Comparison of Supervector and Majority Voting in Acoustic Scene Identification

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—**Acoustic scene identification aims to identify the acoustic environment from the acoustic signal. Usually one first divides a piece of acoustic signal into multiple short-time frames and then calculates frame-level features. A natural question is then how to make use of these frame-level features for identification purposes. In this paper, we compare two feature aggregation methods. One method is Majority Voting (MV), which treats each frame-level feature as an independent feature vector and then perform identification using majority voting strategies. In this way, an acoustic signal is represented by multiple feature vectors. The other method is Supervector, which maps the frame-level features to a single feature vector. In this way, an acoustic signal is represented by one feature vector. Particularly, we consider three types of Supervector, which are Gaussian Supervector, Factor Analysis Supervector, and i-vector. We then compare Supervector with MV in an acoustic identification task. Different classifiers are employed, including Gaussian Mixture Model (GMM), Support Vector Machine (SVM), Multilayer Perceptron (MLP), and Deep Neural Network (DNN). Experimental results indicate that these two feature aggregation methods give very similar performances, nonetheless, each has its own advantages and disadvantages.**

Keywords—acoustic scene identification; majority voting; Gaussian supervector; factor analysis supervector; i-vector

I. Introduction

Acoustic scene identification aims at identifying the acoustic environment (e.g. supermarket, park, train, etc.) based on the information extracted from the acoustic signal. Its applications include audio authentication [1] and context-aware services such as robot navigation [2]. The length of a piece of acoustic signal is usually not fixed, thus to generate fixed-dimension features for classification, we first divide the acoustic signal into equal-length frames and then extract frame-level features [2]. These frame-level features can be averaged to form a single feature vector for classification, or directly used together with a majority-voting classification scheme [3]. Among different frame-level features, the most widely used one is the Mel-frequency Cepstral Coefficient (MFCC), which is also used as the baseline feature in Detection and Classification of Acoustic Scenes and Events (DCASE) [4]. Some studies also extract temporal features as an auxiliary, such as the Recurrence Quantification Analysis (RQA) features [5], the Local Discriminant Base (LDB) features [6], and the Local Binary Pattern (LBP) features [7]. However, acoustic scenes may not possess strong temporal characteristics [8][9].

Given the frame-level features, apart from directly averaging them, it is also feasible to average the basis vectors obtained from them. The basis vectors can be obtained using Matching Pursuit (MP) [10] or Nonnegative Matrix Factorization (NMF) [11]. However, this averaging process does not make full use of all the frame-level features. Majority Voting (MV) can work better than simple averaging [12]. It is also possible to map the frame-level features to an i-vector [4], which is a type of Supervector.

In this paper, we explore the usage of Supervector in acoustic scene identification, which has not been well explored in the literature. In particular, we consider three types of Supervector prevailing in speaker recognition studies, which are Gaussian Supervector (GSV), Factor Analysis Supervector (FASV), and i-vector [13]. We then compare the performance of different types of Supervector with different MV schemes. Different classifiers are employed, including Gaussian Mixture Model (GMM) [14], Support Vector Machine (SVM) [15], Multilayer Perceptron (MLP), and Deep Neural Network (DNN). An overview of the difference between Supervector and MV is illustrated in Fig. 1.

This paper is organized as follows. In Section II, we give the formulation of different types of Supervector. In Section III, we formulate different MV schemes. In Section IV, we briefly describe the acoustic scene dataset. In Section V, we show the experimental results together with some discussions. A conclusion will be drawn in Section VI.

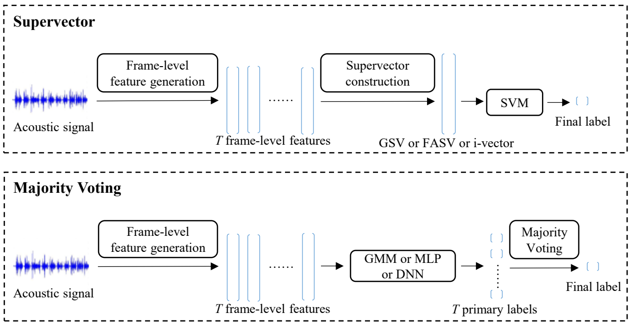

Fig. 1. Overview of Supervector method and Majority Voting method.

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

II. SUPERVECTOR

Supervector is calculated based on a Universal Background Model (UBM), which is a GMM. Given a set of acoustic signals used for UBM construction, we first calculate the frame-level features, which are MFCC vectors [16] in this paper. Then the UBM is constructed using the mixture splitting technique [17] and the Expectation-Maximization (EM) algorithm [18]. In the following, we denote the model parameters of a UBM with M Gaussian mixture components as $\theta_M = \{\pi_m, \mu_m, \sigma_m \mid m = 1,2...M\}$, where π_m, μ_m and σ_m denote the weight, mean and standard deviation of the m-th Gaussian mixture component respectively.

A. Gaussian Supervector (GSV)

Given a sequence of T frame-level features $\{x_1, x_2 \ldots x_T\}$ corresponding to signal s, and a UBM denoted as $\theta_M = \{\pi_m, \mu_m, \sigma_m \mid m = 1,2...M\}$, GSV is calculated as follows.

First, we calculate the posterior probability for the m-th mixture component using (1), where $p(x_t \mid \mu_m, \sigma_m)$ is the Gaussian probability. We then use (2) and (3) to calculate the Maximum A Posterior (MAP) estimation of the mean vector for the m-th mixture component, which is $E_m(s)$. Finally, we calculate the adapted mean vector $\mu'_m(s)$ for the m-th mixture component using (4), which is a weighted sum of $E_m(s)$ and μ_m [19]. In (4), γ is the relevance factor [18].

$$\Pr(m \mid x_t, \theta_M) = \frac{\pi_m p(x_t \mid \mu_m, \sigma_m)}{\sum_{j=1}^{M} \pi_j p(x_t \mid \mu_j, \sigma_j)} \tag{1}$$

$$n_m(s) = \sum_{t=1}^{T} \Pr(m \mid x_t, \theta_M) \tag{2}$$

$$E_m(s) = \frac{1}{n_m(s)} \sum_{t=1}^{T} \Pr(m \mid x_t, \theta_M) x_t \tag{3}$$

$$\mu'_m(s) = \frac{n_m(s)}{n_m(s)+\gamma} E_m(s) + \frac{\gamma}{n_m(s)+\gamma} \mu_m \tag{4}$$

GSV, denoted as $\mu_{GSV}(s)$, is then the concatenation of $\mu'_m(s)$ for m=1, 2…M, as given by (5) [20]. If the dimensionality of a frame-level feature x_t is D×1, and the number of mixture components in the UBM is M, then the dimensionality of GSV will be MD×1.

$$\mu_{GSV}(s) = \begin{bmatrix} \mu'_1(s) \\ \mu'_2(s) \\ \vdots \\ \mu'_M(s) \end{bmatrix} \tag{5}$$

B. Factor Analysis Supervector (FASV) and I-vector

In a Factor Analysis model, Factor Analysis Supervector (FASV), denoted as $\mu_{FASV}(s)$, is given by (6), where μ_{UBM} is the concatenation of μ_m for m=1, 2…M. In (6), V is the factor-loading matrix estimated using the EM algorithm, and $z(s)$ is the i-vector, which is the posterior expected mean of the latent variable $y(s)$ in a Factor Analysis model [21]. So, both FASV and i-vector originate from the Factor Analysis model, and once the model parameters are determined, both FASV and i-vector can be easily computed.

$$\mu_{FASV}(s) = \mu_{UBM} + VE[y(s)] = \mu_{UBM} + Vz(s) = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_M \end{bmatrix} + Vz(s) \tag{6}$$

Given a sequence of T frame-level features $\{x_1, x_2 \ldots x_T\}$ corresponding to signal s, a UBM denoted as $\theta_M = \{\pi_m, \mu_m, \sigma_m \mid m = 1,2...M\}$, and the factor-loading matrix V, FASV and i-vector are calculated as follows.

First, we calculate the centralized first-order Baum-Welch statistics $S_{X,m}(s)$ for the m-th mixture component using (7), and the centralized second-order Baum-Welch statistics $S_{XX,m}(s)$ using (8) [21][22].

$$S_{X,m}(s) = \sum_{t=1}^{T} \Pr(m \mid x_t, \theta_M)(x_t - \mu_m) \tag{7}$$

$$S_{XX,m}(s) = \sum_{t=1}^{T} \Pr(m \mid x_t, \theta_M)(x_t - \mu_m)(x_t - \mu_m)^T \tag{8}$$

Then, $S_{X,m}(s)$ is column-wisely concatenated to form a super vector $S_X(s)$ for m=1, 2…M, as given by (9); $S_{XX,m}(s)$ is diagonally concatenated to form a super matrix $S_{XX}(s)$ for m=1, 2…M, as given by (10); $n_m(s)$ calculated in (2) is used to form a diagonal super matrix $N(s)$ as given by (11), where I is an identity matrix [22]. If the dimensionality of a frame-level feature x_t is D×1, and the number of mixture components in the UBM is M, then the dimensionality of $S_X(s)$ will be MD×1, the dimensionality of $S_{XX}(s)$ will be MD×MD, the dimensionality of the identity matrix I in (11) will be D×D, and the dimensionality of $N(s)$ will be MD×MD.

$$S_X(s) = \begin{bmatrix} S_{X,1}(s) \\ S_{X,2}(s) \\ \vdots \\ S_{X,M}(s) \end{bmatrix} \tag{9}$$

$$S_{XX}(s) = \begin{bmatrix} S_{XX,1}(s) & 0 & \cdots & 0 \\ 0 & S_{XX,2}(s) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & S_{XX,M}(s) \end{bmatrix} \tag{10}$$

$$N(s) = \begin{bmatrix} n_1(s)I & 0 & \cdots & 0 \\ 0 & n_2(s)I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & n_M(s)I \end{bmatrix} \tag{11}$$

Then i-vector is calculated using (12), where Σ is a diagonal matrix estimated using the EM algorithm. After obtaining $z(s)$, FASV is then calculated using (6).

$$z(s)=E[y(s)]=(I+V^T\Sigma^{-1}N(s)V)^{-1}V^T\Sigma^{-1}S_X(s) \quad (12)$$

In the following, we describe how to estimate the model parameters V and Σ using the EM algorithm. On using the EM algorithm, the E-step and the M-step are repeated until convergence or the total number of EM iterations exceeds some predefined threshold.

In the E-step, we calculate the posterior estimated mean and the posterior estimated covariance of the latent variable $y(s)$ using (13) and (14) respectively [22].

$$E[y(s)]=(I+V^T\Sigma^{-1}N(s)V)^{-1}V^T\Sigma^{-1}S_X(s) \quad (13)$$

$$E[y(s)y(s)^T]=(I+V^T\Sigma^{-1}N(s)V)^{-1}+E[y(s)]E[y(s)]^T \quad (14)$$

In the M-step, we calculate V by solving a system of linear equations as given by (15), where we use S to denote the total number of training acoustic signals. After finding V, we can calculate Σ using (16), where $diag(.)$ is to diagonalize a matrix by setting all the non-diagonal elements to zero. More details can be found in [22].

$$\sum_{s=1}^{S}N(s)VE[y(s)y(s)^T]=\sum_{s=1}^{S}S_X(s)E[y(s)]^T \quad (15)$$

$$\Sigma=diag\left(\left(\sum_{s=1}^{S}N(s)\right)^{-1}\left(\sum_{s=1}^{S}S_{XX}(s)-\sum_{s=1}^{S}VE[y(s)]S_X(s)^T\right)\right) \quad (16)$$

Given the model parameters V and Σ, i-vector is calculated using (12), and FASV is calculated using (6). If the dimensionality of a frame-level feature x_t is $D\times1$, and the number of mixture components in the UBM is M, then the dimensionality of FASV will be $MD\times1$, and the dimensionality of i-vector is the number of columns of V.

III. Majority Voting

Given a sequence of T frame-level features $\{x_1, x_2 \ldots x_T\}$ corresponding to signal s, Majority Voting (MV) is carried out as follows.

Let the total number of categories be C. Given a classifier, assume a feature x_t will be classified to category $l(x_t)$ (where $l(x_t)\in\{1, 2\ldots C\}$) according to some criterion, and x_t has a probability $p(c|x_t)$ to be classified to category c (where $c\in\{1, 2\ldots C\}$). Then, there are three MV schemes that can be used to determine the category $L(s)$ (where $L(s)\in\{1, 2\ldots C\}$) that the acoustic signal s should be classified to. Scheme 1, Scheme 2 and Scheme 3 are given by (17), (18) and (19) respectively, where $g(.,.)$ is an indicator function as given by (20).

$$L(s)=\underset{c}{\operatorname{argmax}}\sum_{t=1}^{T}g(c,l(x_t)) \quad (17)$$

$$L(s)=\underset{c}{\operatorname{argmax}}\sum_{t=1}^{T}p(c\mid x_t) \quad (18)$$

$$L(s)=\underset{c}{\operatorname{argmax}}\sum_{t=1}^{T}p(c\mid x_t)g(c,l(x_t)) \quad (19)$$

where

$$g(a,b)=\begin{cases}1 & if\ a=b\\ 0 & if\ a\neq b\end{cases} \quad (20)$$

In this paper, on using different MV schemes, GMM, MLP and DNN are employed as the classifier. On using MLP and DNN, the output layer is a softmax layer, which already gives the probability $p(c|x_t)$ for each category. On using GMM, one GMM will be constructed for each category. Suppose the number of mixture components in each GMM is M, and the c-th GMM is denoted as $\theta_M^{(c)}=\{\pi_m^{(c)},\mu_m^{(c)},\sigma_m^{(c)}\mid m=1,2...M\}$, then $p(c|x_t)$ is given by (21).

$$p(c\mid x_t)=\frac{\sum_{m=1}^{M}\pi_m^{(c)}p(x_t\mid\mu_m^{(c)},\sigma_m^{(c)})}{\sum_{k=1}^{C}\sum_{m=1}^{M}\pi_m^{(k)}p(x_t\mid\mu_m^{(k)},\sigma_m^{(k)})} \quad (21)$$

Actually, (19) is a weighted majority voting strategy, whose weights are determined by $p(c|x_t)$. Instead of using classifier-dependent weights, we can also use feature-dependent weights which are determined by some pre-defined weighting functions, similar to [12]. However, manually designing a suitable weighting function is difficult.

IV. Dataset

In this paper, we conduct experiments on the TUT Acoustic Scenes 2016 development set [23]. TUT2016 development set consists of 15 different acoustic scenes, such as bus, restaurant, library, park, train, etc. Each acoustic scene involves 78 audio segments lasting for 30 seconds, so totally there are 1170 audio segments available. The development set is organized into 4 different folds, and in each fold, there are about 880 audio segments used for training and about 290 audio segments used for testing. The training data are also used to construct UBM. Different folds merely involve different combinations of the training data and the testing data, and the identification accuracy results of the 4 folds are then averaged and reported.

V. Experimental Results and Discussions

In this paper, we use 24-dimension MFCC vector as the frame-level feature, extracted using Hamming window with 50ms length and 10ms shift. Details about MFCC can be found in [16]. For i-vector, its dimensionality is set to be half that of FASV and GSV, and 20 EM iterations are executed to estimate the parameters of the factor analysis model. On using Supervector, linear SVM implemented by LIBSVM [24] is employed as the classifier. On using different MV schemes, GMM, MLP and DNN are employed as the classifier. MLP consists of one input layer, one output layer and one hidden layer, while DNN consists of one input layer, one output layer and three hidden layers. The activation function for the input and hidden layers is sigmoid function, while the output layer is a softmax layer. Both MLP and DNN are pre-trained for 50 epochs using Stacked Autoencoder (SAE) with momentum 0.5 and learning rate 1, and fine-tuned for 2000 epochs with momentum 0.5 and a decaying learning rate. Both MLP and DNN are implemented by DeepLearnToolbox [25].

The average identification accuracy results of the 4 folds are shown in Tables I ~ IV. Table I and Table II show the

TABLE I. ACCURACY USING GSV (%)

Relevance Factor	Number of Components in UBM						
	2	4	8	16	32	64	128
0	65.20	66.75	66.49	67.16	68.44	**68.71**	67.78
1	66.73	66.07	68.61	68.53	68.25	**69.13**	68.12
5	67.67	66.99	67.85	69.29	**69.37**	67.67	66.22
10	68.19	67.76	67.00	**68.51**	68.18	67.74	65.87
15	67.42	67.75	67.00	67.91	67.34	**68.25**	65.88

TABLE II. ACCURACY USING FASV AND I-VECTOR (%)

Supervector Type	Number of Components in UBM						
	2	4	8	16	32	64	128
FASV	63.01	64.17	65.56	**66.14**	65.95	65.88	65.79
i-vector	61.45	65.72	66.58	66.48	**69.30**	66.47	68.44

results of using different types of Supervector, while Table III and Table IV show the results of using different MV schemes. On using Supervector, UBMs with different numbers of mixture components are investigated. Particularly, on using GSV, different values of the relevance factor γ are evaluated. On using GMM, different numbers of mixture components are investigated. On using MLP, different numbers of neurons in the hidden layer are investigated. On using DNN, there are 96 neurons in the first hidden layer, 64 neurons in the second hidden layer, and 32 neurons in the third hidden layer.

Regarding Supervector (Tables I and II), the performance is highly dependent on the number of mixture components in the UBM, and tends to fluctuate with different numbers of mixture components. Increasing the number of mixture components does not guarantee performance improvement, although the dimensionality of different types of Supervector is increased. In addition, GSV is also affected by the choice of the relevance factor. In terms of the highest accuracy achieved, GSV and i-vector give very similar performances, while FASV performs worse than the other two. Practically, GSV is simpler and faster to be calculated than i-vector and FASV (because the EM algorithm used for parameter estimation in the factor analysis model is time consuming).

Regarding MV (Tables III and IV), on using GMM as the classifier, the performance is highly dependent on the number of mixture components in the GMM, and a larger number of mixture components does not always give a better performance. On using MLP as the classifier, the performance is slightly influenced by the number of neurons in the hidden layer; however, increasing the number of neurons in the hidden layer does not guarantee performance improvement. DNN seems to perform a little better than MLP, but worse than GMM. As indicated in [9] and [26], MLP and DNN do not work very well purely as a classifier. In addition, no matter using GMM, MLP or DNN, different MV schemes tend to give quite similar performances.

In terms of the highest accuracy achieved, Supervector tends to give a slightly better performance than MV, nonetheless, the performances are quite similar. Both feature aggregation methods have their own advantages and disadvantages. On using Supervector, the frame-level features are mapped to a single feature vector, meaning that each acoustic signal is represented by only one feature vector. This reduces the time for training and testing, but the feature vector may be contaminated and sensitive to noise. Nevertheless, since the number of feature vectors is small, extra processing techniques can be applied for possible performance improvement, for example, applying Fisher Discriminant Analysis (FDA) based projection techniques [19][27][28]. On using MV, the frame-level features are directly fed to the classifier, meaning that each acoustic signal is represented by multiple feature vectors. This increases the time for training and testing, but can be noise robust, as the decision is made by a group of feature vectors instead of a single feature vector.

VI. CONCLUSION

In this paper, we compare two feature aggregation methods for acoustic scene identification. One method is Majority Voting (MV), and the other is Supervector. Regarding MV, we propose three classification schemes, and experimental results indicate that these schemes show few differences. Regarding Supervector, we investigate three popular types of Supervector, namely Gaussian Supervector (GSV), Factor Analysis Supervector (FASV), and i-vector. Experimental results indicate that GSV and i-vector give similar performances whereas FASV performs worse than the other two. We also compare the performance of different MV schemes and different types of Supervector. Although Supervector gives a slightly better performance than MV, their performances are quite similar. Both feature aggregation methods have their advantages and disadvantages. On using MV, one acoustic signal is represented by multiple features, meaning that the computation takes more time; however, the performance is more robust to noise, as the decision is made based on a group of features but not a single feature. On using Supervector, one acoustic signal is represented by a single feature, meaning that the computation takes less time; however, the performance is more sensitive to noise, as the decision is made based on only one feature. In particular, on using Supervector, extra processing techniques can be applied to the single feature to improve its quality, whereas it is difficult to apply these techniques if there are multiple features, especially when there are too many features, which is the case of using MV.

TABLE III. ACCURACY USING GMM (%)

Majority Voting	Number of Components in GMM						
	2	4	8	16	32	64	128
Scheme 1	64.67	63.56	**68.27**	66.82	67.78	66.58	66.58
Scheme 2	64.32	63.22	**68.61**	67.33	67.95	66.84	66.40
Scheme 3	64.23	63.64	**68.78**	66.99	67.77	66.66	66.66

TABLE IV. ACCURACY USING MLP AND DNN (%)

Majority Voting	Number of Neurons in the Hidden Layer			
	MLP			DNN
	32	64	96	96-64-32
Scheme 1	60.02	**61.12**	60.43	62.57
Scheme 2	59.83	61.37	**61.54**	63.59
Scheme 3	59.24	**60.44**	**60.44**	62.05

References

[1] S. Gupta, S. Cho, and C. C. J. Kuo, “Current developments and future trends in audio authentication,” *IEEE Multimedia*, vol. 19, pp. 50-59, 2012.

[2] 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.

[3] S. Chachada and C. C. J. Kuo, “Environmental sound recognition: a survey,” in *Proc. IEEE Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2013, pp. 1-9.

[4] 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.

[5] G. Roma, W. Nogueira, and P. Herrera, “Recurrence quantification analysis features for environmental sound recognition,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013, pp. 1-4.

[6] K. Umapathy, S. Krishnan, and R. K. Rao, “Audio signal feature extraction and classification using local discriminant bases,” *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1236-1246, 2007.

[7] W. Yang and S. Krishnan, “Combining temporal features by local binary pattern for acoustic scene classification,” *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1315-1321, 2017.

[8] I. M. Morato, M. Cobos, and F. J. Ferri, “A case study on feature sensitivity for audio event classification,” in *Proc. IEEE Int. Workshop on Machine Learning for Signal Processing (MLSP)*, 2016, pp. 1-6.

[9] J. Li, W. Dai, F. Metze, S. Qu, and S. Das, “A comparison of deep learning methods for environmental sound detection,” in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 126-130.

[10] S. Chu, S. Narayanan, and C. C. J. Kuo, “Environmental sound recognition with time-frequency audio features,” *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1142-1158, 2009.

[11] V. Bisot, R. Serizel, S. Essid, and G. Richard, “Feature learning with matrix factorization applied to acoustic scene classification,” *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1216-1229, 2017.

[12] 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.

[13] J. H. L. Hansen and T. Hasan, “Speaker recognition by machines and humans: a tutorial review,” *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74-99, 2015.

[14] A. J. Eronen *et al.*, “Audio-based context recognition,” *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321-329, 2006.

[15] J. T. Geiger, B. Schuller, and G. Rigoll, “Large-scale audio feature extraction and SVM for acoustic scene classification,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013, pp. 1-4.

[16] X. Huang, A. Acero, and H. W. Hon, “Speech signal representations,” in *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Upper Saddle River, NJ: Prentice Hall PTR, 2001, ch. 6, pp. 273-333.

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

[18] D. Reynolds, “Gaussian mixture models,” in *Encyclopedia of Biometrics*, 2015, pp. 827-832.

[19] 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.

[20] 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.

[21] 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.

[22] P. Kenny, G. Boulianne, and P. Dumouchel, “Eigenvoice modeling with sparse training data,” *IEEE Trans. on Speech and Audio Processing*, vol. 13, no. 3, pp. 345-354, 2005.

[23] A. Mesaros, T. Heittola, and T. Virtanen, “TUT database for acoustic scene classification and sound event detection,” in *Proc. IEEE 24th European Signal Processing Conference (EUSIPCO)*, 2016, pp. 1128-1132.

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

[25] R. B. Palm, “Prediction as a candidate for learning deep hierarchical models of data,” *Technical University of Denmark*, 2012.

[26] M. Cowling and R. Sitte, “Comparison of techniques for environmental sound recognition,” *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2895-2907, 2003.

[27] Y. Jiang and F. H. F. Leung, “Using double regularization to improve the effectiveness and robustness of Fisher discriminant analysis as a projection technique,” in *Proc. IEEE Int. Joint. Conf. on Neural Networks (IJCNN)*, 2018, pp. 3275-3281.

[28] Y. Jiang and F. H. F. Leung, “Generalized Fisher discriminant analysis as a dimensionality reduction technique,” in *Proc. Int. Conf. on Pattern Recognition (ICPR)*, 2018.