MOBILE PHONE CLUSTERING FROM ACQUIRED SPEECH RECORDINGS USING DEEP GAUSSIAN SUPERVECTOR AND SPECTRAL CLUSTERING

Yanxiong Li, Xue Zhang, Xianku Li, Xiaohui Feng, Jichen Yang, Aiwu Chen, Qianhua He

School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China
{eexhfeng, eeyxli}@scut.edu.cn

ABSTRACT

Acquisition device clustering from speech recordings is a new and critical problem in the field of speech forensic, which aims at merging speech recordings acquired by the same device into one cluster without both pre-knowing prior information of the processed data and pre-training classifier. We propose a mobile phone clustering method, in which *deep Gaussian supervector* learned by deep neural network is used to represent the intrinsic trace left behind by mobile phone in speech recordings, and then spectral clustering technique is adopted to merge speech recordings acquired by the same mobile phone into one cluster. The performance of the proposed method is evaluated on a public corpus of speech recordings acquired by mobile phones. The results show that the proposed method is effective for mobile phone clustering from acquired speech recordings.

Index Terms— mobile phone clustering, deep Gaussian supervector, spectral clustering, speech forensic

1. INTRODUCTION

Various acquisition devices (e.g. mobile phone, telephone handsets) do not possess exactly the same frequency response due to the tolerance in the nominal values of their electronic components and structures [1]. As a result, each acquisition device leaves behind unique 'intrinsic trace' in the acquired speech recordings. Hence, acquisition devices can be recognized from their acquired speech recordings [2]. What's more, recognition of acquisition devices has been proved to be useful in the court for authenticating speech recordings presented as evidence [3, 4]. With the development of speech forensic technique during recent years, many researchers have carried out studies on acquisition device recognition based on the acquired speech recordings, e.g. microphone identification [5-13]; telephone handset identification [13-19]; mobile phone identification [1, 2, 19-23], verification [22, 24] and matching [25]. For instance, Hanilçi et al [1] identified the brands and models of mobile phones by the acquired speech recordings using Mel-frequency cepstral coefficients (MFCCs) as front-end feature and support vector machine (SVM) as back-end classifier. Afterwards, Kotropulos et al [2] adopted sketches of features as the input of sparse representation-based classifiers and SVM for identifying landline telephone and mobile phones. Recently, Zou et al presented sparse representation based feature for mobile phone verification and matching [24, 25].

Most of the previous studies focused on the problems of acquisition device recognition (i.e. identification or verification) from acquired speech recordings in a supervised way. That is, various audio features (e.g. MFCCs) are first extracted and then a classifier is trained for each acquisition device, and finally each test speech recording is identified or verified by using the pre-trained classifiers (e.g. SVM). In these studies, it was assumed that the identities and numbers of speech acquisition devices were known a priori. Hence, the main task of these studies is to determine which pre-defined identity of acquisition devices the test speech recording belongs to, or determine whether the test speech recording is acquired by the claimed device or not. However, the identities and numbers of acquisition devices are not always available for the court in practice due to various causes, e.g. label loss, acquisition device damage, uncertainty of acquisition device identity. On the other hand, when huge mass of speech recordings are provided by police officers or anybody else, the court probably cares about which speech recordings are acquired by the same device instead of knowing the specific identities of acquisition devices in the first place. In these cases, the problem confronted by the court becomes how to merge the speech recordings acquired by the same device into one cluster without knowing any prior information of the acquisition devices and without pre-training any classifiers. We call this new problem ***Acquisition Device Clustering*** (ADC) here. Main aims of ADC are to obtain the number (instead of the specific identity) of acquisition devices and to determine which speech recordings are acquired by the same device in an unsupervised way. ADC is significant in the forensic context, because the court often obtains many speech recordings without knowing acquisition device labels and meanwhile these speech recordings are quite crucial for solving criminal cases. To the best of our knowledge, no studies have been done on the problem of ADC.

It is well known that mobile phone has become one of the most frequently used communicational tools and is essential in our daily life. Speech evidences acquired by mobile phones have been increasingly submitted to the court or other law enforcement agencies as one of the most common form of evidences [24]. In this study, we take mobile phone as the representative acquisition device and try to address this new problem, i.e. ***Mobile Phone Clustering*** (MPC) from speech recordings. Inspired by the success of deep learning technique for feature representation [26] and spectral clustering technique for data clustering [27], we propose a method for MPC from acquired speech recordings. In the proposed method, we first

use Deep Neural Network (DNN) to extract deep acoustic feature and then use GMM-UBM (Gaussian Mixture Model-Universal Background Model) to extract *Deep Gaussian Supervector* (DGS) for representing the intrinsic trace left behind by mobile phone in speech recordings, and finally adopt spectral clustering to determine which speech recordings are acquired by the same mobile phone. The performance of the proposed method is evaluated on a public corpus of speech recordings acquired by mobile phones. Therefore, the main contribution of this study is to solve a new problem, i.e. MPC, using DGS and spectral clustering. The rest of the paper is organized as follows. Section 2 describes the proposed method, and Section 3 presents the experiments. Finally, conclusions and future works are given in Section 4.

2. THE METHOD

The block diagram of the proposed method for MPC is shown in Fig. 1, comprising two modules: deep Gaussian supervector and spectral clustering. N_p is the total number of mobile phones.

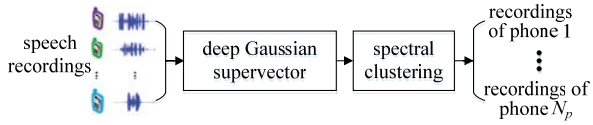

Fig. 1 Block diagram of the proposed method.

2.1. Deep Gaussian supervector

The block diagram for extracting DGS is illustrated in Fig. 2. Waveform of each speech recording is first segmented into frames for extracting MFCC feature, and then a feature extractor of DNN is built for extracting bottleneck feature from each frame of MFCC. Finally, a GMM-UBM is constructed for extracting DGS from bottleneck feature.

Fig. 2 Block diagram for extracting DGS.

2.1.1. MFCC

MFCC is the most popular feature for mobile phone recognition in the previous studies [1]. Hence, we use MFCC as one component for extracting DGS in this study. The extraction procedure of MFCC is shown in Fig. 3. Waveform of speech recording is first split into overlapping frames and windowed using a window function. The Discrete Fourier Transform (DFT) is used to compute the power spectrum which is then smoothed with a bank of triangular filters. The center frequencies of these triangular filters are uniformly spaced on the Mel-scale. Finally, logarithmic filterbank outputs are converted into MFCC by taking the Discrete Cosine Transform (DCT). We use 30ms frames with 15ms overlap and a Hamming window. The extraction of MFCC is detailed in [1].

Fig. 3 The extraction procedure of MFCC.

2.1.2. Bottleneck feature

The activation signals in the bottleneck layer (i.e. the narrowest hidden layer) can be used as a compact representation of the original high-dimensional inputs fed to the input layer of DNN [28]. We create a feature representation from the bottleneck layer neuron activations of DNN, which is called ***bottleneck feature***. The extraction process of the bottleneck feature is illustrated in Fig. 4.

For extracting bottleneck feature, we first extract 39 dimensional acoustic features for each speech recording, i.e. 13 MFCCs, 13 first-order deltas (ΔMFCCs) and 13 second-order deltas (ΔΔMFCCs). To model the dynamic properties of mobile phones, adjacent frames are also taken into consideration. A context of 31 frames of acoustic features is constructed and then discrete cosine transform with 16 bases is carried out on the acoustic features for being fed the input layer of DNN. Hence, the neuron number of the input layer of DNN is 624 (i.e. 39×16). The number of neuron in the hidden layer is the same for all hidden layers: 500 neurons, except for the bottleneck layer. The number of neuron of bottleneck layer, N_b (as shown in Fig. 4), i.e. the dimension of bottleneck feature, can be tuned using development data for obtaining the best performance. The impact of N_b on the performance of MPC will be discussed in the experiments. The number of neuron of output layer generally equals the number of category needed to be identified, and thus depends on the specific task (e.g. 21 mobile phones here). The bottleneck feature extractor of DNN is trained using the development data and then bottleneck feature is extracted for each speech recording of the test data using the DNN-based feature extractor. It should be noted that the trained DNN is used as feature extractor in this study instead of classifier in the previous studies.

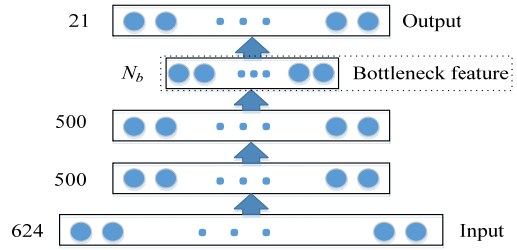

Fig. 4 The extraction of bottleneck feature. Digits and N_b denote neuron numbers in each layer.

2.1.3. Gaussian supervector

Gaussian Supervector (GS) has been proved success in representation of the intrinsic trace left behind by acquisition device in speech recordings [13]. GS extraction is briefly described as follows. Suppose that $\theta_{UBM} = \{\omega_m, \boldsymbol{u}_m, \boldsymbol{\Sigma}_m\}_{m=1}^{M}$ is a diagonal covariance Universal Background Model (UBM) with M Gaussian mixtures, where ω_m, $\boldsymbol{u}_m$, and $\boldsymbol{\Sigma}_m$ represent weight coefficient, mean vector and covariance matrix of the m^{th} Gaussian mixture, respectively. The UBM θ_{UBM} is trained using all speech recordings of the test data. Then, a GMM $\theta_{GMM} = \{\omega_m', \boldsymbol{u}_m', \boldsymbol{\Sigma}_m'\}_{m=1}^{M}$ is adapted from the UBM θ_{UBM} for each speech recording of the test data by using MAP (Maximum A

posteriori) algorithm [29]. Finally, M mean vectors of each GMM are successively concatenated as a super mean vector with total length of $M \times N_b$. For example, assume the number of Gaussian mixtures $M = 256$ and the dimension of bottleneck feature (input feature of UBM/GMM) $N_b = 39$, then the total length of GS for each speech recording is 9984.

The GMM-UBM for generating GS is fed by bottleneck feature which is created by DNN. Hence, the GS here is called DGS (Deep GS) which is used to represent the unique characteristic of each mobile phone.

2.2. Spectral clustering

Spectral clustering is an optimization problem of grouping together similar feature vectors based on eigenvectors of an affinity matrix that contains the similarity values measured between each pair of feature vectors [27]. Inspired by its success for image and speaker clustering [30, 31], we use spectral clustering technique for MPC in this study.

Assume that $\boldsymbol{x}_l$ denotes feature vector (e.g. DGS) of the l^{th} speech recording and $\boldsymbol{X}$ denotes a set of feature vectors for clustering, i.e. $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L\}$, where L is the total number of feature vectors (or speech recordings). The steps of spectral clustering is carried out as follows.

Step 1: Compute an affinity matrix $\boldsymbol{A}$ by,

$$A_{kl} = \exp\left(-\frac{d(\boldsymbol{x}_k, \boldsymbol{x}_l)^2}{2\sigma_k\sigma_l}\right), \qquad 1 \le k, l \le L\ , \tag{1}$$

where $d(\boldsymbol{x}_k, \boldsymbol{x}_l)$ is the Euclidean distance between $\boldsymbol{x}_k$ and $\boldsymbol{x}_l$, and σ_k (or σ_l) is a scaling factor for the feature vector $\boldsymbol{x}_k$ (or $\boldsymbol{x}_l$). The scaling factor σ_k is defined by,

$$\sigma_k = \sum_{\boldsymbol{x}_l \in close(\boldsymbol{x}_k)} d(\boldsymbol{x}_k, \boldsymbol{x}_l)/Q \quad , \tag{2}$$

where $close(\boldsymbol{x}_k)$ denotes the set containing Q nearest neighbors of $\boldsymbol{x}_k$. Q is experimentally set to 5 in this study.

Step 2: Generate diagonal matrix $\boldsymbol{D}$ whose element D_{kk} is the sum of all elements of the k^{th} row of $\boldsymbol{A}$, and then create the normalized affinity matrix $\boldsymbol{L}$ by

$$\boldsymbol{L} = \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{1/2} \ . \tag{3}$$

Step 3: Obtain eigenvalues λ_l and the corresponding eigenvectors $\boldsymbol{s}_l$ of $\boldsymbol{L}$ by decomposing $\boldsymbol{L}$. Rank the eigenvalues λ_l in descending order and assume $\lambda_1 \ge \lambda_2 \ge \ldots \ge \lambda_L$. The optimal cluster number N_c is estimated based on the gaps between adjacent eigenvalues by

$$N_c = \arg\max_{l \in [1,\ L]} \left(1 - \lambda_{l+1}/\lambda_l\right) \ . \tag{4}$$

Then form the matrix $\boldsymbol{S} = [\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_{N_c}] \in \boldsymbol{R}^{L \times N_c}$ by stacking the first N_c eigenvectors in columns.

Step 4: Generate the matrix $\boldsymbol{Y}$ by renormalizing each row of $\boldsymbol{S}$ to yield unit length,

$$Y_{ij} = \frac{S_{ij}}{\left(\sum_j S_{ij}^2\right)^{1/2}} \quad , \qquad 1 \le i \le L,\ 1 \le j \le N_c \ , \tag{5}$$

Step 5: Treat the rows of $\boldsymbol{Y}$ as points in $\boldsymbol{R}^{N_c}$ and cluster them into N_c clusters by K-means algorithm [32]. Assign the l^{th} speech recording to cluster c_k if and only if the l^{th} row of the matrix $\boldsymbol{Y}$ is assigned to c_k.

3. EXPERIMENTS

3.1. Experimental setup

The proposed method for MPC is evaluated on MOBIPHONE which is a public corpus of speech recordings acquired by 21 unique mobile phones of various models from 7 different brands, including HTC, LG, Nokia, Sony Ericsson, Apple, Samsung, and Vodafone. MOBIPHONE is the most common corpus used in the previous studies [19], whose speech recordings are acquired in a silent controlled environment. MOBIPHONE includes 24 speakers (12 males and 12 females) randomly chosen from the TIMIT database [33]. Each speaker reads 10 sentences (about 3s per sentence). The first two sentences are the same for every speaker, but the rest 8 are different. The audio data are saved as WAV format with sampling frequency of 16 kHz and 16 bits quantization. Thus there are 240 speech recordings for each mobile phone, and 5040 (i.e. 240×21) speech recordings in total. The development data are used to train DNN feature extractor, while the test data are used to evaluate the performance of the method for MPC. They are totally different and listed in Table 1.

Table 1 The details of the development and test data. #Phone: phone number, #Spkr: speaker number, #Rec./Phone: recording number per phone, and #Rec.: recording number in total.

	#Phone	#Spkr	#Rec./Phone	#Rec.
Dev.	21	24	120	2520
Test	21	24	120	2520

The entire speech recording, including speech and non-speech segments, is split into frames by a 30 ms Hamming window with half overlap. MFCCs + ΔMFCCs + ΔΔMFCCs with 39 dimensions are extracted from waveform of each speech recording. The neuron numbers of the input, hidden and output layers of the DNN feature extractor (one input layer, two hidden layers, one bottleneck layer and one output layer) are set to 624, 500 and 21, respectively, as given in Fig. 4. The number of Gaussian mixtures M is set to 256.

Let n_{ij} be the total number of speech recordings in cluster i acquired by mobile phone j; N_p be the total number of mobile phones; N_c be the total number of clusters; N be the total number of speech recordings; $n_{\bullet j}$ be the total number of speech recordings acquired by mobile phone j; $n_{i\bullet}$ be the total number of speech recordings in cluster i. The following three equations establish relationships between the above variables:

$$n_{i\bullet} = \sum_{j=1}^{N_p} n_{ij} \ , \qquad n_{\bullet j} = \sum_{i=1}^{N_c} n_{ij} \ , \qquad N = \sum_{i=1}^{N_c}\sum_{j=1}^{N_p} n_{ij} \ . \tag{6}$$

The purity of cluster i, $\pi_{i\bullet}$, is defined below:

$$\pi_{i\bullet} = \sum_{j=1}^{N_p} \frac{n_{ij}^2}{n_{i\bullet}^2} \ , \tag{7}$$

Average Cluster Purity (*ACP*) is defined below:

$$ACP = \frac{1}{N}\sum_{i=1}^{N_c} \pi_{i\bullet} n_{i\bullet} \ . \tag{8}$$

The phone purity for mobile phone j, $\pi_{\bullet j}$, is defined below:

$$\pi_{\bullet j} = \sum_{i=1}^{N_c} \frac{n_{ij}^2}{n_{\bullet j}^2} \ , \tag{9}$$

Average Phone Purity (*APP*) is defined below:

$$APP = \frac{1}{N}\sum_{j=1}^{N_p} \pi_{\bullet j} n_{\bullet j} \quad . \tag{10}$$

Finally, K score is used to characterize the overall performance of mobile phone clustering methods, which is equal to:

$$K = \sqrt{ACP \times APP} \quad . \tag{11}$$

Besides the K score, we use another two metrics: Normalized Mutual Information (*NMI*) [30] and Clustering Accuracy (*CA*) [30], to measure the quality between the produced clusters and the ground truth categories. The *NMI* score is defined as

$$NMI = \frac{\sum_{i=1}^{N_c}\sum_{j=1}^{N_p} n_{ij} \log\left(\frac{N \times n_{ij}}{n_{i\bullet} \times n_{\bullet j}}\right)}{\sqrt{\left(\sum_i n_{i\bullet} \log\frac{n_{i\bullet}}{N}\right)\left(\sum_j n_{\bullet j} \log\frac{n_{\bullet j}}{N}\right)}} \quad , \tag{12}$$

The variables in Eq. (12) are same as that defined in Eq. (6). The *NMI* score is 1 if the clustering results perfectly match the true labels, and the score is close to 0 if feature vectors are randomly partitioned.

The *CA* is defined by,

$$CA = \left[\sum_{i=1}^{N} \delta\left(y_i, \mathrm{map}\left(c_i\right)\right)\right] \Big/ N \quad , \tag{13}$$

where y_i and c_i denote the true label of mobile phone and the obtained cluster label of the i^{th} speech recording, respectively. $\delta(y, c)$ is a function that is equal to 1 if $y = c$ and 0 otherwise. map(•) is a permutation function that maps each cluster label to a true label and optimal matching can be obtained by the Hungarian algorithm [34]. The higher the scores of K, *NMI* and *CA*, the better the clustering quality.

3.2. Experimental results

We first discuss the impact of N_b (the dimension of bottleneck feature) on the performance of the proposed method evaluated on the development data, and compare the proposed DGS with the previous features, i.e. MFCCs [1], Gaussian Supervector (GS) [13], I-Vector (IV) [35] and Sparse Representation based Feature (SRF) [24] adopted in the previous studies on the test data. The detailed procedures for extracting these features can be found in [1, 13, 35, 24], respectively. Here, spectral clustering is used as clustering algorithm for all features.

As can be seen from Fig. 5 that all metrics (K scores, *CA* and *NMI*) are influenced by N_b and they reach the highest values when N_b = 39. Hence, when the proposed method is evaluated on the test data, N_b is set to 39. K scores, *CA* and *NMI* obtained by different features for mobile phone clustering are given in Table 2. The proposed DGS consistently obtains the best performance among all features in terms of all metrics when evaluated on the test data. As for the K score, the proposed DGS obtains 93.81% and achieves gains of 15.26%, 9.02%, 8.23% and 4.35% compared to MFCC, GS, IV and SRF, respectively. As for the *CA*, the proposed DGS yields 96.75% and attains improvements of 17.47%, 5.56%, 4.69% and 3.22% compared to MFCC, GS, IV and SRF, respectively. As for the *NMI*, the proposed DGS also obtains the highest value of 95.11% and earns betterments of 13.44%, 5.50%, 6.02% and 2.43% compared to MFCC, GS, IV and SRF, respectively.

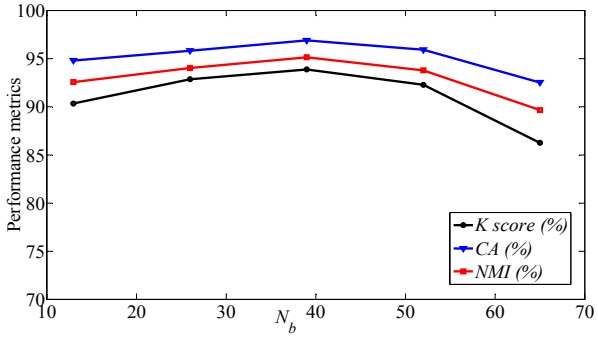

Fig. 5 The impact of N_b on the K score, *CA* and *NMI* of the proposed method evaluated on the development data.

Table 2 Performance comparison of different features for mobile phone clustering. DGS: Deep Gaussian Supervector; MFCC: Mel Frequency Cepstral Coefficient; GS: Gaussian Supervector; IV: I-Vector; SRF: Sparse Representation based Feature. *K*: *K* score; *CA*: Clustering Accuracy; *NMI*: Normalized Mutual Information.

	DGS	MFCC	GS	IV	SRF
K	**93.81**	78.55	84.79	85.58	89.46
CA	**96.75**	79.28	91.19	92.06	93.53
NMI	**95.11**	81.67	89.61	89.09	92.68

The results above have proved the effectiveness of the proposed method for mobile phone clustering. This is a preliminary study for acquisition device clustering. It should be noted that this study focused only on mobile phone due to its popularity. But, the proposed method can be extended to other types of acquisition devices.

4. CONCLUSIONS

In this study, we try to tackle a new problem of mobile phone clustering from acquired speech recordings by using both the feature representation technique of deep Gaussian supervector and spectral clustering technique. The proposed deep Gaussian supervector outperforms other features adopted in the previous studies. The experimental results have shown that the proposed method is effective for solving the new problem of mobile phone clustering. The future work includes enlarging the size of the experimental data and exploring other features and clustering algorithms for further improving the performance of the methods for acquisition device clustering.

5. ACKNOWLEDGMENT

The work was supported by the NSFC (61101160, 61301300, 61401161, 61571192), the Project of the Pearl River Young Talents of S&T, Guangzhou (2013J2200070), S&T Planning Project of Guangdong (2015A030313600, 2015A010103006, 2015A010103003), Fundamental Research Funds for the Central Universities (2015ZZ102, 2015ZM143, 2015ZM145). We acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

6. REFERENCES

[1] C. Hanilci, F. Ertas, T. Ertas, and O. Eskidere, "Recognition of brand and models of cell-phones from recorded speech signals," *IEEE Trans. on Information Forensics Security*, vol. 7, no. 2, pp. 625–634, 2012.

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

[3] H. Zhao and H. Malik, "Audio recording location identification using acoustic environment signature," *IEEE Trans. Information Forensics Security*, vol. 8, no. 11, pp. 1746–1759, 2013.

[4] H. Malik and H. Zhao, "Recording environment identification using acoustic reverberation," in *Proc. of ICASSP*, 2012, pp. 1833–1836.

[5] C. Kraetzer, M. Schott, and J. Dittmann, "Unweighted fusion in microphone forensics using a decision tree and linear logistic regression models," in *Proc. of 11th ACM Multimedia and Security Workshop*, 2009, pp. 49–56.

[6] C. Kraetzer, A. Oermann, J. Dittmann, and A. Lang, "Digital audio forensics: a first practical evaluation on microphone and environment classification," in *Proc. of the 9th Workshop on Multimedia and Security*, 2007, pp. 63–74.

[7] C. Kraetzer, K. Qian, M. Schott, and J. Dittmann, "A context model for microphone forensics and its application in evaluations," *Media Watermarking, Security, and Forensics III*, vol. 7880, 2011.

[8] H. Malik and J. Miller, "Microphone identification using higher-order statistics," in *Proc. of AES 46th Conf. on Audio Forensics*, 2012, pp. 1-10.

[9] R. Buchholz, C. Kraetzer, and J. Dittmann, "Microphone classification using Fourier coefficients," in *Lecture Notes in Computer Science*, 2010, vol. 5806/2009, pp. 235–246.

[10] D. Garcia-Romero and C. Espy-Wilson, "Speech forensics: automatic acquisition device identification," *Acoustical Society of America,* vol. 127, no. 3, pp. 2044–2044, 2010.

[11] Ö. *ESKİDERE*, "Source microphone identification from speech recordings based on a Gaussian mixture model," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 22, no. 3, pp. 754–767, 2014.

[12] L. Cuccovillo and P. Aichroth, "Open-set microphone classification via blind channel analysis," in *Proc. of ICASSP*, 2016, pp. 2074-2078.

[13] D. Garcia-Romero and C. Espy-Wilson, "Automatic acquisition device identification from speech recordings," in *Proc. of ICASSP*, 2010, pp. 1806–1809.

[14] D. A. Reynolds, "HTIMIT and LLHDB: speech corpora for the study of handset transducer effects," in *Proc. of ICASSP*, 1997, vol. 2, pp. 1535–1538.

[15] D. Garcia-Romero and C. Espy-Wilson, "Automatic acquisition device identification from speech recordings," *J. of Audio Engineering Society*, vol. 124, no. 4, pp.2530–2530, 2009.

[16] Y. Panagakis and C. Kotropoulos, "Telephone handset identification by feature selection and sparse representations" in *Proc. of IEEE Int. Workshop on Information Forensics and Security*, 2012, pp. 73–78.

[17] Y. Panagakis and C. Kotropoulos, "Automatic telephone handset identification by sparse representation of random spectral features," in *Proc. of 14th ACM Multimedia and Security Workshop*, 2012, pp. 91–96.

[18] C. Kotropoulos, "Telephone handset identification using sparse representations of spectral feature sketches," in *Proc. of First Int. Workshop on Biometrics and Forensics*, 2013.

[19] C. Kotropoulos and S. Samaras, "Mobile phone identification using recorded speech signals," in *Proc. of the 19th Int. Conf. on Digital Signal Processing,* 2014, pp. 586–591.

[20] L. Zou, J. C. Yang, and T. S. Huang, "Automatic cell phone recognition from speech recordings," in *IEEE China Summit & Int. Conf. on Signal & Information Process.*, 2014, pp.621–625.

[21] M. Jahanirad, A. W. A. Wahab, N. B. Anuar, M. Y. I. Idris, and M. N. Ayub, "Blind source mobile device identification based on recorded call," *Engineering Applications of Artificial Intelligence,* vol. 36, pp. 320–331, 2014.

[22] C. Hanilçi and T. Kinnunen, "Source cell-phone recognition from recorded speech using non-speech segments," *Digital Signal Processing*, vol. 35, pp. 75–85, 2014.

[23] *Ömer ESKİDERE,* Identifying acquisition devices from recorded speech signals using wavelet-based features, *Turkish Journal of Electrical Engineering & Computer Sciences*, 2016, vol. 24, pp. 1942-1954.

[24] L. Zou, Q. H. He, and X. H. Feng, "Cell phone verification from speech recordings using sparse representation," in *Proc. of ICASSP*, 2015, pp. 1787–1791.

[25] L. Zou, Q. H. He, J. Yang and Y. Li, "Source cell phone matching from speech recordings by sparse representation and KISS metric," in *Proc. of ICASSP*, 2016, pp. 2079–2083.

[26] D. Yu, and L. Deng, "Automatic speech recognition: a deep learning approach," Springer Press, 2015 Edition.

[27] A.Y. Ng, M.I. Jordan,Y. Weiss, "On spectral clustering: analysis and an algorithm," *Advances in neural information processing systems*, 2001, pp. 849-856.

[28] D. Yu, and M.L. Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Proc. of INTERSPEECH*, 2011, pp.237–240.

[29] 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.

[30] W.Y. Chen, Y. Song, H. Bai, C.J. Lin and E.Y. Chang, "Parallel spectral clustering in distributed systems," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33. no. 3, 2011.

[31] K. Iso, "Speaker clustering using vector quantization and spectral clustering," in *Proc. of ICASSP*, 2010, pp. 4986-4989.

[32] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Information Theory*, vol. 28, no. 2, pp. 129-137, 1982.

[33] J. Garofolo, "Getting started with the DARPA TIMIT cd-rom: An acoustic phonetic continuous speech database," National Inst. Standards and Technology, Tech. Rep., 1988.

[34] C.H. Papadimitriou and K. Steiglitz, "Combinatorial Optimization: Algorithms and Complexity," Dover Publications, 1998.

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