Received: 14 July 2019 | Revised: 11 December 2019 | Accepted: 2 March 2020

DOI: 10.4218/etrij.2019-0336

ORIGINAL ARTICLE

ETRI Journal WILEY

Online nonparametric Bayesian analysis of parsimonious Gaussian mixture models and scenes clustering

Ri-Gui Zhou | **Wei Wang**

College of Information Engineering, Shanghai Maritime University, Shanghai, China

Correspondence
Wei Wang, College of Information Engineering, Shanghai Maritime University, Shanghai, China.
Email: wangjiusi_ex@163.com

Funding information
This research is supported by the National Key R&D Plan (No. 2018YFC1200200 and 2018YFC1200205) and the National Natural Science Foundation of China (No. 61463016).

The mixture model is a very powerful and flexible tool in clustering analysis. Based on the Dirichlet process and parsimonious Gaussian distribution, we propose a new nonparametric mixture framework for solving challenging clustering problems. Meanwhile, the inference of the model depends on the efficient online variational Bayesian approach, which enhances the information exchange between the whole and the part to a certain extent and applies to scalable datasets. The experiments on the scene database indicate that the novel clustering framework, when combined with a convolutional neural network for feature extraction, has meaningful advantages over other models.

KEYWORDS
Gaussian distribution, mixture model, neural network, nonparametric, scenes clustering

1 | INTRODUCTION

Cluster analysis is a subfield of machine learning. The main idea was to group a set of samples in such a way that samples in the same group are more similar (in a sense) to each other than to those in other groups. This refers to describing the structural information of the data source. In the past few years, it has made great progress and has been implemented in many interesting applications [1–3].

In model-based cluster analysis, the finite mixture model is an exceedingly popular and powerful statistical method [4–6]. From a statistical perspective, the actual datasets, such as computer vision, image processing, and signal processing, are often viewed as being generated from intractable distributions [7–9]. The statistical methods can estimate the parameters of the complex distributions, which can accurately describe the mathematical features of the data sources and divide datasets into unrelated clusters. In addition, real-world datasets are sometimes insufficient, and the finite mixture model can also be used as a data augmentation technique to meet the needs of actual production.

The finite mixture model based on Gaussian distributions (GMM) is a well-known probabilistic tool that possesses good generalization ability and achieves favorable performance in practice [10–12]. On one hand, the partial sum of random variable sequences asymptotically follows Gaussian distribution owing to the central limit theorem, making the GMM a robust and steady method. On the other hand, the GMM is analytically tractable owing to the probability density function having features that are easy to manage. Gaussian distributions are symmetrical and unbounded, which potentially assumes that the values of datasets observed range from negative infinity to positive infinity and the spread of datasets adopts symmetrical characteristics for some unknown samples. Furthermore, the main form of GMM with K components ($K \geq 1$), where each component is built by the same type of Gaussian distribution, can be described as follows:

$$p(x_i|\Theta) = \sum_{k=1}^{K} \pi_k p(x_i|\vartheta_k), \tag{1}$$

where $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$ is a random vector with value in D-dimensional Euclidean space. Different values of the subscripts (i) represent different samples that are independent of each other. $\Theta = (\pi_1, \ldots, \pi_k, \vartheta_1, \ldots, \vartheta_k)$ is the parameter set of GMM. π_k is the probability that one sample belongs to the kth component, subject to the constraints $\pi_k > 0$ and $\sum_{k=1}^{K} \pi_k = 1$. $\vartheta_k = (\mu_k, \varepsilon_k)$ indicates a specific Gaussian component, where μ_k represents the mean vector and ε_k represents the variance matrix.

Another challenging problem is the proper determination of the number of components in the finite mixture model, which is related to the performance of the model. The usual practice is to presuppose a K value, but this often leads to overfitting or underfitting unless the researchers have sufficient empirical knowledge of the data sources and are able to make the right choices. To deal with these troubles, the Bayesian nonparametric mixture model is gradually emerging [13,14], which has the Dirichlet process as its stepping stone [15]. By providing the model a special prerequisite, the number of components is not fixed in advance, but the model is assumed to have infinite hybrid components which means it has infinite parameters. When the model updates, the complexity is constantly and automatically adjusted to fit the acquired datasets [16,17].

In addition, effective feature processing is of great significance to the performance of the model. For instance, a combination of multi-source feature vectors has better practical effects than single-source feature vectors [18]. With the development of deep learning, unsupervised feature extraction using convolutional neural networks has become increasingly popular [19–21]. One of the key points is the adaptability of the domain [22–24]. Compared with traditional feature extraction methods, convolutional neural networks can better describe deep-level spatial structure information.

For the inference of the models, the Monte Carlo Markov chain (MCMC) is a very common method used with Bayesian nonparametric learning from source data [15]. Although MCMC is effective for parameters estimation, the parameters converge slowly, and it is difficult to diagnose their convergence. This is especially true for high-dimensional data that requires the computation of multidimensional integrals. Therefore, an alternative method of variational inference, that is a powerful deterministic approximation technique and has a faster convergence speed, is proposed [17,25]. However, as it processes whole datasets at once, it is only suitable for small-scale problems and can easily generate a locally optimal solution. In this context, we propose an online variational inference method that can effectively extend to large-scale problems while ensuring effective performance [26].

Motivated by the abilities of Bayesian nonparametric methods in dealing with model selection problem and the good performance obtained by the variational methods, we propose, in this paper, a new nonparametric Gaussian mixture model for large-scale scenes clustering based on the Dirichlet process and parsimonious Gaussian distribution. Combined with neural networks [27] and online variational inference [26], the new framework can make full use of feature information, strengthen the iteration of parameters in the model, and achieve good performance in real-life applications.

The remainder of this paper is as follows: In Section 2, the nonparametric mixture model is fully presented. Section 3 details the process of learning the parameters of the model through online variational inference. Experimental results are shown in Section 4. Finally, the conclusion is given in the last Section.

2 | NONPARAMETRIC PARSIMONIOUS GAUSSIAN MIXTURE MODELS

2.1 | Parsimonious Gaussian mixture models

Based on the local independence assumption: each element in x_i is conditionally independent of each other given the category of x_i [28], our framework first adds constraints to the variance matrix ε_k to construct a parsimonious Gaussian mixture model, in which ε_k is characterized by a diagonal structure. Thus, the probability density function $p(x_i|\vartheta_k)$ in (1) can be written as follows:

$$p(x_{i1}, x_{i2}, *, x_{iD}) = \begin{pmatrix} \mu_{k1} & \tau_{k1}^{-1} & & \\ \mu_{k2} & & \tau_{k2}^{-1} & \\ * & , & & * \\ \mu_{kD} & & & \tau_{kD}^{-1} \end{pmatrix}, \quad (2)$$

where $\varepsilon_k = I\sigma_k$ and defines precision as $\tau_k = 1/\sigma_k$.

The number of parameters in the parsimonious Gaussian mixture model grows linearly with the data dimension, which is especially important in high dimension situations [29]. However, the number of parameters in the non-diagonal covariance structure is quadratic with respect to the data dimension in (1). To some extent, the parsimonious Gaussian mixture model is more flexible and efficient for high dimension data.

2.2 | Nonparametric prior

We set the nonparametric prior knowledge of Dirichlet process (DP) for the model in (1), which addresses

the challenging problem of selecting the appropriate number of components [17]. DP can be succinctly described as: let G_0 be a non-atomic base probability distribution defined on measurable space (Φ,B), and α_0 is a positive real number. A random distribution G is called to be $G \sim DP(\alpha_0, G_0)$, if any m-partitions $\{A_1, A_2, \ldots, A_m\}$ of Φ with $A_l \in B$ meets the standard Dirichlet distribution:

$$(G(A_1), G(A_2), *, G(A_m)) \rightarrow Dir(\alpha_0 G_0(A_1), \alpha_0 G_0(A_2), \alpha_0 G_0(A_m)), \tag{3}$$

where m is a natural number.

Combined with the stick-breaking construction, we sample independently from the Beta distribution with parameters 1 and α_0. Then, the representation of $G \sim DP(\alpha_0, G_0)$ is given by

$$\begin{aligned} &\pi_k = v_k \prod_{l=1}^{k-1} (1 - v_l), \quad v_k \rightarrow Beta(1, \alpha_0), \\ &G = \sum_{k=1}^{\infty} \pi_k \delta_{\vartheta_k}, \quad \vartheta_k \rightarrow G_0, \end{aligned} \tag{4}$$

where $0 \le \pi_k \le 1$, $\sum_{k=1}^{\infty} \pi_k = 1$, and δ_{ϑ_k} denotes the Dirac delta measure centered at ϑ_k. Evidently, G is discrete and implies that the number of parameters of the model is infinite. Thus, the new model can be called Par-InGMM and the formula can be written as follows:

$$p(x_i|\Theta) = \sum_{k=1}^{\infty} \pi_k p(x_i|\vartheta_k). \tag{5}$$

Furthermore, the potential vector $z_i = (z_{i1}, z_{i2}, \ldots, z_{ik}, \ldots)$ is raised to denote a specific hybrid component, from which x_i is generated, and subject to the constraints: $z_{ik} \in \{0,1\}$ and $\sum_{k=1}^{\infty} z_{ik} = 1$. We enforce truncation (K) to make the model easier to handle. According to conditional independence and $X = (x_1, x_2, \ldots, x_N)$, the joint distribution between the observed datasets X and hidden variables in (5) can be factorized as follows:

$$p(X, z, v, \mu, \tau) = p(X|z, \mu, \tau) p(\mu) p(\tau) p(z|v) p(v). \tag{6}$$

For the sake of increasing the flexibility and plasticity of the model, it is necessary to add some extra layers. As we can see, the parameter τ_k is defined in the loose support $(0, \infty)$, then a vague and flexible prior of Gamma distribution is selected. And we let the Gaussian distribution be the prior distribution of μ_k, as shown below:

$$\begin{aligned} &p(\tau_k|a_k, b_k) = \frac{b_k^{a_k}}{\Gamma(a_k)} \tau_k^{a_k - 1} e^{-b_k \tau_k}, \\ &p(\mu_k|m_k, (\lambda_k \tau_k)^{-1}) = \frac{\sqrt{\lambda_k \tau_k}}{\sqrt{2\Pi}} e^{-\frac{\lambda_k \tau_k (\mu_k - m_k)^2}{2}}, \end{aligned} \tag{7}$$

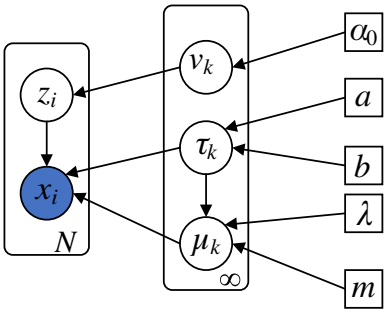

FIGURE 1 Probabilistic graphical model representation of the Par-InGMM [Colour figure can be viewed at wileyonlinelibrary.com]

where a, b, λ, m are hyperparameters like α_0. Figure 1 shows the corresponding probabilistic graphical model, in which all circular nodes represent stochastic variables with unfixed values, every rectangular node indicates a confirmed hyperparameter, arrows display the conditional dependent relations between nodes, and the boxes indicate that the variables need to be independently and identically repeated a certain number of times. Under the Bayesian analysis, the variables in (6) can be written as follows:

$$\begin{aligned} &p(X|z, \mu, \tau) = \prod_{n=1}^{N} \prod_{k=1}^{K} \left(\prod_{d=1}^{D} N(x_{nd}|\mu_{kd}, \tau_{kd}^{-1})^{z_{nk}} \right), \\ &p(v) = \prod_{k=1}^{K} Beta(v_k|1, \alpha_0), \\ &p(z|v) = \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{nk}}, \\ &p(\tau) = \prod_{k=1}^{K} \prod_{d=1}^{D} Gam(\tau_{kd}|a_{kd}, b_{kd}), \\ &p(\mu) = \prod_{k=1}^{K} \prod_{d=1}^{D} N(\mu_{kd}|m_{kd}, (\lambda_k \tau_{kd})^{-1}). \end{aligned} \tag{8}$$

3 | MODEL INFERENCE

Taking the advantages of exponential family distributions and conjugate priors, we utilize the online variational inference algorithm deduced from Reference [26] to solve the posterior distributions of Par-InGMM. Considering the variational distribution $q(z, v, \mu, \tau)$ of the fully factorization (in (10)) to approximate the true posterior distribution $p(z, v, \mu, \tau|X)$ and optimizing the parameters of q, we obtain the evidence lower bound objection (ELBO in (9)) that needs to be maximized to improve the degree of approximation.

$$\begin{aligned} ELBO(q) = E_q[&\log p(X, v, z, \mu, \tau|a, b, \lambda, m, \alpha_0) \\ &- \log q(v, z, \mu, \tau)]. \end{aligned} \tag{9}$$

$$\begin{aligned} q(z, v, \mu, \tau) = &\prod_{n=1}^{N} q(z_{nk}|\hat{r}_{nk}) \prod_{k=1}^{K} q(v_k|\hat{\alpha}_{k1}, \hat{\alpha}_{k2}) \\ &\times \prod_{k=1}^{K} \prod_{d=1}^{D} q(\tau_{kd}|\hat{a}_{kd}, \hat{b}_{kd}) \\ &\times \prod_{k=1}^{K} \prod_{d=1}^{D} q(\mu_{kd}|\hat{m}_{kd}, \hat{\lambda}_k). \end{aligned} \tag{10}$$

In the above equations, $\hat{r}_{nk}, \hat{\alpha}_{k1}, \hat{\alpha}_{k2}, \hat{a}_{kd}, \hat{b}_{kd}, \hat{\lambda}_k, \hat{m}_{kd}$ are the corresponding parameters that determine the approximate distributions.

Under the knowledge of mean-field variational inference and conjugate prior [17], we can tractably analyze the parameters in (10). The local parameter $\hat{r}_{nk}$ can be understood as the posterior probability that component k generates x_n, and satisfies the constraints

$$0 \le \hat{r}_{nk} \le 1, \sum_{k=1}^{K} \hat{r}_{nk} = 1. \tag{11}$$

The updated equations are

$$\begin{aligned} r_{nk} &= \exp\left(E_q[\log \pi_k(v)] + E_q[\log p(x_n|\mu_k, \tau_k^{-1})]\right), \\ \hat{r}_{nk} &= \frac{r_{nk}}{\sum_{l=1}^{K} r_{nl}}. \end{aligned} \tag{12}$$

Usually, we maintain expected mass $\hat{N}_k$ and sufficient statistics $s_k(X)$, $s_k\left(X^2\right)$ for each component k:

$$\begin{aligned} \hat{N}_k &= \sum_{n=1}^{N} \hat{r}_{nk}, \\ s_k(X) &= \sum_{n=1}^{N} \hat{r}_{nk} x_n, \\ s_k(X^2) &= \sum_{n=1}^{N} \hat{r}_{nk} x_n^2. \end{aligned} \tag{13}$$

As for global parameters, there are some simple modalities under the influence of the conjugate characteristic. The shape of Beta distribution is jointly governed by $\hat{\alpha}_{k1}$ and $\hat{\alpha}_{k2}$, as follows:

$$\begin{aligned} \hat{\alpha}_{k1} &= 1 + \sum_{n=1}^{N} \hat{r}_{nk} = 1 + \hat{N}_k, \\ \hat{\alpha}_{k2} &= \alpha_0 + \sum_{j=k+1}^{K} \sum_{n=1}^{N} \hat{r}_{nj} = \alpha_0 + \sum_{j=k+1}^{K} N_j. \end{aligned} \tag{14}$$

Analogously, $\hat{a}_{kd}$, $\hat{b}_{kd}$, $\hat{\lambda}_k$ and $\hat{m}_{kd}$ determine respectively the Gamma and Gaussian distributions:

$$\begin{aligned} \hat{a}_{kd} &= a_{kd} + 0.5 \times \sum_{n=1}^{N} \hat{r}_{nk} = a_{kd} + 0.5 \times \hat{N}_k, \\ \hat{b}_{kd} &= b_{kd} + 0.5 \times \Big(\sum_{n=1}^{N} (x_{nd} - \hat{m}_{kd})^2 \hat{r}_{nk} + \lambda_k (\hat{m}_{kd} - m_{kd})^2 \Big) \\ &= b_{kd} + 0.5 \\ &\quad \times (s_{kd}(X^2) - 2\hat{m}_{kd} s_{kd}(X) + \hat{m}_{kd}^2 \hat{N}_k + \lambda_k (\hat{m}_{kd} - m_{kd})^2), \\ \hat{\lambda}_k &= \lambda_k + \sum_{n=1}^{N} \hat{r}_{nk} = \lambda_k + \hat{N}_k, \\ \hat{m}_{kd} &= \frac{1}{\hat{\lambda}_k} \times (\lambda_k m_{kd} + \sum_{n=1}^{N} \hat{r}_{nk} x_{nd}) \\ &= \frac{1}{\hat{\lambda}_k} \times (\lambda_k m_{kd} + s_{kd}(X)). \end{aligned} \tag{15}$$

Variational posteriors are optimized by iteratively computing nether expected logarithmic values until convergence:

$$\begin{aligned} E_q[\log v_k] &= \psi(\hat{\alpha}_{k1}) - \psi(\hat{\alpha}_{k1} + \hat{\alpha}_{k2}), \\ E_q[\log (1 - v_k)] &= \psi(\hat{\alpha}_{k2}) - \psi(\hat{\alpha}_{k1} + \hat{\alpha}_{k2}), \\ E_q[\log \pi_k(v)] &= E_q[\log v_k] + \sum_{l=1}^{k-1} E_q[\log (1 - v_l)], \\ E_q[\log \tau_{kd}] &= \psi(\hat{a}_{kd}) - \log \hat{b}_{kd}, \\ E_q[\tau_{kd}] &= \hat{a}_{kd} / \hat{b}_{kd}, \\ E_q[\mu_{kd}] &= \hat{m}_{kd}, \\ E_q[\log p(x_n|\mu_k, \tau_k^{-1})] &\propto \sum_{d=1}^{D} (0.5 \times E_q[\log \tau_{kd}] \\ &\quad - 0.5 \times E_q[\tau_{kd}] \times (x_{nd} - E_q[\mu_{kd}])^2), \end{aligned} \tag{16}$$

where $\Psi(\cdot)$ is the digamma function. At this point, we can easily rewrite ELBO (q) to (17).

$$\sum_{k=1}^{K} \left(\begin{aligned} &\hat{N}_k E_q[\log \pi_k(v)] + E_q[\log \frac{Beta(v_k|1, \alpha_0)}{Beta(v_k|\hat{\alpha}_{k1}, \hat{\alpha}_{k2})}] \\ &+ E_q[\log \frac{N(\mu_k|m_k, (\lambda_k \tau_k)^{-1})}{N(\mu_k|\hat{m}_k, (\hat{\lambda}_k \tau_k)^{-1})}] \\ &+ E_q[\log \frac{Gam(\tau_k|a_k, b_k)}{Gam(\tau_k|\hat{a}_k, \hat{b}_k)}] \\ &+ \sum_{n=1}^{N} \hat{r}_{nk} E_q[\log N(x_n|\mu_k, \tau_k^{-1})] - \sum_{n=1}^{N} \hat{r}_{nk} \log \hat{r}_{nk} \end{aligned} \right) \tag{17}$$

Given sufficient statistics $\hat{N}_k, s_k(X)$ and $s_k(X^2)$, we can easily obtain updates to global parameters for each component. And ELBO (q) can be accurately calculated with extra corresponding expectations and $-\sum_{n=1}^{N} \hat{r}_{nk} \log \hat{r}_{nk}$. Once summary sufficient statistics are determined for all components, the structure of the entire model can be determined. Traditional variational inference computes summary sufficient statistics by processing whole datasets, which increases running time and reduces flexibility. Note that sufficient statistics and $-\sum_{n=1}^{N} \hat{r}_{nk} \log \hat{r}_{nk}$ have an additive property, which implies that we can process complete datasets in batches. From this point of view, we divide the datasets into several fixed blocks $\{B_1, B_2, \ldots, B_b\}$ and access those in random order. In the first pass, when accessing a single block (B_l) of datasets, we record and save local sufficient statistic $S_k^{B_l} = [\hat{N}_k, s_k(X), s_k(X^2)]$ for each component. And by means of additivity, we begin to construct and track global sufficient statistic $S_k = [\hat{N}_k, s_k(X), s_k(X^2)]$ by $S_k += S_k^{B_l}$. In the later pass, whenever visiting the data blocks, we not only update S_k with additivity, but also subtract the corresponding $S_k^{B_l}$ previously stored. As long as the overall sufficient statistics are obtained, we can update the global posterior parameters and prepare to calculate the subsequent local parameters. Similarly, for

$$H_k^{B_l} = -\sum_{n \in B_l} \hat{r}_{nk} \log \hat{r}_{nk}, H_k = \sum_{l=1}^{b} H_k^{B_l}, \tag{18}$$

we can also calculate ElBO (q) very quickly. In this way, the information flow between the local and the whole is enhanced, and the convergence speed is improved. On the

premise of elevating ELBO (q), birth and merge moves are also executed during accessing the data blocks to build a compact infinite mixture model (see Reference [26] for more details.).

4 | EXPERIMENT RESULTS AND ANALYSIS

We evaluated the model on the Places365-Standard image set which contains 18 million train images from 365 scene categories [30]. In our case, we randomly selected 10 classes from Places365-Standard, with 5000 images per class. The size of each image is 256 × 256 pixels. We compare Par-InGMM with three different Gaussian mixture models based on the Dirichlet process: full Gauss, zero-mean Gauss, and the parsimonious Gauss, with the prior knowledge of τ_k set to Wishart distribution. Three different feature extraction methods were used to illustrate the experimental results, including HOG, LBP, and VGG19 [27,31,32]. In addition, we took purity, completeness (COM), normalization mutual information (NMI), V-measure and adjusted mutual information (AMI) as performance evaluation indicators [33–35], and compared the efficiency by computation cycles after ten runs. Specifically, after obtaining the initial feature vector, we reduced the dimension to $D = 100$ by PCA to facilitate subsequent experimental processing. We ran the provided codes with default settings for other models to guarantee fairness. For the proposed Par-InGMM, we set the initial hyperparameters: $a_{kd} = 0.91, b_{kd} = 0.28, \lambda_k = 0.22, m_{kd} = 0$ and $\alpha_0 = 1$. Moreover we completely kept 50 epochs, initial $K = 1$ and 15 batches in the runtime.

Note that when we used VGG19 to extract image features, its weights are pre-trained based on the ImageNet and were not modified [36]. Then, we removed the top layer to obtain the feature vectors which fully represent the spatial and semantic information of the images. Throughout the clustering process, the label information was not used.

Table 1 shows the performance comparison of various clustering algorithms based on different features, it can be seen that Par-InGMM always achieves relatively better performance in a variety of evaluation criteria except for completeness. Par-InGMM, combined with VGG19 feature extraction, obtains satisfactory experimental results. Evidently, compared with the traditional feature descriptors, the deep neural network extracts more applicable structural information and greatly improves the experimental performance.

Generalization performance is determined by the ability of the learning algorithm, the adequacy of the datasets, and the difficulty of the learning task itself. From a quantitative point of view, the generalization error can be decomposed into the sum of deviation, variance, and noise. In fact, variance measures the change in learning performance caused by changes in the training set of the same size, which characterizes the impact of data perturbations. It is usually caused by the complexity of the model, as determined by the number of training samples. With this background, we analyzed the stability of Par-InGMM when used with deep neural networks for feature extraction. Specifically, we randomly sampled 10 datasets of the same size and calculated the output variance of all corresponding models. Table 2 shows the experimental results, where we can see that parsimonious Gaussian mixture models have stronger stability and anti-interference than non-parsimonious Gaussian mixture

TABLE 1 Performance comparison of different algorithms under different evaluation indexes condition on different feature processing.

Feature Metric Alg.	Full Gauss	Zero-mean Gauss	Par-Gauss (Wishart)	Par-InGMM
HOG				
purity	0.226	0.252	0.265	0.356
COM	0.303	0.283	0.220	0.221
NMI	0.209	0.214	0.182	0.231
V-measure	0.195	0.206	0.179	0.231
AMI	0.144	0.161	0.151	0.220
LBP				
purity	0.200	0.201	0.222	0.316
COM	0.116	0.117	0.140	0.164
NMI	0.089	0.090	0.120	0.181
V-measure	0.086	0.087	0.119	0.180
AMI	0.069	0.069	0.104	0.163
VGG19				
purity	0.420	0.376	0.476	0.746
COM	0.643	0.616	0.515	0.511
NMI	0.524	0.472	0.480	0.592
V-measure	0.513	0.456	0.479	0.585
AMI	0.427	0.362	0.448	0.510

TABLE 2 Variance comparison of different algorithms under different evaluation indexes condition on VGG19.

Feature Metric Alg.	Full Gauss	Zero-mean Gauss	Par-Gauss (Wishart)	Par-InGMM
VGG19				
purity	0.0042	0.0033	0.0034	0.0011
COM	0.0007	0.0003	0.00007	0.0002
NMI	0.0014	0.0012	0.0008	0.0003
V-measure	0.0020	0.0015	0.0007	0.0003
AMI	0.0031	0.0025	0.0008	0.0002

models. This may be because the parsimonious Gaussian distribution reduces the number of parameters in the non-parametric mixture models.

Concretely, Figure 2 presents the operational aspect of different algorithms under VGG19 feature extraction. We have observed two main situations. First, the parsimonious Gaussian model always discovers the diversity and complexity of datasets faster than the non-parsimonious Gaussian model. Second, the increase in the number of components will require significantly more computing resources, but under the same conditions, the parsimonious Gaussian model will consume less. Figure 3 shows some images from three random components developed by Par-InGMM, which indicates good actual effect.

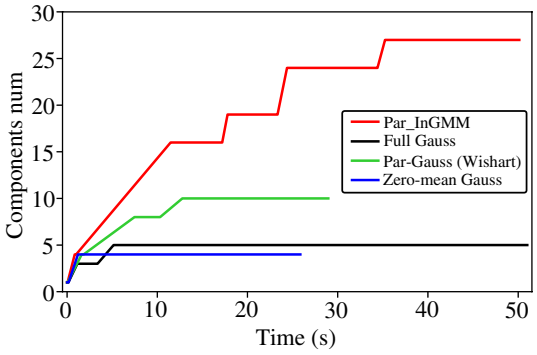

FIGURE 2 Comparison of the number of components generated by different algorithms during running time. The length of the line represents the execution time [Colour figure can be viewed at wileyonlinelibrary.com]

5 | CONCLUSION

In this paper, motivated by the importance of streaming data in some real-world applications, we proposed a Bayesian nonparametric statistical framework based on the Par-InGMM for large-scale scenes clustering. The Par-InGMM is based on the Dirichlet process, and the mixture components of this model are the parsimonious Gaussian distributions. Owing to these characteristics, the Par-InGMM overcomes the difficulty of model selection and has more flexibility compared to non-parsimonious Gaussian distributions. Furthermore, an online variational method, derived from truncated variational interference, was used as inference for the Par-InGMM, achieving scalability. With the pre-trained convolutional neural network performing early feature processing, the proposed Par-InGMM completes considerable large-scale scenes clustering in real data evaluation. Experiments demonstrated that Par-InGMM has more evident advantages than other nonparametric Gaussian models for a small number of iterations.

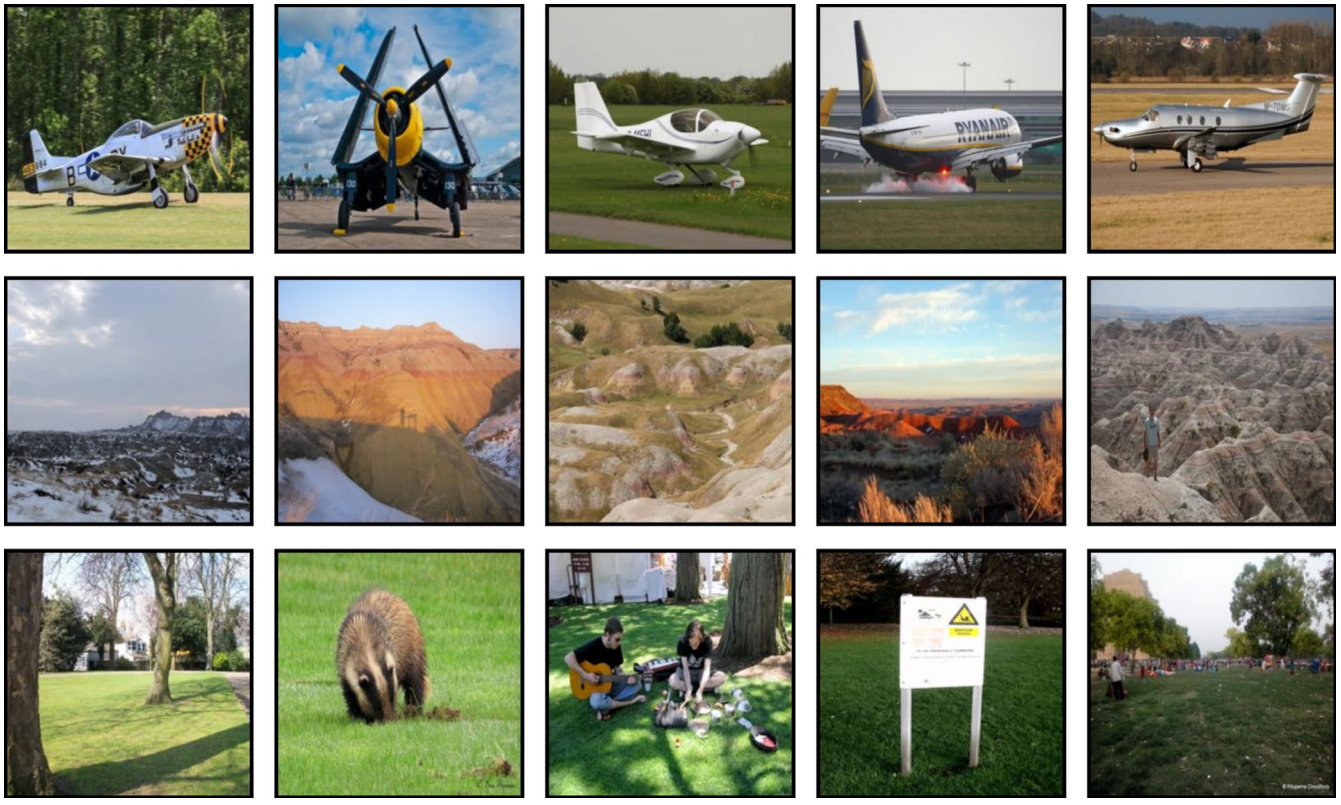

FIGURE 3 Three rows represent three learned components, separately. Each component shows five images

ORCID

Wei Wang https://orcid.org/0000-0002-6675-7474

REFERENCES

1. A. R. Bahrehdar and R. S. Purves, *Description and characterization of place properties using topic modeling on georeferenced tags*, Geo-Spatial Inf. Sci. **21** (2018), 173–184.
2. Z. Jiang et al., *Variational deep embedding: An unsupervised generative approach to clustering*, in Proc. IJCAI Int. Joint Conf. Artif. Intell. (Melbourne, Australia), 2017, pp. 1965–1972.
3. M. Caron et al., *Deep clustering for unsupervised learning of visual features*, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in, Bioinformatics) 2018.
4. V. Melnykov and R. Maitra, *Finite mixture models and model-based clustering*, Stat. Surv. **4** (2010), 80–116.
5. L. Qiu, F. Fang, and S. Yuan, *Improved density peak clustering-based adaptive Gaussian mixture model for damage monitoring in aircraft structures under time-varying conditions*, Mech. Syst. Signal Process. **126** (2019), 281–304.
6. G. J. McLachlan, S. X. Lee, and S. I. Rathnayake, *Finite Mixture Models*, Annu. Rev. Stat. Its Appl. **6** (2019), 355–378.
7. C. M. Bishop, *Pattern recognition and machine learning*, Springer, 2006.
8. A. K. Jain, R. P. W. Duin, and J. Mao, *Statistical pattern recognition: A review*, IEEE Trans. Pattern Anal. Mach. Intell. **22** (2000), 4–37.
9. A. R. Webb, *Statistical Pattern Recognition*, Wiley, England, vol. 2002.
10. D. Reynolds, *Gaussian mixture models*, S. Z. Li, A. Jain (eds) Encyclopedia of Biometrics, Boston, MA, 2009.
11. D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, *Speaker verification using adapted gaussian mixture models*, Digit. Signal Process. **10** (2000), 19–41.
12. J. P. Vila and P. Schniter, *Expectation-maximization gaussian-mixture approximate message passing*, IEEE Trans. Signal Process. **61** (2013), 4658–4672.
13. I. C. McDowell et al., *Clustering gene expression time series data using an infinite Gaussian process mixture model*, PLoS Comput. Biol. **14** (2018), e1005896.
14. X. Zhu and D. R. Hunter, *Clustering via finite nonparametric ICA mixture models*, Adv. Data Anal. Classif. **13** (2019), 65–87.
15. C. E. Rasmussen, *The infinite Gaussian mixture model*, Adv. Neural Inf. Process. Syst. **12** (2000), 554–560.
16. N. Bouguila, D. Ziou, *A dirichlet process mixture of generalized dirichlet distributions for proportional data modeling*, IEEE Trans. Neural Networks **21** (2010), 107–122.
17. D. M. Blei and M. I. Jordan, *Variational inference for Dirichlet process mixtures*, Bayesian Anal. **1** (2006), 121–144.
18. Y. Yu, M. Li, and Y. Fu, *Forest type identification by random forest classification combined with SPOT and multitemporal SAR data*, J. For. Res. **29** (2018), 1407–1414.
19. H. M. Ebied, K. Revett, and M. F. Tolba, *Evaluation of unsupervised feature extraction neural networks for face recognition*, Neural Comput. Appl. **22** (2013), 1211–1222.
20. T. Wiatowski and H. Bolcskei, *A mathematical theory of deep convolutional neural networks for feature extraction*, IEEE Trans. Inf. Theory **64** (2018), 1845–1866.
21. A. Dosovitskiy et al., *Discriminative unsupervised feature learning with exemplar convolutional neural networks*, IEEE Trans. Pattern Anal. Mach. Intell. **38** (2016), 1734–1747.
22. W. Zhang et al., *Collaborative and adversarial network for unsupervised domain adaptation*, in Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn. (Salt Lake City, UT, USA), 2018, pp. 3801–3809.
23. A. Pirbonyeh et al., *A linear unsupervised transfer learning by preservation of cluster-and-neighborhood data organization*, Pattern Anal. Appl. **22** (2019), 1149–1160.
24. S. Nejatian et al., *An innovative linear unsupervised space adjustment by keeping low-level spatial data structure*, Knowl. Inf. Syst. **59** (2019), 437–464.
25. Y. W. The et al., *Hierarchical Dirichlet processes*, J. Am. Stat. Assoc. **101** (2006), 1566–1581.
26. M. C. Hughes and E. B. Sudderth, *Memoized online variational inference for Dirichlet process mixture models*, Adv. Neural Inf. Process. Syst. **26** (2013), 2013.
27. K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, arXiv e-prints, arXiv: 1409.1556, 2014.
28. D. Bartholomew, M. Knott, and I. Moustaki, *Latent variable models and factor analysis: A unified approach (3rd ed.)*, Wiley, 2011.
29. P. D. McNicholas, and T. B. Murphy, *Parsimonious Gaussian mixture models*, Stat. Comput. **18** (2008), 285–296.
30. B. Zhou et al., *Places: A 10 million image database for scene recognition*, IEEE Trans. Pattern Anal. Mach. Intell. **40** (2018), 1452–1464.
31. N. Dalal and B. Triggs, *Histograms of oriented gradients for human detection*, in Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn. (San Diego, CA, USA), 2005, pp. 1–8.
32. T. Ojala, M. Pietikäinen, and T. Mäenpää, *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns*, IEEE Trans. Pattern Anal. Mach. Intell. **24** (2002), 971–987.
33. A. Rosenberg and J. Hirschberg, *V-measure: A conditional entropy-based external cluster evaluation measure*, in Proc. Conf. Empir. Methods Nat. Lang. Process. Comput. Nat. Lang. Learn. (Prague, Czech Republic), 2007, pp. 410–420.
34. N. X. Vinh, J. Epps, and J. Bailey, *Information theoretic measures for clusterings comparison: Is a correction for chance necessary?*, in Proc. Annu. Int. Conf. Mach. Learn. (Montreal, Canada), 2009, pp. 1–8.
35. N. X. Vinh, J. Epps, and J. Bailey, *Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance*, J. Machine Learn. Res. **11** (2010), 2837–2854.
36. J. Deng et al., *ImageNet: A large-scale hierarchical image database*, in Proc. IEEE Conf. Comput. Vision Pattern Recogn. (Miami, FL, USA), 2009, pp.

AUTHOR BIOGRAPHIES

Ri-Gui Zhou received his BS degree from the Shandong University, China, in 1977, his MS degree from the department of Computer Science and Technology of Nanchang Hangkong University, China, in 2003, and his PhD degree from the department of Computer Science and Technology of Nanjing University of Aeronautics and Astronauts, China, in 2007. From 2008 to 2010, he was a Postdoctoral Fellow in the Tsinghua University, China. From 2010 to 2011, he was a Postdoctor in Carleton University, Ottawa, Canada. From 2014 to 2015, he was a Visiting Scholar in North Carolina State University, Raleigh, NC, USA. He is currently a professor with the College of Information Engineering, Shanghai Maritime University, China. His main research interests include quantum image processing, quantum reversible logic and quantum genetic algorithm, et al. He is a senior member of China Computer Federation, and the recipient of the New Century Excellent Talents program, Ministry of Education of China in 2013.

Wei Wang received his BS degree from the Anqing Normal University, China, in 2017. He is currently pursuing a MS degree in computer application technology at Shanghai Maritime University, China. His research interests include machine learning, nonparametric Bayesian and probabilistic graphical model.