorithm uses the following objective function ,

$$\min \|C\|_1 \;\; subject\, to \;\; Y = YC, \;\; diag(C) = 0 \tag{2.2}$$

where $C = [\, c_1 \; \ldots \; c_N \,] \in \mathbb{R}^{NxN}$ is the matrix whose i-th column corresponds to the sparse representation of y_i, c_i. The C matrix allows each point in the Y data set to be written as a linear combination of the other points. The first term of the objective function, ℓ_1-norm of $c_i \in \mathbb{R}^N$, is defined as

$$\|c_i\|_1 = \sum_{j=1}^{N} |c_{ij}| \;\; subject\, to \;\; c_{ii} = 0 \, .$$

Hence, the minimizing this expression allows the C matrix to use a small number of terms when expressing the data points as linear combinations of the others. There are some consequences of the forcing the C matrix to be sparse. In order to express a point as a linear combination of a small number of points, the selected points must be close to a plane passing through that point. Consequently, the other selected points are both forced to be close to this point and forced to be in the same subspace with this point. Because, as mentioned in section 2.3.2 , even if the data is globally lying on or close to multiple non-linear manifolds, if the manifolds are sampled adequately, each data point and its neighbors locally construct an approximately linear structure.

In order to obtain the similarity matrix, firstly the C matrix is normalized, then it is summed with its transpose and made symmetrical. Algorithm 2 summarizes the SSC algorithm.

Algorithm 2 Sparse Subspace Clustering [17]

```
Input: A set of points $(y_i)_{i=1,\dots,N}$ lying in a union of $n$ linear subspaces $(S_i)_{i=1,\dots,n}$ .
Solve the sparse optimization program (2.2).
Normalize the columns of $C$ as $c_i \leftarrow \frac{c_i}{\|c_i\|_\infty}$
Form a similarity graph with $N$ nodes representing the data points. Set the weights on the edges between the nodes by $W = |C| + |C|^T$
Apply spectral clustering 1 to the similarity graph.
Output: Segmentation of the data: $Y_1, Y_2, \dots, Y_n$.
```

SSC is a method based on linear representations, which aims to capture the global non-linear data structure via locally linear representations. Although this may be possible in favorable sampling conditions, the representations learned by SSC may fail to capture the global geometry under more challenging sampling conditions.

2.4.2 Low Rank Representations For Subspace Clustering

Another approach for obtaining the affinity matrix using the self-expressiveness property of the data is to use low rank representations (LRR). The LRR methods [38], [40],[74],[39],[19] try to find the lowest rank representation of all data jointly. In this way, each data point can be represented as a linear combination of certain basis vectors.

Let $X = \{x_1, \dots, x_n\}$ be a set of data points lying in the union of multiple subspaces. Each of these points can be represented as the linear combination of some vectors in a *"dictionary"*. The data X itself is used as the dictionary and thus the following objective function is obtained as

$$\min_{Z} rank(Z) \quad subject\,to \quad X = XZ \; .$$

This optimization problem is difficult to solve because of the discrete nature of the rank function and since its solution may not be unique. [8],[32],[7] suggest the following optimization problem which provides a good surrogate for the above optimization problem,

$$\min_{Z} \|Z\|_* \quad subject\,to \quad X = XZ \; .$$

Here $\|Z\|_*$ denotes the nuclear norm of Z, defined as the sum of all singular values of Z [20]. This approach argues that if the lowest rank representation matrix is searched, the data in the same cluster will be forced to be spanned with the same basis vectors.

However, it is hard for this method to obtain accurate representations when a sufficient number of data can not be observed and the amount of noise is high. The latent low rank representation (LatLRR) methods [40],[74] are developed to overcome these challenges, which try to obtain the low rank matrix not only for the columns but also for the rows. The LatLRR minimizes the following objective function,

$$\min_{Z,L} \|Z\|_* + \|L\|_* \quad subject\,to \quad X = XZ + LX \; .$$

Thus, the loss of information due to noise is solved by approaching the matrix from the other viewpoint, *"the points of row view"*. Minimizing the rank of the L matrix allows the data coordinates to be written using a few selected coordinates. Therefore, we can think of these selected coordinates as the learned features. In addition, thanks to this point of view, the data cloud, which was previously examined only as ambient space (column vectors), is now considered in a feature space (row vectors), so that we can learn new information about the geometric structure of data cloud from the information that the feature space provides. However, in these low-rank approaches, as in sparsity-based approaches, the ability to grasp the geometric structure of the manifold globally is not achieved. This is because in all these approaches the global geometric information of data is tried to be obtained only from the generated graphs.

CHAPTER 3

PROPOSED MANIFOLD CLUSTERING ALGORITHM

In this chapter, we will first overview some basic concepts in order to provide a better insight of our proposed algorithm. Then we will mention some theoretical findings that motivate our work. Finally we will describe our proposed algorithm.

3.1 Basic Concepts

Some basic concepts will be discussed in this section to provide a better insight of our proposed algorithm, such as, subspaces, non-linear manifolds, the local linearity on manifolds, tangent spaces and their estimation, the principal component analysis, the difference between the geodesic and the Euclidean distances and the Dijkstra's algorithm.

3.1.1 Subspaces

A subspace is a vector space that is a subset of another higher-dimensional vector space. Therefore, a subspace W must satisfy the following conditions [33];

$1-)$ W contains the zero vector, 0.
$2-)$ For each u and v which are elements of W, $u + v$ is an element of W. It is said that W is closed under vector addition.
$3-)$ For each u which is an element of W and each scaler c, cu is an element of W. It is said that W is closed under scalar multiplication.

The above conditions tell us that, if the points of a cluster lie on a subspace and are sampled sufficiently, these points can be expressed in terms of each other, as shown in Figure 3.1. As mentioned in Section 2.1, there are some earliest multi-manifold methods [10],[6],[9], which assume that the data has a linear structure like in Figure 3.1. Since these methods are based on linearity, they fail when the clusters have non-linear manifold distributions. In next section, we will discuss these manifold structures.

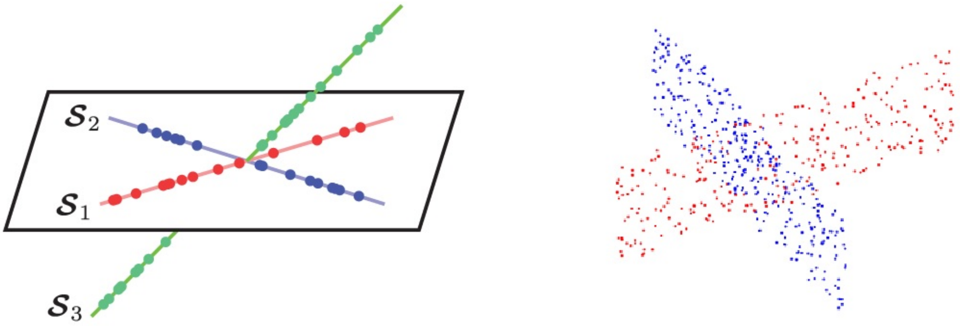


Figure 3.1: There are three 1-dimensional subspaces and they span the 3-dimensional space (left). There are two 2-dimensional subspaces and they span the 3-dimensional space (right). Adapted from [17],[61].

3.1.2 Manifolds

A manifold is a topological space M which has the following property [62]: *"If $x \in M$, then there is some neighbourhood U of x and some integer $n \geq 0$ such that U is homeomorphic to $\mathbb{R}^n$."*

In the simplest terms, a manifold is a topological space that locally looks like some Euclidean space $\mathbb{R}^n$ near each point, and on which one can do calculus [35]. A topological space is assumed to be a set that has adequate structure to meaningfully describe continuous functions on it [54].

In order to illustrate this concept, consider the ancient belief that the Earth was flat. This misbelief results from the fact that the Earth indeed looks flat on the small scales that we see. In general, any object that is nearly "flat" on small scales is a manifold, just as in the case of the "Earth".

We can summarize as follows, if all points in a set can be represented as linear combinations of each other, the points in this set are distributed on a subspace. If these points have linear characteristics only locally, which can also be observed in Figure 3.2, these points lie in a non-linear manifold. The tangent space of a point provides the best linear approximation of the manifold around that point. In next section, we will discuss these tangent spaces.

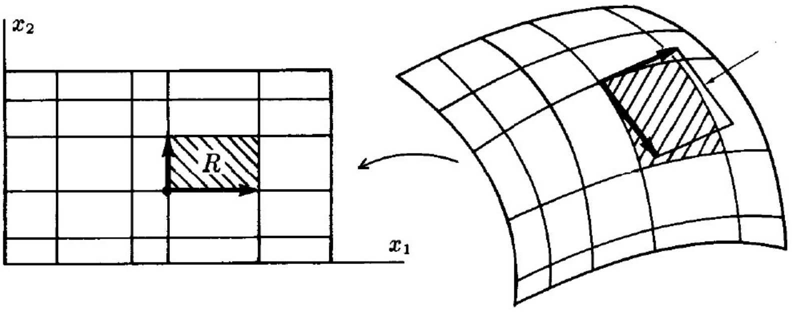


Figure 3.2: Locally linearity of non-linear manifold. Adopted from [46].

3.1.3 Tangent Spaces

In differential geometry, each point x of a differentiable manifold has a tangent space. This space is a real vector space that contains the possible directions in which one can tangentially pass through x. So, the tangent vectors, which pass through x, are the elements of the tangent space at x. The dimension of the tangent space at each point of this differentiable manifold is the same as the dimension of the manifold.

In Figure 3.3, The tangent space $T_p\mathcal{M}$ at the point p on the manifold $\mathcal{M} \in \mathbb{R}^N$ is illustrated. Tangent spaces are critical to our proposed algorithm. In manifold learning, another important issue is the accurate estimation of tangent spaces. In our proposed method, we estimated the tangent spaces using the principal component analysis (PCA), as mentioned in [67],[30]. To find the tangent space of a point, firstly we obtain a set of points which contains the point and its neighbors. Then, this set of points is analysed via the PCA method. The first critical decision here is how many neighbors we should choose. Choosing fewer points than necessary prevents us from having enough knowledge about that region of the manifold. However, choosing more

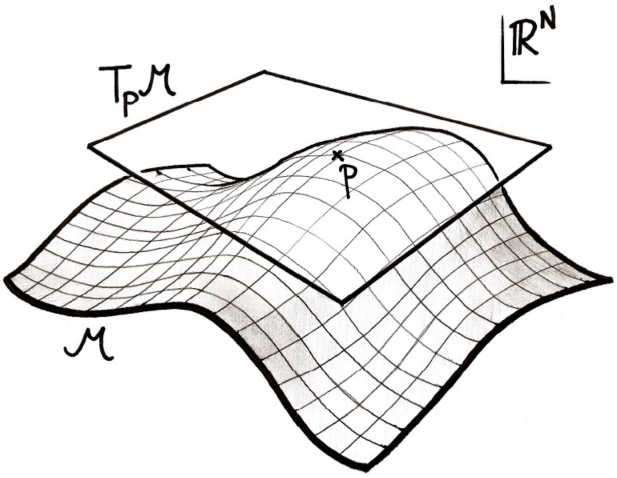


Figure 3.3: The tangent space $T_p\mathcal{M}$ at the point p on the manifold $\mathcal{M} \in \mathbb{R}^N$ [49].

points than necessary makes the effects of curvature prominent, and the local linearity fails. Figure 3.4 illustrates the effect of choosing different neighborhood sizes in order to estimate a tangent space. The effect of noise can be observed in the estimation of tangent spaces using small neighborhood along all surface. If the manifold curvature is sufficiently small, the tangent spaces are estimated more accurately from large neighborhood. Another important decision is how many principal components are chosen for projection. In our work, we aim to choose values close to the intrinsic dimensions of the manifolds. We obtain the closest values to the intrinsic dimensions of manifolds by trial and error. There are some significant works [37] on the estimation of the intrinsic dimensions of data sets. In next section, we will discuss the PCA method, which is critical for our algorithm.

3.1.4 Principal Component Analysis

In the simplest form of expression, the principal component analysis (PCA) is a basis vectors replacement method. For this purpose, the PCA method takes a set of data as an input. Then the covariance matrix of these samples is estimated. This covariance matrix has the correlation information between the coordinates of the data samples. Finally, the eigenvectors of the covariance matrix and their associated eigenvalues are analysed. The eigenvectors are sorted according to the magnitude of their corre-

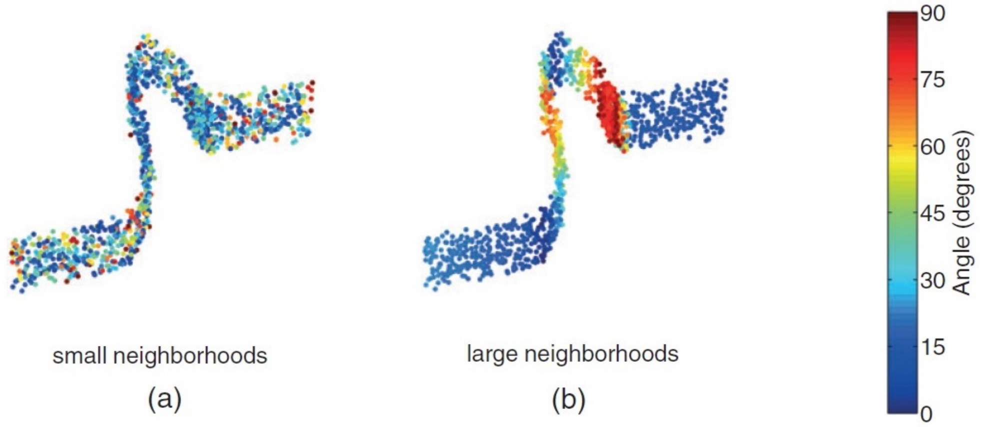


Figure 3.4: The angles between estimated and actual tangent spaces at each point of a noisy two-dimensional data set in $\mathbb{R}^3$. The tangent spaces are estimated (a) from small fixed neighborhood; (b) from large fixed neighborhood. Adopted from [30].

sponding eigenvalues. The eigenvectors of the covariance matrix define a new basis for the data set, which decorrelates the coordinates of data samples. The stages of the PCA method are summarized in Algorithm 3.

Algorithm 3 Principal Component Analysis

```
1: **Input:** $m$ point dataset $X = \{x_1, \ldots, x_m\} \in \mathbb{R}^n$.
2: Find the covariance matrix $C \in \mathbb{R}^{nxn}$ of the dataset.
3: Find the eigenvectors and the eigenvalues of $C$ using the eigendecomposition.
4: **Output:** The eigenvectors $V = \{v_1, \ldots, v_n\} \in \mathbb{R}^n$ as the new basis vectors
   sorted according to the magnitude of their eigenvalues $D = \{d_1, \ldots, d_n\} \in \mathbb{R}^n$.
```

What is the point of this basis vectors replacement? For example, we have a data set which lies in an n-dimensional space and we need to express the points in the data set with k dimensions, subject to $k < n$. The question we have to ask is, which directions should be chosen as the basis vectors to keep the maximum information about the data set? It is certain that there is a loss of information as long as $k < n$. However, this loss can be reduced to a minimum level by choosing the right directions.

During this process, the important point is to identify the directions along which the dataset is distributed with higher variances. To put it more clearly, the directions along which the points on the data set differ the most are the directions that capture the most information. The variances of the data along different directions are illustrated in

Figure 3.5. For instance, if all data points have the same coordinate in some direction, the information in that direction has no distinguishing feature and it is pointless to choose this direction as a basis vector.

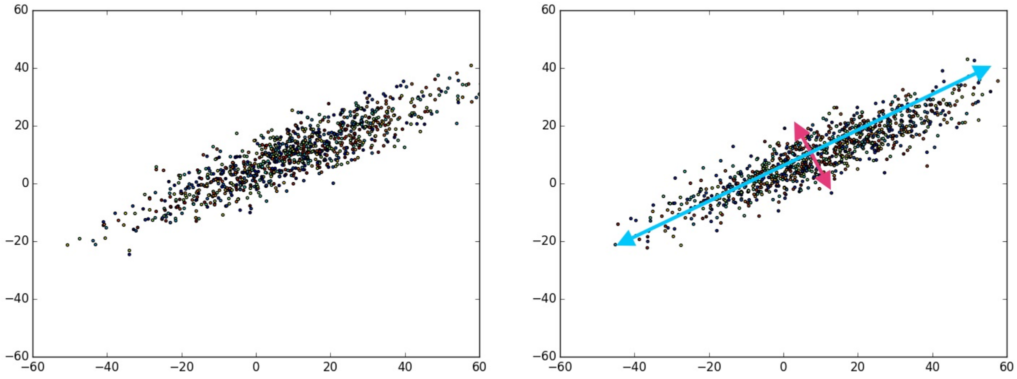


Figure 3.5: A random Gaussian distribution (left), the principal components of the data (right) [71].

Another important point is that the new basis vectors are uncorrelated. We can think of this as follows; if there is a common information along more than one basis vector, there is a redundancy of information here. So, as seen in the Figure 3.6, the new basis vectors do not carry common information.

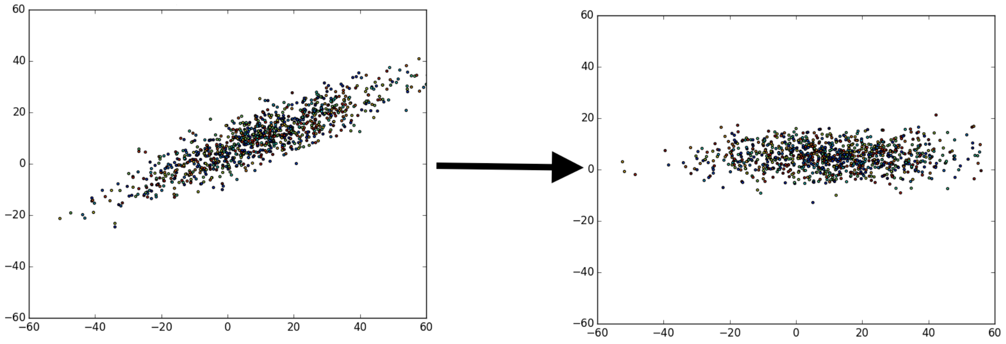


Figure 3.6: The data which lie in correlated basis vectors (left), the data which lie in uncorrelated basis vectors (right) [71].

The PCA method orders the basis vectors (eigenvectors) that span the space in which the data lie, according to the amounts of information which they carry (the magnitudes of their eigenvalues). When we need the directions with the highest variance for a specific purpose, we can make a choice among these ordered eigenvectors according to the magnitudes of their eigenvalues. Thus we can reduce the dimension of our data set by projecting it onto the matrix constructed with these selected eigenvectors.

An important parameter is the number we will reduce the dimension to. One of the preferred methods is to select as many eigenvalues as to retain 95% of the total amount of the energy which can be defined as the sum of all eigenvalues. Another method is to determine the point where the eigenvalues suddenly decrease and reduce the dimension according to this eigenvalue.

In our work, we use the PCA method to estimate the tangent space around a point. As we mentioned in section 3.1.3, we create a matrix consisting of a point and its neighbors. By analysing the covariance matrix of this matrix via PCA, we can think of the space spanned by the most dominant eigenvectors as the tangent space of that point, provided that the neighborhood is suitably chosen and the manifold curvature is sufficiently small.

3.1.5 Geodesic Path and Geodesic Distance

The geodesic path is the shortest path between any two points along a curved surface. The total distance travelled along this path is called the geodesic distance. As mentioned in Section 3.1.2, the surface of the Earth is a good example of a manifold. So the geodesic distance is needed in many real-world problems.

An example of the use of geodesic distance is to determine the shortest distance between any two cities for the flight path of an airplane. The maps are topologically fallacious, as the three-dimensional Earth is reduced to the two dimensions. In fact, they depict a curved surface, which we can define as a manifold, as a flat surface. As a result of this, when we observe the flight path of an airplane over the map, we see that the airplane follows a curved path instead of going straight. When we follow the curved path on the Earth, this line is actually a geodesic path which has the shortest distance between these two points.

This concept is also very important for our work. As mentioned in 1.1, the data sets we deal with are distributed over a manifold surface. Thus, while defining the relationships between the points of the data set, using a non-linearity based geodesic distance instead of a linearity based Euclidean distance allows us to obtain better results. The difference between these distance measurements can be observed in the Figure 3.7. There are a number of methods that can be used to determine these shortest paths and their distance. One of these methods is the Dijkstra's algorithm that we use in our work. We summarize this algorithm in the following section.

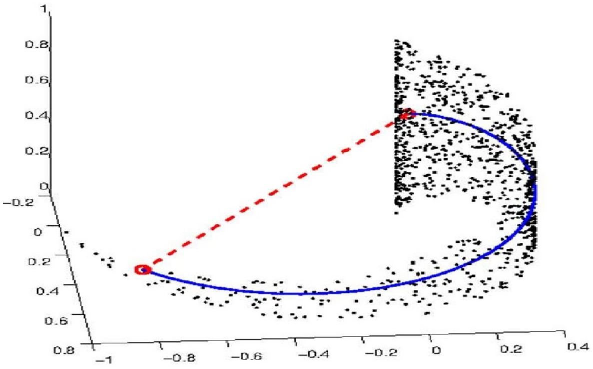


Figure 3.7: The blue solid line illustrates the geodesic distance and the red dashed line illustrates the Euclidean distance [27].

3.1.6 Dijkstra's Algorithm

The Dijkstra's algorithm is a method used to find the shortest paths and the distances over these paths between the nodes in a graph. The algorithm constructs the tree of shortest paths from the starting point to all other points in the graph. This is applied to all starting points in order to find the shortest paths and the distances between all pairs of the data samples. In the ideal case, the Dijkstra's algorithm connects the sample pairs in the same manifold by following the surface of this manifold, when applied to the data sets with a non-linear structure.

The graph G is defined over all data points by connecting the points i and j if they are closer than ϵ or if one of them is among the k-nearest neighbor points of the other. If i and j are connected, an Euclidean distance is defined between them, if they are not, the distance between them is assigned as infinity. Then, for each pair of the data samples, it is checked whether there is a shorter connection that can be established over another point in the data set. See Algorithm 4 for a summary of the Dijkstra's

Algorithm.

Algorithm 4 Dijkstra's Algorithm

```
Input:
    $X$: The collection of $N$ data points.
    $d(x_i, x_j)$: The Euclidean distance between $i$ and $j$; for $i, j = 1, \dots, N$.
Initialization: Initialize the distance function $F$:
if $x_j \in \mathcal{N}(x_i)$ or $d(x_i, x_j) < \epsilon$ then
    $F(x_i, x_j)$=$d(x_i, x_j)$
else
    $F(x_i, x_j)$=$\infty$
end if
Compute similarities via shortest paths:
for $k = 1, \dots, N$;  $i, j = 1, \dots, N$ do
    if $F(x_i, x_k) + F(x_k, x_j)$<$F(x_i, x_j)$ then
        $F(x_i, x_j)$=$F(x_i, x_k) + F(x_k, x_j)$
    end if
end for
Output:
    $F$: Distance function
```

3.2 Theoretical Insights For Our Proposed Algorithm

Before presenting our method, we first give a brief theoretical analysis of the variation of the tangent space over Riemannian manifolds, which will provide a basis for the proposed manifold clustering algorithm. In particular, our purpose here is to show that the change in the tangent space between two nearby points on the same manifold of bounded curvature is also bounded.

Let $\mathcal{M}$ be a Riemannian manifold of dimension d and let $T_p\mathcal{M}$ and $T_q\mathcal{M}$ denote the tangent spaces of the manifold respectively at the points $p, q \in \mathcal{M}$. We consider an immersion $\mathcal{M} \to \overline{\mathcal{M}}$ of $\mathcal{M}$ into a higher dimensional Riemannian manifold $\overline{\mathcal{M}}$ [12]. Let $U(t)$ be the parallel transport [12] of a tangent vector $u \in T_p\mathcal{M}$ along the arc-length parameterized geodesic curve $\gamma(t)$ joining p and q, such that $\gamma(0) = p$,

$\gamma(\ell) = q$ (ℓ being the geodesic distance), and $U(0) = u$. Let us denote by $\overline{u}$ and $\overline{U}(t)$ respectively the extensions of u and $U(t)$ on $\overline{\mathcal{M}}$.

We base our analysis on the following definition of the curvature $\kappa(p)$ of the manifold at point p:

$$\kappa(p) = \sup_{q\in\mathcal{M},\, u\in T_p\mathcal{M},\, t\in[0,\ell]} \left\| \frac{d\overline{U}(t)}{dt} \right\|.$$

Intuitively, the entity $\kappa(p)$ represents the maximum possible rate at which the parallel transport of a tangent vector at p may vary along all possible geodesics starting from the point p. Hence, $\kappa(p)$ provides a measure of curvature. If the overall curvature of the manifold $\mathcal{M}$ is bounded, then we can find a curvature upper bound $\mathcal{K}$ such that

$$\mathcal{K} = \sup_{p\in\mathcal{M}} \kappa(p).$$

Let $\{u_1, \dots u_d\}$ be an orthonormal basis for the tangent space $T_p\mathcal{M}$. We first propose the following lower bound on the inner product between the extension of the tangent vector $\overline{u}_i$ and its parallel transport $\overline{U}_i(\ell)$ at q:

Proposition 1. $\langle \overline{U}_i(\ell), \overline{u}_i \rangle \geq 1 - \mathcal{K}\ell$ *for all* $i = 1, \dots, d$.

Proof.

$$\begin{aligned}\langle \overline{U}_i(\ell), \overline{u}_i \rangle &= \langle \overline{u}_i + \int_0^\ell \frac{d\overline{U}_i(t)}{dt} dt,\ \overline{u}_i \rangle = 1 + \langle \int_0^\ell \frac{d\overline{U}_i(t)}{dt} dt,\ \overline{u}_i \rangle \\ &\geq 1 - \int_0^\ell |\langle \frac{d\overline{U}_i(t)}{dt},\ \overline{u}_i \rangle|\, dt \geq 1 - \int_0^\ell \left\| \frac{d\overline{U}_i(t)}{dt} \right\| dt \\ &\geq 1 - \mathcal{K}\ell.\end{aligned}$$

□

The result in Proposition 1 states that when the curvature of the manifold is bounded, for two nearby points p, q on the manifold having a small geodesic distance, a tangent vector at p is close to its parallel transport to q. This could be used to check whether two nearby points p and q are likely to belong to the same manifold or not. However, the parallel transport is not easy to estimate numerically when the manifold is not known and only data points sampled from the manifold are available. For this reason, in the sequel we extend this result to obtain a bound in terms of the angle between the tangent spaces at two manifold points, which is easy to estimate numerically with classical methods such as PCA.

$T_p\mathcal{M}$ and $T_q\mathcal{M}$ are subspaces of $\mathbb{R}^d$, so that the proximity between them can be characterized via the principal angles between them [42]. In practical data analysis problems, the manifold $\mathcal{M}$ resides in an ambient space $\mathbb{R}^n$, hence, we consider that the immersion of $\mathcal{M}$ is into the space $\overline{\mathcal{M}} = \mathbb{R}^n$. Let V_p and V_q be matrices whose columns are orthonormal bases for $T_p\mathcal{M}$ and $T_q\mathcal{M}$. Then the principal angles $\theta_1, \dots, \theta_d$ are such that $\cos\theta_i = \sigma_i$, where $\sigma_1 \geq \cdots \geq \sigma_d$ are the singular values of $V_q^T V_p$ [42]. Then a commonly used similarity measure between the subspaces is the sum of the cosines of the principal angles, which corresponds to $\mathrm{tr}(V_p^T V_q V_q^T V_p)$ [42]. In the following proposition, we present an upper bound on the dissimilarity between the tangent spaces at two points based on the principal angles.

Proposition 2. *Let the geodesic distance between the manifold points $p, q \in \mathcal{M}$ be $\ell \leq 1/\mathcal{K}$, where the curvature of the manifold is bounded by $\mathcal{K}$. Then we have*

$$1 - \mathrm{tr}(V_p^T V_q V_q^T V_p)/d \leq 2\mathcal{K}\ell.$$

Proof. Remembering that the principal angles do not depend on the choice of the orthonormal bases, and that the parallel transport preserves inner products, without loss of generality, we can pick V_p and V_q such that the i-th column of V_q is the parallel transport $\overline{U}_i(\ell)$ of the i-th column $\overline{u}_i$ of V_p. Let $\mathcal{P}_{T_q\mathcal{M}}(\overline{u}_i)$ denote the orthogonal projection of $\overline{u}_i$ onto the tangent space $T_q\mathcal{M}$. Noticing that the i-th column of $V_q V_q^T V_p$ gives the projection of the i-th column of V_p onto $T_q\mathcal{M}$, we have

$$\mathrm{tr}(V_p^T V_q V_q^T V_p) = \sum_{i=1}^{d} \langle \overline{u}_i, \mathcal{P}_{T_q\mathcal{M}}(\overline{u}_i) \rangle.$$

Here

$$\begin{aligned}\langle \overline{u}_i, \mathcal{P}_{T_q\mathcal{M}}(\overline{u}_i) \rangle &= \langle \overline{u}_i, \sum_{j=1}^{d} \langle \overline{u}_i, \overline{U}_j(\ell) \rangle \, \overline{U}_j(\ell) \, \rangle \\ &= \sum_{j=1}^{d} |\langle \overline{u}_i, \overline{U}_j(\ell) \rangle|^2 \geq |\langle \overline{u}_i, \overline{U}_i(\ell) \rangle|^2 \geq (1 - \mathcal{K}\ell)^2\end{aligned}$$

where the last inequality follows from Proposition 1. Using this in the previous equality, we get

$$\mathrm{tr}(V_p^T V_q V_q^T V_p) \geq d(1 - \mathcal{K}\ell)^2$$

which yields

$$1 - \mathrm{tr}(V_p^T V_q V_q^T V_p)/d \leq 1 - (1 - \mathcal{K}\ell)^2 = 2\mathcal{K}\ell - \mathcal{K}^2\ell^2 \leq 2\mathcal{K}\ell.$$

□

Proposition 2 can be interpreted as follows: When two nearby points p and q belong to the same manifold of bounded curvature, if the geodesic distance between them is sufficiently small, the entity $1 - \mathrm{tr}(V_p^T V_q V_q^T V_p)/d$ is also expected to be small. Hence, we can regard

$$C(p, q) = 1 - \mathrm{tr}(V_p^T V_q V_q^T V_p)/d \tag{3.1}$$

as a geometry-based dissimilarity measure or distance between two neighboring points. Assuming that the data manifolds to be clustered are of bounded curvature, $C(p, q)$ needs to be small when the nearby points p and q belong to the same manifold, while it is likely to be larger if p and q come from different manifolds. Hence, we use this measure to assess the dissimilarity between pairs of data samples in the manifold clustering algorithm proposed in the next section.

3.3 Proposed Manifold Clustering Algorithm

In this thesis, we propose a manifold clustering method that aims to estimate a clustering robust to the sampling conditions and the high-curvature geometry of data manifolds. We consider a setting where the samples from each cluster belong to a different manifold. Unlike the aforementioned clustering methods based on linear representations, our algorithm is based on tracing the variation of the tangent space over the data manifolds; hence, takes into account the geometry of the data globally rather than locally. In order to achieve robustness to difficult conditions caused by noise, intersecting manifolds, etc., we propose a progressive clustering scheme.

In the first stage of the algorithm, we find the nearest neighbors of each data sample, estimate the tangent space around each sample and compute shortest paths between pairs of samples. We define a distance measure based on the variation of the tangent space between neighboring points and obtain an initial pre-clustering such that each pre-cluster consists of samples connected by paths over which the tangent space varies smoothly.

This initial pre-clustering groups the data set into smooth and connected manifold regions. However, due to the noisy nature of the data or because of abruptly increasing manifold curvatures, several of these pre-clusters, which are disconnected from each other, may in fact belong to the same manifold. Hence, in the second stage of our method, we merge these pre-clusters by first reducing the dimension of the data by globally applying PCA to all data in order to reduce the effect of noise and normal components that are dominant in high curvature manifold regions. The pre-clusters are then combined with each other with respect to an affinity measure that represents the agreement of the tangent space between pairs of pre-clusters.

Finally, in the third and last stage of our method, we finalize the clustering by assigning the most problematic data samples to the computed clusters, again according to the agreement of the local tangent space computed at these points to those of the clusters.

The proposed three-stage clustering scheme that begins with the smoothest manifold regions and progressively moves on to more problematic regions can successfully handle difficult data geometries and sampling conditions.

We now present the proposed manifold clustering algorithm, which is motivated by the analysis in Section 3.2. Let $X = \{x_1, \ldots, x_N\} \subset \mathbb{R}^n$ be a set of N data samples belonging to M clusters, such that the samples from each cluster are drawn from a distribution concentrated around a manifold $\mathcal{M}_m \subset \mathbb{R}^n$ of intrinsic dimension d, for $m = 1, \ldots, M$. We then consider the problem of grouping the samples in X into M clusters $Y_1, \ldots Y_M$, each of which is concentrated around a different low-dimensional manifold, such that $X = \cup_m Y_m$.

Let $\mathcal{N}(x_i)$ denote the set of the K nearest neighbors of the sample x_i in X. Motivated by the findings of Section 3.2 and the dissimilarity measure in (3.1), ideally we would like to find M clusters such that the maximum change in the tangent space between neighboring manifold samples is minimal:

$$\begin{aligned} \min_{Y_1,\ldots,Y_M} \max\{C(x_i,x_j): \quad & x_j \in \mathcal{N}(x_i), \\ & x_i, x_j \in Y_m \text{ for some } m = 1,\ldots,M\}. \end{aligned} \tag{3.2}$$

However, there are some complications of the minimization of the above objective: Due to noise and the possible uneven sampling of the manifolds, the numerical estimation of the tangent space will be erroneous at some of the manifold samples in practice. Hence, the minimization of the objective in (3.2) will often not give a robust estimate of the clusters.

Due to these reasons, we propose the following constructive clustering solution that consists of three stages. We explain these stages in the following subsections:

3.3.1 Shortest path algorithm based on the variation of the tangent space

In the first stage of clustering, our purpose is to find a set of pre-clusters, each of which is constructed such that the tangent space varies smoothly among neighboring points in the cluster. For this reason we first define a distance F between neighboring samples such that

$$F(x_i,x_j) = C(x_i,x_j) \quad \text{for } x_j \in \mathcal{N}(x_i),$$

where the tangent space at a data sample can be numerically estimated with PCA. Note that in order to estimate the tangent spaces, the intrinsic dimensions of the manifolds should be known; nevertheless, even if the dimension is not known, it can be estimated with methods such as [37]. We then would like to extend this distance function $F(x_i,x_j)$ to the rest of the data samples, such that for any two samples x_i, x_j, the distance $F(x_i,x_j)$ gives the maximum change in the tangent space between any two neighboring samples along the best possible path connecting x_i and x_j. That is, if $P = (x_{k_1}, x_{k_2}, x_{k_3}, \ldots, x_{k_L})$ is a path of length L connecting x_i and x_j with $x_{k_m} \in \mathcal{N}(x_{k_{m-1}})$ and $x_i = x_{k_1}$, $x_j = x_{k_L}$, the distance function $F(x_i,x_j)$ is given by

$$F(x_i,x_j) = \min_{P=(x_{k_1},x_{k_2},\ldots,x_{k_L})} \max\{C(x_{k_{m-1}},x_{k_m})\}.$$

We obtain this distance function $F(x_i,x_j)$ by applying a modified version of the Dijsktra's algorithm as described in Algorithm 5. We then continue by inspecting the values of the distance function F for all sample pairs and obtain a pre-clustering of the data samples such that each pre-cluster consists of samples connected by paths and the distance F does not exceed a certain threshold value τ along these paths. In the ideal case, F must yield M connected components, where each component corresponds to one of the desired clusters. However, because of noise and uneven sampling of data, there may be some discontinuities in the manifolds as illustrated in Figure 3.8(a), which may lead to the presence of more than one connected component associated with the same manifold. Therefore, this procedure typically yields a pre-clustering with more pre-clusters than the desired number of clusters M. Finally, in order to prevent the noisy points from affecting the performance of clustering in the next stages of the algorithm, we check the number of samples in each pre-cluster and discard the pre-clusters whose cardinalities are smaller than a threshold. The points in these pre-clusters are identified as problematic "noisy" points, and these "noisy" pre-clusters are excluded from the merging procedure in the second stage of our method to achieve robustness. The formation of the pre-clusters in the first stage of the clustering is illustrated in Figure 3.8(b).

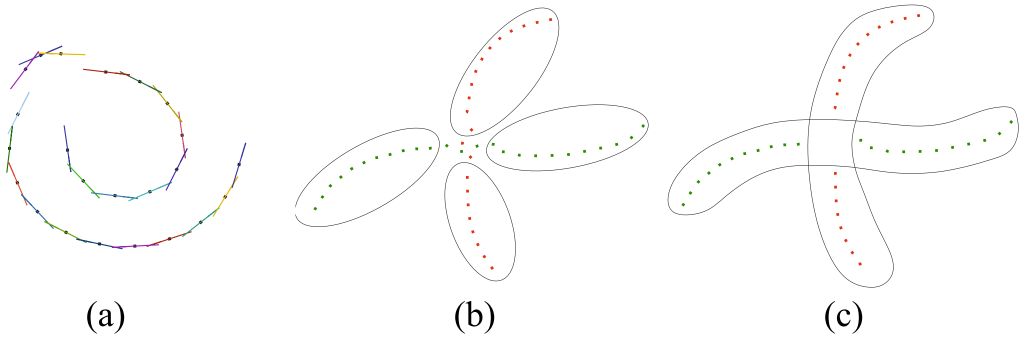


Figure 3.8: (a) Variation of the tangent space between neighboring points on a single noisy manifold (b) First stage of the proposed method. Pre-clusters are formed according to the connected components based on the distance F (c) Second stage of the proposed method: Pre-clusters are merged until the desired number of clusters is obtained

3.3.2 Merging the pre-clusters

In the second stage of our method, we merge the pre-clusters $Y_1, Y_2, \ldots, Y_{M+L}$ formed in the first stage towards obtaining the final clustering. In this step of merging, in order

Algorithm 5 Shortest path algorithm based on variation of the tangent space

```
Input:
    $X$: Collection of $N$ data points
    $M$: Number of clusters
Initialization: Initialize the distance function $F$ at nearest neighbors:
if $x_j \in \mathcal{N}(x_i)$ then
    $F(x_i, x_j)$=$C(x_i, x_j)$ according to (3.1)
else
    $F(x_i, x_j)$=$\infty$
end if
Compute similarities via shortest paths:
for $k = 1, \ldots, N$;  $i, j = 1, \ldots, N$ do
    if max($F(x_i, x_k)$,$F(x_k, x_j)$)<$F(x_i, x_j)$ then
        $F(x_i, x_j)$=max($F(x_i, x_k)$,$F(x_k, x_j)$)
    end if
end for
Output:
    $F$: Distance function based on the variation of tangent spaces
```

to reduce the effects of noise and the normal components of the manifolds dominant in high curvature manifold regions, we first map all samples in the data set X to a lower-dimensional domain by applying PCA globally to all data. Let $\overline{X} = \{\overline{x}_1, \ldots, \overline{x}_N\}$ be the low-dimensional embedding of the data samples obtained via PCA. We merge the pre-clusters in view of the objective (3.2) based on the following observation: If two pre-clusters belong to the same manifold, then for a pair of points from these pre-clusters that are close to each other, the tangent spaces at these points should not be too dissimilar.

In order to make use of this observation, we first estimate the geodesic distance $D_G(\overline{x}_i, \overline{x}_j)$ in the low-dimensional space between all pairs (x_i, x_j) of data samples, which we compute with the classical Dijkstra's shortest path algorithm [11] based on the Euclidean distance between nearest neighbors. The geodesic distances are thus computed such that $D_G(\overline{x}_i, \overline{x}_j) < \infty$ if there is a path connecting the points $\overline{x}_i$ and $\overline{x}_j$, and $D_G(\overline{x}_i, \overline{x}_j) = \infty$ otherwise.

We then define an affinity measure $A(Y_k, Y_m)$ for each pair of pre-clusters $Y_k, Y_m \in \{Y_1, \ldots, Y_{M+L}\}$ as

$$A(Y_k, Y_m) = \frac{1}{|S_{km}|} \sum_{(x_i, x_j) \in S_{km}} \exp\left(-H(x_i, x_j)\right) \tag{3.3}$$

where

$$S_{km} = \{(x_i, x_j) : \ x_i \in Y_k, \ x_j \in Y_m, D_G(\overline{x}_i, \overline{x}_j) < \infty\}$$

and $H(x_i, x_j)$ is the maximum change in the tangent space

$$H(x_i, x_j) = \max_{m=2,\ldots,L} \{C(x_{k_{m-1}}, x_{k_m})\}$$

along the shortest path $(\overline{x}_i = \overline{x}_{k_1}, \overline{x}_{k_2}, \ldots, \overline{x}_{k_L} = \overline{x}_j)$ between $\overline{x}_i$ and $\overline{x}_j$.

Thus, the affinity measure $A(Y_k, Y_m)$ represents the average similarity between the tangent spaces of all pairs of samples connected via a path between the clusters Y_k, Y_m. We merge the pre-clusters iteratively such that in each iteration we identify the two clusters having the highest affinity $A(Y_k, Y_m)$ and merge them into a new cluster. We continue the iterations until the number of clusters is reduced to the desired number of clusters M. The merging process is described in Algorithm 6 and illustrated in Figure 3.8(c).

3.3.3 Assignment of noisy points to clusters

In the third and last stage of our method, we finalize the clustering by assigning the "noisy" data samples discarded in the first stage of clustering, which were identified as problematic points. Let X_d denote the set of points discarded in the first stage. In order to assign each point in X_d to one of the clusters $Y_1, \ldots, Y_M$, we first extend the affinity measure defined in Section 3.3.2 to define the affinity $a(x_i, Y_m)$ between each point $x_i \in X_d$ and each cluster Y_m as

$$a(x_i, Y_m) = \frac{1}{|Y_m|} \sum_{x_j \in Y_m} \exp\left(-H(x_i, x_j)\right). \tag{3.4}$$

The affinity measure $a(x_i, Y_m)$ thus represents the average similarity between the tangent spaces at the sample x_i and at each one of the samples in the cluster Y_m.

Algorithm 6 Merging the pre-clusters

```
Input:
    $X$: Collection of $N$ data points
    $Y$= { $Y_1, \ldots, Y_{M+L}$ }: Pre-clusters
    $M$: Number of clusters
Initialization: Reduce the dimension of data via PCA
Find geodesic distances $D_G(\overline{x}_i, \overline{x}_j)$ with Dijkstra's algorithm
Merge pre-clusters:
while Number of clusters is greater than $M$ do
    for $k, m$ do
        Find $A(Y_k, Y_m)$ as defined in (3.3)
    end for
    Merge: $Y_{k'} = Y_{k'} \cup Y_{m'}$ for $k', m'$ maximizing $A(Y_k, Y_m)$
end while
Output:
    $Y = \{Y_1, \ldots, Y_M\}$: Merged clusters
```

Algorithm 7 Assignment of noisy points to clusters

```
Input:
    $X_d$: Excluded data points in the first stage
    $Y = \{Y_1, \ldots, Y_M\}$: Clusters
for $x_i \in \ X_d$ do
    for $m = 1, \ldots, M$ do
        Find $a(x_i, Y_m)$ as defined in (3.4)
    end for
end for
Clustering: Add $x_i$ to cluster $Y_{m'}$ maximizing $a(x_i, Y_m)$.
```

We then finalize the clustering by identifying the cluster that has the highest affinity to x_i and assigning x_i to that cluster, for each sample $x_i \in X_d$. This procedure is described in Algorithm 7. We call the proposed manifold clustering method consisting of these three stages as Progressive Geometric Manifold Clustering (PGMC).

CHAPTER 4

EXPERIMENTAL RESULTS

We now present experimental results to evaluate the performance of the proposed clustering method. We compare the proposed progressive geometric manifold clustering (PGMC) method to the k-means algorithm, the spectral clustering (SC) algorithm [48], the geodesic k-medoids algorithm, the sparse manifold clustering and embedding (SMCE) method [16] and the sparse subspace clustering (SSC) method [17]. We evaluate the performances of the clustering algorithms with respect to the two following common measures:

1. Clustering error in percentage: The one-to-one mapping giving the best match between the computed clusters and the true clusters is found. Then the clustering error is taken as the percentage of data points that are assigned to a wrong cluster.

2. Normalized Mutual Information (NMI): The NMI between the true clustering and the computed clustering is found, which is a widely used evaluation metric. The NMI between a clustering $Y = \{Y_1, \ldots, Y_M\}$ and the true clustering $Z = \{Z_1, \ldots, Z_M\}$ is given by

$$\frac{1}{\max(H(Y), H(Z))} \sum_{k,m} p(Y_k, Z_m) \log_2 \frac{p(Y_k, Z_m)}{p(Y_k)p(Z_m)}$$

where p denotes the probability that a randomly selected point would lie in a cluster or a pair of clusters and H is the entropy of the clustering defined as [34]

$$H(X) = -\sum_{m} p(X_m) \log_2 p(X_m) \ \ according\ to\ \ X_m \in X.$$

The NMI varies between 0 and 1, where higher values of the NMI closer to 1 indicate that the estimated clustering is closer to the true clustering.

Some experiments with synthetic and real datasets are presented below.

4.1 Synthetic Data Set

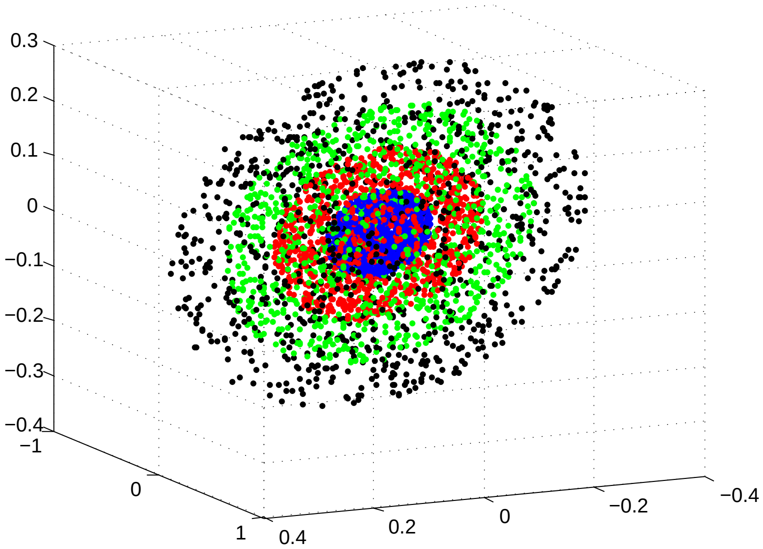
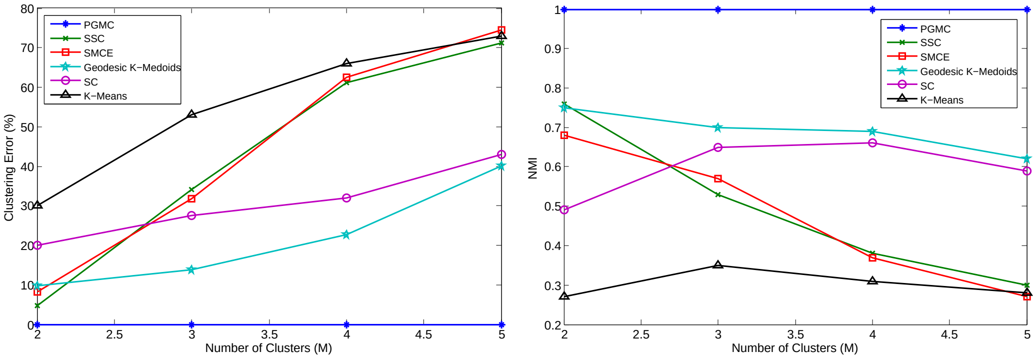


Figure 4.1: 4 concentric ellipsoids of dimension $d = 2$.

We first test the clustering methods on a synthetic data set, that consists of M concentric ellipsoids of dimension $d = 2$ residing in the ambient space $\mathbb{R}^{10}$, as shown in Figure 4.1. 200 samples are drawn from each ellipsoid so that the data set contains a total of $N = 200M$ samples. The samples from each ellipsoid are regarded as a different cluster. The data set is grouped into M clusters with the compared clustering methods. In Figures 4.2(a) and 4.2(b), the clustering error and the NMI obtained with the compared algorithms are plotted with respect to the number of clusters M. The proposed PGMC method outperforms the other clustering algorithms in comparison, as it yields a perfect clustering with 0% of clustering error and an NMI value of 1 on this synthetic data set. It can also be observed that the geodesic k-medoids algorithm and the spectral clustering algorithm yield a better performance than the other algorithms. Because there are enough space between different clusters in this data set, these graph-based clustering methods can make well connections between sample pairs belonging to same clusters.



(a) (b)

Figure 4.2: Clustering error and the NMI obtained on the synthetic data set

4.2 The COIL-20 Data Set

Next, we test the algorithms on the COIL-20 image data set [47]. The data set consists of the images of 20 different objects captured under varying viewpoints (Figure 1.4), with 72 images for each object (Figure 1.3). The images of each object are considered as a different cluster. We convert the images to greyscale and downsample each image to a resolution of 32 $\times$ 32 pixels.

As seen in Figure 1.5, it is observed that these object images actually have a low-dimensional structure. However, there are some challenges due to discontinuities along the points of some objects. Along these points, not only the angle of view, but also the scale of the object image may vary (Figure 4.3). This leads to the presence of several unconnected surfaces within the same cluster. One of the most significant challenges about the COIL-20 data set is that some of these problematic regions, which belong to different objects, are very close to each other with respect to the Euclidean distance in ambient space as shown in Figure 4.3 . Therefore, neither the methods using only the information coming from the Euclidean metric [25][48], nor the methods using the information coming from the tangent spaces in addition to Euclidean metric [72], nor methods based on linear representations [17],[16],[38],[40],[74],[39],[19] can give reasonable performance on such data sets of an intricate geometric structure, as reported in [72],[74] and our experimental results below. The proposed method we have elaborated above can successfully handle these problematic regions progressively.

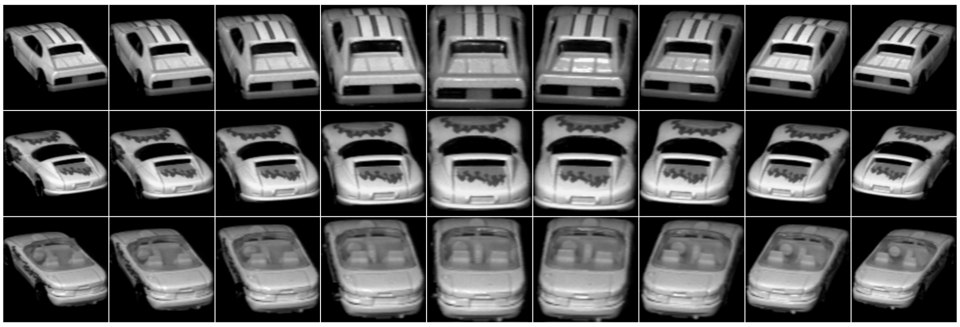
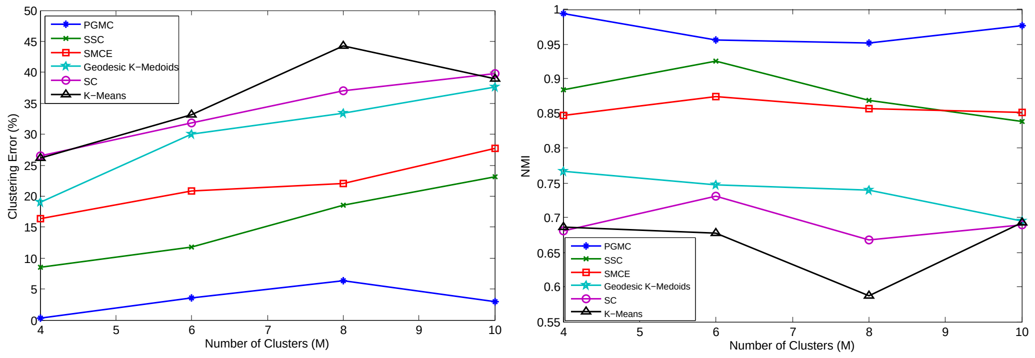


Figure 4.3: Images of three similar objects that are very close to each other according to Euclidean metric in ambient space.

Figures 4.4(a) and 4.4(b) show the clustering errors and the NMI values of the algorithms with respect to the number of clusters M. The results are obtained as the average of 6 trials where the M clusters are chosen randomly among the 20 object classes in each trial. The results show that the proposed PGMC method outperforms the other clustering algorithms in comparison, with respect to both the clustering error and the NMI measures. It can also be observed that the SSC method [17] and the SMCE method [16], which are based on sparse representations of data, yield a better performance than the traditional clustering algorithms. As the number of clusters increases, the clustering errors of all algorithms tend to increase, and their NMI values tend to decrease. The occasional drops in the clustering errors despite the increase in the number of clusters is due to the random choice of the objects in each repetition of the experiment and the presence of some particularly challenging objects in the data set, which may affect the clustering performance. Compared to the other algorithms, the proposed PGMC method seems to be less affected by the change in the number of clusters, as it can successfully make use of geometric priors related to the manifold model of data.

4.3 The VidTIMIT Data Set

Finally, we test the algorithms on the VidTIMIT image data set [55]. The data set consists of the images of 43 different people captured under varying viewpoints. The



(a) (b)

Figure 4.4: Clustering error and the NMI obtained on the COIL-20 data set

images of a subject from the VidTIMIT data set are given in Figure 4.5 . The images of each person are considered as a different cluster. We convert the images to greyscale and downsample each image to a resolution of 24×32 pixels from 384×512.

Table 4.1 show the clustering errors and the NMI values of the algorithms with respect to the number of clusters M. The results are obtained as the average of 5 trials where the M clusters are chosen randomly among the 43 object classes in each trial. The results show that the proposed PGMC method and the SSC method [17] outperforms the other clustering algorithms in comparison, with respect to both the clustering error and the NMI measures. It can also be observed that the recent SMCE method [16] yields a better performance than the traditional clustering algorithms.

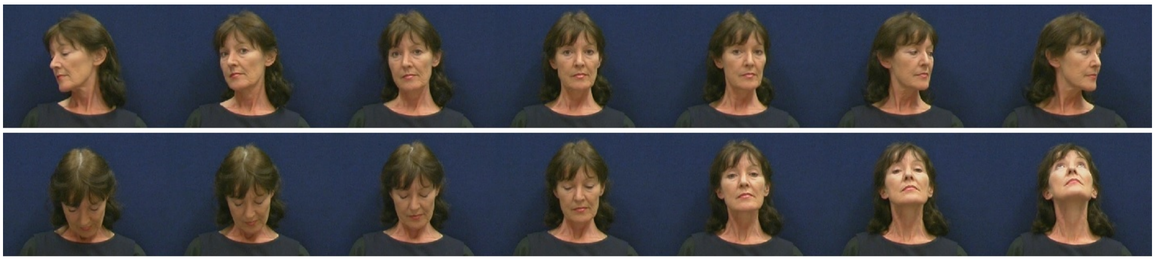


Figure 4.5: Given the images of a person obtained from different angles.

Table 4.1: Clustering error and the NMI obtained on the VidTIMIT data set

Measurement Methods	Clustering Error (%)				NMI			
Number of Clusters (M)	3	5	7	10	3	5	7	10
PGMC	0.000	0.000	0.000	0.000	1.000	1.000	1.000	1.000
SSC	0.000	0.000	0.000	0.000	1.000	1.000	1.000	1.000
SMCE	0.000	0.000	0.500	2.200	1.000	1.000	0.997	0.962
Geodesic K-Medoids	0.000	7.200	18.21	37.75	1.000	0.930	0.900	0.760
SC	8.930	22.50	33.00	46.50	0.910	0.840	0.760	0.670
K-Means	17.81	34.00	40.00	47.00	0.760	0.680	0.640	0.630

4.4 Analysis of Algorithm Parameters

Our proposed algorithm has some parameters that need to be set in order to work well. One of these is the number of neighbors K used in constructing the data graph and the estimation of the local tangent spaces. The other one is the estimated value d of the intrinsic dimension of the manifolds and the last parameter of our method is the threshold value τ that separates the sub-manifolds in the first stage of the algorithm. In this section, we study the effects of these parameters on the clustering performance.

4.4.1 Sensitivity to the Choice of "K"

We study the effect of the neighborhood size parameter on the clustering performance. We experiment on the COIL-20 data set and study the variation of the clustering accuracy with the number of neighbors K used in constructing the data graph and the estimation of the local tangent spaces. The results given in Table 4.2 show the clustering error in percentage and the NMI obtained at different K values for different numbers of clusters. The results show that the best performance is obtained at the rather small number of neighbors $K = 2$. This can be explained with the fact that, due to the one-dimensional rotational motion of the camera and the scale of the object image which sometimes varies, the intrinsic dimension of this data set is about two, hence, increasing of the number of neighbors introduces some error in the tangent space estimation. Nevertheless, the increase in the clustering error is not dramatic and the proposed method seems to tolerate non-optimal choices of the number of neighbors reasonably well. We refer the reader to [67] for further results on how the estimation of the tangent space is affected by parameters such as the intrinsic and ambient space dimensions, the neighborhood size, and the sampling density of data.

Table 4.2: The variation of the clustering performance with the number of neighbors

Cluster	Clustering error (%)			NMI		
Number	K=2	K=3	K=4	K=2	K=3	K=4
4	0,3	4,9	4,5	0,994	0,956	0,960
6	3,5	10,1	15,2	0,956	0,906	0,908
8	6,3	16,0	20,0	0,951	0,910	0,888
10	3,0	12,4	18,1	0,977	0,932	0,908

4.4.2 Sensitivity to the Choice of "d"

Secondly, we study the effect of the estimated intrinsic dimension value d of the manifold on the clustering performance. We experiment on the COIL-20 data set again and study the variation of the clustering accuracy with the estimated value d used in the estimation of the local tangent spaces. The results given in Table 4.3 show the clustering error in percentage and the NMI obtained at different d values for different numbers of clusters. The results show that the best performance is obtained at $d = 2$. As in the preceding experiment, this can be explained with the fact that the intrinsic dimension of the data set is 2; hence, the best estimates of the tangent space are given by the first two most significant principal components. It is observed that the clustering error increases as the choice of d diverges from its optimal value especially when the number of clusters gets larger. Nevertheless, the increase in the clustering error does not seem to be dramatic.

4.4.3 Sensitivity to the Choice of "τ"

Lastly, we study the effect of the threshold value τ on the clustering performance. We experiment on the COIL-20 data set and the VidTIMIT data set. We study the variation of the clustering accuracy with the threshold value τ applied to the distance function F that separates the sub-manifolds in the first stage of the algorithm

Table 4.3: The variation of the clustering performance with the intrinsic dimension d

Cluster	Clustering error (%)				NMI			
Number	d=1	d=2	d=3	d=4	d=1	d=2	d=3	d=4
4	0,3	0,3	3,8	9,6	0,973	0,994	0,948	0,927
6	10,0	3,5	14,7	20,0	0,907	0,956	0,842	0,768
8	11.2	6,3	18,4	22,8	0,893	0,951	0,791	0,746
10	9,6	3,0	15,2	18,3	0,948	0,977	0,850	0,793

Table 4.4: The variation of the clustering performance with the threshold τ on the COIL-20 data set

Cluster	Clustering error (%)					NMI				
Number	τ=0,5	τ=0,6	τ=0,7	τ=0,8	τ=0,9	τ=0,5	τ=0,6	τ=0,7	τ=0,8	τ=0,9
4	0,3	0,3	0,3	10,6	13,5	0,994	0,994	0,994	0,885	0,873
6	3,5	3,5	3,5	15,0	17,7	0,956	0,956	0,956	0,862	0,858

as discussed in Section 3.3.1 . The results given in Table 4.4 and Table 4.5 show the clustering error in percentage and the NMI obtained at different τ values for different numbers of clusters.

The threshold value τ can vary between 0 and 1, because the distance F varies between 0 and 1. It has not been possible to obtain a pre-clustering at very low threshold values as the choice of very low threshold values causes even good connections between points to be ignored. Therefore, we have not been able to test the threshold values below $\tau = 0.5$. The results given in Table 4.4 and Table 4.5 show that the values in which the best performance is obtained are spread over a wide range. The best choice of τ is seen to lie within the rather intermediate range of values $[0.5, 0.7]$. Even at very high threshold values, the increase in the clustering error is rather limited. The exclusion of the pre-clusters obtained in the first step, if their cardinalities are smaller than a certain value, contributes to this result. This is because the pre-clusters that are connected erroneously due to the high threshold values usually consist of the points identified as problematic.

Table 4.5: The variation of the clustering performance with the threshold t on the VidTIMIT data set

Cluster	Clustering error (%)					NMI				
Number	τ=0,5	τ=0,6	τ=0,7	τ=0,8	τ=0,9	τ=0,5	τ=0,6	τ=0,7	τ=0,8	τ=0,9
3	0	0	0	0	1,2	1	1	1	1	0,986
5	0	0	0	5,8	7,3	1	1	1	0,966	0,957

4.5 Conclusions Based on Experiments

The results of the first two experiments (synthetic data set and the COIL-20 data set) show that the proposed geometry-based manifold clustering algorithm outperforms reference clustering solutions relying on the assessment of dissimilarity via the Euclidean distance, the geodesic distance or sparse linear representations. In the VidTIMIT data set, the methods based on linear representations such as SSC and SMCE also get successful results. Because, unlike the COIL-20 data set, the rotation of objects is restricted in the VidTIMIT data set. This causes the data set to have a more linear structure. So SSC and SMCE give reasonable performance in the VidTIMIT data set.

In the synthetic data set, when the radial magnitudes of the concentric ellipsoids are chosen so as to prevent the ellipsoids from being too far apart relative to one another, the geodesic k-medoids algorithm and the spectral clustering algorithm establish connections between data pairs belonging to different clusters. These connections cause some members belonging to different clusters to have a relatively high affinity value. In addition, SSC and SMCE yield unsatisfactory results due to the fact that points from different clusters can be located in the same subspace. Because of the fact that the data set is non-convex, the k-means algorithm does not perform well as expected.

Next, in the COIL-20 data set, in the regions where some manifolds are close (for example the three very similar objects 3, 6 and 19 in Figure 4.3), the spectral clustering method determines high similarity values between the points belonging to different clusters as distances between them are small. Since this erroneous information is diffused across different clusters, SC can not achieve a reasonable result for the COIL-20 data set. Inasmuch as the k-means algorithm also uses the Euclidean distance as a dissimilarity measure, this approach fails in the COIL-20 data set, which contains clusters having a non-linear structure and high variation. Because of this highly non-linear structure due to the high curvature of the underlying data manifolds, it does not admit well linear representations, e.g., in contrast to data sets where each cluster can be well approximated with a single subspace. For this reason, SSC and SMCE do not give a reasonable performance. The geodesic k-medoids algorithm also can not achieve a reasonable result for the COIL-20 data set because of erroneous geodesic connections between sample pairs belonging to different clusters. These connections cause some members belonging to different clusters to have a relatively high affinity value.

Finally, the analysis of algorithm parameters shows that the correct choice of these parameters is important for our proposed algorithm. All of them have an optimal value in order to perform well and these optimal values are consistent with the intrinsic dimension of the manifold and the geometric structure of the data set. Therefore, the performance degradation due to the deviation from these values is a result of the geometric structure of the data.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Summary and Conclusions

We have presented a clustering method developed for data sets of intrinsic low dimension. We have considered a setting where the samples of each cluster are distributed around a low-dimensional manifold, and proposed to cluster the data by making use of some geometric priors respected by manifolds of bounded curvature. We have first proposed a theoretical analysis of the variation of the tangent space on Riemannian manifolds of bounded curvature. Then, motivated by these theoretical findings, we have presented a manifold clustering algorithm that is based on the observation that the change in the tangent space must be limited between two nearby data points sampled from the same manifold. We have proposed a three-stage progressive clustering solution that intends to identify problematic data samples and exclude them from the initial stages of the clustering to in order to achieve better robustness against noise and inconvenient sampling conditions.

What led us to this progressive clustering solution is the observation that the change in the tangent space over the same manifold can be inconsistent in some cases such as noise, uneven sampling of data, intersecting manifolds. This caused us to develop a greedy algorithm. Thus, we achieve robustness against challenges due to discontinuities in the manifolds, noise, manifold intersections, and high curvature.

An important observation in this thesis is that different approaches may be favorable for the clustering of different data sets. As can be seen in our synthetic data set experiment, graph-based methods such as the geodesic k-medoids algorithm and the spectral clustering method achieve reasonable results in data clouds that contain clusters having a non-linear structure. However, these clusters need to be sufficiently distant from each other. In linear structured data sets such as the VidTIMIT data set, the methods based on linear representations such as SSC and SMCE get successful results. Our proposed algorithm also gets successful results in these two kinds of data sets. In addition, our method is more favorable in the data sets which have a highly non-linear structure due to the high curvature of the underlying data manifolds such as the COIL-20 data set.

5.2 Future Directions

The data sets used in this thesis have a well-defined low-dimensional structure such that each cluster is highly concentrated around a manifold. The aim of this thesis has been to show that one may efficiently employ the information of the geometric structure of data in order to improve the clustering performance for such low-dimensional data collections. However, in several real problems, the data at hand might not be so well-structured. One important question is then how the ideas proposed in this thesis can be extended to data collections captured under rather uncontrolled environments, with irregular background, large within-class variability, etc. The extraction of the relevant features from such data sets of large variability to permit the analysis of its essential characteristics is an active research problem. In the future, new feature extraction methods may be developed for better and more efficient extraction of the meaningful low-dimensional structures constituting the essence of data. The fusion of such learning algorithms with some of the ideas proposed in this thesis may lead to exciting new directions.

We have worked on a greedy algorithm in this thesis. A possible handicap of the proposed method is that the erroneous estimates made in one of the stages may be propagated to other stages and may affect negatively the overall result. As future work, again based on geometric insights (such as tangent space), it may be possible to achieve better results by expressing the clustering problem with a more global optimization problem.

Another possible work is to develop a method based on both tracing the variation of the tangent space and making use of linear representations. Thus, erroneous affinities due to incorrect tangent space estimations can be improved using linear approaches and reasonable results can be obtained by observing the variation of the tangent space in the data sets with a non-linear structure, where linearity-based approaches fail. In addition, this combination may also allow clustering by solving an optimization problem over a single objective function.

REFERENCES

[1] D. Barbará and P. Chen. Using the fractal dimension to cluster datasets. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, Boston, MA, USA, August 20-23, 2000*, pages 260–264, 2000.

[2] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(2):218–233, 2003.

[3] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 585–591, 2001.

[4] R. Bellman. *Dynamic Programming*. Dover Publications, 1957.

[5] J. Bezdek, R. Ehrlich, and W. Full. FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3):191–203, 1984.

[6] P. S. Bradley and O. L. Mangasarian. k-plane clustering. *J. Global Optimization*, 16(1):23–32, 2000.

[7] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *CoRR*, abs/0912.3599, 2009.

[8] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[9] R. Cappelli, D. Maio, and D. Maltoni. Multispace KL for pattern representation and classification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(9):977–996, 2001.

[10] G. Chen and G. Lerman. Spectral curvature clustering (SCC). *International Journal of Computer Vision*, 81(3):317–330, 2009.

[11] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[12] M. do Carmo. *Riemannian Geometry*. Mathematics (Boston, Mass.). Birkhäuser, 1992.

[13] D. L. Donoho and C. Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. In *Proc. Natl. Acad. Sci. USA*, volume 100, pages 5591–5596, May 2003.

[14] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 2790–2797, 2009.

[15] E. Elhamifar and R. Vidal. Clustering disjoint subspaces via sparse representation. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010, 14-19 March 2010, Sheraton Dallas Hotel, Dallas, Texas, USA*, pages 1926–1929, 2010.

[16] E. Elhamifar and R. Vidal. Sparse manifold clustering and embedding. In *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain.*, pages 55–63, 2011.

[17] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *CoRR*, abs/1203.1005, 2012.

[18] J. Fan. Spectral Clustering. [Online] Available: http://webpages.uncc.edu/jfan/itcs6157.html.

[19] P. Favaro, R. Vidal, and A. Ravichandran. A closed form solution to robust subspace estimation and clustering. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 1801–1807, 2011.

[20] M. Fazel. *Matrix rank minimization with applications*. Mathematics (Boston, Mass.). PhD Thesis, 2002.

[21] Z. Fu, W. Hu, and T. Tan. Similarity based vehicle trajectory clustering and anomaly detection. In *Proceedings of the 2005 International Conference on Image Processing, ICIP 2005, Genoa, Italy, September 11-14, 2005*, pages 602–605, 2005.

[22] A. Gionis, A. Hinneburg, S. Papadimitriou, and P. Tsaparas. Dimension induced clustering. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, Illinois, USA, August 21-24, 2005*, pages 51–60, 2005.

[23] A. B. Goldberg, X. Zhu, A. Singh, Z. Xu, and R. D. Nowak. Multi-manifold semi-supervised learning. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics, AISTATS 2009, Clearwater Beach, Florida, USA, April 16-18, 2009*, pages 169–176, 2009.

[24] G. Haro, G. Randall, and G. Sapiro. Translated poisson mixture model for stratification learning. *International Journal of Computer Vision*, 80(3):358–374, 2008.

[25] J. A. Hartigan and M. A. Wong. A k-means clustering algorithm. *JSTOR: Applied Statistics*, 28(1):100–108, 1979.

[26] T. Hastie and P. Simard. *Metrics and Models for Handwritten Character Recognition*, pages 203–219. Birkhäuser Basel, Basel, 1997.

[27] R. Heylen and P. Scheunders. A distance geometric framework for nonlinear hyperspectral unmixing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6):1879–1888, June 2014.

[28] J. Ho, M. Yang, J. Lim, K. Lee, and D. J. Kriegman. Clustering appearances of objects under varying illumination conditions. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003), 16-22 June 2003, Madison, WI, USA*, pages 11–18, 2003.

[29] S. Karygianni and P. Frossard. Tangent-based manifold approximation with locally linear models. *Signal Processing*, 104:232–247, 2014.

[30] D. N. Kaslovsky and F. G. Meyer. Non-asymptotic analysis of tangent space perturbation. *Information and Inference: A Journal of the IMA*, 3(2):134–187, 2014.

[31] M. J. Kearns, Y. Mansour, and A. Y. Ng. An information-theoretic analysis of hard and soft assignment methods for clustering. In *UAI*, pages 282–293. Morgan Kaufmann, 1997.

[32] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from noisy entries. *Journal of Machine Learning Research*, 11:2057–2078, 2010.

[33] D. Lay. *Linear Algebra and its Applications (Third edition)*. Pearson, Addison Wesley, 2006.

[34] E. G. Learned-Miller. Entropy and mutual information. *Department of Computer Science, University of Massachusetts, Amherst*, 2013.

[35] J. Lee. *Introduction to Smooth Manifolds*. Springer, New York, 2003.

[36] E. Levina and P. J. Bickel. Maximum likelihood estimation of intrinsic dimension. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*, pages 777–784, 2004.

[37] A. V. Little, Y. Jung, and M. Maggioni. Multiscale estimation of intrinsic dimensionality of data sets. In *AAAI Fall Symposium*, 2009.

[38] 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 Anal. Mach. Intell.*, 35(1):171–184, 2013.

[39] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 663–670, 2010.

[40] G. Liu and S. Yan. Latent low-rank representation for subspace segmentation and feature extraction. In *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, pages 1615–1622, 2011.

[41] Y. Ma, A. Y. Yang, H. Derksen, and R. M. Fossum. Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM Review*, 50(3):413–458, 2008.

[42] J. Miao and A. Ben-Israel. On principal angles between subspaces in Rn. *Linear Algebra and its Applications*, 171:81 – 98, 1992.

[43] B. Mohar. The Laplacian spectrum of graphs. *Graph Theory, Combinatorics, and Applications*, 2:871–898, 1991.

[44] B. Mohar. Some applications of Laplace eigenvalues of graphs. *Graph Symmetry: Algebraic Methods and Applications*, 497:227–275, 1997.

[45] P. Mordohai and G. G. Medioni. Unsupervised dimensionality estimation and manifold learning in high-dimensional spaces by tensor voting. In L. P. Kaelbling and A. Saffiotti, editors, *IJCAI*, pages 798–803. Professional Book Center, 2005.

[46] J. Munkres. *Analysis On Manifolds*. Advanced Books Classics. Avalon Publishing, 1997.

[47] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-20). Technical Report CUCS-005-96, Department of Computer Science, Columbia University, February 1996.

[48] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proc. Advances in Neural Information Processing Systems 14*, pages 849–856, 2001.

[49] B. Oblak. From the Lorentz Group to the Celestial Sphere. 2015.

[50] V. M. Patel, H. V. Nguyen, and R. Vidal. Latent space sparse subspace clustering. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 225–232, 2013.

[51] V. M. Patel and R. Vidal. Kernel sparse subspace clustering. In *2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27-30, 2014*, pages 2849–2853, 2014.

[52] P. Ren, R. C. Wilson, and E. R. Hancock. Graph characterization via ihara coefficients. *IEEE Trans. Neural Networks*, 22(2):233–245, 2011.

[53] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, Dec. 2000.

[54] V. Runde. *A Taste of Topology*. Universitext. Springer New York, 2007.

[55] C. Sanderson and B. C. Lovell. Multi-region probabilistic histograms for robust and scalable identity inference. In M. Tistarelli and M. S. Nixon, editors, *ICB*, volume 5558 of *Lecture Notes in Computer Science*, pages 199–208. Springer, 2009.

[56] L. K. Saul and S. T. Roweis. Think globally, fit locally: Unsupervised learning of low dimensional manifold. *Journal of Machine Learning Research*, 4:119–155, 2003.

[57] B. Shaw and T. Jebara. Minimum volume embedding. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, AISTATS 2007, San Juan, Puerto Rico, March 21-24, 2007*, pages 460–467, 2007.

[58] B. Shaw and T. Jebara. Structure preserving embedding. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, pages 937–944, 2009.

[59] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

[60] M. Soltanolkotabi and E. J. Candès. A geometric analysis of subspace clustering with outliers. *CoRR*, abs/1112.4258, 2011.

[61] R. Souvenir and R. Pless. Manifold clustering. In *10th IEEE International Conference on Computer Vision (ICCV 2005), 17-20 October 2005, Beijing, China*, pages 648–653, 2005.

[62] M. Spivak. *A Comprehensive Introduction to Differential Geometry*. Publish or Perish, inc., 1999.

[63] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319, 2000.

[64] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[65] V. Torra. On the selection of m for fuzzy c-means. In *2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (IFSA-EUSFLAT-15), Gijón, Spain., June 30, 2015.*, 2015.

[66] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA*, 2007.

[67] H. Tyagi, E. Vural, and P. Frossard. Tangent space estimation for smooth embeddings of Riemannian manifolds. *Information and Inference: A Journal of the IMA*, 2(1):69–114, 2013.

[68] R. Vidal. Subspace clustering. *IEEE Signal Process. Mag.*, 28(2):52–68, 2011.

[69] R. Vidal and Y. Ma. A unified algebraic approach to 2-d and 3-d motion segmentation and estimation. *Journal of Mathematical Imaging and Vision*, 25(3):403–421, 2006.

[70] U. von Luxburg. A tutorial on spectral clustering. *CoRR*, abs/0711.0189, 2007.

[71] A. G. Walters. PCA: Principal Component Analysis. [Online] Available: http://austingwalters.com/pca-principal-component-analysis/.

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

[73] K. Q. Weinberger and L. K. Saul. Unsupervised learning of image manifolds by semidefinite programming. *International Journal of Computer Vision*, 70(1):77–90, 2006.

[74] M. Yin, J. Gao, Z. Lin, Q. Shi, and Y. Guo. Dual graph regularized latent low-rank representation for subspace clustering. *IEEE Trans. Image Processing*, 24(12):4918–4933, 2015.

[75] K. Zhang and J. T. Kwok. Clustered nyström method for large scale manifold learning and dimension reduction. *IEEE Trans. Neural Networks*, 21(10):1576–1587, 2010.

[76] Z. Zhang and H. Zha. Principal manifolds and nonlinear dimension reduction via local tangent space alignment. *CoRR*, cs.LG/0212008, 2002.