Person identification using multiple cues

Roberto Brunelli and **Daniele Falavigna**
Istituto per la Ricerca Scientifica e Tecnologica
I-38050 Povo, Trento, ITALY
e-mail: brunelli@irst.it falavi@irst.it

Abstract— **This paper presents a person identification system based on acoustic and visual features. The system is organized as a set of non-homogeneous classifiers whose outputs are integrated after a normalization step. In particular, two classifiers based on acoustic features and three based on visual ones provide data for an integration module whose performance is evaluated. A novel technique for the integration of multiple classifiers at an hybrid rank/measurement level is introduced using HyperBF networks. Two different methods for the rejection of an unknown person are introduced. The performance of the integrated system is shown to be superior to that of the acoustic and visual subsystems. The resulting identification system can be used to log personal access and, with minor modifications, as an identity verification system.**

Keywords— **template matching, robust statistics, correlation, face recognition, speaker recognition, learning, classification.**

1. Introduction

The identification of a person interacting with computers represents an important task for automatic systems in the area of information retrieval, automatic banking, control of access to security areas, buildings and so on. The need for a reliable identification of interacting users is obvious. At the same time it is well known that the security of such systems is too often violated in every day life. The possibility to integrate multiple identification cues, such as password, identification card, voice, face, fingerprints and the like will, in principle, enhance the security of a system to be used by a selected set of people.

This paper describes in detail the theoretical foundations and design methodologies of a person recognition system that is part of MAIA, the integrated AI project under development at IRST [26].

Previous works about speaker recognition [30], [16] have proposed methods for classifying and combining acoustic features and for normalizing [27], [22] the various classifier scores. In particular, score normalization is a fundamental step when a system is required to confirm or reject the identity given by the user (user verification): in this case, in fact, the identity is accepted or rejected according to a comparison with a preestimated threshold. Since the integration of voice and images in an identification system is a new concept, new methods for both classifier normalization and integration were investigated. Effective ways for rejecting an unknown person by considering score and rank information and for comparing images with improved similarity measures are proposed. A simple method for adapting the acoustic models of the speakers to a real operating environment also was developed.

The speaker and face recognition systems are decomposed into two and three single feature classifiers respectively. The resulting five classifiers produce non-homogeneous lists of scores that are combined using two different approaches. In the first approach, the scores are normalized through a robust estimate of the location and scale parameters of the corresponding distributions. The normalized scores are then combined using a weighted geometric average and the final identification is accepted or rejected according to the output of a linear classifier, based on score and rank information derived from the available classifiers. Within the second approach, the problem of combining the normalized outputs of multiple classifiers and of accepting/rejecting the resulting identification is considered a learning task. A mapping from the scores and ranks of the classifiers into the interval $(0, 1)$ is approximated using an HyperBF network. A final threshold is then introduced based on cross-validation. System performance is evaluated and discussed for both strategies. Because of the novelty of the problem, standard data-bases for system training and test are not yet available. For this reason, the experiments reported in this paper are based on data collected at IRST. A system implementation operating in real-time is available and was tested on a variety of IRST researchers and visitors. The joint use of acoustic and visual features proved effective in increasing system performance and reliability.

The system described here represents an improvement over a recently patented identification system based on voice and face recognition [6], [9]. The two systems differ in many ways: in the latter the speaker and face recognition systems are not further decomposed into classifiers, the score normalization does not rely on robust statistical techniques and, finally, the rejection problem is not addressed.

The next sections will introduce the speaker and face recognition systems. The first approach to the integration of classifiers and the *linear* accept/reject rule for the final system identification are then discussed. Finally, the novel rank/measurement level integration strategy using an HyperBF network is introduced with a detailed report on system performance.

2. Speaker recognition

The voice signal contains two types of information: individual and phonetic. They have mutual effects and are difficult to separate; this represents one of the main problems in the development of automatic speaker and speech recognition systems. The consequence is that speaker recognition systems perform better on speech segments having specific phonetic contents while speech recognition systems provide higher accuracy when tuned on the voice of a particular speaker. Usually the acoustic parameters for a speech/speaker recognizer are derived by applying a bank of band-pass filters to adjacent short time windows of the input signal. The energy outputs of the filters, for various frames, provide a good domain representation. Figure 1 gives an example of such an analysis. The speech waveforms correspond to utterances of the Italian digit 4 (/kwat:ro/) by two different speakers. The energy outputs of a 24 triangular band-pass filter bank are represented below the speech waveforms (darker regions correspond to higher energy values).

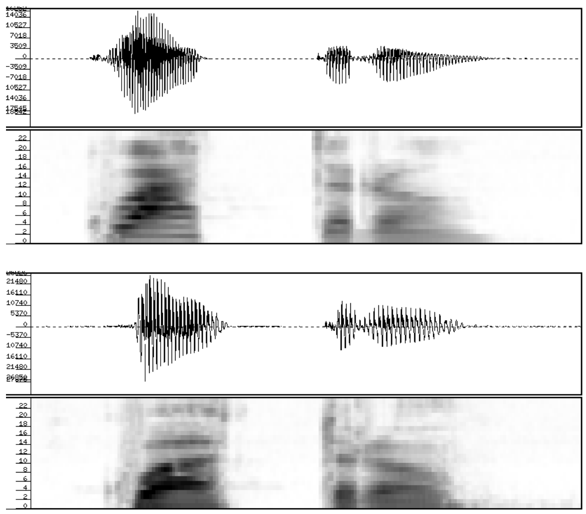

Fig. 1. Acoustic analysis of two utterances of the digit 4 (/kwat:ro/) by two different speakers.

In the past years several methods and systems for speaker identification [13], [16] were proposed that perform more or less efficiently depending on the text the user is required to utter (in general, systems can be distinguished into *text dependent* or *text independent*), the length of the input utterance, the number of people in the reference database and, finally, the time interval between test and training recordings.

For security applications, it is desirable that the user utter a different sentence during each interaction. The content of the utterance can then be verified to ensure that the system is not cheated by prerecorded messages. For this work, a text independent speaker recognition system based on Vector Quantization (VQ) [28] was built. While it cannot yet verify the content of the utterance, it can be modified (using supervised clustering or other techniques) to obtain this result.

A block diagram of the system is depicted in Figure 2. In the system, each reference speaker is represented by means of two sets of vectors (codebooks) that describe his/her acoustic characteristics. During identification, two sets of acoustic features (static and dynamic), derived from the short time spectral analysis of the input speech signal, are classified by evaluating their distances from the prototype vectors contained in the speaker codebook couples. In this way, two lists of scores are sent to the integration module. In the following both the spectral analysis and vector quantization techniques will be described in more detail (see also [21] and a reference book such as [23]).

Since the power spectrum of the speech signal decreases as frequency increases a preemphasis filter that enhances the higher frequencies is applied to the sampled input signal. The transfer function of the filter is $H(z) = 1/(1 - 0.95 \cdot z^{-1})$.

The preemphasized signal, $x(n)$, $1 \leq n \leq N$, is subdivided into frames $y_t(n)$, $1 \leq t \leq T$, having length L. Each frame is obtained by multiplying $x(n)$ by an Hamming window $h_t(n)$:

$$
\begin{aligned}
y_t(n) &= x(n) \cdot h_t(n) \; 1 \leq t \leq T = \frac{N}{S} & (1)\\
h_t(n) &= 0.54 - 0.46 \cdot \cos\left(\frac{2\pi(n - tS)}{L}\right) & (2)\\
& \quad tS - \frac{L}{2} < n \leq tS + \frac{L}{2}
\end{aligned}
$$

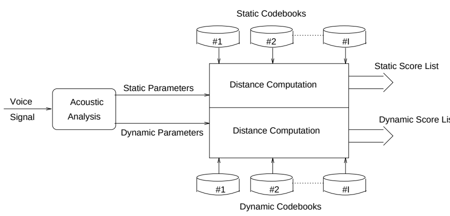

Fig. 2. The speaker recognition system based on Vector Quantization.

In the equation above L represents the length, in samples, of the Hamming window and S is the analysis step (also expressed in samples). For the system L and S were chosen to correspond to 20 ms and 10 ms respectively.

The signal is multiplied by an Hamming window (raised cosine) to minimize the sidelobe effects on the spectrum of the resulting sequence $y_t(n)$.

The acoustic analysis of each frame is performed as follows:

1. the power spectrum of the sequence $y_t(n)$ is evaluated;
2. a bank of $Q = 24$ triangular filters, spaced according to a logarithmic scale (Mel scale), is applied to the power spectrum and the energy outputs s_{tq}, $1 \leq q \leq Q$, from each filter are evaluated;
3. the Mel Frequency Cepstrum Coefficients (MFCC) [29], θ_{tp}, $1 \leq p \leq P = 8$, are computed, from the filterbank outputs, according to the following equation:

$$\theta_{tp} = \sum_{q=1}^{Q}[log(s_{tq})] \cos\left[p\left(q - \frac{1}{2}\right)\frac{\pi}{Q}\right]; \quad (3)$$

   the MFCCs are arranged into a vector, $\boldsymbol{\theta}_t$, which is called static, since it refers to a single speech frame;
4. to account for the transitional information contained in the speech signal a linear fit is applied to the components of 7 adjacent MFCCs; the resulting regression coefficients are arranged into a vector that is called dynamic;
5. a binary variable is finally evaluated that allows marking the frame as *speech* or *background noise*; this parameter is computed by means of the algorithm described in [12].

The Mel scale is motivated by auditory analysis of sounds. The inverse Fourier transform of the log-spectrum (cepstrum) provides parameters that improves performance at both speech and speaker recognition [23], [29]. Furthermore, the Euclidean distance between two cepstral vectors represents a good measure for comparing the corresponding speech spectra. The static and dynamic 8-dimensional vectors related to windows marked as background noise are not considered during both system training and testing. As previuosly said, VQ is used to design the static and dynamic codebooks of a given reference speaker, say the i^{th} one. Starting from a set of training vectors (static or dynamic) $\boldsymbol{\Theta}_i = \{\boldsymbol{\theta}_{i1}, \ldots, \boldsymbol{\theta}_{iK}\}$, derived from a certain number of utterances, the objective is to find a new set $\boldsymbol{\Psi}_i = \{\boldsymbol{\psi}_{i1}, \ldots, \boldsymbol{\psi}_{iM}\}$, with $M \ll K$, that represents well the acoustic characteristics of the given speaker. To do this a clustering algorithm, similar to that described in [21], is applied to the $\boldsymbol{\Theta}_i$ set. The algorithm makes use of an iterative procedure that allows determination of codebook centroids, $\boldsymbol{\Psi}_i$, by minimizing their average distance, $D(\boldsymbol{\Theta}_i, \boldsymbol{\Psi}_i)$, from the training vectors:

$$D(\boldsymbol{\Theta}_i, \boldsymbol{\Psi}_i) = \frac{1}{K}\sum_{k=1}^{K} \min_{m=1}^{M}[d(\boldsymbol{\theta}_{ik}, \boldsymbol{\psi}_{im})] \quad (4)$$

The distance $d(\boldsymbol{\theta}_{ik}, \boldsymbol{\psi}_{im})$ is defined as follows:

$$d(\boldsymbol{\theta}_{ik}, \boldsymbol{\psi}_{im}) = (\boldsymbol{\theta}_{ik} - \boldsymbol{\psi}_{im})^t \boldsymbol{W}^{-1} (\boldsymbol{\theta}_{ik} - \boldsymbol{\psi}_{im}) \quad (5)$$

In the equation above t denotes transpose and $\boldsymbol{W}$ is the covariance matrix of the training vectors. The matrix $\boldsymbol{W}$ is estimated from the training data of all the speakers in the reference database. This matrix was found to be approximately diagonal, so that only the diagonal elements are used to evaluate distances.

In the recognition phase the distances, D_{Si}, D_{Di}, between the static and dynamic vector sequences, derived from the input signal, and the corresponding speaker codebooks are evaluated and sent to the integration module.

If $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_T\}$ is the static (or dynamic) input sequence and $\boldsymbol{\Psi}_i$ is the i^{th} static (or dynamic) codebook, then the total static (or dynamic) distance will be:

$$D_i(\boldsymbol{\Theta}, \boldsymbol{\Psi}_i) = \frac{1}{T}\sum_{t=1}^{T} \min_{m=1}^{M}[d(\boldsymbol{\theta}_t, \boldsymbol{\psi}_{im})] \quad 1 \leq i \leq I \quad (6)$$

where I is the total number of speakers in the reference database.

To train the system, 200 isolated utterances of the Italian digits (from 0 to 9) were collected for each reference user. The recordings were realized by means of a Digital Audio Tape (DAT): the signal on the DAT tape, sampled at 48 kHz, was downsampled to 16 kHz, manually end-pointed, and stored on a computer disk. The speech training material was analyzed and clustered as previously described. As demonstrated in [28], system performance depends on both input utterance length and codebook size; preliminary experiments have suggested that the speaker, to be identified, should utter a string of at least 7 digits in a continuous way and in whatever order. In the reported experiments the number of digits was kept equal to 7 and the codebook size was set to $M = 64$ because higher values did not improve recognition accuracy. Furthermore, if input signal duration is too short, the system requires the user to repeat the digit string.

To evaluate integrated system performance (see section 4.1) the reference users interacted 3 times with the system during 3 different sessions. The test sessions were carried out in an office environment using an ARIEL board as acquisition channel. Furthermore the test phase was performed about five months after the training recordings. Due to both the different background noise and acquisition conditions between training and test, the codebooks must be adapted.

Adaptation means designing a new codebook, starting from a given one, that better resembles the acoustic characteristics of both the operating environment and the acquisition channel. Adaptation should also take into account variations in time of the speaker's voice (intraspeaker variations). Adaptation requires the use of few utterances to modify the codebook as it is not necessary to design it from scratch (this would require at least 30-40 seconds of speech). In our case, the adaptation vectors are derived from the digit strings uttered by the users during a single test session. The dynamic codebooks were not adapted since they represent temporal variations of the speech spectra and therefore they are less sensitive to both intraspeaker voice variability and acquisition channel variations.

The adaptation process of the i^{th} codebook, $\boldsymbol{C}_i$ can be summarized as follows:

1. the mean vectors $\boldsymbol{\mu}_i$ and $\boldsymbol{\nu}_i$ of the adaptation vectors and of the given codebook respectively are evaluated;
2. the difference vector $\boldsymbol{\Delta}_i = \boldsymbol{\mu}_i - \boldsymbol{\nu}_i$ is evaluated;
3. the vectors of $\boldsymbol{C}_i$ are shifted by a quantity equal to $\boldsymbol{\Delta}_i$ obtaining a new set $\boldsymbol{C}_i^{'} = \{\boldsymbol{c}_{i1} + \boldsymbol{\Delta}_i, \ldots, \boldsymbol{c}_{iM} + \boldsymbol{\Delta}_i\}$; the set $\boldsymbol{C}_i^{'}$ is placed in the region of the adaptation vectors;
4. the adaptation vectors are clustered using the set $\boldsymbol{C}_i^{'}$ as initial estimate of the centroids; therefore a new set of centroids $\boldsymbol{O}_i = \{\boldsymbol{o}_{i1}, \ldots, \boldsymbol{o}_{iM}\}$ and the corresponding cell occupancies $N_i = \{n_{i1}, \ldots, n_{iM}\}$ are evaluated;
5. the adapted codebook $\boldsymbol{\Psi}_i$ is obtained according to the following equation:

$$\boldsymbol{\psi}_{im} = \boldsymbol{c}_{im}^{'} + (1 - e^{-\alpha n_{im}})(\boldsymbol{o}_{im} - \boldsymbol{c}_{im}^{'}) \quad 1 \leq m \leq M \tag{7}$$

   In the equation above the parameter n_{im} determines the fraction of deviation vector $\delta_{im} = (\boldsymbol{o}_{im} - \boldsymbol{c}_{im}^{'})$, that has to be summed to the initial centroid $\boldsymbol{c}^{'}_{im}$. Eqn. 7 is a simple method to modify the centroids of a codebook according to the number of data available for their estimates. δ_{im} can be zero when the utterance used for adaptation does not contain sounds whose spectra are related to the m-th centroid.

For the system, α was chosen equal to 0.1. The two shifts applied by the adaptation procedure can be interpreted as follows:

1. Δ_i, the major shift, accounts for environment and channel variations with respect to training;
2. δ_{im}, the minor shift, accounts for intraspeaker voice variations in time.

3. Face recognition

Person identification through face recognition is the most familiar among the possible identification strategies. Several automatic or semiautomatic systems were realized since the early seventies - albeit with varying degree of success. Different techniques were proposed, ranging from the geometrical description of salient facial features to the expansion of a digitized image of the face on an appropriate basis of images (see [8] for references). The strategy used by the described system is essentially based on the comparison, at the pixel level, of selected regions of the face [8]. A set of regions, respectively encompassing the eyes, nose, and mouth of the user to be identified are compared with the corresponding regions stored in the database for each reference user (see Figure 3). The images should represent a frontal view of the user face without marked expressions. As will be clear from the detailed description, these constraints could be relaxed at the cost of storing a higher number of images per user in the database. The fundamental steps of the face recognition process are the following:

Fig. 3. The highlighted regions represent the templates used for identification.

1. acquisition of a frontal view of the user face;
2. geometrical normalization of the digitized image;
3. intensity normalization of the image;
4. comparison with the images stored in the database.

The image of the user face is acquired with a CCD camera and digitized with a frame grabber.

To compare the resulting image with those stored in the database, it is necessary to register the image: it has to be translated, scaled, and rotated so that the coordinates of a set of reference points take corresponding *standard* values. As frontal views are considered, the centers of the pupils represent a *natural* set of control points that can be located with good accuracy. Eyes can be found through the following steps:

1. locate the (approximate) symmetry axis of the face;
2. locate the left/right eye by using an eye template for which the location of the pupil is known; if the confidence of the eye location is not sufficiently high, declare failure (the identification system will use only acoustic information);
3. achieve translation, scale and rotation invariance by fixing the origin of the coordinate system at the midpoint of the interocular segment

and the interocular distance and the direction of the eye-to-eye axis at predefined values.

Under the assumption that the user face is approximately vertical in the digitized image, a good estimate of the coordinate S of the symmetry axis is given by

$$S = \mathrm{median}\{P_V(|\mathcal{I} * K_V|)_i\} \tag{8}$$

where $*$ represent convolution, $\mathcal{I}$ the image, K_V the convolution kernel $[-1, 0, 1]^t$, P_V the vertical projection whose index i runs over the columns of the image. The face can then be split vertically into two, slightly overlapping parts containing the left and right eye respectively. The illumination under which the image is taken can impair the template matching process used to locate the eye. To minimize this effect a filter, $\mathcal{N}(\mathcal{I})$, is applied to image $\mathcal{I}$:

$$\mathcal{N} = \begin{cases} \mathcal{N}' & \text{if } \mathcal{N}' \leq 1 \\ 2 - \frac{1}{\mathcal{N}'} & \text{if } \mathcal{N}' > 1 \end{cases} \tag{9}$$

where

$$\mathcal{N}' = \frac{\mathcal{I}}{\mathcal{I} * K_{G(\sigma)}} \tag{10}$$

and $K_{G(\sigma)}$ is a Gaussian kernel whose σ is related to the expected interocular distance Δ_{ee}. The arithmetic operations act on the values of corresponding pixels. The process mapping $\mathcal{I}$ into $\mathcal{N}$ reduces the influence of ambient lighting while keeping the necessary image details. This is mainly due to the removal of linear intensity gradients that are mapped to the constant value 1. Extensive experiments, using ray-tracing and texture-mapping techniques to generate synthetic images under a wide range of lighting directions have shown that the *local contrast* operator of eqn. (9) exhibits a lower illumination sensitivity than other operators such as the laplacian, the gradient magnitude or direction [5] and that there is an optimal value of the parameter σ (approximately equal to the iris radius).

The same filter is applied to the eye templates. The template matching process is based on the algorithm of hierarchical correlation proposed by Burt [11]. Its final result is a map of correlation values: the center of gravity of the pixels with maximum value representing the location of the eye. Once the two eyes have been located, the confidence of the localization is expressed by a coefficient, C_E, that measures the symmetry of the eye positions with respect to the symmetry axis, the horizontal alignment and the scale relative to that of the eye templates:

$$C_E = \frac{(C_l + C_r)}{2} \frac{\min(C_l, C_r)}{\max(C_l, C_r)} e^{-\frac{(s-1)^2}{2\sigma_s^2}} e^{-\frac{\Delta\theta^2}{2\sigma_\theta^2}} \tag{11}$$

where C_l and C_r represent the (maximum) correlation value for the left/right eye, s the interocular distance expressed as a multiple of the interocular distance of the eyes used as templates, $\Delta\theta$ represents the angle of the interocular axis with respect to the horizontal axis while σ_θ and σ_s represent tolerances on the deviations from the prototype scale and orientation.

The first factor in the RHS of eqn. (11) is the average correlation value of the left and right eye: the higher it is the better the match with the eye templates. The second factor represents the symmetry of the correlation values and equals 1 when the two values are identical. The third and fourth factors allow weighing the deviation from both the assumed scale and (horizontal) orientation of the interocular axis, respectively. The parameters of the Gaussians, σ_s and σ_θ, were determined by the analysis of a set of interactions.

If the value of C_E is too low, the face recognition system declares failure and the identification proceeds using the acoustic features alone. Otherwise, the image is translated, scaled and rotated to match the location of the pupils to that of the database images. In the reported experiments the interocular distance was set equal to 28 pixels. Alternative techniques for locating eyes are reported in [17], [32]. Due to the geometrical standardization, the subimages containing the eyes, nose, and mouth are approximately characterized by the same coordinates in every image. These regions are extracted from the image of the user face and compared in turn to the corresponding regions extracted from the database entries, previously filtered according to eqns. (9)(10). Let us introduce a similarity measure $\mathcal{C}$ based on the computation of the L_1 norm of a vector $\|\boldsymbol{x}\|_{L_1} = \sum_i |x_i|$ and on the corresponding distance $d_{L_1}(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|_{L_1}$:

$$\mathcal{C}(\boldsymbol{x}, \boldsymbol{y}) = 1 - \frac{d_{L_1}(\boldsymbol{x}, \boldsymbol{y})}{\|\boldsymbol{x}\|_{L_1} + \|\boldsymbol{y}\|_{L_1}} \tag{12}$$

The L_1 distance of two vectors is mapped by $\mathcal{C}(\cdot, \cdot)$ into the interval $[0, 1]$, higher values representing smaller distances. This definition can be easily adapted to the comparison of images. For the comparison to be useful when applied to real images, it is necessary to *normalize* the images so that they have the same average intensity μ and standard deviation (or scale) σ. The latter is particularly sensitive to values far from the average μ so that the scale of the image intensity distribution can be better estimated by the following quantity:

$$\sigma_{L_1} = \frac{1}{n} \sum_{i=1}^{n} |x_i - \mu| \tag{13}$$

where the image is considered as a one dimensional vector $\boldsymbol{x}$. The matching of an image B to an image A can then be quantified by the maximum value of $\mathcal{C}(A, B)$, obtained by sliding the smaller of the

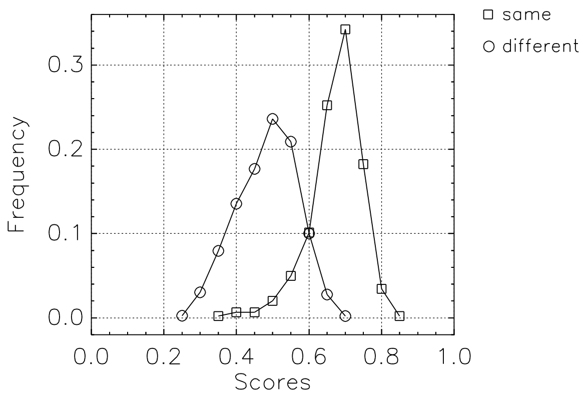

Fig. 4. The distribution of the correlation values for corresponding features of the same person and of different people.

two images over the larger one. A major advantage of the image similarity computed according to eqn. (12) over the more common estimate given by the cross-correlation coefficient [1], based on the L_2 norm, is its reduced sensitivity to small amounts of unusually high differences between corresponding pixels. These differences are often due to noise or image specularities such as iris highlights. A detailed analysis of the similarity measure defined in eqn. (12) is given in [7]. An alternative technique for face identification is reported in [31]. Let us denote with $\{U_{km}\}_{m=1,\ldots,p_k}$ the set of images available for the k^{th} user. A comparison can now be made between a set of regions of the unknown image $\mathcal{N}$ and the corresponding regions of the database images. The regions currently used by the system correspond to the eyes, nose and mouth. A list of similarity scores is obtained for each region $\mathcal{F}_\alpha$ of image U_{km}:

$$\{s_{k\alpha}\} = \{\max_m \mathcal{C}(\mathcal{R}_\alpha(\mathcal{N}), \mathcal{F}_\alpha(U_{km})\} \qquad (14)$$

where $\mathcal{R}_\alpha(\mathcal{N})$ represents a region of $\mathcal{N}$ containing $\mathcal{F}_\alpha$ with a frame whose size is related to the interocular distance. The lists of matching scores corresponding to eyes, nose, and mouth are then available for further processing. The distribution of the correlation values for corresponding features of the same person and of different people are reported in Figure 4.

Integration with the scores derived from the acoustic analysis can now be performed with a single or double step process. In the first case, the two acoustic and the three visual scores are combined simultaneously, while in the second the acoustic and visual scores are first combined separately and the final score is given by the integration of the outputs of the speaker and face recognition systems (see [9] for an example of the latter). The next section will introduce two single-step integration strategies for classifiers working at the measurement level.

4. Integration

The use of multiple cues, such as face and voice, provides in a natural way the information necessary to build a reliable, high performance system. Specialized subsystems can identify (or verify) each of the previous cues and the resulting outputs can then be combined into a unique decision by some integration process. The objective of this section is to describe and evaluate some integration strategies. The use of multiple cues for person recognition proved beneficial for both system performance and reliability[1].

A simplified taxonomy of multiple classifier systems is reported in [33]. Broadly speaking, a classifier can output information at one of the following levels:

- *the abstract level:* the output is a subset of the possible identification labels, without any qualifying information;
- *the rank level:* the output is a subset of the possible labels, sorted by decreasing confidence (which is not supplied);
- *the measurement level:* the output is a subset of labels qualified by a confidence measure.

The level at which the different classifiers of a composite system work clearly constrains the ways their responses can be merged. The first of the following sections will address the integration of the speaker/face recognition systems at the measurement level. The possibility of rejecting a user as unknown will then be discussed. Finally, a novel, hybrid level approach to the integration of a set of classifiers will be presented.

4.1 Measurement level integration

The acoustic and visual identification systems already constitute a multiple classifier system. However, both the acoustic and visual classifiers can be further split into several subsystems, each one based on a single type of feature. In our system, five classifiers were considered (see secs. 2, 3) working on the static, dynamic acoustic features, and on the eyes, nose and mouth regions.

A critical point in the design of an integration procedure at the measurement level is that of measurement normalization. In fact, the responses of the different classifiers usually have different scales (and possibly offsets), so that a sensible combination of the outputs can proceed only after the scores are properly normalized. As already detailed, the outputs of the identification systems are not homogeneous: the acoustic features provide distances while the visual ones provide correlation values. A

[1] Two aspects of reliability are critical for a person identification system: the first is the ability of rejecting a user as unknown, the second is the possibility of working with a reduced input, such as only the speech signal or the face image.

first step towards the normalization of the scores is to reverse the sign of distances, thereby making them concordant with the correlation values: the higher the value, the more similar the input patterns. Inspection of the score distributions shows them to be markedly unimodal and roughly symmetrical. A simple way to normalize scores is to estimate their average values and standard deviations so that distributions can be translated and rescaled in order to have zero average and unit variance. The values can then be forced into a standard interval, such as $(0,1)$, by means of an hyperbolic tangent mapping. The normalization of the scores can rely on a fixed set of parameters, estimated from the score distributions of a certain number of interactions, or can be adaptive, estimating the parameters from the score distribution of the current interaction. The latter strategy was chosen mainly because of its ability to cope with variations such as different speech utterance length without the need to re-estimate the normalization parameters.

The estimation of the location and scale parameters of the distribution should make use of *robust* statistical techniques [18], [20]. The usual arithmetic average and standard deviation are not well suited to the task: they are highly sensitive to outlier points and could give grossly erroneous estimates. Alternative estimators exist that are sensitive to the main bulk of the scores (i.e. the central part of a unimodal symmetric distribution) and are not easily misled by points in the extreme tails of the distribution. The median and the Median Absolute Deviation (MAD) are examples of such location and scale estimators and can be used to reliably normalize the distribution of the scores. However, the median and the MAD estimators have a low efficiency relative to the usual arithmetic average and standard deviation. A class of robust estimators with higher efficiency was introduced by Hampel under the name of *tanh-estimators* and is used in the current implementation of the system (see [18] for a detailed description). Therefore each list of scores $\{S_{ij}\}_{i=1,\ldots,I}$ from classifier j, being I the number of people in the reference database, can be transformed into a normalized list by the following mapping:

$$S'_{ij} = \frac{1}{2}\left[\tanh\left(0.01\,\frac{(S_{ij}-\mu_{tanh})}{\sigma_{tanh}}\right)+1\right] \in (0,1) \tag{15}$$

where μ_{tanh} and σ_{tanh} are the average and standard deviation estimates of the scores $\{S_{ij}\}_{i=1,\ldots,I}$ as given by the Hampel estimators. An example of distributions of the resulting normalized scores is reported in Figure 5 for each of the five features used in the classification.

In the following formulas, a subscript index i_m indicates the m^{th} entry within the set of scores sorted

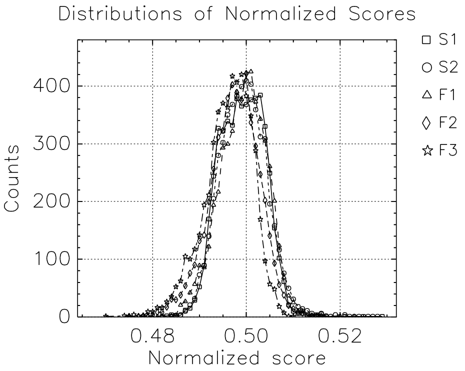

Fig. 5. The density distribution of the normalized scores for each of the classifiers: $S1$, $S2$ represent the static and dynamic speech scores while $F1$, $F2$ and $F3$ represent the eyes, nose and mouth scores respectively.

by decreasing value. The normalized scores can be integrated using a weighted geometric average:

$$S_i = \left(\prod_j {S'_{ij}}^{w_j}\right)^{1/\sum_j w_j} \tag{16}$$

where the weights w_j represent an estimate of the score dispersion in the right tail of the corresponding distributions:

$$w_j = \frac{S'_{i_1 j} - 0.5}{S'_{i_2 j} - 0.5} - 1.0 \tag{17}$$

The main reason suggesting the use of geometric average for the integration of scores relies on probability: if we assume that the features are independent the probability that a feature vector corresponds to a given person can be computed by taking the product of the probabilities of each single feature. The normalized scores could then be considered as equivalent to probabilities. Another way of looking at the geometric average is that of *predicate conjunction* using a continuous logic [2], [3]. The weights reflect the importance of the different features (or predicates). As defined in eqn. (17), each feature is given an importance proportional to the separation of the two best scores. If the classification provided by a single feature is ambiguous, it is given low weight. A major advantage of eqn. (16) is that it does not require a detailed knowledge of how each feature is distributed (as would be necessary when using a Bayes approach). This eases the task of building a system that integrates many features.

The main performance measure of the system is the percentage of persons correctly recognized. Performance can be further qualified by the average value of the following ratio R_x:

$$R_x = \frac{S'_x - S'_{i_I}}{\max_{i\neq x}(S'_i) - S'_{i_I}} \quad 1 \le i \le I. \tag{18}$$

Feature	Recognition (%)	R
Voice	**88**	**1.14**
Static	77	1.08
Dynamic	71	1.08
Face	**91**	**1.56**
Eyes	80	1.25
Nose	77	1.25
Mouth	83	1.28
ALL	**98**	**1.65**

TABLE I

THE RECOGNITION PERFORMANCE AND AVERAGE SEPARATION RATIO R FOR EACH SINGLE FEATURE AND FOR THEIR INTEGRATION. DATA ARE BASED ON 164 REAL INTERACTIONS AND A DATABASE OF 89 USERS.

The ratio R_x measures the *separation* of the correct match S'_x from the wrong ones. This ratio is invariant against the scale and location parameters of the integrated score distribution and can be used to compare different integration strategies (weighted/unweighted geometric average, adaptive/fixed normalization). The weighted geometric average of the scores adaptively normalized exhibits the best performance and separation among the various schemes on the available data.

Experiments have been carried out using data acquired during 3 different test sessions. Of the 89 persons stored in the database, 87 have interacted with the system in one or more sessions. One of the three test sessions was used to adapt the acoustic and visual databases (in the last case the images of the session were simply added to those available); therefore, session 1 was used to adapt session 2 and session 2 to adapt session 3. As the number of interactions for each adapted session is 82 the total number of test interactions was 164. As each session consisted of 82 interactions, the system was tested on 164 interactions. The recognition performance and the average value of R_x for the different separate features and for their integration are reported in Table I.

4.2 Rejection

An important capability of a classifier is to reject input patterns that cannot be classified in any of the available classes with a sufficiently high degree of confidence. For a person verification system, the ability to reject an impostor is critical. The following paragraphs introduce a rejection strategy that takes into account the level of agreement of all the different classifiers in the identification of the best candidate.

A simple measure of confidence is given by the integrated score itself: the higher the value, the higher the confidence of the identification. Another is given by the difference of the two best scores: it is a measure of how sound the ranking of the best candidate is. The use of independent features (or feature sets) also provides valuable information in the form of the rankings of the identification labels across the classifier ouputs: if the pattern does not belong to any of the known classes, its rank will vary significantly from classifier to classifier. On the contrary, if the pattern belongs to one of the known classes, rank agreement will be consistently high. The average rank and the rank dispersion across the classifiers can then be used to quantify the *agreement* of the classifiers in the final identification. The confidence in the final identification can then be quantified through several measures. The decision about whether the confidence is sufficient to accept the system output can be based on one or several of them. In the proposed system, a linear classifier, based on absolute and relative scores, ranks and their dispersion, will be used to accept/reject the final result. The following issues will be discussed:

1. degree of dependence of the features used;
2. choice of the confidence measures to be used in the accept/reject rule;
3. training and test of the linear classifier used to implement the accept/reject rule.

As a preliminary step, the independence of the features used in the identification process will be evaluated. It is known that the higher the degree of independence, the higher the information provided to the classifier. Let us consider a couple of features X and Y. Let $\{(x_i, y_i)\}_{i=1,\ldots,I}$ represent the corresponding normalized scores. They can be considered as random samples from a population with a bivariate distribution function. Let A_i be the rank of x_i among $x_1, \ldots, x_I$ when they are arranged in descending order, and B_i the rank of y_i among $y_1, \ldots, y_I$ defined similarly to A_i. Spearman's rank correlation [19] is defined by:

$$r_s = \frac{\sum_i (A_i - \bar{A})(B_i - \bar{B})}{\sqrt{\sum_i (A_i - \bar{A})^2}\sqrt{\sum_i (B_i - \bar{B})^2}} \qquad (19)$$

where $\bar{A}$ and $\bar{B}$ are the average values of $\{A_i\}$ and $\{B_i\}$, respectively. An important characteristic of rank correlation is its non-parametric nature. To assess the independence of the features it is not necessary to know the bivariate distribution from which the (X_i, Y_i) are drawn, since the distribution of their ranks is known, under the assumption of independence. It turns out that

$$t = r_s \sqrt{\frac{I-2}{1-r_s^2}} \qquad (20)$$

	S1	S2	F1	F2	F3
S1	1 (1)	0.64 (1.0)	0.06 (0.7)	0.08 (0.8)	0.04 (0.7)
S2		1.00 (1.0)	0.03 (0.6)	0.07 (0.8)	0.03 (0.6)
F1			1.00 (1.0)	0.11 (0.8)	0.10 (0.8)
F2				1.00 (1.0)	0.50 (1.0)
F3					1.00 (1.0)

TABLE II

The rank correlation value of the couples of features. The parenthesized values represent the significance of the correlation. S1 and S2 represent the dynamic and static acoustic features respectively; F1, F2, F3 represent the eyes, nose and mouth.

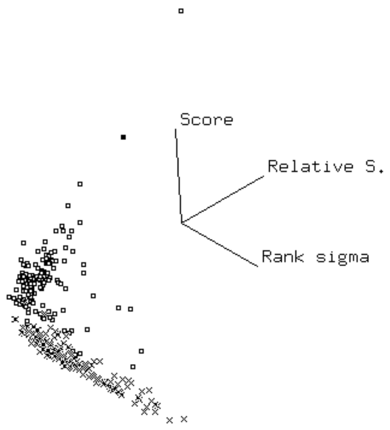

Fig. 6. Let us represent the match with the database entries by means of the integrated score, the standard deviation of the rankings across the different features and the normalized ratio of the first to second best integrated score. The resulting three dimensional points are plotted and marked with a $\square$ if they represent a correct match or with a $\times$ if the match is incorrect. Visual inspection of the resulting point distribution shows that the two classes of points can be separated well by using a plane.

is distributed approximately as a Student's distribution with $I-2$ degrees of freedom [19]. It is then possible to assess the dependence of the different features used by computing the rank correlation of each couple and by testing the corresponding significance. Results for the features used in the system developed are given in Table II.

The acoustic features are clearly correlated, as well as the nose and mouth features. The latter correlation is due to the overlapping of the nose and mouth regions, which was found to be necessary in order to use facial regions characterized by the same coordinates for the whole database. Acoustic and visual features are independent, as could be expected.

The feasibility of using a linear classifier was investigated by looking at the distribution of acceptable and non-acceptable[2] best candidates in a 3D space whose coordinates are the integrated score, a normalized ratio of the first to second best score and the standard deviation of the rankings. As can be seen in Figure 6 a linear classifier seems to be appropriate.

The full vector $\mathbf{d} \in \mathcal{R}^{18}$ used as input to the linear classifier is given by:

1. the integrated score, S_1, of the best candidate;
2. the normalized ratio of the first to the second best integrated score:
$$R = \frac{S_{i_1} - 0.5}{S_{i_2} - 0.5}\,; \tag{21}$$
3. the minimum and maximum ranks of the first and second final best candidates (4 entries);
4. the rank standard deviation of the first and second final best candidates (2 entries);
5. the individual ranks of the first and second final best candidates (10 entries).

To train the linear classifier the following procedure was used. A set of positive examples $\{\boldsymbol{p}_i\}$ is derived from the data relative to the persons correctly classified by the system. A set of negative examples $\{\boldsymbol{n}_j\}$ is given by the data relative to the best candidate when the system did not classify the user correctly. The set of negative examples can be augmented by the data of the best candidate when the correct entry is removed from the database, thereby simulating the interaction with a stranger. The linear discriminant function defined by the vector $\mathbf{w}$ can be found by minimizing the following error:

$$\begin{aligned} E &= \alpha \sum_i \left(1 - \frac{1 - e^{-(\sum_{k=1}^{l} w_k p_{ik} + w_{l+1})}}{1 + e^{-(\sum_{k=1}^{l} w_k p_{ik} + w_{l+1})}}\right)^2 \qquad (22) \\ &+ \beta \sum_j \left(1 + \frac{1 - e^{-(\sum_{k=1}^{l} w_k n_{jk} + w_{l+1})}}{1 + e^{-(\sum_{k=1}^{l} w_k n_{jk} + w_{l+1})}}\right)^2 \end{aligned}$$

where α and β represent the weight to be attributed to false negatives and to false positives respectively and $l = 18$ is the dimensionality of the input vectors. When $\alpha = \beta = 1$, E represents the output error of a linear perceptron with a symmetric sigmoidal unit.

Final acceptance or rejection of an identification, associated to a vector $\mathbf{d}$, is done according to the simple rule:

$$\sum_{i=1}^{l} w_i d_i + w_{l+1} > 0 \quad \text{accept} \tag{23}$$

$$\sum_{i=1}^{l} w_i d_i + w_{l+1} \leq 0 \quad \text{reject} \tag{24}$$

[2]Non-acceptable best candidates derive from two sources: misclassified users from real interactions and best candidates from *virtual* interactions, characterized by the removal of the user entry from the data base.

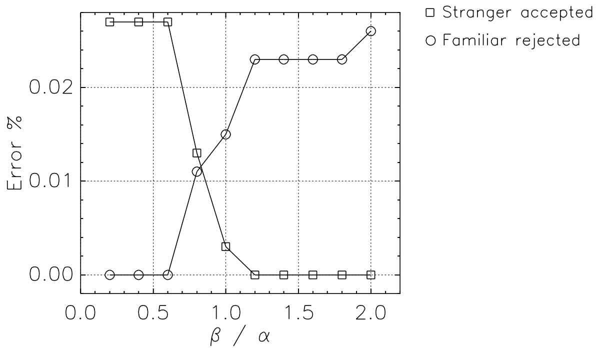

Fig. 7. System performance when false positives and false negatives are weighted differently.

	Error (%)
Face	
Stranger accepted	4.0
Familiar rejected	8.0
Familiar misrecog.	0.5
Voice	
Stranger accepted	14.0
Familiar rejected	27.0
Familiar misrecog.	1.0
Integrated	
Stranger accepted	**0.5**
Familiar rejected	**1.5**
Familiar misrecog.	**0.0**

TABLE III

Error rates of the subsystems and of the complete system when a rejection threshold is introduced. Data are based on the subset of interactions for which both face and speech data were available (155 out of 164).

Note that the LHSs of eqns. (23)(24) represent the signed distance, in arbitrary units, of point $\mathbf{d}$ from the plane defined by $\mathbf{w}$ that divides the space into two semispaces. Points lying in the correct semispace contribute to E inversely to their distance from plane $\mathbf{w}$. Points lying near the plane contribute with α or β while points lying in the wrong semispace and at great distance from the discriminating plane contribute with 2α or 2β. If the two classes of points are linearly separable it is possible to drive E to zero (see [14], [24]). A stochastic minimization algorithm [4], [10] was used to minimize E.

When the system is required to work in a strict mode (no errors allowed, that is, no strangers accepted), $\beta >> \alpha$ should be considered in the training phase. Note that a similar discriminant function can be computed for each of the recognition subsystems (i.e. face recognition and voice recognition), thereby enabling the system to reject an identification when it is not sufficiently certain even when not all of the identification cues are available. The training/test of the classifier followed a leave-one-out strategy to maximize the number of data available in the training phase [15]. The classifier is trained by using all but one of the available samples and tested on the excluded one. The performance of the classifier can be evaluated by excluding in turn each of the available samples and averaging the classification error.

In the reported experiments, the available examples were grouped per interacting user. The leave-one-out method was then applied to the resulting 87 sets (the number of users that interacted with the system) to guarantee the independence of the training and test sets.

Each set was used in turn for testing, leaving the remaining 86 for training. The results are reported in Table III. A complete operating characteristic curve for the integrated performance shown in Table III is reported in Figure 7 where the stranger-accepted and familiar-rejected rates at different β/α ratios are plotted.

Similar experiments were run on the acoustic and visual features separately and are also reported in Table III. The results show that the use of the complete set of features provides a relevant increase in reliable performance over the separate subsystems.

4.3 Hybrid level integration

In this sub-section, a hybrid rank/measurement level at which multiple classifiers can be combined will be introduced. The approach is to reconstruct a mapping from the sets of scores, and corresponding ranks, into the set $\{0, 1\}$. The matching to each of the database entries, as described by a vector of five scores and the corresponding ranks should be mapped to 1, if it corresponds to the correct label, and to 0 otherwise. The reconstruction of the mapping proceeds along the following steps:

1. find a set of positive and negative examples;
2. choose a parametric family of mappings;
3. choose the set of parameters for which the corresponding mapping minimizes a suitable error measure over the training examples.

Another way to look at the reconstruction of the mapping is to consider the problem as a learning task, where, given a set of acceptable and non acceptable inputs, the system should be able to appropriately classify unseen data.

Let $\{C_j\}$ be the set of classifiers. Each of them associates to each person X some numerical data X_j that can be considered a vector. By comparison with the i^{th} database entry, a normalized similarity score S_{ij} can be computed. Each score S_{ij} can be associated to its rank r_{ij} in the list of scores produced by classifier C_j. The output of each classifier can then be regarded as a list of couples $\{(S_{ij}, r_{ij})\}_{i=1,\ldots,I}$ where I represents the number of people in the reference database. A mapping is sought $\mathcal{L}_{01}$ such that:

$$\mathcal{L}_{01}(S_{i1}, r_{i1}, \cdots, S_{i5}, r_{i5}) = \begin{cases} 1 & \text{if label}(X) = i \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

If, after mapping the list of scores, more than a label qualifies, the system rejects the identification. It is possible to *relax* the definition of $\mathcal{L}_{01}$ by letting the value of the mapping span the whole interval $[0, 1]$. In this way the measurement level character of the classification can be retained. The new mapping $\mathcal{L}$ can be interpreted as a fuzzy predicate. The following focuses on the fuzzy variant, from which the original formulation can be obtained by introducing a threshold ω:

$$\mathcal{L}_{01} = \theta(\mathcal{L}(\mathbf{x}_i) - \omega) \tag{26}$$

where $\theta(\cdot)$ is the Heavyside unit-step function and $\mathbf{x}_i = (S_{i1}, r_{i1}, \cdots, S_{i5}, r_{i5})$ is a ten dimensional vector containing the feature normalized matching scores and corresponding ranks. The goal is to approximate the characteristic function of the correct matching vectors as a sum of Gaussian bumps. Therefore the search for $\mathcal{L}$ is conducted within the following family of functions:

$$\mathcal{L}(\boldsymbol{x}; \{(c_\alpha, \mathbf{t}_\alpha)\}_\alpha, \Sigma) = \sigma(\sum_\alpha c_\alpha G(|\mathbf{x} - \mathbf{t}_\alpha|_\Sigma)) \tag{27}$$

where

$$G(x) = e^{-x^2} \tag{28}$$

$$\sigma(x) = \frac{1}{1 + e^{4(x-1/2)}} \tag{29}$$

$$|\mathbf{x} - \mathbf{t}_\alpha|_\Sigma = \sqrt{(\mathbf{x} - \mathbf{t}_\alpha)^T \Sigma^{-1} (\mathbf{x} - \mathbf{t}_\alpha)} \tag{30}$$

Σ^{-1} being a diagonal matrix with positive entries, $\boldsymbol{x}$, $\boldsymbol{t}_\alpha \in \mathcal{R}^{10}$ and $c_\alpha \in \mathcal{R}$. The approximating function can be represented as an HyperBF network [25] whose topology is reported in Figure 8.

The sigmoidal mapping is required to ensure that the co-domain be restricted to the interval $(0, 1)$. The location $\boldsymbol{t}_\alpha$, shape Σ and height c_α of each bump are chosen by minimizing the following error measure:

$$E = \sum_i [y_{ij} - \sigma(\sum_\alpha c_\alpha G(|\mathbf{x}_{ij} - \mathbf{t}_\alpha|_\Sigma))]^2 \tag{31}$$

where $\{(\mathbf{x}_{ij}, y_{ij})\}_i$ is a set of examples (points at which the value of the mapping to be recovered is known). The first subscript i denotes the database

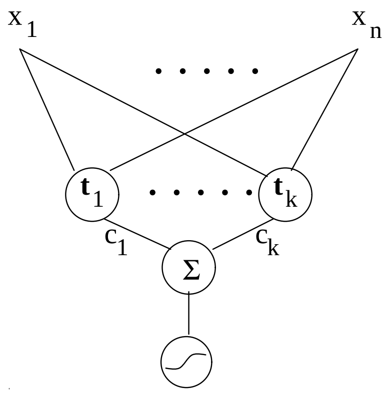

Fig. 8. The function used to approximate the mapping from the score/rank domain into the interval $(0, 1)$ can be represented as an HyperBF network.

entry from which $\mathbf{x}_{ij}$ is derived and the second subscript j represents the example.

The required value of the mapping at $\mathbf{x}_{ij}$ is 1 when i is the correct label (*class*) for the j-th example and 0 otherwise. The error measure E is minimized over the parameter space $\{(c_\alpha, \mathbf{t}_\alpha)\}_\alpha, \Sigma$ by means of a stochastic algorithm with adaptive memory [10]. The number of free parameters involved in the minimization process dictates the use of a large set of examples. As a limited number of real interactions was available, a leave-one-out strategy was used for training and testing the system as for the linear classifier previously described. From each of the available user-system interactions, a virtual interaction was derived by removing from the database the entry of the interacting user, thereby simulating an interaction with a stranger. For each interaction j

1. the vector corresponding to the correct database entry provides a positive example;
2. the vectors of the first ten, non correct, entries of the real interaction (as derived from sorting the integrated scores of Section 4.1) and the vectors of the first ten entries of the virtual interaction provide the negative examples.

The reason for using only the first ten non correct entries is that the matching scores decay quickly with rank position in the final score list and additional examples would not provide more information. Data from different interactions of the same user were then grouped. The resulting set of examples was used to generate an equal number of different training/testing set pairs. Each set was used in turn for testing, leaving the remaining ones for training. The problem of matching the number of free parameters in the approximation function to the complexity of the problem was solved by testing the performance of networks with increasing size. For each network size, a value for threshold ω of eqn. (26) was computed to minimize the total error defined as the sum of the percentage of accepted

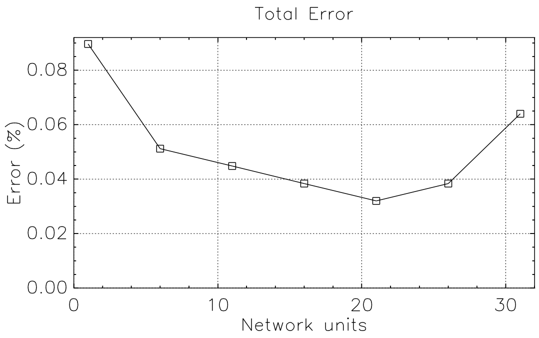

Fig. 9. The total error achieved by networks with different number of units. The total error is computed by summing the percentage of accepted strangers, misrecognized and rejected database people. For each net size a threshold was chosen to minimize the cumulative error.

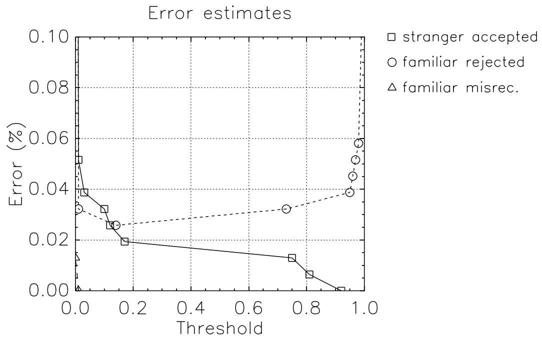

Fig. 10. Error percentages as a function of the rejection threshold for a Gaussian based expansion.

Errors	ω_0	$(\omega_0+\omega_1)/2$	ω_1
Stranger accepted (%)	0.5	0.5	**0.0**
Familiar rejected (%)	**3.0**	**3.0**	3.5
Familiar misrecog. (%	0.0	0.0	0.0

TABLE IV
The performance of the system when using an HyperBF network with 21 units to perform score integration.

strangers, misrecognized and rejected database persons. In Figure 9, the total error is reported as a function of the network size. Note that the threshold is computed on the test set, so that it gives an optimistic estimate.

To obtain a correct estimate of system performance, a cross-validation approach was used for the net giving the best (optimistic) total error estimate. Let $[\omega_0, \omega_1]$ be the interval over which the total error assumes its minimum value (see Figure 10). The threshold value can be chosen as:

- ω_0 favouring acceptance over rejection;
- $(\omega_0+\omega_1)/2$;
- ω_1 favouring rejection over acceptance.

The resulting performance is reported in Table IV. Note that using ω_1 the system was able to reject all of the strangers, which is the ultimate requirement for a reliable system, missing only 3.5% of the known users.

5. Conclusions

A system that combines acoustic and visual cues in order to identify a person has been described. The speaker recognition sub-system is based on Vector Quantization of the acoustic parameter space and includes an adaptation phase of the codebooks to the test environment. A different method to perform speaker recognition, which makes use of the Hidden Markov Model technique and pitch information, is under investigation.

A face recognition sub-system also was described. It is based on the comparison of facial features at the pixel level using a similarity measure based on the L_1 norm.

The two sub-systems provide a multiple classifier system. In the implementation described, 5 classifiers (2 acoustic and 3 visual) were considered. The multiple classifier operates in two steps. In the first one, the input scores are normalized using robust estimators of location and scale. In the second step, the scores are combined using a weighted geometric average. The weights are adaptive and depend on the score distributions. While normalization is fundamental to compensate for input variations (e.g. variations of illumination, background noise conditions, utterance length and of speaker voices), weighting emphasizes the classification power of the most reliable classifiers. The use of multiple cues, acoustic and visual, proved to be effective in improving performance. The correct identification rate of the integrated system is 98% which represents a significant improvement with respect to the 88% and 91% rates provided by the speaker and face recognition systems respectively. Future use of the Hidden Markov Model technique is expected to improve performance of the VQ based speaker recognizer.

An important capability of the multiple classifier itself is the rejection of the input data when they can not be matched with sufficient confidence to any of the database entries.

An accept/reject rule is introduced by means of a linear classifier based on measurement and rank information derived from the five recognition systems. A novel, alternative, approach to the integration of multiple classifiers at the hybrid rank/measurement level is also presented. The problem of combining the outputs of a set of classifiers is considered as a learning task. A mapping from the scores of the classifiers and their ranks into the interval $(0, 1)$ is approximated using an HyperBF network. A final rejection/acceptance threshold is then introduced using the cross-validation technique. System performance is evaluated on data acquired during real interactions of the users in the reference database. Performance of the two techniques is similar.

The current implementation of the system is working on an HP 735 workstation with a Matrox Magic frame grabber. In order to optimize system throughput, it relies on a hierarchical match with the face database.

The incoming picture, represented by a set of features is compared at low resolution with the complete database. For each person in the database, the most similar feature, among the set of available images, is chosen and the location of the best matching position stored. The search is then continued at the upper resolution level by limiting the search to the most promising candidates at the previous level.

These candidates are selected by integrating their face scores according to the procedure described in section 4.1. All available data must be used to secure a reliable normalization of the scores. However, new scores at higher resolution are computed only for a selected subset of persons and this constitutes a problem for the integration procedure. In fact, scores from image comparisons at different levels would me mixed, similarity values deriving from lower resolutions being usually higher. To overcome this difficulty, the scores from the previous level are reduce (scaled) by the highest reduction factor obtained comparing the newly computed scores to the corresponding previous ones.

The performance, measured on the datasets used for the reported experiments, does not decrease and the overall identification time (face and voice processing) is approximately 5 seconds.

The same approach, using codebooks of reduced size could be applied to the speaker identification system, thereby increasing system throughput. Adding a subject to the database is a simple task for both subsystems. This is due to the modularity of the databases, each subject being described independently of the others. The integration strategy itself does not require any update. The rejection and the combined identification/rejection procedures do require updating. However, the training of the linear perceptron and of the HyperBF network can be configured more as a refinement of a suboptimal solution (available from the previous database) than as the computation of a completely unknown set of optimal parameters. While the system, as presented, is mainly an identification system, a small modification transforms it into a verification system. For each person in the database it is possible to select a subset containing the most similar people (as determined by the *identification* system). When the user must be *verified* the *identification* system can be used using the appropriate subset, thereby limiting the computational effort, and verifying the identity of the user using the techniques reported in the paper.

Future work will have the purpose of further improving the global efficiency of the system with the investigation of more accurate and reliable rejection methods.

Acknowledgement

The authors would like to thank Dr. L. Stringa, Prof. T. Poggio and Prof. R. de Mori for valuable suggestions and discussions. The authors are grateful to the referees for many valuable comments.

References

[1] D. H. Ballard and C. M. Brown. *Computer Vision.* Prentice Hall, Englewood Cliffs, NJ, 1982.

[2] P. B. Bonissone and K. S. Decker. Selecting uncertainty calculi and granularity: an experiment in trading-off precision and complexity. In J. F. Lemmer L. N. Karnak, editor, *Uncertainty in Artificial Intelligence*, pages 217–247. North Holland, 1986.

[3] P. B. Bonissone, S. S. Gans, and K. S. Decker. Rum: A layered architecture for reasoning with uncertainty. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, pages 891–898, Milan, August 1987.

[4] R. Brunelli. On Training Neural Nets through Stochastic Minimization. Technical Report 9212-06, I.R.S.T, 1992. To appear on *Neural Networks.*

[5] R. Brunelli. Estimation of Pose and Illuminant Direction for Face Processing. A.I. Memo No. 1499, Massachusetts Institute of Technology, 1994.

[6] R. Brunelli, D. Falavigna, T. Poggio, and L. Stringa. A recognition system, particularly for recognising people. Patent Nr 93112738, 1993, Priority IT/11.08.92/IT TO920695.

[7] R. Brunelli and S. Messelodi. Robust Estimation of Correlation: an Application to Computer Vision. Technical Report 9310-05, I.R.S.T, 1993. to appear on *Pattern Recognition.*

[8] R. Brunelli and T. Poggio. Face Recognition: Features versus Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, 1993.

[9] R. Brunelli, T. Poggio, D. Falavigna, and L. Stringa. Automatic Person Recognition by Using Acoustic and Geometric Features. Technical Report 9307-43, I.R.S.T., 1993. to appear on *Machine Vision and Applications.*

[10] R. Brunelli and G. Tecchiolli. Stochastic minimization with adaptive memory. Technical Report 9211-14, I.R.S.T, 1992. To appear on *Journal of Computational and Applied Mathematics.*

[11] P. J. Burt. Smart sensing within a pyramid vision machine. *Proceedings of the IEEE*, 76(8):1006–1015, 1988.

[12] G. Carli and R. Gretter. A start-end point detection algorithm for a real-time acoustic front-end based on dsp32c vme board. In *Proceedings of ICSPAT*, pages 1011–1017, Boston, November 1992.

[13] G. R. Doddington. Speaker Recognition, Identifying People by Their Voices. *Proceedings of IEEE*, 73(11), 1985.

[14] R. O. Duda and P. E. Hart. *Pattern Recognition and Scene Analysis.* Wiley, New York, 1973.

[15] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1990.

[16] S. Furui. Cepstrum Analysis Technique for Automatic Speaker Verification. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 29(1):254–272, 1981.

[17] P. W. Hallinan. Recognizing Human Eyes. In *SPIE Proceedings*, volume 1570, pages 214–226, 1991.

[18] F. R. Hampel, P. J. Rousseeuw, E. M. Ronchetti, and W. A. Stahel. *Robust statistics: the approach based on influence functions.* John Wiley & Sons, 1986.

[19] R. V. Hogg and A. T. Craig. *Introduction to mathematical statistics*. Collier-Macmillan, 1978.

[20] P. J. Huber. *Robust Statistics*. Wiley, 1981.

[21] H. Gish J. Makhoul, S.Roucos. Vector Quantization in Speech Coding. *Proceedings of IEEE*, 73(11):1551–1588, 1985.

[22] T. Matsui and S. Furui. Similarity normalization method for speaker verification based on a posteriori probability. In *Proceedings of ESCA Workshop on Automatic Speaker Recognition Identification Verification*, pages 59–62, Martigny, Switzerland, April 1994.

[23] D. O'Shaughnessy. *Speech Communication*. Addison-Wesley, 1987.

[24] Yoh-Han Pao. *Adaptive Pattern Recognition and Neural Networks*. Addison-Wesley, Reading, MA, USA, 1989.

[25] T. Poggio and F. Girosi. Regularization algorithms for learning that are equivalent to multilayer networks. *Science*, 247:978–982, 1990.

[26] T. Poggio and L. Stringa. A Project for an Intelligent System: Vision and Learning. *International Journal of Quantum Chemistry*, 42:727–739, 1992.

[27] A. E. Rosenberg, J. DeLong, C. H. Lee, B. H. Juang, and F. K. Soong. The use of cohort normalized scores for speaker verification. In *Proceedings of ICSLP*, volume 1, pages 599–602, Banff, Canada, October 1992.

[28] A. E. Rosenberg and F. K. Soong. Evaluation of a Vector Quantization Talker Recognition System in Text Independent and Text Dependent Modes. *Computer Speech and Language*, 2(3–4):143–157, 1987.

[29] P. Melmerstein S. B. Davis. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuosly Spoken Sentences. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 28(4):357–366, 1980.

[30] F. K. Soong and A. E. Rosenberg. On the Use of Istantaneous and Transitional Spectral Information in Speaker Recognition. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 36(6):871–879, 1988.

[31] L. Stringa. Automatic Face Recognition using Directional Derivatives. Technical Report 9205-04, I.R.S.T, 1991.

[32] L. Stringa. Eyes Detection for Face Recognition. *Applied Artificial Intelligence*, 7:365–382, 1993.

[33] L. Xu, A. Krzyzak, and C. Y. Suen. Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3):418–435, 1992.