THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS INC.

SIGNAL PROCESSING
ALGORITHMS, ARCHITECTURES, ARRANGEMENTS, AND APPLICATIONS
SPA 2013
September 26-28th, 2013, Poznań, POLAND

Automatic speaker recognition using a unique personal feature vector and Gaussian Mixture Models

Kamil Kaminski
Military University of Technology
Faculty of Electronics
Warsaw, Poland
kkw.kaminski@gmail.com

Ewelina Majda, Andrzej P. Dobrowolski
Military University of Technology
Faculty of Electronics
Warsaw, Poland
adobrowolski@wat.edu.pl

Abstract - **This article presents an automatic speaker recognition system implemented in Matlab, which uses a unique feature vector, the so-called "Voice Print" (VP), to describe the voice. The system uses Gaussian Mixtures Models (GMM) in the classification process. The final part of the paper presents research on the efficiency of speaker recognition for different variants of the system, as well as the results of optimisation of the system.**

Keywords: speaker recognition, speech signal, feature extraction, ASR systems, Gaussian Mixtures Models, GMM.

I. INTRODUCTION

Nowadays, the need for employing automatic speaker recognition systems, [ASR (Automatic Speakers Recognition)], is increasing. This stems from the constant increase in the level of interaction between humans and computers. Thanks to the increasing use of biometrics in the authorisation process in systems that require a high level of safety, the ASR system is used in both military and civilian industry.

The voice, one of the distinctive features of each human being, enables the identification of persons without the need for additional attributes in the form of documents that may be damaged or lost. Recognising a person using ones auditory system, something relatively easy for a human, becomes a complex challenge for a machine. Both the machine and the human require some listening time to identify a person. This time will in the following part be referred to as the length of the test segment. Undoubtedly, a human must become accustomed to a voice and commit it to memory in order to be able to recognise the voice of another person. The same applies to computer systems that also require voice models against which they can compare voices of people that are to be recognised. The time for learning and memorising models will be referred to later in the paper as the length of the training segment.

This paper presents an automatic speaker recognition system implemented in Matlab, which uses a unique feature vector, a digital representation of the key features of the voices of particular speakers that is simultaneously a criterion used for comparison in speaker identification, so it will be referred to as a "Voice Print" (VP) [2] from here on. The system uses Gaussian Mixtures Models (GMM) in the classification process. This allows for obtaining voice models that are relatively sparing in memory and contain a significant amount of valuable information about the speaker's voice. In the last phase of the system's operations, a decision is made which of the created voice models can with the highest level of probability serve as a model for the set of multi-dimensional points that constitutes the distinctive features of the voice of the speaker that is to be recognised. Fig. 1 presents the architecture of the ASR system.

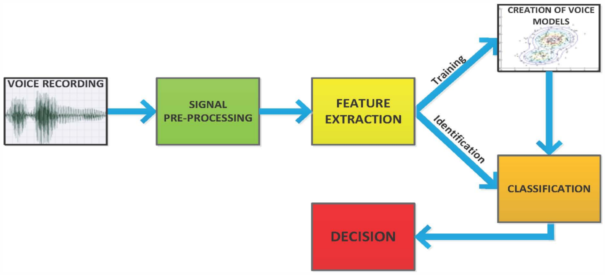

Figure 1. ASR system architecture

II. VOICE DATABASE

The first phase of building an ASR system is to compile an appropriate database of voices of various speakers. If the main task of the system is intended to efficiently recognize different speakers, it is necessary to collected relatively long recordings, which permit a suitably long training phase. The collected recordings of speakers varied considerably and were spoken with different intonations and speeds. This allows the system to be text-independent. Furthermore, in order to obtain objective results, a voice database as large as possible is used. For the presented implementation of the system 50 voices of speakers of various ages and both sexes have been gathered. The recordings were sampled at 22 050 S/s, mono, with a 16-bit amplitude resolution. The recordings were made using a dynamic DM-500 Monacor microphone, a computer sound card and Matlab software. Recording sessions were conducted in one of the lecture rooms of the Faculty of Electronics at Military University of Technology. The conditions in the recording room allowed for a relatively quiet environment, but it was not subject to any further soundproofing. Speakers were recorded for approx. 4 minutes. In order to increase the authenticity of the obtained results, 10 test segments, independent from each other and from the training segments, were extracted. This made the results more reliable because 500 test segments were subject to testing as a result. In order to optimise settings of the ASR system the authors used different lengths of training segments and test segments during the study.

III. Pre-processing

Pre-processing is performed mainly in order to make recorded audio signals independent of the settings of the recording equipment. At this phase, thanks to an application developed in Matlab, the following are performed: removal of DC offset, signal normalisation, filtering and silence removal. The application uses a digital band-pass filter with a finite impulse response, whose poles and cut-off frequency were optimised (Table II). The silence removal algorithm is based on cutting frames that do not meet the empirically set power criterion for a frame – the multiplication of the power of the quietest frame of the voice recording and the constant set in the optimisation process. Thanks to this operation it is possible to reduce the number of signal frames analysed, which improves the speed of voice recognition. Furthermore, analysing only relevant frames from the viewpoint of speaker recognition increases the efficiency of the system. Both, recordings found in the model database, as well as recordings of utterances subject to speech recognition, were automatically subject to signal pre-processing.

IV. Determination of a set of distinctive features

Automatic speaker recognition requires the development of a numerical description of the speech signal in the form of appropriately defined descriptors that best characterise a particular speaker. This phase, often called signal parametrisation or generation of features, is crucial from the point of view of quality of the proposed recognition system, as any mistakes and shortcomings at this phase reduce the discriminatory ability of generated features, something that cannot be rectified in later phases (classification, verification, identification). The goal of speech signal parameterisation for the ASR is to transform the input waveform into a limited number of descriptors containing information relevant to the particular speaker, while minimizing their sensitivity to signal variability irrelevant from the point of view of ASR. The presented system is based on the vector of distinctive features developed during previous studies, referred to by the authors as the voice print - VP [1, 2].

The original and primary form for registering a speech signal is the waveform. The time domain, however, is not the most appropriate for performing further operations because a speech signal demonstrates a very high level of redundancy within it. From the standpoint of further analysis it is much more efficient to transform the signal into the frequency domain. One of the principal reasons for this approach is to try to imitate nature, which has over the course of evolution developed the human speech organ in such a way that the speech signal is generated - and then received and analysed by the auditory system - in the frequency domain.

One particular method of parameterisation derived directly from spectral analysis is called cepstral analysis [6]. A complex cepstrum is defined as the inverse Fourier transform of the logarithm of the simple Fourier transform of the analysed waveform. Since the calculation of the complex logarithm engenders complications stemming from the need to ensure the continuity of the phase and in the case of a speech signal the essential information is contained in the spectrum amplitude, in practice one usually calculates the so-called real cepstrum, where the simple Fourier transform is replaced by its modulus.

An analysis of the presented idea allows for a simple interpretation of the real cepstrum as the spectrum of the logarithmic amplitude spectrum. By observing the amplitude spectrum of the speech signal it is possible to notice that it is composed of a high-frequency variable resulting from excitation and a low-frequency one modulating the amplitude of subsequent pulses resulting from the excitation. The interpretation of the logarithm of the amplitude spectrum is similar, but here the low-frequency component is not multiplied by the amplitudes of the individual pulses originating from the excitation, but added to them. The calculation of the spectrum of such a signal causes low-frequency waveforms associated with the vocal tract transfer function to be close to zero on the quefrency axis, while the pulses associated with laryngeal sound start roughly in the vicinity of the laryngeal signal period and are repeated every such period. Information related to the vocal tract transfer function is focused around the zero index on the cepstral time axis, referred to as quefrency, and therefore it is in this area that one should search for concise information about the utterance. Meanwhile, for times above the laryngeal sound period information about what is being said is minimised and only readable information about the laryngeal sound remains. Since the laryngeal sound is closely linked with the anatomy of the larynx and the glottis, it is therefore also a good carrier of information about an individual. It is therefore easy to understand the usefulness of the real cepstrum for speaker recognition.

In the presented approach the generation of features consists of the creation of descriptors based on three techniques derived from cepstral analysis. In each of these methods a set of distinctive features was identified and then an initial selection of them performed.

A. *Cepstral features*

As a result of extensive preliminary studies, the following features were chosen from this group: the speaker's basic frequency, which is the inverse of the first maximum cepstrum (a zero value maximum occurs for a zero quefrency value), and the values of 4 successive maximum values normalised to the value of the first maximum.

B. *Mel-frequency cepstral features*

The most popular method of parameterisation of the speech signal is a method using MFCC (Mel-Frequency Cepstrum Coefficients). This is a method based on a sub-band analysis of a signal using band-pass filters equally spaced on the mel-frequency scale. The major difference with this transformation consists in the conversion of the spectrum from a linear to a mel scale, in accordance with the following formula [7]

$$f[mel] = 1127\ln\left(1+\frac{f[Hz]}{700}\right). \tag{1}$$

This transformation accounts for the non-linearities in the perception of sound frequencies by the human auditory system, as well as a significant reduction of the data. In order to achieve a non-linear spectrum transfer, a collection of filters is created for subsequent half-overlapping frequency bands evenly spaced on the non-linear mel scale. Triangular-shaped filters are defined in the frequency domain, which makes it possible to determine responses from them as the sum of the results of the multiplication of the spectrum modulus and the course of the triangular function. The vector of signals from the outputs of all filters is subject to a log transform and then to a discrete cosine transform. The resulting MFCC vector has a length equal to the number of bands.

At the phase of generation of mel-frequency cepstrum characteristics it was decided to apply the 30 filters, which yielded 30 distinctive features. Therefore a problem appeared of determining which of the MFCC features are representative only of the pronounced sound and which of the speaker himself or herself. Features related to the content of the utterance should not be considered and, as previously, it is necessary to search for features with indices higher than a certain threshold. Employing earlier studies [7] an initial pre-selection of features was utilised. This reduced the number of elements of the MFCC vector to 7, while minimising the decrease in its representativeness through checking the results on the basis of PCA (ang. Principal Component Analysis). This method was used due to the large preliminary size of the initial MFCC feature vector. Displaying the 30-dimensional MFCC feature vector on a plane has made it possible to perform an efficient initial pre-selection of features relevant from the point of view of the modelled feature generator.

C. *Weighted cepstral features*

Inspiration from the idea of the MFCC method has resulted in an extension of the feature vector by subsequent features in the cepstrum domain through the application of sum filters in sub-bands. The proposed algorithm does not search for maximum values in the bands where their position is predicted, but adds the amplitudes of all lines from these bands with a set weight. The selection of the optimal filter characteristic (weighting function) and the band width was one of the optimisation tasks for the system. A rectangular function was selected as a result of optimisation. The 4 weighted cepstral features have been defined as the sums of four subsequent bands starting with the second, normalised to the sum obtained in the first band, corresponding to the pitch.

Therefore 16 numerical descriptors that differentiate speakers were defined at the phase of feature generation. The aforementioned descriptors constitute the maximum set of potential distinctive features that can be used in the automatic recognition system. Various studies [3] demonstrate that the use of a maximum set of features does not always lead to the best results, as they are not equally important in model recognition. Some features may have the form of measurement noise to the detriment of the ability to recognise the speaker, while strongly correlated features usually have an adverse effect on the quality of classification by dominating over the others and suppressing their beneficial effect. An important part of the process is therefore to perform a quality assessment of descriptors and apply appropriate selection methods when creating an optimal vector. At the next phase feature selection was performed using Fisher's method and PCA. An optimal 11-dimensional feature vector VP was defined and implemented in Matlab as a result of the performed feature selection [2].

V. Classification

A. *Creating GMMs representative of speakers' voices*

GMMs used in the classification process are the parametric probability density functions represented by sums C of Gaussian distributions [4]. They make it possible to obtain voice models that are relatively sparing in memory and contain a significant amount of valuable information about the speaker's voice using only the training data set *X*, where $X = \{x_1, x_2, ..., x_T\}$ is the set *T* of cells containing *d*-diemnsional feature vectors VP extracted for the analysed speaker's voice. The classifying system is implemented using functions available in Matlab. In the first phase of voice model creation, the starting values of distributions parameters $\lambda = \{x_i, \mu_i, \Sigma_i\}$, $i = 1,...,C$ are selected in a pseudo-random manner, i.e. expected values μ_i, covariance matrices Σ_i and distribution weights w_i. Next, the probability of occurrence of individual feature vectors belonging to a particular speaker in the created model of his or her voice is calculated [8]:

$$p(x_t \mid \lambda) = \sum_{i=1}^{C} w_i, N(x_t, \mu_i, \Sigma_i). \tag{2}$$

The function *N* that defines the multidimensional Gaussian distribution is as follows [8]:

$$N(x_t, \mu_i, \Sigma_i) = \frac{\exp\left\{-\frac{1}{2}(x_t-\mu_i)^{'\Sigma_i^{-1}}(x_t-\mu_i)\right\}}{(2\pi)^{d/2}\,|\,\Sigma_i\,|^{1/2}}. \tag{3}$$

The next stage is to estimate model paramters that maximise the likelihood of the GMM by using only *X* training data. These are usually set in accordance with the maximum likelihood (ML) estimator. GMM likelihood can be represented as [8]:

$$p(X|\lambda) = \prod_{t=1}^{T} p(x_t|\lambda). \tag{4}$$

Due to the fact that the training set is very large and the values of the feature vectors of the same person are not equal, the probability density function has a number of maxima and it is not possible to perform direct maximisation. Therefore, the optimisation of the probability density function is executed iteratively, while the estimation of model parameters is conducted through an expectation maximization [EM] algorithm in such a way that the new model $\bar{\lambda}$ for which $p(X|\bar{\lambda}) \geq p(X|\lambda)$ becomes the initial model for the next iteration, until the convergence threshold is not reached. The execution of the algorithm determines the likelihood *a posteriori* for the *i*-th distribution by way of the following formula [8]:

$$p\left(i|x_t,\lambda\right)=\frac{w_i N(x_t,\mu_i,\Sigma_i)}{\sum_{i=1}^{C} w_k N(x_t,\mu_i,\Sigma_k)}\,. \tag{5}$$

On the basis of equation (5) it is possible to estimate the values of model parameters in each iteration [8]:

Distribution weights

$$\overline{w}_i=\frac{1}{T}\sum_{t=1}^{T}p\left(i|x_t,\lambda\right), \tag{6}$$

Expected values

$$\overline{\mu}_i=\frac{\sum_{t=1}^{T}p\left(i|x_t,\lambda\right)x_t}{\sum_{t=1}^{T}p\left(i|x_t,\lambda\right)}, \tag{7}$$

Covariance matrices

$$\overline{\Sigma}_i=\frac{\sum_{t=1}^{T}p\left(i|x_t,\lambda\right)\left(x_t-\mu_i\right)\left(x_t-\mu_i\right)'}{\sum_{t=1}^{T}p\left(i|x_t,\lambda\right)}\,. \tag{8}$$

B. *Speaker recognition*

Let λ_k for $k = 1,\dots,N$ mean voice models, where N is the number of different voices in the database. The classifier is designed in such a way as to assign a feature vector X of a voice to be recognised to one of the voice N models. For this purpose, the probability of occurrence of the analysed feature vectors is calculated for each model. The model that can most accurately serve as a model for the set of analysed feature vectors (i.e. for which the probability of occurrence of the analysed vectors is the greatest) is the voice model of the recognised person. Ultimately, the decision on identifying the speaker is made on the basis of log-likelihood [8]:

$$k^{*}=\arg\max_{1\le k\le N}\sum_{t=1}^{T}\log p\left(x_t|\lambda_k\right). \tag{9}$$

VI. Tests of the implemented ASR system in *Matlab*

This chapter contains the results of studies on the automatic speaker recognition system using Gaussian mixtures models. Efficiency tests of speaker recognition based on the number of Gaussian distributions used to model each voice and the set length of training and test segments have been performed. The relation between the number of correctly recognised test fragments in relation to the total number of recordings (500) was used as the measure of the efficiency of the system. Tests have also been performed on a new voice database containing longer recordings, which makes it possible to use longer training segments. An efficiency test has also been performed with lower sampling rates in order to check the scope of application of the discussed ASR system.

A. *Efficiency tests of speaker recognition based on the number of Gaussian distributions, the length of the training segment and the test segment.*

The speaker recognition system has been tested with different parameter values. The research covered the following: length of training segment, number of Gaussian distributions and the length of the test segment. The results are presented in Table I.

TABLE I. Efficiency of speaker recognition depending on the length of the training segment and the test segment, as well as the number of Gaussian distributions.

Training segment length	**Number of Gaussian distributions**	**Test segment length**				
		1 s	***2 s***	***3 s***	***4 s***	***5 s***
15 s	***1***	42	53.6	57.6	59.6	63.2
	2	42	52.6	54.2	56.2	61
	4	43	56.2	60.2	61.8	65.6
	8	46.6	59.8	61.4	65.6	68.4
	16	45.6	60.6	62.4	66	68.2
	32	45.6	60.4	62.6	65.4	67.8
	64	44.6	54.8	58.8	60	61.8
	128	44.8	53.8	55.8	55.2	55.6
30 s (8)	***1***	45.2	59	63.8	66.6	72
	2	45.6	59.6	63.2	63	69.6
	4	49	64	68	72.6	76.8
	8	51.6	66	74	75.2	78.6
	16	53	70	75.8	78.4	78.2
	32	54	71	76	78.6	80.4
	64	53.4	71.2	75.8	79	80.2
	128	54	68.4	74.2	77.4	75.8
60 s	***1***	72	62.4	68	72	77
	2	69.6	64	68	73	75.8
	4	76.8	70.6	74	78.4	81.2
	8	78.6	71.4	77.2	80	83.4
	16	78.2	74.6	79.2	82.8	85
	32	80.4	75.2	81	84.2	88
	64	80.2	77	81.2	85.2	86.8
	128	75.8	78.8	80.8	84.2	86.8
90 s	***1***	49.8	65	70	73.6	78
	2	50.4	65.2	69.2	73.4	76.8
	4	54.6	68.2	73.6	76.8	81.2
	8	57.2	74	78.8	83.4	86.2
	16	60.8	77	82.8	85.6	87.2
	32	62.2	81	83.6	88	89.6
	64	65.4	81	85.8	89.6	89.8
	128	65	81.2	85.6	88.4	90.2
120 s	***1***	51.6	66.6	73.2	78.2	81.4
	2	52.2	69.4	73.8	77.6	81.6
	4	57.8	71.6	78.2	82.6	87.2
	8	60.4	77.2	82	87	90.4
	16	62.8	79.8	85.8	89.4	91.2
	32	66.4	84.4	87.4	90.2	91.8
	64	67.2	84.8	88	90.4	92
	128	69.2	85.2	89.8	91.8	93
150 s	***1***	54.8	69.6	76	80.6	83.8
	2	55.2	69.8	75.8	80.4	83.6
	4	61.2	76.2	81.2	84.6	89.6
	8	62.8	81.4	86.4	90.8	93.2
	16	66.4	84.2	89.2	91.8	93.8
	32	69.2	85.8	89.6	92.2	94.8
	64	69.4	87.2	90.8	92.8	95.2
	128	72.2	88	90.8	92.6	95.6

The conducted studies highlight the impact of increasing the length of the training segment and the test segment on the increase in efficiency of recognition of particular speakers. The creation of a training segment during the use of the system is performed only once at the initial phase. This makes it possible to record a training segment of any length, while the recording, which is tiring for the speaker, comes down to only one recording session.

Test segments are recorded each time the system is used. It is therefore beneficial to choose an in-between test segment length, one which allows for the greatest possible recognition efficiency while keeping the length as short as possible.

The third variable in the conducted research was the number of Gaussian distributions used for modelling the voice of each speaker. Increasing the number of distributions has a beneficial effect on speaker recognition, but if their number is very large (128), in some cases excessive correspondence of distributions to training data occurs (overtraining), which results in poor generalisation of the created models for data that is not in the training database. This results in a decrease of recognition efficiency despite the increased volume of data describing the tested voice.

B. Tests of system efficiency using records with an extended training segment

Research conducted as part of the previous test suggests that a further increase in the length of the training segment can have a positive influence on the efficiency of recognition speakers' voices. The database of voices used so far did not allow for the execution of longer training segments. For this purpose, a new database has been created containing 15 recordings of speakers, each of which recorded at least 5 minutes. This has made it possible to create training segments with a length of 240 s and 10 sets of test segments of varying lengths (1-5 s) for each voice. The test results are shown in Fig. 2.

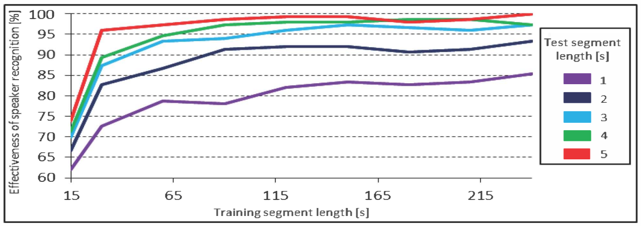

Figure 2. Efficiency of speaker recognition with an extended training segment

The graph above demonstrates that a further increase in the length of the training segment has a positive influence on the efficiency of speaker voice recognition. It can be assumed, however, that a training segment length of 240 s is close to the optimal length. This statement is motivated by the fact that there is a visible downward trend for the 4 s long test segment with a training segment longer than 210 s.

C. Test of the efficiency of speaker recognition with lower sampling frequencies

The last test presented in this paper is aimed at determining the efficiency of speaker recognition at lower sampling frequencies, including the one used in telephony (8 kS/s). Results shown in Fig. 3 indicate that the reduction of sampling frequency has a negative effect on speaker recognition efficiency. However, for test segments of at least 3 s in length this decrease is acceptable, while for a sampling rate of not less than 8 kS/s the speaker recognition efficiency exceeds 90%. This demonstrates that the proposed method can be successfully used in a telephone system for caller recognition. The visibly better efficiency of speaker recognition for higher frequencies is due to the fact that, even though the level to which speech is understandable does not improve, but it is possible to better capture individual features contained in subtle variations in tone, which requires a wider band.

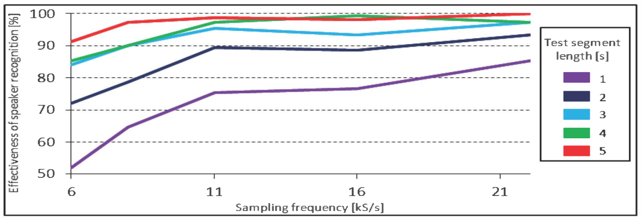

Figure 3. Effect of sampling frequency on the efficiency of speaker recognition

VII. Optimisation of System Parameters

An attempt at optimisation has been made for various system parameters in order to increase the efficiency of voice recognition using the shortest possible training segment. As shown in previous tests, the length of the test segment should be not shorter than 3 s in order to obtain reliable results. For this reason for subsequent analyses a fixed length of the test segment equal to 3 s has been assumed.

The creation models for the same set of *VP* vectors for describing the training recordings were tested ten times with the same system parameters maintained. It has been observed that the obtained results differed slightly, which is a result of the random selection of initial values of the covariance matrix, the expected values and the distribution weights. However, with subsequent iterations these values underwent constant changes in order to perform the most accurate generalisation of parameters for a particular voice. In order to obtain the best possible set of Gaussian mixture models for the analysed parameter configuration, it is necessary to repeatedly train the models using the same *VP* training vectors, while maintaining identical system parameters, and select the models that allow the most efficient speaker recognition. When studying the impact of system parameter variability on the efficiency of recognition, the authors trained models ten times in each possible parameter configuration, which was a very time-consuming process. Then the highest efficiency value from the obtained values for the analysed parameter configuration was taken into account and acted as a benchmark when choosing their optimal settings.

A. Optimisation of the silence removal function in the recordings

The silence removal function used in the presented implementation of the ASR system performs the removal only for clear pauses between words spoken by the speaker in order to distort collected recordings as little as possible. Furthermore, more accurate silence removal is not justified because in the initial phase of feature generation the signal will be checked for the presence of voiced phones and the level of noise of analysed frames, as well as the frame variance value. On the basis of this information it will be possible to effectively eliminate the remaining silent fragments of the signal that are devoid of information about the speaker. The frames whose power was lower than the frame with the lowest power level in the whole recording, multiplied by a constant coefficient. The silence removal function has been optimised in terms of the length of the removed frame, the degree to which frames overlap and the coefficient that was used for the multiplication of the frame with the least power in the voice recording. The results are presented in Table II.

B. *Optimisation of filter parameters*

Using the optimum silence removal function parameters obtained in the previous study, further system optimisation was undertaken in the subsequent phase for filtration of the waveform and filtration of the cepstrum obtained in the parameterisation process. The waveform has been subjected to a high-pass filter, eliminating the DC offset and the low-frequencies not carrying information on individuals. On the other hand the cepstrum has been subjected to a low-pass filter, which made it possible to smooth it and remove information that is irrelevant for pitch recognition. Filter poles and their cut-off frequencies were optimised.

TABLE II. SUMMARY OF OPTIMUM PARAMETER VALUES FOR THE ASR SYSTEM

Phase	Parameter	Value
Silence removal	Frame length	900 ms
	Frame overlap	50 %
	Threshold	$2{,}1E_{min}$
Signal filtering	High-pass filter pole	8
	Cut-off frequency	97 Hz
Cepstrum filtering	Low-pass filter pole	18
	Cut-off frequency	775 Hz
Feature generation	Frame length	55 ms
	Frame overlap	4 %
	Voicing threshold	6
	Power threshold	18
	Pitch differences threshold	20

C. *Optimisation of frame length and overlap, and thresholds at the phase of feature generation*

The next step in the research was to optimise the frame length and overlap at the stage of *VP* vector generation. Three threshold values were also optimised. The first optimised threshold at the feature generation phase was the voicing threshold that separates voiced fragments of the signal, significant from the point of view of speech recognition, from the unvoiced ones. Voiced fragments are characterised by the regular occurrence of maxima of the autocorrelation function. Therefore, in order to check whether an analysed phone is voiced, it is necessary to check the level of the second global maximum and compare with the applied voicing threshold. The next threshold optimised by the authors is the power threshold, which, like at the previously presented silence removal phase, eliminates silence from the signal, but, this time, on the basis of the variance of the signal in the frame. Further optimisation has been performed on the percentage-based pitch difference threshold determined using two methods. According to the theory, the determination of the pitch using the cepstral method is less accurate, but more reliable than autocorrelation, particularly for a highly noisy speech signal. For this reason the difference between these two values can be employed as an excellent tool for checking signal quality in the frame and, if necessary, makes it possible to skip excessively noisy frames.

D. *Selection of the most relevant GMM*

Due to the fact that the initial pseudo-random values of distribution parameters result in different efficiency results of the recognition system being obtained for the same parameters, it is necessary to perform modelling a number of times and choose those which fare best in efficiency tests.

VIII. SUMMARY

The speaker recognition system described in the article has been tested with different parameter values. Finally, at a sampling rate of 8 kS/s of a the test segment with a length of 3 s and a training segment with a length of 1 min, a recognition rate of 93% was achieved. Increasing the length of the test segment or the training segment results only in the increase of system efficiency.

The research conducted is a valuable source of information for the authors, whose ultimate aim is to implement the described system in a signal processor. The tests show that the system can be successfully used even during telephone communication. Further work entails testing the system with a unified voice database.

IX. BIBLIOGRAPHY

[1] A. P. Dobrowolski, E. Majda, “Cepstral analysis in the speakers recognition systems, 15th IEEE SPA Conference, pp. 85-90, Poznan, 2011.

[2] A. P. Dobrowolski, E. Majda, “Application of homomorphic methods of speech signal processing in speakers recognition system”, Przeglad Elektrotechniczny, vol. 88 (6), pp. 12-16, 2012.

[3] R. C. SGuyon, A. Elisseeff, “An introduction to variable and feature selection”, Journal of Machine Learning Res., vol. 3, 2003, pp. 1157-1182.

[4] K. Kaminski, J. Wojtun, Z. Piotrowski, “Subscriber authentication using GMM and TMS320C6713DSP”, Przeglad Elektrotechniczny, vol. 88 (12a), pp. 127-130, 2012.

[5] T. Kinnunen, H. Li, “An overview of text-independent speaker recognition: From feature to supervectors,” Speech Communication, 2010, pp. 12-40.

[6] R. Maison, E. Majda, A. P. Dobrowolski, M. Zakrzewicz, “Similarity based join over audio feeds in a multimedia data stream management system”, Bell Labs Technical Journal, vol. 18, no. 1, 2013, pp. 195–212.

[7] E. Majda, A. P. Dobrowolski, “Feature generator for speaker recognition using the fusion of cepstral and melcepstral parameters”, Joint Conference NTAV/SPA, Łódź 2012, pp. 203-208.

[8] D. Reynolds, “Gaussian Mixture Models, MIT Lincoln Laboratory, Massachusetts,” USA 2007.