ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

PhD THESIS

Gökay DİŞKEN

SPEAKER RECOGNITION FOR SECURITY SYSTEMS UNDER NOISE EFFECTS

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

ADANA-2018

ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES

SPEAKER RECOGNITION FOR SECURITY SYSTEMS UNDER NOISE EFFECTS

Gökay DİŞKEN

PhD THESIS

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

We certify that the thesis titled above was reviewed and approved for the award of degree of the Doctor of Philosophy by the board of jury on 07/03/2018.

……..	…….……	……
Prof. Dr. ULUS ÇEVİK	Prof. Dr. Ergun ERÇELEBİ	Assoc. Prof. Dr. Sami ARICA
SUPERVISOR	MEMBER	MEMBER
…….	……..	
Assoc. Prof. Dr. Esen YILDIRIM	Assoc. Prof. Dr. Turgay İBRİKÇİ	
MEMBER	MEMBER	

This PhD Thesis is written at the Department of Institute of Natural And Applied Sciences of Çukurova University.
Registration Number:

Prof. Dr. Mustafa GÖK
Director
Institute of Natural and Applied Science

ABSTRACT

PhD THESIS

SPEAKER RECOGNITION FOR SECURITY SYSTEMS UNDER NOISE EFFECTS

Gökay DİŞKEN

ÇUKUROVA UNIVERSITY
INSTITUTE OF NATURAL AND APPLIED SCIENCES
DEPARTMENT OF ELECTRICAL AND ELECTROICS ENGINEERING

Supervisor : Prof. Dr. Ulus ÇEVİK
Co-Supervisor : Asst. Prof. Dr. Lütfü SARIBULUT
Year: 2018, Page: 149
Jury : Prof. Dr. Ergun ERÇELEBİ
: Assoc. Prof. Dr. Sami ARICA
: Assoc. Prof. Dr. Esen YILDIRIM
: Assoc. Prof. Dr. Turgay İBRİKÇİ

Nowadays, many different devices and applications such as vehicles, smart home devices, mobile banking, automatic dictation programs, and legal surveillance comprise speech and Speaker Recognition (SR) systems. Noise is one of the most important factors that affect the performances of these systems. Therefore, reducing the susceptibilities of the systems to noise is very important.

Two different methods are proposed within this thesis to reduce the negative effects of the noises for SR systems. One of these methods is creating impostor models by clustering speaker models. The other method is a Polynomial Regression (PR) based Voice Activity Detector (VAD), which aims to determine the high energy speech regions under additive noise.

Recent, and widely used SR methods, and the proposed algorithms within this thesis were realized experimentally, and performance analyzes were made by comparatively presenting results of the case studies.

Keywords: Text-independent speaker recognition, Voice activity detector, Polynomial regression, Speaker model clustering

ÖZ

DOKTORA TEZİ

GÜVENLİK SİSTEMLERİ İÇİN GÜRÜLTÜ ETKİSİ ALTINDA KONUŞMACI TANIMA

Gökay DİŞKEN

ÇUKUROVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Danışman : Prof. Dr. Ulus ÇEVİK
İkinci Danışman : Asst. Prof. Dr. Lütfü SARIBULUT
Year: 2018, Page: 149
Jüri : Prof. Dr. Ergun ERÇELEBİ
: Doç. Dr. Sami ARICA
: Doç. Dr. Dr. Esen YILDIRIM
: Doç. Dr. Turgay İBRİKÇİ

Günümüzde arabalar, akıllı ev aletleri, telefon bankacılığı, otomatik dikta programları, adli telefon dinlemeleri gibi pek çok farklı araç ve uygulama bünyesinde konuşma ve Konuşmacı Tanıma (KT) sistemleri yer almaktadır. Bu sistemlerin performansını etkileyen en önemli faktörlerden biri gürültüdür. Bu nedenle sistemlerin gürültüye karşı hassaslığının azaltılması oldukça önemlidir.

Bu tez kapsamında KT sistemleri için gürültülerin olumsuz etkilerini azaltmaya yönelik iki farklı yöntem önerilmiştir. Bu yöntemlerden biri, konuşmacı modellerinin gruplandırılarak taklitçi modelleri oluşturulması olup, diğer yöntem ise eklenebilir gürültü altında yüksek enerjili konuşma bölgelerini belirlemeye yönelik Polinom Regresyonu (PR) tabanlı Ses Aktivitesi Detektörü (SAD) yöntemidir.

Literatürde güncel ve yaygın kullanılan konuşmacı tanıma metotları ile tez çalışmasında önerilenlerin algoritmalar deneysel olarak gerçekleştirilmiş olup, durum çalışmalarının sonuçları karşılaştırılmalı sunularak performansları analizleri yapılmıştır.

Anahtar Kelimeler: Metinden bağımsız konuşmacı tanıma, Ses aktivitesi dedektörü, Polinom uydurma, Konuşmacı modeli kümeleme

EXTENDED ABSTRACT

Mismatch between the training and testing utterances is one of the main reasons for degraded performance of Speaker Recognition (SR) systems. Although the usage of SR systems spreads with many practical applications, the mismatch problem is still an active research area.

In this thesis, two different algorithms were developed to overcome the mismatch effects originated by additive noises, and channel mismatch, respectively. In general, additive noise can be defined as the sounds captured by the microphone other than the speaker's voice. On the other hand, channel mismatch is more related to the hardware, such as microphone type, transmission line, etc.

One of the proposed algorithms is the Polynomial Regression based Voice Activity Detector (PR-VAD), which is effective against additive noises. Usually, VADs aim to separate the speech and noise/silence regions of the utterances. Also, the VADs that use a fixed magnitude/energy threshold fail when the Signal-to-Noise Ratio (SNR) of the utterances varies. The proposed PR-VAD algorithm includes a pseudo SNR estimation step to automatically adjust the threshold for a given utterance. Further, the proposed algorithm considers the noise presence besides the speech information. If a frame is dominated by the noise components, it is discarded even though some speech information may present in a few frequencies. Therefore, the harmful effects of the noise components are avoided.

The core of the PR-VAD algorithm is the polynomial regression step, which is applied in each filter band independently. The main process of the PR-VAD algorithm consists of k-means clustering, speech enhancement, and binary voting for the final decision. The PR-VAD is applied per utterance, and does not require a training session.

The robustness of the PR-VAD was examined in text-independent speaker verification experiments. Five different noise types were considered with SNR

levels varying from -10 dB to 10 dB with 5 dB steps for each noise type. A recently proposed artificial neural network based VAD, and a noise tracking algorithm were used as baseline methods. Gender-dependent tests were made with both conventional, and state-of-the-art speaker modeling methods. The results verified that the PR-VAD algorithm had achieved better recognition performances than the baseline methods.

The other method is the Speaker Model Clustering (SMC). In the conventional speaker modeling method, namely Gaussian Mixture Model-Universal Background Model (GMM-UBM), only one impostor model is used, which is the UBM. In the SMC method, several background models are obtained by clustering the speaker models derived from the UBM by adaptation. Speakers in the same group share the related group's background model in this approach.

In the experiments, both matched, and mismatched channel conditions were considered to verify the performance of the SMC algorithm. The GMM-UBM method, and the state-of-the-art speaker modeling method i-vectors were used as baseline methods. The results proved that the SMC algorithm yields better recognition performances against the GMM-UBM method for both channel conditions, and all test utterance durations. Furthermore, comparable results were achieved against the i-vector method, without increasing the system's complexity. Even better results were observed by including a handset score normalization.

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to my supervisor Prof. Dr. Ulus ÇEVİK for his guidance and support during this thesis. I would like to express my gratitude to my co-supervisor Asst. Prof. Dr. Lütfü SARIBULUT for his encouragement and motivation.

I would like to express my special thanks to Assoc. Prof. Dr. Zekeriya TÜFEKCİ for his patience, inspirational discussions, sharing his knowledge, and providing me with the datasets used in this thesis.

I would like to thank my committee members Prof. Dr. Ergun ERÇELEBİ, Assoc. Prof. Dr. Sami ARICA, Assoc. Prof. Dr. Esen YILDIRIM, Assoc. Prof. Dr. Turgay İBRİKÇİ.

I also want to thank Dr. Thomas DRUGMAN for sharing the codes and implementation details of his algorithm.

CONTENTS **PAGE**

LIST OF TABLES **PAGE**

LIST OF FIGURES **PAGE**

LIST OF ABBREVATIONS

ALSA : Advanced Linux Sound Architecture
ANN : Artificial Neural Networks
CMN : Cepstral Mean Normalization
DET : Detection Error Tradeoff
DCF : Detection Cost Function
DCT : Discrete Cosine Transform
DNN : Deep Neural Networks
DTW : Dynamic Time Warping
EER : Equal Error Rate
FA : False Acceptance
FAR : False Acceptance Rate
FFT : Fast Fourier Transform
FR : False Rejection
FRR : False Rejection Rate
FT : Fourier Transform
GMM : Gaussian Mixture Model
GPIO : General Purpose Input Output
HMM : Hidden Markov Model
HTK : Hidden Markov Model Toolkit
IBM : Ideal Binary Mask
JFA : Joint Factor Analysis
K-MC : K-Means Clustering
LDA : Linear Discriminant Analysis
LP : Linear Prediction
MFCC : Mel Frequency Cepstral Coefficients
MSR : Microsoft Research
NAP : Nuisance Attribute Projection

NIST : National Institute of Standards and Technology
NT : Noise Tracking
PLDA : Probabilistic Linear Discriminant Analysis
PMC : Parallel Model Combination
PR : Polynomial Regression
RASTA : Relative Spectra
SMC : Speaker Model Clustering
SNR : Signal-to-Noise Ratio
SR : Speaker Recognition
SRE : Speaker Recognition Evaluation
SV : Speaker Verification
SVM : Support Vector Machines
UBM : Universal Background Model
VAD : Voice Activity Detector
VQ : Vector Quantization
VTS : Vector Taylor Series
WCCN : Within Class Covariance Normalization
WT : Wavelet Transform

1. INTRODUCTION

Speech is the most natural and convenient way to communicate for humans. Various information can be obtained from speech signals, therefore speech processing has become an important research area in the last decades. According to (Hafen and Henry, 2012) information extracted from speech signals can be used in event detection, stress and emotion classification, speaker diarization and recognition, speech recognition, multilingual audio analysis, and acoustic fingerprinting.

Among these topics, automatic Speaker Recognition (SR), which means using a machine to recognize a speaker's identity by using his speech signal, has a special importance in terms of biometrics. The development, and accessibility of smart phones and internet gave people endless opportunities. By using their phones, people can access their social media accounts, made money transfers over bank accounts, start the conditioner at their home while they're away, etc. These are just a few examples of what one can achieve by using his/her phone. However, the technology also brings many concerns, such as security.

Many biometrics are offered to increase the security of various systems such as face, fingerprint, hand geometry, iris, and voice recognition (Jain et al., 2006). The traditional verification methods such as signature, password, pin codes, etc. cannot satisfy the remote operation, or high security demands of the users. On the other hand, Speaker Verification (SV) systems are cheaper, and easily accepted by the users compared to the other systems. As an example, only a microphone to record the voice of a user is enough. Further, many commercial applications are already in use. Some of the banks around the world, and also mobile operators in Turkey are using voice biometrics. With this technology, users do not have to share their personal information, or memorize passwords.

In general, SR can be split into two parts as verification and identification (Campbell, 1997; Furui, 1997). The verification is a one-to-one problem, where the

user claims an identity, and the system checks whether the user is really who he/she claims to be or not. The identification is a little bit more complicated problem, where the user's identity is searched in a group of enrolled users. Usually, as the size of the group increases, identification performance of the systems decreases. On the other hand, the verification problem is not related to the group size (Furui, 1997).

Both the verification and identification systems can be text-dependent, or text-independent. In the text-dependent case, the users of the system are restricted to utter few words (such as digits) or fixed phrases. For the text-independent case, the users can freely talk. Between them, text-independent verification is much harder, since there is no limitation on the users' utterances. Therefore, all possible phonemes should be represented in the training data to build a feasible recognition system. Figure 1.1 shows the two types of SR with their working philosophy.

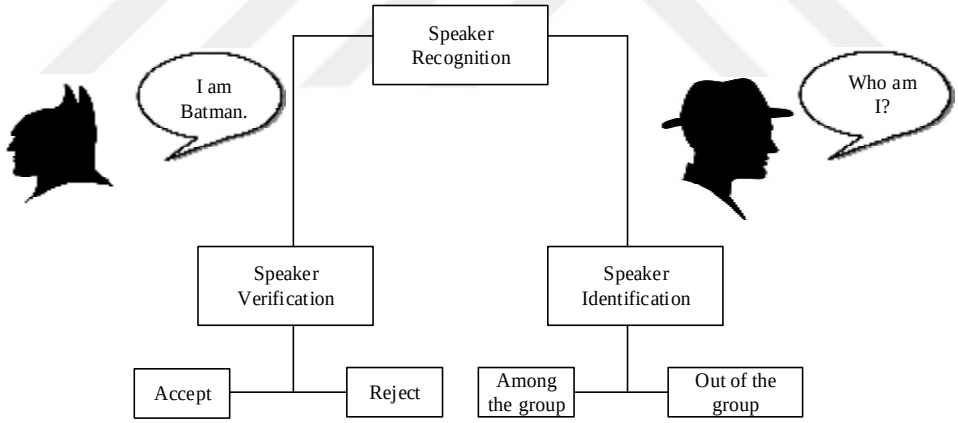

Figure 1.1. Types and working principle of speaker recognition.

SR systems consist of two processes called front-end, and back-end. The front-end process is where the signal processing techniques are applied to the speech signal. The core of this part is the feature extraction. The raw speech signal includes redundant data. In terms of SR, speaker specific information should be extracted from the speech, and the other information can be discarded. By using

the extracted features and machine learning methods, speaker models are trained. The back-end process generally refers to these machine learning methods used to model/classify speakers, and scoring utterances in the training, and test phases, respectively. The trained speaker models are then used in the tests to score unknown utterances.

SR researches can be tracked back to a few decades; between-speaker variations are discovered in spectral patterns by (Denes and Mathews, 1960), and further SR experiments are conducted in (Das and Mohn, 1971; Das, 1969; Li et al., 1966).

Despite the progress over the years, there is still space for improvements in many aspects. One of the fundamental problems of speech/speaker recognition systems is the robustness against noise. In general, speaker models are trained by using their respective training data. The training data are collected in controlled environments such as a recording room, an office, etc. Therefore, the records are “clean”, which indicates that there is no sound besides the speaker’s voice. Feature extraction and speaker model training methods are applied to this clean data. However, in the real-life experiments, the same training conditions may be impossible to replicate in the test stage. This situation leads to recognition performance degradations.

Two main noise sources can be considered. The first one is the convolutive noise, which is the result of modifications applied to the signal in the transmission channel. Hence it is also called channel noise. When a device other than that used to collect training data is used in the test stage, a channel mismatch occurs (Gish et al., 1985). This mismatch occurred by using different channels degrades the performance of the SR systems (Li et al., 2017; Rao et al., 2016; Sahidullah and Saha, 2012a; Zhu et al., 2015). The second one is the additive noise, which can be considered as the sounds captured by the recording device other than the speaker’s voice. The other sound signals deteriorate the speaker specific information, hence

they negatively affect the recognition performance (Al-Kaltakchi et al., 2017; Edwards et al., 2017; Frankle and Ramachandran, 2016; May et al., 2012).

In this thesis, a novel Polynomial Regression (PR) based Voice Activity Detector (VAD), and a Speaker Model Clustering (SMC) algorithm are developed, and their performances are tested experimentally. In the SMC algorithm, K-Means Clustering (K-MC) is applied to the conventional Gaussian Mixture Model-Universal Background Model (GMM-UBM) method to obtain impostor models. Compared to the state-of-the-art i-vector method, better SV results are observed for both matched and mismatched channel conditions.

The proposed VAD consists of PR, K-MC, spectral subtraction, and pseudo Signal-to-Noise Ratio (SNR) dependent thresholding. The performance of the proposed VAD is tested experimentally under five different noise types, and five different noise levels. Also, performance comparisons with a state-of-the-art Artificial Neural Networks (ANN) based VAD, and a Noise Tracking (NT) algorithm are presented.

This thesis is organized as follows.

In Chapter 2, the SR literature is reviewed in a few subtitles. Various feature extraction methods are mentioned with their pros and cons. Conventional and state-of-the-art modelling techniques are briefly described. Noise types, and their deteriorative effects are investigated. Most common methods to achieve robustness (i.e. VAD, robust features, robust classifiers, speech enhancement) are analyzed based on the recent publications.

In Chapter 3, the feature and classifier types used in the thesis are explained in details. Then, the proposed algorithms to achieve robustness against channel noise, and additive noise are given. For a better understanding, block diagrams are provided besides the mathematical expressions. The database used in the experiments is also described. The proposed VAD is tested on a sample signal to analyze it visually by giving various spectra, and filter-bank magnitudes. At the

end of Chapter 3, the system configurations (both hardware, and software) for real-time isolated word recognition is demonstrated.

In the beginning of Chapter 4, performance metrics are defined, and then the experimental results of the proposed algorithms and performance comparisons with well-known methods are given. The performance of the proposed SMC algorithm is tested with both the conventional and state-of-the-art classifiers, and for both matched/mismatched channel conditions. The proposed VAD algorithm's robustness against additive noise is tested by using five different noise types, and five different noise levels. Real-time isolated word recognition experiment results are also provided at the end of this chapter.

In Chapter 5, results obtained from the proposed algorithms are discussed and their advantages/disadvantages are presented. In Chapter 6, the contributions of the thesis are summarized, and possible future research directions are given.

2. LITERATURE REVIEW

In this chapter, a literature review of studies on SR is provided, with an emphasis on robustness issues. The most common feature and classifier types are mentioned. Then, the negative effects of the various noise sources are discussed, and some of the solutions provided in the literature for each noise source are given.

2.1. Feature Extraction

Feature extraction is the process where the redundant information in the raw speech signals are discarded. The analog speech signal is captured by a microphone, and sampled by an analog-to-digital converter. After the conversion, the digital speech signal can be processed with a computer, microcontroller, etc. There are numerous feature types in the literature. The most important point is to know the requirements of the specific applications. As mentioned in the introduction chapter, word, language, speaker, gender, emotion etc. information is present in the raw speech signal. As an example, if one seeks for the word information, then he should not extract emotion related features. So, for the SR case, the extracted features should be discriminative between speakers. Properties of an ideal feature are listed as the following (Kinnunen and Li, 2010; Wolf, 1972).

- The feature should have large variations between speakers, and small variations within speaker.
- The feature should be difficult to mimic for increased security.
- The feature should occur frequently, and naturally in speech. Hence no special training, or effort is required to produce them.
- The feature should be easy to obtain from the speech signal.
- The feature should be robust against undesired effects such as noise.
- The feature should not be affected by the speaker’s health or long-term variations such as aging.

Obviously, the ideal feature does not exist. Also, since the speech production system of the humans is a physiological system, deformations caused by health issues, or aging, affects the produced voice. As an example, when a person gets cold, even his family cannot recognize his voice. Therefore, it is a challenging, and unresolved problem to obtain features close to these specifications. Also, it is possible to fuse different types of features (Li et al., 2016; Venturini et al., 2014). One feature type should cover the information that the other feature type does not have. By this way, features are used in a complementary manner to increase the recognition performance of the systems.

Features can be divided into categories such as short-term spectral, prosodic, high-level features, etc. Various feature types are proposed in each category. However, short-term features seem to be most popular for both research purposes, and practical applications due to their ease of implementation and good recognition performance. Short-term features' aim is to capture vocal tract information.

Speech is a highly non-stationary signal. However, in a short segment called frame (10-30 ms), it is assumed to be stationary. Short-term spectral features analyses these short frames. Usually, the frames are windowed to prevent frequency leakage. Although it is not critical, the Hamming window is the most preferred window type in the speech processing literature. Fast Fourier Transform (FFT) is then applied to obtain magnitude spectrum. Phase spectrum is usually discarded, however some studies given later in this chapter shows that it may be beneficial for increasing the robustness of the system.

The magnitude spectrum captures the resonance properties of the vocal tract (Kinnunen and Li, 2010). Based on the human audio perception system, a filter-bank is used to combine energies of the neighbor frequency bands. Filters with narrow bandwidths are used in the lower frequency region, while larger bandwidths are used in the higher frequency region. The resulted coefficients are called sub-band energy, or filter-bank energy values, and used in many SR, and

speech recognition studies (Besacier et al., 2000; Besacier and Bonastre, 2000; Damper and Higgins, 2003; Erell and Weintraub, 1993; Kua et al., 2010; Tufekci and Gowdy, 2001).

The filter-bank energy vectors usually have a high dimensionality. High dimension data make the model training and scoring processes longer. Therefore, researches have been investing much more compact feature vectors. One of the most popular feature types is the Mel-Frequency Cepstral Coefficients (MFCCs) (Davis and Mermelstein, 1980). MFCCs are obtained by using a triangular filter-bank, and applying Discrete Cosine Transform (DCT) to the log-compressed filter-bank energies. In general, 12-19 coefficients are extracted per frame in this method. Also, MFCCs are used as feature vectors in this thesis, due to their success and widespread usage in the literature. They are analyzed in detail in the next chapter.

Besides MFCCs, many Linear Prediction (LP) based coefficients have been used for SR (Mammone et al., 1996). LP is used to obtain the spectrum of the signal, and generally predictor coefficients are further processed to observe more robust features. Linear prediction coefficients (Li et al., 2014), perceptual linear prediction cepstral coefficients (McLaren et al., 2013), linear predictive residual (Khan et al., 2012), and line spectral frequencies (Klein and Feldes, 2016) are among the examples of the LP based feature used in the speech related research. A more detailed analysis of the LP is presented in (Pati and Mahadeva Prasanna, 2010).

Wavelet Transform (WT) has become a powerful tool in the signal processing area. WT is used in numerous speech and image processing studies. Contrary to the Fourier Transform (FT), or its variant short-term FT, WT enables decompositions other than sinusoidal. WT also has the multi resolution property. Many wavelet basis functions have been developed to extract different features in the data. In terms of speech recognition, WT is used to extract robust features, replaced with FT in MFCC extraction scheme, or combined with other features to

increase recognition performance of the systems (Daqrouq and Al Azzawi, 2012; Jun Yao and Yuan-Ting Zhang, 2001; Malik and Afsar, 2009; Singhai and Singhai, 2007; Srinivas et al., 2014; Turner et al., 2011).

Many other short-term features are proposed in order to achieve better recognition, or robustness, (Jiang et al., 2015; Li and Huang, 2011; McLaren et al., 2013; Plchot et al., 2013; Sadjadi and Hansen, 2015). However, MFCCs and LP based features' popularity have not decreased. They are still used in many state-of-the-art systems, and they provide baseline results for the researchers.

Glottal source feature extraction is another segmental analysis, and can be used to capture speaker specific information. However, their extraction is much more complicated than the MFCCs, or LP based features (Drugman et al., 2014). Several SR studies investigated the effects of glottal excitation based features (Amin et al., 2014; Drugman and Dutoit, 2012; Gudnason and Brookes, 2008; Kinnunen and Alku, 2009; Ostrogonac et al., 2013; Yegnanarayana et al., 2001). As reported in (Drugman et al., 2014), these features alone do not perform better than vocal tract features. However, by combining them with the vocal tract features, better recognition performances can be achieved.

Spectro-temporal clues such as energy variations, formant frequency transitions, etc. may also include speaker information. A very basic approach to obtain temporal dependencies is to extract time derivative estimate called delta features (Furui, 1981). Delta features are computed by using a few neighbor feature vectors. Usually, they are concatenated to the main feature vectors (i.e. MFCCs). Other spectro-temporal features can be found in the literature such as (Kockmann et al., 2011; Magrin-Chagnolleau et al., 2002), but using delta features has become a common practice.

Prosodic and high-level features' main difference from the short-term features is that they span over long durations such as phonemes, words, etc. They represent behavioral properties of the speaker. Speaking style, emotions, speaking rate, fundamental frequency (pitch), energy modulations, characteristic vocabulary,

etc. can be extracted from these features (Andrews et al., 2002; Daqrouq and Tutunji, 2015; Leung et al., 2006; Li Hui et al., 2006; Reynolds et al., 2003; Sarma and Sarma, 2013; Shriberg et al., 2005). These features are more robust than the short-term features, however they require complex extraction processes, and more training data (Kinnunen and Li, 2010). Therefore, they are mostly used as complementary features to the short-term features.

2.2. Speaker Modelling

Speakers are enrolled into the recognition system via creating a model for his/her voice. The feature vectors extracted from a speaker's training data is used to train speaker's model, hence the back-end performance is highly dependent to the quality of the features. Each speaker has his/her own model, which is going to be involved in pattern matching process to compute a score in the tests. The speaker models are stored in the memory, contrary to the feature vectors which can be discarded once the speaker model is constructed. In this sub-section, some of the most used modelling approaches are reviewed, considering both the conventional, and state-of-the-art methods.

One of the conventional template based methods is the Dynamic Time Warping (DTW). In the early ages of the SR studies, direct template matching methods were used (Luck, 1969; Pruzansky, 1963). The training and testing models' distance to each other are computed using Euclidean distance, Mahalanobis distance, etc. As an example, the model for a speaker could be the mean of a set of training vectors. However, temporal variation is usually ignored in this modelling.

DTW is actually introduced into word recognition to compensate temporal variability between training and testing utterances (Sakoe and Chiba, 1978). The DTW concept is used for text-dependent SR in (Furui, 1981). The core of the DTW concept is to expand, or shrink, feature vectors to find the best match between training and testing templates. Each vector of the test utterance is matched with the

closest vector of the training template (piece-wise linear mapping). Once all the testing vectors are matched, their distances are summed to obtain a final score. For long utterances, DTW becomes slower, since the number of feature vectors increases.

Vector Quantization (VQ), which is another template-based method, is used to reduce the number of feature vectors by clustering. The cluster centers are concatenated to build a codebook. K-means is one of the most used clustering algorithm for codebook generation (Linde et al., 1980). VQ is used in many SR studies to model speakers (Burton, 1987; Hautamki et al., 2008; Singhai and Singhai, 2007; Soong et al., 1985). The k-means is a hard decision algorithm, which means that a feature vector may belong only to one cluster. Fuzzy c-means algorithm is introduced as a soft decision alternative to the k-means, and achieved better SR performance (Chatzis et al., 1999; Lin and Wang, 2006). Nearest neighbor method is also used for SR as an alternative to DTW and VQ (Higgins et al., 1993).

Besides template based models which have dominated early research on speech processing, stochastic models such as Hidden Markov Model (HMM), and Gaussian Mixture Models (GMM) are developed to increase recognition performances. Consideration of the speaker model, stochastic methods measure the likelihood of the feature vectors. Hence, the probability that the observed feature vectors was generated by the given speaker model can be calculated. HMM, which is a fundamental method in speech recognition, used for SR in (Matsui and Furui, 1992; Rosenberg and Parthasarathy, 1996). Using the training data, state transition probabilities, feature vector probability distributions, and initial state probabilities of HMM are learned.

An important point in the back-end methods is using the GMM for speaker models (Reynolds, 1995; Reynolds and Rose, 1995; Rose and Reynolds, 1990). A GMM consists of a number of multivariate Gaussians. GMMs can be thought as a single state HMM. Also, similar to the fuzzy c-means, a feature vector is associated

with each Gaussian, i.e. each Gaussian has a non-zero probability of generating each of the feature vectors.

GMMs can model arbitrary shapes with a good accuracy, hence it outperformed uni-modal Gaussian (Gish et al., 1985), VQ (Soong et al., 1985), tied Gaussian mixture, and radial basis function (Oglesby and Mason, 1991) modeling techniques in text-independent SR (Reynolds and Rose, 1995). A GMM is defined by its mixture weights, mean vectors, and covariance matrices. Diagonal covariance matrices are preferred because of numerical and computational reasons. Estimating a full covariance matrix requires much more training data, and computational sources (Kinnunen and Li, 2010).

Expectation maximization algorithm is used to train GMMs (Bilmes, 1998). Monogaussian models, a single Gaussian with a full covariance matrix, is used as speaker models (Besacier et al., 2000; Besacier and Bonastre, 2000; Zilca, 2002). They offer computational efficiency compared to the GMMs, in trade off recognition performance.

To train a speaker's GMM, sufficient amount of training data is required. In practical applications, collecting large amounts of data from each user may not be appropriate. If the GMMs are not well-trained, the system performance will indeed decrease. Another problem with the modelling methods in general is to obtain an anti-model, i.e. a model for the speakers that are not enrolled to the system (impostors). Traditionally, a model called cohort for each speaker is observed. Cohort speakers are chosen by a similarity measure between their models and the target speaker's model (Reynolds, 1997). The cohort models increase the memory requirements, and they may not be beneficial for each speaker. A solution to this problem is to obtain speaker models via adaptation from a well-trained model. Universal Background Model (UBM) method is developed for this purpose (Reynolds et al., 2000). Actually, the UBM is just a GMM with a large number of Gaussians (512-2048).

The UBM is trained by pooling all available training data. It is also called as world model, because it aims to represent all available speakers, and is used as the impostor model. Speakers' models are then adapted from the UBM by using their respective training data. All of the model parameters can be adapted, but it is shown that adapting only the means is sufficient (Reynolds et al., 2000). The adaptation process is visualized for 2-dimensional data in Figure 2.1 (Kinnunen and Li, 2010).

The GMM-UBM method becomes the standard back-end for the text-independent SR. Nevertheless, more recent back-end developments are based on this method. Therefore, they are utilized in the proposed methods. The mathematical expressions of GMM-UBM method, and its application details are given in the next chapter.

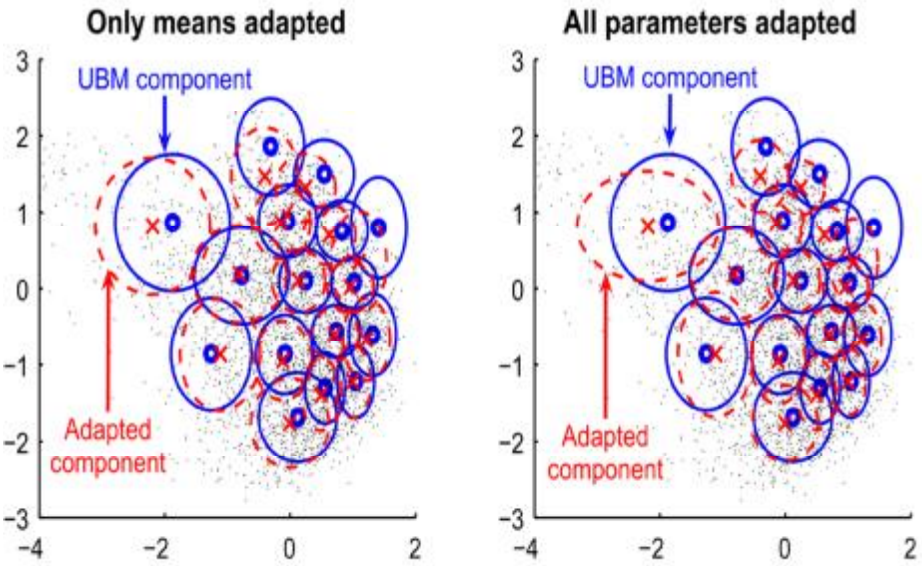

Figure 2.1. UBM mixtures (blue), and adapted speaker model's mixtures (red) for 2-dimensional data (Kinnunen and Li, 2010)

The adaptation process to derive speaker models also bounds the mixtures of the models, therefore gives a fast scoring opportunity in the testing stage.

Although the GMM-UBM method is faster in terms of training, and testing than the previous methods, speaker identification in a large population can still be time consuming. Several methods were proposed to achieve speed-ups. GMM hashing, where top scoring mixtures for each feature vector can be predicted by a GMM that is smaller than the UBM (Auckenthaler and Mason, 2001; McClanahan and De Leon, 2015; McClanahan and DeLeon, 2012).

Hierarchically clustering the mixtures of UBM is investigated in (Bing Xiang and Berger, 2003; Saeidi et al., 2010). Speaker clustering at model level (Apsingekar and De Leon, 2009; Beigi et al., 1999; de Leon and Apsingekar, 2007), and at feature level (Xiong et al., 2006) are also used as speed-up techniques. Although these studies provided some degree of speed-ups, there is a tradeoff between the identification accuracy and identification time. The reason for this tradeoff is that not all the mixtures are scored, or not all the speakers' models are considered.

Support Vector Machines (SVM) is another important machine learning algorithm, which successfully implemented into SR studies (Campbell et al., 2006a, 2006b; Ferrer et al., 2007; Hou Fenglei and Wang Bingxi, 2001; Shriberg et al., 2005; Wan and Renals, 2005; Zhang et al., 2009). Unlike the GMMs, SVM is a discriminative classifier. It separates two class with a decision boundary. In terms of SR, one of the classes is the target speaker. The other class is the impostor speaker, i.e. all speakers other than the target speaker.

The combination of the GMM and SVM is highly attractive. For this combination, means of the GMMs are stacked to construct a vector called supervector. The speakers are represented with these supervectors, and they are fed into SVM classifier. This combination beats the GMM-UBM approach in text-independent SV experiments (Campbell et al., 2006b).

The idea of using supervectors to define speaker space was actually proposed previously (Kenny et al., 2005; Kenny and Dumouchel, 2004). (Kuhn et al., 2000) proposed a rapid speaker adaptation method, where adapted models are

assumed to be a linear combination of few basis vectors. In that work, speakers are represented by a low dimensional vector named eigenvoice.

The Joint Factor Analysis (JFA) is developed to model speaker, and session variabilities separately (Kenny, 2005). The UBM is also used in the JFA to obtain a speaker independent supervector. The speaker- and channel-dependent supervector is decomposed into speaker, and channel supervectors. Once the estimated channel supervector is discarded, the remaining supervector is used as the speaker model. Channel compensation is also achieved by this way. Compared with the GMM-UBM, and SVM methods, JFA achieved better recognition performance (Kinnunen and Li, 2010).

Recently, it is found that the channel space has some speaker discriminative information. Therefore, instead of separately modeling speaker and channel spaces, a single low dimensional total variability space is introduced (Dehak et al., 2011b). The high dimensional GMM supervectors are represented with intermediate sized vectors, commonly known as i-vectors. Also, channel variability can be compensated in the i-vector space by using methods such as Linear Discriminant Analysis (LDA), Within Class Covariance Normalization (WCCN), and Nuisance Attribute Projection (NAP) (Dehak et al., 2011b).

The i-vector method gained a high reputation due to its ability to represent variable-length utterances with a fixed low dimensional vector, and its superior SR performance. It is considered as the state-of-the-art speaker model for the text-independent SR. Numerous research have been made on i-vectors to further improve its performance (Biswas et al., 2015; Cumani et al., 2014; Cumani and Laface, 2014, 2013; Kanagasundaram et al., 2014; Kua et al., 2013; Liu and Kang, 2014; McLaren and van Leeuwen, 2012; Rao et al., 2015, 2016; Rao and Mak, 2013; Tingting Liu et al., 2014; Verma and Das, 2015). Most of the studies focused on i-vector extraction in an efficient manner in terms of speed, and memory. Some of the studies considered channel compensation problem. The i-vector classifier methods are another research topic.

Besides SR, i-vectors are also used in accent recognition (Bahari et al., 2013; Behravan et al., 2015), speaker age estimation (Bahari et al., 2014), language recognition (Vazquez-Machado et al., 2016), utterance verification (Choi et al., 2016). Refer to (Verma and Das, 2015) for a review of i-vectors in speech processing.

Although the conventional GMM-UBM method, and advanced models based on the UBM (i.e. GMM-SVM, JFA, i-vectors) dominated the SR back-end, Artificial Neural Networks (ANN) gained a high interest in the last years. Actually, ANNs have been used for many years (Bennani and Gallinari, 1995; Farrell et al., 1994; Murty and Yegnanarayana, 2006; Oglesby and Mason, 1990; Yegnanarayana et al., 2001). However, the GMM based methods dominated the area. Recent studies on speech recognition showed that Deep Neural Networks (DNN) outperforms the GMMs for acoustic modeling (Hinton et al., 2012). The ANN based methods in speech processing gained a huge interest thanks to the success of the DNNs.

The DNNs, or ANNs more generally, utilized in many SR studies by the researchers. In (Garimella et al., 2012), speaker specific weights between third hidden layer and the output layer of the ANN is projected onto a subspace to obtain i-vectors. This study is improved in (Garimella and Hermansky, 2013) by applying a factor analysis to the weight matrix. (Lei et al., 2014) replaced the UBM with a DNN in i-vector extraction scheme, and achieved lower SR errors.

(Bie et al., 2015) used a DNN to reconstruct clipped speech for SR. (Variani et al., 2014) used a DNN to extract speaker specific features called d-vector. In the test phase, distances between speakers' d-vectors are considered. D-vector reportedly outperformed i-vector in text-dependent SV experiments.

DNN based features are combined with MFCCs in (Ghalehjegh and Rose, 2015) to increase SV accuracy. (Snyder et al., 2016) proposed an end-to-end text-independent SV system, where test and training utterances are given to the network

at the same time. In (Richardson et al., 2015) investigated the use of a single DNN for both SR and language recognition.

In conclusion, although the GMM based approaches are still dominating the back-end systems, recent ANN based systems are achieving comparable, even better results. ANN based features are also increased system performances as reported in the literature. However, ANNs usually require a large amount of training data, and training time. In this thesis, the conventional GMM-UBM, and the state-of-the-art i-vector methods are preferred as to model speakers. Their theoretical frameworks, and implementation details are given in Chapter 3.

2.3. Noise Sources

In this sub-section, noise sources that affect the recognition performances of the systems are considered in three parts. First one is the channel noise, which is a result of the hardware effects. Additive noise is examined in the second part. The other sources are mentioned in the last part. Then, methods to achieve robustness against these noise types are explored.

2.3.1. Channel Noise

Channel noise, or convolutional noise, occurs due to the variable frequency characteristics of transmission channels, microphones, and hand-sets. Using different hardware in the training and testing stages creates a mismatch between the extracted features. This mismatch reduces the recognition accuracy of the systems.

A simple solution to overcome the mismatch problems in general is to train the system with data that represent the operating conditions (channel types, environment, etc.) However, this may not be feasible in most practical applications. As an example, consider a phone banking system which includes an automatic SV stage. Consumers can be at any environment such as office, outdoor, traffic, airport, etc. Also, it is not possible to ask the users to use same mobile phone

model. Therefore, it is a nearly impossible task to obtain training data from every mobile phone model, and every possible place in the world.

Normalization methods are provided in the literature to reduce the channel mismatch effects at the feature level, or score level. One of the standard methods is Cepstral Mean Normalization (CMN) (Furui, 1981). The main idea behind the CMN is that convolution becomes multiplication in the spectral domain. Then, it becomes additive in the log-spectral domain. A mean vector is calculated by using the extracted feature vectors. The mean vector is subtracted from the feature vectors, hence channel effects are diminished.

Relative Spectra (RASTA) filtering is another normalization method, where slowly varying frequency signals are removed in the log-spectral domain. (Hermansky and Morgan, 1994). Feature warping is used to constrain the distribution of cepstral features to a standard distribution, reducing the noise effects (Lung et al., 2014). Short-time Gaussianization (Xiang et al., 2002), speaker model synthesis (Teunen et al., 2000), feature mapping (Reynolds, 2003), acoustic factor analysis (Hasan and Hansen, 2013), and various score normalization techniques (Auckenthaler et al., 2000; Reynolds et al., 2000) are among the other methods proposed to reduce channel mismatch.

2.3.2. Additive Noise

Additive noises can be summarized as the sounds captured by the recording device other than the speaker's voice. Various additive noise sources can be listed for practical applications. Consider again a SV application over mobile phones. Users can be in a street so that the microphone may also capture the sounds of cars and people passing by, engine noises, wind, construction work, etc. The captured sounds will be added to the speaker's voice, therefore the error rate of the system will be increased.

Additive noises can be stationary such as the fan noise from computers, or nonstationary such as cars passing by, car horns, etc. The temporal and spectral

characteristics of the nonstationary noises constantly change. Hence, it is much harder to deal with the nonstationary noises.

As the noise level increases, SNR decreases. At low SNRs, it is important to get rid of the noisy components to achieve a satisfactory recognition performance. A simple way to select speech-dominant frames is to use energy based VADs (Kinnunen and Li, 2010). However, a fixed energy threshold to detect speech activity is not suitable for nonstationary noises. Also, if the level of a stationary noise changes, the fixed threshold may harm the system's performance. Therefore, many other methods have been proposed to robustly detect voice activity, as shown later in the literature review of VADs.

The cepstral features' performances in the controlled environments such as recording studio, quit office, etc. are decent. However, they are not robust to additive noises. Extraction of robust features, or increasing the robustness of the cepstral features, are another important research area (Alam et al., 2014; Sahidullah and Saha, 2012a). The robust features aim to cancel the deteriorative effects of the noises at the feature level.

Compensating the noise at the model/classifier level is also possible (Gales and Young, 1995; Tufekci et al., 2006). However, noise statistics are required forehand in general. Same as the channel noises, it is not possible to know the statistics of every noise source. A common practice to overcome this issue is to estimate the noise statistics by using a noise estimation algorithm, or simply assuming that several frames in the beginning of a record includes only the noise signal.

Speech enhancement is another way to reduce the noisy components of a signal. The basic approach is to detect the noise energy, and subtract it from the noisy speech signal (Boll, 1979). Another milestone for speech enhancement is to continuously track the noise signal (Martin, 1994). Usually, speech enhancement methods are used in hearing aids and cochlear implants. The main reason is that enhancement methods also increase the intelligibility of the signals. The

performance of the speech enhancement algorithms mostly depends on the estimation of the noise statistics.

2.3.3. Other Mismatch Sources

Speaking styles of humans may change due to their emotions. In general, neutral training data are used to obtain speaker models. However, the speaker may be angry, stressed, happy, etc. in the test phase. A mismatch occurs due to these emotional changes, which will affect the SR systems. This variability should be compensated for robustness. On the other hand, emotion recognition from speech is another research area by itself (Anagnostopoulos et al., 2015).

Vocal effort mismatches may also be a problem for practical applications. Vocal effort determines the loudness of the speech. As an example, if a speaker whispers during the training, and shouts during the testing, the recognition performance is expected to be poor. Researchers even investigated effects of singing for speaker clustering systems (Mehrabani and Hansen, 2013).

Reverberation is another mismatch source, which is not taken into account as much as the channel noise, or additive noise. Based on the acoustic environment, the effects of reverberation may decrease the recognition performance. The reverberation problem is addressed in many studies such as (Sadjadi and Hansen, 2014; Zhao et al., 2015, 2014)

2.4. Robust Speaker Recognition

The main mismatch sources were given previously. In this sub-section, VADs, feature based methods, model/classifier based methods, and speech enhancement methods are explored as the main solutions to reduce the unintentional mismatch effects.

2.4.1. Voice Activity Detectors

VADs are essential to separate voice from silence/noise regions. Since there is no speech information in the silence regions of an utterance, it will be beneficial to neglect these parts in terms of recognition performance, and computational load. For a high SNR speech signal, which is called as clean signal, a simple energy based VAD may be sufficient (Kinnunen and Li, 2010). However, as the noise level increases, the speech information is going to be vanished into noise. Therefore, a fixed energy based threshold will either eliminate low energy speech information, or falsely accept high energy noise signal as speech.

Two different thresholds are used in to make a more reliable VAD decision (Woo et al., 2000). Also, a two-band scheme is proposed to prevent the low frequency noise cover the high frequency speech information. The first four frames are assumed to be noise only, and the noise statistics are recursively updated.

The log energy distribution is modelled with a bi-Gaussian in (Bimbot et al., 2004). Gaussian with the highest mean indicated the speech regions. Based on a relative change criteria, GMMs are used in (Sun et al., 2010) as VAD. A VAD based on three discriminative features is proposed for real-time applications (Moattar and Homayounpour, 2009).

Statistical model based methods are proposed to construct VAD classifiers (Gorriz et al., 2008; Jongseo Sohn et al., 1999; Joon-Hyuk Chang et al., 2006; Ramirez et al., 2005, 2004; Yong Duk Cho and Kondoz, 2001). Likelihood ratio test is applied to the features for final VAD decision. Generally, it is assumed that the noise and speech spectra obey the Gaussian distribution. It is shown that Gamma, and Laplacian distributions can be effective than Gaussian distribution (Joon-Hyuk Chang et al., 2006). (Kim et al., 2007) developed a new decision rule, which compares the magnitude of noisy spectral component to a threshold, and improvements over (Jongseo Sohn et al., 1999) are observed. These methods assume that first few frames of an utterance are nonspeech signal. Ying et.al.

proposed an unsupervised approach by using sequential GMM, and updating the models for each frame (Ying et al., 2011).

The VAD used in G.729 standard uses line spectral frequency features, full band energy, low band energy, long-term minimum energy, and zero-crossing rate. This VAD has become a standard baseline for VAD researches. Zero-crossing rate and short-term energy are among the popular VAD features (Jing Pang, 2017; Yiming and Rui, 2015)

Another standard VAD is based on the long-term spectral envelope (Ramírez et al., 2004). It assumes that the voice activity can be detected on the spectrum magnitude. The VAD decision based on the long-term signal probabilities is assigned to the frame that is in the middle of a long segment. (Ghosh et al., 2011) also used long-term signal variability for VAD, but the final decision is obtained over long windows.

(Kotnik, 2003) used a time-domain based VAD, and a frequency-domain based VAD. In the time-domain, short-term energy, and zero-crossing rate are utilized. In the frequency-domain, an SNR value, corresponding to the difference between short-term and long-term log energy estimates, is compared with a threshold.

Features extracted from the raw utterances can also be used in VADs (Hongzhi Wang et al., 2011; In-Chul Yoo et al., 2015; Rabiner and Sambur, 1977). (Sadjadi and Hansen, 2013) used three time-domain features (harmonicity, clarity, prediction gain) and two frequency domain features (periodicity, perceptual spectral flux). Principal component analysis is used to map these features to a one-dimensional space. In addition to the features used in this work, MFCCs and two pitch trackers are used in (Drugman et al., 2016).

Edge detection is utilized to detect beginning and ending edges, considering the raising and descending edges on the energy contours of utterances (Qi Li et al., 2002). Vowel-like regions are used for SV in (Prasanna and Pradhan, 2011). Vowels, semivowels, and diphthongs are less affected by degradations.

Improvements over a fixed energy threshold VAD baseline are observed. This work is enhanced in (Pradhan and Prasanna, 2013) by considering non-vowel-like regions. (Vuppala and Rao, 2013) extracted features from steady vowel regions for speaker identification.

Many examples of ANNs used as VADs can be found in the literature. (Zhang and Wu, 2013) used deep belief networks, where multiple acoustic features are concatenated and used as input. (Drugman et al., 2016) used a single layer with 32 neurons ANN, where the input features are mentioned in the previous paragraphs. Neural fuzzy networks (Wu and Lin, 2000), long short-term memory recurrent neural networks (Eyben et al., 2013), recurrent neural networks (Hughes and Mierle, 2013), convolutional neural networks (Thomas et al., 2014) are among the examples of the ANNs used to detect speech activity. Comparison of robustness between deep learning VADs can be found in (Tong et al., 2016).

(Mak and Yu, 2014) compared several VADs for robust SV and proposed a GMM based VAD. The frames are sorted in ascending order of energy, then 99% of the frames are discarded. Only the frames that are assumed to be correctly classified as speech, or nonspeech, are used to train the GMMs. Another comparison of VADs for SR is given in (Sahidullah and Saha, 2012b). It is reported that bi-Gaussian modeling is the best performing VAD among the state-of-the-art VAD methods.

2.4.2. Robust Features

Extracting features that are less effected from the noises are highly demanded. Since the model/classifier performance are mainly determined by the quality of the features, it is very important have features with less variability under different noise types and levels. Numerous papers in the literature justified the need for robust features. In this sub-section, the latest of them are summarized.

(Garreton and Yoma, 2012) modelled the channel distortion as a p-th order polynomial function, where p is smaller than the number of cepstral coefficients.

Compared to a baseline system, RASTA, and CMN methods, the proposed method achieved better results, although the processing time is increased.

Cepstral features and phonetically discriminative features are combined in (Sarkar et al., 2014). The discriminative features are extracted by a multilayer perceptron. Feature dimension reduction techniques are also applied, and compared to the baseline system, 50% relative improvement is achieved.

Locally normalized cepstral coefficients, based on Seneff's generalized synchrony detector, are proposed in (Poblete et al., 2015). Compensating the spectral tilt provided by the channel response is the main goal of these features. It is reported that these features are also effective against fast varying channels. Also, they can be supported by conventional methods such as CMN, or RASTA.

Normalized dynamic features, which are modified spectral features, are proposed (Chougule and Chavan, 2015). The conventional dynamic features are warped to obtain Gaussian distribution. It is shown that these features achieved higher recognition accuracy then the MFCCs and LP cepstral coefficients.

(Ambikairajah et al., 2015) considered channel bandwidth problem. To use speaker information above 4 kHz, spectral shifting method is proposed. The main idea is to get rid of the frequency band that does not have significant speaker information, then fill this band by shifting the higher frequency parts. By this way, it is assumed that all the speaker information is in the channel bandwidth.

Invariant integration features are used in SV experiments to overcome channel mismatch and additive noises (Alimohad et al., 2014). Improvements over the MFCCs are observed, further performance boost is achieved by fusing both features.

A main drawback for the conventional cepstral features is the noisy spectrum estimation. In (Hanilci et al., 2012), regularized linear prediction is utilized to reduce the mismatch. It is a parametric spectral modeling, where rapid changes in the spectral envelopes is penalized, and smooth spectra are obtained without changing the formant positions. Compared to the standard Fourier

transform and LP methods, better results are reported. To eliminate noise effects, two-dimensional auto-regressive spectra are used in (Ganapathy et al., 2014). Two LP analysis are applied to the speech signal. Constant spectral regions are de-emphasized by dividing a higher order envelope with a lower order one. Cepstral coefficients are then extracted.

Hilbert envelope of gamma-tone filter-bank outputs are used in (Sadjadi and Hansen, 2015). Short-term spectral representation is obtained by windowing smoothed Hilbert envelope frames. Logarithmic compression of the conventional MFCC scheme is replaced with a power-law compression. Results indicated that the proposed features are less susceptible to noise than the MFCCs.

The standard Fourier transform results in a high variance spectrum estimation, which affects the recognition accuracy. Low variance spectrum estimates are observed with multi-tapers (i.e. multiple window functions) (Kinnunen et al., 2012). Theoretical and experimental proves of the variance reduction are reported. Under additive noise, improvements against the Hamming window are achieved.

Ideal Binary Mask (IBM) can be used to segregate the noisy frames (Wang and Brown, 2006). The noisy features can then be recovered or neglected (marginalized). By estimating local SNR of components in the time-frequency matrix representation, An IBM is constructed (May et al., 2012). Feature reconstruction by using sub-bands are examined (Yan et al., 2014). Compared with the full-band method of (May et al., 2012), a higher performance gain is earned.

The performance of the IBM is mostly depending on the accurate noise estimation. In (Ribas González and Calvo de Lara, 2014), several complementary features that are extracted from the spectrum are used to obtain the IBM. The reliable and noisy features are detected by a Bayesian classifier, which is trained with these features. The recognition performance is increased, on the other hand, the computational load is also increased. Multilayer perceptron network is trained to create the IBM in (Zhao et al., 2012). Gamma-tone filter-bank based features are

derived. Also, combining the reconstruction and marginalization modules achieved the best performance.

To avoid the effects of the noisy spectrum on the coefficients, a new feature based on the magnitude difference at two frequency point is developed (Roy et al., 2012). All possible pairs are considered, therefore, Adaboost algorithm is used to choose the most discriminative ones. These features gave better recognition performance than the MFCCs under additive white and pink noises. The noisy spectrum problem tackled in (Sahidullah and Saha, 2012a) by applying a block transformation similar to the DCT. The data divided into blocks to avoid contribution of all sub-band energies for cepstral conversion, then a linear transformation is utilized.

A non-inversible transform is replaced with the Fourier transform of the conventional MFCC framework (Montalvão and Rodrigues Araujo, 2012). Instead of summing weighted energies around a frequency bin, maximum values are chosen to eliminate noise components. Although the computational load is increased, significant performance enhancements are reported. Fractional Fourier transform is utilized in (Ajmera and Holambe, 2013), which is a more suitable transform for non-stationary signals. It is shown that it is more effective than the traditional Fourier transform.

Usually, the speaker and the noise source are in different locations. This spatial difference can be used in multi-channel records. A nonlinear transformation that is effective against nonlinear noises is proposed (Squartini et al., 2012). Prior to the transformation, the noisy cumulative density function is estimated for each channel separately.

Methods do not require prior knowledge about the noise distribution are also studied in the literature. A clean vector is estimated from a noisy test vector, by deriving a conditional probability distribution function from a GMM. Higher performance improvements are obtained when clean and noisy vectors are concatenated to train the GMM. Coefficients with a small amplitude are assumed

to be noise features in (Govindan et al., 2014). A dual threshold shrinkage function is used to truncate the coefficients lower than the below threshold, and to retain the ones higher than the upper threshold. The coefficients between these thresholds are piece-wise linear suppressed.

Multi-style training is considered for an alpha-GMM classifier (Venturini et al., 2014). The multi-style training corrupts the clean training signals to reduce the mismatch between testing features and training features. MFCCs are fused with discrete wavelet transform based features and pH features (Sant'Ana et al., 2006). Speech enhancement methods are also used to further reduce recognition errors.

Based on peripheral auditory system models, a sigmoidal nonlinearity function is used, which is optimized to discriminate between the noisy and clean parts of an utterance (Poblete et al., 2014). The speech signal is passed to a Seneff filter-bank, then sigmoidal compression is applied to the logarithm of the filter outputs. Nevertheless, sigmoidal functions are not effective if the speech and noise spectra are close to each other.

Noisy frames are recovered with the aid of neighboring frames (Huang et al., 2015). To minimize the mismatch between a clean utterance and its degraded version, a transformation matrix is trained. MFCCs, prosodic features, articulatory features, and bigrams are combined to achieve better performance than the MFCCs alone (Drgas and Dabrowski, 2015). It is also reported that the articulatory features' contribution to the performance are less than the other features. In (Daqrouq and Al Azzawi, 2012), wavelet transform is utilized, following LP coefficients calculations for the sub-signals. Average of the frames are taken to reduce the dimensionality.

Vocal effort mismatch problem is tackled by using time-weighted LP modelling (Pohjalainen et al., 2014). LP method with the power spectrum compression/expansion is also shown to be effective against vocal effort mismatch (Saeidi et al., 2016).

2.4.3. Robust Models

Increasing the robustness of speaker models, or classifiers, can also be beneficial for recognition performance in adverse environments. One of the most popular model based compensation method is Parallel Model Combination (PMC) proposed in (Gales and Young, 1995). The core of the PMC is modelling the effects of additive noise based on the noise statistics. Vector Taylor Series (VTS) is another method used to characterize the unknown noises (Moreno et al., 1996). Reliable estimation of the noise statistics is required for these methods to be effective. Other drawbacks are computational loads and requirements of high amount of training data.

In (Moreno et al., 1998), statistical re-estimation algorithms are presented. Similar to the PMC method, it is assumed that the noise effects can be represented as an additive factor to the means and variances of the feature vectors. Once the correction factors are calculated, clean speech vectors can be estimated from the noisy vectors. Another clean speech vector estimation is proposed in (Li Deng et al., 2001), which is independent from any stationarity assumption. Clean feature vectors can be generated from its noisy counterparts of Gaussian components in this method.

The methods mentioned in the previous paragraph associates the clean feature space and the noisy feature space with a bias vector for each GMM component. For multiple types of environment in the noisy space, estimates based on a single GMM may not suffice. In (Buera et al., 2004), noisy features are modeled with individual GMMs by dividing into several environments. The correction vectors are computed for each environment.

Instead of independently modelling noise and speech signals, some researchers suggested using joint probability models. Joint vectors are obtained in these methods by concatenating noisy and clean training vectors. Then, a single GMM is used to model the joint vectors. A clean test vector is estimated by using the GMM and the given noisy test vector. Standard joint probability model based

methods are stereo based stochastic mapping and trajectory based stochastic mapping methods (Afify et al., 2009; Zen et al., 2009). Comparisons of these model based methods for SV can be found in (Sarkar and Sreenivasa Rao, 2014).

Joint factor analysis and i-vector methods are fused to achieve robustness against channel mismatch (Reza et al., 2014). Probabilistic LDA (PLDA) is applied to the i-vectors in (Dehak et al., 2011a) to achieve a SV system without conditioning to a channel type. Multi-condition training is applied to the i-vector extractor and to the PLDA, and robustness against mismatch is achieved (Rajan et al., 2013).

A set of SNR dependent PLDA models are fused in (Pang and Mak, 2015). This work is extended by training mixtures of PLDA with utterances degraded by noise at different SNRs (Mak et al., 2016). Acoustic factor analysis is used instead of the UBM, and more robust i-vectors are extracted (Hasan and Hansen, 2014). Vector Taylor series are integrated in the i-vector extraction framework to model nonlinear distortions (Lei et al., 2013). This work is improved in (Martinez et al., 2014) by replacing the Taylor series with an unscented transform, and better recognition results are obtained with multi-style PLDA.

LDA is replaced by non-parametric discriminant analysis in (Sadjadi et al., 2014), where the between and within class variations are estimated with a nearest neighbor rule. The UBM is replaced by an ANN to extract sufficient statistics for the i-vector (McLaren et al., 2014). Using the prior knowledge about clean i-vectors distribution, a noise compensation method is applied in the i-vector space in (Ben Kheder et al., 2015). It is reported that the proposed method outperformed multi-style training, but the computational load is increased.

2.4.4. Speech Enhancement

Speech enhancement methods' goal is to improve the intelligibility and quality of noisy speech signal. This is achieved by reducing the background noise, hence these methods also known as noise suppression algorithms.

One of the first speech enhancement method is the spectral subtraction (Boll, 1979). In this method, the noise spectrum is estimated, and simply subtracted from the noisy speech spectrum. The remaining spectrum is the clean speech spectrum. The noise spectrum can also be updated during speech absence regions, assuming stationary noise characteristics.

Various algorithms have been developed over the basic subtraction scheme. In the original subtraction algorithm, if the noise spectrum is bigger than the noisy speech spectrum, the result is set to zero (since the spectrum cannot be negative). This situation introduces an artifact known as musical noise. In (Berouti et al., 1979), an overestimate of the noise spectrum is used, while a spectral floor is utilized as the minimum value threshold for the subtraction result. Modulation domain is also considered for spectral subtraction in (Paliwal et al., 2010).

Most noise types affect the speech spectrum nonuniformly. Therefore, the same subtraction factor for all frequency bands may deteriorate the results. Multi-band spectral subtraction is introduced to compensate this problem, and shown to be more effective than the original method (Kamath and Loizou, 2002).

Wiener filtering is another approach for speech enhancement (Abd El-Fattah et al., 2014, 2008; Lim and Oppenheim, 1979; Xia and Bao, 2014). It is a linear filter utilized to recover clean speech signal from the noisy one by minimizing the mean square error between them. Correlation of adjacent frames are also considered with Wiener filter in (Fischer and Gerkmann, 2016). To reduce the noise in multi-channel records, multi-channel Wiener filters are used (Doclo et al., 2007; Spriet et al., 2004).

A DNN architecture is used to map noisy speech to clean speech in (Xu et al., 2014), using multi-condition training with a large amount of training data. Transfer learning approach is utilized in a DNN for speech enhancement in (Wang et al., 2017).

Constrained low-rank and sparse matrix decomposition is used for speech enhancement in (Sun et al., 2014). It is assumed that the noise spectrogram is in a

low-rank subspace, and speech signals are relatively sparse in the time-frequency domain. Also, contrary to the conventional methods, this approach does not require a noise estimation.

The estimation of the noise statistics is vital for the enhancement performance. An obvious method is using a VAD to separate speech and noise frames, then the noise statistics can be calculated, or updated, by using only the noise frames. Some researchers proposed NT algorithms, which eliminates the need for the VAD.

A classical NT algorithm is the minimum statistics (Martin, 1994). It assumes that the speech and noise are statistically independent, and the noise spectrum can be estimated by tracking the minimum of the noisy signal's spectrum. The author improved the algorithm's accuracy for nonstationary noise in (Martin, 2001). Some of the bias compensation methods for minimum statistics is reviewed in (Martin, 2006).

A fast adapting NT algorithm is proposed in (Rangachari and Loizou, 2006). In addition to the minimum statistic methods, speech presence probability is considered. Compared to several other methods, it is reported that much faster adaptation to the noise is achieved. The adaptation time is further reduced in (Hendriks et al., 2008), however, the computational load is increased.

A minimum mean-squared error estimation is utilized in (Hendriks et al., 2010). Although a prior SNR estimate is given in this method, a second SNR estimation is required for bias compensation. In (Gerkmann and Hendriks, 2012), the previous work is modified with a soft speech presence probability decision. A comparison between several noise estimation algorithms can be found in (Taghia et al., 2011).

2.5. Conclusions

The speaker literature is reviewed in this chapter under several sub-sections. For the front-end, the MFCCs are still widely used by the researchers. The

recent review papers about the feature extraction supports this situation (Dişken et al., 2017c; Tirumala et al., 2017). Therefore, MFCCs are preferred in this thesis as the features to be extracted from the speech signals. To model the speakers, both the conventional GMM-UBM and the state-of-the-art i-vector methods are considered.

It is clear that VADs play a fundamental role to achieve robustness, especially under the additive noise. Basically, VADs can be used to separate noise and noisy speech frames. Then, only the noisy speech frames can be used for SV. Also, noise frames can be used to estimate noise statistics, hence a speech enhancement algorithm can be used for further reduce the noise effects. Considering the advantages of a robust VAD, a PR based VAD algorithm is developed in this thesis work. The next chapter gives the details of the proposed algorithms.

3. MATERIAL AND METHODS

In this chapter, the speaker, and the noise databases used in the SV experiments are introduced. Then, the performance metrics used to examine the performances of the methods are explained. The details of selected feature extraction (MFCC) and speaker modeling (GMM-UBM, i-vector) methods are given.

The proposed algorithms are introduced after these fundamental concepts. First, SMC to obtain impostor models are described, then the VAD based on the PR is demonstrated. Two methods, selected for performance comparison under the additive noise, from the literature are explained briefly. One of them is a VAD that uses ANN, the other is a NT algorithm with short adaptation delay. The NT algorithm is used to enhance the speech, then a simple energy based VAD is used for a fair comparison.

A real-time text-dependent SV system is realized on a single board computer, Banana Pi. This real-time system is briefly explained as a case study.

3.1. Database and Toolkits

National Institute of Standards and Technology (NIST) of United States organized Speaker Recognition Evaluation (SRE) series, which are highly contributed to the research interest in text-independent SR since 1996. For each evaluation, a speaker database is distributed to the participants. Each year, different aspects were emphasized, such as channel mismatch, duration mismatch, language recognition, multi-speaker recognition etc.

In this thesis, NIST SRE 1998 database is used (Doddington et al., 2000). This database consists of 250 speakers for each gender. For each speaker, approximately five minutes of training data is available.

Also, channel mismatch is present between some of the speakers' training and testing data. The channel mismatch is due to the different microphone types

used in the training and testing sessions. The microphone types are electret, and carbon. So, if a speaker's training and testing data are recorded with the same microphone type, there is no microphone mismatch. However, if the microphone types changes, a channel mismatch occurs.

Testing data includes 3, 10, and 30 seconds durations. Therefore, it is possible to observe the proposed methods performances for different speech durations. For each test duration, there are 1308 test speech files for male speakers, and 1379 test speech files for female speakers in the matched handset condition. For the mismatched handset condition and for each test duration, there are 1192 test speech files for male speakers, and 1121 test speech files for female speakers Also, for each test file, there is one trial for the target speaker, and nine trials for the non-target speakers. The target speaker's speech data is used as an impostor data for these non-target speakers' models.

Additive noises are chosen from the widely used noise database named NOISEX-92 (Varga and Steeneken, 1993). Five different noises (F16, Lynx, Car, Babble, and Stitel) are added to the test speech files. The SNR values are varied from -10 dB to 10 dB with 5 dB steps.

A short utterance spoken by a male speaker is chosen from the NOIZEUS corpus as a sample speech signal (Hu and Loizou, 2007). This sample signal is used to analyze the proposed VAD algorithm's working principle, which is given later. The selected signal is clean, i.e. no additive noise is present. It is degraded by Lynx noise in the case studies, and some of the VAD steps are illustrated for a better understanding.

Hidden Markov Model Toolkit, known as HTK, is a trusted toolkit for feature extraction, HMM training, and GMM training purposes (Young et al., 2000). In this thesis, it is used to extract MFCC features from the speech signals. A Linux based operation system, Ubuntu, is used. The features extracted by the HTK are used in the proposed speaker clustering algorithm. Except the MFCC

extraction, all other processes (UBM training, model adaptation, clustering, scoring) are done with the codes written in the C++ programming language.

The proposed VAD algorithm is realized in the MATLAB program. The MFCC extraction, and operations related with the VAD are written in MATLAB. MSR Identity Toolbox is used for the GMM-UBM and i-vector back-end methods.

3.2. Performance Metrics

A performance criterion must be defined to examine the recognition system's accuracy. Several metrics can be found in the literature, however, two of the most widely accepted metrics are selected for this thesis. One of them is the Equal Error Rate (EER), which is a very popular metric in biometric authentication. The other is the Detection Cost Function (DCF), which is an important metric for the NIST SREs.

Equal Error Rate (EER) can be defined as the point where false acceptance rate and false rejection rates are equal. In terms of SV, the false acceptance means the impostor speakers falsely accepted as claimed speakers. Contrary, the false rejection indicates the genuine speakers misclassified as impostors. A test utterance is scored by using the claimed speaker's model, and if the score is bigger than the EER threshold score, the unknown speaker is verified. If the score is less than the EER threshold score, the unknown speaker is rejected.

In practical applications, if the security is main concern, the threshold can be set to a higher value than the EER. This way, the system will be more robust to impostor attacks. Nevertheless, the probability of rejecting true speakers will also be increased. The EER concept is visualized in Figure 3.1.

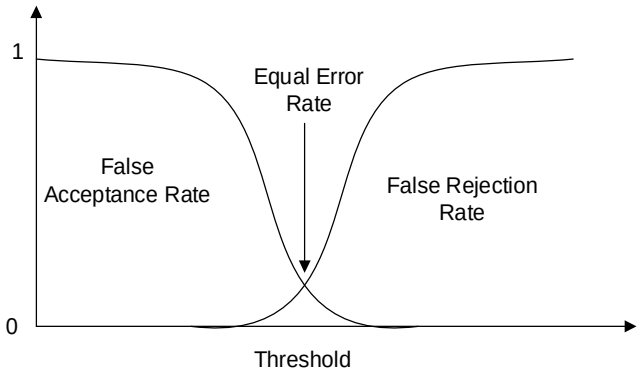

Figure 3.1: Visual representation of EER

The DCF is defined as given in Equation (3.1) below.

$$DCF = C_{FA}P_{FAR|N}P_N + C_{FR}P_{FRR|T}P_T \quad (3.1)$$

Where $P_{FAR|N}$ is the False Acceptance Rate (FAR), $P_{FRR|T}$ is the False Rejection Rate (FRR), cost of the false acceptance is $C_{FA}=10$, cost of the false rejection is $C_{FR}=1$, the prior probability of target tests is $P_T=0.1$, and the prior probability of nontarget tests is $P_N=0.9$. This cost function penalizes the false acceptance of the impostors. The minimum of the DCF is used as the performance metric (minDCF). A lower value of both EER and minDCF indicates a high recognition performance. Therefore, when comparing different systems, their performances will be compared based on the EER, or minDCF, value.

3.3. Mel-Frequency Cepstral Coefficients

Mel-Frequency Cepstral Coefficients (MFCCs) are widely used in speech processing area, despite the fact that they are proposed a few decades ago. The recent literature review on feature extraction methods prove that MFCCs' popularity is yet to be deceased (Dişken et al., 2017c; Tirumala et al., 2017). In this

thesis, MFCCs are chosen as the features to be extracted. Therefore, it will be beneficial to summarize the conventional MFCC extraction framework.

As with the most short-term features, to extract the MFCCs the raw speech signal is divided into overlapping frames. These frames' durations are typically between 10 and 30 milliseconds. In this range, the speech signal is assumed to be stationary. A window function such as Hamming is applied to these frames. The main purpose of the windowing is to reduce frequency leakage when taking the Fourier Transform (FT).

The FT is used to transform the time domain signal into the frequency domain. Although several studies benefited from the phase spectrum, it is neglected in the conventional scheme. The magnitude spectrum is passed to a bank of overlapping band-pass filters. The MFCCs specialty is due to this filter-bank. The filters are linearly placed in the Mel scale. The Mel scale is constructed to mimic the human auditory system, which is linear up to 1 kHz, and logarithmic at the higher frequencies. The mathematical relation between the Mel scale and Hertz is given in Equation (3.2).

$$m = 2595 log_{10}(1 + \frac{f}{700}) \tag{3.2}$$

where f is the frequency in Hertz, and m is the Mel scale. The Mel scale filter-bank allows the representation of the lower frequencies with a higher resolution. In terms of the conventional spectrum, filters with narrow bandwidth are placed in the lower frequencies, and higher bandwidth filters are placed in the higher frequencies.

The magnitudes, or energies, inside the filters are summed, and then compressed with log operation to mimic human's audio perception. Finally, Discrete Cosine Transform (DCT) is applied to the log filter-bank values to de-correlate the coefficients. This step can be interpreted as taking the spectrum of the

spectrum. In the literature, this new spectrum is named as cepstrum. The zeroth coefficient is related to the signal energy. Usually it is discarded, and some of the following coefficients are used. Nevertheless, some researchers use the logarithm of the zeroth coefficients. Mathematical expression for the given operations are shown in Equation (3.3).

$$c_n = \sum_{m}^{M} \left(logY(m)\right) \cos\left[\frac{\pi n}{M}\left(m - \frac{1}{2}\right)\right], \quad m = 1, \dots, M \tag{3.3}$$

where *Y(m)* denotes the output of *M*-channel filter-bank, *M* is the number of filters used in the filter-bank, *m* is the filter-bank index, *c* denotes the static coefficients, and *n* is the coefficient index.

After these operations, static coefficients are obtained. To decrease the negative effects of slowly varying channel noises, and to incorporate temporal information, additive processes such as delta features can be appended to the static coefficients. The delta features can be calculated by using Equation (3.4).

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2\sum_{n=1}^{N} n^2} \tag{3.4}$$

where d_t is the delta coefficient of frame *t*, *N* is the number of neighbor frames, and *c* is the static feature. Double delta, or delta-delta, features can be also calculated with the same equation. To calculate delta-delta features, the static coefficients must be replaced with the delta coefficients.

The conventional MFCC extraction scheme is summarized as a block diagram in Figure 3.2 below. In the experiments, 13 coefficients excluding the zeroth, and their deltas are used as the feature vector.

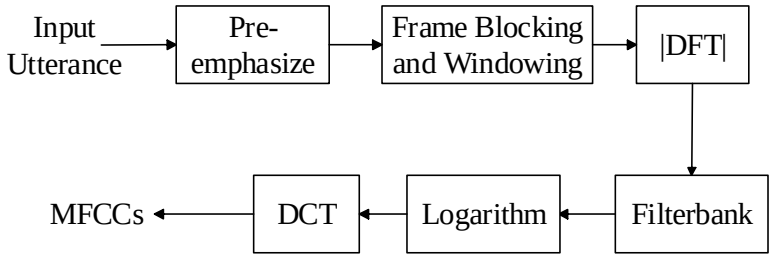

Figure 3.2. Conventional MFCC extraction scheme (Dişken et al., 2017c)

Figure 3.3 shows examples of MFCC features. The HTK toolkit is used to extract these features from the sample speech signal taken from the NOIZEUS corpora. Figure 3.4 is the configuration file, which is given to the HTK. It consists of the parameters to be used in the MFCC extraction process. As an example, the first line of the configuration file indicates that zeroth coefficient is added to the static coefficients. The previous figure verifies this fact, as the last coefficients are bigger than the others.

The HTK toolkit offers various parameters to modify. In the given example, frame length, frame shift, number of filters in the filter-bank, number of statics coefficients, window type, lowest and highest frequencies of the filter-bank, format of the raw speech signal, etc. are shown. If a parameter is not written in the configuration file, its default value is used.

Since the toolkit does not have a graphical user interface, all commands are executed via command line. The MFCC values are given here as an example to illustrate how the toolkit is used. Although the HTK toolkit is a trusted program for speech recognition, it lacks the state-of-the-art SR methods. Also, it lacks visualization oppurtunities. Therefore, MATLAB is also used besides the HTK toolkit.

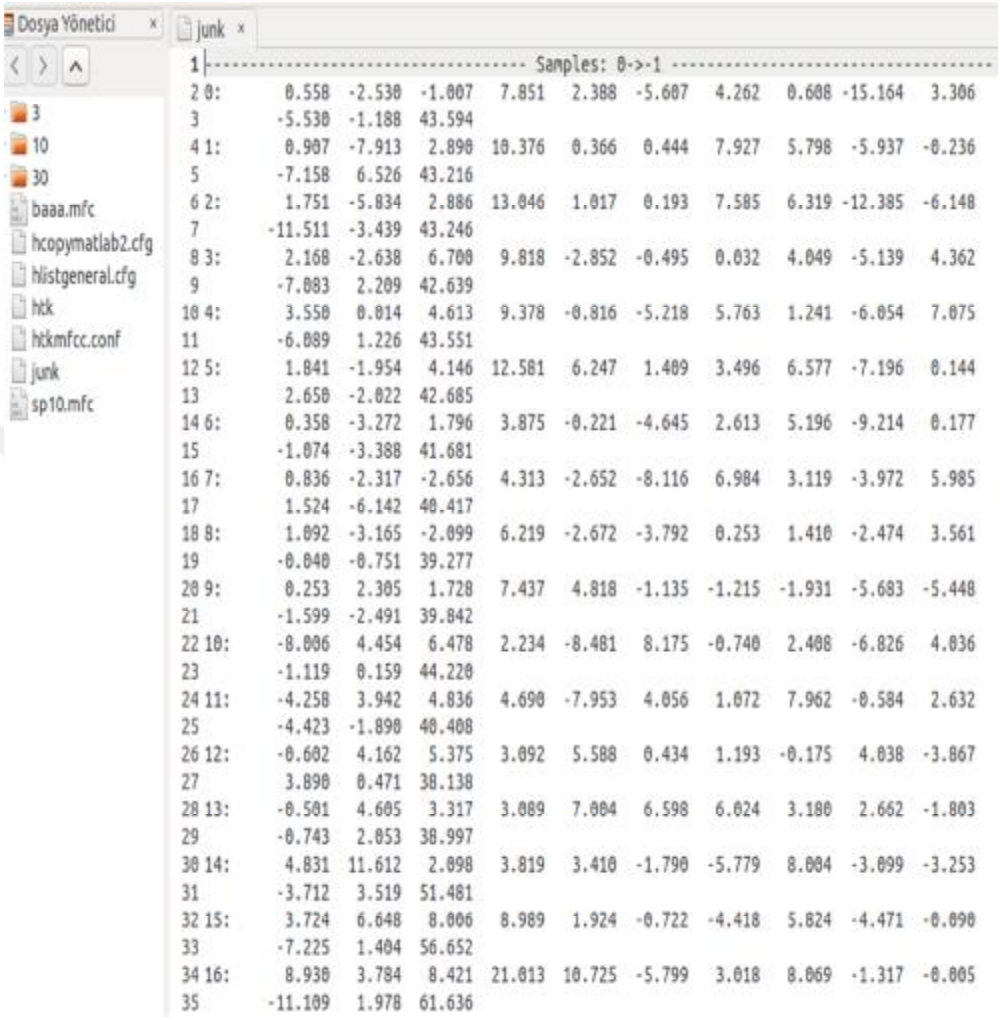

Figure 3.3. MFCC features of the first eighteen frames, extracted with the HTK toolkit

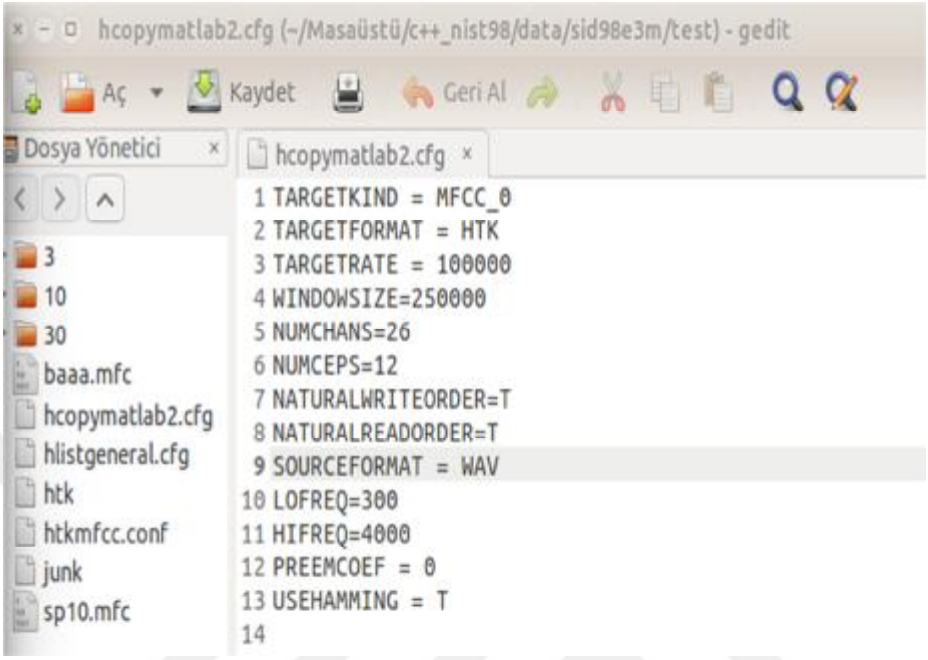

Figure 3.4. Configuration parameters of the MFCCs extracted with the HTK toolkit

Some of the outputs obtained within the MFCC extraction steps are visualized in Figure 3.5, by using one of the test data is chosen from the NIST SRE 1998 corpus. The 1000th frame is chosen for the analysis Figure 3.5(a). Figure 3.5(b) shows the magnitudes of the Fourier transform. Figure 3.5(c) is the result of filter-bank aplied to the magnitude spectrum. Finally, Figure 3.5(d) illustrates the static MFCC coefficients.

The MFCCs take smaller values as the coefficient index increases. Note that twenty-six filters are used in the filter-bank, but only thirteen MFCC coefficients are extracted, excluding the zeroth. The reason is for this situation is that the smaller coefficients' impact on the recognition accuracy is neglegible.

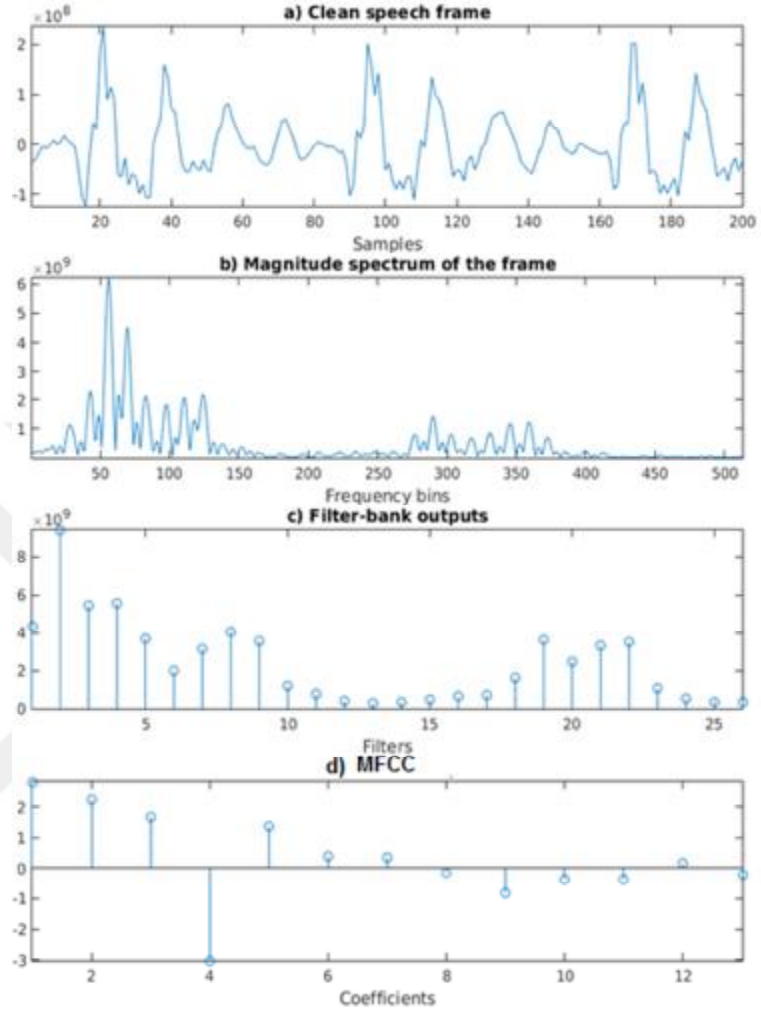

Figure 3.5. a) 1000th frame of the clean speech signal, b) Magnitude spectrum obtained with Fourier transform, c) Mel filter-bank magnitudes, d) MFCC coefficientsexcluding the zeroth

To illustrate the effects of the additive noise on the MFCCs, the same speech signal is degraded by the Lynx noise, taken from the NOISEX-92 database. Figure 3.6 illustrates the speech signal (top), and the degraded signal with a 10 dB overall SNR. As seen in the figure, the regions with high amplitudes preserved their shapes. However, lower amplitudes are vanished in the noise signal.

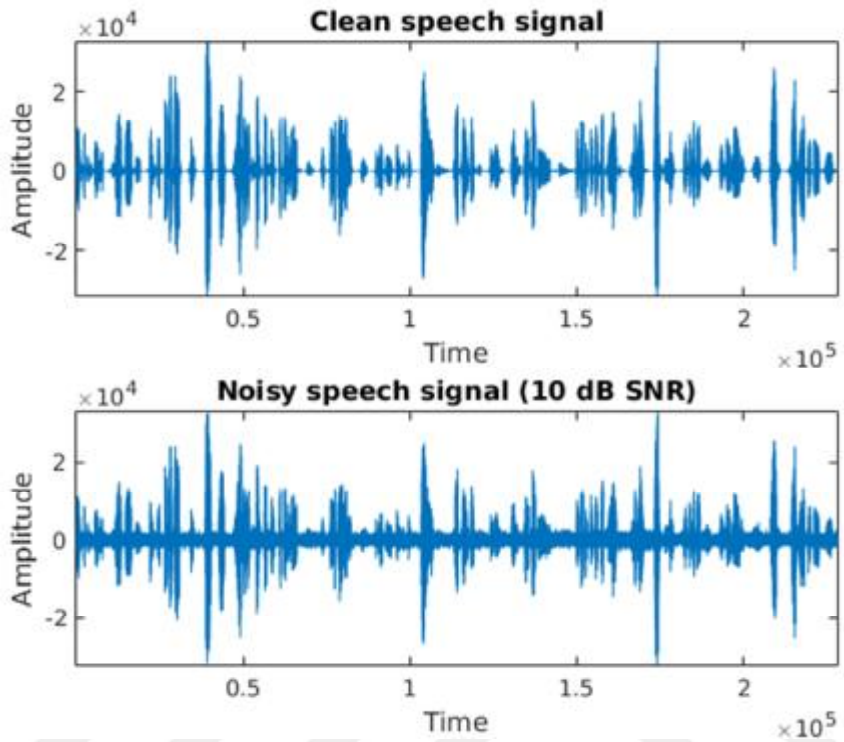

Figure 3.6. Clean speech signal (top), and the same signal degraded with the Lynx noise (bottom)

Figure 3.7 shows the 1000th frame, taken from the clean speech signal, and the MFCCs extracted from it. The signal seems to be a periodic one, and has high amplitudes. Therefore, the frame is clearly dominated by speech information.

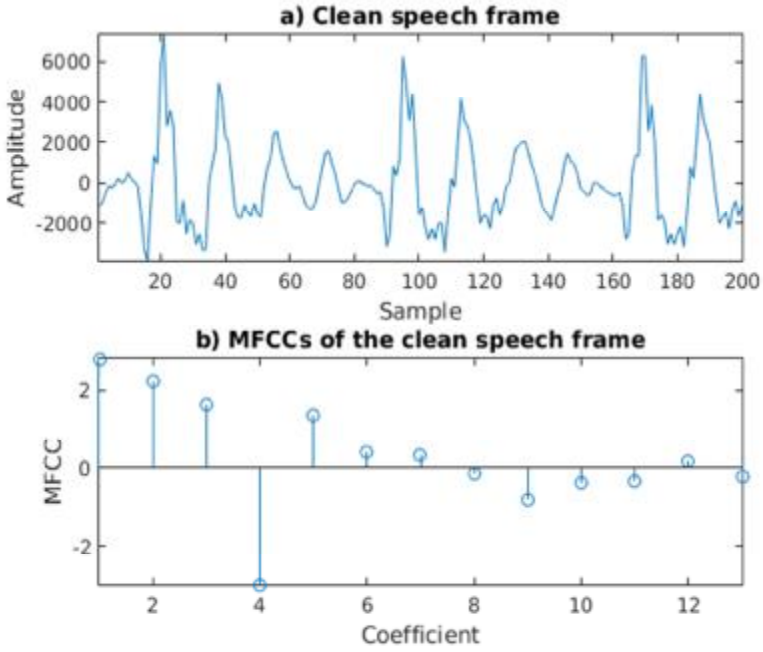

Figure 3.7. 1000th frame of the clean speech signal (top), and its MFCC coefficients (bottom)

Since the frame shown in Figure 3.7 is a high energy frame, the negative effects of the additive noise will be modest. To prove this fact, the same frame (1000th) under the additive noise is given in Figure 3.8. As seen in the figures, the shape of the signals is almost the same. As a consequence, the extracted MFCCs are very similar. These figures prove the usefulness of the high energy speech frames for the recognition.

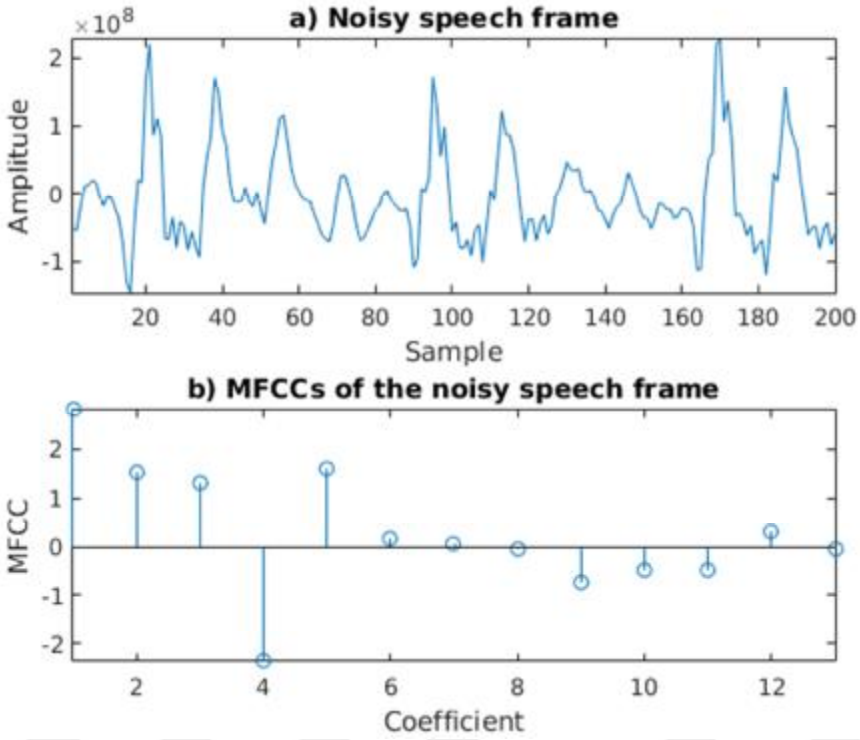

Figure 3.8. 1000th frame of the noisy speech signal (top), and its MFCC coefficients (bottom)

The low energy signals on the other hand, are much more effected from the additive noise. Figure 3.9 shows the 900th frame of the same signal (top), and its respective MFCC coefficients (bottom). This frame is not seem like a periodic signal, and has a lower amplitude compared to the signal given in Figure 3.7. This frame has a high probability to be a silence frame, or a low energy speech frame such as unvoiced phoneme.

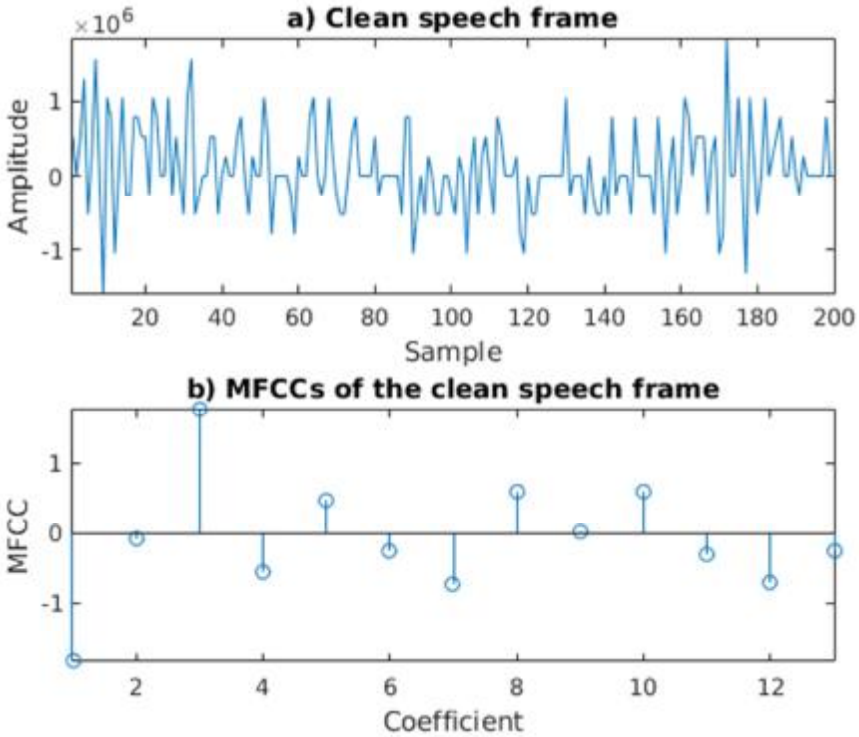

Figure 3.9. 900th frame of the clean speech signal (top), and its MFCC coefficients (bottom)

The same frame (900th) under the additive noise is given in Figure 3.10. The effects of the additive noise can be clearly observed. The signal's temporal shape is completely changed. Further, the MFCCs extracted from the same frame is highly disturbed. This mismatch between the coefficients degrades the recognition performance of the systems. The novel VAD proposed in this thesis aims to detect the less affected, high energy speech regions to reduce the mismatch between the clean training signals and the noisy testing signals.

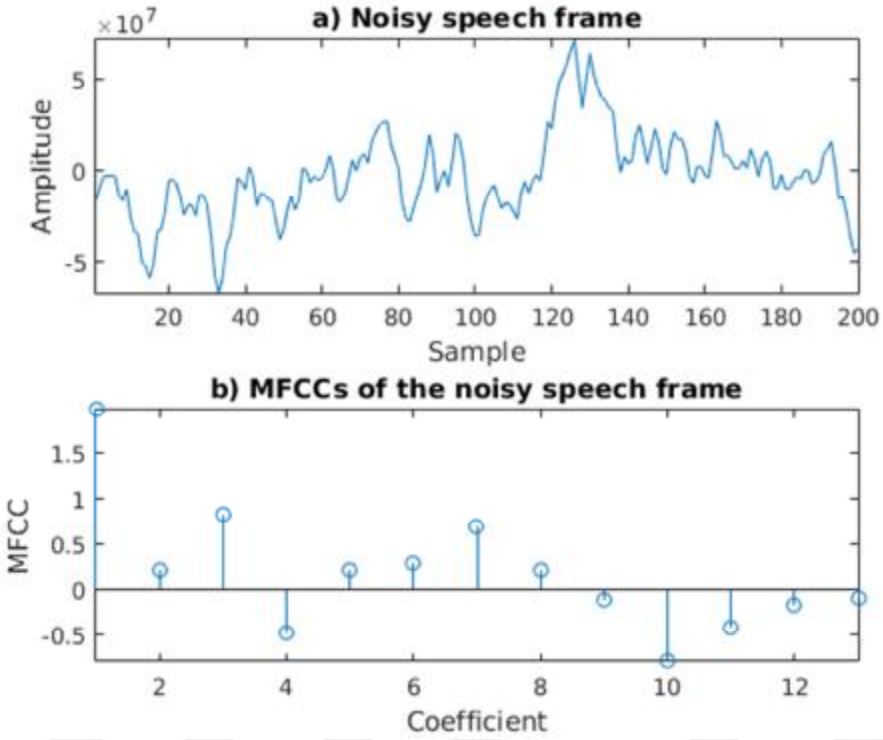

Figure 3.10. 900th frame of the noisy speech signal (top), and its MFCC coefficients (bottom)

3.4. Universal Background Model

The conventional speaker modeling method known as GMM-UBM is a fundamental tool that has been the state-of-the-art back-end for text-independent SR. Further, the recent developments are mostly based on the UBM, as discussed in the literature review chapter. Also, the speaker modeling method proposed in this thesis is an extension to the traditional GMM-UBM method. Therefore, details of the UBM method is given in this sub-section.

A GMM is defined by its mixture parameters, i.e. mixture weights, mean vectors, and covariance matrices. An M mixtures model can be notated as given in Equation (3.5).

$$\lambda = \{p_i, \vec{\mu}_i, \Sigma_i\} \quad i = 1, \dots, M \tag{3.5}$$

where λ is the GMM model, p_i is the weight, $\vec{\mu}_i$ is the mean vector, and Σ_i is the covariance matrix of mixture i, respectively. The covariance matrix is usually diagonal, which reduces the computational loads. Also, using a full covariance matrix does not make substantial performance differences (Reynolds, 1995). Also, the weights in a GMM must sum to one (Equation (3.6)).

$$\sum_{i=1}^{M} p_i = 1 \tag{3.6}$$

Let $\vec{x}$ be a D-dimensional observation vector. Its Gaussian density for the i-th mixture, $b_i(\vec{x})$, can be calculated by using Equation (3.7) given below

$$b_i(\vec{x}) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|} exp\left\{-\frac{1}{2}(\vec{x} - \overrightarrow{\mu_i})'\Sigma_i^{-1}(\vec{x} - \overrightarrow{\mu_i})\right\} \tag{3.7}$$

The mixture density is then found as the weighted sum of all component densities (Equation (3.8)).

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} p_i b_i(\vec{x}) \tag{3.8}$$

where $p(\vec{x}|\lambda)$ denotes the mixture density of the observation vector $\vec{x}$, given the model λ.

A UBM model can be trained by using the expressions given above. The UBM is intended to represent the acoustic space of all available speakers, so it is speaker independent. In many practical situations, a speaker's data may not be

sufficient for effectively training his/her own GMM. The speaker's model can be adapted from the well-trained UBM by using the available speaker dependent data. The mathematical expressions used for speaker adaptation are given in the following.

Let $S = \{\vec{s}_1, \vec{s}_2, \vec{s}_3, \dots, \vec{s}_T\}$ be the speaker dependent training feature vectors, where T is the frame index. The probabilistic alignment of these vectors into the UBM components are calculated by using Equation (3.9).

$$P(i|(\vec{x}_t) = \frac{w_i p_i(\vec{x}_t)}{\sum_{j=1}^{M} w_j p_j(\vec{x}_t)} \tag{3.9}$$

The sufficient statistics for the weight, mean, and variance parameters are calculated as given in Equations (3.10)-(3.12).

$$n_i = \sum_{t=1}^{T} P(i|\vec{x}_t) \tag{3.10}$$

$$E_i(\vec{x}) = \frac{1}{n_i} \sum_{t=1}^{T} P(i|\vec{x}_t)\vec{x}_t \tag{3.11}$$

$$E_i(\vec{x}^2) = \frac{1}{n_i} \sum_{t=1}^{T} P(i|\vec{x}_t)\vec{x}_t^2 \tag{3.12}$$

These statistics are used to update the UBM statistics for the *i*-th mixture, hence an adapted model is created. All of the mixture parameters, i.e. weights, means, and variances) can be adapted. However, it is shown that adapting only the mean vectors are found to be more effective (Reynolds et al., 2000). Therefore, the adopted approach in the SR literature is to adapt the mean vectors, and use the

same weights and variances of the UBM mixtures in the speaker's model. Equation (3.13) is used to calculate the adapted mean parameters ($\hat{\mu}$) of the speaker model.

$$\hat{\mu} = \alpha_i E_i(\vec{x}) + (1 - \alpha_i)\,\overrightarrow{\mu_i} \tag{3.13}$$

where α represents a data-dependent adaptation coefficient, which controls the balance between the UBM means and the estimated means. It is calculated by using Equation (3.14) given below.

$$\alpha_i = \frac{n_i}{n_i + r} \tag{3.14}$$

where r is a fixed relevance factor parameter, usually 16. This adaptation creates a relation between the UBM and the speaker models. The UBM is used as the model of all possible impostors. In the test stage, an unknown utterance is first scored with the UBM. Mixtures with the highest scores are detected. Indexes of a few number of top scoring mixtures are determined, usually top five mixtures. Then, the unknown utterance is scored with only these mixtures of the speaker's model.

As an example, if the models have M mixtures, $M+5$ mixtures are considered instead of $2M$. Therefore, the adaptation process offers a fast scoring scheme. If the score of the UBM is higher, the unknown speaker is rejected, and vice versa.

3.5. I-vector

Initially proposed in (Dehak et al., 2011b), the i-vector modelling method has become very popular in speech processing research, especially for the text-independent SR. The i-vector method assumes that speaker and channel variability can be modelled in a low dimensional space called total variability space. In another view, the i-vector is a compressed form of the supervector. A supervector

is obtained by concatenating the mean vectors of a GMM. If the feature vector dimension is *D*, and the number of components in the GMM is *M*, then the resulted supervector's is *MDx1*. Considering that there are usually 1024-2048 mixtures in the UBM, and the feature vector length varies between 20 and 60, the resulted supervector's dimension becomes very high.

The compression applied to the supervector enables a much lower dimensional representation. The new representation is named as intermediate sized vector, hence i-vector. Researchers usually prefer i-vector lengths between 100 and 1000. This approach gives the opportunity to map an unknown length utterance to a low dimensional space. As an example, two different utterances with 3-seconds, and 1-minute durations can be both modelled as 200-dimensional i-vectors.

Another important aspect of the i-vector is that each utterance is considered as coming from a different speaker. Therefore, to model a speaker, i-vectors are extracted for each of his records. Then, their average is taken, and the resulted i-vector represents the speaker.

The main mathematical expressions are given below. Consider a trained UBM model, $\lambda = \{p_i, \vec{\mu}_i, \Sigma_i\} \quad i = 1, \dots, M$, as described previously, and an utterance from speaker S with a T frame feature sequence as $S = \{\vec{s}_1, \vec{s}_2, \vec{s}_3, \dots, \vec{s}_T\}$. By using the UBM, the zeroth *(Nm)*, and the centered first order ($\vec{F}_m$) statistics can be calculated as following equations, respectively.

$$N_m = \sum_{t=1}^{T} P(m|\vec{s}_t, \lambda) \tag{3.15}$$

$$\vec{F}_m = \sum_{t=1}^{T} P(m|\vec{s}_t, \lambda)(\vec{s}_t - \vec{\mu}_i) \tag{3.16}$$

where m is the mixture index, and $P(m|\vec{s}_t, \lambda)$ is the probability of obtaining the given feature vector from the mixture m. The centered mean supervector $\tilde{F}$ is obtained by concatenating $\vec{F}_m$ for all mixtures. The goal of the i-vector analysis is to project this supervector on a low rank factor loading matrix, $\boldsymbol{T}$, following the factor analysis framework. $\boldsymbol{T}$ is a rectangular matrix sized *MFxK*, where *K<<MD*. $\boldsymbol{T}$ is also called i-vector extractor, or total variability matrix. The training process of the $\boldsymbol{T}$ matrix is the same as training the eigenvoice matrix given in (Kenny et al., 2005). Usually, $\boldsymbol{T}$ is iteratively updated in 20 iterations. Once the centered mean supervector, and total variability matrix are obtained, the i-vector is computed as follows.

$$\vec{x} = (\boldsymbol{I} + \boldsymbol{T^t \Sigma^{-1} N T})^{-1} \boldsymbol{T^t \Sigma^{-1} N \tilde{F}} \quad (3.17)$$

where $\boldsymbol{N}$ is a diagonal matrix of *MDxMD* whose diagonal blocks are $\boldsymbol{N_m I}$, $\boldsymbol{I}$ is the identity matrix, Σ is the diagonal covariance matrix of dimension *MDxMD* estimated in the total variability matrix training. Its purpose is to model the residual variabilities that are not captured by the total variability matrix.

Several channel compensation methods can be applied to reduce the channel variability (Dehak et al., 2011b). In this thesis, Linear Discriminant Analysis (LDA) is used within the MSR Identity toolbox. LDA is mainly used for dimensionality reduction. Orthogonal axes between classes that maximize between-class variance, and minimize intra-class variance are detected. The data are then projected on the new dimensional space, separating the classes from each other.

The LDA can be summarized in a few steps. First, scatter matrices for between-class and within-class are computed, and corresponding eigenvectors and eigenvalues are calculated for each matrix. A number of eigenvectors with the largest eigenvalues are selected to form a matrix, where every column represents

an eigenvector. The obtained matrix is simply multiplied by the feature matrix, hence the samples are transformed into a lower subspace.

In the test phase, an unknown utterance's i-vector is scored with the claimed speaker's i-vector. Probabilistic LDA is used for this purpose, which is analogous to the LDA method. The i-vector is assumed to be generated from a Gaussian density.

3.6. Speaker Model Clustering

The conventional GMM-UBM method uses one model (UBM) to represent all of the impostor speakers. However, using several impostor models may increase the recognition accuracy. For this purpose, speaker models are clustered, and impostor models are created for each cluster. This process can be viewed as dividing the space represented by the UBM into several subspaces. Hence, the proposed speaker clustering method is an extension of this conventional modelling approach.

The adapted speaker models are clustered with the well-known K-MC algorithm. Assume that a UBM is trained, and speaker models are derived from it by adapting the mean vectors. Once the speaker models are obtained, their mean vectors are element-wise divided by their respective standard deviation vectors for normalization as shown in Equation (3.18),

$$\tilde{\mu}_{i,s} = \frac{\vec{\mu}_{i,s}}{\vec{\sigma}_i} \tag{3.18}$$

where i is the mixture index, s is the speaker index, $\tilde{\mu}_{i,s}$ is the normalized mean vector of the i-th mixture for speaker s, and $\vec{\sigma}_i$ is the standard deviation vector of the i-th mixture. Note that the standard deviation vector is speaker independent. Since the speaker models are derived by adaptation from the UBM, they share the same standard deviation vector.

Mean supervectors are constructed for each speaker by concatenating the normalized mean vectors of the models for each mixture, $\tilde{\mu}_s = \{\tilde{\mu}_{1,s}, \tilde{\mu}_{2,s}, \dots, \tilde{\mu}_{M,s}\}$. These supervectors, which represent speaker models, are clustered by using the traditional k-means algorithm. The similarity measure used in the clustering is the Euclidean distance. The formal expression of clustering is given in Equation (3.19)

$$J_{s,c} = \sum_{s=1}^{S} ||\tilde{\mu}_s - \vec{v}_c||^2 \tag{3.19}$$

where c is the cluster index, $\vec{v}_c$ is the vector representing the centroid of cluster c, $\tilde{\mu}_s$ is the mean supervector of speaker s, S is the total number of speakers, and $J_{s,c}$ represents the distance of speaker s to the cluster c. The distances of each speaker to each of the clusters can be calculated by this way. Each speaker is then assigned to the cluster which gives the minimum distance value. Once all the speakers are assigned to the closest cluster, the centroids of the clusters are calculated by using Equation (3.20)

$$\vec{v}_c = \frac{1}{N_c} \sum_{s=1}^{N_c} \tilde{\mu}_{s,c} \tag{3.20}$$

where N_c is the number of speakers assigned to the cluster c, and $\tilde{\mu}_{s,c}$ is the mean supervector of the speakers assigned to the same cluster. The initial values of the cluster centroids are chosen from the speaker supervectors, depending on the number of clusters. As an example, if the speakers are going to be clustered into two classes, two speaker supervectors are chosen as the initial cluster centroids. Then, by using equations (3.19) and (3.20), the centroids are recalculated iteratively.

Once the cluster centroids are fixed (i.e. when negligible changes occur between iterations, or a predetermined number of iterations are executed) final values of cluster centroids ($\vec{v}_c$) are decomposed into mixture mean vectors. Also, a piece-wise multiplication by the standard deviation vector of their respective components is applied (Equation (3.21)).

$$\vec{\mu}_{i,c} = \vec{v}_{i,c}\vec{\sigma}_i \tag{3.21}$$

where $\vec{v}_{i,c}$ is the i-th mixture's mean vector of cluster c, $\vec{\mu}_{i,c}$ is the final (de-normalized) values for the i-th mixture's mean vector of cluster c. The expectation of these process is gathering the speakers sharing the similar acoustic space in the same group by clustering their supervector models. This shared acoustic space will be the impostor model for the speakers in that group. The scoring phase of the proposed method is given in Figure 3.11 with the conventional UBM approach for comparison.

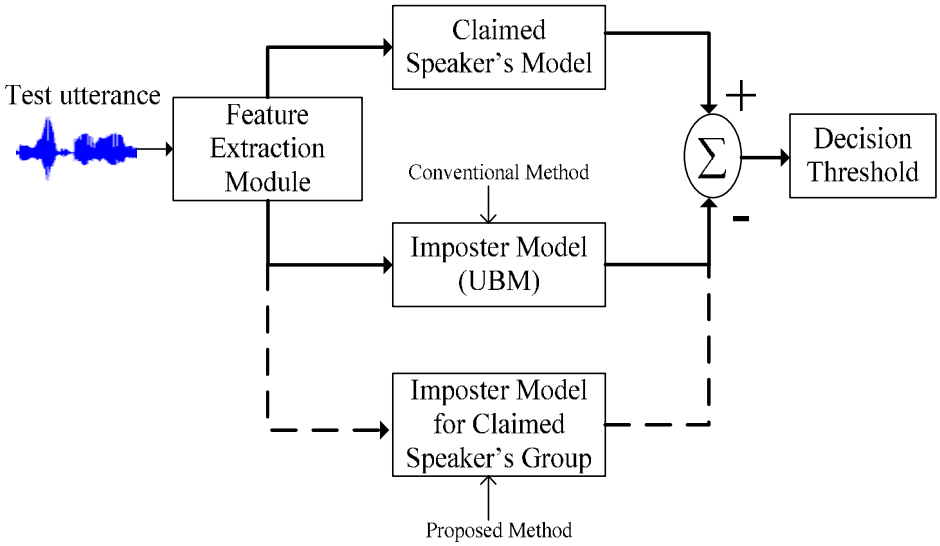

Figure 3.11. Scoring algorithms of the UBM (solid line), and the proposed clustering method (dashed line) (Dişken et al., 2017a)

The proposed algorithm can be viewed as a combination of the UBM and cohort methods. The cohort models, which are used to detect impostors, can be generated by combining the speakers closest to the target speaker's acoustic space. In this approach, a cohort model for each individual speaker is constructed. It is clear that this method needs more memory to store cohort models, and the scoring of test utterances may not be fair (since a speaker's cohort model may not accurately define the acoustic space around him.).

As reported in (Reynolds, 1997), the UBM method performs better than the cohort modeling. Therefore, it is extensively used in the literature, as it offers some other benefits discussed previously. However, cohort impostor models are still investigated by the researchers (McLaren et al., 2010; Zhu et al., 2011). Nevertheless, combining the UBM and the cohort is considered in the score space by utilizing the SVM to find an optimum decision value (Brew and Cunningham, 2010, 2009). However, the conventional GMM-UBM scoring framework is used in the proposed method. It is argued that if the performance of the traditional method is increased, it is also possible to achieve a higher performance gain with more complicated methods (i.e. SVM with mean supervectors, i-vectors, etc.)

The cluster centroids (impostor models) represent the acoustic space defined by the speakers in the cluster, not the acoustic space of all possible speakers, which is the case with the UBM. By this approach, a cohort-like representation is therefore achieved. Another advantage is that there is no need to create impostor models for each speaker, since an impostor model is shared between the speakers in the same cluster. The computational and memory loads are also reduced with the proposed clustering approach. Figure 3.12 shows the proposed clustering algorithm as a block diagram.

Speaker clustering algorithms generally used to achieve speed ups in speaker identification, as discussed in the introduction chapter. There is a trade-off between the speed and the identification accuracy. The main reason of this trade-off is that some of the speakers, or mixtures, are not considered in the scoring

phase. In the proposed method on the hand, speed is not the main concern. Since only the claimed speaker's model, besides the impostor model, is taken into account, all the model mixtures can be scored, hence no trade-off occurs.

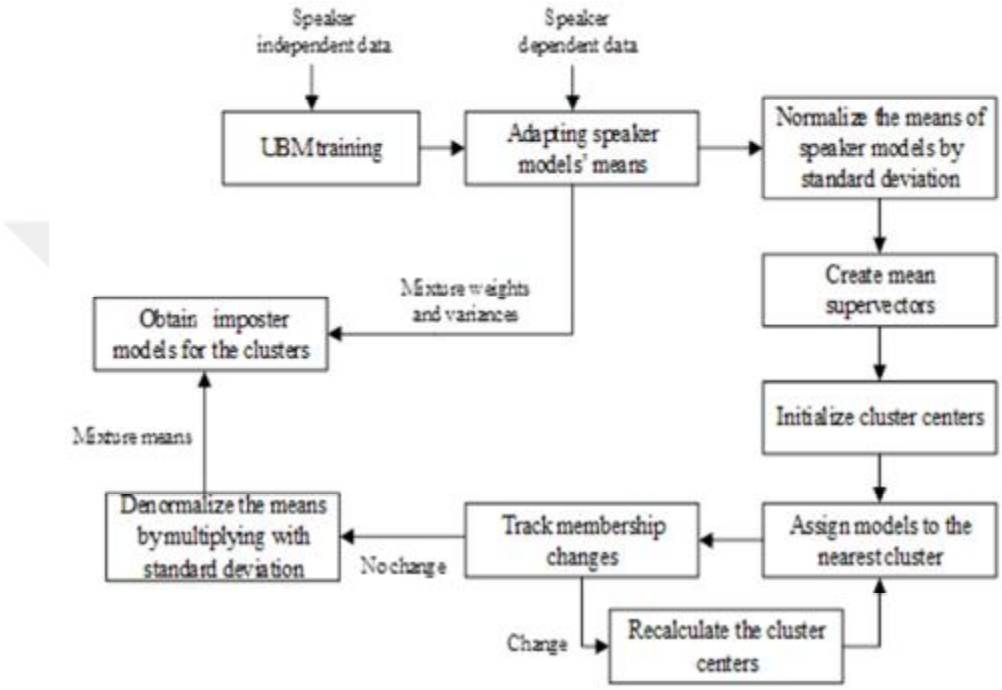

Figure 3.12. Block diagram of the proposed speaker clustering algorithm (Dişken et al., 2017a)

The proposed clustering approach is not suitable for speaker identification. Since the imposter models used for speakers vary, the identification process will take longer time. However, since an identity must be claimed in SV, there is no difference in the scoring time copmared to the UBM. As the cluster of the claimed speaker is known from the training phase, the impostor model of the respective cluster is used as the UBM.

When an impostor that has a voice characteristics similar to the claimed speaker, the system will probabily misclassify his identity. The reason for this misclassification is the distance between the mean vectors of the UBM and the

adapted speaker model. The speaker's model is likely to represent the impostor's voice more accurately. On the other hand, the proposed clustering approach reduces the distance between the speaker's model, and its respective cluster's background model. It is clear that this clustering offers a more accurate representation of speakers with similar voice characteristics. So, the cluster-dependent background models have a higher chance to detect the impostors that cannot be detected by the UBM.

3.7. Polynomial Regression Based VAD

The literature review chapter summarized the general methods to achieve robust SR systems. As a reminder, the methods can be roughly separated as feature domain methods, model domain methods, VADs, and speech enhancement methods. In this thesis, a novel VAD algorithm is proposed for robustness. Although the feature type used in this work is the MFCC, the VAD can be combined with any other short-term feature. Also, speech enhancement is implemented via spectral subtraction method.

Usually, VADs use parameters such as harmonicity, periodicity, long-term variability, or energy to detect voice activity. Based on these parameters, a frame-based speech/non-speech decision is made. It should be noted that some VAD methods assign the decisions to a few frames, but frame-based decisions are more common.

The aforementioned parameters may be sufficient for the voiced phonemes. The voiced phonemes have a relatively high energy, and harmonic structure. On the other hand, the unvoiced phonemes do not possess harmonic content, and have less energy value. Therefore, the unvoiced phonemes may be treated as a noise signal, and misclassified by the VADs.

The proposed VAD's main decision parameter is the filter-bank magnitude, which can be related to the filter-bank energy. The problem with the energy based VADs is that a high energy noise signal can be falsely detected as a speech signal.

Also, a low energy speech signal can be misclassified as a noise signal. To overcome this situation, a PR is used to group frames together, and each group is represented by its average energy.

The regression is applied to the filter-bank outputs of the Mel scale spectrum. Hence, it can be integrated to the conventional MFCC extraction scheme seamlessly. The frame range to be grouped is chosen as a minimum of five, and a maximum of ten frames. For a 25 ms frame length with a 10 ms overlap, this range is equal to 65-115 ms, which covers the average duration of a vowel-like (i.e. vowel, semivowel, and diphthong) regions in a continuous speech (Prasanna and Pradhan, 2011).

The temporal contour of the speech energy of a filter-bank output approximately resembles a bell-like shape in the spectrum, which is another motivation behind the proposed VAD. In general, speech energy starts rising with the beginning of the utterance, and falls to the noise/silence level at the end of the utterance. With a fixed energy threshold, only the regions that exceeds the threshold level are accepted as speech. However, the frames that are near to the beginning-ending edges may also contain speech information. The PR is used to connect these frames with the nearest high energy frames, based on a minimum error criterion. This edge phenomenon was also utilized in (Qi Li et al., 2002). In that study, the rising and falling energy levels, named as beginning and ending edges, respectively, used for end point detection.

The proposed VAD algorithm can also be thought as an energy boost for the low energy frames in a speech region. The low energy frames, especially the unvoiced phonemes, are likely to be suppressed in the low SNR values. When a group of frames are represented by their average energy, it indicates that the lower energy frames are supported by the higher energy frames.

Another advantage of the proposed VAD is that avoiding misclassification of sudden energy ripples. The ripples occur when a noise frame with a higher energy than its neighbor noise frames, or a speech frame with a lower energy than

its neighbor speech frames. In a traditional fixed energy based VAD, these ripples may be misclassified as a speech frame, and a noise frame, respectively.

As an important note on the working principle of the proposed VAD, it is not expected from a single polynomial to capture the entire bell-like shape of a speech segment. It may not be possible due to the frame range limitations (minimum of five, and maximum of ten frames). Instead, the expectation is to capture at least the rising edge, the falling edge, and (if exists) the steadier peak regions.

General PR equations with least squares sense are given as a reminder in the following. A *k*-th order polynomial is defined as $a_0 + a_1 x_i + \cdots + a_k x_i^k$, where x_i is the intermediate variable, i is the frame index, and 'a's are the coefficients of the polynomial. The summed difference (error, E) between the observed value and the estimated value is minimized in the least squares method. The error can be defined as in Equation (3.22),

$$E = \sum_{i=1}^{n} [y_i - (a_0 + a_1 x_i + \cdots + a_k x_i^k)]^2 \tag{3.22}$$

where i is the frame index, y_i is the observation vector (filter-bank magnitude vector for a given band), n is the length of the vector (number of frames considered for regression). It should be noted that for the proposed VAD, $x_i = i$, where i can take values between 1 and 10. By taking the partial derivative of the error, E, with respect to a coefficient, the optimum value of the coefficient can be found (Equation (3.23)).

$$\frac{\partial E}{\partial a_0} = -2\sum_{i=1}^{n}\left[y_i - \left(a_0 + a_1 x_i + \cdots + a_k x_i^k\right)\right] = 0$$
$$\vdots \qquad (3.23)$$
$$\frac{\partial E}{\partial a_k} = -2\sum_{i=1}^{n}\left[y_i - \left(a_0 + a_1 x_i + \cdots + a_k x_i^k\right)\right] x_i^k = 0$$

Taking the terms with y to one side, a matrix form representation can be possible as in Equation (3.24).

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \cdots & \sum_{i=1}^{n} x_i^k \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \cdots & \sum_{i=1}^{n} x_i^{k+1} \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{i=1}^{n} x_i^k & \sum_{i=1}^{n} x_i^{k+1} & \cdots & \sum_{i=1}^{n} x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \\ \vdots \\ \sum_{i=1}^{n} x_i^k y_i \end{bmatrix} \qquad (3.24)$$

This equation can be written in a compact format as $X^T X\vec{a} = X^T\vec{y}$, where X is defined as in Equation (3.25), $\vec{a}$ is the coefficients vector, and $\vec{y}$ is the observation vector. The coefficients of the polynomial then can be found as $\vec{a} = (X^T X)^{-1} X^T\vec{y}$.

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^k \\ 1 & x_2 & x_2^2 & \ldots & x_2^k \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^k \end{bmatrix} \qquad (3.25)$$

The order of the polynomial, *k*, is selected as two in the experiments. A first order polynomial is a straight line and cannot accurately capture the variations of the speech signals. As the order of the polynomial increases, the error decreases, but the computational load also increases. In the SV experiments, the order is

chosen as two, since an approximation of the general trend over the frames is sufficient. Nevertheless, preliminary experiments showed that no substantial advantage was found by using a third-degree polynomial.

Smoothed Mel spectrum filter-bank magnitudes are directly subjected to the regression. The regression is applied in each band of the Mel spectrum separately. In fact, the conventional spectrum can be used, but the increased number of frequency bins will affect the computation time. Moreover, the Mel spectrum is a part of the traditional MFCC extraction process. The proposed VAD can be integrated to the MFCC extraction module.

Consider a noisy speech signal at frame *t* and filter *m*. Let ***S(t,m)*** denote its filter-bank output. The filter-bank outputs are smoothed to reduce the ripples as in Equation (3.26),

$$S^s(t,m) = \sum_{n=-2}^{2} p_n S(t+n,m) \tag{3.26}$$

where $S^s(t,m)$ is the smoothed filter-bank output, p_n is the smoothing coefficient with $p_{-2} = p_2 = 0.1$, $p_{-1} = p_1 = 0.2$, and $p_0 = 0.4$.

The second order PR is applied to the smoothed filter-bank outputs in each filter band independently. Equation (3.27) gives the regression error, which is defined as the normalized distance between the smoothed filter-bank outputs and the fitted polynomial.

$$e_{N,m} = \frac{\sqrt{\sum_{i=0}^{N-1}(S^s(t+i,m) - F_N(t+i,m))^2}}{N} \tag{3.27}$$

Where $F_N(t,m)$ is the value of the fitted polynomial at frame *t* and filter *m*. *N* is the number of the frames used for regression (*N=5,6,…,10*), and $e_{N,m}$ is the

error observed for *N*-length fitting. The adjacent frames which gave the minimum $e_{N,m}$ are grouped, and groups are represented by their respective average magnitudes. Therefore, if a noise frame with a higher magnitude than its neighbor noise frames is present, the average magnitude representation prevents its misclassification. Similarly, if a speech frame with a lower magnitude than its neighbor frames exists, the neighbors' magnitudes boost this speech frame. Hence, the errors that may occur with a fixed threshold and a frame based decision is tackled with the proposed average magnitude representation.

Once *N* frames are grouped, the same process starts from the next ungrouped frame, and continues until all frames have been grouped and represented by its respective average polynomial magnitude.

For a better understanding, consider the first 25 frames of an utterance. F is calculated from the 1st frame to 5th, then 1st to 6th, and goes on this way up to 1st to 10th. If the minimum error is assumed to be found in the range of 7 frames, the frames 1, 2…,7 are grouped. Average magnitudes of these frames are calculated to represent them. Next, *F* is calculated from the 8th frame to 12th, then from 8th to 13th, and so on.

Once all the frames are grouped, the number of data is at most *TF/5*, where *TF* is the total number of the frames. Note that the proposed method also reduces the data that are going to be used in the clustering. If the utterance durations are not so long, such as less then several minutes, simple clustering algorithms are expected to give sufficient results. This is due to the facts that only two clusters are needed, and the relatively low amount of data to be clustered.

The clustering algorithm is therefore chosen as the conventional k-means. This selection is inspired from the success of the bi-Gaussian modeling approach (Sahidullah and Saha, 2012b). If the weights and the variances of Gaussian mixture models are assumed to be equal, it can be simplified into the difference of the given data and the cluster means, which is also calculated in the k-means algorithm.

Therefore, the k-means algorithm can be viewed as a simplified version of the GMMs.

The initial cluster centroids are chosen randomly and updated iteratively. The k-means algorithm results in two magnitude levels as the class centroids. The assumption is that the frames who belong to the higher magnitude class speech-dominant frames. On the other hand, speech regions with a low magnitude contribute to the lower magnitude class. Therefore, the lower magnitude centroid is selected as the fixed threshold of the given filter band. Only the frames below this threshold are treated as noise-dominant frames. As the overall SNR of an utterance is decreased (due to a low speech energy, or a high noise energy), it is expected that the cluster centroids will come closer. This situation will aid

The frames below the threshold are also used to estimate the average noise magnitude of the band. Then, the spectral subtraction method used this average noise magnitude to enhance the speech. A magnitude floor is included in the subtraction to avoid the spectrum becoming too small, which may lead to numerical problems in the log compression step. Equation (3.28) shows the subtraction expression,

$$S^e(t,m) = \max(S^s(t,m) - S^n(m), 0.001S^s(t,m)) \quad (3.28)$$

where $S^e(t,m)$ is the enhanced speech signal, and $S^n(m)$ is the estimated noise energy for the m-th filter.

The aforementioned k-means process separates the filter-bank magnitudes into one of the two classes, indicating the reliable and unreliable components, similar to the missing data approaches as mentioned before. A binary representation of the utterance is obtained by using ones for each frame in the groups above the threshold and using zeros for each frame in the groups below the threshold. This binary matrix is used in a voting scheme that aids the final decision on the frames.

For a frame in the binary matrix, if ones dominate the frame, it is likely to be a speech frame. However, as the noise level increases, the number of ones are also expected to increase, which is a result of misclassifying the noise frames as speech frames in the bands. Therefore, a fixed threshold is not suitable for all SNR levels. To tackle this problem, a pseudo SNR-dependent threshold called clarity level is defined, which takes the cluster centroids into account. The operations described above to obtain the binary matrix representation of the frames are summarized as a block diagram in Figure 3.13.

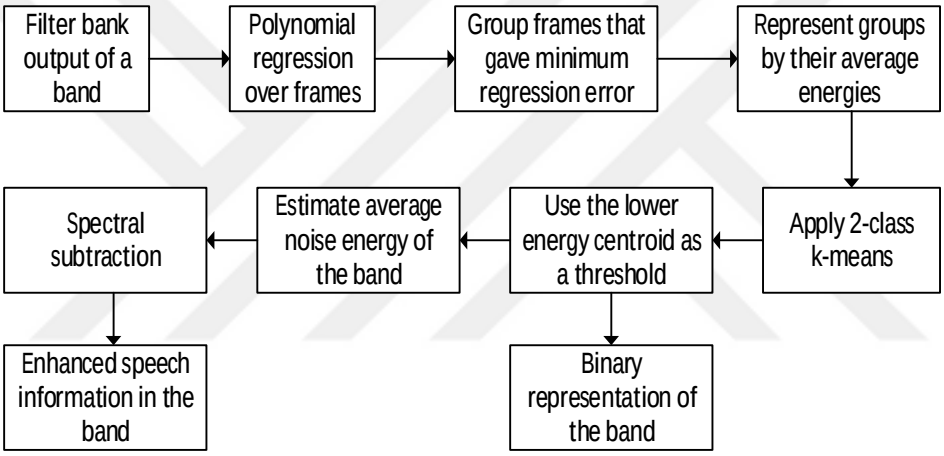

Figure 3.13. Speech enhancement and binary representation of the frames in a given band as a block diagram (Dişken et al., 2017b)

The clarity level uses the k-means cluster centroids to estimate the pseudo SNR. It is based on the fact that as the overall SNR of the utterance decreases, the cluster centroids are expected to become closer. Equation (3.29) shows the formal definition of the clarity level.

$$L = \frac{\sum_{m=1}^{M} log_{10}(\frac{C^{hi}(m)}{C^{low}(m)})}{M} \tag{3.29}$$

where L is the clarity level, $C^{hi}(m)$ is the centroid of the speech-dominant cluster, and $C^{low}(m)$ is the centroid of the noise-dominant cluster of the m-th filter, and M is the total number of filters in the filter-bank. Note that $C^{low}(m)$ is used as the threshold described previously.

As the noise level increases, the L gets smaller values. The main reason for this situation is that speech signal vanishes into the noise signal as it increases. Since the boarders of the signals gets vague as the noise level increases, the clustering algorithm cannot define two distinct clusters. The resulting cluster centroids are expected to become closer.

The pseudo SNR term is preferred to emphasize that the goal is not to estimate the actual SNR value. Rather, the clarity level gives hint about how noisy the signal is. The relation between the SNR and clarity level calculations are obvious. The formal expression for the SNR is given in Equation (3.30), where S_{rms} is the root-mean square of the speech signal, and N_{rms} is the root-mean square of the noise signal.

$$SNR\ (dB) = 20\log(\frac{S_{\text{rms}}}{N_{\text{rms}}}) \tag{3.30}$$

The clarity level expression used cluster centroids instead of the root-mean square values. The SNR is based on the speech and the noise energy ratio. A similar relation between the cluster centroids can be found in Equation (3.29). Therefore, the clarity level can be defined as a pseudo SNR estimation.

For the final output decision of the proposed VAD, the binary matrix and the clarity level are combined in a voting scheme. As a consequence of the clarity level's definition, the signal is treated as a high SNR signal if its clarity level is high, and, vice versa. In the binary matrix, the ones indicate the speech activity, and zeros indicate the noise presence. To label a frame as a speech-dominant frame, the number of ones must be more than the number of zeros. If a signal's

clarity level is high, one can assume that the signal is relatively clean. Therefore, speech activity (ones) in a few bands may be sufficient to confidently say the frame is a speech-dominant frame.

On the other hand, if a signal's clarity level is low, then one must expect that some of the noise components may be misclustered as speech signals. Then, more evidence is needed to label a frame as the speech. It means that the number of ones in a given frame must be increased, compared to a signal with a high clarity level value.

A final decision threshold, called sufficient speech evidence, is developed according to the relations mentioned in the previous paragraph. The proposed threshold is a linear line between the best and worst SNR cases. The calculation of the threshold is given in Equation (3.31),

$$L^s = \begin{cases} 7, & L > 0.8 \\ round(28.36 - 25.45 * L), & 0.8 \geq L \geq 0.25 \\ 23, & L < 0.25 \end{cases} \quad (3.31)$$

where L^s is the sufficient speech evidence, i.e. minimum number of ones required for a frame to be labeled as speech-dominant. Note that equations (3.29) and (3.31) calculated per utterance. Therefore, the proposed VAD eliminates the need for priori information about the noise type, or the actual SNR estimate.

The sufficient limit for the best-case scenario is chosen as seven. If the clarity level exceeds 0.8, it indicates that the signal of interest is a relatively clean signal (SNR > 15 dB). So, if the number of ones for a frame of the binary matrix is equal to, or greater than, seven, the frames are labeled as speech-dominant, and it can be subjected to the feature extraction process. Frames whose summation are less than seven are ignored, since they do not convey any useful speech information.

The best-case limit, seven, is determined by assuming that the speech signal should cover at least several bands. Nevertheless, the best-case limit is not as critical as the worst-case limit, as found with preliminary experiments. Similar SV results were obtained by choosing 6, 7, and 8 as the best-case limit values.

The worst-case limit ($L<0.25$) indicates a severely degraded signal, hence more evidence is needed to label a frame as speech-dominant. 18 and 23 were investigated as the worst-case limits. It is found that when the worst-case limit is 23, a 10% absolute EER reduction was obtained, compared to 18. Therefore, 23 is preferred as the worst-case limit, and 7 is preferred as the best-case limit. The threshold between these limits is simply the linear line passing through these points. Also, it should be noted that the worst-case limits were not reached for the SNR levels used in the SV experiments. The worst-case limit rather adjusts the slope of the threshold line.

Another important point is that the L values are determined on the sample signal, taken from the NOIZEUS corpora. This sample speech signal is not included in the verification experiments. Hence, the parameters are not tuned for a specific database. However, a preliminary test is suggested to verify that these values are suitable for the data type of interest.

The second part of the VAD algorithm, which consists of the calculations for the clarity level (L), sufficient speech evidence (L^S), and final output decision, is summarized as a block diagram in Figure 3.14.

The proposed VAD algorithm is tested on the sample signal to illustrate its effectiveness. Lynx noise is added to the sample signal taken from the NOIZEUS corpora. The overall SNR is 5 dB in this case. Figure 3.15 illustrates the Mel spectrum of both clean speech, and noisy speech signals. Comparing the spectra, the effect of the noise can be easily observed. The speech signals with relatively high magnitudes (red regions) remain in both spectra. However, the lower magnitudes are vanished into the noise.

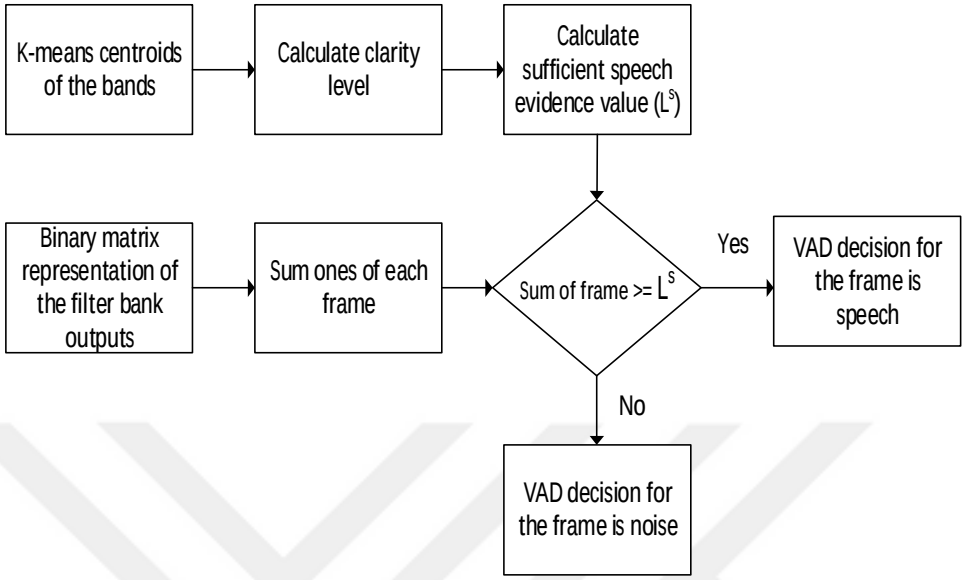

Figure 3.14. Final decision process of the proposed VAD algorithm as a block diagram (Dişken et al., 2017b)

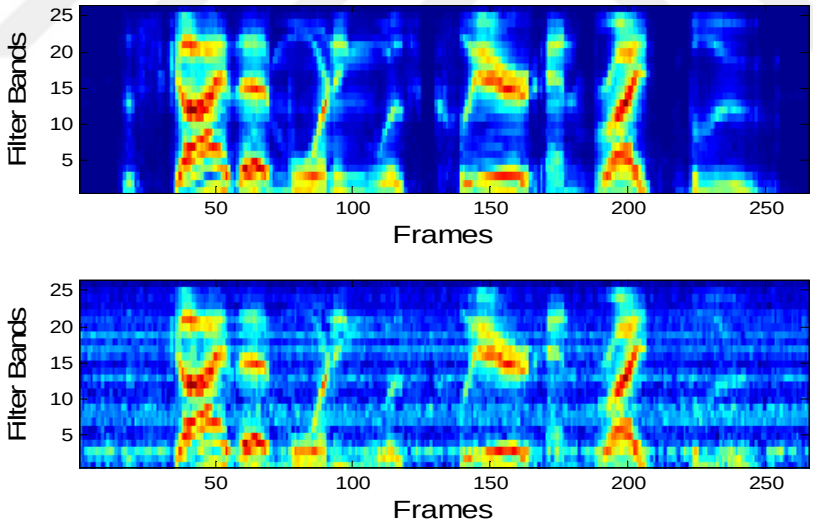

Figure 3.15. Mel spectrum of the cleans sample signal (top), and the degraded signal with a 5 dB overall SNR (bottom)

The results of the PR are given in Figure 3.16, along with the cluster centroids found by the k-means algorithm. The blue lines are the smoothed filter-bank magnitudes. The red line shows the average magnitudes of the frames in the same group, as observed from the horizontal parts. The straight solid line is the centroid of high magnitude cluster. The dashed line is the centroid of the low magnitude cluster, which is also used as the threshold level for this band. As observed from the figure, the average magnitude representation closely follows the original filter-bank magnitudes.

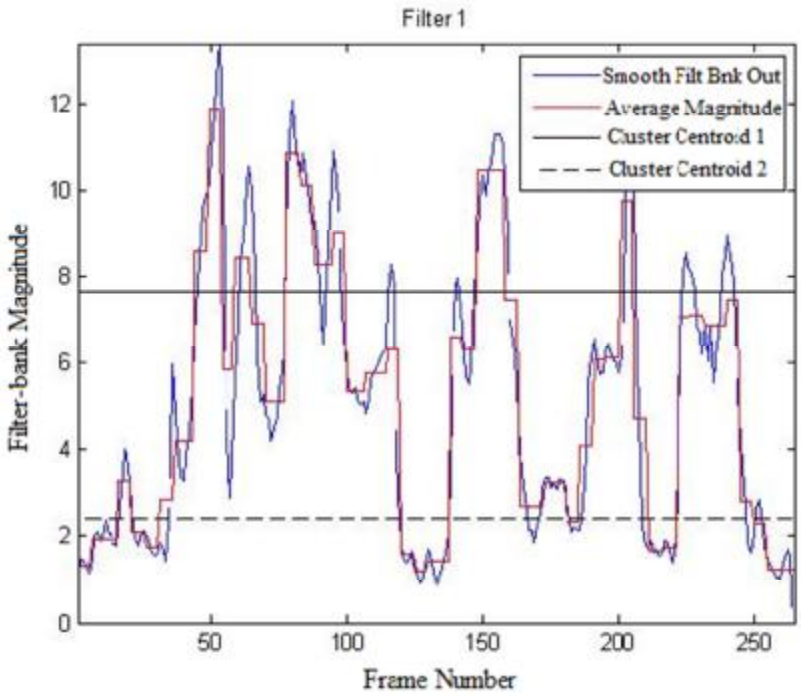

Figure 3.16. Polynomial regression and clustering results for the 1st filter of the filter-bank

Figure 3.17 illustrates the same analysis for the 8th filter of the filter-bank. A small region with a relatively very high magnitude stands out, and considered as a class by itself. This region implies that the proposed VAD algorithm is not

suitable for non-stationary noises. In fact, dealing with the non-stationary noises is a very challenging problem that is yet to be solved.

On the other hand, this figure supports the idea of using the lower magnitude cluster centroid as the threshold. If the frames were directly clustered as the noise frames and speech frames, then only a few frames should be detected as speech in this band, which will be an obvious error. By using the centroid as a threshold level, this error is admirably avoided. Another solution may be adding a minimum number of membership to the clusters. This solution however, may affect the clarity level calculation since it will change the cluster centroid levels.

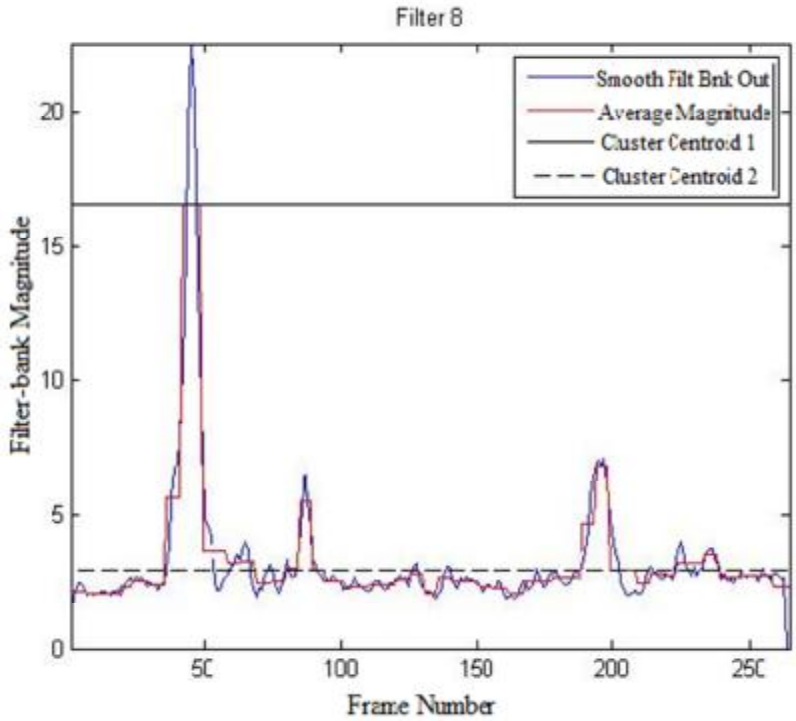

Figure 3.17. Polynomial regression and clustering results for the 8th filter of the filter-bank

Figure 3.18 and Figure 3.19 illustrates the results for the 16th and 24th filters, respectively. As discussed before, the cluster centroids come closer as the

speech energy decreases. This fact can be clearly observed by comparing the distance between centroids of the 24th filter to the others'. The 24th filter covers the high frequencies, where speech energy is lesser than the lower frequencies. The distance between the centroids are therefore decreased. As the overall SNR decreases for an utterance, the expectation is that the centroids of the other filters also get closer. Hence, the clarity level parameter can estimate how noisy the utterance is, as explained above.

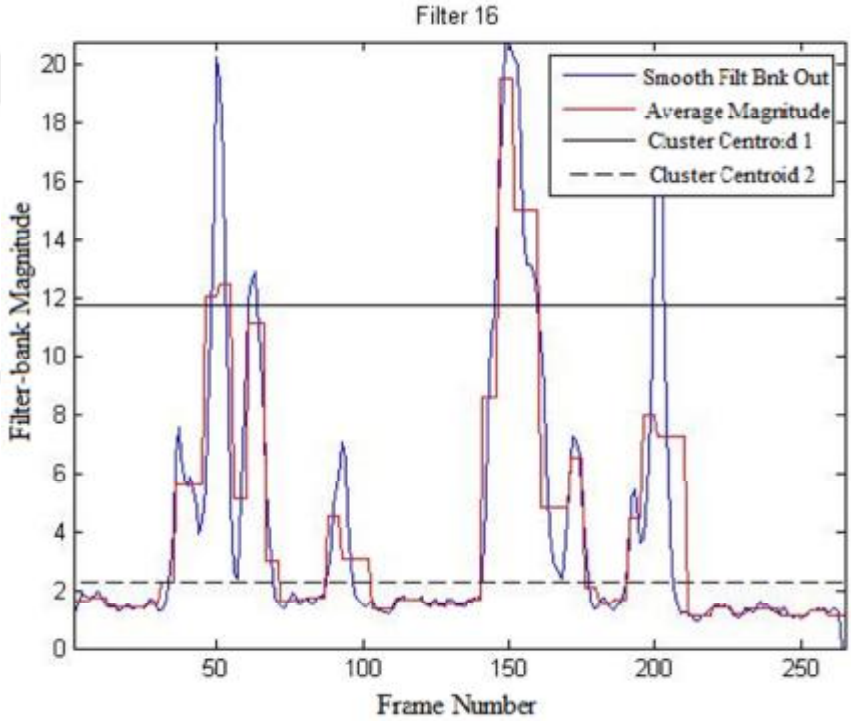

Figure 3.18. Polynomial regression and clustering results for the 16th filter of the filter-bank

The frames under the threshold (dashed lines) are shown with zeros, and the others are shown with ones. The binary matrix is build this way. Average noise magnitudes are calculated by considering only the frames that are below the threshold. Spectral subtraction is then used to enhance the speech. Finally, clarity

level and sufficient speech evidence values are calculated for the final VAD output decision.

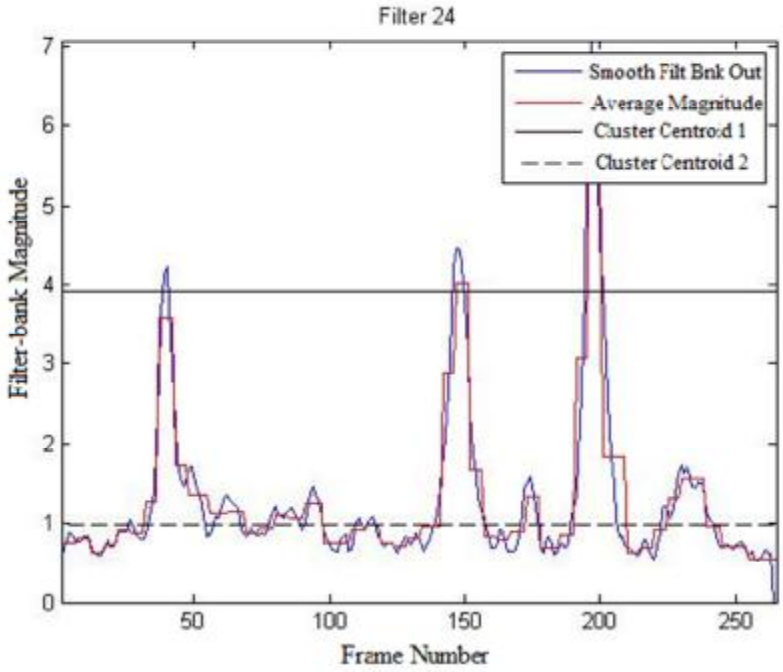

Figure 3.19. Polynomial regression and clustering results for the 24th filter of the filter-bank

Spectra of the noisy utterance, enhanced utterance, and the frames detected as speech by the VAD are given together in Figure 3.20, for the 5 dB overall SNR. The noise artifacts are clearly seen at the top spectrum. The result of spectral subtraction is given in the middle. The noise effects are almost completely removed in this case. The final VAD outputs are shown at the bottom of the figure. It should be noted that only the frames selected as speech are shown, and only these frames are used in the further processes.

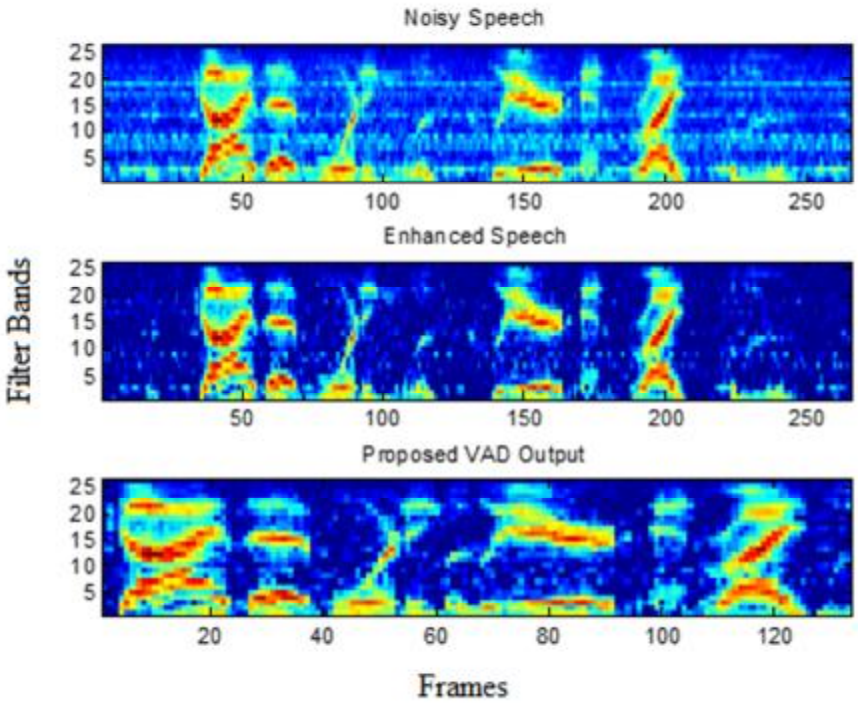

Figure 3.20. Noisy speech signal with a 5 dB overall SNR (top), enhanced speech signal (middle), and final output of the proposed VAD (bottom)

Same spectra are illustrated in Figure 3.21 for the 0 dB overall SNR. In this case, the noise effects are visible in the enhanced speech. However, the final VAD output still successfully captures the high magnitude speech regions. Note that the number of frames at the output is around 100 in this case.

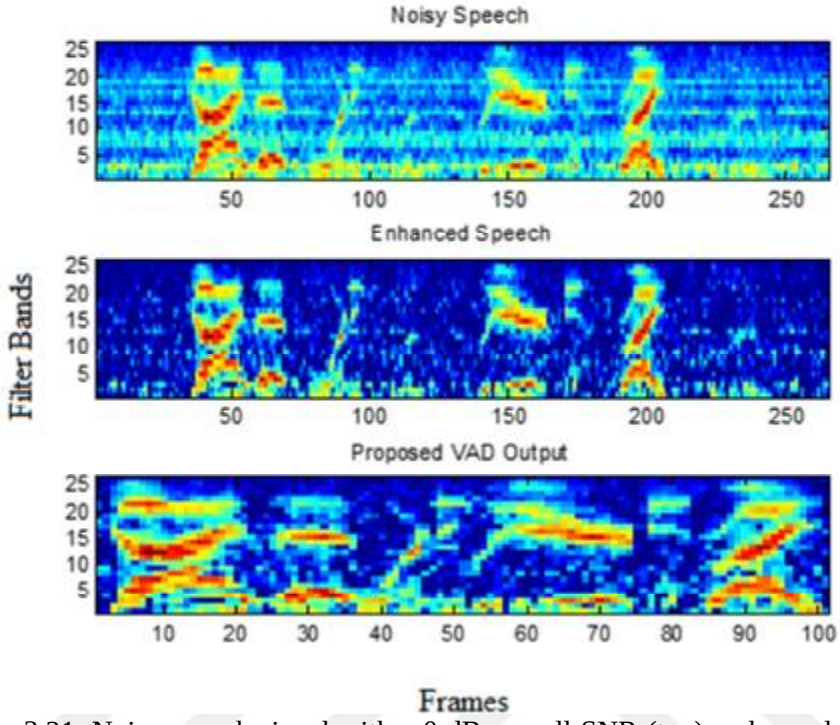

Figure 3.21. Noisy speech signal with a 0 dB overall SNR (top), enhanced speech signal (middle), and final output of the proposed VAD (bottom)

Finally, Figure 3.22 shows the spectra for the -5 dB overall SNR. Note that despite the increased noise artifacts, the proposed VAD algorithm determines the speech region thanks to the sufficient speech evidence thresholding.

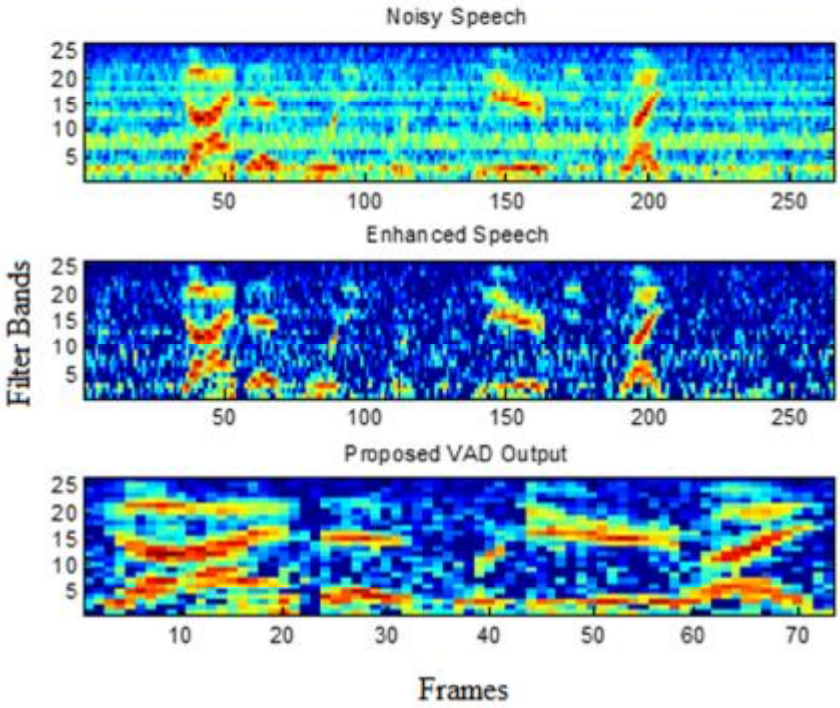

Figure 3.22. Noisy speech signal with a -5 dB overall SNR (top), enhanced speech signal (middle), and final output of the proposed VAD (bottom)

These figures prove that the proposed VAD can effectively calculate a threshold that is suitable for the SNR of the signal. The high energy regions are extracted successfully in each case.

3.8. Methods Selected for Performance Comparison

Two methods from the literature are selected to compare the performance of the proposed VAD algorithm. One of these methods is a NT algorithm proposed in (Rangachari and Loizou, 2006), and the other is a ANN based VAD proposed in (Drugman et al., 2016).

The NT algorithm (called Rangachari's method from here on), is chosen due to its continuous noise spectrum update property, which can be beneficial for non-stationary environments, although only stationary noise types are considered in

this thesis. The speech enhancement methods' success depends on the accuracy of the noise estimation. Rangachari's method offers a fast adapting NT, which considers the speech presence probability in each frequency bin. Compared with several other noise estimation techniques, including the minimum statistics, better estimation results were reported. Formal listening tests were also included, where the listeners preferred Rangachari's method over the other algorithms (Rangachari and Loizou, 2006).

To make a fair comparison, Rangachari's method is modified to operate on the Mel scale, similar to the proposed regression based VAD. Since the noise estimation algorithm does not label the frames as VADs do, a fixed threshold is applied once the speech is enhanced via spectral subtraction. The intention here is that if the true noise spectrum can be estimated and subtracted from the noisy speech signal, only the clean speech signal lefts. Then, a simple energy based threshold can be applied to eliminate silence frames. Frames that have an energy higher than the average energy of all enhanced frames are accepted as speech frames. Only these accepted frames are subjected to the feature extraction process. Rangachari's method is independent from the noise type and does not require any training, similar to the proposed VAD. Rangachari's method is implemented in MATLAB, using the parameters given in the related publication.

The other method is a recently proposed VAD (called Drugman's method from here on) (Drugman et al., 2016). An ANN with a single layer of 32 neurons is used to obtain posterior speech probabilities of the frames. Besides the MFCCs, four voicing measures, and two pitch trackers are used as the features. The ANN is trained on 1500 speech files degraded by noises chosen from the NOISEX-92 database. The author kindly shares his MATLAB codes in his website (http://tcts.fpms.ac.be/~drugman/Toolbox/ Accessed 8 January 2018). Since the same noise database is used in this thesis work, and due to the lack of another speech/speaker database to train an ANN, the author's trained ANN is used in the SV experiments. Once the posterior speech probabilities are obtained, a threshold

is applied to eliminate non-speech frames. Also, to make a fair comparison, spectral subtraction is also included, where the average noise energy is calculated by considering the non-speech frames.

The reasons for choosing Drugman's method for comparison are a) it is one of the recently published VADs, b) it is reported that it outperformed state-of-the-art VADs including (Ghosh et al., 2011; Jongseo Sohn et al., 1999). However, a disadvantage of Drugman's method may be the requirement of the ANN training. Nevertheless, a general problem with the supervised methods is that their performances are expected to drop in an unseen environment.

The mathematical details of these algorithms are not given here, since no significant modifications were made. They are implemented with the parameters given in their respective publications, hence interested readers may refer them.

3.9. Real-Time Text-Dependent Speaker Verification

A real-time SV application is written in C++ programming language, and realized on a single-board computer as a case study. Banana-Pi (BPI-M1 model) single-board computer is used, which has an ARM A7 dual-core processor, 1 GB SDRAM, GPIO pins, audio output, video input/output, USB ports, and many other specifications. The single-board computers can be used as a light-weight, low cost computer with a suitable operating system. Furthermore, they can be configured as a conventional microcontroller via the GPIO pins.

The purpose of this real-time application is to develop a baseline system which can be improved in the future works. The system can be modified for hands-free applications, security systems, etc. However, it is limited to text-dependent SV, and is a part of a project supported by the scientific research projects coordination unit of Adana Science and Technology University, with the project number 17103031. In the project, the goal is to develop a speaker-independent isolated word recognition to control an elevator via speech commands. As this thesis focuses on the SV, the real-time system is modified to recognize the speaker.

The MFCC features are also used in the real-time system, due to its ease of implementation, and high performances in the controlled environments. Since the vocabulary of the system is limited to a few words, each word is modelled by a GMM, whose details were given previously. When the system starts recording, the initial 200 milliseconds are used as a silence level threshold, and a energy based VAD is implemented, which considers the energies of ten adjacent frames. If the frames' energies are over the threshold, the system stores the recorded data until another ten adjacent frames' energies stay below the threshold. Then, the stored sound is subjected to feature extraction.

Figure 3.23 shows the equipment used to build the recognition system. A Linux based operation system, UBUNTU MATE, is used to execute the necessary C++ language commands. Note that the Banana Pi is connected to the monitor, mouse, and keyboard. These connections are not necessary once the recognition program is developed. It can be configured as a start-up application, hence whenever the board is opened, the recognition can start automatically. Also, note that an external microphone is used, since Banana Pi does not have an internal microphone. The microphone is connected via the USB port. Advanced Linux Sound Architecture (ALSA) is used to configure the recording parameters.

The vocabulary of the system consists of the digits, and four commands in Turkish language. The vocabulary is chosen as suitable words to control an elevator. Therefore, the command words are yes, no, close, and open in Turkish language. Twenty-four volunteer individuals uttered these words, ten times for each word. Seven of these utterances are used to train word models. The remaining three words are used to test the system. A total of 1008 test speech files are used.

For the SV, all speakers' training data are used to construct an UBM. The target speaker's (the author of this thesis) data is used to train the speaker model. In the test stage, the unknown speaker's utterance is scored with both models. If the speaker's model gives a higher score, the unknown speaker is verified, else, rejected.

Figure 3.23. Real-time speaker recognition system implemented on a single-board computer

4. EXPERIMENTAL RESULTS

In this chapter, the SV results for the proposed algorithms are given. First, the speaker clustering method's performance is investigated. The results of the robust SV experiments are given after. Finally, the real-time SV system's performance is analyzed. The experimental setups are described in each subsection for completeness, although most of the parameters are same, such as MFCCs.

4.1. Speaker Verification with Speaker Model Clustering

The SMC method was tested with the male speakers of the NIST SRE 1998 database. It was reported that the performance difference between the male and female speakers is fairly small for this database (Doddington et al., 2000). Hence, female speakers were ignored in this case. All available training data for the male speakers were pooled to train a UBM, which consists of 1024 Gaussians. Then, the speaker models were derived by adaptation form the UBM with a relevance factor of 16. Only the mean vectors were adapted, as in the conventional approach.

All the test data durations were considered, i.e. 3, 10, and 30 seconds. As described before, two microphone types are available in this dataset, namely, electret, and carbon-button. Therefore, a same-handset condition implies that the training and testing records for a speaker were both collected by using the same microphone type. A different-handset condition indicates that the microphone types were changed between the training and testing records. As an example, if the training data was recorded with an electret type, then the testing data is recorded with a carbon-button type.

For each test utterance duration, there were 1308 speech files for the same-handset condition, and 1192 speech file for the different-handset condition. For each of these test files, there was one trial for the target speaker, and nine trials for

the non-target speakers. Hence, the total number of trials in each utterance duration was 13080 for the same-handset, and 11920 for the different-handset conditions.

The HTK toolkit was used to extract MFCCs from the utterances. A Hamming window with a 25 ms length, and a 10 ms shift was utilized. The filter-bank consists of 26 triangular bandpass filters. Twelve static coefficients, excluding the zeroth coefficient, were extracted, and the normalized log-energy was appended to them. The cepstral mean subtraction was also used to suppress the slowly varying signals, which are mainly related to the channel effects. The final feature vector dimension was 26 with delta coefficients. The other operations such as training a UBM model, speaker model adaptation, model clustering, scoring, etc. were implemented using the C++ programming language.

In the scoring phase, a test file was first scored with the UBM. Top scoring 5 mixtures were detected for each feature vector. Then, the same mixtures of the speaker models were scored using the respective feature vector. The baseline results obtained by the conventional GMM-UBM method is given in the third row of Table 4.1. The EER values increased as the utterance duration decreased. Also, the negative effect of the channel mismatch is obvious.

The validity of the proposed clustering algorithm was tested with 2, 3, 4, and 5 clusters. The speaker models adapted from the baseline UBM were used in the clustering. As seen in the table, all the impostor models obtained by the clustering showed improved recognition performance compared to the baseline system.

Table 4.1. EER(%) values obtained for the conventional GMM-UBM method, and the proposed clustering method

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
UBM	11.21	5.3	3.58	25.57	20.31	16.12
2-cluster	**10.23**	4.89	3.22	25.38	19.99	16.09
3-cluster	10.54	**4.73**	**3.18**	24.15	**18.53**	**14**
4-cluster	10.38	4.96	3.29	24.05	18.75	14.16
5-cluster	11	4.92	3.35	**23.98**	19.13	14.29

The relative EER reductions are shown in Table 4.2, where the best improvements were obtained by 3-cluster, except the 3-seconds cases. The average EER reductions are calculated as 4.85%, 9.22%, 7.92%, and 6.44% for 2-, 3-, 4-, and 5-cluster, respectively.

Table 4.2. Relative EER reductions compared to the baseline UBM method.

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
2-cluster	**8.71%**	7.69%	10.19%	0.76%	1.56%	0.17%
3-cluster	5.98%	**10.73%**	**11.13%**	5.57%	**8.76%**	**13.17%**
4-cluster	7.34%	6.41%	8.05%	5.93%	7.66%	12.13%
5-cluster	1.89%	7.05%	6.39%	**6.23%**	5.78%	11.32%

The Detection Error Tradeoff (DET) curves are given for the GMM-UBM system, and the best performing clusters of the proposed method in Figure 4.1 for the same-handset condition, and Figure 4.2 for the different-handset condition. Note that the other clusters' curves are not shown to avoid confusing illustrations since the curves highly interfere with each other. As observed from the figures, the impostor models obtained by the SMC highly reduced the false negatives for the different-handset condition.

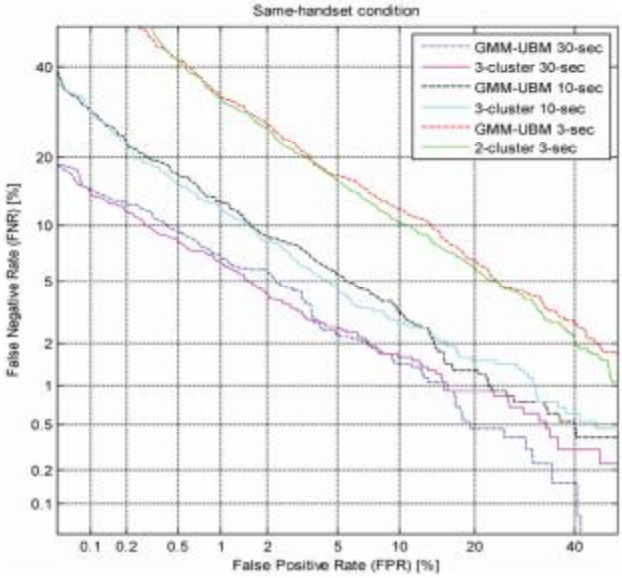

Figure 4.1. DET curves of the GMM-UBM and the best performing clusters for the same-handset condition (Dişken et al., 2017a)

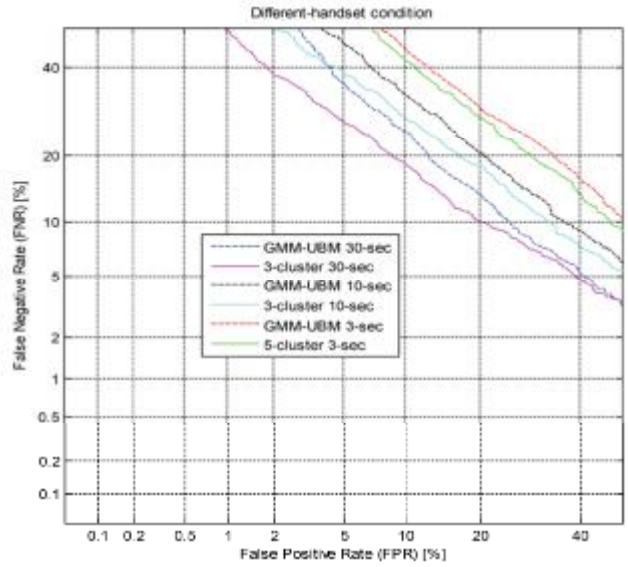

Figure 4.2. DET curves of the GMM-UBM and the best performing clusters for the different-handset condition (Dişken et al., 2017a)

Re-adapting the speaker models from their respective impostor models were also investigated. This type of adaptation can be considered as subsets of the baseline GMM-UBM method. The results of this approach are shown in Table 4.3. The relative improvements over the baseline method are given in Table 4.4. Similar to the previous case, 3-cluster gave the best overall performance improvement with an average of 7.24% EER reduction. The other average reductions are 5.02%, 6.9%, and 6.15%, for 2-, 4-, and 5-cluster, respectively. Although improvements over the baseline can be observed, this method is less effective than the UBM adapted speaker models under channel mismatch conditions.

Table 4.3. EER(%) values for the speaker models adapted from their respective impostor models

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
2-cluster	10.38	4.85	**3.12**	25.32	20.12	16.07
3-cluster	**10.22**	4.85	3.20	**24.26**	19.45	15.25
4-cluster	10.32	4.85	3.16	24.41	19.70	15.11
5-cluster	10.61	4.85	3.27	24.81	**19.33**	**14.92**

Table 4.4. Relative EER reductions for the re-adapted models compared to the baseline UBM method.

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
2-cluster	7.34%	7.84%	**12.79%**	0.98%	0.91%	0.29%
3-cluster	**10.22%**	7.84%	10.66%	**5.13%**	4.21%	5.37%
4-cluster	7.95%	7.84%	11.85%	4.54%	2.98%	6.24%
5-cluster	5.30%	7.84%	8.53%	2.98%	**4.81%**	**7.45%**

The GMM-UBM approach is recently outperformed by the i-vector approach. To compare the proposed speaker clustering method with the i-vectors, MSR Identity toolkit is utilized in the MATLAB. The baseline UBM and pooled

training data were used to train the i-vector extractor in twenty iterations. 100 dimensional i-vectors were extracted from each utterance. LDA was used to reduce the channel mismatch effects, and probabilistic LDA was employed for scoring the i-vectors.

As the i-vector approach includes the LDA for channel compensation, handset normalization (Reynolds, 1997) was added to the proposed method in order to make a fair comparison. Handset normalization technique is a score normalization process to reduce the channel mismatch. The handset normalization is achieved by detecting the response of the models to the different channel types. Then, during the testing, the test utterance's channel type is detected, and its score is normalized by the scores obtained for the same channel type. This operation restricts the speaker models to produce zero mean and unit standard deviation scores. Hence, the handset characteristics are reduced.

In Table 4.5, the EERs for the i-vector and the proposed method are given. The results indicated that the impostor models created by the proposed clustering model can achieve state-of-the-art SV results. Only 5-cluster at 30-seconds utterance duration performed worse than the i-vectors. The relative improvements are given in Table 4.6. a relative improvement as high as 23.62% was achieved by using two clusters.

Table 4.5. EER(%) values for i-vectors and the proposed method with handset normalization

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
i-vector	11	4.70	3.02	24.65	18.69	14.42
2-cluster	**8.40**	4.34	2.97	22.64	16.43	13.91
3-cluster	8.81	**4.11**	**2.93**	21.88	**16.09**	13.33
4-cluster	8.86	4.25	3	21.95	16.18	13.05
5-cluster	8.94	4.19	3.20	**21.71**	**16.09**	**12.74**

Table 4.6. Relative EER reductions compared to the i-vector baseline

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
2-cluster	**23.62%**	7.58%	1.68%	8.16%	12.11%	3.55%
3-cluster	19.84%	**12.45%**	**3.08%**	11.22%	**13.9%**	7.61%
4-cluster	19.45%	9.56%	0.56%	10.96%	13.45%	9.49%
5-cluster	18.68%	10.83%	5.89%	**11.9%**	**13.9%**	**11.69%**

On the average, significant performance improvement over the i-vector was obtained with the proposed clustering method. The 5-cluster yielded the best average performance improvement with a 12.14% average relative EER reduction. The average reductions for the 2-, 3-, and 4-cluster were 9.45%, 11.35%, and 10.58%, respectively. Figure 4.3 shows the DET curves for the same-handset condition, and Figure 4.4 shows the DET curves for the different-handset condition.

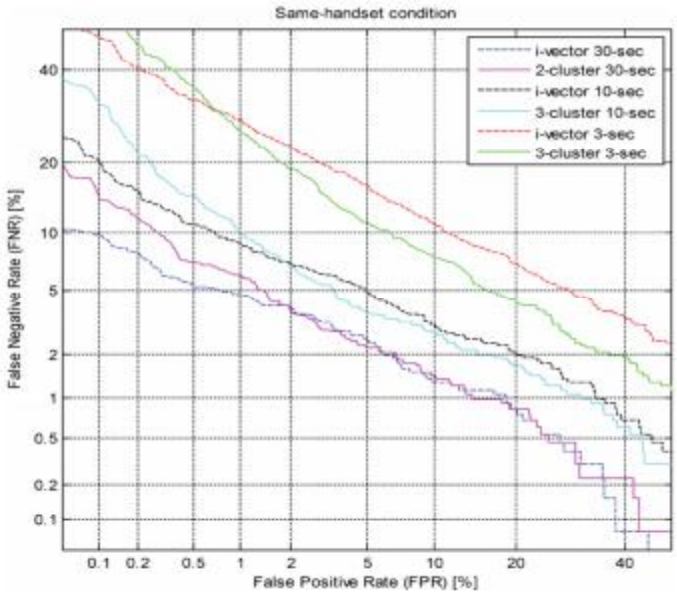

Figure 4.3. DET curves of the baseline i-vector, and the best performing clusters for the same-handset condition (Dişken et al., 2017a)

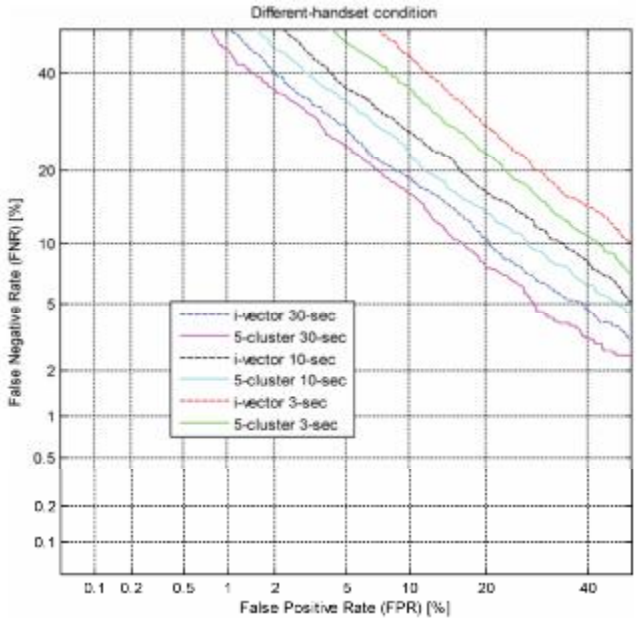

Figure 4.4. DET curves of the baseline i-vector, and the best performing clusters for the same-handset condition (Dişken et al., 2017a)

The performance of the proposed clustering algorithm showed better performances than the conventional GMM-UBM method, and the state-of-the-art i-vector method. The EERs and the DET curves proved that the using several impostor models, instead of only one, can improve the system performance, without highly increasing the computational complexity, or storage requirements.

On the other hand, the EERs were close to each other. Therefore, a statistical significance test is also provided to further support the proposed method's performance. McNemar's test is utilized, which is used in speech recognition area before (Gillick and Cox, 1989; Pallet et al., 1990).

Consider two classifiers named A, and B, which are tasted with the same data, and the following variables are counted.

- N00: Number of examples misclassified by both A, and B
- N01: Number of examples misclassified by A, but not B

- N10: Number of examples misclassified by B, but not A
- N11: Number of examples misclassified by neither A, nor B

The null hypothesis expects that the two algorithms have the same error rate, i.e. N01=N10. McNemar's test is given in Equation (4.1).

$$X^2 = \frac{(|N01 - N10| - 1)^2}{N01 + N10} \tag{4.1}$$

Under the null hypothesis, X^2 has a chi-square distribution with 1 degree of freedom. The value of test at 5% significance level for 1 degree of freedom is 3.84. Hence, if the test is greater than this value, the null hypothesis is rejected, which implies that the classifiers have different performances. In Table 4.7, the proposed clusters were compared with the i-vector, based on the EER values given in Table 4.5. Except the 30-second case, significant performance differences are observed. The highest differences were found in the 3-second cases, which indicates that the proposed method is more suitable for the short utterance durations.

Table 4.7. X^2 values obtained by using the proposed method and the i-vector

	Same-handset coniditon			**Different-handset condition**		
Duration	3-s	10-s	30-s	3-s	10-s	30-s
2-cluster	54.98	7.42	1.5	24.44	17.73	9.07
3-cluster	36.16	14.6	2.18	26.78	25.74	16.2
4-cluster	34.41	10	1.12	25.41	23.9	20.33
5-cluster	31.33	11.84	0.0017	30.82	25.58	28.83

The reason behind the similarity occurred in the 30-second same-handset condition may be due to the better i-vector representations acquired as the utterance duration increases. In general, 3-cluster showed the best verification performance,

based on the results given in the tables. Hence, a straightforward solution to increase the performance of the conventional GMM-UBM method is verified.

4.2. Speaker Verification Under Additive Noise

The proposed PR based VAD's performance was examined with both male and female data of the NIST SRE 1998 corpus. 30-second length test utterances were used. Different-handset condition is ignored in this case, the focus is solely on the additive noise artifacts. There were 1308 test speech files for the male speakers, and 1379 test speech files for the female speakers. A simple energy based VAD as given in (Kinnunen and Li, 2010) was applied to the clean training data to eliminate silence regions. The type of VAD won't make much difference on the verification, since the training data is relatively clean. 26 dimensional MFCCs (13 static excluding the zeroth coefficient, and their deltas) were extracted as feature vectors from each utterance.

The proposed VAD, and the other methods selected for comparison were only applied to the noisy test files. In the test stage, MFCCs were extracted from the frames that are detected as speech-dominant.

For the back-end, both the conventional GMM-UBM method, and the state-of-the-art i-vectors were considered. All methods were implemented in the MATLAB environment. The parameters for the back-end systems were the same as in the previous experiments.

The test data were degraded with Lynx, F16, car, babble, and Stitel noises from the NOISEX-92 noise database. Overall SNR levels were changed from -10 dB to 10 dB with 5 dB steps.

Table 4.8 shows verification results for the male data, with GMM-UBM back-end method, where minDCF values are shown in parenthesis.

Table 4.8. Speaker verification results for the male speakers with UBM method in terms of percent EER (minDCF)

Noise Type	SNR Level (dB)	Proposed Algorithm	Drugman's VAD	Rangachari's Noise Tracking
Lynx	-10	**34.25 (0.64)**	46.10 (0.85)	47.4 (0.87)
	-5	**25.30 (0.47)**	32.18 (0.60)	39.22 (0.72)
	0	15.29 (0.28)	**14.60 (0.27)**	22.47 (0.42)
	5	**8.41 (0.15)**	**8.41 (0.15)**	13.45 (0.24)
	10	**5.42 (0.10)**	6.50 (0.12)	9.93 (0.18)
F16	-10	**41.28 (0.78)**	48.31 (0.88)	48.16 (0.89)
	-5	**31.88 (0.60)**	41.82 (0.80)	45.18 (0.84)
	0	**20.87 (0.39)**	24.38 (0.46)	33.4 (0.60)
	5	11.85 (0.22)	**11.54 (0.21)**	18.19 (0.34)
	10	**6.95 (0.13)**	7.8 (0.14)	12.46 (0.23)
Car	-10	**5.96 (0.10)**	6.27 (0.11)	8.94 (0.16)
	-5	**4.74 (0.08)**	5.88 (0.10)	8.35 (0.15)
	0	**4.35 (0.08)**	5.50 (0.10)	8.18 (0.15)
	5	**4.05 (0.07)**	5.27 (0.09)	7.95 (0.14)
	10	**4.05 (0.07)**	5.12 (0.09)	7.95 (0.14)
Babble	-10	**36.85 (0.69)**	48.08 (0.87)	47.85 (0.88)
	-5	**26.83 (0.50)**	38.45 (0.72)	43.94 (0.87)
	0	**17.50 (0.33)**	19.49 (0.36)	28.28 (0.51)
	5	**10.01 (0.18)**	10.16 (0.18)	14.52 (0.27)
	10	**6.72 (0.12)**	7.26 (0.13)	10.93 (0.20)
Stitel	-10	**42.66 (0.79)**	47.17 (0.86)	45.18 (0.84)
	-5	**33.71 (0.62)**	37.23 (0.69)	37.15 (0.69)
	0	19.95 (0.37)	**19.26 (0.36)**	20.41 (0.38)
	5	**9.40 (0.17)**	9.71 (0.18)	11.62 (0.21)
	10	**5.96 (0.11)**	6.95 (0.12)	9.32 (0.17)

The proposed VAD showed superior performances compared to the baseline methods, especially at the lower SNR levels. Except four cases, the proposed method produced lower errors than the others. Also, the performance is not dependent on the noise type, or noise level.*Table 4.9* Table *4.9* shows the relative EER reductions obtained by the proposed VAD, compared to the baseline methods.

Table 4.9. Relative percent EER reductions for the male speakers with UBM back-end. A: Proposed VAD. B: Drugmans's method. C: Rangachari's method

Noise Type	SNR Level (dB)	((B-A)/B) * 100	((C-A)/C) * 100
Lynx	-10	25.70	27.74
	-5	21.38	35.49
	0	-4.72	31.95
	5	0	37.47
	10	16.61	45.41
F16	-10	14.55	14.28
	-5	23.77	29.43
	0	14.40	37.51
	5	-2.68	34.85
	10	10.89	44.22
Car	-10	4.94	33.33
	-5	19.38	43.23
	0	20.91	46.82
	5	23.15	49.05
	10	20.90	49.05
Babble	-10	23.35	22.98
	-5	30.22	42.84
	0	10.21	38.11
	5	1.47	31.06
	10	7.44	38.51
Stitel	-10	9.56	5.57
	-5	9.45	9.26
	0	-3.58	2.25
	5	3.19	19.10
	10	14.24	36.05

Results for the verification experiment with the GMM-UBM back-end for the female speakers are given in Table 4.10 For the female data, the proposed VAD's performance dropped below the baseline methods only one time, i.e. car noise at -10 dB overall SNR. Similar to the male data, the proposed algorithm is especially effective at lower SNR levels.

Table 4.10. Speaker verification results for the female speakers with UBM method in terms of percent EER (minDCF)

Noise Type	SNR Level (dB)	Proposed Algorithm	Drugman's VAD	Rangachari's Noise Tracking
Lynx	-10	**36.91 (0.69)**	43.80 (0.82)	47.71 (0.88)
	-5	**27.55 (0.52)**	34.51 (0.64)	41.40 (0.78)
	0	**16.67 (0.31)**	18.63 (0.34)	28.86 (0.53)
	5	**9.86 (0.18)**	10.80 (0.20)	17.76 (0.32)
	10	**6.60 (0.12)**	7.61 (0.14)	11.16 (0.20)
F16	-10	**42.13 (0.79)**	47.50 (0.88)	48.73 (0.89)
	-5	**33.57 (0.63)**	42.78 (0.79)	46.62 (0.86)
	0	**23.71 (0.45)**	29.51 (0.55)	37.63 (0.69)
	5	**13.63 (0.25)**	15.15 (0.28)	23.93 (0.44)
	10	**8.41 (0.15)**	8.77 (0.16)	13.92 (0.26)
Car	-10	6.89 (0.12)	**5.94 (0.11)**	8.70 (0.16)
	-5	**5.14 (0.09)**	5.57 (0.10)	8.33 (0.15)
	0	**4.78 (0.08)**	5.51 (0.10)	8.12 (0.15)
	5	**4.49 (0.08)**	5.58 (0.10)	8.04 (0.15)
	10	**4.56 (0.08)**	5.58 (0.10)	8.12 (0.14)
Babble	-10	**37.05 (0.69)**	46.33 (0.86)	48.22 (0.89)
	-5	**27.70 (0.52)**	38.50 (0.70)	44.01 (0.81)
	0	**18.05 (0.33)**	22.84 (0.42)	33.43 (0.62)
	5	**11.38 (0.21)**	12.25 (0.23)	20.66 (0.38)
	10	**7.25 (0.13)**	8.19 (0.15)	12.18 (0.23)
Stitel	-10	**41.55 (0.78)**	45.68 (0.86)	46.84 (0.87)
	-5	**30.96 (0.58)**	36.26 (0.68)	40.24 (0.76)
	0	**19.50 (0.36)**	21.32 (0.40)	27 (0.5)
	5	**10.95 (0.20)**	11.89 (0.22)	16.24 (0.30)
	10	**6.67 (0.12)**	8.48 (0.16)	10.51 (0.20)

Table 4.11 shows the relative percent EER reduction for the female data, compared to the baseline methods. High EER reductions, especially at the lower SNR levels are achieved.

Table 4.11. Relative percent EER reductions for the female speakers with UBM back-end. A: Proposed VAD. B: Drugmans's method. C: Rangachari's method

Noise Type	SNR Level (dB)	((B-A)/B) * 100	((C-A)/C) * 100
Lynx	-10	15.73	22.63
	-5	20.16	33.45
	0	10.52	42.23
	5	8.70	44.48
	10	13.27	40.86
F16	-10	11.30	13.54
	-5	21.52	27.99
	0	19.65	36.99
	5	10.03	43.04
	10	4.10	39.58
Car	-10	-15.99	20.80
	-5	7.72	38.29
	0	13.24	41.13
	5	19.53	44.15
	10	18.28	43.82
Babble	-10	20.03	23.16
	-5	28.05	37.06
	0	20.97	46
	5	7.10	44.91
	10	11.47	40.47
Stitel	-10	9.04	11.29
	-5	14.61	23.06
	0	8.53	27.77
	5	7.90	32.57
	10	21.34	36.53

The SNR based average relative percent EER reduction rates for the GMM-UBM back-end are given in Table 4.12. As seen in the table, a minimum of 12.59% average relative EER reduction was achieved with the proposed VAD. Also, note that improvements for the male and female speakers were similar for the compared methods. Hence, it is clear that the proposed VAD is also do not affected by the gender differences.

Table 4.12. SNR based relative percent EER reduction rates for the GMM-UBM back-end

	MALE		FEMALE	
SNR Level (dB)	**Compared to Drugman's method**	**Compared to Rangachari's method**	**Compared to Drugman's method**	**Compared to Rangachari's method**
-10	15.62	20.78	8.02	18.28
-5	20.84	32.05	18.41	31.97
0	7.44	31.33	14.58	38.82
5	5.02	34.31	10.65	41.83
10	14.02	42.65	13.69	40.25
Average	12.59	32.22	13.07	34.23

EER values averaged over all noise types are also given as bar graphs for visual analysis in Figure 4.5. The proposed VAD (blue bar) remained lower than the baseline methods at all SNR levels. Although the Drugman's method gave similar results at high SNR levels, it fell behind the proposed VAD at low SNR levels.

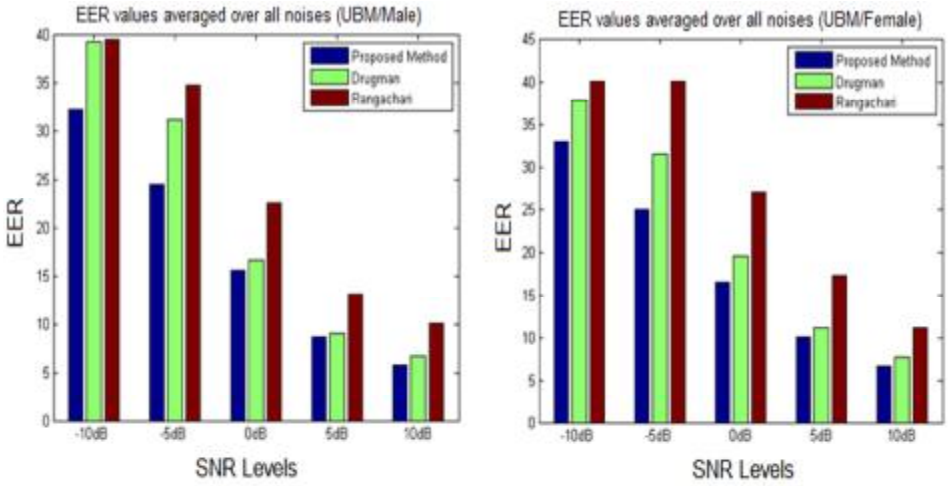

Figure 4.5. EERs averaged over all noise types for the male data (left), and the female data (right)

Before giving the results for the i-vector back-end, it may be beneficial to verify that the PR was the main reason for the performance of the proposed VAD. To this end, the K-MC algorithm was directly applied to the filter-bank outputs, without grouping them with the regression. All other parts of the algorithm remain the same. Table 4.13 shows the results for the female data degraded by the Lynx noise.

Table 4.13. Verification results with and without the polynomial regression, using the UBM back-end

SNR Level (dB)	With polynomial regression	Without polynomial regression
-10	36.91 (0.69)	43.36 (0.81)
-5	27.55 (0.52)	34.08 (0.64)
0	16.67 (0.31)	20.30 (0.37)
5	9.86 (0.18)	11.38 (0.21)
10	6.60 (0.12)	7.61 (0.14)

The benefits of the PR are clearly seen in the table. Directly clustering the frames yielded worse verification than the Drugman's VAD. Hence, the performance of the proposed VAD can be mainly attributed to the PR.

The experimental results with the i-vector back-end are given below. Table 4.14 shows the verification results for the male speakers. The proposed VAD showed superior performance, compared to the other methods. As in the UBM method, the proposed VAD's performance was much better at the lower SNR levels, independent from the noise type.

Table 4.14. Speaker verification results for the male speakers with i-vector method in terms of percent EER (minDCF)

Noise Type	SNR Level (dB)	Proposed Algorithm	Drugman's VAD	Rangachari's Noise Tracking
Lynx	-10	**30.04 (0.55)**	43.57 (0.81)	43.57 (0.81)
	-5	**17.58 (0.33)**	28.36 (0.53)	29.51 (0.55)
	0	**9.48 (0.17)**	14.37 (0.27)	17.43 (0.32)
	5	**5.58 (0.09)**	8.25 (0.14)	11.39 (0.21)
	10	**3.90 (0.06)**	5.35 (0.09)	7.72 (0.14)
F16	-10	**38.60 (0.73)**	48.16 (0.89)	48.93(0.89)
	-5	**27.44 (0.52)**	38.07 (0.71)	37.08 (0.70)
	0	**15.82 (0.30)**	21.71 (0.40)	23.39 (0.43)
	5	**8.48 (0.15)**	11.31 (0.21)	14.98 (0.27)
	10	**5.35 (0.09)**	7.41 (0.13)	10.16 (0.18)
Car	-10	**3.74 (0.06)**	4.66 (0.08)	6.04 (0.11)
	-5	**3.28 (0.05)**	4.43 (0.07)	3.66 (0.06)
	0	**2.98 (0.05)**	4.20 (0.06)	5.58 (0.09)
	5	**3.13 (0.05)**	4.05 (0.06)	5.65 (0.09)
	10	**3.13 (0.04)**	3.97 (0.06)	5.58 (0.09)
Babble	-10	**31.88 (0.60)**	47.24 (0.87)	47.09 (0.88)
	-5	**19.95 (0.37)**	32.95 (0.61)	42.66 (0.80)
	0	**10.85 (0.19)**	18.19 (0.34)	20.18 (0.38)
	5	**5.65 (0.10)**	9.25 (0.17)	12.00 (0.22)
	10	**4.35 (0.07)**	6.11 (0.11)	8.56 (0.15)
Stitel	-10	**37.53 (0.71)**	46.56 (0.87)	45.87 (0.86)
	-5	**22.24 (0.42)**	32.11 (0.60)	31.72 (0.60)
	0	**11.23 (0.20)**	17.35 (0.32)	15.75 (0.29)
	5	**5.81 (0.11)**	9.17 (0.17)	10.24 (0.19)
	10	**4.05 (0.07)**	6.34 (0.11)	7.41 (0.13)

Table 4.15 shows the relative improvements over the baseline methods. The i-vector back-end further improved the proposed VAD's verification performance, compared to the UBM back-end. The Rangachari's NT algorithm yielded the worst scores similar to the previous case.

Table 4.15. Relative percent EER reductions for the male speakers with i-vector back-end. A: Proposed VAD. B: Drugmans's method. C: Rangachari's method

Noise Type	SNR Level (dB)	((B-A)/B) * 100	((C-A)/C) * 100
Lynx	-10	31.05	31.05
	-5	38.01	40.42
	0	34.03	45.61
	5	32.36	51
	10	27.10	49.48
F16	-10	19.85	21.11
	-5	27.92	26
	0	27.13	32.36
	5	25.02	43.39
	10	27.8	47.34
Car	-10	19.74	38.08
	-5	25.95	10.38
	0	29.04	46.59
	5	22.71	44.60
	10	21.15	43.90
Babble	-10	32.51	32.3
	-5	39.45	53.23
	0	40.35	46.23
	5	38.92	52.91
	10	28.80	49.18
Stitel	-10	19.39	18.18
	-5	30.73	29.88
	0	35.27	28.7
	5	36.64	43.26
	10	36.12	45.34

Results for the female data are given in Table 4.16. As expected, the proposed algorithm gave the best verification results, independent from the noise type, or noise level. Table 4.17 shows the relative EER reduction rates.

Table 4.16. Speaker verification results for the female speakers with i-vector back-end in terms of percent EER (minDCF)

Noise Type	SNR Level (dB)	Proposed Algorithm	Drugman's VAD	Rangachari's Noise Tracking
Lynx	-10	**31.32 (0.58)**	41.84 (0.78)	44.52 (0.84)
	-5	**20.16 (0.38)**	29.44 (0.55)	36.76 (0.86)
	0	**11.82 (0.22)**	15.80 (0.30)	23.93 (0.44)
	5	**6.81 (0.12)**	8.34 (0.15)	14.35 (0.26)
	10	**4.13 (0.07)**	4.85 (0.08)	9.64 (0.18)
F16	-10	**38.79 (0.71)**	46.26 (0.85)	47.71 (0.88)
	-5	**27.99 (0.52)**	37.63 (0.70)	42.20 (0.78)
	0	**17.4 (0.33)**	24.43 (0.46)	31.54 (0.59)
	5	**9.93 (0.18)**	11.89 (0.22)	19.29 (0.36)
	10	**5.87 (0.10)**	6.16 (0.11)	12.54 (0.23)
Car	-10	**3.62 (0.06)**	3.77 (0.06)	6.74 (0.12)
	-5	**2.82 (0.05)**	3.19 (0.05)	6.23 (0.11)
	0	**2.75 (0.04)**	3.12 (0.05)	6.09 (0.11)
	5	**2.75 (0.04)**	3.04 (0.05)	6.02 (0.11)
	10	**2.75 (0.04)**	3.04 (0.05)	6.09 (0.11)
Babble	-10	**33.21 (0.63)**	44.81 (0.84)	46.12 (0.84)
	-5	**21.68 (0.40)**	34.15 (0.63)	40.32 (0.75)
	0	**12.98 (0.24)**	20.08 (0.37)	27.19 (0.51)
	5	**6.89 (0.12)**	9.42 (0.17)	16.75 (0.31)
	10	**4.06 (0.07)**	5.07 (0.09)	10.73 (0.19)
Stitel	-10	**33.21 (0.63)**	46.04 (0.85)	45.17 (0.83)
	-5	**26.83 (0.50)**	34.37 (0.64)	35.53 (0.65)
	0	**15.08 (0.28)**	18.92 (0.35)	22.33 (0.42)
	5	**8.12 (0.14)**	10.37 (0.19)	13.77 (0.26)
	10	**4.20 (0.07)**	6.23 (0.11)	8.99 (0.17)

Table 4.17. Relative percent EER reductions for the female speakers with i-vector back-end. A: Proposed VAD. B: Drugmans's method. C: Rangachari's method

Noise Type	SNR Level (dB)	((B-A)/B) * 100	((C-A)/C) * 100
Lynx	-10	25.14	29.65
	-5	31.52	45.15
	0	25.19	50.60
	5	18.34	52.54
	10	14.84	57.15
F16	-10	16.14	18.69
	-5	25.61	33.67
	0	28.77	44.83
	5	16.48	48.52
	10	4.70	53.19
Car	-10	3.97	47.29
	-5	11.59	54.73
	0	11.86	54.84
	5	9.54	54.32
	10	9.54	54.82
Babble	-10	25.88	27.99
	-5	36.51	46.23
	0	35.35	52.26
	5	26.85	58.86
	10	19.92	62.16
Stitel	-10	27.86	26.47
	-5	21.93	24.48
	0	20.29	32.46
	5	21.69	41.03
	10	32.58	53.28

Table 4.18 gives the averaged results over the noise types. The minimum improvement rate was increased to 20.88% with the i-vector method. The frames detected as speech-dominant with the proposed VAD gives the opportunity to estimate the i-vectors more accurately than the other methods.

Table 4.18. SNR based relative percent EER reduction rates for the i-vector back-end

	MALE		FEMALE	
SNR Level (dB)	**Compared to Drugman's method**	**Compared to Rangachari's method**	**Compared to Drugman's method**	**Compared to Rangachari's method**
-10	24.50	28.14	19.79	30.01
-5	32.41	31.98	25.43	40.85
0	33.16	40.13	24.29	46.99
5	31.13	47.03	18.58	51.05
10	28.19	47.04	16.31	56.12
Average	29.87	38.86	20.88	45

The averaged EER values are illustrated in Figure 4.6. The proposed VAD outperformed the other methods at each SNR level, and for both genders. The performance differences between the proposed VAD and the others were increased as the SNR level decreases. This fact indicates that the linear thresholding of the proposed VAD was effective against low SNRs.

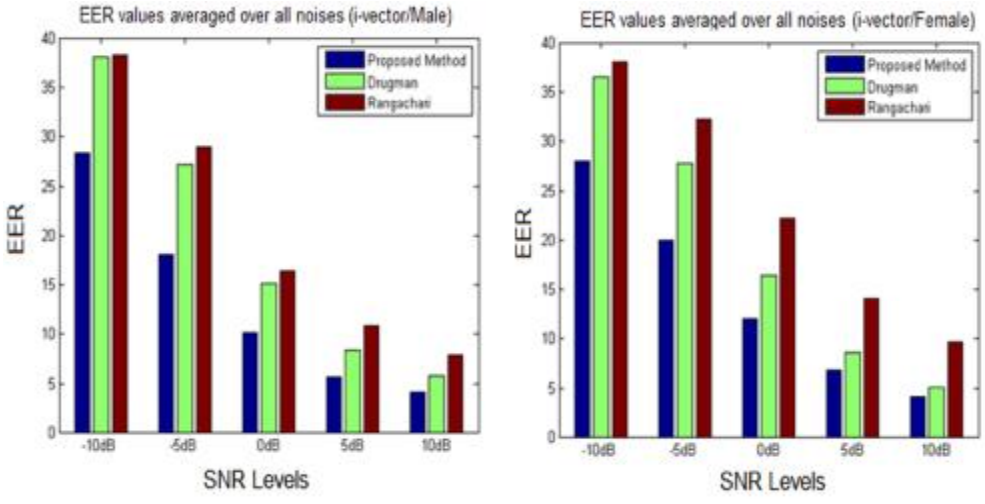

Figure 4.6. EERs averaged over all noise types for the male data (left), and the female data (right)

4.3. Real-Time Speaker Verification

The MFCCs (13 static features and their deltas) were used as the feature vectors. GMMs with 16 Gaussians were trained per word. A UBM model was trained by pooling all available training data. The purpose of this UBM was act as a garbage model, which aims to detect out of the vocabulary words. An unknown utterance was scored with each word model, and with the UBM. If the highest score was obtained by the UBM, the utterance is ignored.

To detect the speech activity in real-time, the initial 200 ms of the recording when the system starts to operate was used to estimate the average silence/noise energy of the environment. Then, the data storage was started/stopped as described in the previous chapter.

As the system is intendent to be used in a project, where an elevator controlled via voice commands will be realized, the system's speaker-independent isolated word recognition results are presented below in Table 4.19.

Table 4.19. Off-line test results for speaker independent isolated word recognition

Uttered Word	**# of Correct Recognition**	**# of Missed Recognition**
0	71	1
1	71	1
2	71	1
3	70	2
4	69	3
5	72	0
6	72	0
7	72	0
8	70	2
9	71	1
Evet (yes)	71	1
Hayır (no)	70	2
Kapan (close)	71	1
Açıl (open)	68	4

Note the results given in Table 4.19 are the off-line results, where three utterances for each word from twenty-four individuals were used. Only nineteen utterances were misclassified. Also, the UBM can effectively capture the out of the vocabulary words, if they are not consisted of phonemes that are very similar to the vocabulary words'.

SV is also realized to recognize the author's identity. Since the database is very limited, and no other speaker was enrolled to the system, a meaningful performance metric cannot be assigned. However, the following figures illustrate example results obtained by the author's voice, and an impostor's voice.

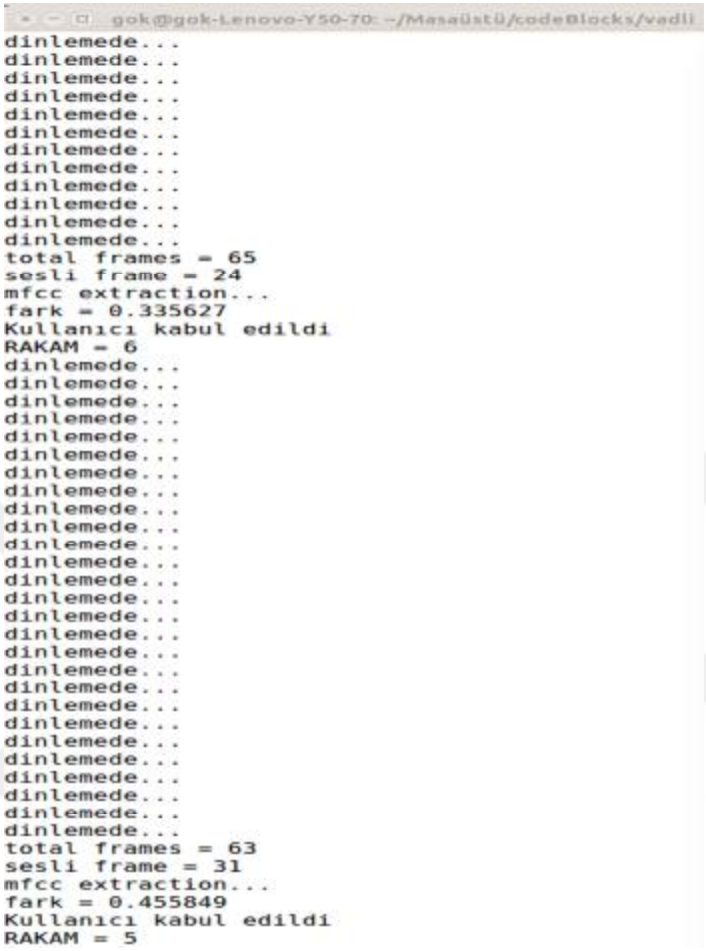

Figure 4.7. Display of an accepted speaker in real-time speaker verification

Figure 4.7 shows the output displayed when the unknown speaker is verified. In this example, the uttered words were six, and five. The utterances were scored with the UBM and the claimed speaker's model. For the first case, where the uttered word was six, the difference between the model scores was 0.335627. In the other case, the difference was 0.455849. Since the positive values indicates that the speaker's model produced a higher score, the unknown speaker was verified.

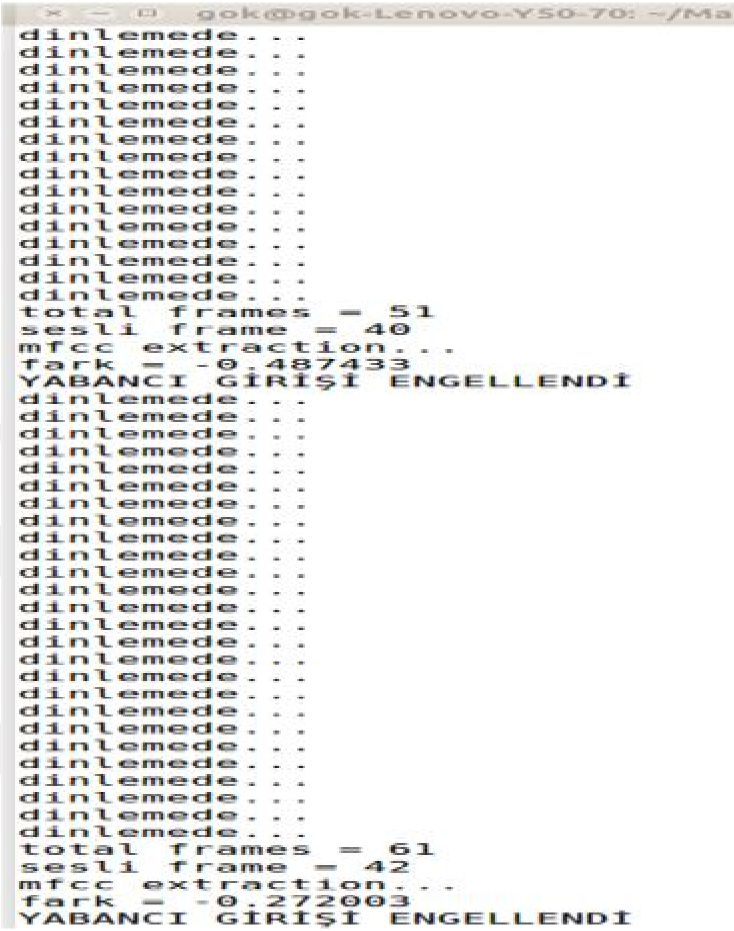

Figure 4.8. Display of a rejected speaker in real-time speaker verification

Figure 4.8 shows the example outputs for a rejected speaker. The differences between the models become negative in this case. Therefore, the unknown speaker is labeled as an imposter, and rejected.

5. DISCUSSION

The experimental results given in the previous chapter proved that the proposed algorithms increased the performance of the SV systems. In this chapter, the results will be analyzed in a more detailed manner.

The SMC method achieved an improved performance over the conventional GMM-UBM, and the state-of-the-art i-vector methods. The improvement is mainly due to a better estimation of the impostor models, as expected from the proposed algorithm.

The algorithm can be viewed as a combination of UBM and the cohort methods. DET curves supported this idea. For a given FRR, the proposed algorithm produces lower FARs as observed from the DET curves. This fact can be seen especially in the short duration utterances (3- and 10-second), and for different-handset conditions. This property makes the proposed method favorable for practical applications. A phone banking system, which verifies the clients (speakers) over phone calls, can be considered as an example. Any enrolled speaker may use different phones at different times, which will create a channel mismatch and also speakers probably want to be verified with a few words, or short phrases. Therefore, a short verification time is preferred. Further, the statistical significance of the verification performances implies that the results found in the experiments were not by coincidence.

Separating the speakers into three clusters yielded the best performance based on the average EER reductions. Nevertheless, the performances of the others were close to each other. This situation is a kind of expected since only the speaker model means were considered for the clustering. As explained before, the speaker model uses the mean vectors of a mixture of the UBM if there is not enough speaker data related to the same mixture.

The speaker clustering method is a simple alternative to achieve verification performances close, or even better, to the I-Vector method. It is

effective either in the matched or mismatched channel conditions. Also, it does not require complex training procedures or excessive storage.

The robustness against additive noise is achieved by the novel VAD algorithm, which includes several parts such as PR, K-MC, speech enhancement, binary voting scheme with a linear thresholding. Compared to the two other methods, the proposed VAD achieved superior verification performances under different noise types and different noise levels. Also, it was shown that I-Vector method increased all competitors' performances, but the proposed VAD benefits from the I-Vectors more than the others.

The higher verification performance obtained by the proposed VAD could be due to the accurately separating the speech-dominant frames from the noisy signal. Further, some of the low magnitude regions should be recovered by the average magnitude representation if their respective group's average magnitude exceeds the threshold.

Also, some of the frames with high magnitudes speech in a few bands can be discarded if the noise is presented in most of the bands, which are expected to decline the verification performance. The utilization of a linear function as a threshold to determine the sufficient speech evidence was proved to be a good approximation to separate speech-dominant frames under noise.

Drugman's VAD method detects speech regions, but as a conventional VAD principle, does not give any hint about the usefulness of the detected frames. The verification performance is expected to decrease if noise is also highly affected the frames. It was verified that a conventional VAD framework did not produce higher verification performances than the proposed method in this thesis.

According to the experimental results, Rangachari's NT algorithm gave the worst results. In the implementation of this algorithm, the enhanced speech was treated as a clean speech. The frames having an energy value higher than the average energy of the utterance were used in the further process. A more suitable threshold to detect speech-dominant regions may increase the verification results.

Since the purpose of this algorithm is also to track the noise, even under the speech activity, the speech/speaker information may be degraded while estimating the noise characteristics. In fact, in (Ramírez et al., 2004) it is stated that while these kind of algorithms can quickly update the noise statistics, they usually capture signal energy during the speech regions.

6. CONCLUSION

The mismatch between the training and testing utterances due to the channel, and additive noises has been a challenging problem that reduces the performances of speech processing systems. In this thesis, two different methods were proposed to overcome this problem for SV systems.

One of the proposed algorithms was the SMC, which is an extension over the conventional GMM-UBM method. The proposed SMC algorithm divides the speaker space of UBM by clustering the speaker models. The cluster centroids were used to create cluster-dependent impostor models. Since each cluster own an impostor model, speakers out of a given cluster could be detected more accurately with this method.

The experimental results showed that the SMC algorithm performed better than the GMM-UBM method at different utterance durations. Also, SV performance was increased for both matched, and mismatched channel conditions. Moreover, the SMC gave comparable SV performances against the state-of-the-art i-vector method. By adding a handset normalization, even better results were achieved.

The performance of the proposed SMC algorithm could be mainly attributed to a more accurate impostor modeling. Despite the improved SV performance, the SMC algorithm does not require a high computational power as the i-vectors do.

The other algorithm proposed in this thesis was the PR based VAD. Its main purpose was to estimate the average noise magnitudes in each filter separately, and enhance the speech information by using the spectral subtraction. The sufficient speech evidence thresholding was proposed to consider the noise bands along with the speech bands in a given frame. This threshold was used to extract the most useful frames, i.e. less affected from the additive noise.

The performance of the proposed VAD was examined under five different noise types, and five different SNR levels for each noise type. The proposed algorithm was compared to a state-of-the-art ANN based VAD, and a highly adaptive noise estimation algorithm. For a fair comparison, the spectral subtraction was used for all the methods.

Compared to the others, the proposed VAD achieved better SV results, especially at low SNR levels for both male, and female speakers. The main reason behind its performance is the PR step. As shown in the experiments, the SV performance of the proposed VAD was decreased without the PR.

Besides, the average noise magnitude could be more accurately estimated with the PR. Also, eliminating the noisy speech frames at the final VAD output decision could be the reason for the high performance at lower SNR levels. This elimination was achieved by the linear thresholding step of the proposed method. As the noise level increased, the threshold was also increased to extract the most useful speech information. Since the SV performance depends on the quality of the frames, this fact explains the better performance observed at lower SNR levels.

Several improvements for each method can be considered in the future works. For the SMC algorithm, sufficient statistics for the i-vectors can be extracted from the impostor models, instead of the UBM.

The linear threshold used in the proposed VAD can be replaced by an exponential function, other speech enhancement algorithms such as Wiener filtering can be added, and different clustering methods can be examined instead of the K-MC.

REFERENCES

Abd El-Fattah, M.A., Dessouky, M.I., Abbas, A.M., Diab, S.M., El-Rabaie, E.-S.M., Al-Nuaimy, W., Alshebeili, S.A., Abd El-samie, F.E., 2014. Speech enhancement with an adaptive Wiener filter. Int. J. Speech Technol. 17, 53–64. doi:10.1007/s10772-013-9205-5

Abd El-Fattah, M.A., Dessouky, M.I., Diab, S.M., Abd El-samie, F.E., 2008. Speech Enhancement Using an Adaptive Wiener Filtering Approach, in: Progress In Electromagnetics Research M. pp. 167–184.

Afify, M., Cui, X., Gao, Y., 2009. Stereo-Based Stochastic Mapping for Robust Speech Recognition. IEEE Trans. Audio. Speech. Lang. Processing 17, 1325–1334. doi:10.1109/TASL.2009.2018017

Ajmera, P.K., Holambe, R.S., 2013. Fractional Fourier transform based features for speaker recognition using support vector machine. Comput. Electr. Eng. 39, 550–557. doi:10.1016/j.compeleceng.2012.05.011

Al-Kaltakchi, M.T.S., Woo, W.L., Dlay, S.S., Chambers, J.A., 2017. Comparison of I-vector and GMM-UBM approaches to speaker identification with TIMIT and NIST 2008 databases in challenging environments, in: 2017 25th European Signal Processing Conference (EUSIPCO). IEEE, Kos, Greece, pp. 533–537. doi:10.23919/EUSIPCO.2017.8081264

Alam, M.J., Kenny, P., O'Shaughnessy, D., 2014. Robust feature extraction based on an asymmetric level-dependent auditory filterbank and a subband spectrum enhancement technique. Digit. Signal Process. 29, 147–157. doi:10.1016/j.dsp.2014.03.001

Alimohad, A., Bouridane, A., Guessoum, A., 2014. Efficient Invariant Features for Sensor Variability Compensation in Speaker Recognition. Sensors 14, 19007–19022. doi:10.3390/s141019007

Ambikairajah, E., Li, H., Thiruvaran, T., Sethu, V., 2015. Spectral shifting of speaker-specific information for narrow band telephonic speaker recognition. Electron. Lett. 51, 2149–2151. doi:10.1049/el.2015.3117

Amin, T. Bin, Marziliano, P., German, J.S., 2014. Glottal and Vocal Tract Characteristics of Voice Impersonators. IEEE Trans. Multimed. 16, 668–678. doi:10.1109/TMM.2014.2300071

Anagnostopoulos, C.-N., Iliou, T., Giannoukos, I., 2015. Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011. Artif. Intell. Rev. 43, 155–177. doi:10.1007/s10462-012-9368-5

Andrews, W.D., Kohler, M.A., Campbell, J.P., Godfrey, J.J., Hernandez-Cordero, J., 2002. Gender-dependent phonetic refraction for speaker recognition, in: IEEE International Conference on Acoustics Speech and Signal Processing. IEEE, p. I-149-I-152. doi:10.1109/ICASSP.2002.5743676

Apsingekar, V.R., De Leon, P.L., 2009. Speaker Model Clustering for Efficient Speaker Identification in Large Population Applications. IEEE Trans. Audio. Speech. Lang. Processing 17, 848–853. doi:10.1109/TASL.2008.2010882

Auckenthaler, R., Carey, M., Lloyd-Thomas, H., 2000. Score Normalization for Text-Independent Speaker Verification Systems. Digit. Signal Process. 10, 42–54. doi:10.1006/dspr.1999.0360

Auckenthaler, R., Mason, J.S., 2001. Gaussian selection applied to text-independent speaker verification, in: Proc. Speaker Odyssey: The Speaker Recognition Workshop. Crete, Greece, pp. 83–88.

Bahari, M.H., McLaren, M., Van hamme, H., van Leeuwen, D.A., 2014. Speaker age estimation using i-vectors. Eng. Appl. Artif. Intell. 34, 99–108. doi:10.1016/j.engappai.2014.05.003

Bahari, M.H., Saeidi, R., Van hamme, H., Van Leeuwen, D., 2013. Accent recognition using i-vector, Gaussian Mean Supervector and Gaussian posterior probability supervector for spontaneous telephone speech, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Vancouver, BC, Canada, pp. 7344–7348. doi:10.1109/ICASSP.2013.6639089

Behravan, H., Hautamäki, V., Kinnunen, T., 2015. Factors affecting i-vector based foreign accent recognition: A case study in spoken Finnish. Speech Commun. 66, 118–129. doi:10.1016/j.specom.2014.10.004

Beigi, H.S.M., Maes, S.H., Chaudhari, U. V., Sorensen, S., 1999. A hierarchical approach to large-scale speaker recognition, in: European Conference on Speech Communication and Technology. pp. 2203–2206.

Ben Kheder, W., Matrouf, D., Bonastre, J.-F., Ajili, M., Bousquet, P.-M., 2015. Additive noise compensation in the i-vector space for speaker recognition, in: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Brisbane, QLD, Australia, pp. 4190–4194. doi:10.1109/ICASSP.2015.7178760

Bennani, Y., Gallinari, P., 1995. Neural networks for discrimination and modelization of speakers. Speech Commun. 17, 159–175. doi:10.1016/0167-6393(95)00014-F

Berouti, M., Schwartz, R., Makhoul, J., 1979. Enhancement of speech corrupted by acoustic noise, in: ICASSP ’79. IEEE International Conference on Acoustics, Speech, and Signal Processing. Institute of Electrical and Electronics Engineers, Washington, DC, USA, pp. 208–211. doi:10.1109/ICASSP.1979.1170788

Besacier, L., Bonastre, J.-F., 2000. Subband architecture for automatic speaker recognition. Signal Processing 80, 1245–1259. doi:10.1016/S0165-1684(00)00033-5

Besacier, L., Bonastre, J.F., Fredouille, C., 2000. Localization and selection of speaker-specific information with statistical modeling. Speech Commun. 31, 89–106. doi:10.1016/S0167-6393(99)00070-9

Bie, F., Wang, D., Wang, J., Zheng, T.F., 2015. Detection and reconstruction of clipped speech for speaker recognition. Speech Commun. 72, 218–231. doi:10.1016/j.specom.2015.06.008

Bilmes, J. a., 1998. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. Int. Comput. Sci. Inst. 4, 1–13. doi:10.1.1.119.4856

Bimbot, F., Bonastre, J.-F., Fredouille, C., Gravier, G., Magrin-Chagnolleau, I., Meignier, S., Merlin, T., Ortega-García, J., Petrovska-Delacrétaz, D., Reynolds, D.A., 2004. A Tutorial on Text-Independent Speaker Verification. EURASIP J. Adv. Signal Process. 2004, 430–451. doi:10.1155/S1110865704310024

Bing Xiang, Berger, T., 2003. Efficient text-independent speaker verification with structural gaussian mixture models and neural network. IEEE Trans. Speech Audio Process. 11, 447–456. doi:10.1109/TSA.2003.815822

Biswas, S., Rohdin, J., Shinoda, K., 2015. Autonomous selection of i-vectors for PLDA modelling in speaker verification. Speech Commun. 72, 32–46. doi:10.1016/j.specom.2015.05.001

Boll, S., 1979. Suppression of acoustic noise in speech using spectral subtraction. IEEE Trans. Acoust. 27, 113–120. doi:10.1109/TASSP.1979.1163209

Brew, A., Cunningham, P., 2010. Combining cohort and UBM models in open set speaker detection. Multimed. Tools Appl. 48, 141–159. doi:10.1007/s11042-009-0381-x

Brew, A., Cunningham, P., 2009. Combining Cohort and UBM Models in Open Set Speaker Identification, in: 2009 Seventh International Workshop on Content-Based Multimedia Indexing. IEEE, pp. 62–67. doi:10.1109/CBMI.2009.30

Buera, L., Lleida, E., Miguel, A., Ortega, A., 2004. Multi-environment models based linear normalization for speech recognition in car conditions, in: 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing. IEEE, Montréal, Canada, p. I-1013-16. doi:10.1109/ICASSP.2004.1326160

Burton, D., 1987. Text-dependent speaker verification using vector quantization source coding. IEEE Trans. Acoust. 35, 133–143. doi:10.1109/TASSP.1987.1165110

Campbell, J.P., 1997. Speaker recognition: a tutorial. Proc. IEEE 85, 1437–1462. doi:10.1109/5.628714

Campbell, W.M., Campbell, J.P., Reynolds, D.A., Singer, E., Torres-Carrasquillo, P.A., 2006a. Support vector machines for speaker and language recognition. Comput. Speech Lang. 20, 210–229. doi:10.1016/j.csl.2005.06.003

Campbell, W.M., Sturim, D.., Reynolds, D.A., 2006b. Support vector machines using GMM supervectors for speaker verification. IEEE Signal Process. Lett. 13, 308–311. doi:10.1109/LSP.2006.870086

Chatzis, V., Bors, A.G., Pitas, I., 1999. Multimodal decision-level fusion for person authentication. IEEE Trans. Syst. Man, Cybern. - Part A Syst. Humans 29, 674–680. doi:10.1109/3468.798073

Choi, W.Y., Song, H.J., Chung, H., Kang, J., Park, J.G., 2016. I-vector based utterance verification for large-vocabulary speech recognition system, in: 2016 First IEEE International Conference on Computer Communication and the Internet (ICCCI). IEEE, Wuhan, China, pp. 316–319. doi:10.1109/CCI.2016.7778933

Chougule, S. V., Chavan, M.S., 2015. Robust Spectral Features for Automatic Speaker Recognition in Mismatch Condition. Procedia Comput. Sci. 58, 272–279. doi:10.1016/j.procs.2015.08.021

Cumani, S., Laface, P., 2014. Factorized Sub-Space Estimation for Fast and Memory Effective I-vector Extraction. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 248–259. doi:10.1109/TASLP.2013.2290505

Cumani, S., Laface, P., 2013. Memory and Computation Trade-Offs for Efficient I-Vector Extraction. IEEE Trans. Audio. Speech. Lang. Processing 21, 934–944. doi:10.1109/TASL.2013.2239291

Cumani, S., Plchot, O., Laface, P., 2014. On the use of i-vector posterior distributions in Probabilistic Linear Discriminant Analysis. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 846–857. doi:10.1109/TASLP.2014.2308473

Damper, R.I., Higgins, J.E., 2003. Improving speaker identification in noise by subband processing and decision fusion. Pattern Recognit. Lett. 24, 2167–2173. doi:10.1016/S0167-8655(03)00082-5

Daqrouq, K., Al Azzawi, K.Y., 2012. Average framing linear prediction coding with wavelet transform for text-independent speaker identification system. Comput. Electr. Eng. 38, 1467–1479. doi:10.1016/j.compeleceng.2012.04.014

Daqrouq, K., Tutunji, T.A., 2015. Speaker identification using vowels features through a combined method of formants, wavelets, and neural network classifiers. Appl. Soft Comput. 27, 231–239. doi:10.1016/j.asoc.2014.11.016

Das, S., Mohn, W., 1971. A scheme for speech processing in automatic speaker verification. IEEE Trans. Audio Electroacoust. 19, 32–43. doi:10.1109/TAU.1971.1162158

Das, S.K., 1969. A Method of Decision Making in Pattern Recognition. IEEE Trans. Comput. C-18, 329–333. doi:10.1109/T-C.1969.222660

Davis, S., Mermelstein, P., 1980. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE Trans. Acoust. 28, 357–366. doi:10.1109/TASSP.1980.1163420

de Leon, P.L., Apsingekar, V., 2007. Reducing Speaker Model Search Space in Speaker Identification, in: 2007 Biometrics Symposium. IEEE, pp. 1–6. doi:10.1109/BCC.2007.4430544

Dehak, N., Karam, Z.N., Reynolds, D.A., Dehak, R., Campbell, W.M., Glass, J.R., 2011a. A channel-blind system for speaker verification, in: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Prague, Czech Republic, pp. 4536–4539. doi:10.1109/ICASSP.2011.5947363

Dehak, N., Kenny, P.J., Dehak, R., Dumouchel, P., Ouellet, P., 2011b. Front-End Factor Analysis for Speaker Verification. IEEE Trans. Audio. Speech. Lang. Processing 19, 788–798. doi:10.1109/TASL.2010.2064307

Denes, P., Mathews, M. V., 1960. Spoken Digit Recognition Using Time-Frequency Pattern Matching. J. Acoust. Soc. Am. 32, 1450–1455. doi:10.1121/1.1907936

Dişken, G., Tüfekci, Z., Çevik, U., 2017a. Speaker Model Clustering to Construct Background Models for Speaker Verification. Arch. Acoust. 42, 127–135. doi:10.1515/aoa-2017-0014

Dişken, G., Tüfekci, Z., Çevik, U., 2017b. A robust polynomial regression-based voice activity detector for speaker verification. EURASIP J. Audio, Speech, Music Process. 2017, 23. doi:10.1186/s13636-017-0120-6

Dişken, G., Tüfekçi, Z., Saribulut, L., Çevik, U., 2017c. A Review on Feature Extraction for Speaker Recognition under Degraded Conditions. IETE Tech. Rev. 34, 321–332. doi:10.1080/02564602.2016.1185976

Doclo, S., Spriet, A., Wouters, J., Moonen, M., 2007. Frequency-domain criterion for the speech distortion weighted multichannel Wiener filter for robust noise reduction. Speech Commun. 49, 636–656. doi:10.1016/j.specom.2007.02.001

Doddington, G.R., Przybocki, M.A., Martin, A.F., Reynolds, D.A., 2000. The NIST speaker recognition evaluation – Overview, methodology, systems, results, perspective. Speech Commun. 31, 225–254. doi:10.1016/S0167-6393(99)00080-1

Drgas, S., Dabrowski, A., 2015. Speaker recognition based on multilevel speech signal analysis on Polish corpus. Multimed. Tools Appl. 74, 4195–4211. doi:10.1007/s11042-013-1502-0

Drugman, T., Alku, P., Alwan, A., Yegnanarayana, B., 2014. Glottal source processing: From analysis to applications. Comput. Speech Lang. 28, 1117–1138. doi:10.1016/j.csl.2014.03.003

Drugman, T., Dutoit, T., 2012. The Deterministic Plus Stochastic Model of the Residual Signal and Its Applications. IEEE Trans. Audio. Speech. Lang. Processing 20, 968–981. doi:10.1109/TASL.2011.2169787

Drugman, T., Stylianou, Y., Kida, Y., Akamine, M., 2016. Voice Activity Detection: Merging Source and Filter-based Information. IEEE Signal Process. Lett. 23, 252–256. doi:10.1109/LSP.2015.2495219

Edwards, J.S., Ramachandran, R.P., Thayasivam, U., 2017. Robust speaker verification with a two classifier format and feature enhancement, in: 2017 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, Baltimore, MD, USA, pp. 1–4. doi:10.1109/ISCAS.2017.8050775

Erell, A., Weintraub, M., 1993. Filterbank-energy estimation using mixture and Markov models for recognition of noisy speech. IEEE Trans. Speech Audio Process. 1, 68–76. doi:10.1109/89.221385

Eyben, F., Weninger, F., Squartini, S., Schuller, B., 2013. Real-life voice activity detection with LSTM Recurrent Neural Networks and an application to Hollywood movies, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Vancouver, BC, Canada, pp. 483–487. doi:10.1109/ICASSP.2013.6637694

Farrell, K.R., Mammone, R.J., Assaleh, K.T., 1994. Speaker recognition using neural networks and conventional classifiers. IEEE Trans. Speech Audio Process. 2, 194–205. doi:10.1109/89.260362

Ferrer, L., Shriberg, E., Kajarekar, S., Sonmez, K., 2007. Parameterization of Prosodic Feature Distributions for SVM Modeling in Speaker Recognition, in: 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07. IEEE, Honolulu, HI, USA, p. IV-233-236. doi:10.1109/ICASSP.2007.366892

Fischer, D., Gerkmann, T., 2016. Single-microphone speech enhancement using MVDR filtering and Wiener post-filtering, in: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings. Shanghai, China, pp. 201–205.

Frankle, M.N., Ramachandran, R.P., 2016. Robust speaker identification under noisy conditions using feature compensation and signal to noise ratio estimation, in: 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, Abu Dhabi, United Arab Emirates, pp. 1–4. doi:10.1109/MWSCAS.2016.7869973

Furui, S., 1997. Recent advances in speaker recognition. Pattern Recognit. Lett. 18, 859–872. doi:10.1016/S0167-8655(97)00073-1

Furui, S., 1981. Cepstral analysis technique for automatic speaker verification. IEEE Trans. Acoust. 29, 254–272. doi:10.1109/TASSP.1981.1163530

Gales, M.F.J., Young, S.J., 1995. Robust speech recognition in additive and convolutional noise using parallel model combination. Comput. Speech Lang. 9, 289–307. doi:10.1006/csla.1995.0014

Ganapathy, S., Mallidi, S.H., Hermansky, H., 2014. Robust Feature Extraction Using Modulation Filtering of Autoregressive Models. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 1285–1295. doi:10.1109/TASLP.2014.2329190

Garimella, S., Hermansky, H., 2013. Factor Analysis of Auto-Associative Neural Networks With Application in Speaker Verification. IEEE Trans. Neural Networks Learn. Syst. 24, 522–528. doi:10.1109/TNNLS.2012.2236652

Garimella, S., Mallidi, S.H., Hermansky, H., 2012. Regularized Auto-Associative Neural Networks for Speaker Verification. IEEE Signal Process. Lett. 19, 841–844. doi:10.1109/LSP.2012.2221706

Garreton, C., Yoma, N.B., 2012. Telephone Channel Compensation in Speaker Verification Using a Polynomial Approximation in the Log-Filter-Bank Energy Domain. IEEE Trans. Audio. Speech. Lang. Processing 20, 336–341. doi:10.1109/TASL.2011.2157495

Gerkmann, T., Hendriks, R.C., 2012. Unbiased MMSE-Based Noise Power Estimation With Low Complexity and Low Tracking Delay. IEEE Trans. Audio. Speech. Lang. Processing 20, 1383–1393. doi:10.1109/TASL.2011.2180896

Ghalehjegh, S.H., Rose, R.C., 2015. Deep bottleneck features for i-vector based text-independent speaker verification, in: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, Scottsdale, AZ, USA, pp. 555–560. doi:10.1109/ASRU.2015.7404844

Ghosh, P.K., Tsiartas, A., Narayanan, S., 2011. Robust Voice Activity Detection Using Long-Term Signal Variability. IEEE Trans. Audio. Speech. Lang. Processing 19, 600–613. doi:10.1109/TASL.2010.2052803

Gillick, L., Cox, S.J., 1989. Some statistical issues in the comparison of speech recognition algorithms, in: International Conference on Acoustics, Speech, and Signal Processing. IEEE, pp. 532–535. doi:10.1109/ICASSP.1989.266481

Gish, H., Karnofsky, K., Krasner, M., Roucos, S., Schwartz, R., Wolf, J., 1985. Investigation of text-independent speaker indentification over telephone channels, in: ICASSP ’85. IEEE International Conference on Acoustics, Speech, and Signal Processing. Institute of Electrical and Electronics Engineers, Tampa, FL, USA, pp. 379–382. doi:10.1109/ICASSP.1985.1168410

Gorriz, J.M., Ramirez, J., Lang, E.W., Puntonet, C.G., 2008. Jointly Gaussian PDF-Based Likelihood Ratio Test for Voice Activity Detection. IEEE Trans. Audio. Speech. Lang. Processing 16, 1565–1578. doi:10.1109/TASL.2008.2004293

Govindan, S.M., Duraisamy, P., Yuan, X., 2014. Adaptive wavelet shrinkage for noise robust speaker recognition. Digit. Signal Process. 33, 180–190. doi:10.1016/j.dsp.2014.06.007

Gudnason, J., Brookes, M., 2008. Voice source cepstrum coefficients for speaker identification, in: 2008 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Las Vegas, NV, USA, pp. 4821–4824. doi:10.1109/ICASSP.2008.4518736

Hafen, R.P., Henry, M.J., 2012. Speech information retrieval: a review. Multimed. Syst. 18, 499–518. doi:10.1007/s00530-012-0266-0

Hanilci, C., Kinnunen, T., Ertas, F., Saeidi, R., Pohjalainen, J., Alku, P., 2012. Regularized All-Pole Models for Speaker Verification Under Noisy Environments. IEEE Signal Process. Lett. 19, 163–166. doi:10.1109/LSP.2012.2184284

Hasan, T., Hansen, J.H.L., 2014. Maximum Likelihood Acoustic Factor Analysis Models for Robust Speaker Verification in Noise. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 381–391. doi:10.1109/TASLP.2013.2292356

Hasan, T., Hansen, J.H.L., 2013. Acoustic factor analysis for robust speaker verification. IEEE Trans. Audio, Speech Lang. Process. 21, 842–853. doi:10.1109/TASL.2012.2226161

Hautamki, V., Kinnunen, T., Krkkinen, I., Saastamoinen, J., Tuononen, M., Frnti, P., 2008. Maximum a Posteriori Adaptation of the Centroid Model for Speaker Verification. IEEE Signal Process. Lett. 15, 162–165. doi:10.1109/LSP.2007.914792

Hendriks, R.C., Heusdens, R., Jensen, J., 2010. MMSE based noise PSD tracking with low complexity, in: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Dallas, TX, USA, pp. 4266–4269. doi:10.1109/ICASSP.2010.5495680

Hendriks, R.C., Jensen, J., Heusdens, R., 2008. Noise Tracking Using DFT Domain Subspace Decompositions. IEEE Trans. Audio. Speech. Lang. Processing 16, 541–553. doi:10.1109/TASL.2007.914977

Hermansky, H., Morgan, N., 1994. RASTA processing of speech. IEEE Trans. Speech Audio Process. 2, 578–589. doi:10.1109/89.326616

Higgins, A.L., Bahler, L.G., Porter, J.E., 1993. Voice identification using nearest-neighbor distance measure, in: IEEE International Conference on Acoustics Speech and Signal Processing. IEEE, Minneapolis, MN, USA, pp. 375–378 vol.2. doi:10.1109/ICASSP.1993.319317

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., Kingsbury, B., 2012. Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. IEEE Signal Process. Mag. 29, 82–97. doi:10.1109/MSP.2012.2205597

Hongzhi Wang, Yuchao Xu, Meijing Li, 2011. Study on the MFCC similarity-based voice activity detection algorithm, in: 2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC). IEEE, Dengleng, China, pp. 4391–4394. doi:10.1109/AIMSEC.2011.6009945

Hou Fenglei, Wang Bingxi, 2001. Text-independent speaker recognition using support vector machine, in: 2001 International Conferences on Info-Tech and Info-Net. Proceedings (Cat. No.01EX479). IEEE, Beijing, China, pp. 402–407. doi:10.1109/ICII.2001.983090

Hu, Y., Loizou, P.C., 2007. Subjective comparison and evaluation of speech enhancement algorithms. Speech Commun. 49, 588–601. doi:10.1016/j.specom.2006.12.006

Huang, H., Xu, Y., Zhou, R., Yan, Y., 2015. Feature recovery for noise-robust speaker verification. Electron. Lett. 51, 1459–1461. doi:10.1049/el.2015.1418

Hughes, T., Mierle, K., 2013. Recurrent neural networks for voice activity detection, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Vancouver, BC, Canada, pp. 7378–7382. doi:10.1109/ICASSP.2013.6639096

In-Chul Yoo, Hyeontaek Lim, Dongsuk Yook, 2015. Formant-Based Robust Voice Activity Detection. IEEE/ACM Trans. Audio, Speech, Lang. Process. 23, 2238–2245. doi:10.1109/TASLP.2015.2476762

Jain, A.K., Ross, A., Pankanti, S., 2006. Biometrics: A Tool for Information Security. IEEE Trans. Inf. Forensics Secur. 1, 125–143. doi:10.1109/TIFS.2006.873653

Jiang, S., Frigui, H., Calhoun, A.W., 2015. Speaker Identification in Medical Simulation Data Using Fisher Vector Representation, in: 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA). IEEE, Miami, FL, USA, pp. 197–201. doi:10.1109/ICMLA.2015.187

Jing Pang, 2017. Spectrum energy based voice activity detection, in: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, Las Vegas, NV, USA, pp. 1–5. doi:10.1109/CCWC.2017.7868454

Jongseo Sohn, Nam Soo Kim, Wonyong Sung, 1999. A statistical model-based voice activity detection. IEEE Signal Process. Lett. 6, 1–3. doi:10.1109/97.736233

Joon-Hyuk Chang, Nam Soo Kim, Mitra, S.K., 2006. Voice activity detection based on multiple statistical models. IEEE Trans. Signal Process. 54, 1965–1976. doi:10.1109/TSP.2006.874403

Jun Yao, Yuan-Ting Zhang, 2001. Bionic wavelet transform: a new time-frequency method based on an auditory model. IEEE Trans. Biomed. Eng. 48, 856–863. doi:10.1109/10.936362

Kamath, S., Loizou, P., 2002. A multi-band spectral subtraction method for enhancing speech corrupted by colored noise, in: IEEE International Conference on Acoustics Speech and Signal Processing. IEEE, Orlando, FL, USA, p. IV-4164-IV-4164. doi:10.1109/ICASSP.2002.5745591

Kanagasundaram, A., Dean, D., Sridharan, S., McLaren, M., Vogt, R., 2014. I-vector based speaker recognition using advanced channel compensation techniques. Comput. Speech Lang. 28, 121–140. doi:10.1016/j.csl.2013.04.002

Kenny, P., 2005. Joint factor analysis of speaker and session variability: Theory and algorithms. CRIM, Montr. CRIM-06/08-13 1–17.

Kenny, P., Boulianne, G., Dumouchel, P., 2005. Eigenvoice modeling with sparse training data. IEEE Trans. Speech Audio Process. 13, 345–354. doi:10.1109/TSA.2004.840940

Kenny, P., Dumouchel, P., 2004. Experiments in Speaker Verification using Factor Analysis Likelihood Ratios. Odyssey - Speak. Lang. Recognit. Work.

Khan, S., Basu, J., Bepari, M.S., 2012. Performance Evaluation of PBDP Based Real-Time Speaker Identification System with Normal MFCC vs MFCC of LP Residual Features, in: Perception and Machine Intelligence. Lecture Notes in Computer Science, Vol 7143. pp. 358–366. doi:10.1007/978-3-642-27387-2_44

Kim, D.K., Jang, K.W., Chang, J.-H., 2007. A New Statistical Voice Activity Detection Based on UMP Test. IEEE Signal Process. Lett. 14, 891–894. doi:10.1109/LSP.2007.900225

Kinnunen, T., Alku, P., 2009. On separating glottal source and vocal tract information in telephony speaker verification, in: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Taipei, Taiwan, pp. 4545–4548. doi:10.1109/ICASSP.2009.4960641

Kinnunen, T., Li, H., 2010. An overview of text-independent speaker recognition: From features to supervectors. Speech Commun. 52, 12–40. doi:10.1016/j.specom.2009.08.009

Kinnunen, T., Saeidi, R., Sedlak, F., Lee, K.A., Sandberg, J., Hansson-Sandsten, M., Li, H., 2012. Low-Variance Multitaper MFCC Features: A Case Study in Robust Speaker Verification. IEEE Trans. Audio. Speech. Lang. Processing 20, 1990–2001. doi:10.1109/TASL.2012.2191960

Klein, A., Feldes, S., 2016. HMM Embedded Conditional Vector Estimation Applied to Noisy Line Spectral Frequencies, in: Proceedings of 12. ITG Speech Communication Syposium. Paderborn, Germany, pp. 332–336.

Kockmann, M., Burget, L., “Honza” Černocký, J., 2011. Application of speaker- and language identification state-of-the-art techniques for emotion recognition. Speech Commun. 53, 1172–1185. doi:10.1016/j.specom.2011.01.007

Kotnik, B., 2003. Efficient Noise Robust Feature Extraction Algorithms for Distributed Speech Recognition (DSR) Systems. Int. J. Speech Technol. 6, 205–219. doi:10.1023/A:1023410018862

Kua, J.M.K., Epps, J., Ambikairajah, E., 2013. i-Vector with sparse representation classification for speaker verification. Speech Commun. 55, 707–720. doi:10.1016/j.specom.2013.01.005

Kua, J.M.K., Thiruvaran, T., Nosratighods, M., Ambikairajah, E., Epps, J., 2010. Investigation of Spectral Centroid Magnitude and Frequency for Speaker Recognition, in: Odyssey 2010 - Speaker and Language Recognition Workshop. Brno, Czech Republic, pp. 34–39.

Kuhn, R., Junqua, J.C., Nguyen, P., Niedzielski, N., 2000. Rapid speaker adaptation in eigenvoice space. IEEE Trans. Speech Audio Process. 8, 695–706. doi:10.1109/89.876308

Lei, Y., Burget, L., Scheffer, N., 2013. A noise robust i-vector extractor using vector taylor series for speaker recognition, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Vancouver, BC, Canada, pp. 6788–6791. doi:10.1109/ICASSP.2013.6638976

Lei, Y., Scheffer, N., Ferrer, L., McLaren, M., 2014. A novel scheme for speaker recognition using a phonetically-aware deep neural network, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Florence, Italy, pp. 1695–1699. doi:10.1109/ICASSP.2014.6853887

Leung, K.Y., Mak, M.W., Siu, M.H., Kung, S.Y., 2006. Adaptive articulatory feature-based conditional pronunciation modeling for speaker verification. Speech Commun. 48, 71–84. doi:10.1016/j.specom.2005.05.013

Li, K.P., Dammann, J.E., Chapman, W.D., 1966. Experimental Studies in Speaker Verification, Using an Adaptive System. J. Acoust. Soc. Am. 40, 966–978. doi:10.1121/1.1910221

Li, M., Kim, J., Lammert, A., Ghosh, P.K., Ramanarayanan, V., Narayanan, S., 2016. Speaker verification based on the fusion of speech acoustics and inverted articulatory signals. Comput. Speech Lang. 36, 196–211. doi:10.1016/j.csl.2015.05.003

Li, N., Mak, M.-W., Chien, J.-T., 2017. DNN-Driven Mixture of PLDA for Robust Speaker Verification. IEEE/ACM Trans. Audio, Speech, Lang. Process. 25, 1371–1383. doi:10.1109/TASLP.2017.2692304

Li, P., Hu, F., Li, Y., Xu, Y., 2014. Speaker identification using linear predictive cepstral coefficients and general regression neural network, in: Proceedings of the 33rd Chinese Control Conference. IEEE, Nanjing, China, pp. 4952–4956. doi:10.1109/ChiCC.2014.6895780

Li, Q., Huang, Y., 2011. An Auditory-Based Feature Extraction Algorithm for Robust Speaker Identification Under Mismatched Conditions. IEEE Trans. Audio. Speech. Lang. Processing 19, 1791–1801. doi:10.1109/TASL.2010.2101594

Li Deng, Acero, A., Li Jiang, Droppo, J., Xuedong Huang, 2001. High-performance robust speech recognition using stereo training data, in: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221). IEEE, Salt Lake City, UT, USA, pp. 301–304. doi:10.1109/ICASSP.2001.940827

Li Hui, Bei-Qian Dai, Lu Wei, 2006. A Pitch Detection Algorithm Based on AMDF and ACF, in: 2006 IEEE International Conference on Acoustics Speed and Signal Processing Proceedings. IEEE, Toulouse, France, p. I-377-I-380. doi:10.1109/ICASSP.2006.1660036

Lim, J.S., Oppenheim, A.V., 1979. Enhancement and bandwidth compression of noisy speech. Proc. IEEE 67, 1586–1604. doi:10.1109/PROC.1979.11540

Lin, L., Wang, S., 2006. A Kernel Method For Speaker Recognıtıon Wıth Lıttle Data, in: 2006 8th International Conference on Signal Processing. IEEE, Beijing, China, pp. 1–4. doi:10.1109/ICOSP.2006.345523

Linde, Y., Buzo, A., Gray, R., 1980. An Algorithm for Vector Quantizer Design. IEEE Trans. Commun. 28, 84–95. doi:10.1109/TCOM.1980.1094577

Liu, T., Kang, K., 2014. I-vector Based Text-Independent Speaker Identification 5420–5425.

Luck, J.E., 1969. Automatic Speaker Verification Using Cepstral Measurements. J. Acoust. Soc. Am. 46, 1026–1032. doi:10.1121/1.1911795

Lung, J.W.J., Salam, M.S.H., Rehman, A., Rahim, M.S.M., Saba, T., 2014. Fuzzy Phoneme Classification Using Multi-speaker Vocal Tract Length Normalization. IETE Tech. Rev. 31, 128–136. doi:10.1080/02564602.2014.892669

Magrin-Chagnolleau, I., Durou, G., Bimbot, F., 2002. Application of time-frequency principal component analysis to text-independent speaker identification. IEEE Trans. Speech Audio Process. 10, 371–378. doi:10.1109/TSA.2002.800557

Mak, M.-W., Pang, X., Chien, J.-T., 2016. Mixture of PLDA for Noise Robust I-Vector Speaker Verification. IEEE/ACM Trans. Audio, Speech, Lang. Process. 24, 130–142. doi:10.1109/TASLP.2015.2499038

Mak, M.-W., Yu, H.-B., 2014. A study of voice activity detection techniques for NIST speaker recognition evaluations. Comput. Speech Lang. 28, 295–313. doi:10.1016/j.csl.2013.07.003

Malik, S., Afsar, F.A., 2009. Wavelet transform based automatic speaker recognition, in: 2009 IEEE 13th International Multitopic Conference. IEEE, pp. 1–4. doi:10.1109/INMIC.2009.5383083

Mammone, R.J., Xiaoyu Zhang, Ramachandran, R.P., 1996. Robust speaker recognition: a feature-based approach. IEEE Signal Process. Mag. 13, 58. doi:10.1109/79.536825

Martin, R., 2006. Bias compensation methods for minimum statistics noise power spectral density estimation. Signal Processing 86, 1215–1229. doi:10.1016/j.sigpro.2005.07.037

Martin, R., 2001. Noise power spectral density estimation based on optimal smoothing and minimum statistics. IEEE Trans. Speech Audio Process. 9, 504–512. doi:10.1109/89.928915

Martin, R., 1994. Spectral Subtraction Based on Minimum Statistics, in: European Signal Processing Conference. Edinburgh, pp. 1182–1185.

Martinez, D., Burget, L., Stafylakis, T., Lei, Y., Kenny, P., Lleida, E., 2014. Unscented transform for ivector-based noisy speaker recognition, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Florence, Italy, pp. 4042–4046. doi:10.1109/ICASSP.2014.6854361

Matsui, T., Furui, S., 1992. Comparison of text-independent speaker recognition methods using VQ-distortion and discrete/continuous HMMs, in: [Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing. IEEE, pp. 157–160 vol.2. doi:10.1109/ICASSP.1992.226096

May, T., van de Par, S., Kohlrausch, A., 2012. Noise-Robust Speaker Recognition Combining Missing Data Techniques and Universal Background Modeling. IEEE Trans. Audio. Speech. Lang. Processing 20, 108–121. doi:10.1109/TASL.2011.2158309

McClanahan, R., De Leon, P.L., 2015. Reducing computation in an i-vector speaker recognition system using a tree-structured universal background model. Speech Commun. 66, 36–46. doi:10.1016/j.specom.2014.07.003

McClanahan, R.D., DeLeon, P.L., 2012. Mixture Component Clustering for Efficient Speaker Verification, in: Interspeech. Portland, USA, pp. 1086–1090.

McLaren, M., Lei, Y., Ferrer, L., 2014. Application of Convolutional Neural Networks to Speaker Recognition in Noisy Conditions, in: INTERSPEECH 2014. Singapore, pp. 686–690.

McLaren, M., Scheffer, N., Graciarena, M., Ferrer, L., Lei, Y., 2013. Improving speaker identification robustness to highly channel-degraded speech through multiple system fusion, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Vancouver, BC, Canada, pp. 6773–6777. doi:10.1109/ICASSP.2013.6638973

McLaren, M., van Leeuwen, D., 2012. Source-Normalized LDA for Robust Speaker Recognition Using i-Vectors From Multiple Speech Sources. IEEE Trans. Audio. Speech. Lang. Processing 20, 755–766. doi:10.1109/TASL.2011.2164533

McLaren, M., Vogt, R., Baker, B., Sridharan, S., 2010. Data-Driven Background Dataset Selection for SVM-Based Speaker Verification. IEEE Trans. Audio. Speech. Lang. Processing 18, 1496–1506. doi:10.1109/TASL.2009.2035786

Mehrabani, M., Hansen, J.H.L., 2013. Singing speaker clustering based on subspace learning in the GMM mean supervector space. Speech Commun. 55, 653–666. doi:10.1016/j.specom.2012.11.001

Moattar, M.H., Homayounpour, M.M., 2009. A Simple but Efficient Real-Time Voice Activity Detection Algorithm, in: 17th European Signal Processing Conference (EUSIPCO 2009). Glasgow, Scotland, August 24-28, pp. 2549–2553.

Montalvão, J., Rodrigues Araujo, M.R., 2012. Is masking a relevant aspect lacking in MFCC? A speaker verification perspective. Pattern Recognit. Lett. 33, 2156–2165. doi:10.1016/j.patrec.2012.07.023

Moreno, P.J., Raj, B., Stern, R.M., 1998. Data-driven environmental compensation for speech recognition: A unified approach. Speech Commun. 24, 267–285. doi:10.1016/S0167-6393(98)00025-9

Moreno, P.J., Raj, B., Stern, R.M., 1996. A vector Taylor series approach for environment-independent speech recognition, in: 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings. IEEE, Atlanta, GA, USA, pp. 733–736. doi:10.1109/ICASSP.1996.543225

Murty, K.S.R., Yegnanarayana, B., 2006. Combining evidence from residual phase and MFCC features for speaker recognition. IEEE Signal Process. Lett. 13, 52–55. doi:10.1109/LSP.2005.860538

Oglesby, J., Mason, J.S., 1991. Radial basis function networks for speaker recognition, in: [Proceedings] ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing. IEEE, Toronto, Ontario, Canada, pp. 393–396 vol.1. doi:10.1109/ICASSP.1991.150359

Oglesby, J., Mason, J.S., 1990. Optimisation of neural models for speaker identification, in: International Conference on Acoustics, Speech, and Signal Processing. IEEE, Albuquerque, NM, USA, pp. 261–264. doi:10.1109/ICASSP.1990.115617

Ostrogonac, S., Secujski, M., Knezevic, D., Suzic, S., 2013. Extraction of glottal features for speaker recognition, in: 2013 IEEE 9th International Conference on Computational Cybernetics (ICCC). IEEE, pp. 369–373. doi:10.1109/ICCCyb.2013.6617621

Paliwal, K., Wójcicki, K., Schwerin, B., 2010. Single-channel speech enhancement using spectral subtraction in the short-time modulation domain. Speech Commun. 52, 450–475. doi:10.1016/j.specom.2010.02.004

Pallet, D.S., Fisher, W.M., Fiscus, J.G., 1990. Tools for the analysis of benchmark speech recognition tests, in: International Conference on Acoustics, Speech, and Signal Processing. IEEE, pp. 97–100. doi:10.1109/ICASSP.1990.115546

Pang, X., Mak, M.-W., 2015. Noise robust speaker verification via the fusion of SNR-independent and SNR-dependent PLDA. Int. J. Speech Technol. 18, 633–648. doi:10.1007/s10772-015-9310-8

Pati, D., Mahadeva Prasanna, S., 2010. Speaker Recognition from Excitation Source Perspective. IETE Tech. Rev. 27, 138–157. doi:10.4103/0256-4602.60167

Plchot, O., Matsoukas, S., Matejka, P., Dehak, N., Ma, J., Cumani, S., Glembek, O., Hermansky, H., Mallidi, S.H., Mesgarani, N., Schwartz, R., Soufifar, M., Tan, Z.H., Thomas, S., Zhang, B., Zhou, X., 2013. Developing a speaker identification system for the DARPA RATS project, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Vancouver, BC, Canada, pp. 6768–6772. doi:10.1109/ICASSP.2013.6638972

Poblete, V., Espic, F., King, S., Stern, R.M., Huenupán, F., Fredes, J., Yoma, N.B., 2015. A perceptually-motivated low-complexity instantaneous linear channel normalization technique applied to speaker verification. Comput. Speech Lang. 31, 1–27. doi:10.1016/j.csl.2014.10.006

Poblete, V., Yoma, N.B., Stern, R.M., 2014. Optimization of the parameters characterizing sigmoidal rate-level functions based on acoustic features. Speech Commun. 56, 19–34. doi:10.1016/j.specom.2013.07.006

Pohjalainen, J., Hanilci, C., Kinnunen, T., Alku, P., 2014. Mixture Linear Prediction in Speaker Verification Under Vocal Effort Mismatch. IEEE Signal Process. Lett. 21, 1516–1520. doi:10.1109/LSP.2014.2339632

Pradhan, G., Prasanna, S.R.M., 2013. Speaker Verification by Vowel and Nonvowel Like Segmentation. IEEE Trans. Audio. Speech. Lang. Processing 21, 854–867. doi:10.1109/TASL.2013.2238529

Prasanna, S.R.M., Pradhan, G., 2011. Significance of Vowel-Like Regions for Speaker Verification Under Degraded Conditions. IEEE Trans. Audio. Speech. Lang. Processing 19, 2552–2565. doi:10.1109/TASL.2011.2155061

Pruzansky, S., 1963. Pattern-Matching Procedure for Automatic Talker Recognition. J. Acoust. Soc. Am. 35, 354–358. doi:10.1121/1.1918467

Qi Li, Jinsong Zheng, Tsai, A., Qiru Zhou, 2002. Robust endpoint detection and energy normalization for real-time speech and speaker recognition. IEEE Trans. Speech Audio Process. 10, 146–157. doi:10.1109/TSA.2002.1001979

Rabiner, L., Sambur, M., 1977. Voiced-unvoiced-silence detection using the Itakura LPC distance measure, in: ICASSP '77. IEEE International Conference on Acoustics, Speech, and Signal Processing. Institute of Electrical and Electronics Engineers, Hartford, CT, USA, pp. 323–326. doi:10.1109/ICASSP.1977.1170330

Rajan, P., Kinnunen, T., Hautamäki, V., 2013. Effect of Multicondition Training on i-Vector PLDA Configurations for Speaker Recognition, in: INTERSPEECH 2013. Lyon, France, pp. 3694–3697.

Ramírez, J., Segura, J.C., Benítez, C., de la Torre, Á., Rubio, A., 2004. Efficient voice activity detection algorithms using long-term speech information. Speech Commun. 42, 271–287. doi:10.1016/j.specom.2003.10.002

Ramirez, J., Segura, J.C., Benitez, C., DelaTorre, A., Rubio, A.J., 2004. A New Kullback–Leibler VAD for Speech Recognition in Noise. IEEE Signal Process. Lett. 11, 266–269. doi:10.1109/LSP.2003.821762

Ramirez, J., Segura, J.C., Benitez, C., Garcia, L., Rubio, A., 2005. Statistical voice activity detection using a multiple observation likelihood ratio test. IEEE Signal Process. Lett. 12, 689–692. doi:10.1109/LSP.2005.855551

Rangachari, S., Loizou, P.C., 2006. A noise-estimation algorithm for highly non-stationary environments. Speech Commun. 48, 220–231. doi:10.1016/j.specom.2005.08.005

Rao, W., Mak, M.-W., 2013. Boosting the Performance of I-Vector Based Speaker Verification via Utterance Partitioning. IEEE Trans. Audio. Speech. Lang. Processing 21, 1012–1022. doi:10.1109/TASL.2013.2243436

Rao, W., Mak, M.-W., Lee, K.-A., 2015. Normalization of total variability matrix for i-vector/PLDA speaker verification, in: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 4180–4184. doi:10.1109/ICASSP.2015.7178758

Rao, W., Xiao, X., Xu, C., Xu, H., Lee, K., Chng, E.S., Li, H., 2016. Neural networks based channel compensation for i-vector speaker verification, in: 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP). IEEE, Tianjin, China, pp. 1–5. doi:10.1109/ISCSLP.2016.7918436

Reynolds, D.A., 2003. Channel robust speaker verification via feature mapping, in: 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). IEEE, p. II-53-6. doi:10.1109/ICASSP.2003.1202292

Reynolds, D.A., 1997. Comparison of Background Normalization Methods for Text-Independent Speaker Verification, in: European Conference on Speech Communication and Technology. Rhodes, Greece.

Reynolds, D.A., 1995. Speaker identification and verification using Gaussian mixture speaker models. Speech Commun. 17, 91–108. doi:10.1016/0167-6393(95)00009-D

Reynolds, D.A., Quatieri, T.F., Dunn, R.B., 2000. Speaker Verification Using Adapted Gaussian Mixture Models. Digit. Signal Process. 10, 19–41. doi:10.1006/dspr.1999.0361

Reynolds, D.A., Rose, R.C., 1995. Robust text-independent speaker identification using Gaussian mixture speaker models. IEEE Trans. Speech Audio Process. 3, 72–83. doi:10.1109/89.365379

Reynolds, D., Andrews, W., Campbell, J., Navratil, J., Peskin, B., Adami, A., Qin Jin, Klusacek, D., Abramson, J., Mihaescu, R., Godfrey, J., Jones, D., Bing Xiang, 2003. The SuperSID project: exploiting high-level information for high-accuracy speaker recognition, in: 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). IEEE, Hong-Kong, China, p. IV-784-7. doi:10.1109/ICASSP.2003.1202760

Reza, S., Azadi, T.E., Kabudian, J., Shekofteh, Y., 2014. A robust speaker recognition system combining factor analysis techniques, in: 2014 21th Iranian Conference on Biomedical Engineering (ICBME). IEEE, Tehran, Iran, pp. 343–347. doi:10.1109/ICBME.2014.7043948

Ribas González, D., Calvo de Lara, J.R., 2014. Feature classification criterion for missing features mask estimation in robust speaker recognition. Signal, Image Video Process. 8, 365–375. doi:10.1007/s11760-012-0299-z

Richardson, F., Reynolds, D., Dehak, N., 2015. Deep Neural Network Approaches to Speaker and Language Recognition. IEEE Signal Process. Lett. 22, 1671–1675. doi:10.1109/LSP.2015.2420092

Rose, R.C., Reynolds, D.A., 1990. Text independent speaker identification using automatic acoustic segmentation, in: International Conference on Acoustics, Speech, and Signal Processing. IEEE, pp. 293–296. doi:10.1109/ICASSP.1990.115638

Rosenberg, A.E., Parthasarathy, S., 1996. Speaker background models for connected digit password speaker verification, in: 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings. IEEE, Atlanta, GA, USA, pp. 81–84. doi:10.1109/ICASSP.1996.540295

Roy, A., Magimai.-Doss, M., Marcel, S., 2012. A Fast Parts-Based Approach to Speaker Verification Using Boosted Slice Classifiers. IEEE Trans. Inf. Forensics Secur. 7, 241–254. doi:10.1109/TIFS.2011.2166387

Sadjadi, S.O., Hansen, J.H.L., 2015. Mean Hilbert envelope coefficients (MHEC) for robust speaker and language identification. Speech Commun. 72, 138–148. doi:10.1016/j.specom.2015.04.005

Sadjadi, S.O., Hansen, J.H.L., 2014. Blind Spectral Weighting for Robust Speaker Identification under Reverberation Mismatch. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 937–945. doi:10.1109/TASLP.2014.2311329

Sadjadi, S.O., Hansen, J.H.L., 2013. Unsupervised Speech Activity Detection Using Voicing Measures and Perceptual Spectral Flux. IEEE Signal Process. Lett. 20, 197–200. doi:10.1109/LSP.2013.2237903

Sadjadi, S.O., Pelecanos, J., Zhu, W., 2014. Nearest Neighbor Discriminant Analysis for Robust Speaker Recognition, in: INTERSPEECH 2014. Singapore, pp. 1860–1864.

Saeidi, R., Alku, P., Backstrom, T., 2016. Feature Extraction Using Power-Law Adjusted Linear Prediction With Application to Speaker Recognition Under Severe Vocal Effort Mismatch. IEEE/ACM Trans. Audio, Speech, Lang. Process. 24, 42–53. doi:10.1109/TASLP.2015.2493366

Saeidi, R., Kinnunen, T., Sadegh Mohammadi, H.R., Rodman, R., Franti, P., 2010. Joint frame and Gaussian selection for text independent speaker verification, in: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, pp. 4530–4533. doi:10.1109/ICASSP.2010.5495576

Sahidullah, M., Saha, G., 2012a. Design, analysis and experimental evaluation of block based transformation in MFCC computation for speaker recognition. Speech Commun. 54, 543–565. doi:10.1016/j.specom.2011.11.004

Sahidullah, M., Saha, G., 2012b. Comparison of Speech Activity Detection Techniques for Speaker Recognition.

Sakoe, H., Chiba, S., 1978. Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoust. 26, 43–49. doi:10.1109/TASSP.1978.1163055

Sant'Ana, R., Coelho, R., Alcaim, A., 2006. Text-independent speaker recognition based on the Hurst parameter and the multidimensional fractional Brownian motion model. IEEE Trans. Audio, Speech Lang. Process. 14, 931–940. doi:10.1109/TSA.2005.858054

Sarkar, A.K., Do, C.-T., Le, V.-B., Barras, C., 2014. Combination of Cepstral and Phonetically Discriminative Features for Speaker Verification. IEEE Signal Process. Lett. 21, 1040–1044. doi:10.1109/LSP.2014.2323432

Sarkar, S., Sreenivasa Rao, K., 2014. Stochastic feature compensation methods for speaker verification in noisy environments. Appl. Soft Comput. 19, 198–214. doi:10.1016/j.asoc.2014.02.016

Sarma, M., Sarma, K.K., 2013. Vowel Phoneme Segmentation for Speaker Identification Using an ANN-Based Framework. J. Intell. Syst. 22, 111–130. doi:10.1515/jisys-2012-0050

Shriberg, E., Ferrer, L., Kajarekar, S., Venkataraman, A., Stolcke, A., 2005. Modeling prosodic feature sequences for speaker recognition. Speech Commun. 46, 455–472. doi:10.1016/j.specom.2005.02.018

Singhai, J., Singhai, R., 2007. Automatic Speaker Recognition: An Approach using DWT based Feature Extraction and Vector Quantization. IETE Tech. Rev. 24, 395–402.

Snyder, D., Ghahremani, P., Povey, D., Garcia-Romero, D., Carmiel, Y., Khudanpur, S., 2016. Deep neural network-based speaker embeddings for end-to-end speaker verification, in: 2016 IEEE Spoken Language Technology Workshop (SLT). IEEE, San Diego, CA, USA, pp. 165–170. doi:10.1109/SLT.2016.7846260

Soong, F., Rosenberg, A., Rabiner, L., Juang, B., 1985. A vector quantization approach to speaker recognition, in: ICASSP ’85. IEEE International Conference on Acoustics, Speech, and Signal Processing. Institute of Electrical and Electronics Engineers, pp. 387–390. doi:10.1109/ICASSP.1985.1168412

Spriet, A., Moonen, M., Wouters, J., 2004. Spatially pre-processed speech distortion weighted multi-channel Wiener filtering for noise reduction. Signal Processing 84, 2367–2387. doi:10.1016/j.sigpro.2004.07.028

Squartini, S., Principi, E., Rotili, R., Piazza, F., 2012. Environmental robust speech and speaker recognition through multi-channel histogram equalization. Neurocomputing 78, 111–120. doi:10.1016/j.neucom.2011.05.035

Srinivas, V., Santhirani, C., Madhu, T., 2014. Neural Network based Classification for Speaker Identification. Int. J. Signal Process. Image Process. Pattern Recognit. 7, 109–120. doi:10.14257/ijsip.2014.7.1.11

Sun, C., Zhu, Q., Wan, M., 2014. A novel speech enhancement method based on constrained low-rank and sparse matrix decomposition. Speech Commun. 60, 44–55. doi:10.1016/j.specom.2014.03.002

Sun, H., Ma, B., Khine, S.Z.K., Li, H., 2010. Speaker diarization system for RT07 and RT09 meeting room audio, in: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Dallas, TX, USA, pp. 4982–4985. doi:10.1109/ICASSP.2010.5495077

Taghia, J., Taghia, J., Mohammadiha, N., Sang, J., Bouse, V., Martin, R., 2011. An evaluation of noise power spectral density estimation algorithms in adverse acoustic environments, in: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Prague, Czech Republic, pp. 4640–4643. doi:10.1109/ICASSP.2011.5947389

Teunen, R., Shahshahani, B., Heck, L., 2000. A Model-Based Transformational Approach To Robust Speaker Recognition, in: 6th International Conference on Spoken Language Processing (ICSLP). Beijing, China.

Thomas, S., Ganapathy, S., Saon, G., Soltau, H., 2014. Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Florence, Italy, pp. 2519–2523. doi:10.1109/ICASSP.2014.6854054

Tingting Liu, Kai Kang, Shengxiao Guan, 2014. I-vector based text-independent speaker identification, in: Proceeding of the 11th World Congress on Intelligent Control and Automation. IEEE, pp. 5420–5425. doi:10.1109/WCICA.2014.7053640

Tirumala, S.S., Shahamiri, S.R., Garhwal, A.S., Wang, R., 2017. Speaker identification features extraction methods: A systematic review. Expert Syst. Appl. 90, 250–271. doi:10.1016/j.eswa.2017.08.015

Tong, S., Gu, H., Yu, K., 2016. A comparative study of robustness of deep learning approaches for VAD, in: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Shanghai, China, pp. 5695–5699. doi:10.1109/ICASSP.2016.7472768

Tufekci, Z., Gowdy, J.N., 2001. Subband feature extraction using lapped orthogonal transform for speech recognition, in: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221). IEEE, Salt Lake City, USA, pp. 149–152. doi:10.1109/ICASSP.2001.940789

Tufekci, Z., Gowdy, J.N., Gurbuz, S., Patterson, E., 2006. Applied mel-frequency discrete wavelet coefficients and parallel model compensation for noise-robust speech recognition. Speech Commun. 48, 1294–1307. doi:10.1016/j.specom.2006.06.006

Turner, C., Joseph, A., Aksu, M., Langdon, H., 2011. The wavelet and fourier transforms in feature extraction for text-dependent, filterbank-based speaker recognition. Procedia Comput. Sci. 6, 124–129. doi:10.1016/j.procs.2011.08.024

Varga, A., Steeneken, H.J.M., 1993. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems. Speech Commun. 12, 247–251. doi:10.1016/0167-6393(93)90095-3

Variani, E., Lei, X., McDermott, E., Moreno, I.L., Gonzalez-Dominguez, J., 2014. Deep neural networks for small footprint text-dependent speaker verification, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Florence, Italy, pp. 4052–4056. doi:10.1109/ICASSP.2014.6854363

Vazquez-Machado, C., Colon-Hernandez, P., Torres-Carrasquillo, P.A., 2016. I-vector speaker and language recognition system on Android, in: 2016 IEEE High Performance Extreme Computing Conference (HPEC). IEEE, pp. 1–6. doi:10.1109/HPEC.2016.7761630

Venturini, A., Zao, L., Coelho, R., 2014. On speech features fusion, alpha-integration Gaussian modeling and multi-style training for noise robust speaker classification. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 1951–1964. doi:10.1109/TASLP.2014.2355821

Verma, P., Das, P.K., 2015. i-Vectors in speech processing applications: a survey. Int. J. Speech Technol. 18, 529–546. doi:10.1007/s10772-015-9295-3

Vuppala, A.K., Rao, K.S., 2013. Speaker identification under background noise using features extracted from steady vowel regions. Int. J. Adapt. Control Signal Process. 27, 781–792. doi:10.1002/acs.2357

Wan, V., Renals, S., 2005. Speaker verification using sequence discriminant support vector machines. IEEE Trans. Speech Audio Process. 13, 203–210. doi:10.1109/TSA.2004.841042

Wang, D., Brown, G.J., 2006. Computational Auditory Scene Analysis: Principles, Algorithms, and Applications. Wiley-IEEE Press.

Wang, S., Li, K., Huang, Z., Siniscalchi, S.M., Lee, C.-H., 2017. A transfer learning and progressive stacking approach to reducing deep model sizes with an application to speech enhancement, in: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, New Orleans, LA, USA, pp. 5575–5579. doi:10.1109/ICASSP.2017.7953223

Wolf, J.J., 1972. Efficient Acoustic Parameters for Speaker Recognition. J. Acoust. Soc. Am. 51, 2044–2056. doi:10.1121/1.1913065

Woo, K.-H., Yang, T.-Y., Park, K.-J., Lee, C., 2000. Robust voice activity detection algorithm for estimating noise spectrum. Electron. Lett. 36, 180. doi:10.1049/el:20000192

Wu, G.-D., Lin, C.-T., 2000. Word boundary detection with mel-scale frequency bank in noisy environment. IEEE Trans. Speech Audio Process. 8, 541–554. doi:10.1109/89.861373

Xia, B., Bao, C., 2014. Wiener filtering based speech enhancement with Weighted Denoising Auto-encoder and noise classification. Speech Commun. 60, 13–29. doi:10.1016/j.specom.2014.02.001

Xiang, B., Chaudhari, U. V., Navratil, J., Ramaswamy, G.N., Gopinath, R.A., 2002. Short-time Gaussianization for robust speaker verification, in: IEEE International Conference on Acoustics Speech and Signal Processing. IEEE, p. I-681-I-684. doi:10.1109/ICASSP.2002.5743809

Xiong, Z., Zheng, T.F., Song, Z., Soong, F., Wu, W., 2006. A tree-based kernel selection approach to efficient Gaussian mixture model–universal background model based speaker identification. Speech Commun. 48, 1273–1282. doi:10.1016/j.specom.2006.06.011

Xu, Y., Du, J., Dai, L.-R., Lee, C.-H., 2014. An Experimental Study on Speech Enhancement Based on Deep Neural Networks. IEEE Signal Process. Lett. 21, 65–68. doi:10.1109/LSP.2013.2291240

Yan, F., Zhang, Y., Yan, J., 2014. A sub-band-based feature reconstruction approach for robust speaker recognition. EURASIP J. Audio, Speech, Music Process. 2014, 1–13. doi:10.1186/s13636-014-0040-7

Yegnanarayana, B., Sharat Reddy, K., Kishore, S.P., 2001. Source and system features for speaker recognition using AANN models, in: 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221). IEEE, Salt Lake City, USA, pp. 409–412. doi:10.1109/ICASSP.2001.940854

Yiming, S., Rui, W., 2015. Voice Activity Detection Based on the Improved Dual-Threshold Method, in: 2015 International Conference on Intelligent Transportation, Big Data and Smart City. IEEE, Halong Bay, Vietnam, pp. 996–999. doi:10.1109/ICITBS.2015.252

Ying, D., Yan, Y., Dang, J., Soong, F.K., 2011. Voice activity detection based on an unsupervised learning framework. IEEE Trans. Audio, Speech Lang. Process. 19, 2624–2632. doi:10.1109/TASL.2011.2125953

Yong Duk Cho, Kondoz, A., 2001. Analysis and improvement of a statistical model-based voice activity detector. IEEE Signal Process. Lett. 8, 276–278. doi:10.1109/97.957270

Young, S., Kershaw, D., Odell, J., Ollason, D., Valtchev, V., Woodland, P., 2000. The HTK Book Version 3.0. Cambridge University Press.

Zen, H., Nankaku, Y., Tokuda, K., 2009. Stereo-based stochastic noise compensation based on trajectory GMMS, in: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, Taipei, Taiwan, pp. 4577–4580. doi:10.1109/ICASSP.2009.4960649

Zhang, X.-L., Wu, J., 2013. Deep Belief Networks Based Voice Activity Detection. IEEE Trans. Audio. Speech. Lang. Processing 21, 697–710. doi:10.1109/TASL.2012.2229986

Zhang, Z., Jia, Y., Xie, S., Zhang, M., 2009. Speaker Recognition Based on Support Vector Machines and Multi-Scale Wavelet Analysis, in: 2009 International Symposium on Computer Network and Multimedia Technology. IEEE, Wuhan, China, pp. 1–4. doi:10.1109/CNMT.2009.5374719

Zhao, X., Shao, Y., Wang, D., 2012. CASA-Based Robust Speaker Identification. IEEE Trans. Audio. Speech. Lang. Processing 20, 1608–1616. doi:10.1109/TASL.2012.2186803

Zhao, X., Wang, Y., Wang, D., 2015. Cochannel Speaker Identification in Anechoic and Reverberant Conditions. IEEE/ACM Trans. Audio, Speech, Lang. Process. 23, 1727–1736. doi:10.1109/TASLP.2015.2447284

Zhao, X., Wang, Y., Wang, D., 2014. Robust Speaker Identification in Noisy and Reverberant Conditions. IEEE/ACM Trans. Audio, Speech, Lang. Process. 22, 836–845. doi:10.1109/TASLP.2014.2308398

Zhu, D., Ma, B., Li, H., 2011. Speaker Verification With Feature-Space MAPLR Parameters. IEEE Trans. Audio. Speech. Lang. Processing 19, 505–515. doi:10.1109/TASL.2010.2051269

Zhu, W., Sadjadi, S.O., Pelecanos, J.W., 2015. Nearest neighbor based i-vector normalization for robust speaker recognition under unseen channel conditions, in: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, Brisbane, QLD, Australia, pp. 4684–4688. doi:10.1109/ICASSP.2015.7178859

Zilca, R.D., 2002. Text-independent speaker verification using utterance level scoring and covariance modeling. IEEE Trans. Speech Audio Process. 10, 363–370. doi:10.1109/TSA.2002.803419

Dişken, G., Tüfekci, Z., Çevik, U., 2017a. Speaker Model Clustering to Construct Background Models for Speaker Verification. Arch. Acoust. 42, 127–135. doi:10.1515/aoa-2017-0014

Dişken, G., Tüfekci, Z., Çevik, U., 2017b. A robust polynomial regression-based voice activity detector for speaker verification. EURASIP J. Audio, Speech, Music Process. 2017, 23. doi:10.1186/s13636-017-0120-6

Dişken, G., Tüfekçi, Z., Saribulut, L., Çevik, U., 2017c. A Review on Feature Extraction for Speaker Recognition under Degraded Conditions. IETE Tech. Rev. 34, 321–332. doi:10.1080/02564602.2016.1185976

BIOGRAPHY

Gökay DİŞKEN was born in Antalya, Turkey, in 1990. He received his B.S. and M.Sc. degrees in Electronics and Communication Engineering Department from Süleyman Demirel University in 2011, and 2014, respectively. He has been working as a research assistant at Adana Science and Technology University since November, 2013. His research interests include signal processing, speaker recognition, and speech recognition.