

**VULNERABILITY OF SPEAKER VERIFICATION
SYSTEM UNDER REPLAY SPOOFING ATTACK AND
INVESTIGATION OF SYNTHETIC SPEECH
DETECTION**

A Thesis

by

Mustafa Caner Özbay

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Electrical and Electronics Engineering

Özyeğin University
May 2016

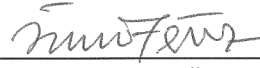
Copyright © 2016 by Mustafa Caner Özbay

**VULNERABILITY OF SPEAKER VERIFICATION
SYSTEM UNDER REPLAY SPOOFING ATTACK AND
INVESTIGATION OF SYNTHETIC SPEECH
DETECTION**

Approved by:



Assistant Professor Cenk Demiroğlu,
Advisor
Department of Electrical and Electronics
Engineering
Özyeğin University

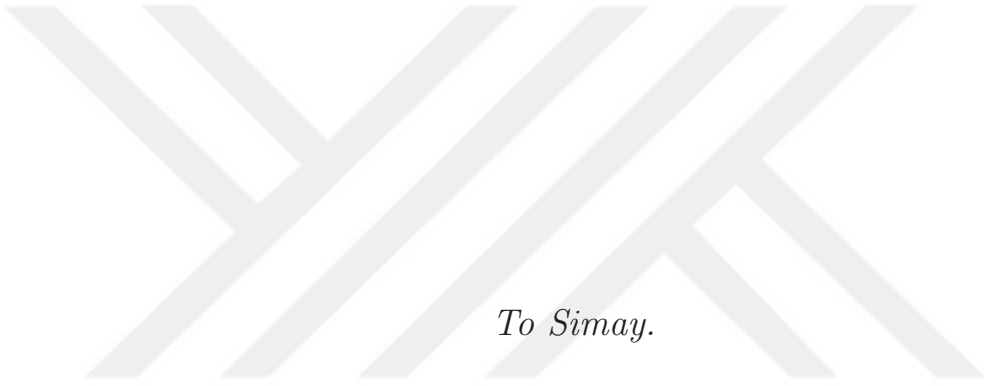


Assoc. Professor Ümit Güz
Department of Electrical and Electronics
Engineering
Işık University



Assoc. Professor Hakan Gürkan
Department of Electrical and Electronics
Engineering
Işık University

Date Approved: 31 May 2016



To Simay.

ABSTRACT

Security issues have become major subject along with developments in the speaker verification system. Previous studies have shown that performance is still not fully proven for vulnerability. Simple and easy methods may be preferred to attack however the verification system can be attacked with high-tech infrastructure. The first example of this method is "Replay Attack". In this thesis, firstly we downloaded data from the Internet in ways that everyone can easily success. Then the data has enrolled into the system that was previously created as universal background model (UBM). Operability of the system have been proved by tests with same people voice data and then system was exposed to attack again with most similar speech that obtained from other speakers. The results also showed that, even it is not high rated but spoofing attack may cause false acceptance by the verification system. After that, noise added to simulate effect of electronic devices and that added noise was found to increase the system performance. With this work it is aimed to point the open field in the anti-spoofing area against the replay attack. Then the specification of synthetic speech system and the differences according to natural sound is investigated. How it is used as spoofing attack, and the results are also represented.

ÖZETÇE

Konuşmacı doğrulama sistemlerinde ki gelişmeler beraberinde olayın güvenlik boyutunu da gündeme getirdi. Önceki çalışmalar gösterdi ki, güvenlik konusunda hala performans tam olarak kanıtlanamadı. Yüksek teknolojik altyapılar ile doğrulama sistemlerine saldırı olabileceği gibi basit ve kolay yöntemler de tercih edilebilir. Bunun ilk örneği "Yeniden Oynatma" yöntemidir. Bu tezde biz ilk olarak herkesin kolaylıkla elde edebileceği yollarla internetten veri indirdik. Daha sonra bu veriler önceden oluşturulan evrensel ses modeli (UBM) sistemine tanıtıldı. Sistemin çalışırılığı aynı kişiler ile test edilerek kanıtlandı ve sonrasında elde edilen diğer en benzer konuşmalar ile sistem tekrar saldırıya maruz bırakıldı. Burada ki sonuç gösterdi ki saldırı da yüksek oran olmasa bile sistemi yanıltan denemeler olabiliyor. Daha sonra kullanılacak elektronik araçların etkisini simüle edebilmek için gürültü eklendi ve eklenen gürültünün sistem performansını arttırdığı görüldü. Bu çalışma ile yeniden oynatma alanında yanıltmaya karşı koruma tekniği alanında ki açık gösterilmeye çalışıldı. Daha sonra sentetik ses sentezleme sisteminin özellikleri ve doğal ses ile farkları incelendi. Yanıltıcı saldırılarda ne şekilde kullanıldığı ve sonuçları da paylaşıldı.

ACKNOWLEDGEMENTS

I would like to thank my advisor Cenk Demirođlu for his guidance, support and patience throughout my master study. Also, I would like to thank Özyeđin University Speech Processing Laboratory members Ekrem Güner, Fatih Yeşil, Osman Soyyiđit and Ali Khodabakhsh. I would like to thank Vestel Electronics during this thesis I worked for, too. I would also like to give my special thanks to my wife for continuous ambition she provided me.

TABLE OF CONTENTS

DEDICATION	iii
ABSTRACT	iv
ÖZETÇE	v
ACKNOWLEDGEMENTS	vi
LIST OF TABLES	ix
LIST OF FIGURES	x
GLOSSARY	1
I INTRODUCTION	1
1.1 Overview of a Speaker Verification System	1
1.2 Performance Measures	4
1.3 Spoofing	5
1.3.1 Impersonation	6
1.3.2 Speech Synthesis	6
1.3.3 Voice Conversion	6
1.3.4 Replay	7
1.4 Anti-Spoofing	7
1.5 Outline of This Thesis	8
II PREVIOUS WORK	9
2.1 Literature Review	9
2.1.1 Spoofing	9
2.1.2 Anti-Spoofing	11
2.2 Speaker Verification System	12
2.2.1 Universal Background Model	12
2.2.2 Total Variability Space	14
2.2.3 Channel Compensation Algorithms	17

III REPLAY ATTACK	19
3.1 Impostor Data Collection	20
3.2 Scoring Algorithm	20
3.2.1 Cosine Distance Scoring	20
3.3 Spoofing Attack	21
3.3.1 Noise Added Attack	21
IV SYNTHETIC SPEECH DETECTION	23
4.1 Frame and Segment Specific Importance Weighting	24
4.2 Synthetic Speech Detectors	25
V EXPERIMENTS	26
5.1 Replay Spoofing Attack Experiments	26
5.1.1 Effect of Noise on Speaker Verification System	29
5.2 Synthetic Speech Detection	30
5.3 Discussion	43
VI CONCLUSION	45
BIBLIOGRAPHY	48
VITA	54

LIST OF TABLES

1	Comparison of all used spoofing attack types according to spoken by whom, effort and effect on system.	10
2	Distributon of Speakers and Contents	27
3	Effect of Noise to Spoofing Attack EER	30
4	EER Of The Voice Verification System For Different Noise Types and SNRs.	37
5	Perfomance of the Baseline and Proposed Detectors in Termsof EERs for the Development and Evaluation Data. Results are Presented with and without Duration-Weighting. S1, S2, and S5 Systems Use Voice Conversion (VC). S3 and S4 Systems Use HMM-Based Synthesis. Best Performing Algorithm For Each Attack Type is Shown in Bold	42

LIST OF FIGURES

1	A typical speaker verification system.	3
2	FAR, FRR and EER relationship.	5
3	A flowchart of acoustic model of textdependent speaker verification system.	10
4	Flowchart of Replay Spoofing Attack	19
5	Effect of Gaussian Noise on MFCC features.	22
6	Overview of the proposed synthetic speech detectors.	25
7	Baseline DET Curve	28
8	Attack DET Curve	29
9	Attack DET Curve with 5dB Gaussian Noise	31
10	Attack DET Curve with 10dB Gaussian Noise	32
11	Attack DET Curve with 15dB Gaussian Noise	33
12	Attack DET Curve with 20dB Gaussian Noise	34
13	Clean Synthetic and Natural Data	35
14	Noisy Natural and Synthetic Data (Matched Condition)	35
15	Noisy Natural and Synthetic Data (Mismatched Condition)	36
16	Verification False Alarm Rates Under Attack With Synthetic Speech .	38
17	Detector Performance for Mismatched Case	39
18	Detection performance of each Gaussian component versus its logarithm of number of occurrence in the development utterances is shown.	43
19	Detection performance of each phoneme versus its logarithm of number of occurrence in the development utterances is shown. Phonemes that are in the same sound-class are shown with the same color and shape.	44

CHAPTER I

INTRODUCTION

The latest point of current technological improvement and the requirements which has to be satisfied, has increased the work and actions about identity verification. These identity verification can consist all personal features like fingerprints without glove, signature, eye scan and voice. Voice is the main used human biological feature in this thesis. In speech processing domain, speaker recognition is the identification of a speaker by features of speaker's voice. Recognition of a speaker using speech data and features which are obtained from that speech data, can be separated in two groups of research topic which are speaker identification and speaker verification. First title of this group is speaker identification problem in which there is a pool of target speakers and the system tries to determine the identity of the trial speaker by matching the most possible one from the speaker set. On the other hand, speaker verification is the decision process of whether a trial speaker it matches the claimed id or not.

1.1 Overview of a Speaker Verification System

For a short section of spoken voice data, Y , and a hypothesized speaker, S , the duty of speaker verification, is to decide if Y was spoken by S . In the speaker detection task, a simple and mostly used hypothesis test can be used to restate between;

H_0 : Y was spoken by the hypothesized speaker S

and

H_1 : Y was not spoken the hypothesized speaker S .

Likelihood ratio test which is optimum one, can be used to determine between these two hypotheses and it is given by;

$$\frac{p(Y|H_0)}{p(Y|H_1)} \begin{cases} \geq \theta & \text{accept } H_0 \\ < \theta & \text{reject } H_0 \end{cases} \quad (1)$$

in this formula the probability density function (pdf), which is $p(Y|H_i)$, for the assumption of H_i , $i = 0, 1$, is examined for the used speech data section Y . Also θ is used as a threshold level to make a decision whether Y is spoken by hypothesized speaker or not. The main aim of a speaker verification system is to decide about methods to verify data and study on the results of this formula for the two pdfs, $p(Y|H_0)$ and $p(Y|H_1)$ [1].

Fig. [1] illustrate how a typical speaker verification system operates . The speech pool of probability density the function of $p(Y|H_1)$ is a speech model which are spoken by sufficiently amount of speaker. This large speaker pool is called as the Universal Background Model (UBM) (detailed analysis is given in section 2.2.1). The speaker data expressed as $p(Y|H_0)$ is data which is obtained from hypothesized speaker and enrolled to the system. $p(Y|H_0)$ is not a speech pool as in $p(Y|H_1)$, because only a short segment of speech is enough for enrollment part. Finally, decision is made by using threshold θ for the recorded speech from hypothesized speaker which has a identity label from verification system [2].

In literature, two different types of ASV systems which are text-independent speaker verification and text-dependent speaker verification systems, exist. Text-dependent systems based on concerted speakers and requires the speaker to speak pre-defined and enrolled or instantly determined sentences, while text-independent systems is more practical because the speaker can speak any sentences to the system during both enrolment and verification. But unfortunately text-independent system needs more training and testing utterances than a text-dependent system because

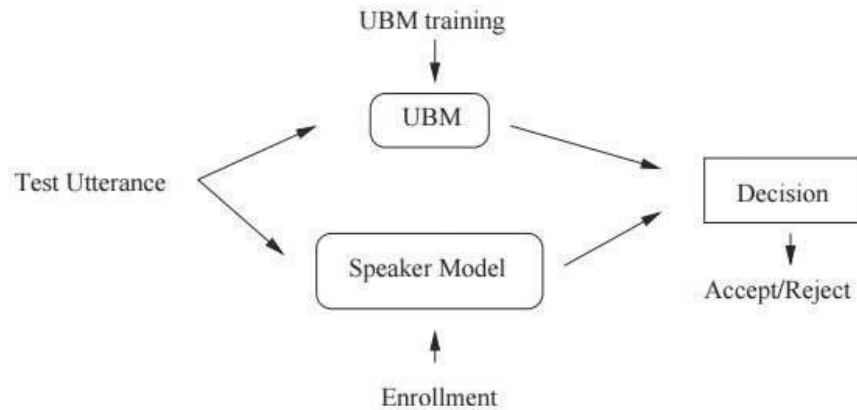


Figure 1: A typical speaker verification system.

expressive features are not existing.

The individuality of speakers can be characterized as three different level of features which are high level, spectro-temporal and short-term features.

- High level features are robust against noise but to extract that features require more effort than others. Phoneme, accent and pronunciation are example of high level features. So, automatic speech recognition systems require high level features extraction. According to sensitivity to channel and noise effects High-level features have more advantage than spectral and prosodic features. But as a disadvantage of high level features, the features extraction requires a far amount more complex front-ends, as in automatic speech recognition systems.
- Spectro-temporal features involve prosodic, temporal modulation features, etc. Longer segments are required to extract the prosodic features such as syllables and word-like units to characterise intonation and speaking style. Prosodic features, such as pitch, energy and duration, are more strong to the effects of channel.
- Short-term features requires shorter frames which is 20-30 milliseconds, to be

extracted. This 20-30 milliseconds window is moved on speech data while in extraction process. They describe the short-term spectral envelope which is an acoustic correlate of voice timbre. Mel-frequency cepstral coefficient (MFCC), linear predictive cepstral coefficient (LPCC), and perceptual linear prediction (PLP) are the most popular short-term features.

Short-term spectral features are usually preferred features in speaker verification area. They are also called as low-level features too [3].

1.2 Performance Measures

In single-speaker detection scenario there are two possible output that may occur: if the tested speech data is spoken by the same speaker as hypothesised speaker which means they have same identity, then tested speaker will be labelled as client speaker. As a second possibility, if tested speaker speaks like a pre-used client speaker identity to spoof the system by claiming then tested speaker will be labelled as a non-client speaker. False acceptance (FA) is the rate of acceptance of an attempt from a non-client or impostor speaker by the verification system. False rejection (FR) is the rate of rejection of an attempt from a true speaker. The relationship between false rejection rate (FRR) and false acceptance rate (FAR) specifies by the threshold η . This trade-off can be figured using Detection Error Trade-off (DET) curve so that performance and calibration of the system can be visually analyzed [4].

Equal error rate (EER) is the most commonly used performance measurement methods. EER of a verification or detection system can be set by adjusting η until $FAR=FRR=EER$. The relationship between FAR, FRR and EER can be easily seen at the below Fig. [2].

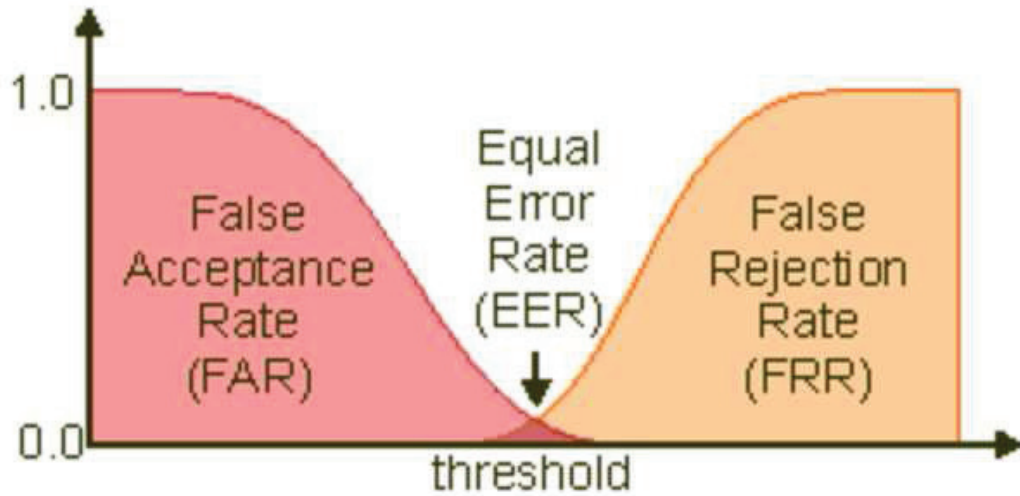


Figure 2: FAR, FRR and EER relationship.

1.3 Spoofing

Offline enrolment and runtime verification are two different steps in a typical Automatic Speaker Verification (ASV) system. Features are extracted from a set of speech sample. Then they are used to train a target speaker model during the offline enrolment. Target model is used to make a decision to either accept or reject the identity claim during the runtime verification step. That decision module occurs the relative scoring which is usually log-likelihood ratio. If spoofing attack types need to be categorised which are [5]:

Indirect attacks, which is not a subject of this thesis but it demands the control of some system parts. For example an attack at feature extraction, modelling, decision part or scoring logic part is called as indirect attack.

Direct attacks; should be applied as a pre-processing of indirect attack because it can be applied at microphone or transmission points. that's why it is also called as spoofing attacks and direct attack types are as follows:

1.3.1 Impersonation

Impersonation is the easiest spoofing attack type to apply and it does not require any technological knowledge. It just requires human-altered voices.

1.3.2 Speech Synthesis

Speech can be synthesized by using text-to-speech (TTS) method. Speech synthesis should satisfy easy to understand and native human sounded artificial speech specifications. Front-end component of speech synthesis process is text analysis in which phonemes and other linguistic components are extracted from converted input text. Back-end component of speech synthesis is speech waveform generation component in which waveform of speech is generated as output from converted linguistic input text [6]. Here, it is investigated the effectiveness of spoofing attacks with statistical speech synthesis systems using limited amount of adaptation data and additive noise. Experiment results show that effective spoofing is possible using limited adaptation data.

1.3.3 Voice Conversion

Voice conversion is a computer transfer function that converts an input speech signal to another sounded one. The aim of this conversion is that converted speech looks like of target speaker's voice. Input speech signal of a voice conversion system should be a natural sound unlike speech synthesis systems always need a text as an input. A voice conversion system include two different conversion block which are spectral mapping and prosody conversion. Prosody conversion change spectro-temporal features which are pitch, fundamental frequency, energy and duration. On the other hand spectral mapping relates to voice timbre [7].

1.3.4 Replay

Replay attack is a spoofing attack form to an ASV system by using a ready to play recorded speech which is spoken by target speaker which is genuine. Pre-recorded set of sample speech can be obtained from various ways such as recording device, arranged samples which are put together from different short speech segments [8], or can be downloaded via internet as in this thesis.

There are also two kind of replay spoofing attack according to used data. One is using the data which is obtained from hypothesized speaker, and the other one is using the data which is similar to the hypothesized speaker. In section 3.2 it is explained how we measure the similarity score.

1.4 *Anti-Spoofing*

Speaker verification has started to become a major application because identity verification needed applications takes up considerable space in our lives such as bank call center. Call center verification system based on using voice signature as vocal password. Earning the trust of customers to this kind of systems, it should be vulnerable to spoofing attacks in real life scenarios. For text-dependent speaker verification systems, building a vulnerable and safe detection function is easier than text-independent one against spoofing attacks. In this thesis, it has been aimed to show how a speaker verification system is weak against replay spoofing attack. Also as countermeasure to synthesized speech, two methods can be proposed. In one approach, distributions of Gaussian components are used to detect repetitions of Gaussians in speech. In a second approach, automatic voice quality assessment tools are used to detect synthetic speech [9].

1.5 Outline of This Thesis

The rest of this thesis is organized as follows. An overview of previous works on this topic is presented in chapter 2, along with a brief background on the technologies used. Chapter 3 and chapter 4 explains proposed algorithms. Chapter 5 presents experiments and discusses the results obtained from each. Finally chapter 6 concludes this thesis and discusses future works.



CHAPTER II

PREVIOUS WORK

2.1 Literature Review

Some of the prior methods for spoofing the SV systems and detection of spoofing attacks are described below.

2.1.1 Spoofing

In the literature, four different spoofing attack attempt [10], [11] can be listed as impersonation, speech synthesis, voice conversion and replay. Impersonation is the easiest spoofing attack approach and it only requires human-altered voices. Impersonation is mostly investigated in past researches. Vulnerability of this system has been studied in [12], [13], [14], [15]. Speech synthesis is synthesising a genuine speaker's speech with a given text by using a speech synthesis system to spoof ASV systems and this has been studied in [16], [17]. Voice conversion is a conversion function based spoofing attack attempt to manipulating a natural input voice to imitate the target speaker. Vulnerability and success ratio of voice conversion systems has been studied in [18], [19], [20], [21], [22], [23], [24]. The most easy to apply and low effort requirement attack type is replay attack which investigated in [25], [26], [27], [28], [29] to make a decision if speaker verification systems are vulnerable to spoofing attack or not. In this thesis, replay attack and synthesized speech attack are focused.

In [29] it is aimed to investigate the threat of replay attacks by using a large speech database to compare the vulnerability between replay spoofing attacks and other high-tech knowledge required spoofing attacks. Also there is comparison in Table [1] of all used spoofing attack types according to spoken by whom, effort and effect on system.

Attack	Naïve impostor	Replay	Voice conversion	Speech synthesis
Speech used	impostor's (genuine)	client's	impostor's (converted)	synthetic
Effort	zero	low	medium-high	high
Effectiveness	low	(?)	high	high

Table 1: Comparison of all used spoofing attack types according to spoken by whom, effort and effect on system.

The work in [12] showed that if non-professional impersonators have similar natural voice to voice of target speakers, then they can imitate their voice to spoof ASV systems. Previous works proved that impersonation spoofing attack increased FAR rates from about %0 to between %10 and %60, but there is no significant difference in vulnerability by the attack either non-professional or professional impersonators.

In [30] it is employed a flowchart of acoustic model of text dependent speaker verification system. In Fig. [3] this system is represented. Text-dependency is main difference between this work and my research. This model is builded of three layers and they can be listed as respectively: universal background model (UBM) is first step, text-dependent Gaussian mixture model (GMM) is second step, and text-dependent hidden Markov model (HMM) is latest step.

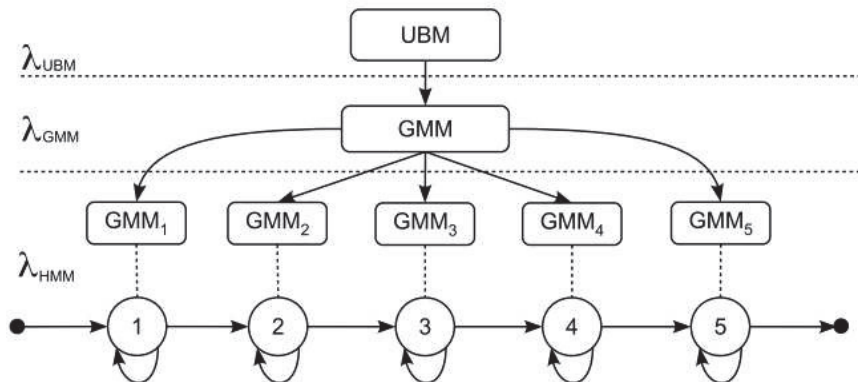


Figure 3: A flowchart of acoustic model of textdependent speaker verification system.

Also as a disadvantage of replay attack it is observed that some features are same with each other. Because of replay speech is spoken by genuine speaker, it probably has exactly same features (spectral attributes, prosodic and high-level features) with target speaker. It is also possible to have exactly indistinguishable spectrogram and formant tracks to target speech. If replay spectrogram is used to extract the features, then it will be possible to see a closer verification score to the hypothesized speaker. By following this method verification system has not got capability of detecting a replay spoofing attack because of indistinguishable features. So some anti-spoofing or detection algorithm for replay attack are proposed in literature.

2.1.2 Anti-Spoofing

There is a common output of all different work that is FARs clearly increase when replay attack applied. So, replay attack's anti-spoofing technique should be developed to increase the vulnerability against to the speaker verification systems. To differentiate a speech if it is a replay attack or genuine speakers speech, noise level can be used. In replay attack there are two device noises and one speaker noise but in licit recording there is only one recording device noise.

Because of it is possible for replay speech has same spectral attributes, prosodic and high level features as that of target genuine speaker, bitmap detection technique can be used to detect a spoofing attack. Two bitmaps are used to calculate the similarity by using spectral peaks. With this method detection score is being calculated and detection score is the inverse of similarity score [30].

In [31] the application of remote interaction via telephone has been proposed as playback attack detection (PAD) algorithm. The task of a PAD is to decide the incoming record matches with any previous stored recording or not. If it matches then replay attempt will be called as spoofing attack. The PAD algorithm occurs from three

different stages which are feature extraction, similarity measure, and attack/non-attack classification. A detection score has been measured to define similarity between input utterance and stored one. A playback attack determined is if an high similarity score has been assessed between input and any other pre-stored utterance.

In [28] channel pattern noise method is investigated by extracting from genuine and playback recordings for PAD. The main difference between records is base channel noise. Authentic recordings consist only channel noise of recording device of the speaker recognition system like telephone banking system, while the channel noise of playback recordings consist three different devices: the intruder recording device, the playback speaker, the recording device of the system. Different electronic devices that capable of recording and playback will result in various channel noise in speech signals (microphone, loudspeaker, pre-amplifier, power amplifier, input and output filters, A/D, D/A, sample and hold circuit will cause channel noise [32]). These channels which are from transducers and different circuits noise is called as channel pattern noise.

2.2 Speaker Verification System

In [33] a general description is given for Speaker Verification System. The main topics are Universal Background Model (UBM), Total Variability Space (TVS) and Channel Compensation Algorithms.

2.2.1 Universal Background Model

Gaussian mixture models (GMM) are mostly basic component of the State-of-the-art speaker verification systems. A whole speaker space representative background model which include all possible variabilities, is used within GMM approaches. The mathematical expression of background model is a trained GMM with lots of data by using acoustic features of speakers. This model is called Universal Background Model (UBM).

While training UBM, algorithm and data related parameters are used. Examples of algorithm parameters are proper mixture number, initialization method, training method, number of iterations and examples of data parameters are number of speakers, data amount per speaker, selection way of features, variability that can be captured.

2.2.1.1 Gaussian Mixture Model (GMM) Training

A Gaussian mixture model (GMM) is well-known approach of probability density function which represented as weighted sum of Gaussian component densities. Mean, variance and weights are main parameters of GMM where mean (μ) and variance (σ^2) are only occurs for one single Gaussian besides weights (ω) are belonged to the Gaussian mixture [34]. The probability distribution function (pdf) of a GMM is defined as below;

$$p(x) = \sum_{m=1}^M \omega_m N(x|\mu_m, \Sigma_m) \quad (2)$$

where the weight parameters w_m satisfy these two conditions: $0 \leq w_m \leq 1$ and $\sum_{m=1}^M w_m = 1$. Ideal GMM related parameters can be estimated with training data by defining likelihood function. To simplify the equation and to remove exponential terms logarithm of likelihood function is used instead of Eq. 2. Logarithmic function of GMM is described in Eq. 3.

$$\ln p(X|w, \mu, \Sigma) = \sum_{f=1}^F \ln \left(\sum_{m=1}^M w_m N(x_f|\mu_m, \Sigma_m) \right) \quad (3)$$

Expectation-maximization(EM) algorithm is used to estimate the maximizing optimum parameters which are the ratio of data likelihood to the model. EM algorithm is an iterative model and equation steps are;

1. As initialization step, log likelihood is computed and GMM parameters which are means μ_m , covariances Σ_m and mixture weights w_m are initialized.

2. In Expectation (**E**) step, GMM parameters which are found at first step, are used to calculate the responsibilities of each mixture on the generation of each sample

$$\gamma(z)_{fm} = \frac{w_m N(x_f | \mu_m, \Sigma_m)}{\sum_{j=1}^M w_j N(x_f | \mu_j, \Sigma_j)} \quad (4)$$

3. In Maximization (**M**) step, GMM parameters are aimed to be used in next iteration step for responsibilities evaluated above Eq. 4 by estimating and updating.

$$\mu_m^{new} = \frac{1}{F_m} \sum_{f=1}^F \gamma(z_{fm}) x_f \quad (5)$$

$$\Sigma_m^{new} = \frac{1}{F_m} \sum_{f=1}^F \gamma(z_{fm}) (x_f - \mu_m^{new})(x_f - \mu_m^{new})^T \quad (6)$$

$$w_m^{new} = \frac{F_m}{F} \quad (7)$$

where F_m , number of frames belonging to GMM component m is

$$F_m = \sum_{f=1}^F \gamma(z_{fm}) \quad (8)$$

4. The log likelihood function in Eq. 3 is computed and the condition of convergence is examined to make a decision whether to keep iterating starting from step 2 or to stop.

2.2.2 Total Variability Space

Extracting the i-vector for each speaker is the goal of a Total Variability Space (TVS) system. In data training term T matrix are trained. In [35] TVS modelling was introduced. In state of art speaker verification systems TVS modeling has been most commonly used model. Following formulation can be a summary of this model;

$$M_s = M_0 + T w_s \quad (9)$$

In Eq. 9 M_0 is a supervector which is independent from speaker. It projects UBM means. Also in above formula, T is the low rank total variability matrix and it is also

called as projection matrix or i-vector extractor. In addition w_s is a low-dimension i-vector.

Both of dimensionality reduction and channel compensation gathering can be provided with a TVS modelling.

2.2.2.1 Training of The T-Matrix

Extracting a low dimensional i-vector for each speaker is the main goal of TVS modelling. To obtain this low dimensional i-vector, a large database of speech which are spoken by many different speaker in many different session, should be used to train i-vector extractor or in another name T-matrix. However the eigenvoice modeling in [36] estimate as many eigenvoices from a given training set, in TVS modelling each of these sessions are assumed as a different speaker. This is the difference of training procedure between eigenvoice modeling and TVS modeling.

Considering Eq. 9, it can be said that for any speaker s , frames aligned with k^{th} mixture of UBM, are distributed with mean $M_k(s)$ and covariance matrix Σ_k . Σ denote the DKxDK block diagonal matrix whose block diagonals are $\Sigma_1, \dots, \Sigma_K$. In Eq. 10 all speakers who are in database, likelihood function is given.

$$\prod_{s=1}^S \max_w P(x(s) | M_0 + Tw, \Sigma) \quad (10)$$

In Eq. 10 $x(s)$ is the training data of speakers and s ranges in the all training set's utterances. In TVS modelling each of this utterances is assumed as spoken by a different speaker because of both speaker and channel variabilities are considered.

2.2.2.2 EM Algorithm

Maximum likelihood (ML) estimation problem in Eq. 10 has no closed form solution so the parameters are estimated using EM algorithm. The goal of EM algorithm is to compute the model parameters iteratively which has two steps;

1. Speaker supervector increase the ratio of likelihood for all training data and T and Σ can be used to find this speaker supervector. Training data $x(s)$ is shown in Eq. 11.

$$w(s) = \underset{w}{\operatorname{arg\,max}} P(x(s)|M_0 + Tw, \Sigma) \quad (11)$$

2. After miximization T and Σ will be updated.

$$\prod_{s=1}^S P(x(s)|M_0 + Tw_s, \Sigma) \quad (12)$$

In Expectation (E) step of EM algorithm, only calculation is in finding posterior distribution of $w(s)$ given the speakers training data mixture-components. Posterior distribution need to be calculated for all speakers by using current estimates of T and Σ . If it is first iteraton initial random values are used.

Each data frame should be labeled by using a mixture model. This process is called as alignment and training data $x(s)$ for each speaker need to be aligned with universal background model to calculate the distribution of posterior. Using the alignment informations $N_k(s)$ which is the number of frames of aligned with the k^{th} mixture and first and second order statistics $S_{X,k}(s)$ and $S_{XX^T,k}(s)$ are computed. Where $k= 1, \dots, K$ and computation is as follow;

$$S_{X,k}(s) = \sum_t (X_t - \mu_k) \quad (13)$$

$$S_{XX^T,k}(s) = \sum_t (X_t - \mu_k)(X_t - \mu_k)^T \quad (14)$$

where in summation speaker independent UBM's k^{th} mixture is used for the alignment of all frames X_t of speaker s . In addition k^{th} mixture component of UBM mean vector is shown as μ^k .

In Eq. 15 $N_1(s)I, \dots, N_K(s)I$ block diagonals compose $N(s)$ block diagonal matrix in format of DKxDK. The identity matrix I is also DxD matrix unlike $S_x(s)$ is defined as KD dimensional column vector as $S_{X,1}, \dots, S_{X,K}$ and $l(s)$ is formulated as below;

$$l(s) = I + T^T \sum^1 N(s)T \quad (15)$$

All above formulas satisfy the requirement of definitions and expectation of posterior distribution of $w(s)$, so $E[w(s)]$ and $E[w(s)w^T(s)]$ can be formulated as below two equations:

$$E[w(s)] = l^{-1}(s)T^T \sum^1 S_X(s) \quad (16)$$

$$E[w(s)w^T(s)] = E[w(s)]E[w^T(s)] + l^{-1}(s) \quad (17)$$

In the Maximization (M) step, new model parameters T and Σ that maximize the Eq. 12 are calculated as below:

$$T^i \sum_{s=1}^S N_k(s) E[w(s)w^T(s)] = \sum_{s=1}^S S_X^i(s) E[w^T(s)] \quad (18)$$

$$\Sigma_k = \frac{1}{n_c} \left(\sum_{s=1}^S S_{XX^T,k} - M_k \right) \quad (19)$$

Eq. 18 is just a linear equation system that is RxR. It is solved using basic linear algebra.

2.2.3 Channel Compensation Algorithms

The problem of channel variability dealt with in constructing classifiers for speaker recognition using i-vectors as features because of i-vector extraction method don't have the ability to separate channel and speaker variability. For this thesis Probabilistic Linear Discriminant Analysis (PLDA) technique is used. PLDA is a probabilistic version of the Linear Discriminant Analysis (LDA) technique.

LDA technique is generally used in pattern recognition to reduce the dimension. LDA principle is based on better discriminate between different classes by exploring new orthogonal axes. These orthogonal axes must maximize between-class variance and minimize intra-class variance [37]. To satisfy that maximizing the between class

covariance S_b and minimizing the within class covariance S_w is applied in LDA method by maximizing the Rayleigh quotient in Eq. 20.

$$J(v) = \frac{v^t S_b v}{v^t S_w v} \quad (20)$$

Probabilistic version of LDA (PLDA) and standart LDA can be compared by investigating the relationship between factor analysis and principal components analysis. In the PLDA method x_{ij} which is j_{th} utterance of i_{th} speaker, can be computed as follows;

$$x_{ij} = \mu + Fh_i + Gw_{ij} + \epsilon_{ij} \quad (21)$$

where μ is the mean of all i-vectors in the training data. Also this formula consists of signal component $\mu + Fh_i$ and the noise component $Gw_{ij} + \epsilon_{ij}$. Signal component represents the speaker and noise component represents the variability of session for a given speaker. F and G are factor loading matrices, h_i are the speaker factors, and w_{ij} are the channel factors. h_i and w_{ij} have Gaussian prior distributions, $N(0; I)$. ϵ_{ij} is admitted as Gaussian residual noise and it can be expressed with a diagonal covariance matrix.

CHAPTER III

REPLAY ATTACK

Vulnerability of a speaker verification system became the most important part of recently researchs because of used applications of that verification systems exists in security required areas. As mentioned in Chapter 1, replay attack is a low cost and low effort spoofing attack type and it does not require high technology knowledge. Obtaining voice data via internet is easiest way nowadays, espacially for well-known people. Second step is finding nearest neighbored voice data to the hypothesized speaker. This voice data can be used in spoofing attack attempt. The flowchart in Fig. [4] summarize which steps are followed in this thesis for replay spoofing attack.

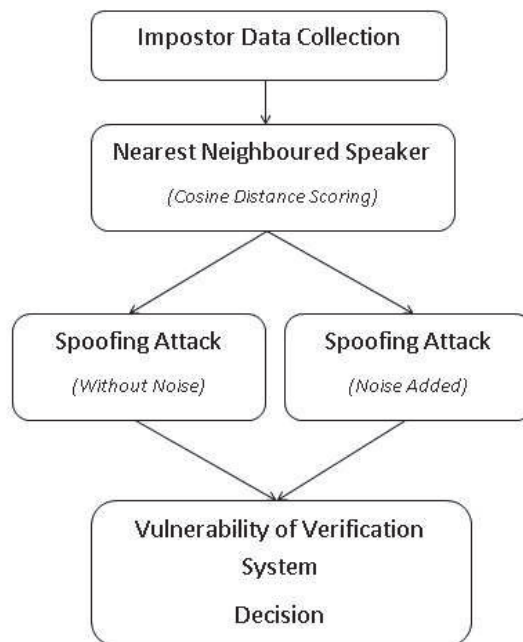


Figure 4: Flowchart of Replay Spoofing Attack

3.1 Impostor Data Collection

In replay attack part of this thesis, Data should be obtained to enroll speakers to UBM system and then attack to verification system by using nearest neighbour speaker. In replay attack data does not need be produced, just need to be recorded or downloaded. In our scenario, pre-recorded voice data of some famous people is downloaded from youtube.com via an online media downloader website. By using that website all media content downloaded with same specification such as extension which is mp3. Other specifications of downloaded data will be indicated in Experiments section.

3.2 Scoring Algorithm

Spoofing attack can be applied by using nearest neighboured speaker to previously enrolled speaker. Nearest neighboured speaker can be obtained by using scoring algorithm. In Speaker Verification space, scoring means how two different speeches match to eachother. Likelihood Ratio Test, Cosine Distance Scoring and Super Vector Machine are three different algorithm can be applied. In this case Cosine Distance Scoring is used to find nearest speaker to target speaker.

3.2.1 Cosine Distance Scoring

Cosine Distance Scoring is a similarity measurement method between two vectors. It measures the cosine of the angle between them. In this scenario the angle of i-vector of target speaker and i-vector of test speaker is measured. The test speaker who has lowest angle is labeled as nearest neighbour speaker and it is used to pass speaker verification system with spoofing attack. The cosine of two vectors can be derived by using the Euclidean dot product formula.

$$a \cdot b = \|a\| \|b\| \cos(\theta) \quad (22)$$

And the score is the cosine of the angle θ .

$$\cos(\theta) = \frac{a \cdot b}{\|a\| \|b\|} \quad (23)$$

In speaker verification system, the cosine distance between the claimed speakers i-vector, $w_{claimed}$, and the test speakers i-vector, w_{test} , is calculated as follows:

$$score(w_{claimed}, w_{test}) = \frac{w_{claimed}^T w_{test}}{\|w_{claimed}\| \|w_{test}\|} \quad (24)$$

Then calculated score is compared with a pre-defined threshold θ to decide test speaker is same with claimed speaker or not.

3.3 Spoofing Attack

Recorded data will be used for enrollment and the most similar speaker will be used for spoofing attack which will be applied with a replay device. In this thesis Matlab is used to create player environment. As mentioned in section 1.3.4, replay attack is the most low effort spoofing attack. First step is creating the UBM and then enrolling some preferred data to the system. Then nearest neighbour speakers to the enrolled speakers are found with making a decision with scoring algorithm. These found most similar speakers speech data are used to attack to spoof the verification system. According to calculated score, test speaker is labelled as a client speaker or impostor speaker after decision process.

3.3.1 Noise Added Attack

Recorded speech data has a recording device noise and also will has player device noise in spoofing attack stage. In [38] noisy condition is investigated and current state-of-the-art spoofing detection algorithms are examined if they work well under additive noisy conditions or not. They also investigate how additive noises affect the spoofing detection performance and what kind of noise is more dangerous than others to degrade the vulnerability of spoofing detection systems. As conclusion they

shown that, the performance of detection systems degrade in all the noise scenarios and the system performance varies significantly under different noise scenarios and the phase-based features are noise robust than magnitude-based features.

In this thesis experiment, there is no extra player noise but to simulate noisy situation, gaussian noise with different levels are added to the original speech data. Noise levels are applied from 5dB to 20dB. In Fig. [5] features are compared with eachother for clean and noisy speech. Below figure proof that, noise which is applied to speech, has a serious effect even on extracted features. So additive noise can be used to change detection algorithm result. Results after these process are examined in chapter 5.

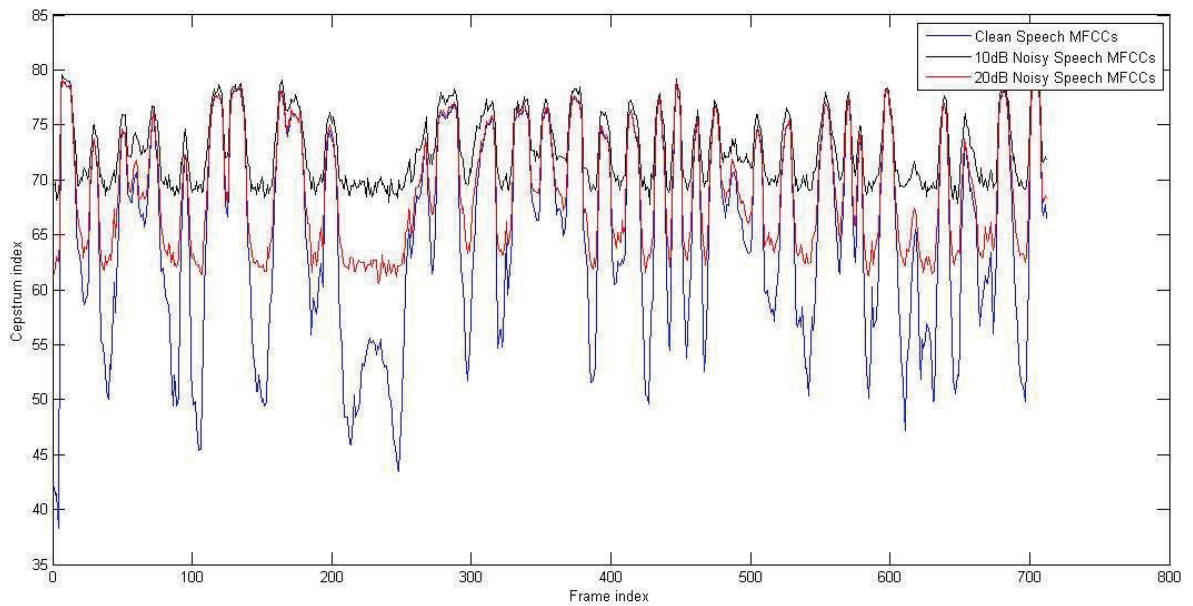


Figure 5: Effect of Gaussian Noise on MFCC features.

CHAPTER IV

SYNTHETIC SPEECH DETECTION

In this thesis, the effectiveness of spoofing attacks was also investigated with Statistical Speech Synthesis (SSS) systems using limited amount of adaptation data and additive noise. A UBM is first trained and then speaker-specific models are obtained by adapting the UBM using a Maximum A Posteriori adaptation (MAP) approach. Typically, supervector of mean vectors in UBM is very high dimensional which increases the number of parameters to adapt. In the factor analysis (FA) approach [14], mean vectors of speakers, m_s , are represented in a lower dimensional total variability space in which

$$m_s = m_0 + Tw_s \quad (25)$$

where w_s is called an identity vector (i-vector). T matrix is trained using a database where multiple sessions are available for each speaker. Even though removing the session effects from the i-vectors is important for successful verification, session differences contain valuable information for detecting synthetic speech. For session-i, channel vector can be defined as

$$m_{c,i} = m_{s,i} + m_s \quad (26)$$

where $m_{s,i}$ is the i-vector extracted in session-i and m_s is the mean i-vector for speaker s. Channel vectors contain information about the distortions that are session-specific. In the case of synthetic speech, there is additional variability. So synthetic speech can be defined with that session variability.

The differences between i-vectors of synthetic and natural speech is investigated

through visualization. To that end, Fisher linear discriminant analysis (LDA) is used to reduce dimensionality of the channel vectors to 2. Clean and noisy cases are compared according to the position of clusters. But these all comparisons are made for the attacker and defender who have same SSS technologies. An answer is investigated for the question of what if the attacker and the defender use different SSS technologies. To test that condition, STRAIGHT vocoding and GV adjustment is used at the attacker side but not at the defender side.

4.1 Frame and Segment Specific Importance Weighting

In this thesis, it is investigated several detectors without attack specific prior assumptions. The approach is based on the hypothesis that long- and/or short-duration artifacts will be observed in the synthetic speech without any constraints on the type of artifacts. Artifacts that typically occur in stop sounds during synthesis because of their rapidly changing dynamics and sudden glitches that occur frequently with the unit selection systems are examples of short-duration artifacts. Overly smooth parameters generated with HMM-based synthesis is an example to long-duration artifacts. The SSD algorithm should be sensitive to both types of artifacts to be effective.

The first SSD was developed using an unsupervised approach where a GMM is trained for natural speech and a GMM is trained for synthetic speech. After aligning each speech frame with a Gaussian, each Gaussian component is treated as an independent detector and detector scores are fused with logistic regression. Second method is based on designing detectors that are focused on detecting artifacts in specific phonemes. This approach can be successful at detecting phoneme-specific artifacts in synthetic speech. To reduce the data sparsity issue, broad-level sound class detectors are used in a third approach. All three methods performed substantially better than the baseline detector that treats all Gaussians and phonemes equally for the known attack types. However, the proposed systems did not substantially improve

the baseline system for unknown attack types. Fusing the three proposed detectors further improved the SSD performance both in known and unknown conditions.

4.2 Synthetic Speech Detectors

Detailed structure of SSDs can be explained as following steps and in Fig. [6]. Mel-frequency cepstral coefficients (MFCC) are first extracted from the speech utterance. Then, the feature vectors are grouped together into J groups. After grouping, log-likelihood ratio (LLR) detection is done for each group of feature vectors. To compute LLR, a GMM is trained for natural speech and a GMM is trained for synthetic speech. Same GMMs are used for all J groups. Once the score of each group is computed, score fusion is done using a logistic regression function to compute the final score $S(u)$. A hard threshold is used to compute the final decision.

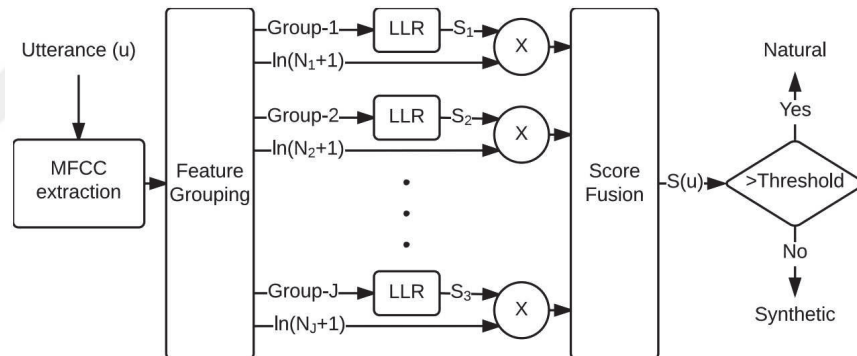


Figure 6: Overview of the proposed synthetic speech detectors.

CHAPTER V

EXPERIMENTS

In this chapter details of replay spoofing attack are given about the used universal speaker model and downloaded voice data for enrollment. Then system tests made by replay spoofing attack with and without added noise. As second experiment synthesized speech detector investigation results are shared. Then a discussion section written for about what happened in this thesis and what should be next iterations in future works section.

5.1 Replay Spoofing Attack Experiments

As explained in section 2.2.1 Universal Background Model which is assumed to represent whole speaker space is the necessary part for the information of our systems should know how people averagely say the voices. In this thesis "Turkish Broadcast Database" used as UBM and it is created by the member of Özyeğin Speech Processing Laboratory.

Turkish Broadcast Database includes 85 different people and 24500 different ".wav" file which are recorded voice data's of famous people from news. By using these data, training system process is started which includes MFCC feature extraction, T-matrix training and i-vector extracting and PLDA technique. All related Matlab codes about these process are prepared before by a member group of the Özyeğin Speech Processing Laboratory and this algorithms are used in this research and thesis. *All used OzuLibrary functions and are listed in Appendix A.*

Then youtube.com website is used to download voice data from famous people in three different categories which are called as "general", "news" and "test". As a summary of the distrubition of speakers and contents are given in Table [2].

Categories	Number of Speakers	Number of .wav file
News	102	627
General	102	493
Test	197	1290
Total	401	2410

Table 2: Distrubiton of Speakers and Contents

All these voice data are downloaded by the help of an online media content downloader website ("http://www.clipconverter.cc/") and then they are converted to have same specification between eachother. These specifications are sample rate, number of channel, length and format. For all speech data, sampling rate is 16 kHz, number of channel is 1, length is 5 seconds and data are 16 bit. Also these contents are normalized and DC parts are removed by using speacial Matlab functions.

Once UBM is created as a whole speaker space, next step should be enrollment. For this process one wav file is picked up from every different speakers from one of the above categories. It can be any of them but for this experiment "news" category is used which has 102 speakers. Only one speech data is selected from every 102 speakers. As a normal training procedure, i-vectors are extracted for the remaining data from "news" category and a baseline system is builded. The difference between UBM and baseline system is that baseline is a small data used speech space which is created by taking the UBM as references. As described in Section (1.2) most common scalar performance measure is EER which is measured for this new baseline system and result is **0,9010**. DET curve in Fig. [7] shows the performance of baseline system. The main aim to find EER in that step to see how system acceptance and rejection mechanisms work. Although it is not a spoofing attack, all speakers' voice data except the one used for enrollment, used to test system and verification system worked with a high percentage which means low EER (0.9010).

Enrolled voice data can be identified by the used speaker verification system with high accuracy as an expected scenario.

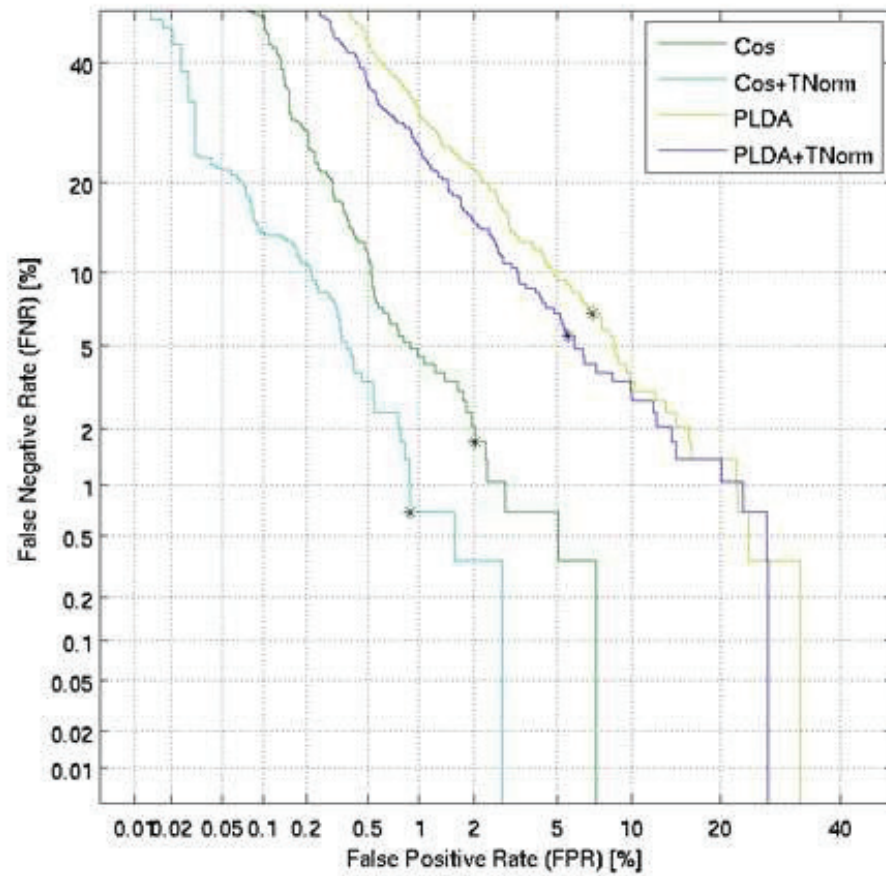


Figure 7: Baseline DET Curve

Then it was the time to test system with another people's voice data but the key point is to find nearest neighbour voices. Other downloaded categories are used to find 3 mostly similar speaker voices data to 102 enrolled speakers to the system. "General" and "test" categories are used for nearest neighbour and system is tested with new speakers as a spoofing attack. New EER result is 7.5 and that means only 7.5 speaker made a successful attempt from every 100 trial to infiltrate the system. DET Curve in Fig. [8] shows the performane after spoofing attack.

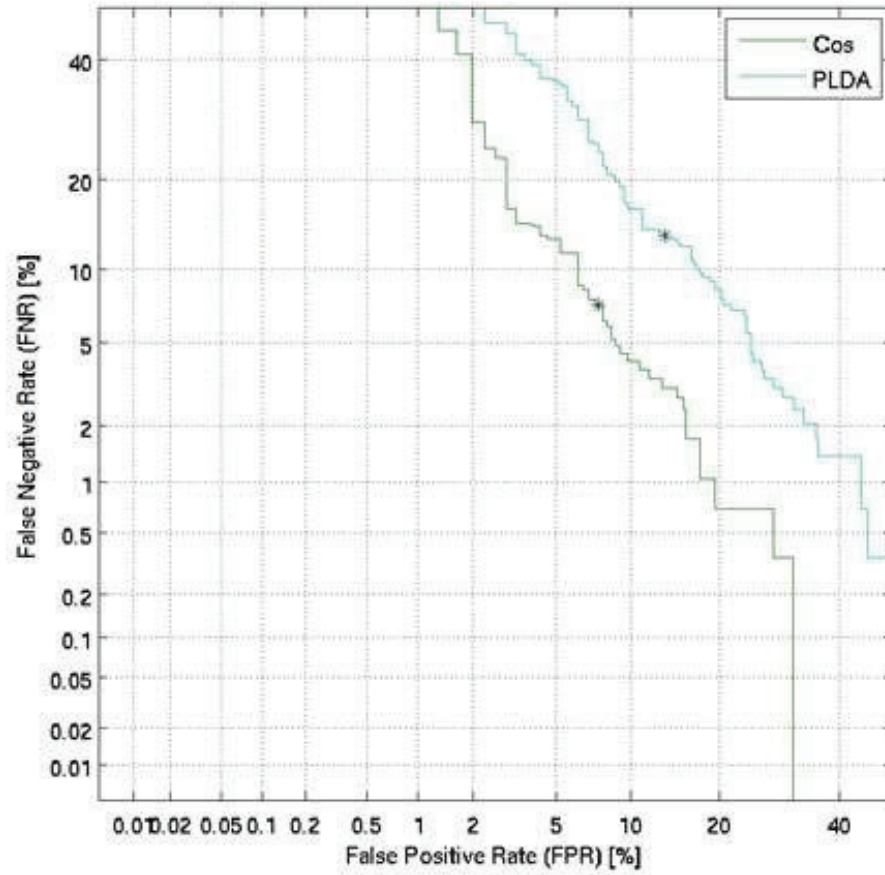


Figure 8: Attack DET Curve

5.1.1 Effect of Noise on Speaker Verification System

Above experiments and results are a representation of a system and spoofing attack under ideal condition but in real life it is impossible to reach that conditional environment. In this thesis **Gaussian Noise** in some different levels are applied to system to understand how it will affect.

Table [3] can easily shows the effect of noise on EER for different levels. As a summary of table; lower EER means higher accuracy for the system and higher noise means lower EER because all tests applied as replay spoofing attack. Nearest

neighbour data which can pass from verification system without noise, can't achieve to pass verification system anymore after gaussian noise added. The reason of that behaviour is that system is under attack by replay spoofing attack and verification system try to reject all attempt. Because of all attempts are spoofing attack, false rejection is not a case at this research. By the help of added noise false acceptance goes down and it increases the vulnerability of speaker verification system.

DET Curves show effect of added gaussian noises on EER in Fig. [9], [10], [11], [12].

	No Noise	White 5 dB	White 10 dB	White 15 dB	White 20 dB
Attack EER	7.5163	1.73	1.578	1.38	1.3

Table 3: Effect of Noise to Spoofing Attack EER

5.2 *Synthetic Speech Detection*

The synthetic speech detectors can be investigated by evaluating spoofing attack channel vector. Channel vectors of synthetic and natural speech is compared in Fig. [13] and Fig. [14]. Both test and train synthetic data are generated with STRAIGHT and GV for these two scenario. In the clean case, there is a clear separation between synthetic and natural vectors. In the noisy case, the two clusters are still clearly separable. However, the margin is not as large as the clean case. Thus, noise distorts the smooth structure of the synthetic features and make clean and noisy channel less separable. In Fig. [15] effect of mismatch in synthesis technologies are shown. STRAIGHT vocoding and GV adjustment is used at the attacker side but not at the defender side. Using different synthesis technologies by the attacker and defender caused significant overlap between the clusters which makes the detection problem harder.

WSJ1 database [39] is used for the verification experiments. 69 male test speakers are enrolled into the system. Each enrollment utterance is around 4-6 seconds long.

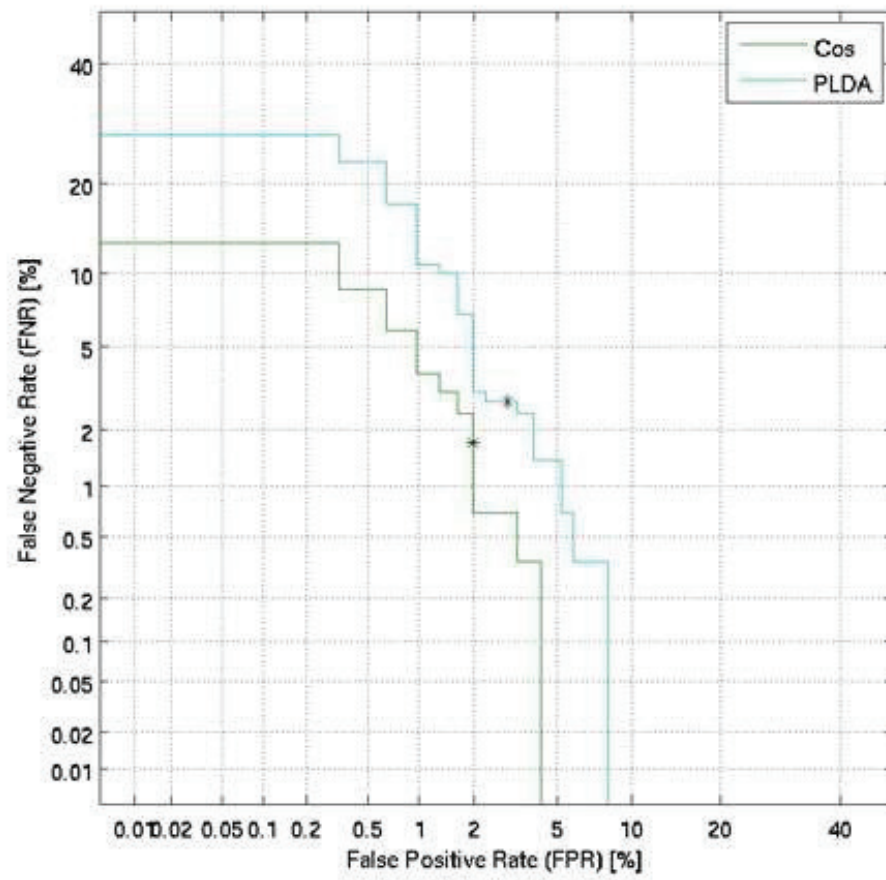


Figure 9: Attack DET Curve with 5dB Gaussian Noise

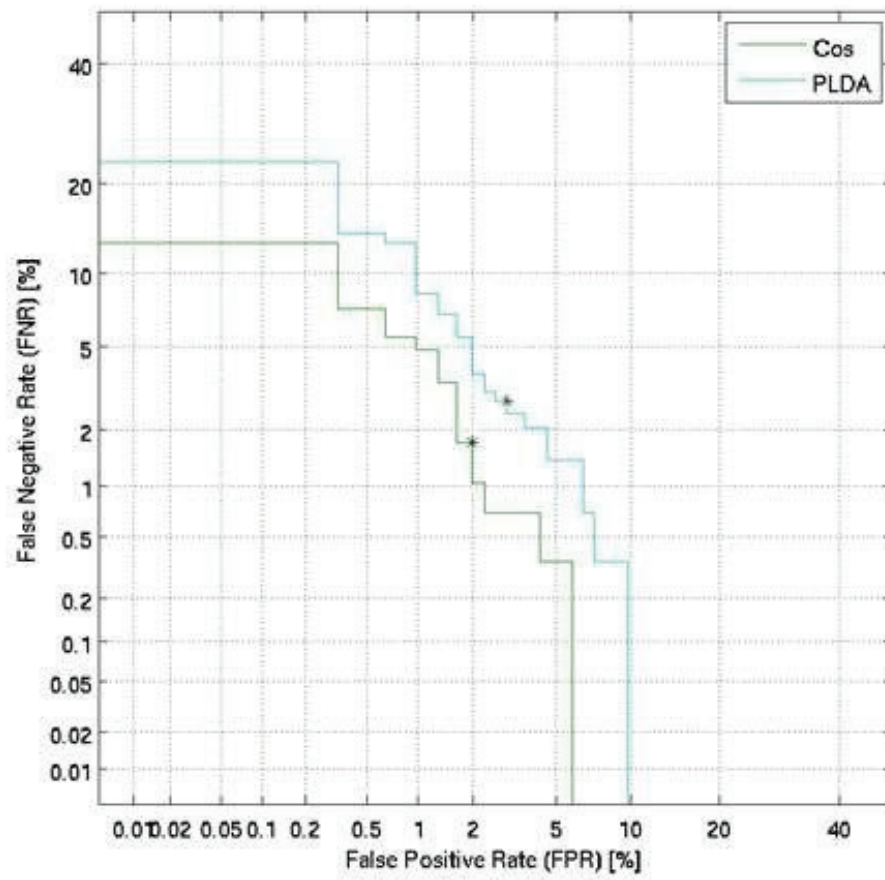


Figure 10: Attack DET Curve with 10dB Gaussian Noise

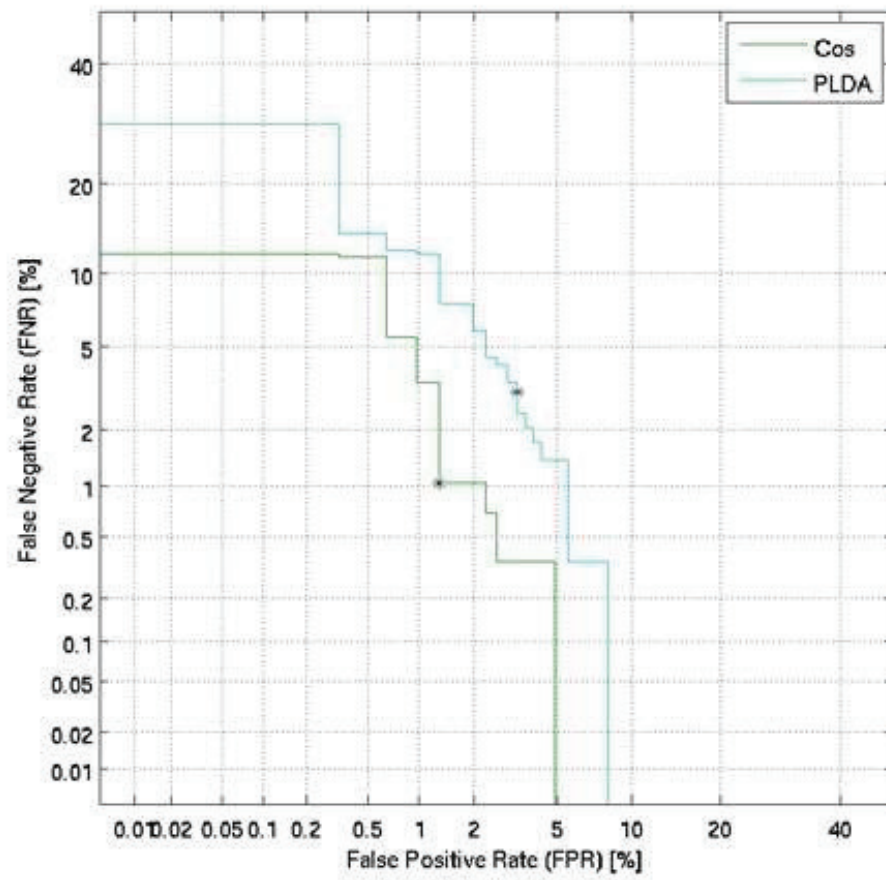


Figure 11: Attack DET Curve with 15dB Gaussian Noise

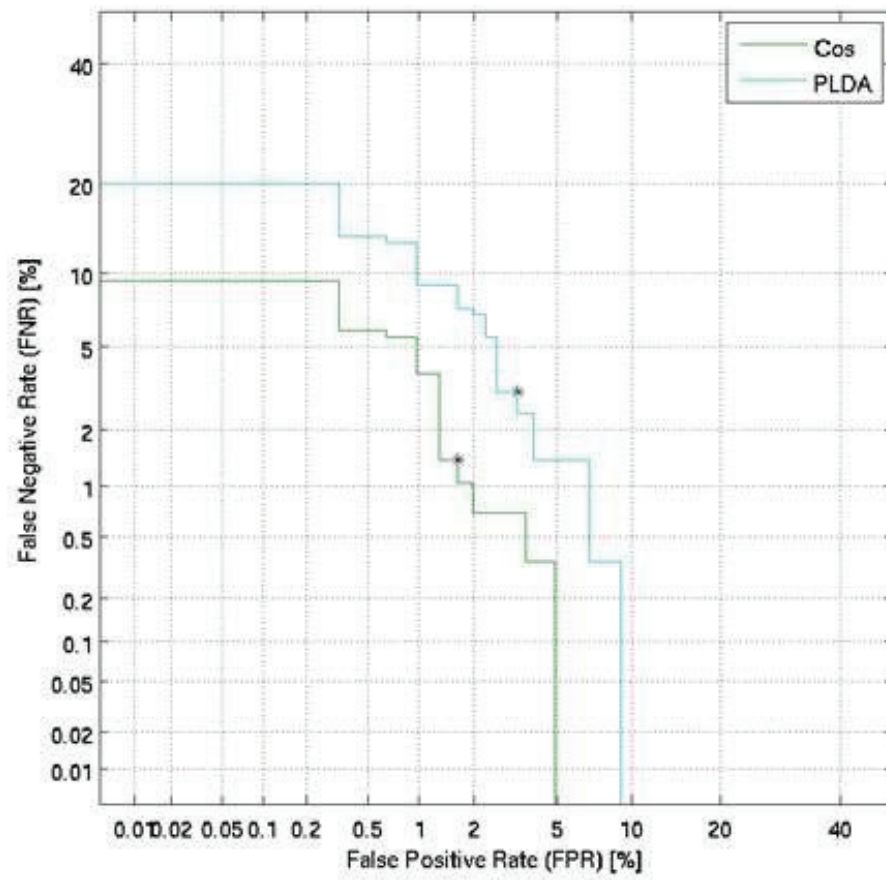


Figure 12: Attack DET Curve with 20dB Gaussian Noise

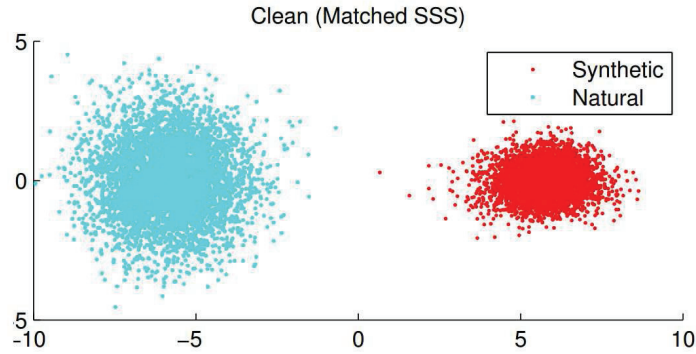


Figure 13: Clean Synthetic and Natural Data

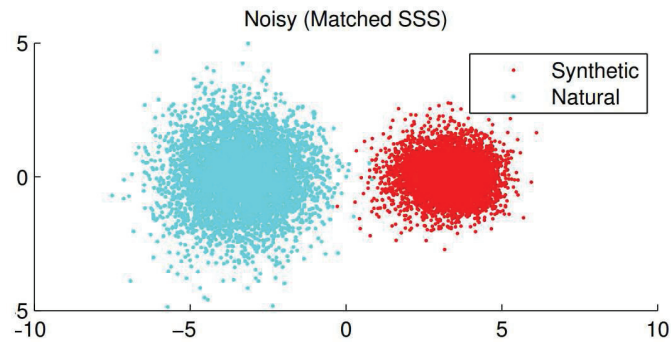


Figure 14: Noisy Natural and Synthetic Data (Matched Condition)

For each enrolled speaker, 59 client tests and 340 impostor tests are done. Impostor tests are created by using 5 utterances from each of the 68 impostor speakers among the enrolled speakers. Each test is done using one utterance. Verification system uses 19 dimension MFCC plus 1 energy static features and their delta and delta-delta features. However, static energy is not used which makes the total dimension of features 59. 256 mixture UBM is trained using 84 male speakers, and 60 utterances from each speaker. T matrix is trained using those same speakers and utterances. Rank of the T matrix is set to 400.

Experiments are done for clean training and test data as well as noisy training and

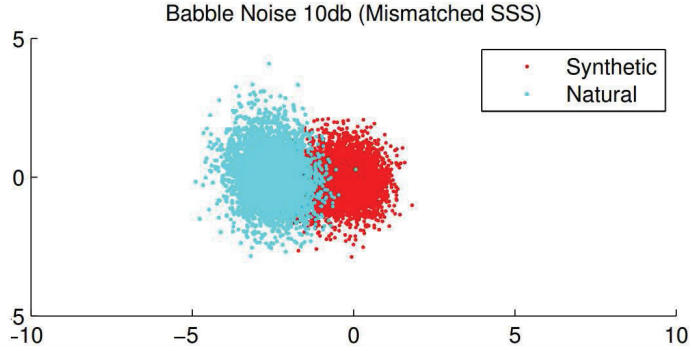


Figure 15: Noisy Natural and Synthetic Data (Mismatched Condition)

test data. Noise is added to clean speech samples at 10, 15, and 20dB SNRs because when the SNR is below 10 dB, performance of the verification system is found to be unacceptably poor. The detector and the verification systems are trained using a mixture of white, babble, car, and station noisy samples under 10, 15, and 20dB SNRs in noisy conditions. Bus, cafe, metro, and office noises are used only during testing.

For each enrolled speaker, different statistical models are created for attacks using adaptation with one, two, three, and four utterances. Synthesis is done for all of the 69 speakers enrolled into the verification system. Enrollment and test data are not used for adaptation. Experiments when 150 utterances are used for adaptation are also done for comparison purposes. Speaker-independent (SI) model is generated using four male speakers and 1250 utterances from each speaker. Constrained structural maximum a posteriori linear regression (CSMAPLR) algorithm is used for adaptation.

SSS systems were trained with 198 dimensional vectors consisting of 40 Mel-Generalized Cepstral (MGC), 1 LogFundamental frequency (LF0), and 25 Band APeriodicity (BAP) coefficients and their delta and delta-delta parameters. 25 msec analysis window with 5 msec frame rate is used for feature extraction. Phonemes are modeled with 5 state hidden semi-Markov models (HSMM). STRAIGHT vocoding

and global variance adjustments are done to improve the synthesis quality.

Baseline performance of the voice verification system in clean training and test conditions in terms of EER is %0.23. Performance of the system for individual noise types and SNRs are shown in Table [4].

Seen noises	10db	15db	20db
white	4.53	1.98	1.16
babble	1.27	1.23	1.11
car	1.21	1.19	1.26
station	0.96	0.97	1.03

Unseen noises	10db	15db	20db
bus	1.27	1.24	1.22
metro	1.26	1.10	1.13
office	1.25	1.28	1.25
cafe	1.13	1.13	1.15

Table 4: EER Of The Voice Verification System For Different Noise Types and SNRs.

For spoofing attacks, threshold of the voice verification system is set to %1.81 average EER point. Results with clean train/test and noisy train/test are shown in Fig. [16]. Noise substantially increases the effectiveness of the attacks. Effectiveness of car and bus noises are below others since those noise types have lower bandwidth. Interestingly, effectiveness of the attacks are close to each other at different SNRs. This is thought to be a result of the fact the system is trained with a mix of all SNRs and all noises. Moreover, the calibration is also done with a mix of all conditions. Thus, the system does not seem to substantially favor any particular SNR.

Effectiveness of the spoofing attacks in such mismatch conditions are reported in Fig. [17]. Under the mismatched SSS synthesis conditions, detection performance decreases substantially especially for babble and white noises. This result calls for

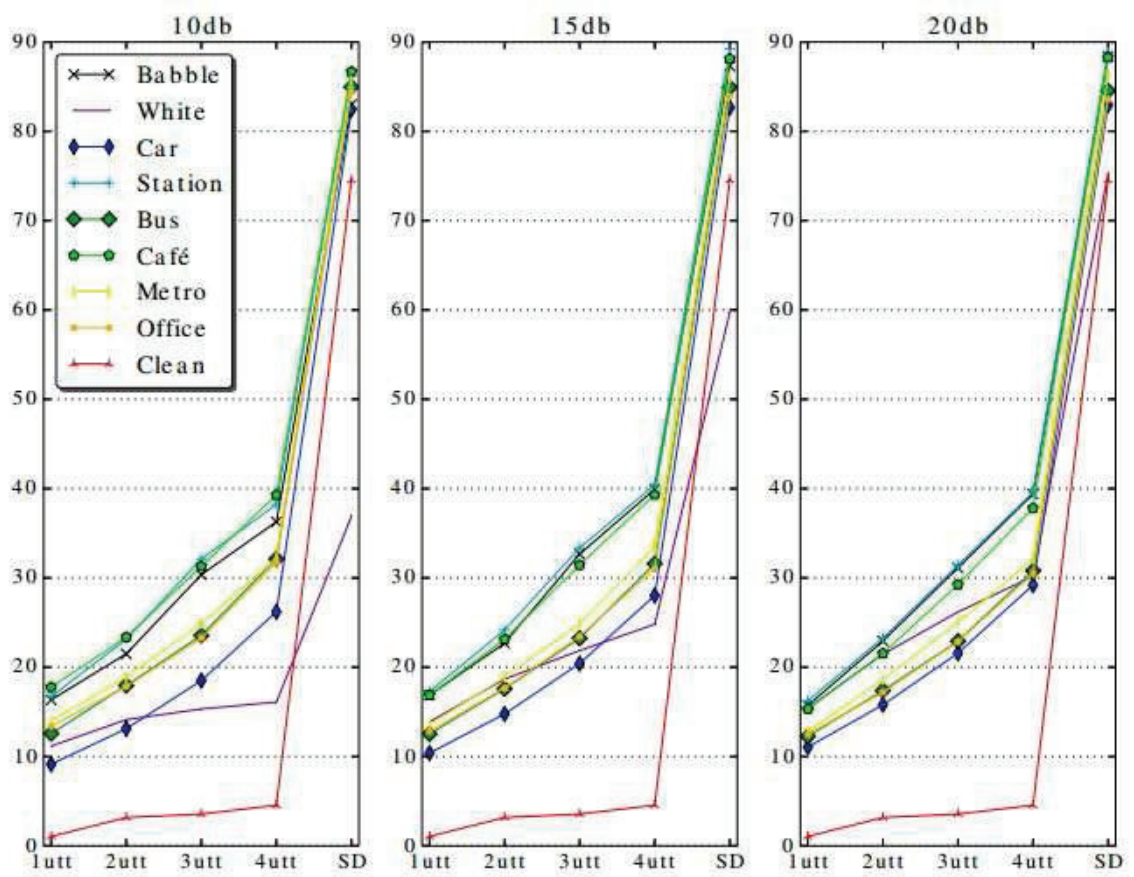


Figure 16: Verification False Alarm Rates Under Attack With Synthetic Speech

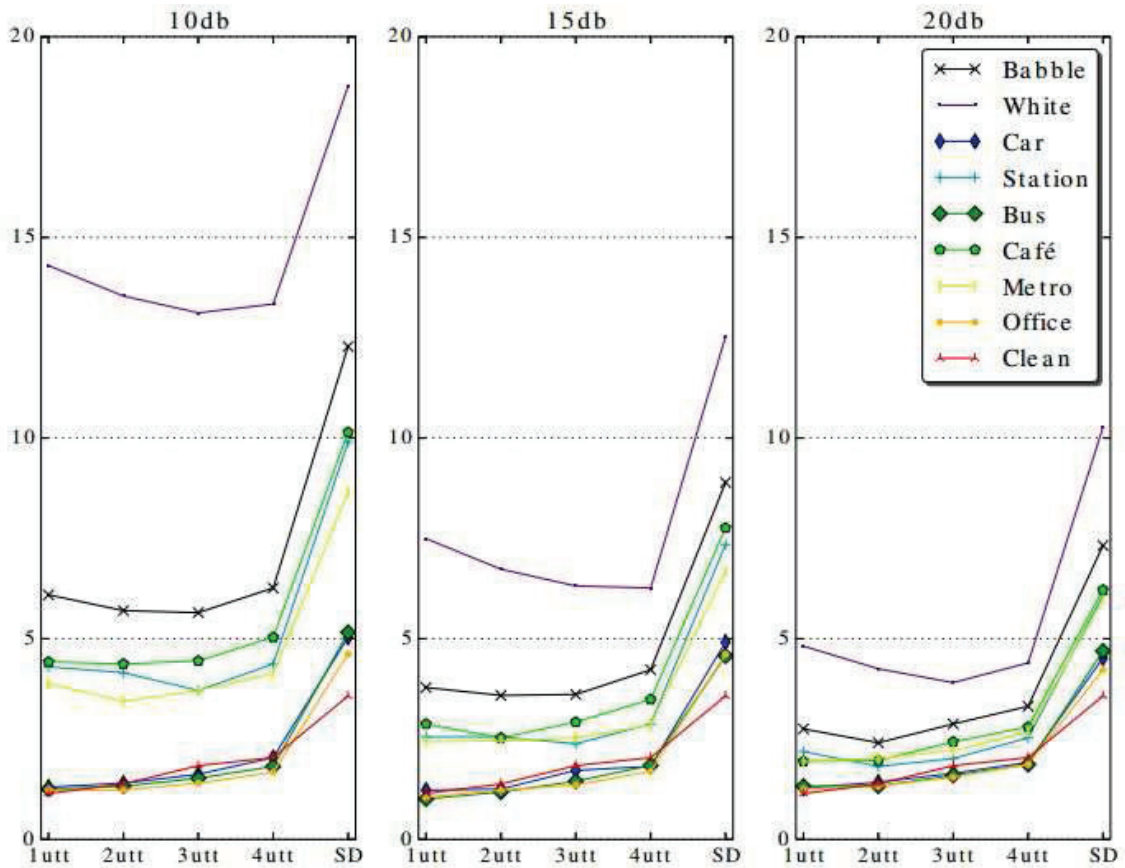


Figure 17: Detector Performance for Mismatched Case

training detectors with different synthesis conditions and not fit the detector on one particular type of SSS.

After investigating i-vector based spoofing attack Synthetic Speech Detection is experimented. The synthetic speech detectors were trained with 19 MFCCs together with the delta and delta-delta features. In short-time analysis, frame length was 25msec and frame rate was 10msec. Bigaussian voice activity detection (VAD) was used where energy of the speech and noise frames are modeled with single Gaussians and likelihood ratio detector is used to detect speech frames.

The baseline synthetic speech detector had a 512-component GMM to model natural speech. Similarly, synthetic speech was modeled with 512-component GMM.

For natural speech, GMM training was initialized using k-means clustering. The GMM for synthetic speech was adapted from the GMM of the natural speech using a maximum a posteriori (MAP) approach. Experiments with synthetic speech GMM that was trained independent of the natural speech GMM were also performed for comparison.

Experimental results for the development and evaluation data are shown in Table [5]. The baseline LLR detector is trained with two different methods. In one approach (LLRnoAdapt), two independent GMMs are trained for the natural and synthetic speech. In the second approach (LLR-Adapt), a GMM is trained for natural speech and then adapted to the synthetic speech using MAP adaptation.

The LLR-Adapt system performed better for known conditions while LLR-noAdapt performed better for unknown conditions. Thus, even though LLR-Adapt performed better than LLR-noAdapt on average, it could not generalize as good as the LLR-noAdapt. This result indicates that, during GMM training, some of the novel clusters in the synthetic data that were useful for ambiguity detection, could not be modeled well with adaptation of GMM for natural speech.

Gaussian-based system performed better than class- and phoneme-based methods both for known and unknown conditions. In particular, Gaussian-based approach performed better for the S1, S2, and S5 methods, all of which are voice conversion algorithms. Unlike the phoneme- and class-based systems, Gaussian-based detector can learn to detect shortduration artifacts. Thus, the presence of short-duration acoustic distortions seems to be more informative for detecting voice conversion attacks.

Class-based system performed better for S3 and phonemebased system performed better for S4 attack methods. Both S3 and S4 are generated with HMM-based TTS. Unlike the voice conversion systems, HMM-based TTS systems generate smooth trajectories. Thus, sudden acoustic distortions are rarely generated with those systems. In this case, overlysmooth longer segments seem to be more informative for detection.

Small distortions in a long segment can be detected well with class- and phoneme-specific detectors that are focused on particular segments. However, Gaussian-based approach is not expected to be as successful with this type of attack because speech frames are generated with a maximum likelihood approach in HMM-based synthesis. Thus, the parameter generation algorithm is designed to generate high likelihoods for each frame and individual Gaussians are not expected to detect the artifacts in features.

Duration-based weighting consistently improved class- and phoneme-based performance. However, for the Gaussianbased approach, performance improved slightly for the unknown systems and degraded slightly for the known systems. We believe there are at least two major factors behind this result. Firstly, because an important strength of the Gaussianapproach is its ability to detect short-time artifacts, weighting with duration can hurt its performance. Secondly, duration of observed Gaussians can change significantly depending on the spoofing system used which can increase the variability of features and make the detection task harder. Because ASR systems take phoneme durations into account during recognition, that effect is not as important in the phonemeand class-based methods.

The core hypothesis in the proposed system was that different Gaussians, phonemes, sound-classes contribute different amounts of information for synthetic speech detection. To test that hypothesis, experiments were performed with each Gaussian, phoneme, and sound-class separately. For the Gaussian case, results are shown in Fig. [18], for the phoneme case, results are shown in Fig. [19]. In both cases, large variation in detection performance can be observed which verifies our hypothesis.

Fig. [18] shows the correlation of number of occurrences vs EER computed with each of the 512 Gaussians. Even though EER and durations have a negative correlation, the pattern is weak and does not impact the overall detector performance significantly. This result is inline with the finding that durationbased weighting does

		Normal						Duration-based weighted			
		LLR		Logistic Regression				Logistic Regression			
		noAdapt	Adapt	Class	Phoneme	Gaussian	Fusion	Class	Phoneme	Gaussian	Fusion
Development	S1	0.47	0.76	0.68	0.69	0.47	0.41	0.54	0.54	0.51	0.46
	S2	10.24	5.12	3.37	3.41	1.89	1.83	2.99	3.13	2.26	2.20
	S3	0.07	0.07	0.03	0.09	0.20	0.17	0.03	0.09	0.18	0.11
	S4	0.04	0.09	0.05	0.03	0.25	0.20	0.03	0.07	0.20	0.13
	S5	4.63	3.04	2.78	2.86	1.72	1.57	2.65	2.72	1.59	1.47
	Total	4.21	2.42	1.92	1.77	1.17	1.11	1.67	1.66	1.19	1.14
Evaluation	S1	0.54	0.57	0.55	0.56	0.48	0.41	0.51	0.50	0.46	0.42
	S2	9.24	4.47	2.78	2.71	1.89	1.75	2.63	2.44	2.15	2.03
	S3	0.07	0.02	0.04	0.04	0.18	0.12	0.02	0.03	0.15	0.09
	S4	0.07	0.03	0.05	0.05	0.17	0.11	0.03	0.04	0.13	0.08
	S5	3.95	1.72	1.99	2.14	1.48	1.36	1.89	1.97	1.50	1.40
	S6	3.49	1.35	1.39	1.40	1.09	0.98	1.31	1.24	1.13	1.01
	S7	1.91	1.65	0.84	0.87	0.75	0.63	0.85	0.94	0.70	0.65
	S8	0.46	1.03	0.76	0.85	0.83	0.70	0.71	0.87	0.79	0.70
	S9	0.43	1.26	0.93	1.02	0.76	0.65	0.94	1.02	0.68	0.64
	S10	27.24	29.62	32.14	33.59	30.05	29.81	31.39	32.25	29.88	29.78
	Known	2.77	1.36	1.08	1.10	0.84	0.75	1.02	0.99	0.88	0.81
	Unknown	6.70	6.98	7.21	7.54	6.70	6.55	7.04	7.27	6.64	6.55
	All	4.74	4.17	4.15	4.32	3.77	3.65	4.03	4.13	3.76	3.68

Table 5: Performance of the Baseline and Proposed Detectors in Terms of EERs for the Development and Evaluation Data. Results are Presented with and without Duration-Weighting. S1, S2, and S5 Systems Use Voice Conversion (VC). S3 and S4 Systems Use HMM-Based Synthesis. Best Performing Algorithm For Each Attack Type is Shown in **Bold**.

not improve the performance of the Gaussian-based system.

The effect of duration is more significant with phoneme-based detector compared to the Gaussian-based detector. Duration versus EER is shown in Fig. [19] where a stronger negative correlation is observed compared to the Gaussian case especially for the vocalic sounds. The correlation disappears for some of the highly informative stop and fricative sounds.

The proposed detectors performed substantially better than the baseline detectors for known attack types. However, the difference is not substantial for the unknown attack types. To further boost the performance, the detectors were fused with a second stage of logistic regression algorithm. The fusion improved performance both for known and unknown attack types which indicate that the detectors generate complementary information

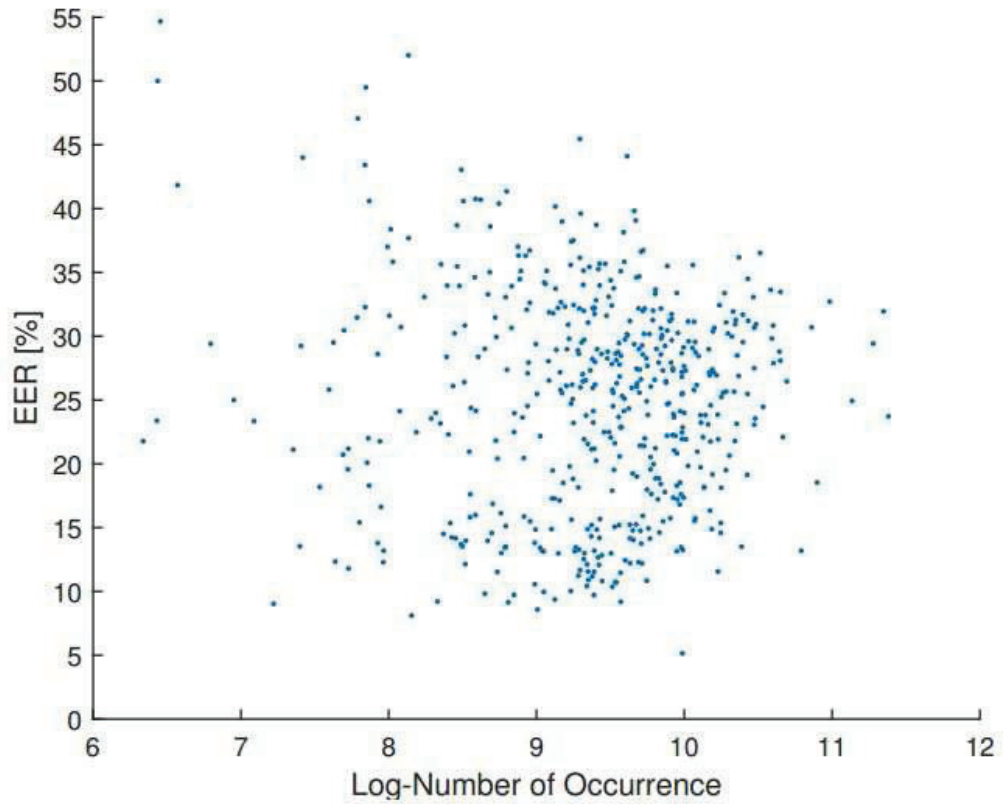


Figure 18: Detection performance of each Gaussian component versus its logarithm of number of occurrence in the development utterances is shown.

5.3 Discussion

It is examined that my findings in the light of the previous state of the subject about replay spoofing attack as outlined in the background, and make judgments as to what has been learnt in my work. This is first research for Turkish language by using downloaded replay attack. Calculated EER for enrolled speaker to system is 0,9010. That means the verification system I used, can recognize people with high accuracy. But because of these speakers are enrolled (known by system), low EER is an expected result. When nearest neighbour speaker used as replay attack data, EER becomes 7.5 and that means, any impostor can hack verification system algorithm by

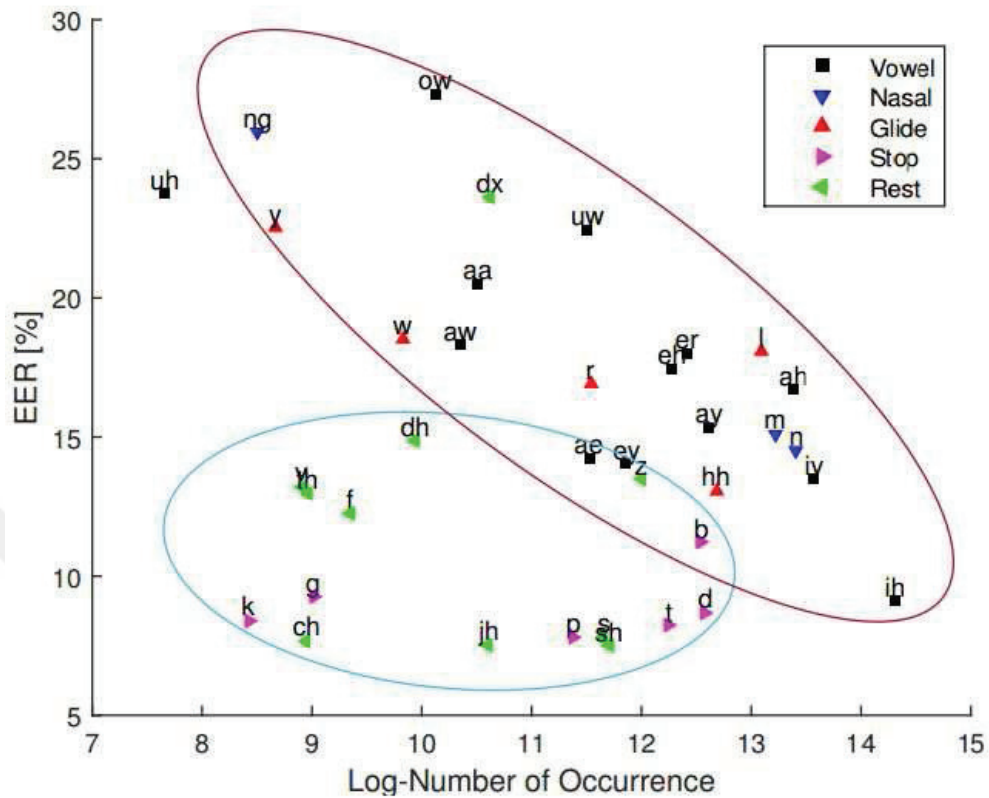


Figure 19: Detection performance of each phoneme versus its logarithm of number of occurrence in the development utterances is shown. Phonemes that are in the same sound-class are shown with the same color and shape.

using downloaded voice data from someone else.

Main difference between this research and others, previous works investigated far-field or telephone record with an external record device. But in this research external record device was not required.

For synthetic speech spoofing attack and detection algorithm, firstly it is proposed to attack an i-vector based voice verification system with SSS when limited amount of adaptation data is available. substantial performance gains are obtained when the verification system is trained with mixed noise conditions at and above 10 dB and noise is intentionally added to synthetic speech. It is also proposed a synthetic speech detector that is found to have excellent performance in noisy conditions.

CHAPTER VI

CONCLUSION

Vulnerability of a speaker verification system has been proposed under replay attack using voice data which are downloaded from youtube.com website. An online content downloader website is used to obtain data. Firstly an UBM system is created to simulate whole speaker space and some of downloaded data are enrolled to system. By using other voice data from same speakers are used to see how system works. It is obviously showed that system can detect attack attempts with high accuracy. Then other speakers data are used to test system performance under replay attack by using nearest neighbour. False acceptance rate increased and system performance went down a bit. Next research step was adding noise to data and after that process, system vulnerability increased because of noise decreased that false acceptance.

Replay attack is the most low cost and low technology required attack type and this thesis try to show that, speaker verification systems are not too strong against replay attack. Some anti-spoofing methods need to be applied because of people personal data safety.

In addition a multi-detector approach for synthetic speech detection is investigated where each detector is focused on a particular acoustic segment. The Gaussian-based detector performed better in voice conversion attacks. Phoneme- and class-based detectors performed better for HMM-based synthesis attacks. Duration-based feature normalization improved the phoneme and class-based systems but not the Gaussian-based system. As a future work, all attack attempts can be recorded and test for remaining attempts if it was used before or not? If it is used, it can be signed as spoofing attack even if it has ability to pass verification system.

For the replay attack case, the main subject of this thesis is using downloaded nearest neighbour data. Performance for enrolled speakers' voice data are examined in Section (3.2). But these data are another windowed data which is from exactly same conversation with the data used for enrollment. Same people but different conversation or data spoken in different time can be used for next research. In this these it is proved that noise has a positive effect on the detection of spoofing attack. But only gaussian noise type is investigated and other type noise can be investigated as babble noise or car noise. And also as an open point this is a research to investigate effect of the spoofing attack and don't suggest any anti-spoofing technique to literature. Vulnerability of speaker verification system under replay spoofing attack is a security related and hot topic nowadays and should be fixed that open point.

For the synthetic speech case, The hypothesis here was that different segments contribute different amounts of information and their scores should be weighted accordingly. Results confirmed the hypothesis. Because only commonly used MFCC features have been used. In the future work, a richer set of features and other classifiers such as SVM to further improve the detection performance is planned to use by focusing increasing the robustness of the detector to mismatch in SSS techniques.

APPENDIX A

UTILIZED FUNCTIONS AND TOOLBOXES

- OzULibrary

extractFeaturesOZU.m : A MATLAB function for feature extraction including different VADs

gmm em.m : A MATLAB function for GMM training

UbmCreating EM.m : A MATLAB wrapper function for UBM training

PLDA Train.m : A MATLAB function for PLDA model training

extract ivector.m : A MATLAB function for i-vector extraction

train tv space : A MATLAB function for T-Matrix training

UbmCreating EM.m : A MATLAB wrapper function for UBM training

- Toolboxes

DETware : DET-Curve plotting software written in MATLAB

BIBLIOGRAPHY

- [1] S. Vuuren, "Speaker Verification in a Time-Feature Space", Ph.D., (1999) thesis, Oregon Graduate Institute, March.
- [2] D. Neiberg, "Text Independent Speaker Verification Using Adapted Gaussian Mixture Models" , Ph. D. Thesis, Centre for Speech Technology (CTT) Department of Speech, Music and Hearing KTH, Stockholm, Sweden supervisor: Hakan Melin 2001-12-11
- [3] Z. Wu, H. Li, "Voice conversion and spoofing attack on speaker verification systems", *Asia-Pacific Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Proceedings of (2013).
- [4] Syris Technology Corporation, "Technical Document About FAR, FRR and EER", 2004
- [5] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, " Spoofing and countermeasures for speaker verification: A survey", *Speech Communication*, vol. 66, no. 0, pp. 130 153, 2015.
- [6] T. Masuko, K. Tokuda, T. Kobayashi, S. Imai, "Speech synthesis using HMMs with dynamic features", *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 1996
- [7] Y. Stylianou, "Voice transformation: a survey", *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009
- [8] J. Villalba, E. Lleida, "Detecting replay attacks from far-field recordings on speaker verification systems", *Vielhauer, C., Dittmann, J., Drygajlo, A., Juul, N.*,

- Fairhurst, M. (Eds.), Biometrics and ID Management. Springer. Lecture Notes in Computer Science, pp. 274285., 2011*
- [9] F. Alegre, R. Vippera, and N. Evans, "Spoofing countermeasures for the protection of automatic speaker recognition systems against attacks with artificial signals", in *INTERSPEECH, 13th Annual Conference of the International Speech Communication Association*, 2012.
- [10] N. Evans, T. Kinnunen, and J. Yamagishi, "Spoofing and countermeasures for automatic speaker verification", in *Proc. Interspeech*, 2013.
- [11] N. Evans, T. Kinnunen, J. Yamagishi, Z. Wu, F. Alegre, and P. DeLeon, "Voice anti-spoofing, in Handbook of biometric anti-spoofing", *S. Marcel, S. Z. Li, and M. Nixon, Eds. Springer*, 2014.
- [12] Y. W. Lau, M. Wagner, and D. Tran, "Vulnerability of speaker verification to voice mimicking", in *Proc. Int. Symposium on Intelligent Multimedia, Video and Speech Processing*, 2004.
- [13] Y. Lau, D. Tran, and M. Wagner, "Testing voice mimicry with the YOHO speaker verification corpus", in *Knowledge-Based Intelligent Information and Engineering Systems. Springer*, 2005, pp. 907907.
- [14] J. Mariethoz and S. Bengio, "Can a professional imitator fool a GMMbased speaker verification system ?", *IDIAP Research Report (No. IdiapRR-61-2005)*, 2005.
- [15] R. G. Hautamaki, T. Kinnunen, V. Hautamaki, T. Leino, and A.M. Laukkanen, "I-vectors meet imitators: on vulnerability of speaker verification systems against voice mimicry", in *Proc. Interspeech*, 2013.

- [16] T. Satoh, T. Masuko, T. Kobayashi, and K. Tokuda, "A robust speaker verification system against imposture using an HMM-based speech synthesis system", *in Proc. European Conference on Speech Communication and Technology (Eurospeech)*, 2001.
- [17] P. L. De Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, "Evaluation of speaker verification security and detection of HMMbased synthetic speech", *IEEE Trans. Audio, Speech and Language Processing*, vol. 20, no. 8, pp. 2280-2290, 2012.
- [18] J.F. Bonastre, D. Matrouf, and C. Fredouille, "Artificial impostor voice transformation effects on false acceptance rates", *in Proc. Interspeech*, 2007.
- [19] T. Kinnunen, Z. Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, and H. Li, "Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech", *in Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.
- [20] Z. Wu, T. Kinnunen, E. S. Chng, H. Li, and E. Ambikairajah, "A study on spoofing attack in state-of-the-art speaker verification: the telephone speech case", *in Proc. Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2012.
- [21] F. Alegre, A. Amehraye, and N. Evans, "Spoofing countermeasures to protect automatic speaker verification from voice conversion", *in Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.
- [22] Z. Wu, A. Larcher, K. A. Lee, E. S. Chng, T. Kinnunen, and H. Li, "Vulnerability evaluation of speaker verification under voice conversion spoofing: the effect of text constraints", *in Proc. Interspeech*, 2013.

- [23] Z. Kons and H. Aronowitz, "Voice transformation-based spoofing of text-dependent speaker verification systems", *in Proc. Interspeech*, 2013.
- [24] Z. Wu and H. Li, "Voice conversion and spoofing attack on speaker verification systems", *in Proc. Asia-Pacific Signal Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2013.
- [25] J. Lindberg, M. Blomberg et al., "Vulnerability in speaker verification- a study of technical impostor techniques", *in Proc. European Conference on Speech Communication and Technology (Eurospeech)*, 1999.
- [26] J. Villalba and E. Lleida, "Speaker verification performance degradation against spoofing and tampering attacks", *in FALA 10 workshop*, 2010, pp. 131134.
- [27] , "Detecting replay attacks from far-field recordings on speaker verification systems", *in Biometrics and ID Management, ser. Lecture Notes in Computer Science*, C. Vielhauer, J. Dittmann, A. Drygajlo, N. Juul, and M. Fairhurst, Eds. Springer, 2011, pp. 274285.
- [28] Z.-F. Wang, G. Wei, and Q.-H. He, "Channel pattern noise based playback attack detection algorithm for speaker recognition", *in Proc. IEEE Int. Conf. Machine Learning and Cybernetics (ICMLC)*, 2011.
- [29] F. Alegre, A. Janicki, and N. Evans, "Re-assessing the threat of replay spoofing attacks against automatic speaker verification", *in Proc. Int. Conf. of the Biometrics Special Interest Group (BIOSIG)*, 2014.
- [30] Z. Wu, S. Gao, E. S. Cling, and H. Li. "A study on replay attack and anti-spoofing for text-dependent speaker verification", *In Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA)*, pp. 15, Dec 2014.

- [31] Shang Wei, and S. Maryhelen, "Score normalization in playback attack detection", *Proceeding of IC ASSP2010 Conference, Dallas*, pp. 1678-1681, 2010.
- [32] K. C. Pohlmann, "Principles of Digital Audio", Sixth Edition, Mc Graw-Hill, New York, 2010.
- [33] F. Yeşil, "Comparision of Text-Independent Speaker Verification Systems in a Multi-Class, Semi-Automatic Detection Scenario", Ms. Thesis, Özyegin University , June 2013
- [34] D. A. Reynolds, "A Gaussian Mixture Modeling Approach to Text-Independent Speaker Identification", Ph.D. Thesis, Georgia Institute of Technology, September 1992.
- [35] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification", *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, pp. 788798, May 2011.
- [36] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice modeling with sparse training data", *Speech and Audio Processing, IEEE Transactions on*, vol. 13, pp. 345 354, 2005.
- [37] S. Prince and J. Elder, " Probabilistic linear discriminant analysis for inferences about identity ", *in Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 18, Oct 2007.
- [38] X. Tian, Z. Wu, X. Xiao, E. S. Chng, and H. Li, Spoofing detection under noisy conditions: A preliminary investigation and an initial database, <http://arxiv.org/pdf/1602.02950v1.pdf>, 2016.

- [39] D. B. Paul and J. M. Baker, The design for the wall street journal-based csr corpus, in *Proceedings of the workshop on Speech and Natural Language. Association for Computational Linguistics*, 1992, pp. 357 - 362.



VITA

Mustafa Caner Özbay was born in Istanbul, Turkey. After completing high school in Vefa Lisesi, he started to study Electrical and Electronics Engineering program in Dokuz Eylül University. In 2010, he started M.Sc in Electrical and Electronics Engineering department in Özyeğin University on Speech Processing at the Speech Processing laboratory.

