Nonparametric Bayesian Models Based on Asymmetric Gaussian Distributions

Ziyang Song

A Thesis
in
The Concordia Institute
for
Information Systems Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Master of Applied Science (Quality Systems Engineering)
Concordia University
Montreal, Quebec, Canada

June 2020

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Ziyang Song**

Entitled: **Nonparametric Bayesian Models Based on Asymmetric Gaussian Distributions**

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

Master of Applied Science (Quality Systems Engineering)

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. Chadi Assi _____ Chair

Dr. Nizar Bouguila _____ Supervisor

Dr. Farnoosh Naderkhani _____ CIISE Examiner

Dr. Daria Terekhov _____ External Examiner

Approved _____

Dr. Chadi Assi Graduate Program Director

2019 / 11 / 08 _____

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

Abstract

Nonparametric Bayesian Models Based on Asymmetric Gaussian Distributions

Ziyang Song

Data clustering is a fundamental unsupervised learning approach that impacts several domains such as data mining, computer vision, information retrieval, and pattern recognition. Various clustering techniques have been introduced over the years to discover the patterns. Mixture model is one of the most promising techniques for clustering. The design of mixture models hence involves finding the appropriate parameters and estimating the number of clusters in the data.

The Gaussian mixture model has especially shown good results to tackle this problem. However, the Gaussian assumption is not ideal for modeling asymmetrical data. For achieving an accurate approximation, I investigate the asymmetric Gaussian distribution which is capable of modeling asymmetric data.

A prevalent challenge researchers face when applying mixture models is the correct identification of the adequate number of mixture components to model the data at hand. Hence, in this thesis, I propose statistical algorithms based on asymmetric Gaussian mixture models. I also present novel Bayesian inference frameworks to estimate parameters and learn model structure.

Here, I thoroughly investigate the Bayesian inference framework, including Markov chain Monte Carlo and variational inference approaches, to learn appropriate model structure and precisely estimate parameters. I also incorporate feature selection within the frameworks to choose relevant features set and avoid noisy influence from uninformative features. Furthermore, I investigate nonparametric hierarchical models by introducing Dirichlet process and Pitman-Yor process.

Acknowledgments

I would like to express my deepest gratitude to my parents who support me purse dream and study aboard. For the past one year, I did not have opportunity to come back my hometown and meet my parents which make me feel apologetic. Therefore, I want to deliver my appreciation for their selfless sacrifice and contribution.

I also would like to express my profound appreciation to my supervisor Prof. Nizar Bouguila, who offer me opportunity to study at Concordia University although I did not have research experience before. Without his insightful guidance, none of my research achievements could become accomplished. He also help me widen my view of machine learning research with his vast knowledge on the subject. Thus, I want to thanks for his wit, patience and support which inspired me to move forward.

I also want to express my gratitude to my cooperator Dr. Wentao Fan and Miss. Samr Ali who help me during the last year. They contribute so much in my works as co-author and help me bulid research abilities. Besides, I also want to thanks for Mr. Xavier Sumba Toral and Mr. Kamal Maanicshah since they provide me so much in my master period. Without the technical advises and aids, the completion of my research projects and thesis would have been very hard. Besides, I am extremely fortunate to to be working with many fantastic lab mates who have made this one year unforgettable.

A special thanks to Mr. Hao Wu, and Miss. Jiaoyang Dong, and Miss. Qianwen Huang who believe I have potential in research area and always encourage me to continue research when I feel frustrated of technical difficulties. I could always find courage and motivation from my precious friends. I also want to mention my friends, including Mr. Zhikun Chen and Miss. Hui Cai, who spend to share their valuable experience and discuss research topics with me. I am also grateful to Miss. Sainan Zhao and Mr. Haolong Wu who help me deal with my files of undergraduate which allows me to stay at Montreal and focus on my research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Clustering is a common unsupervised learning methodology for data analysis and has been widely applied to uncover the structure of observations representing distinct groups. A mixture model, one of most prevalent statistical clustering techniques, divides data into a collection of homogeneous groups. This can be modeled by a density and the overall model is represented by a weighted sum of a number of components. The Gaussian distribution assumption has been extensively applied in many fields because it provides interpretable results and is easily generalized to new tasks [1]. However, it is not always an adequate choice since the shape of the distribution of the observations may not be strictly symmetric [2]. Indeed, this is the case especially for natural images. For achieving an accurate approximation, I investigate the asymmetric Gaussian distribution (AGD) which is capable of modeling asymmetric data: AGD has left and right standard deviation parameters to better control the shape of distribution to reflect the asymmetry of data [3].

Parameter estimation is one of the challenges required for the use of latent variable models. Various algorithms have been studied to achieve this objective. The expectation maximization (EM) algorithm is one of the well-known methods to estimate the parameters of density function. Nevertheless, the EM algorithm as a deterministic approach is not guaranteed to converge to a global optimal due to vulnerability to initialization conditions and overfitting problems. Practised solutions include Bayesian inference techniques which are extensively discussed in approximating intractable distributions [4]. It provides a robust theoretical framework to employ clustering algorithms. As such, Markov Chain

Monte Carlo (MCMC) is one of the most prevalent methods to estimate parameters because it is capable of precisely approximating a given distribution which lead to remarkable performance [4] [5]. As another prevalent inference approach, variational inference, could approximate the ideal distribution requiring relatively smaller amount of computational time and resources compared with MCMC algorithms [6].

Several studies have been devoted to the automatic selection of the mixture components number which best depicts the observations. Mixture models which allow the number of components to grow to infinity as required to fit the data can be viewed as nonparametric models [7]. I am interested in Bayesian nonparametric approaches for modeling, especially those based on the Dirichlet process (DP). The DP allows unbounded growth of the number of mixture components as necessary to fit the data, where the individual variables still follow certain parametric distributions. Through DP based mixture models, it is possible to determine the correct number of components and to extend a finite mixture model to an infinite one. Thereby, I propose DP based infinite mixture model with Bayesian inference framework.

On the other hand, theoretically, the higher the number of features used to represent a given dataset, the better the clustering algorithm is expected to perform. In practice, however, some features can be noisy, redundant, or uninformative. Thus, they can hinder the clustering performance [8]. The presence of many irrelevant features introduces a bias resulting in unreliable homogeneity measures. Feature selection is the process of reducing the number of collected features to a subset of relevant features. Hence, it increases the performance of models by eliminating noise in the data, improving model interpretation, and decreasing the risk of overfitting [9]. Here, I consider a feature saliency approach which consider feature selection as parameter estimation problem and recast probability distribution as dependent and independent distributions [10]. Feature saliency is added as new parameter to the conditional distribution of the mixture model and used to find clusters embedded in feature subspace. Since feature saliency represents the probability of belonging to a mixture-dependent distribution, it can be interpreted as the probability that a feature is relevant.

In addition, I investigate the possibility of extending the proposed model to hierarchical nonparameteric model which allows it to model grouped data with shared clusters. Within the same group, each observation is drawn independently from a mixture model, and the number of observations within each group may be different. The dependencies among

groups are caused by the assumption that the mixture models in different groups may share mixture components. Under the settings of hierarchical modeling, parameters are shared among groups, and the randomness of the parameters induces dependencies among different groups. Hierarchical Bayesian models have been an attractive research topic and been successfully applied in various fields such as language modeling, image segmentation, etc [11]. A sound alternative to DP is Pitman-Yor process (PYP) which can be viewed as a generalization to the DP prior for nonparametric Bayesian modeling [12]. I further extend it to hierarchical Dirichlet process mixture and hierarchical Pitman-Yor process mixture.

1.2 Contributions

The main objective of the thesis is to demonstrate the advantage of asymmetric Gaussian distribution and investigate thoroughly Bayesian inference framework. I also investigate the extension of proposed model using nonparametric Bayesian approaches and, probabilistic feature selection. The contributions of the thesis are listed as follows:

- ☞ **Data Clustering using asymmetric Gaussian mixture**

  I propose asymmetric Gaussian mixture which could accurately capture asymmetry structure of data. I also extend the mixture model with nonparametric prior to adjust components number according to the structure of the dataset. Furthermore, I also introduce a feature selection framework for infinite mixture model.

- ☞ **Introduction of hierarchical Bayesian learning for the proposed model**

  In this thesis, I consider MCMC and variational inference approaches. The MCMC methodology includes Gibbs sampling and Metropolis-Hastings algorithm and their efficiency is validated in Chapter 2 and Chapter 3. The variational inference based learning framework includes mean field variational inference and black box variational inference. Since black box variational inference is gradient descent approach, it has to control the variance to ensure the convergence. The effectiveness of the variational inference learning framework is evaluated in Chapter 3, Chapter 4, and Chapter 5 on several models and different applications.

☞ **Hierarchical nonparameteric Bayesian model for modelling grouped data**

I introduce the Dirichlet process and Pitman-Yor process, which are popular nonparameteric priors, to form hierarchical infinite mixture models. The parameters and structures of hierarchical infinite mixtures are learned by variational inference framework and validated on grouped datasets.

These contributions have been published in the International Conference on Image Analysis and Recognition (ICIAR 2019) and IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2019). The contributions have been submitted and are under review in Soft Computing, IET Image Processing and, Signal Processing journals.

1.3 Thesis Overview

- ❑ Chapter 1 introduces the concept of clustering and a brief overview of various concepts related to the thesis. I also explain clearly the motivations behind the conducted research work.
- ❑ In Chapter 2, I explain in detail the MCMC learning framework for infinite asymmetric Gaussian mixture model. The efficiency of the proposed model is validated by the challenging task of background subtraction and evaluated on several datasets.
- ❑ In chapter 3, I integrate a simultaneous feature selection algorithm within the proposed infinite asymmetric Gaussian mixture. The MCMC based Bayesian inference framework is presented to solve parameter estimation and structure learning problems. The experiments with various applications including dynamic textures clustering and scene categorization are described in detail.
- ❑ Chapter 4 describes the finite asymmetric Gaussian mixture model with variational inference framework which includes mean-field variational inference and black box variational inference. The model has been tested with challenging application.
- ❑ In chapter 5, I integrate simultaneous feature selection algorithm within infinite asymmetric Gaussian mixture model. The Bayesian inference framework consists of mean-field inference and black box variational inference. Since the gradient ascent method lead to high variance, I propose variational reduction technique and reparameterization trick to control the variance and ensure convergence.

- ❏ Chapter 6 describes the hierarchical Bayesian nonparametric model and statistical inference framework which consists of several variational inference methods. Specifically, the Dirichlet process and Pitman-Yor process are considered in the research. The models have been tested via image clustering.
- ❏ In conclusion, I briefly summarize the contributions.

Chapter 2

Bayesian Learning of Infinite Asymmetric Gaussian Mixture Models

In this chapter, I introduce an infinite asymmetric mixture model (IAGM) which provides a better fit for asymmetric shaped observations. It estimates the parameters and chooses the optimal number of components through the employment of Bayesian learning and the extension of the finite asymmetric Gaussian mixture (AGM) to infinity. Furthermore, I demonstrate the efficiency of the model by utilizing it for the background subtraction task. the achieved results are comparable to three different methods in terms of precision, and superior in terms of the recall metric.

2.1 Infinite Asymmetric Gaussian Mixture

2.1.1 Finite Asymmetric Gaussian Mixture

The definition of a finite AGM model with respect to observations, weights and probability density is illustrated as follows:

$$p\big(X \mid \Theta\big) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j p\big(X_i \mid \xi_j\big) \tag{2.1}$$

where $X = (X_1, ..., X_N)$ is the N observations dataset, each observation $X_i = (X_{i1}, \ldots, X_{iD})$ could be represented as D-dimensional random variable. $M \geq 1$ is the number of mixture components, $\Theta = (\pi_1, ..., \pi_M, \xi_1, \ldots, \xi_M)$ defines the complete set of parameters fully characterizing the mixture model where $\pi = (\pi_1, \ldots, \pi_M)$ are the mixing weights

which must be positive and sum to one, and ξ_j is the set of parameters of mixture component j.

The AGD for each component j, the probability density of each observation X_i $p(X_i \mid \xi_j)$ is then given by:

$$p(X_i \mid \xi_j) \propto \prod_{k=1}^{D} \frac{1}{(S_{ljk})^{-\frac{1}{2}} + (S_{rjk})^{-\frac{1}{2}}} \times \begin{cases} \exp\big[-\frac{S_{ljk}(X_{ik}-\mu_{jk})^2}{2}\big] & \text{if} X_{ik} < \mu_{jk} \\ \exp\big[-\frac{S_{rjk}(X_{ik}-\mu_{jk})^2}{2}\big] & \text{if} X_{ik} \geq \mu_{jk} \end{cases} \tag{2.2}$$

where $\xi_j = (\mu_j, S_{lj}, S_{rj})$ is the parameter set for AGD with $\mu_j = (\mu_{j1}, \ldots, \mu_{jd})$, $S_{lj} = (S_{lj1}, \ldots, S_{ljd})$ and $S_{rj} = (S_{rj1}, \ldots, S_{rjd})$. μ_{jk}, S_{ljk} and S_{rjk} are the mean, the left precision and the right precision of the kth dimensional distribution. Here, I assume independence so that the covariance matrix of X_i is diagonal matrix. This assumption allows us to avoid costly computation during deployment.

I introduce the latent indicator variables $Z = (Z_1, \ldots, Z_N)$, Z_i for each observation X_i to indicate which component it belongs to. $Z_i = (Z_{i1}, \ldots, Z_{iM})$ where hidden label Z_{ij} is assigned as 1 if X_i belongs to component j otherwise will be set to 0. The likelihood function is then defined by:

$$p(X \mid Z, \Theta) = \prod_{i=1}^{N} p(X_i \mid \xi_j)^{Z_{ij}} \tag{2.3}$$

Given the mixing weights π, for $j = 1, \ldots, M$, the indicators Z are given Multinomial prior:

$$p(Z \mid \pi) = \text{Multi}(\pi) = \prod_{j=1}^{M} \pi_j^{n_j} \tag{2.4}$$

where n_j is the number of observations that are associated with component j. The mixing weights are considered to follow symmetric Dirichlet distribution with a concentration parameter α/M:

$$p(\pi \mid \alpha) \sim \text{Dir}(\frac{\alpha}{M}, ..., \frac{\alpha}{M}) = \frac{\Gamma(\alpha)}{\Gamma(\frac{\alpha}{M})^M} \prod_{j=1}^{M} \pi_j^{\frac{\alpha}{M}-1} \tag{2.5}$$

It then integrates out the mixing weights π to obtain the prior of Z:

$$p(Z \mid \alpha) = \int p(Z \mid \pi) p(\pi \mid \alpha) d\pi = \frac{\Gamma(\alpha)}{\Gamma(N+\alpha)} \prod_{j=1}^{M} \frac{\Gamma(\frac{\alpha}{M} + n_j)}{\Gamma(\frac{\alpha}{M})} \tag{2.6}$$

The conditional prior for a single indicator is then denoted by:

$$p(Z_{ij} = 1 \mid \alpha, Z_{-i}) = \frac{n_{-i,j} + \frac{\alpha}{M}}{N - 1 + \alpha} \tag{2.7}$$

where the subscript $-i$ defines all indexes except i, $Z_{-i} = (Z_1, \ldots, Z_{i-1}, Z_{i+1}, \ldots, Z_N)$, $N_{-i,j}$ is the number of observations excluding X_i allocated to the component j.

Next, I extend the model to infinity by updating the posterior of indicators in Eq. (2.7) with $M \to \infty$:

$$p(Z_{ij} = 1 \mid \alpha, Z_{-i}) = \begin{cases} \frac{n_{-i,j}}{N-1+\alpha}, & if\, n_{-i,j} > 0 \\ \frac{\alpha}{N-1+\alpha}, & if\, n_{-i,j} = 0 \end{cases} \tag{2.8}$$

where $n_{-i,j} > 0$ occurs only when component j is represented. Thus, an observation X_i is associated with an existing component by a certain probability proportional to the number of observations already allocated to this component; while a new (when unrepresented) component is proportional to α and N. Given the conditional prior in Eq. (2.7), the conditional posterior is obtained by multiplying the prior with Eq. (2.3) resulting in:

$$p(Z_{ij} = 1 \mid ...) = \begin{cases} \frac{n_{-i,j}}{N-1+\alpha} \prod_{k=1}^{d} p\big(X_{ik} \mid \xi_{jk}\big), & if\, n_{-i,j} > 0 \\ \frac{\alpha}{N-1+\alpha} \int p\big(X_i \mid \xi_j\big) p\big(\xi_j \mid \lambda, r, \beta_l, \beta_r, w_l, w_r\big) d\xi_j, & if\, n_{-i,j} = 0 \end{cases} \tag{2.9}$$

where the hyperparameter α is defined by an inverse Gamma prior with shape a and mean b chosen as follows:

$$p(\alpha^{-1}) \propto \alpha^{-\frac{3}{2}} \exp(-\frac{1}{2\alpha}) \tag{2.10}$$

Given the likelihood of α in Eq. (2.6), the posterior is then:

$$p(\alpha \mid M, N) \propto \frac{\alpha^{\frac{M-3}{2}} \exp(-\frac{1}{2\alpha}) \Gamma(\alpha)}{\Gamma(N + \alpha)} \tag{2.11}$$

The conditional posterior for α depends only on number of observations, N, and the number of components, M. The logarithmic representation of posteriors is log-concave, so it can sample α by using the Adaptive Rejection Sampling (ARS) method [13].

2.1.2 Bayesian Learning

In this section, I describe an MCMC-based approach for learning the proposed IAGM model. The means of the components μ_{jk} are given Gaussian prior with hyperparameters λ and r as follows:

$$p(\mu_{jk} \mid \lambda, r) \sim \mathcal{N}(\lambda,\, r^{-1})\,. \tag{2.12}$$

where the mean, λ, and precision, r, hyperparameters are common to all components in a specific dimension. λ is given Gaussian prior with mean e and variance f, and r is given Gaussian prior and inverse Gamma prior with shape parameter g and mean parameter h respectively:

$$p(\lambda) \sim \mathcal{N}(e, f) \tag{2.13}$$

$$p(r) \sim \Gamma(g, h) \tag{2.14}$$

where e and f will be μ_y and σ_y^2, the mean and variance of the observations which are used for the parameters of the Gaussian prior. The Gamma prior uses constant values 1 as shape g and σ_y^2 as mean h.

The conditional posterior for the mean μ_{jk} is then computed by multiplying the likelihood from Eq. (2.3) by the prior Eq. (2.12) as follows:

$$p(\mu_{jk} \mid X_k, S_{ljk}, S_{rjk}, \lambda, r) \propto \mathcal{N}(\frac{S_{ljk} \sum_{i:X_{ik}<\mu_{jk}}^{n} X_{ik} + s_{rjk} \sum_{i:X_{ik}\geq\mu_{jk}}^{n} X_{ik} + r\lambda}{r + ps_{ljk} + (n_j - p)s_{rjk}}, \frac{1}{r + ps_{ljk} + (n_j - p)s_{rjk})} \tag{2.15}$$

where X_k is the kth dimensional observations allocated to component j. n_j is the count number of observations X_k and p is the count number of observations X_k which are less than μ_{jk}. For the hyperparmeters λ and r, it uses hyperposteriors to update parameters. Eq. (2.12) plays the role of the likelihood function. As such, it combines Eq. (2.12), Eq. (2.13) and Eq. (2.14) to obtain the following posteriors:

$$p(\lambda \mid \mu_{1k}, \ldots, \mu_{Mk}, r) \propto \mathcal{N}(\frac{\mu_y \sigma_y^{-2} + r \sum_{j=1}^{M} \mu_{jk}}{\sigma_y^{-2} + Mr}, \frac{1}{\sigma_y^{-2} + Mr}) \tag{2.16}$$

$$p(r \mid \mu_{1k}, \ldots, \mu_{Mk}, \lambda) \propto \Gamma(M+1, \frac{M+1}{\sigma_y^2 + \sum_{j=1}^{M}(\mu_{jk} - \lambda)^2}) \tag{2.17}$$

The component precisions S_{ljk} and S_{rjk} are given Gamma priors with common hyperparameters β and w^{-1} as follows:

$$p(S_{ljk} \mid \beta, w) \sim \Gamma(\beta, w^{-1}), \quad p(S_{rjk} \mid \beta, w) \sim \Gamma(\beta, w^{-1}) \tag{2.18}$$

where β is given inverse Gamma prior with shape parameter s and mean parameter t, and w is given Gamma prior with u and v:

$$p(\beta^{-1}) \sim \Gamma(s, t) \tag{2.19}$$

$$p(w) \sim \Gamma(u, v) \tag{2.20}$$

where I set both of mean and shape parameters of hyperprior β as constant value 1, and mean and shape parameters of hyperprior w are defined as 1 and σ_y^2 respectively. The conditional posterior distribution for S_{ljk} and S_{rjk} are obtained by multiplying the likelihood from Eq. (2.3) by the prior Eq. (2.18) as follows:

$$\begin{aligned} p(S_{ljk} \mid X_k, \mu_{jk}, S_{rjk}, \beta, w) \propto (S_{ljk}^{-\frac{1}{2}} + S_{ljk}^{-\frac{1}{2}}) S_{ljk}^{\frac{\beta}{2}-1} \\ \exp\big[- \frac{S_{ljk} \sum_{i:X_{ik}<\mu_{jk}}^{n} (x_{ik} - \mu_{jk})^2}{2} - \frac{w\beta S_{ljk}}{2} \big] \end{aligned} \tag{2.21}$$

Random samples of posteriors can be drawn by using the MCMC method. In this chapter, I use Metropolis-Hastings algorithm to sample precision parameters. For the hyperparameters β and w, Eq. (2.18) plays the role of the likelihood function. Combining Eq. (2.12), Eq. (2.19), and Eq. (2.20), I obtain the following posteriors:

$$p(\beta_l \mid S_{l1k}, \ldots, S_{lMk}, w_l) \propto \Gamma(\frac{\beta_l}{2})^{-M} \exp(-\frac{1}{2\beta_l})(\frac{\beta_l}{2})^{\frac{M\beta_l - 3}{2}} \prod_{j=1}^{M} (w_l S_{ljk})^{\frac{\beta_l}{2}} \exp(-\frac{\beta_l w_l s_{ljk}}{2}) \tag{2.22}$$

$$p(w_l \mid S_{l1k}, \ldots, S_{lMk}, \beta_l) \propto \Gamma(M\beta_l + 1, \frac{M\beta_l + 1}{\sigma_y^{-2} + \beta_l \sum_{j=1}^{M} S_{ljk}}) \tag{2.23}$$

where I only show the left side of β and w parameters with similar posteriors for the right side parameters. The posterior distribution of precision β is not a standard form, but its logarithmic posterior is log-concave. Therefore, it can sample from the distribution for $\log(\beta)$ using ARS technique and transform the resultant to get values for β.

The proposed complete algorithm can be summarized by the following:

Algorithm 1 Infinite Asymmetric Gaussian Mixture

1: **procedure**
2: Initialize assignments and parameters.
3: *loop*:
4: Update mixture parameters μ_j, S_{ljk} and S_{rjk} from posteriors in Eq. (2.15) and Eq. (2.21).
5: Update hyperparameters λ, r, β, w and DP concentration parameter α from posteriors in Eq. (2.16), (2.17), (2.22), (2.23) and (2.11).
6: Update the indicators conditioned on the other indicators and the hyperparameters from Eq. (2.9).
7: The convergence criteria is reached when the difference of the current value of joint posteriors and the previous value is less than 10^{-4}. Otherwise, repeat above procedures until convergence.

2.2 Experimental Setup

2.2.1 Background Subtraction Application

In this section, I employ the proposed IAGM model for video background subtraction with a pixel-level evaluation approach as in [14]. The background subtraction methodology starts off by constructing the model using the proposed IAGM. After applying the learning algorithm for the model, it discriminates between the mixture components for the representation of foreground and background pixels for each of the new input frames.

Assume that each video frame has P pixels such that $\vec{X} = (X_1, \ldots, X_P)$ then each pixel X is assigned as a foreground or background according to the trained IAGM model $p\big(X \mid \Theta\big) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j p\big(X_i \mid \xi_j\big)$. Components that occur frequently, i.e. with high π value, and with a low standard deviation $S^{-\frac{1}{2}}$ are modeled as the background.

Accordingly, the value of $\pi_j/(||{S_{l_j}}^{-\frac{1}{2}}|| + ||{S_{r_j}}^{-\frac{1}{2}}||)$ is used to order the mixture components, where π_j is the mixing weight for component j, $||{S_{l_j}}^{-\frac{1}{2}}||$ and $||{S_{r_j}}^{-\frac{1}{2}}||$ are the

respective norms of left and right standard deviations of the jth component [14]. The first B number of components are chosen to model the background, with B estimated as:

$$B = \underset{b}{\operatorname{argmin}} \sum_{j=1}^{b} \pi_j > T \tag{2.24}$$

where T is a measure of the minimum proportion of the data that represents the background in the scene, and the rest of the components are defined as foreground components.

2.2.2 Results and Discussion

I apply the proposed algorithm to the Change Detection 2012 dataset [15]. The dataset consists of six categories with a total of 31 videos totaling 90,000 frames. Each of the categories (baseline, dynamic background, camera jitter, shadows, intermittent object motion, and thermal) contains around 4 to 6 different video sequences from low-resolution IP cameras.

In this chapter, I have selected five videos from the Change Detection dataset to evaluate the proposed methodology. I initialize the IAGM by incrementally increasing the threshold multiple times and choosing the optimum parameter setting. I adopt the threshold factor $T = 0.9$ in the method. I set the maximum component number for the algorithm as 9 and the standard deviation factor $K = 2$. Evaluation of the proposed IAGM can be observed in the confusion matrices in Figure. 2.1. Moreover, Figure. 2.2 shows visual results of the proposed method on sample frames in the Library and Street Light video sequences.

I also compare the results with three other methods from the literature. These include the Gaussian mixture model-based background subtraction algorithms by Sauffer et al. [14] and Zivkovic [16] as well as the finite asymmetric Gaussian mixture model by Elguebaly et al. [17]. I evaluate the performance of the algorithms in terms of the recall and the precision metrics.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2.25}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2.26}$$

Table 2.1: Experimental results for the background subtraction application.

	Stauffer	Zivkovic	Elguebaly	IAGM
Boulevard				
Recall	83.21%	79.77%	79.54%	84.72%
Precision	40.02%	43.79%	61.13%	55.80%
Abandoned Box				
Recall	45.74%	45.64%	45.18%	81.53%
Precision	65.52%	62.14%	67.41%	56.23%
Street Light				
Recall	32.25%	33.94%	30.33%	57.41%
Precision	89.16%	92.47%	97.56%	99.99%
Sofa				
Recall	51.62%	51.41%	59.90%	53.56%
Precision	85.92%	89.25%	92.52%	93.41%
Library				
Recall	28.00%	28.68%	31.33%	94.74%
Precision	84.76%	81.76%	94.66%	86.52%

where TP is the total number of true positives correctly identified by approaches, FN is the number of false negatives, and FP represents the number of false positives. The results can be seen in Table. 2.1.

As can be observed in Table. 2.1, the proposed IAGM mostly outperforms the other approaches in terms of the recall metric, while achieving comparable precision results. For instance, IAGM attains better recall results for the Street Light video sequence with a near perfect precision. This clearly demonstrates the effectiveness of the proposed model.

In particular, the approach detects more foreground pixels; most of which are clustered correctly. This ensures comparable precision results compared with the other algorithms. The method does not remarkably improve the precision metric due to the sensitivity of the proposed method to the change in environments. With higher number of detected foreground pixels, the approach shows significant improvement in the recall metric. This improvement is especially distinct for the Library video.

These improvements are due to the nature of the IAGM that is capable of accurately capturing the asymmetry of the observations. This higher flexibility of AGD allows the incorporation of the different shape distributions of objects. Furthermore, the extension to the infinite mixture using the DP with a Chinese restaurant process construction increases adaptability of the proposed model. Hence, I addressed both the parameter learning and the component number determination challenges. These advantages provide a more efficient

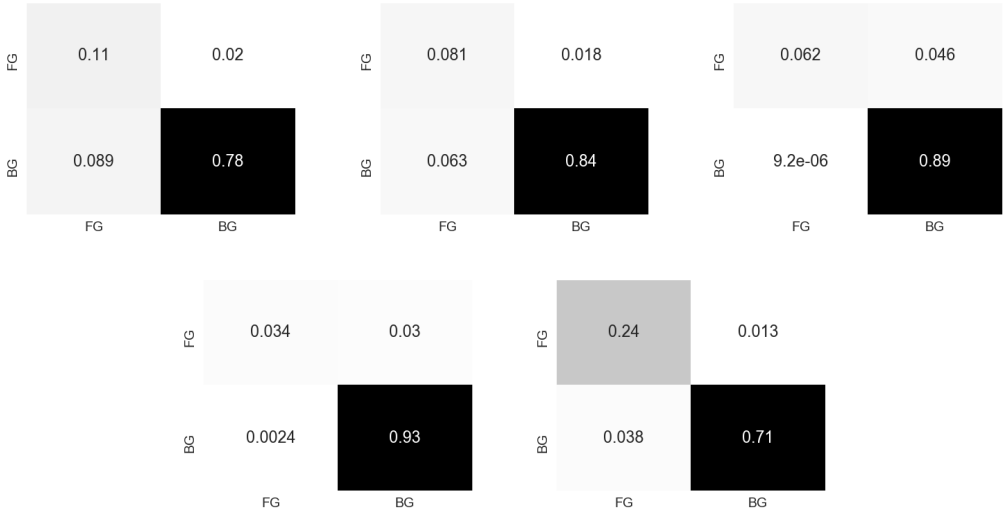


Figure 2.1: Confusion matrices of the proposed method employed for background subtraction on the boulevard (top left), abandoned box (top center), street light (top right), sofa (bottom left), and library (bottom right) videos where FG denotes the foreground and BG denotes the background.

Figure 2.2: A sample frame from Street Light (left) and Library (right) video sequences and the detected foreground object respectively.

model for background subtraction.

Chapter 3

Bayesian Learning for Infinite Asymmetric Gaussian Mixture with feature selection

In this chapter, I integrate infinite mixture model with a feature selection technique for the purpose of choosing the set of features that are most informative in order to construct an appropriate model in terms of clustering accuracy. I report results based on experiments that concern dynamic textures clustering as well as scene categorization. These show the merits of the developed approach.

3.1 Infinite Asymmetric Gaussian Mixture with Feature Selection

In this section, I incorporate IAGM model, which is proposed in chapter 2, with feature selection algorithm. I start by introducing the concept of feature saliency and represent the proposed model combined with feature selection.

3.1.1 Feature Saliency

In this section, I introduce the concept of feature saliency and consider the feature selection problem as a parameter estimation problem [10]. It is natural to consider that different features may have different weights for each of the mixture components. Thus, I define

feature saliency as the weight of feature importance.

It assumes that a feature is relevant if it follows a mixture-dependent distribution AGD. Otherwise, it may be modeled as a mixture-independent background distribution. In this chapter, I propose a Gaussian assumption for the background distribution. By introducing latent relevant indicator $\phi_i = (\phi_{i1}, \ldots, \phi_{iM})$ with $\phi_{ij} = (\phi_{ij1}, \ldots, \phi_{ijD})$, I could then infer if a given feature is relevant or not. The binary indicator $\phi_{ijk} = 1$ if feature k in observation X_i is relevant for component j, otherwise $\phi_{ijk} = 0$. Thus, it is possible to rewrite the probability density function as follows:

$$p(\mathcal{X} \mid \Theta, \xi^{irr}, \Phi) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j \prod_{k=1}^{D} \left[p(X_{ik} \mid \xi_{jk})^{\phi_k} p(X_{ik} \mid \xi_{jk}^{irr})^{1-\phi_k} \right] \tag{3.1}$$

where the $\xi^{irr} = (\xi_1^{irr}, \ldots, \xi_M^{irr})$ represents the set of parameters for background Gaussian distribution with $\xi_j^{irr} = (\mu_j^{irr}, (S_j^{irr})^{-1})$, $\mu_j = (\mu_{j1}, \ldots, \mu_{jD})$, $S_j = (S_{j1}, \ldots, S_{jD})$. μ_{jk}^{irr} and S_{jk}^{irr} represent the mean and precision for Gaussian distribution, respectively.

Feature saliency defined as $\rho = (\rho_1, \ldots, \rho_M)$ such that $\rho_j = (\rho_{j1}, \ldots, \rho_{jD})$. $\rho_{jk} = p(\phi_j = 1)$ represents the prior probability that the feature k is relevant in mixture component j. Thus, it could recast the likelihood function after introducing the feature saliency ρ. This can be denoted by:

$$p(X_i \mid \Theta_F) = \sum_{j=1}^{M} \pi_j \prod_{k=1}^{D} (\rho_{jk} p(X_{ik} \mid \xi_{jk}) + (1 - \rho_{jk}) p(X_{ik} \mid \xi_{jk}^{irr})) \tag{3.2}$$

where $\Theta_F = (\Theta, \rho, \xi^{irr})$ is the full set of parameters of the mixture model after introducing feature saliency. Eq. (3.2) offers sound generative interpretation for the model. First, the model selects the component j by sampling from a Multinomial distribution with mixing proportions $\pi = (\pi_1, \ldots, \pi_k)$. Then, for each feature dimension $k = 1, \ldots, D$, it follows a Bernoulli distribution with feature saliency ρ_{jk}; if successful, it uses the relevant mixture component $p(\ X_{ik} \mid \xi_{jk})$ to generate feature k; otherwise, the background component $p(X_{ik} \mid \xi_{jk}^{irr})$ will be used. Therefore, the model of previous chapter could be viewed as special case when all of the features are relevant.

The conditional posteriors of DP mixture could be rewritten after bringing feature

saliency into model as:

$$p(Z_i = j \mid ...) = \begin{cases} \frac{n_{-i,j}}{N-1+\alpha} \prod_{k=1}^{D} (\rho_{jk} p(X_{ik} \mid \xi_{jk}) + (1-\rho_{jk}) p(X_{ik} \mid \xi_{jk}^{irr})) & \text{if } n_{-i,j} > 0 \\ \frac{\alpha}{N-1+\alpha} \int p(\xi_j \mid \cdots) p(\xi_j^{irr} \mid \cdots) \times p(X_i \mid \xi_j) d\xi_j & \text{if } n_{-i,j} = 0 \end{cases} \tag{3.3}$$

It could use these posteriors to generate new components or allocate observations. For latent allocation variables $Z = (Z_1, \ldots, Z_N)$, $\pi_j = p(Z_i = j)$ represents the prior probability that observation X_i is associated with component j. It could obtain the posterior probability that the observation X_i is allocated to component j conditional on having observation X_i to be:

$$p(Z_i = j \mid X_i) = \frac{p(X_i \mid \Theta_F, Z_i = j)}{p(X_i \mid \Theta_F)} \propto \pi_j \prod_{k=1}^{D} (\rho_{jk} p(X_{ik} \mid \theta_{jk}) + (1-\rho_{jk}) p(X_{ik} \mid \theta_{jk}^{irr})) \tag{3.4}$$

Latent relevancy variable ϕ_{ijk} indicates whether the feature k is relevant for component j given the observation X_i. $\rho_j = p(\phi_{ijk} = 1)$ represents the prior probability that the feature k is relevant for component j given observation X_i. The posterior probability that the feature k is relevant for component j conditioned on X_i is given by:

$$p(\phi_{ijk} = 1, Z_i = j \mid X_i) = p(Z_i = j \mid X_i) \frac{\rho_{jk} p(X_{ik} \mid \xi_{jk})}{\rho_{jk} p(X_{ik} \mid \xi_{jk}) + (1-\rho_{jk}) p(X_{ik} \mid \xi_{jk}^{irr})} \tag{3.5}$$

Posteriors for irrelevant features could be deduced in the same way.

$$p(\phi_{ijk} = 0, Z_i = j \mid X_i) = p(Z_i = j \mid X_i) \frac{(1-\rho_{jk}) p(X_{ik} \mid \xi_{jk}^{irr})}{\rho_{jk} p(X_{ik} \mid \xi_{jk}) + (1-\rho_{jk}) p(X_{ik} \mid \xi_{jk}^{irr})} \tag{3.6}$$

The likelihood function of X conditioned on the complete set of mixture parameters can be obtained. It will be used for further Bayesian inference derivation:

$$p(X \mid Z, \Phi, \xi, \xi^{irr}) = \prod_{i=1}^{N} \prod_{k=1}^{D} \left[p(X_{ik} \mid \xi_{jk})^{\phi_k} p(X_{ik} \mid \xi_{jk}^{irr})^{1-\phi_k} \right] \tag{3.7}$$

3.2 Non-parametric Bayesian Inference

In the Bayesian context, the most important step is the determination of the posteriors for inference. In this section, I describe a MCMC-based inference approach to learn the proposed model. The goal of inference is to approximate the posteriors of parameters which absorb the information to update the priors. Thus, I define a hierarchical Bayesian model and use conjugacy to develop the appropriate posteriors. The parameters are inferred based on a MCMC method.

3.2.1 Estimation for μ_{jk} and μ_{jk}^{irr}

I consider that the relevant and irrelevant mean parameters μ_{jk} and μ_{jk}^{irr} follow Gaussian priors with common hyperparameters mean λ and precision r respectively as follows:

$$p(\mu_{jk} \mid \lambda, r) \sim \mathcal{N}(\lambda,\, r^{-1}) \qquad p(\mu_{jk}^{irr} \mid \lambda^{irr}, r^{irr}) \sim \mathcal{N}(\lambda^{irr},\, (r^{irr})^{-1}) \tag{3.8}$$

where the hyperparameters mean λ and precision r are considered as common to all components in a specific dimension k. λ and r are given Gamma and inverse Gamma priors with the following shape and mean hyperparameters:

$$p(\lambda) \sim \mathcal{N}\,(e, f) \qquad p(r) \sim \gamma\,(g, h) \tag{3.9}$$

where λ, λ^{irr}, r, r^{irr} have same prior forms and I will omit replicated representation. The conditional posteriors for μ_{jk} and μ_{jk}^{irr} are obtained by combining the likelihood in Eq. (3.7) and the priors in Eq. (3.8).

$$\begin{aligned} p(\mu_{jk} \mid \dots) &\propto p(\mu_{jk} \mid \lambda, r)p(X \mid Z, \Phi, \xi, \xi^{irr}) \\ p(\mu_{jk}^{irr} \mid \dots) &\propto p(\mu_{jk}^{irr} \mid \lambda^{irr}, r^{irr})p(X \mid Z, \Phi, \xi, \xi^{irr}) \end{aligned} \tag{3.10}$$

For the posteriors of hyperparameters λ and r, Eq. (3.8) plays the role of likelihood and combined with priors Eq. (3.9) to obtain:

$$p(\lambda \mid \dots) \propto p(\lambda) \prod_{j=1}^{M} p(\mu_{jk} \mid \lambda, r)$$
$$p(r \mid \dots) \propto p(r) \prod_{j=1}^{M} p(\mu_{jk} \mid \lambda, r) \tag{3.11}$$

3.2.2 Estimation for S_{ljk}, S_{rjk} and S_{jk}^{irr}

The precision parameters S_{ljk}, S_{rjk} and S_{jk}^{irr} are endowed with Gamma priors of common hyperparameters β and w respectively:

$$\begin{aligned} &p(S_{ljk} \mid \beta_l, w_l \sim \gamma(\beta_l, w_l^{-1}) \\ &p(S_{rjk} \mid \beta_r, w_r) \sim \gamma(\beta_r, w_r^{-1}) \\ &p(S_{jk}^{irr} \mid \beta^{irr}, w^{irr}) \sim \gamma(\beta^{irr}, (w^{irr})^{-1}) \end{aligned} \tag{3.12}$$

where the hyperparameters β, w are common to all components in specific dimension k. β and w are given Gamma and inverse Gamma priors with the respective shape and mean hyperparameters:

$$p(\beta^{-1}) \sim \gamma\,(s, t) \qquad p(w) \sim \gamma\,(u, v) \tag{3.13}$$

where β_l, β_r, β^{irr}, w_l, w_r, w^{irr} have the same prior forms. The conditional posteriors for S_{ljk}, S_{rjk} and S_{jk}^{irr} are obtained by combining the likelihood in Eq. (3.7) and the priors in Eq. (3.12) as follows:

$$\begin{aligned} &p(S_{ljk} \mid \dots) \propto p(S_{ljk} \mid \beta_l, w_l) p(X \mid Z, \Phi, \xi, \xi^{irr}) \\ &p(S_{rjk} \mid \dots) \propto p(S_{rjk} \mid \beta_r, w_r) p(X \mid Z, \Phi, \xi, \xi^{irr}) \\ &p(S_{jk}^{irr} \mid \dots) \propto p(S_j^{irr} \mid \beta^{irr}, w^{irr}) p(X \mid Z, \Phi, \xi, \xi^{irr}) \end{aligned} \tag{3.14}$$

For the posteriors of hyperparameters β and w, Eq. (3.12) plays the role of likelihood and combined with priors Eq. (3.13), I can then obtain the following:

$$p(\beta \mid \dots) \propto p(\beta) \prod_{j=1}^{M} p(S_{jk} \mid \beta, w)$$
$$p(r \mid \dots) \propto p(w) \prod_{j=1}^{M} p(S_{jk} \mid \beta, w) \tag{3.15}$$

3.2.3 Estimation for ρ

Feature saliency ρ_{jk} has support over [0, 1] and considered naturally as Beta distribution with common hyperparameters a and b as following:

$$p(\rho_{jk} \mid a, b) \sim \text{Beta}\,(a, b) \tag{3.16}$$

where the shape hyperparameters a and b are common to all components and follow Gamma priors:

$$p(a) \sim \gamma\,(\delta_1, \delta_2) \qquad p(b) \sim \gamma\,(\varphi_1, \varphi_2) \tag{3.17}$$

I assume that the latent relevancy parameter ϕ_{jk} follows Bernoulli distribution with ρ_{jk}, so I have:

$$p(\phi_{jk} \mid \rho_{jk}) \sim \prod_{i=1}^{N} \rho_{jk}^{\phi_{ijk}} (1 - \rho_{jk})^{(1-\phi_{ijk})} = \rho_{jk}^{n_{jk}} (1 - \rho_{jk}^{N-n_{jk}}) \tag{3.18}$$

where $n_{jk} = \sum_{i=1}^{N} I_{\phi_{jk}=1}$ represents the amount of feature k relevant for component j given all of the observations. Considering Eq. (3.16) as the likelihood, I can obtain the conditional posterior by multiplying the prior in Eq. (3.18):

$$p(\rho_{jk} \mid \dots) \sim p(\phi_{jk} \mid \rho_{jk}) p(\rho_{jk} \mid a, b) \tag{3.19}$$

Conditional posteriors can then be obtained by combing Eq. (3.16) and Eq. (3.17) as follows:

$$\begin{aligned} p(a \mid \dots) &\propto p(a) \prod_{j=1}^{M} p(\rho_{jk} \mid a, b) \\ p(b \mid \dots) &\propto p(b) \prod_{j=1}^{M} p(\rho_{jk} \mid a, b) \end{aligned} \tag{3.20}$$

3.2.4 Complete Algorithm

Following the inference approach above, I propose a MCMC based algorithm for inferring the hierarchical Bayesian mixture model. Among Monte Carlo methods, Gibbs sampling is one of the most popular methods, and it is also widely used for complicated posterior sampling. I also use Metropolis-Hastings algorithm to generate non-standard posteriors. The Gibbs sequence converges to the joint posterior distribution. The algorithm can be summarized in Algorithm 2.

Algorithm 2 Infinite Asymmetric Gaussian Mixture with Feature Selection

procedure
    Initialization:
    Initialize the truncation levels K and T.
    repeat:
    Update the latent relevancy variables ϕ from Multivariate Bernoulli distribution with probability $p\big(\phi_{ijk} = 1, Z_i = j \mid X_i\big)$ in Eq. (3.5).
    Update mixture parameters μ, μ^{irr}, S_l, S_r, S^{irr} and ρ from conditional posteriors in Eq. (3.10), Eq. (3.14) and Eq. (3.19).
    Update hyperparameters λ, r, β, w, a, b from conditional posteriors and update DP concentration parameter α from conditional posterior in Eq. (3.11), Eq. (3.15) and Eq. (3.20) and Eq. (2.11).
    Update the latent indicator variables Z in Eq. (3.3).
    Update the component number M.
    The convergence criteria is reached when the difference of the current value of joint posteriors and the previous value is less than 10^{-4}. Otherwise, repeat step 1-5 until convergence.
    until convergence

3.3 Experimental Results

In this section, I validate the algorithm on several challenging experiments; particularly, dynamic textures clustering and scene categorization. I compare the results with multiple state-of-the-art methods.

Among these applications, the hyperparameters chosen are $e = \mu_y$, $f = \sigma^2$, g=2, h=$\frac{2}{\sigma^2}$, s=0.5, t=2, u=0.5, v=$\frac{2}{\sigma^2}$, δ_1=2, δ_2=0.5, φ_1=2, φ_2=0.5, κ=0.5, and η=2. μ_x and σ_x^2 are the mean and variance of observations.

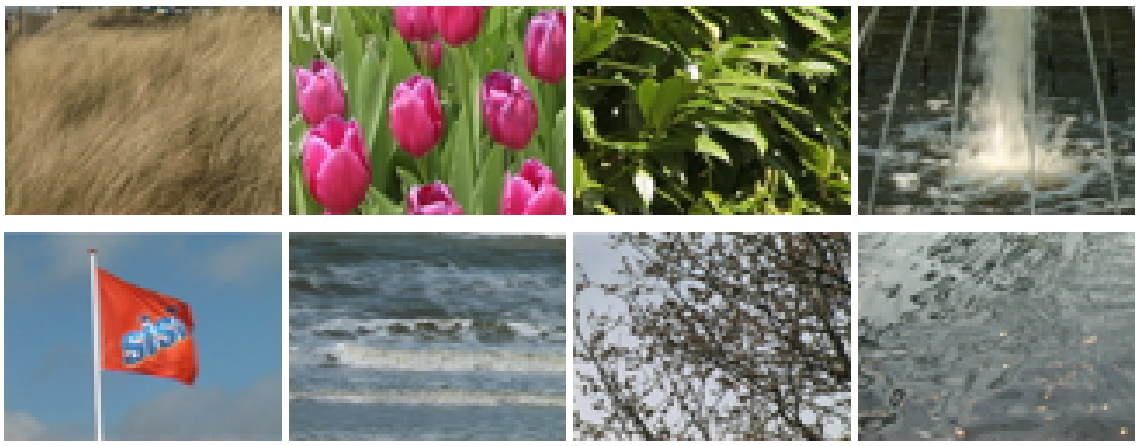


Figure 3.1: Sample frames from the DynTex database.

Approach	IGM	IDM	IGDM	IBLM	IAGM
Acc (%)	74.87	77.75	80.62	83.37	88.79

Table 3.1: Average accuracy of different algorithms for dynamic textures clustering.

3.3.1 Dynamic Textures Clustering

Dynamic textures are the temporal extension of spatial textures which are defined as sequences of images of moving scenes that exhibit certain stationarity properties in time (sea-waves, smoke, foliage, whirlwind) [18]. Dynamic textures have drawn tremendous attention during the past years due to their application in several domains in image processing and pattern recognition, such as motion classification, video registration, and computer games [19]; [20]. In the experiment, I apply the proposed IAGM with simultaneous feature selection for clustering dynamic textures with a representation of LBP-TOP features.

I carry out the experimentation on the challenging dynamic textures dataset; DynTex [21], for evaluating the performance of the algorithm. This dataset contains over 650 dynamic texture video sequences from several categories. In the case, I use a subset of video sequences from 8 different categories: candle, flag, flower, fountain, grass, sea, smoke and tree. Each category has about 20 video sequences. The sample frames from each category are shown in Figure. 3.1. As a preprocessing step, I extract LBP-TOP descriptors from the selected video sequence.

In the experiment, I adopt the parameter choice of 4,4,4,1,1,1 as suggested in [22]. The chosen setting of the LBP-TOP descriptor achieves a good performance while it also provides a comparative shorter 48-length feature vector.

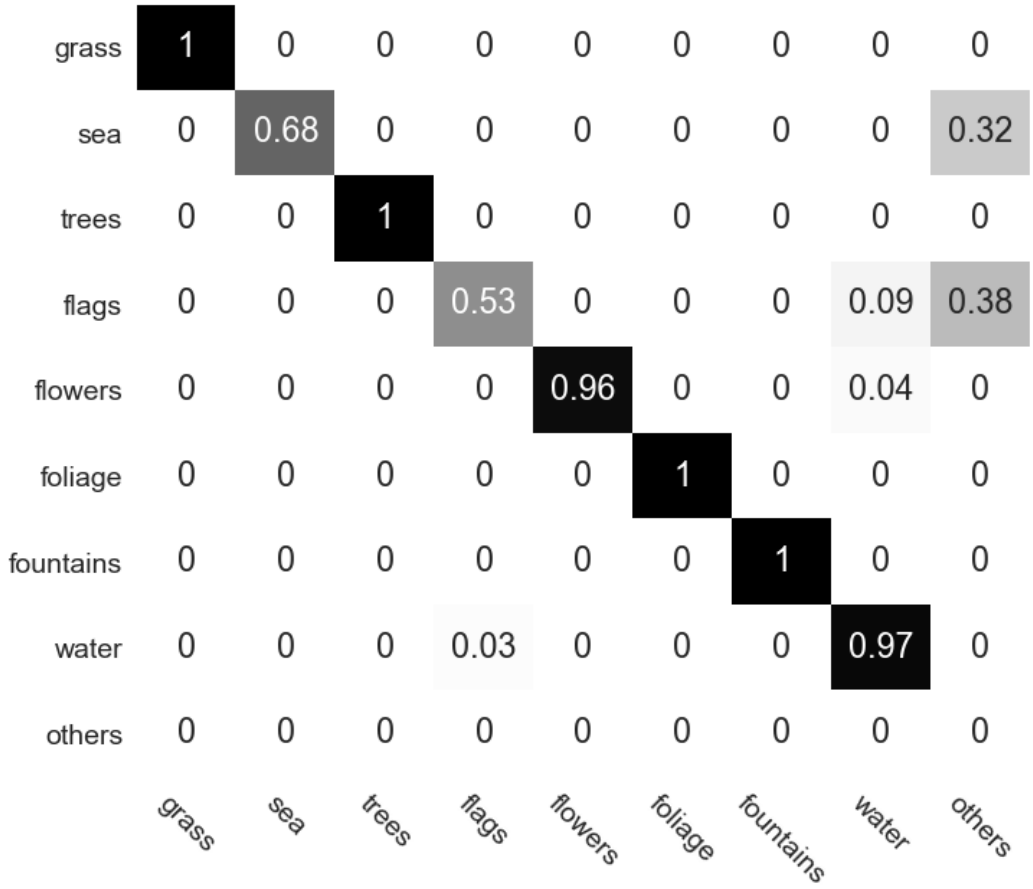


Figure 3.2: Confusion matrix of the IAGM with feature selection for the DynTex database.

Approach	GMM-EM	GMM-RPEM	prob	SPM	BOW	MLE-Scene	MM-Scene	IAGM
Acc (%)	69.51	69.76	63.88	66.00	71.57	69.87	71.70	73.33

Table 3.2: Average accuracy of different algorithms for scene categorization.

Obtained features are modeled using proposed IAGM algorithm. In order to evaluate the performance of the proposed method, I compare the proposed approach with other methods; infinite Beta-Liouville mixture, infinite generalized Dirichlet mixture, infinite Dirichlet mixture, and infinite Gaussian mixture models. I run these approaches 30 times and get average results for validating the performance. The averages of the clustering accuracy can be observed in Table. 3.1. Figure. 3.2 shows the confusion matrix for the dataset using IAGM with feature selection.

According to the results, IAGM with feature selection approach outperforms other methods in terms of the highest categorization accuracy rate (87.02%). It shows significant improvement compared with other methods because it could successfully distinguish 6 categories leading to a higher overall accuracy

The results of dynamic texture clustering demonstrate the advantage of applying mixture model which includes asymmetry characteristics of observations for modelling non-standard shaped observations. Meanwhile, simultaneously performing feature selection allows for the inclusion of background noise while accurately representing important features that contribute to better performance.

3.3.2 Scene Categorization

Humans are proficient at perceiving, recognizing and understanding natural scenes. The representation of scene images has drawn considerable interests in recent years. In this section, I apply the proposed algorithm to the challenging scene categorization task. Thus, I divide the approach into three parts: feature extraction, image representation, and scene classification.

In this application, I use the UIUC sports event dataset [23] to validate the performance of the algorithm. This dataset consists of 8 different sport event classes: rowing (250 images), badminton (200 images), polo (182 images), bocce (137 images), snowboarding (190 images), croquet (236 images), sailing (190 images), and rock climbing (194 images). Fig. 3.3 demonstrates its diverse nature.

I represent each image by a collection of local image patches. Particularly, I adopt

Figure 3.3: Sample frames from UIUC sport event dataset. the samples show the diversity of background and complexity of information

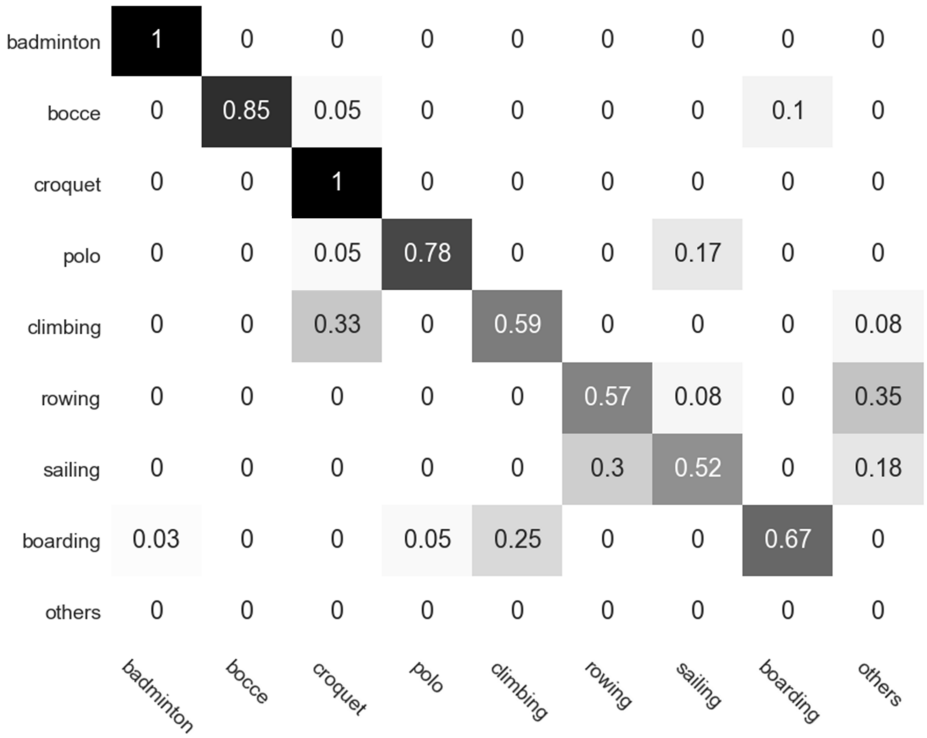


Figure 3.4: Confusion matrix of the the IAGM with feature selection for the UIUC sport event dataset

scale-invariant feature transform (SIFT) descriptors of 16 $\times$ 16 pixel patches computed over a grid with spacing of 8 pixels. Then, I employ bag of visual words (BoVW) approach to have an overall representation of each image. I then use k-means algorithm to cluster the training dataset in a vocabulary of V visual words. Each SIFT keypoint will be allocated to the nearest vocabulary in codebook. The points in the image can be approximated by each of the visual words. Thus, each image can be represented as a frequency histogram over the V visual words. Then, I use IAGM with feature selection model to classify the processed data. For each sport event class, I randomly select 70 images as a training and 60 images as a testing. I run the proposed algorithm 30 times to obtain the average accuracy results for comparison.

In order to demonstrate the advantages of the algorithm, I compared the model with a number of state-of-the-art approaches within similar area. These approaches include Gaussian mixture model with Expectation Maximization algorithm (GMM-EM) [10], Gaussian mixture model with Rival Penalized Expectation Maximization (GMM-RPEM) [24], GIST [25], multi-class supervised Latent Dirichlet Allocation and multi-class supervised Latent Dirichlet Allocation with annotations (probabilistic) [26], Spatial pyramid matching (SPM) [27], bag of keypoints (BOK) [28], maximum likelihood estimation Scene (MLE-Scene) and Max-Margin Scene (MM-Scene) [29]. The evaluation results are shown at Table. 3.2. Fig. 3.4 displays the confusion matrix for IAGM applied on sport dataset.

We can observe from the results that the proposed IAGM with simultaneous feature selection outperforms other approaches under consideration and provides better average accuracy results for the task of scene categorization.

Chapter 4

Variational Inference for Finite Asymmetric Gaussian mixture

In this chapter, I consider a finite mixture model based on AGD which provides a better fit for the data. I apply a variational learning framework to estimate the parameters and adjust model complexity automatically. I handle the problem of inferring non-conjugate variables by introducing gradient ascent inference method.

4.1 Finite Asymmetric Gaussian Mixture

In this chapter, the definition of the AGD comes with standard deviation parameters instead of with precision parameters which appear in previous chapters due to the convenience of this setting for variational Bayes inference. Mathematically, this is denoted as follows:

$$p(X_i \mid \xi_j) \propto \prod_{k=1}^{D} \frac{1}{\sigma_{ljk} + \sigma_{rjk}} \times \begin{cases} \exp\big[-\frac{(X_{ik}-\mu_{jk})^2}{2\sigma_{ljk}^2}\big] & \text{if } X_{ik}<\mu_{jk} \\ \exp\big[-\frac{(X_{ik}-\mu_{jk})^2}{2\sigma_{rjk}^2}\big] & \text{if } X_{ik} \geq \mu_{jk} \end{cases} \tag{4.1}$$

where $\xi_j = (\mu_j, \sigma_{lj}, \sigma_{rj})$ is the complete set of parameters for AGD with $\mu_j = (\mu_{j1}, \ldots, \mu_{jD})$, $\sigma_{lj} = (\sigma_{lj1}, \ldots, \sigma_{ljD})$, and $\sigma_{rj} = (\sigma_{rj1}, \ldots, \sigma_{rjD})$. μ_{jk}, σ_{ljk} and $_{rjk}$ are the mean, the left and right standard deviations for the kth-dimensional distribution, respectively. I still consider each dimension of observation X_i as independent and thus its covariance matrix is diagonal. This assumption reduces the computational time during deployment.

The latent indicator variables Z, $Z = (Z_1, \ldots, Z_N)$, indicate which components the observations belong to. $Z_i = (Z_{i1}, \ldots, Z_{iM})$ and each element Z_{ij} is assigned value 1 when the observation X_i is associated with component j; otherwise, it is 0. The mixing coefficient $\pi_j = p(Z_i = j)$, $j = \{1, \ldots, M\}$ specifies the probability that an observation X_i is allocated to component j. Hence, the marginal distribution over Z given a Multinomial prior is as follows:

$$p(Z \mid \pi) \sim \text{Multi}(\pi) = \prod_{i=1}^{N} \prod_{j=1}^{M} \pi_j^{\mathbb{I}(Z_i = j)} \tag{4.2}$$

I choose a Dirichlet distribution prior over the mixing coefficients π:

$$p(\pi) = \text{Dir}(\pi \mid \alpha_0) = \frac{\Gamma(\sum_{j=1}^{M} \alpha_0)}{\prod_{j=1}^{M} \Gamma(\alpha_0)} \prod_{j=1}^{M} \pi_j^{\alpha_0 - 1} \tag{4.3}$$

where by symmetry I choose the same parameter α_0 for each component. I assume that μ follows a Gaussian distribution with mean λ and precision r, i.e. the inverse variance of Gaussian distribution. The standard deviations σ_l and σ_r follow a Gaussian distribution with a mean value whose value is set experimentally and a high value standard deviation setting [30]:

$$\begin{aligned} p(\mu_{jk} \mid \lambda, r) &\sim \mathcal{N}(\lambda_{jk}, r_{jk}) \\ p(\sigma_{ljk} \mid m_l, s_l) &\sim \mathcal{N}(m_{ljk}, s_{ljk}^2) \\ p(\sigma_{rjk} \mid m_r, s_r) &\sim \mathcal{N}(m_{rjk}, s_{rjk}^2) \end{aligned} \tag{4.4}$$

4.2 Variational Inference Framework

4.2.1 Mean field Variational Approximation

In this section, I use variational inference to closely approximate the parameter set $w = \big(Z, \pi, \mu, \sigma_l, \sigma_r\big)$ of the mixture. I consider the problem of calculating the posterior density $p(w \mid x)$ given the model evidence $p(x)$ which is hard to compute with latent parameter set w [31]. The explicit rationale behind analytical intractability is that the evidence term is usually hard to compute.

The idea behind variational inference is to approximate the posterior $p(w \mid x)$ with variational distribution $q(w)$ from a constrained family of distributions. The objective is to adopt the closest one in a given variational distribution family. I choose the Kullback-Leibler (KL) divergence to measure the distance between the posteriors and the variational distributions:

$$KL\big(q(w) \parallel p(w \mid x)\big) = E_q[\log q(w)] - E_q[\log p(w \mid x)] \tag{4.5}$$

Thus, variational inference amounts to solving an optimization problem: choosing the variational parameters that minimizes KL divergence. The family of distributions is chosen to make the optimization problem tractable.

However, the divergence is difficult to compute since it requires finding the distribution that I wish to approximate. I then expand the KL divergence and find the evidence lower bound (ELBO) in addition to the log marginal distribution of the observations as follows:

$$\begin{aligned} & KL\big(q(w) \parallel p(w \mid x)\big) \\ =& E_q[\log q(w)] - E_q[\log p(w, x)] + \log p(x) \\ =& \mathcal{L}(w) + \log p(x) \end{aligned} \tag{4.6}$$

As such, it minimizes the ELBO that is equal to log marginal likelihood term, which is constant with respect to variational distribution $q(w)$, minus KL divergence. It reaches a maximum when $q(w) = p(w \mid x)$; the KL divergence is zero.

Typically, q will be constrained to a family of simpler distributions, and the ELBO is optimized to find the distribution in the family that is closest (in terms of KL divergence) to the true posterior. Here, I follow the mean field assumption [32]. This approach assumes independence between latent variables to factorize the family of variational distributions so that the true posterior is easy to compute. Then, the variational distributions have the factorized form:

$$q(Z, \pi, \mu, \sigma_l, \sigma_r) = \prod_{j=1}^{M} q(\pi_j) \prod_{i=1}^{N} q(Z_i) \prod_{j=1}^{M} \prod_{k=1}^{D} q(\mu_{jk}) q(\sigma_{ljk}) q(\sigma_{rjk}) \tag{4.7}$$

where $q(\pi_j)$ is a Dirichlet prior with parameter α_j , $q(Z_i)$ is a Multinomial prior with parameter ϕ and $q(\mu_{jk})$ is considered as a Gaussian distribution with mean m and variance

Σ. I also define the variational distributions of σ_l and σ_r as a Gaussian with mean ι and standard deviation υ:

$$
\begin{aligned}
q(\pi_j) &\sim \text{Dir}(\pi \mid \alpha_j) \\
q(Z_i) &= \text{Multi}(\phi_i) \\
q(\mu_{jk}) &= \mathcal{N}(m_{jk}, \Sigma_{jk}) \\
q(\sigma_{ljk}) &= \mathcal{N}(\iota_{ljk}, \upsilon^2_{ljk}) \\
q(\sigma_{rjk}) &= \mathcal{N}(\iota_{rjk}, \upsilon^2_{rjk})
\end{aligned} \tag{4.8}
$$

For the proposed finite AGM and using the mean field assumption, the ELBO is:

$$
\begin{aligned}
\mathcal{L}(\Theta) = \sum_{i=1}^{N} \big(E_q[\log p(X_i \mid Z_i, \mu, \sigma_l, \sigma_r)] + E_q[\log p(Z_i)] \big) + E_q[\ln p(\pi)] \\
+ E_q[\log p(\mu)] + E_q[\log p(\sigma_l)] + E_q[\log p(\sigma_r)] - E_q[\log q(\pi, Z, \mu, \sigma_l, \sigma_r)]
\end{aligned} \tag{4.9}
$$

By applying Eq. (4.9) to each factor, I obtain the optimal solutions for the factors of the variational posteriors. I next present the explicit coordinate ascent variational inference (CAVI) to optimize the ELBO in Eq. (4.9) where I find the updates of the variational parameters of mixing proportions V and indicators Z are:

$$
\phi_{ij} = \frac{r_{ij}}{\sum_j r_{ij}} \tag{4.10}
$$

$$
\begin{aligned}
r_{ij} = \exp\{ E_q[\log \pi_j] - \sum_{k}^{D} E_q[\log(\sigma_{ljk} + \sigma_{rjk})] \\
- \sum_{k, X_{ik} < \mu_{jk}}^{D} \frac{X^2_{ik} + E_q[\mu^2_{jk}] - 2X_{ik} E_q[\mu_{jk}]}{2E_q[\sigma^2_{ljk}]} \\
- \sum_{k, X_{ik} \geq \mu_{jk}}^{D} \frac{X^2_{ik} + E_q[\mu^2_{jk}] - 2X_{ik} E_q[\mu_{jk}]}{2E_q[\sigma^2_{rjk}]} \}
\end{aligned} \tag{4.11}
$$

$$
\alpha_j = \alpha_0 + \sum_{i=1}^{N} \phi_{ij} \tag{4.12}
$$

The variational updates of latent variable μ can be obtained as:

$$\Sigma_{jk} = \Big(\sum_{i, X_{ik} < \mu_{jk}}^{N} \frac{\phi_{ij}}{E_q[\sigma_{ljk}^2]} + \sum_{i, X_{ik} \geq \mu_{jk}}^{N} \frac{\phi_{ij}}{E_q[\sigma_{rjk}^2]} + r \Big)^{-1}$$

$$m_{jk} = \Sigma_{jk} \Big(\sum_{i, X_{ik} < \mu_{jk}}^{N} \phi_{ij} \frac{X_{ik}}{E_q[\sigma_{ljk}^2]} + \sum_{i, X_{ik} \geq \mu_{jk}}^{N} \phi_{ij} \frac{X_{ik}}{E_q[\sigma_{rjk}^2]} + \lambda r \Big) \tag{4.13}$$

It is intuitive to update the parameters (π, Z, μ) but closed forms are not achieved for the standard deviation variables (σ_l, σ_r) because of non-conjugate characteristics. Although there are multiple solutions for non-conjugate models, such as Delta Method Variational Inference proposed in [33], Taylor Expansion cannot be used to approximate intractable density of (σ_l, σ_r) because it is unviable to get first-order derivative solution. Thus, I consider a gradient-based optimization method, the Black Box variational Inference (BBVI), to approximate standard deviation parameters [34].

4.2.2 Black Box Variational Inference

For the BBVI, the variational lower bound of probabilistic model associated with parameter $\sigma = (\sigma_l, \sigma_r)$ is given as follows:

$$\mathcal{L}(\sigma) = E_{q(\sigma)}[\log p(x, \sigma) - \log q(\sigma \mid \theta)] \tag{4.14}$$

where θ is a set of free parameters of variational distribution $q(\sigma \mid \theta)$. the objective is to accurately approximate $p(x \mid \sigma)$ with a setting of θ and optimize the ELBO. In BBVI, I use stochastic optimization approach to maximize the ELBO based on the noisy estimation of its gradient.

Given a certain learning rate ρ_t following Robbins-Monro conditions where t denotes the current iteration, it is possible to guarantee that the optimized function $f(x)$ converges to a maximum:

$$x_{t+1} \leftarrow x_t + \rho_t h_t(x_t) \tag{4.15}$$

where $h_t(x_t)$ is a realization of the random variable $H(x)$ whose expectation is the gradient of objective $f(x)$. The derivative of the ELBO with respect to the variational distribution can be obtained:

$$\nabla_\theta \mathcal{L}(\sigma) = E_q[\nabla_\theta \log q(\sigma \mid \theta) \big(\log p(x, \sigma) - \log q(\sigma \mid \theta) \big)] \tag{4.16}$$

where the gradient of log variational distribution, $\nabla_\theta \log q(\sigma \mid \theta)$, is considered as the score function. Using above gradient of the objective, it can sample from variational posterior to get noisy but unbiased gradients, which I utilize to update the parameters. The noisy unbiased estimation of gradients of the ELBO with Monte Carlo samples from the variational distribution can be denoted as:

$$\nabla_\theta \mathcal{L}(\sigma) = \frac{1}{S}\sum_{s=1}^{S} \nabla_\theta \log q(\sigma_s \mid \theta)\big(\log p(x, \sigma_s) - \log q(\sigma_s \mid \theta)\big)$$
$$\text{where } \sigma_s \sim q(\sigma \mid \theta) \tag{4.17}$$

where s indexes the samples and S indicates the number of samples drawn from the variational distribution. I consider the factorized parameters σ_{ljk} and σ_{rjk} to follow a diagonal Gaussian variational family with mean ι and standard deviation υ. Thus, the inference using gradient ascent is performed:

$$\begin{aligned}
&\nabla_{\iota_{ljk},\upsilon_{ljk}} \mathcal{L}(\sigma_{ljk}) \\
=&\frac{1}{S}\sum_{s=1}^{S} \nabla_{\iota_{ljk},\upsilon_{ljk}} \log q(\sigma^s_{ljk} \mid \iota_{ljk}, \upsilon_{ljk})\big(\log p(x, \sigma^s_{ljk}) - \log q(\sigma^s_{ljk} \mid \iota_{ljk}, \upsilon_{ljk})\big) \\
&\nabla_{\iota_{rjk},\upsilon_{rjk}} \mathcal{L}(\sigma_{rjk}) \\
=&\frac{1}{S}\sum_{s=1}^{S} \nabla_{\iota_{rjk},\upsilon_{rjk}} \log q(\sigma^s_{rjk} \mid \iota_{rjk}, \upsilon_{rjk})\big(\log p(x, \sigma^s_{rjk}) - \log q(\sigma^s_{rjk} \mid \iota_{rjk}, \upsilon_{rjk})\big)
\end{aligned} \tag{4.18}$$

The expectations included in the above formulas are calculated by:

$$E_q[\mu_{jk}] = m_{jk} \quad E_q[\mu^2_{jk}] = m^2_{jk} + \Sigma_{jk} \tag{4.19}$$

$$E_q[\sigma_{ljk}] = \iota_{ljk} \quad E_q[\sigma^2_{ljk}] = \iota^2_{ljk} + \upsilon^2_{ljk} \tag{4.20}$$

$$E_q[\sigma_{rjk}] = \iota_{rjk} \quad E_q[\sigma^2_{rjk}] = \iota^2_{rjk} + \upsilon^2_{rjk} \tag{4.21}$$

I use Jensen's Inequity to approximate the $E_q[\log(\sigma_{ljk} + \sigma_{rjk})]$ by replacing with the upper bound:

$$E_q[\log(\sigma_{ljk} + \sigma_{rjk})] \leq \log(E_q[\sigma_{ljk} + \sigma_{rjk}]) = \log(\iota_{ljk} + \iota_{rjk}) \tag{4.22}$$

Algorithm 3 Finite Asymmetric Gaussian Mixture

procedure
    Initialization:
    Initialize a relatively large starting number of mixture components K and hyperparameters m_0, v_0, d_{l0}, s_{l0}, d_{r0}, and s_{r0}.
    Initialize variational parameters r, α, m and v.
    repeat:
    Update local variational parameters r_{ik} using Eq. (4.10) and Eq. (4.11).
    Update global variational parameters α_k, m_{kd} and v_{kd} using Eq. (4.12) to Eq. (4.13).
    Update global latent variables σ_{ljk} and σ_{rjk} by BBVI from Section 4.2.2.
    Check for convergence, i.e. the difference between the current value of ELBO and previous value is less than 10^{-3}.
    until convergence
    Compute the expected values of π_k, μ, σ_l, and σ_r

4.2.3 Complete Algorithm for the Proposed Framework

In this subsection, I detail the steps of the proposed AGM framework, including CCVI and BBVI. I trace the convergence by monitoring the ELBO difference between epochs. Convergence is achieved when the ELBO difference is less than a threshold set experimentally to 10^{-3} for each iteration. The variational inference of the AGM is summarized in Algorithm 3.

4.3 Experimental Results

4.3.1 Experimental Setup

In this section, I apply the proposed finite AGM framework for the background subtraction task with a pixel-level approach. Pixel-level methods model the value of a particular pixel over time as a mixture of poised distribution; the AGD in this case. I start by modeling the background by using the proposed AGM then divide the mixtures into foreground and background components.

Each pixel $\mathcal{X}_p$ is allocated a label as foreground or background according to the measured model $p(\mathcal{X}_p \mid \Theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} \pi_j p(\mathcal{X}_p \mid \xi_k)$ where the pixel sequence $\mathcal{X}$ has P pixels represented as $\mathcal{X} = (\mathcal{X}_1, \ldots, \mathcal{X}_P)$. I assume that the background objects persist with relatively low standard deviation and high weight. This is because they usually remain stationary and occur regularly for a given pixel location. Therefore, I consider the components

that appear frequently and vary on a limited range as the background objects of a scene.

Accordingly, I rank the mixture components by the fitness value of $\pi_k/(|\sigma_{lk}| + |\sigma_{rk}|)$, where π_k, $|\sigma_{lk}|$, and $|\sigma_{rk}|$ are the weight value, the corresponding norms of left and right standard deviations of the kth component of the mixture model. The fitness value increases as the distribution appears more frequently and remains stable. Hence, the first B components are associated with the background:

$$B = \underset{b}{\operatorname{argmin}} \sum_{j=1}^{b} \pi_j > T \tag{4.23}$$

where threshold variable T is a proportion of the minimum share of the observations treated as the background in a given image sequence. Thus, I rank the distributions according to the probability of belonging to the background. A pixel value is fit to the closest distribution whereby a match occurs when a pixel value is no more than 3 standard deviations away from the distribution. The parameters of the first matched component will be adjusted as follows:

$$\pi_{kt} = (1 - \beta)\pi_{k(t-1)} + \beta\mathcal{M}_{kt} \tag{4.24}$$

$$\mu_{kt} = (1 - \beta)\mu_{k(t-1)} + \rho X_t \tag{4.25}$$

$$\begin{aligned} \sigma_{l_{kt}}^2 &= (1 - \beta)\sigma_{l_{k(t-1)}}^2 + \rho(X_t - \mu_{kt})^2 \quad \text{if } \mu_{kt} < X_t \\ \sigma_{r_{kt}}^2 &= (1 - \beta)\sigma_{r_{k(t-1)}}^2 + \rho(X_t - \mu_{kt})^2 \quad \text{if } \mu_{kt} \geq X_t \end{aligned} \tag{4.26}$$

where β defines the learning speed and indicator $\mathcal{M}_{kt}$ indicates whether the pixel value fits component k. π_{kt}, μ_{kt}, σ_{lkt} and σ_{rkt} are expectations of variables given by the j the component at t.

Finally, ρ is defined as:

$$\rho = \beta p(X_t \mid \mu_{kt}, \sigma_{lkt}, \sigma_{rkt}) \tag{4.27}$$

where $p(X_t \mid \mu_{kt}, \sigma_{lkt}, \sigma_{rkt})$ represents the AGD density. When a new pixel is checked against the existing distributions, the lowest ranking distribution is replaced by new emerged component with low weight, high standard deviations, and the same mean value.

Table 4.1: Experimental results for the background subtraction task on skating image sequences

	Stauffer et al. [14]	***AGM***
recall	69.40%	74.72%
precision	63.22%	92.62%

(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by GMM. (d) Result frame by AGM.

Figure 4.1: Sample results from skating video sequences.

4.3.2 Results and Discussion

I employ the proposed approach on the Change Detection 2014 dataset (CDnet 2014) [15]. This dataset consists of numerous videos grouped into 11 categories describing a wide range of change detection tasks. These videos obtained by different cameras varying from low-resolution Internet Protocol (IP) cameras, through higher resolution consumer grade camcorders to thermal cameras. Accordingly, the spatial resolutions of the video sequences in the 2014 CDnet are 320×240 to 720×486. Videos captured by low-resolution IP cameras suffer from noticeable radial distortion. Besides, various cameras suffer different bias due to diverse white balancing algorithms employed. Some cameras also apply an automatic exposure adjustment algorithm which causes a fluctuation in the brightness.

In this thesis, I select video sequences in the bad weather category, which show the low-visibility winter storm conditions. This dataset includes a snowing scene and people skating in the snow. It presents a double challenge: not only should the algorithm detect the snow accumulation, it also needs to distinguish the dark tire tracks left in the snow which have the potential to cause false positives.

In order to evaluate the performance of the proposed approach, I compare the developed approach with state-of-the-art method introduced by Sauffer et al. [14]. In this application, I set the initial mixture components number K as 9, the distribution matching variable $M = 3$, and the threshold factor $T = 0.8$. For the hyperparameters used for learning,

I sample them randomly from their respective support intervals. Figure. 4.1 shows the samples from the input frame, the ground truth frame, the foreground segmentation result evaluated by the GMM as well as the proposed AGM.

Figure 4.2: Confusion matrix of the proposed method for the background subtraction task on the skating image sequences. BG denotes background and FG denotes the foreground.

For quantitative analysis, I adopt two evaluation metrics: recall and precision. Recall identifies the number of correctly classified foreground pixels over total number of foreground pixels in the ground truth, while the precision represents the percentage of the number of correctly identified foregrounds by the number of pixels detected as foreground.

The results evaluated by the proposed AGM and GMM are shown in Table. 4.1 and the confusion matrix of the proposed method is displayed in Figure. 4.2. It is shown that the proposed AGM outperforms GMM in terms of the precision and the recall metrics. It considerably improves the precision due to the higher precision achieved in capturing the shape of pixels with the asymmetric Gaussian assumption compared with Gaussian distribution. The proposed approach is also more robust to background change. This includes the motion of the pedestrians and the snow accumulation on the trees. Moreover, it has the merit to distinguish the widespread snow in the frames. However, GMM faces a difficulty for determining the background condition of snow; thus, achieve inferior results compared with the adaptive asymmetric assumption.

Chapter 5

Variational Inference for Infinite Asymmetric Gaussian Mixture Models with Simultaneous Feature Selection

In this chapter, I detail the variational Bayes inference framework for infinite asymmetric Gaussian mixture based on the DP. I also incorporate feature selection approach to determine informative features set. This helps us eliminate the irrelevant features and improve the effectiveness of the algorithm. I evaluate the performance of the models with background subtraction task.

5.1 Infinite Asymmetric Gaussian Mixture

5.1.1 Dirichet Process with the stick-breaking process

The DP is a stochastic process whose realization is a probability distribution, with a non-negative scaling parameter α and base distribution G_0 [35]. It is used to form a distribution over discrete distributions that place their mass on a countably infinite set of atoms. For a DP distributed random measure $G \sim \text{DP}(\alpha, G_0)$ is drawn from k-partitions of measure sets $\{B_1, \ldots, B_k\}$ which are discrete with probability one [36]:

$$(G(B_1), \ldots, G(B_k)) \sim (\alpha G_0(B_1), \ldots, \alpha G_0(B_k)) \tag{5.1}$$

When applied with variational inference methods, the learning approach is usually

based on the stick-breaking process representation. This representation provides a set of latent variables on which to place an approximate posterior [35] [37]. The stick-breaking process gives an explicit representation of the DP which is based on two infinite sequences of independent and identically distributed random variables V_j and η_j, for $j \in \{1, \ldots, \infty\}$ [38]. Here, I use this construction to form the DP mixture model as:

$$p(V_j \mid \alpha) = \text{Beta}(1, \alpha) \quad p(\eta_j^* \mid \alpha, G_0) \sim G_0 \tag{5.2}$$

where V_j is the stick-breaking length distributed according to Beta distribution with concentration variable α. η_j^* is the atom drawn independently from base distribution G_0. Considering stick pieces as the proportion of unit length, I define the stick-breaking representation of the random representation G as follows:

$$\pi_j = V_j \prod_{s=1}^{j-1}(1 - V_s) \quad G = \sum_{j=1}^{\infty} \pi_j \delta_{\eta_j^*} \tag{5.3}$$

The mixing proportions $\pi = (\pi_j)_{j=1}^{\infty}$ are formed by repetitively breaking a unit length stick into an infinite number of pieces and noting that these proportions sum to one. δ_{η^*} is a probability measure concentrated at η^* with weights π. This infinite collection of variables forms a point on the infinite simplex.

One of the most common applications of the DP is as a nonparametric prior on the parameters of a mixture model. Hence, I can interpret the DP mixture as a mixture model with unbounded number of components that can grow as new data is observed. Then, I have a set of observations $x = \{x_1, \ldots, x_N\}$ with parameters $\eta = \{\eta_1, \ldots, \eta_N\}$ where N is the number of given samples.

Combining these processes and representation, I form the distribution of random measure G according to following step:

$$\begin{aligned} G \mid \{\alpha, G_0\} &\sim \text{DP}(\alpha, G_0) \\ \eta_n \mid G &\sim G \\ x_n \mid \eta_n &\sim p(x_n \mid \eta_n) \end{aligned} \tag{5.4}$$

where the random measure G is drawn from a DP prior $DP(\alpha, G_0)$ and atom η_n is drawn independently and identically from measure G_0 with the probability π_n given by the nth stick-breaking length V_n. This distribution is considered as a discrete distribution with

the mass on an infinite set of atoms. The datapoint x_n has a distribution $p(x_n \mid \eta_n)$ and clusters into a small number of G although these measures place mass on an infinite set of atoms.

I use the above DP mixture model with the stick-breaking process representation. The random variable η_n will takes on value η_j^* with weight π_j. The assignment will be denoted by the latent indicator variable Z_n representing the allocation of datapoint x_n. I can elucidate the generative process of the DP mixture model as follows:

1. Draw $V_j \mid \alpha \sim \text{Beta}(1, \alpha)$, $j \in \{1, \ldots, \infty\}$.
2. Draw $\eta_j^* \mid G_0 \sim G_0$, $j \in \{1, \ldots, \infty\}$.
3. Draw the n-th observations, $n \in \{1, \ldots, N\}$:
   - Draw $Z_n \mid V \sim \text{Multi}(\pi)$.
   - Draw $x_n \mid Z_n \sim p(x_n \mid \eta_{Z_n}^*)$.

In this construction, the measures η are drawn from the base distribution and stick lengths V to define a probability distribution on these measures, which specifies a set of relative prevalence in the mixture model. For the observations, the latent indicators Z are distributed according to a Multinomial distribution with mixing weights π, and π is generated from sticks V.

5.1.2 Dirichlet Process of Asymmetric Gaussian Distributions

Here, I restrict the proposed distribution of $p(X \mid \eta)$ in Eq. (5.1) to AGD with the set of parameters ξ to obtain Dirichlet process asymmetric Gaussian mixture (DPAGM). Furthermore, I set a truncation on the maximum component number M of the stick-breaking representation. I consider a truncation level which restrict the mixture model to M component mixture model:

$$p(X \mid \Theta) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j p(X_i \mid \xi_j) \tag{5.5}$$

where $p(X_i \mid \xi_j)$ denotes the density function of AGD which is given in chapter 4. I assume that each dimension of observation X_i is independent and its covariance matrix is diagonal. This assumption reduces the computational time during deployment.

The latent indicator variables Z, $Z = (Z_1, \ldots, Z_N)$, indicate which components the observations belong to and $\mathbb{I}(Z_i = j)$ is the indicator function. According to Eq. (5.3), which expresses the stick-breaking process construction, mixing proportions π are represented by sticks V. Hence, the marginal distribution over Z given a Multinomial prior is as follows:

$$p(Z \mid V) = \prod_{i=1}^{N} \prod_{j=1}^{M} [V_j \prod_{s=1}^{j-1} (1 - V_s)]^{\mathbb{I}(Z_i = j)} \tag{5.6}$$

With the Beta prior of sticks V given in Eq. (5.2), I truncate the number of components to M:

$$p(V \mid \alpha) = \prod_{j=1}^{M} \text{Beta}(1, \alpha) = \prod_{j=1}^{M} \alpha (1 - V_j)^{\alpha - 1} \tag{5.7}$$

where μ follows a Gaussian prior with mean λ and precision r, i.e. the inverse variance of Gaussian distribution. The standard deviations σ_l and σ_r follow a Gaussian distribution with a mean whose value is set experimentally and a high value standard deviation setting [30]:

$$\begin{aligned} p(\mu_{jk} \mid \lambda, r) &\sim \mathcal{N}(\lambda_{jk}, r_{jk}) \\ p(\sigma_{ljk} \mid m_l, s_l) &\sim \mathcal{N}(m_{ljk}, s_{ljk}^2) \\ p(\sigma_{rjk} \mid m_r, s_r) &\sim \mathcal{N}(m_{rjk}, s_{rjk}^2) \end{aligned} \tag{5.8}$$

5.2 Variational Inference Framework

5.2.1 Variational approximation

In this section, I use variational inference to precisely approximate the parameter set $w = \big(V, Z, \mu, \sigma_l, \sigma_r\big)$ of the DP mixture model. I consider the problem of calculating the posterior density $p(w \mid x)$ given the model evidence $p(x)$ which is hard to compute with hidden parameter set w [31]. The explicit rationale behind analytical intractability is that the evidence term is usually hard to compute.

As such, I minimize the ELBO that is equal to log marginal likelihood term, which is constant with respect to variational distribution $q(w)$, minus KL divergence. It reaches a maximum when $q(w) = p(w \mid x)$; the KL divergence is zero.

Typically, q will be constrained to a family of simpler distributions, and the ELBO is optimized to find the distribution in the family that is closest (in KL) to the true posterior. In this thesis, I follow the mean field assumption [32]. This approach assumes independence between hidden variables to factorize the family of variational distributions so that the true posterior is easy to compute. Then, the variational distributions have the factorized form:

$$q(V, Z, \mu, \sigma_l, \sigma_r) = \prod_{j=1}^{M} q(V_j) \prod_{i=1}^{N} q(Z_i) \prod_{j=1}^{M} \prod_{k=1}^{D} q(\mu_{jk}) q(\sigma_{ljk}) q(\sigma_{rjk}) \tag{5.9}$$

where $q(V_j)$ is a Beta prior with parameters γ_1 and γ_2, $q(Z_i)$ is a Multinomial prior with parameter ϕ and $q(\mu_{jk})$ is considered as a Gaussian distribution with mean m and variance Σ. I also define the variational distributions of σ_l and σ_r as Gaussian priors with mean ι and standard deviation υ:

$$\begin{aligned}
q(V_j) &= \text{Beta}(\gamma_{j1}, \gamma_{j2}) \\
q(Z_i) &= \text{Multi}(\phi_i) \\
q(\mu_{jk}) &= \mathcal{N}(m_{jk}, \Sigma_{jk}) \\
q(\sigma_{ljk}) &= \mathcal{N}(\iota_{ljk}, \upsilon^2_{ljk}) \\
q(\sigma_{rjk}) &= \mathcal{N}(\iota_{rjk}, \upsilon^2_{rjk})
\end{aligned} \tag{5.10}$$

For the proposed DPAGM and using the mean field assumption, the ELBO is:

$$\begin{aligned}
\mathcal{L}(\Theta) = \sum_{i=1}^{N} \big(E_q[\log p(X_i \mid Z_i, \mu, \sigma_l, \sigma_r)] + E_q[\log p(Z_i)] \big) + E_q[\log p(V)] \\
+ E_q[\log p(\mu)] + E_q[\log p(\sigma_l)] + E_q[\log p(\sigma_r)] - E_q[\log q(V, Z, \mu, \sigma_l, \sigma_r)]
\end{aligned} \tag{5.11}$$

By applying Eq. (5.9) to each factor, I obtain the optimal solutions for the factors of the variational posteriors. I next present the explicit coordinate ascent variational inference (CAVI) to optimize the ELBO in Eq. (5.9) where I find the updates of the variational parameters of stick lengths V and indicators Z are:

$$\phi_{ij} = \frac{r_{ij}}{\sum_j r_{ij}} \tag{5.12}$$

$$
\begin{aligned}
r_{ij} = \exp\big\{ & E_q[\log V_j] + \sum_{m=1}^{j-1} E_q[\log(1 - V_m)] \\
& - \sum_{k}^{D} E_q[\log(\sigma_{ljk} + \sigma_{rjk})] \\
& - \sum_{k, X_{ik} < \mu_{jk}}^{D} \frac{X_{ik}^2 + E_q[\mu_{jk}^2] - 2X_{ik}E_q[\mu_{jk}]}{2E_q[\sigma_{ljk}^2]} \\
& - \sum_{k, X_{ik} \geq \mu_{jk}}^{D} \frac{X_{ik}^2 + E_q[\mu_{jk}^2] - 2X_{ik}E_q[\mu_{jk}]}{2E_q[\sigma_{rjk}^2]} \big\}
\end{aligned}
\tag{5.13}
$$

$$
\gamma_{j1} = 1 + \sum_{i=1}^{N} \phi_{ij} \quad \gamma_{j2} = \alpha + \sum_{i=1}^{N} \sum_{m=j+1}^{M} \phi_{im} \tag{5.14}
$$

The expectation used in the calculation of updates is:

$$
\begin{aligned}
& q(Z_i = j) = \phi_{i,j} \\
& q(Z_i > j) = \sum_{m=j+1}^{M} \phi_{i,m} \\
& E_q[\log(V_j)] = \Psi(\gamma_{j,1}) - \Psi(\gamma_{j,1} + \gamma_{j,2}) \\
& E_q[\log(1 - V_j)] = \Psi(\gamma_{j,2}) - \Psi(\gamma_{j,1} + \gamma_{j,2})
\end{aligned}
\tag{5.15}
$$

where $\Psi(\cdot)$ denotes the digamma function that arises from the derivative of the log normalization factor in the Beta distribution. The variational parameters of μ can be obtained as:

$$
\begin{aligned}
\Sigma_{jk} &= \big(\sum_{i, X_{ik} < \mu_{jk}}^{N} \frac{\phi_{ij}}{E_q[\sigma_{ljk}^2]} + \sum_{i, X_{ik} \geq \mu_{jk}}^{N} \frac{\phi_{ij}}{E_q[\sigma_{rjk}^2]} + r \big)^{-1} \\
m_{jk} &= \Sigma_{jk} \big(\sum_{i, X_{ik} < \mu_{jk}}^{N} \phi_{ij} \frac{X_{ik}}{E_q[\sigma_{ljk}^2]} + \sum_{i, X_{ik} \geq \mu_{jk}}^{N} \phi_{ij} \frac{X_{ik}}{E_q[\sigma_{rjk}^2]} + \lambda r \big)
\end{aligned}
\tag{5.16}
$$

I consider the BBVI method proposed in chapter 4 to infer the parameters since the black box method can easily extends to different models and usually exactly captures the probability density.

However, in practice, the high variance gradient function would impede the convergence. Thereby, I need to develop variance control method to effectively employ inference. The expectations that appear in the above formulas are calculated by:

$$E_q[\mu_{jk}] = m_{jk} \quad E_q[\mu_{jk}^2] = m_{jk}^2 + \Sigma_{jk} \tag{5.17}$$

$$E_q[\sigma_{ljk}] = \iota_{ljk} \quad E_q[\sigma_{ljk}^2] = \iota_{ljk}^2 + \upsilon_{ljk}^2 \tag{5.18}$$

$$E_q[\sigma_{rjk}] = \iota_{rjk} \quad E_q[\sigma_{rjk}^2] = \iota_{rjk}^2 + \upsilon_{rjk}^2 \tag{5.19}$$

I use Jensen's Inequity to approxmiate the $E_q[\log(\sigma_{ljk} + \sigma_{rjk})]$ by replacing with the upper bound:

$$E_q[\log(\sigma_{ljk} + \sigma_{rjk})] \leq \log(E_q[\sigma_{ljk} + \sigma_{rjk}]) = \log(\iota_{ljk} + \iota_{rjk}) \tag{5.20}$$

5.2.2 Variance Control

To tackle the high variance issue, I introduce an accessible technique to reduce the variance of stochastic gradients for variational inference [39]. I adopt a reparameterization trick that omits the score function from the derivatives and presents a new gradient estimator with zero variance. Theoretically, the ELBO will have low variance when $q(\sigma \mid \theta) = p(\sigma \mid x)$, i.e. the variational distribution precisely approximates the true posterior:

$$\begin{aligned} \hat{\mathcal{L}}_{MC}(\sigma) &= \log p(x, \sigma) - \log q(\sigma \mid \theta) \\ &= \log p(\sigma \mid x) + \log p(x) - \log q(\sigma \mid \theta) \\ &= \log p(x) = \text{const} \end{aligned} \tag{5.21}$$

Specifically, the variance of the full Monte Carlo estimator of the ELBO $\hat{\mathcal{L}}_{MC}$ will exactly become zero. Its value is constant and the samples z are independent and identically distributed according to the variational distribution $z \overset{iid}{\sim} q(\sigma \mid \theta)$. This suggests that Eq. (5.21) is preferred when I believe that $q(\sigma \mid x) \approx p(\sigma \mid x)$.

Using the reparameterization in [40], I can decompose the gradient estimator of the ELBO. I represent the sample σ from $q(\sigma \mid \theta)$ as deterministic function parameterized by

θ and a random variable ϵ with the independent marginal distribution $p(\epsilon)$. Because of the diagonal Gaussian distribution, the representation is $\sigma_{ljk} = \iota_{ljk} + \upsilon_{ljk}\epsilon$ and $\sigma_{rjk} = \iota_{rjk} + \upsilon_{rjk}\epsilon$. Noise variable ϵ is given a standard Gaussian distribution $\epsilon \sim \mathcal{N}(0, 1)$.

Under such a reparameterization of variable set σ, I decompose the total derivative (TD) of the integrand of the estimator as follows:

$$\begin{aligned}\hat{\nabla}_{TD}(\epsilon, \theta) &= \nabla_\theta \big[\log p(x, \sigma) - \log q(\sigma \mid \theta) \big] \\ &= \nabla_\theta \big[\log p(\sigma \mid x) + \log p(x) - \log q(\sigma \mid \theta) \big] \\ &= \nabla_\sigma \big[\log p(\sigma \mid x) + \log p(x) \big] \nabla_\theta t(\epsilon, \theta) \\ &\quad - \nabla_\theta \log q(\sigma \mid \theta) \end{aligned} \tag{5.22}$$

The reparameterization gradient estimator is divided into two components: the path derivative and the score function. The first part depends on the set of variational parameters θ, and the second term measures the log variational distribution $\log q$ without considering the explicit value σ as a function of θ. For stochastic gradient descent algorithm to converge, I require an unbiased estimator of its gradient. By construction, the gradient estimator of w is unbiased. As the score function term has a zero expected value, I can just simply exclude score function term without biasing the stochastic gradients.

Considering the assumption $q(\sigma \mid \theta) = p(\sigma \mid x)$, the path derivative component reaches zero when variational distribution is exactly equal to the true distribution of latent variables. Thus, I get a desirable reparameterized gradient estimator of the ELBO whose variance approaches to zero when as $q(\sigma \mid \theta)$ gradually gets closer to $p(\sigma \mid x)$.

5.3 Feature Selection Approach

The main purpose of feature selection is to find the most informative feature set that better discriminate groups and alleviate the noise influence. In this section, I introduce the concept of feature saliency which considers feature selection as a parameter estimation problem [10]. Feature saliencies take into consideration the potential presence of irrelevant features and distinguish the noise each feature contains which can used to mitigate the influence of redundant features.

I consider a feature as relevant if it follows the mixture-dependent distribution AGD;

else follows a mixture-independent background distribution and be independent of the cluster labels. In this chapter, I propose a Gaussian assumption for the irrelevant distribution with parameter mean μ^{irr} and variance τ^{irr}. The prior of mean variable μ^{irr} is defined as a Gaussian distribution and variance τ_{jk}^{irr} follows an inverse Gamma distribution [41].

$$p(\mu_{jk}^{irr} \mid \lambda^{irr}, r^{irr}) \sim \mathcal{N}(\lambda^{irr},\, r^{irr}) \qquad p(\tau_{jk}^{irr} \mid v_0, w_0) \sim \gamma(v_0, w_0) \tag{5.23}$$

I use a series of latent indicator variables $\varphi = (\varphi_1, \ldots, \varphi_D)$ to represent the assignments of relevancy, where $\phi_k = 1$ if a feature is relevant; otherwise, $\phi_k = 0$. Thus, I represent the mixture density function from Eq. (5.5) as follows:

$$p(X \mid \Theta, \xi^{irr}, \varphi) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j \prod_{k=1}^{D} \left[p(X_{ik} \mid \xi_{jk})^{\varphi_k} p(X_{ik} \mid \xi_{jk}^{irr})^{1-\varphi_k} \right] \tag{5.24}$$

where the $\xi^{irr} = (\xi_1^{irr}, \ldots, \xi_M^{irr})$ represents the set of parameters for background Gaussian distribution with $\xi_j^{irr} = (\mu_j^{irr}, S_j^{irr})$, $\mu_j = (\mu_{j1}, \ldots, \mu_{jD})$, $\tau_j = (\tau_{j1}, \ldots, \tau_{jD})$. μ_{jk} and τ_{jk} represent the mean and variance for kth dimensional shared Gaussian distribution.

I define the feature saliency $\mathrm{P} = (\varrho_1, \ldots, \varrho_M)$ such that $\varrho_j = (\varrho_{j1}, \ldots, \varrho_{jD})$. $\varrho_{jk} = p(\varphi_j = 1)$ represents the probability that the kth feature is relevant for component j. Hence, the feature saliency P is associated with the Bernoulli prior over missing relevancy label φ and given a Beta prior with hyperparameters a_0 and b_0:

$$p(\varrho \mid a_0, b_0) = \text{Beta}(a_0, b_0) \tag{5.25}$$

I can then rewrite the likelihood function after introducing the feature saliency P:

$$p(X_i \mid \Theta_F) = \sum_{j=1}^{M} \pi_j \prod_{k=1}^{D} \left(\varrho_{jk} p(X_{ik} \mid \xi_{jk}) + (1 - \varrho_{jk}) p(X_{ik} \mid \xi_{jk}^{irr}) \right) \tag{5.26}$$

where $\Theta_F = (\pi, \xi, \mathrm{P}, \xi^{irr})$ is the complete set of parameters of mixture model. Eq (5.26) offers sound generative interpretation. First, the model selects the component j by sampling from a Multinomial distribution with mixing proportion $(\pi_1, \ldots, \pi_k)$. Then, each feature $k = 1, \ldots, D$ follows a Bernoulli prior with feature saliency ρ_{jk}; if successful, I consider the relevant mixture component $p(X_{ik} \mid \xi_{jk})$ generating feature k; otherwise, the background component $p(X_{ik} \mid \xi_{jk}^{irr})$ will be used. Therefore, I consider the model of previous section as a special case when all of the features are relevant.

For latent relevancy variable $\varphi_i = (\varphi_{i1}, \ldots, \varphi_{iM})$, where $\varphi_{ij} = (\varphi_{ij1}, \ldots, \varphi_{ijD})$ and φ_{ijk} indicates whether the feature k is relevant for component j given the observation X_i, $\varrho_j = p(\varphi_{ijk} = 1)$ represents the prior probability that the feature k is relevant for component j given observation X_i. Given observations, the model can be written hierarchically with a set of distribution parameters, the allocation variables, and the relevancy variables as follows:

$$p(X \mid Z, \varphi, \xi, \xi^{irr}) = \prod_{i=1}^{N} \prod_{j=1}^{M} [\prod_{k=1}^{D} p(X_{ik} \mid \xi_{jk})^{\varphi_{ijk}} p(X_{ik} \mid \xi_{jk}^{irr})^{1-\varphi_{ijk}}]^{\mathbb{I}(Z_i=j)} \tag{5.27}$$

$$p(\varphi \mid \mathrm{P}) = \prod_{i=1}^{N} \prod_{j=1}^{M} [\prod_{k=1}^{D} \varphi_{ijk}^{\varphi_k} (1 - \varphi_{ijk})^{1-\varphi_k}]^{\mathbb{I}(Z_i=j)} \tag{5.28}$$

Given the representation of the ELBO in Eq. (5.11), I rewrite the ELBO after introducing the relevancy parameters as follows:

$$\begin{aligned} \mathcal{L}(\Theta) = \sum_{i=1}^{N} (E_q[\log p(X_i \mid Z_i, \mu, \sigma_l, \sigma_r, \varrho, \mu^{irr}, \tau^{irr})] + E_q[\log p(Z_i)]) + E_q[\log p(V)] \\ + E_q[\log p(\mu)] + E_q[\log p(\sigma_l)] + E_q[\log p(\sigma_r)] + E_q[\log p(\varrho)] \\ + E_q[\log p(\mu^{irr})] + E_q[\log p(\tau^{irr})] - E_q[\log q(V, Z, \mu, \sigma_l, \sigma_r, \varrho, \mu^{irr}, \tau^{irr})] \end{aligned} \tag{5.29}$$

Because the relevant distribution is the same as the original DPAGM model, I do not need to change the updating process proposed in Section 5.2. However, I must build an additional irrelevant component and alter the inference of latent variable Z because I introduced the relevancy saliency in this section. Therefore, I propose Variational EM framework to learn the parameters of the DPAGM with feature selection approach [42]. I also adopt the factorized representation of ϱ, μ^{irr}, and τ^{irr}. Feature saliency ϱ_{jk} has support over [0, 1] and as considered naturally to follow a Beta distribution with common hyperparameters a and b. For the irrelevant Gaussian distribution, I consider a Gaussian prior for mean μ^{irr} and Gamma prior for precision τ^{irr}.

$$q(\varrho, \mu^{irr}, \Sigma^{irr}) = \prod_{j}^{M} \prod_{k}^{D} q(\varrho_{jk} \mid a, b) q(\mu_{jk}^{irr} \mid m^{irr}, \Sigma^{irr}) q(\tau_{jk}^{irr} \mid v_{jk}, w_{jk}) \tag{5.30}$$

$$
\begin{aligned}
q(\mu_{jk}^{irr} \mid m^{irr}, \Sigma^{irr}) &= \mathcal{N}(m^{irr}, \Sigma^{irr}) \\
q(\tau_{jk}^{irr} \mid v, w) &= \gamma(v, w) \\
q(\varrho \mid a, b) &= \text{Beta}(a, b)
\end{aligned} \tag{5.31}
$$

The expectation update process is then defined as follows:

$$
\begin{aligned}
A &= \log p\big(X_{ik} \mid Z_i = j, \mu, \sigma_l, \sigma_r, \big) + \log p\big(\mu, \sigma_l, \sigma_r \mid Z_i = j\big) + \log \varrho_{jk} \\
&\quad - \log q\big(\mu, \sigma_l, \sigma_r \mid Z_i = j\big) \\
B &= \log p\big(X_{ik} \mid Z_i = j, \mu^{irr}, \tau^{irr}\big) + \log p\big(\mu^{irr}, \tau^{irr} \mid Z_i = j\big) + \log(1 - \varrho_{jk}) \\
&\quad - \log q\big(\mu^{irr}, \tau^{irr} \mid Z_i = j\big)
\end{aligned} \tag{5.32}
$$

Using the above representation, $q(\varphi_{ijk} = 1 \mid Z_i = j)$ is written as:

$$
q(\varphi_{ijk} = 1 \mid Z_i) = \frac{\exp(A)}{\exp(A) + \exp(B)} \tag{5.33}
$$

and $q(\varphi_{ijk} = 0 \mid Z_i = j) = 1 - q(\varphi_{ijk} = 1 \mid Z_i = j)$. The expected values of φ are defined as:

$$
\begin{aligned}
E_q[\varphi_{ijk}]^1 &= q(\varphi_{ijk} = 1 \mid Z_i = j) \\
E_q[\varphi_{ijk}]^0 &= q(\varphi_{ijk} = 0 \mid Z_i = j)
\end{aligned} \tag{5.34}
$$

If I define the quantity:

$$
\begin{aligned}
r_{ij} = E_q[\log V_j] + \sum_{m=1}^{j-1} E_q[\log(1 - V_m)] + \sum_{k}^{D} E_q[\varphi_{ijk}]^1 \Big(& \log p\big(X_{ik} \mid Z_i = j, \mu, \sigma_l, \sigma_r, \big) + \log \varrho_{jk} \\
+ \log p\big(\mu, \sigma_l, \sigma_r \mid Z_i = j\big) - \log q\big(\mu, \sigma_l, \sigma_r \mid Z_i = j\big) \Big) + E_q[\varphi_{ijk}]^0 \Big(& \log p\big(X_{ik} \mid Z_i = j, \mu^{irr}, \tau^{irr}\big) \\
+ \log(1 - \varrho_{jk}) + \log p\big(\mu^{irr}, \tau^{irr} \mid Z_i = j\big) - \log q\big(\mu^{irr}, \tau^{irr} \mid Z_i = j\big) \Big)
\end{aligned} \tag{5.35}
$$

The variational parameters ϕ_{ij} can be updated by normalizing the quantity r_{ij}:

$$
\phi_{ij} = \frac{r_{ij}}{\sum_j r_{ij}} \tag{5.36}
$$

Using CAVI to update variational parameters, I get:

$$\begin{aligned}
\Sigma_{jk}^{irr} &= \big(\sum_{i}^{N} \phi_{ij} E_q[\tau_{ljk}^{irr}] + r^{irr}\big)^{-1} \\
m_{jk}^{irr} &= \Sigma_{jk}^{irr}\big(\sum_{i}^{N} \phi_{ij} X_{ik} E_q[\tau_{ljk}^{irr}] + \lambda^{irr} r^{irr}\big)
\end{aligned} \tag{5.37}$$

$$\begin{aligned}
v_{jk} &= \frac{N_j + v_{0jk}}{2} \\
w_{jk} &= \frac{2}{v_{0jk} w_{0jk} + \sum_{i:Z_i=j}^{N} (X_{ik} - \mu_{jk})^2}
\end{aligned} \tag{5.38}$$

$$\begin{aligned}
a_{jk} &= a_{0jk} + \sum_{i=1}^{N} E_q[\varphi_{ijk}]^1 \phi_{ij} \\
b_{jk} &= b_{0jk} + \sum_{i=1}^{N} E_q[\varphi_{ijk}]^0 \phi_{ij}
\end{aligned} \tag{5.39}$$

5.4 Complete Learning algorithm

An important aspect when applying variational inference is the convergence assessment. In this chapter, I trace the convergence systematically by monitoring the variational lower bound and find the variational parameters vary narrowly when the ELBO difference is less than 10^{-3} between epoches. The variational inference for DPAGM is summarized in Algorithm 4.

In the DPAGM model, I need to test the appropriate truncation level which depends on the data structure. Usually, I first set a truncated component number, and then rely on variational inference to infer a smaller number to model the observations. Although an infinite mixture model may appear complicated because of the number of involved parameters, the final model remains concise as the self-correcting component reduction process cuts the least effective components and leave well-separated clusters. Because I consider BBVI method to infer parameters σ_l and σ_r, I also need to adopt adequate hyparameters sample size and epoch number to ensure convergence. The complete algorithm for DPAGM with feature selection process can be summarized in Algorithm 5.

Algorithm 4 Dirichlet Process Asymmetric Gaussian Mixture

procedure
    Initialization:
    Initialize the truncation level M and hyperparameters α, λ and r, m_{ljk}, s_{ljk}, m_{rjk} and s_{rjk}.
    Initialize variational parameters ϕ, γ_1, γ_2, μ and Σ.
    repeat:
    Update local variational parameters ϕ_{ij} using Eq. (5.12) and Eq. (5.13).
    Update global variational parameters γ_{j1}, γ_{j2}, μ_{jk} and Σ_{jk} using Eq. (5.14) and Eq. (5.16).
    Update variational parameters of global latent variables σ_{ljk} and σ_{rjk} by Black Box Variational Infernece from Section 4.2.2, with variance control approach from Section 5.2.2.
    The convergence criteria is reached when the difference of the current value of ELBO and previous value is less than 10^{-3}.
    until convergence
Compute the expected value of stick length V_j as $E_q[V_j] = \gamma_{j1}/(\gamma_{j1} + \gamma_{j2})$ and the value of mixing proportions using Eq. (5.3)
Detect the optimal number of components M by eliminating the components with small mixing coefficients close to 0.

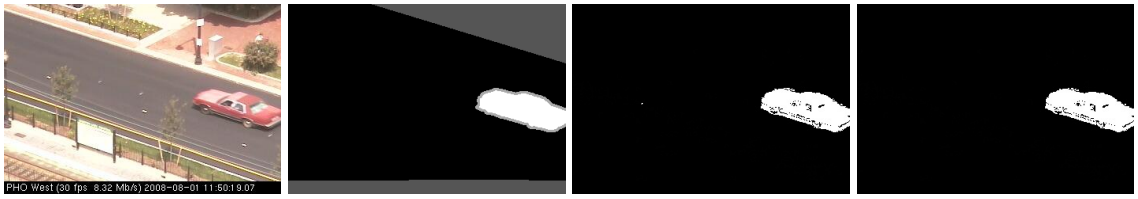
For the initialization step, I start by assuming all of the features are relevant, and update relevancy assignments and feature saliencies by variational EM algorithm. In above process, the model strives to take advantage of discriminative features for clustering the data into diverse components.

(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.1: Sample results from fall video sequences.

Algorithm 5 Dirichlet Process Asymmetric Gaussian Mixture with Feature Selection

procedure
    Initialization:
    Initialize the truncation level M and hyperparameters α, λ, r, m_{ljk}, s_{ljk}, m_{rjk}, s_{rjk}, λ^{irr}, r^{irr}, v_0, w_0, a_0 and b_0.
    Initialize variational parameters ϕ, γ_1, γ_2, μ, Σ, m^{irr}, Σ^{irr}, v, w, a and b.
    repeat:
    VB E-step:
    Update latent relevancy assignments φ using Eq. (5.32) and Eq. (5.33)
    Update local variational hyperparameter ϕ_{ij} using Eq. (5.34) and Eq. (5.35).
    VB M-step:
    Update variational parameters γ_{j1}, γ_{j2}, μ_{jk}, Σ_{jk}, m^{irr}_{jk}, Σ^{irr}_{jk}, v_{jk}, w_{jk}, a_{jk} and b_{jk} using Eq. (5.14), Eq. (5.16), Eq. (5.36), Eq. (5.37), Eq. (5.38), and Eq. (5.39).
    Update varaitional parameters from latent variables σ_{ljk} and σ_{rjk} by Black Box Variational Infernece from Section 4.2.2, with variance control approach from Section 5.2.2.
    The convergence criteria is reached when the difference of the current value of ELBO and previous value is less than 10^{-3}.
    until convergence
    Compute the expected value of stick length V_j as $E_q[V_j] = \gamma_{j1}/(\gamma_{j1} + \gamma_{j2})$ and the value of mixing proportions using Eq. (5.3)
    Detect the optimal number of components M by eliminating the components with small mixing coefficients close to 0.



(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.2: Sample results from boulevard video sequences.

5.5 Experimental setup and results

5.5.1 Background subtraction setup

In this section, I employ the proposed DPAGM model for image background subtraction with a pixel-level evaluation approach as in [14]. The background modeling starts off by constructing the model using the proposed DPAGM. After applying the learning algorithm for the model, I discriminate between the mixture components for the representation of foreground and background pixels for each of the new input frames.

Assume that a particular pixel of a video frame sequences $\mathcal{X}$ has P pixels as $\mathcal{X} = (\mathcal{X}_1, \ldots, \mathcal{X}_P)$ then each pixel $\mathcal{X}_p$ is assigned as a foreground or background pixel with respect to the trained DPAGM $p(\mathcal{X}_p \mid \Theta) = \prod_{i=1}^{N} \sum_{j=1}^{M} \pi_j p(\mathcal{X}_{pi} \mid \xi_j)$. Usually, background objects maintain persistent appearance with relatively low variance as they usually maintain static status compared with movable foreground objects. Besides, the background always appears frequent in a given pixel but a foreground object appears abruptly with a low rate of occurrence. Heuristically, I consider components that occurs frequently, i.e. with high π value and with a low standard deviation σ, as the background of the scene.

(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.3: Sample results from traffic video sequences.

Accordingly, I use the fitness value of $\pi_j/(|\sigma_{lj}| + |\sigma_{rj}|)$ as a criteria to rank the mixture

(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.4: Sample results from abandonedBox video sequences.

components, where π_j is the mixing proportions for component j, $|\sigma_{lj}|$ and $|\sigma_{rj}|$ are the respective norms of left and right standard deviations of the jth component. The fitness value increases both as a distribution gains more evidence and as it remains stable. The first B number of components are chosen as the background model estimated as:

$$B = \underset{b}{\operatorname{argmin}} \sum_{j=1}^{b} \pi_j > T \tag{5.40}$$

where the threshold T is a measure of the minimum share of the data that should be counted as the background in a given pixel sequence. The rest of the observations are defined as foreground scene. Thereby, the most probable distribution remains on the top with the lowest one replaced by new distribution.

The pixels match a given distribution when they are no more than K standard deviation away from the distribution. In the case, the matching occurs when it is less than 3 left or right standard deviations given a distribution. Then, pixels are more than 3 standard deviations away are considered foreground. The first mixture that matches the pixel value will be updated by the following equation:

(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.5: Sample results from library video sequences.

$$\pi_{jt} = (1 - \beta)\pi_{j(t-1)} + \beta \mathcal{M}_{jt} \tag{5.41}$$

(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.6: Sample results from corridor video sequences.

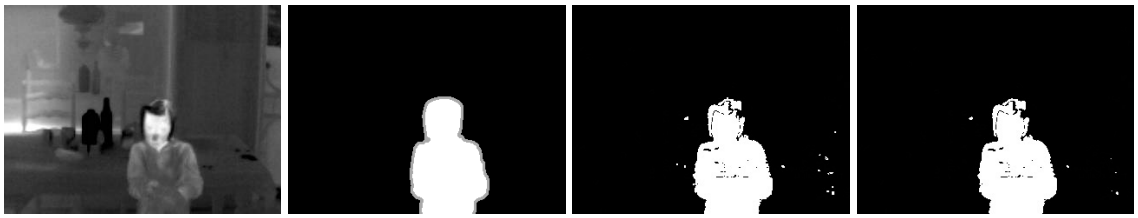
$$\mu_{jt} = (1-\beta)\mu_{j(t-1)} + \rho\mathcal{X}_t \tag{5.42}$$

$$\begin{aligned} \sigma^2_{ljt} &= (1-\beta)\sigma^2_{lj(t-1)} + \rho(\mathcal{X}_t - \mu_{jt})^2 \quad \text{if } \mu_{jt} < \mathcal{X}_t \\ \sigma^2_{rjt} &= (1-\beta)\sigma^2_{rj(t-1)} + \rho(\mathcal{X}_t - \mu_{jt})^2 \quad \text{if } \mu_{jt} \geq \mathcal{X}_t \end{aligned} \tag{5.43}$$

where β determines the speed of parameters change and indicator symbol $\mathcal{M}_{jt}$ denotes whether the pixel matches kth component. π_{jt}, μ_{jt}, σ_{ljt} and σ_{rjt} are expected values of the weight, mean and standard deviations of the jth component of DPAGM at frame t, which can be computed with Eq. (5.17), Eq. (5.18), and Eq. (5.19). The estimated values of mixing weights π are derived from the expected value of stick length V_j as $E_q[V_j] = \gamma_{j1}/(\gamma_{j1} + \gamma_{j2})$ and the stick-breaking process generation from Eq. (5.3).

Finally, ρ is defined as:

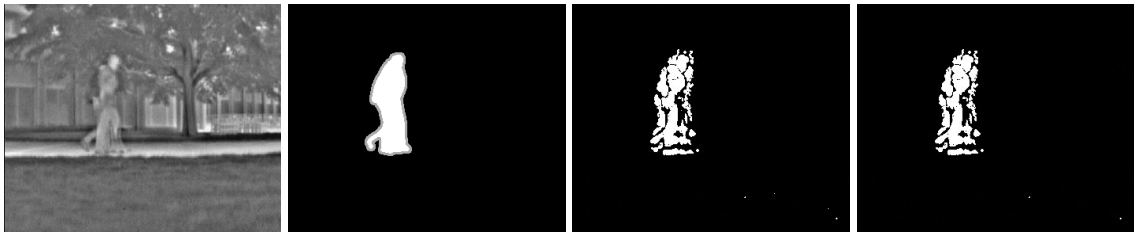
$$\rho = \beta p(\mathcal{X}_t \mid \mu_{jt}, \sigma_{ljt}, \sigma_{rjt}) \tag{5.44}$$



(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.7: Sample results from diningRoom video sequences.

where $p(\mathcal{X}_t \mid \mu_{jt}, \sigma_{ljt}, \sigma_{rjt})$ represents the AGD function with mean μ_{jt} and standard deviations σ_{ljt} and σ_{rjt}. If a new pixel value matches against all existing distributions,



(a) The input frame of video. (b) The groundtruth frame of video. (c) Result frame by DPAGM. (d) Result frame by DPAGM+FS.

Figure 5.8: Sample results from park video sequences.

the least probable distribution will be replaced by a new distribution with initially low frequency, high standard deviation value, and the same mean as the current value. For the proposed feature selection, the parameter update procedures remain same as the original process proposed in [14] because the density function of irrelevant features is also assumed as Gaussian.

5.5.2 Results and discussion

I apply the proposed approach on the Change Detection 2014 dataset (CDnet 2014) [15]. This dataset spans 53 realistic camera-captured videos totalling 160,000 frames organized in 11 categories that describe a wide range of change detection tasks. Each category contains 4 to 6 video sequences. The videos had been recorded using different cameras from low-resolution Internet Protocol (IP) ones, through higher resolution consumer grade camcorders, and commercial pan-tilt-zoom (PTZ) cameras to thermal cameras. Consequently, spatial resolutions of the videos vary from 320×240 to 720×486. The level of noise and compression artifacts varies considerably from one to another due to the diverse lighting conditions and compression settings. Low-end IP cameras suffer from apparent radial distortion. Different cameras may contain different bias and global brightness fluctuations as a result of the employment of different white balancing methods and automatic exposure adjustment.

In this chapter, I incorporated several video sequences under diverse surveillance settings. I selected 4 video sequences (two outdoor and two indoor) in thermal category, which were captured by far-infrared cameras, to evaluate infrared image background detection task. These video sequences include typical thermal artifacts, heat reflection and camouflage effects. As for visible image task, I have chosen 4 intricate sequences from the

Dynamic Background, Camera Jitter, Intermittent Object Motion and Shadow categories:

- Dynamic Background category depicts scenes with strong background motion.
- Camera Jitter category contains videos captured by unstable cameras and the shiver magnitude varies from one video to another.
- Shadow category consists of videos exhibiting strong as well as faint shadows. Some shadows are fairly narrow as large objects occupy most of the scene and some are cast by moving objects.
- Intermittent Object Motion category includes videos with scenarios known for causing ghosting artifacts in the detected motion, i.e., the moving object suddenly stop for a while, after which they start moving again.

In order to assess the performance of the proposed approach DPAGM and DPAGM with feature selection (DPAGM+FS), I implement 7 other state-of-the-art methods. These algorithms can be grouped into two main categories: pixel based and nonparametric Kernel Density Estimation (KDE) methods. For pixel based methods, I have chosen the well-known Gaussian mixture model based background subtraction introduced by Sauffer et al. [14] and Zivkovic [16], and also compare with the approach of KaewTrakulPong et al. [43], and Evangelio et al. [44]. I also include finite mixture of asymmetric Gaussian distributions proposed by Elguebaly et al. [17]. Others are from Elgammal et al. [45] and Nonaka et al [46].

In this application, I specified the initial truncation number M as 20, the distribution matching factor $K = 3$ and the threshold factor $T = 0.8$. The concentration parameter α is set as $1/M$ and feature saliency variable φ is fixed to 1 as I assume all of features are relevant initially. The hyperparameters mentioned in Algorithm 4 and Algorithm 5 are randomly drawn from their support. The relevant and irrelevant mean parameters μ and μ^{irr} are sampled from Gaussian prior with a mean calculated by the average value of the observations. The left and right standard deviations σ_l and σ_r are sampled from Gaussian distribution with high mean value and irrelevant τ^{irr} is sampled from inverse Gamma distribution with shape parameter 2 and mean parameter 0.5.

For quantitative analysis, recall and precision are utilized to assess the performance. Recall and precision are widely used in pattern recognition and image processing for quantitative analysis of binary classification. The recall and precision metrics are defined in Eq. 2.25 and Eq. 2.26:

In this case, I can compute recall results by dividing the number of correctly identified foreground pixels by the number of foreground pixels in ground truth which can be seen as a measure of fidelity. Precision is calculated by dividing the number of correctly identified foreground pixels by the number of foreground pixels detected which can be seen as a measure of completeness of foreground. The recall and precision are calculated based on the averages on all the evaluated frames. The results for the selected method can be seen in Table. 5.1 and Table. 5.2. The samples of input frame, groundtruth frame and results frame for all sequences are shown in Figure. 5.1 to Figure. 5.8.

From Table. 5.1 and Table. 5.2, it can be observed that the proposed DPAGM and DPAGM with feature selection approach remarkably outperform other algorithms in terms of precision metrics while also giving relatively higher recall results. According to the definition of precision, the results shows a relatively lower number of FP compared with TP which indicates DPAGM provides robust detection compared with other methods. In Table. 5.2, DPAGM gives better precision results although other methods imprecisely capture the foreground pixels. For infrared images, the method achieves relatively lower precision but with higher recall and completely detect foreground objects.

DPAGM also provides higher recall results when the precision is close to other approaches; otherwise, it significantly improves recall. Hence, the proposed approach usually preserves the completeness of the foreground objects with similar exactness. In the Library video sequence, the approach is the only one to completely detect the foreground objects with approximately similar precision results with other methods barely discriminating between foreground pixels and the background. This demonstrates how the proposed method can notably improve recall without sacrificing too much fidelity.

The reason why the approach performs better in terms of precision without sacrificing recall results is because the AGD is capable of modeling complex asymmetric characteristics of the observations for completely incorporating the structure of the objects. The higher flexibility with DPAGM results in mixture models that are more adaptive and give higher precision results. I also include DP which could automatically allocate observations precisely through determination of the number of components for an exact representation. These advantages show how the approach provides a more accurate and adaptive background model. Therefore, the method does not detect many incorrectly distinguished pixels as foreground with a high proportion of groundtruth's foreground pixels discerned.

Table 5.1: Experimental results for the background subtraction task on infrared image .

	Stauffer	Zivkovic	Evangelio	Elgammal	Nonaka	Elguebaly	DPAGM	DPAGM+FS
Corridor								
Rec.(%)	82.52	83.26	84.68	83.20	56.00	89.24	83.86	81.51
Prec.(%)	80.75	83.93	84.68	88.0	89.55	90.72	91.42	92.14
Library								
Rec.(%)	28.00	28.68	30.23	92.20	8.07	31.34	94.01	93.62
Prec.	84.76	81.76	93.86	97.14	96.35	94.66	82.49	84.81
Park								
Rec.(%)	63.96	59.30	39.98	60.81	89.03	64.00	63.34	60.11
Prec.	80.66	85.07	92.57	85.85	80.42	88.14	89.53	92.12
Dining Room								
Rec.(%)	70.21	69.43	77.45	75.74	40.11	79.57	85.77	84.41
Prec.(%)	93.37	92.31	94.03	88.42	95.55	93.74	92.10	93.49

Based on the results shown in Table. 5.1 and Table. 5.2, the DPAGM with feature selection algorithm greatly outperforms in precision metric at the expense of recall. The results indicate that the feature selection algorithm could detect foreground pixels more thoroughly compared with original DPAGM but it also misinterprets more background pixels as foreground. Simultaneous feature selection clustering method prefers to identify more pixels as foreground because it always discriminate features to find the most informative to represent clusters. Because of the sensitivity of the proposed DPAGM with feature selection approach, it is vulnerable to a rise of noise, such as uneven illumination. Thus, the approach sacrifices recall to improve the overall performance.

Table 5.2: Experimental results for the background subtraction task on visible image .

	Stauffer	Zivkovic	Evangelio	Elgammal	Nonaka	Elguebaly	DPAGM	DPAGM+FS
Fall								
Rec.(%)	88.38	85.60	84.79	89.21	81.75	89.14	68.40	66.75
Prec.(%)	3.91	28.17	40.33	18.75	32.12	66.12	92.15	95.94
Boulevard								
Rec.(%)	83.21	79.77	75.82	77.61	58.73	79.54	60.47	59.58
Prec.(%)	40.02	43.79	65.21	33.59	70.57	61.13	84.89	86.09
Traffic								
Rec.(%)	76.47	73.68	76.76	85.89	87.63	78.46	78.71	76.79
Prec.(%)	58.61	52.58	64.57	44.31	68.88	66.10	76.79	77.65
Abandon Box								
Rec.(%)	45.74	45.64	42.23	87.45	40.54	45.18	59.95	57.31
Prec.(%)	65.52	62.14	66.53	53.73	79.67	67.41	81.25	83.42

Chapter 6

Variational Inference for Nonparametric Hierarchical Infinite Mixture with Asymmetric Gaussian Distribution

In this chapter, I present a Variational inference framework for hierarchical Bayesian nonparametric model. Specifically, I propose the DP and PYP to endow nonparametric property and extend to hierarchical cases. I illustrate the models and learning algorithms with the challenging task of image clustering.

6.1 Hierarchical infinite asymmetric Gaussian mixture

In this section, I briefly introduce the hierarchical DP mixture model of AGD, which may also be referred to as the hierarchical infinite asymmetric Gaussian mixture model.

6.1.1 Hierarchical Dirichlet process mixture model

The DP is a parameterized stochastic process with a positive scaling factor and base distribution. The DP forms a distribution over discrete distribution that place its mass on a countably infinite collection of atoms. The base distribution places location of atoms and the concentration variable controls the range of the mass spreading around atoms [36].

The hierarchical Dirichlet process (HDP) constructs a global random probability measure G_0 and an indexed collection of random measures $\{G_j\}$. Thus, this model binds a collection of group-level DPes at a single top-level DP [47]:

$$
\begin{aligned}
&G_0 \sim DP(\omega, H) \\
&G_j \sim DP(\alpha, G_0) \quad \text{for each } j,\ j \in \{1, \ldots, M\}
\end{aligned} \tag{6.1}
$$

where j is the index for each group of the observations, random measure G_j attached to jth group. A two-level HDP model can be defined as the following: given a grouped dataset X with M groups, each group is associated with a DP G_j, and this indexed set of DP G_j shares a global measure G_0 which is itself distributed according to a DP with the base distribution H and concentration ω as a result of the discreteness of the top-level DP.

In this chapter, the representation of the global-level base measure G_0 and each group measure G_j are formed by the stick-breaking process [38]. The stick-breaking process gives an explicit representation of the HDP which is based on two infinite sequences of independent and identically distributed random variables $\{V_k'\}$ and $\{\Omega_k\}$, for $k \in \{1, \ldots, \infty\}$. The stick-breaking construction of the global measure G_0 is defined as:

$$
\begin{aligned}
&\Omega_k \sim H \\
&V_k' \sim \text{Beta}(1, \omega) \\
&V_k = V_k' \prod_{l=1}^{k-1} (1 - V_l') \\
&G_0 = \sum_{k=1}^{\infty} V_k \delta_{\Omega_k}
\end{aligned} \tag{6.2}
$$

where $\{\Omega_k\}$ is a set of independent random variables drawn from global measure H and δ_{Ω_k} indicates a probability measure mass at Ω_k with proportion V_k. The stick-breaking proportions $\{V_k\}$ denotes the corresponding prevalence for atoms and satisfy the constraint $\sum_{k=1}^{\infty} V_k = 1$. The proportions are obtained by recursively cutting a unit length stick into an infinite number of pieces according to a series of stick-breaking lengths V'. This infinite collection of variables combine to construct a point on the infinite simplex. Since G_0 is the base distribution of the DP G_j and has the stick-breaking representation as shown in Eq. (6.2), G_j contains all of the atoms in G_0 with different weights.

Since the stick-breaking weights are closely coupled between two-level DP [47], this kind of construction does not allow an amendable closed form formula for variational inference. I apply another stick-breaking representation for each group-level DP G_j which allows for closed form update. [48]:

$$
\begin{aligned}
\varpi_{jt} &\sim G_0 \\
\pi'_{jt} &\sim \text{Beta}(1, \alpha) \\
\pi_{jt} &= \pi'_{jt} \prod_{s=1}^{t-1} (1 - \pi'_{js}) \\
G_j &= \sum_{t=1}^{\infty} \pi_{jt} \delta_{\varpi_{jt}}
\end{aligned} \tag{6.3}
$$

where $\delta_{\varpi_{jt}}$ represents the group-level Dirac delta measure concentrated at ϖ_{jt}, and $\{\pi_{jt}\}$ is a set of mixing weights which must be positive and sum to one. Since group-level atom ϖ_{jt} is distributed according to the base distribution G_0, each atom maps onto the base-level atoms Ω_k with probability V_k. Notice there may be multiple atoms ϖ_{jt} which map to a same top-level atom Ω_k.

Here, I introduce a series of latent indicator variables C_{jtk} to denote which global-level atom ϖ maps to. If the indicator $C_{jtk} = 1$, group-level atom ϖ_{jt} maps onto the global-level atom Ω_k which is indexed by k; otherwise, C_{jtk} assigned value 0. Accordingly, I have a mapping $\varpi_{jt} = \Omega_k^{C_{jtk}}$. Thereby, there is no need to explicitly maintain representation for group-level atom ϖ_{jt}. The indicator variable $C_{jt} = (C_{jtk})_{k=1}^{\infty}$ is distributed according to Multinomial distribution given stick parameter V as follows:

$$
p(C \mid V) = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \text{Multi}(V) = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} V_k^{C_{jtk}} \tag{6.4}
$$

Since V is a function to take a collection of V' and to return the mixing weights of each k component according to the stick-breaking construction of the DP in Eq. (6.2), I can rewrite Eq. (6.4) as:

$$
p(C \mid V') = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} [V'_k \prod_{l=1}^{k-1} (1 - V'_l)]^{C_{jtk}} \tag{6.5}
$$

The stick lengths V' are independently drawn from Beta distribution with concentration variable ω in Eq. (6.2). The realization of stick lengths V' are defined as:

$$
p(V' \mid \omega) = \prod_{k=1}^{\infty} \text{Beta}(1, \omega_k) = \prod_{k=1}^{\infty} \omega_k (1 - V'_k)^{\omega_k - 1} \tag{6.6}
$$

One of the most common applications of HDP is placing DP as nonparametric prior on the parameters of mixture model for grouped data. I then interpret the topic in HDP as a factor which is associated with the observation X_{ji}, where i index data point in the jth group of the dataset and $i \in \{1, \ldots, N\}$. HDP mixture model generates factor θ_{ji} corresponding to each observation X_{ji} and $\theta_j = (\theta_{j1}, ..., \theta_{jN})$ which is distributed according to the DP G_j. Then I can generate the observation X_{ji} from that factor. The likelihood function can be defined as:

$$\begin{aligned} \theta_{ji} \mid G_j &\sim G_j \\ X_{ji} \mid \theta_{ji} &\sim F(\theta_{ji}) \end{aligned} \tag{6.7}$$

where $F(\theta_{ji})$ indicates the distribution of the observation X_{ji} given factor θ_{ji}. The base distribution H of G_0 provides the prior for the factors θ_{ji}. Based on above setting, known as the HDP mixture model, each group j is associated with a mixture model, and the mixture components are shared among these mixture models since G_j contains all of the atoms in G_0.

Since each factor θ_{ji} is distributed according to G_j based on Eq. (6.7), it takes the value ϖ_{jt} with probability π_{jt}. I introduce another collection of latent indicator variables Z as follows:

$$p(Z \mid \pi) = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} \pi_{jt}^{Z_{jit}} \tag{6.8}$$

The indicator Z_{jit} represents which component θ_{ji} belongs to. $Z_{ji} = (Z_{jit})_{t=1}^{\infty}$ and each element Z_{jit} assigned as value 1 if θ_{ji} is allocated to component t and maps to the group-level atom ϖ_{jt}; otherwise, $Z_{jit} = 0$. I then have the mapping $\theta_{ji} = \varpi_{jt}^{Z_{jit}}$. Since group-level atom also maps to the global level atom Ω_k through the indicator variables C as well, I also write $\theta_{ji} = \varpi_{jt}^{Z_{jit}} = \Omega_k^{C_{jtk} Z_{jit}}$.

Since π is a function of π' according to the stick-breaking construction of the DP as shown in Eq. (6.3), I then have

$$p(Z \mid \pi') = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} [\pi'_{jt} \prod_{s=1}^{t-1} (1 - \pi'_{js})]^{Z_{jit}} \tag{6.9}$$

The prior distribution of π' is a specific Beta distribution with concentration α as described in Eq. (6.3) as

$$p(\pi' \mid \alpha) = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \text{Beta}(1, \alpha_{jt}) = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \alpha_{jt}(1 - \pi'_{jt})^{\alpha_{jt}-1} \tag{6.10}$$

The discreteness of the corpus-level draw G_0 can ensures that all the groups share the same set of factors. The group-level draw G_j inherits the factor from G_0, but weight them according to group-level specific factors according to mixing proportions. Combining these processes and representation, I form the generative process of the HDP is as follows:

1. Draw an infinite factor Ω_k from G_0, $k \in \{1, \ldots, \infty\}$.
2. Draw global stick length $V'_k \sim \text{Beta}(1, \omega)$, $k \in \{1, \ldots, \infty\}$.
3. For each group j, $j \in \{1, \ldots, M\}$:
   (a) Draw group-level topic assignment, $C_{jt} \sim \text{Multi}(V)$, $t \in \{1, \ldots, \infty\}$.
   (b) Draw group-level stick length $\pi'_{jt} \sim \text{Beta}(1, \alpha)$, $t \in \{1, \ldots, \infty\}$.
   (c) For each word n, $n \in \{1, \ldots, N\}$:
      i. Draw word indicator $Z_{ji} \sim \text{Multi}(\pi)$.
      ii. Draw the n th observation X_{ji}.

The infinite number of factors are drawn as in the DP. The global-level stick lengths V' describes probability distributions for these factors and drawn from Beta prior, which denote the relative prevalence across over the grouped data. At group level, sticks π' create a set of probabilities and topic indices C_j, drawn from π, attach each group-level stick length to a factor. This creates a group-level distribution over factors, and observations are then drawn as for HDP mixture model.

6.1.2 Hierarchical Pitman-Yor process mixture model

The PYP is also known as two-parameter Poisson-Dirichlet process [49]. The PYP is a generalization of the DP with an extra discount parameter γ_a in addition to the concentration parameter γ_b, and satisfying $0 < \gamma_a < 1, \gamma_b > -\gamma_a$. When $\gamma_a = 0$, it is the special case of DP with concentration parameter γ_b. Similar to DP, the sample drawn from PYP also

associated with discrete distribution with support of the base distribution H [**?**]. The Hierarchical Pitman-Yor process (HPYP) defines the global-level measure G_0 and group-level distribution G_j similar to HDP as shown in Eq. (6.11).

I can describe HPYP by applying the stick-breaking construction for the global-level measure G_0 and group-level measure G_j. The base measure is defined via stick-breaking process as:

$$\begin{aligned}
&\Lambda_k \sim H \\
&\eta'_k \sim \text{Beta}(1-\gamma_a, \gamma_b + k\gamma_a) \\
&\eta_k = \eta'_k \prod_{l=1}^{k-1}(1-\eta'_l) \\
&G_0 = \sum_{k=1}^{\infty} \eta_k \delta_{\Lambda_k}
\end{aligned} \tag{6.11}$$

where Λ_k is independent sample drawn from base distribution H and δ_{Λ_k} represents probability mass concentrated at Λ_k, and η_k denotes the relative prevalence of each atoms which satisfy $\sum_{k=1}^{\infty} \eta_k = 1$. The stick lengths η' are distributed according to Beta distribution with two parameters γ_a and γ_b.

The group-level measures of HPYP also constructed via stick-breaking construction similar to the HDP described in Section 6.1.1:

$$\begin{aligned}
&\psi_{jt} \sim G_0 \\
&p'_{jt} \sim \text{Beta}(1-\beta_a, \beta_b + t\beta_a) \\
&p_{jt} = p'_{jt} \prod_{s=1}^{t-1}(1-p'_{js}) \\
&G_j = \sum_{t=1}^{\infty} p_{jt} \delta_{\psi_{jt}}
\end{aligned} \tag{6.12}$$

where ψ_{jt} is the atom of second-level PYP and δ_{Λ_k} indicates the corresponding realization, and p_{jt} represents the probability mass associated with each atoms with constraint of sum equal to 1. The stick lengths p' are given a Beta prior with discount parameters γ_a and concentration γ_b.

I also introduce the global-level indicator variables I and group-level indicator variables W as HDP in Eq. (6.4) and Eq. (6.8). The indicators W could map the component θ_{ji} to group-level atom ψ_{jt} and I assign the lower-level atom to global-level one Λ_k.

$$p(I \mid \eta') = \prod_{j}^{M} \prod_{t}^{\infty} \prod_{k}^{\infty} \eta_k^{I_{jtk}} = \prod_{j}^{M} \prod_{t}^{T} \prod_{k}^{K} [\eta'_k \prod_{l=1}^{k-1} (1 - \eta'_l)]^{I_{jtk}} \tag{6.13}$$

$$p(W \mid p') = \prod_{j}^{M} \prod_{i}^{N} \prod_{t}^{\infty} p_{jt}^{W_{jit}} = \prod_{j}^{M} \prod_{i}^{N} \prod_{t}^{T} [p'_{jt} \prod_{s=1}^{t-1} (1 - p_{js})]^{W_{jit}} \tag{6.14}$$

where the latent indicators are distributed according to the Multinomial distribution with stick lengths. Thus, I obtain generalized hierarchical mixture model.

6.1.3 Hierarchical infinite mixture models of asymmetric Gaussian distributions

I restrict the base distribution of H in Eq. (6.1) as AGD with the set of parameters $(\mu,\ \sigma_l, \sigma_r)$. If a D dimensional input vector $X = (X_1, \ldots, X_D)$ follows AGD, then the probability density function is given by:

$$p(X \mid \mu, \sigma_l, \sigma_r) \propto \prod_{d=1}^{D} \frac{1}{\sigma_{ld} + \sigma_{rd}} \times \begin{cases} \exp\left\{ -\frac{(X_d - \mu_d)^2}{2\sigma_{ld}^2} \right\} & \text{if } X_d < \mu_d \\ \exp\left\{ -\frac{(X_d - \mu_d)^2}{2\sigma_{rd}^2} \right\} & \text{if } X_d \geq \mu_d \end{cases} \tag{6.15}$$

where $(\mu, \sigma_l, \sigma_r)$ is the complete set of parameters for AGD, where $\mu = (\mu_1, \ldots, \mu_D)$, $\sigma_l = (\sigma_{l1}, \ldots, \sigma_{lD})$, and $\sigma_r = (\sigma_{r1}, \ldots, \sigma_{rD})$. μ_d, σ_{ld} and $_{rd}$ are the mean, the left and right standard deviation for the dth-dimensional AGD, respectively. Here I assume each dimension of observation X is independent and its covariance matrix will be diagonal leading to the reduction of computational expense during deployment stage.

Here I have a dataset X with N random vectors categorized into M groups, where each D dimensional observation $X_{ji} = (X_{ji1}, \ldots, X_{jiD})$ is distributed according to Hierarchical infinite mixture model with asymmetric Gaussian, the likelihood function can be illustrated with respect to the complete parameters set of asymmetric Gaussian $(\mu, \sigma_l, \sigma_r)$ and latent indicators as follows:

$$p(X \mid Z, C, \mu, \sigma_l, \sigma_r) = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} p(X_{ji} \mid \mu_k, \sigma_{lk}, \sigma_{rk})^{Z_{jit} C_{jtk}}$$
$$p(X \mid W, I, \mu, \sigma_l, \sigma_r) = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} p(X_{ji} \mid \mu_k, \sigma_{lk}, \sigma_{rk})^{W_{jit} I_{jtk}} \tag{6.16}$$

where the two functions represent the HDP asymmetric Gaussian mixture (HDPAGM) and HPYP asymmetric Gaussian mixture (HPYPAGM). The mean parameters μ follow a Gaussian distribution with mean λ and precision r, i.e. the inverse variance of Gaussian distribution. The standard deviation variables σ_l and σ_r are given Gaussian distribution with high value standard deviation variable suggested by [30]:

$$\begin{aligned} p(\mu_{kd} \mid \lambda, r) &\sim \mathcal{N}(\lambda_{kd}, r_{kd}) \\ p(\sigma_{lkd} \mid m_l, s_l) &\sim \mathcal{N}(m_{lkd}, s_{lkd}^2) \\ p(\sigma_{rkd} \mid m_r, s_r) &\sim \mathcal{N}(m_{rkd}, s_{rkd}^2) \end{aligned} \tag{6.17}$$

6.2 Variational inference

6.2.1 Variational approximation

In this section, I consider the complete Variational Bayes inference framework proposed in chapter 5. Variational inference is a well-defined method to approximate probability densities through optimization [6] [31].

The mean field assumption renders the latent variables independent and discovers the true density from the posed family. Here, I adopt fully factorized form of variational distributions and perform mean field variational inference on HDPAGM and HPYPAGM mixture. The variational lower bound can be found as:

$$q(C, V', Z, \pi', \mu, \sigma_l, \sigma_r) = q(C) q(V') q(Z) q(\pi') q(\mu) q(\sigma_l) q(\sigma_r) \tag{6.18}$$

$$q(I, \eta', W, p', \mu, \sigma_l, \sigma_r) = q(I) q(\eta') q(W) q(p') q(\mu) q(\sigma_l) q(\sigma_r) \tag{6.19}$$

where the latent variables of AGD have same factorized form. Consider suitable family of variational approximations, I can obtain the following distributions as:

$$
\begin{aligned}
q(C) &= \prod_{j}^{M}\prod_{t}^{T}\prod_{k}^{K}\text{Multi}(C_{jtk} \mid \phi_{jtk}) \\
q(Z) &= \prod_{j}^{M}\prod_{i}^{N}\prod_{t}^{T}\text{Multi}(Z_{jit} \mid \rho_{jit}) \\
q(V') &= \prod_{k}^{K}\text{Beta}(V'_{k} \mid u_k, v_k) \\
q(\pi') &= \prod_{j}^{M}\prod_{t}^{T}\text{Beta}(\pi'_{jt} \mid a_{jt}, b_{jt}) \\
q(I) &= \prod_{j}^{M}\prod_{t}^{T}\prod_{k}^{K}\text{Multi}(I_{jtk} \mid \varphi_{jtk}) \\
q(W) &= \prod_{j}^{M}\prod_{i}^{N}\prod_{t}^{T}\text{Multi}(W_{jit} \mid \varrho_{jit}) \\
q(\eta') &= \prod_{k}^{K}\text{Beta}(\eta'_{k} \mid c_k, d_k) \\
q(p') &= \prod_{j}^{M}\prod_{t}^{T}\text{Beta}(p'_{jt} \mid e_{jt}, f_{jt}) \\
q(\mu) &= \prod_{k}^{K}\prod_{d}^{D}\mathcal{N}(m_{kd} \mid m_{kd}, \Sigma_{kd}) \\
q(\sigma_l) &= \prod_{k}^{K}\prod_{d}^{D}\mathcal{N}(\sigma_{lkd} \mid \iota_{lkd}, \upsilon^2_{lkd}) \\
q(\sigma_r) &= \prod_{k}^{K}\prod_{d}^{D}\mathcal{N}(\sigma_{rkd} \mid \iota_{rkd}, \upsilon^2_{rkd})
\end{aligned}
\tag{6.20}
$$

The variational distributions for indicator variables C and Z are Multinomial. The stick lengths V' and π' follow the Beta distribution. The variational distribution of mean parameters μ is considered as Gaussian distribution with mean m and variance Σ. σ_l, and σ_r are given Gaussian variational distributions with mean ι and standard deviation υ.

For proposed HDPAGM, I could expand the ELBO in Eq. 6.18 by using the mean field assumption:

$$\mathcal{L} = E_q[\log p(X \mid C, Z, \mu, \sigma_l, \sigma_r)]) + E_q[\log p(C \mid V')] + E_q[\log p(V' \mid \omega)] + E_q[\log p(Z \mid \pi')] + E_q[\log p(\pi' \mid \alpha)] + E_q[\log p(\mu \mid \lambda, r)] + E_q[\log p(\sigma_l \mid \iota_l, \upsilon_l)] + E_q[\log p(\sigma_r \mid \iota_r, \upsilon_r)] - E_q[\log q(C, V', Z, \pi', \mu, \sigma_l, \sigma_r)] \tag{6.21}$$

I perform CAVI to optimize the ELBO in Eq. (6.21) with respect to the repeated updates of each parameter. I obtain the optimal solutions for the variables of the posterior densities by applying Eq. (6.18) and Eq. (6.20), excluding the variables associated with the standard deviations. Since I cannot reach the closed form of standard deviation variables (σ_l, σ_r) without the non-conjugate priors. Therefore, I apply the BBVI and variance reduction technique proposed in chapter 5 and achieve the desired approximation since the black box variational method is easy to extend to different models.

6.2.2 Coordinate ascent variational inference

I present the explicit coordinate ascent method to update variational parameters. The variational parameters of Multinomial distributions ϕ and ρ are normalized with the solutions to each parameter in the HDP mixture as follows:

$$\phi_{jtk} = \frac{\hat{\phi}_{jtk}}{\sum_k^K \hat{\phi}_{jtk}} \qquad \rho_{jit} = \frac{\hat{\rho}_{jit}}{\sum_t^T \hat{\rho}_{jit}} \tag{6.22}$$

$$\begin{aligned} \hat{\phi}_{jtk} &= \exp\big\{E_q[\log V_k'] + \sum_{l=1}^{k-1} E_q[\log(1 - V_l')] \\ &\quad - \sum_i^N E_q[Z_{jit}]\widetilde{R}\big\} \\ \hat{\rho}_{jit} &= \exp\big\{E_q[\log \pi_{jt}'] + \sum_{s=1}^{t-1} E_q[\log(1 - \pi_{jt}')] \\ &\quad - \sum_k^K E_q[C_{jtk}]\widetilde{R}\big\}\hat{\phi}_{jtk} \end{aligned} \tag{6.23}$$

$$
\begin{aligned}
\widetilde{R} = &\sum_{d}^{D} E_q[\log(\sigma_{lkd} + \sigma_{rkd})] \\
&+ \sum_{d, X_{jid} < \mu_{kd}}^{D} \frac{X_{jid}^2 + E_q[\mu_{kd}^2] - 2X_{jid}E_q[\mu_{kd}]}{2E_q[\sigma_{lkd}^2]} \\
&+ \sum_{d, X_{jid} \geq \mu_{kd}}^{D} \frac{X_{jid}^2 + E_q[\mu_{kd}^2] - 2X_{jid}E_q[\mu_{kd}]}{2E_q[\sigma_{rkd}^2]}
\end{aligned} \tag{6.24}
$$

$$
\begin{aligned}
u_k &= 1 + \sum_{j}^{M} \sum_{t}^{T} E_q[C_{jtk}] \\
v_k &= w_k + \sum_{j}^{M} \sum_{t}^{T} \sum_{l=k+1}^{K} E_q[C_{jtl}] \\
a_{jt} &= 1 + \sum_{i}^{N} E_q[Z_{jit}] \\
b_{jt} &= \alpha_{jt} + \sum_{i}^{N} \sum_{s=t+1}^{T} E_q[Z_{jis}]
\end{aligned} \tag{6.25}
$$

$$
\begin{aligned}
\Sigma_{kd} = &\big(r + \sum_{j}^{M} \sum_{t}^{T} \sum_{i, X_{jid} < \mu_{kd}}^{N} \frac{E_q[Z_{jit}]E_q[C_{jtk}]}{E_q[\sigma_{lkd}^2]} \\
&+ \sum_{i, X_{jid} \geq \mu_{kd}}^{N} \frac{E_q[Z_{jit}]E_q[C_{jtk}]}{E_q[\sigma_{rkd}^2]}\big)^{-1} \\
m_{kd} = &\Sigma_{kd}\big(\lambda r + \sum_{j}^{M} \sum_{t}^{T} \sum_{i, X_{jid} < \mu_{kd}}^{N} \frac{E_q[Z_{jit}]E_q[C_{jtk}]X_{jid}}{E_q[\sigma_{lkd}^2]} \\
&+ \sum_{i, X_{jid} \geq \mu_{kd}}^{N} \frac{E_q[Z_{jit}]E_q[C_{jtk}]X_{jid}}{E_q[\sigma_{rkd}^2]}\big)
\end{aligned} \tag{6.26}
$$

The expected values for stick lengths and indicator variables in the above equations can be calculated:

$$
E_q[C_{jtk}] = \phi_{jtk} \quad E_q[Z_{jit}] = \rho_{jit} \tag{6.27}
$$

$$
\begin{aligned}
E_q[\log V_k] &= E_q[\log V'_k] + \sum_{l=1}^{k-1} E_q[\log(1 - V'_l)] \\
E_q[\log(V'_k)] &= \Psi(u_k) - \Psi(u_k + v_k) \\
E_q[\log(1 - V'_k)] &= \Psi(v_k) - \Psi(u_k + v_k)
\end{aligned} \tag{6.28}
$$

$$
\begin{aligned}
E_q[\log \pi_{jt}] &= E_q[\log \pi'_{jt}] + \sum_{s=1}^{t-1} E_q[\log(1 - \pi'_{jt})] \\
E_q[\log(\pi'_{jt})] &= \Psi(a_{jt}) - \Psi(a_{jt} + b_{jt}) \\
E_q[\log(1 - \pi'_{jt})] &= \Psi(b_{jt}) - \Psi(a_{jt} + b_{jt})
\end{aligned} \tag{6.29}
$$

where $\Psi(\cdot)$ denotes the Digamma function that arises from the derivative of the log normalization factor in the Beta distribution.

The approximated expectations for the parameter set of AGD are calculated by:

$$
\begin{aligned}
E_q[\mu_{kd}] &= m_{kd} \quad E_q[\mu^2_{kd}] = m^2_{kd} + \Sigma_{kd} \\
E_q[\sigma_{lkd}] &= \iota_{l,kd} \quad E_q[\sigma^2_{lkd}] = \iota^2_{l,kd} + \upsilon^2_{l,kd} \\
E_q[\sigma_{rkd}] &= \iota_{r,kd} \quad E_q[\sigma^2_{rkd}] = \iota^2_{r,kd} + \upsilon^2_{r,kd}
\end{aligned} \tag{6.30}
$$

I present the expected value of $E_q[\log(\sigma_{lkd} + \sigma_{rkd})]$ by applying the Jensen's inequality and replacing with a upper bound:

$$
E_q[\log(\sigma_{lkd} + \sigma_{rkd})] \leq \log(E_q[\sigma_{lkd} + \sigma_{rkd}]) = \log(\iota_{l,kd} + \iota_{r,kd}) \tag{6.31}
$$

Since the variational solutions of HPYP asymmetric Gaussian share similar characteristics, the explicit formulas are detailed in Appendix. B.

6.3 Learning Algorithm

An important aspect when applying variational inference is the convergence assessment. In this chapter I trace the convergence systematically by monitoring the ELBO. Convergence

Figure 6.1: Sample frames from video sequence in different categories in the DynTex dataset.

Algorithm 6 Hierarchical Dirichlet Process Asymmetric Gaussian Mixture

procedure
    Initialization:
    Initialize the truncation levels K and T.
    Initialize the parameters of priors: ω, α, λ and r, m_{lkd}, s_{lkd}, m_{rkd} and s_{rkd}.
    Initialize the parameters of variational distributions: ϕ, ρ, u, v, a, b μ and Σ.
    repeat:
    VB E-step:
    Estimate the expected values in Eq. (6.27), Eq. (6.28), Eq. (6.29), and Eq. (6.30).
    VB M-step:
    Update the variational solutions for each factors using Eq. (6.22), Eq. (6.23), Eq. (6.25), and Eq. (6.26).
    Update variational hyperparameters from latent variables σ_{lkd} and σ_{rkd} by BBVI from Section 4.2.2, with variance control approach from Section 5.2.2.
    The convergence criteria is reached when the difference of the current value of ELBO and previous value is less than 10^{-2} or the epochs number exceeds 300.
    until convergence

is reached when the ELBO is less than 10^{-2} between epochs or the number of iterations is more than 300. The Bayesian inference framework of the HDPAGM is summarized in Algorithm 6.

The detailed learning equations of HPYPAGM is presented in Appendix B. The complete learning algorithm is summarized in Algorithm 7.

Algorithm 7 Hierarchical Pitman-Yor Process Asymmetric Gaussian Mixture

procedure
    Initialization:
    Initialize the truncation levels K and T.
    Initialize the parameters of priors: γ_a, γ_b, β_a, β_b, λ and r, m_{lkd}, s_{lkd}, m_{rkd} and s_{rkd}.
    Initialize the parameters of variational distributions: φ, ϱ, c, d, e, f μ and Σ.
    repeat:
    VB E-step:
    Estimate the expected values in Eq. (B.6), Eq. (B.7), Eq. (B.8), and Eq. (6.30).
    VB M-step:
    Update the variational solutions for each factors using Eq. (B.2), Eq. (B.3), Eq. (B.4), and Eq. (B.5).
    Update variational hyperparameters from latent variables σ_{lkd} and σ_{rkd} by BBVI from Section 4.2.2, with variance control approach from Section 5.2.2.
    The convergence criteria is reached when the difference of the current value of ELBO and previous value is less than 10^{-2} or the epochs number exceeds 300.
    until convergence

6.4 Experimental Results

I evaluate the effectiveness of the proposed HDP mixture and HPYP mixture model with AGD using challenging dynamic texture clustering application. In the experiments, I initialize the global truncation level K and group level truncation level T to 120 and 60, respectively. For HDP mixture, the hyperparameters of the stick lengths ω and α are initialized to 0.25; I set the parameters of HPYP mixture γ_a, γ_b, β_a and β_b as 0.25.

The hyperparameters of asymmetric Gaussian base distribution are initialized by sampling from priors. The mean parameters μ are sampled from Gaussian prior with a mean calculated by the average value of the observations. The left and right standard deviations σ_l and σ_r are sampled from Gaussian distribution with a high mean value as studied by [30].

Table 6.1: The accuracy results of dynamic texture clustering evaluated by different algorithms.

Approach	HPYPDM	HDPGM	HPYPGM	HDPAGM	HPYPAGM
Accuarcy (%)	82.75	74.96	75.19	83.47	84.21

6.4.1 Dynamic Texture Clustering

Dynamic textures are the extension of texture to the temporal domains, which can be defined as sequences of images of moving scenes that exhibit certain stationary properties in time (e.g., sea-waves, smoke, foliage, whirlwind etc.) [18]. Dynamic textures have been applied in a vast number of applications in image processing, such as motion classification, video registration, computer games, and motion segmentation [50].

In this chapter, I apply the proposed two hierarchical infinite mixtures to clustering dynamic textures with the representation of scale-invariant feature transform (SIFT). The methodology can be summarized as follows. The first step of the approach is to extract 128-dimensional SIFT descriptors [51] from each test scene frame using the difference-of-Gaussians interest points detector and then normalized. Then, I apply the geometric transformation for the resultant vectors and model it using the proposed HDPAGM and HPYPAGM. In that case, each dynamic texture image I_j is treated as a group and is associated with a infinite mixture model G_j (e.g., the DP mixture model and PYP mixture model). Thus, each extracted SIFT feature vector X_{ji} of the dynamical texture I_j is supposed to be drawn from G_j, where the mixture components of G_j can be considered as visual words. Next, a global vocabulary is constructed and is shared among all groups (dynamic textures) through the common global infinite mixture model G_0 of the hierarchical model.

It is worth mentioning that most of the previously proposed bag-of-visual-words approaches have to apply a separate vector quantization technique, such as K-means algorithm, to build the visual vocabulary, where the size of vocabulary is normally manually selected. In contrast, the construction of the visual vocabulary in the approach is part of the framework of the hierarchical infinite mixture models, and therefore, the size of the vocabulary (i.e., the number of mixture components in the global-level mixture model) can be automatically inferred from the data due to the property of nonparametric Bayesian models. Then, I employ the paradigm of bag-of-visual-words and compute a histogram of visual words from each image. Since the objective is to determine the texture cluster that a testing dynamic texture I_j belongs to, I introduce the indicator variable B_{jm} and associate each

image with cluster label in the hierarchical infinite mixture frameworks. B_{jm} represents the dynamic texture I_j allocated to the cluster m and is drawn from another infinite mixture model, which is truncated at level J. As a result, the methodology requires a new level of hierarchy to the proposed hierarchical infinite mixture modelx with a sharing vocabulary among all clusters. In this experiment, I truncate J to 50 and initialize the hyperparameters of the mixing probability B_{jm} to 0.2. Finally, I assign a testing dynamic texture into the cluster that has the highest posterior probability according to Bayes decision rule.

6.4.2 Dataset and Results

In this experiment, the proposed HPDAGM and HPYPAGM are validated on clustering challenging dynamic texture dataset which is known as the DynTex database [21]. This dataset consists of more than 650 dynamic texture videos from several categories and recorded using SONY 3 CCD camera mounted on a tripod. All sequences are captured in Phase Alternate Line (PAL) format (720×576), 25 Frames Per Second (fps), interlaced. In that case, I use a subset of video sequences from 10 different categories: sea, vegetation, trees, flags, clam water, fountains, smoke, escalator, traffic, and rotation. Each category has about 20 video sequences. The sample frames from each category are shown in Fig. 6.1.

For preprocessing, all features in the dataset are normalized into the range of $[0, 1]$. I do not need to include the class labels in the experiment since I were performing clustering analysis. I use cross validation method to partition the dataset. I evaluate the proposed approach and obtain average results from 30 runs. In order to evaluate the performance of the proposed method, I compare with three other approaches, hierarchical Pitman-Yor process mixture of Dirichlet distribution (HPYPDM) [20], Hierarchical Dirichlet Process mixture of Gaussian distribution (HDPGM), and hierarchical Pitman-Yor process mixture of Gaussian distribution (HPYPGM). The average results in terms of the clustering accuracy are summarized in Table. 6.1. Figure. 6.2 and Figure. 6.3 show the confusion matrices for the Dyntex dataset using proposed hierarchical infinite mixture models.

According to the results shown in the table, I can observe that HDPAGM and HPYPAGM have shown to outperform the other three methods in terms of the categorization accuracy rate. By contrast, the hierarchical infinite mixture models with Gaussian base distribution has obtained the worst performance. This confirms the merit of AGD in incorporating the asymmetry structure inside the dataset. Furthermore, the proposed model outperforms HPYPDM which is considered in modeling positive data accurately [20]. Moreover,

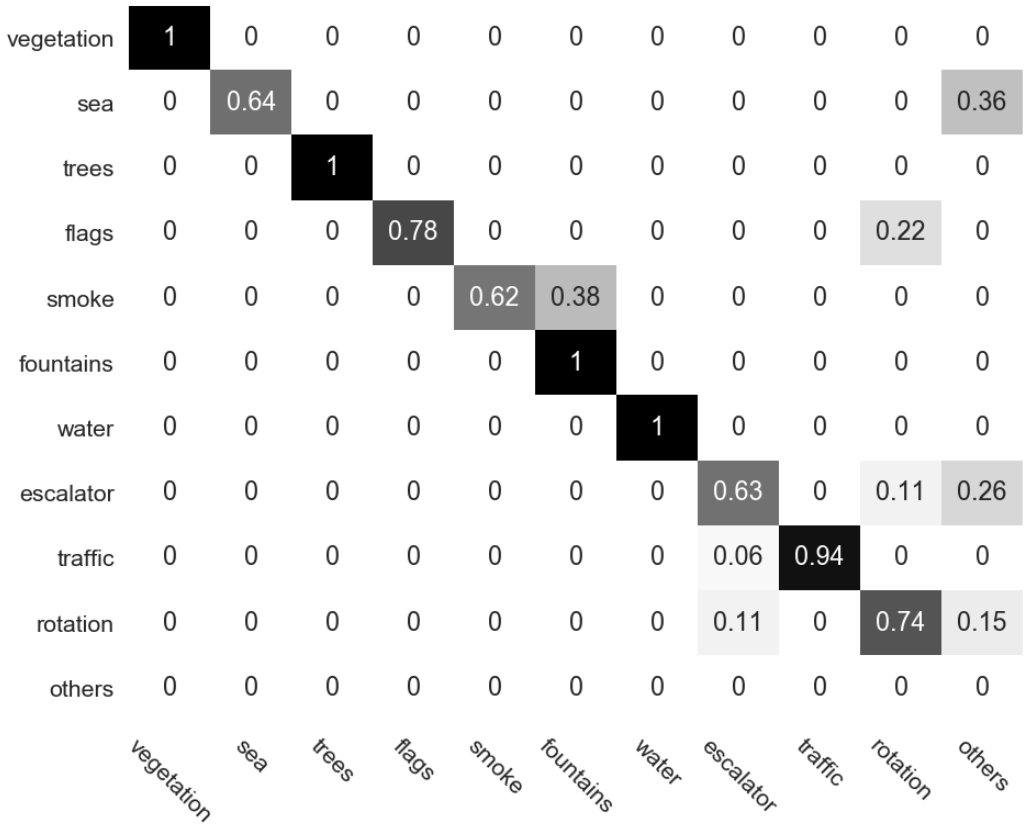


Figure 6.2: Confusion matrix of HDPAGM for the DynTex dataset.

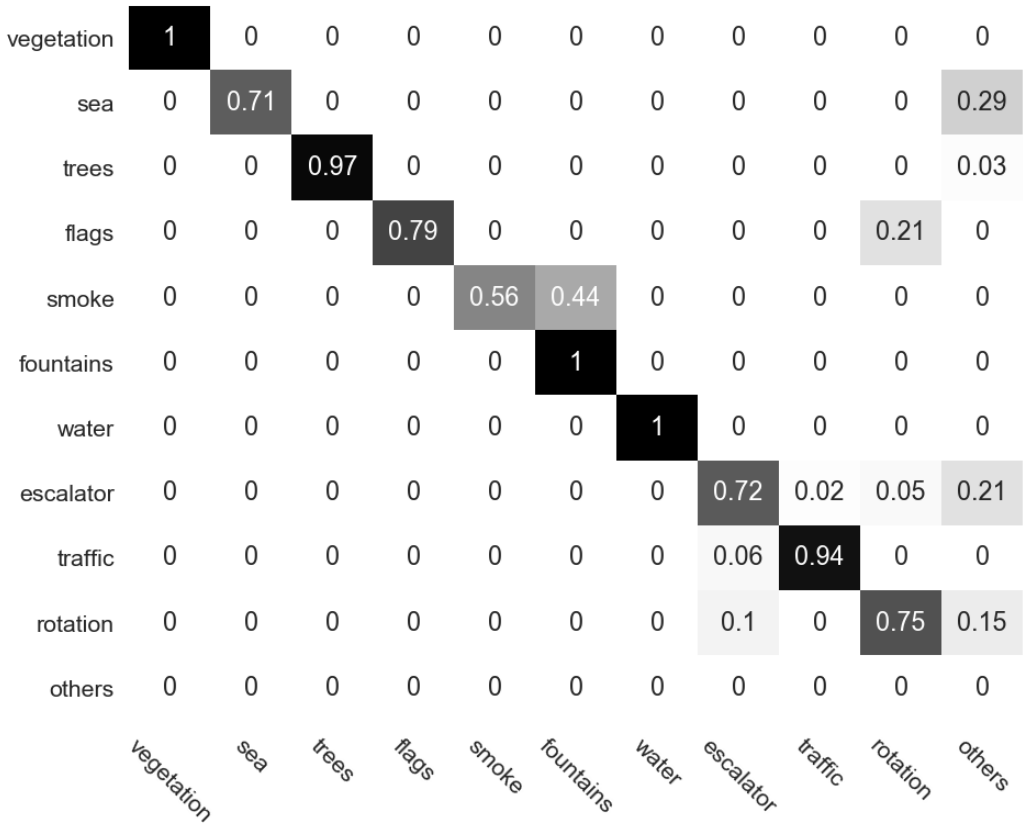


Figure 6.3: Confusion matrix of HPYPAGM for the DynTex dataset.

I also find that HPYP mixture model achieves better accuracy compared with HDP mixture model with specific distribution. It demonstrates that a PYP prior could lead to a better modeling capability.

Chapter 7

Conclusion

Clustering has become an inevitable part of image processing and pattern recognition domains. I have explored several extensions and statistical inference approaches and demonstrated its efficiency.

In chapter 2, I introduce a novel infinite mixture model, IAGM, that is capable of modeling the asymmetry of data in contrast to the traditionally deployed GMM. Moreover, I address the challenges of learning parameter and choosing the adequate number of components through the employment of MCMC algorithm and the extension of model to infinity with nonparametric prior.

In chapter 3, I integrate feature selection algorithm into mixture model for clustering high-dimensional data which plagues unsupervised learning algorithm frequently. Through Bayesian framework, identifying relevant features and parameter inference are unified into the same framework.

Then, in chapter 4, I have presented and implemented an efficient variational learning framework for finite mixture model with AGD. I choose a factorized coordinate ascent variational approximation as well as the black box variational inference method which apply Monte Carlo estimation. It demonstrated the effectiveness of gradient ascent inference method to deal with non-conjugate problem.

In chapter 5, I have proposed a nonparametric prior for asymmetric Gaussian mixture to combine the learning of model complexity and the estimation of parameters together. Moreover, I incorporate feature selection within the proposed framework which alleviates the noisy influence of irrelevant features. Although the variance of the noisy gradient estimator consistently obstructs utility, a simple and general variant of variance reduction

algorithm is developed to guarantee the convergence. All though an infinite mixture model may appear complicated because of the number of involved parameters, the final model remains concise as the self-correcting component reduction process cuts the least effective components and leaves well-separated clusters.

Finally, in chapter 6 I have presented hierarchical nonparametric Bayesian models. I consider the AGD and proposed an effective variational inference framework to estimate latent variables for hierarchical infinite mixtures. The hierarchical nonparametric Bayesian mixtures are evaluated on grouped image features.

The experiments with proposed frameworks are motivating and proves to be a better solution than prevalent considered GMM for appropriate data. A promising future work is to consider advanced automatic variational inference approach; for instance, [52] propose Automatic Differentiation Variational Inference approach to save effort for building model. Since a nonparametric process is not differentiable, a potential way to include such models is to approximate the discrete latent variables by introducing adequate variational families.

Bibliography

[1] Sangwoo Park, Erchin Serpedin, and Khalid A. Qaraqe. Gaussian assumption: The least favorable but the most useful [lecture notes]. *IEEE Signal Processing Magazine*, 30:183–186, 2013.

[2] Ivan Laptev. Improving object detection with boosted histograms. *Image Vision Comput.*, 27(5):535–544, April 2009.

[3] Tarek Elguebaly and Nizar Bouguila. Bayesian learning of finite generalized gaussian mixture models on images. *Signal Processing*, 91(4):801 – 820, 2011.

[4] David Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, New York, NY, USA, 2012.

[5] Christophe Andrieu, Nando de Freitas, Arnaud Doucet, and Michael I. Jordan. An introduction to mcmc for machine learning. *Machine Learning*, 50(1):5–43, Jan 2003.

[6] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, Nov 1999.

[7] Jim Griffin and Mark Steel. Bayesian nonparametric modelling with the dirichlet process regression smoother. *Statistica Sinica*, 20, 10 2010.

[8] Sabri Boutemedjet, Nizar Bouguila, and Djemel Ziou. A hybrid feature extraction selection approach for high-dimensional non-gaussian data clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:1429–1443, 2009.

[9] Wentao Fan, Nizar Bouguila, and Xin Liu. A nonparametric bayesian learning model using accelerated variational inference and feature selection. *Pattern Analysis and Applications*, 22(1):63–74, Feb 2019.

[10] Martin H. C. Law, Mário A. T. Figueiredo, and Anil K. Jain. Simultaneous feature selection and clustering using mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:1154–1166, 2004.

[11] Yee Whye Teh and Gatsby. Hierarchical bayesian nonparametric models with applications . 2008.

[12] Jim Pitman and Marc Yor. The two-parameter poisson-dirichlet distribution derived from a stable subordinator. *Ann. Probab.*, 25(2):855–900, 04 1997.

[13] Carl Edward Rasmussen. The infinite gaussian mixture model. In S. A. Solla, T. K. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 554–560. MIT Press, 2000.

[14] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, volume 2, pages 246–252. IEEE, 1999.

[15] Yi Wang, Pierre-Marc Jodoin, Fatih Porikli, Janusz Konrad, Yannick Benezeth, and Prakash Ishwar. Cdnet 2014: An expanded change detection benchmark dataset. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 06 2014.

[16] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. volume 2, pages 28 – 31 Vol.2, 09 2004.

[17] Tarek Elguebaly and Nizar Bouguila. Background subtraction using finite mixtures of asymmetric gaussian distributions and shadow detection. *Machine Vision and Applications*, 25, 07 2013.

[18] Gianfranco Doretto, Alessandro Chiuso, Ying Nian Wu, and Stefano Soatto. Dynamic textures. *International Journal of Computer Vision*, 51(2):91–109, Feb 2003.

[19] Wentao Fan and Nizar Bouguila. Learning finite beta-liouville mixture models via variational bayes for proportional data clustering. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pages 1323–1329. AAAI Press, 2013.

[20] Wentao Fan and Nizar Bouguila. Dynamic textures clustering using a hierarchical pitman-yor process mixture of dirichlet distributions. In *2015 IEEE International Conference on Image Processing, ICIP 2015, Quebec City, QC, Canada, September 27-30, 2015*, pages 296–300, 2015.

[21] Renaud Péteri, Sándor Fazekas, and Mark J. Huiskes. DynTex : a Comprehensive Database of Dynamic Textures. *Pattern Recognition Letters*, 2010.

[22] Guoying Zhao and Matti Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(6):915–928, June 2007.

[23] Li-Jia Li and Li Fei-Fei. What, where and who? classifying events by scene and object recognition. *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

[24] Yiu-ming Cheung and Hong Zeng. Feature weighted rival penalized em for gaussian mixture clustering: Automatic feature and model selections in a single paradigm. volume 1, pages 1018–1028, 09 2007.

[25] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, May 2001.

[26] Chong Wang, David M. Blei, and Li Fei-Fei. Simultaneous image classification and annotation. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1903–1910, 2009.

[27] Svetlana Lazebnik, Cordelia Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. volume 2, pages 2169 – 2178, 02 2006.

[28] Gabriela Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cdric Bray. Visual categorization with bags of keypoints. *Work Stat Learn Comput Vision, ECCV*, Vol. 1, 01 2004.

[29] Jun Zhu, Li-Jia Li, Li Fei-Fei, and Eric P. Xing. Large margin learning of upstream scene understanding models. In *NIPS*, 2010.

[30] Andrew Gelman. Prior distributions for variance parameters in hierarchical models. *Bayesian Anal.*, 1(3):515–534, 09 2006.

[31] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[32] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[33] Chong Wang and David M. Blei. Variational inference in nonconjugate models. *J. Mach. Learn. Res.*, 14(1):1005–1031, April 2013.

[34] Rajesh Ranganath, Sean Gerrish, and David M. Blei. Black Box Variational Inference. *arXiv e-prints*, page arXiv:1401.0118, Dec 2013.

[35] David M. Blei and Michael I. Jordan. Variational inference for dirichlet process mixtures. *Bayesian Anal.*, 1(1):121–143, 03 2006.

[36] Thomas S. Ferguson. A bayesian analysis of some nonparametric problems. *Ann. Statist.*, 1(2):209–230, 03 1973.

[37] Kenichi Kurihara, Max Welling, and Yee Whye Teh. Collapsed variational dirichlet process mixture models. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2796–2801, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[38] Jayaram Sethuraman. A constructive definition of the dirichlet prior. *Statistica Sinica*, 4:639–650, 01 1994.

[39] Geoffrey Roeder, Yuhuai Wu, and David K. Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In *NIPS*, 2017.

[40] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, page arXiv:1312.6114, Dec 2013.

[41] Carl Edward Rasmussen. The infinite gaussian mixture model. In S. A. Solla, T. K. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 554–560. MIT Press, 2000.

[42] Jianyong Sun, Aimin Zhou, Simeon Keates, and Shengbin Liao. Simultaneous bayesian clustering and feature selection through student's t mixtures model. *IEEE Transactions on Neural Networks and Learning Systems*, PP:1–13, 03 2017.

[43] Pakorn KaewTrakulPong and Richard Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. 2002.

[44] Ruben Evangelio, Michael Patzold, and Thomas Sikora. Splitting gaussians in mixture models. pages 300–305, 09 2012.

[45] Ahmed M. Elgammal, David Harwood, and Larry S. Davis. Non-parametric model for background subtraction. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, ECCV '00, pages 751–767, London, UK, UK, 2000. Springer-Verlag.

[46] Yosuke Nonaka, Atsushi Shimada, Hajime Nagahara, and Rin-ichiro Taniguchi. Evaluation report of integrated background modeling based on spatio-temporal features. pages 9–14, 06 2012.

[47] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[48] Emily B. Fox, Erik B. Sudderth, Michael Jordan, and Alan S. Willsky. An hdp-hmm for systems with state persistence. pages 312–319, 01 2008.

[49] Hemant Ishwaran and Lancelot F James. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453):161–173, 2001.

[50] Avinash Ravichandran, Rizwan Chaudhry, and Ren Vidal. Categorizing dynamic textures using a bag of dynamical systems. *IEEE transactions on pattern analysis and machine intelligence*, 35:342–53, 02 2013.

[51] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov 2004.

[52] Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M. Blei. Automatic differentiation variational inference. *J. Mach. Learn. Res.*, 18(1):430–474, January 2017.

Appendix

A Full Equations of Infinite Asymmetric Gaussian with Feature Selection

Based on the hyperparameters setting we chosen in Section 3.2, we deduce the posteriors for all of the parameters.

For parameter α, the posteriors depend only on the number of observations N and the number of components M, and not on how the distributions are distributed among the mixtures:

$$p(\alpha \mid k, n) \propto \frac{\alpha^{M-\frac{3}{2}} \exp(-\frac{1}{2\alpha})\Gamma(\alpha)}{\Gamma(N+\alpha)} \tag{A.1}$$

The complete posteriors for μ, μ_{irr}, λ and r are obtained as follows:

$$p(\mu_{jk} \mid \dots) \propto \mathcal{N}\big(\frac{r\lambda + S_{ljk}\sum_{i:\phi_{ijk}=1, X_{ik}<\mu_{jk}} X_{ik} + s_{rjk}\sum_{i:\phi_{ijk}=1, X_{ik}\geq\mu_{jk}} X_{ik}}{r + ps_{ljk} + (n_j - p)s_{rjk})}, \frac{1}{r + ps_{ljk} + (n_j - p)s_{rjk}}\big) \tag{A.2}$$

$$p(\mu_{jk}^{irr} \mid \dots) \propto \mathcal{N}\big(\frac{\sum_{i,\phi_{ijk}=0} x_{ik}^{irr} S_{jk}^{irr} + r_k^{irr}\lambda_k^{irr}}{r_k^{irr} + n_j^{irr} S_{jk}^{irr}}, \frac{1}{r_k^{irr} + n_j^{irr} S_{jk}^{irr}}\big) \tag{A.3}$$

$$p(\lambda \mid \mu_{1k}, \dots, \mu_{Mk}, r) \propto \mathcal{N}\big(\frac{r\sum_{j=1}^{M}\mu_{jk} + \mu_x\sigma_x^{-2}}{\sigma_x^{-2} + Mr_k}, \frac{1}{\sigma_x^{-2} + Mr_k}\big) \tag{A.4}$$

$$p(r \mid \mu_{1k}, \dots, \mu_{Mk}, \lambda) \propto \gamma\big(\frac{M+1}{2}, \frac{2}{\sigma_x^2 + \sum_{j=1}^{M}(\mu_{jk} - \lambda_k)^2}\big) \tag{A.5}$$

The complete posteriors for s_{ljk}, s_{rjk}, s_{jk}^{irr}, β and w are obtained as follows:

$$p(S_{ljk} \mid X, \mu_j, S_{rj}, \beta_l, w_l) \propto \exp\Big\{ - \frac{S_{ljk}\sum_{i:X_{ik}<\mu_{jk}}^{n}(x_{ik} - \mu_{jk})^2}{2} - \frac{w_{lk}\beta_{lk}S_{ljk}}{2}\Big\} \tag{A.6}$$

$$p(S_j^{irr} \mid X, \mu_j^{irr}, \beta^{irr}, w^{irr}) \propto \Gamma\big(\frac{N_{jk}^{irr}\beta_k^{irr}}{2}, \frac{2}{\beta_k^{irr} w_k^{irr} + \sum_{i,\phi_{ijk}=0}(X_{ik} - \mu_{jk}^{irr})^2}\big) \quad \text{(A.7)}$$

$$p(\beta_l \mid S_{l1k}, \ldots, S_{lMk}, w_l) \propto \Gamma(\frac{\beta_l}{2})^{-M} \exp(-\frac{1}{2\beta_l})(\frac{\beta_l}{2})^{\frac{M\beta_l - 3}{2}} \prod_{j=1}^{M}(w_l S_{ljk})^{\frac{\beta_l}{2}} \exp(-\frac{\beta_l w_l s_{ljk}}{2}) \quad \text{(A.8)}$$

$$p(w_l \mid S_{l1k}, \ldots, S_{lMk}, \beta_l) \propto \Gamma\big(\frac{M\beta_l + 1}{2}, \frac{2}{\sigma_y^{-2} + \beta_l \sum_{j=1}^{M} S_{ljk}}\big) \quad \text{(A.9)}$$

N_{jk}^{re} and N_{jk}^{irr} are the number of observations allocated to mixture j with feature k considered as relevant and irrelevant, respectively.

The complete posteriors for feature saliency ϕ with gamma parameters a and b, with n_{jk} the number of feature k relevant for component j can then be expressed by:

$$p(\rho_{jk} \mid \ldots) \propto \text{Beta}\,(a + n_{jk}, b + N - n_{jk}) \quad \text{(A.10)}$$

$$\begin{aligned} p(a \mid \ldots) &\propto a e^{-\frac{a}{2}} (\frac{\Gamma(a+b)}{\Gamma(a)})^M \prod_{j=1}^{M} \rho_{jk}^{a-1} \\ p(b \mid \ldots) &\propto b e^{-\frac{b}{2}} (\frac{\Gamma(a+b)}{\Gamma(b)})^M \prod_{j=1}^{M} (1-\rho_{jk})^{a-1} \end{aligned} \quad \text{(A.11)}$$

B The Variational Inference framework for Hierarchical Pitman-Yor Process mixture.

The variational lower bound is shown as:

$$
\begin{aligned}
\mathcal{L}_2 = & E_q[\log p(X \mid I, W, \mu, \sigma_l, \sigma_r)]) \\
& + E_q[\log p(I \mid \eta')] + E_q[\log p(\eta' \mid \gamma_a, \gamma_b)] \\
& + E_q[\log p(W \mid p')] + E_q[\log p(p' \mid \beta_a, \beta_b)] \\
& + E_q[\log p(\mu \mid \lambda, r)] + E_q[\log p(\sigma_l \mid \iota_l, \upsilon_l)] \\
& + E_q[\log p(\sigma_r \mid \iota_r, \upsilon_r)] \\
& - E_q[\log q(W, \eta', I, p', \mu, \sigma_l, \sigma_r)]
\end{aligned} \tag{B.1}
$$

The following steps are to optimize variational distribution:

$$
\varphi_{jtk} = \frac{\hat{\varphi}_{jtk}}{\sum_k^K \hat{\varphi}_{jtk}} \qquad \varrho_{jit} = \frac{\hat{\varrho}_{jit}}{\sum_t^T \hat{\varrho}_{jit}} \tag{B.2}
$$

$$
\begin{aligned}
\hat{\varphi}_{jtk} = & \exp\big\{E_q[\log \eta'_k] + \sum_{l=1}^{k-1} E_q[\log(1-\eta'_l)] \\
& - \sum_i^N E_q[W_{jit}]\widetilde{R}\big\} \\
\hat{\rho}_{jit} = & \exp\big\{E_q[\log p'_{jt}] + \sum_{s=1}^{t-1} E_q[\log(1-p'_{jt})] \\
& - \sum_k^K E_q[I_{jtk}]\widetilde{R}\big\}
\end{aligned} \tag{B.3}
$$

$$
\begin{aligned}
c_k &= 1 - \gamma_{ak} + \sum_j^M \sum_t^T E_q[I_{jtk}] \\
d_k &= \gamma_{bk} + k\gamma_{ak} + \sum_j^M \sum_t^T \sum_{l=k+1}^K E_q[I_{jtl}] \\
e_{jt} &= 1 - \beta_{ajt} + \sum_i^N E_q[W_{jit}] \\
f_{jt} &= \beta_{bjt} + t\beta_{ajt} + \sum_i^N \sum_{s=t+1}^T E_q[W_{jis}]
\end{aligned} \tag{B.4}
$$

$$
\begin{aligned}
\Sigma_{kd} =& \big(r + \sum_{j}^{M}\sum_{t}^{T}\sum_{i,X_{jid}<\mu_{kd}}^{N} \frac{E_q[W_{jit}]E_q[I_{jtk}]}{E_q[\sigma_{lkd}^2]} \\
&+ \sum_{i,X_{jid}\geq\mu_{kd}}^{N} \frac{E_q[W_{jit}]E_q[I_{jtk}]}{E_q[\sigma_{rkd}^2]}\big)^{-1} \\
m_{kd} =& \Sigma_{kd}\big(\lambda r + \sum_{j}^{M}\sum_{t}^{T}\sum_{i,X_{jid}<\mu_{kd}}^{N} \frac{E_q[W_{jit}]E_q[I_{jtk}]X_{jid}}{E_q[\sigma_{lkd}^2]} \\
&+ \sum_{i,X_{jid}\geq\mu_{kd}}^{N} \frac{E_q[W_{jit}]E_q[I_{jtk}]X_{jid}}{E_q[\sigma_{rkd}^2]}\big)
\end{aligned} \tag{B.5}
$$

The expectation of HPYP mixture of Asymmetric Gaussian are shown as:

$$
E_q[I_{jtk}] = \varphi_{jtk} \quad E_q[W_{jit}] = \varrho_{jit} \tag{B.6}
$$

$$
\begin{aligned}
E_q[\log \eta_k] &= E_q[\log \eta'_k] + \sum_{l=1}^{k-1} E_q[\log(1-\eta'_l)] \\
E_q[\log(\eta'_k)] &= \Psi(c_k) - \Psi(c_k + d_k) \\
E_q[\log(1-\eta'_k)] &= \Psi(d_k) - \Psi(c_k + d_k)
\end{aligned} \tag{B.7}
$$

$$
\begin{aligned}
E_q[\log p_{jt}] &= E_q[\log p'_{jt}] + \sum_{s=1}^{t-1} E_q[\log(1-p'_{jt})] \\
E_q[\log(p'_{jt})] &= \Psi(e_{jt}) - \Psi(e_{jt} + f_{jt}) \\
E_q[\log(1-p'_{jt})] &= \Psi(f_{jt}) - \Psi(e_{jt} + f_{jt})
\end{aligned} \tag{B.8}
$$

The expectation of parameters of AGD are same as the HDP mixture. Since BBVI adapts to different model, we would not need to drive again.