This is the Pre-Published Version.
The following publication Y. Jiang and H. F. Frank Leung, "Gaussian Mixture Model and Gaussian Supervector for Image Classification," 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), Shanghai, China, 2018, pp. 1-5 is available at https://dx.doi.org/10.1109/ICDSP.2018.8631558

Gaussian Mixture Model and Gaussian Supervector for Image Classification

Yuechi Jiang
Electronic and Information Engineering
The Hong Kong Polytechnic University
Hong Kong, China
yuechi.jiang@connect.polyu.hk

Frank H. F. Leung
Electronic and Information Engineering
The Hong Kong Polytechnic University
Hong Kong, China
frank-h-f.leung@polyu.edu.hk

***Abstract*—Gaussian Mixture Model (GMM) has been widely used in speech signal and image signal classification tasks. It can be directly used as a classifier, or used as the representation of speech or image signals. Another important usage of GMM is to serve as the Universal Background Model (UBM) to generate speech representations such as Gaussian Supervector (GSV) and i-vector. In this paper, we borrow GSV from speech signal classification studies and apply it as an image representation for image classification. GSV is calculated based on a Universal Background Model (UBM). Apart from employing the conventional GMM as the UBM to calculate GSV, we also propose the Equal-Variance GMM (EV-GMM), where all the variables in all the Gaussian mixture components share the same variance. Moreover, we derive the kernel version of EV-GMM, which generalizes EV-GMM by introducing a kernel. We then compare GSV to the raw image feature and other popular image representations such as Sparse Representation (SR) and Collaborative Representation (CR). Experiments are carried out on a handwritten digit recognition task, and classification results indicate that GSV can work very well and can be even better than other popular image representations. In addition, as the UBM, the proposed EV-GMM can work better than the conventional GMM.**

Keywords—Gaussian mixture model; equal-variance Gaussian mixture model; Gaussian supervector; image classification

I. INTRODUCTION

Gaussian Mixture Model (GMM) is a widely-used model for signal classification, especially for speech and image signals. It can be used as a speech representation for speaker recognition [1]-[3] and language recognition [4][5]. It can also be used to generate multiple features from a speech sample for recording mobile phone recognition [6]. Besides speech-related signal classification, GMM has also been used to generate features for image-based signal classification, such as using Gaussian maps [7] or GMM-based vocabulary of visual words [8] as image representations. GMM can also be used to represent human faces in a video for face recognition [9].

Another usage of GMM is to serve as the Universal Background Model (UBM). GMM-based UBM can be used to calculate Gaussian Supervector (GSV) [10] and i-vector [11], which are two popular speech representations for speaker recognition. Actually, besides of speaker recognition, GSV has also been used for recording device recognition [12]-[14].

GMM can also be used as a classifier for signal classification, such as phoneme classification [15], microphone classification [16], room classification [17], acoustic scene classification [18], acoustic event classification [19], image texture classification [20]-[22], human identification [23], and hyperspectral image classification [24].

In this paper, we employ GSV as an image representation for image classification. The performance of GSV is compared to other image representations, such as Sparse Representation (SR) [25]-[29] and Collaborative Representation (CR) [30]-[33]. Experimental results show that GSV can perform as well as or even better than SR and CR. GSV is calculated based on a GMM. In this paper, we propose the Equal-Variance GMM (EV-GMM), where all the variables in all the Gaussian mixture components share the same variance. We then compare the performance of the conventional GMM and the proposed EV-GMM as the UBM to calculate GSV. Experimental results show that EV-GMM can work better. Furthermore, we also derive the kernel version of EV-GMM, which generalizes EV-GMM by introducing a kernel.

This paper is organized as follows. In Section II, we give the formulation for the construction of the conventional GMM as well as EV-GMM. In Section III, we give the formulation of GSV. In Section IV, we extend EV-GMM to its kernel version. In Section V, experimental results are presented and discussed. A conclusion is drawn in Section VI.

II. GMM AND EV-GMM

A. Conventional GMM

The parameters of an M-component GMM can be denoted as $\theta_M = \{\omega_m, \mu_m, \sigma_m \mid m = 1,2...M\}$, where ω_m, μ_m and σ_m are the weight, mean vector and standard deviation vector of the m-th Gaussian component. The parameters can be estimated using the Expectation-Maximization (EM) algorithm. Given N training feature vectors denoted as $\{x_1, x_2 \ldots x_N\}$, the EM algorithm is described as follows [34].

In the E-step, we calculate the posterior probability $\Pr(m \mid x_n, \theta_M)$ for each mixture component m using (1), where $p(x_n \mid \mu_m, \sigma_m)$ is the Gaussian probability.

The work described in this paper was substantially supported by a grant from The Hong Kong Polytechnic University (Project Account Code: RUG7).

$$\Pr(m \mid x_n, \theta_M) = \frac{\omega_m p(x_n \mid \mu_m, \sigma_m)}{\sum_{k=1}^{M} \omega_k p(x_n \mid \mu_k, \sigma_k)} \quad (1)$$

In the M-step, we re-estimate the parameters ω_m, μ_m and σ_m for each mixture component m using (2) ~ (4), where $(\sigma_m)_i$ and $(x_n-\mu_m)_i$ denote the i-th element of σ_m and $(x_n-\mu_m)$ respectively.

$$\omega_m = \frac{1}{N} \sum_{n=1}^{N} \Pr(m \mid x_n, \theta_M) \quad (2)$$

$$\mu_m = \frac{\sum_{n=1}^{N} \Pr(m \mid x_n, \theta_M) x_n}{\sum_{n=1}^{N} \Pr(m \mid x_n, \theta_M)} \quad (3)$$

$$((\sigma_m)_i)^2 = \frac{\sum_{n=1}^{N} \Pr(m \mid x_n, \theta_M)((x_n - \mu_m)_i)^2}{\sum_{n=1}^{N} \Pr(m \mid x_n, \theta_M)} \quad (4)$$

To obtain an M-component GMM where M is assumed to be a power of 2, we adopt the mixture splitting technique [35] to increase the number of mixture components step by step. On using this technique, we start from a single Gaussian model (i.e. 1-component GMM), and in each step, we double the number of mixture components in the current GMM. For example, suppose we already have a k-component GMM $\theta_k = \{\omega_m, \mu_m, \sigma_m \mid m = 1,2...k\}$, then in the next step, we initialize the parameters of the $2k$-component GMM to be $\theta_{2k} = \{\frac{\omega_m}{2}, \mu_m - 0.2\sigma_m, \sigma_m \mid m = 1,2...k\} \cup \{\frac{\omega_m}{2}, \mu_m + 0.2\sigma_m, \sigma_m \mid m = 1,2...k\}$ and then re-estimate the parameters using EM algorithm.

B. Equal-Variance GMM (EV-GMM)

For Equal-Variance GMM (EV-GMM), we assume that the variables in all the Gaussian mixture components share the same standard deviation σ. In this condition, the parameters are denoted as $\theta_M = \{\omega_m, \mu_m, \sigma \mid m = 1,2...M\}$ where σ is now a scalar. Then in the E-step, we calculate the posterior probability using (5), where the Gaussian probability is given by (6). In the M-step, we re-estimate ω_m and μ_m using (2) and (3), but do not re-estimate σ. Parameter σ will be pre-defined.

$$\Pr(m \mid x_n, \theta_M) = \frac{\omega_m p(x_n \mid \mu_m, \sigma)}{\sum_{k=1}^{M} \omega_k p(x_n \mid \mu_k, \sigma)} \quad (5)$$

where

$$\begin{aligned} & p(x_n \mid \mu_m, \sigma) \\ &= \frac{1}{\sqrt{(2\pi)^D \mid \sigma^2 I \mid}} \exp\left\{-\frac{1}{2}(x_n - \mu_m)^T (\sigma^2 I)^{-1} (x_n - \mu_m)\right\} \quad (6) \\ &= \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp\left\{-\frac{1}{2\sigma^2}(x_n - \mu_m)^T (x_n - \mu_m)\right\} \end{aligned}$$

To obtain an M-component EV-GMM, we use a similar mixture splitting technique. Suppose we already have a k-component EV-GMM $\theta_k = \{\omega_m, \mu_m, \sigma \mid m = 1,2...k\}$, we initialize the parameters of the $2k$-component EV-GMM to be $\theta_{2k} = \{\frac{\omega_m}{2}, 0.8\mu_m, \sigma \mid m = 1,2...k\} \cup \{\frac{\omega_m}{2}, 1.2\mu_m, \sigma \mid m = 1,2...k\}$ and then re-estimate the parameters using EM algorithm. For EV-GMM, parameter σ is a pre-defined value. It is different from parameter σ_m of the conventional GMM, which is calculated based on the training data. Thus, parameter σ is not instructive for the mixture splitting process and is therefore excluded in the initialization strategy.

III. GSV

GSV is calculated based on a UBM. Given an M-component GMM or EV-GMM as the UBM, for a feature vector x_t, its corresponding GSV y_t is calculated as follows.

For each Gaussian component m, we calculate the posterior probability using (1) or (5), and then calculate the adapted mean vector μ'_m using (7), where γ is the relevance factor [34]. Then y_t is the concatenation of μ'_m as given by (8) [10][13].

$$\mu'_m = \frac{\Pr(m \mid x_t, \theta_M)}{\Pr(m \mid x_t, \theta_M) + \gamma} x_t + \frac{\gamma}{\Pr(m \mid x_t, \theta_M) + \gamma} \mu_m \quad (7)$$

$$y_t = \begin{bmatrix} \mu'_1 \\ \mu'_2 \\ \vdots \\ \mu'_M \end{bmatrix} \quad (8)$$

IV. Kernel Version of EV-GMM

An interesting characteristic of (6) is that, only the inner product of two vectors are involved, which enables the usage of a kernel for EV-GMM. When using a kernel function, an implicit feature mapping can be involved in the computation. For example, suppose a kernel function $k(.,.)$ is defined to be $k(a,b)=\varphi(a)^T\varphi(b)$, where φ is a feature mapping, then as long as this kernel function is valid, we do not need to know what φ is. Following this idea, suppose we would like to map each vector x_n to $\varphi(x_n)$ using the mapping function φ, then the EM algorithm shall be modified as follows.

In the E-step, we use (9) to calculate the posterior probability $\Pr(m \mid \varphi(x_n), \theta_M)$, where the Gaussian probability $p_m(\varphi(x_n) \mid \sigma)$ is given in (11).

$$\Pr(m \mid \varphi(x_n), \theta_M) = \frac{\omega_m p_m(\varphi(x_n) \mid \sigma)}{\sum_{k=1}^{M} \omega_k p_k(\varphi(x_n) \mid \sigma)} \quad (9)$$

In the M-step, we calculate the weight using (10) and the Gaussian probability for the m-th mixture component using (11), where $k(a,b)=\varphi(a)^T\varphi(b)$ is a kernel function.

$$\omega_m = \frac{1}{N} \sum_{n=1}^{N} \Pr(m \mid \varphi(x_n), \theta_M) \quad (10)$$

$$
\begin{aligned}
& p_m(\varphi(x_n)\,|\,\sigma) \\
&= \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp\left\{-\frac{1}{2\sigma^2}\left(\varphi(x_n)-\varphi(\mu_m)\right)^T\left(\varphi(x_n)-\varphi(\mu_m)\right)\right\} = \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp\left\{-\frac{1}{2\sigma^2}\left(\varphi(x_n)^T\varphi(x_n)-2\varphi(\mu_m)^T\varphi(x_n)+\varphi(\mu_m)^T\varphi(\mu_m)\right)\right\} \\
&= \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp\left\{-\frac{1}{2\sigma^2}\left(\varphi(x_n)^T\varphi(x_n)-2\frac{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\varphi(x_i)^T}{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)}\varphi(x_n)+\frac{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\varphi(x_i)^T}{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)}\frac{\sum_{j=1}^{N}\Pr(m\,|\,\varphi(x_j),\theta_m)\varphi(x_j)}{\sum_{j=1}^{N}\Pr(m\,|\,\varphi(x_j),\theta_m)}\right)\right\} \\
&= \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp\left\{-\frac{1}{2\sigma^2}\left(\varphi(x_n)^T\varphi(x_n)-2\frac{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\varphi(x_i)^T\varphi(x_n)}{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)}+\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\Pr(m\,|\,\varphi(x_j),\theta_m)\varphi(x_i)^T\varphi(x_j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\Pr(m\,|\,\varphi(x_j),\theta_m)}\right)\right\} \\
&= \frac{1}{\sqrt{(2\pi\sigma^2)^D}} \exp\left\{-\frac{1}{2\sigma^2}\left(k(x_n,x_n)-2\frac{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)k(x_i,x_n)}{\sum_{i=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)}+\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\Pr(m\,|\,\varphi(x_j),\theta_m)k(x_i,x_j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\Pr(m\,|\,\varphi(x_i),\theta_m)\Pr(m\,|\,\varphi(x_j),\theta_m)}\right)\right\}
\end{aligned}
\tag{11}
$$

As can be seen from (11), the calculation of the Gaussian probability only requires the calculation of the posterior probability and the kernel function, meaning that the explicit feature mapping function φ is unnecessary to be known, as long as the kernel function $k(.,.)$ is valid.

V. Experimental Results and Discussions

In this part, we compare GSV to the raw image feature, Sparse Representation (SR) and Collaborative Representation (CR). We also compare the performance of GMM and EV-GMM as the UBM to calculate GSV.

A. Experimental Setting and Image Dataset

Suppose an image is of size $P\times Q$, we concatenate all the Q columns to form a long column vector, whose dimensionality is then $PQ\times 1$. This long column vector is the raw image feature. The raw image feature is then used as the feature vector to calculate UBM, GSV, SR and CR. On using GSV as the image representation, the classifier is SVM, which is implemented using LIBSVM [36]. The UBM is calculated using the raw image features used for training. For a feature vector x_t, its corresponding SR y_t is obtained by solving the optimization problem $y_t = \arg\min_y \|y\|_1$ subject to $x_t = Ay$, where A is a dictionary [26]. The solution y_t is obtained using SparseLab [37]. The classifier for SR is the Sparse Representation-based Classifier (SRC) [26]. For a feature vector x_t, its corresponding CR is obtained by solving the optimization problem $y_t = \arg\min_y \|x_t - Ay\|_2^2 + \lambda\|y\|_2^2$, whose solution is given by $y_t = \left(A^TA+\lambda I\right)^{-1}A^Tx_t$ [30]. λ is the regularization parameter and is set to be 0.01 in our experiment. The classifier for CR is the Collaborative Representation-based Classifier (CRC) [30]. For SR and CR, the dictionary A is a matrix, whose column vectors are the raw image features used for training.

The image dataset is USPS handwritten digit dataset containing 10 handwritten digits (i.e. digit 0 ~ 9) [25][38]. For each digit, 500 samples are used for training and 600 samples are used for testing. Totally 5000 images are used for training and 6000 images are used for testing. The training images are also used to build the UBM. Each image is of size 16×16, thus the dimensionality of the raw image feature is 256×1.

B. Experimental Results

On using GSV, different values of γ are evaluated. As the UBM, GMM and EV-GMM with different numbers of Gaussian components (i.e. M) are investigated. When utilizing EV-GMM to calculate GSV, different values of σ are evaluated. Classification results are illustrated in Figs. 1 ~ 5. Fig. 1 shows the results of using GSV calculated from the conventional GMM, while Figs. 2 ~ 5 show the results of using GSV calculated from EV-GMM. In the figures, "raw+SVM" means that the image representation is the 256-dimension raw image feature and the classifier is SVM; "GSV+SVM" means that the image representation is GSV and the classifier is SVM; "SR+SRC" means that the image representation is SR and the classifier is SRC; "CR+CRC" means that the image representation is CR and the classifier is CRC.

From Fig. 1, when the conventional GMM is used, the performance of GSV is similar to other image representations. The number of mixture components M in the GMM highly influences the quality of GSV, and the quality of GSV tends to degrade with the increase of M.

From Figs. 2 ~ 5, when EV-GMM is used, GSV can outperform other image representations when a suitable relevance factor γ is used. When γ is small (e.g. γ=0.0001), the

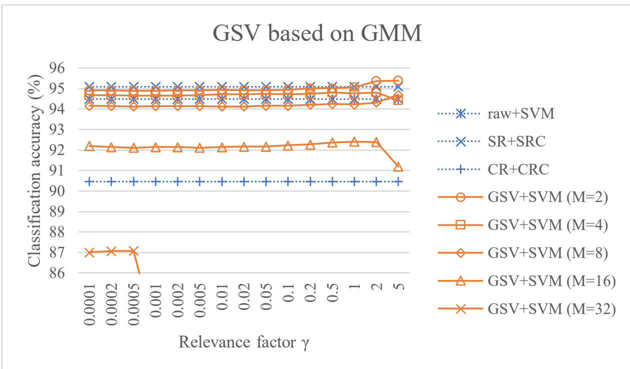

Fig. 1. GSV calculated from conventional GMM.

larger the M, the better the performance of GSV; when γ is large (e.g. γ=1), the smaller the M, the better the performance of GSV. In addition, when M is small (e.g. M=4), the performance of GSV is more stable with respect to different values of γ; when M is large (e.g. M=16), the performance of GSV drops rapidly with the increase of the value of γ. Moreover, comparing Figs. 2 ~ 5, the larger the σ, the less the GSV will be influenced by M.

C. Discussions

In summary, the performance of GSV is influenced by the relevance factor as well as the UBM. As can be seen from (7), the relevance factor γ controls the proportion of information that GSV absorbs from the raw image feature and the UBM. The larger the γ, the more information GSV absorbs from the UBM. With suitable relevance factors, GSV can work very well and even better than other popular image representations. The number of mixture components M in the UBM also influences the performance of GSV, but increasing M does not guarantee performance improvement. In particular, on using EV-GMM as the UBM, σ also plays an important role. As the UBM, experimental results show that the proposed EV-GMM can outperform the conventional GMM. The reason lies in the fact that, the image features do not follow the multivariate Gaussian distribution very well, causing the variance of the conventional GMM to be inaccurately estimated. On the contrary, for EV-GMM, the standard deviation parameter is pre-defined and shared by all the variables in all the Gaussian mixture components, making EV-GMM less affected by the inaccuracy in parameter estimation.

VI. Conclusion

In this paper, we utilize GSV as the image representation for handwritten digit recognition, and compare GSV with other popular image representations used in image classification tasks, such as Sparse Representation (SR) and Collaborative Representation (CR). Experimental results show that GSV can work as well as other image representations, and with suitable chosen parameters, GSV can work even better.

GSV is calculated based on a Universal Background Model (UBM), and this UBM is usually a Gaussian Mixture Model (GMM). To better model the characteristics of images, we propose the Equal-Variance GMM (EV-GMM), where all the variables in all the Gaussian mixture components share the same standard deviation. We compare the performance of EV-GMM and the conventional GMM in terms of the performance of GSV, and experimental results indicate that EV-GMM can work better, when the standard deviation parameter is properly chosen. The improved performance of EV-GMM over GMM lies in the fact that, image features do not follow the multivariate Gaussian distribution very well. By pre-defining the standard deviation instead of estimating the standard deviation from data, EV-GMM is less affected by the inaccurate estimation of the parameters than GMM. Moreover, when using EV-GMM, we can have a kernel version, where an implicit feature mapping can be involved. The kernel version EV-GMM is also the generalization of EV-GMM, but the usage of the kernel version EV-GMM needs more exploration in the future.

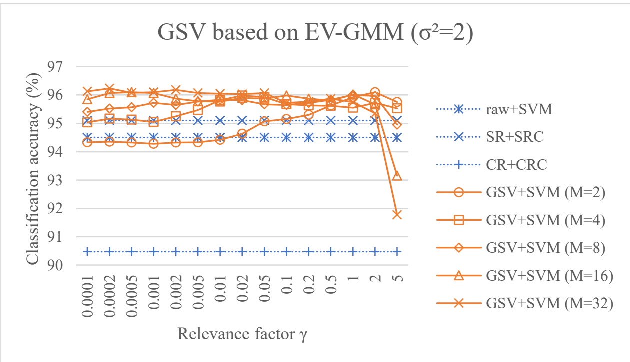

Fig. 2. GSV calculated from EV-GMM with σ^2=2.

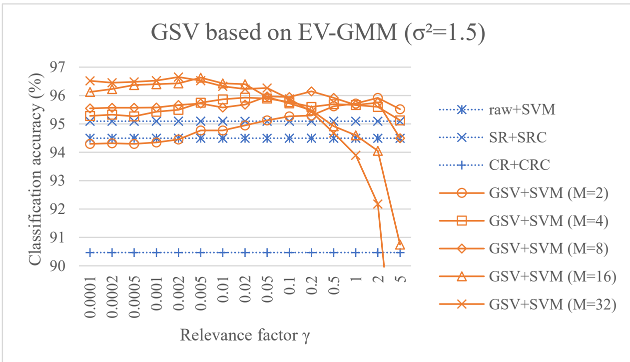

Fig. 3. GSV calculated from EV-GMM with σ^2=1.5.

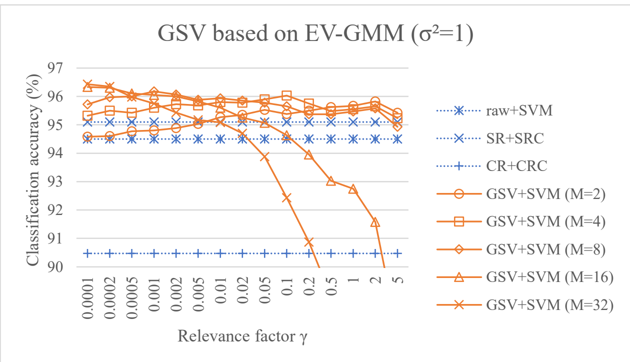

Fig. 4. GSV calculated from EV-GMM with σ^2=1.

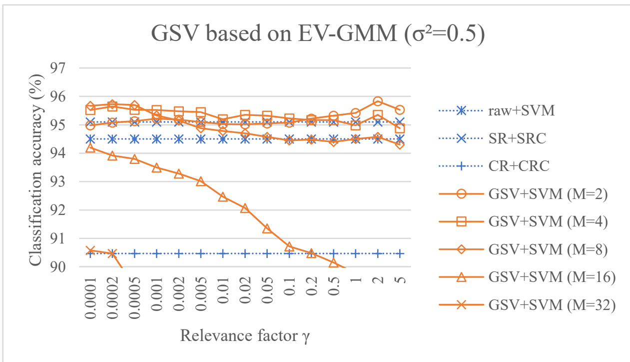

Fig. 5. GSV calculated from EV-GMM with σ^2=0.5.

REFERENCES

[1] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. on Speech and Audio Processing*, vol. 3, no. 1, pp. 72-83, 1995.

[2] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*, vol. 17, no. 1, pp. 91-108, 1995.

[3] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, pp. 19-41, 2000.

[4] P. A. T. Carrasquillo, D. A. Reynolds, and J. R. Deller, "Language identification using Gaussian mixture model tokenization," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2002, pp. 757-760.

[5] P. A. T. Carrasquillo *et al.*, "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *Proc. Interspeech*, 2002.

[6] Y. Jiang and F. H. F. Leung, "Mobile phone identification from speech recordings using weighted support vector machine," in *Proc. IEEE 42nd Annual Conf. on Industrial Electronics (IECON)*, 2016, pp. 963-968.

[7] X. Zhou, N. Cui, Z. Li, F. Liang, and T. S. Huang, "Hierarchical Gaussianization for image classification," in *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, 2009, pp. 1971-1977.

[8] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1-8.

[9] H. Bredin, N. Dehak, and G. Chollet, "GMM-based SVM for face recognition," in *Proc. IEEE Int. Conf. on Pattern Recognition (ICPR)*, 2006, pp. 1111-1114.

[10] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308-311, 2006.

[11] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788-798, 2011.

[12] D. G. Romero and C. Y. E. Wilson, "Automatic acquisition device identification from speech recordings," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2010, pp. 1806-1809.

[13] Y. Jiang and F. H. F. Leung, "Using regularized Fisher discriminant analysis to improve the performance of Gaussian supervector in session and device identification," in *Proc. IEEE Int. Joint Conf. on Neural Networks (IJCNN)*, 2017, pp. 705-712.

[14] C. L. Kotropoulos, "Source phone identification using sketches of features," *IET Biometric*, vol. 3, no. 2, pp. 75-83, 2014.

[15] F. Sha and L. K. Saul, "Large margin Gaussian mixture modeling for phonetic classification and recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2006, pp. 265-268.

[16] O. Eskidere, "Source microphone identification from speech recordings based on a Gaussian mixture model," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 22, pp. 754-767, 2014.

[17] N. Peters, H. Lei, and G. Friedland, "Name that room: room identification using acoustic features in a recording," in *Proc. 20th ACM Int. Conf. on Multimedia*, 2012, pp. 841-844.

[18] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 16-34, 2015.

[19] D. Stowell, D. Giannoulis, E. Benetos, M. Lagrange, and M. D. Plumbley, "Detection and classification of acoustic scenes and events," *IEEE Trans. on Multimedia*, vol. 17, no. 10, pp. 1733-1746, 2015.

[20] H. Permuter, J. Francos, and I. H. Jermyn, "Gaussian mixture models of texture and colour for image database retrieval," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003, pp. 569-572.

[21] H. Permuter, J. Francos, and I. Jermyn, "A study of Gaussian mixture models of color and texture features for image classification and segmentation," *Pattern Recognition*, vol. 39, no. 4, pp. 695-706, 2006.

[22] S. C. Kim and T. J. Kang, "Texture classification and segmentation using wavelet packet frame and Gaussian mixture model," *Pattern Recognition*, vol. 40, no. 4, pp. 1207-1221, 2007.

[23] S. Mitra and M. Savvides, "Gaussian mixture models based on the frequency spectra for human identification and illumination classification," in *Proc. IEEE Workshop on Automatic Identification Advanced Technologies*, 2005, pp. 245-250.

[24] W. Li, S. Prasad, and J. E. Fowler, "Hyperspectral image classification using Gaussian mixture models and Markov random fields," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 1, pp. 153-157, 2014.

[25] K. Huang and S. Aviyente, "Sparse representation for signal classification," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2007, pp. 609-616.

[26] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210-227, 2009.

[27] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 625-632.

[28] J. Yang, D. Chu, L. Zhang, Y. Xu, and J. Yang, "Sparse representation classifier steered discriminative projection with applications to face recognition," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 24, no. 7, pp. 1023-1035, 2013.

[29] H. Zhang, Y. Zhang, and T. S. Huang, "Pose-robust face recognition via sparse representation," *Pattern Recognition*, vol. 46, no. 5, pp. 1511-1521, 2013.

[30] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: which helps face recognition," in *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, 2011, pp. 471-478.

[31] R. Timofte and L. V. Cool, "Weighted collaborative representation and classification of images," in *Proc. IEEE Int. Conf. on Pattern Recognition (ICPR)*, 2012, pp. 1606-1610.

[32] M. Yang, L. Zhang, D. Zhang, and S. Wang, "Relaxed collaborative representation for pattern classification," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2224-2231.

[33] W. Deng, J. Hu, and J. Guo, "Face recognition via collaborative representation: its discriminant nature and superposed representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2017.

[34] D. Reynolds, "Gaussian mixture models," in *Encyclopedia of Biometrics*, 2015, pp. 827-832.

[35] S. Young *et al.*, *The HTK book (v3.4),* Cambridge: Cambridge University Press, 2006, pp. 156-157.

[36] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1-27, 2011.

[37] D. Donoho, V. Stodden, and Y. Tsaig, *About SparseLab(v2.1)*, Stanford, 2007.

[38] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550-554, 1994.