Dimensionality Reduction Techniques and SVM Algorithms for Large Population Speaker Identification

P. R. K. Rao
ECE Dept., Usha Rama College of Engg. and Technology, India
Email: prkr74@gmail.com

Y. S. Rao
Instrument Technology Dept., AU College of Engg., Andhra University, India
Email: srinniwasarao@gmail.com

***Abstract*—The ever-increasing size of datasets in speaker recognition systems is the primary reason why challenges arise with regard to accuracy and computational complexity. Inadequate speaker-specific information on vocal tracts may lead to poor modeling and can adversely affect the performance under large-scale data conditions. In this work, we have developed a speaker recognition system, based on the excitation source information by blending pitch and pitch strength vectors. We investigate various approaches to improve the performance from two directions. First, we investigate various dimensionality reduction techniques during the feature extraction phase such as Multi-linear Principle Component Analysis (MPCA), Principle Factor Analysis (PFA), and Maximum Likelihood Factor Analysis (MLFA). We have evaluated the performance of different large-scale Support Vector Machine (SVM) algorithms as a function of training time to attain convergence. Combinations of the proposed dimensionality reduction methods and SVM algorithms have successfully produced an effective recognition system. We have demonstrated the performances of our approaches by conducting experiments on standard large-scale data bases, and then, compared these with the existing state-of-the-art recognition systems. The experimental results have shown that these approaches significantly improve the performance under large-scale data conditions in comparison with the conventional procedures.**

***Index Terms*—maximum likelihood factor analysis, large scale-SVM, pitched, dimensionality reduction, support vector machine**

I. Introduction

Speaker recognition is the process of authenticating persons with the speaker-specific information from speech signal. Based on the task objectives, speaker recognition is classified into verification and identification. Verification deals with the process of evaluating the identity claim of a speaker. Whereas in identification, a machine determines the identity of a specific person using the existing reference models. Speaker recognition modes are further divided into text-dependent and text-independent systems [1]. A text-dependent system makes assumptions regarding the parameters of speaker's vocal tract. But, in text-independent modality, the same text is employed for enrolment and verification stages. The present work is focused on text-independent speaker recognition.

Real-Time implementation of speaker recognition technology involves multiple application-specific trade-offs such as cost, performance, robustness, enrolment procedures, training time, adaptability, response time, etc. [2]. Since the database size for the real-world recognition tasks is ever-increasing, large population speaker recognition systems pose challenges such as large training time, vast memory requirements and poor response time [3]. Though accuracy is always the first consideration, efficient recognition and adaptability are also the significant aspects in many real-world speaker recognition systems under large-scale data conditions. This motivates us to investigate new methods at various stages involved in typical speaker recognition systems.

Generic speaker recognition process mainly consists of two phases: training (also known as enrolment) and identification [4]. In the enrolment stage, speaker-specific information, from speech signal is extracted in chronological mode to develop speaker models. A cluster of such models in turn establishes the speaker database for the later test phase. During the identification phase, an anonymous speaker model is compared with the existing database and then the results are expedited. In fact, both the phases comprise of feature extraction that transforms the raw speech signal into a compact and effective representation which is comparatively more stable and discriminating than the original signal [5]. A typical speaker recognition system consists of the following phases: feature extraction, dimensionality reduction and classification [3]. In this work, in view of development of an effective large-scale recognition system, the state-of-art methods have been proposed and experimental investigations have been conducted for each stage.

Manuscript received May 23, 2014; revised November 30, 2014.

doi: 10.12720/ijsps.4.2.86-96

Speaker characteristics in a speech signal are differentiated by the dimensions of vocal tract, vocal excitation and learning habits of speakers [6]. Physiological structure of the speech production system is reflected by vocal tract characteristics and is relatively more robust and less prone to mimicry [7]. Therefore, recognition systems mostly use vocal tract information related features such as Mel Frequency Cepstral Coefficients (MFCC), Linear Predictive Cepstral Coding (LPCC) and other non-conventional features [8]. These methods have utilized the information from linear predictive (LP) residual signal. Two approaches known as temporal and frequential representations are examined. The former consists of an auto-regressive (AR) modeling of a signal followed by a cepstral transformation similar way to LPC-LPCC transformation [9]. However, under noisy environments and large data sets, performance of vocal tract information related features degrades severely [10]. Hence, it is necessary to derive robust features for the speaker recognition task. From the evidences of earlier studies [11]-[13], it has been identified that the features extracted from the excitation source are less prone to environmental noise and need small amount of data. Since the characteristics of the excitation source like pitch and pitch strength exhibit both the physiological and behavioral aspects of the speaker, it is possible to achieve good accuracy and lessen computational complexity. This work mainly explores feature extraction from the excitation source by accumulating speech characteristics such as pitch and pitch strength.

The next stage in feature extraction is dimensionality reduction in which a high-dimensional space is transformed into a space of fewer dimensions. Dimension reduction is employed not only for the benefit of computational efficiency, but also to improve the accuracy of recognition [14]. In this work, much effort has been put to reduce the feature matrix by using different dimensionality reduction techniques such as Multi-linear Principal Component Analysis (MPCA) [15], Principal Factor Analysis (PFA) [16] and Maximum Likelihood Factor Analysis (MLFA) [17]. The proposed methods are not strictly novel; however, the proposers have conducted several investigations and performed the comparative analysis by varying the number of users from standard databases. Since the methods have been successful in large-scale classification, original contributions are carried out in this work in the interest of large-scale speaker recognition that employs pitch and pitch strength as features.

Finally, in the classification stage, verification is tested by Support Vector Machine (SVM) which is a successful discriminator and a very effective method in the field of speaker recognition [18]. Specifically, SVM is employed in the context of statistical learning and thereby attributed to minimize the risk function. However, the main constraints of the SVMs are its computational complexity and relatively poor performance for the large-scale classification [19]. In view of this, investigations for optimization of large-scale SVM algorithms have been carried out in this work. Four large-scale SVM algorithms namely, Pegasos [20], BMRM [21], FOLOS [22], and CVM [23], have been demonstrated and experiments have been conducted with combinations of different dimensionality reduction techniques. Though the proposed algorithms are not strictly novel, the approaches are very efficient for large datasets in classification tasks such as pattern and speech recognitions. This work demonstrates the original contributions of interest in the field of large-scale speaker recognition.

II. Feature Extraction

Pitch is the quality of subjective sensation where all the tones perceived by a listener are assigned to relative positions on the musical scale and the frequency of vibrations is observed. Though sounds vary in pitch, some of the sounds have strong pitch sensation (e.g., vowels) whereas some do not (e.g., consonants). Accordingly, sounds are classified into pitched and non-pitched. Pitch information is valuable in speech applications such as music transcription, speech coding and query by humming [24]. As pitch differentiates the user's speech greatly, pitch and its strength are suggested to extract as features for the proposed speaker recognition system.

Let $A_{ij}(t)$; $0 \le i \le N_u$-1 and $0 \le j \le N_s^{(i)}$-1, is the speech signal obtained from different users, where N_u is the number of users and $N_s^{(i)}$ is the number of samples taken from the i^{th} user, provided, $N_s^{(0)}=N_s^{(1)}=N_s^{(2)}=\ldots=N_s^{N_u-1}$. The sample speech signals from different users are subjected to feature extraction that extracts pitch and its strength as the feature for each sample. In the feature extraction stage, the continuous speech signals of all users are converted to discrete speech signals as $A_{ij}(n)$; $n=1,2\ldots N_l$. The speech samples of every user are processed in sequence for the feature matrix representation. The primary speech sample of the first user is subjected to the process of extracting pitch and pitch strength. The process is initiated by selecting windows of signal instants with different sizes. N_c classes of windows are generated in which each class has its own window size [25]. Afterward, the obtained windows of signal sequences are processed by the following Hanning window function,

$$W_{kl}(m) = 0.5\left(1-\left(cos\left(\frac{2\pi w_{kl}}{|w_k|-1}\right)\right)\right) \quad (1)$$

where $0 \le m \le |w_k|$-1, $0 \le k \le N_c$-1, $0 \le 1 \le N_w^{(k)}$-1, N_w is the number of windows that belongs to k^{th} class, $w_{kl}(m)$ is the m^{th} instant of time in l^{th} window of k^{th} class and the parameters $W_{kl}(m)$ are the Hanning window coefficients derived from equation (1). The size of window that belongs to each class is calculated as $|w_{kl}| = 2^{k+2}$. A pitch vector $[P_{kl}]$, $P_{kl}(m) \in \{0,1\}$ is generated with the same size of $W_{kl}(m)$ in which each element of vector $P_{kl}(m)$ is arbitrarily generated from {0,1} (i.e. $P_{kl}(m) \in \{0, 1\}$. For every window element, centroids for pitched and non-pitched classes are determined as,

$$G_{P_{kl}}=\frac{1}{\sum_{m=0}^{|W_{kl}|-1} W_{kl}(m)P_{kl}(m)}\sum_{m=0}^{|W_{kl}|-1} a_{00}\,W_{kl}(m)P_{kl}(m) \quad (2)$$

$$G_{\mathrm{UP}_{kl}}=\frac{1}{\sum_{m=0}^{|W_{kl}|-1} W_{kl}(m)\overline{P_{kl}(m)}}\sum_{m=0}^{|W_{kl}|-1} a_{00}\,W_{kl}(m)\overline{P_{kl}(m)} \quad (3)$$

Using (2) and (3), centroids can be determined for the classes pitched and non-pitched respectively based on pitch vector, window coefficients [26] and speech signal. In (2) and (3), $a_{00}W_{kl}(m)$ is the magnitude of speech signal at a specific time interval indicated by the window element $W_{kl}(m)$. Once the centroids are calculated, then the time instant at which the pitch is present $\{P_I\}$ and the strength of pitch $\{P_S\}$ are determined as $\{P_I\}=\{P_I'\}-\varphi$ and $\{P_S\}=\{P_S'\}-\varphi$ respectively, where, the set $\{P_I'\}$ and $\{P_S'\}$ are calculated as,

$$\{P_I'\} \ll \begin{cases} n; \frac{2(a_{00}(n)-G_{UP}^{max})}{G_P^{max}-G_{UP}^{max}} > 1 \\ \varphi\ ; \qquad otherwise \end{cases} \quad (4)$$

$$\{P_S'\} \ll \begin{cases} a_{00}(n); if\ n \in \{P_I\} \\ \varphi\ ;\ otherwise \end{cases} \quad (5)$$

In (4), pair of centroids (G_P^{max} and G_{UP}^{max}) exhibits the maximum distance among all centroid pairs. The centroids G_P^{max} and G_{UP}^{max} are determined by firstly calculating the distance between each centroid pair as $d_{kl} = G_{P_{kl}} - G_{UP_{kl}}$. Then, the parameters $G_{P_{kl}}$ and $G_{UP_{kl}}$ contribute to the maximum d_{kl} and are converted into G_P^{max} and G_{UP}^{max} respectively. The obtained feature set of $\{P_I\}$ and $\{P_S\}$ is stored as the first sample for the first user. The process is repeated for all speech samples of the same user and the obtained feature set is stored in feature matrix P_{ab}. Where $0 \le a \le P_{max}$, $0 \le b \le N_T - 1$, each column is composed of elements of feature set from each speech sample for a single user [27]. Thus the obtained feature matrix is of size $P_{max} * \mathrm{N_T}$, where P_{max} is the feature set for a sample which has maximum elements and $N_T = N_u.N_s$. All the remaining feature sets are filled up with zeros to attain the size of P_{max}. Afterword, the obtained feature matrix of higher dimension is processed for the dimensionality reduction.

III. Dimension Reduction

The dimension of the feature matrix that is obtained from the previous section can be further minimized by the following Dimensionality Reduction techniques:

- Multi-Linear Principle Component Analysis (MPCA)
- Principal Factor Analysis (PFA)
- Maximum Likelihood Factor Analysis (MLFA)

The theoretical background of each technique is described in the following subsections.

A. Multi-Linear Principle Component Analysis (MPCA)

The proposed technique is a multilinear algorithm [28], which can perform dimensionality reduction for all tensor modes by seeking the bases in each mode. More precisely, these tensor bases in turn allow the projected tensors to capture the variation present in the original tensors. In order to perform the dimensionality reduction, a selected number of feature values are extracted from the feature matrix [15]. This can be accomplished by extracting N_{sel} elements from every column of the feature matrix, where, N_{sel} has to be a perfect square integer. Hence, N_T column vectors are obtained of size $N_{sel} \times 1$ each. Every column vector is denoted as $C_a^{(b)}$, where $0 \le a \le N_{sel} - 1$ and the corresponding matrix M is determined as,

$$M_{xy}^{(b)} = C_z^{(b)}; 0 \le x \le \sqrt{N_{sel}} - 1, 0 \le y\sqrt{N_{sel}} - 1 \quad (6)$$

$$z = x\sqrt{N_{sel}} + \mathrm{Y} \quad (7)$$

In the above modified matrix, every b^{th} column vector is subjected to dimensionality reduction by using MPCA.

During the process of dimensionality reduction, the distance matrix D can be determined for every b^{th} matrix as,

$$D^{(b)} = M^{(b)} - \mu \quad (8)$$

where

$$\mu_{xy} = \frac{1}{N_T}\sum_{b=0}^{N_T-1} M_{xy}^{(b)} \quad (9)$$

In (8), μ is the mean matrix, determined for $M^{(b)}$ and it is used for calculating the distance matrix. In order to obtain the projection matrix Ψ, representations $T_1^{(b)}$ and $T_2^{(b)}$ (here the mode is 2) are applied to the distance matrix $D^{(b)}$:

$$\Psi = \sum_{b=0}^{N_T-1} T^{(b)}\,(T^{(b)})^T \quad (10)$$

The generalized form (Ψ_1 and Ψ_2) for calculation is given in (10), and it is used to determine both the tensor representations $T_1^{(b)}$ and $T_2^{(b)}$. The projection matrix is subjected for the generalized eigenvector problems. Therefore, the eigenvectors (V_1 and V_2) and eigenvalues (λ_1 and λ_2) are determined for the projection matrices Ψ_1 and Ψ_2 respectively. Furthermore, the determined eigenvalues are sorted in descending order and the rows of the eigenvectors are arranged by using the index of sorted eigenvalues. The arranged eigenvector is transposed for further representations and then the modified eigenvectors V_1' and V_2' are computed. In order to sort eigenvalues, cumulatively distributed eigenvalues are generally calculated as,

$$\lambda_x' = \frac{\lambda_x^{cdf}}{\sum_{x=0}^{|\lambda|-1} \lambda_x^{sort}} \quad (11)$$

where,

$$\lambda_x^{cdf} = \begin{cases} \lambda_x^{sort} + \lambda_{x-1}^{cdf}; & if\ x > 0 \\ \lambda_x^{sort}; & otherwise \end{cases} \quad (12)$$

In (11) and (12), λ_x^{cdf} is the set of cumulatively distributed eigenvalues and λ_x^{sort} is the set of sorted eigenvalues. From the obtained λ_x', the new dimension λ_T is calculated by using a dimensional threshold D_{TH}. This can be accomplished by identifying the indices of all eigenvalues that need to satisfy the condition $\lambda_x \ge D_{TH}$. Afterword, the dimensionality reduced eigenvectors V_1''

and V_2'' are determined from V_1' and V_2' respectively by extracting the first λ_T rows. Furthermore, the reduced eigenvectors V_1'' and V_2'' and the tensor matrices $T_1^{(b)}$ and $T_2^{(b)}$ are computed again. This process is already done for the projection matrices as it is repeated for the tensor matrices also. As a result, new vectors $\lambda_{x_1}^{new}$ and $\lambda_{x_2}^{new}$, V_1^{new} and V_2^{new} are obtained. The weights of these tensor eigenvalues are determined as,

$$W_x = \sqrt{\lambda_{x_1}^{new}\lambda_{x_2}^{new}} \tag{13}$$

Therefore, by using the MPCA projections, the dimensionality reduced matrix $M^{b'}$ is obtained, which is of size $N^{red} \times N_T$, where N^{red} is determined as $N^{red} = \lambda_{T1}\lambda_{T2}$.

B. Principle Factor Analysis

Factor analysis (FA) is a linear method which assumes that the measured variables depend on some unknown and often immeasurable common factors. Variables of various test scores for individuals can be taken as factors. Where, these test scores are assumed to be related to a common intelligence factor [29]. The main aim of the factor analysis is to uncover such relations. The q-dimensional random vector x_{q*1} with the covariance matrix Σ satisfies the k-factor model if,

$$x = \Lambda f + u \tag{14}$$

where Λ_{p*k} is a matrix of constants and u_{q*1} and f_{k*1} are the specific factors and random common factors respectively. However, the common factors are standardized to have variance of value one and all the factors to be uncorrelated as,

$$E(f) = 0, \quad Var(f) = I \tag{15}$$

$$E(u) = 0, \quad Cov(u_i, u_j) = 0 \quad for\ i \neq j \tag{16}$$

$$Cov(f, u) = 0 \tag{17}$$

The diagonal covariance matrix *u* is determined from the above assumptions as,

$$Cov(u) = \psi = diag(\psi_{11}, \ldots\ldots\ldots \psi_{qq}) \tag{18}$$

The k-factor model is defined by decomposing the data covariance matrix into,

$$\Sigma = \Lambda\Lambda^T + \psi \tag{19}$$

Since

$$x_i = \sum_{j=1}^{k} \lambda_{ij} f_j + u_i \qquad i = 1, \ldots\ldots, q \tag{20}$$

The variance of $\boldsymbol{x_i}$ may be decomposed as,

$$\sigma_{ii} = \lambda_{ij}^2 + \psi_{ii} \tag{21}$$

where $h_i^2 = \sum_{j=1}^{k} \lambda_{ij}^2$ is called the communality and represents the variance of x_i, which is unique to all variables. ψ_{ii} is called the unique variance and it contributes to the variability of x_i due to its specific u_i part. The magnitude of the dependence of x_i on the common factor f_j is measured by the term λ_{ij}^2 for the given factor f_j. If several x_i variables have high loadings λ_{ij}, then variables measure the same unobservable quantity and hence it is redundant.

The factor model also holds for orthogonal rotations of the factors but does not depend on the scale of the variables. By considering an orthogonal matrix *G* and the given model in (14), the new model can be defined as,

$$x = (\Lambda G)(G^T f) + u \tag{22}$$

It also holds the new factors $G^T f$ and the corresponding loadings ΛG. Therefore, to satisfy some additional constraints the factors are generally rotated.

$$\Lambda^T D^{-1} \Lambda \text{ is diagonal} \tag{23}$$

$$\text{And} \quad D = diag(\sigma_{11}, \ldots\ldots, \sigma_{qq}) \tag{24}$$

where, the diagonal elements are arranged in decreasing order. The varimax method is used to rotate the factors to obtain a reduced representation with few significantly non-zero loadings (i.e. sparse matrix **Λ**). In many cases, a k-order factor model in (19) provides a better explanation for the data instead of the alternative full covariance model *Var(x) = Σ*. Therefore, it is possible to derive the parameter estimates $\hat{\lambda}$ and $\hat{\psi}$ in such cases.

Let *S*, *R* and $\bar{x}$ denote the correlation matrix, covariance matrix and sample mean respectively for the observed data matrix X. Accordingly the equations are,

$$\widehat{\sigma_{ii}} = s_{ii} \quad i = 1, \ldots\ldots, q \tag{25}$$

$$\hat{\Sigma} = \overline{\Lambda}\,\overline{\Lambda^T} + \hat{\psi} \tag{26}$$

$$\widehat{\sigma_{ii}} = \sum_{j=1}^{k} \hat{\lambda}_{ij}^2 + \hat{\psi}_{ii} \tag{27}$$

When the data [30] is standardized, its covariance matrix is equal to the correlation matrix. To obtain the estimates $\widehat{\lambda}$ and $\hat{\psi}$ for the standardized variables, the first estimation becomes $\hat{h}_i^2$ for *i* = 1,....., *q*. Common estimates $\hat{h}_i^2$ include the square of the multiple correlation coefficients of the i_{th} variable, and the largest correlation coefficient between the i^{th} variable and one of the other variables. The reduced correlation matrix is derived from *R*-$\hat{\psi}$, where the diagonal elements of 1 in *R* are replaced by the elements.

Decomposing the reduced correlation matrix in terms of the eigenvalues $a_1 \geq \ldots \geq a_q$ and orthonormal eigenvectors $\gamma_{(1)}, \ldots, \gamma_{(q)}$ as,

$$R - \widehat{\psi} = \sum_{i=1}^{k} a(i)\gamma(i)\gamma(i)^T \tag{28}$$

The above equation estimates the $\boldsymbol{i_{th}}$ column of Λ by considering the first $\boldsymbol{k}$ eigen values as positive,

$$\lambda_{(i)} = a_{(i)}^{1/2} \lambda_{(i)}, \quad i = 1, \ldots, q \tag{29}$$

Equivalently,

$$\Lambda = \Gamma_1 A_1^{1/2} \tag{30}$$

where, $\Gamma_1 = (\gamma_{(1)}, \ldots, \gamma_{(q)})$, and $A_1 = diag(a_1, \ldots a_k)$. The constraint (24) holds when eigenvectors are orthogonal. Finally, the specific variance estimates are updated as,

$$\hat{\psi}_{ii} = 1 - \sum_{j=1}^{k} \hat{\lambda}_{ij}^2 \qquad i = 1, \ldots q \tag{31}$$

The k-factor model is permissible if all the *p* terms are non-negative.

Generally, the number of factors may be determined by taking into account eigenvalues a_i from the reduced correlation matrix, and choosing 'g' as the index with a sharp drop in the eigenvalue magnitudes.

C. Maximum Likelihood Factor Analysis

If it is assumed that the factors *f* and *u* are distributed as multivariate normal variables, then the parameters for the model can be estimated by maximizing the likelihood [17]. The k-factor model describes the data more accurately than the unconstrained variance model. The log likelihood function can be written as,

$$l = -\frac{1}{2}\, n \log|2\pi\Sigma| - \frac{1}{2}\, n\, tr\, \Sigma^{-1}S \quad (32)$$

The main aim is to maximize it with respect to the parameters Λ and ψ, subject to the constraint on the parameter Λ in the equation (24).

The covariance matrix under the factor model is defined as,

$$\Sigma = \Lambda\Lambda^T \quad (33)$$

In order to optimize, the following assumptions are noted,

$$F(\Lambda, \psi) = F(\Lambda, \psi, S) = tr\Sigma^{-1}S - \log|\Sigma^{-1}S| - q \quad (34)$$

It is a linear function of the log-likelihood *l*, with a maximum in *l* corresponding to a minimum in *F*. It is also described in terms of the arithmetic mean 'a' and the geometric mean 'g' of the eigenvalues of $\Sigma^{-1}S$ as,

$$F = q(a - \log g - 1) \quad (35)$$

$F(\Lambda, \psi)$ can be minimized by the following two stages:

Stage-1: The minimization over Λ for a fixed ψ has an analytical solution.

Stage-2: The minimization over ψ is carried out numerically.

The dimensionality reduced matrices obtained from the above reduction techniques are subjected to one of the two processes of SVM-based training and recognition.

IV. Support Vector Machines

Support Vector Machine is a classifier which can group the patterns according to the maximum margin separation principle [31]. Since the real-world classification tasks evaluate computations on large data-sets, the allocation of main memory becomes infeasible even for modern computers [32]. The main constraint of SVM comes from its requirement in terms of memory and training time. But the previous section describes that a large number of high dimensional patterns are needed to train SVM. Therefore, approaches for SVM algorithms are drawn with the emphasis on linear scaling in memory occupation and the training time complexity with the user count.

SVM discriminates the reference models into two classes according to the maximum margin separation criterion. Regularized Risk Minimization factor estimates the hyperplane that describes the boundary for the two given classes of patterns [33].

$$\min_{w}[\tfrac{1}{2}\|w\|^2 + C.\sum_{i=1}^{n} l(w, x_i, y_i)] \quad (36)$$

where, *d*-dimensional training pattern (x_i) denotes $x_i \in \mathbb{R}^d$ with the corresponding label $y_i \in \{-1, +1\}$. The objective function is denoted by the sum of two terms. The first term represents a regularized contribution that can be estimated by square rule of hyperplane *w*. In the second term, the empirical risk is evaluated on the training dataset and it is estimated by the parameter *C*. The SVM standard loss Hinge function ($l\mathrm{L}_1$) evaluates the maximum-soft margin as,

$$\mathrm{lL1} = max(0, 1 - y_i\, w^T x_i) \quad (37)$$

From (36) and (37), the primal SVM solver conceptualization is,

$$w^* = \arg\min_{w} \tfrac{1}{2} w^T w + C.\sum_{i=1}^{n} max(\,0, 1 - y_i\, w^T x_i) \quad (38)$$

where, *n* denotes the number of patterns. The classification score can be evaluated by the set of Support Vectors from the separation plane.

A. Large-Scale SVM Algorithms

Numerous large-scale algorithms have been proposed by the researchers to handle large-scale SVM optimization problems. These algorithms mainly focus on the complexity involved to speed up the classification task for the standard databases. Performance evaluation procedures are formulated in terms of memory and training time requirements to reach convergence [34]. In order to optimize, large-scale SVM algorithms rely on the following two approaches.

1) Dual solvers

Dual solver optimization methods evaluate many dot-product implementations from the classical nonlinear SVM algorithms. Since the input pitch based feature vectors represent pairs, the training trails of SVM grows linearly with $O(n^2)$. The feature component representation limits to $O(d^2)$ and the data set size approximates to $O(n^2d^2)$ iterations. Complete buffered feature matrix is impracticable even for the small feature set because it would occupy $O(n^4)$ memory space. Therefore, the application of dual solvers for the speaker recognition is inefficient.

2) Primal solvers

The Primal solvers estimate the time complexity as $O(n^4d)$ because of efficient evaluation of the loss function and its gradient. The following sections describe the four primal solvers and provide the steps to derive their corresponding solutions.

Pegosis: It is a SVM optimization algorithm and based on gradient descent [35] in the primal form. Since the run-time is independent of sample size, the solver varies between Stochastic Gradient Descent (SGD) and Subgradient Descent [20]. The updated rule for the loss function L_1 is defined as,

$$w_{t+1} = w_t - \eta_t[w_t + C\sum_{i=1}^{n} \nabla_w\, max(0, 1 - y_i w^T x_i)\,|w_t \quad (39)$$

where η_t is the learning rate and w_t is the normal vector at iteration t.

Since the estimation of learning rate values is critical for fast convergence, the subgradient projection on loss function L_1 can be computed by differentiability as,

$$\nabla_w max(0,1 - y_i w^T x_i) = - y_i \quad \text{if } y_i w^T x_i \leq 1$$
$$= 0 \quad \text{otherwise} \tag{40}$$

The SGD estimates the gradient evaluation step by computing the subgradient of the objective function on a set of patterns.

$$w_{t+1} = w_t - \eta_t [w_t + nC\nabla_w l(w, x_{i_t}, y_{i_t}) w_t] \tag{41}$$

where, the index i_t is chosen randomly for all iterations. Thus the SVM loss function L_1 is computed by combining SGD with the projection step for fast convergence. At each iteration, a set of training patterns A_t is randomly chosen. Then the subgradient is estimated for the objective function as,

$$\nabla_t = w_t - \frac{nC}{|A_t|} \sum_{\substack{i | x_i \in A_t \\ y_i w^t x_i < 1}} y_i x_i \tag{42}$$

The hyperplane can be updated as,

$$w_{t+1/2} = w_t - \eta_t \nabla_t \tag{43}$$

The optimal SVM solution [20] is constrained by $\|w\| \leq \sqrt{nC}$. Thus, the solution is estimated into a ball of radius $\sqrt{nC}$ by projecting $w_{t+\frac{1}{2}}$ as,

$$s_t = \min \left\{ 1, \frac{\sqrt{nC}}{\left\| w_{t+\frac{1}{2}} \right\|} \right\} \tag{44}$$

$$w_{t+\frac{1}{2}} = \left| s_t w_{t+\frac{1}{2}} \right| \tag{45}$$

For the efficient optimization ε, the fast-decaying learning rate is combined with the current projection step and bounded to $o(\frac{1}{\varepsilon})$ average number of iterations.

Bundle methods: Bundle methods optimize the general risk minimization and hence BMRM is such a solver [21]. Based on the optimization problem, this method approximates the convex function by means of a set of sub-gradients and minimizes the simpler optimization problem. In order to obtain the optimal solution for the error, a small and incremental subset of constraints is built. The BMRM provides us an easy and extensible framework for support vector machines in the risk regularization. Specifically, at each iteration, an incremental set of approximate solutions $\{w_0, w_1, w_2, ...\}$ is formulated. A set of hyperplanes which are tangent to the objective function-are described as,

$$f_t(w) = l_{emp}(w_t) + \nabla l_{emp}(w_t).(w - w_t) \tag{46}$$

where, l_{emp} is the empirical loss function and is defined as $\sum_i^n l(w, x_i y_i)$. A new point is selected from the approximated function at the given iteration as,

$$w_{t+1} = \underset{w}{argmin} \left[\frac{1}{2} \|w\|^2 + C.max \left[0, \max_{t' \leq t+1} f_t'(w) \right] \right] \tag{47}$$

According to [21], it is estimated that the problem is similar to the dual quadratic problem.

$$\min_{\beta} D_i(\beta) = \frac{C}{2} \beta^T A^T \mathrm{A} \beta - \beta^T \mathrm{b} \tag{48}$$

Subject to $\beta \geq 0$, $e^T \beta \leq 1$, where A is the matrix $[a_1 \quad a_2 \ldots\ldots \quad a_i]$ of gradients $a_{t+1} = \nabla l_{emp}(w_t)$ and the vector b is denoted as $[b_1 \quad b_2 \ldots\ldots \quad b_i]^T$ of offsets $b_{t+1} = l_{emp}(w_t) - a_{t+1}^T w_t$, and the solution is computed as,

$$w_{t+1} = -CA\beta \tag{49}$$

Since the complexity increases with the number of iterations, the quadratic problem is not expensive and its complexity does not increase with the training set size.

Folos: An objective function describes and analyzes the FOLOS solver in the form $\hat{g}(w) = \ell_{emp}(w) + r(w)$, where $\ell_{emp}(w)$ is the empirical loss for the convex measure and $r(w)$ is the convex regularization term [22]. The solver mainly focuses on the loss-regularization class of convex optimization problems. Instead of performing projection on subgradients, this algorithm does an analytic minimization. Since we require the objective function minimization, the parameter w_t minimizes the term $\ell_{emp}(w)$ and hence the FOLOS requires the following updates,

$$\mathrm{w}_{t+1/2} = \mathrm{w}_t - \eta_t \nabla^{(s)} \ell_{emp}(\mathrm{w}_t) \tag{50}$$

$$\mathrm{w}_{t+1} = \underset{\mathrm{w}}{argmin} \left(\frac{1}{2} \mathrm{w} || \mathrm{w} - \mathrm{w}_{t+\frac{1}{2}} ||^2 + \eta_t + \frac{1}{2} \mathrm{r(w)} \right) \tag{51}$$

A weight vector is calculated from the updates w_{t+1} and it is in turn close to the updates $w_{t+\frac{1}{2}}$ for the minimum value of $r(w)$. Since it is described such that $O \in \partial f(w) \iff (\forall_v) \ f(v) \geq f(w)$, w estimates the minimum weight vector for the function ℓ_{emp}. The property implies the approximation as,

$$\mathrm{w}_{t+1} = \mathrm{w}_t - \eta_t \nabla^{(s)} \ell_{emp}(\mathrm{w}_t) - \eta_{t+\frac{1}{2}} \nabla^{(s)} \mathrm{r}(\mathrm{w}_{t+1}) \tag{52}$$

The equation denotes the alternate expression for the weight vector w_{t+1}. This equation in tern includes the subgradient $\nabla^{(s)} \mathrm{r}(w_{t+1})$ which is the gradient of $r(w)$ at weight vector (w_{t+1}). The vector (w_{t+1}) influences the update of the gradient anvit is even evaluated in advance. If the regularization ℓ_1 is chosen to the optimal result, the algorithm convergences with $O(\frac{1}{\sqrt{t}})$. The detailed relations can be expressed as,

$$\eta_t \, \alpha \, \frac{1}{\sqrt{t}}, \tag{53}$$

$$\min_{t \in \{1 \ldots T\}} \hat{g}(w_t) - \hat{g}(w^*) = O(GD\frac{\log T}{\sqrt{T}}) \tag{54}$$

where, $\|w^*\| < D$ for the optimal solution.

The subgradients ∂f and ∂r can be derived and these are bounded to the parameter G. The parameters ℓ_{emp} and r are H-strongly convex for the online case. The learning rate η_t is defined as $\eta_t = \frac{1}{t}$ and the target function T is computed as,

$$R(T) = O\left(\frac{G^2 \log T}{H}\right) \quad (55)$$

The FOLOS reaches the convergence in $\tilde{o}\left(\frac{1}{H_\epsilon}\right)$ when batch and online generalizations are combined.

Core vector machine: CVM algorithm allows the results to be obtained from the computational geometry. Training time is linear with the number of examples in this method and it reaches convergence within run time of $O\left(d\,.\,(\frac{n}{\rho^2} + \frac{1}{\rho^4})\right)$ for the accuracy ρ. As described in [23], an exact solution of the SVM optimization problem is not necessary for a good generalization.

The CVM algorithm aims at connecting SVMs to the problems of computational geometry which is known as the minimum enclosing ball (MEB). It requires a ball of minimum radius to enclose the set of points in the plane. The algorithm is expensive in terms of time and space, but it is possible to find a solution with respect to linear time for the specified points.

The run time is computed [23] as $o\left(\frac{d}{\rho^8}\right)$ and is independent of the number of samples. It is subsequently reduced to $o(\frac{d}{\rho^4})$ but not sufficient for the requirement. However, in case of strenuous optimization, it is faster than that of super linear limitation. For any value of x, the kernel is specified as $K(x, x)$. The implicit kernel is mapped onto a hyperplane from the given examples and many real world problems satisfy this mapping. CVM has not still achieved zero error results as it is required for the Support Vector Machines.

V. Results and Discussion

A. Experimental Setup

Speaker database: The proposed speaker recognition technique is implemented in research tool, MATLAB of version 7.10 and its performance is evaluated by using the NIST Speaker Recognition Evaluation 2010 (SRE-10) database [36]. The database consists of both the training and test data which are in turn involved in the recognition process. The data-base has been collected from the speakers who submitted the results without hearing the audios and without knowing the speaker assignments. This work employs one of the conditions available in the database, namely, the core-core condition in which the training and testing speech is collected from a two-channel telephone conversation with the duration of approximately five minutes or a microphone conversation with the duration of three to fifteen minutes. However, the system is well known in advance of speech data for the evaluation. From the evidences, the database can provide a maximum of 25000 test segments and up to 6000 speaker models with a maximum of 750000 trials.

Feature extraction: For each user, twenty five different speech samples have been extracted from the database in order to evolve speaker-specific features. Twenty speech samples are used for the training process and the remaining five samples are used for the testing [37]. The speech signals are specifically sampled at 16 KHz and are framed as windows of size 25ms. Each frame consists of FFT-based 256 dimensional power spectrum vectors in order to develop feature vectors. The feature vectors are in turn applied to the dimensionality reduction phase in order to derive an optimized feature set for the classification stage.

SVM based classification: Classification process comprises of both training and testing based on different SVM algorithms. Classification task has been carried out in this work for the proposed algorithms by conducting several experiments on the proposed database. The training set consists of approximately 71000 utterances from 3150 speakers of about 50 hours continuous speech [38]. In the testing process, the test set employed for connecting the recognition which comprises of approximately 20000 utterances from 500 different speakers of 5 hours duration. In general, the number of test samples represents just 10% of the whole test set samples.

B. Results

The experimental setup is employed to analyze the performances of the identification systems by evaluating the parameters such as accuracy, Detection Cost Function defined by NIST for 2010 (DCF10) and Equal Error Rate. The preceding section specifically describes the variation of accuracy with the sample size at various proposed dimensionality reduction techniques and SVM algorithms.

Accuracy: Accuracy is a crucial performance metric in speaker recognition and is evaluated for the proposed dimensionality reduction techniques and SVM algorithms by varying the sample size. The number of samples is varied from 15 to 150 and the performances are compared as described in the following figures. The variation of accuracy for the MPCA dimensionality reduction technique is depicted in Fig. 1.

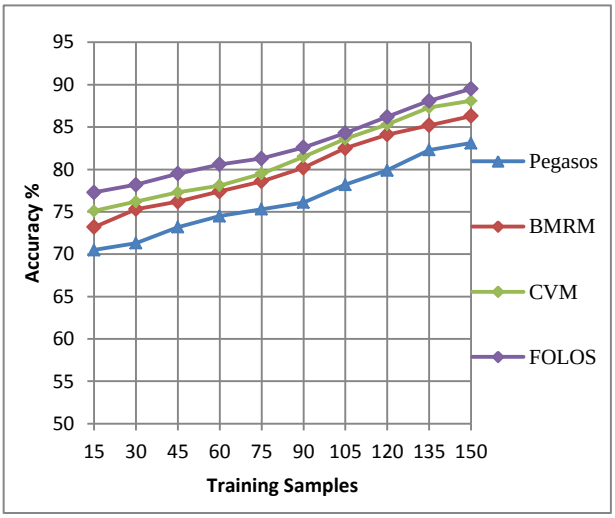

Figure 1. Variation of accuracy with training samples for MPCA

As depicted in Table I, it is evidenced from the experimental evaluations that the accuracy has been significantly improved for the FOLOS algorithm with the increased number of training samples. Since the algorithm tries to optimize the original and the objective functions simultaneously, the selection of cutting planes

possesses a higher chance to contribute the approximate objective function. Therefore, the performance of Pegasos algorithm is poorer as compared to others because the choice of cutting plane allows a higher execution time at the cost of the number of iterations to reach convergence.

TABLE I. RECOGNITION RATE WITH RESPECT TO VARIOUS SVM ALGORITHMS FOR THE MPCA DIMENSIONALITY REDUCTION TECHNIQUE

Sample size	Pegasos	BMRM	CVM	FOLOS
15	70.5	73.2	75.1	77.3
30	71.3	75.3	76.2	78.2
45	73.2	76.2	77.3	79.5
60	74.5	77.4	78.1	80.6
75	75.3	78.6	79.5	81.3
90	76.1	80.2	81.5	82.6
105	78.2	82.5	83.6	84.3
120	79.9	84.1	85.3	86.2
135	82.3	85.2	87.3	88.1
150	83.1	86.3	88.1	89.5

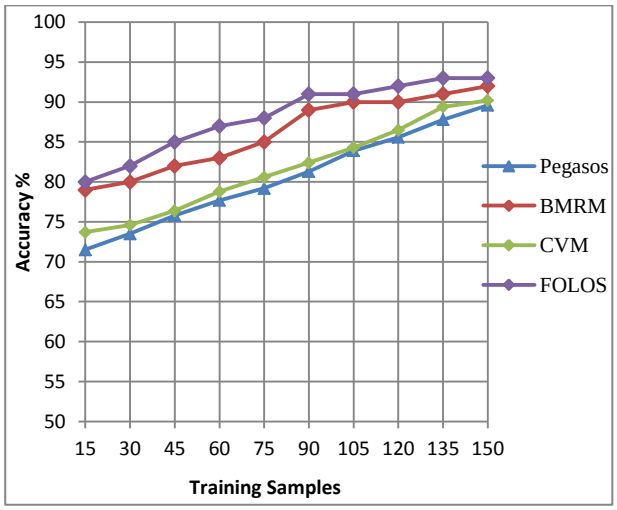

Figure 2. Variation of accuracy with training samples for PFA

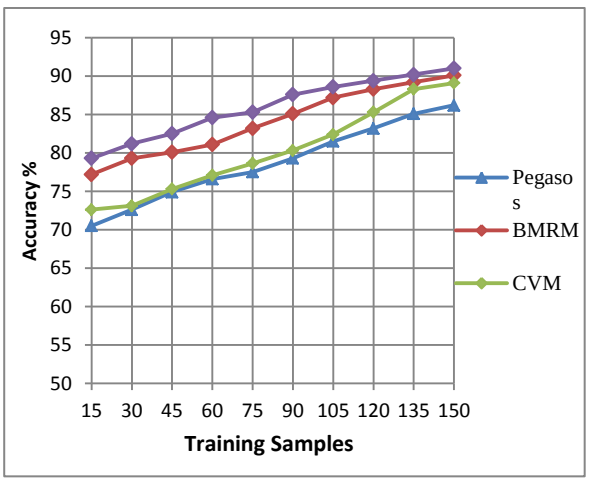

Figure 3. Variation of accuracy with training samples for MLFA

In Fig. 2, it is evidenced that the accuracy has been improved for the FOLOS algorithm because the principle factor components are nearer to the hyperplane.

In Fig. 3, it has been observed that the accuracy is comparable to the PFA reduction technique, but it is slightly less as compared to PFA and more than that of MPCA. Furthermore, the principle factors reduce the fluctuations towards the convergence and the MLFA dimensionality reduction method reaches the asymptotic performance.

Detection cost function (DCF): The performance of a speaker recognition system can be measured by using Detection Cost Function (DCF). Based on the experimental results that have been conducted for SVM algorithms under various Dimensionality Reduction techniques, the following section describes the DCF parameter metric.

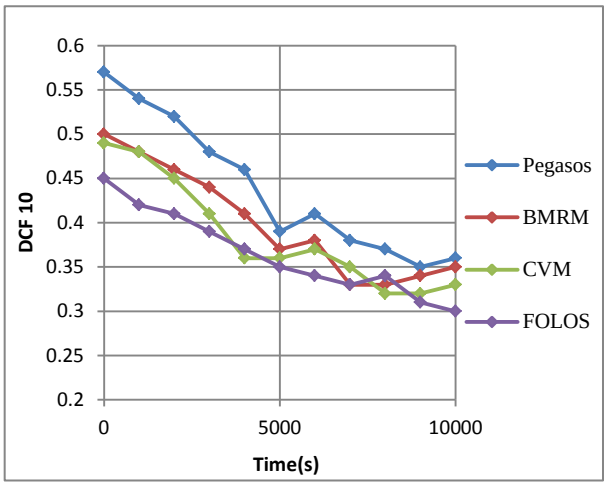

Figure 4. Variation of DCF10 in training time for MPCA

In Fig. 4, it has been identified that as the training time increases, the performance metric DCF10 substantially degrades. Furthermore, the Pegosis algorithm yields significantly good results because a reduced number of iterations is used to train data for evaluation. Moreover, it is observed that the CVM algorithm produces slightly differ from the Pegosis at low values, but closely meet at high time measures. Since the objective function is unstable, the iterations require considerably more time to reach convergence.

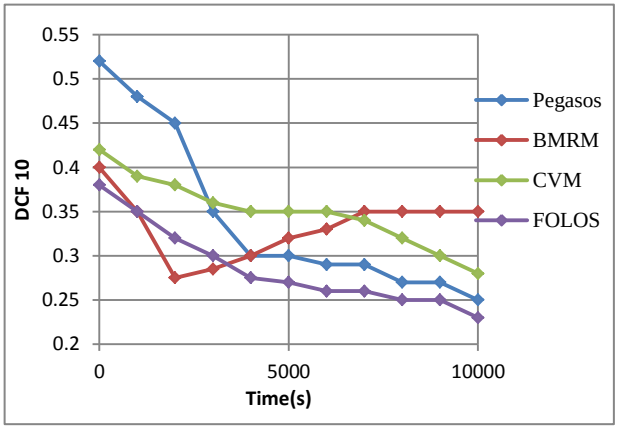

Figure 5. Variation of DCF10 with training time for PFA

TABLE II. DETECTION COST FUNCTION WITH RESPECT TO VARIOUS SVM ALGORITHMS FOR THE PFA DIMENSIONALITY REDUCTION TECHNIQUE.

Time(s)	Pegasos	BMRM	CVM	FOLOS
0	0.52	0.4	0.42	0.38
1000	0.48	0.35	0.39	0.35
2000	0.45	0.275	0.38	0.32
3000	0.35	0.285	0.36	0.3
4000	0.3	0.3	0.35	0.275
5000	0.3	0.32	0.35	0.27
6000	0.29	0.33	0.35	0.26
7000	0.29	0.35	0.34	0.26
8000	0.27	0.35	0.32	0.25
9000	0.27	0.35	0.3	0.25
10000	0.25	0.35	0.28	0.23

In Table II, Detection Cost Function with respect to various SVM algorithms for the PFA dimensionality reduction technique has been summarized. It has been noticed that as the training time increases, the bunch size of datasets cannot be decomposed because of its faster convergence. Therefore, it is evidenced from Fig. 5 that the DCF for FOLOS algorithm is considerably small as compared to the other algorithms under PFA dimensionality reduction technique.

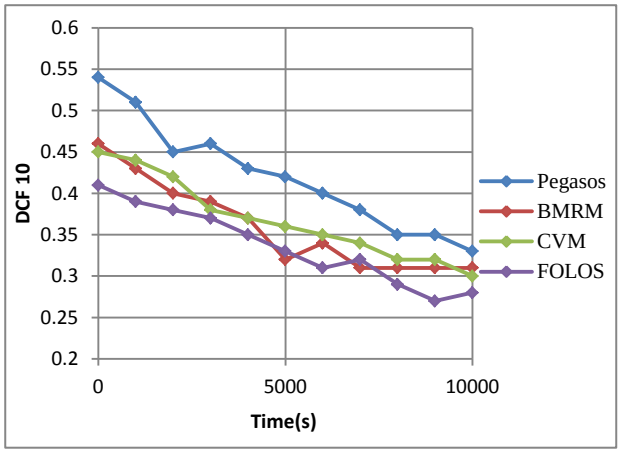

Figure 6. Variation of DCF10 with training time for MLFA

TABLE III. VARIATION OF EER WITH RESPECT TO SVM ALGORITHMS AND DIMENSIONALITY REDUCTION TECHNIQUES

Algorithm	Equal Error Rate (%)		
	MPCA	PFA	MLFA
Pegasos	3.15	2.54	2.65
BMRM	3.10	2.54	2.58
CVM	2.85	2.32	2.48
FOLOS	2.80	2.12	2.35

In Fig. 6, it has been observed that the DCF drastically reduces with respect to time for all SVM algorithms. In order to estimate the FOLOS algorithm at the end of iterations, the measured training time is enough to reach convergence for the target speaker.

Equal error rate (EER): The equal error rates for various SVM algorithms under different dimensionality reduction methods have been summarized in Table III.

In order to estimate the Equal Error Rates (EER), experiments were conducted for the sample size of 150 per user. From the experimental results, it has been identified that the Pegosis algorithm has produced poor results because of the imbalance in the classes, whereas the FOLOS algorithm takes the advantage of global complexity and the choice of reducing cutting planes. Furthermore, it has been observed that the combined PFA-FOLOS technique outperforms the other combinations. Therefore, the experimental setup has successfully estimated the EER performance metric for the recognition.

Population vs accuracy: Due to the fact that the recognition rate considerably degrades with respect to the database size, the performances of the recognition systems are severely affected. Compared to the state-of-the-art systems, the proposed techniques significantly enhance the recognition rate under large-scale data conditions as depicted in the following figure.

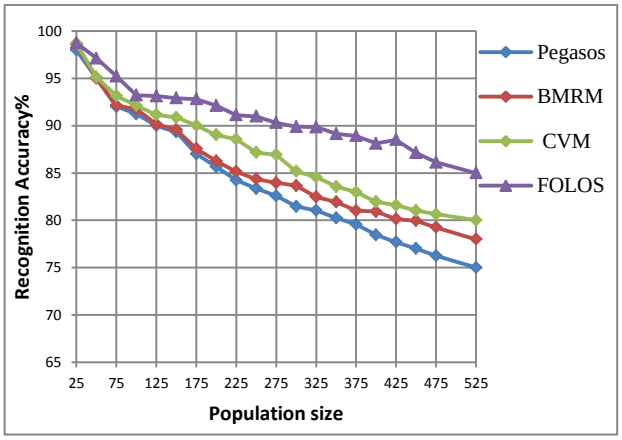

Figure 7. Variation of accuracy with population size for PFA

In Fig. 7, it is evident that the recognition accuracy has been improved for the FOLOS SVM algorithm with respect to population size. Furthermore, the experimental results of the parallel implementations of CVM and FOLOS have been measured for various characteristics to reach convergence. However, it has been noticed that the BMRM algorithm exhibits large fluctuations and spends more time to train its models.

VI. CONCLUSION

An effective speaker recognition system has been presented by using various dimensionality reduction techniques and SVM algorithms under large-scale data. Pitch and its strength have been employed as the feature set in the enrolment stage. The PFA dimensionality reduction technique has produced reasonably good results in order to minimize the dataset. Furthermore, the performance of various SVM algorithms has been tested under different dimensionality reduction techniques.

Compared to the various possible combinations, it can be concluded that the PFA with FOLOS SVM algorithm gives better performance. The effectiveness of the recognition rate makes the proposed techniques a promising solution for the speaker recognition under large-scale data.

REFERENCES

[1] J. P. J. Campbell, "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437-1462, Sep. 1997.

[2] T. Kinnunen, E. Karpov, and P. Franti, "Real-Time speaker identification and verification," *IEEE Trans. Audio, Speech, and Language Process*, vol. 14, no. 1, pp. 277-288, Jan. 2006.

[3] H. Beigi, *Speaker Recognition: Advancements and Challenges*, Intech Open Access Publisher, 2012.

[4] D. A. Reynolds, "An overview of automatic speaker recognition technology," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2002, pp. 4072-4075.

[5] A. M Othman and M. H. Riadh, "Speech Recognition using Scaly Neural Networks," *Proc. of Intl. Journal on Computer Systems Science and Engineering*, vol. 3, no. 2, pp. 253-258, 2008.

[6] J. J. Wolf, "Efficient acoustic parameters for speaker recognition," *Journal of the Acoustical Society of America*, vol. 51, no. 2, pp. 2044-2055, 1972.

[7] D. O. Shaughnessy, "Speaker recognition," *IEEE ASSP Magazine*, vol. 3, pp. 4-17, Oct. 1986.

[8] M. Chetouania, M. Faundez-Zanuyb, B. Gasa, and J. L. Zaradera, "Investigation on LP-residual representations for speaker identification," *Pattern Recognition*, vol. 42, no. 3, pp. 487-494, 2009.

[9] M. D. Plumpe, T. F. Quatieri, and D. A. Reynolds, "Modelling of glottal flow derivative waveform with application to speaker identification," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 5, pp. 569-586, 1999.

[10] W. N. Chan, N. Zheng, and T. Lee, "Discrimination power of vocal source and vocal tract related features for speaker segmentations," *IEEE Transactions on Audio, Speech and Signal Processing*, vol. 15, no. 6, pp. 1884-1892, 2007.

[11] L. Mary and B. Yegnanarayana, "Extraction and representation of prosodic features for language and speaker recognition," *Speech Communication*, vol. 50, pp. 782-796, 2008.

[12] K. S. R. Murty, S. R. M. Prasanna, and B. Yegnanarayana, "Speaker specific information from residual phase," in *Proc. Int. Conf. on Signal Processing and Comm. (SPCOM)*, IISc, Bangalore, Dec. 2004, pp. 516-519.

[13] D. Pati and S. R. M. Prasanna, "Non-Parametric vector quantization of excitation source information for speaker recognition," in *Proc. TENCON 2008 - 2008 IEEE Region 10 Conference*, Nov. 2008, pp. 1-4.

[14] S. D. Villalba, "Dimension reduction," Technical Report UCD-CSI-2007-7, University College Dublin, 2007.

[15] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Multilinear principal component analysis of tensor objects for recognition," in *Proc. International Conference on Pattern Recognition, ICPR 2006*, 2006, pp. 776-777.

[16] C. Liang, L. Yang, H. Suo, J. Wang, and Y. Yan, "Factor analysis of Laplacian approach for speaker recognition," in *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 4221-4224.

[17] S. Park and S. Gupta, "A simulated maximum likelyhood estimator for the random coefficient logit model using aggregate data," *Journal of Marketing Research*, vol. 46, no. 4, pp. 531-542, 2009..

[18] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, 1998.

[19] W. Campbell, J. Campbell, T. Gleason, D. Reynolds, and W. Shen, "Speaker verification using support vector machines and high-level features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2085-2094, 2007.

[20] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal estimated sub-gradient solver for SVM," in Proc. *24th International Conference on Machine Learning*, Corvallis, 2007, pp. 807-814.

[21] C. H. Teo, A. Smola, S. V. Vishwanathan, and Q. V. Le, "Bundle methods for regularized risk minimization," *Journal of Machine Learning Research.*, vol. 11, pp. 311-365, Mar. 2010.

[22] J. Duchi and Y. Singer, "Efficient online and batch learning using forward backward splitting," *Journal of Machine Learning Research*, vol. 10, pp. 2899-2934, Dec. 2009.

[23] I. W. Tsang, J. T. Y. Kwok, and J. M. Zurada, "Generalized core vector machines," *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1126-1140, Sep. 2006.

[24] W. Hess, "Pitch determination of speech signals," *IEEE Transactions on Neural Networks*, vol. 17, no. 5, pp. 1126-1140, 2006.

[25] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, Prentice-Hall Publishers, 1999, pp. 468-471.

[26] R. Rozman and D. M. Kodek, "Using asymmetric windows in automatic speech recognition," *Journal of Elsevier, Speech Communication*, vol. 49, pp. 268-276, Jan. 2007.

[27] J. Zhu, S. Sun, X. Liu, and B. Lei, "Pitch in speaker recognition," in *Proc. The Ninth International Conference on Hybrid Intelligent Systems*, Aug. 2009, vol. 1, pp. 33-36.

[28] H. Hung, P. Wu, I. Tu, and S. Huang, "On multilinear principal component analysis of order-two tensors," *Journal of Biometrika*, vol. 99, no. 3, pp. 569-583, 2012.

[29] C. W. Ting and J. T. Chien, "Factor analysis of acoustic features for streamed hidden Markov modeling," in *Proc. IEEE Automatic Speech Recognition Understanding Workshop*, 2007, pp. 30-35.

[30] P. Filzmoser, K. Hron, C. Reimann, and R. Garrett, "Robust factor analysis for compositional data," *Journal of Computers and Geo-Sciences*, vol. 35, no. 9, pp. 1854-1861, Sep. 2009.

[31] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, 1998.

[32] W. Campbell, J. Campbell, T. Gleason, D. Reynolds, and W. Shen, "Speaker verification using support vector machines and high-level features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2085-2094, 2007.

[33] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Primal estimated sub-gradient solver for SVM," in *Proc. Proc. ICML 2007*, 2007, pp. 807-814.

[34] J. Shawe-Taylor and S. Sun, "A review of optimization methodologies in support vector machines," *Journal of Neurocomputing*, vol. 74, no. 17, pp. 3609-3618, Oct. 2011.

[35] J. Duchi, E. Hazan, and Y. Singer, "Adaptive sub-gradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121-2159, Jul. 2011.

[36] NIST SRE 2010 evaluation plan. [Online]. Available: http://www.itl.nist.gov/iad/mig//tests/sre/2010/NIST_SRE10_eval plan.r6.pdf

[37] K. Saeed and M. K. Nammous, "A speech and speaker identification system: feature extraction, description, and classification of speech-signal image," *IEEE Transactions on Industrial Electronics*, vol. 54, no. 2, pp. 887-897, Apr. 2007.

[38] A. Abad, T. Pellegrini, I. Trancoso, and J. Neto, "Context dependent modelling approaches for hybrid speech recognizers," in *Proc. 11th Annu. Conference on Int. Speech Commun. Assoc. (Interspeech 2010)*, Chiba, Japan, Sep. 2010, pp. 2950-2953.

Mr. P. Rama Koteswara Rao is currently working as Associate Professor in ECE Department, Usha Rama College of Engineering and Technology, Telaprolu, AP, India. He is working towards his Ph.D. at Andhra University College of Engineering, Visakhapatnam, AP, India. He received his M.Tech. from the JNTU college of Engineering Anantapur. He has fifteen years of experience in teaching undergraduate students and three years industrial experience. His research interests are in the areas of speech signal processing, embedded systems and speaker recognition techniques. He is a life member of ISTE and ISOI.

Dr. Y. Srinivasa Rao received his Ph.D. in Electrical Communication Engineering from Indian Institute of Science, Bangalore in 1998. At present, he is a UGC - PDF Research Fellow and also Professor and BOS Member in Instrument Technology Department, AU College of Engineering (A), Andhra University, Visakhapatnam, AP, India. He is having more than 18 years of teaching and research experience and published more than 88 research papers in the International journals and presented few of them in the International Conferences. He was in Eritrean Institute of Technology, Asmara, N.E. Africa for one year on academic assignment. He had given several Invited Talks and Guest Lectures at several International and National Conferences, and Workshops for the last few years. He will be a Visiting Researcher, EEE Department, Nanyang Technological University, Singapore in 2012. He received "Research Award" in 2012from University Grants Commission, India, "Outstanding Paper Award Winner" at the Emerald Literati Network Awards for Excellence 2008, England (UK), Five "Best Paper Awards" at International Conferences (two in 2012, one in 2011 and twice in 2007), "Vignan Pratibha Award-2006" for Research from Andhra University and selected as "Science Researcher for Asia–Pacific Region" in 2005 by UNESCO and Australian Expert group in Industry Studies (AEGIS) at the University of Western Sydney (UWS) and received "Young Scientist Award" from Department of Science and Technology, Government of India in 2002. His name is listed in "LEADING SCIENTISTS OF THE WORLD-2007, 2008, 2010" by The International Biographical Centre, Cambridge, England and his Biography was listed in MARQUIS Who's Who in Science and Engineering (USA)-2006, 2007, and 2010. He is acting as Reviewer for several International and National Journals. His present research focuses on Carbon Nanotube Electronics including Modeling, Design and Simulation, Fabrication of Carbon Nanotube Field Effect Transistor Devices and Circuits including Ambipolar electronic devices and also on Fabrication, Characterization, Modeling, Trimming and reliability studies of polymer Thick Film Resistors, Nanobiosensors, Flexible Electronics using Carbon Nanotubes and Spintronic logic devices using graphene. His research work "Speech processing and Synthesis" has been accepted for inclusion as one chapter for publication in the book "Speech Technologies/Book 1" publishing in June 2011 from Vienna, Austria European Union. He is a life member of IEEE, IETE, LISTE and member of IMAPS (USA and India).