Using Regularized Fisher Discriminant Analysis To Improve The Performance Of Gaussian Supervector In Session And Device Identification

Yuechi Jiang
Centre for Signal Processing
Dept of EIE, PolyU
Hung Hom, Hong Kong
yuechi.jiang@connect.polyu.hk

Frank H. F. Leung
Centre for Signal Processing
Dept of EIE, PolyU
Hung Hom, Hong Kong
frank-h-f.leung@polyu.edu.hk

***Abstract*—In this paper, we propose Regularized Fisher Discriminant Analysis (RFDA) as a projection method applied on Gaussian Supervector (GSV). GSV was originally applied on speaker recognition and verification, and has exhibited good performance. Recently GSV has also been applied in audio forensics area, such as recording device identification. It has been shown that GSV can also capture useful information related to the recording device. In this paper, we show that GSV can also be applied in telephone session identification. However, although GSV can capture useful information for different identification purposes, the performance of the raw GSV may not be so good. Thus, we apply RFDA-based projection method on the raw GSV, and find that this projection method can significantly improve the performance of the raw GSV, in both telephone session identification and recording device identification tasks.**

Keywords—audio forensics, Regularized Fisher Discriminant Analysis, projected Gaussian Supervector, telephone session identification, recording device identification

I. INTRODUCTION

Besides the content, a recorded audio signal also embeds the clues about the encoding algorithm, the recording device, or the recording date. These clues can be useful in some situations, for example, the recorded audio signal is to be used as court evidence [1]. In this paper, we focus on the telephone session identification task, where different sessions are corresponding to different recording dates. We also consider the microphone identification task, where each audio recording is recorded using one microphone.

In the literature, some researchers try to extract the recording date information from a recorded audio signal using the embedded Electric Network Frequency (ENF) signal. ENF signal is the operating frequency of the power grid. If the recording device is near the power grid, the ENF signal will be embedded in the recorded audio signal due to electro-magnetic induction [1]. The ENF signal has been found to be useful to authenticate recorded audio signals, such as detecting whether the audio has been edited (insertion or deletion) by analysing the phase discontinuity of the ENF signal [2] or the higher order harmonics of the ENF signal [3]. Audio edit detection can also be performed by examining the instantaneous magnitude variations of the ENF signal [4].

It was found that the ENF signal could be used to verify the recording date of an audio recording by comparing the embedded ENF signal with the reference ENF signals in the database [5]. However, the ENF signal may be tampered, which reduces the robustness of the ENF-based recording date verification system [6]. In addition, the ENF signal may not always be available in an audio recording if the recording device is far from the power grid.

Session variability has always been a consideration in speaker recognition studies. In particular, for telephone speeches, the session variability is induced by the telephone network variability, which may be used as a clue for identifying the recording date of the telephone conversation.

In this paper, we try to tackle the recording date identification problem by identifying different telephone sessions using Machine Learning techniques. Given a set of training audio signals recorded in different sessions, we first construct a model containing the session information, and then classify an unknown audio recording into a session based on the model. Different feature extraction methods, such as the averaged frame-level feature vectors and the Gaussian Supervector (GSV) will be used. Besides, we also propose a projection method called Regularized Fisher Discriminant Analysis (RFDA), and then extend it to a kernel version of RFDA. The kernel RFDA-based projected GSV will be used and compared with the raw GSV. Linear Support Vector Machine (SVM) will be used as the identifier. An overview of different feature extraction methods is shown in Fig. 1.

In fact, GSV has also been applied in recording device identification tasks and exhibits good performance [7]. Device identification aims to identify which device is used to record the audio signal, from a set of candidate devices. This device information helps authenticate whether a claim on the recording device is true. In this paper, we consider microphone identification, which aims to identify the underlying recording microphone for an audio recording. Interestingly, we find that using RFDA-based projection can also improve the performance of the GSV for doing

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

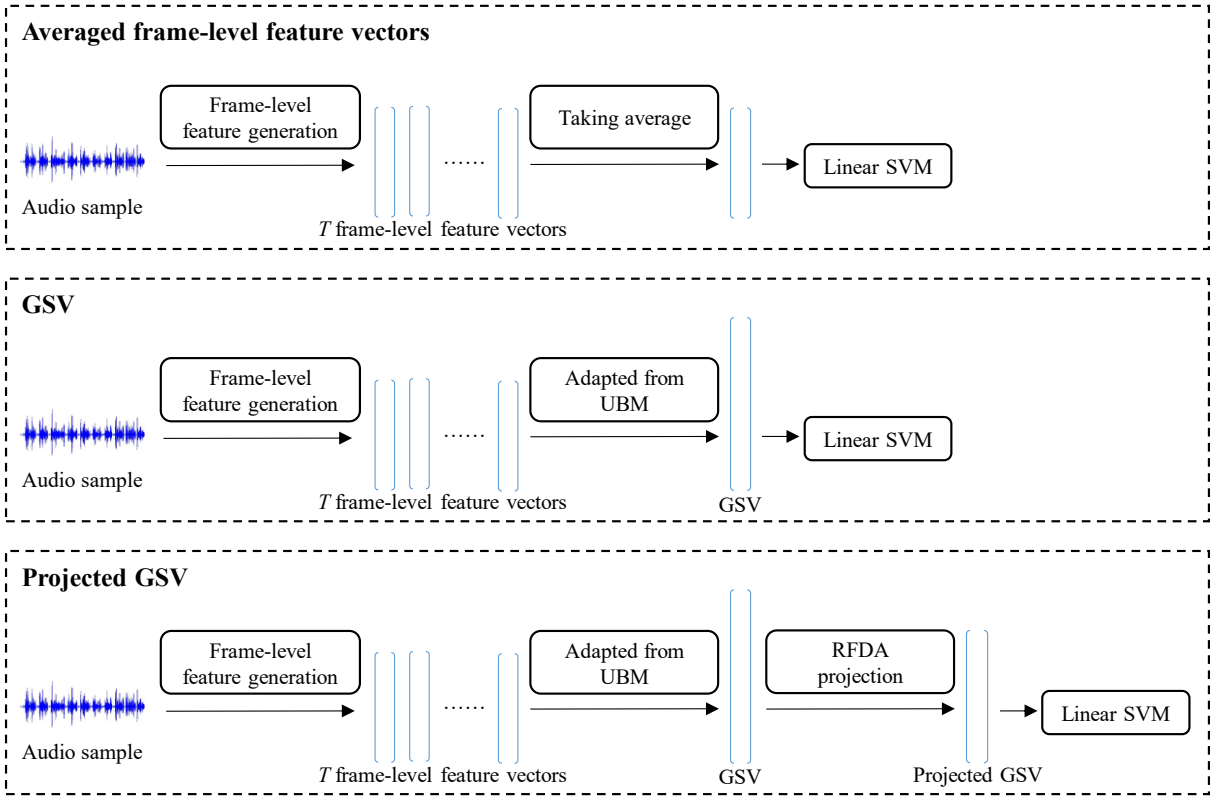

Fig.1. Overview of different feature extraction methods.

microphone identification. It seems the proposed RFDA method has the potential to reorganize the GSV serving different identification purposes.

This paper is organized as follows. In Section II, the GSV, the RFDA method and the kernel version of RFDA are described. In Section III, the dataset used in our experiment is briefly described. In Section IV, experimental results are presented and discussed. A conclusion will be drawn in Section V.

II. GSV AND RFDA

A. Frame-level Feature Extraction

In this paper, Mel-frequency Cepstral Coefficients (MFCC) is used as the frame-level feature vector, which is commonly chosen for speech signals. MFCC can be obtained as described in [8]. In our experiment, we use Hamming window with 50ms frame length and 10ms frame shift to extract short-time frames. We then apply 48 triangular filters in Mel scale to form a 24-dimension MFCC excluding the energy coefficient.

B. Gaussian Supervector

Gaussian Supervector (GSV) is obtained by adapting a Universal Background Model (UBM), and a UBM is in fact a Gaussian Mixture Model (GMM). Given a set of training audios, we first compute the MFCCs, and then construct a 1-component GMM, which is a simple Gaussian model. Then for a desired M-component GMM (where M is assumed to be a power of 2), we use the mixture splitting technique [9] and Expectation-Maximization (EM) algorithm [10] to split and then retrain the GMM for $\log_2 M$ times. Each time, we split the current GMM to double the number of mixture components using the mixture splitting technique, and then retrain the newly split GMM using the EM algorithm.

Suppose that we have constructed a UBM with M components, denoted as $\theta_M = \{\omega_i, \mu_i, \sigma_i \mid i = 1,2...M\}$, where ω_i, μ_i and σ_i are the weight, the mean vector and the standard deviation vector (assuming diagonal covariance matrix in GMM) for the i-th mixture component. We then use (1) ~ (4) to compute the adapted mean vector for a speech audio signal whose MFCCs are $\{z_1, z_2 \ldots z_T\}$,

$$\Pr(i \mid z_t, \theta_M) = \frac{\omega_i p(z_t \mid \mu_i, \sigma_i)}{\sum_{j=1}^{M} \omega_j p(z_t \mid \mu_j, \sigma_j)} \tag{1}$$

$$n_i = \sum_{t=1}^{T} \Pr(i \mid z_t, \theta_M) \tag{2}$$

$$E_i = \frac{1}{n_i} \sum_{t=1}^{T} \Pr(i \mid z_t, \theta_M) z_t \tag{3}$$

$$\mu_i' = \frac{n_i}{n_i + \gamma} E_i + \frac{\gamma}{n_i + \gamma} \mu_i \tag{4}$$

where in (1) $p(z_t \mid \mu_i, \sigma_i)$ is the i-th component Gaussian probability density function, and in (4) γ is the relevance factor reflecting the relationship between the adapted mean vector μ_i' and the UBM mean vector μ_i [10]. Then the GSV is obtained by concatenating all the μ_i'.

C. Regularized Fisher Discriminant Analysis (RFDA)

Suppose that we have a set of N training vectors $X=(x_1, x_2 \ldots x_N)$ belonging to K classes, where class k is denoted as C_k, containing N_k training vectors, and $\sum_{k=1}^{K} N_k = N$. Fisher Discriminant Analysis (FDA) then aims to find a set of corresponding projected vectors $Y=(y_1, y_2 \ldots y_N)$ such that the vectors belonging to the same class are moved together while the vectors belonging to different classes are separated. The projection is given by (5), where W is the projection matrix containing I columns, whose i-th column vector is denoted by w_i, and each column of W is a projection direction [11].

$$y_n = W^T x_n \tag{5}$$

In (5), the projection matrix W is obtained by maximizing the objective function $J(W)$ given in (6), where S_B is the between-class scatter matrix and S_W is the within-class scatter matrix as given by (7) and (8).

$$J(W) = Trace\left\{\frac{W^T S_B W}{W^T S_W W}\right\} \tag{6}$$

$$S_B = \sum_{k=1}^{K} N_k (m_k - m)(m_k - m)^T \tag{7}$$

$$S_W = \sum_{k=1}^{K} \sum_{x_n \in C_k} (x_n - m_k)(x_n - m_k)^T \tag{8}$$

In (7) and (8), m_k is the mean vector of the training vectors belonging to class k, and m is the mean vector of all the training vectors, given by (9) and (10) respectively.

$$m_k = \frac{1}{N_k} \sum_{x_n \in C_k} x_n \tag{9}$$

$$m = \frac{1}{N} \sum_{n=1}^{N} x_n = \frac{1}{\sum_{k=1}^{K} N_k} \sum_{k=1}^{K} N_k m_k \tag{10}$$

Maximizing (6) is equivalent to finding the eigenvectors of $S_W^{-1} S_B$ [11]. However, as the rank of S_B is at most K-1, the rank of $S_W^{-1} S_B$ is at most K-1, there are at most K-1 independent eigenvectors, which means there are at most K-1 orthogonal projection directions [11]. Normally the number of classes K is small, which resulting in low efficiency of using the traditional FDA to construct projected vectors, as the number of projection directions is small.

The objective function in (6) aims to make the between-class covariance $W^T S_B W$ large (i.e. separate vectors coming from different classes), and make the within-class covariance $W^T S_W W$ small (i.e. group vectors coming from the same class). Instead of maximizing (6), we can realize the similar target by maximizing the objective function defined in (11) subject to some constraints, where α is a pre-defined regularization parameter. Eq. (11) is the objective function of our proposed Regularized FDA (RFDA). There is a similar objective function as Eq. (11) in [12], but is more complicated and lacking discussions. It seems the objective function in (6) is quite popular, but we are going to show the modified objective function (11) is more useful.

$$\begin{aligned} J'(W) &= Trace\left\{W^T S_B W\right\} - Trace\left\{\alpha W^T S_W W\right\} \\ &= \sum_{i=1}^{I} w_i^T S_B w_i - \alpha \sum_{i=1}^{I} w_i^T S_W w_i \\ &subject\ to \quad w_i^T w_i = 1 \ for\ i = 1,2 \ldots I \end{aligned} \tag{11}$$

By applying Lagrange multiplier, we can reformulate (11) as (12), where λ_i is a non-zero coefficient.

$$\begin{aligned} L(W, \lambda_1, \lambda_2 \ldots \lambda_I) &= J'(W) - \sum_{i=1}^{I} \lambda_i \left(w_i^T w_i - 1\right) \\ &= \sum_{i=1}^{I} w_i^T S_B w_i - \alpha \sum_{i=1}^{I} w_i^T S_W w_i - \sum_{i=1}^{I} \lambda_i \left(w_i^T w_i - 1\right) \end{aligned} \tag{12}$$

By setting the derivative of $L(W, \lambda_1, \lambda_2 \ldots \lambda_I)$ to zero with respect to w_i, we obtain (13); by setting the derivative of $L(W, \lambda_1, \lambda_2 \ldots \lambda_I)$ to zero with respect to λ_i, we obtain (14).

$$\begin{aligned} &\frac{\partial L(W, \lambda_1, \lambda_2 \ldots \lambda_I)}{\partial w_i} = 2 S_B w_i - 2\alpha S_W w_i - 2\lambda_i w_i = 0 \\ &\Leftrightarrow \quad (S_B - \alpha S_W) w_i = \lambda_i w_i \end{aligned} \tag{13}$$

$$\frac{\partial L(W, \lambda_1, \lambda_2 \ldots \lambda_I)}{\partial \lambda_i} = w_i^T w_i - 1 = 0 \tag{14}$$

Thus w_i is an eigenvector of $S_B - \alpha S_W$ with λ_i being the corresponding eigenvalue. Compared with $S_W^{-1} S_B$ whose rank is limited by the lower rank of S_B and S_W, which is at most K-1, the rank of $S_B - \alpha S_W$ is limited by the larger rank of S_B and S_W, which can be larger than K-1. As long as (13) is solved, (14) can be satisfied easily, because we can easily normalize the eigenvector to have unit length. The regularization parameter α controls the trade-off between the separating ability and the grouping ability of the projection. We emphasize more on

separating vectors from different classes by choosing a smaller α, and emphasize more on grouping vectors from the same class by choosing a larger α.

D. Kernel-based RFDA

In [13], the kernel version for two-class FDA is proposed as an extension to the traditional two-class FDA. In this part, we derive the kernel version for multiple-class RFDA in a similar way. From (7), (9) and (10), we have,

$$\begin{aligned}
S_B w_i &= \sum_{k=1}^{K} N_k (m_k - m)(m_k - m)^T w_i \\
&= \sum_{k=1}^{K} N_k m_k {m_k}^T w_i - \sum_{k=1}^{K} N_k m_k m^T w_i \\
&\quad - \sum_{k=1}^{K} N_k m {m_k}^T w_i + \sum_{k=1}^{K} N_k m m^T w_i \\
&= \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i - \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n m^T w_i \\
&\quad - \sum_{k=1}^{K} m N_k {m_k}^T w_i + N m m^T w_i \\
&= \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i - \sum_{n=1}^{N} x_n m^T w_i \\
&\quad - m N m^T w_i + N m m^T w_i \\
&= \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i - \sum_{n=1}^{N} x_n m^T w_i
\end{aligned} \tag{15}$$

Similarly, from (8), (9) and (10), we have,

$$\begin{aligned}
S_w w_i &= \sum_{k=1}^{K} \sum_{x_n \in C_k} (x_n - m_k)(x_n - m_k)^T w_i \\
&= \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {x_n}^T w_i - \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i \\
&\quad - \sum_{k=1}^{K} \sum_{x_n \in C_k} m_k {x_n}^T w_i + \sum_{k=1}^{K} \sum_{x_n \in C_k} m_k {m_k}^T w_i \\
&= \sum_{n=1}^{N} x_n {x_n}^T w_i - \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i \\
&\quad - \sum_{k=1}^{K} m_k \sum_{x_n \in C_k} {x_n}^T w_i + \sum_{k=1}^{K} N_k m_k {m_k}^T w_i \\
&= \sum_{n=1}^{N} x_n {x_n}^T w_i - \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i \\
&\quad - \sum_{k=1}^{K} m_k N_k {m_k}^T w_i + \sum_{k=1}^{K} N_k m_k {m_k}^T w_i \\
&= \sum_{n=1}^{N} x_n {x_n}^T w_i - \sum_{k=1}^{K} \sum_{x_n \in C_k} x_n {m_k}^T w_i
\end{aligned} \tag{16}$$

From (15) and (16), we can reformulate (13) as,

$$\sum_{k=1}^{K} \sum_{x_n \in C_k} x_n (1+\alpha) {m_k}^T w_i - \sum_{n=1}^{N} x_n \left(m^T + \alpha {x_n}^T \right) w_i = \lambda_i w_i \tag{17}$$

We can further reformulate (17) as (18),

$$\sum_{n=1}^{N} x_n \beta_n^{(k)} = \lambda_i w_i \tag{18}$$

where

$$\beta_n^{(k)} = (1+\alpha) {m_k}^T w_i - \left(m^T + \alpha {x_n}^T \right) w_i \quad for\ x_n \in C_k \tag{19}$$

Further reformulating (18) gives (20), where v_i is a column vector containing N elements, and X is the data matrix whose n-th column is the training vector x_n. Eq. (20) implies that w_i must lie in the space spanned by the set of training vectors x_n, as w_i can be expressed as a linear combination of the training vectors.

$$w_i = \sum_{n=1}^{N} \frac{\beta_n^{(k)}}{\lambda_i} x_n = X v_i \tag{20}$$

Reconsidering (7), (8) and (11) in view of (20), we have,

$${w_i}^T m_k = {v_i}^T X^T \left(\frac{1}{N_k} \sum_{x_n \in C_k} x_n \right) = {v_i}^T \frac{1}{N_k} X^T \sum_{x_n \in C_k} x_n = {v_i}^T M_k \tag{21}$$

$${w_i}^T m = {v_i}^T X^T \left(\frac{1}{N} \sum_{n=1}^{N} x_n \right) = {v_i}^T \frac{1}{N} X^T \sum_{n=1}^{N} x_n = {v_i}^T M \tag{22}$$

$${w_i}^T x_n = {v_i}^T X^T x_n = {v_i}^T Q_n \tag{23}$$

where we define M_k, M and Q_n as a column vector containing N elements, whose j-th element is given by (24), (25) and (26) below respectively.

$$\begin{aligned}
(M_k)_j &= \frac{1}{N_k} {x_j}^T \sum_{x_n \in C_k} x_n = \frac{1}{N_k} \sum_{x_n \in C_k} {x_j}^T x_n \\
&= \frac{1}{N_k} \sum_{x_n \in C_k} k(x_j, x_n)
\end{aligned} \tag{24}$$

$$(M)_j = \frac{1}{N} {x_j}^T \sum_{n=1}^{N} x_n = \frac{1}{N} \sum_{n=1}^{N} {x_j}^T x_n = \frac{1}{N} \sum_{n=1}^{N} k(x_j, x_n) \tag{25}$$

$$(Q_n)_j = {x_j}^T x_n = k(x_j, x_n) \tag{26}$$

In (24) ~ (26), $k(x_j, x_n)$ is a kernel function, defining the inner product of two vectors x_j and x_n. Plugging (21) ~ (26) into (11), we have,

$$
\begin{aligned}
J'(W) &= \sum_{i=1}^{I} w_i^T S_B w_i - \alpha \sum_{i=1}^{I} w_i^T S_W w_i \\
&= \sum_{i=1}^{I} w_i^T \left(\sum_{k=1}^{K} N_k (m_k - m)(m_k - m)^T \right) w_i \\
&\quad - \alpha \sum_{i=1}^{I} w_i^T \left(\sum_{k=1}^{K} \sum_{x_n \in C_k} (x_n - m_k)(x_n - m_k)^T \right) w_i \\
&= \sum_{i=1}^{I} \sum_{k=1}^{K} N_k w_i^T (m_k - m)(m_k - m)^T w_i \\
&\quad - \alpha \sum_{i=1}^{I} \sum_{k=1}^{K} \sum_{x_n \in C_k} w_i^T (x_n - m_k)(x_n - m_k)^T w_i \\
&= \sum_{i=1}^{I} \sum_{k=1}^{K} N_k v_i^T (M_k - M)(M_k - M)^T v_i \\
&\quad - \alpha \sum_{i=1}^{I} \sum_{k=1}^{K} \sum_{x_n \in C_k} v_i^T (Q_n - M_k)(Q_n - M_k)^T v_i \\
&= \sum_{i=1}^{I} v_i^T \sum_{k=1}^{K} N_k (M_k - M)(M_k - M)^T v_i \\
&\quad - \alpha \sum_{i=1}^{I} v_i^T \sum_{k=1}^{K} \sum_{x_n \in C_k} (Q_n - M_k)(Q_n - M_k)^T v_i
\end{aligned}
\tag{27}
$$

If we define U_B and U_W as in (28) and (29) below, $J'(W)$ can be reformulated as (30), where we define a new matrix V whose i-th column is v_i.

$$
U_B = \sum_{k=1}^{K} N_k (M_k - M)(M_k - M)^T \tag{28}
$$

$$
U_W = \sum_{k=1}^{K} \sum_{x_n \in C_k} (Q_n - M_k)(Q_n - M_k)^T \tag{29}
$$

$$
J'(W) = \sum_{i=1}^{I} v_i^T U_B v_i - \alpha \sum_{i=1}^{I} v_i^T U_W v_i = J(V) \tag{30}
$$

This time, instead of finding w_i, we need to find v_i. In order to obtain a unique solution for maximizing the objective function in (30), we need to normalize v_i such that

$$
v_i^T v_i = 1 \; for \; i = 1,2...I \tag{31}
$$

Then we can combine (30) and (31) using Lagrange multiplier, as shown in (32), where λ_i' is a non-zero coefficient.

$$
\begin{aligned}
L(V, \lambda_1', \lambda_2' ... \lambda_I',) &= J(V) - \sum_{i=1}^{I} \lambda_i' \left(v_i^T v_i - 1 \right) \\
&= \sum_{i=1}^{I} v_i^T U_B v_i - \alpha \sum_{i=1}^{I} v_i^T U_W v_i - \sum_{i=1}^{I} \lambda_i' \left(v_i^T v_i - 1 \right)
\end{aligned}
\tag{32}
$$

Then after setting the derivative of $L(V, \lambda_1', \lambda_2' ... \lambda_I',)$ to zero with respect to v_i and λ_i', we have

$$
\begin{aligned}
&\frac{\partial L(V, \lambda_1', \lambda_2' ... \lambda_I',)}{\partial v_i} = 2U_B v_i - 2\alpha U_W v_i - 2\lambda_i' v_i = 0 \\
&\Leftrightarrow \quad (U_B - \alpha U_W) v_i = \lambda_i' v_i
\end{aligned}
\tag{33}
$$

$$
\frac{\partial L(V, \lambda_1', \lambda_2' ... \lambda_I',)}{\partial \lambda_i'} = v_i^T v_i - 1 = 0 \tag{34}
$$

Thus, instead of finding w_i, which is the eigenvector of $S_B - \alpha S_W$, equivalently, we now need to find v_i, which is the eigenvector of $U_B - \alpha U_W$. Eq. (33) and (34) are the kernel version of RFDA. After finding v_i based on the training vectors, for a given input vector t, we can then calculate its projected version t' as (35) below, where t'_i is the i-th element of t', $(v_i)_n$ is the n-th element of v_i, x_n is the n-th training vector, $k(x_n, t)$ is the kernel function in (24) ~ (26).

$$
\begin{aligned}
t_i' &= w_i^T t = (X v_i)^T t = v_i^T X^T t = \sum_{n=1}^{N} (v_i)_n x_n^T t \\
&= \sum_{n=1}^{N} (v_i)_n k(x_n, t)
\end{aligned}
\tag{35}
$$

We can then normalize t'_i with respect to w_i implicitly:

$$
\begin{aligned}
\frac{t_i'}{\|w_i\|} &= \frac{t_i'}{\sqrt{w_i^T w_i}} = \frac{\sum_{n=1}^{N} (v_i)_n k(x_n, t)}{\sqrt{(X v_i)^T (X v_i)}} = \frac{\sum_{n=1}^{N} (v_i)_n k(x_n, t)}{\sqrt{v_i^T X^T X v_i}} \\
&= \frac{\sum_{n=1}^{N} (v_i)_n k(x_n, t)}{\sqrt{\sum_{n=1}^{N} (v_i)_n x_n^T \sum_{j=1}^{N} x_j (v_i)_j}} = \frac{\sum_{n=1}^{N} (v_i)_n k(x_n, t)}{\sqrt{\sum_{n=1}^{N} \sum_{j=1}^{N} (v_i)_n x_n^T x_j (v_i)_j}} \\
&= \frac{\sum_{n=1}^{N} (v_i)_n k(x_n, t)}{\sqrt{\sum_{n=1}^{N} \sum_{j=1}^{N} (v_i)_n k(x_n, x_j) (v_i)_j}}
\end{aligned}
\tag{36}
$$

In the above, (33) ~ (36) are the core equations for the formulation of kernel-based Regularized FDA (RFDA), where the kernel function plays an important role. It can be seen that,

TABLE I. TELEPHONE SESSION INFORMATION

Telephone Speech Data Set	Number of Speeches		Speech Duration
	Training	Testing	
Session 1	240	259	2s ~ 10s
Session 2	240	260	
Session 3	240	260	
UBM	300		20s ~ 2min

TABLE II. MICROPHONE INFORMATION

Set	Microphone Model	Number of Speeches		Duration
		Training	Testing	
M1	AKG C410B Head Mounted	240	260	2s ~ 5s
M2	AKH D80S Desktop	240	260	
M3	SONY ECM 66B Lapel	240	260	
M4	TARGET Lapel	240	260	
UBM	All the models	599		10s ~ 100s

during the calculation of the projected vector, only the kernel function is necessary. Through using a kernel function, we can implicitly map the input vector into another dimensional space, and then calculate the projected vector using the mapped vector. Interestingly, we do not need to know the mapping explicitly, as long as we have the kernel function, which is the inner product of the mapped vectors.

In this paper, we will consider two different kernel functions, one is the linear kernel function defined in (37) below, the other is the Gaussian kernel function defined in (38) below, where a and b are two column vectors, and ρ is a positive kernel parameter.

$$k_1(a,b) = a^T b \tag{37}$$

$$k_2(a,b) = e^{-\|a-b\|^2/\rho} = \varphi(a)^T \varphi(b) \tag{38}$$

While using kernel RFDA to project a given vector a, if we use the linear kernel function in (37), we are directly projecting the given vector onto another dimensional space; if we use the Gaussian kernel function in (38), we are first implicitly mapping it to a high dimensional vector $\varphi(a)$ which is of infinite dimension, and then projecting it onto a finite dimensional space. Gaussian kernel function can map the vector to an infinite dimensional space implicitly, although the inner product of the two vectors in the infinite dimensional space is finite [14].

III. SPEECH RECORDING DATASET

In our experiment, we use the Ahumada-25 speech corpus, which is a part of the AHUMADA Spanish speech corpus [15]. Ahumada-25 contains telephone conversations from 25 speakers recorded in 3 different sessions, and different sessions are separated for several days. Besides, Ahumada-25 also contains speech recordings recorded using different microphones. The contents include isolated numbers, sentences, specific texts and spontaneous speeches.

TABLE III. TELEPHONE SESSION IDENTIFICATION USING AVERAGED MFCC AND GSV (%)

Feature	Relevance Factor	Identification Accuracy
Averaged MFCC	n/a	65.08
Raw GSV	5	75.87
	10	76.38
	15	75.48

In terms of telephone session identification, for each session, we use 240 speeches coming from 12 speakers to form part of the training set and 260 speeches coming from the other 13 speakers to form part of the testing set (for session 1, only 259 speeches are used for testing, as one speech recording is corrupted). Another 300 speeches are used to construct the Universal Background Model (UBM), where all of the 25 speakers are involved, with each speaker contributing 12 speeches. Details about the training set, testing set and the UBM set are listed in Table I.

In terms of recording device identification, we obtain totally 4 different microphones, as shown in Table II. For each microphone, we use 240 speeches coming from 12 speakers to form part of the training set and 260 speeches coming from the other 13 speakers to form part of the testing set. Another 599 speeches are used to construct the UBM, where all of the 25 speakers are involved.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

A. Telephone Session Identification

In this part, we compare the performances of using different feature extraction methods, such as the averaged MFCC, the raw GSV, and the kernel RFDA-based projected GSV, to identify 3 different telephone sessions. We use a UBM with 32 mixture components to compute the GSV, and different relevance factors are used. As for the projected GSV, we also compare the performances of using different regularization parameters, different kernel functions and different kernel parameters. We use the linear SVM as the identifier, implemented using LIBSVM [16].

The identification results of using the averaged MFCC, and the GSV with different relevance factors, as the features, are shown in Table III. It can be seen that the GSV outperforms the averaged MFCC a lot. The reasons lie in that, first, the

TABLE IV. TELEPHONE SESSION IDENTIFICATION USING PROJECTED GSV WITH DIFFERENT PARAMETERS (%)

Relevance Factor (γ)	Regularization Parameter (α)	Kernel			
		Linear	Gaussian		
			Kernel Parameter (ρ)		
			10000	20000	50000
5	10	74.20	68.42	68.68	70.09
	20	75.74	73.04	73.94	74.45
	50	78.43	75.22	75.99	77.66
	100	78.56	**76.12**	**76.25**	78.31
	200	79.33	75.35	**76.25**	**78.43**
	500	**79.46**	74.97	75.99	78.05
10	10	74.20	68.16	68.55	69.70
	20	76.89	73.04	74.33	75.87
	50	79.08	76.38	78.69	80.87
	100	79.33	76.77	**79.33**	83.31
	200	**79.72**	**76.89**	79.20	**83.83**
	500	79.33	76.25	78.18	82.93
15	10	73.43	70.86	70.99	70.60
	20	76.64	73.17	73.81	75.48
	50	78.95	75.61	76.89	78.56
	100	**79.59**	76.38	78.69	80.49
	200	**79.59**	**76.77**	**79.08**	**81.64**
	500	**79.59**	76.12	78.95	**81.64**

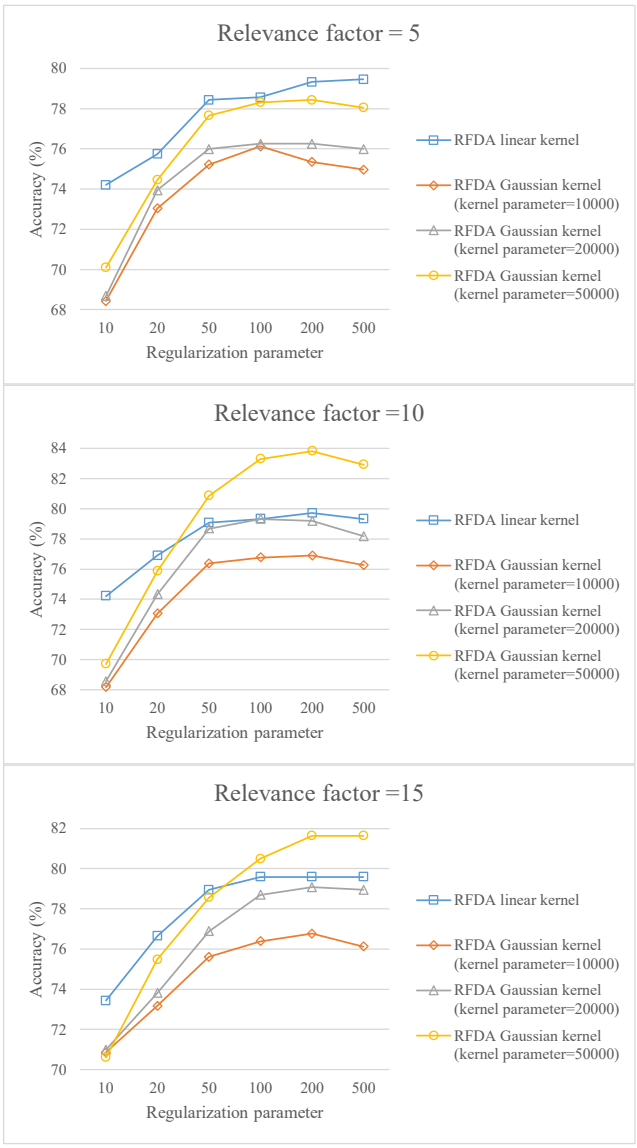

Fig.2. Telephone session identification using projected GSV with different parameters.

averaged MFCC does not make good use of all the frame-level feature vectors while the GSV does; second, the GSV obtains extra information from the UBM while the averaged MFCC only relies on the training data.

The performance of the kernel RFDA-based projected GSV is evaluated under different regularization parameters, different kernel functions and different kernel parameters. The identification accuracy results are listed in Table IV and plotted in Fig. 2. It can be seen that, generally the performance of the projected GSV with linear kernel and Gaussian kernel can be improved by increasing the regularization parameter α. In particular, on using Gaussian kernel, the performance of the projected GSV can be improved by increasing the kernel parameter ρ. The highest accuracy achieved by the projected GSV with linear kernel is 79.72%, and the highest accuracy achieved by the projected GSV with Gaussian kernel is 83.83%, and both are better than the raw GSV (76.38%). With suitable kernel parameters, we see that Gaussian kernel RFDA can work better than linear kernel RFDA. While using Gaussian kernel, we first implicitly map the input feature vector onto a high dimensional space and then project the mapped feature vector from the higher dimensional space to a lower dimensional space. The mapping from a lower dimensional space to a higher dimensional space can better reveal the relationship between different dimensions of the input feature vector.

TABLE V. MICROPHONE IDENTIFICATION USING RAW GSV (%)

Feature	Relevance Factor	Identification Accuracy
Raw GSV	5	78.17
	10	78.65
	15	79.62

B. *Recording Device Identification*

In this part, we compare the performances of using the raw GSV and the kernel RFDA-based projected GSV to identify 4 different microphones. A UBM with 32 mixture components is used to compute the GSV, and different relevance factors, different regularization parameters and different kernel parameters are evaluated. The identification results of using the raw GSV with different relevance factors are shown in Table V. The identification results of using the projected GSV with different relevance factors, different kernel parameters and different regularization parameters are shown in Table VI and plotted in Fig. 3.

By comparing the results in Table V and Table VI, it can be seen that the kernel RFDA-based projected GSV can outperform the raw GSV. In particular, with suitable kernel parameters, Gaussian kernel RFDA can work better than linear kernel RFDA, which is similar to the results we observe in telephone session identification. The highest accuracy achieved by the projected GSV with linear kernel is 82.60%, and the highest accuracy achieved by the projected GSV with

TABLE VI. MICROPHONE IDENTIFICATION USING PROJECTED GSV WITH DIFFERENT PARAMETERS (%)

Relevance Factor (γ)	Regularization Parameter (α)	Kernel			
		Linear	Gaussian		
			Kernel Parameter (ρ)		
			10000	20000	50000
5	10	78.37	87.21	84.90	83.85
	20	79.42	88.75	88.46	**87.50**
	50	79.13	**88.94**	**88.56**	86.83
	100	**81.63**	88.65	87.69	85.77
	200	81.06	88.37	87.21	84.81
	500	81.25	88.37	87.02	84.42
10	10	78.27	83.75	83.65	83.85
	20	79.52	**87.79**	**87.69**	**88.08**
	50	81.92	87.69	86.92	86.44
	100	**82.31**	87.12	87.02	86.35
	200	79.81	87.12	86.73	86.15
	500	81.35	86.92	86.83	86.25
15	10	78.17	82.60	82.21	81.83
	20	79.71	84.13	84.33	84.81
	50	82.40	**85.67**	**85.48**	**85.19**
	100	**82.60**	85.00	84.81	85.10
	200	82.31	84.52	84.71	84.90
	500	81.92	84.42	84.23	84.90

Fig.3. Microphone identification using projected GSV with different parameters.

Gaussian kernel is 88.94%, while the highest accuracy achieved by the raw GSV is only 79.62%. We also observe that, on using Gaussian kernel, the performance of the projected GSV can be improved by decreasing the kernel parameter ρ. Although this tendency is different from what we observe from telephone session identification, yet it is reasonable, as we are dealing with different tasks.

V. Conclusion

In this paper, we propose the Regularized Fisher Discriminant Analysis (RFDA), as an alternative to the traditional Fisher Discriminant Analysis (FDA). We also develop the kernel version of RFDA, which generalizes RFDA by introducing different kernel functions. we try to identify different telephone sessions based on some training telephone speeches recorded in different sessions. We borrow the idea of Gaussian Supervector (GSV) from the studies on speaker recognition, and show that GSV also works well for telephone session identification tasks. Then we apply the RFDA-based projection on the GSV for telephone session identification, and find that the projected GSV gives better performance. Besides, we also apply the RFDA-based projection on the GSV used in recording device identification, and find that the projected GSV also exhibits improvement. These results show the efficiency and potential of RFDA as a feature de-noising and purifying technique for improve the performance of the GSV serving different identification purposes.

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, Jan. 2012.

[2] D. P. N. Rodriguez, J. A. Apolinario, and L. W. P. Biscainho, "Audio authenticity: detecting ENF discontinuity with high precision phase analysis," *IEEE Trans. on Information Forensics and Security*, vol. 5, no. 3, pp. 534-543, 2010.

[3] D. P. N. Rodriguez, J. A. Apolinario, and L. W. P. Biscainho, "Audio authenticity based on the discontinuity of ENF higher harmonics," in *Proc. 21st European Signal Processing Conference*, 2013, pp. 1-5.

[4] P. A. A. Esquef, J. A. Apolinario, and L. W. P. Biscainho, "Edit detection in speech recordings via instantaneous electric network frequency variations," *IEEE Trans. on Information Forensics and Security*, vol. 9, no. 12, pp. 2314-2326, 2014.

[5] R. Garg, A. L. Varna, and M. Wu, "Modeling and analysis of electric network frequency signal for timestamp verification," in *Proc. IEEE Int. Workshop on Information Forensics and Security*, 2012, pp. 67-72.

[6] W. H. Chuang, R. Garg, and M. Wu, "How secure are power network signature based time stamps," in *Proc. ACM Conf. on Computer and Communications Security*, 2012, pp. 428-438.

[7] D. Garcia-Romero and C. Y. Espy-Wilson, "Automatic acquisition device identification from speech recordings," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP'10)*, Dallas, TX, USA, 2010, pp. 1806-1809.

[8] 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.

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

[10] D. A. Reynolds, "Gaussian mixture models," in *Encyclopedia of Biometrics*, Springer US, 2009, pp. 659-663.

[11] C. M. Bishop, "Linear models for classification," in *Pattern Recognition and Machine Learning*, Springer, 2006, ch. 4, pp. 179-224.

[12] K. Fukunaga, "Feature Extraction and Linear Mapping for Classification," in *Introduction to Statistical Pattern Recognition*, 2nd Ed., San Diego, California: Academic Press, 1990, ch. 10, pp. 441-507.

[13] S. Mika *et al.*, "Fisher discriminant analysis with kernels," in *Proc. IEEE Neural Networks for Signal Processing Workshop*, 1999, pp. 41-48.

[14] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, vol. 2, no. 2, pp. 121-167, 1998.

[15] J. O. Garcia, J. G. Rodriguez, and V. M. Aguiar, "AHUMADA: a large speech corpus in Spanish for speaker characterization and identification," *Speech Communication*, vol. 31, no. 2, pp. 255-264, 2000.

[16] 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.