Robust Many-to-Many Location Sharing Based on Diffie-Hellman Encrypted Key Exchange

Abstract—**A novel unsupervised Bayesian image categorization framework based on asymmetric Gaussian mixture (AGM) model is proposed and the mixture parameter estimation is achieved by sampling-based reversible jump Markov chain Monte Carlo (RJMCMC) method. Previous researches have reveled that AGM outperforms classic symmetric mixture models (i.e Gaussian mixture model (GMM)) since the model adapts both symmetric and asymmetric datasets yielding better fitting accuracy. Moreover, the introduction of RJMCMC, a hybrid self-adapted sampling-based MCMC implementation, enables model transfer throughout parameter learning process, therefore, automatically converges to the optimal number of categories. In order to better identify visual features from the challenging UIUC sport events dataset, the image representative data is generated by adopting scale-invariant feature transform (SIFT), bag-of-visual-words (BOVW) and probabilistic latent semantic analysis (pLSA) techniques. Eventually, irrelevant and unneeded information will be filtered by feature selection. A comparison between AGM and other popular classifiers is given to discover its merits and the direction of future work is suggested.**

Index Terms—**Asymmetric Gaussian Mixture, Image Categorization, RJMCMC, SIFT, BOVW, pLSA**

I. INTRODUCTION

Recently, as the consequence of frequent usage of mobile phone, social media and cloud storage, digitalized visual data such as photos and pictures brings difficulties for management and analysis. Unlike those within text-based documents, indexing and comparison among images could be challenging. Therefore, image categorization is becoming one of the most interesting research topics in computer vision community. Indeed, finding relevant images from a rapidly growing unannotated image database is challenging which makes the previous time-consuming manual categorization methods infeasible. Automated methods such as machine-learning-based approaches [1] introduce image representation methodologies for visual feature extraction and both generative and discriminative classifiers for categorization. Meanwhile, modern machine-learning-based solutions can be divided into two main streams, classification-based supervised and clustering-based unsupervised ones. Compared to supervised solutions, unsupervised approach has no assumption on the number of groups, therefore, friendly to new added images and categories which makes it more suitable for increasing datasets. Moreover, it also immunizes against learning biases and overfitting problems that commonly exist in most supervised approaches if model training is inappropriate. Consequently, unsupervised categorization of images or image parts [2]–[4] acts an important role for image and video summarization, human action recognition, and image search etc,. It can also be seen as a pre-

processing step for supervised methodologies for classification or segmentation. As upgrade of single-mathematical-model-based methodologies, mixture models [5]–[7] can be seen as a superimposition of certain mixture components sharing dependencies with each other, therefore, lead to outstanding performance especially for high-dimensional and multi-cluster datasets. More precisely, for Gaussian-like datasets, Gaussian mixture model (GMM) [8] demonstrated satisfactory fitting abilities on applications such as computer vision, pattern recognition and data mining. However, under a more general circumstances regarding to non-Gaussian or asymmetric datasets, asymmetric Gaussian mixture (AGM) model [9] leads to a better accuracy by introducing two variance parameters for both left and right parts of asymmetric Gaussian distribution, providing more flexibility for variant real applications. Therefore, the justification of choosing AGM model and its merits will be discussed in the following chapters.

As an essential step for mixture model in practise, parameter estimation could be challenging and highly affects the performance. One of the most popular learning algorithm is the maximum-likelihood-based expectation maximization (EM) [10] which has been proven as effective under some circumstances. However, as a deterministic solution, the efficiency and usability of EM could be significantly compromised by bad initialization and overfitting problems [11] [12] which usually cause slow convergence and improper approximation, and eventually, compromise the accuracy of the model. As a consequence, researchers are paying more attention to sampling-based stochastic Bayesian learning algorithms [13], [14], such as Markov Chain Monte Carlo (MCMC) based implementations [15] to address previous issue by introducing prior and posterior distributions for every mixture parameter which decouple the dependencies between mixture parameters and mixture components, therefore, allow model adjustment by substituting proposed distributions to adapt to varied application datasets in a more flexible way [16], [17]. Accordingly, we chose an advanced MCMC implementation called reversible jump MCMC (RJMCMC) [12] which is based on a hybrid Metropolis-Hastings within Gibbs sampling [11] solution, combining both Metropolis-Hastings [18] and Gibbs sampling [19] methods because the main difficulty of applying traditional MCMC method is that, under some circumstances, direct sampling is not always straightforward that distributions of mixture parameters are latent and dependencies between parameters are unknown. By integrating the merits of both methods, mixture parameters will be evaluated iteratively and, eventually, the optimal parameter

values will be identified after convergence. Furthermore the self-adapted learning process [12] treats component number as an extra parameter and adjust it throughout iterations by automatically increasing (component birth/death step) and decreasing (component merge/split step) according to current status, therefore, enables model transfer which significantly improves the learning performance.

Before deploying UIUC sports event database [20] to validate AGM framework, image processing is needed because images should be represented as visual features and, eventually, be translated into numerical data. For this reason, we decided to adopt scale-invariant feature transform (SIFT) [21] to detect and describe image features even under changes in image scale, noise and illumination. However, generated SIFT features have to be categorized and features belong to each image should be counted as histograms to be the input of AGM model. Therefore, bag-of-visual-words [22] and probabilistic latent semantic analysis (pLSA) [23] are responsible for the generation of feature histogram and reduction of dimensionality.

The next sections are organized as follows. Section II and III elaborate the AGM model and its Bayesian learning algorithm. Section IV is devoted to results analysis and Section V gives a conclusion of the paper and proposes future research directions.

II. ASYMMETRIC GAUSSIAN MIXTURE MODEL

The likelihood function of AGM model [9] with M mixture components can be illustrated as follows:

$$p(\mathcal{X}|\Theta) = \prod_{i=1}^{N} \sum_{j=1}^{M} p_j p(X_i|\xi_j) \tag{1}$$

where $\mathcal{X} = (X_1, ..., X_N)$ reprensents the dataset with N observations, $\Theta = \{p_1, ..., p_M, \xi_1, ..., \xi_M\}$ defines the mixture parameters set of AGM mixture model including component weight p_j ($0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$) and asymmetric Gaussian distribution (AGD) parameters set ξ_j for mixture component j. Assuming the dataset $\mathcal{X}$ is d-dimensional, for each observation $X_n = (x_{n1}, ..., x_{nd}) \in \mathcal{X}$, the probability density function [9] for j-th component of the model can be defined as follows:

$$p(X_n|\xi_j) \propto \prod_{k=1}^{d} \frac{1}{(\sigma_{l_{jk}} + \sigma_{r_{jk}})}$$
$$\times \begin{cases} \exp\left[-\frac{(x_{nk}-\mu_{jk})^2}{2(\sigma_{l_{jk}})^2}\right] & if \ x_{nk} < \mu_{jk} \\ \exp\left[-\frac{(x_{nk}-\mu_{jk})^2}{2(\sigma_{r_{jk}})^2}\right] & if \ x_{nk} \geq \mu_{jk} \end{cases} \tag{2}$$

parameters set of component j is $\xi_j = (\mu_j, \sigma_{lj}, \sigma_{rj})$ where $\mu_j = (\mu_{j1}, ..., \mu_{jd})$ is the mean, $\sigma_{lj} = (\sigma_{lj1}, ..., \sigma_{ljd})$ and $\sigma_{rj} = (\sigma_{rj1}, ..., \sigma_{rjd})$ represents the left and right standard deviation vectors of AGD .

We bring a M-dimensional membership vector Z to each observation $X_i \in \mathcal{X}, Z_i = (Z_{i1}, ..., Z_{iM})$, indicating which specific component X_i belongs to [24], such that:

$$Z_{ij} = \begin{cases} 1 & \text{if } X_i \text{ belongs to component } j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

that being said, $Z_{ij} = 1$ only when observation X_i has the highest probability of belonging to component j and accordingly, for other components, $Z_{ij} = 0$.

Hence, the complete likelihood function can be obtained by combining Eq. (1) and Eq. (3) as follows:

$$p(\mathcal{X}, Z|\Theta) = \prod_{i=1}^{N} \prod_{j=1}^{M} (p_j p(X_i|\xi_j))^{Z_{ij}} \tag{4}$$

A. Priors and Posteriors

As mentioned before, MH-within-Gibbs based RJMCMC learning algorithm implementation introduces definitions of priors and posteriors for mixture weights and parameters to avoid direct sampling. For a specific iteration t, since mixture weight p_j satisfies $0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$, a natural choice of the prior is Dirichlet distribution [25] as follows:

$$\pi(p_j^{(t)}) \sim \mathcal{D}(\gamma_1, ..., \gamma_M) \tag{5}$$

where $\gamma = \gamma_j = \cdots = \gamma_M$ is a known hyperparameter. By taking the membership vector Z into account as a condition, the posterior probability of mixture weight p_j is defined as follows:

$$p(p_j^{(t)}|Z^{(t)}) \sim \mathcal{D}(\gamma_1 + n_1^{(t)}, ..., \gamma_M + n_M^{(t)}) \tag{6}$$

where n_j represents the number of observations of component j which could be calculated using membership vectors as follows:

$$n_j^{(t)} = \sum_{i=1}^{N} Z_{ij} \ (j = 1, ..., M) \tag{7}$$

The sampling process of mixture parameters employs the same concept. The proposal posterior distribution is $\xi^{(t)} \sim q(\xi|\xi^{(t-1)})$. More specifically, for parameters of AGM model $\xi^{(t)} = (\mu^{(t)}, \sigma_l^{(t)}, \sigma_r^{(t)})$, we choose d-dimensional Gaussian distributions as posterior distributions respectively:

$$\mu_j^{(t)} \sim \mathcal{N}_d(\mu_j^{(t-1)}, \Sigma)$$
$$\sigma_{lj}^{(t)} \sim \mathcal{N}_d(\sigma_{lj}^{(t-1)}, \Sigma)$$
$$\sigma_{rj}^{(t)} \sim \mathcal{N}_d(\sigma_{rj}^{(t-1)}, \Sigma) \tag{8}$$

where Σ is d x d identity matrix which makes the sampling a random walk MCMC process. Correspondingly, the priors are $\mu \sim \mathcal{N}_d(\eta, \Sigma)$ and $\sigma_l, \sigma_r \sim \mathcal{N}_d(\tau, \Sigma)$ given known hyperparameters η and τ.

B. Feature Selection

The AGM model defined in Eq. (1) assumes that all the d features of observations have the same weight of importance and carry pertinent information which is not always the case and many of those features can be irrelevant for clustering purpose. In order to tackle this problem and define relevance and importance of features, feature selection techniques [25] [26] should be taken into consideration. By denoting background Gaussian distributions for all the d features with parameter set $\Psi = \{\mu'_1, ..., \mu'_d, \sigma'_1, ..., \sigma'_d\}$, where μ' and σ' represent the mean and standard deviation of the Gaussian distribution respectively. Then, Eq. (1) can be reformulated with the feature relevancy approach suggested in Ref. [27] as follows:

$$p(\mathcal{X}|\Theta, \Psi, \Phi) = \prod_{i=1}^{N} \sum_{j=1}^{M} p_j \prod_{k=1}^{d} p(X_{ik}|\xi_{jk})^{\phi_k} p(X_{ik}|\psi_k)^{1-\phi_k} \tag{9}$$

where $\psi_k = (\mu'_k, \sigma'_k)$ and $\Phi = (\phi_1, \ldots, \phi_d)$ is a binary relevancy vector where $\phi_k = 1$ if k-th feature is relevant or $\phi_k = 0$ otherwise. If we consider the relevancy vector Φ as a latent variable, the complete likelihood function of AGM model with full parameter set will be given as follows:

$$p(\mathcal{X}|\Theta') = \prod_{i=1}^{N} \sum_{j=1}^{M} p_j \prod_{k=1}^{d} [\omega_k p(X_{ik}|\xi_{jk}) + (1 - \omega_k)p(X_{ik}|\psi_k)] \tag{10}$$

where $\Theta' = (\Theta, \Psi, \Omega)$ and $\Omega = (\omega_1, \ldots, \omega_d)$ is the relevancy weight with value range of $0 \leq \omega_d \leq 1$ which represents the probability that k-th feature is relevant. Finally, the calculation of relevancy weight ω_k is given as follows:

$$\omega_k = \frac{\sum_{j=1}^{M} p_j p(X_k|\xi_{jk})}{\sum_{j=1}^{M} p_j p(X_k|\xi_{jk}) + \sum_{j=1}^{M} p_j p(X_k|\psi_k)} \tag{11}$$

Therefore, irrelevant features only have few contribution for the clustering process, thus the usability of AGM model is extended to more common and complicated cases such as high-dimensional noisy applications.

III. BAYESIAN LEARNING ALGORITHM VIA RJMCMC

MH-within-Gibbs: MH-within-Gibbs method, a sampling-based learning algorithm, performs random sampling from posteriors of parameters, and then calculates the acceptance ratio r in order to make a decision whether the new samples should be accepted or discarded for next iteration. Due to the usage of membership vector Z, the mixture weight p_j can be derived within Gibbs sampling part. Therefore, it will be excluded from the calculation of the acceptance ratio r which is defined as follows:

$$r = \frac{p(\mathcal{X}|\Theta^{(t)})\pi(\Theta^{(t)})q(\Theta^{(t-1)}|\Theta^{(t)})}{p(\mathcal{X}|\Theta^{(t-1)})\pi(\Theta^{(t-1)})q(\Theta^{(t)}|\Theta^{(t-1)})} \tag{12}$$

Further information about the calculation of acceptance ratio r is explained in Appendix A. Once acceptance ratio r is derived, acceptance probability $\alpha = min[1, r]$ [28] could be computed. Then $u \sim U_{[0,1]}$ is supposed to be generated randomly. If $\alpha < u$, the proposed move should be accepted and parameters should be updated by $p^{(t)}$ and $\xi^{(t)}$ for next iteration. Otherwise, we discard $p^{(t)}, \xi^{(t)}$ and set $p^{(t)} = p^{(t-1)}$, $\xi^{(t)} = \xi^{(t-1)}$.

RJMCMC moves: Traditional MH-within-Gibbs algorithm assumes that the component number M is given and persistent throughout the learning process. However, because of bad initialization or just information leakage, M could be inaccurate or unknown. Under these circumstances, RJMCMC algorithm presents its merits by providing extra four independent steps (birth/death steps and merge/split steps) into learning process which could change component number M, therefore, brings more generalities.

In practice, within every RJMCMC learning iteration, the current component number m is considered as an extra parameter which has a proposed Poisson prior $\mathcal{P}(\lambda)$ with $\lambda = 4$ particularly in our case [8]. Accordingly, let M_{min} and M_{max} denote the minimum and maximum number of components M, and assume the probabilities of performing birth/split and death/merge steps are b_m and $d_m = 1 - b_m$ for $m = M_{min}, \ldots, M_{max}$ respectively. Obviously, $b_{M_{max}} = 0$ and $d_{M_{min}} = 0$. Correspondingly, $d_{M_{max}} = 1 - b_{M_{max}} = 1$ and $b_{M_{min}} = 1 - d_{M_{min}} = 1$. For $m = M_{min} + 1, \ldots, M_{max} - 1$, for simplification purpose, we choose the same value for both b_m and d_m as $b_m = d_m = 0.5$. Within every iteration, we generate a random value $u' \sim U_{[0,1]}$ respectively for the four RJMCMC steps. If $b_m >= u'$ or $d_m >= u'$, birth/split or death/merge steps should be performed correspondingly [8].

Merge and Split Steps: Randomly choose two components (j_1, j_2) satisfying that $\mu_{j_1} < \mu_{j_2}$ with no other μ_j in the interval $[\mu_{j_1}, \mu_{j_2}]$. The newly merged component j' will contain the
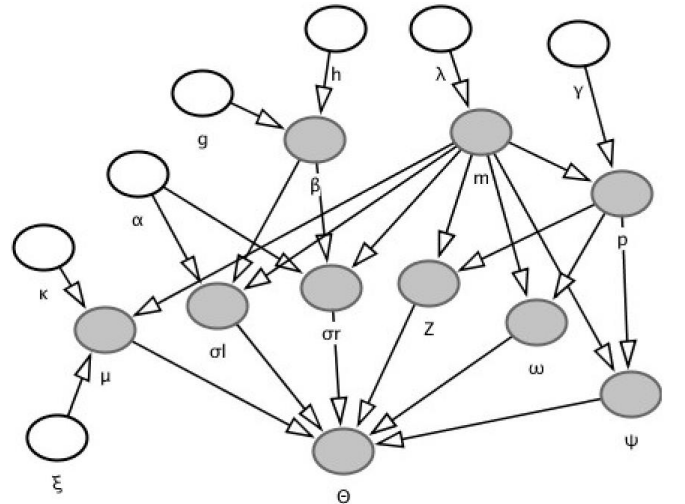

Fig. 1: DAG of RJMCMC Bayesian parameter learning network

observations that previously belonged to both component j_1 and j_2. Meanwhile, reduce current value of component number m to $m-1$, then calculate mixture weight and parameters for j' as follows:

$$p_{j'} = p_{j_1} + p_{j_2}$$
$$p_{j'}\mu_{j'} = p_{j_1}\mu_{j_1} + p_{j_2}\mu_{j_2}$$
$$p_{j'}(\mu_{j'}^2 + \sigma_{j'l}^2) = p_{j_1}(\mu_{j_1}^2 + \sigma_{j_1 l}^2)$$
$$+ p_{j_1}(\mu_{j_1}^2 + \sigma_{j_1 l}^2)$$
$$p_{j'}(\mu_{j'}^2 + \sigma_{j'r}^2) = p_{j_1}(\mu_{j_1}^2 + \sigma_{j_1 r}^2)$$
$$+ p_{j_1}(\mu_{j_1}^2 + \sigma_{j_1 r}^2) \quad (13)$$

As a reverse of merge step, we split component j' into two (j_1 and j_2) with 3 degrees of freedom ($u_1 \sim Beta(2,2), u_2 \sim Beta(2,2), u_3 \sim Beta(1,1)$) and, accordingly, increase m to $m+1$. Therefore, mixture parameters for split components can be calculated as follows:

$$p_{j_1} = p_{j'}u_1, \quad p_{j_2} = p_{j'}u_2$$
$$\mu_{j_1} = \mu_{j'} - \frac{u_2(\sigma_{j'l} + \sigma_{j'r})}{2}\sqrt{\frac{p_{j_2}}{p_{j_1}}}$$
$$\mu_{j_2} = \mu_{j'} + \frac{u_2(\sigma_{j'l} + \sigma_{j'r})}{2}\sqrt{\frac{p_{j_1}}{p_{j_2}}}$$
$$\sigma_{j_1 l}^2 = u_3(1 - u_2^2)\sigma_{j'l}^2\frac{p_{j'}}{p_{j_1}}$$
$$\sigma_{j_1 r}^2 = u_3(1 - u_2^2)\sigma_{j'r}^2\frac{p_{j'}}{p_{j_1}}$$
$$\sigma_{j_2 l}^2 = (1 - u_3)(1 - u_2^2)\sigma_{j'l}^2\frac{p_{j'}}{p_{j_2}}$$
$$\sigma_{j_2 r}^2 = (1 - u_3)(1 - u_2^2)\sigma_{j'r}^2\frac{p_{j'}}{p_{j_2}} \quad (14)$$

In order to decide whether the merge and split steps should be accepted or not, the acceptance probability [8] can be derived as follows:
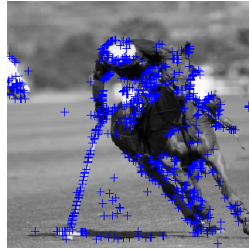
Fig. 2: a) Original UIUC sport image. b) Enhanced image with SIFT extracted visual features.

$$\mathcal{A} = \frac{p(\mathcal{X}, Z|\Theta')}{p(\mathcal{X}, Z|\Theta)}\frac{m'\mathcal{P}(m'|\lambda)}{\mathcal{P}(m|\lambda)}\frac{p_{j_1}^{\gamma-1+n_1}p_{j_2}^{\gamma-1+n_2}}{p_{j'}^{\gamma-1+n_1+n_2}Beta(\gamma, m\gamma)}$$
$$\times \sqrt{\frac{\kappa}{2\pi}}\exp[-\frac{1}{2}\kappa(\mu_{j_1} - \xi) + (\mu_{j_2} - \xi) + (\mu_{j'} - \xi)]$$
$$\times \frac{\beta^\alpha}{\Gamma(\alpha)}(\frac{\sigma_{j_1 l}^2\sigma_{j_1 r}^2\sigma_{j_2 l}^2\sigma_{j_2 r}^2}{\sigma_{j'l}^2\sigma_{j'r}^2})^{-\alpha-1}$$
$$\times \exp[-\beta(\sigma_{j_1 l}^2 + \sigma_{j_1 r}^2 + \sigma_{j_2 l}^2 + \sigma_{j_2 r}^2 - \sigma_{j'l}^2 - \sigma_{j'r}^2)]$$
$$\times \frac{d_{m'}}{b_m P_{alloc}}[Beta(\mu_1|2,2)Beta(\mu_2|2,2)Beta(\mu_3|1,1)]^{-1}$$
$$\times \frac{p_{j'}|\mu_{j_1} - \mu_{j_2}|\sigma_{j_1 l}^2\sigma_{j_1 r}^2\sigma_{j_2 l}^2\sigma_{j_2 r}^2}{\mu_2(1 - \mu_2^2)\mu_3(1 - \mu_3)\sigma_{j'l}^2\sigma_{j'r}^2} \quad (15)$$

where Θ' and $m' = m+1$ denote the mixture parameters set and the component number respectively before merge or after split steps. κ is a known hyperparameter and ξ is the midpoint of the variation interval of the involved data observations. Besides, P_{alloc} is the probability of which this particular allocation is made. Therefore, the acceptance probability for merge step is $\min(1, \mathcal{A})$ and, correspondingly, for split step is $\min(1, \mathcal{A}^{-1})$.

Birth and Death Steps: Compared to merge and split steps, birth and death steps are relatively straightforward because the newborn and dead components are empty ones which means parameter re-calculation is not needed. Mixture weight p_{new} in birth step can be obtained by sampling from Beat distribution $p_{new} \sim Beta(1, m)$ and mixture parameters can be derived from the priors as follows [29]:

$$\mu \sim \mathcal{N}(\xi, \kappa^{-1}), \quad \sigma_l^{-2}, \sigma_r^{-2} \sim \Gamma(\alpha, \beta), \quad \beta \sim \Gamma(g, h) \quad (16)$$

where hyperparameters κ, α, g and h are estimated by data. For death step, an empty component should be randomly selected and deleted among the existing components if there is any. Otherwise, this step will be skipped. After birth and death steps, mixture weights p_j should be re-scaled so that all weights sum to 1. Acceptance probability for birth and death steps is also required as the one for merge and split steps whose definition is as follows:

$$\mathcal{A}' = \frac{\mathcal{P}(m'|\lambda)}{\mathcal{P}(m|\lambda)}\frac{1}{Beta(m\gamma, \gamma)}p_{j'}^{\gamma-1}(1 - p_{j'})^{N+m\gamma-m}m'$$
$$\times \frac{d_{m'}}{(m_0 + 1)b_m}\frac{1}{Beta(p_{j'}|1, m)}(1 - p_{j'})^m \quad (17)$$

where m_0 is the amount of empty components. Thus, the probabilities of occurrence of birth and death steps are $\min(1, \mathcal{A}')$ and $\min(1, \mathcal{A}'^{-1})$ [8].

Finally, Figure 1 describes the dependencies between constants and variables involved in the Bayesian network of RJMCMC mixture parameter learning, and then, a typical learning procedure of AGM can be summarized as follows:

Input: Data observations $\mathcal{X}$ and component number M
Output: AGM mixture parameter set Θ

1) Initialization
2) Step t: For $t = 1, \ldots$

      Gibbs sampling part
  a) Generate $Z^{(t)}$ from Eq. (3)
  b) Compute $n_j^{(t)}$ from Eq. (7)
  c) Generate $p_j^{(t)}$ from Eq. (6)
      Metropolis-Hastings part
  d) Sample $\xi_j^{(t)}$ $(\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})$ from Eqs. (8)
  e) Compute acceptance ratio r from Eq. (12)
  f) Generate $\alpha = min[1, r]$ and $u \sim U_{[0,1]}$
  g) If $\alpha \geq u$ then $\xi^{(t)} = \xi^{(t-1)}$
      RJMCMC part
  h) Generate $u' \sim U_{[0,1]}$. If $b_m >= u'$, perform split or birth step, then calculate acceptance probability $\mathcal{A}$. If the step is accepted, set $m = m + 1$.
  i) Generate $u' \sim U_{[0,1]}$. If $d_m >= u'$, perform merge or death step, then calculate acceptance probability $\mathcal{A}'$. If the step is accepted, set $m = m - 1$.

IV. Experimental results

This section gives a detailed discussion around model adjustment, evaluation and effectiveness validation through the UIUC sports event dataset. A comparison is made between K-means, Gaussian mixture model (GMM) and proposed approach concerning different scenarios. Before apply the proposed model onto real dataset, hyperparameters related to the calculations of mixture weights and other parameters are set as follows: $\gamma_j = 1$ and both η and τ are considered as d-dimensional zero vectors. Consequently, hyperparameters for priors of mixture parameters are set as follows [30]:

$$\kappa = \frac{1}{\mathcal{R}^2}, \quad \alpha = 3, \quad g = 0.3, \quad h = \frac{100g}{\alpha \mathcal{R}^2} \quad (18)$$

where $\mathcal{R}$ is the interval of variation of observations.

A. Image Categorization

A challenging UIUC sports event database [20] is selected for our target application which has been evaluated by previous researches [31] [2]. It has 1579 images in total and consists of 8 sports event categories: 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). The first step of image pre-processing is applying scale-invariant feature transform (SIFT) on the original image files using difference of Gaussian (DoG) [21] (Figure 2) as interest point detector and then, visual feature will be translated into 128-dimensional feature descriptor vectors. Next, bag-of-visual-words (BOVW) [32] clusters feature vectors into a visual vocabulary $\mathcal{W}$ with variant vocabulary size using K-means algorithm. Consequently, each image will be represented by a frequency histogram of occurrence of visual words in $\mathcal{W}$. Since the vocabulary size in our test is between 200 to 1000, we also adopted probabilistic latent semantic analysis (pLSA) to describe each image as several latent topics (aspect) and therefore, reduce the dimension of image representation. Before applying AGM model for clustering, normalization based on feature scaling method is added to restrain the range of numerical attributes which will improve the Bayesian learning performance. Finally, we deploy the proposed AGM model as an unsupervised classifier to categorize the whole database. The proposed model is tested against image representation generated with different vocabulary size between 200 to 1000 with an interval of 100 and latent aspect number between 15 to 50 with an interval of 5 in order to identify the best accuracy.

According to a comparison between K-means, Gaussian mixture model (GMM) and proposed AGM model revels the fact that applying AGM to the sports event database leads to a significant clustering accuracy boost due to the best accuracy numbers of all the 3 classifiers illustrated in figure 3(a) (K-means: 22.17%, GMM: 29.26% and AGM: 46.04%). At the meanwhile, the impact of accuracy from different latent aspect numbers can be found in figure 3(b). Finally, the detailed clustering confusion matrix of AGM under the optimal vocabulary size (1000) and latent aspect number (30) is described in figure 4.
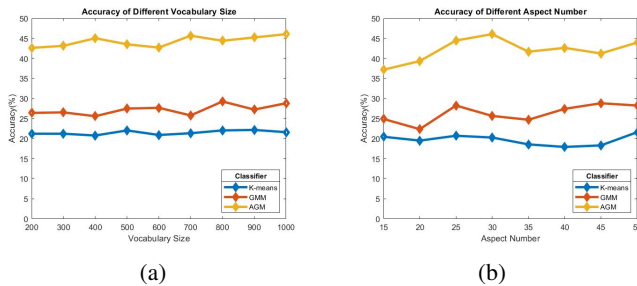

(a)   (b)

Fig. 3: a) Clustering accuracy by vocabulary size. b) Clustering accuracy by latent aspect number.

Fig. 4: Confusion matrix of AGM model (Vocabulary size is 1000 and latent aspect number is 30)

V. CONCLUSION AND FUTURE WORK

In this paper, we proposed a novel Bayesian framework based on asymmetric Gaussian mixture model via reversible jump Markov chain Monte Carlo method with feature selection and then, extended it to image categorization application which is a challenging UIUC sports event database. The proposed AGM model includes feature selection which can not only filters irrelevant and unneeded features but also weights relevant features based on the pertinent information they carry. For the parameter learning part, the proposed approach combines the merits of both Metropolis-Hastings and Gibbs sampling methods and allows model transfer throughout iterations by adopting RJMCMC methodology. A horizontal comparison between other unsupervised classifiers showing a significant accuracy improvement and future research directions will be focus on model adjustments and improvements to tackle high-dimensional datasets and achieve high clustering accuracy.

REFERENCES

[1] Y. Han and X. Qi, "Machine-learning-based image categorization," in *International Conference Image Analysis and Recognition*. Springer, 2005, pp. 585–592.

[2] W. Fan and N. Bouguila, "Infinite dirichlet mixture model and its application via variational bayes," in *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, vol. 1. IEEE, 2011, pp. 129–132.

[3] N. Bouguila, "Bayesian hybrid generative discriminative learning based on finite liouville mixture models," *Pattern Recognition*, vol. 44, no. 6, pp. 1183–1200, 2011. [Online]. Available: https://doi.org/10.1016/j.patcog.2010.12.010

[4] M. Azam and N. Bouguila, "Unsupervised keyword spotting using bounded generalized gaussian mixture model with ICA," in *2015 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2015, Orlando, FL, USA, December 14-16, 2015*. IEEE, 2015, pp. 1150–1154.

[5] J. Yang, X. Liao, X. Yuan, P. Llull, D. J. Brady, G. Sapiro, and L. Carin, "Compressive sensing by learning a gaussian mixture model from measurements," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 106–119, Jan 2015.

[6] C. K. Wen, S. Jin, K. K. Wong, J. C. Chen, and P. Ting, "Channel estimation for massive mimo using gaussian-mixture bayesian learning," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1356–1368, March 2015.

[7] N. Bouguila, "Count data modeling and classification using finite mixtures of distributions," *IEEE Trans. Neural Networks*, vol. 22, no. 2, pp. 186–198, 2011.

[8] S. Richardson and P. J. Green, "On bayesian analysis of mixtures with an unknown number of components (with discussion)," *Journal of the Royal Statistical Society: series B (statistical methodology)*, vol. 59, no. 4, pp. 731–792, 1997.

[9] T. Elguebaly and N. Bouguila, "Background subtraction using finite mixtures of asymmetric gaussian distributions and shadow detection," *Mach. Vis. Appl.*, vol. 25, no. 5, pp. 1145–1162, 2014.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[11] N. Bouguila, D. Ziou, and R. I. Hammoud, "On bayesian analysis of a finite generalized dirichlet mixture via a metropolis-within-gibbs sampling," *Pattern Anal. Appl.*, vol. 12, no. 2, pp. 151–166, 2009.

[12] N. Bouguila and T. Elguebaly, "A fully bayesian model based on reversible jump MCMC and finite beta mixtures for clustering," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5946–5959, 2012.

[13] S. Bourouis, M. A. Mashrgy, and N. Bouguila, "Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection," *Expert Syst. Appl.*, vol. 41, no. 5, pp. 2329–2336, 2014. [Online]. Available: https://doi.org/10.1016/j.eswa.2013.09.030

[14] N. Bouguila, J. H. Wang, and A. B. Hamza, "A bayesian approach for software quality prediction," in *2008 4th International IEEE Conference Intelligent Systems*, vol. 2, Sept 2008, pp. 11–49–11–54.

[15] S. Fu and N. Bouguila, "Bayesian learning of finite asymmetric gaussian mixtures," in *in Proceedings of The 31st International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems Montreal, QC, CA, June 25-28, 2018*, 2018.

[16] N. Bouguila, D. Ziou, and R. I. Hammoud, "A bayesian non-gaussian mixture analysis: Application to eye modeling," in *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA*, 2007. [Online]. Available: https://doi.org/10.1109/CVPR.2007.383439

[17] S. Fu and N. Bouguila, "Asymmetric gaussian mixtures with reversible jump MCMC," in *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE) (CCECE 2018)*, Quebec City, Canada, May 2018.

[18] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[19] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," in *Readings in Computer Vision*. Elsevier, 1987, pp. 564–584.

[20] L.-J. Li and L. Fei-Fei, "What, where and who? classifying events by scene and object recognition," 2007.

[21] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[22] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 524–531.

[23] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine learning*, vol. 42, no. 1-2, pp. 177–196, 2001.

[24] N. Bouguila, D. Ziou, and E. Monga, "Practical bayesian estimation of a finite beta mixture through gibbs sampling and its applications," *Statistics and Computing*, vol. 16, no. 2, pp. 215–225, 2006.

[25] T. Elguebaly and N. Bouguila, "Simultaneous bayesian clustering and feature selection using rjmcmc-based learning of finite generalized dirichlet mixture models," *Signal Processing*, vol. 93, no. 6, pp. 1531–1546, 2013.

[26] ——, "Simultaneous high-dimensional clustering and feature selection using asymmetric gaussian mixture models," *Image and Vision Computing*, vol. 34, pp. 27–41, 2015.

[27] M. H. Law, M. A. Figueiredo, and A. K. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 9, pp. 1154–1166, 2004.

[28] D. Luengo and L. Martino, "Fully adaptive gaussian mixture metropolis-hastings algorithm," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*. IEEE, 2013, pp. 6148–6152.

[29] G. Casella, C. P. Robert, and M. T. Wells, "Mixture models, latent variables and partitioned importance sampling," *Statistical Methodology*, vol. 1, no. 1-2, pp. 1–18, 2004.

[30] M. Stephens, "Bayesian analysis of mixture models with an unknown number of components-an alternative to reversible jump methods," *Annals of statistics*, pp. 40–74, 2000.

[31] F. Najar, S. Bourouis, A. Zaguia, N. Bouguila, and S. Belghith, "Unsupervised human action categorization using a riemannian averaged fixed-point learning of multivariate ggmm," in *International Conference Image Analysis and Recognition*. Springer, 2018, pp. 408–415.

[32] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, vol. 1, no. 1-22. Prague, 2004, pp. 1–2.