Phone-based filter parameter optimization of filter and sum robust speech recognition using likelihood maximization

Bahram Kouhi-Jelehkaran a,*, Hamidreza Bakhshi b, Farbod Razzazi a

a Department of Electrical Engineering, Islamic Azad University, Science and Research Branch, Tehran, Iran
b Department of Electrical Engineering, Shahed University, Tehran, Iran

A R T I C L E  I N F O

Article history:
Received 15 February 2009
Accepted 29 October 2009

Keywords:
Maximum likelihood filtering optimization
Microphone array processing
Robust speech recognition

A B S T R A C T

Because of noise and reverberation, accuracy of speech recognition systems decreases when the distance between talker and microphone increases. By the using of microphone arrays and appropriate filtering of received signals, the accuracy of recognizer can be increased. Many different methods for using microphone arrays have been proposed that can be classified into two main approaches: systems that perform in two independent stages of array processing and then recognition and systems that use array processing to generate a sequence of features which maximize the likelihood of generating the correct hypothesis in recognition phase. Following second approach, in this paper a new method for microphone array processing is proposed in which the parameters of array processing are adjusted in calibration phase based on phones used in language and maximum likelihood method. Optimized filter parameters are stored and used during recognition phase. A new modified Viterbi algorithm using optimal phone-based filter parameters is used for recognition phase. The proposed algorithm is analytically formulated and Persian language is used to find any improvement in speech recognition accuracy compared with results of delay and sum and utterance-based filter and sum algorithms. The results show 12.2% improvement in accuracy compared to utterance-based algorithm.

© 2009 Elsevier GmbH. All rights reserved.

1. Introduction

In current automatic speech recognition (ASR) systems, when the speech signals are captured using a close-talking microphone worn near the mouth of the speaker performance is reasonable [1,2]. But, in many applications, for reasons of safety or convenience, use of such a microphone is undesirable. In these cases, such as cars, meeting rooms, and information kiosks, a fixed microphone can be placed at some distance from the talker. As the distance between the talker and the microphone increases, the speech signal becomes degraded by the effects of additive noise and reverberation, which results in degraded speech recognition accuracy. The use of an array of microphones, rather than a single microphone, can compensate for this distortion in these distant-talking environments by providing spatial filtering to the speech signal and so effectively focusing attention in a desired direction [3].

In researches, many microphone array processing techniques which improve the quality of the output signal and increase the signal-to-noise ratio (SNR) have been proposed. Delay and sum beamforming is the simplest and most common microphone array processing method [4]. In this method, the signals received by the microphones in the array are adjusted with respect to each other in order to compensate for the path length differences between the speech source and each of the microphones. These signals are then weighted and added together. Signals that are not coincident with the speech source remain misaligned and are thus attenuated when the signals are combined. An extension of delay and sum beamforming is filter and sum beamforming, in which each microphone signal has an associated filter and the captured signals are filtered before they are combined.

Many methods for optimization of parameters of filters to improve certain specification of output signal for improved recognition accuracy have been proposed [1,5–7]. These methods can be divided into two main categories: in the first type, array processing parameters are optimized based on quality of output waveform and then processed array output is sent for recognition. In second type, the likelihood acoustic information of recognition stage is used to calibrate and optimize the parameters of array processing stage.

In some of the first type adaptive beamforming schemes, such as the generalized sidelobe canceler (GSC), the array parameters are updated based on samples or frames according to a specified criterion. Typical criteria used in first type adaptive beamforming include a response without distortion in the look direction or the minimization of the energy from all directions other than look
direction. In some cases, the array parameters can be calibrated to a particular environment or user prior to use, e.g., [6].

All of these microphone array processing methods were designed for signal enhancement, and so process incoming signals according to various signal-level criteria, e.g., minimizing the signal error, maximizing the SNR, or improving the quality as received by human listeners. These algorithms are used to generate the best output waveform, which then is sent as a single-channel input to a recognition system. This approach, shown in Fig. 1a, implicitly assumes that by producing a higher quality output waveform, we will get improved recognition performance.

In second approach, the microphone array processing problem is considered as one of finding the set of array parameters that maximizes the likelihood of the correct recognition hypothesis. As shown in Fig. 1b, the array processor and the speech recognizer are no longer considered two independent systems cascaded together, but rather two connected components of a single system, with the common goal of improved speech recognition accuracy. In these methods acoustic information from the recognition engine itself is used to optimize the parameters of a filter and sum beamformer.

These methods have several advantages over current array processing methods. First, by incorporating the statistical models of the recognizer into the array processing stage, we ensure that the processing enhances those signal components important for recognition accuracy without undue emphasis on less important components. Second, in contrast to conventional adaptive filtering methods, no assumptions about the interfering signals are made. Third, the proposed approach requires no a priori knowledge of the room configuration, array geometry, or source-to-sensor room impulse responses.

In this paper we introduce a new approach that uses the information from recognizer to optimize filter parameters on a phone base. In this method, instead of having single filter optimized by using calibration utterance, we use the calibration utterance to optimize sets of filters based on different phones that are used in language, and record these phone-based parameters of filters in a database and use them according to state under processing in Viterbi algorithm in recognition phase, as shown in Fig. 1c.

Comparative results with utterance-based and delay and sum methods showed that the proposed algorithm outperforms the traditional solutions.

2. Filter and sum array processing

Consider a microphone array as Fig. 2. Signals \( x_0(t) \) to \( x_{M-1}(t) \) after sampling are processed by FIR filters \( H_0(z) \) to \( H_{M-1}(z) \) and finally the output of array processing unit \( y[n] \) is produced by adding outputs of filters. If \( h_m[k] \) is considered as weight of \( m \) th microphone, then we have

\[
y[n] = \sum_{m=0}^{M-1} \sum_{k=0}^{L-1} h_m[k] x[n-k-r_m] \tag{1}
\]

3. Phone-based filter parameters optimization using maximum likelihood

In the utterance-based filter optimization method using likelihood maximization, filter coefficients are optimized in average [8,9]. This method has two limitations: First, because speech signal is wideband and there are large changes in frequency contents of an utterance, filter parameter estimation based on total utterance gives us only the average behavior of parameters. Second, when we are performing array processing for speech recognition applications speech recognition features are derived from the spectrum of the incoming speech signal. Therefore, it is worthwhile to examine the spectral response of the filters learned during calibration and because we are utilizing multiple spatially separated microphones, the filters also produce a spatial response. In utterance-based calibration, while we can look at the spectral and spatial response of filters in isolation, it is impossible to decouple the relative contributions of the two. In other words, while an interesting question might be whether the likelihood was maximized as a result of the spatial response of the array, by the overall filter and sum frequency response, or some contribution of them, there is no way to find the answer.

These limitations have been removed in phone-based calibration. Phone-based filters are optimized based on each phone which has a relatively constant frequency behavior and also the
filters can maximize likelihood spatially based on constant frequency content for each phone.

In suggested approach, filter coefficients are optimized locally by considering phones as the basic segment for assigning filter parameters for them. In addition, in utterance-based filter parameter optimization, analysis of filters frequency response shows that filters have a peak-value nature similar to speech signals. This shows a possible attempt to preserve the features of the speech spectrum important for accurate feature extraction, i.e., the formant frequencies. In phone-based filter parameter optimization, optimum parameters for each phone that has a relatively fixed behavior is preserved. The calibration utterance is made so that covers all phones used in language. These sets of filter parameters are stored in a database.

Speech recognizer finds the word string most likely to generate observation sequence of \( \mathcal{O} = \{o_1, o_2, \ldots, o_T\} \), as measured by statistical models. The feature vectors are function of both incoming speech signal and the array parameters. Recognition hypotheses are generated according to Bayes optimal classification as

\[
\hat{o} = \arg\max_o P(O|\mathbf{h}^i_f))/(o)^P(o)
\]

where \( O(\mathbf{h}^i_f) \) is the MFCC feature vector extracted from the output of filter and sum stage and dependence of \( O \) to the array processing parameters \( \mathbf{h}^i_f \) of a single phone \( f \) is explicitly shown. The acoustic score \( P(O|\mathbf{h}^i_f))/(o) \) is computed using the statistical models of the recognizer and the language score \( P(o) \) is computed from language model.

In calibration phase, the goal is to find the parameter vector \( \mathbf{h}^i_f \) for each phone for optimal recognition performance. Assuming the correct transcription \( o_c \) is known, we can maximize (2) for array parameters \( \mathbf{h}^i_f \) by

\[
\hat{\mathbf{h}}^i_f = \arg\max_{\mathbf{h}^i_f} \log P(O|\mathbf{h}^i_f))/(o_c)
\]

Because the transcription is assumed to be known a priori, the language score can be neglected. We assume that the likelihood of a given transcription is largely represented by the single most likely HMM state sequence. If \( S_c \) shows the set of all possible sequences and \( s \) represents one such sequence, then (3) can be written as

\[
\hat{\mathbf{h}}^i_f = \arg\max_{\mathbf{h}^i_f} \left\{ \sum_i \log P(O_i(\mathbf{h}^i_f)|S_i)), + \sum_i \log P(S_i|S_{i-1}, o_c) \right\}
\]

where \( i \) sums over all frames.

Second term can be optimized by forced alignment using Viterbi algorithm, assuming the array parameters are fixed and determined. For optimizing first term, given a state sequence \( s \), we must find \( \mathbf{h}^i_f \) so that

\[
\hat{\mathbf{h}}^i_f = \arg\max_{\mathbf{h}^i_f} \sum_i \log P(O_i(\mathbf{h}^i_f)|S_i))
\]

Maximization of Eq. (5) should be done by nonlinear optimization methods, because the state distribution used is a complicated density Gaussian function and the acoustic likelihood of an utterance and the parameter vector \( \mathbf{h}^i_f \) are related through a series of linear and nonlinear relations. Therefore we employ a nonlinear optimization approach to find the optimal value for \( \mathbf{h}^i_f \).

We define the target function as

\[
F(\mathbf{h}^i_f) = \sum_i \log P(O_i(\mathbf{h}^i_f)|\hat{S_i}))
\]

For single multivariate Gaussian state output distribution with diagonal covariance matrices, if we define \( \mu_i \) and \( \Sigma_i \) as the mean vector and covariance matrix, the log likelihood for a phone can be expressed as

\[
F(\mathbf{h}^i_f) = \sum_i \left[ -0.5(O_i(\mathbf{h}^i_f) - \mu_i)^T \Sigma_i^{-1} \cdot (O_i(\mathbf{h}^i_f) - \mu_i) + \kappa_i \right]
\]

where \( \kappa_i \) is a normalizing parameter [9]. Method of conjugate gradients is used to optimize the array parameters \( \mathbf{h}^i_f \).

To compute \( O_i(\mathbf{h}^i_f) \), we use (1). We assume that time-delay compensation (TDC) has already been performed. The output signal \( y[n] \) is then segmented into a series of overlapping frames. If we define \( N \) as the length of a frame in samples and \( R \) the number of samples between starting points of consecutive frames, we can represent the short-time Fourier transform (STFT) of frame of the output signal as

\[
Y_i[k] = \sum_{n=0}^{N-1} w[n]y[n+R+i]e^{-j2\pi nk/N}, \quad 0 \leq k \leq N/2 - 1
\]

where \( w[n] \) is Hamming window applied to the signal at frame \( i \). Note that we have only computed the non-negative frequencies of the DFT in this case. Because the input signal is real, the STFT has conjugate symmetry and the features are only extracted from non-negative half of the STFT.

The squared magnitude of the STFT is computed and used to derive the mel spectrum, a vector of length \( L \) that represents the energy in a series of overlapping frequency bands defined by a set of \( L \) triangular weighting functions called mel filters. We can define the \( t \)th component of the mel spectral vector as

\[
M^t = \sum_{k=0}^{N/2-1} V^t[k]Y_i[k]Y^*_i[k], \quad 0 \leq t \leq L - 1
\]

where \( Y^*_i[k] \) is the complex conjugate to \( Y_i[k] \).

Finally, the mel frequency cepstral vector is derived from the mel spectral vector by first taking the logarithm of each component of the mel spectral vector, producing the log mel spectrum, and then performing a truncated discrete cosine transform (DCT) operation. The DCT operation is performed in order to reduce the dimensionality of the feature vector and decorrelate its components for better classification performance in the recognizer. Thus, for a cepstral vector of length \( C \), we define \( \Phi \) as the \( \mathcal{C} \times L \) DCT matrix \((C \leq L)\), and express the \( c \)th cepstral coefficient as

\[
O^c_i(\mathbf{h}^i_f) = \sum_{t=0}^{L-1} \Phi_{ct} \log(M^t_i), \quad 0 \leq c \leq C - 1
\]

In order to optimize array parameters using (7) and conjugate gradient method, we need to compute Jacobian matrix of feature vector \( \mathbf{O} \), with respect to array processing parameter vector \( \mathbf{h}^i_f \) for each phone expressed as

\[
\mathbf{h}^i_f = [h_0[0], \ldots, h_{m-1}[P-2], h_m[1][P-1]]^T
\]

where \( h_m[p] \) represents the \( p \)th tap of the FIR filter associated with microphone \( m \). Let \( C \) be the length of the cepstral vector \( \mathbf{O} \). We define \( \mathbf{J} \) to be the \( MP \times C \) Jacobian matrix composed of the partial derivatives of each element of the feature vector \( \mathbf{O} \) in frame \( i \) with respect to each of the array parameters \( h_m[p] \) as

\[
\mathbf{J}_i = \frac{\partial \mathbf{O}_i}{\partial \mathbf{h}^i_f} = \left[ \begin{array}{cccc}
\frac{\partial O^0_0}{\partial h_0[0]} & \frac{\partial O^1_0}{\partial h_0[0]} & \cdots & \frac{\partial O^{C-1}_0}{\partial h_0[0]} \\
\frac{\partial O^0_1}{\partial h_0[1]} & \frac{\partial O^1_1}{\partial h_0[1]} & \cdots & \frac{\partial O^{C-1}_1}{\partial h_0[1]} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial O^0_{M-1}[P-2]}{\partial h_{M-1}[P-2]} & \frac{\partial O^1_{M-1}[P-2]}{\partial h_{M-1}[P-2]} & \cdots & \frac{\partial O^{C-1}_{M-1}[P-2]}{\partial h_{M-1}[P-2]} \\
\frac{\partial O^0_{M-1}[P-1]}{\partial h_{M-1}[P-1]} & \frac{\partial O^1_{M-1}[P-1]}{\partial h_{M-1}[P-1]} & \cdots & \frac{\partial O^{C-1}_{M-1}[P-1]}{\partial h_{M-1}[P-1]} \\
\end{array} \right]
\]
We use (8)–(10) to derive the expression for each term \( \partial \Omega_f / \partial h_n[p] \) in the Jacobian matrix. Using (10), the partial derivative of \( \Omega_f \) with respect to the filter coefficient \( h_n[q] \) can be expressed as

\[
\frac{\partial \Omega_f}{\partial h_n[q]} = \sum_{l=1}^{L-1} \Phi_{ij} \frac{\partial M_l}{\partial h_n[q]}
\]

(13)

The partial derivative term \( \partial M_l / \partial h_n[q] \), computed from Eq. (9) using the product rule, can be expressed as

\[
\frac{\partial M_l}{\partial h_n[q]} = \sum_{k=0}^{N/2} V[k] \left( Y_l[k] \frac{\partial Y_l'}{\partial h_n[q]} + \frac{\partial Y_l'}{\partial h_n[q]} Y_l[k] \right)
\]

(14)

To compute \( \frac{\partial Y_l'}{\partial h_n[q]} \) after passing through the most likely state sequence, we use (8)–(10) to derive the expression for each term

\[
Y_l[k] = \sum_{n=0}^{N-1} w[n] \cdot \left( \sum_{m=0}^{M-1} \sum_{p=0}^{P-1} h_n[p] X_m[n-p] \right) e^{-j2\pi kn/N}
\]

(15)

Rearranging the order of summation to

\[
Y_l[k] = \sum_{m=0}^{M-1} \sum_{p=0}^{P-1} h_n[p] \left( \sum_{n=0}^{N-1} w[n] X_m[n-p] \right) e^{-j2\pi kn/N}
\]

(16)

where \( X_m[n-p] \) is a weighted sum of the STFT of \( X_m[n-p] \) over all channels and taps delays. If we define the STFT of \( X_m[n-p] \) as

\[
X_m'[n] = \sum_{n=0}^{N-1} w[n] X_m[n-p] e^{-j2\pi kn/N}
\]

(17)

then \( Y_l[k] \) can be expressed as

\[
Y_l[k] = \sum_{m=0}^{M-1} \sum_{p=0}^{P-1} h_n[p] X_m'[n]
\]

(18)

where \( X_m'[n] \) is the DFT of a frame of speech beginning \( p \) samples prior to the starting sample of the \( t \)th frame.

Using Eq. (18), the partial derivatives in Eq. (14) can be expressed as

\[
\frac{\partial Y_l'}{\partial h_n[q]} = X_m'[n]
\]

(19)

\[
\frac{\partial Y_l'}{\partial h_n[q]} = X_m'[n]
\]

(20)

Substituting Eqs. (19) and (20) into Eq. (14), we obtain the following expression for \( \partial M_l / \partial h_n[q] \) as

\[
\frac{\partial M_l}{\partial h_n[q]} = \sum_{k=0}^{N/2} V[k] Y_l[k] X_m'[n] + X_m'[n] Y_l[k]
\]

\[
= 2 \sum_{k=0}^{N/2} V[k] |Y_l| X_m'[n] Y_l[k]
\]

(21)

Finally by substituting Eq. (21) into Eq. (13), we obtain the complete expression for \( \partial \Omega_f / \partial h_n[q] \) as

\[
\frac{\partial \Omega_f}{\partial h_n[q]} = \sum_{l=1}^{L-1} \Phi_{ij} \sum_{k=0}^{N/2} V[k] |Y_l| X_m'[n] Y_l[k]
\]

(22)

The Jacobian matrix \( J_f \) is formed by substituting Eq. (22) into Eq. (12) for \( c = [0, \ldots, C-1], n = [0, \ldots, M-1], \) and \( p = [0, \ldots, P-1] \).

Total calibration procedure is as follows. Calibration utterance is received from user and signals of microphones are compensated for time delays. After initializing parameters of filters for each phone, filter and sum algorithm is applied and feature vectors are extracted and used to compute likelihood of specific phone. These parameters are used to estimate state sequence for calibration utterance and using them to compute new optimum filter parameters. This procedure continues until the likelihood for that phone converges. This routine is applied for all phones of language. Fig. 3 shows flowchart of this procedure.

4. Modified Viterbi for decoding using phone-based filters

In our suggested method, during test phase, when Viterbi algorithm uses observation signal to compute the emission likelihood, we apply sets of parameters of filters assigned to each phone to the observation likelihood function.

In convenient decoding, the most likely state sequence is computed using Viterbi trellis [10]. In the Viterbi trellis, \( v_i(j) \) shows that the probability the model to be in state \( j \) after seeing and passing through the most likely state sequence, \( q_1, \ldots, q_t, \) is computed recursively. This means:

\[
v_i(j) = P(q_0=q_i, \ldots, q_t, q_t=j; \lambda)
\]

(23)

where \( q_1, \ldots, q_t \) are observations and \( \lambda \) is the hidden Markov model. For computing \( v_i(j) \), dynamic programming is used. For a given state \( q_i \) at time \( t \), \( v_i(j) \) is computed as

\[
v_i(j) = \max_{1 \leq l \leq N-1} v_{l-1}(j) a_{lj} b_i(o_l)
\]

(24)

where \( a_{lj} \) is transition probability from previous state \( q_i \) to current state \( q_t \) and \( b_i(o_l) \) is the state observation likelihood of the observation symbol \( o_l \) given the current state.

In the phone-based parameter optimization, in decoding phase of recognition, we use a modified Viterbi algorithm in which
instead of using observations directly, we apply the corresponding phone filter, \( h_f \), to observation signal and then compute the most likelihood path. Corresponding phone filter parameters, \( h_f \), have been determined and stored in database in calibration phase as one filter parameter set for each phone. This means for new Viterbi path probability we have
\[
v_h^b(j) = P(q_0, q_1, \ldots, q_{t-1}, \rho_l^1, \ldots, \rho_l^t, q_t = j|\lambda)
\]
(25)
where \( \rho_l^t \) is the filtered observation signal using filter corresponding to \( j \) th state which belongs to related phone \( f \).

For computing \( v_h^b \) using dynamic programming, we will have
\[
v_l(j) = \max_{1 \leq s \leq N-1} v_h^{b-1}(j) a_{js} b_j(\rho_l^t)
\]
(26)
This modified Viterbi algorithm will use the parameters of optimized filter to search for the most probable state sequence in trellis.

5. Results

For implementing the proposed recognition system, we used HTK 3.4 [11]. Original speech signals were sampled at 22.05 kHz that downsampled to 16 kHz. The feature vectors used were 12 MFFCs and the zeroth cepstral coefficient and their first and second derivatives, using hamming windows of 25 ms with a feature vector calculated every 10 ms. Context-independent three-state left-to-right HMMs were trained using FARSAT database which consists of 6080 utterances recorded by 304 Persian talkers from 10 different dialect regions of Iran. For training 5400 utterances are used that told by 270 talker. Considering Persian phones [12], 30 HMM models were trained and corresponding filter parameters were optimized.

Reverberation effect is simulated in a room \( 10 \times 5 \times 3 \) m using image method [13]. The microphone array consists of 4 microphones with 7 cm spacing placed on one of 10 m walls at 1.6 m heights. Talker is at 1 m distance and 1 m left of center of microphone array at the same height. A babble noise source (from NOISEX database) is also placed at 1 m distance and 2 m right of center of microphone array at the same height. A room impulse response between talker and each microphone in array and also between noise source and each microphone in array is created.

For calibration, two utterances of each talker of test set that covers all Persian phones except “f” are used to calibrate the corresponding filters of phones. For optimization, algorithm of section 3 is used. These optimum parameters are saved for test phase.

For test, 612 test utterances are used. Markov models and room specification remain same as calibration phase. Clean test signal of FARSAT is processed with four speech impulse responses and added to babble noise signal that processed by four noise impulse responses. Because speech signals are processed in batch form during test phase, it was necessary that during likelihood computation of Viterbi algorithm, according to (6), filtered observations be available and since in this research no attempt is done to optimize speed of suggested algorithms, so all test signals are filtered with saved optimum filters and corresponding observation vectors were created and saved for use by modified Viterbi algorithm.

At first, effect of length of FIR filter parameters on improvement of absolute value of likelihood, as an indicator, was studied. According to Fig. 4 for three phones “m”, “ee” and “ch”, as good samples of Persian phones with different time-spectrum patterns, it is obvious that by increasing number of taps of filters to 4, absolute likelihood decreases but for more taps, it increases. Because of small length of calibration data for each filter, risk of overfitting of filter parameters is high and it seems this is the reason of increase in absolute likelihood for taps number more than 4. For all remaining experiments, four taps for filters were used.

For the test, one utterance from each talker was used for calibration. For each individual talker, first we reset filter parameters and then calibrate them with corresponding calibration utterances of that talker. In next experiments, effect of noise
on accuracy for delay and sum, utterance-based optimization and phone-based optimization algorithms was compared (Fig. 5). In these tests, 0.2 s reverberation time was assumed. In low SNR, 12.2% improvement relative to utterance-based and 18.8% relative to delay and sum algorithm is achieved. In larger SNR, this improvement reduces. For large SNR, knowing that reverberation is very low, effect of noise on phones with smaller power is reduced and it is predictable that phone-based optimization gets more similar to utterance-based optimization.

Finally, effect of reverberation on accuracy is studied as shown in Fig. 6. Accuracy for clean signal is shown with zero reverberation time. It is assumed SNR is fixed and equals to 20 dB. For low reverberation time, there are no big differences in accuracy between two of the studied methods, but when reverberation time increases, considering allocated filters and better spatial beamforming for each phone with relatively stationary frequency pattern, a relative improvement of 16.7% in 1 s reverberation time is achieved.

6. Conclusion

In this paper experimental results of new approach to microphone array filter and sum processing is proposed in which the parameters of filters are adjusted based on phones used in language. Improvement of 12.2% in recognition accuracy compared to utterance-based processing for 4 dB SNR and no reverberation, and also 16.7% for 1 s reverberation time in 20 dB SNR, has been achieved.

References


Bahram Kouhi-jelehkaran was born in Bandar Anzali, Iran, 1969. He received the Ph.D. in 2009 from the Islamic Azad University, Science and Research Branch of Tehran, Iran. He is an expert in speech processing. Currently he is working on adaptive microphone array robust speech recognition for applications in noisy and reverberant environment.

Hamidreza Bakhshi was born in Tehran, Iran, on April 25, 1971. He received the B.Sc. degree in Electrical Engineering from Tehran University, Iran, in 1992, the M.Sc. and Ph.D. degrees in Electrical Engineering from Tarbiat Modares University, Iran, in 1995 and 2001, respectively. Since 2001, he has been as an Assistant Professor on Electrical Engineering at Shahed University, Tehran, Iran. His research interests include smart antennas, wireless communications, and multiuser detection.

Farbod Razzazi was born in Tehran, Iran, in 1972. He received his B.Sc. and M.Sc. degrees in Electrical Engineering from Sharif University of Technology in 1994 and 1996, respectively. He received his Ph.D. from Amirkabir University of Technology (Tehran Polytechnic) in 2003 in Electrical Engineering. He is currently an Assistant Professor in Department of Electrical Engineering, Islamic Azad University, Science and Research Branch, Tehran, Iran. His current research interests are pattern recognition methods and their applications in speech recognition systems.