On-Line Linear Combination of Classifiers Based on Incremental Information in Speaker Verification

Fernando Huenupán, Néstor Becerra Yoma, Claudio Garretón, and Carlos Molina

A novel multiclassifier system (MCS) strategy is proposed and applied to a text-dependent speaker verification task. The presented scheme optimizes the linear combination of classifiers on an on-line basis. In contrast to ordinary MCS approaches, neither a priori distributions nor pre-tuned parameters are required. The idea is to improve the most accurate classifier by making use of the incremental information provided by the second classifier. The on-line multiclassifier optimization approach is applicable to any pattern recognition problem. The proposed method needs neither a priori distributions nor pre-estimated weights, and does not make use of any consideration about training/testing matching conditions. Results with Yoho database show that the presented approach can lead to reductions in equal error rate as high as 28%, when compared with the most accurate classifier, and 11% against a standard method for the optimization of linear combination of classifiers.

Keywords: Speaker verification, multiclassifier system, incremental information.

I. Introduction

In pattern recognition, the problem of using multiclassifier systems (MCS) has been addressed in several fields [1]. The motivation behind MCS is the fact that the response to the same input signal is classifier dependent, so the error of a given classifier could be corrected by the whole system. From pattern recognition theory, the most straightforward formal strategy to fuse classifiers (Fig. 1) is certainly the Bayes classification theory [1], [2]:

$$D(X) = \arg\max_m \left\{ \Pr\left[C_m \mid S(X)\right] \right\} = \arg\max_m \left\{ \frac{\Pr\left[S(X) \mid C_m\right] \cdot \Pr(C_m)}{\sum_{\tilde{m}=1}^{M} \Pr\left[S(X) \mid C_{\tilde{m}}\right] \cdot \Pr(C_{\tilde{m}})} \right\}, \quad (1)$$

where $S(X)=\left[S_{\mathrm{CL}_1}(X),\cdots,S_{\mathrm{CL}_j}(X),\cdots,S_{\mathrm{CL}_J}(X) \right]$ and $S_{\mathrm{CL}_j}(X)$ is the score of classifier j; J is the total number of classifiers; C_m denotes the m-th class; and M is the total number of classes; and $D(X)$ is the final decision or classification that corresponds to input X. As can be seen in Fig. 1, the classifier outputs can be combined at abstract level or score level. In the former case, the individual classifier decisions, $d_{\mathrm{CL}_j}(X)$, are combined; in the latter case, the scores of individual classifiers, $S_{\mathrm{CL}_j}(X)$, are merged.

Theoretically, the classification error is optimally minimized by (1). However, the $\Pr\left[S(X) \mid C_m\right]$ of the a priori multivariable probability density function (PDF) may require an unmanageable amount of training data to be reliably estimated [1]. As a consequence, the problem is

Manuscript received May 27, 2009; revised Dec. 23, 2009; Jan. 12, 2010.

Fernando Huenupán (phone: +56 2 978 4218, email: fhuenupa@ing.uchile.cl), Néstor Becerra Yoma (email: nbecerra@ing.uchile.cl), Claudio Garretón (email: cgarreto@ing.uchile.cl), and Carlos Molina (email: cmolina@ing.uchile.cl) are with the Speech Processing and Transmission Laboratory, Department of Electrical Engineering, Universidad de Chile, Santiago, Chile.

doi:10.4218/etrij.10.0109.0301

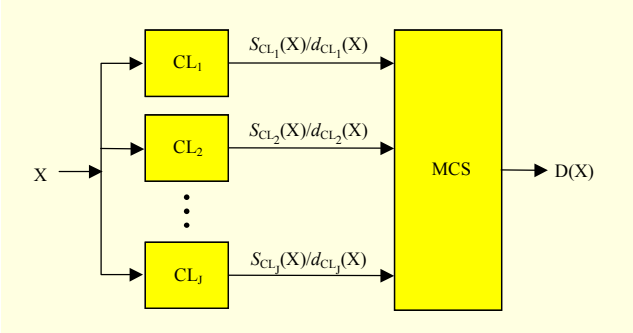

Fig. 1. Classical MCS scheme.

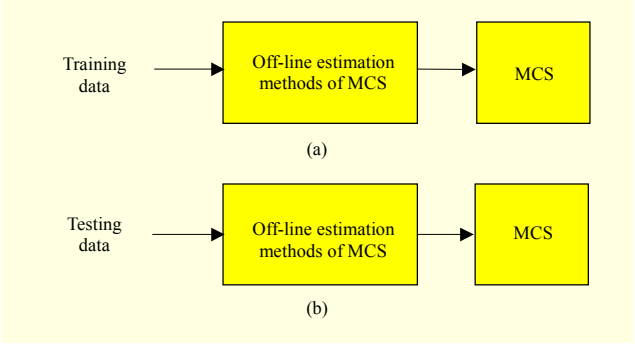

Fig. 2. (a) Off-line and (b) on-line estimation of fusion parameters in MCS.

substantially simplified if the maximization in (1) can be expressed in terms of computations performed by individual classifiers.

The classical techniques to simplify the Bayesian fusion [1], [3], [4] are product rule, sum rule, max rule, min rule, mean rule, and majority vote rule. However, it is also necessary to estimate a distribution for each individual classifier, and the training/testing mismatch problem remains. Mismatch between training and testing conditions is one of the most severe problems in pattern recognition. It can dramatically degrade the accuracy of the whole classifier system.

In order to counteract some of the limitations presented by Bayes-based fusion techniques, methods based on maximum entropy and mutual information theory have been proposed elsewhere to combine several sources of information [5]-[7]. Entropy and mutual information have been used to combine classifiers in ordinary multiclassifier fusion [7], [8], multimodal classifiers [9], [10], and multisensor systems [11], [12], among others. Maximum entropy is a versatile modeling criterion that allows straightforward integration of constraints such as correlation between classifiers and reliability of experts [7]. The motivation behind the use of information theory is to take into consideration the uncertainty of each information source and then, to improve the accuracy of the a priori conditional distribution estimation. Some examples of the applicability of information theory to pattern recognition problems are entropy [5], [7], [13], [14], mutual information [15], [16], and conditional entropy [6]. However, all those methods usually assume matching conditions between training and testing data with variable requirements for the estimation database size.

Moreover, the maximization of entropy is not a suitable criterion to optimize the linear combination of classifiers. The classifier score with the highest variance will tend to provide the maximum entropy. Also, conditional entropy and mutual information require the evaluation of multivariable distributions, which in turn worsen the requirements associated with training-testing matching condition and estimation database size.

In speaker verification (SV), neural networks [17]-[19], linear combination [20]-[22], and binary methods [23] are the most popular approaches to tackle the problem of optimizing the use of multiple classifiers. In [24], Bayes-based confidence measure is proposed as a framework for multiclassifier fusion.

In summary, in classical MCS schemes (Fig. 1), the fusion parameters are optimized a priori by employing a training data set as shown in Fig. 2(a). As a consequence, the optimized MCS is vulnerable to mismatching between training/testing conditions. In contrast, the method proposed here optimizes the fusion parameters on an on-line basis by making use of testing data only (Fig. 2(b)).

In this paper, the mutual information criterion is applied to address the problem of on-line optimization of multiclassifier fusion in text-dependent (TD) SV with limited testing data, that is, utterances shorter than 3 or 5 seconds. The proposed method attempts to improve the most accurate classifier by making use of the incremental information provided by the second classifier. The addressed MCS scheme is the weighted linear combination (also known as the sum rule) of classifier scores. The term *on-line* denotes that the classifier combination optimization takes place in the verification or testing procedure, no training-testing matching condition is required, and, no a priori distributions are employed. The restriction of limited testing data is especially suitable for TD-SV, which in turn provides more interesting potential applications from the commercial point of view than text-independent SV (TI-SV). However, the proposed approach is also applicable to any pattern recognition problem and is promising from the practical and theoretical points of view. Finally, the presented method leads to highly significant reductions in equal error rate (EER) when compared with the most accurate classifier and with a standard classification fusion method that requires a training-testing matching condition.

II. Linear Combination of Classifiers and Mutual Information

Linear combination, usually denominated as *weighted sum rule*, is one of the most common strategies of metrics combination [25] and is a simplification of the Bayesian fusion [1]. In this paper, the optimization of the weighted linear combination scheme with two classifiers, CL_1 and CL_2, is addressed. Given an input utterance X composed of I frames, $X=\{x_1,x_2,\cdots,x_i,\cdots,x_I\}$, there are two sets of output scores, $P^{(1)}=\{P_1^{(1)},\cdots,P_i^{(1)},\cdots,P_I^{(1)}\}$ and $P^{(2)}=\{P_1^{(2)},\cdots,P_i^{(2)},\cdots,P_I^{(2)}\}$, provided by classifiers CL_1 and CL_2, respectively. $P_i^{(1)}$ and $P_i^{(2)}$ denote the classifier scores with frame x_i. The linear combination (or weighted sum rule) of CL_1 and CL_2 at frame i, $\hat{P}_i$, is expressed as

$$\hat{P}_i(\alpha_i)=(1-\alpha_i)\cdot P_i^{(1)}+\alpha_i\cdot P_i^{(2)}, \quad (2)$$

where $0\le\alpha_i\le 1$ is a weighting or scaling factor that defines the linear combination. As a result, $\hat{P}_i$ is a function of $\alpha_i, P_i^{(1)}$ and $P_i^{(2)}$. Then, the linear combination of classifier scores associated to X is evaluated as

$$\hat{P}(A)=\frac{1}{I}\sum_{i=1}^{I}\hat{P}_i(\alpha_i)=\frac{1}{I}\sum_{i=1}^{I}\left[(1-\alpha_i)\cdot P_i^{(1)}+\alpha_i\cdot P_i^{(2)}\right], \quad (3)$$

where $A=\{\alpha_1,\alpha_2,\cdots,\alpha_i,\cdots,\alpha_I\}$ denotes the whole set of scaling factors in utterance X.

SV is a two class problem [25] where the final decision, $D(X)$, takes place according to

$$D(X)=\begin{cases}\hat{P}(A)>Th\Rightarrow\text{claimed identity is accepted},\\ \hat{P}(A)\le Th\Rightarrow\text{claimed identity is rejected},\end{cases} \quad (4)$$

where Th is a decision threshold. Observe that the estimation of $A=\{\alpha_i\}$, $1\le i\le I$ in (2) that minimizes the error rate could be seen as an information source fusion problem. It is worth highlighting that linear combination is probably the most straightforward procedure to combine metrics or scores in a pattern recognition problem.

Information theory, particularly the maximization of entropy, is a popular approach that is employed in several problems [5]-[7], [13], [14]. As mentioned, the estimation of $A=\{\alpha_i\}$, $1\le i\le I$, according to the maximization of entropy of $\hat{P}$ in (2), is not applicable. For instance, if the distributions of $P^{(1)}$ and $P^{(2)}$ are considered Gaussians, the entropy of $P^{(1)}$ and $P^{(2)}$ is proportional to the natural logarithm of their variances [26]. Consequently, it can easily be shown that the entropy of $\hat{P}$ is monotonically decreasing or increasing between the entropies of $P^{(1)}$ and $P^{(2)}$. Mutual information and conditional entropy [5], [6], [15], [16], could be interesting candidates to optimize the linear combination in (2) by taking into consideration the incremental information of $\hat{P}$ with respect to $P^{(1)}$. In this case, mutual information is defined as [27]

$$I(P^{(1)},\hat{P})=H(P^{(1)})-H(P^{(1)}\mid\hat{P}), \quad (5)$$

and conditional entropy corresponds to

$$H\left(P^{(1)}\mid\hat{P}\right)=H\left(P^{(1)},\hat{P}\right)+H\left(P^{(1)}\right). \quad (6)$$

According to (5) and (6), mutual information and conditional entropy require the estimation of cross entropy $H(P^{(1)},\hat{P})$, which in turn requires evaluation of the joint probability distribution (PDF) $\Pr(P^{(1)},\hat{P})$. As a result, the optimization of $I(P^{(1)},\hat{P})$ and $H(P^{(1)}\mid\hat{P})$ should be highly dependent on the size of available data and can hardly be employed on an on-line basis with limited data.

Consider that classifier CL_1 provides a lower error rate than CL_2. The optimization of the linear combination of classifiers proposed in this paper attempts to improve the performance of the most accurate classifier (CL_1) with the information provided by the least accurate expert, CL_2. Accordingly, the distribution of $\hat{P}(A)$ in (3) could be interpreted as the distribution of $P^{(1)}$ modified by using the information of the second classifier, CL_2. The proposed multiclassifier method optimizes the mutual information between the score of the most accurate classifier, $P^{(1)}$, and the distribution of $\hat{P}(A)$ in (3).

It is worth emphasizing that the optimization of the mutual information between $P^{(1)}$ and $P^{(2)}$ criterion would require the estimation of joint distributions of $P^{(1)}$ and $P^{(2)}$, which in turn requires a significant amount of data. In contrast, as shown in section I, the presented scheme makes use of the distribution of $\hat{P}(A)$ whose estimation can easily be achieved on an utterance-by-utterance basis.

1. Estimation of the Distribution of $\hat{P}(A)$

Figure 3 shows the histograms of classifier scores within two verification utterances. On average, each utterance provides 300 frames. The scores are computed on a frame-by-frame basis. According to Fig. 3, the distribution of classifier scores within a given utterance could be modeled as a Gaussian PDF. Consider that $\varepsilon\left[P^{(1)}\mid\Phi_{p^{(1)}}\right]$, $f\left[P^{(2)}\mid\Phi_{P^{(2)}}\right]$, and $g\left[\hat{P}\mid\Phi_{\hat{P}}(A)\right]$ are the PDFs of $P_i^{(1)}$, $P_i^{(2)}$, and $\hat{P}_i$ within an utterance, respectively, where $\Phi_{p^{(1)}}=\left\{\mu_{p^{(1)}},\sigma^2_{p^{(1)}}\right\}$ are the mean and variance of $P_i^{(1)}$, $\Phi_{P^{(2)}}=\left\{\mu_{P^{(2)}},\sigma^2_{P^{(2)}}\right\}$ are the mean and variance of $P_i^{(2)}$, and $\Phi_{\hat{P}}(A)=\left\{\mu_{\hat{P}}(A),\sigma^2_{\hat{P}}(A)\right\}$ are the mean and variance of $\hat{P}_i$, respectively. Observe

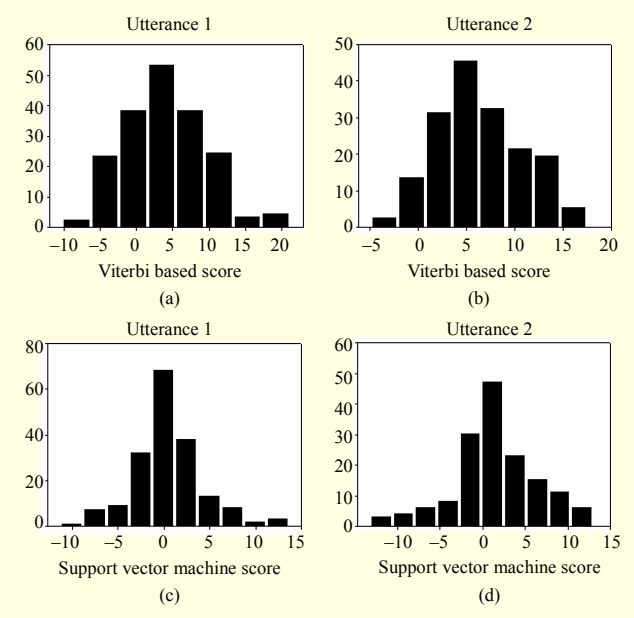

Fig. 3. Classifier score histograms within two verification utterances: (a) Viterbi based score in utterance 1; (b) Viterbi based score in utterance 2; (c) support vector machine score in utterance 1; and (d) support vector machine score in utterance 2. The scores are estimated on a frame-by-frame basis.

that $\Phi_{P^{(1)}}$ and $\Phi_{P^{(2)}}$ are estimated on an utterance-by-utterance basis according to

$$\mu_{P^{(1)}} = \frac{1}{I}\sum_{i=1}^{I} P_i^{(1)}, \qquad \mu_{P^{(2)}} = \frac{1}{I}\sum_{i=1}^{I} P_i^{(2)},$$

$$\sigma^2_{P^{(1)}} = \frac{1}{I}\sum_{i=1}^{I}\left(P_i^{(1)} - \mu_{P^{(1)}}\right)^2, \quad \sigma^2_{P^{(2)}} = \frac{1}{I}\sum_{i=1}^{I}\left(P_i^{(2)} - \mu_{P^{(2)}}\right)^2.$$

Then, $\Phi_{\hat{P}}(A)$ is computed according to

$$\mu_{\hat{P}}(A) = \frac{1}{I}\sum_{i=1}^{I}\hat{P}_i(\alpha_i) = \frac{1}{I}\sum_{i=1}^{I}\left[(1-\alpha_i)\cdot P_i^{(1)} + \alpha_i \cdot P_i^{(2)}\right], \quad (7)$$

$$\sigma^2_{\hat{P}}(A) = \frac{1}{I}\sum_{i=1}^{I}\left[\hat{P}_i - \mu_{\hat{P}}(A)\right]^2. \quad (8)$$

If α is made equal to a constant within a given utterance, that is, $\alpha_i = \alpha$, where $1 \le i \le I$, the mean and variance of $\hat{P}_i$ corresponds to

$$\mu_{\hat{P}}(\alpha) = (1-\alpha)\cdot\mu_{P^{(1)}} + \alpha\cdot\mu_{P^{(2)}}, \quad (9)$$

and

$$\sigma^2_{\hat{P}}(\alpha) = \alpha^2\cdot\left[\sigma^2_{P^{(2)}} + \sigma^2_{P^{(1)}} - 2\cdot E\left[P^{(1)}\cdot P^{(2)}\right] + 2\mu_{P^{(1)}}\cdot\mu_{P^{(2)}}\right] - 2\alpha\cdot\left[\sigma^2_{P^{(1)}} - E\left[P^{(1)}\cdot P^{(2)}\right] + \mu_{P^{(1)}}\cdot\mu_{P^{(2)}}\right] + \sigma^2_{P^{(1)}}, \quad (10)$$

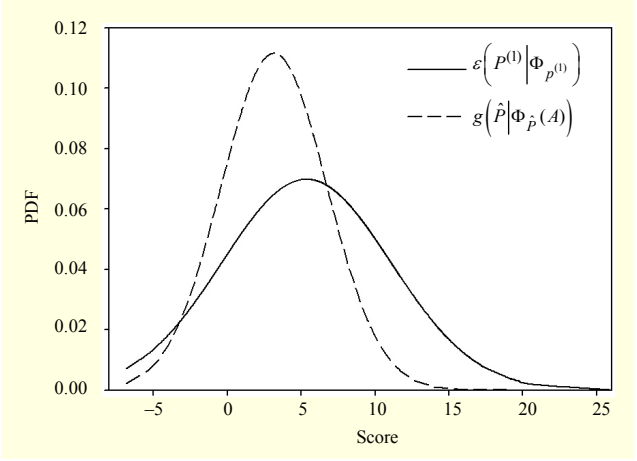

Fig. 4. Comparison of the PDFs corresponding to $P^{(1)}$ (the most accurate classifier score), $\varepsilon\left[P^{(1)}\middle|\Phi_{P^{(1)}}\right]$, and to the optimal $\hat{P}$ (the linearly combined classifier score), $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$.

where, $E\left[P^{(1)}\cdot P^{(2)}\right] = \frac{1}{I}\sum_{i=1}^{I} P_i^{(1)}\cdot P_i^{(2)}$.

As can be seen in (7) and (8), $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$ is a function of A, which in turn is optimized to improve the discrimination ability of the linear combination of classifiers when compared with the most accurate classifier. As an example, Fig. 4 shows the distribution of the most accurate classifier score, $P^{(1)}$, and the distribution of the score resulted from the linear combination, $\hat{P}(A)$, when optimized with respect to A. As is suggested in Fig. 4, the problem of optimizing the linear combination of experts can be interpreted as correcting the most accurate classifier score distribution in order to improve the classification accuracy.

2. Optimization of Linear Combination of Classifiers Based on Mutual Information

As mentioned above, this paper proposes the optimization of linear combination of classifiers by making use of the mutual information between $P^{(1)}$ and $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$. The mutual information [27] between $P^{(1)}$ and $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$ can be written as

$$I\left\{P^{(1)}; g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\} = H\left[P^{(1)}\right] - H\left\{P^{(1)} \mid g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}. \quad (11)$$

Maximizing the additional information provided by $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$ to $P^{(1)}$ is equivalent to minimizing the mutual information between $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$ and $P^{(1)}$,

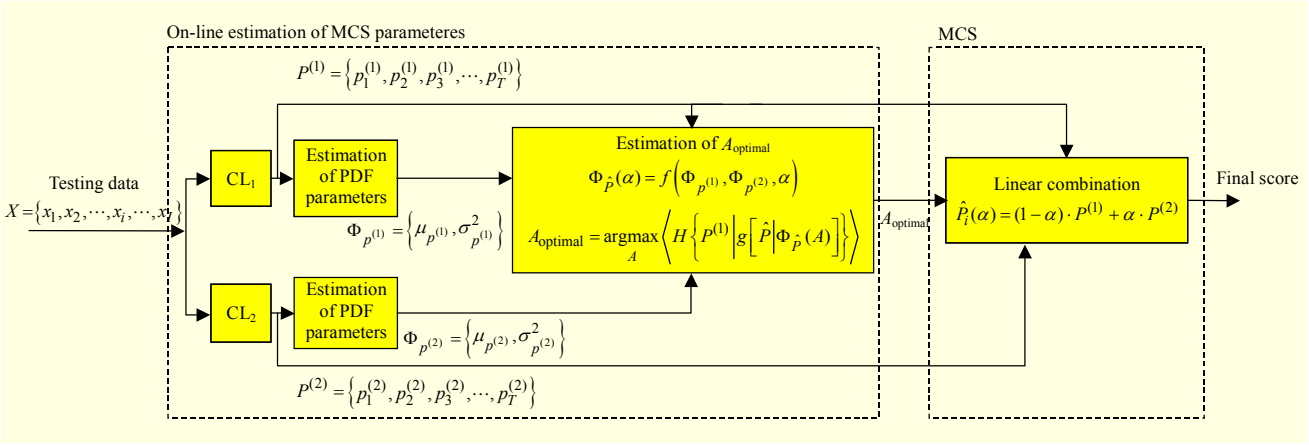

Fig. 5. Block diagram of the incremental information based on-line optimization of linear combination of classifiers.

$I\left\{P^{(1)}; g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}$. As a consequence, the optimal A, A_{optimal}, that defines the linear combination of classifiers according to (3) could be estimated as

$$A_{\text{optimal}} = \arg\min_{A}\left\langle I\left\{P^{(1)}; g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}\right\rangle = \arg\min_{A}\left\langle H\left[P^{(1)}\right] - H\left\{P^{(1)} \mid g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}\right\rangle. \quad (12)$$

As can be seen in (12), $H\left[P^{(1)}\right]$ does not depend on A, and minimizing $I\left\{P^{(1)}; g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}$ is equivalent to maximizing $H\left\{P^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}$.

$$A_{\text{optimal}} = \arg\max_{A}\left\langle H\left\{P^{(1)} \mid g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}\right\rangle. \quad (13)$$

As a result, A_{optimal} can be computed by estimating the partial derivative of $H\left\{P^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}$ with respect to A and then equating to zero:

$$\frac{\partial\left\{H\left[P^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right]\right\}}{\partial A} = 0, \quad (14)$$

where

$$H\left\{P^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right], A\right\} = -\sum_{i=1}^{I}\Pr\left\{P_i^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}\cdot\ln\left\langle\Pr\left\{P_i^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}\right\rangle. \quad (15)$$

$\Pr\left\{P_i^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}$ is estimated by evaluating $P_i^{(1)}$ in $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$. Notice that PDF $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$ in (15) is evaluated previously with (9) and (10). Figure 5 summarizes the MCS scheme proposed here.

III. Signal-by-Signal Based On-Line Optimization of Linear Combination of Classifiers

Ordinary methods of multiclassifier combination such as the Bayesian fusion and neural networks require training data to estimate a priori distributions and weights. As a result, those fusion techniques implicitly require matching conditions between training and testing. As is very well known, most real problems in the field of pattern recognition hardly comply with those training-testing matching condition requirements, which in turn degrades the performance of the multiclassifier system. The on-line classifier fusion method proposed here optimizes the combination of experts without the use of a priori distributions or pre-estimated weights. When applied to the SV problem the optimization of the multiclassifier system takes place on an utterance-by-utterance basis. As mentioned above, the on-line optimization of A in (3) is achieved by maximizing $H\left\{P^{(1)}|g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]\right\}$ in (3) by means of (12). In this paper, three procedures to estimate A_{optimal} were evaluated, frame independent optimization; frame-dependent optimization; and average frame dependent optimization.

1. Estimation of A_{optimal} as a Constant within the Whole Utterance

If α is made equal to a constant within a given utterance, that is, $\alpha_i = \alpha$ and $1 \le i \le I$, the mean and variance of $g\left[\hat{P}|\Phi_{\hat{P}}(A)\right]$ as functions of α can be estimated according to (9) and (10). Then, $A_{\text{optimal}} = \left\{\alpha_i = \alpha_{\text{optimal}}\right\}$, with $1 \le i \le I$, according to (3), can be estimated by means of (14), which leads to solving the following equation:

$$\frac{\partial\left\{H\left[P^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha_{\text{optimal}}\right)\right]\right.\right]\right\}}{\partial\alpha_{\text{optimal}}}=0\,, \qquad (16)$$

where $H\left\{P^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha\right)\right]\right.\right\}$ is defined as in (15) with $A=\{\alpha_i=\alpha\}$, with $1\le i\le I$, and is expressed as

$$H\left\{P^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha\right)\right]\right.\right\} = -\sum_{i=1}^{I}\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha\right)\right]\right.\right\}\cdot\ln\left\langle\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha\right)\right]\right.\right\}\right\rangle. \qquad (17)$$

The estimation of α_{optimal} by applying (16) does not lead to an analytical solution, and a numerical estimation for solving (16) is adopted:

Step 1: Estimate N_{samples} of α, α^j, uniformly distributed in the interval [0, 1] with

$$\alpha^j=\frac{1}{N_{\text{samples}}}\cdot j, \qquad 0\le j\le N_{\text{samples}}-1\,.$$

Step 2: Estimate $g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha^j\right)\right]$ with (9) and (10), where α^j is computed as described in the previous step.

Step 3: Obtain $H\left\{P^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha^j\right)\right]\right.\right\}$ according to (17), where α^j is defined in step 1.

Step 4: α_{optimal} is estimated by using a polynomial approximation [28].

Step 5: Finally, score $\hat{P}$ is computed as

$$\hat{P}\left(\alpha_{\text{optimal}}\right)=\frac{1}{I}\sum_{i=1}^{I}\left\{\left(1-\alpha_{\text{optimal}}\right)\cdot P_i^{(1)}+\alpha_{\text{optimal}}P_i^{(2)}\right\}. \qquad (18)$$

2. Frame-by-Frame Based Estimation of A_{optimal}

An optimal α_i on each frame i could be defined according to the following maximization:

$$\alpha_{\text{optimal}}\left(i\right)=\arg\max_{\alpha}\left\langle\begin{array}{l}\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha\right)\right]\right.\right\}\\ \times\ln\left\langle\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha\right)\right]\right.\right\}\right\rangle\end{array}\right\rangle. \qquad (19)$$

Observe that $\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}(\alpha)\right]\right.\right\}\cdot\ln\left\langle\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}(\alpha)\right]\right.\right\}\right\rangle$ is the term within the summation in (17). The optimization in (19) can be achieved as follows:

Step 1: Estimate N_{samples} of α, α^j, uniformly distributed in the interval $[0,1]$, with

$$\alpha^j=\frac{1}{N_{\text{samples}}}\cdot j, \qquad 0\le j\le N_{\text{samples}}-1\,.$$

Step 2: Estimate $g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha^j\right)\right]$ with (9) and (10), where α^j is computed as described in the previous step.

Step 3: At each frame i, obtain

$$\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha^j\right)\right]\right.\right\}\cdot\ln\left\langle\Pr\left\{P_i^{(1)}\left|g\left[\hat{P}|\Phi_{\hat{P}}\left(\alpha^j\right)\right]\right.\right\}\right\rangle,$$

where α^j is defined in step 1.

Step 4: At each frame i, estimate $\alpha_{\text{optimal}}(i)$ by making use of the same polynomial approximation based method mentioned in subsection III.1.

Step 5: Finally, score $\hat{P}$ is computed as

$$\hat{P}\left(A_{\text{optimal}}\right)=\frac{1}{I}\sum_{i=1}^{I}\left\{\left[1-\alpha_{\text{optimal}}\left(i\right)\right]\cdot P_i^{(1)}+\alpha_{\text{optimal}}\left(i\right)\cdot P_i^{(2)}\right\}. \qquad (20)$$

3. Estimation of α_{optimal} by Averaging $\alpha_{\text{optimal}}(i)$

If the optimal α is estimated on a frame-by-frame basis as described in subsection III.2, A_{optimal} could also be obtained by making $A_{\text{optimal}}=\left\{\alpha_i=\overline{\alpha_{\text{optimal}}(i)}\right\}$, with $1\le i\le I$, where $\overline{\alpha_{\text{optimal}}(i)}$ is defined as

$$\overline{\alpha_{\text{optimal}}(i)}=\frac{1}{N}\sum_{i=1}^{N}\alpha_{\text{optimal}}\left(i\right)\,. \qquad (21)$$

Finally, score $\hat{P}$ is computed as

$$\hat{P}\left(\overline{\alpha_{\text{optimal}}\left(i\right)}\right)=\frac{1}{T}\sum_{i=1}^{N}\left[\left(1-\overline{\alpha_{\text{optimal}}\left(i\right)}\right)\cdot P_i^{(1)}+\overline{\alpha_{\text{optimal}}\left(i\right)}\cdot P_i^{(2)}\right]. \qquad (22)$$

IV. Experiments

This paper presents results with Yoho database [29]. The Yoho Speaker Verification Corpus (Linguistic Data Consortium) supports development, training, and testing of speaker verification systems that use limited vocabulary and free-text input. The vocabulary is composed of two-digit numbers spoken continuously in sets of three (for example, "62-31-53" or "sixty-two thirty-one fifty-three"). The database is divided into "enrollment" and "verification" segments; each segment contains data from all 138 speakers (106 males and 32 females). There are four enrollment sessions per speaker and each session contains 24 utterances. Each verification segment contains 10 sessions, and each session contains four utterances per speaker. The proposed technique is compared with the method to optimize a linear combination of classifiers described in [20], where the fusion weights are obtained a

priori by logistic regression [30]. Two classifiers are evaluated: Viterbi based score (VBS) and support vector machine (SVM). The database was divided in three groups: Yoho_A, Yoho_B, and Yoho_C. Yoho_A database, composed of 80 speakers (65 males and 15 females), is used for testing. Yoho_B, composed of 17 speakers (12 males and 5 females), was employed to estimate the optimal weights of the linear combination of classifiers according to [20]. Finally, Yoho_C, composed of 41 speakers (29 males and 12 females), was used to train the speaker independent (SI) model required by the Viterbi based classifier according to [31] and the non-target class in the support vector classifier [32], [33].

1. Classifiers

The two standard SV techniques employed were VBS and SVM.

A. VBS Classifier

The input signal is processed with the forced-Viterbi algorithm in order to estimate the normalized log-likelihood $\log L(X)$ [31]:

$$\log L(X) = \sum_{i=1}^{I} \{\log \Pr(x_i \mid \lambda_{SD}) - \log \Pr(x_i \mid \lambda_{SI})\}, \quad (23)$$

where X is the observation sequence $X=[x_1, x_2, \cdots, x_i, \cdots, x_I]$, $\Pr(x_i \mid \lambda_{SD})$ and $\Pr(x_i \mid \lambda_{SI})$ represent the likelihood related to the speaker dependent (λ_{SD}) and speaker independent (λ_{SI}) hidden Markov models (HMMs), respectively. Both models, λ_{SD} and λ_{SI}, correspond to the sequence of triphone HMMs that compose the testing sequence X. The normalized log-likelihood Log $L(X)$ is divided by the number of frames, I, in the verification utterance: $\log L(X)' = \log L(X)/I$. It is worth highlighting that λ_{SD} is computed with the enrolling data pronounced by each client, and λ_{SI} is estimated as explained above.

B. SVM Classifier

A support vector machine is a two-class classifier constructed with a sum of a kernel function $k(\cdot,\cdot)$ [32], [33]:

$$f(y) = \sum_{i=1}^{N} \gamma_i \cdot t_i \cdot K(y, y_i) + b_i, \quad (24)$$

where t_i are targets, γ_i is the Lagrange multiplier of the i-th constraint, and $\sum_{i=1}^{N} \gamma_i \cdot t_i = 0$. Parameters y_i are the support vectors obtained from the training set. The target values are 1 or −1 depending on class 1 or class 2, respectively. For classification, a class decision is based upon whether the value, $f(y)$, is above or below a threshold.

2. Experimental Setup

Enrolling and verification utterances are decomposed as a sequence of triphones. Thirty-three cepstral coefficients are computed per frame: the frame energy plus ten static coefficients and their first and second time derivatives. As mentioned above, there are 300 frames per utterance on average. In VBS, the HMMs are trained with the Viterbi algorithm. Each triphone is modeled with a three-state left-to-right HMM topology without skip-state transition, with one multivariate Gaussian density per state in speaker-dependent models, and eight multivariate Gaussian densities per state in the speaker-independent model. Both models employ diagonal covariance matrices. SVM is applied as follows: 100 and 4000 codewords for the client and non-target classes, respectively; a K-means algorithm is used to estimate the codebooks; and, a polynomial kernel is employed [31]. FA and FR error rates are computed with Yoho_A as follows: FR curves are estimated with 80 speakers × 40 verification signals per client = 3,200 signals; and FA curves are obtained with 79 impostors × 3 verification signals per impostor × 80 users = 18,960 experiments.

The baseline result is given by VBS that is more accurate than SVM, as shown in Table 1. The lower accuracy of SVM should be due to the fact that SVM was not specifically proposed to address the problem of TD-SV. According to the method proposed here, the linear combination of classifiers defined in (3) is optimized on an utterance-by-utterance basis, with no a priori PDFs or pre-estimated parameters or weights, by using only Yoho_A data. As described in section III, three strategies for the on-line optimization of linear combination of

Table 1. EER and area below DET curve with individual classifiers: VBS and SVM.

Classiffier	EER (%)	Area below DET curve
VBS	0.78	3.2
SVM	2.46	25.1

Table 2. EER vs. the polynomial approximation based optimization with the linear combination of classifiers VBS/SVM and AFDO according to (19).

(%)

$N_{samples}$	Polynomial order						
	3	4	5	6	7	8	9
6	0.57	0.6	0.6	-	-	-	-
8	0.57	0.58	0.57	0.61	0.61	-	-
10	0.56	0.59	0.6	0.59	0.59	0.58	0.59

Table 3. EER and area below the DET curve with the polynomial approximation based optimization (N_{sample} and polynomial order equal to 10 and 3, respectively). The linear combination of classifier VBS/SVM is optimized according to FIO (15); FDO (17); and AFDO (19).

On-line optimization of the linear combination of classifiers using mutual information	EER	Area below the DET curve
FIO according to (15)	0.69	3.1
FDO according to (17)	0.75	3.1
AFDO according to (19)	0.56	1.9

Table 4. Off-line optimization of the linear combination of classifiers VBS/SVM with LLR, logistic regression linear combination [20]. The fusion weights are estimated with Yoho_B and tested with Yoho_A.

Off-line optimization logistic regression linear combination	EER	Area below the DET curve
VBS/SVM	0.63	2.14

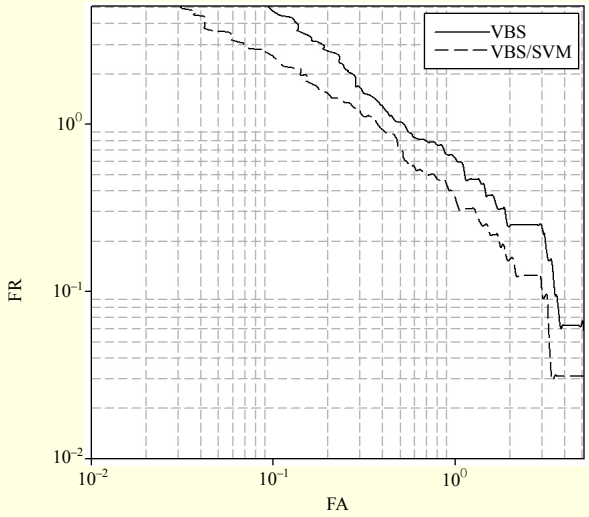

Fig. 6. DET curves of VBS in solid line and the dashed line shows the linear combination of classifiers VBS/SVM, optimized with AFDO according to (19).

classifiers are presented: frame independent optimization (FIO) according to (18), frame dependent optimization (FDO) as defined in (20), and averaged frame dependent optimization (AFDO) as indicated in (22). The polynomial approximation is implemented with the linear least square fitting method [34]. Results are show in Tables 2 to 4 and Fig. 6.

V. Discussion

Table 2 shows results with AFDO and VBS/SVM where (12) is optimized as described in subsection III.3. As can be seen in Table 2, there is a wide range of values of N_{sample} and polynomial order where AFDO and VBS/SVM give a significant improvement in EER when compared with the most accurate classifier (VBS). This result validates the optimization criterion and numerical approximation. When compared with VBS, N_{sample} and polynomial order equal to 10 and 3, respectively, provide reductions in EER as high as 28%. Table 3 presents results with FIO, FDO, and AFDO to optimize the linear combination of classifiers VBS/SVM. According to Table 2, FIO, FDO, and AFDO can lead to reductions in EER equal to 12%, 4%, and 28%, respectively, when compared with VBS (the most accurate classifier). The improvement in accuracy of the method presented in this paper can also be observed in Fig. 6 where the DET curves from VBS/SVM, when optimized according to AFDO (22), and from VBS are compared. The same behavior is observed in Table 3. When compared with VBS, VBS/SVM optimized with FIO, FDO, and AFDO leads to reductions in the area below the DET curve equal to 3%, 3%, and 41%, respectively.

The improvement in accuracy resulted from the optimization of the linear combination of classifiers with logistic linear regression (LLR) [20], is presented in Table 4. When compared with LLR, AFDO leads to reductions in EER equal to 11%. This result strongly validates the proposed approach. In contrast to ordinary multiclassifier fusion methods, the mutual information based method does not require any a priori distribution or pre-estimated weights. Consequently, the comparison scenario is the worst possible where the training-testing matching conditions are highly satisfied in the experiment reported here. As a result, the method described in this paper is especially promising to tackle the problem of multiclassifier fusion when the training-testing matching hypothesis loses accuracy.

Despite the fact that SVM gives a much higher EER than VBS, the former is able to improve the accuracy of the latter. This is a very interesting result. The EER achieved by VBS is comparable to the state-of-the-art EER found in the literature (0.5 – 0.8) [35]-[39]. As a result, MCS becomes an interesting method to reduce the error rate of the most accurate classifier given a set of experts. Observe that to improve the accuracy of an optimized classifier is not a trivial task. In addition, MCS is a prominent subdiscipline in the field of pattern recognition [40], although it is an incipient topic in speech processing.

VI. Conclusion

The problem of optimization of linear combination of classifiers using information theory is addressed and tested in a

text-dependent speaker verification task. Linear combination is one of the most popular approximations for Bayesian classifier fusion in pattern recognition. A mutual information criterion is proposed to optimize the classifier fusion on an utterance-by-utterance basis. The method does not require a priori distributions or pre-estimated weights with training data. As a consequence, the proposed technique is able to capture the dependence of the classifiers on the input signal. This is especially promising to address the problem of multiclassifier fusion when the training-testing matching hypothesis is not valid. The idea is to improve the most accurate classifier by taking into account the additional information provided by the second classifier. The results presented here show that the on-line optimization of linear combination of classifiers can lead to reductions in EER as high as 28% and 11% when compared, respectively, with the most accurate classifier and with the combination according to [20]. The comparison scenario is the worst possible for the presented scheme due to the fact that the training-testing matched conditions adopted in the experiments are highly satisfied. It is worth highlighting that the proposed on-line multiclassifier optimization approach is applicable to any pattern recognition problem. The presented method does not make use of any consideration about training-testing environments and leads to highly significant reductions in EER. Finally, to evaluate the proposed approach with mismatch between training-testing conditions, to apply the presented scheme to the combination of three or more classifiers and to other classification problems, and to improve the computational efficiency of the optimization procedure are proposed for future research.

References

[1] J. Kittler et al., "On Combining Classifiers," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, 1998, pp. 226-239.

[2] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, NY: John Wiley and Sons, 1973.

[3] L.I. Kuncheva, "Using Measures of Similarity and Inclusion for Multiple Classifier Fusion by Decision Templates," *Fuzzy Sets and Systems*, vol. 122, no. 3, 2001, pp. 401-407.

[4] L.I. Kuncheva, "A Theoretical Study on Six Classifier Fusion Strategies," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 24, no. 2, 2002, pp. 281-286.

[5] Y. Chen, C.Y. Wan, and L.S. Lee, "Entropy-Based Feature Parameter Weighting for Robust Speech Recognition," *Int. Conf. Acoustics, Speech and Signal Process.*, 2006, Toulouse, France.

[6] B. Fassinut-Mombot and J.-B. Choquel, "A New Probabilistic and Entropy Fusion Approach for Management of Information Sources," *Information Fusion*, vol. 5, 2004, pp. 35-47.

[7] M. Saerens and F. Fouss, "Yet Another Method for Combining Classifiers Outputs: A Maximum Entropy Approach," *Lecture Notes in Computer Science*, vol. 3077, 2004, pp. 82-91.

[8] H.J. Kang and S.W. Lee, "Combining Classifers Based on Minimization of a Bayes Error Rate," *Proc. 5th Int. Conf. Document Anal. Recognition*, 1999, Bangalore, India, pp. 398-401.

[9] G. Gravier et al., "Maximum Entropy and Mce Based HMM Stream Weight Estimation for Audio-Visual Asr," *ICASSP*, 2002.

[10] S. Tamura, K. Iwano, and S. Furui, "Toward Robust Multimodal Speech Recognition," *LKR*, 2005, Tokyo, Japan, pp. 163-166.

[11] A.C.S. Chung and H.C. Shen, "Dependence in Sensory Data Combination," *Int. Conf. Intelligent Robots and Systems,* 1998, Victoria, BC, Canada, pp. 1676-1681.

[12] Y. Zhou and H. Leung, "Minimum Entropy Approach for Multisensor Data Fusion," *IEEE Signal Process. Workshop Higher-Order Statistics*, 1997, pp. 336-339.

[13] A.L. Berger, S.A. Della Pietra, and V.J. Della Pietra, "A Maximum Entropy Approach to Natural Language Processing," *Computational Linguistics*, vol. 22, 1996, pp. 42-71.

[14] B. Nasersharif and A. Akbari, "Improved HMM Entropy for Robust Sub-Band Speech Recognition," *Eusipco*, 2005.

[15] M. Matton et al., "Maximum Mutual Information Training of Distance Measures for Template Based Speech Recognition," *Int. Conf. Speech and Computer*, 2005, pp. 511-514.

[16] M.K. Omar et al., "An Evaluation of Using Mutual Information for Selection of Acoustic-Features Representation of Phonemes for Speech Recognition," *ICSLP*, 2002, pp. 2129-2132.

[17] K.R. Farrel, "Text-Dependent Speaker Verification Using Data Fusion," *ICASSP*, 1995, pp. 349-352.

[18] K.R. Farrel et al., "Sub-Word Speaker Verification Using Data Fusion Methods," *IEEE Workshop Neural Networks Signal Process.*, 1997, pp. 531-540.

[19] B. Yegnanarayana et al., "Combining Evidence from Source, Suprasegmental and Spectral Features for a Fixed-Text Speaker Verification System," *IEEE Trans. Speech Audio Process*., vol. 13, no. 4, 2005, pp. 578-582.

[20] N. Brümmer et al., "Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006," *IEEE Trans. Audio, Speech, and Language Process*., vol. 15, no. 7, 2007, pp. 2072-2084.

[21] M.F. Benzeghiba and H. Bourlard, "Hybrid HMM/Ann and Gmm Combination for User-Customized Password Speaker Verification," *ICASSP,* 2003, pp. 225-228.

[22] M.W. Mak, M.C. Cheung, and S.Y. Kung, "Robust Speaker Verification from Gsm-Transcoded Speech Based on Decision Fusion and Feature Transformation," *ICASSP*, 2003, pp. 745-748.

[23] D. Genoud et al., "*Combining Methods to Improve Speaker Verification*," *ICSLP*, 1996, pp. 1756-1759.

[24] F. Huenupan et al., "Confidence Based Multiple Classifier Fusion

in Speaker Verification," *Pattern Recognition Lett.,* vol. 29, no. 7, 2008, pp. 957-966.
[25] F. Bimbot et al., "A Tutorial on Text-Independent Speaker Verification," *EURASIP J. Applied Signal Process.,* 2004, pp. 430-451.
[26] A.V. Lazo and P.N. Rathie, "On the Entropy of Continuous Probability Distributions," *IEEE Trans. Inf. Theory,* vol. IT-24, 1978, pp. 120-122.
[27] R. Gray, *Entropy and Information Theory*, NY: Springer-Verlag, 1990.
[28] C. Molina et al., "Unsupervised Re-Scoring of Observation Probability Based on Maximum Entropy Criterion by Using Confidence Measure with Telephone Speech," *Interspeech*, Australia, 2008, pp. 1016-1019.
[29] J. Campbell and A. Higgins, "YOHO Speaker Verification," *Linguistic Data Consortium*, 1994.
[30] S. Pigeon, P. Druyts, and P. Verlinde, "Applying Logistic Regression to the Fusion of the Nist'99 1-Speaker Submissions," *Digit. Signal Process.*, vol. 10, 2000, pp. 237-248.
[31] S. Furui, "Recent Advances in Speaker Recognition," *Pattern Recognition Letters,* vol. 18, 1997, pp. 859-872.
[32] C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery,* vol. 2, 1998, pp. 121-167.
[33] W.M. Campbell et al., "High-Level Speaker Verification with Support Vector Machines," *ICASSP*, 2004, Montréal, Canada, pp. 73-76.
[34] J. Rice, *Mathematical Statistics and Data Analysis*, Florence, Ky., USA: Brooks Cole, 1995, pp. 507-570.
[35] X. Dong and W. Zhaohui, "Speaker Recognition Using Continuous Density Support Vector Machines," *Electron. Letters*, vol. 37, no. 17, 2001, pp. 1099-1101.
[36] Y. Gu and T. Thomas, "A Hybrid Score Measurement for HMM-Based Speaker Verification," *ICASSP*, 1999, pp. 317-320.
[37] Z. Lei, Y. Yang, and Z. Wu, "An Ubm-Based Reference Space for Speaker Recognition," *Int. Conf. Pattern Recognition*, 2006, pp. 318-321.
[38] Y. Liu, M. Russell, and M. Carey, "*The Role of Dynamic Features in Text-Dependent and Independent Speaker Verification*," *ICASSP*, 2006, pp. 669-672.
[39] B.L. Pellom and J.H.L. Hansen, "An Efficient Scoring Algorithm for Gaussian Mixture Model Based Speaker Identification," *IEEE Signal Process. Lett.,* vol. 5, no. 11, 1998, pp. 281-284.
[40] L.I. Kuncheva and C. Whitaker, "Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy," *Machine Learning*, vol. 51, no. 2, 2003, pp. 181-207.

Fernando Huenupán received his BSc in electronic engineering from the Universidad de La Frontera, Temuco, Chile, in 2004. He is currently pursuing the PhD degree in electrical engineering at the Universidad de Chile, Santiago, Chile. Since 2003, he has been a research student at the Speech Processing and Transmission Laboratory where he is currently working on speaker verification and multiple classifier systems. Mr. Huenupán is the co-author of two journal articles and five conference papers. His research interests include robustness in speaker verification and multiple classifier systems.

Néstor Becerra Yoma received his PhD from the University of Edinburgh, UK, in 1998. He received his MSc and BSc from UNICAMP (Campinas State University), Sao Paulo, Brazil, all of them in electrical engineering, in 1993 and 1986, respectively. In 1998 and 1999, he was a post-doctoral researcher at UNICAMP and a full-time professor at Mackenzie University in Sao Paulo, Brazil. From 2000 to 2002, he was an assistant professor at the Department of Electrical Engineering, Universidad de Chile, in Santiago. At this university he is currently lecturing on telecommunications and speech processing, and working on robust speech recognition/speaker verification, computer aided pronunciation training and language learning, dialogue systems, and voice over IP. At the Universidad de Chile he established the Speech Processing and Transmission Laboratory to do research on speech technology applications on the Internet and telephone lines. Dr. Becerra Yoma has been an associate professor since 2003 and is the co-author of 20 journal articles and 27 conference papers. His research interests include speech processing, language learning, real time Internet protocols, QoS, and usability evaluation of human-machine interfaces. Professor Becerra Yoma is a member of the IEEE and the International Speech Communication Association.

Claudio Garretón received his MSc and BSc in electrical engineering from the Universidad de Chile, Santiago, Chile, in 2007 and 2005, respectively. He is currently pursuing the PhD in electrical engineering at the Universidad de Chile. Since 2005, he has been a research student at the Speech Processing and Transmission Laboratory where he is currently working on techniques for channel distortion and noise canceling in speech recognition, and speaker verification and second language learning. In the last four years, he has co-authored two journal articles and six conference papers. His general research interests include channel and noise robustness in speech applications on the Internet and telephone lines.

Carlos Molina received his MSc and BSc in electrical engineering from the Universidad de Chile, Santiago, Chile, in 2005 and 2003, respectively. He is currently pursuing his PhD in electrical engineering at the Universidad de Chile, Santiago, Chile. Since 2003, he has been a research student at the Speech Processing and Transmission Laboratory where he is currently working on noise canceling and speaker adaptation techniques for speech recognition, and computer aided pronunciation training and second language learning. He is the co-author of six journal articles and first author of three conference papers. His research interests include robustness in automatic speech recognition and second language learning. Mr. Molina is a student member of the IEEE and the International Speech Communication Association.