COLOR IMAGE SEGMENTATION BY INTEGRATING SPATIAL INFORMATION USING SEMI-BOUNDED FINITE MIXTURE MODELS

Jaspreet Singh Kalsi

A thesis
in
The Department
of
The Concordia Institute for Information Systems Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Applied Science
(Information Systems Security)
Concordia University
Montréal, Québec, Canada

March 2019

Concordia University
School of Graduate Studies

This is to certify that the thesis prepared

By: **Jaspreet Singh Kalsi**

Entitled: **Color Image Segmentation by Integrating Spatial Information using Semi-Bounded Finite Mixture Models**

and submitted in partial fulfillment of the requirements for the degree of

Degree of Master of Applied Science
(Information Systems Security)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

Dr. Walter Lucia _____ Chair

Dr. Nizar Bouguila _____ Supervisor

Dr. Jamal Bentahar _____ CIISE Examiner

Dr. Farjad Shadmehri _____ External Examiner

Approved _____

Dr. Chadi Assi Graduate Program Director

2019.02.28 _____

Dr. Amir Asif, Dean
Faculty of Engineering and Computer Science

Abstract

Color Image Segmentation by Integrating Spatial Information using Semi-Bounded Finite Mixture Models

Jaspreet Singh Kalsi

In computer vision, image segmentation plays foundational role. Innumerable techniques, such as active contour, graph-cut-based, model-based, machine learning, and clustering-based methods have been proposed for tackling the image segmentation problem. But, none of them is universally applicable. Thus, the hunt for optimized and robust models for image segmentation is still under-process and also an open question. The challenges faced in image segmentation are the integration of spatial information, finding the exact number of clusters (M), and to segment the image smoothly without any inaccuracy specially in the presence of noise, a complex background, low contrast and, inhomogeneous intensity. The use of finite mixture model (FMMs) for image segmentation is very popular approach in the field of computer vision. The application of image segmentation using FMM ranges from automatic number plate recognition, content-based image retrieval, texture recognition, facial recognition, satellite imagery etc. Image segmentation using FMM undergoes some problems. FMM-based image segmentation considers neither spatial correlation among the peer pixels nor the prior knowledge that the adjacent pixels are most likely belong to the same cluster. Also, color images are sensitive to illumination and noise. To overcome these limitations, we have used three different methods for integrating spatial information with FMM. First method uses the prior knowledge of M. In second method, we have used Markov Random Field (MRF). Lastly, in third, we have used weighted geometric and arithmetic mean template. We have implemented these methods with inverted Dirichlet mixture model (IDMM), generalized inverted Dirichlet mixture model (GIDMM) and inverted Beta Liouville mixture model (IBLMM). For experimentation, the Berkeley 500 (BSD500) and MIT's Computational Visual Cognition Laboratory (CVCL) datasets are employed. Furthermore, to compare the image segmentation results, the outputs of IDMM, GIDMM, and IBLMM are compared with each other, using segmentation performance evaluation metrics.

Acknowledgments

I wish to thank everyone who helped me to complete this thesis. I would like to express my very profound gratitude to my supervisor Prof. Nizar Bouguila. The completion of this thesis would have not been possible without his endless support, effort, encouragement and most importantly patience. I will be forever grateful to him for giving me the opportunity to work under his guidance.

I would like to offer my special thanks to my uncle Dr. Jagpal Singh Ubhi who supported and helped me in different steps of my long journey to Canada and starting my education.

I would like to thank all my friendly fellow lab mates who made our lab a delightful and convenient place for research, in particular many thanks goes to Muhammad Azam, Narges, Kamal, Hieu, Omar, Soudhe, Basim, Meeta and Eddie that I took benefit of their discussions during my research.

It is noteworthy to mention my gratitude to the faculty members and administrative staff of CIISE department, which were always ready to help with smiling faces.

My gratitude extends to my friends Harminder Singh, Amanpreet Kaur and Mohit Thakur for their continuous support and encouragement.

Last but not the least, my gratitude goes to my family for their endless love and support through- out my life, without their support, simply I could not do this. I also extend my warmest thanks to my little sister.....

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Over the past couple of decades, computer vision experienced tremendous advancement as well as growth. Image segmentation has received attention in satellite imagery [12], medical image analysis [6], texture recognition [4], facial recognition [13], automatic number plate recognition [65], content-based image retrieval [3], humans skin detection [75] etc. The success and failure of any application depends on the model used. Extensive research has been conducted to propose suitable models for image segmentation but still, it is an open challenge [15, 11]. The main problems faced in image segmentation are finding the exact number of regions and integrating spatial information. FMM have been proven to be one of the most strong and flexible tools for model-based clustering and have seen a real boost in popularity. But, image segmentation using model-based clustering undergoes few difficulties. Model-based image clustering does not consider spatial correlation among the peer pixels. Also, color images are sensitive to noise as well as illumination [14]. It is noteworthy that the Gaussian mixture model (GMM) is not a proper tool to express the latent structure of non-Gaussian data. Recently, other distributions which are more flexible have been considered as a powerful alternative [18, 19].

1.2 Objectives

The main goal of this thesis is to propose algorithms using novel finite mixture modelling approach by focusing on few capable distributions namely inverted Dirichlet (ID), generalized inverted Dirichlet (GID), and inverted Beta Liouville. We developed a learning framework based on maximum likelihood estimation to infer the optimal parameters of our proposed mixture models and applied it to address following challenging issues:

1. Choosing flexible mixture densities which offer more flexibility to approximate many shapes.

2. The estimation of parameters as one of the compelling and critical challenges when deploying mixture models.

3. Assessment and validation of the feasibility and effectiveness of the proposed models by experimental results involving well-known and standardized datasets.

In this work, we introduce unsupervised learning based methods which incorporates the spatial information with IDMM, GIDMM, and IBLMM which could be applied in various challenging computer vision problems. Our proposed learning framework will deploy deterministic and efficient techniques such as Maximum likelihood (ML), Expectation maximization (EM) and Newton Raphson methods, followed by the use of segmentation evaluation metrics for the comparison of the results.

1.3 Contributions

Our major contributions in this thesis are as follows:

- **Incorporation of spatial information using indirect information of M as prior knowledge**: We propose an approach for the integration of spatial information into two different FMM namely IDMM and GIDMM to produce more meaningful, robust and, smooth regions in color image segmentation while offering more flexibility and ease of use for data modeling in comparison to the well popular and commonly used GMM.

- **Incorporation of spatial information using MRF**: In this contribution, the spatial information is incorporated with different FMM namely IDMM and GIDMM using MRF.

- **Incorporation of spatial information using weighted mean template**: In this contribution, the spatial information is integrated with different FMM namely IDMM, GIDMM, and IBL using weighted geometric and arithmetic conditional mean templates. Moreover, weighted geometric and arithmetic prior mean templates are also employed for the same.

- **Qualitative analysis of segmentation results using performance evaluation metrics**: We compare the segmentation results from the proposed algorithms with each other using performance evaluation metrics. We have used evaluation metrics such as Jaccard Similarity Score, Calinski-Harabaz Index etc.

- **Revealing the effect of color space on color image segmentation**: We compare the effect of two different color spaces namely rgb and $l_1l_2l_3$ on the color image segmentation process and we show that different color spaces can lead to different segmentation results.

1.4 Thesis Overview

The rest of the thesis is organized as follows:

1. In chapter 2, we present the incorporation of spatial information by using indirect knowledge of M as prior information by using MRF with IDMM and GIDMM. Afterwards in experiments, we compare the segmentation outputs from ID and GID with GMM using performance evaluation metrics.

2. Chapter 3 is devoted to the integration of spatial information using weighted conditional and prior mean templates with IDMM, GIDMM and IBLMM. Futhermore, these mean templates consist of WGCMT (Weighted geometric conditional mean template), WACMT (Weighted arithmetic conditional mean template), WGPMT (Weighted geometric prior mean template), and WAPMT (Weighted arithmetic prior mean template). For experimentation, we compared the segmentation outputs from IDMM, GIDMM, and IBLMM with each other using segmentation performance evaluation metrics. Lastly, we compare the effect of rgb and $l_1l_2l_3$ color spaces on the segmentation outputs with the proposed algorithms.

3. Finally in chapter 4, we conclude our work and suggest future works.

Chapter 2

Color Image Segmentation Using Generalized Inverted Finite Mixture Models By Integrating Spatial Information

2.1 Introduction

In this chapter, we propose two methods. In first one, we adapt a segmentation approach proposed in [16] where the authors have used indirect information about M as prior knowledge. This prior knowledge does not give any direct value of M. Instead, it provides an indirect information which can further be used to estimate the exact value of M. In second method, finite mixture model based on MRF [10] is used. MRF ensures the spatial correlation between the pixels. It is given as follows:

$$\mathcal{W}(\Pi) = \mathcal{N}^{-1} \exp\left\{ -\frac{1}{\mathcal{T}}\mathcal{U}(\Pi)\right\} \tag{1}$$

where $\mathcal{N}$ is a normalizing constant, $\mathcal{T}$ is a Temperature constant and $\mathcal{U}(\Pi)$ is the smoothing prior. Many researchers have performed extensive research on determining the optimized value of smoothing prior. But, it has been found that most of them are very complex and requires a large amount of computational power. Below is an

example of one of such smoothing priors:

$$\mathcal{U}(\Pi) = \sum_{n=1}^{N} \sum_{j=1}^{K} \sum_{s=1}^{S} \left[\frac{1}{2} \log \beta_{js}^2 - \frac{1}{2} \frac{\left(\sum_{m \in \delta_i} (\pi_{ij} - \pi_{mj}) \right)^2}{\beta_{js}^2} \right] \tag{2}$$

where S is defined as the total number of directions under consideration, β_{js} is a variable parameter and π is a mixing parameter explained in later section. As we can observe, (2) is very complex and thus, the M-Step of the EM algorithm [8] cannot be applied directly to prior distribution Π. In order to remove these demerits, a novel factor has been given by [10] which is discussed in later section.

The remaining chapter is organized as follows. In section 2.2, the general definition of traditional FMM is presented in detail. Section 2.3 is devoted to the incorporation of Spatial Information using prior knowledge of M with IDMM and GIDMM. This section contains the pdfs of ID and GID, following by their mixture models learning. In section 2.4, the spatial information is integrated with IDMM and GIDMM using MRF approach. Section 2.5 summarizes the proposed algorithm. Lastly, section 2.6, contains the experimental results.

2.2 Finite Mixture Model

Consider an image $\mathcal{X} = \{\vec{X_1}, \ldots, \vec{X_N}\}$ consisting of N pixels, where each pixel $\vec{X_n}$ has a dimension D such that $\vec{X_n} = (X_{n1}, \ldots, X_{nD})$. We assume that $\mathcal{X}$ can be segmented into M clusters and thus, it is appropriate to use distribution as:

$$p(\vec{X}|\vec{\Theta}) = \sum_{j=1}^{M} \pi_j p(\vec{X}|\vec{\theta_j}) \tag{3}$$

where each cluster j has a weight π_j, $\sum_{j=1}^{M} \pi_j = 1$. M is the number of components in the FMM. $p(\vec{X}|\vec{\theta_j})$ is the density associated with cluster j and $\vec{\Theta} = (\pi_1, \ldots, \pi_M, \vec{\theta}_1, \ldots, \vec{\theta}_M)$ is the set of all the mixture parameters.

2.3 Integration of Spatial Information using Prior knowledge of M with IDMM and GIDMM.

2.3.1 The Probability Density Function of ID

If $\vec{X}$ is a positive vector which consists of D dimensions and following an ID distribution, then it has a joint density function given as follows [52]:

$$p(\vec{X}|\vec{\alpha}) = \frac{\Gamma(|\vec{\alpha}|)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \prod_{d=1}^{D+1} X_d^{\alpha_d - 1} \left(1 + |\vec{X}|\right)^{-|\vec{\alpha}|} \tag{4}$$

where $|\vec{X}| = \sum_{d=1}^{D} X_d$, each $X_d > 0$. The parameter of ID is $\vec{\alpha} = [\alpha_1, \ldots, \alpha_{D+1}]$, $|\vec{\alpha}| = \sum_{d=1}^{D+1} \alpha_d$, $\alpha_d > 0$ where $d = 1, \ldots, D+1$. The mean and the variance of ID are given as follows:

$$E(X_d) = \frac{\alpha_d}{\alpha_{D+1} - 1} \tag{5}$$

$$Var(X_d) = \frac{\alpha_d(\alpha_d + \alpha_{D+1} - 1)}{(\alpha_{D+1} - 1)^2(\alpha_{D+1} - 2)} \tag{6}$$

2.3.2 ID Mixture Model Learning

In this section, we present MLE for IDMM followed by EM algorithm.

Maximum Likelihood Estimation (MLE)

The pixels class labels are considered as latent variables. Each pixel $\vec{X}_n$ and its peer pixel $\vec{\hat{X}}_n$ are the observed data. The membership vector is defined as $Z = \{\vec{Z}_1, \ldots, \vec{Z}_N\}$. where $\vec{Z}_n = (Z_{n1}, ..., Z_{nM})$. If $\vec{X}$ and $\vec{\hat{X}}$ are in same cluster c, then $Z_{nc} = 1$ and $Z_{nl} = 0$ where $l = \{1, \ldots, M\} - \{c\}$, otherwise $Z_{nc} = 0$. The complete likelihood is denoted as:

$$p(X, \hat{\mathcal{X}}, Z|\Theta) = \prod_{n=1}^{N} \prod_{\jmath=1}^{M} \left[\pi_\jmath p(\vec{X}_n|\vec{\alpha}_\jmath)\pi_\jmath p(\vec{\hat{X}}_n|\vec{\alpha}_\jmath)\right]^{Z_{n\jmath}} \tag{7}$$

The complete log-likelihood is as follows:

$$L(\Theta, Z, X, \hat{X}) = \sum_{n=1}^{N} \sum_{\jmath=1}^{M} Z_{n\jmath} \left(2 \log \pi_\jmath + \log p(\vec{X}_n|\vec{\alpha}_\jmath) + \log p(\vec{\hat{X}}_n|\vec{\alpha}_\jmath) \right) \tag{8}$$

EM algorithm consists of two phase: E-Step and M-Step [9]. In E-Step, the conditional expectation of the maximum log-likelihood is estimated as follows:

$$E[L(\Theta, Z, X, \hat{X})] = Q(\Theta, Z, X, \hat{X}) = \sum_{n=1}^{N} \sum_{\jmath=1}^{M} p(\jmath|\vec{X}_n, \vec{\hat{X}}_n, \vec{\alpha_\jmath}) \times \left(2 \log \pi_\jmath + \log p(\vec{X}_n|\vec{\alpha}_\jmath) + \log p(\vec{\hat{X}}_n|\vec{\alpha_\jmath}) \right) \tag{9}$$

where $p(\jmath|\vec{X}_n, \vec{\hat{X}}_n, \vec{\alpha}_\jmath) = \hat{Z}_{n\jmath}$ is a posterior probablity which signifies that both pixels $\vec{X}_n$ and its neighbor $\vec{\hat{X}}_n$ belong to the same cluster $\jmath$.

$$\hat{Z}_{n\jmath} = \frac{\pi_\jmath p(\vec{X}_n|\vec{\alpha}_\jmath)\pi_\jmath p(\vec{\hat{X}}_n|\vec{\alpha}_\jmath)}{\sum_{m'=1}^{M} \pi_{m'} p(\vec{X}_n|\vec{\alpha}_{m'})\pi_{m'} p(\vec{\hat{X}}_n|\vec{\alpha}_{m'})} \tag{10}$$

In M-Step, we have to maximize the complete log-likelihood and thus have to solve:

$$\frac{\partial Q(\Theta, Z, X, \hat{X})}{\partial \theta_\jmath} = 0 \tag{11}$$

The partial derivative of $Q(\Theta, Z, X, \hat{X})$ with respect to $\alpha_{\jmath d}$ where $\jmath = 1, \ldots, M$ and $d = 1, \ldots, D$ is as follows:

$$\frac{\partial Q(\Theta, Z, X, \hat{X})}{\partial \alpha_{\jmath d}} = \sum_{n=1}^{N} \sum_{j=1}^{M} \hat{Z}_{nj} \left\{ 2\big(\Psi|\vec{\alpha}_{\jmath}| - \Psi|\alpha_{\jmath d}|\big) + \log\left(\frac{X_{nd}}{1+|\vec{X}_n|}\right) + \log\left(\frac{\hat{X}_{nd}}{1+|\vec{\hat{X}}_n|}\right) \right\} \quad (12)$$

The partial derivative of $Q(\Theta, Z, X, \hat{X})$ with respect to $\alpha_{\jmath D+1}$ is as follows:

$$\frac{\partial Q(\Theta, Z, X, \hat{X})}{\partial \alpha_{jD+1}} = \sum_{n=1}^{N} \sum_{j=1}^{M} \hat{Z}_{nj} \left\{ 2\Big(\Psi|\alpha_j| - \Psi|\alpha_{jD+1}|\Big) + \log\left(\frac{1}{1+|\vec{X}_n|}\right) + \log\left(\frac{1}{1+|\vec{\hat{X}}_n|}\right) \right\} \quad (13)$$

Considering Eq. (12) and Eq. (13), it can be observed that no closed solution exists for $\alpha_{\jmath}$. Therefore, we have used Newton-Raphson method as follows:

$$\vec{\alpha}_{\jmath}^{(k+1)} = \vec{\alpha}_{\jmath}^{\,(k)} - \vec{G}_{\jmath} H_{\jmath}^{-1} \quad (14)$$

where $\vec{\alpha}_{\jmath}^{(k+1)}$ is the updated hyper-parameter, $\vec{\alpha}_{\jmath}^{\,(k)}$ is the old hyper-parameter, $H_{\jmath}^{-1}$ is the inverse of Hessian matrix followed by $\vec{G}_{\jmath}$, which is a gradient. The Gradient is the first partial order derivative of $Q(\Theta, Z, X, \hat{X})$ and is described as follows:

$$\vec{G}_{\jmath} = \left(\frac{\partial Q(\Theta, Z, X, \hat{X})}{\partial \alpha_{\jmath 1}}, \ldots, \frac{\partial Q(\Theta, Z, X, \hat{X})}{\partial \alpha_{\jmath D+1}} \right) \quad (15)$$

For computing $H_{\jmath}^{-1}$, the approach given in [2] is used. And for the mixing parameter $\pi_{\jmath}$, there exists a closed-form solution:

$$\pi_{\jmath} = \sum_{n=1}^{N} \frac{\hat{Z}_{n\jmath}}{N} \quad (16)$$

2.3.3 The Probability Density Function of GID

If $\vec{X}$ is a positive vector which consists of D dimensions and following a GID, then its joint density function is given by:

$$p(\vec{X}|\vec{\alpha}_j, \vec{\beta}_j) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_{jd} + \beta_{jd})}{\Gamma(\alpha_{jd})\Gamma(\beta_{jd})} T_{nd}^{\alpha_{jd}-1} \left(1 + |\vec{X}|\right)^{-\Omega_{jd}} \quad (17)$$

where $\vec{\alpha_j} = [\alpha_{j1}, \ldots, \alpha_{jD}]$, $\vec{\beta_j} = [\beta_{j1}, \ldots, \beta_{jD}]$. $|\vec{X}| = \sum_{d=1}^{D} X_d$. We define Ω such that $\Omega_{jd} = \alpha_{jd} + \beta_{jd} - \beta_{jd+1}$ for $d = 0, \ldots, D$ with $\beta_{jD+1} = 0$. The GID possesses a property which makes its estimation simple. If there exists a vector $\vec{X}$ that follows GID, then we can create another vector $\vec{W}_n = [W_{n1}, \ldots, W_{nD}]$ where elements follow inverted Beta (IB) distributions via the following transformation:

$$W_{nd} = f(X_{nd}) = \begin{cases} X_{nd} & d = 1 \\ \frac{X_{nd}}{1 - X_{n1} - , \ldots, - X_{nd-1}} & d = 2, \ldots, D \end{cases} \tag{18}$$

The pdf of IB is defined as:

$$p_{iBeta}(W_{nd}|\alpha_{jd}, \beta_{jd}) = \frac{\Gamma(\alpha_{jd} + \beta_{jd})}{\Gamma(\alpha_{jd})\Gamma(\beta_{jd})} W_{nd}^{\alpha_{jd}-1}(1 + W_{jd})^{-(\alpha_{jd}+\beta_{jd})} \tag{19}$$

The mean of inverted Beta (IB) is given by:

$$E(W_d) = \frac{\alpha_d}{\beta_d - 1} \tag{20}$$

The variance of IB is as follows:

$$Var(W_d) = \frac{\alpha_d(\alpha_d + \beta_d - 1)}{(\beta_d - 2)(\beta_d - 1)^2} \tag{21}$$

2.3.4 GID Model Learning

Maximum Likelihood Estimation

The new form of Eq. (8) for GID is as follows :

$$L(\Theta, Z, \mathcal{W}, \hat{\mathcal{W}}) = \sum_{n=1}^{N} \sum_{\jmath=1}^{M} Z_{n\jmath} \Big(2 \log \pi_\jmath + \sum_{d=1}^{D} \log p_{nB}(\vec{W}_{nd}|\vec{\theta}_{\jmath d}) \quad + \sum_{d=1}^{D} \log p_{nB}(\vec{\hat{W}}_{nd}|\vec{\theta}_{\jmath d}) \Big) \tag{22}$$

In E-Step, the conditional expectation of the maximum log-likelihood is estimated as follows:

$$Q(\Theta, Z, \mathcal{W}, \hat{\mathcal{W}}, \Lambda) = \sum_{n=1}^{N} \sum_{\jmath=1}^{M} \hat{Z}_{n\jmath} \Big\{ 2 \log \pi_\jmath + \sum_{d=1}^{D} \log p_{nB}(\vec{W}_{nd}|\vec{\alpha}_\jmath, \vec{\beta}_\jmath) + \sum_{d=1}^{D} \log p_{nB}(\vec{\hat{W}}_{nd}|\vec{\alpha}_\jmath, \vec{\beta}_\jmath \Big\} + \Lambda(1 - \sum_{m=1}^{M} \pi_\jmath) \tag{23}$$

where $\hat{Z}_{nj}$ signifies that both $\vec{W}_n$ and $\vec{\hat{W}}_n$ belong to the same cluster $\jmath$ and Λ is the Langrange multiplier.

$$\hat{Z}_{nj} = \frac{{\pi_\jmath}^2 \prod_{d=1}^{D} p_{nB}(\vec{W}_{nd}|\vec{\alpha}_{\jmath d}, \vec{\beta}_{\jmath d}) \prod_{d=1}^{D} p_{nB}(\vec{\hat{W}}_{nd}|\vec{\alpha}_{\jmath d}, \vec{\beta}_{\jmath d})}{\sum_{\jmath'=1}^{M} {\pi'_\jmath}^2 \prod_{d=1}^{D} p_{nB}(\vec{W}_{nd}|\vec{\alpha}_{\jmath' d}, \vec{\beta}_{\jmath' d}) \prod_{d=1}^{D} p_{nB}(\vec{\hat{W}}_{nd}|\vec{\alpha}_{\jmath' d}, \vec{\beta}_{\jmath' d})} \tag{24}$$

In M-Step, we have to maximize the complete log-likelihood and thus have to estimate:

$$\frac{\partial Q(\Theta, Z, \mathcal{W}, \hat{\mathcal{W}}, \Lambda)}{\partial \Theta} = 0 \tag{25}$$

The partial derivatives of $Q(\Theta, Z, \mathcal{W}, \hat{\mathcal{W}}, \Lambda)$ with respect to $\alpha_{\jmath d}$ and $\beta_{\jmath d}$ are as follows:

$$\frac{\partial Q(\Theta, Z, \mathcal{W}, \hat{\mathcal{W}}, \Lambda)}{\partial \alpha_{\jmath d}} = \sum_{i=1}^{N} \hat{Z}_{n\jmath} \Big\{ 2\Big(\Psi(\alpha_{\jmath d} + \beta_{\jmath d}) - \Psi(\alpha_{\jmath d})\Big) + \log\left(\frac{W_{nd}}{1 + W_{nd}}\right) + \log\left(\frac{\hat{W}_{nd}}{1 + \hat{W}_{nd}}\right) \Big\} \tag{26}$$

$$\frac{\partial Q(\Theta, Z, \mathcal{W}, \hat{\mathcal{W}}, \Lambda)}{\partial \beta_{\jmath d}} = \sum_{i=1}^{N} \hat{Z}_{n\jmath} \Big\{ 2\Big(\Psi(\alpha_{\jmath d} + \beta_{\jmath d}) - \Psi(\beta_{\jmath d})\Big) + \log\left(\frac{W_{nd}}{1 + W_{nd}}\right) + \log\left(\frac{\hat{W}_{nd}}{1 + \hat{W}_{nd}}\right) \Big\} \tag{27}$$

where $\Psi(.)$ is the digamma function. From (26) and (27), it can be observed that no closed-form solution exists for $\theta_{\jmath d}$. Therefore, we have to use Newton-Raphson method as follows:

$$\vec{\Theta}_{\jmath d}^{(k+1)} = \vec{\Theta}_{\jmath d}^{(k)} - H_{\jmath d}^{-1} G_{\jmath d} \tag{28}$$

where $H_{\jmath d}$ and $G_{\jmath d}$ are Hessian matrix and gradient, respectively. To compute $H_{\jmath d}^{-1}$, the inverse matrix theorem by [5] can be used.

2.4 Spatial information integration with IDMM and GIDMM using MRF Approach.

2.4.1 ID Model Learning

Segmentation Approach

The joint conditional probability is as follows:

$$p(\mathcal{X}, \mathcal{Z}|\theta) = \prod_{n=1}^{N} p(\vec{X}_n|\vec{\theta}_\jmath)^{Z_{n\jmath}} = \prod_{n=1}^{N} \sum_{\jmath=1}^{M} \Big(\pi_\jmath p(\vec{X}_n|\vec{\alpha}_\jmath)\Big)^{Z_{n\jmath}} \tag{29}$$

where $Z = \{\vec{Z}_1, ..., \vec{Z}_N\}$ is the membership vector and each $\vec{Z}_n = (Z_{n1}, \ldots, Z_{nM})$. If a pixel $\vec{X}_b$ belongs to cluster c where $b \in N$ and $c \in M$, then $Z_{bc} = 1$ and the rest of elements in $\vec{Z}_b$ will be equal to 0. According to the Bayes rules, the posterior probablity is given by:

$$p(\theta|\mathcal{X}, \mathcal{Z}) \propto p(\mathcal{X}, \mathcal{Z}|\theta)\mathcal{W}(\Pi) \tag{30}$$

Maximum Likelihood Estimation.

The complete log-likelihood is as follows:

$$\begin{aligned} L(\theta|\mathcal{X}, \mathcal{Z}) &= \log\Big(\prod_{n=1}^{N}\sum_{\jmath=1}^{M}\Big(\pi_\jmath p(\vec{X}_n|\vec{\alpha}_\jmath)\Big)^{Z_{n\jmath}} W(\Pi)\Big) \\ &= \sum_{i=1}^{N} \log\Big\{\sum_{j=1}^{M}\Big(\pi_\jmath p(\vec{X}_n|\vec{\alpha}_\jmath)\Big)^{Z_{n\jmath}}\Big\} - \log\mathcal{N} - \frac{1}{\mathcal{T}}\mathcal{U}(\Pi) \end{aligned} \tag{31}$$

In order to solve the issues with smoothing prior which are discussed in the introduction, a novel factor [10] is used which is given as:

$$G_{n\jmath} = \exp\Big[\frac{\beta}{2N_n}\sum_{m\in\delta_n}(Z_{m\jmath}^{(t)} + \pi_{m\jmath}^{(t)})\Big] \tag{32}$$

where β is the temperature value that is responsible for controlling smoothing prior and $Z_{m\jmath}$ is the posterior probablity. The authors in [10] have suggested, the ideal values for $\beta = 12$ and $N_n = 25$. The smoothing parameter is only dependent on $Z_{m\jmath}$ and $\pi_{m\jmath}$ of the previous step in the EM algorithm. Thus, it also acts as a filter to

ensure the smoothness and restore image from corruption. The authors in [10] have proposed the following smoothing:

$$\mathcal{U}(\Pi) = -\sum_{n=1}^{N}\sum_{j=1}^{M} G_{nj}^{(t)} \log \pi_{nj}^{(t+1)} \tag{33}$$

By maximizing the complete log-likelihood, we get $Q(\mathcal{Z}, \mathcal{X}, \Theta)$ which is as follows:

$$\begin{aligned} Q(\mathcal{Z}, \mathcal{X}, \Theta) = \sum_{n=1}^{N}\sum_{j=1}^{M} z_{nj}\Big\{ \log \pi_j + \log p(\vec{X}_n|\vec{\alpha}_j)\Big\} \quad - \log \mathcal{N} \\ + \frac{1}{T}\sum_{n=1}^{N}\sum_{j=1}^{M} G_{nj}^{(t)} \log {\pi_{nj}}^{(t+1)} \end{aligned} \tag{34}$$

The conditional expectation values $\hat{Z}_{nj}$ of hidden variables is as follows:

$$\hat{Z}_{nj} = \frac{\pi_{nj}^{(t)} p(\vec{X}_n|\vec{\alpha}_j)}{\sum_{k=1}^{K} \pi_{nk}^{(t)} p(\vec{X}_n|\vec{\alpha}_k)} \tag{35}$$

For computing the updated value of $\vec{\alpha}$, the approach given in [2] can be used. There exist a closed-form solution for parameter π_j, and is given as:

$$\pi_{nj}^{(k+1)} = \frac{z_{nj}^{(t)} + G_{nj}^{(t)}}{\sum_{s=1}^{M} z_{ns}^{(t)} + G_{ns}^{(t)}} \tag{36}$$

2.4.2 GID Model Learning

Maximum Likelihood Estimation

The complete log-likelihood is as follows:

$$L(p(\mathcal{W}, \mathcal{Z}|\Theta)) = \log\left(\prod_{n=1}^{N}\sum_{j=1}^{M} \pi_j^{Z_{nj}} \prod_{d=1}^{D} p_B(W_{nd}|\alpha_{jd}, \beta_{jd})^{Z_{nj}} \mathcal{W}(\Pi)\right) \tag{37}$$

$$L(p(\mathcal{W}, \mathcal{Z}|\Theta)) = \sum_{n=1}^{N} \log\left(\sum_{j=1}^{M} \pi_j^{Z_{nj}} \prod_{d=1}^{D} p_B(W_{nd}|\alpha_{jd}, \beta_{jd})^{Z_{nj}}\right) - \log \mathcal{N} - \frac{1}{T}\mathcal{U}(\Pi) \tag{38}$$

By maximizing the complete log-likelihood, we get $Q(\Theta, \mathcal{W}, \mathcal{Z})$ which is as follows:

$$Q(\Theta, \mathcal{W}, \mathcal{Z}) = \sum_{n=1}^{N}\sum_{j=1}^{M} z_{nj}^{(t)}\Bigg\{ \log \pi_j^{(t+1)} + \sum_{d=1}^{D} p_B(W_{nd}|\alpha_{jd}, \beta_{jd})\Bigg\} - \log \mathcal{N} + \frac{1}{T}\sum_{n=1}^{N}\sum_{n=1}^{M} G_{nj}^{(t)} \log \pi_j^{(t+1)} \tag{39}$$

The hidden variable is given by:

$$\hat{Z}_{nj} = \frac{\pi_j p(\vec{W}_n|\alpha_j, \beta_j)}{\sum_{j=1}^{M} p(\vec{W}_n|\alpha_j, \beta_j)} = \frac{\pi_j \prod_{d=1}^{D} p_B(\vec{W}_{nd}|\alpha_{jd}, \beta_{jd})}{\sum_{m=1}^{M} \pi_m \prod_{d=1}^{D} p_B(\vec{W}_{nd}|\alpha_{jd}, \beta_{jd})} \tag{40}$$

For the calculation of updated $\vec{\alpha}$ and $\vec{\beta}$, we have used the approach given in [1].

2.5 Estimation Algorithm

The algorithm is as follows:

1. INPUT: An image $\mathcal{X}$ and M.
2. Apply K-means clustering algorithm to group pixels into M clusters.
3. Initialize π.
4. Apply Method of Moments to calculate the initial values of parameters.
5. E-Step:
   - Compute $\hat{Z}_{nj}$ using (10), (24), (35) and (40) for IDMM1, GIDMM1, IDMM2 and GIDMM2 respectively.
6. M-Step:
   - Calculate $\vec{\alpha}$ using (14) for IDMM1 and IDMM2 respectively.
   - Calculate $\vec{\Theta}$ using (28) for GIDMM1 and GIDMM2 respectively.
   - For mixing parameter π, use (16) for IDMM1 and GIDMM1, use (36) for IDMM2 and GIDMM2.
7. Iterate through E-Step and M-Step until convergence.

where IDMM1 and IDMM2 are the two presented IDMM-based methods. GIDMM1 and GIDMM2 are the two presented GIDMM-based methods.

2.6 Experimental Results

In this section, we validate the performance of the proposed algorithms and compare them with Gaussian mixture models proposed in [16, 10]. It is worth to note that the selection of color-space is important for color images. While performing image segmentation, it is desirable to have a color-space robust against varying illumination. We have selected the rgb and $l_1l_2l_3$ color-spaces which are as follows:

2.6.1 Color Spaces for Image Segmentation

The selection of color-space is crucial for image segmentation. It is desirable to have a color-space robust against varying illumination. Few color-spaces are assessed, dissected and examined in [70]. Out of many color spaces, we have selected the rgb and $l_1l_2l_3$ color spaces which are as follows:

$$r(\mathcal{R},\mathcal{G},\mathcal{B}) = \frac{\mathcal{R}}{\mathcal{R}+\mathcal{G}+\mathcal{B}} \tag{41}$$

$$g(\mathcal{R},\mathcal{G},\mathcal{B}) = \frac{\mathcal{G}}{\mathcal{R}+\mathcal{G}+\mathcal{B}} \tag{42}$$

$$b(\mathcal{R},\mathcal{G},\mathcal{B}) = \frac{\mathcal{B}}{\mathcal{R}+\mathcal{G}+\mathcal{B}} \tag{43}$$

$$l_1(\mathcal{R},\mathcal{G},\mathcal{B}) = \frac{(\mathcal{R}-\mathcal{G})^2}{SUM(RGB)} \tag{44}$$

$$l_2(\mathcal{R},\mathcal{G},\mathcal{B}) = \frac{(\mathcal{R}-\mathcal{B})^2}{SUM(RGB)} \tag{45}$$

$$l_3(\mathcal{R},\mathcal{G},\mathcal{B}) = \frac{(\mathcal{G}-\mathcal{B})^2}{SUM(RGB)} \tag{46}$$

where $SUM(RGB) = (\mathcal{R}-\mathcal{G})^2 + (\mathcal{R}-\mathcal{B})^2 + (\mathcal{G}-\mathcal{B})^2$. The $l_1l_2l_3$ and rgb outperform the traditional $\mathcal{RGB}$ color-space and hence used in our experiments.

2.6.2 Metrics for segmentation performance evaluation

In order to compare the performances of the proposed algorithms, we have used eight image segmentation evaluation metrics.

Adjusted Rand Index (ARI)

It is defined as the level of similarity among the actual labels and predicted labels. In ARI, the permutations are not considered. The ARI's value will tend to 0, if the predicted labels are arranged randomly. ARI has a range of [-1, 1] and values closer to zero are considered as bad clustering and values closer to 1 means good clustering. The ARI [54] is given as :

$$ARI = \frac{RI - E[RI]}{max(RI) - E[RI]} \tag{47}$$

where $E[RI]$ is Expected value of RI (Rand index). The RI is defined as :

$$ARI = \frac{a+b}{C_2^{n_{samples}}} \tag{48}$$

where K and C are the actual and predicted labels respectively. The a is the number of element pairs having same class labels in K and C. b is the number of element pairs having different class labels in K and C.

Adjusted Mutual Information Score (AMIS)

The mutual information (MI) is defined as the level of agreement of actual labels and predicted labels, without permutation. The AMIS [55, 56, 57] is the adjusted version of MI and is defined as:

$$AMIS = \frac{MI(T,U) - E[MI(T,U)]}{mean(H(T), H(U)) - E[MI(T,U)]} \tag{49}$$

where T and U are two class labels assignments, $H(T)$ and $H(U)$ defines the entropy for T and U, respectively. $E[MI(T,U)]$ is the expected value of $MI(T,U)$.

Normalized Mutual Information Score (NMIS)

The NMIS [55, 56, 57] is the normalized version of MI and given as:

$$NMIS(T,U) = \frac{MI(T,U)}{mean\big(H(T), H(U)\big)} \tag{50}$$

Homogeneity Score (HS)

HS [58, 59] uses a criteria related to analysis of the conditional entropy. HS is defined as each cluster contains only members of the single class. It has a range of [0, 1], where 1 means each cluster only contains members of just one class. On the other

hand, 0 means that almost every data inside a cluster contains different class labels. It is given by:

$$HS = 1 - \frac{H(C|K)}{H(C)} \tag{51}$$

where $H(C|K)$ is the conditional entropy of the classes given the cluster assignments:

$$H(C|K) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|}\frac{n_{ck}}{n}\log\left(\frac{n_{ck}}{n_k}\right); \qquad H(C) = -\sum_{c=1}^{|C|}\frac{n_c}{n}.\log\left(\frac{n_c}{n}\right) \tag{52}$$

where $H(C)$ is the entropy of the classes, n is the number of pixels in the image; n_c is the number of pixels that belong to class c; n_k is the number of pixels that belong to cluster k.

Completeness Score (CS)

CS [58, 59] is also a criteria related to analysis of the conditional entropy. CS is defined as all the members of a given class that belong to the same clusters. It has a range of [0, 1], where 0 means worst clustering and 1 means perfect clustering.

$$CS = 1 - \frac{H(K|C)}{H(K)} \tag{53}$$

where $H(K|C)$ and $H(K)$ can be computed in a symmetric manner.

V-Measure Score (VMS)

The VMS [58, 59] is defined as the harmonic mean of HS and CS. It is symmetrical in nature.

$$VMS = 2 \times \left(\frac{HS \times CS}{HS + CS}\right) \tag{54}$$

Calinski-Harabaz Index (CHI)

It is one of the most flexible metrics for image segmentation. If the human segmentations are not available, then CHI can be used for model evaluation, where higher value signifies that the clusters are well defined. For M clusters, the CHI [60] is defined as the ratio of the between-clusters dispersion mean and the within-cluster dispersion:

$$CHI(M) = \frac{T_r(B_M)}{T_r(W_M)} \times \frac{n - M}{M - 1} \tag{55}$$

where W_M is the within-cluster dispersion matrix and B_M is defined as the between group dispersion matrix:

$$W_M = \sum_{m=1}^{M} \sum_{x \in C_m} (x - C_m)(x - C_m)^T; \quad B_M = \sum_{m} n_m (c_m - c)(c_m - c)^T \tag{56}$$

C_m: set of pixels in cluster m, c_m: the centre of the cluster m, n_m: number of pixels in cluster m.

Jaccard Similarity Score (JSS)

JSS [61, 62, 63] is also called Jaccard index. It is defined as the ratio of intersection (of predicted labels and actual labels) and the union (of predicted labels and actual labels). It also ranges between 0 and 1 where 0 means very bad score and 1 means the segmentation output is perfect.

2.6.3 Results

For experimentation, we have used two images 29030 and 5096 from BSD500. The rgb and $l_1l_2l_3$ color spaces are used with 29030 and 5096 images respectively. Table 1 and Table 2 contains the qualitative analysis results for 29030 followed by the segmentation output shown in Fig. 1. Specifically, Table 1 and Table 2 depict the results of IDMM, GIDMM and GMM using the first and second methods respectively. Similarly, Table 3 and Table 4 represent the qualitative results for 5096. Table 3 illustrates the results of the proposed models using the first method, followed by Table 4, which contains the results of proposed models using the second method. Fig. 2 contains the segmentation output for image 5096. All the tables show that the proposed algorithms outperformed the GMM proposed in [16, 10].

Table 1: Performance evaluation of the 29030 image using mixture models deployed in first method.

Alg.	Metrics (K=3)							
	Seg.	**ARI**	**AMIS**	**NMIS**	**HS**	**CS**	**VM**	**CHI**
ID	1	0.587	0.371	0.469	0.371	0.593	0.456	**232945.025**
	2	0.550	0.301	0.423	0.301	0.593	0.399	
	3	0.573	0.356	0.446	0.356	0.558	0.434	
	4	0.549	0.284	0.417	0.284	0.612	0.388	
	5	0.447	0.235	0.370	0.235	0.582	0.335	
	6	0.549	0.283	0.411	0.283	0.596	0.384	
Mean		**0.543**	**0.305**	**0.440**	**0.305**	**0.589**	**0.400**	
GID	1	0.706	0.483	0.609	0.483	0.769	0.593	**268627.231**
	2	0.660	0.394	0.552	0.394	0.774	0.522	
	3	0.698	0.464	0.581	0.465	0.727	0.567	
	4	0.659	0.374	0.548	0.374	0.803	0.510	
	5	0.539	0.303	0.476	0.303	0.747	0.431	
	6	0.706	0.366	0.531	0.366	0.768	0.496	
Mean		**0.661**	**0.398**	**0.570**	**0.398**	**0.765**	**0.520**	
GMM	1	0.506	0.342	0.416	0.342	0.507	0.408	**72422.334**
	2	0.464	0.284	0.383	0.284	0.518	0.367	
	3	0.461	0.327	0.395	0.327	0.477	0.388	
	4	0.457	0.275	0.388	0.275	0.549	0.366	
	5	0.369	0.227	0.344	0.227	0.521	0.316	
	6	0.453	0.270	0.377	0.270	0.527	0.357	
Mean		**0.452**	**0.287**	**0.384**	**0.287**	**0.517**	**0.367**	

Table 2: Performance evaluation of the 29030 image using mixture models deployed in second method.

Alg.	Metrics (K=3)							
	Seg.	**ARI**	**AMIS**	**NMIS**	**HS**	**CS**	**VM**	**CHI**
ID	1	0.609	0.396	0.489	0.396	0.604	0.478	**171334.822**
	2	0.570	0.333	0.457	0.333	0.626	0.435	
	3	0.577	0.379	0.464	0.379	0.569	0.455	
	4	0.564	0.319	0.457	0.319	0.655	0.429	
	5	0.473	0.273	0.419	0.273	0.644	0.383	
	6	0.567	0.318	0.451	0.318	0.639	0.425	
Mean		**0.560**	**0.336**	**0.456**	**0.336**	**0.623**	**0.434**	
GID	1	0.676	0.441	0.553	0.441	0.693	0.539	**216480.594**
	2	0.640	0.373	0.519	0.373	0.721	0.492	
	3	0.663	0.426	0.529	0.426	0.657	0.517	
	4	0.633	0.347	0.505	0.347	0.734	0.471	
	5	0.529	0.294	0.458	0.294	0.714	0.417	
	6	0.637	0.350	0.503	0.350	0.723	0.472	
Mean		**0.630**	**0.372**	**0.511**	**0.372**	**0.707**	**0.485**	
GMM	1	0.264	0.192	0.233	0.192	0.284	0.229	**36243.646**
	2	0.245	0.187	0.252	0.187	0.340	0.241	
	3	0.272	0.187	0.240	0.199	0.289	0.235	
	4	0.245	0.199	0.273	0.194	0.386	0.258	
	5	0.266	0.180	0.272	0.180	0.412	0.250	
	6	0.248	0.191	0.266	0.191	0.372	0.252	
Mean		**0.257**	**0.189**	**0.256**	**0.190**	**0.347**	**0.244**	

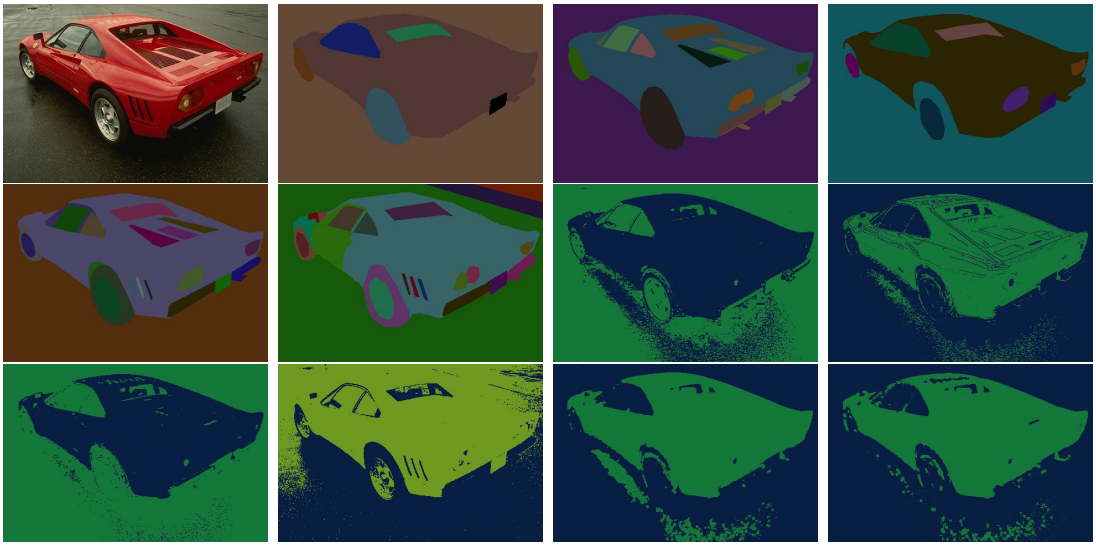

Figure 1: Original image (29030) followed by five ground truth segments from BSD500. Images from seventh to nine are the outputs of IDMM, GIDMM, GMM deployed in method one respectively. Similarly, tenth to eleventh images are the outputs of IDMM, GIDMM, GMM deployed in method two respectively.

Table 3: Performance evaluation of the 5096 image using mixture models deployed in method one.

Alg.	Metrics (K=3)							
	Seg.	**ARI**	**AMIS**	**NMIS**	**HS**	**CS**	**VM**	**CHI**
ID	1	0.232	0.233	0.335	0.233	0.483	0.314	**108713.679**
	2	0.199	0.213	0.321	0.213	0.483	0.296	
	3	0.238	0.240	0.340	0.240	0.482	0.320	
	4	0.216	0.214	0.321	0.214	0.480	0.296	
	5	0.252	0.233	0.321	0.233	0.442	0.305	
Mean		**0.227**	**0.226**	**0.328**	**0.226**	**0.474**	**0.306**	
GID	1	0.269	0.283	0.408	0.283	0.588	0.382	**202214.492**
	2	0.233	0.256	0.386	0.256	0.581	0.355	
	3	0.287	0.289	0.410	0.289	0.582	0.386	
	4	0.262	0.271	0.406	0.271	0.608	0.375	
	5	0.271	0.273	0.377	0.273	0.521	0.359	
Mean		**0.265**	**0.274**	**0.397**	**0.274**	**0.576**	**0.371**	
GMM	1	0.000	0.002	0.017	0.002	0.150	0.004	**636.274**
	2	0.000	0.002	0.018	0.002	0.163	0.004	
	3	0.000	0.002	0.013	0.002	0.113	0.003	
	4	0.000	0.002	0.017	0.002	0.152	0.004	
	5	0.001	0.002	0.018	0.002	0.151	0.004	
Mean		**0.000**	**0.002**	**0.017**	**0.002**	**0.146**	**0.004**	

Table 4: Performance evaluation of the 5096 image using mixture models deployed in method two.

Alg.	Metrics (K=3)							
	Seg.	**ARI**	**AMIS**	**NMIS**	**HS**	**CS**	**VM**	**CHI**
ID	1	0.287	0.300	0.432	0.300	0.622	0.405	**115633.717**
	2	0.268	0.276	0.415	0.276	0.625	0.383	
	3	0.287	0.309	0.439	0.309	0.623	0.413	
	4	0.271	0.285	0.427	0.285	0.639	0.394	
	5	0.290	0.297	0.410	0.297	0.565	0.390	
Mean		**0.280**	**0.293**	**0.421**	**0.294**	**0.615**	**0.397**	
GID	1	0.300	0.305	0.440	0.305	0.633	0.412	**239455.458**
	2	0.268	0.280	0.421	0.280	0.633	0.388	
	3	0.306	0.312	0.443	0.312	0.628	0.417	
	4	0.285	0.289	0.433	0.289	0.648	0.400	
	5	0.317	0.310	0.427	0.310	0.589	0.406	
Mean		**0.295**	**0.299**	**0.433**	**0.299**	**0.626**	**0.405**	
GMM	1	0.274	0.284	0.413	0.284	0.599	0.386	**108849.1252**
	2	0.238	0.265	0.402	0.265	0.609	0.369	
	3	0.262	0.298	0.425	0.298	0.608	0.400	
	4	0.284	0.285	0.429	0.285	0.648	0.396	
	5	0.310	0.301	0.418	0.301	0.581	0.396	
Mean		**0.273**	**0.286**	**0.418**	**0.287**	**0.609**	**0.389**	

Figure 2: Original image (5096) followed by five ground truth segments from BSD500. Images from seventh to nine are the outputs of IDMM, GIDMM, GMM deployed in method one respectively. Similarly, tenth to eleventh images are the outputs of IDMM, GIDMM, GMM from method two respectively.

Chapter 3

Color Image Segmentation using Semi-Bounded Finite Mixture Models by Incorporating Mean Templates

3.1 Introduction

In this chapter, we have applied mean templates for CP (conditional probability) and PP (prior probability), proposed in [51], in order to integrate spatial information with FMM. Furthermore, from the mean template, we can obtain four methods that are geometric conditional geometric prior (GCGP), geometric conditional arithmetic prior (GCAP), arithmetic conditional geometric prior (ACGP), and arithmetic conditional arithmetic prior (ACAP).

The remaining chapter is organized as follows. In section 3.2, the challenges faced during image segmentation using FMM are discussed along with their solutions. Section 3.3 is devoted to the weighted mean templates where the geometric and arithmetic CP along with geometric and arithmetic PP are explained in detail. In section 3.4, the integration of four methods (GCGP, GCAP, ACGP and, GCGP) with IDMM, GIDMM and IBLMM are presented followed by their algorithms. Section 3.5 contains the experimental results in which eight segmentation evaluation metrics, the results in the form of tables for BSD500 dataset and segmentation outputs in the

form of figures for both BSD500 [71] and CVCL [72] datasets are discussed.

3.2 Problem Description

Consider the binary image given in Fig. 1 [51]. The upper-most part of the image is white in color and has intensity value equal to 1. The lower part of the image is black in color, having intensity value equal to 0. This image is distorted by noise. Two 3 by 3 windows are extracted from upper and lower parts of the image, as shown in Fig. 1 (b) and (c). For binary image, let the pixels having intensity value equal to 1 and 0 be assigned to classes $\mathcal{U}$ and $\mathcal{V}$, respectively. It can be easily observed that middle pixels of both the windows are corrupted by noise and may result in mis-classification.

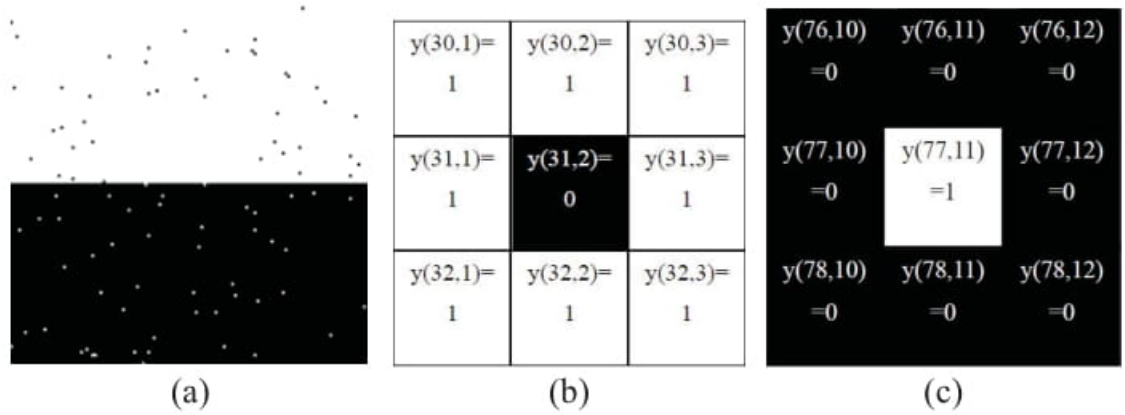

Figure 3: (a) Original image. (b) White window. (c) Black window. The numbers in parentheses are the coordinates of the image; 0 and 1 are the binary image intensity values.

A possible solution for this problem has two requirements: First, the spatial information of each pixel should be incorporated to prior probability π_j, therefore π_j should be changed to π_{ij}. The π_{ij} of the middle pixels for both the windows should be affected by the prior probability π_{st} where $s = i \pm 1$ and $t = j \pm 1$. Each pixel inside the window should have same prior probability which can be calculated using mean prior probability (discussed in later section).

Second, given the component j and intensity value y, the FMM satisfies the same conditional probability $p(y_n|\theta_j)$. Sometimes, this is true but not always. The y of middle pixel in Fig. 1(b) is same as the pixels around the central pixel in Fig. 1(c).

These two types of pixels should belong to different clusters. Thus, the traditional FMM is not capable enough to differentiate among these types of pixels. In order to counter this issue, the authors of [51] have suggested the mean template for CP.

The authors of [51] have calculated the windows CP values (CPV) for Fig. 1(b) and Fig. 1(c) using traditional FMM, GMT and, AMT as illustrated in Fig. 2 [51] and Fig. 3 [51]. In Fig. 2(a), the middle pixels have same CPV ($0.606/\sqrt{2\pi}$) as the pixels around the central pixel in Fig 3(a). Similarly, the central pixel of Fig. 3(a) and surrounding pixels of middle pixel of Fig. 2(a) have the same CPV ($1/\sqrt{2\pi}$). We can observe that the proposed mean templates have removed the effect of noise from the windows shown in Fig. 2(a) and 3(a). It is noteworthy that the model suggested by the authors in [51] is robust to noise.

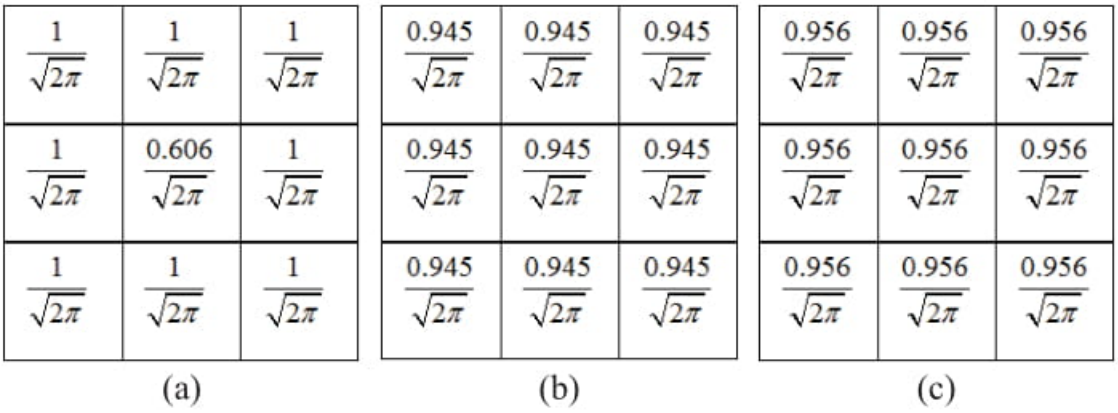

Figure 4: CPV of Fig. 1(b), (a) CPV with traditional FMM. (b) CPV calculated by geometric template. (c) CPV calculated by arithmetic template.

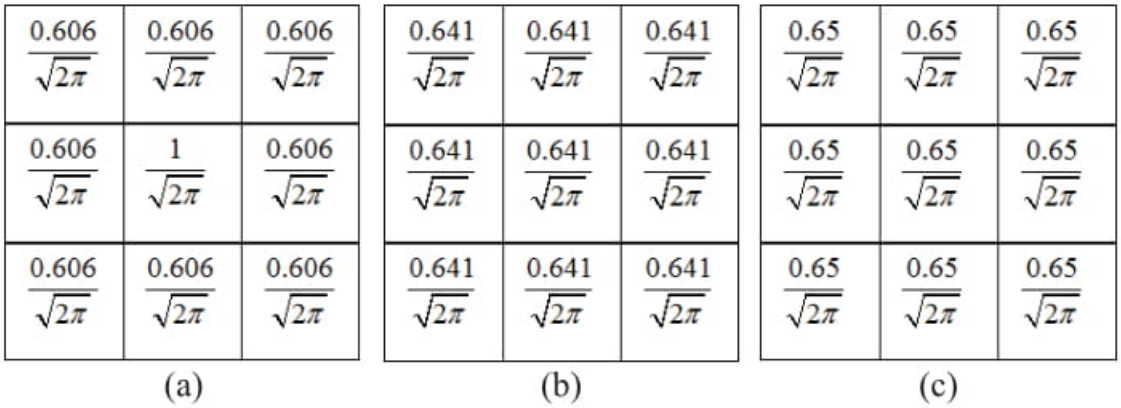

Figure 5: CPV of Fig. 1(c), (a) CPV with traditional FMM. (b) CPV calculated by geometric template. (c) CPV calculated by arithmetic template.

3.3 Mean Templates for Conditional and Prior Probabilities

In order to integrate the spatial information with the PP, Eq. (3) can be redefined as:

$$p(\vec{X}_n|\vec{\Theta}) = \sum_{j=1}^{M} \pi_{nj} p(\vec{X}_n|\vec{\theta_j}) \tag{57}$$

where π_{nj} is a updated mixing parameter where $j = 1, \ldots, M$, $\sum_{j=1}^{M} \pi_{nj} = 1$ and $n = 1, \ldots, N$. In this section, we discuss the geometric and arithmetic CP mean templates followed by their respective mixture models and complete log-likelihood equations. Furthermore, the equations of PP for both geometric and arithmetic mean templates are discussed.

3.3.1 Weighted Geometric Conditional Mean Template

In this section, we are using a weighted geometric conditional mean template (WGCMT) for calculating the CP of a pixel $\vec{X}_n$. Thus, Eq. (3) can be rewritten as:

$$p(\vec{X}_n|\vec{\Theta}) = \sum_{j=1}^{M} \pi_{nj} \prod_{r \in \mathcal{N}_n} p(\vec{X}_r|\vec{\theta_j})^{\frac{w_r}{R_n}} \tag{58}$$

where $\mathcal{N}_n$ is a set of peers of the n^{th} pixel. The conditional probability window is (CPW) = $\{\mathcal{N}_n, \vec{X}_n\}$. R_n is a normalized factor which is defined as

$$R_n = \sum_{r \in \mathcal{N}_n} w_r \tag{59}$$

In order to integrate the spatial information and pixel intensity value, the strength of w_r is inversely proportional to the distance between pixels r and n. Therefore, the authors in [51] have defined w_r as a function of d_{rn}, which is Euclidean distance between pixels r and n.

$$w_r = \frac{1}{\sqrt{2\pi\rho^2}} \exp\left(- \frac{d^2{}_{rn}}{2\rho^2} \right) \tag{60}$$

$$\rho = \frac{sizeofCPW - 1}{4} \tag{61}$$

Maximum Likelihood Estimation (MLE) for WGCMT

The pixels class labels are considered as the latent variables. Each pixel $\vec{X}_n$ is the observed data. The membership vector is defined as $Z = \{\vec{Z}_1, \ldots, \vec{Z}_N\}$ where $\vec{Z}_n = (Z_{n1}, ..., Z_{nM})$. If $\vec{X}_n$ belongs to cluster c, then $Z_{nc} = 1$ and $Z_{nl} = 0$ where $l = \{1, \ldots, M\} - \{c\}$, otherwise $Z_{nc} = 0$. The complete log-likelihood is as follows:

$$Q = \sum_n \sum_j Z_{nj} \left[\log \pi_{nj} + \sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} \log p(\vec{X_r} | \vec{\theta_j}) \right] \tag{62}$$

EM algorithm consists of two phases: E-Step and M-Step [9]. In E-Step, the posterior probability ($\hat{Z}_{nj}$) can be calculated as:

$$\hat{Z}_{nj}^{(t+1)} = \frac{\pi_{nj}^{(t)} \prod_{r \in \mathcal{N}_n} p(\vec{X_r} | \vec{\theta_j^{(t)}})^{\frac{w_r}{R_n}}}{\sum_{h=1}^{M} \pi_{nh}^{(t)} \prod_{r \in \mathcal{N}_n} p(\vec{X_r} | \vec{\theta_h^{(t)}})^{\frac{w_r}{R_n}}} \tag{63}$$

In M-Step, we have to maximize the complete log-likelihood and solve:

$$\frac{\partial Q}{\partial \theta_j} = 0 \tag{64}$$

3.3.2 Weighted Arithmetic Conditional Mean Template

In this part, we are using a weighted arithmetic conditional mean template (WACMT) to calculate the CP of a pixel $\vec{X}_n$. Thus, Eq. (58) can be rewritten as:

$$p(\vec{X}_n|\vec{\Theta}) = \sum_{j=1}^{M} \pi_{nj} \prod_{r \in \mathcal{N}_n} \frac{w_r}{R_n} p(\vec{X}_r|\vec{\theta}_j) \tag{65}$$

Maximum Likelihood Estimation (MLE) for WACMT

The complete log-likelihood is as follows:

$$Q = \sum_n \sum_j Z_{nj} \left[\log \pi_{nj} + \log \left(\sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} p(\vec{X}_r|\vec{\theta}_j) \right) \right] = \sum_n \sum_j Z_{nj} \left[\log \pi_{nj} + G \right] \tag{66}$$

G cannot be calculated directly. It is important to note that $\frac{w_r}{R_n}$ always follows the condition $\frac{w_r}{R_n} >= 0$ and $\sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} = 1$. Therefore, we can apply the Jensen's inequality rule which is defined as, given a set of numbers $\tau >= 0$ and $\sum_n \tau = 1$, we have $\log(\sum_n \tau x_i) >= \sum_n \tau \log(x_i)$. Then, the G can be modified and the complete log-likelihood is:

$$Q = \sum_n \sum_j Z_{nj} \left[\log \pi_{nj} + \sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} \log p(\vec{X}_r|\vec{\theta}_j) \right] \tag{67}$$

In E-Step, $\hat{Z}_{nj}$ can be calculated as

$$\hat{Z}_{nj}^{(t+1)} = \frac{\pi_{nj}^{(t)} \prod_{r \in \mathcal{N}_n} \frac{w_r}{R_n} p(\vec{X}_r|\vec{\theta_j^{(t)}})}{\sum_{h=1}^{M} \pi_{nh}^{(t)} \prod_{r \in \mathcal{N}_n} \frac{w_r}{R_n} p(\vec{X}_r|\vec{\theta_h^{(t)}})} \tag{68}$$

The M-Step can be computed using Eq. (65).

3.3.3 Weighted Prior Probability Estimation

The prior probability for FMM is as follows:

$$\pi_j = \frac{\sum_{n=1}^{N} Z_{nj}}{\sum_{n=1}^{N} \sum_{j=1}^{M} Z_{nj}} \tag{69}$$

According to the authors of [51], the weighted geometric prior mean template (WGPMT) is given as:

$$\pi_{nj}^{(t+1)} = \frac{\pi_{nj}^{(t)} \prod_{r \in \rho_n} {Z_{nj}}^{\frac{w_r}{R_n}}}{\sum_{h=1}^{M} \pi_{nh}^{(t)} \prod_{r \in \rho_n} {Z_{nj}}^{\frac{w_r}{R_n}}} \tag{70}$$

The weighted arithmetic prior mean template (WAPMT) is defined by:

$$\pi_{nj}^{(t+1)} = \frac{\pi_{nj}^{(t)} \prod_{r \in \rho_n} \frac{w_r}{R_n} Z_{nj}}{\sum_{h=1}^{M} \pi_{nh}^{(t)} \prod_{r \in \rho_n} \frac{w_r}{R_n} Z_{nj}} \tag{71}$$

3.4 Integration of Mean templates with IDMM, GIDMM and IBLMM

In this section, we propose our mixture models based on three probability density functions including ID, GID, IBL with incorporation of WGCMT and WACMT.

3.4.1 Incorporation of Mean Template with IDMM

In this subsection, we explain the integration of mean template with IDMM.

Incorporation of IDMM with WGCMT

By substituting Eq. (4) into Eq. (62), the complete log-likelihood is as follows:

$$Q = \sum_{n} \sum_{j} Z_{nj} \left[\log \pi_{nj} + \sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} \log \left(\frac{\Gamma(|\vec{\alpha}|)}{\prod_{d=1}^{D} \Gamma(\alpha_d)} \prod_{d=1}^{D+1} X_d^{\alpha_d - 1} \left(1 + |\vec{X}|\right)^{-|\vec{\alpha}|} \right) \right] \tag{72}$$

In E-Step, $\hat{Z}_{nj}$ can be calculated using Eq. (63). In M-step, we need to maximize the complete log-likelihood. From Eq. (64), the partial derivative of Q with respect to α_{jd} where $j = 1, \ldots, M$ and $d = 1, \ldots, D$ is as follows:

$$\frac{\partial Q}{\partial \alpha_{jd}} = \sum_{n=1}^{N} \sum_{j=1}^{M} \hat{Z}_{nj} \left\{ \Psi|\vec{\alpha}_j| - \Psi|\alpha_{jd}| + \sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} \log \left(\frac{X_{rd}}{1 + |\vec{X}_r|} \right) \right\} \tag{73}$$

where $\Psi(.)$ is the digamma function.
The partial derivative of Q with respect to α_{jD+1} is as follows:

$$\frac{\partial Q}{\partial \alpha_{jD+1}} = \sum_{n=1}^{N} \sum_{j=1}^{M} \hat{Z}_{nj} \left\{ \Psi|\vec{\alpha}_j| - \Psi|\alpha_{jd+1}| + \sum_{r \in \mathcal{N}_n} \frac{w_r}{R_n} \log \left(\frac{X_{rd}}{1 + |\vec{X}_r|} \right) \right\} \tag{74}$$

Considering Eq. (73) and (74), it can be observed that no closed solution exists for $\vec{\alpha}_j$. Therefore, we have used Newton-Raphson method as follows:

$$\vec{\alpha}_j^{(k+1)} = \vec{\alpha}_j^{(k)} - \vec{G}_j H_j^{-1} \tag{75}$$

where $\vec{\alpha_j}^{(k+1)}$ is the updated hyper-parameter, $\vec{\alpha_j}^{(k)}$ is the old hyper-parameter, $\vec{G}_j$ is the gradient followed by H_j^{-1}, which is the inverse of Hessian matrix. The Gradient is the first partial order derivative of Q and described as follows:

$$\vec{G}_j = \left(\frac{\partial Q}{\partial \alpha_{j1}}, \ldots, \frac{\partial Q}{\partial \alpha_{jD+1}}\right) \tag{76}$$

To find the Hessian of Q, we have to calculate the second and mixed derivatives:

$$\frac{\partial^2 Q}{\partial^2 \alpha_{jd}} = \sum_{n=1}^{N} \hat{Z}_{nj}\Big(\Psi'(|\vec{\alpha_j}|) - \Psi'(\alpha_{jd})\Big), \quad d = 1, \ldots, D+1 \tag{77}$$

$$\frac{\partial^2 Q}{\partial^2 \alpha_{jd_1}\alpha_{jd_2}} = \Psi'(|\vec{\alpha_j}|)\sum_{n=1}^{N} \hat{Z}_{nj}, \quad d1 \neq d2, \quad d_1, d_2 = 1, \ldots, D+1 \tag{78}$$

where $\Psi'(.)$ is the trigamma function. The Hessian can be described as:

$$H_j = \sum_{n=1}^{N} \hat{Z}_{nj} \begin{bmatrix} \Psi'(|\vec{\alpha_j}|) - \Psi'(\alpha_{j1}) & \Psi'(|\vec{\alpha_j}|) & \ldots & \Psi'(|\vec{\alpha_j}|) \\ \Psi'(|\vec{\alpha_j}|) & \Psi'(|\vec{\alpha_j}|) - \Psi'(\alpha_{j2}) & \ldots & \Psi'(|\vec{\alpha_j}|) \\ \vdots & \ldots & \ddots & \vdots \\ \Psi'(|\vec{\alpha_j}|) & \ldots & & \Psi'(|\vec{\alpha_j}|) - \Psi'(\alpha_{jD+1}) \end{bmatrix} \tag{79}$$

Thus, H_j can be written as:

$$H_j = D_j + \rho_j {A_j}^T A_j \tag{80}$$

where D_j is a diagonal matrix and described by:

$$D_j = diag\Big(-\sum_{n=1}^{N} \hat{Z}_{nj}\Psi'(\alpha_{j1}), \ldots, -\sum_{n=1}^{N} \hat{Z}_{nj}\Psi'(\alpha_{jD+1})\Big) \tag{81}$$

The constant ρ_j is defined as:

$$\rho_j = \left[\left((\Psi'(|\vec{\alpha_j}|)\sum_{d=1}^{D+1}\frac{1}{\Psi'(\alpha_{jd})}\right) - 1\right]\Psi'(|\vec{\alpha_j}|)\sum_{n=1}^{N}\hat{Z}_{nj} \tag{82}$$

$A_j^T = (a_1, \ldots, a_{D+1}), a_d = 1$ where $d = 1, \ldots, D+1$. In order to find H_j^{-1}, a matrix inverse theorem given in [69] can be used [2]:

$$H_j^{-1} = D_j^{-1} + {\rho^*}_j A_j^{*T} A_j^* \tag{83}$$

D_j^{-1} can be easily computed. ${A_j}^*$ and ${\rho^*}_j$ are expressed by two following equations:

$${A_j}^* = \frac{-1}{\sum_{n=1}^{N}\hat{Z}_{nj}}\left[\frac{1}{\Psi'(\alpha_{j1})}, \ldots, \frac{1}{\Psi'(\alpha_{j,D+1})}\right] \tag{84}$$

$${\rho^*}_j = \Psi'(|\vec{\alpha_j}|)\sum_{n=1}^{N}\hat{Z}_{nj}\left[\Psi'(|\vec{\alpha_j}|)\sum_{n=1}^{N}\frac{1}{\Psi'(\alpha_{j_d})} - 1\right] \tag{85}$$

Incorporation of IDMM with WACMT

In E-Step, the $\hat{Z}_{nj}$ can be calculated using Eq. (68). The M-Step can be calculating using Eq. (64).

IDMM's Algorithm

We have two conditional probabilities that are WGCMT, WACMT and two prior probabilities that are WGPMT, WAPMT. Therefore, we can have four models as following:
1) GCGP: the application of weighted geometric conditional mean template to weighted geometric prior mean template.
2) GCAP: the application of weighted geometric conditional mean template to weighted arithmetic prior mean template.
3) ACGP: the application of weighted arithmetic conditional mean template to weighted geometric prior mean template.
4) ACAP: the application of weighted arithmetic conditional mean template to weighted arithmetic prior mean template.

The algorithm is as follows:

1. INPUT: An image $\mathcal{X}$ and M.
2. Apply K-means clustering algorithm to group pixels into M clusters.
3. Apply Method of Moments to calculate the initial value for $\vec{\alpha}$ parameter.
4. E-Step:
    - Calculate $\vec{\pi}$ using Eq. (70) for GCGP and ACGP, and using Eq. (71) for GCAP and ACAP.
    - Calculate $\hat{Z}_{nj}$ using Eq. (63) for GCGP and GCAP, and using Eq. (68) for ACGP and ACAP.
5. M-Step: Calculate the updated value of $\vec{\alpha}$ parameter for each cluster j, using Eq. (75).
6. Iterate through E-Step and M-Step until convergence.

3.4.2 Mean Template Incorporation with GIDMM

The second mixture model is GIDMM and we are integrating the mean template with it.

Incorporation of GIDMM with WGCMT

By substituting Eq. (17) into Eq. (67) , the complete log-likelihood is given by:

$$Q = \sum_{n=1}^{N}\sum_{j=1}^{M} Z_{nj}\left[\log \pi_{nj} + \sum_{r\in\mathcal{N}_n}\frac{w_r}{R_n}\log\left(\frac{\Gamma(\alpha_{jd}+\beta_{jd})}{\Gamma(\alpha_{jd})\Gamma(\beta_{jd})}W_{nd}^{\alpha_{jd}-1}(1+W_{jd})^{-(\alpha_{jd}+\beta_{jd})}\right)\right] \tag{86}$$

In E-Step, the $\hat{Z}_{nj}$ can be calculated using Eq. (63) and in M-Step, the partial derivatives of Q with respect to α_{jd} and β_{jd} are as follows:

$$\frac{\partial Q}{\partial \alpha_{jd}} = \sum_{i=1}^{N}\hat{Z}_{nj}\left\{\Psi(\alpha_{jd}+\beta_{jd}) - \Psi(\alpha_{jd}) + \log\left(\frac{W_{nd}}{1+W_{nd}}\right)\right\} \tag{87}$$

$$\frac{\partial Q}{\partial \beta_{jd}} = \sum_{i=1}^{N}\hat{Z}_{nj}\left\{\Psi(\alpha_{jd}+\beta_{jd}) - \Psi(\beta_{jd}) + \log\left(\frac{W_{nd}}{1+W_{nd}}\right)\right\} \tag{88}$$

From Eq. (87) and Eq. (88) , it can be observed that no closed-form solution exists for $\vec{\theta_{jd}}$. Therefore, we have to use Newton-Raphson method as follows:

$$\vec{\theta_{jd}}^{(k+1)} = \vec{\theta_{jd}}^{(k)} - H_{jd}^{-1}G_{jd} \tag{89}$$

where H_{jd} is the Hessian matrix [1] and given as:

$$H_{jd} = \begin{bmatrix} \frac{\partial^2 Q}{\partial^2 \alpha_{jd}} & \frac{\partial^2 Q}{\partial^2 \alpha_{jd}\beta_{jd}} \\ \frac{\partial^2 Q}{\partial^2 \alpha_{jd}\beta_{jd}} & \frac{\partial^2 Q}{\partial^2 \beta_{jd}} \end{bmatrix} \tag{90}$$

The second and mixed derivatives of Q are as follows:

$$\frac{\partial^2 Q}{\partial^2 \alpha_{jd}} = \sum_{n=1}^{N}\hat{Z}_{nj}\Big(\Psi'(\alpha_{jd}+\beta_{jd}) - \Psi'(\alpha_{jd})\Big), \quad d = 1,\ldots,D+1 \tag{91}$$

$$\frac{\partial^2 Q}{\partial^2 \beta_{jd}} = \sum_{n=1}^{N}\hat{Z}_{nj}\Big(\Psi'(\alpha_{jd}+\beta_{jd}) - \Psi'(\beta_{jd})\Big) \tag{92}$$

$$\frac{\partial^2 Q}{\partial^2 \alpha_{jd}\beta_{jd}} = \Psi'(\alpha_{jd}+\beta_{jd})\sum_{n=1}^{N}\hat{Z}_{nj} \tag{93}$$

$\vec{G}_{jd}$ is defined as follows:

$$\vec{G}_{jd} = \left(\frac{\partial Q}{\partial \alpha_{jd}}, \frac{\partial Q}{\partial \beta_{jd}}\right) \tag{94}$$

Incorporation of GIDMM with WACMT

In E-Step, the $\hat{Z}_{nj}$ can be calculated using Eq. (68) . The M-Step is based on Eq. (64) .

GIDMM's Algorithm

In this section, we propose an algorithm for the four models that are GCGP, GCAP, ACGP and ACAP.

1. INPUT: An image $\mathcal{X}$ and M.
2. Apply K-means clustering algorithm to group pixels into M clusters.
3. Apply Method of Moments to calculate the initial value for $\vec{\alpha}$ parameter.
4. E-Step:
   - Calculate $\vec{\pi}$ using Eq. (70) for GCGP and ACGP, and using Eq. (71) for GCAP and ACAP.
   - Calculate $\hat{Z}_{nj}$ using Eq. (63) for GCGP and GCAP, and using Eq. (68) for ACGP and ACAP.
5. M-Step: Calculate the updated value of θ parameter for each cluster j, using Eq. (90).
6. Iterate through E-Step and M-Step until convergence.

3.4.3 Incorporation of Mean Template with IBLMM

In this subsection, we integrate the mean template with IBLMM.

The Probability Density Function of IBL

If $\vec{X}$ is a positive vector which consists of D dimensions and following an IBL distribution, then it has a joint density function which is given in [53] as:

$$p(\vec{X}|\alpha_1 \dots \alpha_d, \alpha, \beta, \lambda) = \frac{\Gamma(|\vec{\alpha}|)\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \prod_{d=1}^{D+1} \frac{X_d^{\alpha_d - 1}}{\Gamma(\alpha_d)} \lambda^{\beta} \left(|\vec{X}|\right)^{\alpha - \sum_{d=1}^{D} \alpha_d} \left(\lambda + |\vec{X}|\right)^{-(\alpha+\beta)} \tag{95}$$

where $|\vec{X}| = \sum_{d=1}^{D} X_d$, each $X_d > 0$, $\alpha > 0$, $\beta > 0$ and $\lambda > 0$. The mean and the variance of IBL are given by:

$$E(X_d) = \frac{\alpha_d}{\alpha_{D+1} - 1}; \tag{96}$$

$$Var(X_d) = \frac{\alpha_d(\alpha_d + \alpha_{D+1} - 1)}{(\alpha_{D+1} - 1)^2(\alpha_{D+1} - 2)} \tag{97}$$

Incorporation of IBLMM with WGCMT

By substituting Eq. (95) into Eq. (62), the complete log-likelihood is as follows:

$$Q = \sum_{n}\sum_{j} Z_{nj}\left[\log \pi_{nj} + \sum_{r\in\mathcal{N}_n} \frac{w_r}{R_n} \log\left(\frac{\Gamma(|\vec{\alpha}|)\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \prod_{d=1}^{D+1} \frac{X_d^{\alpha_d - 1}}{\Gamma(\alpha_d)} \lambda^{\beta} \left(|\vec{X}|\right)^{\alpha - \sum_{d=1}^{D}\alpha_d} \left(\lambda + |\vec{X}|\right)^{-(\alpha+\beta)}\right)\right] \tag{98}$$

In E-Step, $\hat{Z}_{nj}$ can be calculated using Eq. (63). In M-step, we need to calculate the partial derivative of Q with respect to the parameters of IBL [73].

The partial derivative of Q with respect to α_j where $j = 1, \ldots, M$ is as follows:

$$\frac{\partial Q}{\partial \alpha_j} = \sum_{i=1}^{N} \hat{Z}_{nj}\left\{\log \sum_{d=1}^{D} X_{nd} - \log\left(\lambda_j + \sum_{d=1}^{D} X_{nd}\right) + \Psi(\alpha_j + \beta_j) - \Psi(\alpha_j)\right\} \tag{99}$$

The partial derivative of Q with respect to β_j is given by:

$$\frac{\partial Q}{\partial \beta_j} = \sum_{i=1}^{N} \hat{Z}_{nj}\left\{\log \lambda_j - \log\left(\lambda_j + \sum_{d=1}^{D} X_{nd}\right) + \Psi(\alpha_j + \beta_j) - \Psi(\beta_j)\right\} \tag{100}$$

The partial derivative of Q with respect to α_{jd} is defined as:

$$\frac{\partial Q}{\partial \alpha_{jd}} = \sum_{i=1}^{N} \hat{Z}_{nj}\left\{\log X_{nd} - \log \sum_{d=1}^{D} X_{nd} + \Psi(\sum_{d=1}^{D} \alpha_{jd}) - \Psi(\alpha_{jd})\right\} \tag{101}$$

The partial derivative of Q with respect to λ_j is expressed as follows:

$$\frac{\partial Q}{\partial \lambda_j} = \sum_{i=1}^{N} \hat{Z}_{nj}\left\{\frac{\beta_j}{\lambda_j} - \frac{\alpha_j + \beta_j}{\lambda_j + \ sum_{d=1}^{D} X_{nd}}\right\} \tag{102}$$

From Eq. (99) to Eq. (102), it can be observed that a closed-form solution does not exist for $\vec{\theta_j}$.

In order to estimate these parameters, Newton-Raphson method can be used:

$$\vec{\theta_j}^{(k+1)} = \vec{\theta_j}^{(k)} - H_j^{-1} G_j \tag{103}$$

Incorporation of IBMM with WACMT

In E-Step, the $\hat{Z}_{nj}$ can be calculated using Eq. (69). The M-Step is performed via Eq. (65).

IBLMM's Algorithm

In this section, we propose an algorithm for the four models namely GCGP, GCAP, ACGP and ACAP.

1. INPUT: An image $\mathcal{X}$ and M.
2. Apply K-means clustering algorithm to group pixels into M clusters.
3. Apply Method of Moments to calculate the initial value of $\vec{\theta}$ parameters.
4. E-Step:
   - Calculate $\vec{\pi}$ using Eq. (70) for GCGP and ACGP, and using Eq. (71) for GCAP and ACAP.
   - Calculate $\hat{Z}_{nj}$ using Eq. (63) for GCGP and GCAP, and using Eq. (68) for ACGP and ACAP.
5. M-Step: Calculate the updated value of $\vec{\theta}$ parameter for each cluster j, using Eq. (103).
6. Iterate through E-Step and M-Step until convergence.

3.5 Experimental Results

To investigate the performance of our proposed framework, we test the models on two different datasets that are BSD500 and CSCV. The BSD500 dataset is known as a reliable source to compare different image segmentation algorithms and contains 500 color images and has at least 5 ground-truth segments for each image. The CSCV dataset is composed of many categories such as Coast and Beach, Highway, etc. Each category contains few hundred images. All the images are in color, in jpeg format, and are 256 x 256 pixels. Their sources varies from digital cameras, websites and commercial databases. This section is composed of two experiments. In first one, we tested the proposed models on BSD500 and evaluated the results using segmentation evaluation metrics. In second experiment, we employed CSCV dataset and compared our models using two color spaces that are rgb and $l_1l_2l_3$, which are explained below.

3.5.1 Experiment 1

Here, we present some results of testing our models on images from the BSD500 using $l_1l_2l_3$ color space. Fig. 6 contains the segmentation outputs of image 29030 (obtained by using ID, GID, and IBL versions of GCGP, GCAP, ACGP, and ACAP models). Considering the second, third and fourth columns of Fig. 6, it can be observed that IBL is able to detect car as an object properly, followed by GID and ID. Table 5 and Table 6 verify our visual analysis by means of qualitative approach using the image segmentation evaluation metrics. Images in BSD500 may contain upto six human segmentations. The output of each model is compared with each human segmentation by using evaluation metrics. By incorporating 3 pdfs with 4 mean template models, we have 12 models. Each result of these models is compared with 6 human segments which led to have upto 72 comparisons. In order to reduce the complexity in understanding, we have compared the mean of each model with the human segmentations.

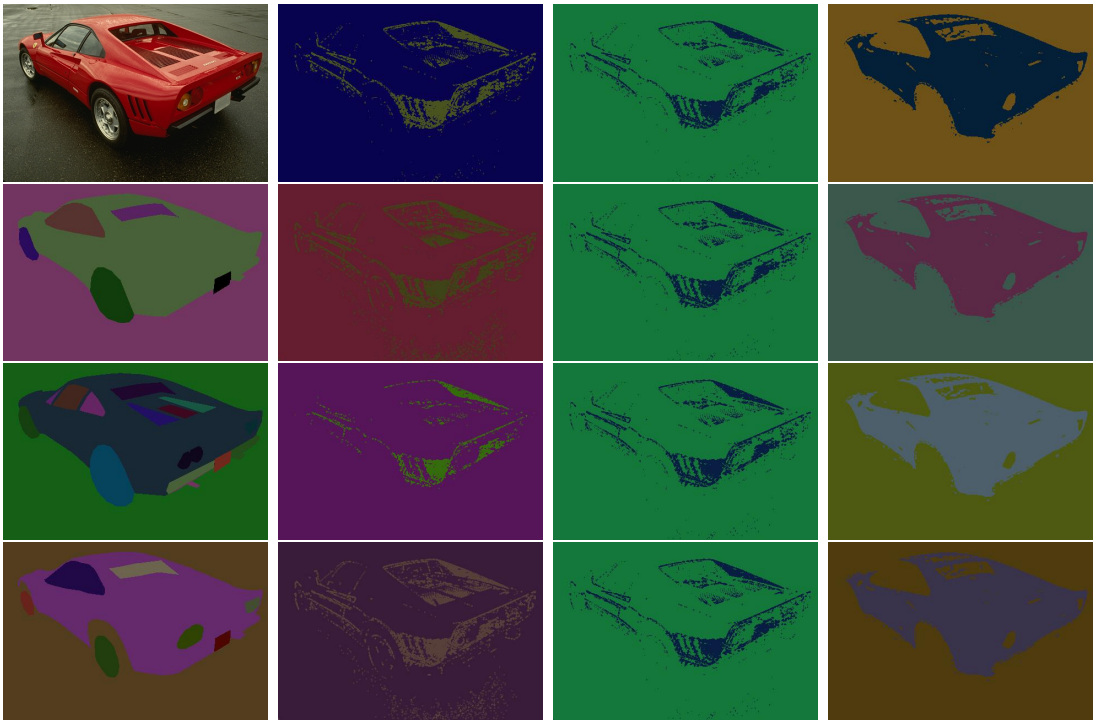

Figure 6: Column 1: Contains the original image (29030) followed by the three ground-truths. Column 2: Contains the segmentation outputs from ID version of GCGP, GCAP, ACGP, and ACAP models, Column 3: Contains the segmentation outputs from GID version of GCGP, GCAP, ACGP, and ACAP models, Column 4: Contains the segmentation outputs from IBL version of GCGP, GCAP, ACGP, and ACAP models.

Table 5: Performance evaluation of the 29030 image with the ARI, AMIS, NMIS, MIS and HS metrics.

	Metrics (K=2)					
Alg.	**Model**	**ARI**	**AMIS**	**NMIS**	**MIS**	**HS**
ID	GCGP	0.092	0.063	0.135	0.078	0.063
	GCAP	0.111	0.052	0.099	0.065	0.052
	ACGP	0.042	0.039	0.104	0.048	0.039
	ACAP	0.112	0.061	0.119	0.075	0.061
Mean		**0.089**	**0.054**	**0.114**	**0.067**	**0.054**
GID	GCGP	0.084	0.059	0.129	0.073	0.059
	GCAP	0.081	0.057	0.126	0.071	0.057
	ACGP	0.079	0.056	0.124	0.069	0.056
	ACAP	0.077	0.055	0.123	0.068	0.055
Mean		**0.080**	**0.057**	**0.126**	**0.070**	**0.057**
IBL	GCGP	0.662	0.420	0.587	0.520	0.420
	GCAP	0.664	0.419	0.585	0.519	0.419
	ACGP	0.650	0.412	0.578	0.511	0.412
	ACAP	0.651	0.412	0.578	0.510	0.412
Mean		**0.6569**	**0.4158**	**0.5821**	**0.5151**	**0.4158**

Table 6: Quality analysis of the 29030 image with the CS, VM, JSS, and CHI metrics.

Alg.	Metrics (K=2)				
	Model	**CS**	**VM**	**JSS**	**CHI**
ID	GCGP	0.289	0.103	0.001	73962.307
	GCAP	0.190	0.081	0.013	54791.895
	ACGP	0.281	0.068	0.000	42512.321
	ACAP	0.236	0.096	0.007	64281.498
Mean		**0.249**	**0.087**	**0.005**	**58887.006**
GID	GCGP	0.285	0.098	0.001	71021.648
	GCAP	0.282	0.095	0.001	67895.016
	ACGP	0.279	0.092	0.001	65785.996
	ACAP	0.277	0.091	0.001	64006.640
Mean		**0.281**	**0.094**	**0.001**	**268709.300**
IBL	GCGP	0.826	0.553	0.003	262224.285
	GCAP	0.823	0.552	0.004	259172.527
	ACGP	0.817	0.545	0.003	265561.038
	ACAP	0.815	0.544	0.003	264630.709
Mean		**0.8202**	**0.5487**	**0.0031**	**262897.140**

Fig. 7 contains the segmentation outputs of image 118035. Again, from the second, third and fourth columns of Fig. 7, it can be observed that IBLMM is able to detect the different components of a building much accurately as compared to GIDMM and IDMM. Table 7 and Table 8 contain the qualitative analysis of outputs from image 118035.

Figure 7: Column 1: Contains the original image (118035) followed by the three ground-truths. Column 2: Contains the segmentation outputs from ID's version of GCGP, GCAP, ACGP, and ACAP models, Column 3: Contains the segmentation outputs from GID's version of GCGP, GCAP, ACGP, and ACAP models, Column 4: Contains the segmentation outputs from IBL's version of GCGP, GCAP, ACGP, and ACAP models

Table 7: Performance evaluation of the 118035 image with the ARI, AMIS, NMIS, MIS and HS metrics.

	Metrics (K=2)					
Alg.	**Model**	**ARI**	**AMIS**	**NMIS**	**MIS**	**HS**
ID	GCGP	0.157	0.174	0.372	0.258	0.174
	GCAP	0.158	0.174	0.371	0.258	0.174
	ACGP	0.171	0.174	0.351	0.260	0.174
	ACAP	0.318	0.223	0.345	0.334	0.223
Mean		**0.201**	**0.186**	**0.360**	**0.278**	**0.186**
GID	GCGP	0.179	0.143	0.249	0.217	0.143
	GCAP	0.159	0.130	0.232	0.198	0.130
	ACGP	0.162	0.133	0.236	0.201	0.133
	ACAP	0.165	0.134	0.237	0.204	0.134
Mean		**0.166**	**0.135**	**0.238**	**0.205**	**0.135**
IBL	GCGP	0.589	0.438	0.631	0.644	0.438
	GCAP	0.589	0.438	0.632	0.644	0.438
	ACGP	0.589	0.438	0.631	0.644	0.438
	ACAP	0.588	0.436	0.629	0.641	0.436
Mean		**0.5887**	**0.4376**	**0.6308**	**0.6432**	**0.4376**

Table 8: Quality analysis of the 118035 image with the CS, VM, JSS, and CHI metrics.

Alg.	Metrics (K=2)				
	Model	**CS**	**VM**	**JSS**	**CHI**
ID	GCGP	0.810	0.283	0.453	184910.025
	GCAP	0.810	0.283	0.453	184682.459
	ACGP	0.722	0.277	0.452	163659.995
	ACAP	0.543	0.311	0.452	68588.861
Mean		**0.721**	**0.288**	**0.452**	**150460.335**
GID	GCGP	0.810	0.283	0.453	184910.025
	GCAP	0.810	0.283	0.453	184682.459
	ACGP	0.722	0.277	0.452	163659.995
	ACAP	0.543	0.311	0.452	68588.861
Mean		**0.721**	**0.288**	**0.452**	**150460.335**
IBL	GCGP	0.927	0.586	0.452	133485.637
	GCAP	0.928	0.586	0.452	133528.359
	ACGP	0.927	0.585	0.452	133460.767
	ACAP	0.923	0.583	0.452	133191.467
Mean		**0.9262**	**0.5850**	**0.4521**	**133416.557**

Similarly, Fig. 8 contains the segmentation results of 124084. From the second, third and fourth columns of Fig. 8, it can be seen that IBLMM is able to detect the flower petals pretty smoothly as compared to its competitors. Table 9 and Table 10 contain the qualitative analysis of outputs from image 124084.

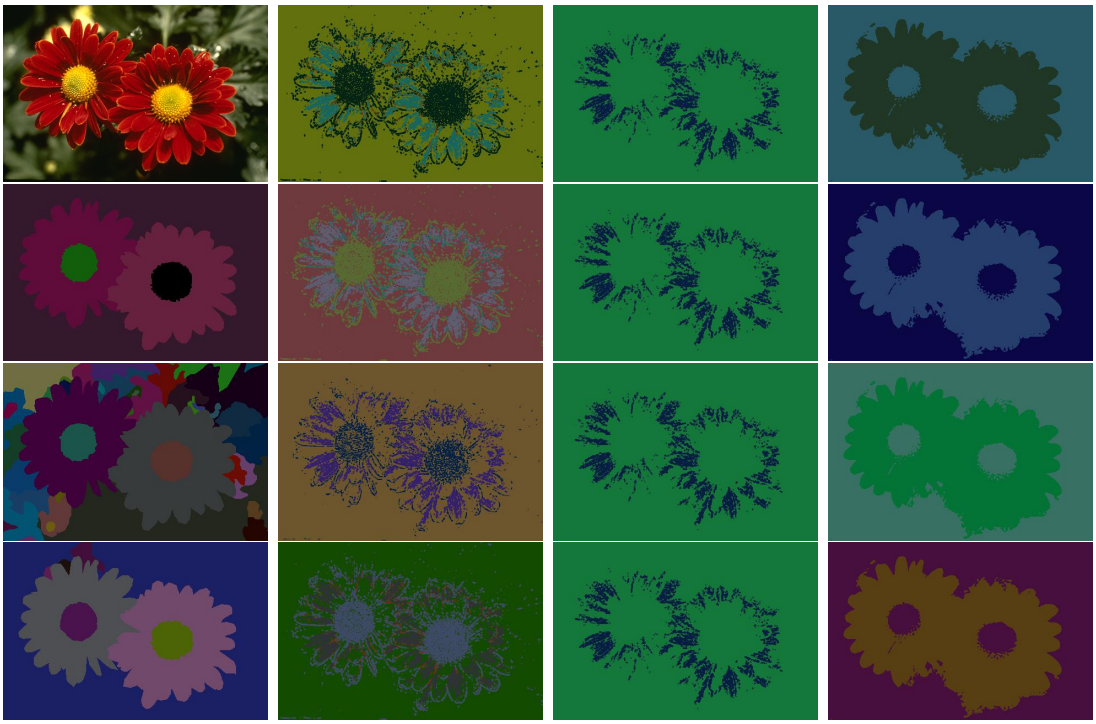

Figure 8: Column 1: Contains the original image (124084.jpg) followed by the three ground-truths. Column 2: Contains the segmentation outputs from ID's version of GCGP, GCAP, ACGP, and ACAP models, Column 3: Contains the segmentation outputs from GID's version of GCGP, GCAP, ACGP, and ACAP models, Column 4: Contains the segmentation outputs from IBL's version of GCGP, GCAP, ACGP, and ACAP models

Table 9: Performance evaluation of the 124084 image with the ARI, AMIS, NMIS, MIS and HS metrics.

	Metrics (K=2)					
Alg.	**Model**	**ARI**	**AMIS**	**NMIS**	**MIS**	**HS**
ID	GCGP	0.046	0.030	0.084	0.044	0.030
	GCAP	0.134	0.085	0.162	0.126	0.086
	ACGP	0.053	0.035	0.092	0.052	0.035
	ACAP	0.065	0.042	0.104	0.062	0.042
Mean		**0.075**	**0.048**	**0.110**	**0.071**	**0.048**
GID	GCGP	0.112	0.071	0.150	0.104	0.071
	GCAP	0.112	0.071	0.150	0.104	0.071
	ACGP	0.112	0.071	0.150	0.104	0.071
	ACAP	0.112	0.071	0.150	0.104	0.071
Mean		**0.112**	**0.071**	**0.150**	**0.104**	**0.071**
IBL	GCGP	0.502	0.354	0.511	0.519	0.354
	GCAP	0.500	0.352	0.508	0.516	0.352
	ACGP	0.500	0.353	0.510	0.517	0.353
	ACAP	0.501	0.353	0.510	0.517	0.353
Mean		**0.5007**	**0.3529**	**0.5097**	**0.5171**	**0.3529**

Table 10: Quality analysis of the 124084 image with the CS, VM, JSS, and CHI metrics.

Alg.	Metrics (K=2)				
	Model	**CS**	**VM**	**JSS**	**CHI**
ID	GCGP	0.239	0.053	0.454	5717.493
	GCAP	0.313	0.132	0.454	15553.412
	ACGP	0.249	0.061	0.454	6769.251
	ACAP	0.264	0.071	0.454	8221.494
Mean		**0.266**	**0.079**	**0.454**	**9065.413**
GID	GCGP	0.327	0.115	0.000	47850.229
	GCAP	0.326	0.115	0.000	47616.519
	ACGP	0.326	0.115	0.000	47471.541
	ACAP	0.326	0.114	0.000	47390.939
Mean		**0.326**	**0.115**	**0.000**	**190329.228**
IBL	GCGP	0.759	0.472	0.427	592730.642
	GCAP	0.754	0.469	0.427	589902.293
	ACGP	0.756	0.471	0.427	587628.857
	ACAP	0.756	0.471	0.427	591670.571
Mean		**0.7562**	**0.4707**	**0.4268**	**590483.091**

Fig. 9 contains the segmentation results of image 376086. From the second, third and fourth columns of Fig. 9, it can be seen that IBLMM is able to detect the two men more accurately as compare to GIDMM and IDMM. Table 11 and 12 contain the image segmentation results for image 376086.

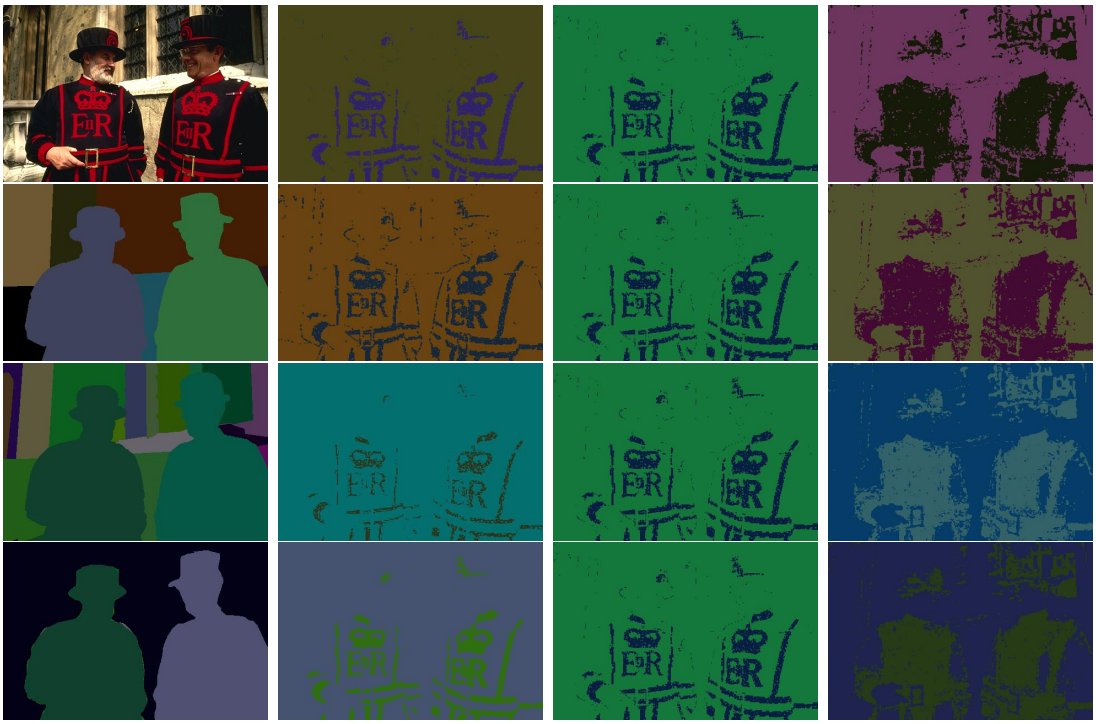

Figure 9: Column 1: Contains the original image (376086) followed by three ground-truths. Column 2: Contains the segmentation outputs from ID's version of GCGP, GCAP, ACGP, and ACAP models, Column 3: Contains the segmentation outputs from GID's version of GCGP, GCAP, ACGP, and ACAP models, Column 4: Contains the segmentation outputs from IBL's version of GCGP, GCAP, ACGP, and ACAP models

Table 11: Performance evaluation of the 376086 image with the ARI, AMIS, NMIS, MIS and HS metrics.

Alg.	Metrics (K=2)					
	Model	**ARI**	**AMIS**	**NMIS**	**MIS**	**HS**
ID	GCGP	-0.004	0.047	0.109	0.097	0.047
	GCAP	-0.003	0.040	0.091	0.084	0.041
	ACGP	-0.006	0.023	0.071	0.049	0.023
	ACAP	-0.004	0.047	0.112	0.098	0.047
Mean		**-0.004**	**0.039**	**0.096**	**0.082**	**0.039**
GID	GCGP	-0.004	0.046	0.107	0.095	0.046
	GCAP	-0.004	0.047	0.110	0.097	0.047
	ACGP	-0.004	0.048	0.112	0.098	0.048
	ACAP	-0.004	0.048	0.112	0.098	0.048
Mean		**-0.004**	**0.047**	**0.110**	**0.097**	**0.047**
IBL	GCGP	0.064	0.103	0.180	0.208	0.103
	GCAP	0.061	0.102	0.178	0.206	0.102
	ACGP	0.062	0.102	0.178	0.206	0.102
	ACAP	0.064	0.103	0.179	0.208	0.103
Mean		**0.0627**	**0.1027**	**0.1788**	**0.2070**	**0.1028**

Table 12: Quality analysis of the 376086 image with the CS, VM, JSS, and CHI metrics.

Alg.	Metrics (K=2)				
	Model	**CS**	**VM**	**JSS**	**CHI**
ID	GCGP	0.267	0.078	0.000	197432.441
	GCAP	0.212	0.066	0.003	165708.431
	ACGP	0.231	0.041	0.000	71589.989
	ACAP	0.274	0.079	0.000	197429.295
Mean		**0.249**	**0.066**	**0.001**	**158040.039**
GID	GCGP	0.260	0.076	0.001	195016.547
	GCAP	0.268	0.078	0.000	198092.848
	ACGP	0.273	0.079	0.000	198893.445
	ACAP	0.273	0.079	0.000	198893.445
Mean		**0.269**	**0.078**	**0.000**	**790896.285**
IBL	GCGP	0.324	0.153	0.136	54345.372
	GCAP	0.322	0.151	0.136	54980.749
	ACGP	0.322	0.151	0.136	56306.326
	ACAP	0.324	0.152	0.136	54602.056
Mean		**0.3229**	**0.1516**	**0.1359**	**55058.626**

3.5.2 Experiment 2

Considering Fig. 10 and Fig. 11, the first image is the original one (n291030), followed by eight outputs, out of which the first four outputs are computed using rgb color space and the remaining four outputs are obtained by using $l_1l_2l_3$ color space.

Figure 10: Original image (n291030) followed by the eight output images from ID's version of the GCGP, GCAP, ACGP, and ACAP models, out of which the first four images, have used rgb color space and remaining four have used $l_1l_2l_3$ color space.

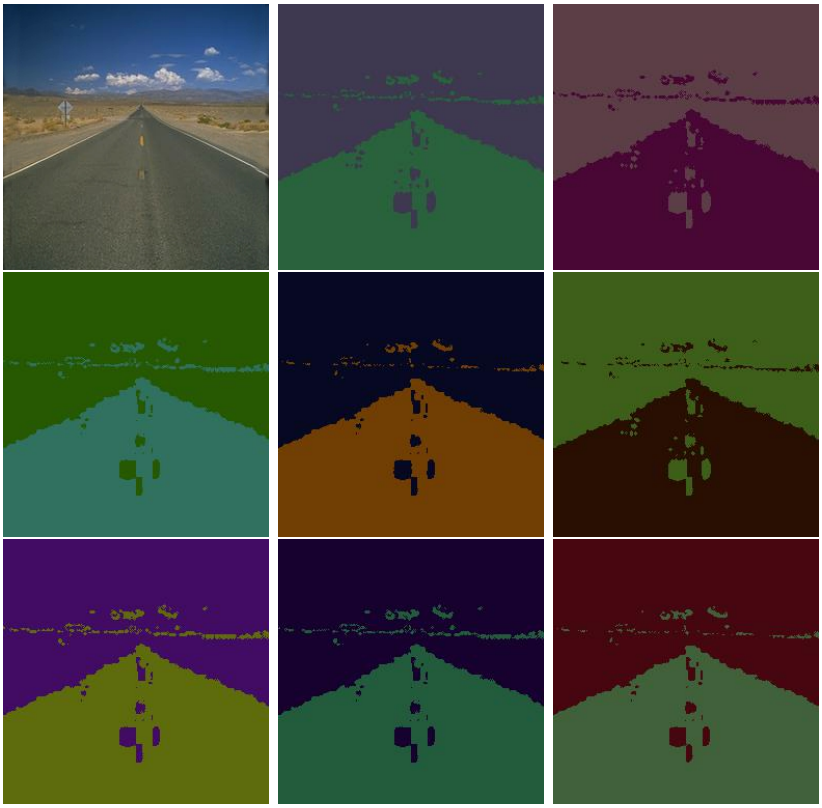

Figure 11: Contains the original image (art255) followed by the eight output images from ID's version of the GCGP, GCAP, ACGP, and ACAP models. Out of which the first four images, have used rgb color space and remaining four have used $l_1l_2l_3$ color space.

Also, Fig. 12 and Fig. 13 contain the segmentation output of images n291030 and art255 respectively. In Fig. 12 and Fig. 13, the first image is the original one, followed by eight outputs, out of which in the first four image, rgb color space is used and for rest segmentation outputs, $l_1l_2l_3$ color space is used.

Figure 12: Original image (n291030) followed by the eight output images from GID's version of the GCGP, GCAP, ACGP, and ACAP models out of which the first four images, have used rgb color space and remaining four have used $l_1l_2l_3$ color space.

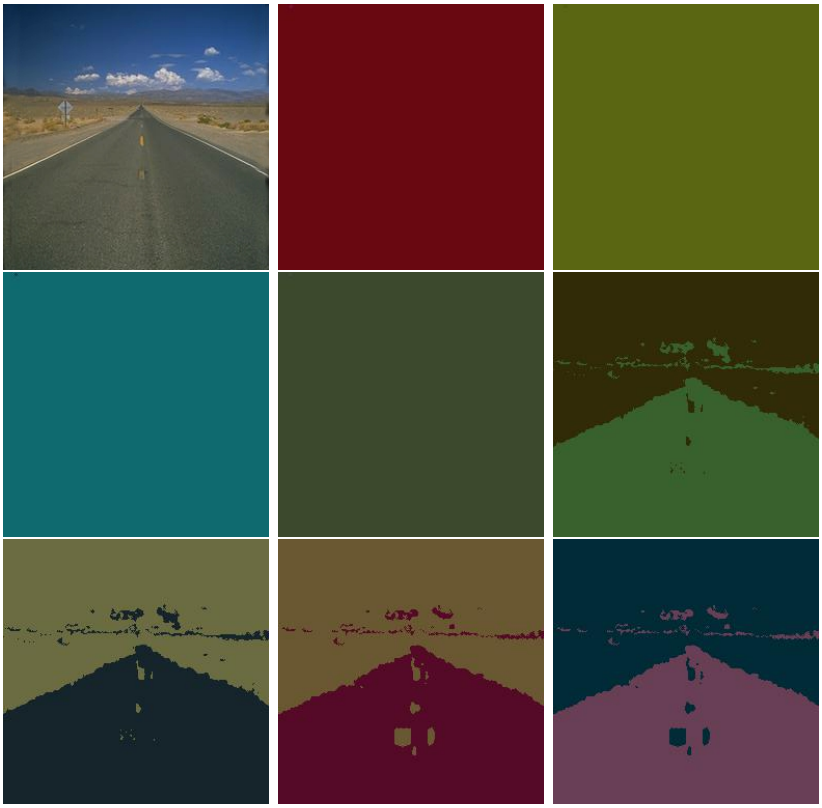

Figure 13: Original image (art255) followed by the eight output images from GID's version of the GCGP, GCAP, ACGP, and ACAP models, out of which the first four images, have used rgb color space and remaining four have used $l_1l_2l_3$ color space.

Similarly, Fig. 14 and Fig. 15 contain the segmentation output of image n291030 and art255 respectively. In Fig. 14 and Fig. 15, the first image is the original one, followed by eight outputs, out of which the first four segmentation output, rgb color space is used and for remaining four outputs, $l_1l_2l_3$ color space is used.

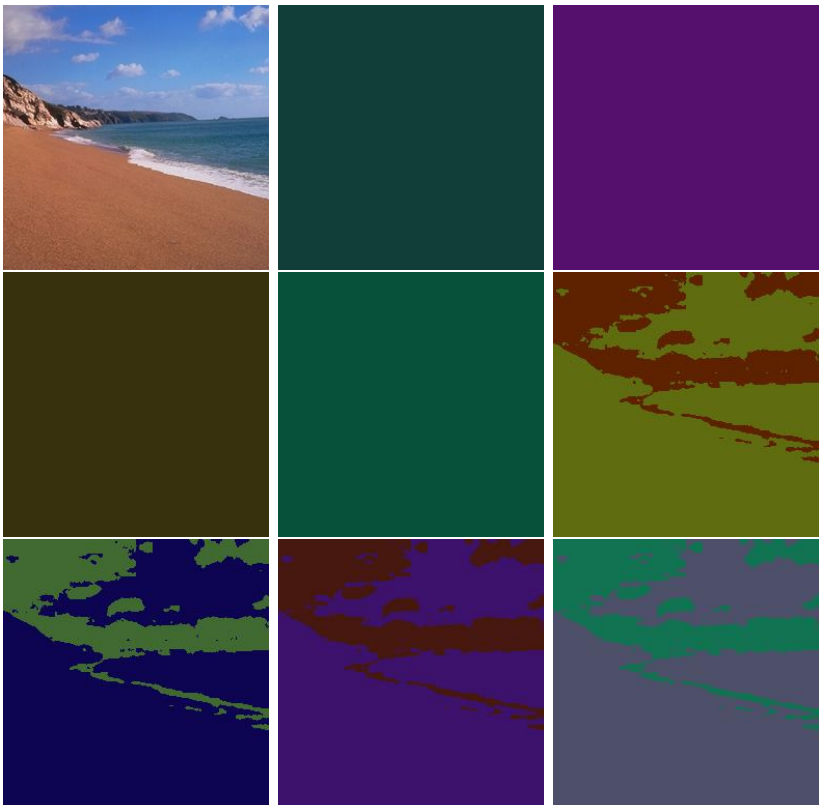

Figure 14: Original image (n291030) followed by the eight output images from IBL's version of the GCGP, GCAP, ACGP, and ACAP models out of which the first four images, have used rgb color space and remaining four have used $l_1l_2l_3$ color space.

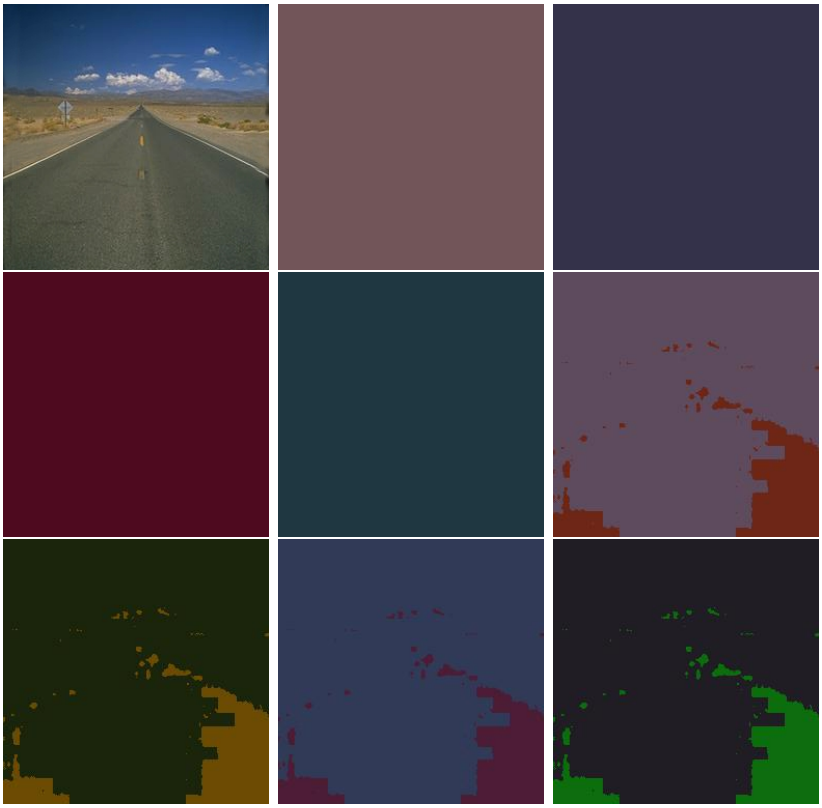

Figure 15: Original image (art255) followed by the eight output images from IBL's version of the GCGP, GCAP, ACGP, and ACAP models out of which the first four images, have used rgb color space and remaining four have used $l_1 l_2 l_3$ color space.

Chapter 4

Conclusion

The main motivation of this thesis is to develop sophisticated algorithms for image segmentation. We employed three approaches for integrating spatial information to FMM. In first two approaches, the spatial information is incorporated by using indirect information of M as prior knowledge [16] and MRF [10]. In chapter two, we have implemented IDMM and GIDMM using these two methods. The ID and GID are chosen precisely because of their flexible property of approximating many shapes [17]. Also, in contrast to Gaussian, it can be skewed to the left or right. We have used BSD500 dataset for experimentation. It has been found that the proposed algorithms outperformed the GMM [16, 10]. Lastly, in third chapter we have used a third approach proposed [51] in which the authors have suggested the incorporation of traditional FMM with CP and PP mean templates. These methods ensure the integration of spatial information by using peer pixels information and thus, makes the FMM more robust to noise. We explained how the mean templates integrate spatial information by introducing the pixel's weight in mixture model estimation. We have implemented IDMM, GIDMM, and IBLMM versions of GCGP, GCAP, ACGP, and ACAP models. These Semi-Bounded FMM are chosen precisely because of their flexibility that allow to describe many shapes. We have used BSD500 and CVCL datasets for experimentation. It has been found that out of the proposed algorithms IBLMM outperformed the GIDMM and IDMM. Also, $l_1l_2l_3$ color space is far better than rgb and the traditional RGB color space. Future works could be devoted to the application of the proposed models and approaches for object detection and tracking as well as video segmentation.

Bibliography

[1] Bdiri, T., Bouguila, N. and Ziou, D., 2016. A statistical framework for online learning using adjustable model selection criteria. Engineering Applications of Artificial Intelligence, (49), pp. 19-42.

[2] Bdiri, T. and Bouguila, N., 2012. Positive vectors clustering using inverted dirichlet finite mixture models. Expert Systems with Applications, 39(2), pp. 1869-1882.

[3] Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H. and Hu, S.M., 2015. Global contrast based salient region detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(3), pp. 569-582.

[4] Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A.L., 2018. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, 40(4), pp. 834-848.

[5] Jackson, J.E., 1985. The User's Guide to Multidimensional Scaling.

[6] Hershkovitch, T. and Riklin-Raviv, T., 2018. Model-dependent uncertainty estimation of medical image segmentation. IEEE 15th International Symposium on Biomedical Imaging, pp. 1373-1376.

[7] Lingappaiah, G.S., 1976. On the generalised inverted dirichlet distribution. Demostratio Mathematica, 9(3), pp. 423-433.

[8] McLachlan, G. and Krishnan, T., 2007. The EM algorithm and extensions.

[9] Moon, T.K., 1996. The expectation-maximization algorithm. IEEE Signal processing magazine, 13(6), pp. 47-60.

[10] Nguyen, T.M. and Wu, Q.J., 2013. Fast and robust spatially constrained Gaussian mixture model for image segmentation. IEEE transactions on circuits and systems for video technology, 23(4), pp. 621-635.

[11] Pal, N.R. and Pal, S.K., 1993. A review on image segmentation techniques. Pattern recognition, 26(9), pp. 1277-1294.

[12] Price, S.R., Price, S.R., Price, C.D. and Blount, C.B., 2018. Pre-screener for automatic detection of road damage in SAR imagery via advanced image processing techniques. In Pattern Recognition and Tracking, International Society for Optics and Photonics, (10649), pp. 1064913.

[13] Saito, S., Li, T. and Li, H., 2016. Real-time facial segmentation and performance capture from rgb input. In European Conference on Computer Vision, pp. 244-261.

[14] Sanjay-Gopal, S. and Hebert, T.J., 1998. Bayesian pixel classification using spatially variant finite mixtures and the generalized EM algorithm. IEEE Transactions on Image Processing, 7(7), pp. 1014-1028.

[15] Shi, J. and Malik, J., 2000. Normalized cuts and image segmentation. Departmental Papers, pp.107.

[16] Yang, X. and Krishnan, S.M., 2004. Image segmentation using finite mixtures and spatial information. Image and Vision Computing, 22(9), pp. 735-745.

[17] Ghorbel, M., 2009. On the inverted dirichlet distribution. Communications in Statistics-Theory and Methods, 39(1), pp. 21-37.

[18] Singh, J.P., 2017. Proportional Data Modeling using Unsupervised Learning and Applications, Concordia University.

[19] Sefidpour, A. and Bouguila, N., 2011. Spatial Finite Non-gaussian Mixture for Color Image Segmentation. In International Conference on Neural Information Processing, pp. 514-521.

[20] Caselles, V., Kimmel, R. and Sapiro, G., 1997. Geodesic active contours. International journal of computer vision, 22(1), pp. 61-79.

[21] Fabijańska, A. and Gocławski, J., 2015. The segmentation of 3D images using the random walking technique on a randomly created image adjacency graph. IEEE Transactions on Image Processing, 24(2), pp. 524-537.

[22] Kass, M., Witkin, A. and Terzopoulos, D., 1988. Snakes: Active contour models. International journal of computer vision, 1(4), pp. 321-331.

[23] Li, C., Kao, C.Y., Gore, J.C. and Ding, Z., 2007. Implicit active contours driven by local binary fitting energy. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-7.

[24] Li, C., Xu, C., Gui, C. and Fox, M.D., 2005. Level set evolution without re-initialization: a new variational formulation. IEEE computer society conference on computer vision and pattern recognition, (1), pp. 430-436.

[25] Ren, Z., 2015. Adaptive active contour model driven by fractional order fitting energy. Signal Processing, (117), pp. 138-150.

[26] Wang, X., Shan, J., Niu, Y., Tan, L. and Zhang, S.X., 2014. Enhanced distance regularization for re-initialization free level set evolution with application to image segmentation, Neurocomputing, (141), pp. 223-235.

[27] Wu, J., Zhao, Y., Zhu, J.Y., Luo, S. and Tu, Z., 2014. A sweeping line multiple instance learning paradigm for interactive image segmentation. IEEE Conference on Computer Vision and Pattern Recognition, pp. 256-263.

[28] Ji, Z., Xia, Y., Sun, Q., Chen, Q., Xia, D. and Feng, D.D., 2012. Fuzzy local Gaussian mixture model for brain MR image segmentation. IEEE Transactions on Information Technology in Biomedicine, 16(3), pp. 339-347.

[29] Boudaren, M.E.Y., An, L. and Pieczynski, W., 2016. Unsupervised segmentation of SAR images using Gaussian mixture-hidden evidential Markov fields. IEEE Geoscience and Remote Sensing Letters, 13(12), pp. 1865-1869.

[30] Xia, Y., Ji, Z. and Zhang, Y., 2016. Brain MRI image segmentation based on learning local variational Gaussian mixture models. Neurocomputing, (204), pp. 189-197.

[31] Bishop, C.M., 2006. Pattern recognition and machine learning, springer.

[32] Shang, R., Tian, P., Jiao, L., Stolkin, R., Feng, J., Hou, B. and Zhang, X., 2016. A spatial fuzzy clustering algorithm with kernel metric based on immune clone for SAR image segmentation. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 9(4), pp. 1640-1652.

[33] Ben-Dor, A., Shamir, R. and Yakhini, Z., 1999. Clustering gene expression patterns. Journal of computational biology, 6(3), pp. 281-297.

[34] Liu, G., Zhang, Y. and Wang, A., 2015. Incorporating adaptive local information into fuzzy clustering for image segmentation. IEEE transactions on image processing, 24(11), pp. 3990-4000.

[35] Li, G., Chen, X., Shi, F., Zhu, W., Tian, J. and Xiang, D., 2015. Automatic liver segmentation based on shape constraints and deformable graph cut in CT images. IEEE Transactions on Image Processing, 24(12), pp. 5315-5329.

[36] Dai, S., Lu, K., Dong, J., Zhang, Y. and Chen, Y., 2015. A novel approach of lung segmentation on chest CT images using graph cuts. Neurocomputing, (168), pp. 799-807.

[37] Salehi, S.S.M., Erdogmus, D. and Gholipour, A., 2017. Auto-context convolutional neural network (auto-net) for brain extraction in magnetic resonance imaging. IEEE transactions on medical imaging, 36(11), pp. 2319-2330.

[38] Orlando, J.I., Prokofyeva, E. and Blaschko, M.B., 2017. A discriminatively trained fully connected conditional random field model for blood vessel segmentation in fundus images. IEEE transactions on Biomedical Engineering, 64(1), pp. 16-27.

[39] Duan, Y., Liu, F., Jiao, L., Zhao, P. and Zhang, L., 2017. SAR Image segmentation based on convolutional-wavelet neural network and markov random field. Pattern Recognition, (64), pp. 255-267.

[40] Zhang, K., Zhang, L., Song, H. and Zhou, W., 2010. Active contours with selective local or global segmentation: a new formulation and level set method. Image and Vision computing, 28(4), pp. 668-676.

[41] Han, B. and Wu, Y., 2017. A novel active contour model based on modified symmetric cross entropy for remote sensing river image segmentation. Pattern Recognition, (67), pp. 396-409.

[42] Chen, Y.T., 2017. A novel approach to segmentation and measurement of medical image using level set methods. Magnetic resonance imaging, (39), pp. 175-193.

[43] Zhang, Y.J., 1996. A survey on evaluation methods for image segmentation. Pattern recognition, 29(8), pp. 1335-1346.

[44] McLachlan, G., Peel., D., 2000. Finite mixture models.

[45] Bouguila, N., 2011. Count data modeling and classification using finite mixtures of distributions. IEEE Transactions on Neural Networks, 22(2), pp. 186-198.

[46] Fan, W., Bouguila, N. and Ziou, D., 2012. Variational learning for finite Dirichlet mixture models and applications. IEEE transactions on neural networks and learning systems, 23(5), pp. 762-774.

[47] Yuksel, S.E., Wilson, J.N. and Gader, P.D., 2012. Twenty years of mixture of experts. IEEE transactions on neural networks and learning systems, 23(8), pp. 1177-1193.

[48] Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H. and Hu, S.M., 2015. Global contrast based salient region detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37(3), pp. 569-582.

[49] Zhang, H., Wu, Q.J. and Nguyen, T.M., 2013, May. Image segmentation by a robust modified gaussian mixture model. IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1478-1482.

[50] Rabiner, L.R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2), pp. 257-286.

[51] Zhang, H., Wu, Q.J. and Nguyen, T.M., 2013. Incorporating mean template into finite mixture model for image segmentation. IEEE transactions on neural networks and learning systems, 24(2), pp. 328-335.

[52] Tiao, G.G. and Cuttman, I., 1965. The inverted Dirichlet distribution with applications. Journal of the American Statistical Association, 60(311), pp. 793-805.

[53] Fang, K.W., 2017. Symmetric Multivariate and Related Distributions, Chapman and Hall/CRC.

[54] Strehl, A. and Ghosh, J., 2002. Cluster ensembles, a knowledge reuse framework for combining multiple partitions. Journal of machine learning research, pp. 583-617.

[55] Vinh, N.X., Epps, J. and Bailey, J., 2009. Information theoretic measures for clusterings comparison: is a correction for chance necessary?.International conference on machine learning, pp. 1073-1080.

[56] Vinh, N.X., Epps, J. and Bailey, J., 2010. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. Journal of Machine Learning Research, (11), pp. 2837-2854.

[57] Yang, Z., Algesheimer, R. and Tessone, C.J., 2016. A comparative analysis of community detection algorithms on artificial networks. Scientific reports, (6).

[58] Rosenberg, A. and Hirschberg, J., 2007. V-measure: A conditional entropy-based external cluster evaluation measure. Joint conference on empirical methods in natural language processing and computational natural language learning.

[59] Wanner, F., Stoffel, A., Jäckle, D., Kwon, B.C., Weiler, A., Keim, D.A., Isaacs, K.E., Giménez, A., Jusufi, I. and Gamblin, T., 2014. State-of-the-art report of visual analysis for event detection in text data streams. In Computer graphics forum 3(33), pp. 1-15.

[60] Caliński, T. and Harabasz, J., 1974. A dendrite method for cluster analysis. Communications in Statistics-theory and Methods, 3(1), pp. 1-27.

[61] Tan, P.N. and Steinbach, M., Kumar, V., 2005. Introduction to Data Mining.

[62] Jaccard, P., 1901. Étude comparative de la distribution florale dans une portion des Alpes et des Jura. Bull Soc Vaudoise Sci Nat, (37), pp. 547-579.

[63] Jaccard, P., 1912. The distribution of the flora in the alpine zone, 11(2), pp. 37-50.

[64] Bdiri, T., Bouguila, N. and Ziou, D., 2016. Variational Bayesian inference for infinite generalized inverted Dirichlet mixtures with feature selection and its application to clustering. Applied Intelligence, 44(3), pp. 507-525.

[65] Panahi, R. and Gholampour, I., 2017. Accurate detection and recognition of dirty vehicle plate numbers for high-speed applications. IEEE Transactions on intelligent transportation systems, 18(4), pp. 767-779.

[66] Bdiri, T., Bouguila, N. and Ziou, D., 2013, November. Visual scenes categorization using a flexible hierarchical mixture model supporting users ontology. IEEE 25th International Conference on Tools with Artificial Intelligence, pp. 262-267.

[67] Al Mashrgy, M., Bdiri, T. and Bouguila, N., 2014. Robust simultaneous positive data clustering and unsupervised feature selection using generalized inverted dirichlet mixture models. Knowledge-Based Systems, (59), pp. 182-195.

[68] Bdiri, T., Bouguila, N. and Ziou, D., 2014. Object clustering and recognition using multi-finite mixtures for semantic classes and hierarchy modeling. Expert Systems with Applications, 41(4), pp. 1218-1235.

[69] Graybill, F.A., 1983. Matrices with applications in statistics.

[70] Gevers, T. and Smeulders, A.W., 1999. Color-based object recognition. Pattern recognition, 32(3), pp. 453-464.

[71] Martin, D., Fowlkes, C., Tal, D. and Malik, J., 2002. A database of human segmented natural images and its application to evaluating segmentation algorithms. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition and Measuring Ecological Statistics, (2), pp. 416.

[72] Oliva, A. and Torralba, A., 2001. Modeling the shape of the scene: A holistic representation of the spatial envelope. International journal of computer vision, 42(3), pp. 145-175.

[73] Hu, C., Fan, W., Du, J.X. and Bouguila, N., 2019. A novel statistical approach for clustering positive data based on finite inverted Beta-Liouville mixture models. Neurocomputing, pp. 110-123.

[74] Bouguila, N., 2007. Spatial color image databases summarization. IEEE International Conference on Acoustics, Speech and Signal Processing, (1), pp. 953

[75] Bouguila, N. and Ziou, D., 2005. Using unsupervised learning of a finite Dirichlet mixture model to improve pattern recognition applications. Pattern Recognition Letters, 26(12), pp.1916-1925.