Ecole Doctorale ED488 “Sciences, Ingénierie, Santé”

Unsupervised 3D image clustering and extension to joint color and depth segmentation

Titre en Français:

Classification non supervisée d’images 3D et extension à la segmentation exploitant les informations de couleur et de profondeur

Thèse préparée par **Md Abul HASNAT** pour obtenir le grade de :

Docteur de l'Université Jean Monnet de Saint-Etienne

Discipline : Sciences et Technologies de l’Information et de la Communication, Traitement d’images

Laboratoire Hubert Curien, UMR CNRS 5516

Faculté des Sciences et Techniques

Soutenance le 1er Octobre 2014 au Laboratoire Hubert Curien devant le jury composé de:

Yves Delignon	Professeur, Telecom Lille, France.	Rapporteur
Christian Germain	Professeur, Bordeaux Science Agro, France.	Rapporteur
Frank Nielsen	Professeur, École Polytechnique, France.	Examinateur
Elisa Fromont	Maître de Conférences, Université Jean Monnet, Saint Etienne, France.	Examinatrice
Radu Horaud	Directeur de recherche, INRIA Grenoble RhôneAlpes, France.	Co-Directeur de thèse
Alain Trémeau	Professeur, Université Jean Monnet, Saint Etienne, France.	Co-Directeur de thèse
Olivier Alata	Professeur, Université Jean Monnet, Saint Etienne, France.	Directeur de thèse

Abstract

Access to the 3D images at a reasonable frame rate is widespread now, thanks to the recent advances in low cost depth sensors as well as the efficient methods to compute 3D from 2D images. As a consequence, it is highly demanding to enhance the capability of existing computer vision applications by incorporating 3D information. Indeed, it has been demonstrated in numerous researches that the accuracy of different tasks increases by including 3D information as an additional feature. However, for the task of indoor scene analysis and segmentation, it remains several important issues, such as: (a) how the 3D information itself can be exploited? and (b) what is the best way to fuse color and 3D in an unsupervised manner? In this thesis, we address these issues and propose novel unsupervised methods for 3D image clustering and joint color and depth image segmentation. To this aim, we consider image normals as the prominent feature from 3D image and cluster them with methods based on finite statistical mixture models. We consider Bregman Soft Clustering method to ensure computationally efficient clustering. Moreover, we exploit several probability distributions from directional statistics, such as the von Mises-Fisher distribution and the Watson distribution. By combining these, we propose novel Model Based Clustering methods. We empirically validate these methods using synthetic data and then demonstrate their application for 3D/depth image analysis. Afterward, we extend these methods to segment synchronized 3D and color image, also called RGB-D image. To this aim, first we propose a statistical image generation model for RGB-D image. Then, we propose novel RGB-D segmentation method using a joint color-spatial-axial clustering and a statistical planar region merging method. Results show that, the proposed method is comparable with the state of the art methods and requires less computation time. Moreover, it opens interesting perspectives to fuse color and geometry in an unsupervised manner. We believe that the methods proposed in this thesis are equally applicable and extendable for clustering different types of data, such as speech, gene expressions, etc. Moreover, they can be used for complex tasks, such as joint image-speech data analysis.

Résume

L'accès aux séquences d'images 3D s'est aujourd'hui démocratisé, grâce aux récentes avancées dans le développement des capteurs de profondeur ainsi que des méthodes permettant de manipuler des informations 3D à partir d'images 2D. De ce fait, il y a une attente importante de la part de la communauté scientifique de la vision par ordinateur dans l'intégration de l'information 3D. En effet, des travaux de recherche ont montré que les performances de certaines applications pouvaient être améliorées en intégrant l'information 3D. Cependant, il reste des problèmes à résoudre pour l'analyse et la segmentation de scènes intérieures comme (a) comment l'information 3D peut-elle être exploitée au mieux? et (b) quelle est la meilleure manière de prendre en compte de manière conjointe les informations couleur et 3D? Nous abordons ces deux questions dans cette thèse et nous proposons de nouvelles méthodes non supervisées pour la classification d'images 3D et la segmentation prenant en compte de manière conjointe les informations de couleur et de profondeur. A cet effet, nous formulons l'hypothèse que les normales aux surfaces dans les images 3D sont des éléments à prendre en compte pour leur analyse, et leurs distributions sont modélisable à l'aide de lois de mélange. Nous utilisons la méthode dite « Bregman Soft Clustering » afin d'être efficace d'un point de vue calculatoire. De plus, nous étudions plusieurs lois de probabilités permettant de modéliser les distributions de directions: la loi de von Mises-Fisher et la loi de Watson. Les méthodes de classification « basées modèles » proposées sont ensuite validées en utilisant des données de synthèse puis nous montrons leur intérêt pour l'analyse des images 3D (ou de profondeur). Une nouvelle méthode de segmentation d'images couleur et profondeur, appelées aussi images RGB-D, exploitant conjointement la couleur, la position 3D, et la normale locale est alors développée par extension des précédentes méthodes et en introduisant une méthode statistique de fusion de régions « planes » à l'aide d'un graphe. Les résultats montrent que la méthode proposée donne des résultats au moins comparables aux méthodes de l'état de l'art tout en demandant moins de temps de calcul. De plus, elle ouvre des perspectives nouvelles pour la fusion non supervisée des informations de couleur et de géométrie. Nous sommes convaincus que les méthodes proposées dans cette thèse pourront être utilisées pour la classification d'autres types de données comme la parole, les données d'expression en génétique, etc. Elles devraient aussi permettre la réalisation de tâches complexes comme l'analyse conjointe de données contenant des images et de la parole.

Contents

Table des matières

List of Figures

List of Tables

List of Algorithms

Abbreviations

AIC Akaike Information Criteria

BD Bregman Divergence

BIC Bayesian Information Criteria

BSC Bregman Soft Clustering

EFD Exponential Family of Distributions

EM Expectation Maximization

GMM Gaussian Mixture Model

HAC Hierarchical Agglomerative Clustering

IC Information Criteria

ICL Integrated Completed Likelihood

JCSA Joint Color-Spatial-Axial

KLD Kullback Leibler Divergence

MBC Model Based Clustering

MLE Maximum Likelihood Estimates

MM Mixture Model

MML Minimum Message Length

mWD multivariate Watson Distribution

NDA Non Dominant Axial

RAG Region Adjacency Graph

RM Region Merging

vMF von Mises-Fisher

vMFMM vMF Mixture Model

WMM Watson Mixture Model

WPLR Weighted Piecewise Linear Regression

List of Symbols

Symbol	Description
$C(.)$	Penalization function to compute Information Criterion.
$D(.)$	Divergence function.
$F(.)$	Log normalizing function.
$G(.)$	Legendre dual of the log normalizing function.
$I(.)$	Modified Bessel function.
$M(.)$	Kummer's confluent hypergeometric function.
$P(.)$	Number of free parameters.
$Q(.)$	Normalization constant of the von Mises-Fisher (or Langevin) distribution.
$V(.)$	Density function of the von Mises-Fisher (or Langevin) distribution.
$W(.)$	Density function of the multivariate Watson distribution.
$d(.)$	Distance function.
$f(.)$	Density function of a distribution belonging to the exponential families.
$f_g(.)$	Density function of the multivariate Gaussian distribution.
$g(.)$	Mixture model.
$k(.)$	Carrier measure.
$p(.\|\mathbf{x_i})$	Posterior probability.
$q(.)$	Kummer-ratio.
$t(.)$	Sufficient statistics.
$w(.)$	Function to compute edge weight.
$\delta(.)$	Shortest distance from a data point to the closest center.
Σ	Variance-covariance symmetric positive-definite matrix.
∇	Gradient operator.
Γ	Set of labels corresponding to set of observations.
Θ_k	Set of parameters corresponding to a mixture model with k components.
Ψ	Matrix associated with the natural parameter of multivariate Gaussian distribution.
Φ	Matrix associated with the expectation parameter of multivariate Gaussian distribution.
E	Set of edges.
M	Number of observations in a subset of samples.
N	Total number of observations.
P	Region merging predicate.
R	Set of regions.
V	Set of nodes.

$\mathbf{X}$	Set of observations.
Z	Total number of regions or nodes.
d	Dimension of an observation.
e	An edge among two nodes.
k	Number of clusters.
k_{max}	Maximum number of clusters.
k_o	Optimal number of clusters.
i	Index of an observation.
j	Index of a cluster or class.
p	Posterior probability.
r	A region of a segmented image.
v	A node in the graph V.
$\mathbf{x}_i$	Single observation.
ω	Weight associated with a line.
τ	Weight associated with the right-sided line.
θ	Natural parameter.
η	Expectation parameter.
γ	Label of an observation.
μ	Mean of a cluster or probability distribution.
π	Mixing proportion or prior class probability.
ψ	Vector associated with the natural parameter of multivariate Gaussian distribution.
ϕ	Vector associated with the expectation parameter of multivariate Gaussian distribution.
λ	A multiplier.
ξ	Initial centers of kmeans++ algorithm.
ρ	Rising factorial.
$\jmath$	Order of Bessel function.
ν	Vector associated with the mean direction for Watson distribution.

Chapter 1

Introduction

The widespread use of consumer color cameras in a variety of applications enlarges the research areas related to image processing, computer vision and robotics. Over the years the capability of these cameras improves significantly to provide rich and quality information, e.g., high resolution color image, high speed image capture, high accuracy, etc. Undoubtedly, such quality of information boosted the performance of the applications in the respective areas. However, the use of only color information is limited up to certain extent because of the several reasons (Dal Mutto et al., 2012b; Rusu, 2013), to name few:

a. These images are the 2D projection of the real world 3D scene, hence there is a loss of shape/geometric information due to the missing third dimension or depth information.

b. These images do not always contain enough information in order to disambiguate and interpret all scene objects properly. For example, they tend to fail in a uniform color region as well as in a heavily textured region.

c. They are often sensitive to the scene properties such as reflection, illumination etc. For example, they are unable to handle environments with spatially varying illumination which causes several effects of shadows, such as in indoor or outdoor scenes.

Researches have shown that, these limitations have numerous effects especially in the context of image understanding and analysis (Dal Mutto et al., 2012b; Rusu, 2013). On the other hand, it is possible to overcome these limitations by incorporating color information with shape/geometric information which is computed or captured in the form of depth image or 3D point clouds. *This provides us the motivation to work with 3D images.*

A variety of different techniques (e.g., shape from X, stereo vision, etc.) and devices (laser scanner, stereo camera, time-of-flight camera etc.) are available for the acquisition or computation of the depth/3D information (Lanman and Taubin, 2009; Moons, 2009; Dal Mutto et al., 2012b). Until a few years ago, the research activities related to depth images manipulation were not as widespread as they were with color images. An obvious reason for this was the limited affordability of the cameras and availability of the computational resources and techniques for depth image acquisition and computation (Dal Mutto et al., 2012b). Interestingly, in the past few years the research activities related to 3D information processing increased significantly (Henry et al., 2012; Izadi et al., 2011; Han et al., 2013; Khoshelham and Elberink, 2012), thanks to the Microsoft Kinect sensors (Zhang, 2012) which provide access to depth image with a camera that costs around 150 USD. A true reflection of this scenario can be observed in this thesis as this work begins after the introduction of the Microsoft Kinect in the consumer market. *The main focus of this research work was to manipulate particularly the depth images from the Kinect camera for the task of scene understanding and analysis.* Our primary interest to depth image solely was motivated by the fact that the color accuracy of kinect camera is very low, particularly in regards to the hue and saturation channels observed in the indoor scenes.

Due to the availability of low-cost 3D depth sensors, access to the depth information at a reasonable frame rate is widespread now. These information have been employed to enhance the capability of existing applications in computer vision, graphics and robotics, see Han et al. (2013) for a detail review. Kinect type low-cost cameras (Han et al., 2013; Zhang, 2012) allow the direct acquisition of the third dimension (also called depth) information of the scene points. Then, using the camera calibration parameters (Herrera et al., 2012; Keane et al., 2011) one can easily reconstruct the 3D position information of the scene being imaged (Khoshelham and Elberink, 2012). Moreover, Kinect also provides synchronize color information along with depth, which opens the possibility to jointly exploit the color and depth for image analysis and relevant tasks. We refer the readers to Chapter 3 of the book of Dal Mutto et al. (2012b) for further technical details related to Kinect camera.

Kinect is a structured light based depth sensing camera (Zhang, 2012; Dal Mutto et al., 2012b). It projects randomly coded infrared speckle patterns to the scene and then compute disparity information by decoding the observed patterns through an infrared camera. It attracts high interest from the research community and industries. Therefore a number of software programs, to interact with Kinect, have been developed and are freely available (Keane et al., 2011). Despite numerous benefits,

there are several limitations of the Kinect like cameras such as (Dal Mutto et al., 2012b; Han et al., 2013) :

a. Depth acquisition is limited within a certain range of distance, preferably less than 3.5 meters.

b. Measurements depend on the scene illumination and lighting condition which may interact with the projected patterns.

c. Measurements depend on the reflectance properties of the scene surfaces that cause overexposed or low reflectivity in the infrared image.

d. Measurement directions and occlusions often cause the absence of depth values, also called missing depth values.

Due to the above limitations, Kinect performs poorly in the outdoor environment. Moreover, in the outdoor environments the depth acquisition is more complex to realize. *Therefore, in this thesis we limit our research only for the indoor environments.*

Kinect captures images at a reasonable frame rate, 30 frames/sec. Therefore, it provides the opportunity to work with motion information. In this thesis, *we mainly focus on the single images from Kinect* and plan to extend it for multi-frame analysis in order to perform several tasks, such as co-segmentation, 3D model reconstruction, etc.

Over the past decades, the task of image analysis and segmentation has received significant attention from the community. It is frequently considered as a low level image/vision task which is employed as a preprocessing step for many advanced applications. A large number of methods for intensity/color image analysis have been proposed in the literature, see Chapter 5 of Szeliski (2011) for a detail review. Many of these methods have been either modified or directly employed to analyze depth images, see Chapter 6 of Dal Mutto et al. (2012b) for a detail review. Beside these, a number of recent research activities, e.g., Gupta et al. (2013) and Taylor and Cowley (2013) provide different methodologies to exploit depth/3D images for indoor scene understanding and analysis. There are several common properties of these proposed methods, such as: (a) they incorporate depth as a complementary information with color image, which is called RGB-D image and (b) most of them are based on learning a classifier from available training data with ground truth, i.e., supervised approach (Gupta et al., 2013; Ren et al., 2012; Silberman et al., 2012; Koppula et al., 2011; Lai et al., 2011). From our study, we observe that the unsupervised approaches received

relatively less attention in the context of depth image analysis. Moreover, it is not completely evident how certain features (e.g. depth, 3D, surface normal) individually contribute for the objective of scene analysis. *To address these issues properly, we initially focus on developing an unsupervised depth image analysis method using the primitive depth features. Later, we focus on extending our method towards RGB-D indoor image analysis.*

A common approach to analyze the depth image is to consider it as a grayscale image (Dal Mutto et al., 2012b) and then apply standard image analysis techniques (Szeliski, 2011) on it. This approach is relatively simpler compare to color image as the edges are sharper and the complex texture patterns are absent in the depth maps (Dal Mutto et al., 2012b). However, such approaches fail to identify long uniform structures when they spread into a wide range of depth values, such as the walls in a room. In general these structures are divided into several regions rather than being identified as a single region. Therefore, it is suggested to use 3D position as the feature rather than only depth value for each pixel (Dal Mutto et al., 2012b,a; Rusu, 2013). Beside the 3D position, surface normal is considered as an important feature, which describes the planar property of each pixel of a depth image (Rusu, 2013; Holz et al., 2012).

The planar surfaces are prominent geometric primitives of the Man-made environment and are often employed for scene decomposition (Silberman et al., 2012; Ren et al., 2012; Gupta et al., 2013; Holz et al., 2012) and grouping (Taylor and Cowley, 2013). Detected and segmented planes are able to adequately model the surface of the main structures in the indoor environment (Holz et al., 2012). These surfaces are generally located with two different approaches: (a) using model (plane) fitting by applying the RANSAC algorithm on the 3D point clouds (Rusu, 2013; Taylor and Cowley, 2013) and (b) by clustering the surface normals using k-means or mean-shift method (Dal Mutto et al., 2012b; Holz et al., 2012). We observed several common facts about these approaches such as:

a. These methods do not consider any particular model (e.g. mixture models with statistical distributions) for generating the depth image, and hence an interesting parametric model based study for the depth data is missing.

b. They require explicit settings of parametric factors, which is often difficult for the non-experts users to analyze scene.

c. They do not explain the pixels which belong to the non-planar surfaces and

d. They do not provide a clear view of how these methods can be extended for scene analysis with additional features.

The above facts motivate us to conduct further research on: (a) how to best exploit the surface normals for analyzing depth images of indoor environment and (b) how it can be extended for further analysis by incorporating additional features in an unsupervised manner.

Cluster analysis is often employed for the task of image analysis and segmentation (Szeliski, 2011). To perform clustering, image pixels are described by different features such as intensity, color, position, texture, etc. We consider the surface normal as a feature and apply clustering to analyze the depth images from it. To this aim, we employ a model based clustering approach (Fraley and Raftery, 2002; Melnykov and Maitra, 2010). This choice was driven due to the following reasons:

a. It employs a generative model, which assumes that the data are issued from a mixture of certain statistical distributions (Murphy, 2012). In statistics, such models are theoretically well-judged and are able to provide greater insight into the anatomy of the clusters (Banerjee et al., 2005a).

b. These models are well fitted into the unsupervised classification paradigm. Learning of parameters is automatically done through the mixture model estimation process (Figueiredo and Jain, 2002). The number of clusters can be automatically determined using certain model selection criteria (Alata and Quintard, 2009; Biernacki et al., 2000; Fraley and Raftery, 2002) or using non-parametric Bayesian approach (Murphy, 2012; Cherian et al., 2011).

c. Obtained clusters are explainable through the parameters of the model. For example, using the prior probability, mean and covariance, one can interpret the clusters provided by a Gaussian Mixture Model (Murphy, 2012). These parameters provide very useful information, e.g., the covariance matrices of multivariate data have been used as feature descriptors in many areas in computer vision (Cherian et al., 2011).

d. These models can be easily extended in several ways, such as: (a) forming a feature vector which concatenates different types of features and (b) with the naïve Bayes (Murphy, 2012) assumption which assumes that features are independent of each other.

Most commonly, the Gaussian distribution is employed for clustering image with mixture models (Alata and Quintard, 2009; Garcia and Nielsen, 2010; Ma et al., 2007; Nguyen and Wu, 2013). Although the Gaussian Mixture Model is well adapted with a variety of computer vision applications (Szeliski, 2011), it can also be argued that it is not always the best choice (Sefidpour and Bouguila, 2012; Gopal and Yang, 2014). For example, the Hue (color attribute) values are circular in nature and therefore a circular probability distribution (e.g., the von Mises distribution (Mardia and Jupp, 2009)) is an appropriate choice for it. Therefore, in practice the best approach is first to understand the true nature of the data and next to select a probability distribution that best suits it.

Surface normal is a 3D unit vector that provides the direction of each pixel in the depth image. The sample space for surface normals is the unit-sphere manifolds. Directional distributions (Mardia and Jupp, 2009) are the standard choice to construct a Mixture Model for such samples (Gopal and Yang, 2014). The fundamental directional distributions (Mardia and Jupp, 2009) are the von Mises-Fisher, Watson, Kent, etc. *Therefore, in this thesis our primary focus is to propose model based clustering methods with the directional distributions (Mardia and Jupp, 2009) in order to perform unsupervised clustering of the depth images with surface normals. Our secondary objective is to extend these methods for clustering heterogeneous (joint color and depth) data and propose an unsupervised RGB-D scene analysis method.*

Expectation Maximization (EM) is the most common method to estimate the parameters of a mixture model. It consists of an Expectation and a Maximization steps which are iteratively employed to maximize log likelihood of the data. Banerjee et al. (2005b) proposed Bregman soft clustering algorithm which simplifies the computationally expensive M-step. Moreover, it has the following attractive features: (a) it is equivalent to EM for a mixture of exponential family of distributions (Murphy, 2012); (b) it is applicable to mixed data types and (c) its computational complexity is linear in the data points. The fundamental directional distributions belong to the exponential family (Mardia and Jupp, 2009). *This motivates us to develop Bregman soft clustering methods for the directional distributions. Moreover, we set several objectives at this point: (a) to exploit such method within the model based clustering framework and (b) to extend such method for joint clustering task.*

In this thesis, we propose methods to analyze depth images. To develop these methods we focus on several issues: (a) theoretically well justified; (b) unsupervised, i.e., no learning from training data; (c) provide better classification accuracy w.r.t.

the state of the art; (d) computationally efficient (e) extendable with additional information and (f) applicable to a variety of domains other than image processing and computer vision. First, we empirically validate the proposed methods using a synthetic data-set which is generated through standard sampling procedures (Dhillon and Sra, 2003). Then, we apply these methods on real depth images to cluster surface normals. As per the observed results, the proposed methods can be considered as potential tools for bottom up depth image analysis and segmentation (Szeliski, 2011).

We are aware about the fact that the directional features alone have limited capability to provide a complete semantic categorization of indoor scenes. For this reason, we extend our initially proposed methods such that they are able to incorporate additional features. To this aim, we consider color, 3D and surface normal as features and propose a combination of joint clustering and region merging method. We apply the proposed method to analyze color image synchronized with depth image provided by Kinect camera, which is also called RGB-D image. We employed standard benchmarks (Arbelaez et al., 2011; Freixenet et al., 2002) to evaluate the proposed method w.r.t. the state of the art methods.

Publications

The following research papers are accepted or submitted during this thesis:

J1 Md. Abul Hasnat, Olivier Alata and Alain Trémeau, "Model Based Clustering with von Mises-Fisher Mixture Model: Application to Depth Image Analysis", Revised version submitted to Statistics and Computing (STCO).

C1 Md. Abul Hasnat, Olivier Alata and Alain Trémeau. "Model Based Clustering for 3D Ddirectional Features: Application to Depth Image Analysis", Accepted in the International Conference on Image Processing (ICIP), October 2014.

C2 Md. Abul Hasnat, Olivier Alata and Alain Trémeau. "Unsupervised Clustering of Depth Images using Watson Mixture Model", Accepted in the 22nd International Conference on Pattern Recognition (ICPR), August 2014.

C3 Md. Abul Hasnat, Olivier Alata and Alain Trémeau. "RGB-D image segmentation using joint clustering and region merging", Accepted in the British Machine Vision Conference (BMVC), September 2014.

W1 Md. Abul Hasnat, Olivier Alata and Alain Trémeau, “Hierarchical 3-D von Mises-Fisher Mixture Model”, In Proc. of the ICML Workshop on Divergences and Divergence Learning, Atlanta, Georgia, USA, 2013.

Oral presentations without publication

1 A. Hasnat, O. Alata and A. Trémeau, "Model based clustering for directional features and application to depth image", PEPS WAVE days, the 18th and 19th of November, 2013, Bordeaux, France.

2 A. Hasnat, O. Alata and A. Trémeau, "Model based clustering using color and depth information", GDR ISIS day on "joint analysis of RGB-D images", the 6th of february, 2014, Telecom Paris, France.

3 A. Hasnat, O. Alata and A. Trémeau, "Model based clustering using color and depth information", SIERRA (Signal et Images en RÃľgion RhÃťne-Alpes) day on "Adaptive methods and models", the 25th of march, 2014, Ecole des Mines de Saint-Etienne, France.

Contributions

We can summarize our contributions in this thesis as follows:

- A Model based clustering method for the fundamental directional distributions called the von Mises-Fisher distribution (vMF) and the multivariate Watson distribution (mWD), published or submitted in the research papers *J1, C1, C2 and W1*. The key contributions are: (a) a mathematical formulation to compute Bregman divergence (Banerjee et al., 2005b) among the vMFs and the mWDs; (b) an efficient soft clustering method for the vMF Mixture Models (Banerjee et al., 2005a) and the mWD Mixture Models (Sra and Karp, 2013); (c) hierarchical mixture models for the vMF and mWD and (d) an empirical model selection strategy based on the combination of model selection criteria (Alata and Quintard, 2009; Biernacki et al., 2000; Figueiredo and Jain, 2002) and linear regression fit (Baudry et al., 2010; Salvador and Chan, 2004).

- An unsupervised RGB-D image segmentation using joint clustering and region merging, published in *C3*. The key contributions are: (a) propose a statistical RGB-D image generation model that incorporates both color and geometry of a scene; (b) develop an efficient soft clustering method by exploiting the Bregman divergence (Banerjee et al., 2005b) to cluster heterogeneous data w.r.t. the

image model; (c) propose a statistical region merging method based on planar geometry, which can be used with other RGB-D segmentation methods and (d) provide a benchmark on the NYU depth database V2 (Silberman et al., 2012) using standard evaluation metrics (Arbelaez et al., 2011; Freixenet et al., 2002).

In this thesis, we developed several methods to cluster unit vectors and also to cluster mixed data types. These methods are device and dataset independent, and hence can be applicable to the data obtained from different types of depth sensing devices and relevant datasets. We experiment these methods mainly in the context of image processing and computer vision. However, we believe that the proposed methods can be equally useful for a number of different domains, for example to cluster motion, speech, text, gene expressions, joint speech-image, joint motion-image data etc.

Organization of this thesis

The outline of this thesis is as follows:

- **Chapter 2** presents the background and methodology to perform model based clustering. Here, first we introduce the model based clustering method and discuss related work. Then, we provide the background of several connected topics: exponential family of distributions, Bregman divergence, Bregman soft clustering, hierarchical meta-clustering and several model selection strategies. Finally, we present a complete model based clustering method, which is developed during this thesis.

- **Chapter 3** presents our proposed (developed during this thesis) model based clustering methods with directional distributions and provides experimental results. Here, first we provide the background related to the directional distributions and associated mixture models. Then, we present the methodologies to compute the Bregman divergence for these distributions and extend it for model based clustering. Finally, we provide the experimental results, first with synthetic data and then with real depth images. We compare the results with the state of the art directional data clustering methods and the relevant clustering based image analysis methods.

- **Chapter 4** presents an extension of the methods, developed in the previous Chapters, to perform RGB-D image analysis. In this Chapter, we present a statistical image generation model that incorporates the color and geometry of

the scene. Then, we present a joint color-spatial-directional clustering method followed by a statistical planar region merging method. Finally, we provide the experimental results and a benchmark of the NYU depth database w.r.t. the state of the art of unsupervised RGB-D segmentation methods.

- **Chapter 5** provides conclusions and possible extensions of the methods to perform different computer vision tasks.

Introduction

L'accès aux séquences d'images 3D s'est aujourd'hui démocratisé, grâce aux récentes avancées dans le développement des capteurs de profondeur ainsi que des méthodes permettant de manipuler des informations 3D à partir d'images 2D. De ce fait, il y a une attente importante de la part de la communauté scientifique de la vision par ordinateur dans l'intégration de l'information 3D. En effet, des travaux de recherche ont montré que les performances de certaines applications pouvaient être améliorées en intégrant l'information 3D. Cependant, il reste des problèmes à résoudre pour l'analyse et la segmentation de scènes intérieures comme (a) comment l'information 3D peut-elle être exploitée au mieux? et (b) quelle est la meilleure manière de prendre en compte de manière conjointe les informations couleur et 3D? Dans cette thèse, nous apportons des éléments de réponses à ces deux questions dans un contexte de classification non supervisée. Nous avons postulé que les informations principales à prendre en compte était la couleur, la position dans l'espace 3D et les normales aux surfaces. Les deux premières informations peuvent être décrites à l'aide de lois de Gauss multivariées et la troisième à l'aide de distributions directionnelles. Ces dernières appartiennent aussi à la famille exponentielle de distributions. Ainsi, dans le deuxième chapitre nous proposons une méthode de type classification basée modèle (Model Based Clustering - MBC) pour la famille exponentielle de distributions exploitant la divergence de Bregman, la classification ascendante hiérarchique ainsi qu'une approche parcimonieuse pour la sélection de modèle. Au cours du troisième chapitre, nous développons la méthode de type MBC pour deux distributions directionnelles: la loi de von Mises-Fisher et la loi de Watson. La méthode de type MBC proposée est ensuite modifiée dans le chapitre quatre pour pouvoir faire de la segmentation conjointe prenant en compte la couleur, les positions spatiales et les normales aux surface, en introduisant une méthode de fusion de régions exploitant un graphe d'adjacence, la couleur et des propriétés géométriques. Au cours des différents chapitres, nous donnons des résultats expérimentaux obtenus sur des données simulées et des données réelles et nous les comparons aux méthodes de l'état de l'art.

Chapter 2

Model Based Clustering with Exponential Family of Distributions

Résumé: La classification "basée" modèle (Model Based Clustering - MBC) est une méthode qui permet de regrouper les données en partant de l'hypothèse que leur distribution est une loi de mélange. Dans ce chapitre, nous proposons une nouvelle méthode de type MBC pour une loi de mélange contenant des composantes dont les distributions appartiennent à la famille exponentielle. Les principaux aspects de cette méthode sont: (a) d'offrir une solution pertinente pour estimer les paramètres de la loi de mélange ; (b) de générer une hiérarchie de modèles et (c) de sélectionner le modèle optimal. La méthode d'estimation des paramètres des modèles est développée en exploitant les propriétés de la divergence de Bregman et la classification ascendante hiérarchique. La méthode de sélection de modèle est construite à partir d'une approche parcimonieuse et d'une méthode d'évaluation exploitant un graphe. Pour finir, la méthode proposée permet d'obtenir une classification non supervisée des données.

Model Based Clustering (MBC) is a method that clusters data with an assumption of mixture model structure. In this Chapter, we propose a novel MBC method for a finite statistical mixture model based on the exponential family of distributions. The main focuses of the proposed method are: (a) provide efficient solution to estimate the parameters of a mixture model; (b) generate a hierarchy of models and (c) select the optimal model. To this aim, we develop a Bregman soft clustering method for a mixture model. Our model estimation strategy exploits Bregman divergence and hierarchical agglomerative clustering. Whereas, our model selection strategy comprises parsimony based approach and an evaluation graph based approach. Overall, the proposed method performs an unsupervised classification of the data.

2.1 Introduction

Clustering or cluster analysis can be defined as the task to automatically identify the groups of similar observations from a given set of data points. For example, to perform image segmentation (Szeliski, 2011), cluster analysis identifies groups of similar pixels based on certain features as well as certain measure of distance. However, most clustering methods have the limitation to pre-specify the number of clusters as an external input. Model based clustering (Fraley and Raftery, 2002, 2007; Zhong and Ghosh, 2003; Melnykov and Maitra, 2010) is a well-established method that can be used to overcome this limitation.

Model based clustering assumes a generative model, i.e. each observation is a sample from a finite mixture of probability distributions (Biernacki et al., 2000). In general, it consists of: (a) defining a probabilistic model (ex: mixture model) of the data; (b) optimizing an objective function, such as maximizing the value of likelihood function; (c) generating a set of models and (d) finally, selecting an optimal model based on a specific criterion. As an outcome, it provides a probabilistic clustering, also called soft clustering of the data. See Fraley and Raftery (2002) for a complete overview of this clustering method and see Zhong and Ghosh (2003) for different variations of this method.

The multivariate Gaussian distribution has been mostly employed in the Model Based Clustering (MBC) framework (Fraley and Raftery, 2002, 2007; Fraley et al., 2012; Zhong and Ghosh, 2003; Wehrens et al., 2004). This provides a principled statistical approach to clustering as it assumes that the samples are issued from a finite mixture of the Gaussian distributions. The goal in this approach is to estimate the Gaussian Mixture Model (GMM) parameters as well as to select the GMM with optimal number of components. Clustering with the GMM requires the correct estimation of the covariance structure (Fraley and Raftery, 2007), such as spherical, diagonal and ellipsoidal. Therefore, a number of GMMs with different choices of covariance structures as well as with different number of components are fitted for the data. Afterwards, the best GMM is selected using a model selection criterion. Although GMM is widely employed for MBC methods, it would be interesting to develop a generalized MBC framework which includes a number of other probability distributions.

Model based clustering methods use the Expectation Maximization (EM) method to estimate a mixture model, i.e. to learn the parameters (Fraley and Raftery, 2007; Fraley et al., 2012; Melnykov and Maitra, 2010). It consists of an Expectation (E-step)

and a Maximization (M-step) step. The E-step and M-step are iteratively employed to maximize log likelihood of the data, while considering constraints in the optimization goal (Murphy, 2012). The M-step of the EM method is often computationally expensive. Banerjee et al. (2005b) proposed Bregman Soft Clustering (BSC) algorithm which performs Maximum Likelihood Estimates (MLE) of the mixture model parameters using the EM method. Compare to the other soft clustering methods, BSC has the following attractive features:

- It is equivalent to the EM method for the mixture of exponential family of distributions (Nielsen and Garcia, 2009; Bishop, 2006).
- It simplifies the computationally expensive M-step.
- It is applicable to mixed data types.
- Its computational complexity is linear in the data points.

Bregman soft clustering is a centroid based parametric clustering method (e.g., k-means), which arises by special choice of Bregman divergence (Banerjee et al., 2005b). Bregman divergence generalizes a large number of distortion functions which are commonly used in the data clustering problems (Banerjee et al., 2005b; Liu et al., 2012). Naturally, this allows the computation of relative entropy (KL Divergence) between statistical distributions. Garcia and Nielsen (2010) exploited this and proposed a method to construct a hierarchy of mixture models. This hierarchy of models can be considered as the set of models with different number of components.

Due to the bijection between Bregman divergence and the Exponential Family of Distributions (EFD), Bregman Soft Clustering (BSC) method can be effectively developed using statistical mixture models with any member of EFD (Nielsen and Garcia, 2009). However, to develop BSC for any distribution, it is necessary to obtain the canonical representation of the density function. Nielsen and Garcia (2009) provided such representation for a number of probability distributions.

The properties of the model based clustering, Bregman soft clustering and Bregman divergence provide us the motivation to exploit them in a single method. Particularly, we want to develop a clustering method which has the following features:

- Applicable to a variety of different types of data.
- Extendable with a number of probability density functions.
- Computationally efficient clustering.

- Efficiently generate the set of models.
- Automatically select the number of clusters.

Moreover, the proposed method will extend the capability and efficiency of the model based clustering framework with numerous benefits which are mentioned above.

Number of components selection is one of the most prominent issues in cluster analysis. An incorrect selection leads to over-fit or under-fit the data (Figueiredo and Jain, 2002). In general, model based clustering methods employ a parsimony based approach (Melnykov and Maitra, 2010; McLachlan and Peel, 2004) to select the best model. A different type of approach performs evaluation on graph/plot generated from certain model selection criteria (Baudry et al., 2010; Salvador and Chan, 2004). The idea is to select optimal model by detecting certain change (called kink/knee/elbow (Murphy, 2012; Salvador and Chan, 2004)) in the plot. In practice, none of these two approaches uniquely exhibits desired performance for all dataset. Therefore, we aggregate the best from both approaches in order to determine our model selection strategy.

In this Chapter, we present a novel clustering method, which follows the principals of model based framework (Fraley and Raftery, 2002). To this aim, we begin with the development of Bregman soft clustering for a statistical mixture model based on the exponential family of distributions. Then, we generate a set of models using hierarchical agglomerative clustering with the objective to minimize Bregman divergence among statistical distributions. Finally, we apply a combination of parsimony based model selection (Melnykov and Maitra, 2010) and evaluation graph based approach (Baudry et al., 2010; Salvador and Chan, 2004) to select the optimal model.

The outline of the rest of this Chapter is as follows: Section 2.2 discusses related work. Section 2.3 describes the necessary background of clustering methods. Then, Section 2.4 presents the method to generate a hierarchy of models and Section 2.5 presents several model selection methods. Next, the complete model based clustering method is presented in Section 2.6. Finally, Section 2.7 provides discussion and conclusions.

2.2 Related Work

Model based clustering estimates a model for the data and produces probabilistic clustering that quantifies the uncertainty of observations belonging to components of

the mixture (Fraley and Raftery, 2007). The resulting model can be used for a variety of problems, such as for multivariate analysis, density estimation, discriminant analysis and automatically select the number of clusters. This clustering technique has been applied in a number of studies (Fraley and Raftery, 2007) such as multivariate image analysis, magnetic resonance imaging, microarray image segmentation, statistical process control and food authenticity. Several software programs, such as mclust (Fraley and Raftery, 2002) and HDclassif (Bergé et al., 2012) are available online to cluster data with this method.

Model based clustering identifies the best model (number of clusters and structure of component parameters if necessary) for the data by fitting a set of models with different parameterizations and/or number of components and then applying a statistical criterion for model selection (Fraley and Raftery, 2007; Melnykov and Maitra, 2010; Figueiredo and Jain, 2002; Biernacki et al., 2000). Therefore, three prominent issues arise: (a) What type of model to estimate?; (b) How many models? and (c) Which criterion to select the best model? Answers of these issues lead to a complete clustering method.

Type of models (issue (a), "what type?") is often specified a priori (Zhong and Ghosh, 2003). Particularly, it is related to the selected probability distribution which is considered to construct a statistical mixture model. The Gaussian distribution is mostly employed in model based clustering methods (Fraley and Raftery, 2002; Zhong and Ghosh, 2003; Fraley and Raftery, 2007; Melnykov and Maitra, 2010; Bergé et al., 2012) as they represent in practice the most commonly used mixture models (Garcia and Nielsen, 2010).

Mixture models, also called latent variable models (Murphy, 2012) have been extensively used in a number of different domains. For example, the Gaussian Mixture Model (GMM) has been used for different tasks such as segmentation (Garcia and Nielsen, 2010; Permuter et al., 2006; Nguyen and Wu, 2013; Verbeek et al., 2003), color space characterization for image analysis (Alata and Quintard, 2009), shape retrieval (Liu et al., 2012), data compression (Ma et al., 2007), speaker verification (Reynolds et al., 2000), large margin classification (Sha and Saul, 2006), supervised classification (Fernando et al., 2012) and cluster analysis (Fonseca and Cardoso, 2007; Biernacki et al., 2000; Figueiredo and Jain, 2002; Fraley and Raftery, 1998; Baudry et al., 2010; Vlassis and Likas, 2002), etc. However, it can be argued that GMM is not always the most appropriate choice (Sefidpour and Bouguila, 2012). Besides the Gaussian distribution, mixture models based on other probability distributions also exist and are used in practice. For example, mixture of multivariate Bernoulli

distributions is used for clustering bit vectors such as digits (Murphy, 2012) and text classification (Juan and Vidal, 2002), mixture of Student's-t distributions is used for image segmentation (Nguyen and Wu, 2012; Sfikas et al., 2007), mixtures of Beta distributions is used for clustering DNA methylation data (Houseman et al., 2008), and so on.

Despite having established methods for mixture models based on different distributions, it is particularly interesting to have a framework that generalizes a group of distributions. The exponential family of distributions is a broad class consists of many important probability distributions, such as Gaussian, Bernoulli, Dirichlet, etc. (Nielsen and Garcia, 2009; Bishop, 2006). Banerjee et al. (2005b) derived bijection between Bregman divergence and the exponential family and proposed Bregman soft clustering algorithm. This algorithm provides clustering method that generalizes all mixture models based on the exponential family of distributions. Nielsen and Garcia (2009) provided formulations for a number of distributions of the exponential family and software (jMEF) for estimating models and parameters. In this thesis work, we follow their methodology and extend it. Therefore, we consider the exponential family as the model type (issue (a), "what type?") to be used in our model based clustering framework.

Number of models to generate (issue (b), "how many?") focuses on generating models with different numbers of components within a certain bound (ex: k_{min} to k_{max}). Methods which employed this type of bound are called deterministic method (Figueiredo and Jain, 2002). A Hierarchical Agglomerative Clustering (HAC) scheme with an objective function is often employed to generate a set of models in a deterministic approach. Fraley and Raftery (2002) used maximization of classification likelihood as the objective function for HAC. However, for large number of samples their approach is inefficient w.r.t. computational time and memory requirements. Moreover, such objective does not perform well when samples are not well separated (Melnykov and Maitra, 2010). Baudry et al. (2010) proposed an objective function based on entropy minimization. In their approach, two components are selected for merging such that the entropy of the resulting clustering is minimized. Zhong and Ghosh (2003) and Goldberger and Roweis (2004) employed minimum KL Divergence as the objective function. Recently, Garcia and Nielsen (2010) proposed a mixture model simplification method with Bregman divergence, which generates a hierarchy of mixture models by fusing centroids in natural/exponential parameter space. We found that, this approach is well suited for us due to the fact that: (a) it can be

employed to efficiently generate a set of mixture models and (b) it guarantees the structural relationship (Zhong and Ghosh, 2003) among the mixture models.

Model selection based on certain criterion (issue (c), "what objective function?") is one of the most critical issues for any model based clustering method (Burnham and Anderson, 2002). In general, such objective function is defined based on minimizing a model selection criterion. Type of approaches that incorporates such objective function is referred to as parsimony based approach (Melnykov and Maitra, 2010). See Figueiredo and Jain (2002) for a list of different criteria and their categorization. For example, Fraley and Raftery (2002) used the Bayesian Information Criteria (BIC), Figueiredo and Jain (2002) employed the Minimum Message Length (MML) and Biernacki et al. (2000) proposed the Integrated Completed Likelihood (ICL). Alata and Quintard (2009) applied a different formulation called Φ_β criterion that computes model penalization term with different β parameter values ($0 < \beta < 1$). An advantage of this criterion is that, certain values of β allow computing other criteria such as Akaike Information Criterion (AIC) (Burnham and Anderson, 2002) and Bayesian Information Criterion (BIC). In general, the above mentioned information criteria should provide the desired model with the true number of mixture components. However, these criteria are mostly successful when the data can be modeled with the assumed mixture model. Unfortunately, in many practical situations the real data cannot be completely described by the assumed models and hence model selection with information criteria fails. A number of different solutions are proposed in literature that we will discuss shortly.

Beside the parsimony based model selection, there exists a different family of approaches that can be used to analyze plot/curve/evaluation graph (Salvador and Chan, 2004). In general, Bayesian Information Criterion (BIC) is used to generate a plot (let us call it BIC plot). The idea of BIC plot analysis is to find optimal number of components by detecting the point in the plot where BIC plot exhibits an abrupt change. In literature, methods associated to detecting such change in a point is often referred to as kink/knee/elbow detection process (Murphy, 2012). For example, Salvador and Chan (2004) proposed the L-method which detects elbow by fitting two lines. Zhao et al. (2008) proposed the global angle detection on the BIC plot in order to detect the knee. Other than the BIC plot, Baudry et al. (2010) employed linear regression fit in a rescaled entropy plot. They demonstrated that with the GMM their approach performs similar to the ICL (Biernacki et al., 2000) criteria.

Apart from the above mentioned methods, there are numerous methods to compute the optimal number of components from a set of candidate clustering models (Murphy, 2012). However, all of them do not fit within the context of this research. Among the closely similar approaches, we studied the method called Gap statistics method proposed by Tibshirani et al. (2001). The idea of such method is to compare two graphs generated from candidate models. However, the method is inefficient for large dataset. In a different context (model simplification), Garcia and Nielsen (2010) and Garcia et al. (2010) employed the KL Divergence based threshold to determine the optimal model. Note that, the KL Divergence is also considered as the fundamental basis for model selection criteria. See Chapter 2.2 and 7.2 of Burnham and Anderson (2002) for the derivation of Akaike Information Criterion (AIC) from the KL information.

The non-parametric Bayesian approach based on Dirichlet Process Mixture Model (DPMM) is currently one of most active approach to automatically determining the number of components (Murphy, 2012) in the context of mixture model. Such methods assume no apriori bound on the number of components and hence allows the number of clusters to grow with the increased amount of data. We refer readers to Chapter 25 of Murphy (2012) for the details of this approach. The drawbacks of this method are that they are non-deterministic and computationally very expensive. Another approach for automatic component selection is based on sampling with Reversible Jump Markov Chain Monte Carlo (RJMCMC) (Kato, 2008). Such sampler is able to explore the parameter subspaces of different dimensionality and hence can be used to find the most likely number of classes. However, it requires high computation time due to involving a large amount of sampling. In this thesis, we do not further explore these methods due to their inefficiency to cluster large amount of data, e.g., $\approx$300k for an image.

Initialization is considered as one of the most prominent issues to be addressed in the Expectation Maximization based methodology (Martinez et al., 2010). A Variety of different strategies exists for initializing the EM algorithm, see Biernacki et al. (2003), Figueiredo and Jain (2002) and McLachlan and Peel (2004) for different choices. However, no single method uniformly outperforms the other from all aspects, such as sensitivity to local minima, Maximum Likelihood function value, speed of convergence or computation time, stability etc. Therefore, it is necessary to experimentally evaluate different strategies and select the suitable one depending on data and probability distribution.

One of the most common forms of initializing Expectation Maximization (EM) method is through random initialization (Biernacki et al., 2003; McLachlan and Peel, 2004). The idea of this initialization consists of drawing one or more random positions, and then computes the mean of these positions. However, it appears from experiments that random initialization can often lead to a suboptimal solution by getting trapped into one of the many local maxima of the Maximum Likelihood function. Experimental evaluation by Biernacki et al. (2003) shows that, algorithms such as short runs of EM (1emEM or $x$emEM), classification EM (CEM), stochastic EM (SEM) outperforms the random initialization. These techniques are less sensitive to noisy data and often they cause faster convergence of the core EM algorithm.

Another approach of initialization considers starting the first Expectation step with an initial partition (McGraw et al., 2006). This initial partition is obtained by clustering algorithms such as widely used k-means type algorithm or hierarchical algorithm. Clustering with model based approach (Fraley and Raftery, 2002) belongs to such family. However, it has several drawbacks (Melnykov and Maitra, 2010), such as it works well only for well separated clusters and it has limited applicability to large datasets. The k-means algorithm is considered as a variant of the Expectation Maximization (EM) by imposing restrictive assumptions of certain parameters of the distribution associated with the mixture model. Therefore, speed of convergence for k-means will be faster than EM. This provides reasonable motivation to choose k-means (and its variants) as an initialization tool for the EM algorithm. However, k-means itself needs initialization and common procedure is to choose k data points at random (Murphy, 2012). Therefore, k-means based EM initialization have the same drawbacks of random initialization. The k-means++ (Arthur and Vassilvitskii, 2007) algorithm appears as very promising to tackle the problems by choosing the starting centers with specific probabilities, see Section 2.6.1 for details.

2.3 Background

Clustering or cluster analysis can be defined as the unsupervised classification of patterns (observations, data items, or feature vectors) into groups (clusters) (Jain et al., 1999). It is considered as one of the oldest techniques for exploratory data analysis and data mining. Figure 2.1 illustrates an example of data for clustering and its true labels that we will use throughout this Chapter. A number of different clustering techniques are available in literature (Murphy, 2012; Martinez et al., 2010). See Jain et al. (1999) for a taxonomy of the common clustering approaches. Among

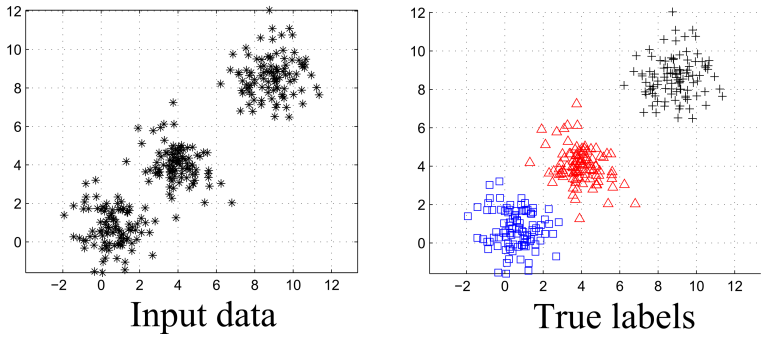


Figure 2.1: Example of 2-dimensional data for clustering and its true labels.

them, the partitional and hierarchical clustering methods are most relevant with this thesis work.

The partitional clustering technique creates groups of the data into disjoint sets. This grouping can be hard, such as k-means, which assigns each observation into one of the groups. In contrast, it can be fuzzy or probabilistic, such as fuzzy logic approaches as Fuzzy C-Means (FCM) (Jain et al., 1999) or Expectation Maximization (Bishop, 2006) for statistical mixture models. In the fuzzy or probabilistic approaches, each data point has a certain degree of membership or probability to be a member of each of the groups or clusters. The hierarchical clustering (Martinez et al., 2010) creates a nested tree of partitions. Below we discuss the relevant clustering techniques which are essential part of our proposed clustering method.

Let us consider a set of observations as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1,...,N}$, where $\mathbf{x}_i \in \mathbb{R}^d$ denotes a single d dimensional sample and N is the total number of samples. The goal of clustering is to partition $\mathbf{X}$ into k clusters and automatically identify the labels $\Gamma = \{\gamma_i\}_{i=1,...,N}$, where $\gamma_i \in \{1, ..., k\}$ denotes the label of sample $\mathbf{x}_i$.

2.3.1 Hierarchical Clustering

Hierarchical clustering methods produce clusters of observations which can be considered as a hierarchy of groups or set of nested partition (Murphy, 2012). There are two main categories of this type of methods: agglomerative or bottom-up and divisive or top-down. Both categories build a dissimilarity matrix from each pair of the observations and perform clustering based on it. The agglomerative method proceeds by merging similar observations or subsets of observations at each step until having a single set containing all observations. Whereas, the divisive method starts from the entire set of observations and recursively splits it to subsets until having subsets

with single observation. We consider only the agglomerative method which follows two steps as:

- Step 1: Start with N subsets each containing a single observation.
- Step 2: Merge two most similar subsets and continues until there is a single set.

The initial observations and their progressively merged subsets information are stored into a hierarchical structure called dendogram (Martinez et al., 2010), see Figure 2.5 for an example.

Two most important issues of the hierarchical clustering are the distance among single observations and the measure of distance between pair of subsets which contains more than one observation. Computing distance among single observations depends on the type of data, for example the Euclidean distance is used for continuous data types belonging to the Euclidean space. The measure of distance between pair of subsets is called the linkage criteria. Different choices exist as the linkage criteria, such as: Single, Complete, Average, Ward, Weighted, Median and Centroid. See Martinez et al. (2010) for details. Among them, the Single, Complete and Average linkage criteria are most commonly used (Murphy, 2012). The Single linkage is also called the nearest neighbor clustering, it measures the distance among two closest members of each group. The Complete linkage is also called the farthest neighbor clustering, it measures the distance among two most distant pairs. The Average linkage measures the average distance between all pairs. It should be noted that there is no recommended distance type and linkage criteria (Martinez et al., 2010). Therefore, the analyst should find the appropriate one to explore the data. The cophenetic correlation coefficient (Martinez et al., 2010), which provides a way to compare a set of nested partitions from hierarchical clustering, can be employed for the purpose of evaluating different criteria and select the appropriate one.

2.3.2 k-means

k-means is one of the most popular, simple and widely used data clustering techniques. It is a partitional clustering method that provides hard clustering of the data. The basic idea of this method is based on the objective to minimize intra-cluster distance and maximize inter-cluster distance. This idea is formalized by discovering the parameters $\Theta_k = \{\mu_j\}_{j=1,\dots,k} \in \mathbb{R}^d$ and the labels Γ, such that the following function is minimized:

$$\sum_{i=1}^{N} \sum_{j=1}^{k} \mathbf{1}\left[\gamma_i = j\right] \left\|\mathbf{x}_i - \mu_j\right\|_2^2 \tag{2.1}$$

Eq. (2.1) is the objective function for the k-means clustering method, where μ_j is called the mean or centroid of each cluster $j = 1, ..., k$, $\mathbf{1}[.]$ is an indicator function for the associated condition and $\|.\|_2$ is the L_2 norm or the Euclidean distance. In order to cluster with the k-means method, the objective in Eq. (2.1) is iteratively evaluated until certain convergence criteria are satisfied. Each iteration consists of assigning the labels γ_i and updating the parameters μ_j as follows:

$$\gamma_i = \underset{j=1,...,k}{\arg\min} \|\mathbf{x}_i - \mu_j\|_2^2 \ , \ i = 1, ..., N \tag{2.2}$$

$$\mu_j = \frac{\sum_{i=1}^{N} \mathbf{1}[\gamma_i = j]\, \mathbf{x}_i}{\sum_{i=1}^{N} \mathbf{1}[\gamma_i = j]} \tag{2.3}$$

The k-means method starts by setting initial values for the parameters, i.e. Θ_k. Most commonly, these parameters are set randomly. However, random initialization often generates a sub-optimal solution as it cannot guarantee to converge into the global minimum. The convergence criteria applied in this method consists of setting a maximum number of iterations and a threshold related to the minimum difference in the objective function (Eq. (2.1)) value in two consecutive steps. One of the concerns about k-means is its spherical assumption about the structure of the clusters (Bishop, 2006). This can be solved with the use of mixture model based method, such as GMM.

2.3.3 Finite Mixture Models

Clustering with finite mixture models, also called latent variable models (Murphy, 2012; Bishop, 2006), is a partitional approach that provides probabilistic clustering. Moreover, they provide better interpretability of the clusters structure by modeling data with the parameters associated with the probability distributions. The most popular model is the Gaussian Mixture Model (GMM), which models data with the mean and covariance of the Gaussian distribution for each cluster (Bishop, 2006). A mixture model of k Gaussian distributions is written as:

$$g(\mathbf{x}_i|\Theta_k) = \sum_{j=1}^{k} \pi_{j,k} f_g(\mathbf{x}_i|\mu_{j,k}, \Sigma_{j,k}) \tag{2.4}$$

where, $\Theta_k = \{(\pi_{1,k}, \mu_{1,k}, \Sigma_{1,k}), ..., (\pi_{k,k}, \mu_{k,k}, \Sigma_{k,k})\}$ is the set of model parameters and $\pi_{j,k}$ is the mixing proportion with $\sum_{j=1}^{k} \pi_{j,k} = 1$. $f_g(\mathbf{x}_i|\mu_{j,k}, \Sigma_{j,k})$ is the multivariate Gaussian distribution for cluster j, which is defined as:

$$f_g(\mathbf{x}_i|\mu_j, \Sigma_j) = \frac{1}{(2\pi)^{d/2} \det(\Sigma_j)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}_i - \mu_j)^T \Sigma_j^{-1} (\mathbf{x}_i - \mu_j)\right) \tag{2.5}$$

where, $\mu_j \in \mathbb{R}^d$ is the mean and Σ_j is the variance-covariance symmetric positive-definite matrix.

Clustering with a mixture model requires the estimation of the model parameters Θ_k as well as the latent variables Γ of the data. Most commonly, this is accomplished by finding the Maximum Likelihood Estimation (MLE) using the Expectation Maximization method, also called EM method. See Chapter 9 of Bishop (2006) for details of the EM method.

Maximum Likelihood Estimation using the EM method consists of Initialization, E-step, M-step and log likelihood evaluation. Initialization is applied only once at the beginning of the method in order to set the initial values for the model parameters Θ_k. It can be done in several ways, such as randomly or using k-means algorithm. After initializing, the log likelihood value of the model parameters is computed as:

$$\log g\left(\mathbf{X}|\Theta_k\right) = \sum_{i=1}^{N} \log \left\{ \sum_{j=1}^{k} \pi_{j,k} f_g\left(\mathbf{x}_i|\mu_{j,k}, \Sigma_{j,k}\right) \right\} \tag{2.6}$$

The E step computes the posterior probability, also called responsibility of the current parameter values as:

$$p_{ij} = p\left(\gamma_i = j|\mathbf{x}_i\right) = \frac{\pi_{j,k} f_g\left(\mathbf{x}_i|\mu_{j,k}, \Sigma_{j,k}\right)}{\sum_{l=1}^{k} \pi_{l,k} f_g\left(\mathbf{x}_i|\mu_{l,k}, \Sigma_{l,k}\right)} \tag{2.7}$$

The M step (for GMM) performs an update or re-estimation of the current parameter values as:

$$\pi_{j,k} = \frac{1}{N}\sum_{i=1}^{N} p_{ij} \text{ and } \mu_{j,k} = \frac{\sum_{i=1}^{N} p_{ij}\mathbf{x}_i}{\sum_{i=1}^{N} p_{ij}} \text{ and } \Sigma_{j,k} = \frac{\sum_{i=1}^{N} p_{ij}(\mathbf{x}_i - \mu_j)(\mathbf{x}_i - \mu_j)^T}{\sum_{i=1}^{N} p_{ij}} \tag{2.8}$$

Then, the log likelihood value is computed with Eq. (2.6). The EM method is an iterative procedure, which employs the E and M steps iteratively until certain convergence criteria are satisfied. Such criteria consist of setting a maximum number of iterations and a threshold related to the minimum difference in the likelihood function (Eq. 2.6) value of two consecutive steps.

The Gaussian distribution is commonly used for finite mixture model. However, it is interesting to have a mixture model framework that generalizes a group of distributions. The exponential family of distributions is a broad class which consists of many important probability distributions (Murphy, 2012), which can be considered for a generalized mixture model framework.

2.3.4 Exponential Family of Distributions (EFD)

A multivariate probability density function $f(\mathbf{x}|\theta)$ belongs to the exponential family if it has the following canonical form (Murphy, 2012; Banerjee et al., 2005b):

$$f(\mathbf{x}|\theta) = \exp\left(\langle t(\mathbf{x}), \theta\rangle - F(\theta) + k(\mathbf{x})\right) \tag{2.9}$$

Here,

- $t(\mathbf{x})$ denotes the sufficient statistics[1];
- θ denotes the natural parameter[1];
- F is the log normalizing function[1], which is strictly convex and differentiable;
- $k(\mathbf{x})$ is the carrier measure[1];
- $< ., . >$ is the inner product.

The expectation of the sufficient statistics $t(\mathbf{x})$ is called the expectation parameter, $\eta = E[t(\mathbf{x})]$. There exists a one-to-one correspondence between expectation (η) and natural (θ) parameters, which exhibits dual relationships among the parameters and functions as (Banerjee et al., 2005b):

$$\eta = \nabla F(\theta) \quad \text{and} \quad \theta = (\nabla F)^{-1}(\eta) \tag{2.10}$$

and

$$G(\eta) = \left\langle (\nabla F)^{-1}(\eta), \eta \right\rangle - F\left((\nabla F)^{-1}(\eta)\right) \tag{2.11}$$

Here, ∇F is the gradient of F. G is the Legendre dual of the log normalizing function F. See Section 3.2 of Banerjee et al. (2005b) for details.

The exponential family encompasses a wide class of familiar distributions (Nielsen and Garcia, 2009), which includes Gaussian or normal, Gamma, Beta, Laplacian, Exponential, Wishart, Rayleigh, Weibull, Dirichlet, Poisson, Bernoulli, Binomial, Multinomial, etc. We refer reader to Chapter 9 of Murphy (2012) to study the important properties of exponential families and Nielsen and Garcia (2009) for the canonical form of a number of probability distributions.

To provide an example, let us consider the Gaussian distribution (Eq. 2.5), which has the following canonical representation (based on Eq. 2.9) (Garcia and Nielsen, 2010):

[1] see the definitions given later for different probability distributions.

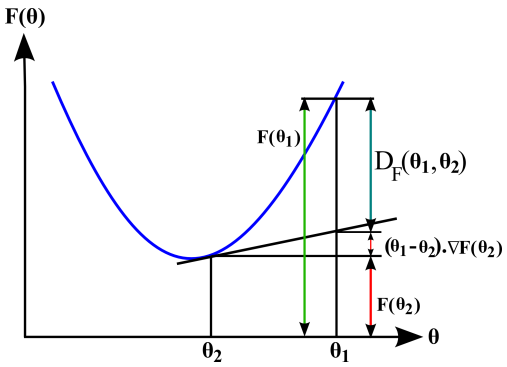


Figure 2.2: Illustration of Bregman divergence.

- sufficient statistics: $t(\mathbf{x}) = (\mathbf{x}, -\mathbf{x}\mathbf{x}^T)$;
- carrier measure $k(\mathbf{x}) = 0$;
- natural parameter $\theta = (\psi, \Psi) = \left(\Sigma^{-1}\mu, \frac{1}{2}\Sigma^{-1}\right)$;
- expectation parameter $\eta = (\phi, \Phi) = \left(\mu, -(\Sigma + \mu\mu^T)\right)$;
- log normalizing function $F(\theta) = \frac{1}{4}\text{tr}\left(\Psi^{-1}\psi\psi^T\right) - \frac{1}{2}\log\det\Psi + \frac{d}{2}\log\pi$ and
- dual log normalizing function $G(\eta) = -\frac{1}{2}\log(1 + \phi^T\Phi^{-1}\phi) - \frac{1}{2}\log(\det(\Phi)) - \frac{d}{2}\log(2\pi e)$.

Banerjee et al. (2005b) developed efficient clustering method for the mixture of exponential families. Their method exploits the relationship between exponential families and Bregman divergence.

2.3.5 Bregman Divergence (BD)

For a strictly convex function F, Bregman divergence, $D_F(\theta_1, \theta_2)$ can be formally defined as (Banerjee et al., 2005b):

$$D_F(\theta_1, \theta_2) = F(\theta_1) - F(\theta_2) - \langle \theta_1 - \theta_2, \nabla F(\theta_2) \rangle \tag{2.12}$$

$D_F(\theta_1, \theta_2)$ measures the error using the tangent function at θ_2 to approximate F. This can be seen as the distance between the first order Taylor approximation to F at θ_2 and the function evaluated at θ_1 (Liu et al., 2012). Figure 2.2 illustrates an example of computing Bregman divergence using Eq. (2.12).

The one-to-one correspondence in Eq. (2.10) provides the dual form of BD (of Eq. (2.12)) as:

$$D_G(\eta_1, \eta_2) = G(\eta_1) - G(\eta_2) - \langle \eta_1 - \eta_2, \nabla G(\eta_2) \rangle \tag{2.13}$$

Due to the bijection[2] between BD and the exponential families, Eq. (2.12) and (2.13) can be used to measure the dissimilarity between distributions of the same exponential family.

Bregman Divergences (BD) generalize the squared Euclidean distance, Mahalanobis distance, Kullback-Leibler divergence, Itakura-Saito divergence etc. See Table 1 of Banerjee et al. (2005b) and Boissonnat et al. (2010) for a list and corresponding F and $D_F(.,.)$. Besides, BD has the following interesting properties (Boissonnat et al., 2010):

- Non-negativity: The strict convexity of F implies that, for any θ_1 and θ_2, $D_F(\theta_1, \theta_2) \geq 0$ and $D_F(\theta_1, \theta_2) = 0$ if and only if $\theta_1 = \theta_2$.
- Convexity: Function $D_F(\theta_1, \theta_2)$ is convex in its first argument θ_1 but not necessarily in the second argument θ_2.
- Linearity: BD is a linear operator, i.e., for any two strictly convex functions F_1 and F_2 and $\lambda \geq 0$:

$$D_{F1+\lambda F2}(\theta_1, \theta_2) = D_{F1}(\theta_1, \theta_2) + \lambda\, D_{F2}(\theta_1, \theta_2)$$

Now, let us consider an example of computing Bregman divergence among two multivariate Gaussian distributions. To this aim, we can use Eq. (2.12) or (2.13) based on the type of parameters derived in Section 2.3.4. However, we can notice that the multivariate Gaussian distribution consists of mixed type vector/matrix parameters. For this reason, the inner product $< .,. >$ in Eq. (2.12) or (2.13) is a composite inner product obtained as a sum of two inner products of vectors and matrices as (Garcia and Nielsen, 2010):

$$\langle \theta_1, \theta_2 \rangle = \langle \Psi_1, \Psi_2 \rangle + \langle \psi_1, \psi_2 \rangle \tag{2.14}$$

where, the inner product of vectors is the dot product $\langle \psi_1, \psi_2 \rangle = \psi_1^T \psi_2$, and the inner product of two matrices is defined as:

$$\langle \Psi_1, \Psi_2 \rangle = \mathrm{tr}\left(\Psi_1\, \Psi_2^T\right) = \mathrm{tr}\left(\Psi_2\, \Psi_1^T\right)$$

The formulations presented in Sections 2.3.4 and 2.3.5 along with the properties of Bregman divergence and the exponential families allow us to develop a generalized clustering method (see Section 2.6 and Figure 2.9) which can be incorporated with any mixture of exponential family of distributions.

[2]The bijection is expressed as: $f(\mathbf{x}|\theta) = \exp(-D_G(t(\mathbf{x}), \eta))J_G(\mathbf{x})$ where J_G is a uniquely determined function. For more details, please see Theorem 3 of Banerjee et al. (2005b).

2.4 Hierarchy of Mixture Models

We assume a generative model (Murphy, 2012), which consists of a mixture of k distributions belonging to the exponential families as:

$$g\left(\mathbf{x}_i|\Theta_k\right) = \sum_{j=1}^{k} \pi_{j,k} f\left(\mathbf{x}_i|\theta_{j,k}\right) \tag{2.15}$$

Here $\Theta_k = \{(\pi_{1,k}, \theta_{1,k}), ..., (\pi_{k,k}, \theta_{k,k})\}$ is the set of component parameters, $\pi_{j,k}$ is the mixing proportion and $f\left(\mathbf{x}_i|\theta_{j,k}\right)$ is the distribution for j^{th} component.

We apply the Hierarchical Agglomerative Clustering (HAC) on the mixture model parameters Θ_k to construct a set of models. In general, the HAC permits a variety of choices based on three principal issues (Martinez et al., 2010):

a. the distance measure between clusters,

b. the criterion to select the clusters to be merged and

c. the representation of the merged cluster.

The first issue can be solved by measuring the distance between two exponential families distributions using the Bregman Divergence (BD) of Eq. (2.12) or (2.13). Since BD is generally an asymmetric measure (Garcia and Nielsen, 2010), we have three choices for distance measure:

Left-sided:

$$d_l\left((\pi_1, \theta_1), (\pi_2, \theta_2)\right) = \pi_1\, \pi_2\, D_F\left(\theta_1, \theta_2\right)$$

or

$$d_l\left((\pi_1, \eta_1), (\pi_2, \eta_2)\right) = \pi_1\, \pi_2\, D_G\left(\eta_1, \eta_2\right)$$

Right-sided:

$$d_r\left((\pi_1, \theta_1), (\pi_2, \theta_2)\right) = \pi_1\, \pi_2\, D_F\left(\theta_2, \theta_1\right)$$

or

$$d_r\left((\pi_1, \eta_1), (\pi_2, \eta_2)\right) = \pi_1\, \pi_2\, D_G\left(\eta_2, \eta_1\right)$$

Symmetric:

$$d_s\left((\pi_1, \theta_1), (\pi_2, \theta_2)\right) = \frac{\pi_1\, \pi_2\, \left(D_F\left(\theta_1, \theta_2\right) + D_F\left(\theta_2, \theta_1\right)\right)}{2}$$

or

$$d_s\left((\pi_1, \eta_1), (\pi_2, \eta_2)\right) = \frac{\pi_1\, \pi_2\, \left(D_G\left(\eta_1, \eta_2\right) + D_G\left(\eta_2, \eta_1\right)\right)}{2}$$

To deal with the second issue (issue (b)), we choose the "minimum BD" as merging criterion. The linkage criteria (single, complete, average, etc.) should be selected empirically.

In our clustering strategy, the set of models is represented by their parameters (also called cluster centroids). After determining the clusters to be merged, we compute their representative centroids (issue (c)). Similar to the distances, there are three types of centroids, called Bregman centroids. See Figure 1 of Garcia and Nielsen (2010) for an example with clear distinctions among different types of centroid, which are computed with the uni-variate Gaussian distributions. For a set of parameters $\{\theta_1, ..., \theta_M\}, M > 1$ with associated weights $\{\pi_1, ..., \pi_M\}$, different types of Bregman centroid (with both natural and expectation parameters) can be computed as:

Left-sided centroid:

$$\theta_L = \nabla F^{-1}\left(\frac{\sum_{i=1}^{M} \pi_i \nabla F(\theta_i)}{\sum_{i=1}^{M} \pi_i}\right)$$

or

$$\eta_L = \frac{\sum_{i=1}^{M} \pi_i \eta_i}{\sum_{i=1}^{M} \pi_i}$$

Right-sided centroid:

$$\theta_R = \frac{\sum_{i=1}^{M} \pi_i \theta_i}{\sum_{i=1}^{M} \pi_i}$$

or

$$\eta_R = \nabla F\left(\frac{\sum_{i=1}^{M} \pi_i \theta_i}{\sum_{i=1}^{M} \pi_i}\right)$$

Symmetric centroid:

$$\theta_S = \nabla F^{-1}\left(\lambda \nabla F(\theta_R) + (1-\lambda) \nabla F(\theta_L)\right)$$

or

$$\eta_S = \nabla F(\theta_S) \text{ and } \theta_S = \nabla F^{-1}\left(\lambda \eta_R + (1-\lambda) \eta_L\right)$$

with $\lambda \in [0, 1]$ (λ is obtained by using a standard bisection search).

Note that, the type of centroid used to merge/fuse clusters parameters, must correspond to the type of distance. The appropriate type of distance (issue (a)) and centroid (issue (c)) should be selected empirically. Figure 2.3 illustrates an example of merging clusters with left-sided Bregman centroid.

Now, let us consider an example of applying the hierarchical mixture models method with a multivariate Gaussian Mixture Model (GMM). Figure 2.4 illustrates

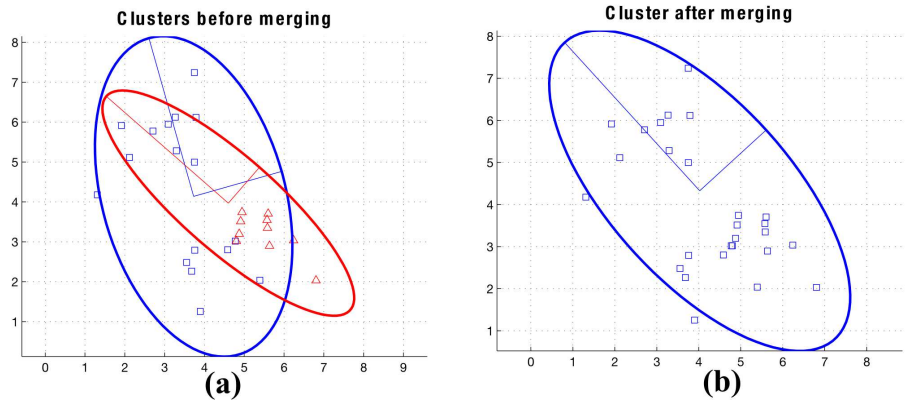


Figure 2.3: Example of merging clusters with left-sided Bregman centroid. (a) two clusters, 1 - blue colored with parameters: $\pi_1 = 0.0934$, $\mu_1 = [3.7298; 4.1386]$, $\Sigma_1 = [0.6836 \; -0.3418; -0.3418 \; 1.7928]$ and 2 - red colored with parameters: $\pi_2 = 0.0676$, $\mu_2 = [4.6003; 3.9701]$, $\Sigma_2 = [1.1124 \; -0.8339; -0.8339 \; 0.8858]$ (b) two clusters merged into a single cluster with parameters: $\pi_m = 0.1610$, $\mu_m = [4.0224; 4.3286]$, $\Sigma_m = [0.8037 \; -0.5661; -0.5661 \; 0.9066]$, where the sub-script "m" denotes the merged cluster.

an example of a hierarchy of GMMs for $k = 9, ..., 2$ number of classes. The GMMs samples correspond to the data shown in Figure 2.1. Notice that, we compute the parameters from data, only for the model with $k_{max} = 9$ components. Then, we use these parameters in the proposed HAC method to compute the parameters for the models with $k = 8, ..., 2$ components. The hierarchical structure of the merged information can be represented by a dendogram which is shown in Figure 2.5.

The set of mixture models generated by the hierarchical agglomerative clustering method can be considered as the candidate models for the model based clustering method. Next, we apply a model selection method to select the optimal model.

2.5 Model Selection

Let us consider that after applying Hierarchical Agglomerative Clustering (HAC), we have a set of mixture models which consists of $k_{max}, ..., 1$ components. The problem of finding an optimal model can be described as the selection of the mixture model with k_o components such that $\Theta_{k_o} = \{(\pi_{1,k_o}, \theta_{1,k_o}), ..., (\pi_{k_o,k_o}, \theta_{k_o,k_o})\}$. Next, we will present different methods for model selection.

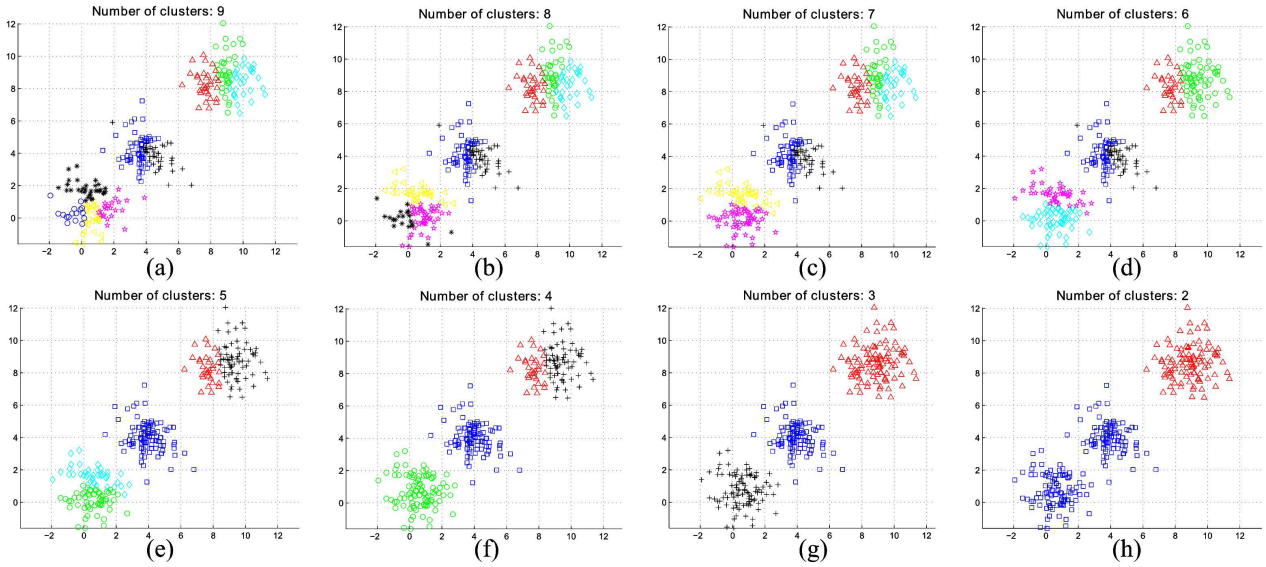


Figure 2.4: Example of a hierarchy of mixture models; generated using the data shown in Figure 2.1. From (a) to (h) the number of components reduces from 9 to 2.

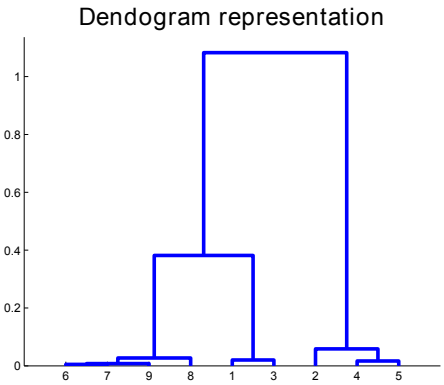


Figure 2.5: Dendogram for constructing the mixture models shown in Figure 2.4.

2.5.1 Parsimony based approach

In this approach, an objective function is employed, which minimizes certain model selection criteria (Figueiredo and Jain, 2002) (also called information criteria (IC)). Many of these criteria involve a negative log likelihood augmented by a penalizing function in order to take into account the complexity of the model. We consider the following form to compute the IC value for a model with k components (Alata and Quintard, 2009):

$$IC(k) = -2\log\left(g(\mathbf{X}|\hat{\Theta}_k)\right) + C(N)P(k) \tag{2.16}$$

with

$$g(\mathbf{X}|\hat{\Theta}_k) = \prod_{i=1}^{N} g(\mathbf{x}_i|\hat{\Theta}_k) \tag{2.17}$$

Here, $g(\mathbf{X}|\hat{\Theta}_k)$ denotes the maximum likelihood value of the data samples $\mathbf{X}$. $\hat{\Theta}_k = \left\{(\hat{\pi}_{1,k}, \hat{\theta}_{1,k}), ...(\hat{\pi}_{k,k}, \hat{\theta}_{k,k})\right\}$ are the parameters that maximize the likelihood value. $C(N)$ denotes the penalization of model complexity depending on the number of observations N and $P(k)$ denotes the number of free parameters. For example, $P(k)$ for the GMM is:

$$P(k) = \alpha k - 1 \text{ with } \alpha = \left(d + \frac{d(d+1))}{2} + 1\right) \tag{2.18}$$

Different information criteria use different values of $C(N)$. Akaike Information Criterion (AIC) uses $C(N) = 3$. Bayesian Information Criterion (BIC) uses $C(N) = \log(N)$. The Integrated Completed Likelihood (ICL) is computed by adding BIC with the estimated mean entropy (Biernacki et al., 2000) as:

$$ICL(k) = -2\log\left(g(\mathbf{X}|\hat{\Theta}_k)\right) + \log(N)P(k) - 2\sum_{i=1}^{N}\log\left(p(\gamma_i|\mathbf{x}_i)\right) \tag{2.19}$$

Here, $p(\gamma_i|\mathbf{x}_i)$ denotes the conditional probability of the classified class label $\gamma_i \in \{1, ..., k\}$ for the sample $\mathbf{x}_i$.

Beside these, we can also adopt the Φ_β criterion (Alata and Quintard, 2009), that computes $C(N)$ with different β values ($0 < \beta < 1$ for having a consistent estimator). In the general form, $C(N)$ for computing Φ_β criterion is:

$$C(N) = N^\beta \log(\log(N)) \tag{2.20}$$

The motivation for choosing this criterion is that, different β values allow us to compute different criteria. For example, several choices of β values in Eq. (2.20) are:

$$\beta_{AIC} = \frac{\log 3 - \log\log\log N}{\log N} \quad (\beta \text{ for Akaike IC})$$

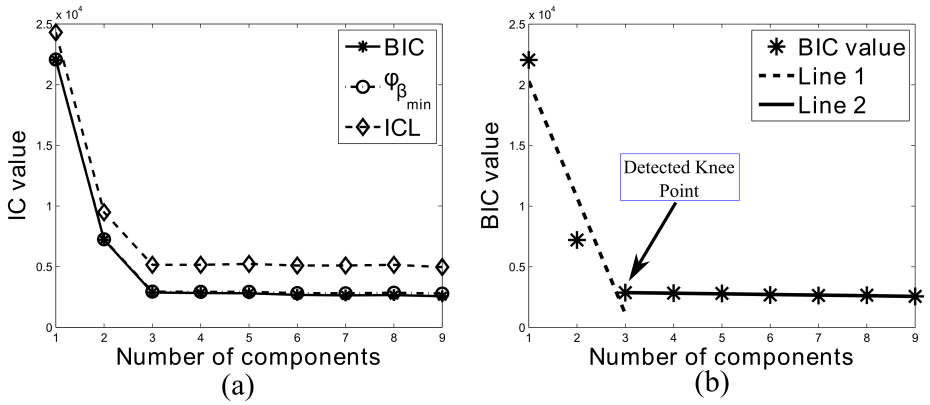


Figure 2.6: Model selection approaches based on the mixture models shown in Figure 2.4. (a) Parsimony based approach based on different model selection criteria and (b) Plot/Evaluation graph based approach.

$$\beta_{BIC} = \frac{\log\log N - \log\log\log N}{\log N} \quad (\beta \text{ for Bayesian IC})$$

$$\beta_{min} = \frac{\log\log N}{\log N} \quad (\text{proposed bound for minimum } \beta \text{ value})$$

After computing the value of model selection criteria for different $k \in \{1, ..., k_{max}\}$ we use the following objective function to obtain k_o (optimal model):

$$k_o = \arg\min_k IC(k) \quad \text{or} \quad k_o = \arg\min_k ICL(k) \tag{2.21}$$

Figure 2.6(a) illustrates graphical examples of model evaluation values obtained by using different model selection criteria. Here, we use the data shown in Figure 2.1 and we consider the GMMs shown in Figure 2.4. One interesting behavior we observe from Figure 2.6(a) is that, the evaluation values changes linearly after 3, which is the correct number of components. The reason for such behavior is that, the likelihood value decreases slowly after $k = 3$ w.r.t. the part $C(N)P(N)$ which is a line with slope $C(N)\alpha$ (see Eq. (2.18)). Such linear changes can be characterized by fitting lines. A particular family of methods selects models based on this assumption. Next we discuss these methods.

2.5.2 Plot/Graph based approach

A different strategy selects optimal number of components by analyzing a plot/evaluation graph (Baudry et al., 2010; Zhao et al., 2008; Salvador and Chan, 2004). This graph is usually obtained by placing numbers of clusters along the x axis and corresponding evaluated values (obtained using a model selection criteria) along the y axis. The

idea is to locate the knee/kink/elbow/transition area in the graph, where the knee exhibits abrupt change (Murphy, 2012). Then the k_o will be the value of the knee. Figure 2.6(b) illustrates an example of these graphs and the detected knee point.

One common graph based approach is called the L-method (Salvador and Chan, 2004). It detects the knee point by fitting a pair of straight lines over the y axis values of the graph. The idea is to fit two lines at the left and right side of each point (within the range 2,...,$k_{max}-1$). Finally, select the point as k_o that minimizes the total weighted root mean squared error (RMSE):

$$k_o = \arg\min_k \left(\omega_l \, RMSE_{k,left} + \omega_r \, RMSE_{k,right}\right) \tag{2.22}$$

$$\omega_l = \frac{k-1}{k_{max}-1} \text{ and } \omega_r = \frac{k_{max}-k}{k_{max}-1}$$

Note that, two weights (ω_l and ω_r) are associated with each line (left and right). These weights are computed from the ratio of the number of points in a line over the total number of points. These weights have significant impact on model selection. Particularly, it is interesting to characterize the linear change shown in the right sided line, see Figure 2.6(b). This can be done by setting higher weight for ω_r compare to ω_l, such that $\omega_l \leq \omega_r$. Setting such weight means that, in order to respect the linear change of the right sided line, the evaluation plot based methods will penalize more on the line fitting error at the right side. Figure 2.7 shows such an example of setting different weights for ω_r while keeping $\omega_l = 1$ fixed, where we used the same BIC plot shown in Figure 2.6(b). In practice, the weight ω_r should be set empirically. Let us call this model selection method the Weighted Piecewise Linear Regression (WPLR) method for further references. We will use and discuss about *WPLR* in Section 2.6.3 of this Chapter and also in the following Chapter. Now, we discuss a different model selection approach based on Kullback Leibler Divergence (Burnham and Anderson, 2002).

2.5.3 Kullback Leibler Divergence (KLD) based approach

Kullback Leibler Divergence (KLD) is one of the fundamental measure (relative entropy) between two statistical distributions (Hershey and Olsen, 2007; Burnham and Anderson, 2002). It is an oriented distance (asymmetric), which is often used as a measure of similarity (Hershey and Olsen, 2007). In the KLD based model selection approach (Garcia and Nielsen, 2010), a model is selected based on a threshold (KLD value) among $\hat{\Theta}_{k_{max}}$ (mixture model with k_{max} components) and $\hat{\Theta}_k$ (mixture model with $k \in \{k_{min}, ..., k_{max}-1\}$ components). No closed-form solution exists to compute

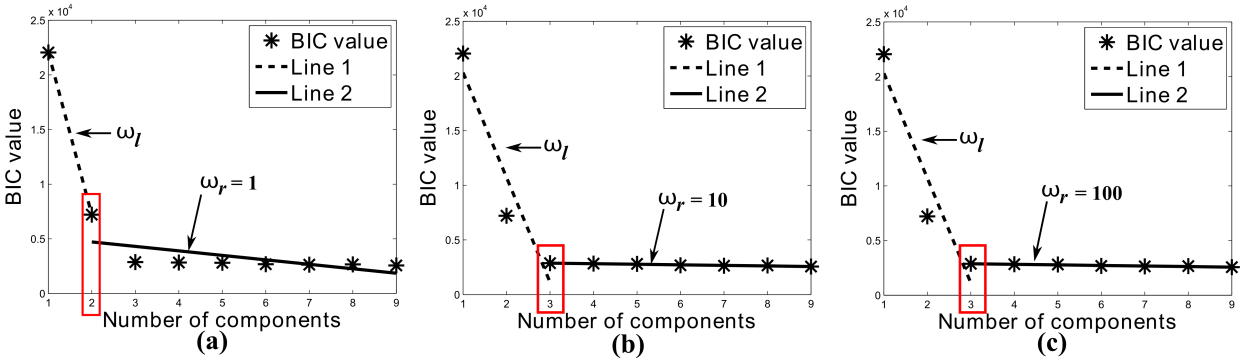


Figure 2.7: Examples of setting different weights for ω_r while keeping $\omega_l = 1$. The selected number.

KLD among mixture models. However, it can be approximated by employing classical Monte-Carlo sampling among two mixture models in the following form (Hershey and Olsen, 2007):

$$D_{KL}\left(\hat{\Theta}_{k_{max}} \,||\, \hat{\Theta}_k\right) = \frac{1}{M}\sum_{i=1}^{M} \log\left(\frac{g(\mathbf{x}_i|\hat{\Theta}_{k_{max}})}{g(\mathbf{x}_i|\hat{\Theta}_k)}\right) \tag{2.23}$$

Here, M is the number of identically and independently distributed samples obtained using a sampling procedure for the mixture model with k_{max} components. Using Eq. (2.23), the KLD values can be computed for different values of $k \in \{k_{min}, ..., k_{max}-1\}$ and then the desired model k_o can be obtained as:

$$k_o = \arg\min_k D_{KL}\left(\hat{\Theta}_{k_{max}} \,||\, \hat{\Theta}_k\right) < threshold \tag{2.24}$$

Note that, the *threshold* is defined externally by the user. This indicates that, to obtain desired clustering results with this model selection approach, the user should have sufficient knowledge about the data and experience of correct threshold selection. Figure 2.8 illustrates an example of employing the KLD based approach for model selection with a threshold value 2. In this example, we use the data shown in Figure 2.1 and we consider the GMMs shown in Figure 2.4. Garcia and Nielsen (2010) employed this approach for selecting the optimal mixture model.

Considering all the elements presented in this section and the previous one thereafter we propose a complete clustering method.

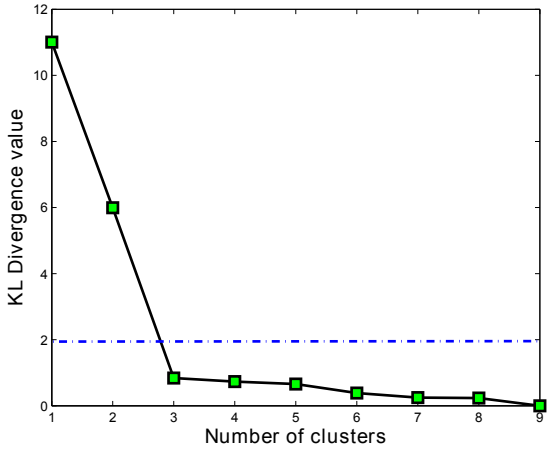


Figure 2.8: Examples of KLD threshold (shown in the blue dotted line) based approach for model selection.

2.6 Model based clustering with exponential family mixture model

We consider a deterministic (Figueiredo and Jain, 2002) Model Based Clustering (MBC) approach where the number of models is bounded within a certain range $k_{min}, ..., k_{max}$. Let $\Theta_k = \{(\pi_{1,k}, \theta_{1,k}), ..., (\pi_{k,k}, \theta_{k,k})\}$ denotes the exponential family mixture model with k components. Therefore, $\Theta_{k_{max}}$ denotes the mixture model with k_{max} components and Θ_{k_o} denotes the optimal mixture model with k_o components. To cluster a set of observations, we propose a complete data clustering method that follows a step-by-step procedure as:

- *Step 1:* Compute $\hat{\Theta}_{k_{max}}$ and perform soft clustering.
- *Step 2:* Generate a set of models $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$ from $\hat{\Theta}_{k_{max}}$.
- *Step 3:* Select the optimal model $\hat{\Theta}_{k_o}$ from $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$.

Figure 2.9 illustrates the block diagram of the proposed method. It begins with applying Bregman soft clustering on the data in Step 1 (section 2.6.1). Then, it applies the Hierarchical Agglomerative Clustering (HAC) in Step 2 (section 2.6.2). Finally, it employs a model selection method on $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$ in Step 3 (section 2.6.3). Below, we briefly describe each method individually.

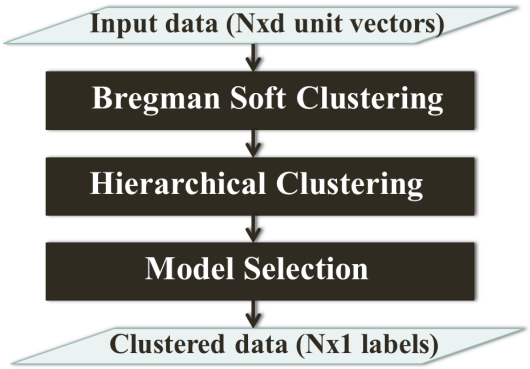


Figure 2.9: Block diagram of the proposed clustering method.

2.6.1 Bregman Soft Clustering (BSC)

Let us recall the exponential family mixture model introduced in section 2.4 and define it as $\Theta_k = \Theta_{k_{max}}$ (for brevity we write $k = k_{max}$). Our goal is to estimate the model (see Eq. (2.15)) with the objective to maximize the likelihood value such that:

$$\hat{\Theta}_k = \arg\max_{\Theta_k} g(\mathbf{X}|\Theta_k) \tag{2.25}$$

Bregman soft clustering exploits Bregman Divergence (BD) in the Expectation Maximization (EM) framework to compute Maximum Likelihood Estimate (MLE) of the parameters (Banerjee et al., 2005b). In the Expectation step (E-step), the posterior probability is computed for $j = 1, ..., k$ as:

$$p_{ij} = p\left(\gamma_i = j|\mathbf{x}_i\right) = \frac{\pi_{j,k} \exp\left(-D_G\left(t(\mathbf{x}_i), \eta_{j,k}\right)\right)}{\sum_{l=1}^{k} \pi_{l,k} \exp\left(-D_G\left(t(\mathbf{x}_i), \eta_{l,k}\right)\right)} \tag{2.26}$$

Here, $t(\mathbf{x}_i)$ denotes the expectation parameter for data sample $\mathbf{x}_i$. $\eta_{j,k}$ and $\eta_{l,k}$ denote the expectation parameters for any cluster j and l given that the total number of components is k. Note that, computing p_{ij} using $D_G(.,.)$ of Eq. (2.13) needs to compute $G(t(\mathbf{x}_i))$. However, such computation causes $G(t(\mathbf{x}_i)) = -\frac{1}{2}\log 0$ in the case of Gaussian distribution. Garcia and Nielsen (2010) provided a solution by factorizing and simplifying $G(t(\mathbf{x}_i))$ in both numerator and denominator. By adopting such solution, we can write Eq. (2.26) as:

$$p_{ij} = \frac{\pi_{j,k} \exp\left(G(\eta_{j,k}) + \langle t(\mathbf{x}_i) - \eta_{j,k}, \nabla G(\eta_{j,k})\rangle\right)}{\sum_{l=1}^{k} \pi_{l,k} \exp\left(G(\eta_{l,k}) + \langle t(\mathbf{x}_i) - \eta_{l,k}, \nabla G(\eta_{l,k})\rangle\right)} \tag{2.27}$$

The Maximization step (M-step) updates the mixing proportion and expectation parameter for each class as:

$$\pi_{j,k} = \frac{1}{N}\sum_{i=1}^{N} p\left(\gamma_i = j|\mathbf{x}_i\right) \text{ and } \eta_{j,k} = \frac{\sum_{i=1}^{N} p\left(\gamma_i = j|\mathbf{x}_i\right)\mathbf{x}_i}{\sum_{i=1}^{N} p\left(\gamma_i = j|\mathbf{x}_i\right)} \tag{2.28}$$

Initialization is a prominent issue and has significant impact on clustering. We initialize π and η of the mixture model using k-means++ (Arthur and Vassilvitskii, 2007) clustering. Let $\delta(\mathbf{x_i})$ defines the shortest distance from a data point $\mathbf{x_i}$ to the closest center we have already chosen. The k-means++ algorithm consists of the following steps:

1. Choose an initial center ξ_1 uniformly at random from $\mathbf{X}$.
2. Choose the next center ξ_j, selecting $\xi_j = \mathbf{x_i}' \in \mathbf{X}$ with probability $\frac{\delta(\mathbf{x_i}')^2}{\sum_{\mathbf{x_i} \in \mathbf{X}} \delta(\mathbf{x_i})^2}$.
3. Repeat Step 2 until we have chosen a total of k centers.
4. Proceed as with the standard k-means algorithm (see Section 2.3.2).

We choose k-means++ because of its: (a) careful seeding strategy; (b) ability to trade off among random selection and parameter search space and (c) faster convergence rate. However, one should empirically select the initialization strategy. After initialization, we iteratively apply the E-step and M-step until convergence.

The above procedures estimate the mixture model $\hat{\Theta}_k$ and provide soft clustering of the dataset. Let us call it BSC-MM algorithm (Algorithm 1). However, if a hard[3] clustering is desired, then it is easily obtained from BD as:

$$\hat{\gamma}_i = \underset{j=1,\ldots,k}{\arg\min}\ G(\eta_{j,k}) + \langle t(\mathbf{x}_i) - \eta_{j,k}, \nabla G(\eta_{j,k}) \rangle \tag{2.29}$$

Figure 2.10 illustrates an example of initialization with k-means++ (Arthur and Vassilvitskii, 2007) and clustering with BSC-MM algorithm. The BSC-MM is employed to cluster data (shown in Figure 2.1) into 9 classes. We set[4] maximum number of iterations to 20 and threshold 'log-likelihood difference among successive steps' to 0.01 as the convergence criteria. The convergence status of the proposed algorithm is illustrated in Figure 2.11. We observe that, the negative log likelihood values reduce at successive iterations, which confirms the convergence of the proposed algorithm (Algorithm 1).

[3] In hard clustering, each observation is assigned to a unique cluster.

[4] In practice, these settings depend on the requirements from clustering methods, such as speed of convergence, computation time, etc. For example, in MATLAB the default values of clustering with Gaussian mixture model are: maximum iteration = 100, threshold log likelihood difference = $1e-6$.

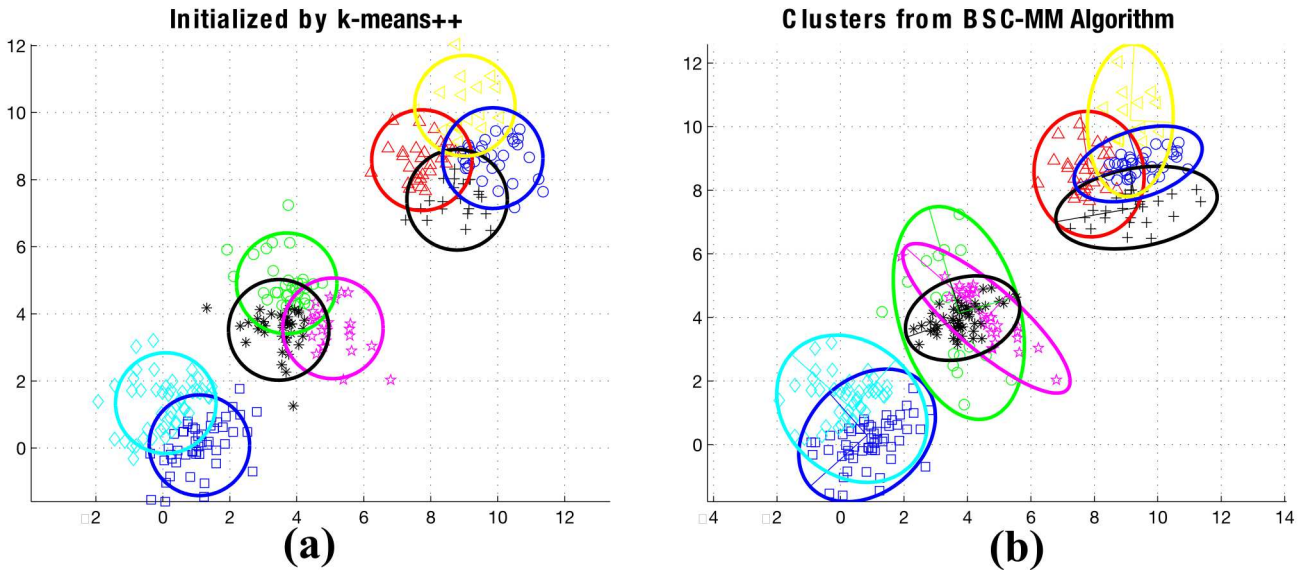


Figure 2.10: Examples of clustering data (of Figure 2.1) with 9 classes. (a) Initialization with k-means++ and (b) Clustering results from BSC-MM algorithm after 20 iterations. Similar to the k-means clustering, for these 2D data, clusters obtained from k-means++ have circular shape. In contrary, the clusters obtained with Gaussian mixture model using BSC-MM algorithm have elliptical shape. This indicates that the Gaussian mixture model is more powerful to model complex structure of data.

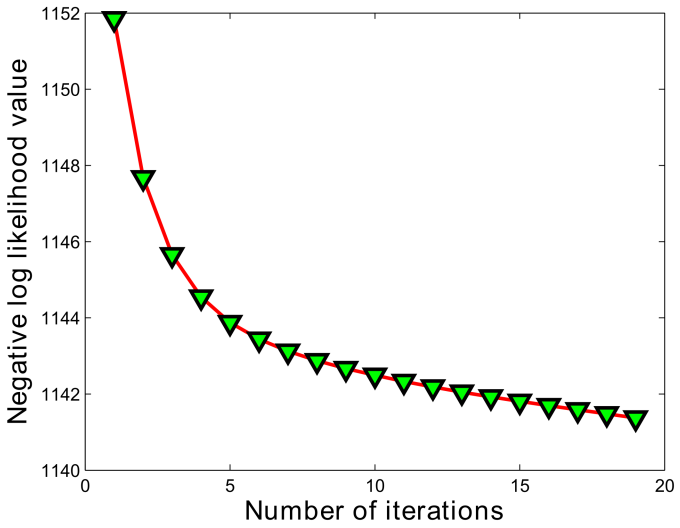


Figure 2.11: Illustration of convergence of the BSC-MM algorithm (Algorithm 1) observed using the negative log-likelihood values. Maximum number of iterations was set to 20 and threshold 'log-likelihood difference among successive steps' was set to 0.01 as the convergence criteria.

Algorithm 1: BSC-MM algorithm for mixture of exponential family of distributions.

Input: $\mathbf{X} = \{\mathbf{x}_i \mid \mathbf{x}_i \in \mathbb{R}^d \wedge 1 \leqslant i \leqslant N\}$ and k
Output: A soft clustering of $\mathbf{X}$ over a mixture model with k components.
Initialize $\pi_{j,k}$ and $\eta_{j,k}$ for $1 \leq j \leq k$ with k-means++;
while *not converged* **do**
  {Perform the E-step of EM};
  foreach *i and j* **do**
    Compute $p_{ij} = p(\gamma_i = j|\mathbf{x}_i)$ using Eq. (2.27)
  end
  {Perform the M-step of EM};
  for $j = 1$ *to* k **do**
    Update $\pi_{j,k}$ and $\eta_{j,k}$ using Eq. (2.28)
  end
end

2.6.2 Model Generation with Hierarchical Clustering

The set of models are the core elements of our Model Based Clustering (MBC) approach, from which we select the optimal model. In a simple approach, one may apply k-means or EM algorithm to generate the desired set of models with different number of components. However, such approach has two important limitations (Zhong and Ghosh, 2003), such as: (a) cannot guarantee structural similarity among different solutions and (b) computation time will increase significantly with the number of desired clustering solutions. We overcome both of these limitations by efficiently employing the Hierarchical Agglomerative Clustering (HAC) to build the set of mixture models $\{\Theta_k\}_{k=k_{min},\ldots,k_{max}-1}$ from a principal model $\Theta_{k_{max}}$. Our proposed HAC method consists of the following three steps:

- *Step 1:* Construct a distance matrix using appropriate type of Bregman divergence (section 2.4) among pairs of clusters (exponential family distributions).
- *Step 2:* Group the objects into a binary, hierarchical cluster tree using appropriate linkage criteria.
- *Step 3:* Compute new cluster representatives using appropriate type of Bregman centroid (section 2.4).

In the above HAC method, one should choose the appropriate distance and centroid type empirically. Figure 2.4 illustrates an example of generating a set of GMMs from the parameters of a GMM with $k_{max} = 9$ components. The GMM with k_{max}

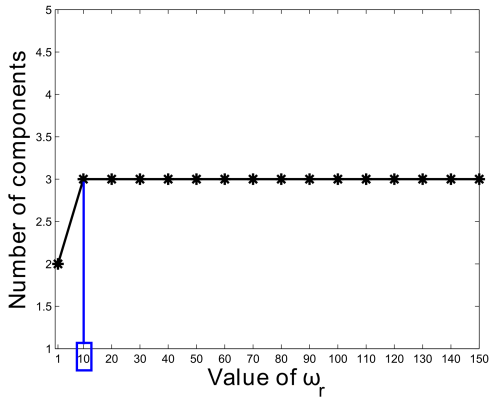


Figure 2.12: Illustration of determining an appropriate weight for $\tau = \omega_r$ with step size 10. The number of components remains same from $\omega_r \geq 10$.

components is estimated using the BSC-MM algorithm (Algorithm 1). To construct the hierarchy of models, we used left-sided BD, 'average-link' criterion and left-sided centroid.

2.6.3 Model Selection

The final task of a Model Based Clustering (MBC) method is to select the best model from a set of models. We propose a method, that combines both parsimony based and graph based methods. Our model selection method to obtain k_o is as follows:

- *Step 1:* Draw an evaluation plot using the BIC criterion.
- *Step 2:* Perform piecewise linear regression fit and calculate $RMSE_{k,left}$ and $RMSE_{k,right}$ for $k = k_{max} - 1, ...k_{min}$.
- *Step 3:* Identify k_o using Eq. (2.22) with $\omega_r \geq \omega_l$.

In step 3, we set $\omega_l = 1$ and we set ω_r empirically, see Figure 2.7. Usually, ω_r can be easily found by obtaining the minimum for the most stable region of values. See Figure 2.12 for an example. The proposed approach is called *weighted piecewise linear regression fit* ($WPLR - \tau$) method, where τ indicates the weight value (with $\tau = \omega_r$). Note that, $WPLR - \tau$ is nearly equivalent to the L-method (Salvador and Chan, 2004) when $k_o = k_{max}/2$ and $\tau = 1$. In the following Chapter, we will further discuss about the setting of ω_r.

2.7 Discussions and Conclusions

In this Chapter, we have presented a novel model based clustering algorithm based on the exponential family of distributions which encompasses a wide class of familiar probability distributions. We provided relevant examples and illustrations with the most familiar multivariate Gaussian distribution. We did not provide the experimental evaluations of any database and applications for any particular tasks in this Chapter. We will provide these in the following Chapter along with our developed model based clustering methods for directional distributions and their applications for depth image analysis.

The proposed model based clustering method employs Bregman soft clustering algorithm to estimate the initial model from data. Then, it constructs a hierarchy of mixture models only from the parameters of the initial model by exploiting the properties of Bregman divergence. Finally, it employs a model selection method to select the best model. The proposed method has the following properties:

1. **Unsupervised:** It is unsupervised, i.e., it does not need to learn from training data. However, similar to any unsupervised method, often it requires setting few parameters to obtain the desired clustering results.

2. **Efficient clustering:** It employs Bregman soft clustering (Banerjee et al., 2005b) algorithm which is an efficient algorithm with additional benefits (see Section 2.1) compared to the traditional EM based methods. We will demonstrate this in the next Chapter.

3. **Structural similarity of models:** The mixture models generated for different number of components guarantees to be structurally similar (Zhong and Ghosh, 2003) as they are computed from the parameters of the model with k_{max} components. This strategy is known as the mixture models simplification process (Garcia and Nielsen, 2010).

4. **Novel model selection:** Besides the widely used parsimony based methods (Melnykov and Maitra, 2010; Alata and Quintard, 2009; Biernacki et al., 2000), it employs a novel model selection approach (called WPLR-τ). WPLR-τ method is a generalized proposal and hence can be incorporated with any other model based clustering methods.

5. **Computationally efficient:** The proposed method applies the EM method to compute the model parameters from data only once. The rest of the models

are generated from the parameters of the initial model, which saves a significant amount of computation time. We will demonstrate this in the next Chapter.

6. **Wide adaptability:** The method is a generalized proposal and can be adapted easily to any probability distributions which belong to the exponential families.

The above discussions reveal that, our method can be an interesting tool for clustering, model simplification, model selection and eventually unsupervised classification. Hence, we believe that the proposed method will be an interesting tool for the machine learning, data mining and pattern recognition community.

Note that, with the Gaussian mixture model, our proposed method has significant similarity with the method proposed by Garcia and Nielsen (2010). However, we propose a novel extension which manipulates it within the model based clustering framework (Fraley and Raftery, 2007) by incorporating different model selection criteria. Moreover, we also propose novel extensions of this method for directional distributions, see next Chapter.

Chapter 3

Clustering with Directional Distributions: Application to Depth Image Analysis

Résumé: Dans ce chapitre, nous utilisons la méthode proposée dans le précédent chapitre afin de classifier des informations directionnelles. De ce fait, nous proposons une méthode de type MBC exploitant les distributions directionnelles. Elle s'appuie sur un modèle "génératif": les données sont supposées être générées par une loi de mélange de distributions directionnelles. Nous avons travaillé avec deux types de distributions directionnelles: la loi de von Mises-Fisher (aussi appelée loi de Langevin) et la distribution de Watson. Tout d'abord, la méthode proposée réalise une classification "douce" permettant d'estimer les paramètres de la loi de mélange pour un nombre maximum de composantes donné. Ensuite, une hiérarchie de modèle est générée sans avoir besoin de réutiliser les données: c'est à partir de cet ensemble de modèle que le modèle optimal (ou le nombre de composantes optimal) sera obtenu à l'aide d'une méthode de sélection empirique. Nous validons les méthodes proposées à l'aide de données simulées. Puis, nous évaluons leurs performances sur des données réelles, en classifiant les normales aux surfaces calculées à partir d'images de profondeur. Les résultats obtenus confirment le fait que les méthodes proposées sont des outils potentiels pour analyser les images de profondeur.

In this Chapter, we extend the methods that we proposed in the previous Chapter in order to cluster directional features. Therefore, we propose a model based clustering approach using the directional distributions. The proposed method is based on the assumption of a generative model, where the data is generated from a finite statistical mixture model. For such models, we particularly consider two fundamental

directional distributions, called the von Mises-Fisher (also called Langevin) and the Watson distribution. Initially, the proposed method applies a soft clustering algorithm in order to obtain the parameters of the mixture model for a given maximum number of components. Then, a hierarchy of mixture models is generated from the parameters. The hierarchy of models represents the desired set of models from where the optimal model should be selected. Finally, an empirical model selection method is applied to select the optimal model, i.e. to select the optimal number of components. First, we validate the proposed methods by applying it on simulated data. Then, to evaluate its performance on real data, we applied them to cluster image normals which are computed from a depth image. As an outcome of the clustering, we obtained a bottom-up segmentation of the depth image. Obtained results confirmed our assumption that the proposed method can be a potential tool to analyze depth images.

3.1 Introduction

Data/features in the form of a unit vector exhibits directional behavior. Normalization is often employed as an important pre-processing step in data analysis, which removes the 'magnitude' of data samples and keeps the directional part as the prominent information (Gopal and Yang, 2014). Directional distributions (Mardia and Jupp, 2009) are the standard choice to model and analyze the directional data. For example, the statistical mixture models with different directional distributions are frequently employed in a variety of domains to analyze images (Da Costa et al., 2012; Grana et al., 2008), speech signals (Vu and Haeb-Umbach, 2010; Souden et al., 2013), text documents (Banerjee et al., 2005b; Maitra and Ramler, 2010; Gopal and Yang, 2014), digits (Bijral et al., 2007), gene expressions (Banerjee et al., 2005a; Sra and Karp, 2013; Maitra and Ramler, 2010), treatment beams (Bangert et al., 2010), shapes (Prati et al., 2008), motion (Kobayashi and Otsu, 2010), pose (Glover et al., 2012), protein structures (Razavian et al., 2011), diffusion MRI (Cabeen et al., 2013; Bhalerao and Westin, 2007), fibrous materials (Zhang, 2013), rock mass (Peel et al., 2001), etc. Several software or packages, such as Mocapy++ (Paluszewski and Hamelryck, 2010) and skmeans (Buchta et al., 2012) are already freely available for these purposes. The wide applicability of directional distributions receives attentions from different communities, which reveals the necessity of developing efficient solutions. We focus on proposing solutions for unsupervised classification with such distributions.

The sample spaces for the directional distributions are the circle (S^1), the sphere (S^2) and the hypersphere ($S^{d-1}, d > 3$). Most prominent distributions in directional statistics are the von Mises-Fisher, Watson, Kent, Bingham etc. (Mardia and Jupp, 2009). These distributions model data concentrated around the mean-direction. For example, the von Mises-Fisher and Watson distributions have minimal set of parameters which are the mean and concentration. These distributions are rotationally symmetric around the mean direction. The Kent and Bingham consist of more parameters to model data. An important property of these distributions is that, they belong to the exponential family of distributions (Mardia and Jupp, 2009). This property allows these distributions to be exploited within the model based clustering (Fraley and Raftery, 2007) framework (discussed in Chapter 2) and hence to develop efficient clustering solutions. In this Chapter, we focus on developing such solutions with the von Mises-Fisher and Watson distributions.

Directional distributions are associated with complicated normalizing constants. For this reason, analytical solution to obtain maximum likelihood estimate (MLE) of the parameters even for a single distribution is difficult (Sra, 2012). Specially, estimation of the concentration parameters is often non-trivial since they involve functional inversion of the ratios of special functions such as Bessel function, Kummer's function, etc. Therefore, unlike the well-known models, such as GMM, it requires special formulations to incorporate the directional distributions in the model based clustering (Fraley and Raftery, 2007) framework. Recently, methods to estimate parameters of these probability distributions have been revisited and better solutions are now provided (Sra, 2012; Sra and Karp, 2013). Although these solutions are within the context of clustering, they do not address the issue of automatic component selection. We address this issue from the perspective of model based clustering. To this aim we develop solutions, not only to estimate parameters efficiently but also to find the number of clusters automatically.

When a clustering method is applied for image analysis, it generates several groups of pixels. Usually these groups represent a distinctive set of regions/segments in the image. Therefore, the problem of image segmentation can be addressed from cluster analysis (Szeliski, 2011). To perform clustering, image pixels are described by different attributes/features. Pixels of a depth image can be described by features such as depth, 3D point, surface normal, etc. (Rusu, 2013). See Fig. 3.1(a) for an example, which shows that segmentation using surface normals is most relevant to the ground truth in certain cases. The reason is that, in some contexts it makes sense to group together the normals belonging to similar planar surfaces in the image. Motivated

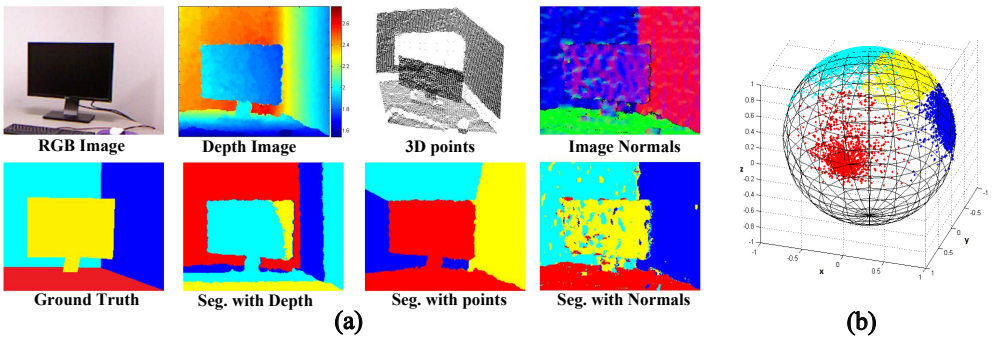


Figure 3.1: (a) Examples of Depth Image clustering. First row presents different image features. Second row illustrates the ground truth and segmentation results using k-means with depth and with 3D points and using SP-kmeans (Banerjee et al., 2005a) with surface normals. Note that, here we explicitly set $k = 4$. (b) Sample space (sphere: S^2) for surface normals.

by this observation, we address the problem of depth image analysis using surface normals. Surface normal is a 3D unit vector that describes the planar property for each pixel of a depth image. Its sample space belongs to the sphere (S^2), see Fig. 3.1(b). Therefore, we can apply our proposed clustering methods (developed in this Chapter) on the normals to segment and analyze the depth images.

In this Chapter, we present model based clustering methods with two fundamental directional distributions: the von Mises-Fisher (also called Langevin) and the Watson distribution. These methods are first evaluated with synthetic data. Then, they are applied on real depth image data to cluster surface normals. We used the depth images from the NYU Depth Database V2 (NUYD2) (Silberman et al., 2012) for the experiments. Evaluations shown in Section 3.4 confirm that, on simulated data the proposed methods are better than the state of the art methods. Moreover, on real data they have potential applications, such as to analyze depth images by clustering image normals.

The remaining of this Chapter is structured as follows: Section 3.2 provides the background related to the directional distributions. Section 3.3 presents the proposed clustering method. Experimental results followed by discussions are reported in Section 3.4. Finally, Section 3.5 draws conclusion and possible future extensions of the proposed methods.

3.2 Directional Distributions, Mixture Models and Bregman Divergence

Directional data arise frequently in a number of practical data analysis applications either due to their natural appearance or due to applying $L2$ normalization on the data (Banerjee et al., 2005a; Gopal and Yang, 2014). The 'magnitude' of these data is unknown or irrelevant, whereas the direction is the prominent information. In several cases the sign of these data is also unknown and hence they are represented with only an axis (Mardia and Jupp, 2009). In both directional and axial forms of these data, the Spherical geometry is the appropriate choice for them rather than the standard Euclidean geometry. Moreover, the popular data modeling approach such as the Gaussian mixture model is inadequate to characterize this type of data (Banerjee et al., 2005a). Directional distributions are the appropriate choice for them. Among the number of directional distributions, we particularly focus on the von Mises-Fisher distribution for signed directional data and the Watson distribution for unsigned directional data or axial data.

3.2.1 von Mises-Fisher (vMF) Distribution

The fundamental directional distribution is called the von Mises-Fisher (vMF) distribution, which models data concentrated around a mean-direction. Originally, it is known as the Langevin distribution (Watson, 1984). Moreover, for $d = 2$ it is called the von-Mises distribution and for $d = 3$ it is called the Fisher distribution (Mardia and Jupp, 2009).

For a $d\,(d \geq 2)$ dimensional random unit vector $\mathbf{x} = [x_1, ..., x_d]^T \in S^{d-1} \subset \mathbb{R}^d$ (i.e., $\|\mathbf{x}\|_2 = 1$), the von Mises-Fisher (or Langevin) distribution is defined as (Mardia and Jupp, 2009):

$$V_d(\mathbf{x}|\mu, \kappa) = Q_d(\kappa) \exp(\kappa \mu^T \mathbf{x}) \tag{3.1}$$

Here, μ denotes the mean (with $\|\mu\|_2 = 1$) and κ denotes the concentration parameter (with $\kappa \geq 0$). The normalization constant $Q_d(\kappa)$ is equal to:

$$Q_d(\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)}$$

Here $I_\jmath(.)$ represents the modified Bessel function of the first kind and order $\jmath$, which has the following power series expression (Mardia and Jupp, 2009):

$$I_\jmath(\kappa) = \sum_{r=0}^{\infty} \frac{1}{\Gamma(\jmath + r + 1)\Gamma(r + 1)} \left(\frac{\kappa}{2}\right)^{2r+p} \tag{3.2}$$

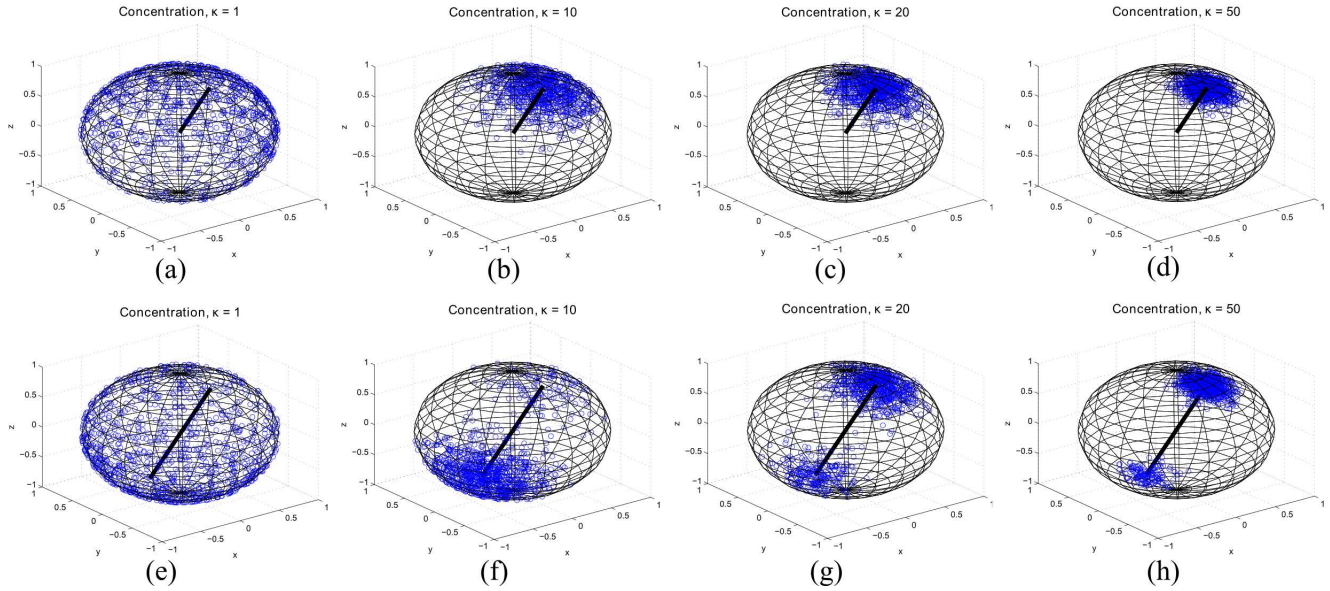


Figure 3.2: 3 dimensional directional samples from the von Mises-Fisher distribution (top row) and the Watson distribution (bottom row). Samples are shown in the S^2 sphere for different values of the concentration (κ) parameters.

For higher ($d > 3$) dimensional data, the analytical solution to estimate the concentration parameter (κ) of vMF is non-trivial since it involves functional inversion of ratios of the Bessel functions (Banerjee et al., 2005a). However, for $d = 3$ the normalizing factor simplifies and can be written without the Bessel function as (Mardia and Jupp, 2009):

$$Q_d(\kappa) = \frac{\kappa}{\sinh(\kappa)}$$

For this reason, we limit our study of vMF for $d = 3$. Considering this normalizing factor, we can rewrite Eq. (3.1) as:

$$V_d(\mathbf{x}|\mu, \kappa) = \exp\left(\kappa \mu^T \mathbf{x} - \log\left(\frac{\sinh(\kappa)}{\kappa}\right)\right) \tag{3.3}$$

The shape of the vMF distribution depends on the value of the concentration parameter κ. For high value of κ, i.e. highly concentrated observations, the distribution has a mode at the mean direction μ. In contrary, for low values of κ the distribution is almost uniform, i.e. the samples appear as to be almost uniformly distributed on the sphere. Beside these, the shape of the distribution is rotationally symmetric about μ as the density function in Eq. (3.1) or (3.3) depends on $\mathbf{x}$ only through $\mu^T\mathbf{x}$. The top row of Figure 3.2 illustrates examples of 3D samples in the S^2 sphere, which are distributed according to the vMF distribution with different values of the concentration κ.

3.2.2 Watson Distribution

Multivariate Watson Distribution (mWD) is a fundamental distribution that models axially symmetric directional data (i.e., unit vectors where $\pm\mathbf{x}$ is equivalent). For a d dimensional axially symmetric unit vector $\pm\mathbf{x} = [x_1, ..., x_d]^T \in S^{d-1} \subset \mathbb{R}^d$ (i.e., $\|\mathbf{x}\|_2 = 1$), the multivariate Watson distribution (mWD) is defined as (Mardia and Jupp, 2009):

$$W_d(\mathbf{x}|\mu, \kappa) = M\left(a, c, \kappa\right)^{-1} \exp\left(\kappa(\mu^T\mathbf{x})^2\right) \tag{3.4}$$
$$\text{and } W_d(-\mathbf{x}|\mu, \kappa) = W_d(\mathbf{x}|\mu, \kappa)$$

Here, μ is the mean direction (with $\|\mu\|_2 = 1$), $\kappa \in \mathbb{R}$ the concentration, $a = 1/2$, $c = d/2$ and $M\left(a, c, \kappa\right)$ is the Kummer's confluent hypergeometric function defined as (Sra and Karp, 2013):

$$M\left(a, c, \kappa\right) = \sum_{\rho \geq 0} \frac{a^{\overline{\rho}}}{c^{\overline{\rho}}} \frac{\kappa^\rho}{\rho!}\ , \quad a, c, \kappa \in \mathbb{R},\ \rho \in \mathbb{N} \tag{3.5}$$

where, $a^{\overline{0}} = 1\ , a^{\overline{\rho}} = a(a+1)...(a+\rho-1), \rho \geq 1$ denotes the rising factorial.

Similar to the vMF distribution, mWD is rotationally symmetric about the mean μ and the shape depends on the value of the concentration parameter κ. However, unlike vMF the κ value can have both positive and negative values. For $\kappa < 0$, the distribution is concentrated around the great circle orthogonal to μ and it is a symmetric girdle distribution (Mardia and Jupp, 2009). For $\kappa > 0$, the distribution has maxima at $\pm\mu$ and it is bipolar. In such case, the Watson distribution exhibits similar shape as the vMF w.r.t. the value of κ. The bottom row of Figure 3.2 illustrates examples of 3D samples in the S^2 sphere, which are distributed according to the mWD distribution with different values of the concentration κ. The line indicates the direction of the axis. We see that, the samples are bipolar and concentrated about μ based on the value of κ.

3.2.3 Clustering with Mixture of Directional Distributions

Clustering is a fundamental tool which has been vastly used for data modeling and analysis. It can be defined as the task of automatically identifying the groups of similar observations from a given set of data points. Numerous clustering methods, such as k-means based (Buchta et al., 2012; Maitra and Ramler, 2010), Bayesian approach (Gopal and Yang, 2014), mixture model based (Banerjee et al., 2005a; Sra and Karp, 2013), non-parametric (Kobayashi and Otsu, 2010) etc. already exist to model and analyze directional data. Among them, the statistical mixture model

based methods are most popular and powerful due to their ability to model and cluster data as well as provide greater insight into the anatomy of the clusters via the model parameters (Banerjee et al., 2005a; Sra and Karp, 2013). In this Chapter, we mainly focus on the methods related to the mixture of directional distributions.

3.2.3.1 von Mises-Fisher (vMF) Mixture Model

Let us recall notations and models from Chapter 2 and denote a set of data samples as $\mathbf{X} = \{\mathbf{x}_i\}_{i=1,...,N}$ and associated labels as $\Gamma = \{\gamma_i\}_{i=1,...,N}$, $\gamma_i \in \{1, ..., k\}$. We assume a generative model (Murphy, 2012), which consists of a mixture of k von Mises-Fisher (vMF) distributions, also called vMF Mixture Model (vMFMM) as:

$$g_v\left(\mathbf{x}_i|\Theta_k\right) = \sum_{j=1}^{k} \pi_{j,k} V_d\left(\mathbf{x}_i|\mu_{j,k}, \kappa_{j,k}\right) \tag{3.6}$$

where $\Theta_k = \{(\pi_{1,k}, \mu_{1,k}, \kappa_{1,k}), ..., (\pi_{k,k}, \mu_{k,k}, \kappa_{k,k})\}$ is the set of component parameters, $\pi_{j,k}$ is the mixing proportion and $V_d\left(\mathbf{x}_i|\mu_{j,k}, \kappa_{j,k}\right)$ is the density function (Eq. (3.3)) of the vMF distribution for the j^{th} component.

Finite vMFMM was introduced by Banerjee et al. (2005a). They proposed soft clustering for mixture of vMF, called soft-MoVMF algorithm, that employs Expectation Maximization (EM) method for computing parameters of the mixture model. Very recently, Gopal and Yang (2014) proposed a Bayesian formulation for vMF clustering models. However, none of the above methods automatically select the number of components. Infinite vMFMM (iMFMM) was proposed by Bangert et al. (2010), which addressed the issue of components selection. However, iMFMM is a non-deterministic approach and computationally very expensive. A nonlinear least-squares technique to compute parameters of vMFMM was proposed by McGraw et al. (2006). However, their method do not explicitly address the clustering issue. To select the number of components for directional data, Banerjee et al. (2005a) suggested the PAC-MDL[1] bound for vMFMM in a semi-supervised case.

3.2.3.2 Watson Mixture Model

Similar to the vMF mixture model (Eq. (3.6)), let us now define a mixture of k Watson distributions, also called Watson Mixture Model (WMM) as (Sra and Karp, 2013):

$$g_w\left(\mathbf{x}_i|\Theta_k\right) = \sum_{j=1}^{k} \pi_{j,k} W_d\left(\mathbf{x}_i|\mu_{j,k}, \kappa_{j,k}\right) \tag{3.7}$$

[1]PAC - Probably Approximately Correct, MDL - Minimum Description Length

where $\mathbf{x}_i$ denotes a single sample, $\Theta_k = \{(\pi_{1,k}, \mu_{1,k}, \kappa_{1,k}), ..., (\pi_{k,k}, \mu_{k,k}, \kappa_{k,k})\}$ is the set of component parameters, $\pi_{j,k}$ is the mixing proportion and $W_d(\mathbf{x}_i|\mu_{j,k}, \kappa_{j,k})$ is the density function (Eq. (3.4)) of the Watson distribution for the j^{th} component.

The multivariate Watson Distribution (mWD) has received relatively less attention in comparison to the other distributions in the directional statistics. Most recently Sra and Karp (2013) provided theoretically well justified estimation of the parameters of mWD. They considered the Watson Mixture Model (WMM) to model axially symmetric data and used the EM algorithm to estimate the model and cluster data. Before that, Bijral et al. (2007) employed WMM for hyperspherical embedding and shown its application to digit clustering. Vu and Haeb-Umbach (2010) employed WMM for blind speech separation. Both of them used Expectation Maximization (EM) methods with different approximations of the model parameters. However, according to Sra and Karp (2013) those approximations are not numerically well justified. Souden et al. (2013) recently used WMM for speech clustering and computed parameters following Sra and Karp (2013). None of these methods explicitly focus on selecting the number of clusters in the data.

Studying the related work on clustering directional data using mixture model based approaches, we observed that there is no method that performs automatic component selection and that considers a model based clustering approach. These observations motivate us to extend the model based clustering method (presented in Chapter 2) for the directional distributions. To this aim, the first step is to derive Exponential family formulations and the computation of Bregman Divergence among the directional distributions.

3.2.4 Bregman Divergence for Directional Distributions

Bregman Divergences (BD) generalize a number of distortion functions which are commonly used in clustering (Banerjee et al., 2005b). It is one of the most important elements of the model based clustering method proposed in Chapter 2. A probability distribution can take the benefits of Bregman Divergence if its canonical exponential family representation is available. While it exists for several commonly used probability distributions (Garcia and Nielsen, 2010), the directional distributions are yet to have such representation. In this sub-Section, we derive the Bregman Divergence for the von Mises-Fisher and the Watson distribution.

Let us shortly recall the Exponential Family of Distributions (EFD) and Bregman Divergence, see Chapter 2 for details. A probability density function $f(\mathbf{x}|\theta)$ belongs

to the EFD if it has the following form:

$$f(\mathbf{x}|\theta) = \exp\left(\langle t(\mathbf{x}), \theta\rangle - F(\theta) + k(\mathbf{x})\right) \tag{3.8}$$

Here, $t(\mathbf{x})$ is the sufficient statistics, θ is the natural parameter, $F(\theta)$ is the log normalizing function, $k(\mathbf{x})$ is the carrier measure and $< ., . >$ is the inner product. The expectation of the sufficient statistics $E[t(\mathbf{x})]$ is called the expectation parameter (η). There exists a one-to-one correspondence between η and θ, which is expressed as:

$$\eta = \nabla_\theta F(\theta) \quad \text{and} \quad \theta = (\nabla_\theta F(\theta))^{-1}(\eta) \tag{3.9}$$

with ∇ is the gradient operator. The Bregman Divergence with the expectation parameter η can be defined as:

$$D_G(\eta_1, \eta_2) = G(\eta_1) - G(\eta_2) - \langle \eta_1 - \eta_2, \nabla G(\eta_2)\rangle \tag{3.10}$$

where, $G(.)$ is the Legendre dual of $F(.)$.

3.2.4.1 Bregman Divergence among vMF Distributions

Considering the canonical form of exponential family (Eq. (3.8)), the vMF defined in Eq. (3.3) can be decomposed as:

- sufficient statistics $t(\mathbf{x}) = \mathbf{x}$,
- natural parameter $\theta = \kappa\mu$,
- log normalizing function $F(\theta) = \log\left(\frac{\sinh(\kappa)}{\kappa}\right)$, which is a convex function and
- carrier measure $k(\mathbf{x}) = 0$.

The mean μ $(\|\mu\|_2 = 1)$ and concentration parameter κ $(\kappa > 0)$ can be written in terms of the natural parameter θ as:

$$\theta = \kappa\mu;\ \mu = \frac{\theta}{\|\theta\|_2} \text{ and } \kappa = \|\theta\|_2 \tag{3.11}$$

The gradient of the log normalizing function $(\nabla_\theta F(\theta))$ can be written as:

$$\nabla_\theta F(\theta) = \nabla_\kappa \log\left(\frac{\sinh(\kappa)}{\kappa}\right).\nabla_\theta \kappa$$

Considering Eq. (3.9) we can write:

$$\eta = \nabla_\theta F(\theta) = \left\{\tanh(\kappa)^{-1} - (\kappa)^{-1}\right\}.\frac{\theta}{\kappa} \tag{3.12}$$

and

$$\theta = \frac{\eta}{R(\kappa)} \tag{3.13}$$

where,

$$R(\kappa) = \left\{(\tanh(\kappa))^{-1} - (\kappa)^{-1}\right\}(\kappa)^{-1} \tag{3.14}$$

Using property of collinear vectors in Eq. (3.12) we can write:

$$\left\{(\tanh(\kappa))^{-1} - (\kappa)^{-1}\right\} = \|\eta\|_2$$

We can then apply the Newton-Raphson method to compute κ from $\|\eta\|_2$ using an iterative update equation as:

$$\kappa_{n+1} = \kappa_n - \frac{a - b - \|\eta\|_2}{1 - a^2 + b^2} \tag{3.15}$$

where, $a = \tanh(\kappa)^{-1}$ and $b = (\kappa)^{-1}$. Now, considering $\theta = \nabla_\eta G(\eta)$ (Nielsen and Garcia, 2009), we can use equations (3.10, 3.13, 3.14 and 3.15) to compute Bregman Divergence among the vMF distributions.

3.2.4.2 Bregman Divergence among Watson Distributions

In order to obtain canonical Exponential Family form of a multivariate Watson distribution, let us rewrite Eq. (3.4) as:

$$W_d(\mathbf{y}|\nu,\kappa) = \exp\left\{\kappa\nu^T\mathbf{y} - \log M\left(\kappa\right)\right\} \tag{3.16}$$

with $\mathbf{y}, \nu \in \mathbb{R}^p$, $p = d + C_2^d$:

$$\mathbf{y} = \left[x_1^2, ..., x_d^2, \sqrt{2}x_1x_2, ..., \sqrt{2}x_{d-1}x_d\right]^T$$

$$\nu = \left[\mu_1^2, ..., \mu_d^2, \sqrt{2}\mu_1\mu_2, ..., \sqrt{2}\mu_{d-1}\mu_d\right]^T$$

where, $\mathbf{y}$ and ν are the vectors associated with the sample ($\mathbf{x}$) and mean (μ). In Eq. (3.16), we write $M(\kappa)$ instead of $M(1/2, p/2, \kappa)$ for the sake of brevity. Following Eq. (3.8), we can decompose the multivariate Watson distribution in Eq. (3.16) as:

- sufficient statistics $t(\mathbf{x}) = \mathbf{y}$,
- natural parameter $\theta = \kappa\nu$,
- log normalizing function $F(\theta) = \log M(\kappa)$ and
- carrier measure $k(\mathbf{x}) = 0$.

Then, we can write ν and κ in terms of natural parameter θ as:

$$\theta = \kappa\nu;\ \nu = \frac{\theta}{\|\theta\|_2}\ and\ \kappa = \|\theta\|_2 \tag{3.17}$$

Now, we can write the gradient of the log normalizing function $F(\theta)$ as:

$$\eta = \nabla_\theta F(\theta) = q(a, c; \kappa)\frac{\theta}{\kappa} \tag{3.18}$$

where, $q(a, c; \kappa)$ is called the Kummer-ratio and defined as (Sra and Karp, 2013):

$$q(a, c; \kappa) = \frac{M'(\kappa)}{M(\kappa)} := \frac{M'(a, c, \kappa)}{M(a, c, \kappa)} = \frac{a}{c}\frac{M(a+1, c+1, \kappa)}{M(a, c, \kappa)} \tag{3.19}$$

From Eq. (3.18) we can define the natural parameter θ as:

$$\theta = \frac{\eta\kappa}{q(a, c; \kappa)} \tag{3.20}$$

Moreover, using Eq. (3.17) and (3.18) we can write:

$$q(a, c; \kappa) = \|\eta\|_2 \tag{3.21}$$

Similar to Sra and Karp (2013), we can apply Newton-Raphson root finder method to approximate κ from $\|\eta\|_2$ (in Eq. (3.21)) using the following iterative update equation:

$$\kappa_{l+1} = \kappa_l - \frac{q(a, c; \kappa_l) - \|\eta\|_2}{q'(a, c; \kappa_l)} \tag{3.22}$$

where, $q'(a, c; \kappa)$ is the first derivative of the Kummer-ratio (Eq. (3.21)) and can be calculated as (Sra and Karp, 2013):

$$q'(a, c; \kappa) = (1 - \frac{c}{\kappa})\ q(a, c; \kappa) + \frac{a}{\kappa} - q(a, c; \kappa)^2 \tag{3.23}$$

Now, considering $\theta = \nabla_\eta G(\eta)$ (Garcia and Nielsen, 2010), we can use equations (3.10, 3.20, and 3.22) to compute Bregman Divergence among the Watson distributions. Note that instead of computing the mean μ directly, we compute ν. Then to obtain μ, we take the square root of the first d elements of ν. However, to recover the sign we use a lookup table.

3.3 Methodology

In this Section, first we present the methodology for the proposed model based clustering method. Then, we present how the clustering method is applied for depth image analysis.

3.3.1 Model Based Clustering

Model based clustering estimates a model for the data and produces probabilistic clustering. It identifies the best model by fitting a set of models with different parameterizations and/or number of components and then applying a statistical criterion for model selection (Fraley and Raftery, 2007). Currently, there exists no model based clustering method with the directional distributions. In this Section, we develop such method with the von Mises-Fisher and the Watson distribution. Since these distributions belong to the Exponential families, we follow the same methodologies presented in Chapter 2.

Model based clustering requires a certain model to be defined for the data. For directional data, we consider the von Mises-Fisher Mixture Model (vMFMM) which is defined in Eq. (3.6) (see Section 3.2.3.1). For axial data, we consider the Watson Mixture Model (WMM) which is defined in Eq. (3.7) (see Section 3.2.3.2). Interestingly, both vMFMM and WMM consist of the same type of parameters. Therefore, using expectation parameters (η), let us uniquely define a k components vMFMM or WMM as $\Theta_k = \{(\pi_{1,k}, \eta_{1,k}), ..., (\pi_{k,k}, \eta_{k,k})\}$. Similarly, $\Theta_{k_{max}}$ denotes the mixture model with k_{max} components and Θ_{k_o} denotes the optimal mixture model with k_o components. To cluster a set of observations (directional/axial), the model based clustering method follows the step-by-step procedure as:

- *Step 1:* Apply Bregman soft clustering algorithm to compute $\hat{\Theta}_{k_{max}}$.
- *Step 2:* Generate a set of models $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$ from $\hat{\Theta}_{k_{max}}$.
- *Step 3:* Select the optimal model $\hat{\Theta}_{k_o}$ from $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$.

As described in Chapter 2, the proposed method begins with applying Bregman soft clustering on the observations to estimate model parameters $\hat{\Theta}_{k_{max}}$. Then, it applies the hierarchical agglomerative clustering on $\hat{\Theta}_{k_{max}}$ to obtain $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$. Finally, it employs a model selection method on $\{\hat{\Theta}_k\}_{k=k_{min},...,k_{max}-1}$.

First, in *Step 1*, we apply Bregman Soft Clustering (BSC) algorithm on the model $\Theta_{k_{max}}$ defined by Eq. (3.6) or (3.7) with k_{max} components. The goal of applying the BSC algorithm is to obtain $\hat{\Theta}_{k_{max}}$ such that the value of likelihood function is maximized. The BSC algorithm for vMFMM and WMM is provided in Algorithm 2. At the beginning, we initialize π and η of the mixture model. We employ the *kmeans++* (Arthur and Vassilvitskii, 2007) to initialize the vMFMM parameters (Eq. (3.6)) and *diametric clustering* (Dhillon and Sra, 2003) to initialize the WMM parameters

Algorithm 2: Bregman Soft Clustering algorithm for vMFMM or WMM.

Input: $\mathbf{X} = \{\mathbf{x}_i\}_{i=1,\ldots,N}$ and K; $\mathbf{x}_i \in S^2$ for vMFMM or $\mathbf{x}_i \in S^{d-1}$ for WMM $(d \geq 2)$

Output: A soft clustering of $\mathbf{X}$ over a vMFMM or WMM with K components.

Initialize $\pi_{j,K}$ and $\eta_{j,K}$ for $1 \leq j \leq K$ with *kmeans++* for vMFMM or *diametric clustering* for WMM;

while *not converged* **do**

{Perform the E-step of EM};

foreach *i and j* **do**

$$p_{ij} = p\left(\gamma_i = j|\mathbf{x}_i\right) = \frac{\pi_{j,K} \exp\left(G(\eta_{j,K}) + \langle t(\mathbf{x}_i) - \eta_{j,K}, \nabla G(\eta_{j,K})\rangle\right)}{\sum_{l=1}^{K} \pi_{l,K} \exp\left(G(\eta_{l,K}) + \langle t(\mathbf{x}_i) - \eta_{l,K}, \nabla G(\eta_{l,K})\rangle\right)} \tag{3.24}$$

end

{Perform the M-step of EM};

for $j = 1$ *to* K **do**

$$\pi_{j,K} = \frac{1}{N}\sum_{i=1}^{N} p_{ij} \text{ and } \eta_{j,K} = \frac{\sum_{i=1}^{N} p_{ij}\,\mathbf{x}_i}{\sum_{i=1}^{N} p_{ij}} \tag{3.25}$$

end

end

(Eq. (3.7)). After initialization, we iteratively apply the E-step and M-step until convergence.

Next, in *Step 2*, we apply Hierarchical Agglomerative Clustering (HAC) on $\hat{\Theta}_{k_{max}}$ and generate a set of models $\{\hat{\Theta}_k\}_{k=k_{min},\ldots,k_{max}-1}$. For different settings of HAC method, we empirically determine the distance type as 'left sided Bregman Divergence', linkage criterion as 'average link' and centroid type as 'left sided Bregman Centroid'. See Section 2.4 of Chapter 2 for details of the computations and see Section 3.4.1.3 in this Chapter for empirical justifications.

Finally, in *Step 3*, we apply an empirical model selection criterion in order to select the best model $\hat{\Theta}_{k_o}$ from the set of models $\{\hat{\Theta}_k\}_{k=k_{min},\ldots,k_{max}-1}$. See Section 2.5 of Chapter 2 for details of the model selection methods.

After applying the above steps, we have the estimated model $\hat{\Theta}_{k_o}$ and a soft clustering of the observations. However, if a hard clustering is desired, then it can be obtained by using Bregman Divergence as:

$$\hat{\gamma}_i = \underset{j=1,\ldots,k_o}{\arg\min}\; G(\eta_{j,k_o}) + \langle t(\mathbf{x}_i) - \eta_{j,k_o}, \nabla G(\eta_{j,k_o})\rangle \tag{3.26}$$

where $\hat{\gamma}_i$ is the class label corresponding to the observation $\mathbf{x}_i$. Now, for further uses let us define several abbreviations for the methods developed so far in this Chapter:

BSC-vMFMM: Bregman soft clustering with von Mises-Fisher Mixture Model. Algorithm 2 with vMFMM is used as the model. Number of components is pre-specified.

BSC-WMM: Bregman soft clustering with Watson Mixture Model. Algorithm 2 with WMM is used as the model. Number of components is pre-specified.

MBC-vMFMM: Model based clustering with von Mises-Fisher Mixture Model. Clustering method presented in Section 3.3.1 with vMFMM is used as the model.

MBC-WMM: Model based clustering with Watson Mixture Model. Clustering method presented in Section 3.3.1 with WMM is used as the model.

3.3.2 Depth Image Analysis

We follow a clustering based approach for depth image analysis. To this aim, our method clusters the surface normals of a depth image. The normal is usually computed by fitting a plane on the neighborhood 3D points of each pixel. For a plane: $ax + by + cz + d = 0$, the vector (a, b, c) is the normal. Therefore, given a depth image, first we obtain the 3D points (using camera parameters) and then compute the normal for each pixel. In the experiments with real images, we used the toolbox of NYU database (Silberman et al., 2012) to compute normals.

Fig. 3.3 illustrates the block diagram of our proposed method. First, we compute the surface normals of the depth image. Then, we apply the MBC-vMFMM or MBC-WMM to cluster the normals. Using hard clustering (Eq. (3.26)), we assign a cluster label to each pixel. This generates a set of regions/segments of the depth image. Based on literature, our method belongs to the family of agglomerative/bottom up image segmentation method (Szeliski, 2011).

3.4 Experiments

We evaluate MBC-vMFMM and MBC-WMM methods by conducting experiments with directional and axial data samples processed from both synthetic and real dataset. The results associated with each method are presented separately in two

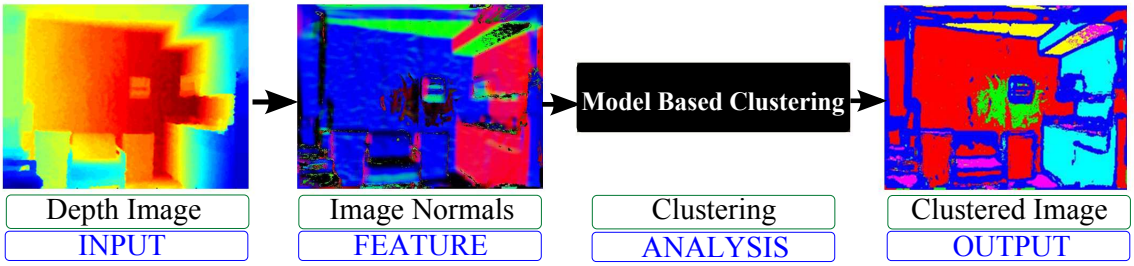


Figure 3.3: Block diagram of the proposed depth image analysis method.

sub-Sections. Note that, we found a number of similarities among these methods, especially for setting the parameters and different criteria. Therefore, we present brief results only for MBC-vMFMM and skip redundant results for MBC-WMM.

For each method, the results are presented in two parts. In the first part, the method is evaluated with simulated data samples for which the true cluster labels are known. We use the global clustering accuracy for evaluation, which is computed as the total number of true positives for all classes divided by the total number of samples. We also computed the Purity, Rand Index and Mutual Information (Murphy, 2012), which provide a complementary result. In the second part, the method is evaluated using real data by applying it to depth image analysis.

3.4.1 Model Based Clustering with von Mises-Fisher Mixture Model (MBC-vMFMM)

The simulated data experiments with MBC-vMFMM method consist of: (1) finding appropriate setting (e.g., initialization, convergence criteria, distance and centroid type, linkage criteria) and (2) comparative evaluation w.r.t. the state of the art methods. Experiments with depth images consist of comparing the results from MBC-vMFMM with the state of the art clustering methods which are commonly employed for image analysis (see Chapter 5.3 of Szeliski (2011)).

3.4.1.1 Simulated Data Samples

Using a standard sampling method for vMFMM (Dhillon and Sra, 2003), we draw a finite set of 3D sample unit vectors $\mathbf{X} = \{\mathbf{x}_i\}_{i,\ldots,N} \in \mathbb{R}^3$, from a vMFMM with different numbers (3, 5 and 7) of components. For the experiments, we generate 100 sets of data from two types of samples: (a) well separated ($\boldsymbol{ws}$) with manually selected parameters and (b) not-well separated ($\boldsymbol{nws}$) with random parameters. For

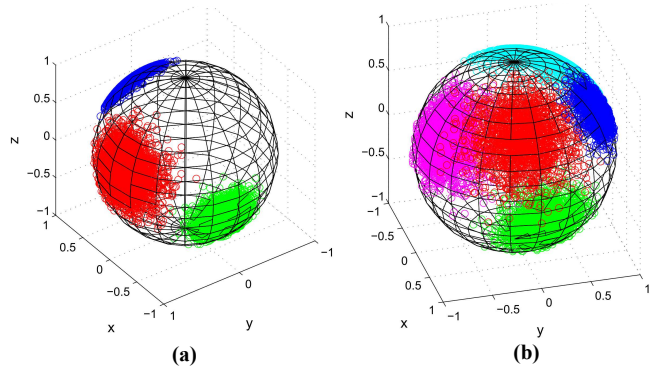


Figure 3.4: Simulated data samples drawn from vMFMM (a) Well separated 3 classes; (b) Not-well separated 5 classes.

each type and each set, we generate 10,000 identically and independently distributed samples. Fig. 3.4 illustrates an example of simulated data samples.

3.4.1.2 Bregman Soft Clustering for vMF Mixture Model (BSC-vMFMM)

Since we follow a deterministic approach for model selection, we set the bounds for the number of components as $k_{max} = 15$ and $k_{min} = 1$. The convergence criteria of the BSC-vMFMM (*Algorithm-2* with vMFMM) are based on maximum number of iterations, set[2] to 100, and a threshold difference, set to 0.001, between the negative log likelihood values ($nLLH$) of successive iterations. We compute the $nLLH$ with $k = k_{max}$ as:

$$nLLH(\Theta_k) = -\log(g_v(\mathbf{X}|\Theta_k)) = -\sum_{i=1}^{N} \log \left(\sum_{j=1}^{k} \pi_{j,k} V_d(\mathbf{x}_i|\mu_{j,k}, \kappa_{j,k}) \right) \tag{3.27}$$

We begin by evaluating the initialization methods for BSC-vMFMM. Table 3.1 presents the results, which shows that, for higher number of clusters with not-well separated samples, the initialization provided by kmeans++ (Arthur and Vassilvitskii, 2007) leads to better classification accuracy. Moreover, from experiments we observed that initialization with kmeans++ is better w.r.t. the stability and convergence time.

Next, we evaluate and compare the performance of BSC-vMFMM w.r.t. the state of the art methods: Gaussian mixture model, Spherical kmeans (Banerjee et al., 2005a), k-means-directions algorithm (Maitra and Ramler, 2010) and soft-MoVMF (Banerjee et al., 2005a). We use the simulated data set (Section 3.4.1.1) for which

[2]In practice, these settings depend on the requirements from clustering methods, such as speed of convergence, computation time, etc. For example, in MATLAB the default values of clustering with Gaussian mixture model are: maximum iteration = 100, threshold log likelihood difference = $1e-6$.

Table 3.1: Evaluation of the initialization methods for clustering with the BSC-vMFMM (clustering accuracy in %). Experimented on not-well separated (*nws*) samples of 3 and 5 classes. **Methods:** randomly initialized kmeans (KM), kmeans++ (KMPP) (Arthur and Vassilvitskii, 2007) and stochastic EM mean (SEMmean) (Biernacki et al., 2003).

	KM	**KMPP**	**SEMmean**
3 cl, nws	99.05	99.05	99.05
5 cl, nws	95.16	97.16	95.18

Table 3.2: Comparison of clustering accuracy (in %). Experimented on simulated data samples containing 3 and 5 components mixture of two types: well separated (*ws*) and not-well separated (*nws*). **Methods:** Gaussian Mixture Model (GMM), Spherical kmeans (SPKM), k-means-directions algorithm (KMDR), soft-MoVMF and BSC-vMFMM (Algorithm 2 with vMFMM). True numbers of components are provided as input.

	GMM	**SPKM**	**KMDR**	**soft-MoVMF**	**BSC-VMFMM**
3 cl, ws	91.71	98.23	98.30	98.92	**99.99**
3 cl, nws	90.5	92.25	98.55	93.07	**99.05**
5 cl, ws	83.93	97.07	97.92	97.6	**99.99**
5 cl, nws	86.06	93.64	93.95	94.96	**97.16**

ground truth labels and the number of components are known. Table 3.2 presents the comparison[3] based on clustering accuracy.

From the results in Table 3.2, it is evident that BSC-vMFMM provides the best clustering accuracy. Particularly, for the not-well separated samples BSC-vMFMM performs notably better than others.

3.4.1.3 Hierarchical Agglomerative Clustering (HAC) for Model Generation

Sided distance, centroid type and linkage criteria

Following Garcia and Nielsen (2010), we evaluate appropriate BD types (left /right /symmetric) and linkage criteria (ex: single, complete, average, etc.) w.r.t. the KLD and number of components. Note that, the choice of centroid type should correspond to the type of BD. In the Table 3.3 and Fig. 3.5, we present results from a vMFMM with well separated 7 components.

[3]In order to compare different methods, we used MATLAB implementation provided either by the authors (SPKM and soft-MoVMF) or by standard toolbox (GMM). For the k-means-directions algorithm (KMDR), we used the available R package ***skmeans*** (Buchta et al., 2012).

Table 3.3: Numerical evaluation using cophenetic correlation coefficient. Each table entry indicates the evaluated value for a particular choice of BD type and linkage criteria.

Linkage type	**Left-sided**	**Right-sided**	**Symmetric**
Single	0.4594	0.5212	0.4679
Complete	0.4051	0.4109	0.4135
Average	**0.5297**	**0.5231**	**0.5331**
Ward	0.4396	0.4455	0.4483
Weighted	0.4438	0.4497	0.4526
Median	0.4222	0.5171	0.4311
Centroid	0.4669	0.4715	0.4753

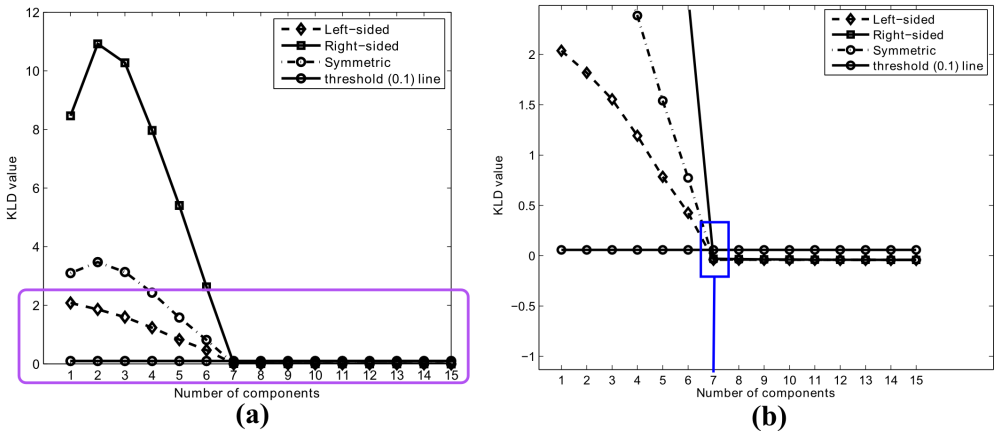


Figure 3.5: Evaluation of distance type and linkage criteria. (a) Average KLD values for different types of distances. Linkage criteria: 'average link'. KLD threshold value: 0.1. (b) a closer view is provided for the selected rectangular area in the left image.

First, we select the linkage criteria. To this aim, we compute cophenetic correlation coefficient (Martinez et al., 2010). Table 3.3 presents the numerical evaluation, which indicates that the 'average linkage' is the best choice (i.e., the highest cophenetic correlation coefficient).

Fig. 3.5 illustrates the results obtained for evaluating the types of divergences. Here the KLD value among $\Theta_{k_{max}}$ and $\{\Theta_k\}_{k=k_{min},\ldots,k_{max}-1}$ is used as a measure (lower is better) of quality. See Garcia and Nielsen (2010) for details of this evaluation criterion. Our experiments reveal that the left-sided BD provides the best simplification quality for the data sampled from a vMFMM with well separated 7 components.

We applied these experiments on all simulated data (see Section 3.4.1.1). Indeed, for all mixture models we observe the same behavior. Therefore, we choose the 'left-sided' BD with the 'average-link' as the linkage criteria for our HAC method.

Table 3.4: Comparison of MBC-MoVMF and MBC-vMFMM.

	MBC-MoVMF	**MBC-vMFMM**
Initialization & EM	Soft-MoVMF (Banerjee et al., 2005a)	BSC
Objective of HAC	Min Entropy (Baudry et al., 2010)	Min BD
Parameter Estimation for HAC merged clusters	Single step EM + Heuristic app. (Banerjee et al., 2005a)	Centroid averaging
Component Annihilation	Yes	No

Comparative evaluation

To the best of our knowledge, it does not exist model based clustering method for vMFMM. However, for the purpose of comparison we follow the state of the art and combine MBC-GMM (Baudry et al., 2010) method with soft-MoVMF (Banerjee et al., 2005a). Let us call this method the MBC-MoVMF and our method the MBC-vMFMM (see Section 3.3.1) for further uses. A methodological comparison among the two methods is presented in Table 3.4. To experiment with both methods, we set $k_{max} = 15$ and provide the true number of components. Note that, we apply component annihilation (Figueiredo and Jain, 2002) for MBC-MoVMF method. This annihilation takes place inside the EM algorithm (soft-moVMF) that we apply immediately after HAC (based on entropy minimization) step. We annihilate a component if its probability is close to zero (e.g., less than 0.0001). The annihilation strategy allows the algorithms to avoid from approaching towards the boundary of the parameter space. Additional advantages observed due to following this strategy are: (i) reduce the number of EM iterations and hence speed up the convergence and (ii) allows skipping several merging steps of HAC and hence reducing computational time.

Next, we perform numerical evaluation (Table 3.5) based on the accuracy of the classification and computational time. For the experiments we used MATLAB on a 64 bit machine with Intel(R) Xenon(R) CPU and 16 GB RAM.

We observe from these results that, the proposed MBC-vMFMM outperforms MBC-MoVMF with both evaluation criteria. Specially, we observe that the MBC-MoVMF is $\sim$3 times slower than the MBC-vMFMM.

Recall that, the MBC-vMFMM employs HAC to estimate the mixture models $\{\Theta_k\}_{k=k_{min},\ldots,k_{max}-1}$ from the parameters of a principal model $\Theta_{k_{max}}$. This guarantees

Table 3.5: Evaluation of MBC based methods (M1: ***MBC-MoVMF***, and M2: ***MBC-vMFMM***) for vMFMM.

	Classific. Acc (%)		**Comp. Time (sec)**	
	M1	**M2**	**M1**	**M2**
3,ws	87.913	**99.992**	8.9187	**2.953**
5,ws	84.487	**99.995**	8.1757	**2.9494**
7,ws	76.991	**99.994**	7.8314	**2.8663**
3,nws	93.788	**99.039**	10.74	**2.9201**
5,nws	90.012	**97.156**	8.6715	**2.9004**
7,nws	80.709	**92.966**	7.9239	**2.8822**

(unlike Maitra and Ramler (2010)) the structural relations, i.e., consistency of the cluster centers among the mixture models with different k. Moreover, this makes MBC-vMFMM faster as it does not incorporate the data points and an iterative procedure to estimate $\{\Theta_k\}_{k=k_{min},\ldots,k_{max}-1}$. However, to observe the effect of model estimation from the data, we include an additional EM step in MBC-vMFMM just after parameter estimation by HAC. Results effectively show that the performance remains almost same while increases a fraction of computational time.

In order to observe the results of estimating models with and without the HAC, we compare results from MBC-vMFMM in Table 3.5 and BSC-vMFMM in Table 3.2. In both cases, the true numbers of components are given as input. Results show that the difference in clustering accuracy is insignificant. However, let us recall that only the MBC-vMFMM permits to proceed towards model selection.

3.4.1.4 Model Selection

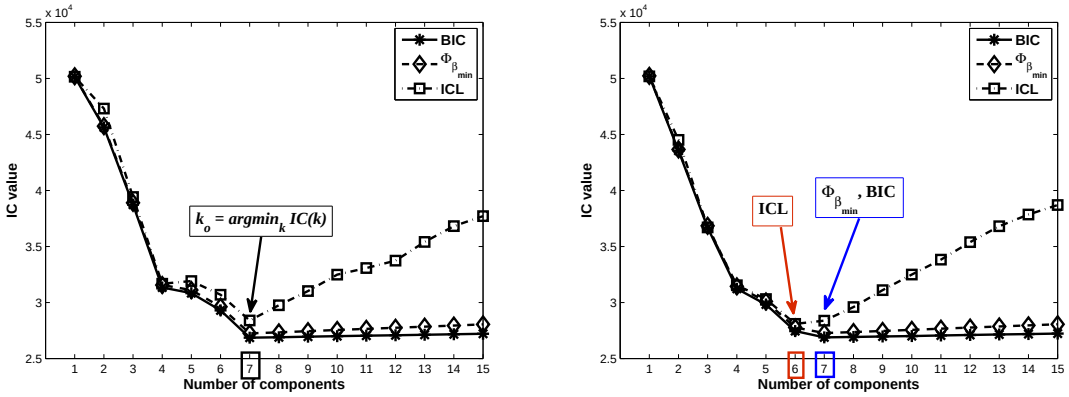
KLD based approach

In this approach, a simplified mixture model is obtained with a user defined threshold value (Garcia and Nielsen, 2010). Fig. 3.5 gives an idea of how to select such threshold. Experimentally we observe that, for the well separated samples, a very small threshold value ($\simeq 0.01$, see Fig. 3.5) perfectly selects the correct number of components. However, this is not trivial for the not-well separated samples. Therefore, for these samples, we learn the threshold from the ground truth data. To this aim, we did experiments using simulated data with different amount of samples (2k, 5k, 10k, 20k, 50k) and different numbers of components (3, 5, 7).

Table 3.6 presents the learned threshold values, which shows that a single threshold is not applicable in all cases. This implies that, the user must choose different thresholds for different number of components, which is impractical. Therefore, the

Table 3.6: Empirical thresholds obtained from learning threshold values from simulated data.

Num. classes	**3**	**5**	**7**
Th. Value	0.1	0.07	0.05



(a) (b)

Figure 3.6: Graphical representation for component selection with different criteria. Arrows indicate the selected number of components. The data for clustering was sampled from two vMFMMs of 7 components (see Section 3.4.1.1) where: (a) all criteria select the same number of components and (b) the selection is different from different criteria.

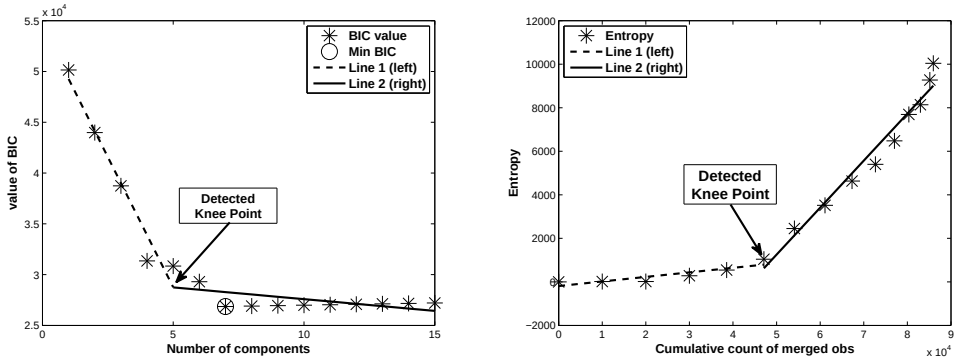
KLD based approach (Garcia and Nielsen, 2010) is not an appropriate choice for our MBC approach.

Parsimony based methods and Evaluation graph

From a wide selection of criteria for parsimony based approach (Melnykov and Maitra, 2010; Figueiredo and Jain, 2002), we select BIC, $\Phi_{\beta_{min}}$ and ICL. This selection is based on the observation (similar to Alata and Quintard (2009)) that other criteria (AIC and MML) do not provide significantly different results than BIC[4]. Fig. 3.6 illustrates two study cases of applying these criteria.

Fig. 3.7 illustrates two examples of the evaluation graph based methods applied on the same data used in Figure 3.6(a). From the results of the L-method (Fig. 3.7(a)) (Salvador and Chan, 2004) we observe that: (a) the fitted lines tend to underestimate the number of components and (b) it does not consider the fact that the BIC values change almost linearly after $k_o = 7$. Indeed, the L-method is a generalized proposal

[4]AIC overparameterize w.r.t. BIC. We are not giving the results with AIC as they are not better than those obtained with BIC



(a) (b)

Figure 3.7: Evaluation graphs and selected optimal numbers of components (k_o) by knee point detection approach (a) the '*L-method on BIC plot*' selects $k_o = 5$ and (b) the '*linear regression fit on rescaled entropy plot*' selects $k_o = 5$. The data was sampled from a vMFMMs of 7 components, i.e. $k_o = 7$. Clusters in the simulated data are not-well separated (see Section 3.4.1.1), which is similar to the data used in Figure 3.6(a).

and is not intended to analyze BIC plot. However, it shows an informative hint to exploit BIC plot in a better way.

Next, we analyze the rescaled entropy plot (Fig. 3.7(b)) (Baudry et al., 2010), fitted with linear regression. We observe that it underestimates k_o. In this plot, unlike BIC curve, it is not possible to find an appropriate reason for the underestimation by analyzing the entropy values.

The hint observed from the BIC curve (Fig. 3.7(a)) is also evident from the KLD plot in Fig. 3.5. The KLD plot shows that from k_{max} to k_o, the KL distance exhibits linear change. Such change can be fitted by linear regression with very small error. In contrary, the change from k_o to k_{min} is not equivalent and hence a linear regression fit produces comparatively higher error. Such phenomenon validates our approach to set higher weight on the right side, such that it is balanced in both side.

Table 3.7 presents numerical evaluation of the parsimony based and evaluation graph based methods for the simulated data (see Section 3.4.1.1). Let us denote the methods as: min BIC (BIC), min Φ_β ($\Phi_{\beta_{min}}$), min ICL (ICL), piecewise linear regression fit on rescaled Entropy plot (REP-LR) (Baudry et al., 2010), L-method (Lm) (Salvador and Chan, 2004), weighted linear regression fit on BIC plot, with $\tau = 1$ (WPLR-1) and with $\tau = 300$ (WPLR-300) and the k-means-directions algorithm (KMDR) (Maitra and Ramler, 2010). We observe (from Table 3.7) that, both $\Phi_{\beta_{min}}$ and WPLR-300 successfully determines the optimal number of components. Among

Table 3.7: Accuracy evaluation of different methods for determining the optimal number of components.

Well Separated samples								
	BIC	$\Phi_{\beta_{min}}$	**ICL**	**REP-LR**	**Lm**	**WPLR-1**	**WPLR-300**	**KMDR**
3	100	**100**	100	82	100	100	**100**	78
5	100	**100**	100	98	100	100	**100**	96
7	100	**100**	100	100	100	100	**100**	52
not-well Separated samples								
	BIC	$\Phi_{\beta_{min}}$	**ICL**	**REP-LR**	**Lm**	**WPLR-1**	**WPLR-300**	**KMDR**
3	100	**100**	100	78	100	16	**100**	96
5	100	**100**	100	84	96	10	**100**	92
7	92	**100**	24	2	0	0	**100**	22

the other methods BIC, ICL, Lm and WPLR-1 are accurate for the well separated samples. However, they are inconsistent for the not-well separated samples. The REP-LR and KMDR methods provide inconsistent results for both types of samples.

Let us concentrate more on the data samples from not-well separated 7 components, where most of the methods perform an underestimation. We compare Lm and WPLR-1 since for detecting 7 components mixture both are nearly same (for Lm, $\omega_r = 1.33$ and for WPLR-1, $\omega_r = \tau = 1$). Now, looking at Fig. 3.7(a) we realize that such small weight does not support the observation that "BIC values from k_{max} to k_o change linearly". And hence, higher weight should be imposed to obtain correct k_o. This is immediately evident from the result provided by WPLR-300 (in Table 3.7).

Now, from the perspective of determining the value of τ, we present additional results about the proposed WPLR-τ method (see Table 3.8). We see that, for $\tau = 1$, the number of components are underestimated; and the number of underestimations decreases with the increase of τ. Additionally, we see that the accuracy is stable after $\tau \geq 300$. Beside this, the results in Table 3.7 show that a single value of $\tau = 300$ successfully determines the correct number of components for the entire data-set (see Section 3.4.1.1) containing mixture of different numbers of components. This validates that, unlike the KLD threshold (see Section 3.4.1.4), a single value of τ is sufficient for a dataset. For different dataset and applications, we propose a two steps heuristic as:

1. Evaluate dataset with a range of τ values.

2. Select the minimum of τ values from which the evaluation is stable.

Table 3.8: Effect of τ for WPLR-τ method. Data for this experiments are sampled from not-well separated 7 components vMFMM. Each row presents the evaluations for a particular value of τ. Evaluation criteria: Correct (**Corr**), over estimation (**OE**) and under estimation (**UE**).

τ	**Corr**	**OE**	**UE**
1	0	0	100
10	18	0	82
20	24	0	76
50	24	0	76
100	42	0	58
200	96	0	4
300	100	0	0
400	100	0	0
500	100	0	0
800	100	0	0
1000	100	0	0

The above experiments and analysis reveal that the proposed MBC-vMFMM method successfully performs unsupervised classification of the simulated 3D directional data. It performs better than the state of the art methods in terms of classification accuracy and detecting the true number of clusters. In the next Section, we demonstrate an application of MBC-vMFMM for depth image analysis.

3.4.1.5 Depth Image Analysis

We consider the NYU depth dataset v2 (Silberman et al., 2012) for our experiments. It contains 1449 synchronized color and depth images of indoor environment. In this research, we consider only the depth images for experiments. Notice that, in Fig. 3.8 and 3.11 the color images are provided to show the readers the contents of the scene.

First, we analyze a depth image (see Section 3.3.2) with the KLD based approach. This helps us to understand the importance of selecting the correct number of components. Fig. 3.8 illustrates such an example. The KLD thresholds exhibit an inverse relation with the number of components. Therefore, we can interpret the clustered images from the perspective of increasing or decreasing the KLD threshold value. Increasing threshold is equivalent to merge image regions. This is evident when the threshold value increases from 0.19 to 0.2 (number of components decreases from 7 to 6). In contrary, decreasing threshold is equivalent to splitting the image regions.

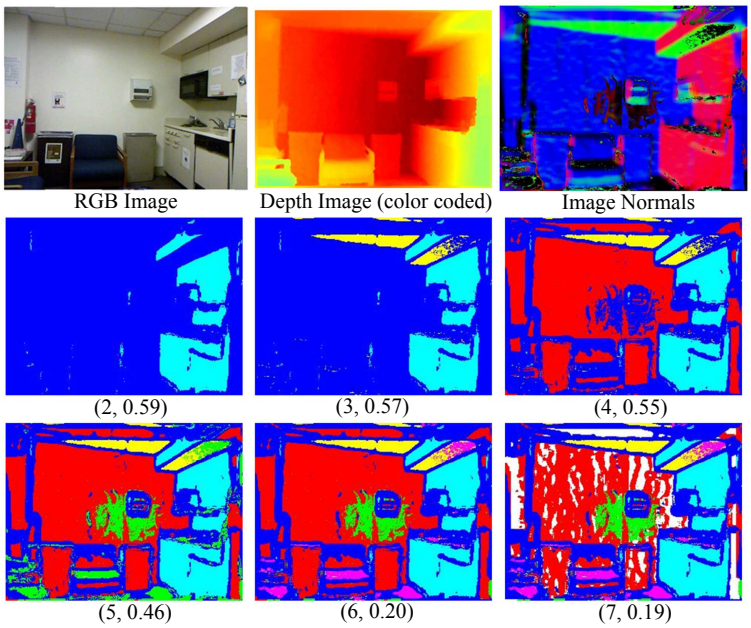
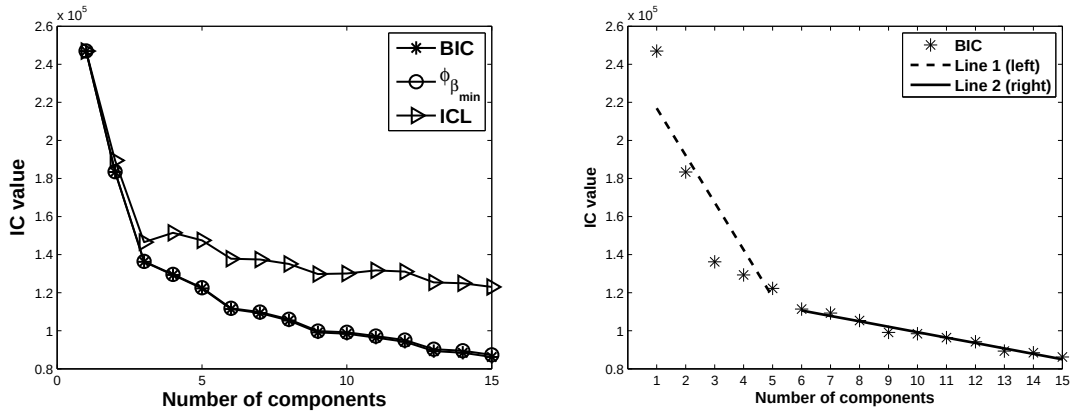


Figure 3.8: Resulting clusters generated for different numbers of components. Associated KLD threshold values are provided.

We observe from the results (Fig. 3.8) that, the best clustering provides sufficient semantic interpretation about the structure of the indoor scene. Most interestingly, it provides the three principal surfaces (planes in the indoor scene) when the number of components is 4. It appears that, the more we increase the number of components (starting from 2), the more we can detect the principal surfaces. However, increasing the number of components too much will enforce over-segmentation (evident from 7 clusters). Therefore, careful choice of the KLD threshold value is very important. On the other hand, based on the observation from Table 3.6 we can say that a unique threshold is not sufficient to provide the true number of clusters for all images. Rather it could create an over-segmentation or under-segmentation. Therefore, we can say that the KLD based approach is not appropriate in the context of unsupervised depth image clustering.

Next, we address the issue of automatically identifying the number of clusters in the depth image. For this reason, we apply parsimony based (BIC, $\Phi_{\beta_{min}}$ and ICL criteria) and evaluation graph based (WPLR-τ and L-method) model selection approaches. Let us consider $\tau = 30$ (based on Fig. 3.13(a)) since it exhibits a good compromise between the over-segmentation and under-segmentation. The plots of Fig. 3.9 and 3.10 illustrate the model selection experiments, where Fig. 3.9 shows details for a single image and Fig. 3.10 shows overall analysis for all images of the



(a) (b)

Figure 3.9: Number of clusters selection of a depth image (same image shown in Fig. 3.8) based on: (a) parsimony based (BIC, $\Phi_{\beta_{min}}$ and ICL criteria) and (b) evaluation graph based (WPLR-τ with τ=30) method.

NYU dataset (Silberman et al., 2012).

We begin with the analysis of a single image (same image of Fig. 3.8). We observe (in Fig. 3.9(a)) that, all the criteria favor the maximum number (i.e. $k_o = k_{max} = 15$) of clusters. This produces over-segmentation (see Fig. 3.8). However, Fig. 3.9(b) shows that WPLR-τ with $\tau = 30$ selects $k_o = 6$, which is the correct number of clusters according to our judgment. Hence, we see that, for this depth image the WPLR-τ method outperforms others.

Now, we evaluate WPLR-τ on the entire NYU database (Silberman et al., 2012). Fig. 3.10 illustrates details of the evaluation. We see that BIC and $\Phi_{\beta_{min}}$ criteria tend to choose a higher number of clusters, which indeed over-segment the images (based on Fig. 3.8). We observe opposite scenario from the L-method, which tends to under-segment the images. The ICL criterion provides a combination of both cases (over and under segmentation). Lastly, let us analyze the performance of our proposed WPLR-τ method with $\tau = 30$. We observe that, unlike other methods WPLR-30 does not perform over or under segmentation. This provides additional evidence that compare to other experimented methods the WPLR-τ shows better compromise both for the simulated and the real data. In order to further clarify this claim, either we need the associated ground truth for this particular image analysis task or we need an unsupervised depth image segmentation quality measure. Since none of these are available at present, we consider providing such evaluation as a future perspective of the proposed method.

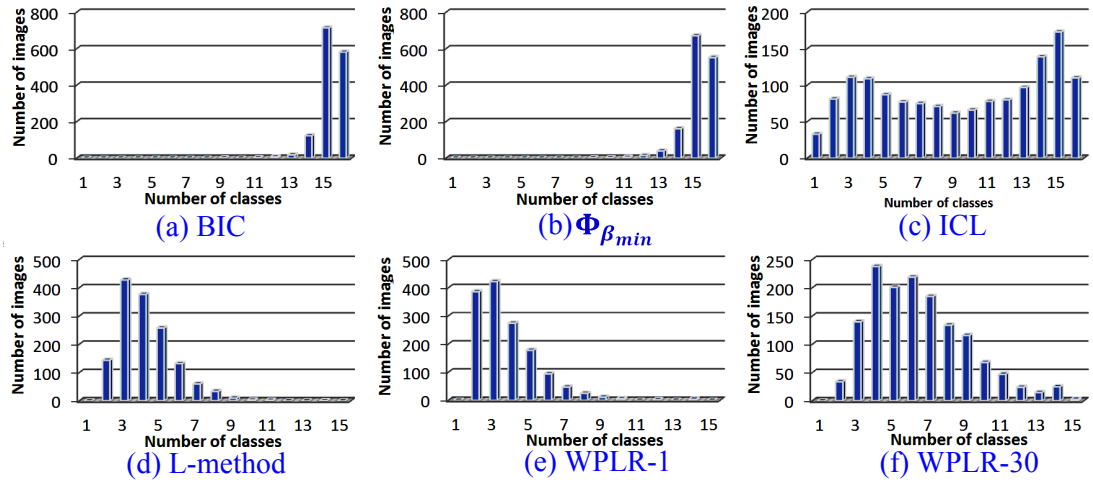


Figure 3.10: Details of the evaluation for selecting the number of components. Methods: min BIC (BIC), min Φ_β ($\Phi_{\beta_{min}}$), min ICL (ICL), L-method, weighted linear regression fit on BIC plot with τ=1 (WPLR-1) and with τ=30 (WPLR-30).

Fig. 3.11 illustrates additional image analysis results with MBC-vMFMM (with τ=30). We noticed that the computed normals contain noisy information, which affects the clustering result. This is evident from Fig. 3.8, where a new cluster appears around the "paper towel dispenser" if the number of components is 6 or more (see 3rd row). The source of noise is caused by the low accuracy of the depth information (addressed by Barron and Malik (2013)) and directional ambiguity of the computed normal (Rusu, 2013).

Now, we compare the MBC-vMFMM w.r.t. the state of the art. Among the most relevant methods for unsupervised image analysis (see Section 5.3 of Szeliski (2011)), we select K-means (KM), Gaussian Mixture Model (GMM) and Mean shift (MS). While MBC-vMFMM, KM and GMM are parametric methods, MS is non-parametric (Szeliski, 2011). Fig. 3.12 illustrates a comparison with settings: $k = 6$ (for KM and GMM), $\tau = 30$ (for MBC-vMFMM) and $bandwidth = 0.5$ (for MS). We observe that, KM and MS methods generate nearly same result, which is smoother and hence visually more pleasing. However, they do not always respect the true nature of the directional data. For example, the pixels which belong to the corners have different normal directions and should form a separate cluster. Interestingly, such clusters often able to characterize the corners and edges. We see that, while KM and MS do not identify such clusters, MBC-vMFMM and GMM can do. However, results from GMM are noisier. This is intuitive since in a unit sphere S^2 data should be explained with concentration (Mardia and Jupp, 2009) rather than ellipsoids in $\mathbb{R}^3$.

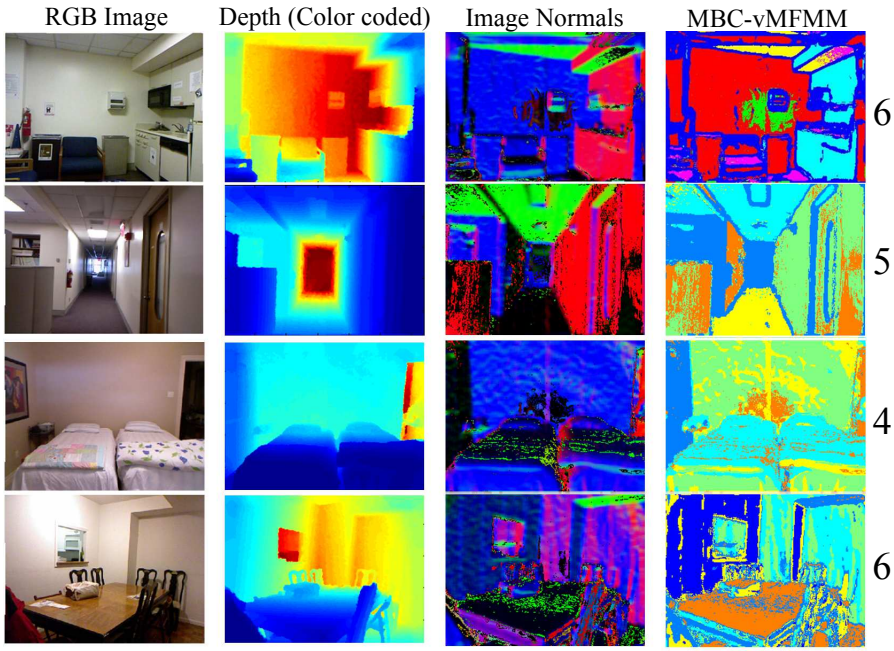


Figure 3.11: Illustration of clustering of the depth images obtained by applying MBC-vMFMM with $\tau = 30$. The last column indicates the associated number of clusters.

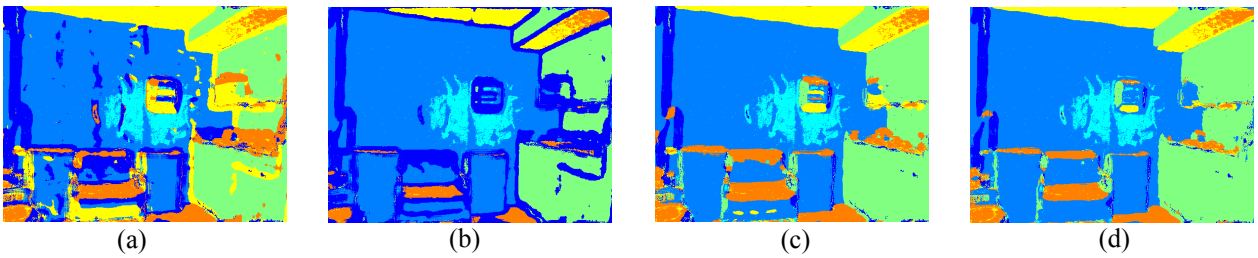
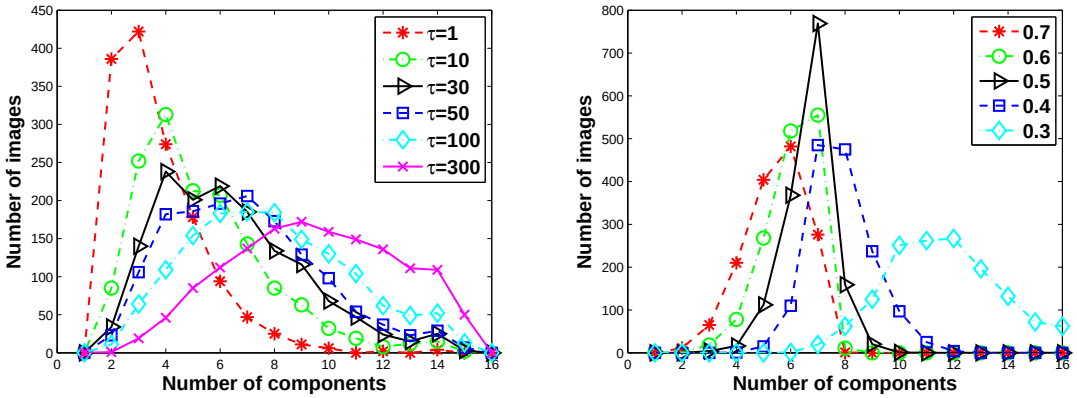


Figure 3.12: Comparison of depth image clustering generated by different methods. (a) GMM with 6 components; (b) MBC-VMFMM with $\tau = 30$; (c) K-means with 6 components and (d)Mean Shift with *bandwidth* $= 0.5$.



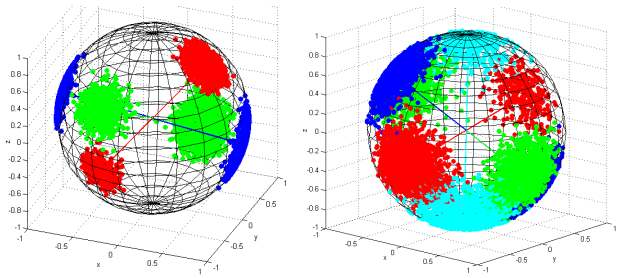
(a) MBC-vMFMM, WPLR-τ (b) Mean-Shift

Figure 3.13: Comparison of component selection with MBC-vMFMM and Mean Shift clustering methods. (a) Effect of τ for WPLR-τ method applied on MBC-vMFMM clustering method and (b) Effect of ***bandwidth*** parameter of the Mean Shift clustering method. For evaluation, we explore different values of τ and ***bandwidth*** parameters.

Parametric methods employ different strategies to automatically identify the number of components. However, a common strategy applicable for all purposes is yet to become available. We propose WPLR-τ method, which shows better compromise for the simulated and the real data. The Mean Shift is a well known non-parametric method, that automatically determines the number of clusters. However, it needs an input for the bandwidth parameter. This is similar to the τ (weight of right sided fitted line) parameter of our proposed (WPLR-τ) method. From Fig. 3.13 we observe that, the τ parameter has an inverse relationship with the bandwidth. Moreover, if we increase τ gradually, then the clustering method moves from generating under-segmentation to over-segmentation. It is balanced in the middle for certain values of τ. We observe similar phenomenon for the Mean Shift method, when the bandwidth decreases gradually.

3.4.2 Model Based Clustering with Watson Mixture Model (MBC-WMM)

To evaluate MBC-WMM, first we use simulated data samples for comparing it w.r.t. the state of the art methods. Next, we apply it on real depth image data samples.



(a) (b)

Figure 3.14: Synthetic data samples from WMM (a) Well separated (ws) 3 classes and (b) Not-well separated (nws) 5 classes.

3.4.2.1 Evaluation with Simulated Data Samples

In order to generate simulated data, we draw a finite set of axially symmetric 3D unit vectors ($d = 3$) from the Watson mixture models with different numbers of components. For this reason, we modified the standard sampling method proposed by Dhillon and Sra (2003). We generate 100 sets of data from two types of samples: (a) well separated (***ws***) and (b) not-well separated (***nws***). Each set consists of 10,000 identically and independently distributed samples. Fig. 3.14 illustrates an example of different types of samples.

The MBC-WMM method requires the setting of parameters and criteria, such as setting: (a) k_{max} and convergence criteria for the BSC-WMM algorithm and (b) the distance type, linkage criterion and centroid type for the Hierarchical Agglomerative Clustering (HAC) algorithm. We set $k_{max} = 10$ and the convergence criteria of BSC-WMM method is set same as the criteria of BSC-vMFMM method, see Section 3.4.1.2. Similar to the experiments in Section 3.4.1.3, we empirically find the setting of the HAC method for MBC-WMM which is: 'left sided' distance measure, 'average link' criterion and 'left sided' centroid.

To evaluate MBC-WMM (without component selection) w.r.t. the state of the art methods, we begin with a comparison of the average clustering accuracy (in %) which is presented in Table 3.9. From the results, we observe that MBC-WMM provides best average clustering accuracy. We also notice in Table 3.9 that EM-moW (Sra and Karp, 2013) is very competitive. However, we see from Table 3.11 that, performance of EM-moW (Sra and Karp, 2013) decreases significantly when it is included in the model based clustering framework.

Table 3.9: Comparison of clustering accuracy (in %). Experimented on several numbers (2 - 5) of classes and two types (*ws* and *nws*) of samples. Methods: diametrical (DM) (Dhillon and Sra, 2003), EM-Watson (EMW) (Bijral et al., 2007), EM-moW (Sra and Karp, 2013) and MBC-WMM.

	DM	**EM-W**	**EM-moW**	**MBC-WMM**
2, ws	99.99	99.99	**100**	**100**
3, ws	99.04	98.05	**99.99**	**99.99**
4, ws	93.26	98.13	**99.99**	**99.99**
5, ws	94.65	96.35	**99.96**	**99.96**
2, nws	97.17	97.22	**97.22**	**97.22**
3, nws	95.63	95.66	**96.4**	94.35
4, nws	97.93	95.21	96.28	**98.06**
5, nws	96.03	93.63	94.2	**96.09**
Avg.	96.71	96.78	98	**98.21**

To the best of our knowledge, no model based clustering method exists for the Watson mixture model. However, for the purpose of comparison, we follow similar strategy as in Section 3.4.1.3 and combine state of the art methods to perform model based clustering. We combine: (a) the diametric clustering method (Dhillon and Sra, 2003), for initialization; (b) the EM-Watson (Bijral et al., 2007) or the EM-moW (Sra and Karp, 2013) method, for parameter estimation and (c) the entropy based cluster merging approach (Baudry et al., 2010), for hierarchical merging of clusters. Let us call these methods the MBC-EMW (with EM-Watson (Bijral et al., 2007)) and MBC-MOW (with EM-moW (Sra and Karp, 2013)) for further uses. A methodological comparison among these methods is presented in Table 3.10. Table 3.11 presents a numerical evaluation of these methods based on clustering accuracy (in %) and computation time (in sec). For the experiments we used MATLAB in a 64 bit machine with Intel Xenon CPU and 16 GB RAM. The average accuracy and computation time (bottom row of Table 3.11) show that the MBC-WMM is better in both cases.

Now, let us focus on selecting the number of components using the methods discussed in Section 2.5 of Chapter 2. We begin with the KLD based approach for component selection and observed similarities with the MBC-vMFMM method presented in Section 3.4.1.4. We do not proceed with this approach because: (a) it requires the threshold as an external input and (b) the learned threshold values change for different number of components, which is impractical to fix in real applications. Next, we

Table 3.10: Methodological comparison of MBC-EMW, MBC-MOW and MBC-WMM.

	MBC-EMW	**MBC-MOW**	**MBC-WMM**
Initialization	Diam. clust. (Dhillon and Sra, 2003)	Diam. clust. (Dhillon and Sra, 2003)	Diam. clust. (Dhillon and Sra, 2003)
EM	EM-Watson (Bijral et al., 2007)	EM-moW (Sra and Karp, 2013)	BSC
Obj of HAC	Min Entropy (Baudry et al., 2010)	Min Entropy (Baudry et al., 2010)	Min BD
Param. Est.	Heuristic (Bijral et al., 2007)	Closed form (Sra and Karp, 2013)	Centroid avg.

Table 3.11: Numerical evaluation of MBC methods (M1: MBC-EMW, M2: MBC-MOW and M3: MBC-WMM)

	Classific. Acc (%)			**Comp.Time (sec)**		
	M1	**M2**	**M3**	**M1**	**M2**	**M3**
Well Separated						
2	100.00	100.00	100.00	6.76	66.50	8.52
3	92.00	82.00	99.99	8.01	201.17	7.17
4	91.12	86.35	99.99	8.52	355.48	7.18
5	87.36	81.51	99.96	9.21	110.14	8.40
Not well Separated						
2	97.36	97.37	97.22	10.58	307.72	7.94
3	93.47	96.19	94.35	14.11	386.26	7.41
4	96.07	95.70	98.05	12.70	135.51	8.50
5	94.68	88.11	96.09	10.69	229.95	8.05
Average	94.01	90.90	**98.21**	10.07	224.09	**7.89**

evaluate different model selection criteria as in Section 3.4.1.4. Table 3.12 presents the rate of correct components selection by different methods. According to the average rate (bottom row of Table 3.12) of correct components selection, the L-method (Salvador and Chan, 2004) provides the best results. From detail results we observed that, the BIC and Φ_β criteria often over-estimate the number of components in comparison with ICL criterion (similar to Alata and Quintard (2009)). Based on these observations, we consider the ICL and the L-method for the further experiments.

Table 3.12: Evaluation of the rate of correct components selection by different methods: min BIC (BIC), min Φ_β ($\Phi_{\beta_{min}}$), min ICL (ICL), L-method (Lm) and Weighted Linear Regression Fit on BIC plot, with $\tau = 1$ (WPLR-1).

Num of cl	**BIC**	Φ_β	**ICL**	**Lm**	**WPLR-1**
Well Separated					
2	0	0	100	100	100
3	62	98	100	100	100
4	88	88	88	88	88
5	26	42	100	98	98
Not well Separated					
2	80	90	98	100	100
3	34	36	86	92	94
4	82	84	100	100	82
5	46	46	66	68	60
Average	52.25	60.5	92.25	**93.25**	90.25

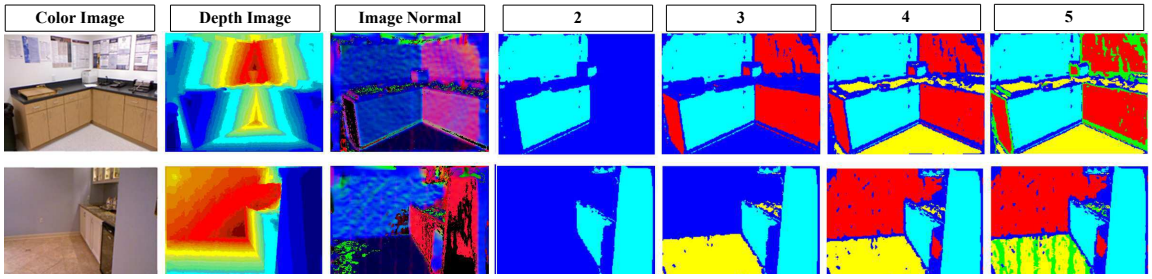


Figure 3.15: Illustration of depth image analysis for different numbers of clusters obtained by applying MBC-WMM method.

3.4.2.2 Evaluation of Depth Image Analysis

We follow the method described in Section 3.3.2 and apply MBC-WMM on the surface normals. The setting for MBC-WMM is the same as for simulated experiments, except we set $k_{max} = 12$. We conduct experiments with the depth images from NYU Depth Dataset V2 (NYUD2) (Silberman et al., 2012). It is worth mentioning that, due to axially symmetric property of Watson distribution, MBC-WMM can handle the noise or directional ambiguity in the surface normals (Rusu, 2013). Moreover, this causes the segments from MBC-WMM to be smoother. Therefore, for depth image analysis, MBC-WMM is more suitable over MBC-vMFMM method in case of the existence of noisy normals.

Fig. 3.15 illustrates the results of applying the MBC-WMM method (without component selection) on two depth images. For brevity let us denote k as the number of clusters. From the results we observe that, for a particular choice of k, the method

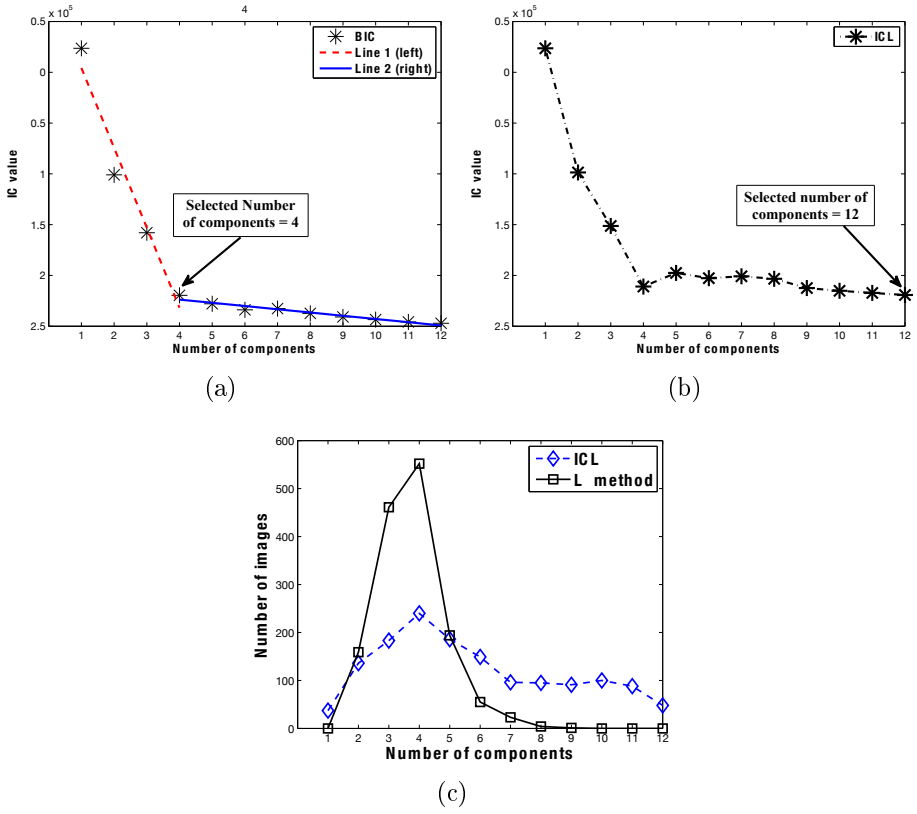


Figure 3.16: Selection of the number of components using: (a) L-method and (b) ICL criterion for a depth image (first row of Fig. 3.15). (c) Evaluation of components selection from NYU database (using both methods).

identifies different image regions w.r.t. the dominant (in terms of total number of pixels) axes of the scene. We see that, the identified regions represent piecewise planar surfaces (associated with a particular axis) of the scene. For example, when $k = 2$ it provides the plane which belongs to the first dominant axis. Similarly, it identifies the planes belonging to other axes for $k = 3$ and 4. Notice also that, one of the k clusters represents the normals which do not belong to a dominant axes. Let us denote this cluster as the Non-Dominant-Axial (NDA) cluster. Often a NDA cluster indicates the presence of non-planar objects such as corners of indoor surface, inhomogeneous shaped objects, noise, etc. Therefore, one could exploit the NDA clusters to discover additional (other than planar) category of objects.

Next, we evaluate MBC-WMM to select k automatically. Our component selection strategy can be explained with Fig. 3.16(a) and 3.16(b) which correspond to the first (top row) depth image shown in Fig. 3.15. The plots show that the L-method (using BIC plot) selects $k = 4$ and the ICL criterion selects $k = 12$. Based on our subjective (w.r.t. the axes) and visual observation we can verify that the L-method is correct.

On the other hand ICL over-estimates the k. Next, we evaluate component selection on the entire NYU database. Fig. 3.16(c) illustrates the results. Let us observe that ICL selects components on the entire range (1 to $k_{max} = 12$). This indicates (based on Fig. 3.16(b)) that ICL performs a large number of over-segmentation. In contrary (based on Fig. 3.16(a)), L-method performs better for selecting k (1 to 8). Therefore, we can justify that L-method is the right choice for the objectives of our analysis with MBC-WMM.

Additional results are given in Fig. 3.17. Let us note that, depending on the contents of images studied, MBC-WMM selects different k for different images. From these results we identify two cases about the NDA clusters. In the first case (case-1), the NDA cluster merges with one of the dominant clusters (see c, d, h, and j, Fig. 3.17). In the second case (case-2), the NDA cluster appears as an independent cluster (see a, b, e, f, g, i, k and l, Fig. 3.17). From our analysis over the entire database, we observed that case-1 occurs when the number of NDA data points is significantly lower (i.e., prior probability of NDA cluster is very low). Such low probability allows MBC-WMM to ignore the NDA cluster and merge it with a dominant cluster. However, one could find such NDA cluster in MBC-WMM method by looking at the next level of the hierarchy of mixture models. Therefore, from a theoretical standpoint MBC-WMM method can characterize the dominant planes as the clusters with high concentration and NDA as the cluster with low concentration.

Now, let us focus on the clustered depth images with higher values of k (see k and l, Fig. 3.17). We identify two cases: (1) more than one NDA cluster (see k) and (2) over-segmentation (see l). While case-1 is acceptable, case-2 (a degenerated case) highlights the necessity to pay more attention on component selection. In order to face this issue, we suggest to pre-process (e.g., spatial filtering) the image normals (which we did not apply) and hence further improve the efficiency of the depth image analysis using the proposed MBC-WMM method.

One should also notice the effectiveness of MBC-WMM method to handle the directional ambiguity of image normals (see 3^{rd} column of Fig. 3.17). The results show that although there is a significant amount of noise (due to low accuracy of depth sensor and incorrect surface normal directions), MBC-WMM could be used to identify the planar and non-planar surfaces in an unsupervised way.

Besides the above analysis, we study the planar statistics of the regions of the images from NYUD2 (Silberman et al., 2012). These regions are obtained using the MBC-WMM method. Each region is associated with a cluster of surface normals. Such cluster can be interpreted with the concentration parameter (κ) of the associated

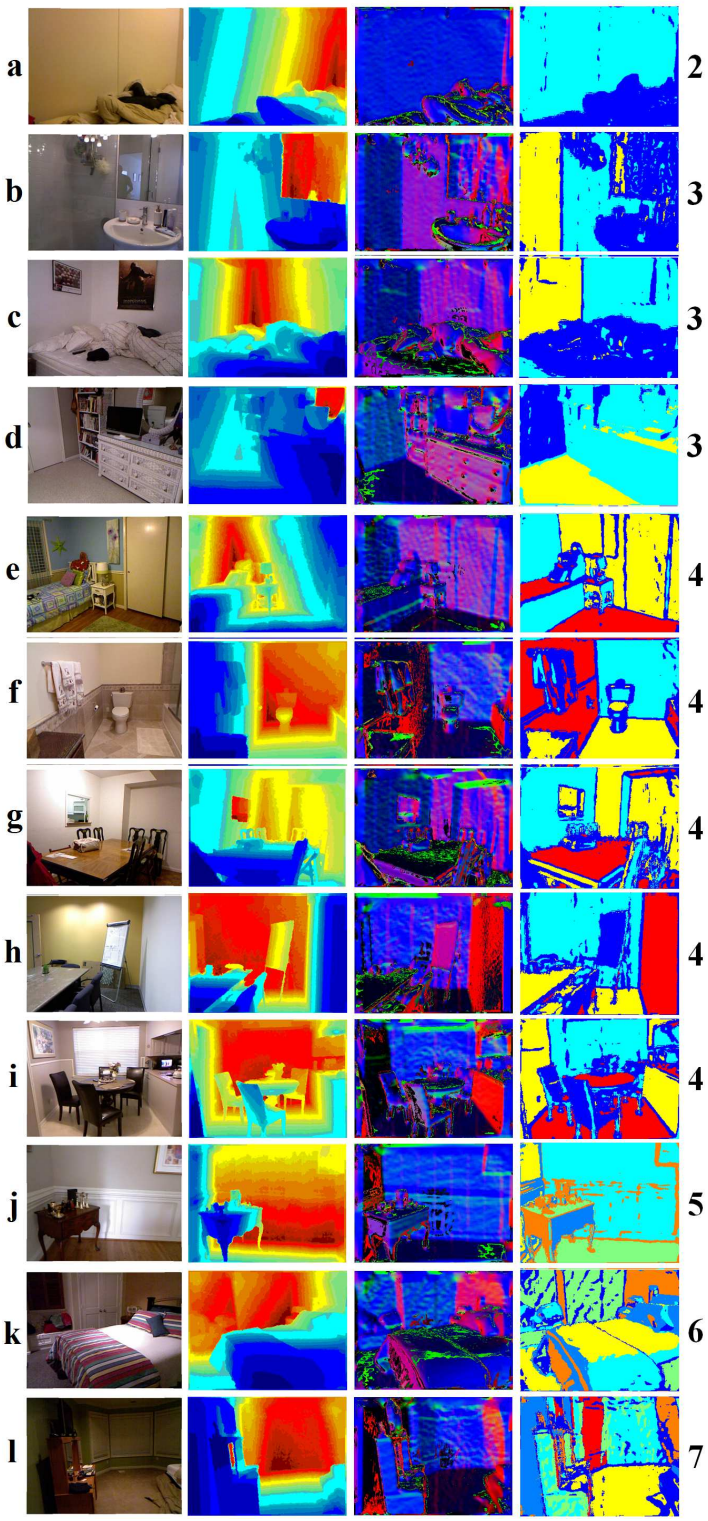


Figure 3.17: Depth image analysis with MBC-WMM method. Results obtained for several images of NYU database (Silberman et al., 2012). The right most column indicates the number of clusters.

Watson distribution. We particularly study the values of κ of the regions with the aim to distinguish between planar and non-planar regions.

Fig. 3.18 illustrates the histograms of κ (concentration of surface normal) values for the planar and non-planar surfaces. These histograms are obtained from an analysis of four category of segmented surfaces: (1) planar; (2) non-planar ; (3) planar + non-planar and (d) unknown (category not sure). We asked an analyst to categorize total 5410 segments obtained from the depth images into one of the four above-mentioned category. After categorizing the segments, we found 2559 segments as planar and 793 segments as non-planar. Then we construct the histogram from the κ values associated to these categories. Besides analyzing the histograms, we also observed that 99.88% of the planar surfaces has $\kappa > 5$ and 99.5% of the non-planar surfaces has $\kappa < 5$. This provides an interesting observation that the planar property of the regions can be characterized with the κ values. In the next Chapter, we will see how we can efficiently exploit this observation to design a semantic scene analysis method.

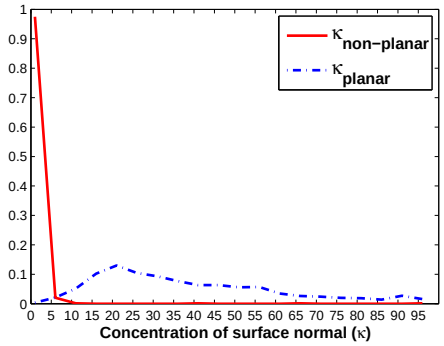


Figure 3.18: Histogram of κ values for planar and non-planar surfaces.

We believe that, if ambiguities and noises are absent in the computed normals, the analysis and discussions above will also be applicable for analyzing depth images with MBC-VMFMM method.

3.5 Discussions and Conclusions

Let us now discuss and summarize the contributions and future perspectives of the research presented in this Chapter. We proposed novel Model Based Clustering (MBC) methods with two directional distributions, called MBC-vMFMM and MBC-WMM. These methods perform unsupervised clustering of the directional and axial data which are in the form of unit vectors. The proposed methods assume a generative

model and exploit Bregman Divergence within the MBC framework. According to our knowledge, no such method exists for directional distributions. Moreover, observing the individual elements of the method, we can highlight several key contributions for directional statistics:

- An efficient soft clustering method, based on Bregman soft clustering (Banerjee et al., 2005b), with the vMFMM and the WMM.
- A hierarchical mixture model (Goldberger and Roweis, 2004) generation method that can be used for simplifying (Garcia and Nielsen, 2010) the vMFMM and the WMM.
- A hybrid MBC method (Zhong and Ghosh, 2003) for the vMFMM and the WMM (by combining Bregman soft clustering and hierarchical agglomerative clustering).

We evaluated these methods first with synthetic data. Results show that they are relevant for clustering directional and axial data. Moreover, they perform better than the state of the art in terms of: (a) accuracy of clustering; (b) rate of correct selection of the optimal number of components and (c) computational efficiency. In practice, we also applied them to cluster image normals with the goal of analyzing real depth images. Results show that, as an unsupervised method they are able to detect and discriminate the planar and non-planar surfaces. Therefore, we show that these methods are also relevant to provide semantic (planar/non-planar) interpretation of indoor scenes using only directional features. There are several future perspectives of the proposed methods:

- Develop model based clustering method (Fraley and Raftery, 2007) with the Kent and Bingham distributions (Mardia and Jupp, 2009). Such development will allow us to model complex structure of the data with more parameters, e.g., the Kent distribution can model data with an elliptical shape whereas the vMF and Watson distributions model data with circular shape. This can be done by deriving Bregman Divergence (Banerjee et al., 2005b) for these distributions.
- Extend the MBC-vMFMM method for high dimensional data, as currently it is limited for 3D data only.
- Include pre-processing and post-processing (e.g., spatial filtering and regularization) to extend the methods, such that they can be used for semantic depth

image segmentation. However, knowing the properties of directional features we should not expect a complete semantic categorization.

- Extend these methods such that they can incorporate additional features (e.g., color) and that they can cluster heterogeneous data. Eventually, extend these methods for joint color and depth (RGB-D) image analysis, see next Chapter.

- Beside image analysis, apply these methods to cluster data from different domains, such as speech (Souden et al., 2013; Vu and Haeb-Umbach, 2010), gene expressions (Sra and Karp, 2013), digits (Bijral et al., 2007), etc.

Number of components selection is yet a challenging problem in clustering and no single method is found to be the best for all purposes. We propose WPLR-τ for vMFMM and select L-method (Salvador and Chan, 2004) for MBC-WMM. They provided satisfactory results for the experiments with synthetic and real data. However, it would be interesting to compare them with other methods, such as the Dirichlet Process Mixture Model (DPMM) (Murphy, 2012). This could be another perspective for the methods presented in this Chapter.

Chapter 4

Unsupervised RGB-D image segmentation using joint clustering and region merging

Résumé: Des avancées récentes dans le domaine des capteurs, comme la caméra Kinect de Microsoft, donnent accès à des données couleur et des données de profondeur synchronisées, appelées images RGB-D. Dans ce chapitre, nous exploitons les méthodes et les observations des précédents chapitres afin de proposer une méthode non supervisée de segmentation d'images RGB-D de scènes intérieures. La nouvelle méthode est basée sur un modèle génératif d'image exploitant la couleur et la géométrie de la scène: elle réalise une classification jointe de données couleur, spatiales et axiales, puis une méthode de fusion de régions de géométrie plane. Nous évaluons la méthode sur la base de données de profondeur NYU et nous la comparons aux méthodes existantes de segmentation non supervisée de données RGB-D. Les résultats obtenus montrent que la nouvelle méthode donne des résultats comparables aux méthodes de l'état de l'art tout en demandant un temps de calcul inférieur. De plus, elle ouvre des perspectives intéressantes pour fusionner des informations couleur et géométriques de manière non supervisée.

Recent advances in imaging sensors, such as Microsoft Kinect camera, provide access to the synchronized depth with color information, called RGB-D image. In this Chapter, we exploit the methods and observations from previous Chapters and propose an unsupervised method for indoor RGB-D image segmentation and analysis. The proposed method considers a statistical image generation model based on the color and geometry of the scene. It consists of a joint color-spatial-axial clustering method followed by a statistical planar region merging method. We evaluate the

method on the NYU Depth Database and compare it with existing unsupervised RGB-D segmentation methods. Results show that, it is comparable with the state of the art methods and it needs less computation time. Moreover, it opens interesting perspectives to fuse color and geometry in an unsupervised manner.

4.1 Introduction

Segmentation is considered as one of the oldest and most widely studied problems in image analysis and computer vision. The central goal of this task is to group perceptually similar pixels based on certain features (e.g., color, texture etc.) in an image, which are based on human perception according to the Gestalt theory in psychology (Nock and Nielsen, 2004). This problem has been addressed from many different perspectives and therefore a variety of different techniques are available in literature (Szeliski, 2011). In this Chapter, we address the problem of segmenting synchronized color and depth images from indoor scene and propose a solution that combines a clustering method (Murphy, 2012) with a statistical region merging technique (Nock and Nielsen, 2004).

After the introduction of Microsoft Kinect camera, the availability and accessibility of RGB-D images is widespread now. As a consequence, traditional computer vision algorithms which are previously developed for color/intensity image, have been enhanced to incorporate depth information (Han et al., 2013). Notable progress have been reported on RGB-D image segmentation of indoor scenes (Gupta et al., 2013; Taylor and Cowley, 2013; Silberman et al., 2012; Ren et al., 2012; Dal Mutto et al., 2012a; Koppula et al., 2011). These researches have shown that depth as an additional feature improves accuracy of scene segmentation. Most of the techniques address the problem with supervised approaches (e.g., Gupta et al. (2013)). In contrary, unsupervised approach (e.g., Dal Mutto et al. (2012a)) to accomplish this task remains underexplored. Moreover, it remains an important issue - what is the best way to fuse color and geometry in an unsupervised manner? This motivates us to conduct further research and contribute towards unsupervised indoor RGB-D image segmentation or scene labeling with the aim to improve the performance of the task. In this Chapter, we focus on this issue and propose a solution.

This Chapter proposes a scene segmentation approach which first identifies the possible image regions using a statistical image generation model. Then it merges regions based on the statistics associated to the planar property. The proposed model

is based on three different cues/features[1] of the RGB-D image: color, 3D location and surface normals. It follows generative model approach for these features in which they are issued independently (*âĂIJnaïveâĂİ Bayes* (Murphy, 2012) assumption) from a finite mixture of certain probability distributions. The model considers the Gaussian distribution (Murphy, 2012) for color and 3D features and the multivariate Watson distribution (mWD) (Mardia and Jupp, 2009) for surface normals. The use of mWD is motivated by the observations from Chapter 3 which are: (a) it overcomes the directional ambiguity and noise (Rusu, 2013) related to surface normals (b) it provides adequate statistics to explain the planar property of regions and (c) it helps us to develop a simple and effective region merging method.

Finite Mixture Models are commonly used for cluster analysis (Fraley and Raftery, 1998; Biernacki et al., 2000; Fraley and Raftery, 2007). In the context of image analysis and segmentation these models have been employed with the Gaussian distribution for clustering the color image pixels (Ma et al., 2007; Alata and Quintard, 2009; Garcia and Nielsen, 2010; Szeliski, 2011; Nguyen and Wu, 2013). These clusters are obtained by using the Expectation Maximization (EM) algorithm that performs Maximum Likelihood Estimate (MLE) of the model parameters (Murphy, 2012; Bishop, 2006). In Chapter 2 and 3, we presented efficient algorithms to estimate mixture models based on individual distributions from the Exponential families (Nielsen and Garcia, 2009), such as the Gaussian, the von Mises-Fisher and the Watson. In this Chapter, we propose a clustering method that combines a mixture model of multiple distributions from the Exponential families.

Bregman Soft Clustering (BSC) is a centroid based parametric clustering method (Banerjee et al., 2005b). It has been effectively employed to estimate parameters of the mixture models which are based on Exponential Family of Distributions (Garcia and Nielsen, 2010; Nielsen and Garcia, 2009). Compare to the traditional EM based algorithm, BSC provides additional benefits, see Chapter 2 for details related to this method. In this Chapter, we extend the BSC algorithm in order to perform efficient clustering with our proposed image generation model.

Image segmentation based on region merging is one of the oldest techniques in computer vision (Murphy, 2012). Existing methods which merge regions in a RGB

[1] Clustering using only 3D points often fails to locate the intersections among the planar surfaces with different orientations such as wall, floor, ceiling, etc. This is due to the fact that the 3D points associated to the intersections are grouped into a single cluster. On the other hand, the use of only normals groups multiple objects with nearly similar orientations into the same cluster irrespective of their 3D location. In order to overcome these limitations and to describe the geometry of indoor scenes, we take both features into account.

image exploit color and edge information (Trémeau and Colantoni, 2000; Nock and Nielsen, 2004; Peng and Zhang, 2011; Martínez-Usó et al., 2013). For indoor scenes, the use of color is often unreliable due to numerous effects caused by spatially varying illumination (Gupta et al., 2013) and the presence of shadows. Therefore, for indoor scenes color based merging is not as effective as it is for outdoor scenes. On the other hand, in indoor scenes the planar surfaces are considered as important geometric primitives. They are often employed for scene decomposition (Silberman et al., 2012; Rusu, 2013; Gupta et al., 2013) and grouping coplanar segments into extended regions (Taylor and Cowley, 2011). This motivates us to develop a region merging algorithm exploiting planar property of the regions rather than color. In Chapter 3, we observed that the concentration parameter (κ) of the directional distributions can be exploited for characterizing planar surfaces. In the proposed merging method, we efficiently exploit the concentration (κ) of the surface normals in order to accept or reject a merging operation.

In this Chapter, we present a novel RGB-D segmentation method. The proposed method first applies a joint clustering method on the features (color, position and normals) extracted from the RGB-D image. As an outcome of clustering, it obtains a set of regions. Next, it applies a statistical region merging method on the initially obtained regions to obtain the final segmentation. We evaluate the proposed method by applying it on RGB-D images of the NYU depth database (NYUD2) (Silberman et al., 2012) and compare the results with the state of the art unsupervised techniques. To benchmark the segmentation task, we consider commonly used evaluation metrics such as (Arbelaez et al., 2011; Freixenet et al., 2002): segmentation covering, probability rand index, variation of information, boundary displacement error and boundary F-measure. Moreover, we consider the computation time of comparable methods as a measure of evaluation.

Finally, the contributions related to the work developed in this Chapter can be highlighted as follows:

- A statistical RGB-D image generation model (section 4.3.1) that incorporates both color and geometric properties of the scene.
- Development of an efficient probabilistic joint clustering method (section 4.3.3) exploiting the Bregman divergence (Banerjee et al., 2005b). It has following properties: (a) performs clustering with respect to the proposed image model; (b) provides an intrinsic view of the indoor scene and (c) provides statistics w.r.t. the planar property of the regions.

- A statistical region merging method (Section 4.3.4) based on certain region merging predicates. This method can be incorporated independently with any other existing indoor RGB-D scene segmentation method.
- A benchmark (Section 4.4) on the NYUD2 (Silberman et al., 2012) for unsupervised scene segmentation. Results from the proposed method show that it is comparable w.r.t. the state of the art and better in terms of computational time.

The outline of the rest of this Chapter is as follows: Section 4.2 discuss the background of RGB-D segmentation methods and related works. Section 4.3 presents the proposed method. Section 4.4 provides the experimental results and discussion. Finally, Section 4.5 draws conclusions and discusses future perspectives.

4.2 Background of RGB-D Segmentation

Color image segmentation of natural and outdoor scene is a well-studied problem due to its numerous applications in computer vision. Different methods to solve the problem have been already established based on different perspectives such as contour, clustering, affinity, energy minimization, etc. Chapter 5 of Szeliski (2011) provides a detail overview of these methods.

Many of the established image analysis methods have been either modified or directly employed to the depth image data in order to analyze and to modelize it, see Chapter 6 of Dal Mutto et al. (2012b) for a detail review. In the simplest cases, the depth image is considered as a grayscale image or converted to a cloud of 3D points. However, such simple approaches have limitations (Dal Mutto et al., 2012b) and hence better features such as surface normals are suggested to use (Rusu, 2013; Holz et al., 2012). We followed such suggestions and developed method in Chapter 3. From the results, we observed that: (a) the use of surface normals solely is not sufficient to extract full semantics of the scene and (b) it is necessary to incorporate additional features, such as color, texture etc. for providing better interpretation of indoor environments. Such observations raise the necessity to jointly exploit depth, color and other features for the task of image analysis.

A number of recent research activities, such as Dal Mutto et al. (2012a), Gupta et al. (2013), Ren et al. (2012) and Silberman et al. (2012), proposed different methodologies for indoor scene understanding and analysis with promising results. Most of these researches incorporate depth as complementary information with color images.

They are different among themselves mainly from two aspects: (a) feature-wise: different types, levels and dimensions of features and (b) method-wise: numerous distinctions, such as supervised, unsupervised, clustering based, graph based, split-merge based, etc. Different methods emphasize on different aspects of the problem, which eventually opens a number of interesting and challenging part to focus on.

A common approach to tackle the RGB-D scene analysis problem is to extract different features, design kernels and classify pixels with learned classifiers. For example, Ren et al. (2012) proposed contextual models in a supervised setting. Their model combines kernel descriptors with a segmentation tree or with superpixels Markov Random Field (MRF). To this aim, they extended the well-known gPb-UCM algorithm (Arbelaez et al., 2011) to incorporate the global probability of boundaries (gPb) of depth image with gPb of RGB image. The RGB-D scene analysis method proposed by Silberman et al. (2012) first gives an over-segmentation of the scene by applying watershed on the gPb of the RGB image. Next, it aligns the over-segmentation with the 3D planes. Finally, using a trained classifier it applies a hierarchical segmentation in order to merge regions. Beside proposing the method, Silberman et al. (2012) released an annotated RGB-D dataset (NYUD2) to perform scene analysis. Recently, Gupta et al. (2013) extended the gPb-UCM (Arbelaez et al., 2011) method in a supervised setting. First, they combine geometric contour cues: convex and concave normal gradients with monocular cues: brightness, color, texture. Then, they detect pixels as contours via learned classifiers for 8 different orientations. Finally, they generate a hierarchy of segmentations from all oriented detectors. All of the above-mentioned methods use supervised approach in order to combine/fuse different features or information extracted from them. Let us now focus on the methods in unsupervised domain.

Dal Mutto et al. (2012a) discussed about the fusion of color with geometry in an unsupervised setting and provide a solution using the normalized cut spectral clustering method. Their approach consists of identifying an optimal multiplier to balance between color and depth. For this reason, they generate several segmentations with different values of the multiplier. Each segmentation is obtained by applying spectral clustering on the fused subsampled features. Finally, they select the best segmentation based on their proposed RGB-D segmentation quality evaluation score. In practice, this method requires more computation time as it generates a number of different segmentations for a single image. Taylor and Cowley (2011) proposed a method which first extract edges from RGB image, apply Delaunay Triangulation on the edges to construct triangular graph and then apply Normalized Cut algorithm to

the graph. In the second step, they extract planar surfaces from the segments using RANSAC (Szeliski, 2011) and finally merge the coplanar segments using a greedy merging procedure. The unsupervised method that we propose in this Chapter is different than the above proposals as: (a) it considers surface normals as features; (b) it employs mixture model based joint clustering rather than Normalized Cut and (c) it merges regions based on statistics rather than a greedy approach.

Beside these approaches, the well-known graph based segmentation (Felzenszwalb and Huttenlocher, 2004) is extended for joint color and depth image segmentation. For example, Niu et al. (2012) extended it by including disparity with color for the purpose of segmenting stereopsis images. Strom et al. (2010) extended it by incorporating surface normals to segment colored 3D laser point clouds. For the purpose of comparison, we develop an extension of the graph based method that considers both 3D and normals along with color.

Despite all of these researches, it remains an interesting issue about what could be an appropriate statistical model to describe RGB-D images of indoor scenes and how to exploit such model to segment the captured images. Scene-SIRFS (Barron and Malik, 2013) is a recently proposed model whose aim is to recover intrinsic scene properties from single RGB-D image. It considers a mixture of shapes and illuminations where the mixture components are embedded in a soft segmentation of 17 eigenvectors. These eigenvectors are obtained from the normalized Laplacian corresponding to the input RGB image. Although the concept of using mixture is similar to the proposed method of this Chapter, the underlying objective, model and methodologies are different. We consider a mixture of shape (via 3D and normals) and color that consists of a feature vector of length 9. In the next Section, we present our proposed scene analysis method.

4.3 Methodology

4.3.1 Image Generation Model

We propose a statistical image model that fuses color and shape (3D and surface normals) features according to the *âĂIJnaïveâĂİ Bayes* assumption (Murphy, 2012), i.e., the features are independent of each other. Furthermore, it is based on a generative model (Murphy, 2012) where the features are issued from a finite mixture of different probability distributions. We consider the multivariate Gaussian (Bishop, 2006) distribution for the color and 3D features and the multivariate Watson (Mardia

and Jupp, 2009) distribution for surface normals. Mathematically, such a model with k components has the following form:

$$g\left(\mathbf{x}_i|\Theta_k\right) = \sum_{j=1}^{k} \pi_{j,k}\, f_g(\mathbf{x}_i^C|\mu_{j,k}^C, \Sigma_{j,k}^C)\, f_g(\mathbf{x}_i^P|\mu_{j,k}^P, \Sigma_{j,k}^P)\, W_d\left(\mathbf{x}_i^N|\mu_{j,k}^N, \kappa_{j,k}^N\right) \tag{4.1}$$

Here $\mathbf{x}_i = \{\mathbf{x}_i^C, \mathbf{x}_i^P, \mathbf{x}_i^N\}$ is the 9 dimensional feature vector of the ith pixel with $i = 1, ..., M$. Superscripts denote: C - color, P - 3D position and N - normal. $\Theta_k = \{\pi_{j,k}, \mu_{j,k}^C, \Sigma_{j,k}^C, \mu_{j,k}^P, \Sigma_{j,k}^P, \mu_{j,k}^N, \kappa_{j,k}^N\}_{j=1...k}$ denotes the set of model parameters where $\pi_{j,k}$ is the prior probability, $\mu_{j,k}$ is the mean, $\Sigma_{j,k}$ is the variance-covariance symmetric positive-definite matrix and $\kappa_{j,k}$ is the concentration of the jth component. $f_g(.)$ and $W_d(.)$ are the density functions of the multivariate Gaussian distribution (Section 4.3.3.2) and the multivariate Watson distribution (Section 4.3.3.3) respectively.

4.3.2 Segmentation method

Figure 4.1 illustrates the work flow of the proposed RGB-D segmentation method that consists of two sub-tasks such as: (1) clustering heterogeneous (color, 3D and Normal) data and (2) merging regions. The first task performs a joint color-spatial-axial clustering and generates a set of regions. The second task performs a refinement on the set with the aim to merge regions which are susceptible to be over-segmented. In the next two sub-sections we present our methods to accomplish these tasks.

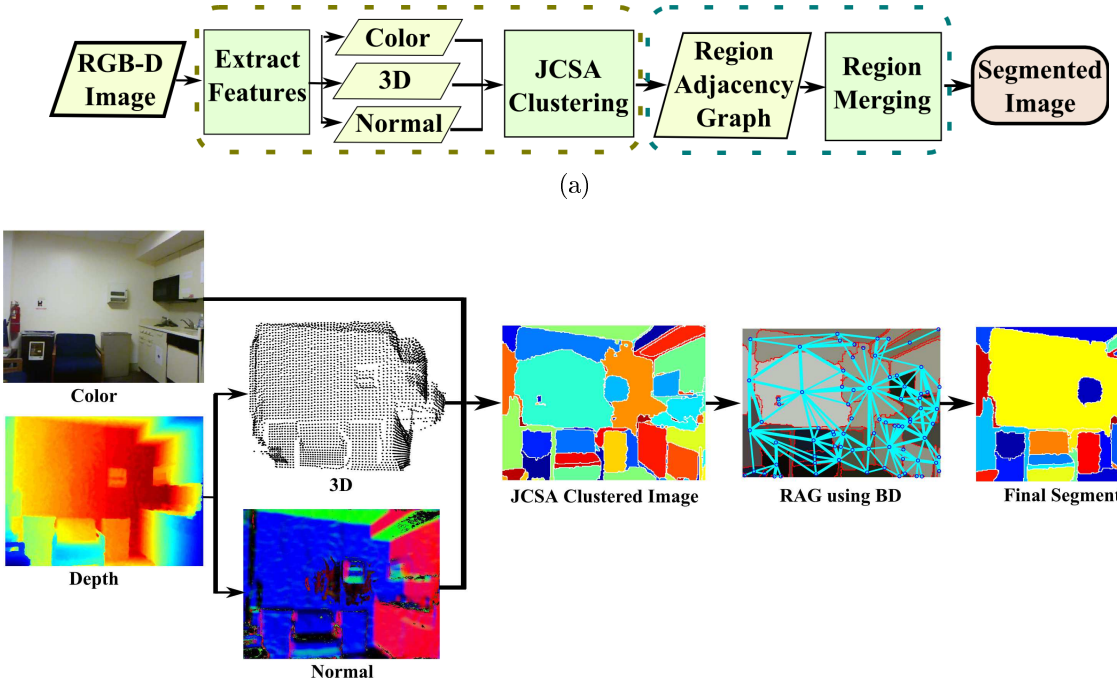


Figure 4.1: Work flow of the proposed segmentation method. (a) Block diagram and (b) Illustration with an example.

4.3.3 Joint Color-Spatial-Axial (JCSA) clustering

In order to cluster heterogeneous data, we develop a Joint Color-Spatial-Axial (JCSA) clustering method. The clustering method estimates the parameters of the mixture model (Eq. (4.1)) as well as clusters the image data/features. As an outcome, we obtain the groups of image pixels which form the regions in the image. However, notice that in an unsupervised setting the true number of segments are unknown. Therefore, we cluster features with the assumption of certain maximum number of clusters ($k = k_{max}$). Such assumption often causes an over-segmentation of the image. In order to tackle this issue, it is necessary to merge the over-segmented regions (see Section 4.3.4).

The proposed joint clustering method exploits and extends the clustering methodologies developed in Chapter 2 and 3. Recall that, both the Gaussian and the Watson distributions belong to the Exponential Family of Distributions. Therefore, based on the Linearity property (Boissonnat et al., 2010) of Bregman divergence (see Section 2.3.5 of Chapter 2), it is possible to compute Bregman divergence among two distributions of the following combined form:

$$f_{comb}(\mathbf{x}_i|\Theta_{j,k}) = f_g(\mathbf{x}_i^C|\mu_{j,k}^C, \Sigma_{j,k}^C)\, f_g(\mathbf{x}_i^P|\mu_{j,k}^P, \Sigma_{j,k}^P)\, W_d\left(\mathbf{x}_i^N|\mu_{j,k}^N, \kappa_{j,k}^N\right) \tag{4.2}$$

where $\Theta_{j,k} = \{\pi_{j,k}, \mu^C_{j,k}, \Sigma^C_{j,k}, \mu^P_{j,k}, \Sigma^P_{j,k}, \mu^N_{j,k}, \kappa^N_{j,k}\}$ denotes the j_{th} component parameters of Θ_k. This eventually allows to develop a joint Bregman soft clustering method for the model in Eq. (4.1).

We refer readers to Chapter 2 and 3 for a detail review of Exponential Family of Distributions, Bregman divergence and Bregman soft clustering. However, to keep the presentation of the proposed joint clustering method independent, in the following sub-sections we will repeat necessary elements in a concise form.

4.3.3.1 Exponential Family of Distributions (EFD) and Bregman Divergence

A multivariate probability density function $f(x|\eta)$ belongs to the exponential family if it has the following (Eq. (3.7) of (Banerjee et al., 2005b), Eq. (60) of (Nielsen and Garcia, 2009)) form[2]:

$$f\left(x|\eta\right) = \exp\left(-D_G\left(t(x), \eta\right)\right) \exp\left(k(x)\right) \tag{4.3}$$

and

$$D_G\left(\eta_1, \eta_2\right) = G(\eta_1) - G(\eta_2) - \langle \eta_1 - \eta_2, \nabla G(\eta_2) \rangle \tag{4.4}$$

with $G(.)$ the Legendre dual of log normalizing function which is a strictly convex function. ∇G the gradient of G. $t(x)$ denotes the sufficient statistics and $k(x)$ is the carrier measure. The expectation of the sufficient statistics $t(x)$ w.r.t. the density function (Eq. (4.3)) is called the expectation parameter (η). D_G is the Bregman divergence computed from expectation parameters: it can be used to compute a measure of distance between two distributions of the same exponential family, defined by two expectation parameters η_1 and η_2. We will define in the following Section the particular forms obtained with the Gaussian distribution and the Watson distribution.

4.3.3.2 Multivariate Gaussian Distribution

For a d dimensional random vector $\mathbf{x} = [x_1, ..., x_d]^T \in \mathbb{R}^d$, the multivariate Gaussian distribution is defined as:

$$f_g(\mathbf{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{d/2}\det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)\right) \tag{4.5}$$

Here, $\mu \in \mathbb{R}^d$ denotes the mean and Σ denotes the variance-covariance symmetric positive-definite matrix. To write the multivariate Gaussian distribution in the form of

[2]In order to keep our formulations concise, we use the expectation parameters η to define the Exponential Family of Distributions. However, we provide the other form: $f\left(x|\theta\right) = \exp\left(\langle t(x), \theta \rangle - F(\theta) + k(x)\right)$ and related derivations in the previous Chapters.

Eq. (4.3), the elements are defined as (Nielsen and Garcia, 2009): sufficient statistics $t(\mathbf{x}) = (\mathbf{x}, -\mathbf{x}\mathbf{x}^T)$; carrier measure $k(\mathbf{x}) = 0$; expectation parameter $\eta = (\phi, \Phi) = \left(\mu, -(\Sigma + \mu\mu^T)\right)$ and $G_g(\eta) = -\frac{1}{2}\log(1 + \phi^T\Phi^{-1}\phi) - \frac{1}{2}\log(\det(\Phi)) - \frac{d}{2}\log(2\pi e)$.

4.3.3.3 Multivariate Watson Distribution

For a d dimensional unit vector $\mathbf{x} = [x_1, ..., x_d]^T \in S^{d-1} \subset \mathbb{R}^d$ (i.e. $\|\mathbf{x}\|_2 = 1$), the multivariate (axially symmetric) Watson distribution (mWD) is defined as (Mardia and Jupp, 2009):

$$W_d(\mathbf{x}|\mu, \kappa) = M\left(1/2, d/2, \kappa\right)^{-1} \exp\left(\kappa(\mu^T\mathbf{x})^2\right) = W_d(-\mathbf{x}|\mu, \kappa) \tag{4.6}$$

Here, μ is the mean direction (with $\|\mu\|_2 = 1$), $\kappa \in \mathbb{R}$ the concentration and $M\left(1/2, d/2, \kappa\right)$ the Kummer's function (Mardia and Jupp, 2009). To write the mWD in the form of Eq. (4.3), the elements are defined as: sufficient statistics $t(\mathbf{x}) = \left[x_1^2, ..., x_d^2, \sqrt{2}x_1x_2, ..., \sqrt{2}x_{d-1}x_d\right]^T$; carrier measure $k(\mathbf{x}) = 0$; expectation parameter η as:

$$\eta = \|\eta\|_2 \, \nu \tag{4.7}$$

where $\nu = \left[\mu_1^2, ..., \mu_d^2, \sqrt{2}\mu_1\mu_2, ..., \sqrt{2}\mu_{d-1}\mu_d\right]^T$ and

$$G_w(\eta) = \kappa \|\eta\|_2 - \log M\left(1/2, d/2, \kappa\right) \tag{4.8}$$

With the above formulation, for a set of observations $\mathbf{X} = \{\mathbf{x}_i\}_{i=1,...,M}$ we estimate $\eta = E[t(\mathbf{X})]$ and κ with a Newton-Raphson root finder method as (Sra and Karp, 2013):

$$\kappa_{l+1} = \kappa_l - \frac{q(1/2, d/2; \kappa_l) - \|\eta\|_2}{q^{'}(1/2, d/2; \kappa_l)} \tag{4.9}$$

where $q(1/2, d/2; .)$ is the Kummer-ratio, $q^{'}(1/2, d/2; .)$ is the derivative of $q(1/2, d/2; .)$. See Chapter 3 for details.

4.3.3.4 Bregman Divergence for the combined model

Our image model (in Eq. (4.1)) combines different exponential family of distributions (associated to color, 3D and normals) based on independent (*naïve Bayes* (Murphy, 2012)) assumption. Therefore, Bregman Divergence (BD) of the combined model can be defined as a linear combination of the BD of each individual distributions:

$$D_G^{comb}(\eta_i, \eta_j) = D_{G,g}^C(\eta_i^C, \eta_j^C) + D_{G,g}^P(\eta_i^P, \eta_j^P) + D_{G,w}^N(\eta_i^N, \eta_j^N) \tag{4.10}$$

where, $D_{G,g}(.,.)$ denotes BD using multivariate Gaussian distribution and $D_{G,w}(.,.)$ denotes BD using multivariate Watson distribution. Then, it is possible to define, with expectation parameter $\eta = \{\eta^C, \eta^P, \eta^N\}$:

$$G^{comb}(\eta) = G_g(\eta^C) + G_g(\eta^P) + G_w(\eta^N) \tag{4.11}$$

4.3.3.5 Bregman Soft Clustering for the combined model

Bregman soft clustering exploits Bregman Divergence (BD) in the Expectation Maximization (EM) framework (Murphy, 2012) to compute the Maximum Likelihood Estimate (MLE) of the mixture model parameters and provides a soft clustering of the observations (Banerjee et al., 2005b). In the expectation step (E-step) of the algorithm, the posterior probability is computed as (Nielsen and Garcia, 2009):

$$p\left(\gamma_i = j|\mathbf{x}_i\right) = \frac{\pi_{j,k} \exp\left(G^{comb}(\eta_{j,k}) + \left\langle t(\mathbf{x}_i) - \eta_{j,k}, \nabla G^{comb}(\eta_{j,k})\right\rangle\right)}{\sum_{l=1}^{k} \pi_{l,k} \exp\left(G^{comb}(\eta_{l,k}) + \left\langle t(\mathbf{x}_i) - \eta_{l,k}, \nabla G^{comb}(\eta_{l,k})\right\rangle\right)}, \; j = 1, ..., k \tag{4.12}$$

Here, $\eta_{j,k}$ and $\eta_{l,k}$ denote the expectation parameters for any cluster j and l given that the total number of components is k. The maximization step (M-step) updates the mixing proportion and expectation parameter for each class as:

$$\pi_{j,k} = \frac{1}{M} \sum_{i=1}^{M} p\left(\gamma_i = j|\mathbf{x}_i\right) \text{ and } \eta_{j,k} = \frac{\sum_{i=1}^{M} p\left(\gamma_i = j|\mathbf{x}_i\right) \mathbf{x}_i}{\sum_{i=1}^{M} p\left(\gamma_i = j|\mathbf{x}_i\right)} \tag{4.13}$$

Initialization is a prominent issue and has significant impact on clustering. Our initialization procedure consists of setting initial values for prior class probability $(\pi_{j,k})$ and the expectation parameters $(\eta_{j,k})$ with $1 \leq j \leq k$. We initialize π and η associated to the Gaussian and Watson using a combined k-means type clustering. After initialization, we iteratively apply the E-step and M-step until the convergence criteria are met. These criteria are based on maximum number of iterations (e.g. 200) and a threshold difference (e.g. 0.001) between the negative log likelihood values (see Eq. (4.14)) of two consecutive steps.

$$nLLH(\Theta_k) = -\sum_{i=1}^{M} \log\left(g\left(\mathbf{x}_i|\Theta_k\right)\right) \tag{4.14}$$

The above procedures lead to a soft clustering algorithm, which generates associated probability and parameters for each component of the proposed model in Eq. (4.1). Let us call this the BSC-COMB algorithm (Algorithm 3). Finally, for each sample we get the class label $(\hat{\gamma}_i)$ using the updated combined BD (Eq. 4.10) as:

$$\hat{\gamma}_i = \underset{j=1,...,k}{\arg\min}\, D_G^{comb}(t(x_i), \eta_{j,k}) \tag{4.15}$$

Algorithm 3: BSC-COMB algorithm for Joint Color-Spatial-Axial clustering.

Input: $\mathbf{X} = \left\{\mathbf{x}_i \mid \mathbf{x}_i = \{\mathbf{x}_i^C, \mathbf{x}_i^P, \mathbf{x}_i^N\},\ \mathbf{x}_i^C \in R^d, \mathbf{x}_i^P \in R^d, \mathbf{x}_i^N \in S^{d-1} \wedge 1 \leqslant i \leqslant M\right\}$

Output: A soft clustering of $\mathbf{X}$ with k components.

Initialize $\pi_{j,k}$ and $\eta_{j,k}$ for $1 \leq j \leq k$ using combined kmeans;

while *not converged* **do**

  {Perform the E-step of EM};

  foreach *i and j* **do**

    Compute $p(\gamma_i = j|\mathbf{x}_i)$ using Eq. (4.12)

  end

  {Perform the M-step of EM};

  for $j = 1$ *to* k **do**

    Update $\pi_{j,k}$ and $\eta_{j,k}$ using Eq. (4.13)

  end

end

Applying Algorithm 3 on RGB-D image features (color, position and normals) performs a joint color-spatial-axial clustering. Note that, we apply this clustering method with the assumption of certain maximum number of components $k = k_{max}$. Image regions obtained by such clustering often lead to over-segmentation. Therefore, it is necessary to merge the over-segmented regions. In the following, we propose a region merging method to tackle such over-segmentation.

4.3.4 Region Merging

In this sub-task, we merge the over-segmented regions which are generated from previous step. To this aim, first we build a Region Adjacency Graph (RAG) (Trémeau and Colantoni, 2000) (see Figure 4.1). The graph considers that each region is a node and each node has edges with its adjacent nodes. In order to define the edge connectivity among nodes, we consider a measure of statistical distance among two regions. Moreover, we consider the boundary strength among regions as a measure of their eligibility to merge. Similar to the standard region merging methods (Trémeau and Colantoni, 2000; Nock and Nielsen, 2004; Peng and Zhang, 2011), we define the region merging predicates and merging order. As an outcome of region merging we obtain the final segmentation.

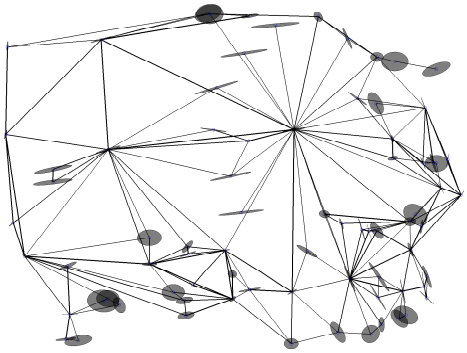


Figure 4.2: Illustration of a Region Adjacency Graph (RAG) constructed from JCSA clustered regions of the image in Figure 4.1(b). The circle at each node represents the concentration of image normals at the region. Each edge represents the weight w_d among two adjacent nodes/regions.

4.3.4.1 Region Adjacency Graph (RAG)

In our proposed region merging method, RAG provides an inherent view of the merging strategy. Figure 4.2 illustrates an example of the RAG constructed from clustered regions of the image in Figure 4.1(b). Let $R = \{r_i\}_{i=1,\ldots,Z}$ be the set of regions that we obtain from the JCSA clustering; $G = (V, E)$ be the undirected graph that represents the RAG, where $v_i \in V$ is the set of nodes corresponding to the regions $r_i \in R$ and E is the set of edges among adjacent nodes.

Each node v_i consists of the parameters (mean direction μ and concentration κ) of the Watson distribution (Section 4.3.3.3) associated with region r_i. In Figure 4.3 the radius of the circles (nodes) represents the κ value and the orientation of the nodes represents the mean direction μ.

Each edge e_{ij} consists of two weights: w_d, based on statistical dissimilarity and w_b, based on boundary strength between adjacent nodes v_i and v_j. The dissimilarity based weight w_d is computed using the Bregman divergence (Eq. (4.4)) among two adjacent nodes v_i and v_j as:

$$w_d(v_i, v_j) = \min\left(D^N_{G,w}(\eta^N_i, \eta^N_j), D^N_{G,w}(\eta^N_j, \eta^N_i)\right) \tag{4.16}$$

where, $D^N_{G,w}(\eta^N_i, \eta^N_j)$ is the Bregman divergence (Eq. (4.4)) among the Watson distributions associated with regions r_i and r_j. The boundary based weight w_b between two nodes v_i and v_j is computed from the average normalized gradient values along the boundary of their corresponding regions r_i and r_j as:

$$w_b(v_i, v_j) = \frac{1}{|r_i \bigcap r_j|} \sum_{b \in r_i \bigcap r_j} I^{rgbd}_G(b) \tag{4.17}$$

where, $r_i \bigcap r_j$ is the set of boundary pixels among two regions, $|.|$ denotes the cardinality and I_G^{rgbd} is the normalized magnitude of image gradient[3] (MoG) (Szeliski, 2011) computed from the RGB-D image. I_G^{rgbd} is obtained by first computing MoG for each color channels (I_G^r, I_G^g, I_G^b) and depth (I_G^d) individually, and then taking the maximum of those MoGs at each pixel.

4.3.4.2 Merging Strategy

Our region merging strategy is an iterative procedure that proceeds by employing merging predicates among adjacent nodes in a certain order. The merging predicates consist of evaluating the *candidacy* of each node, the *eligibility* of merging adjacent nodes and verifying the *consistency* of the merged nodes. Once two nodes are merged, the information regarding the merged node and its edges are updated instantly. This procedure continues until no valid candidates are left to merge.

candidacy of a node/region defines whether it is a valid candidate to be merged with the adjacent nodes. For each node, first we check its *candidacy*. This helps us to filter out a number of nodes which are not a valid candidate to be merged and hence reduce the computational time. Our proposed *candidacy* criterion for a node checks the planar property of its associated region. Since our goal is to merge the adjacent planar regions, we do not consider any region which is non-planar. This property can be easily investigated by analyzing the concentration parameter (κ) associated with each node v_i. We define the *candidacy* of a node v_i as follows:

$$candidacy(v_i) = \begin{cases} true, & \text{if } \kappa_i > \kappa_p, \\ false, & \text{otherwise.} \end{cases} \tag{4.18}$$

Here κ_i is the concentration parameter computed from the region r_i. κ_p is the threshold that defines the planar property of a region. In Chapter 3, we observed that the concentration of the normals associated with a region can be exploited to discriminate among the planar and non-planar surfaces. In Eq (4.18) we are exploiting that observation. See Section 4.4 for details about the κ_p threshold value.

We define the *eligibility* of merging two regions (r_i and r_j) based on the dissimilarity based weight w_d (using Eq. (4.16)) and boundary based weight w_b (using Eq.

[3]To compute image gradient $\Delta I = \left(\frac{\partial I(x,y)}{\partial x}, \frac{\partial I(x,y)}{\partial y}\right)$, with $\frac{\partial I(x,y)}{\partial x} \approx \frac{I(x+1,y)-I(x-1,y)}{2}$ and $\frac{\partial I(x,y)}{\partial y} \approx \frac{I(x,y+1)-I(x,y-1)}{2}$, we used the 'sobel' operator in MATLAB implementation.

(4.17)) among the corresponding nodes (v_i and v_j) as:

$$eligibility(v_i, v_j) = \begin{cases} true, & (a)\ w_b(v_i, v_j) < th_b;\ \text{and} \\ & (b)\ w_d(v_i, v_j) < th_d; \\ false, & \text{otherwise.} \end{cases} \tag{4.19}$$

where, th_b and th_d are the thresholds associated with the boundary weight w_b and the distance weight w_d. See Section 4.4 for details about their influence on region merging and segmentation.

We employ the plane outlier ratio in order to verify the *consistency* (Peng and Zhang, 2011) of a merged region. It is computed by first fitting a plane to the 3D points belonging to the merged region and then computes the ratio of inliers and outliers based on a threshold distance (Taylor and Cowley, 2013). We employed the widely used RANSAC (Szeliski, 2011) algorithm for the purpose of plane fitting. Therefore, we define *consistency* among two regions r_i and r_j as follows:

$$consistency(v_i, v_j) = \begin{cases} true, & \text{if } planar\ outlier\ ratio > th_r, \\ false, & \text{otherwise.} \end{cases} \tag{4.20}$$

where, th_r is the threshold associated with the plane outlier ratio. We set this threshold following the existing methods, such as Taylor and Cowley (2013).

Finally, we define the *region merging predicate* (Peng and Zhang, 2011) P_{ij} based on: (a) candidacy (using Eq. (4.18)); (b) eligibility of merging (using Eq. (4.19)) and (c) consistency of merged node (using Eq. (4.20)) as:

$$P_{ij} = \begin{cases} true, & \text{if } (a)\ candidacy(v_j) = true;\ \text{and} \\ & \quad (b)\ eligibility(v_i, v_j) = true;\ \text{and} \\ & \quad (c)\ consistency(r_i, r_j) = true \\ false, & \text{otherwise.} \end{cases} \tag{4.21}$$

Let us note that the conditions in the merging predicate are applied sequentially and hence reduce computational time. The condition (b) in the merging predicate is related to the statistical properties extracted from the regions. One could ignore this condition and expect similar results. However, this will significantly increase the computational time.

The *region merging order* (Peng and Zhang, 2011) sorts the adjacent regions that should be evaluated and merged sequentially. However, it changes dynamically after each merging occurs. We define the *merging order* based on dissimilarity based weights w_d among the adjacent nodes. The adjacent node v_j which has minimum

$w_d(v_i, v_j)$ is considered to be evaluated first. We use w_d as the merging order constraint due to its ability to provide a measure of dissimilarity among regions. Such a measure is based on the mean direction (μ) and the concentration (κ) of the surface normals of the regions. Therefore, with this constraint, the neighboring region, which is most similar w.r.t. μ and κ will be selected as the first candidate to evaluate using Eq. (4.21).

Algorithm 4 provides the pseudo code for the proposed region merging method. It begins with a set of regions obtained by applying Algorithm 3 on an RGB-D image. As an outcome, it provides the final segmentation result. In the next Section, we evaluate the results obtained from the RGB-D segmentation method developed in this Chapter.

Algorithm 4: Region Merging algorithm.

Input: $R = \{r_i\}_{i=1,\ldots,Z}$, $G = (V, E)$, κ_p, th_b, th_d and th_r
Output: Final segmentation after region merging.
Compute *candidacy*(v_i) for $\{v_i\}_{i=1,\ldots,Z}$ using Eq. (4.18);
Set $i = 1$;
foreach i **do**
  if *candidacy*(v_i) *is* true **then**
    while *no adjacent of* v_i *is left to check* **do**
      Sort e_{ij} in ascending order according to $w_b(v_i, v_j)$;
      Evaluate each v_j with the *merging predicate* P_{ij} (Eq. (4.21)) ;
      if P_{ij} *is* true **then**
        Merge two nodes v_i and v_j and update the RAG;
        Start over again from sorting the adjacents e_{ij} of the node v_i.
      else
        Check the next node
      end
    end
  end
end

4.4 Experiments and Results

In this Section, we evaluate the proposed method on the benchmark image database NYUD2 (Silberman et al., 2012) which consists of 1449 indoor images with RGB, depth and ground-truth information. We convert (using MATLAB function) the RGB color information into $L^*a^*b^*$ (CIELAB space) color because of its perceptual accuracy (Cheng et al., 2011). From the depth images, we compute the 3D coordinates

and surface normals using the toolbox available with the database (Silberman et al., 2012).

Our clustering method requires to set initial labels of the pixels and the number of clusters k. We initialize it following the k-means++ (Arthur and Vassilvitskii, 2007) strategy with $k = 20$. For the region merging we empirically set the thresholds as: $\kappa_p = 5$ to decide a region as planar (see Section 3.4.2.2 of Chapter 3), $th_b = 0.2$ to decide the existence of boundary among two regions, $th_d = 3$ to decide the distance among two regions and $th_r = 0.9$ to determine the goodness of a plane fitting.

We evaluate performance using standard benchmarks (Arbelaez et al., 2011) which are applied to compare the test and ground truth segmentation: (1) probability rand index (PRI), it measures likelihood of a pair of pixels that has same label; (2) variation of information (VoI), it measures the distance between two segmentations in terms of their average conditional entropy; (3) boundary displacement error (BDE) (Freixenet et al., 2002), it measures the average displacement between the boundaries of two segmentations; (4) Ground truth region covering ($GTRC$), it measures the region overlaps between ground truth and test and (5) Boundary based F-measure (BFM), a boundary measure based on precision-recall framework (Arbelaez et al., 2011). With these criteria a segmentation is better if PRI, $GTRC$, BFM are larger whereas VoI and BDE are smaller.

First, we study the sensitivity of the proposed method w.r.t. the parameters (k, κ_p, th_b, th_d), which is presented in table 4.1. The parameter k is related to the clustering method (Section 4.3.3) while κ_p, th_b and th_d are related to the region merging method (Section 4.3.4). Note that, the parameter $th_r = 0.9$ is set by following Taylor and Cowley (2013) and hence we do not analyze it further. From table 4.1, we observe that while PRI (1%) is quite stable, VoI (6%), BDE (8%) and GTRC (7%) provide discriminating view w.r.t the parameters. The parameter k is inversely related to the number of pixels in a cluster. In segmentation, a smaller k causes to loose details in the scene while higher k splits the scene into more regions. We set κ_p based on the study we did on NYUD2 (see Section 3.4.2.2 of Chapter 3) for details) which reveals that planar surfaces can be characterized with concentration $\kappa >= 5$. While, a lower κ value selects non-planar surfaces to be merged, a higher value may reject true planar surfaces for merging. Following the OWT-UCM (Arbelaez et al., 2011) method, we empirically set the value of th_b. Similarly, we set th_d empirically. In theory two regions which belong to the same direction have a negligible value of the Bregman divergence. However, the inaccurate computation of the shape features and the presence of noise in the acquired depth information often causes the Bregman

divergence to be high. From our experience with the images of NYUD2, th_d should be within the range between 2 to 4.

	$\{k, 5, 0.2, 3\}$			$\{20, \kappa_p, 0.2, 3\}$			$\{20, 5, th_b, 3\}$			$\{20, 5, 0.2, th_d\}$		
	15	20	25	2	5	8	0.1	0.2	0.3	2	3	4
VoI	2.31	**2.29**	2.42	2.32	**2.29**	2.38	2.43	**2.29**	2.32	2.37	**2.29**	2.32
BDE	10.64	**9.83**	10.05	10.52	**9.83**	10.00	9.98	**9.83**	10.34	10.10	**9.83**	10.00
PRI	0.89	**0.90**	0.89	0.89	**0.90**	**0.90**	0.89	**0.90**	0.89	**0.90**	**0.90**	**0.90**
GTRC	0.56	**0.58**	0.57	0.56	**0.58**	0.56	0.54	**0.58**	0.56	0.56	**0.58**	0.57

Table 4.1: Sensitivity of JCSA-RM with respect to the parameters $\{k, \kappa_p, th_b, th_d\}$.

We also compare the proposed method ***JCSA-RM*** (joint color-spatial-axial clustering and region merging) with several unsupervised RGB-D segmentation methods such as: RGB-D extension of OWT-UCM (Ren et al., 2012) (UCM-RGBD), modified Graph Based segmentation (Felzenszwalb and Huttenlocher, 2004) with color-depth-normal (GBS-CDN), Geometry and Color Fusion method (Dal Mutto et al., 2012a) (GCF) and the Scene Parsing Method (Taylor and Cowley, 2013) (SP). For the UCM-RGBD method we obtain best score with threshold value 0.1. The best results from GBS-CDN method are obtained by using $\sigma = 0.4$. To obtain the optimal multiplier (λ) in GCF (Dal Mutto et al., 2012a) we exploit the range 0.5 to 2.5. For the SP method, we scaled the depth values (1/0.1 to 1/10 in meters) to use author's source code Taylor and Cowley (2013).

Table 4.2 presents (best appears as bold) the comparison w.r.t. the average score of the benchmarks. Results show that JCSA-RM performs best in PRI, VoI and GTRC and comparable in BDE. However, in the BFM it is not comparable. The reason is that, BFM favors methods like UCM-RGBD which is specialized in contours detection. This indicates that JCSA-RM can be improved by incorporating the boundary information more efficiently, e.g., by incorporating boundary information within the joint clustering method.

Several segmentation examples to visualize the results are illustrated in Fig 4.3. These examples confirm that the segmentation from JCSA-RM (our proposed) and UCM-RGBD are competitive. However, they have several distinctions: (a) JCSA-RM is better in providing the details of indoor scene structures whereas UCM-RGBD loose them sometimes (see ex. 3-5); (b) UCM-RGBD provides better estimation of the object boundaries whereas JCSA-RM gives a rough boundary and (c) UCM-RGBD shows more sensitivity on color whereas JCSA-RM is more sensitive on directions. The GBS-CDN method provides visually pleasing results, however it often

	PRI	VoI	BDE	GTRC	BFM
UCM-RGBD	**0.90**	2.35	**9.11**	0.57	**0.63**
GBS-CDN	0.81	2.32	13.23	0.49	0.53
GCF	0.84	3.09	14.23	0.35	0.42
SP	0.85	3.15	10.74	0.44	0.50
JCSA	0.87	2.72	10.33	0.45	0.46
JCSA-RM	**0.90**	**2.29**	9.83	**0.58**	0.59

Table 4.2: Comparison with the state of the art. Methods: **UCM-RGBD** (Ren et al., 2012), **GBS-CDN** (Felzenszwalb and Huttenlocher, 2004), **GCF** (Dal Mutto et al., 2012a), **SP** (Taylor and Cowley, 2013), **JCSA** and **JCSA-RM** (proposed).

tends to loose details (see ex. 1-4) of the scene structure (e.g. merges wall with ceiling). Results from the SP method seems to be severely sensitive to the varying illumination and rough changes in surfaces (see ex. 3). The GCF method performs over-segmentation (see ex. 1, 3, and 5-7) or under-segmentation (see ex. 2 and 4), which is a drawback of such algorithm as it is often unable to estimate the correct number of clusters in real data. Moreover, the GCF method often fails to discriminate major surface orientations (see ex. 1, 2 and 4) as it does not consider the direction of surfaces (normals).

Comparing JCSA with JCSA-RM (Table 4.2), we can decompose the contributions of *clustering* and *region merging* in JCSA-RM. We see that *region merging* improves clustering output from 0.45 to 0.58 (28.88%) in GTRC. We believe that JCSA-RM can be improved and extended further in the following ways:

- Including a pre-processing stage, which is necessary because the shape features are often computed inaccurately due to noise and quantization (Barron and Malik, 2013). Moreover, we observed significant noise in the color images which are captured especially in low light condition. A method like Scene-SIRFS (shape, illumination and reflectance from shading) (Barron and Malik, 2013), which recover the intrinsic scene properties, can be used for pre-processing purpose.

- Enhancing the clustering method by adding contour information (Arbelaez et al., 2011) efficiently. Additionally, we may consider spatially constrained model such as (Nguyen and Wu, 2013) which incorporates boundary information by adding spatially varying constraints in the clustering task.

- Enhancing the region merging method with color information. To this aim, we can exploit the estimated reflectance information (using (Barron and Malik,

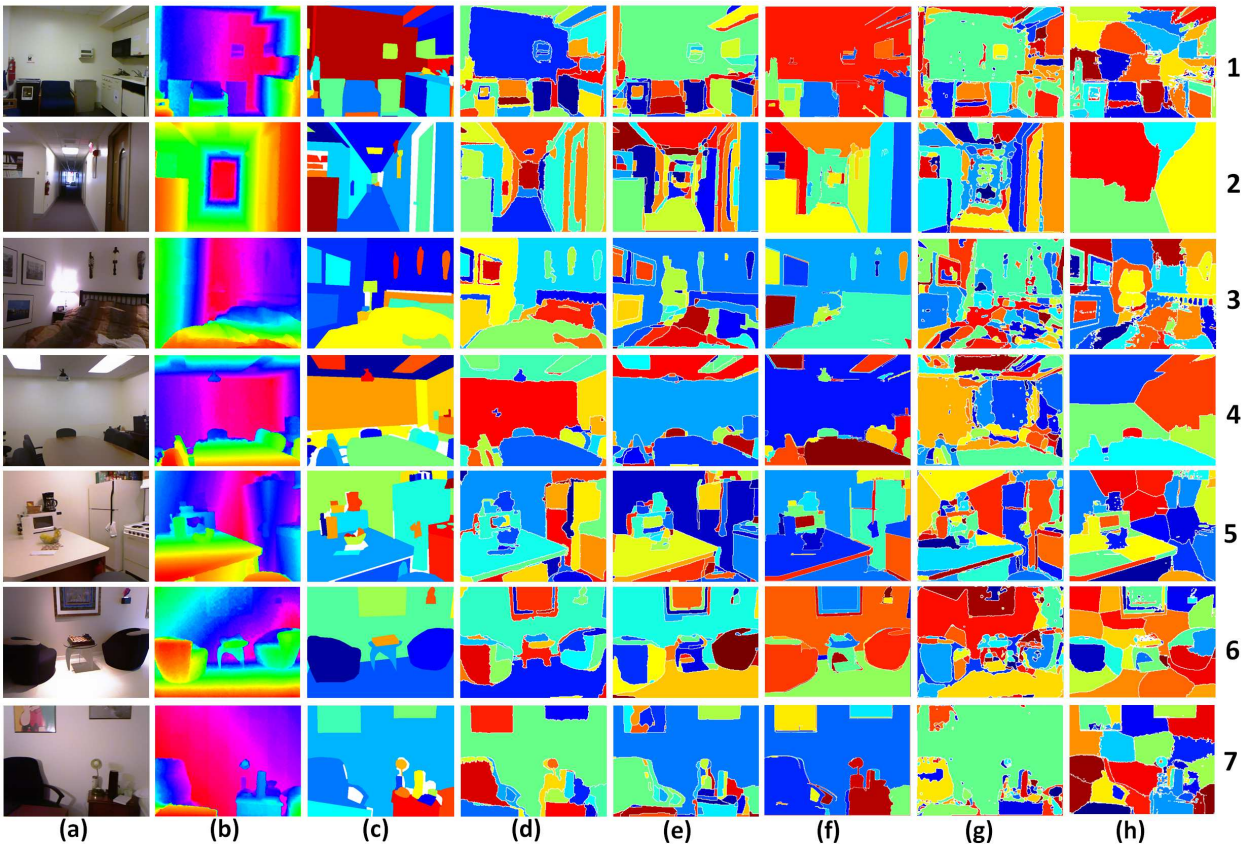


Figure 4.3: Segmentation examples (from top to bottom) on NYU RGB-D database (NYUD2). (a) Input Color image (b) Input Depth image (c) Ground truth (d) JCSA-RM (*our proposed*) (e) UCM-RGBD (Ren et al., 2012) (f) GBS-CDN (Felzenszwalb and Huttenlocher, 2004) (g) SP (Taylor and Cowley, 2013) and (h) GCF (Dal Mutto et al., 2012a).

2013)), such that the varying illumination is discounted.

In order to conduct the experiments we used a 64 bit machine with Intel Xenon CPU and 16 GB RAM. The JCSA-RM method is implemented in MATLAB, which on average takes 38 seconds, where 31 seconds for the clustering and 7 seconds for region merging. In contrast, UCM-RGBD (MATLAB and C++) takes 110 seconds. Therefore, JCSA-RM is ≈3 times faster[4] than UCM-RGBD. Moreover, we believe that implementing JCSA-RM in C++ will significantly reduce the computation time.

To further analyze the computation time of JCSA-RM, we run it for different image scales. Table 4.3 presents relevant information from which we see that the

[4]To perform a fair comparison, we conducted this experiment with half scaled image. This is due to the fact that the computational resource did not support to run UCM-RGBD for the full scale image.

reduction rate of JCSA computation time (in sec) w.r.t. different scales is approximately equivalent to the reduction rate of the number of pixels.

Scale	**1**	**1/2**	**1/4**	**1/8**
Num. pixels	239k	60k	15k	4k
JCSA (req. time in sec)	132	31	8	1.5
RM (req. time in sec)	42	7	1.4	0.33

Table 4.3: Computation time of JCSA-RM w.r.t. different image scales.

In Table 4.1 and 4.2 we observed that the *Ground Truth Region Covering (GTRC)* (Arbelaez et al., 2011) benchmark provides reasonable score to evaluate and differentiate among the different methods. Fig. 4.4 provides further analysis on NYUD2 (Silberman et al., 2012) using histograms of the GTRC scores. We observe that, while the JCSA-RM and UCM-RGBD covers quite similar regions in the histogram, others are quite different specially in the higher GTRC region.

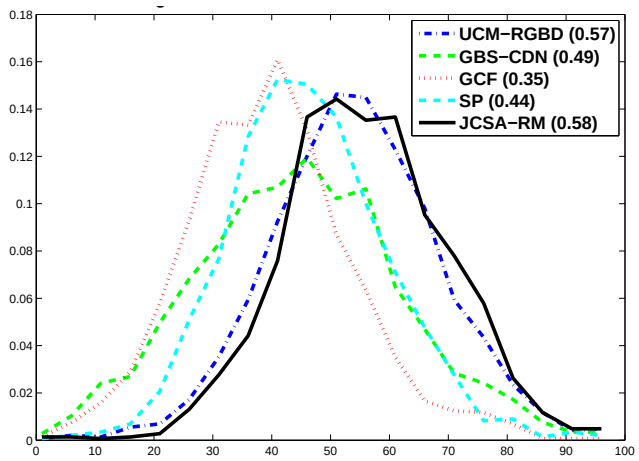


Figure 4.4: Histogram of *GTRC* (Arbelaez et al., 2011) scores of different methods.

Now, in Fig. 4.5 let us focus and analyze some segmentation examples which have lower (less than 0.4) GTRC score. Average GTRC score of JCSA-RM is 0.58 (see Table 4.1 and 4.2). Results show several cases for low scores:

- JCSA-RM method tends to provide more details (over-segment) while the ground truth keeps minimum detail, see ex. 1-3, and 5 in Fig. 4.5.
- JCSA-RM method do not provide enough detail (under-segment) while the ground truth does, see ex. 4 and 6 in Fig. 4.5. This is a very difficult case, as looking at the images we can see that the under-segmented regions have similar

color, depth and normal which in a general case difficult to segment without additional knowledge.

- Example 7 shows a characteristic example of JCSA-RM, which is to be biased on surface normals. This causes the furniture (sofa) to be segmented into several parts. Perhaps this can be improved by incorporating color based merging heuristics in our region merging method.

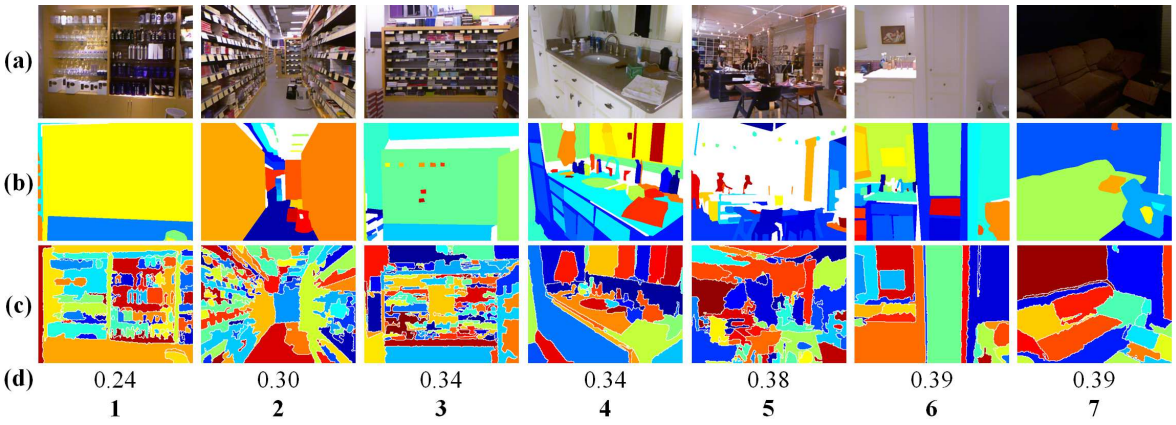


Figure 4.5: Segmentation examples with lower GTRC scores (less than 0.4). (a) Input Color Image (b) Ground Truth Segmentation (c) Segmentation with the JCSA-RM method and (d) GTRC score.

4.5 Conclusion

We proposed an unsupervised indoor RGB-D scene segmentation method. Our method is based on a statistical image generation model, which provides a theoretical basis for fusing different cues (e.g. color and depth) of an image. In order to cluster w.r.t. the image model, we developed an efficient joint color-spatial-axial clustering method based on Bregman divergence. Additionally, we proposed a region merging method that exploits the planar statistics of the image regions. We evaluated the proposed method with a database of benchmark RGB-D images and using widely accepted evaluation metrics. Results show that our method is competitive w.r.t. the state of the art and opens interesting perspectives for fusing color and geometry. We foresee several possible extensions of our method: more complex image model and clustering with additional features, region merging with additional hypothesis based on color. Moreover, we believe that the methodology proposed in this paper is equally

applicable and extendable for other complex tasks, such as joint image-speech data analysis.

Chapter 5

Conclusions

In this thesis, we focused on exploring, evaluating and developing unsupervised methods to analyze indoor images captured by Microsoft Kinect camera which is a synchronized color and depth sensor, also called RGB-D sensor. Kinect camera provides a low cost solution to access color with depth information at a reasonable rate. At present it is very popular and widely employed camera in a variety of applications related to the image processing and computer vision. Numerous researches have already shown that the performance of traditional image and vision algorithms enhances with the use of RGB-D images from Kinect.

This thesis begins shortly after the introduction of Kinect in the consumer market. Therefore, the methods developed during this thesis were concurrent with the demand from communities, particularly in the direction of developing relatively underexplored problems, such as unsupervised methods for indoor scene understanding and analysis. At the beginning, this thesis focused on developing an unsupervised depth image analysis method using the primitive depth features. To this aim, it proposed novel model based clustering algorithms with directional distributions to cluster surface normals. Next, it focused on extending the methods for the RGB-D image analysis. For this, it proposed efficient joint clustering method, which fuses different (color, spatial, directional) information together and performs joint clustering.

We evaluated the methods developed during this thesis w.r.t. the state of the art. Results show that they are better in terms of accuracy and computational efficiency. Although we applied the methods only for image analysis, they are mostly independent of particular domain. Hence, we believe that they will help practitioners and researchers of different domains which have similar requirements, such as unsupervised classification, clustering directional observations, fusion and clustering heterogeneous data, etc.

In the remaining part of this Chapter, first in Section 5.1 we provide a meta-summary of the contributions and findings of Chapters 2 and 3, and then we provide potential future work in Section 5.2.

5.1 Summary of contributions

The contributions of this thesis arise from applying, evaluating and developing clustering algorithms for unsupervised classification of patterns and its applications for indoor depth and RGB-D image analysis. The following is a summary of the principal contributions in this thesis.

5.1.1 Model Based Clustering with Directional Distributions

We consider the surface normals as one of the most important primitive depth features. Therefore, we particularly focused on developing algorithms to cluster normals. To this aim, in Chapter 2 and 3, we proposed novel Model Based Clustering (MBC) methods for the fundamental directional distributions called von Mises-Fisher (vMF) and multivariate Watson distributions. To the best of our knowledge there exists no similar MBC method for any directional distributions.

The proposed unsupervised method consists of several independent contributions such as: (a) Bregman Soft Clustering (Banerjee et al., 2005b) algorithm for vMF Mixture Models (vMFMM) and Watson Mixture Models (WMM); (b) Hierarchical Agglomerative Clustering (HAC) on expectation parameter space using Bregman Divergence (BD) and (c) empirical model selection using information criteria or WPLR-τ method. Now let us discuss each of them individually.

- Compare to the traditional EM based soft clustering methods, Bregman Soft Clustering (BSC) has already proved as an efficient algorithm with additional benefits (Banerjee et al., 2005b). There exists no BSC method for vMFMM and WMM and we are the first to propose one. We empirically validate that to cluster directional and axial data our proposed BSC-vMFMM and BSC-WMM algorithms are better compare to other clustering methods.

- The HAC on the source and natural parameter space of GMM is already proposed in the context of mixture model simplification (Goldberger and Roweis, 2004; Garcia and Nielsen, 2010) and hybrid Model Based Clustering method (Zhong and Ghosh, 2003). We applied it in the expectation parameter space of

vMFMM and WMM using BD. Therefore, our HAC method is also a simplification method for vMFMM and WMM. Note that, there exists no vMFMM and WMM simplification method. The HAC procedure is independent and is able to handle any space of parameters. Therefore, one can easily plug the method in an external soft clustering method (vMFMM and WMM). In such case, this method behaves similar to the hybrid Model Based Clustering method (Zhong and Ghosh, 2003).

- In order to select best model, we applied widely used parsimony based approach (Melnykov and Maitra, 2010; Alata and Quintard, 2009; Biernacki et al., 2000). Beside this, we propose a novel model selection approach (called WPLR-τ). Compare to the parsimony based approaches, WPLR-τ exhibits better compromise for both the simulated and real data. Moreover, we have shown that the ÏĎ parameter exhibits similar behavior of the bandwidth parameter of the non-parametric Mean Shift method.

The above discussion reveals that, for directional and axial data our method can be an interesting tool for clustering, model simplification, model selection and eventually unsupervised classification. Hence we believe that the proposed method will be an interesting tool for the machine learning, data mining and pattern recognition community.

As an application we have shown its usability for depth image analysis through clustering. We demonstrated that our method can be used as a potential tool to perform unsupervised segmentation of the indoor scene. They are able to provide piecewise planar segments which are important geometric primitives of man-made structures, such as the indoor environments. Moreover, we have shown that the methods are able to provide sufficient distinctions among the planar and non-planar surfaces via the concentration parameters. The findings in this work were very helpful for us to develop a novel RGB-D segmentation method based on joint clustering and region merging.

5.1.2 Joint Clustering and Region merging for RGB-D segmentation

The observations from the initially developed clustering methods revealed that we should consider heterogeneous features, such as color, position, depth, etc. in order to obtain better results in scene analysis and understanding. Therefore, we focused on developing a joint clustering method with the aim to fuse different features together.

However, we were also interested to exploit the interesting findings from our previous work. To this aim, in Chapter 4, we developed a RGB-D scene analysis method, which first performs a joint clustering of the color-position-axial features, and then applies a region merging based on planar statistics. The individual contributions of this work can be highlighted as follows:

- A statistical image generation model for RGB-D data that incorporates both color and geometric properties of the scene. Such model provides an interesting formulation of how different features can be incorporated into a single model with simple assumptions. Moreover, this type of model is very flexible to extend with additional features.

- A novel and efficient probabilistic joint clustering method based on Bregman Soft Clustering (Banerjee et al., 2005b) approach. The proposed method is a solution to cluster image pixels based on the proposed image generation model. Such clustering algorithm is computationally efficient and expressive to provide better interpretation in terms of individual features. For example, it provides the planar statistics which can be used efficiently for scene interpretation by incorporating region merging.

- A statistical region merging method (Nock and Nielsen, 2004) based on certain region merging predicates. This method can be incorporated independently with any other existing indoor RGB-D scene segmentation method. This method used the planar statistics from the clustering method.

- A benchmark on the NYU Depth Dataset V2 (Silberman et al., 2012) for unsupervised scene segmentation. At present no such benchmark exists in literature for unsupervised tasks.

The method presented in Chapter 4 shows how we efficiently extended the previously proposed method by exploiting the findings in Chapter 3. Moreover, this method opens many interesting perspectives for further improving the efficiency of the scene analysis task.

5.2 Future Work

There are numerous perspectives and future extensions of the methods that naturally follow on from the work in this thesis. Let us now discuss them individually.

5.2.1 Extension of Model Based Clustering methods

Other Directional Distributions

The proposed Model Based Clustering methods can be extended for the other directional distributions, such as Kent, Bingham, etc. This might be interesting as the Kent and Bingham distributions incorporate more parameters which naturally allow them to provide better model data with complex structure of the data. Note that, the shape of both von Mises-Fisher and Watson distribution is circular around the mean direction, see Section 3.2 of Chapter 3. The Kent allows having elliptical shape of the clusters via additional parameters. Therefore, in certain applications it would be effective to use the Kent distribution rather than the von Mises-Fisher distribution.

Other Probability Distributions

Beside the extension to the directional distributions, one can extend the Model Based Clustering method proposed in Chapter 2 for any probability distributions which belongs to the Exponential Family of Distributions (EFD). Note that, the extension can be accomplished once the canonical EFD form for that distribution is derived and the associated Bregman Divergence is computed.

Spatially Variant Methods

Spatial smoothness is one of the most widely considered constraints for image analysis. There exists several methods based on spatially variant finite mixture models (Nguyen and Wu, 2013). Since, the core assumption of our proposed method is a finite mixture model therefore one can consider to extend the method by adding spatial constraints.

Selecting Number of Component

Selection of number of components remains a challenging problem in clustering. We believe that, it is necessary to invest more effort on finding unique solution for component selection such that it can be applied globally to perform clustering with any probability distribution and particularly clustering real data which contains significant amount of noise.

Extend Applicability

In order to be focused on the core objectives of this thesis, we did not evaluate the applicability of the proposed method for other applications. However, we know that such methods are commonly employed for a variety of different domains. Therefore,

in future we should consider applying them for different tasks associated with different domains.

5.2.2 RGB-D segmentation method

Extend Joint Clustering with Additional Information

For image segmentation, it would be interesting to extend the proposed joint clustering method by adding different constraints, such as spatial smoothness and by adding information, such as contour, texture, etc.

The joint clustering method sometimes exhibits sub-standard performance due to the improper initialization. It should be investigated further to avoid such initialization.

Extend Region Merging Method

Currently, the region merging method only considers the planar information. This method can be easily extended by incorporating color information. At present, color information from Kinect exhibits challenges due to the presence of noise as well as due to the presence of shadows in the scene. One must consider first to reduce their effects and then incorporate color based merging procedure.

One may consider enhancing the influence of edges during region merging. At present the edges associated to the regions are obtained naively from the initially clustered regions. We observed numerous artifacts of such edges. Therefore, it should be properly addressed by incorporating a pre-processing step prior to region merging.

Conclusion

Dans cette thèse, nous avons proposé de nouvelles méthodes non supervisées pour la classification d'images 3D et la segmentation prenant en compte de manière conjointe les informations de couleur et de profondeur. A cet effet, nous avons formulé l'hypothèse que les normales aux surfaces dans les images 3D sont des éléments à prendre en compte pour leur analyse, et leurs distributions sont modélisable à l'aide de lois de mélange. Nous avons utilisé la méthode dite « Bregman Soft Clustering » afin d'être efficace d'un point de vue calculatoire. De plus, nous avons étudié plusieurs lois de probabilités permettant de modéliser les distributions de directions: la loi de von Mises-Fisher et la loi de Watson. Les méthodes de classification « basées modèles » proposées sont ensuite validées en utilisant des données de synthèse puis nous avons montré leur intérêt pour l'analyse des images 3D (ou de profondeur). Une nouvelle méthode de segmentation d'images couleur et profondeur, appelées aussi images RGB-D, exploitant conjointement la couleur, la position 3D, et la normale locale est alors développée par extension des précédentes méthodes et en introduisant une méthode statistique de fusion de régions « planes » à l'aide d'un graphe. Les résultats ont montré que la méthode proposée donne des résultats au moins comparables aux méthodes de l'état de l'art tout en demandant moins de temps de calcul. De plus, elle ouvre des perspectives nouvelles pour la fusion non supervisée des informations de couleur et de géométrie. Nous sommes convaincus que les méthodes proposées dans cette thèse pourront être utilisées pour la classification d'autres types de données comme la parole, les données d'expression en génétique, etc. Elles devraient aussi permettre la réalisation de tâches complexes comme l'analyse conjointe de données contenant des images et de la parole.

Bibliography

Olivier Alata and Ludovic Quintard. Is there a best color space for color image characterization or representation based on multivariate gaussian mixture model? *Computer Vision and Image Understanding*, 113(8):867–877, 2009.

Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011.

David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

Arindam Banerjee, Inderjit S Dhillon, Joydeep Ghosh, and Suvrit Sra. Clustering on the unit hypersphere using von mises-fisher distributions. In *Journal of Machine Learning Research*, pages 1345–1382, 2005a.

Arindam Banerjee, Srujana Merugu, Inderjit S Dhillon, and Joydeep Ghosh. Clustering with bregman divergences. *The Journal of Machine Learning Research*, 6: 1705–1749, 2005b.

Mark Bangert, Philipp Hennig, and Uwe Oelfke. Using an infinite von mises-fisher mixture model to cluster treatment beam directions in external radiation therapy. In *Machine Learning and Applications (ICMLA), 2010 Ninth International Conference on*, pages 746–751. IEEE, 2010.

Jonathan T Barron and Jitendra Malik. Intrinsic scene properties from a single rgb-d image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17–24. IEEE, 2013.

Jean-Patrick Baudry, Adrian E Raftery, Gilles Celeux, Kenneth Lo, and Raphael Gottardo. Combining mixture components for clustering. *Journal of Computational and Graphical Statistics*, 19(2):332–353, 2010.

Laurent Bergé, Charles Bouveyron, and Stéphane Girard. Hdclassif: An r package for model-based clustering and discriminant analysis of high-dimensional data. *Journal of Statistical Software*, 46(6):1–29, 2012.

Abhir Bhalerao and Carl-Fredrik Westin. Hyperspherical von mises-fisher mixture (hvmf) modelling of high angular resolution diffusion mri. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007*, pages 236–243. Springer, 2007.

Christophe Biernacki, Gilles Celeux, and Gérard Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725, 2000.

Christophe Biernacki, Gilles Celeux, and Gérard Govaert. Choosing starting values for the em algorithm for getting the highest likelihood in multivariate gaussian mixture models. *Computational Statistics & Data Analysis*, 41(3):561–575, 2003.

Avleen S Bijral, Markus Breitenbach, and Gregory Z Grudic. Mixture of watson distributions: a generative model for hyperspherical embeddings. In *International Conference on Artificial Intelligence and Statistics*, pages 35–42, 2007.

Christopher M Bishop. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.

Jean-Daniel Boissonnat, Frank Nielsen, and Richard Nock. Bregman voronoi diagrams. *Discrete & Computational Geometry*, 44(2):281–307, 2010.

Christian Buchta, Martin Kober, Ingo Feinerer, and Kurt Hornik. Spherical k-means clustering. *Journal of Statistical Software*, 50(10):1–22, 2012.

Kenneth P Burnham and David R Anderson. *Model selection and multi-model inference: a practical information-theoretic approach*. Springer, 2002.

RyanP. Cabeen, MarkE. Bastin, and DavidH. Laidlaw. Estimating constrained multi-fiber diffusion mr volumes by orientation clustering. In Kensaku Mori, Ichiro Sakuma, Yoshinobu Sato, Christian Barillot, and Nassir Navab, editors, *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2013*, volume 8149 of *Lecture Notes in Computer Science*, pages 82–89. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-40810-6.

Ming-Ming Cheng, Guo-Xin Zhang, Niloy J. Mitra, Xiaolei Huang, and Shi-Min Hu. Global contrast based salient region detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 409–416, 2011.

Anoop Cherian, Vassilios Morellas, Nikolaos Papanikolopoulos, and Saad J Bedros. Dirichlet process mixture models on symmetric positive definite matrices for appearance clustering in video surveillance applications. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3417–3424. IEEE, 2011.

Jean-Pierre Da Costa, Frédéric Galland, Antoine Roueff, and Christian Germain. Unsupervised segmentation based on von mises circular distributions for orientation estimation in textured images. *Journal of Electronic Imaging*, 21(2):021102–1, 2012.

Carlo Dal Mutto, Pietro Zanuttigh, and Guido M Cortelazzo. Fusion of geometry and color information for scene segmentation. *IEEE Journal of Selected Topics in Signal Processing*, 6(5):505–521, 2012a.

Carlo Dal Mutto, Pietro Zanuttigh, and Guido M Cortelazzo. *Time-of-flight cameras and microsoft KinectŹ*. Springer, 2012b.

Inderjit S Dhillon and Suvrit Sra. Modeling data using directional distributions. Technical report, Technical Report TR-03-06, Department of Computer Sciences, The University of Texas at Austin. URL ftp://ftp. cs. utexas. edu/pub/techreports/tr03-06. ps. gz, 2003.

Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

Basura Fernando, Elisa Fromont, Damien Muselet, and Marc Sebban. Supervised learning of gaussian mixture models for visual vocabulary generation. *Pattern Recognition*, 45(2):897–907, 2012.

Mario A. T. Figueiredo and Anil K. Jain. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3): 381–396, 2002.

Jaime RS Fonseca and Margarida GMS Cardoso. Mixture-model cluster analysis using information theoretical criteria. *Intelligent Data Analysis*, 11(2):155–173, 2007.

C. Fraley, A. E. Raftery, T. B. Murphy, and L. Scrucca. mclust version 4 for r: Normal mixture modeling for model-based clustering, classification, and density estimation. Technical Report 597, Department of Statistics, University of Washington., 2012.

Chris Fraley and Adrian E Raftery. How many clusters? which clustering method? answers via model-based cluster analysis. *The computer journal*, 41(8):578–588, 1998.

Chris Fraley and Adrian E Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458): 611–631, 2002.

Chris Fraley and Adrian E Raftery. Model-based methods of classification: using the mclust software in chemometrics. *Journal of Statistical Software*, 18(6):1–13, 2007.

Jordi Freixenet, Xavier Muñoz, David Raba, Joan Martí, and Xavier Cufí. Yet another survey on image segmentation: Region and boundary information integration. In *Computer VisionâĂŤECCV 2002*, pages 408–422. Springer, 2002.

Vincent Garcia and Frank Nielsen. Simplification and hierarchical representations of mixtures of exponential families. *Signal Processing*, 90(12):3197–3212, 2010.

Vincent Garcia, Frank Nielsen, and Richard Nock. Levels of details for gaussian mixture models. In *Computer Vision–ACCV 2009*, pages 514–525. Springer, 2010.

Jared Glover, Gary Bradski, and Radu Bogdan Rusu. Monte carlo pose estimation with quaternion kernels and the bingham distribution. *Robotics: Science and Systems VII*, page 97, 2012.

Jacob Goldberger and Sam T Roweis. Hierarchical clustering of a mixture model. In *Advances in Neural Information Processing Systems*, pages 505–512, 2004.

Siddharth Gopal and Yiming Yang. Von mises-fisher clustering models. In *Proceedings of The 31st International Conference on Machine Learning*, pages 154–162, 2014.

Costantino Grana, Daniele Borghesani, and Rita Cucchiara. Describing texture directions with von mises distributions. In *ICPR*, pages 1–4, 2008.

Saurabh Gupta, Pablo Arbelaez, and Jitendra Malik. Perceptual organization and recognition of indoor scenes from rgb-d images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 564–571. IEEE, 2013.

Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton. Enhanced computer vision with microsoft kinect sensor: A review. *IEEE Transactions on Cybernetics*, 43: 1318 – 1334, 2013.

Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments. *The International Journal of Robotics Research*, 31(5):647–663, 2012.

C Herrera, Juho Kannala, Janne Heikkilä, et al. Joint depth and color camera calibration with distortion correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(10):2058–2064, 2012.

John R Hershey and Peder A Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 4, pages IV–317. IEEE, 2007.

Dirk Holz, Stefan Holzer, Radu Bogdan Rusu, and Sven Behnke. Real-time plane segmentation using rgb-d cameras. In *RoboCup 2011: Robot Soccer World Cup XV*, pages 306–317. Springer, 2012.

E Andres Houseman, Brock C Christensen, Ru-Fang Yeh, Carmen J Marsit, Margaret R Karagas, Margaret Wrensch, Heather H Nelson, Joseph Wiemels, Shichun Zheng, John K Wiencke, et al. Model-based clustering of dna methylation array data: a recursive-partitioning algorithm for high-dimensional data arising as a mixture of beta distributions. *BMC bioinformatics*, 9(1):365, 2008.

Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.

Anil K Jain, M Narasimha Murty, and Patrick J Flynn. Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323, 1999.

Alfons Juan and Enrique Vidal. On the use of bernoulli mixture models for text classification. *Pattern Recognition*, 35(12):2705–2710, 2002.

Zoltan Kato. Segmentation of color images via reversible jump mcmc sampling. *Image and Vision Computing*, 26(3):361–371, 2008.

Sean Keane, Jonathan Hall, and Phoenix Perry. *Meet the Kinect: An Introduction to Programming Natural User Interfaces*. Apress, 2011.

Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.

Takumi Kobayashi and Nobuyuki Otsu. Von mises-fisher mean shift for clustering on a hypersphere. In *20th International Conference on Pattern Recognition (ICPR)*, pages 2130–2133. IEEE, 2010.

Hema Swetha Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. Semantic labeling of 3d point clouds for indoor scenes. In *NIPS*, volume 1, page 4, 2011.

Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824. IEEE, 2011.

Douglas Lanman and Gabriel Taubin. Build your own 3d scanner: 3d photography for beginners. In *ACM SIGGRAPH 2009 Courses*, page 8. ACM, 2009.

Meizhu Liu, Baba C Vemuri, S-I Amari, and Frank Nielsen. Shape retrieval using hierarchical total bregman soft clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(12):2407–2419, 2012.

Yi Ma, Harm Derksen, Wei Hong, and John Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1546–1562, 2007.

Ranjan Maitra and Ivan P Ramler. A k-mean-directions algorithm for fast clustering of data on the sphere. *Journal of Computational and Graphical Statistics*, 19(2), 2010.

Kanti V Mardia and Peter E Jupp. *Directional statistics*, volume 494. Wiley. com, 2009.

Wendy L Martinez, Angel Martinez, and Jeffrey Solka. *Exploratory data analysis with MATLAB, Second Edition*. CRC Press, 2010.

Adolfo Martínez-Usó, Filiberto Pla, and Pedro García-Sevilla. Unsupervised colour image segmentation by low-level perceptual grouping. *Pattern Analysis and Applications*, 16(4):581–594, 2013.

Tim McGraw, Baba Vemuri, Robert Yezierski, and Thomas Mareci. Segmentation of high angular resolution diffusion mri modeled as a field of von mises-fisher mixtures. In *Computer Vision–ECCV 2006*, pages 463–475. Springer, 2006.

Geoffrey McLachlan and David Peel. *Finite mixture models*. Wiley. com, 2004.

Volodymyr Melnykov and Ranjan Maitra. Finite mixture models and model-based clustering. *Statistics Surveys*, 4:80–116, 2010.

Luc ; Vergauwen Maarten Moons, Theo ; Van Gool. *3d Reconstruction from Multiple Images: Part 1: Principles*. Now Publishers Inc, 2009.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. The MIT Press, 2012.

Thanh Minh Nguyen and Qm Wu. Fast and robust spatially constrained gaussian mixture model for image segmentation. *IEEE transactions on circuits and systems for video technology*, 23(4):621–635, 2013.

Thanh Minh Nguyen and Qm Jonathan Wu. Robust student's-t mixture model with spatial constraints and its application in medical image segmentation. *IEEE Transactions on Medical Imaging*, 31(1):103–116, 2012.

Frank Nielsen and Vincent Garcia. Statistical exponential families: A digest with flash cards. *CoRR*, abs/0911.4863:http://arxiv.org/abs/0911.4863, 2009.

Yuzhen Niu, Yujie Geng, Xueqing Li, and Feng Liu. Leveraging stereopsis for saliency analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 454–461. IEEE, 2012.

Richard Nock and Frank Nielsen. Statistical region merging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1452–1458, 2004.

Martin Paluszewski and Thomas Hamelryck. Mocapy++-a toolkit for inference and learning in dynamic bayesian networks. *BMC bioinformatics*, 11(1):126, 2010.

David Peel, William J Whiten, and Geoffrey J McLachlan. Fitting mixtures of kent distributions to aid in joint set identification. *Journal of the American Statistical Association*, 96(453):56–63, 2001.

Bo Peng and David Zhang. Automatic image segmentation by dynamic region merging. *IEEE Transactions on Image Processing*, 20(12):3592–3605, 2011.

Haim Permuter, Joseph Francos, and Ian Jermyn. A study of gaussian mixture models of color and texture features for image classification and segmentation. *Pattern Recognition*, 39(4):695–706, 2006.

Andrea Prati, Simone Calderara, and Rita Cucchiara. Using circular statistics for trajectory shape analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

Narges Sharif Razavian, Hetunandan Kamisetty, and Christopher James Langmead. The von mises graphical model: structure learning. *Carnegie Mellon University School of Computer Science Technical Report*, 2011.

Xiaofeng Ren, Liefeng Bo, and Dieter Fox. Rgb-(d) scene labeling: Features and algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2759–2766. IEEE, 2012.

Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1):19–41, 2000.

Radu Bogdan Rusu. *Semantic 3D Object Maps for Everyday Robot Manipulation*. Springer, 2013.

Stan Salvador and Philip Chan. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *16th IEEE International Conference on Tools with Artificial Intelligence.*, pages 576–584. IEEE, 2004.

Ali Sefidpour and Nizar Bouguila. Spatial color image segmentation based on finite non-gaussian mixture models. *Expert Systems with Applications*, 39(10):8993–9001, 2012.

Giorgos Sfikas, Christophoros Nikou, and Nikolaos Galatsanos. Robust image segmentation with mixtures of student's t-distributions. In *IEEE International Conference on Image Processing*, volume 1, pages I–273. IEEE, 2007.

Fei Sha and Lawrence K Saul. Large margin gaussian mixture modeling for phonetic classification and recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages I–I. IEEE, 2006.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012*, pages 746–760. Springer, 2012.

Mehrez Souden, Keisuke Kinoshita, and Tomohiro Nakatani. An integration of source location cues for speech clustering in distributed microphone arrays. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 111–115. IEEE, 2013.

Suvrit Sra. A short note on parameter approximation for von mises-fisher distributions: and a fast implementation of i s (x). *Computational Statistics*, 27(1):177–190, 2012.

Suvrit Sra and Dmitrii Karp. The multivariate watson distribution: Maximum-likelihood estimation and other aspects. *J Multivar Anal*, 114:256 – 269, 2013.

Johannes Strom, Andrew Richardson, and Edwin Olson. Graph-based segmentation for colored 3d laser point clouds. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2131–2136. IEEE, 2010.

Richard Szeliski. *Computer vision: algorithms and applications*. Springer, 2011.

Camillo J Taylor and Anthony Cowley. Segmentation and analysis of rgb-d data. In *Proceedings of Robotics Science and Systems (RSS)*, 2011.

Camillo J Taylor and Anthony Cowley. Parsing indoor scenes using rgb-d imagery. *Robotics: Science and Systems VIII*, pages 401–408, 2013.

Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

Alain Trémeau and Philippe Colantoni. Regions adjacency graph applied to color image segmentation. *IEEE Transactions on Image Processing*, 9(4):735–744, 2000.

Jakob J Verbeek, Nikos Vlassis, and B Kröse. Efficient greedy learning of gaussian mixture models. *Neural computation*, 15(2):469–485, 2003.

Nikos Vlassis and Aristidis Likas. A greedy em algorithm for gaussian mixture learning. *Neural Processing Letters*, 15(1):77–87, 2002.

Dang Hai Tran Vu and Reinhold Haeb-Umbach. Blind speech separation employing directional statistics in an expectation maximization framework. In *Int. Con. on Acoustics, Speech, and Signal Processing*. IEEE, 2010.

Geoffrey S Watson. The theory of concentrated langevin distributions. *Journal of Multivariate Analysis*, 14(1):74–82, 1984.

Ron Wehrens, Lutgarde MC Buydens, Chris Fraley, and Adrian E Raftery. Model-based clustering for image segmentation and large datasets via sampling. *Journal of Classification*, 21(2):231–253, 2004.

Na Zhang. *Fibre Processes and their Applications*. PhD thesis, 2013.

Zhengyou Zhang. Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2): 4–10, 2012.

Qinpei Zhao, Ville Hautamaki, and Pasi Fränti. Knee point detection in bic for detecting the number of clusters. In *Advanced Concepts for Intelligent Vision Systems*, pages 664–673. Springer, 2008.

Shi Zhong and Joydeep Ghosh. A unified framework for model-based clustering. *The Journal of Machine Learning Research*, 4:1001–1037, 2003.

Resumé

L'accès aux séquences d'images 3D s'est aujourd'hui démocratisé, grâce aux récentes avancées dans le développement des capteurs de profondeur ainsi que des méthodes permettant de manipuler des informations 3D à partir d'images 2D. De ce fait, il y a une attente importante de la part de la communauté scientifique de la vision par ordinateur dans l'intégration de l'information 3D. En effet, des travaux de recherche ont montré que les performances de certaines applications pouvaient être améliorées en intégrant l'information 3D. Cependant, il reste des problèmes à résoudre pour l'analyse et la segmentation de scènes intérieures comme (a) comment l'information 3D peut-elle être exploitée au mieux? et (b) quelle est la meilleure manière de prendre en compte de manière conjointe les informations couleur et 3D? Nous abordons ces deux questions dans cette thèse et nous proposons de nouvelles méthodes non supervisées pour la classification d'images 3D et la segmentation prenant en compte de manière conjointe les informations de couleur et de profondeur. A cet effet, nous formulons l'hypothèse que les normales aux surfaces dans les images 3D sont des éléments à prendre en compte pour leur analyse, et leurs distributions sont modélisable à l'aide de lois de mélange. Nous utilisons la méthode dite « Bregman Soft Clustering » afin d'être efficace d'un point de vue calculatoire. De plus, nous étudions plusieurs lois de probabilités permettant de modéliser les distributions de directions: la loi de von Mises-Fisher et la loi de Watson. Les méthodes de classification « basées modèles » proposées sont ensuite validées en utilisant des données de synthèse puis nous montrons leur intérêt pour l'analyse des images 3D (ou de profondeur). Une nouvelle méthode de segmentation d'images couleur et profondeur, appelées aussi images RGB-D, exploitant conjointement la couleur, la position 3D, et la normale locale est alors développée par extension des précédentes méthodes et en introduisant une méthode statistique de fusion de régions « planes » à l'aide d'un graphe. Les résultats montrent que la méthode proposée donne des résultats au moins comparables aux méthodes de l'état de l'art tout en demandant moins de temps de calcul. De plus, elle ouvre des perspectives nouvelles pour la fusion non supervisée des informations de couleur et de géométrie. Nous sommes convaincus que les méthodes proposées dans cette thèse pourront être utilisées pour la classification d'autres types de données comme la parole, les données d'expression en génétique, etc. Elles devraient aussi permettre la réalisation de tâches complexes comme l'analyse conjointe de données contenant des images et de la parole.