

SES KODLAYICILARININ KODLAMA NİTELİKLERİNE GÖRE
TEK VE İKİ KERE KODLANMIŞ SESLERDEN TESPİTİ

SAMET HIÇSÖNMEZ

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞUSTOS 2013

ANKARA

Fen Bilimleri Enstitü onayı

Prof. Dr. Necip CAMUŐCU

Müdü

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Prof. Dr. Erdoğan DOĐDU

Anabilim Dalı Başkanı

SAMET HIŐSÖNMEZ tarafından hazırlanan SES KODLAYICILARININ KODLAMA NİTELİKLERİNE GÖRE TEK VE İKİ KERE KODLANMIŐ SESLERDEN TESPİTİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Hüsrev Taha SENCAR

Tez DanıŐmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Murat ÖZBAYOĐLU

Üye : Yrd. Doç. Dr. Hüsrev Taha SENCAR

Üye : Prof. Dr. İsmail AVCIBAŐ

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Samet HİÇSÖNMEZ

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Bilgisayar Mühendisliği
Tez Danışmanı : Yrd. Doç. Dr. Hüsrev Taha SENCAR
Tez Türü ve Tarihi : Yüksek Lisans – ağustos 2013

Samet HİÇSÖNMEZ

SES KODLAYICILARININ KODLAMA NİTELİKLERİNE GÖRE TEK VE İKİ KERE KODLANMIŞ SESLERDEN TESPİTİ

ÖZET

Bu çalışmada sunulan araştırma, kodlanmış bir ses verisinin kodeğini tespit etmeye yönelik özgün bir yöntem önermektedir. Önerilen yöntem herhangi bir kod çözümü içermediği, kodlama üstverisinden faydalanmadığı ve bir kodeğin bit dizisi formatına ait bilgileri göz önüne almadığı için basit ve hızlıdır. Yöntemin temel aldığı düşünce, kodek tasarımına etki eden sıkıştırma oranı, ses kalitesi ve karmaşıklık gibi faktörlerin kodlanmış ses üzerinde etkilerini gösterecek olmasıdır. Bu etkileri açığa çıkarmak amacıyla, önerilen yöntem, kodlanmış verinin rasgele yerlerinden aldığı 2 ile 4 KB uzunluğundaki bayt dizilerinin kaotik ve rasgelelik niteliklerini analiz ederek farklı kodeklere ait kodlama süreçlerini ifade eden istatistiksel modeller kurmaktadır. Yöntemin kodek tespit etme başarımını ölçmek adına, testlerde, PSTN, GSM ve VoIP iletişim ortamlarında kullanılan konuşma kodekleri ile yüksek kaliteli ses sıkıştırma amacıyla kullanılan toplam 16 farklı ses kodlayıcı, en sık kullanılan kodlama parametreleri ile kullanıldı. Ses içeriğinin oluşturabileceği etkilerden kurtulmak için testler hem konuşma hem de müzik veri kümeleri üzerinde yapıldı. Sonuçlar incelendiğinde, bir kere kodlanmış bir ses verisinin kodeği %95'ten yüksek bir başarımla tespit edilmiştir. İkili kodlama senaryolarını da içeren birçok deney yapılmıştır. İkili kodlama test sonuçlarına göre, önerilen yöntem, iki kere kodlanmış seslerin ilk kodeğini %80'den daha yüksek bir başarımla tespit edebilmektedir. Bu testlere ait tek kısıt; ikinci kodlayıcının ilk kodlayıcıdan daha yüksek bir bit hızına sahip olması gerekmektedir.

Anahtar Kelimeler: Ses kodlayıcıları, Kodek tespit etme, PSTN, VoIP, GSM, Sıkıştırma.

University : TOBB University of Economics and Technology
Institute : Institute of Natural and Applied Sciences
Science Programme : Computer Engineering
Supervisor : Asst. Prof. Hüsrev Taha SENCAR
Degree Awarded and Date : M.Sc. – AUGUST 2013

Samet HİÇSÖNMEZ

**IDENTIFICATION OF AUDIO CODECS FROM SINGLE CODED
AND TRANSCODED AUDIOS BASED ON ENCODING
PROPERTIES**

ABSTRACT

The research presented in this paper provides a novel technique to identify the codec of an encoded audio. Since the technique does not perform decoding, utilize any coding metadata, or assume information about the structure describing the bit stream format of a codec, it is simple and fast. The main idea of the technique is that design principles of a codec such as compression level, audio quality and complexity will reveal themselves on the coded audio. To exploit this, the technique samples 2-4 kilobytes of data from a coded audio and analyzes the randomness and chaotic nature of the sampled data to build statistical models that represent encoding process associated with different codecs. To evaluate the effectiveness of the technique in identification of the codec, in experiments we use 16 of the most popular audio codecs used for high quality audio compression and in PSTNs, cellular networks, and VoIP networks by setting encoding parameters of each codec to its most commonly used values. Tests are performed on both speech and music data sets to eliminate any effect which may stem from contents of audios. Results show that the codec of a singly encoded audio can be identified with an accuracy of more than %95. Several transcoding scenarios were also performed. Those results show that the scheme can even identify the first encoder of a doubly-encoded audio with an accuracy range of around %80 to %90 or more as long as the second codec operates on higher bit rates than the first one.

Keywords: Audio codecs, Codec identification, PSTN, VoIP, GSM, Compression.

TEŐEKKÖR

Bu tez kapsamında yapılan alıőmalarda grüşlerini ve desteęini esirgemeyen tez danıőmanım Yrd. Do. Dr. Hüsrev Taha SENCAR'a ve bu tezi yazmama telkinleri ile katkıda bulunan aileme ve arkadaşlarıma teşekkür ederim.

İÇİNDEKİLER

1	GİRİŞ	1
2	TEORİK ALTYAPI	3
2.1	Konuşma Kodlama Yöntemleri	6
2.1.1	Dalga Biçimi Kodlayıcılar	6
2.1.2	Model Tabanlı Kodlayıcılar	10
2.2	Müzik Kodlama Yöntemleri	14
2.3	Çalışmada Kullanılan Kodekler	16
3	BENZER ÇALIŞMALAR	18
3.1	Alley'in Konuşma Kodeği Tespiti	18
3.2	Scholz'un Konuşma Kodeği Tespiti	19
3.3	Jenner'in Konuşma Kodeği Tespiti	20
3.4	Memon'un Dosya Tipi Tespiti	21
3.5	PinDr0p	22
4	ÖNERİLEN YÖNTEM	24
4.1	Yöntem	24

4.2	Öznitelikler	26
4.2.1	Kaotik Öznitelikler	27
4.2.2	Rasgelelik Öznitelikleri	31
5	TESTLER	36
5.1	Tekli Kodlama Testleri	37
5.2	İkili Kodlama Testleri	41
5.2.1	Karşılaştırma	44
6	SONUÇLAR VE ANALİZ DEĞERLENDİRME	50
7	GELECEK ÇALIŞMALAR	52
	ÖZGEÇMİŞ	57

ŞEKİLLERİN LİSTESİ

2.1	DPCM kodlama işlemine ait alıcı ve verici taraftaki işlem adımları	10
2.2	LPC modelinde kodlama işlemine ait akış diyagramı	12
2.3	AbS modelinde kodlama işlemine ait akış diyagramı	13
2.4	Psiko-akustik bir modele ait akış diyagramı	15
2.5	MP3 kodeğinin kodlama işlemine ait akış diyagramı	15
4.1	Sistemin çevrim dışı evredeki çalışma adımları	25
4.2	Sistemin çevrim içi evredeki çalışma adımları	26
4.3	Farklı kodekler ile kodlanmış ses örnekleri için 5. gömme boyutuna ait komşuluğun büyüklüğünün ortalama değeri	28
4.4	Farklı kodekler ile kodlanmış ses örnekleri için 5. gömme boyutuna ait komşuluğun büyüklüğünün karesinin ortalama değeri	29
4.5	Farklı kodekler ile kodlanmış ses örnekleri için ilk iterasyona ait germe etmeninin logaritmasının ortalama değerleri	30
4.6	Farklı kodekler ile kodlanmış ses örneklerine ait ortalama varyans değerleri	32
4.7	Farklı kodekler ile kodlanmış ses örneklerine ait öz-ilinti fonksiyonunun 11. katsayısının ortalama değerleri	33

4.8 Farklı kodekler ile kodlanmış ses örneklerine ait ortalama entropi değerleri	33
--	----

ÇİZELGELERİN LİSTESİ

2.1	Ses kodlayıcılarının karşılaştırması	17
3.1	Alley'in konuşma kodlayıcısı belirleme testi sonuçları	19
3.2	Scholz'un konuşma kodlayıcısı belirleme testine ait doğruluk matrisi (640 pencere)	20
3.3	Jenner'in konuşma kodlayıcısı belirleme testine ait doğruluk matrisi (160 pencere)	21
3.4	Memon'un akış içeriği belirleme testine ait doğruluk matrisi(4KB)	22
5.1	Örnekleme Boyutunun Başarım Üzerine Etkisi	38
5.2	Tüm Veri Kümeleri İçin Kodek Grup İçi Test Sonuçları	40
5.3	GSM'den PSTN ağına geçiş senaryosunda a-law kodeği kullanılan teste ait doğruluk matrisi (Müzik veri kümesi)	42
5.4	GSM'den PSTN ağına geçiş senaryosu test sonuçları	42
5.5	GSM'den VoIP ağına geçiş senaryosu test sonuçları	43
5.6	GSM'den PSTN ağına geçiş senaryosunda G.729 kodeği kullanılan teste ait doğruluk matrisi (Konuşma veri kümesi)	43
5.7	VoIP'den PSTN ağına geçiş senaryosunda u-law kodeği kullanılan teste ait doğruluk matrisi (Müzik veri kümesi)	44
5.8	VoIP'ten PSTN ağına geçiş senaryosu test sonuçları	44

5.9	[35] ile önerilen yöntemin tüm veri kümeleri üzerinde yapılan 17 sınıf sınıflandırma testleri ile ortalama başarımlar açısından karşılaştırılması	46
5.10	[35] ile önerilen yöntemin Müzik-I veri kümesi üzerinde yapılan ikili kodlama testleri ile karşılaştırılması	46
5.11	[35] ile önerilen yöntemin Konuşma-I veri kümesi üzerinde yapılan ikili kodlama testleri ile karşılaştırılması	46
5.12	Konuşma-I veri kümesi üzerinde yapılan 17 sınıf sınıflandırma testine ait doğruluk matrisi	48
5.13	Müzik-I veri kümesi üzerinde yapılan 17 sınıf sınıflandırma testine ait doğruluk matrisi	49

1. GİRİŞ

Günümüzde ses kodlaması ve kod çözümü için birçok kodlayıcı mevcuttur. Bu kodlayıcılar farklı amaçlar için kullanılmaktadır. Bu amaçların öne çıkanları olarak, yüksek kaliteli sesin ya da müziğin daha efektif saklama imkanı için sıkıştırılması, internet üzerinden ses yayını (örn.internet üzerinden yayın yapan radyolar), ve sabit telefon hatları, GSM veya VoIP (Voice over Internet Protocol) ağlarında ses iletişiminin sağlanması gösterilebilir. Belirtildiği gibi farklı amaçlarda kullanılmalarına rağmen ses kodlayıcıları, belirli sayıdaki dizayn prensiplerinde yaptıkları seçimlere göre birbirlerinden ayrılmaktadır. Bu prensiplere örnek olarak, kodlayıcının sıkıştırma ve ses kalitesi arasında gözettiği denge, kodlayıcının gürbüzlüğü ile ağ hatalarına ve gürültüye karşı duyarlılığı, ve iletişim ortamının bant genişliğindeki değişimlere gösterdiği uyum verilebilir. Bu çalışmada, ses kodlayıcılarının öncelik verdiği tasarım prensiplerinin ses üzerinde bıraktığı izler kullanılarak kodlayıcıların tespit edilmesi amaçlanmıştır.

Ses kodlamasında kullanılan kodlayıcının kodlama üstverisine (metadata) bağlı kalınmadan ve hızlı bir şekilde tespit edilmesi, mevcut bazı problemlerin çözülmesine katkı sağlayabilir. Bunlardan birincisine, günümüzde ağ trafiğinin büyük çoğunluğunu multimedya uygulamalarının alması ile beraber, ağ içindeki ses trafiğinin hızlı ve doğru karakterize edilmesinin (örn. internet yayını, dosya transferi, VoIP uygulaması) önem kazanması verilebilir [35]. Ağ trafiğinin hızlı bir şekilde sınıflandırılması, bant genişliğinin daha etkin kullanılmasına büyük oranda katkı sağlayacaktır.

Ses kodlayıcısının tespit edilmesinin büyük önem taşıdığı bir başka problem ise bir aramanın kaynağının tespit edilmesidir. Arama kaynağının tespit edilmesi spam ve dolandırıcılık gibi saldırıların önüne geçebilmek açısından son derece önemlidir

[11]. Bunun yanında arama kaynağının tespiti, reklam gibi hedefe yönelik içerik dağıtımında da önemli bir veridir. Arama kaynağının tespiti durumu için günümüzün merkezi olmayan telefon altyapısı (örn. PSTN, GSM, VoIP ağları) göz önüne alındığında, aramalar çoğunlukla birden fazla ağ üzerinden geçmektedir. Bu durumda son kodlayıcının tespitinin yanında, daha önceki kodlayıcıların belirlenmesi, aramanın hangi tür ağlardan geçtiğinin tespit edilmesinde son derece kritiktir. Örneğin, ilk önce AMR (Adaptive Multi-Rate) ses kodeği ile, daha sonra ise SILK ses kodeği ile kodlanmış bir ses verisi, bir GSM hattından bir VoIP ağına (muhtemelen Skype kullanıcısına) yapılmış bir aramayı belirtmektedir.

Bunlara ek olarak, INMD'ler (in-service non-intrusive measurement device) için ses kodlayıcılarının çabuk bir şekilde tespit edilmesi son derece kritiktir. Bu cihazlar iletişim ağlarına yerleştirilerek, konuşma kalitesini düşürebilecek konuşma ve gürültü seviyesi (speech and noise level), yankı kaybı (echo loss) ve konuşma yankı yolu gecikmesi (speech echo path delay) gibi faktörleri gözlemlemektedir [25]. Konuşmanın kalitesinde kullanılan ses kodlayıcının önemli bir etkisi olduğu göz önüne alındığında, bu bilginin INMD'lere bir ek bilgi olarak verilmesi son derece kritiktir.

Bütün bu problemler ve uygulama alanları göz önüne alındığında, ses kodlayıcısı tespit edilirken sadece elimizdeki yani çıkış terminalindeki verilerin kullanılarak bu işlemin gerçekleştirilmesi, elde edilecek bilginin etkin kullanılmasına olanak sağlayacaktır. Bunun yanında çoğu zaman elimizdeki tek bilgi iletişim ortamından alınan belli uzunluktaki bit dizisi olmaktadır. Bu bit dizilerinde kodlayıcı hakkında birçok bilgi içeren üst veriler (metadata) ise sadece belirli paketlerde, çoğunlukla bir ses dizisi için ilk paket, bulunmaktadır. Ayrıca bu pakete ulaşılsa bile üst veri bilgisine ulaşabilmek için en alt katmandan itibaren paketin çözülmesi gerekmektedir. Sunulan problemlerin gerçek zamanlı olması, çözümün de benzer şekilde olmasını gerektirmektedir. Dolayısıyla böyle bir işlem, problemlere çözüm olmaktan uzaktır.

Bu çalışmada, farklı kodlama yöntemleri kullanan kodlayıcıların ses üzerinde bıraktığı izler istatistiksel ve kaotik olarak incelenerek, kodlayıcılar tespit edilmeye çalışıldı. Önerilen yöntem, belirtilen problemlerin çözümünde hem tek başına hem de başka yöntemler ile kullanılabilir. Yöntem herhangi bir kod çözümü içermediği için gerçek zamanda sonuç verebilmektedir.

2. TEORİK ALTYAPI

Ses kodlayıcılarının hem daha rahat analiz edilebilmesi hem de kodlama sonucu oluşan ses üzerinde bıraktıkları etkilerin anlaşılabilmesi için kullanılan kodlama yöntemlerinin incelenmesi gerekmektedir. Bu kısımda öncelikle kodeklerin tasarım prensipleri hakkında genel bilgiler aktarılacak, ardından konuşma ve müzik kodlamasında kullanılan temel yöntemler hakkında detaylı bilgi verilecek, daha sonra ise çalışmada kullanılan kodeklere kısaca göz atılacaktır.

Bir ses kodlayıcısı üç tasarım prensibi göz önüne alınarak değerlendirilebilir. Bunlar kodlanmış sesin kalitesi, kodlayıcının sıkıştırma oranı ve kodlayıcı/kod çözücünün karmaşıklığıdır [13]. Pratikte ise kodek tasarımı bant genişliği kısıtları düşünülerek yapılmaktadır. Bu sebeple kodlayıcının sıkıştırma oranı ve değişen koşullara uyumu önemli bir faktördür. Burada değişen ortam koşullarına uyuma, internet bant genişliğinin düştüğü ve paket kayıplarının arttığı bir durumda kodlayıcının kendi bit hızını düşürmesi ve paketleri yedekli bir şekilde göndermesi örnek olarak verilebilir. Sıkıştırma oranı olarak da ifade edilen kodlayıcı bit hızı ya da veri hızı, örnekleme frekansının her örneği kodlamak için gereken bit sayısı ile çarpımına eşittir. Genellikle konuşma kodekleri 8 KHz, müzik kodekleri ise 44.1 KHz ya da düşük bir örnekleme frekansında çalışmaktadır. Seçilen kodlama yöntemine göre her bir örnek birkaç bit ya da bayt ile kodlanabilmektedir.

Ses kodlayıcıları müzik ya da konuşma kodlamaya yönelik olarak optimize edilmişlerdir. Müzik kodlamanın göz önünde tuttuğu temel sorun müzik dosyalarının efektif saklanmasıya yönelik yüksek kaliteli sıkıştırma iken, konuşma kodlama için gerçek zamanlı iletişim uygulamalarıdır. Konuşma ile karşılaştırıldığında müzik daha geniş bir frekans bandına sahiptir. Aynı zamanda müzik kodlayıcılarının daha yüksek bir duyarlılık sunması beklenmektedir. Bu iki durum

müzik kodlayıcılarının daha yüksek bir örnekleme frekansına ve bit hızına sahip olmalarına sebep olmuştur. İçerik dışında, iletişim ortamının bant genişliği istekleri de kodlayıcı seçimine etki eden bir başka faktördür. GSM iletişimde kullanılan kodekler, kablosuz veri iletişiminin düşük bant genişliğine sahip olması sebebiyle en düşük bit hızına sahiptirler. Buna karşın PSTN ağlarında kullanılan kodekler diğer konuşma kodekleri ile kıyaslandığında, en büyük bit hızına, en yüksek ses kalitesine ve en düşük karmaşıklığa sahiptirler. VoIP iletişimde kullanılan kodekler ise hem ses kalitesi hem de bit hızı olarak bu iki grubun arasında yer almaktadır.

Teorik olarak, kodek tasarımının en ayırt edici özelliği, kullanılan kodlama tekniği ile ilgilidir. Zaman içerisinde başarılı birçok teknik geliştirilmiş ve çok sayıda bilimsel çalışmaya konu olmuştur. Bu yöntemler, müzik ve konuşma kodlamasına göre temelde birkaç yaklaşım üzerine kurulmuştur. Genel olarak konuşma kodlayıcılar dalga formu kodlayıcıları ve model tabanlı kodlayıcılar olmak üzere iki grupta incelenebilir. Bu iki grup arasındaki temel fark, ikinci grup insan konuşma modelini göz önüne almaktadır ve bu sebeple daha yüksek bir sıkıştırma oranı sağlamaktadır. Darbe kod modülasyonu (Pulse Code Modulation - PCM) en basit ve oturmuş dalga formu kodlama tekniğidir. Bu yöntemin bir türü olan, adaptif fark darbe kod modülasyonu (Adaptive Differential PCM - ADPCM) [13], yüksek kaliteli konuşma kodlamasında sıklıkla kullanılmaktadır. En başarılı model tabanlı kodlama yöntemi olarak lineer öngörümlü kodlama (Linear Predictive Coding - LPC) [14] söylenebilir. Pratikte konuşma kodlayıcılarının büyük çoğunluğu darbe-arttırılmış lineer öngörüm (Code-Excited Lineer Prediction - CELP) [34] kullanmaktadır. Bu yöntem konuşmanın LPC modelini kullanmakta ve daha yüksek bir ses kalitesi sağlamaktadır.

Konuşma kodeklerinden farklı olarak, konuşma dışındaki sesleri kodlamada kullanılan temel yöntem, zaman frekans eşleştirmesi yaparak ses üzerindeki tekrarları ortadan kaldırma üzerine kurulmuştur [13]. Sinyali tam olarak tekrar oluşturabilme yeteneğine sahip oldukları için blok tabanlı dönüştürme teknikleri ses kodlaması için önerilmiştir. Bunlar arasında çok yüksek bir kodlama oranı sağladığı için modifiye edilmiş ayrık kosinüs dönüşümü (Modified Discrete Cosine Transform - MDCT) en yaygın olanıdır. Bu sebeple MDCT, MP3, AAC, OGG, WMA ve AC-3 gibi popüler kodeklerde kullanılmaktadır. Müzik ve konuşma kodlama için belirtilen yöntemler 2.1 ve 2.2 kısımlarında daha detaylı

anlatılacaktır.

Ses kodlayıcılarını birbirinden ayıran bir başka özellik ise kodlanmış sesin, kodlanmamış orijinal haline kıyasla sahip olduğu kalitedir. Ses kodlamasının, müzik ve konuşma kalitesi üzerindeki etkilerini tespit edebilmek amacıyla birçok nesnel ve öznel ses kalitesi değerlendirme yöntemleri önerilmiştir. En kapsamlı kullanılan öznel konuşma kalitesi değerlendirme ölçütü ortalama karar puanıdır (Mean Opinion Score - MOS) [28]. Bir sesin MOS puanı 1 ile 5 arasında değer almaktadır ve 4 puan ile üzeri sabit telefon ses kalitesi (toll quality) olarak tanımlanmıştır. Öznel yöntemleri nesnel yöntemler ile değiştirmek amacıyla, ses kalitesinin algısal değerlendirmesi (Perceptual Evaluation of Audio Quality - PEAQ) [18] adlı yöntem öne sürülmüştür. PEAQ yönteminin çıktısı nesnel fark puanı (Objective Difference Grade - ODG) olarak ifade edilmekte ve -4 ile 0 arasında değer almaktadır. Burada yüksek puan yüksek ses kalitesini ifade etmektedir.

Kodeğin karmaşıklığı, MIPS (millions of instructions per second) olarak hesaplanmakta, kodlayıcının gecikmesi ise toplam algoritmik gecikme olan ses örneklerinin kodlama ve kod çözümü için geçen toplam süre olarak ifade edilmekte olup kodekleri değerlendirirken karşımıza çıkan diğer faktörlerdir. Burada gecikme faktörünün gerçek zamanlı iletişim için çok kritik olduğu göz önüne alınmalı ve bu sebeple GSM, VoIP ve PSTN kodeklerinin düşük gecikme değerine sahip olması gerekmektedir.

Kodlayıcı tasarımına etkisi olan tüm bu faktörler direk ya da dolaylı yoldan kodlanmış ses üzerinde kendilerini göstermektedir. Bütün faktörler arasında kodlama tekniği, sıkıştırma oranını, bit hızını ve ses kalitesini belirlediği için en ciddi etkiye o sahip olacaktır. Karmaşıklık ve gecikme daha çok modüllerin kodlanması ve kod çözümü ile ilgili olduğu için, kodlanmış ses üzerindeki etkileri sadece bit dizisi pencereleme ve formatlama işlemleri ile sınırlı kalacaktır. Bu tezin temeli, bu etkilerin tutarlı olacağı, sesin içeriğine bağlı olmayacağı ve kodlanmış sesin istatistiksel nitelikleri ile modellenebileceği üzerine kurulmuştur.

2.1 Konuşma Kodlama Yöntemleri

Konuşma kodlamada sıklıkla kullanılan yöntemler bu kısımda fazla detaya girmeden açıklanacaktır. Bunlardan ilk ikisi PCM tabanlı kodlama yöntemleri olan Companding ile DPCM/ADPCM'dir. Bu yöntemler insan konuşma modelini göz önüne almadan örnekleme, kuantalama ve kodlama işlemlerini yapmaktadırlar. İkinci grup yöntemler ise insan konuşma modelini göz önüne alarak ses kalitesinden az miktar feragat ile yüksek oranda sıkıştırma elde etmektedirler. Bu yöntemler Lineer Öngörümlü Kodlama ve Sentezden-Analiz (Analysis-by-Synthesis - AbS) olarak ana iki grupta incelenebilir. İnsan konuşma modelini temel alan birçok kodlama yöntemi olmasına rağmen çoğunlukla kullanılan iki tanesi burada açıklanacaktır.

2.1.1 Dalga Biçimi Kodlayıcılar

Bir sinyali analogtan dijitale dönüştürürken yapılan ilk işlem sinyalin içindeki yüksek frekans elemanlarının filtrelenmesidir. Konuşma sinyallerinin çoğunluğunun 200-300 Hz ile 2700-2800 Hz. arasında olduğu göz önüne alındığında yaklaşık 4000 Hertz filtre bant genişliği için yeterli olacaktır. Tabiki bu değerler dar bant konuşma (narrowband speech) için geçerlidir. Geniş bant konuşma kodları için frekans aralığı 50 ile 7000 Hz. arasında olmaktadır. Filtreleme işleminden sonraki adım filtrelenmiş sinyalin belirli bir frekansta örneklenmesidir. Örnekleme işleminin frekansı ise Nyquist Teoremi ile belirlenmektedir. Analog sinyaldeki en büyük frekans değerinin iki katından büyük bir frekans ile örnekleme yapılması gerektiği için dar bant için bu değer 8 KHz. olarak seçilirken, geniş bant kodları için ise bu değer 16 KHz. olarak belirlenmiştir.

Bu işlemden sonraki son adım ise kuantalama ve kodlama adımıdır. Kuantalama işlemi, örneklenen her bir analog sinyali bir sayısal değere dönüştürme sürecidir. Tek tip kuantalamada (uniform quantization) analog sinyalin genlik spektrumunu eşit aralıklara bölünmüştür. Kuantalama işlemine giren her bir örnek için genlik değeri bu kuantalama aralıklarından hangisine en yakın ise alacağı değer o aralık için belirlenen sayısal değer olacaktır. Burada bir sinyal kendi genliğine tam olarak eşit olmayan bir aralık ile eşleştirildiğinde, bu durum birçok örnek için

geçerlidir, PCM sürecine gürültü karışmış olmaktadır.

Ses kalitesine etki eden en önemli faktör Sinyal Gürültü Oranı (Signal to Noise Ratio - SNR) olarak tanımlanan, sinyalin gücünün gürültünün gücüne oranı şeklinde hesaplanan bir değerdir. Kuantalama gürültüsünü düşürmenin bir yolu, kuantalama aralıklarını arttırmaktır. Kuantalama aralıkları arttığında sinyalin genliği ile kuantalama aralığı arasındaki fark azalacağı için gürültü miktarı da azalacaktır. Ancak bu durumda kodlama için kullanılacak bit sayısının da artırılması gerekmektedir. Tek tip kuantalama ile ilgili en önemli sorun, kuantalama aralıklarının eşit seçilmesidir. Bu durum düşük genlikteki sinyallerin daha küçük, yüksek genliktekilerin ise daha büyük bir SNR değerine sahip olmalarına neden olmaktadır. İnsan konuşması ile üretilen sinyallerin bir çoğunun düşük genlikte olduğu göz önüne alındığında, yüksek genlikli seslerin kalitesini arttıran bir yöntem son derece verimsiz olacaktır. Bunun yanında insan işitme sistemi düşük genliklerdeki ufak dalgalanmalara, yüksek genlikteki dalgalanmalara göre daha duyarlıdır. Düşük genliklerdeki ses kalitesini arttırmak amacıyla tek tip kuantalama yerine düzensiz (non-uniform) kuantalama işlemleri kullanılmaktadır. Bunlardan ilki companding adı verilen işlemidir.

2.1.1.1 Companding

Companding sinyalin öncelikle kaynakta sıkıştırılması (compression), daha sonra çıkış terminalinde ise ilk haline genişletilmesi (expanding) işlemidir [17]. Companding kelimesi, bu iki işlemi ifade eden terimlerin birleştirilmesi ile oluşturulmuştur. Sıkıştırma adımında sinyaller logaritmik bir sıkıştırma fonksiyonuna tabi tutulurlar. Örneğin, kaynakta ADC ile 14 bit hassasiyetinde örneklenen bir analog sinyal, bir ara logaritmik fonksiyon ile 8 bite düşürülmektedir. Logaritmik ölçekleme fonksiyonu düşük genlikli örneklerin, yüksek genlikli olanlara nazaran daha yüksek hassasiyete sahip olmasına imkan sağlamaktadır. Bir başka ifade ile genliği büyük sinyaller daha çok sıkıştırılmakta, genliği küçük olanlar ise daha az sıkıştırılmaktadır. Bu sayede, düşük genlikli örneklerin sahip olduğu gürültü miktarı da azalmakta ve aynı zamanda SNR bütün örnekler için sabit bir seviyeye çekilmiş olmaktadır.

Companding ITU-T G.711 kodekleri olan a-law ve u-law'da kullanılmaktadır.

Her iki G.711 kodeđi, 8 KHz örnekleme frekansı ile alıřmakta ve örneklenmiř sinyalleri 8 bit ile ifade etmektedir. Bunun sonucunda her iki kodek te 64 kbps kodlama bit hızına sahip olmaktadır. A-law 13 bit hassasiyetindeki örneklere girdi olarak almakta ve çođunlukla Avrupa kıtasında kullanılmaktadır. U-law ise 14 bit hassasiyetindeki örneklere alıp bunları 8 bite düşürmekte ve çođunlukla Amerika ve Japonya'da kullanılmaktadır. Burada a-law ve u-law kullanan iki farklı grup iletişime geeceđi zaman kullanılan ortak standart a-law kodeđidir ve bu dönüşümden u-law kullanan taraf sorumludur. Bu iki kodeđin sıkıřtırma oranları düşük olsa da kodlama ve kod çözümünün son derece basit olması (karmařılıđı çok düşük) ve kodlama sonucu oluřan ses kalitesinin çok yüksek olması bu kodeklerin halen kullanılıyor olmalarının temel sebepleridir. A-law kodeđine ait sıkıřtırma ve genişletme fonksiyonları 2.1 ve 2.2'de verilmiřtir [23]. 2.1 ve 2.2 fonksiyonlarındaki A deđeri 87.7 (Avrupa) olarak seđilmektedir [23].

$$F(x) = (x) \begin{cases} \frac{A|x|}{1 + \ln(A)}, & |x| < \frac{1}{A} \\ \frac{1 + \ln(A|x|)}{1 + \ln(A)}, & \frac{1}{A} \leq |x| \leq 1, \end{cases} \quad (2.1)$$

$$F^{-1}(y) = (y) \begin{cases} \frac{|y|(1 + \ln(A))}{A}, & |y| < \frac{1}{1 + \ln(A)} \\ \frac{\exp(|y|(1 + \ln(A)) - 1)}{A}, & \frac{1}{1 + \ln(A)} \leq |y| < 1. \end{cases} \quad (2.2)$$

U-law kodeđine ait sıkıřtırma ve genişletme fonksiyonları ise 2.3 ve 2.4'te verilmiřtir [23]. 2.3 ve 2.4 fonksiyonlarındaki μ deđeri 255 (Amerika ve Japonya) olarak seđilmektedir [23].

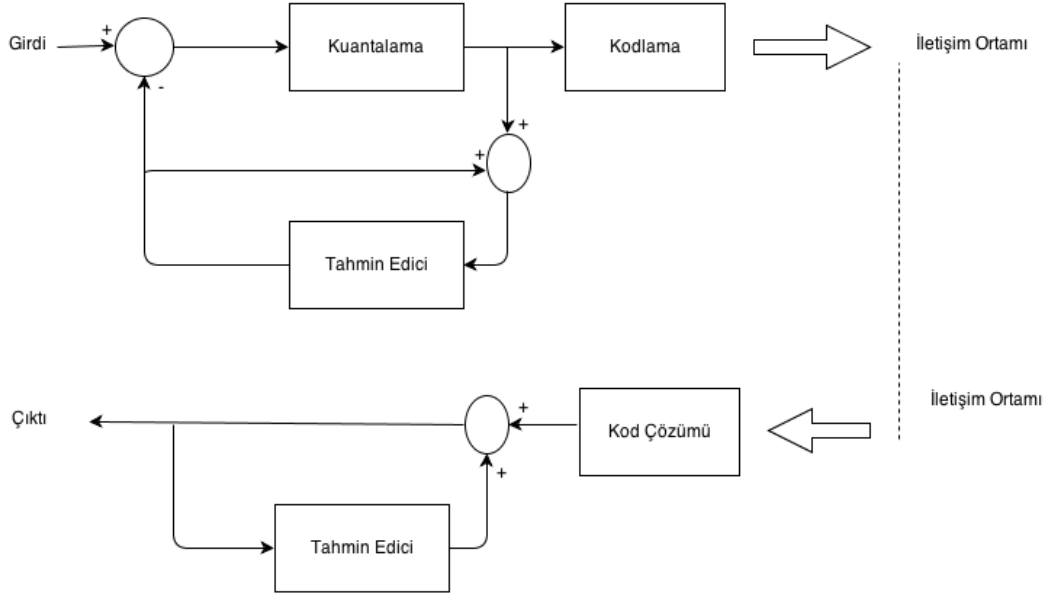
$$F(x) = \text{sgn}(x) \frac{\ln(1 + \mu|x|)}{\ln(1 + \mu)} \quad -1 \leq x \leq 1 \quad (2.3)$$

$$F^{-1}(y) = \text{sgn}(y)(1/\mu)((1 + \mu)^{|y|} - 1) \quad -1 \leq y \leq 1 \quad (2.4)$$

2.1.1.2 DPCM/ADPCM

Çoğunlukla konuşma esnasında bir önceki ses sinyali ile bir sonraki arasındaki fark son derece küçük olmaktadır. Diferansiyel PCM olarak tanımlanan DPCM, tüm sinyali göndermek yerine bu ufak farkı göndererek bant genişliğini düşürmeyi amaçlamaktadır [17]. İki sinyal arasındaki farkın, sinyalin kendi genliğinden her zaman ufak olacağı aşıkardır. Bu fark iki farklı şekilde hesaplanabilir. Bunlardan ilki ve en basiti, sinyalin şu anki değerini elimizde bulunan bir önceki örneğin değerinden çıkartmaktır. Bu aradaki fark kuantalama işlemine sokulur ve PCM işlemine göre daha az bit ile ifade edilebilir. Ancak bu işlem gürültü anlamında son derece verimsizdir. Bunun ilk sebebi kuantalama işlemindeki gürültünün ardışık örnekler için sürekli büyümesidir [8]. Herbir sinyalin kendi değerinin kuantalandığı durumda, bu değer önceki örneklerin üzerindeki gürültüden etkilenmemektedir. Ancak DPCM işleminde, sinyalin alacağı değer önceki örneklere bağlı olacağı için gürültünün sürekli artması ihtimali mevcuttur. Buna ek olarak iletim ortamının sinyale ekleyeceği gürültünün de benzer sebepler ile ardışık örnekler için sürekli artması ihtimali de vardır.

Burada kullanılan ikinci yöntem bu iki gürültüyü azaltma ya da ortadan kaldırmak amacıyla bir tahmin algoritması kullanmaktır. Bu tahmin algoritması ile eldeki sinyal ile bir önceki sinyalin farkı yerine, eldeki sinyal ile tahmin algoritmasının ürettiği değer farkı alınarak, bu değer karşı tarafa iletilmektedir. Burada belirli sayıdaki geçmiş sinyal üzerinden şu anki sinyalin alabileceği değer doğrusal tahmin (linear prediction) ile belirlenmeye çalışılmaktadır [17]. Bu sayede bir sinyal üzerinde oluşan kuantalama gürültüsü sonraki örneklere etki etmemekte sadece o sinyali etkilemektedir. İletim ortamının sisteme ekleyeceği gürültüyü düşürmek amacıyla bu yönteme ek olarak, sinyal daha küçük alt parçalara ayrılır ve her parçanın gönderimi tamamlandıktan sonra alıcı ve verici taraf sıfırlanır. Bu sayede iletim ortamının gürültüsü sadece ufak parçalar içinde etkisini gösterir ve gürültünün sürekli büyümesi de engellenmiş olur [8]. Bu tahmin algoritmalı DPCM işlemine ait verici (encoder) ve alıcı (decoder) taraftaki blok diyagramlar Şekil 2.1'de verilmiştir.



Şekil 2.1: DPCM kodlama işlemine ait alıcı ve verici taraftaki işlem adımları

DPCM üzerine geliştirilen Adaptif DPCM yöntemi, fark sinyaline göre kuantalama aralıklarını değiştirmektedir. Eğer fark sinyalinin değeri küçük ise kuantalama aralıkları arttırılmakta, fark sinyali büyük ise kuantalama aralıkları düşürülmektedir. Yöntem kuantalama aralıklarını giriş fark sinyaline göre adaptif bir şekilde ayarlamaktadır. Bu sayede SNR değeri de tüm sinyal için sabit bir değerde tutulmaya çalışılmıştır. G.726 kodlayıcısı bu yöntem ile çalışmakta ve 16, 24, 32 ve 40 kbps bit hızlarında çalışabilmektedir.

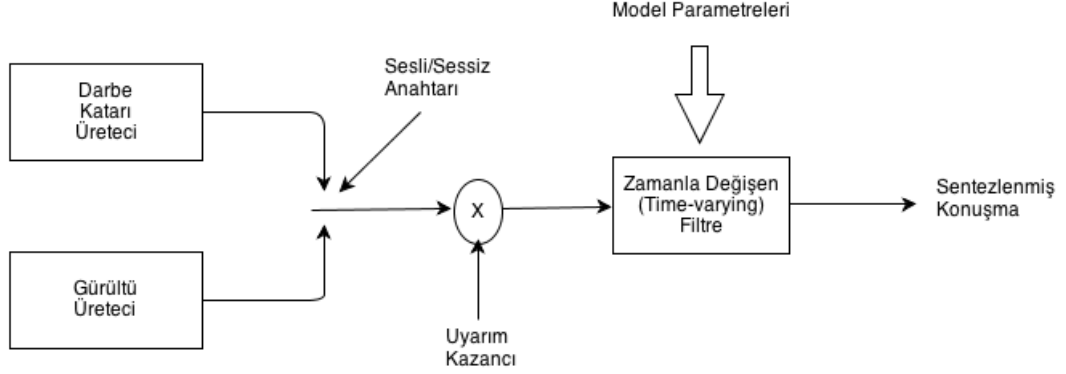
2.1.2 Model Tabanlı Kodlayıcılar

Model tabanlı kodekler (vocoders), dalga biçimli kodlayıcılar gibi giriş sinyalini alıcı tarafta tekrardan oluşturmak yerine, bu giriş sinyalini alıcı ve verici tarafın bildiği bir konuşma modeli ile ifade etmeye çalışmaktadır. Bu kodekler her örneklenmiş sinyali ayrı ayrı kodlamak yerine, belirli uzunluktaki (çoğunlukla 20 ms) bir pencereyi (frame) analiz ederek, bu pencere içindeki sesi tekrardan sentezleyebilecek model parametrelerini bulmaktadır. Uygun model parametreleri bulunduktan sonra ise kod çözücü tarafa sadece bu parametreler iletilir. Alıcı tarafta da alınan bu parametreler bir ses sentezleyicisine (speech synthesizer) verilerek orijinal sese benzer bir ses üretilmektedir. Bu yöntem ile büyük

oranda sıkıştırma elde edilmektedir. Ancak bu yöntemlerin karmaşıklığı fazladır. Örneğin Tablo 2.1'den görüleceği üzere dalga biçimli kodlayıcılar olan a-law ve u-law kodeklerinin MIPS değeri 0.01 iken, vocoder kodlayıcılarından G.729'un MIPS değeri 20 olarak hesaplanmaktadır. Ayrıca dalga biçimli kodlayıcılar ses içeriğinden bağımsız oldukları için müzik kodlamada da kullanılabilirlerdir. Buna karşın model tabanlı kodekler bir insan konuşma modelini baz aldıkları için konuşma dışındaki seslerin, müzik gibi, kodlanmasında yetersiz kalmaktadır.

İnsan sesini modelleyen birçok yöntem mevcuttur. Bunlardan en çok kullanılanı kaynak filtre (source filter) modelidir. Bu modelin popüler olması basit ve başarılı olmasından kaynaklanmaktadır. Diğer insan sesi sentezleme mekanizmaları gibi bu model de bir uyarım kaynağı (excitation source) ve bir spektral zarf filtesinin (spectral envelope filter) birleşiminden oluşmaktadır [25]. Sesli konuşma (voiced speech) için uyarım sinyali, analiz edilen konuşma penceresinin ses seviyesine (pitch) eşit periyotta bir gırtlak darbesi benzetimidir. Sessiz konuşma için ise uyarım sinyali bir gürültü kaynağından, çoğunlukla beyaz gürültü (white noise), gelmektedir. Daha sonra ise bu uyarım sinyalleri, orijinal sinyalin spektral zarfına benzer şekilde cevap veren filtreden (örn. vocal tract filter) geçmektedir. Burada uyarım sinyali ve filtre iyi seçildiğinde bu model son derece kaliteli ses üretebilmektedir.

Kaynak filtre modelini kullanan en önemli uygulamalardan bir tanesi doğrusal öngörümlü kodlama üzerine kuruludur. LPC vocoder, uyarım kaynağı olarak sesli konuşma için bir darbe katarı (impulse train), sessiz konuşma için ise rastgele gürültü kullanmaktadır. Konuşmanın temel niteliklerinden olan dudak yayılması, bazı geniz ve burun hareketlerini göz önüne alan birçok başarılı konuşma modeli geliştirilmiş olsa da, bu yöntemde kullanılan model zamanla değişen tüm kutup filtre (time-varying all-pole filter) şeklinde basitleştirilmiştir [25][14]. Basitleştirilmiş bir LPC modeli Şekil 2.2'de verilmiştir.



Şekil 2.2: LPC modelinde kodlama işlemine ait akış diyagramı

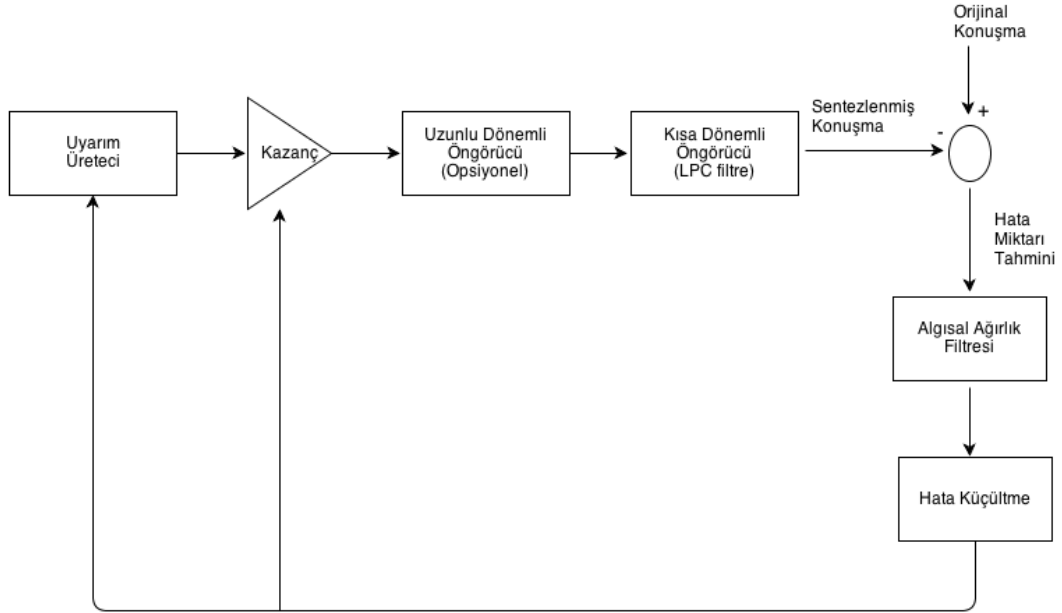
Şekil 2.2’de de görüldüğü üzere, kod çözücü tarafa filtre katsayıları, ses seviye periyodu (pitch period), uyarım kazancı ve sesli/sessiz bilgisi aktarılması gereken bilgilerdir. Bu bilgilerin her ses penceresi için iletilmesi yeterli olacaktır. Bu sayede LPC kodlayıcıları son derece düşük bit hızlarında kodlama yapabilmektedir.

LPC kodekleri çok yüksek sıkıştırma oranlarında çalışmalarına rağmen iletişim ortamlarında nadiren kullanılmaktadırlar. Bunun sebebi sadece LPC üzerine kurulu kodlayıcıların düşük kalitede ses üretmesidir. Bu kodlayıcıların ürettiği ses, uyarım sinyalinin darbe katarı ile tanımlanmasının yetersizliğinden dolayı uğultulu olmaktadır. Bu sebeple LPC kodekleri ses kalitesinin çok önemli olmadığı, ama bant genişliğinin kritik olduğu alanlarda, çoğunlukla askeri iletişimde, sıklıkla kullanılmaktadır. LPC modelini geliştiren sentezden-analiz yöntemi popüler kodeklerin birçoğunda kullanılmaktadır.

Bu yöntemde, kodlayıcı taraf hesapladığı model parametreleri ile yeni bir konuşma sentezlemektedir. Daha sonra ise sentezlediği ses ile orijinal sesi karşılaştırmakta ve hata oranı minimuma gelene kadar model parametrelerini düzeltmektedir. En uygun model parametreleri bulunduğu anda ise bu değerler kod çözücü yani alıcı tarafa iletiminde ve düşük hatalı ses tekrardan sentezlenmektedir. Yöntemin adının sentezden-analiz olması, kodlayıcı tarafın da alıcı taraf gibi ses sentezlemesi ve uygun parametre değerlerini ürettiği sesi analiz ederek hesaplamasından dolayıdır.

Bu kodlayıcıların gerçek zamanlı çalışmaları gerektiği düşünüldüğünde, bütün

model parametre değerlerinin test edilmesi mümkün değildir. Bu yöntemde model parametreleri yerine uyarım sinyalleri test edilmektedir. LPC kodlamasında kullanılan modelin başarılı olduğu ancak uyarım sinyallerinin kötü modellendiği belirtilmişti. Bu sebeple sentezden-analiz yönteminde LPC modeli aynı şekilde kullanılmakta, ancak uyarım sinyali, hata oranı minimuma düşürülene kadar değiştirilmektedir. Hatta uyarım sinyali şekil (shape) ve kazanç (gain) olarak iki parçaya ayrılabilir. Burada kazanç değeri, şekilden bağımsız olarak hesaplanabilmektedir ve bu sayede sentezden-analiz adımında kazanç değerinin değiştirilmesi yeterli olacaktır. Sentezden-analiz yönteminin blok diyagramı Şekil 2.3'te verilmiştir [25].



Şekil 2.3: AbS modelinde kodlama işlemine ait akış diyagramı

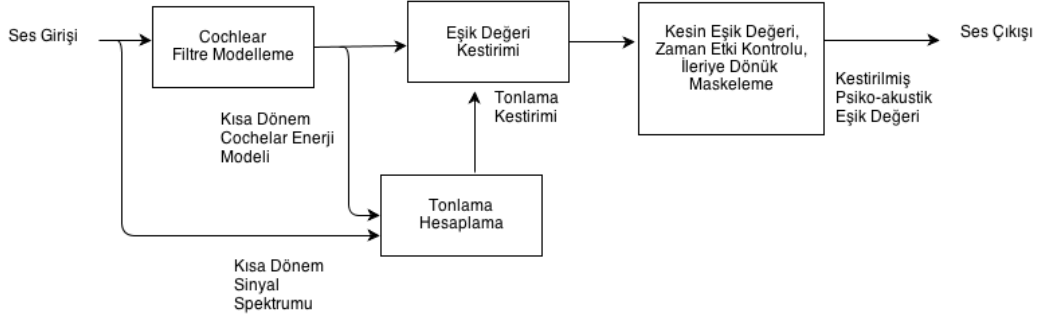
Sentezden-analiz yönteminin en başarılı uygulamalarından bir tanesi CELP (Code-excited linear prediction) kodlama yöntemidir. Günümüzde GSM ve VoIP ağlarında kullanılan birçok kodlayıcı Tablo 2.1'den de görüleceği üzere CELP ve ACELP (Algebraic CELP) yöntemlerini kullanmaktadır.

2.2 Müzik Kodlama Yöntemleri

Konuşma kodeklerinde temel amaç insan konuşmalarını, ses kalitesini belirli bir seviyenin altına düşürmeden en yüksek miktarda sıkıştırmaktır. Bu amaçla temel aldıkları yöntemler çoğunlukla bir insan konuşma modelinin gerçekleşmesi şeklindedir. Müzik ya da yüksek kaliteli ses kodlayıcılarının amacı da benzer şekilde hem yüksek sıkıştırma hem de yüksek ses kalitesi sağlamaktır. Ancak buradaki ses kalitesi bir konuşmadakinden farklıdır. Bu kodlayıcılar algısal kodlama (perceptual coding) olarak ta tanımlanan psiko-akustik (psychoacoustic) modeller kullanmaktadır. Bu modeller insan işitme sisteminin çalışmasını baz almaktadır.

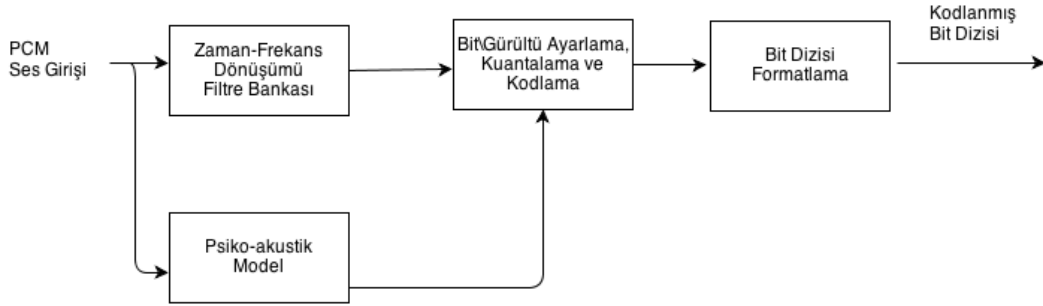
Burada algısal modelleri anlayabilmek için iki kavramın açıklanması gerekmektedir. Bunlardan ilki olan gereksizlik (redundancy), insan işitme limitlerinin dışında kalan frekansların atılmasını ifade etmektedir. CD kalitesi olarak ifade edilen ses kayıtları bu sebeple 44.1 KHz örnekleme frekansına sahiptir. İnsan işitme sistemi 20 KHz'dan yüksek frekanstaki sesleri duyamadığından 22.05 KHz üzerindeki sesler atılmaktadır. İkincisi ise yersizlik (irrelevancy) olarak tanımlanmaktadır. İnsan işitme sınırları içerisinde yer alan ancak ortam şartları vb. sebepler ile birçok insan tarafından algılanamayan bazı seslerin kodlama esnasında tamamen atılması ya da daha az bit ile ifade edilmesini belirtmektedir.

Algısal kodlamanın en önemli adımlarından bir tanesi, irrelevancy konsepti ile ilgili olarak, maskeleyedir. Maskeleye insan işitme sisteminin aynı anda gerçekleşen bazı seslere diğerlerinden daha fazla öncelik vermesi olarak tanımlanabilir. Örneğin bir müzik grubunun provasında davul çalınmaya başlayana kadar elektro gitar baskın sestir. Ancak davul çalınmaya başladığında gitarın sesi arka planda kalmaya başlamaktadır. Maskeleye algısal kodlama kullanan kodlayıcıların çok yüksek oranda sıkıştırma, MP3 için 11 kat, yapabilmelerine imkan vermektedir. Psiko-akustik bir modele ait akış diyagramı Şekil 2.4'te verilmiştir [2].



Şekil 2.4: Psiko-akustik bir modele ait akış diyagramı

Bu kısımdaki kodeklerin hemen hemen hepsi zaman-frekans eşleştirimi (time-frequency mapping) kullanmaktadır. Bu işlemi yaparken kullanılan dönüşüm çoğunlukla MDCT olmaktadır. MDCT'nin bu kadar popüler olmasının sebebi ardışık blokların belirli kısımlarının örtüşmesi ile bloklar arasında bozuklukların (artifacts) oluşmamasıdır [13]. Mp3 kodlamasına ait bir akış şeması Şekil 2.5'te verilmiştir [13].



Şekil 2.5: MP3 kodeğinin kodlama işlemine ait akış diyagramı

Bu şema incelendiğinde, PCM formatındaki ses dosyaları MDCT filtre bankasına ve psiko-akustik modele aynı anda girmektedir. Bir yandan zaman-frekans dönüşümü yapılırken bir yandan da ses dosyalarının akustik nitelikleri incelenmektedir. Bu işlemlerden sonra kuantalama ve kodlama gelmektedir. Burada kullanılan kodlama opsiyonlarına uygun şekilde örnekleme ve bit atama yapılmaktadır. MP3 ve diğer yüksek kaliteli ses kodlayıcıları farklı örnekleme ve bit hızlarında kodlama yapabilmektedir.

2.3 Çalışmada Kullanılan Kodekler

Çalışmada kullanılan kodekleri belirlerken bu kısımda anlatılan farklı kodlama yöntemlerine sahip ve farklı alanlarda ya da iletişim ortamlarında kullanılan popüler kodlayıcılar seçilmeye çalışıldı. Bu kodlayıcıları, PSTN, GSM ve VoIP iletişim ortamlarında ve yüksek kaliteli ses sıkıştırmasında kullanılan kodekler olmak üzere dört grupta inceleyebiliriz. Bu kodeklere ait ayırıcı nitelikler Tablo 2.1'de verilmiştir. Tablonun ikinci sütunu kodlayıcının en çok kullanılan bit hızını, üçüncü sütun ise ideal koşullar altında ses kalitesini MOS (konuşma kodekleri için) ve ODG (müzik kodekleri için) skorları olarak göstermektedir. Dördüncü sütun kodlayıcının kullandığı kodlama tekniğini, beşinci sütun da kodlayıcının kompleksliğini MIPS değeri olarak belirtmektedir. Son sütun ise varsa kodlayıcının gecikme zamanını ifade etmektedir.

Çizelge 2.1: Ses kodlayıcılarının karşılaştırması

Kodek Grubu	Kodek	Tanımlı Bit hızı (Kbps)	MOS ¹ ODG ²	Kodlama Tek.	MIPS	Gecikme (ms)
PSTN	a-law[6]	64	4.44	PCM	0.01	0.125
	u-law	64	4.45	PCM	0.01	0.125
	PCM	32	N/A	ADPCM	~ 0	~ 0
GSM	AMR [19]	12.2	4.14	ACELP	20	25
	AWB[24]	12.65	4.20	ACELP	40	25
	GSM[20]	13	3.5	RPE-LTP	5	20
	GSM (WAV)	18	3.9	RPE-LTP	6	20
VoIP	G.729[27]	8	4.1	CS-ACELP	20	15
	G.726[37]	32	4.3	ADPCM	2	0.125
	iLBC[10]	13.33	4.1	LPC	18	40
	Speex[4]	22	3.84	CELP	40	30
Yüksek Kaliteli	AAC[15]	128	-0.975	MDCT	N/A	N/A
	MP3[15]	128	-1.179	MDCT	N/A	N/A
Sıkıştırma	OGG	128	-0.485	MDCT	N/A	N/A
	FLAC	kayıpsız	N/A	Linear Pred.	N/A	N/A
	WMA	128	-0.661	MDCT	N/A	N/A

¹ MOS değerlerinin alındığı adres

http://www.vocal.com/speech_coders/psqm_data.html

² ODG değerlerinin alındığı adres [18]

3. BENZER ÇALIŞMALAR

Her ne kadar ses kodlama uzun bir geçmişe sahip olsa da bir sesin kodlamasında kullanılan kodlayıcının belirlenmesine yönelik sınırlı sayıda çalışma mevcuttur. Bu kısımda bu çalışmaya ilham kaynağı olmuş birkaç çalışma hakkında bilgi verilecektir. Bu çalışmaların bazıları birincil hedef, bazıları ise yardımcı adım olarak ses kodlayıcısını tespit etmeyi amaçlamaktadır.

3.1 Alley'in Konuşma Kodeği Tespiti

Ses kodlayıcısı belirleme alanında yapılan ilk çalışma, Alley tarafından bir telefon kanalındaki kodlama tipini bulmaya yönelik olarak gerçekleştirilmiştir [9]. Metod, daha sonra yapay sinir ağlarını eğitmek için kullanılmak üzere, iletişim ortamına adaptif en küçük ortalama kareler filtresi (least mean squares adaptive filter) yerleştirilerek filtre katsayılarından istatistikler elde etmektedir. Kullanılan nitelikler olarak en büyük filtre katsayısının varyansı, giriş sinyal gücü ve filtre hatasının olasılık dağılım histogramı seçilmiştir. Çalışmada kullanılan yapay sinir ağı modeli ise çok katmanlı perceptron yapısıdır.

Tablo 3.1'de yapılan testlere ait sonuçlar verilmiştir. Sonuçlar incelendiğinde, farklı öğrenme oranları için birçok test yüksek başarımlı göstermiştir. Ancak bu çalışmanın en büyük eksiği sınırlı sayıda ses kodlayıcısının kullanılmış olmasıdır. Burada çalışmanın yapıldığı dönemdeki (1993) en büyük ve belkide tek iletişim altyapısının sabit telefon hatları (ilk GSM araması tarihi 1991) olduğu göz önüne alınmalıdır. Bunun yanında böyle bir sınıflandırma için yapay sinir ağlarından ziyade daha basit sınıflandırıcılar ile de benzer sonuçlar elde edilebilirdi.

Çizelge 3.1: Alley'in konuşma kodlayıcısı belirleme testi sonuçları

Öğrenme Oranı	Kanal Tipi	Başarım(%)
0.025	Linear	95
	A-law	86
	ADPCM	91
0.050	Linear	97
	A-law	86
	ADPCM	92
0.075	Linear	91
	A-law	83
	ADPCM	97
0.100	Linear	94
	A-law	83
	ADPCM	87

3.2 Scholz'un Konuşma Kodeği Tespiti

Konuşma kodlayıcısı tespit etme alanında farklı iletişim ortamlarına ait kodlayıcıları içeren önemli çalışmalardan biri de Scholz'a aittir [33]. Scholz'un çalışmasında kullandığı tespit yöntemi çok bandlı uyarım (multi-band excitation) ses modeli içerisinde geliştirilmiş harmonik ve gürültü ayrıştırımına (harmonic-plus-noise decomposition) dayanmaktadır [21].

Bu çalışmada öncelikle ses örneklerinin sesli kısımları bulunmuş ve bu sesli kısımlar (voiced frames) harmonik ve gürültü ayrışımına tabi tutulmuştur. Bu işlem sonucunda iki farklı spektrum ortaya çıkmıştır. Bunlardan ilki sadece sesli parçalardan oluşan harmonik spektrum ve gürültü spektrumu olarak adlandırılan ses örneklerine ait konuşma spektrumu ve harmonik spektrum arasındaki farklılıkları ortaya koyan ikinci bir spektrumdur. Bu ikinci spektrumun da büyük oranda giriş ses örneklerine bağlı olduğu öngörülmektedir.

Bu öngörüden yola çıkarak farklı konuşma kodlayıcılarına ait ses örnekleri belirtilen işlemlerden geçirilmiş ve bir eğitim kümesi oluşturulmuştur. Aynı şekilde test edilmek istenen konuşma örneklerinden de gürültü spektrumu çıkarılmış ve çıkartılan gürültü spektrumu daha önce eğitilmiş kümedeki gürültü spektrumlarına en çok benzeyen ile eşleştirilmiş, bu şekilde kodlayıcı tespiti

yapılmıştır. Buradaki eşleştirme normalize edilmiş çapraz ilinti (normalized cross-correlation) ile yapılmıştır.

Farklı uzunluktaki ses örnekleri üzerinde deneyler yapılmıştır. Yaklaşık 30 sn. (640 pencere) uzunluğundaki ses örnekleri üzerinde yapılan testlere ait doğruluk matrisi Tablo 3.2’de verilmiştir. Her ne kadar bazı kodlayıcılar yüksek başarımla tespit edilse de ortalama başarımın %80’den düşük olduğu görülmektedir. Özellikle EFR (Enhanced Full Rate) ve ADPCM (G.726) kodekleri diğerleri ile ciddi oranda karıştırılmıştır. Bunun yanında 30 saniyelik bir ses örneği üzerinde belirtilen işlemlerin yapılması da son derece maliyetlidir.

Çizelge 3.2: Scholz’un konuşma kodlayıcısı belirleme testine ait doğruluk matrisi (640 pencere)

Ses Kodlayıcısı		Sınıflandırma Sonucu						
		ADPCM	AMR	EFR	G.723.1	G.729	HR	G.711
ADPCM	16kbit/s	%100	-	-	-	-	-	-
	24kbit/s	%95.35	-	%3.49	%1.16	-	-	-
	32kbit/s	%74.12	-	%0.32	-	%3.19	-	%22.36
	40kbit/s	%16.82	-	%0.47	-	%8.88	-	%73.88
AMR	4.75kbit/s	-	%92.41	-	-	%3.63	-	%3.96
	5.9kbit/s	-	%88.44	-	-	%2.51	-	%9.05
	10.2kbit/s	-	%82.01	-	-	%5.02	-	%12.97
EFR		%7.06	%28.24	%51.76	%7.06	-	-	%5.88
G.723.1	6.3kbit/s	-	-	%19.32	%77.27	-	%3.41	-
HR		-	-	-	%15.93	-	%84.07	-
G.711	u-law	-	-	-	-	%9.64	-	%90.36

3.3 Jenner’in Konuşma Kodeği Tespiti

Jenner, Scholz’un çalışmasını geliştirerek daha başarılı bir sınıflandırıcı oluşturmayı amaçlamıştır. Önerdiği yöntem, Scholz’un kullandığı frekans alanı niteliğine (gürültü spektrumu) ek olarak giriş sinyalinin zaman alanındaki niteliklerini de göz önüne alan bir genlik histogramı içermektedir [26].

Yöntem kod çözümü yapılmış ses örneklerinden çıkartılan gürültü spektrumu ve genlik histogramının, daha önce eğitilmiş örnekler ile normalize edilmiş çapraz ilinti (normalized cross-correlation) yardımıyla karşılaştırıp, en yakın profil ile eşleştirme yaparak ses kodlayıcısını tespit etmektedir.

Önerilen yöntemin başarımı TIMIT veri tabanından alınan konuşma örnekleri ile test edilmiştir. Örneklerden 160 pencere uzunluğu (3.94 sn.) üzerinde nitelik çıkartma işlemi yapılmış ve 3 kodek hariç tam başarımlar elde edilmiştir. Scholz'un yöntemine kıyasla daha kısa ses örnekleri kullanmalarına rağmen çok daha yüksek başarımlar elde edilmiştir. Testlere ait doğruluk matrisi Tablo 3.3'de verilmiştir. Düşük başarımlar elde edilen kodekler olarak iLBC ve Silk göze çarpmaktadır.

Çizelge 3.3: Jenner'in konuşma kodlayıcısı belirleme testine ait doğruluk matrisi (160 pencere)

Ses Kodlayıcısı		Sınıflandırma Sonucu						
		G.711	G.726	G.728	G.729	iLBC	AMR	Silk
G.711	u-law	%100	-	-	-	-	-	-
G.726	40kbit/s	-	%100	-	-	-	-	-
	32kbit/s	-	%100	-	-	-	-	-
	24kbit/s	-	%100	-	-	-	-	-
	16kbit/s	-	%100	-	-	-	-	-
G.728	16kbit/s	-	-	%100	-	-	-	-
G.729	11.8kbit/s	-	-	-	%96.43	%0.60	-	%2.98
	8kbit/s	-	-	-	%100	-	-	-
	6.4kbit/s	-	-	-	%100	-	-	-
iLBC	15.2kbit/s	-	-	-	%11.90	%88.10	-	-
	13.33kbit/s	-	-	-	%14.29	%85.71	-	-
AMR	12.2kbit/s	-	-	-	-	-	%100	-
	10.2kbit/s	-	-	-	-	-	%100	-
	7.95kbit/s	-	-	-	-	-	%100	-
	7.4kbit/s	-	-	-	-	-	%100	-
	6.7kbit/s	-	-	-	-	-	%100	-
	5.9kbit/s	-	-	-	-	-	%100	-
	5.15kbit/s	-	-	-	-	-	%100	-
4.75kbit/s	-	-	-	-	-	%100	-	
Silk	VBR	-	-	-	%21.43	-	-	%78.57

3.4 Memon'un Dosya Tipi Tespiti

Memon'un ağlardaki kaynakların istismarını tespit edebilmek amacıyla önerdiği yöntem bu çalışmanın da ilham kaynağı olmuştur. Yoğun akış trafiğinin olduğu ağlarda kaynakların hem adil kullanımını sağlamak hem de amacı dışında kullanılmasını engellemek son derece zor bir iştir. Bu tespit işleminin ağdaki tüm akışların takip edilip, her akıştaki paketlerin tek tek açılıp içerdiği bilgiye

bakılmadan yapılması, önerilecek yöntemin kullanılabilir olmasını sağlayacaktır.

Memon'un önerdiği yöntem, ağdaki akışlardan belirli uzunlukta paketler(payload) alınıp, bu paketler üzerinde istatistiksel analiz yapılarak akışın içerdiği dosya tipini, örneğin text, html ya da jpeg, belirlemeyi amaçlamıştır [35]. İstatistiki analiz için kullanılan öznitelikler 3 kısımda incelenebilir. Bunlardan ilki zaman alanında çıkartılan ortalama değer, varyans, entropi ve öz-ilinti gibi basit istatistiklerden oluşmaktadır. İkinci grup frekans alanında oluşturulan güç spektrumu üzerinden hesaplanan ortalama değer, varyans ve çarpıklığı[3] içermektedir. Son grup ise bicoherence [30], basıklık[1] ve çarpıklığı içeren yüksek dereceli istatistikler grubudur. Bu istatistikler de zaman alanında hesaplanmıştır.

Testlerde 8 farklı dosya tipini sınıflandırmayı amaçlamışlardır. Her dosyadan 4KB veri alınarak yapılan testlere ait doğruluk matrisi Tablo 3.4'de verilmiştir. Sonuçlar incelendiğinde ortalama başarımın %80 civarında olduğu görülmektedir. Bunun yanında Zip ve Jpeg tipindeki dosyaların tespitinde düşük başarım elde edilmiştir. Her dosyadan 16KB veri alındığı durumda yapılan testlerde başarımın büyük oranda yükseldiği belirtilmiştir ancak ağdaki herhangi bir akışa ait 16 paket toplamak için bir anda çok fazla paket alınması gerektiği de belirtilmiştir.

Çizelge 3.4: Memon'un akış içeriği belirleme testine ait doğruluk matrisi(4KB)

Dosya Tipi	Sınıflandırma Sonucu(%)							
	Txt	Bmp	Wav	Zip	Jpg	Mp3	Mpg	Enc
Txt	96.83	2.5	0	0.67	-	-	-	-
Bmp	3.17	91.67	3.33	0.5	0.67	0.17	0.50	-
Wav	0.5	6.67	82.17	0.67	2	7.5	0.5	-
Zip	0.67	0.33	1.33	53	12	3.83	1.17	27.67
Jpg	-	0.33	0.67	12.50	77.67	4.33	3.33	1.17
Mp3	0.33	0.17	0.33	3.17	3.17	90.50	2.33	-
Mpg	0.67	2	1.5	1	4.67	6	84.17	-
Enc	0.17	-	-	12.67	-	0.33	-	86.83

3.5 PinDr0p

Asıl amacı arama doğrulama olan PinDr0p adlı çalışma, kodek tespitini ara-maların kaynağını ve geçtikleri iletişim ortamlarını (Örn. VoIP, GSM, PSTN) belirlemede kullanmaktadır [11]. İletişim ortamlarında kullanılan kodekleri

genelleyerek; VoIP için G.711, iLBC, Speex veya G.729, GSM için GSM-FR ve PSTN için G.711 aday kodekler olarak belirtilmiştir.

Farklı iletişim ortamlarına ait kodekleri tespit ederken, kodeklerin ve iletişim ortamlarının kendilerine özgü nitelikleri kullanılmıştır. VoIP ağlarında paket kayıplarını ve varsa paket kaybı gizleme yöntemlerini göz önüne alarak VoIP kodeğinin tipini belirlemeyi amaçlamışlardır. Bunlara ek olarak, ses üzerinde gürültü profileme yapılmış ve tüm kodekler için spektral açıklık (spectral clarity), spektral seviye aralığı ve spektral seviye sapması hesaplanarak kodek tespitinde kullanılmıştır. Spektral açıklığı hesaplamada P.563 tek taraflı arama kalite tespit aracı kullanılmıştır.

Her bir ses örneği her kodeğin varlığını ya da yokluğunu belirten beş etikete sahip olmaktadır. Örneğin bir arama, sabit telefon hattından başlayıp, Speex kullanan bir VoIP ağından geçerek bir GSM kullanıcıya ulaşıyorsa, bu arama G.711, Speex ve GSM-FR etiketleri için 1, iLBC ve G.729 etiketleri için ise 0 değerine sahip olacaktır. Hesaplanan etiketler daha sonra çoklu sınıflandırıcıya girdi olarak verilerek her bir arama için bir ağ akış geçmişi (Örn. GSM -> PSTN veya PSTN -> VoIP -> GSM) tespit edilmiştir. Bu çalışmada aramaların kaynakları %90 ile %100 arasında başarımla tespit edilmiştir. Her ne kadar bu çalışmanın asıl amacı kodek tespit etmek olmasa da kodek tespitinin kritik olduğu bir uygulamayı göstermesi açısından önemlidir.

4. ÖNERİLEN YÖNTEM

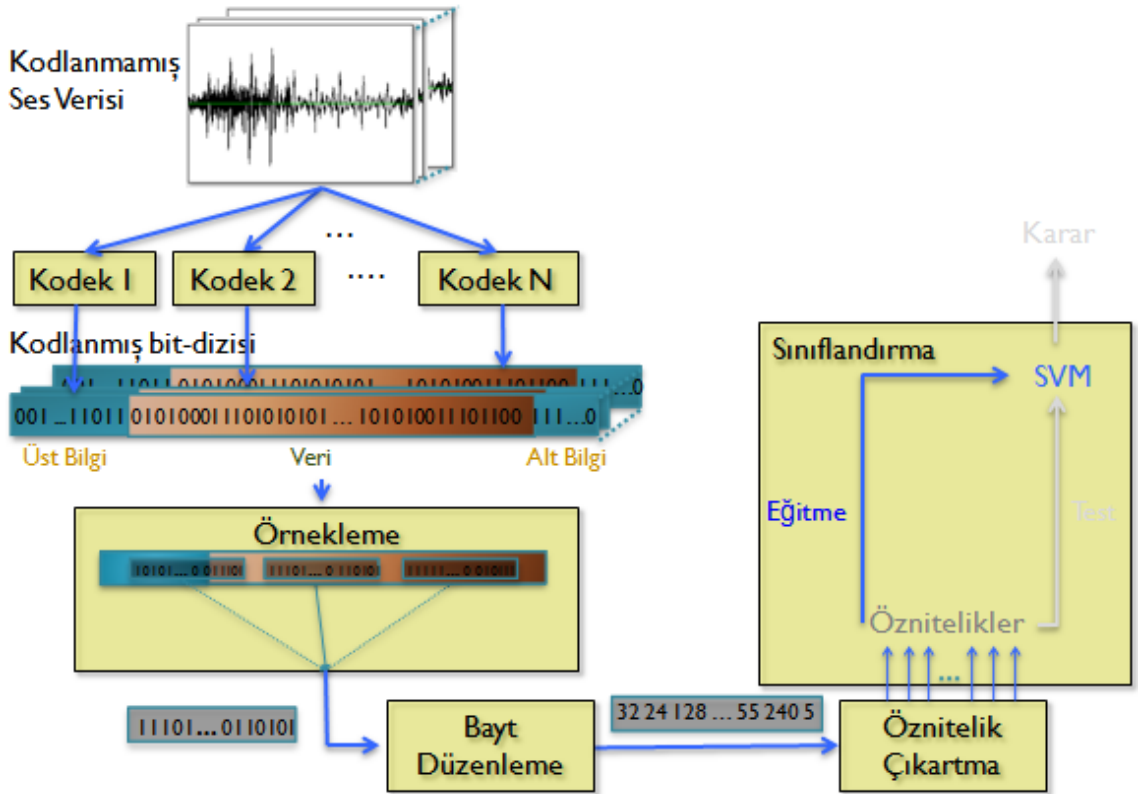
Önceki kısımlarda bu çalışmanın arkasındaki motivasyona ve uygulama alanlarına, konuşma ve müzik kodlamada kullanılan temel yöntemlere ve kodek tespit etme alanında yapılmış çalışmalara değinildi. Bu kısımda ise kodek tespit etme için önerdiğimiz hızlı ve yüksek başarıma sahip yöntem anlatılacaktır.

4.1 Yöntem

Önerilen yöntem kodek tespit ederken, kodlama üst bilgisini(metadata) ve kodlanmış verinin yapısını dikkate almamaktadır. Bunların yerine kodlanmış sesi, kodlama süreci ile ilişkili olan istatistiksel nitelikler ile karakterize etmeye çalışmaktadır. Bilindiği gibi kodlama, sıkıştırma oranı, ses kalitesi ve kodlama süresi gibi nitelikler arasında bir denge gözetmeye çalışmaktadır. Örneğin sıkıştırma oranı çok yüksek olan kodlayıcılar ses kalitesinden feragat etmektedir. PSTN'lerde kullanılan kodlayıcılar (a-law, u-law), iletişim ortamında kullanılan cihazların basitliği ve band genişliğinin geniş olması sebebiyle düşük kodlama süresi ve yüksek ses kalitelerine sahip olmalarının karşılığında, son derece düşük bir sıkıştırma oranına sahiptirler. Burada farklı iletişim ortamlarında kullanılan kodekler için örnekler çoğaltılabilir.

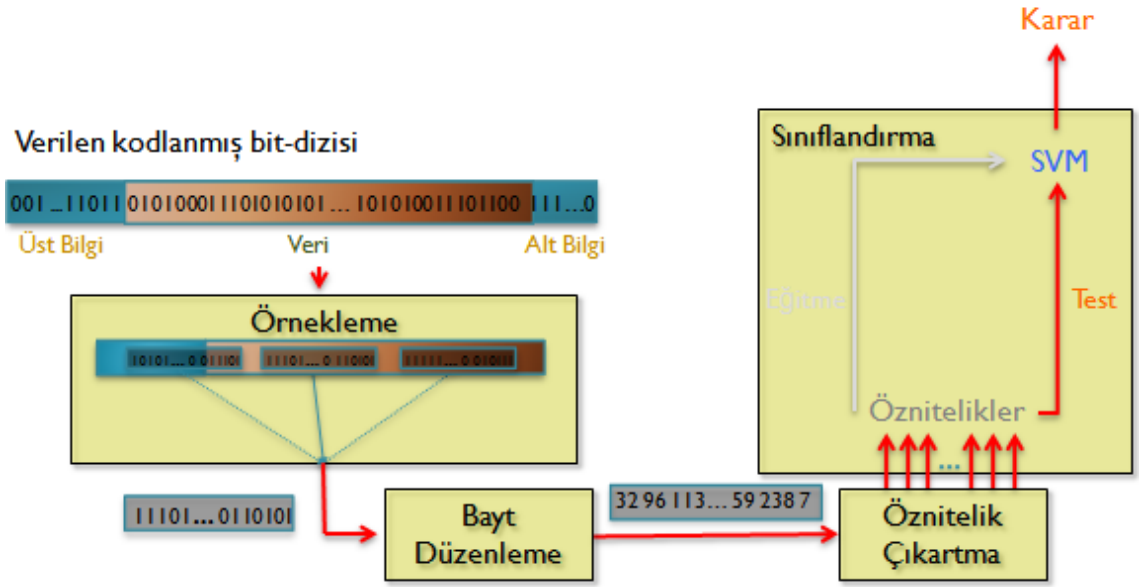
Görüldüğü üzere kodlanmış sesin içeriği ile alakalı olmayan, tamamen kodlayıcının tasarım prensiplerine dayanan karakteristikler ortaya çıkartılabilir. Yöntem, bu karakteristikleri ortaya çıkartmak amacıyla, kodlanmış sesin rasgelelik ve kaotik niteliklerini ölçmekte ve bu ölçümleri bir sınıflandırma sisteminde kullanarak istatistiksel bir model oluşturmaktadır.

Sistem çevrim dışı (offline) ve çevrim içi (online) olmak üzere iki evreden oluşmaktadır. Çevrim dışı evrede sistem eğitim adı verilen süreç ile sıfırdan kurulmaktadır. Bu süreçteki adımları dört kısımda inceleyebiliriz. İlk adımda, veri kümelerindeki kodlanmamış(raw) ses örnekleri, seçilmiş kodekler ile kodlanmaktadır. Bu adım sonucunda her orijinal ses dosyasının farklı kodekler ile kodlanmış versiyonları elde edildi. Daha sonra ikinci adımda, tüm kodlanmış ses dosyaları örnekleme işlemine tabi tutuluyor. Örnekleme işlemi ses dosyaları üzerinde belirlenmiş bir noktadan başlayarak belirli bir uzunluktaki verinin alınması olarak özetlenebilir. Burada ses dosyalarının yapısının sonuçlara etki etme ihtimalini ortadan kaldırmak amacıyla örneklemenin başlayacağı nokta her ses örneği için rasgele belirlenmektedir. Üçüncü adımda ise ses dosyalarından alınan bit dizisi, her sekiz bitin işaretli tam sayı olarak (0 - 255 değerlerinde) ifade edilmesi ile bayt vektör olarak organize edilmektedir. Son adımda, istatistiksel karakterize etme için gerekli özellikler bayt vektör üzerinden hesaplanmakta ve çok sınıflı sınıflandırma sistemi kurulmaktadır. Sistemin çevrim dışı evredeki çalışma adımları Şekil 4.1'de gösterilmiştir.



Şekil 4.1: Sistemin çevrim dışı evredeki çalışma adımları

Çevrim içi evrede ise çevrim dışı evreden farklı olarak ilk adım hiç gerçekleştirilmemekte, son adımda ise sadece öznitelik hesaplaması yapılmaktadır. Bu evrede verilen ses dosyaları oluşturulan sınıflandırıcı üzerinde test edilmektedir. Sistemin başarımı, doğru tespit edilen aynı kodek ile kodlanmış seslerin ortalaması alınarak hesaplanmaktadır. Sistemin çevrim içi evredeki çalışma adımları Şekil 4.2’de sunulmuştur.



Şekil 4.2: Sistemin çevrim içi evredeki çalışma adımları

4.2 Öznitelikler

Sistemin en önemli adımının, istatistiksel sınıflandırma işleminde kullanılacak özniteliklerin belirlenmesi olduğu görülmektedir. Sınıflandırma modelini oluştururken kullandığımız öznitelikleri iki grupta inceleyebiliriz. İlk grup kodlama işlemlerinin sesin kaotik yapısı üzerinde bıraktığı etkileri incelemektedir. İkinci grup ise sesin istatistiksel niteliklerini incelemektedir. Bu iki kategoriye ait öznitelikler alt başlıklarda detaylı bir şekilde açıklanacaktır.

4.2.1 Kaotik Öznitelikler

Ses üzerinde lineer modeller ile açığa çıkartılamayan kaotik fenomenlerin varlığına dair teorik ve deneysel kanıtlar mevcuttur [29]. Bir ses sinyalinin kaotik bir sistem ile oluşturulduğu varsayıldığında, farklı kodlayıcıların ses sinyalinin kaotik yapısı üzerinde farklı etkileri olacaktır. Kaotik niteliklerin temel konsepti, ses sinyal vektörlerinin faz uzayındaki komşuluklarına dayanmaktadır. Bir sinyalin faz uzayı vektörü olan $\mathbf{s}(n) = [x(n), x(n+T) \dots x(n+(D_E-1)T)]$, Taken'ın gömme teoremine (embedding theorem) [7] göre tekrar oluşturulmaktadır. Bu ifadede $x(n)$ sinyalin n inci örneği, T zaman gecikmesi ve D_E faz uzayının gömme boyutu olarak tanımlanmıştır. Taken'ın zaman gecikmeli gömme teoremi, faz uzayının, sinyalin orijinal bilinmeyen kaotik dinamikleri hakkında faydalı bilgiler açığa çıkartabileceğini belirtmektedir. Ancak bunun için uygun bir D_E değeri seçilmelidir. Hatalı komşular oranı yöntemi uygun değer (gömme boyutu) bulunmasında sıklıkla kullanılmaktadır [36]. Bu yöntem belirtilen boyut, D , için Hatalı Komşular Oranı (False Neighbors Fraction - FNF) sağlamakta ve uygun D_E değerini, boyutu (D), FNF sıfır olana kadar arttırarak bulmaktadır. Bir komşuluğu doğru ya da hatalı olarak işaretleme kriteri, en yakın komşu noktaları arasındaki uzaklığın kıyaslanması ile oluşturulmuştur. Bu iki komşuluk noktasının, $\mathbf{s}(n)$ ve $\mathbf{s}(m)$, ardışık artan boyutlarda gömülü olması gerekmektedir.

D boyutlu uzaydaki;

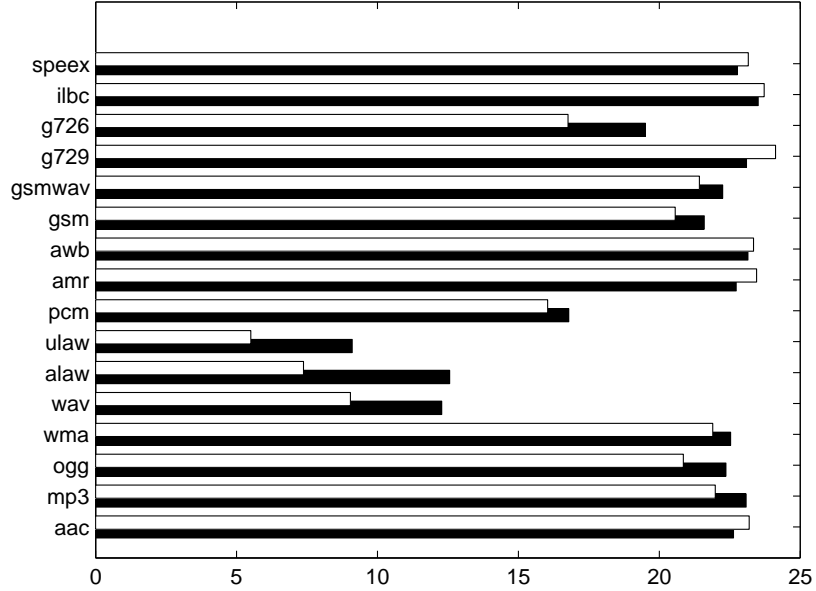
$$d_D(\mathbf{s}(n), \mathbf{s}(m)) = \sqrt{\sum_{k=0}^{D-1} (x(n+k \times T) - x(m+k \times T))^2}$$

uzaklığı, $(D+1)$ boyutlu uzaydaki;

$$d_{D+1}(\mathbf{s}(n), \mathbf{s}(m)) = \left((x(n+T \cdot D) - x(m+T \cdot D))^2 + \sum_{k=0}^{D-1} (x(n+k \times T) - x(m+k \times T))^2 \right)^{1/2}$$

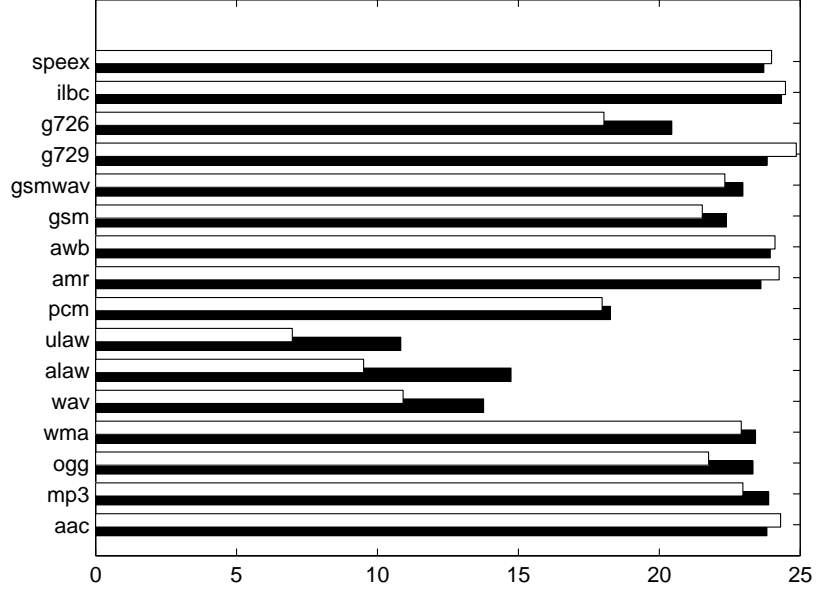
uzaklığından ciddi oranda farklı ise bunlar hatalı komşu çifti olarak değerlendirilmektedir [7]. Her komşuyu doğru ve hatalı olarak işaretledikten sonra, FNF hatalı komşuların tüm komşulara oranı olarak tanımlanmaktadır. İlk beş boyut için en yakın hatalı komşular oranı, komşuluğun büyüklüğünün

ortalaması ve komşuluğun büyüklüğünün karesinin ortalaması hesaplanmış ve bu 15 değer nitelik vektörüne eklenmiştir. Boyut uzayı arttırıldıkça FNF'lerin hesaplanma süresi her yeni boyut için giderek artmaktadır. Burada sadece ilk beş boyutun hesaplanma sebebi işlem zamanı ve başarımların dengelemesinden kaynaklanmaktadır. Testlerde kullanılan iki farklı müzik ve konuşma veri kümesi üzerinde hesaplanan FNF değerleri Şekil 4.3 ve Şekil 4.4'de sunulmuştur. İlk şekil farklı kodlayıcılar ile kodlanmış ses örnekleri için 5. boyuta ait komşuluğun büyüklüğünün ortalamasını göstermektedir. İkinci şekil ise yine 5. boyut için komşuluğun büyüklüğünün karesinin ortalamasının farklı kodlayıcılara ait ortalama değerini belirtmektedir.



Şekil 4.3: Farklı kodekler ile kodlanmış ses örnekleri için 5. gömme boyutuna ait komşuluğun büyüklüğünün ortalama değeri

Lyapunov üsleri (Lyapunov Exponents - LE) kaotik bir sinyalin tahmin edilebilirliğini ölçmektedir [22]. LE faz uzayında yakın yörüngeler arasındaki uzaklaşmaları (divergence) belirten genel bir ölçüttür. Pozitif üs başlangıçta birbirine yakın olan yörüngelerin zaman içinde uzaklaştıklarını belirtmektedir. Pozitif üssün genliği bu iki yörüngenin birbirinden ne kadar hızlı uzaklaştıklarını göstermektedir. LE genliği büyük olan sistemler tahmin edilemez olarak ifade



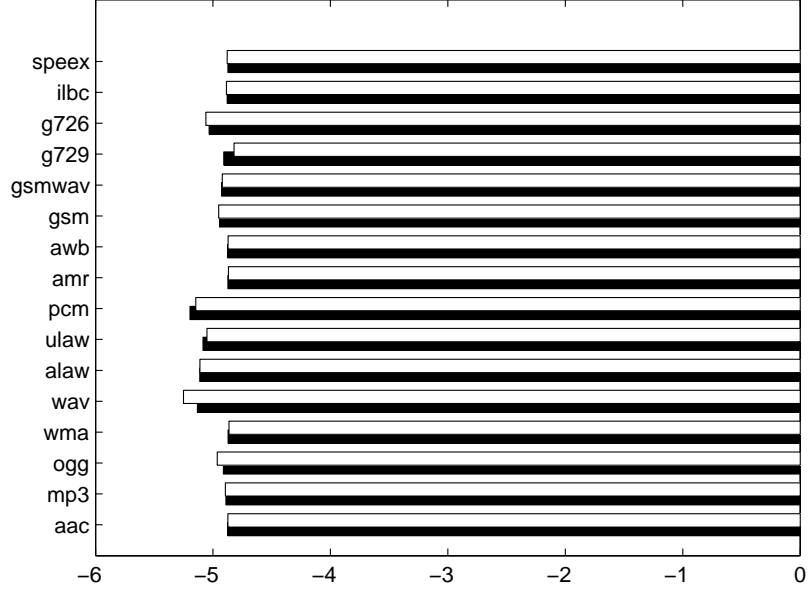
Şekil 4.4: Farklı kodekler ile kodlanmış ses örnekleri için 5. gömme boyutuna ait komşuluğun büyüklüğünün karesinin ortalama değeri

edilmektedir. Herbir gömme boyutu D_E için LE

$$\lambda = \lim_{N \rightarrow +\infty} \frac{1}{N} \sum_{n=1}^N \ln \frac{d(s(n+1), s(m+1))}{d(s(n), s(m))}$$

formülü ile hesaplanmaktadır. Burada $s(n)$ referans noktasını, $s(m)$ ise $s(n)$ 'nin yakın yörünge üzerindeki en yakın komşusunu belirtmektedir. Lyapunov üsleri iki komşu yörünge arasındaki uzaklaşmanın (yada yakınlaşmanın) ortalama oranıdır. Azalan sıra ile $\lambda_1, \lambda_2, \dots, \lambda_{D_E}$ olmak üzere toplam D_E adet Lyapunov Üsleri mevcuttur. λ_{D_E} en büyük LE olarak bilinmektedir ve pozitif en büyük LE kaosu varlığına işaretler. Farklı yörüngelerdeki her en yakın komşu çiftleri için LE'ler hesaplandıktan sonra, tüm sinyalin LE'si bütün hesaplanmış LE'lerin ortalaması şeklinde belirlenir. Testlerde kullanılan iki farklı müzik ve konuşma veri kümesi üzerinde hesaplanan ortalama LE değerleri Şekil 4.5'de sunulmuştur. Bu şekilde gösterilen değerler ilk iterasyon için ortalama germe etmeninin (stretching factor) logaritmasına aittir.

Sıkıştırma algoritmaları sinyalin içindeki tekrarları açığa çıkarmaktadır ve performansları çıktılarının ilintisizleşmesi (decorrelation) ile doğrudan alakalıdır.



Şekil 4.5: Farklı kodekler ile kodlanmış ses örnekleri için ilk iterasyona ait germe etmenin logaritmasının ortalama değerleri

Pratikte mükemmel bir sıkıştırma algoritması olmadığı için açığa çıkartılmamış ilintiler (correlations) çıktı sinyalinde kalmaktadır. Bu noktadaki temel öngörümüz her sıkıştırma algoritması için bu niteliklerin istatistiksel olarak farklı olacağı ve kodlanmış verideki arta kalan ilintilerin başarılı sınıflandırıcılar kurabilmek adına yeterli oranda benzersiz olduğudur. Bu farklılıkları yakalayabilmek adına sinyallerin FNF ve LE değerleri TISEAN [31] yazılım paketi kullanılarak hesaplandı. FNF niteliğinin nitelik vektörü F_{DE} üç elemandan oluşmaktadır; hatalı komşuların oranı, komşuluğun ortalama boyutu ve komşuluğun karelerinin kare kök ortalama (root mean squared) boyutu.

$$F_{DE} = [FNF, mean(d_{DE}(s(n), s(m))), RMS(d_{DE}(s(n), s(m)))]$$

Toplam kaotik nitelik vektörü, F , 26 elemandan oluşmaktadır;

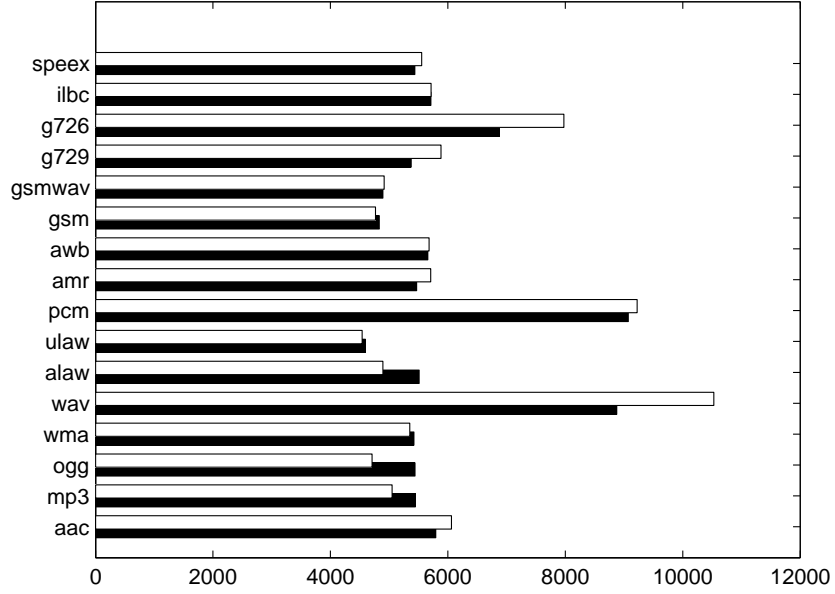
$$F = \{F_{DE} | D_E = 1, 2, 3, 4, 5\} \cup \{\lambda_i | i = 1, 2, \dots, 11\}$$

4.2.2 Rasgelelik Öznitelikleri

Bu öznitelikler, NIST tarafından kriptografik bileşenlerin rasgelelik özelliklerini değerlendirmek amacıyla düzenlenmiş rasgelelik testlerinden esinlenerek kullanılmışlardır [32]. Örneklenmiş bayt vektörlerinin rasgelelik özniteliklerini karakterize etmek amacıyla bu testlerde kullanılan niteliklerin bir alt kümesini kullandık. Kullanılan nitelikleri zaman ve frekans alanı olmak üzere iki kategoride inceleyebiliriz.

Zaman alanında, ortalama değer, varyans, öz-ilinti, entropi gibi basit istatistikler ve bicoherence, çarpıklık, basıklık gibi yüksek dereceli istatistikler hesaplanmıştır. Her kodeğin farklı kodlama tekniği kullandığı, birçok bit hızı opsiyonuna sahip olduğu ve farklı örnekleme frekanslarında çalıştığı göz önüne alındığında, bu farklılıkların kodlanmış ses üzerinde ciddi etkileri olacağı aşıkardır. Örneğin varyans, değişkenlik ya da verideki değerlerin dağılımı ile ilgilidir. Bu sebeple veri üzerindeki bazı paternlerin tekrarı ölçülen varyans değerini etkilemektedir. Sıkıştırma işleminin ses üzerindeki mevcut yapıları ya da paternleri ortadan kaldırdığı dikkate alındığında, kodlanmış ses örneklerinden alınan örneklerin eşit oranda bir dağılıma sahip olacağı öngörülebilir.

Testlerde kullanılan müzik ve konuşma örneklerinden oluşan iki farklı veri kümesinden alınan örnekler üzerinde hesaplanan varyans değerleri Şekil 4.6'da verilmiştir. Bu şekil incelendiğinde PCM tabanlı kodekler olan G.726 ve ADPCM ile WAV formatındaki dosyaların varyans değerleri diğerlerine göre ciddi oranda fazla çıkmıştır. PCM işleminin herhangi bir sıkıştırma adımı içermemesi bu duruma sebep olmuştur.

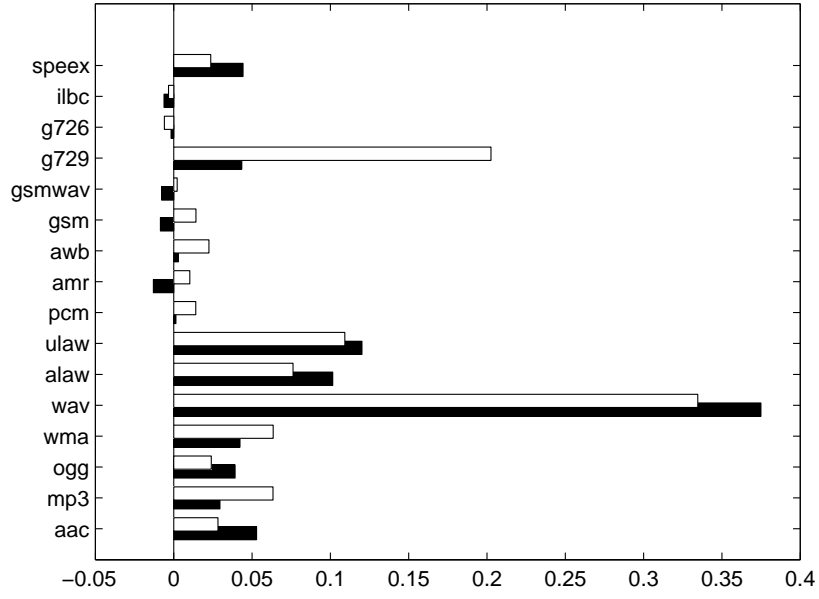


Şekil 4.6: Farklı kodekler ile kodlanmış ses örneklerine ait ortalama varyans değerleri

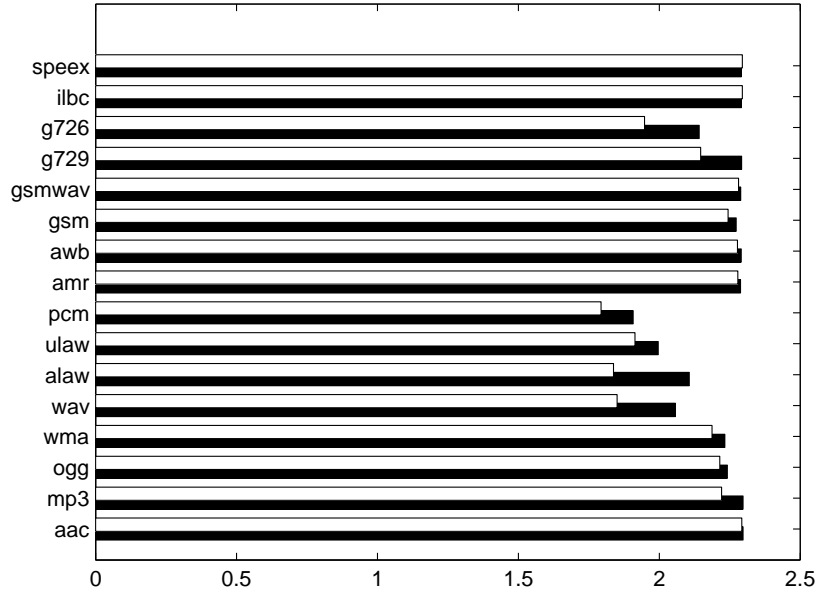
Öz-ilinti, kodlanmış veri üzerindeki tekrar eden paternlerin tespit edilmesinde kullanılabilecek başka bir niteliklidir. Bu sebeple öz-ilinti fonksiyonunun ilk 21 katsayısını nitelik vektörümüze ekledik. Entropi, veri üzerindeki rasgeleliği ve belirsizliği ölçen bir niteliklidir. Bu sebeple entropi, kodlayıcıları sesi sıkıştırma yeteneklerine göre ayırt edebilen bir faktör olarak kullanılabilir. Entropy fonksiyonunun formülü şu şekilde ifade edilebilir:

$$H(X) = - \sum_{i=1}^n p(x_i) \log p(x_i) \quad (4.1)$$

Şekil 4.7 ve Şekil 4.8 müzik ve konuşma içeren veri kümelerinden alınan ses örneklerine ait sırası ile öz-ilinti fonksiyonunun 11. katsayısının ve entropi değerlerinin ortalamasını göstermektedir.



Şekil 4.7: Farklı kodekler ile kodlanmış ses örneklerine ait öz-ilinti fonksiyonun 11. katsayısının ortalama değerleri



Şekil 4.8: Farklı kodekler ile kodlanmış ses örneklerine ait ortalama entropi değerleri

Yüksek dereceli istatistiklerden olan ve 0 ile 1 arasında değerler alabilen bicoherence, byte vektör üzerindeki doğrusalsızlığı (non-linearity) ve non-Gaussianity'yi tespit etmekte ve miktarını ölçmekte kullanılmaktadır. Bu sayede farklı seviyelerdeki sıkıştırmaları ayırt etmede yardımcı olmaktadır. Farklı kodlayıcıların etkilerini yakalayabilmek amacıyla hesaplanan ortalama bicoherence, nitelik vektörüne eklenmiştir.

Çarpıklık (Skewness) ve Basıklık (Kurtosis) verinin dağılımı ile ilgili iki farklı yüksek dereceli istatistiktir. Bu istatistikler, verinin dağılımının normal dağılımdan ne derece saptığının belirlenmesinde kullanılmaktadır. Benzer şekilde bu iki istatistik verinin dağılımının şekilsel olarak incelenmesine de olanak sağlamaktadır. Bu iki nitelik örneklenmiş bayt vektörüne ait olasılık kütle fonksiyonu kullanılarak hesaplanmıştır.

Çarpıklık bir olasılık dağılımının simetrik olmayışının ölçülmesidir. Pozitif çarpıklık dağılımın kütesinin grafiğin sol tarafında yoğunlaştığı, tersine negatif çarpıklık ise dağılımın kütesinin grafiğin sağ tarafında yoğunlaştığı durumları ifade etmektedir. Çarpıklık üçüncü standardize edilmiş moment olup şu şekilde ifade edilmektedir:

$$\gamma_1 = \frac{\mu_3}{\sigma^3} \quad (4.2)$$

Bu ifadede μ_3 üçüncü ortalama etrafındaki moment ve σ standart sapma olarak ifade edilmektedir.

Basıklık bir olasılık dağılımdaki dorukluluk (peakedness) miktarının ölçümü olarak tanımlanmaktadır. Basıklığı aldığı değerlere göre üç gruba ayırabiliriz. Sıfır basıklık değeri normal dağılıma aittir. Basıklık değerinin pozitif olması, dağılımın ortalama değerinde sivri ve kuyrukları daha şişman bir görüntüye sahip olduğunu göstermektedir. Değerin 0 ile -2 arasında olması ise dağılımın ortalama değerinde normal dağılıma göre daha az sivri ve kuyrukları daha zayıf bir görüntüsünün olduğunu belirtmektedir. Basıklık, bir olasılık dağılımının "dördüncü kümülant değeri bölü varyans karesi" olarak şöyle tanımlanır:

$$\gamma_2 = \frac{\kappa_4}{\kappa_2^2} = \frac{\mu_4}{\sigma^4} - 3 \quad (4.3)$$

Bu ifadede μ_4 dördüncü ortalama etrafındaki moment ve σ standart sapma olarak ifade edilmektedir. Buradaki -3 değeri normal dağılımın basıklığını 0 yapmak için yapılmış bir ayarlama olarak açıklanmaktadır.

Frekans alanında hesaplanan nitelikler daha basit olmakla beraber enerjinin birkaç spektral banddaki dağılımı ile ilgili istatistikler vermektedir. Bu alandaki nitelikleri hesaplamak için örneklenmiş bayt vektörlerinin FFT'leri (Fast Fourier Transform) alınmıştır. Daha sonra ise frekans spektrumu dört eşit frekans bandına bölünmüştür. Her frekans bandı için ortalama değer, varyans ve çarpıklık hesaplanarak nitelik vektörüne eklenmiştir. Rasgelelik nitelikleri olarak 27 tanesi zaman alanında, 12 tanesi ise frekans alanında olmak üzere toplam 39 adet nitelik hesaplanmıştır.

5. TESTLER

Testlerde dört farklı veri kümesi kullanıldı. İlk veri kümesi farklı müzik türlerine ait CD kalitesindeki (1411 Kbps) 500 şarkıdan alınan 1000 örnekten oluşmaktadır. Her örnek beş saniye uzunluğunda olup, bir şarkıdan alınan iki örnek şarkının örtüşmeyen ve ardışık olmayan kısımlarından alınmıştır. İkinci veri kümesi ise VoxForge konuşma veri tabanından alınan 2000 farklı konuşma örneğini içermektedir [5]. Bu konuşma örnekleri 1 ile 13 sn. arasında değişen uzunlukta olup 256 Kbps bit hızına sahiptirler. Tezin ileriki kısımlarında ilk veri kümesine Müzik-I, ikincisine ise Konuşma-I şeklinde atıf yapılacaktır. Sonuçların güvenilirliğini tesis etmek amacıyla her biri 4000 örnekten oluşan iki büyük müzik ve konuşma veri kümesi daha oluşturuldu. Müzik veri kümesi önceki ile aynı yöntemle oluşturuldu ancak tek fark bu küme içindeki her örnek farklı şarkılardan alındı. İkinci konuşma veri kümesi ise aynı şekilde VoxForge konuşma veri tabanında alınan örnekler ile oluşturuldu. Bu konuşma örnekleri Konuşma-I ile tamamen farklı örneklerdir. Bu son iki veri kümesi de bundan sonra Müzik-II ve Konuşma-II şeklinde belirtilecektir.

Tüm veri kümelerindeki örnekler öncelikle Tablo 2.1’de verilen kodekler ve bit hızları ile kodlandı. Kodlama adımından sonra, kodlanmış her örnekten belirli uzunluktaki byte dizileri rastgele konumlardan başlanarak alındı ve 65 boyutlu nitelik vektörü hesaplandı. Sınıflandırma için standart bir makine öğrenme yöntemi olan destek vektör makine (support vector machine - svm) kullanıldı. SVM için Libsvm [16] paketi radial basis kernel (rbs) ile kullanıldı. Tüm testlerde veri kümesindeki örneklerin yarısı eğitim için diğer yarısı ise test için kullanıldı. Burada Müzik-I veri kümesindeki bir şarkıdan alınan iki örneğin aynı grup (eğitim ya da test) içinde kalması sağlandı.

Yapılan testleri iki grupta inceleyebiliriz. İlk gruptaki testlerde, kodlanmış bir sesin kodlanmasında kullanılan kodeği, örneklerden farklı boyutlarda byte dizileri olarak tespit etmeyi amaçladık. İkinci grup testlerde ise ikili kodlama senaryoları incelendi. Bu testlerde kodlanıp kod çözümü yapılmış sesler bir kez daha kodlama işlemine tabi tutuldu ve ilk kodlamada kullanılan kodek tespit edilmeye çalışıldı.

5.1 Tekli Kodlama Testleri

Dört veri kümesi üzerinde birçok deney yapıldı. Testlerin tümünde kodlanmamış ses örnekleri de testlere ayrı bir sınıf olarak eklenmiş ve 17 sınıflık bir sınıflandırma problemi ortaya çıkmıştır. Yöntemin en kritik adımlarından bir tanesi örnekleme boyutunun (bayt dizisi uzunluğu) belirlenmesidir. Bu uzunluğun, başarımlar ve çalışma zamanını göz önünde bulundurarak, belirlenmesine yönelik 1 KB, 2 KB, 4 KB ve 8 KB bayt uzunlukları için çok sayıda test gerçekleştirildi. Tablo 5.1 Müzik-I ve Konuşma-I veri kümeleri için farklı bayt dizisi uzunluklarına ait başarımları göstermektedir. Müzik-I veri kümesine ait sonuçlar incelendiğinde, başarımların 1 KB'tan 8 KB'a kadar artış gösterdiği görülmektedir. Ancak programın çalışma zamanı göz önüne alındığında, 8 KB'lık bayt dizileri üzerinden niteliklerin çıkartılması 4 KB'a göre ciddi oranda uzun sürmektedir. Konuşma-I veri kümesindeki ses örneklerinin boyutları kısa olduğundan, bu örneklerin bazı kodekler ile kodlanması sonucu 4 KB'tan daha küçük boyutlarda ses örnekleri oluştu. Bu sebeple Konuşma-I veri kümesi üzerinde sadece 1 ve 2 KB örnekleme boyutları için testler yapıldı. Bu iki testin başarımlar oranları Müzik-I'e kıyasla daha yüksek gelmiştir. Hesaplama zamanı ve başarımlar arasında bir denge gözetmek amacıyla bundan sonraki testlerde örnekleme boyutu Müzik-I veri kümesi için 4 KB, Konuşma-I veri kümesi için ise 2 KB olarak alınacaktır.

Önerilen yöntemin farklı kodlayıcıları ayırt etme yeteneği Konuşma-I ve Müzik-I veri kümeleri için sırasıyla Tablo 5.12 ve Tablo 5.13'de doğruluk matrisleri şeklinde verilmiştir. Konuşma veri kümesine ait sonuçlar incelendiğinde birçok kodeğin tam başarımlar ile tespit edildiği, birbirleri ile karıştırılan kodeklerin ise benzer kodlama tekniğine sahip kodlayıcılar olduğu görülmektedir. Örneğin,

AMR, AWB ve G.729 kodlayıcıları kodlama tekniği olarak ACELP kullanılmaktadır ve bu üç kodlayıcı sadece birbirleri ile karıştırılmıştır. Benzer şekilde a-law, u-law, PCM, WAV (kodlanmamış) ve G.726 ile kodlanmış örnekler sadece kendi aralarında karıştırılmıştır ve bu kodlayıcılar PCM veya ADPCM kodlama tekniğini kullanmaktadırlar. Müzik veri kümesine ait sonuçlar incelendiğinde, bu testte de benzer durumların olduğu görülmektedir. Aynı test Müzik-II ve Konuşma-II veri kümeleri üzerinde de yapılmıştır ve başarımları 4 KB örnekleme boyutu için sırası ile %97.91 ve %97.88 olarak hesaplanmıştır. Bu iki testin başarımlarının daha yüksek olması, sınıflandırıcıyı eğitmek için kullanılan örneklerin daha fazla olması ve konuşma veri kümesi için daha büyük bir örnekleme penceresi kullanması ile açıklanabilir.

Çizelge 5.1: Örnekleme Boyutunun Başarım Üzerine Etkisi

Örnekleme Boyutu	Başarım(%)	
	Müzik-I	Konuşma-I
1KB	85.1	87.04
2KB	90.91	97.34
4KB	95.88	n/a
8KB	98.13	n/a

Sınıflandırma başarımında en yüksek etkisi olan niteliklerin belirlenmesi için Müzik-II ve Konuşma-II veri kümeleri üzerinde nitelik seçme işlemi uygulanmıştır. Nitelik seçme algoritması olarak sıralı kayan ileri arama (sequential floating forward search - SFFS) kullanılmıştır [38]. Algoritma başlangıçta tek başına en yüksek başarımları sağlayan niteliği seçer. Daha sonra seçtiği ilk nitelik ile birleştirildiğinde en yüksek başarımları sağlayan ikinci niteliği belirler. Bu süreç belirlenen nitelik sayısına ulaşıncaya kadar aynı şekilde devam eder. Kayan arama metodu daha önce çıkartılan niteliklerin ya da seçilmeyen niteliklerin daha sonra nitelik vektörüne eklenebilmesini sağlamaktadır. Müzik ve Konuşma veri kümeleri üzerinde yapılan nitelik belirleme testlerinde, 10 nitelikten sonra başarımın az bir miktar yükseldiği tespit edildi. Test sonuçlarına göre çarpıklık, entropi, ortalama değer, varyans, 3., 4. ve 5. gömme boyutları için komşuluğun büyüklüğünün ortalaması, 5. gömme boyutu için komşuluğun büyüklüğünün karesinin ortalaması, 5. gömme boyutu için en yakın hatalı komşular oranı, öz ilinti fonksiyonunun 2., 5., 11. ve 12. katsayıları ile bir LE niteliği

olan ilk iterasyonun gerilme katsayısının logaritması değerleri, daha belirleyici ayırım yapma yeteneğine sahiptirler. Bu 14 nitelik, Müzik-II ve Konuşma-II veri kümeleri üzerinde koşturulan iki nitelik seçme algoritması sonucu ortaya çıkan 20 niteliğin (6 tanesi ortak) birleştirilmesi ile oluşan farklı nitelikleri kapsamaktadır. Tekli kodlama testlerini her veri kümesi için seçilen 10 nitelik ile tekrar ettiğimizde, Müzik-II ve Konuşma-II için sırası ile %96.15 ve %95.80 başarımlar elde edildi. Bu sonuçlar incelendiğinde, nitelik sayısını ciddi oranda azaltarak yöntemin çalışma zamanı, başarımda gerçekleşecek ufak bir azalma karşılığında, düşürülebilmektedir.

Sınıflandırma başarımının seçilen kodeklerin bit hızlarına bağlı olmadığını göstermek amacıyla, benzer bit hızı ve en kötü bit hızı kodlama senaryoları adlı iki farklı test daha yapıldı. Benzer bit hızı kodlama senaryosunda, kodlayıcıların geçerli bit hızı seçeneklerini kullanmak yerine tüm ses örnekleri benzer bit hızlarında kodlandı. Burada tek bit hızı opsiyonu olan kodlayıcılar için o seçenek kullanıldı. Bu sebeple, sadece 64 kbps bit hızı seçeneği sunan PCM tabanlı kodekler ile eşleştirme yapabilmek adına, yüksek kaliteli ses kodekleri için bu bit hızı seçildi. GSM ve VoIP kodekleri için 13 kbps kodlama bit hızı seçildi. Sonuç olarak, PSTN ve yüksek kaliteli kodlama kodekleri 64 kbps ile GSM ve VoIP kodekleri ise 13 kbps bit hızına en yakın olan bit hızı seçenekleri ile kodlandı. En kötü bit hızı kodlama senaryosunda ise tüm ses örnekleri kodlayıcıların önerdiği en düşük bit hızı seçeneği ile kodlandı. Her iki testte de Müzik-I veri kümesi 2 KB örnekleme boyutu ile kullanıldı. Testlerin başarımları benzer bit hızı ve en kötü bit hızı testleri için sırası ile %94.13 ve %94.37 olarak hesaplandı. Bu sonuçlar, önerilen yöntemin kodek tespit etme başarımının kodlama seviyesine bağlı olmadığını göstermektedir.

Kodlayıcıları ayrıca sadece kendi kodek grupları içinde test ettik. Bu testleri yapabilmek adına dört farklı kodek grubu (PSTN, GSM, VoIP ve yüksek kaliteli sıkıştırma) için dört farklı sınıflandırıcı oluşturuldu. Tüm veri kümeleri üzerinde bu testler tekrarlandı. Müzik-I veri kümesi üzerinde 4 KB örnekleme boyutu ile yapılan teste ait başarımlar PSTN, GSM, VoIP ve yüksek kaliteli sıkıştırma kodekleri için sırası ile %99.05, %98.8, %99.95 ve %95.33 olarak hesaplandı. Konuşma-I veri kümesi ile yapılan test için 2 KB pencere boyutu kullanıldığında %95.65, %97.8, %99.02 ve %100 oranlarında başarımlar elde edildi. Bu iki test ile Müzik-II ve Konuşma-II veri kümeleri üzerinde yapılan testlere ait başarımlar

Tablo 5.2’de verilmiştir. Bu sonuçlar incelendiğinde, eğitim için kullanılan örnek sayısı arttırıldığında ve örnekleme boyutu büyütüldüğünde yöntemin başarımlı yükselmektedir ve aynı zamanda ses dosyalarının içeriği (müzik ya da konuşma) yöntemin başarımlı etkilememektedir.

Çizelge 5.2: Tüm Veri Kümeleri İçin Kodek Grup İçi Test Sonuçları

Kodlayıcı Grubu	Başarım (%)			
	Müzik-I (4KB)	Konuşma-I (2KB)	Müzik-II (4KB)	Konuşma-II (4KB)
GSM	99.05	95.65	99.01	98.46
PSTN	98.8	97.8	99.65	98.87
VoIP	99.95	99.02	99.46	99.38
Yüksek Kaliteli Sıkıştırma	95.33	100	98.59	98.93

Tablo 5.12 ve Tablo 5.13’de elde edilen sonuçlar ile Tablo 5.2’de verilen sonuçlar karşılaştırıldığında, önerilen yöntemin kodek tespit etme yeteneği testlerde kullanılan veri kümesine çok az derecede bağlıdır. Genel olarak, konuşma içerikli sesleri kodlamak üzere geliştirilmiş kodekler, müzik veri kümeleri üzerinde test edildiğinde daha başarılı sınıflandırıldıkları gözlemlenmiştir. Benzer şekilde, müzik içerikli sesleri kodlamak üzere geliştirilmiş kodekler, konuşma veri kümeleri ile yapılan testlerde daha yüksek oranda başarılı tespit edilmişlerdir. Bunun sebebi, belirli bir içerik için geliştirilmiş kodlayıcılar, o içeriğe ait karakteristikler göz önüne alınarak tasarlanmıştır. Burada Tablo 2.1’de görüldüğü üzere aynı kodek grubundaki kodlayıcılar benzer kodlama yöntemlerine ve bit hızlarına sahiptir. Sonuç olarak, kodlamada kullanılan kodek ile kodlama yapılacak sesin içeriği örtüştüğünde, kodlayıcılar çok daha başarılı davranmaktadır. Ancak sesin içeriği ile kodeğin hedef içeriği örtüşmediğinde ise kodlayıcılar daha kötü davranmakta, bu sayede kodekler arasındaki farklar daha fazla ortaya çıkmakta ve sınıflandırma işlemi daha yüksek başarımlı gerçekleştirilebilmektedir. Bu tespitteki tek istisnai durum PSTN kodekleri için geçerlidir. Bu durum PSTN kodeklerinin aslında karmaşık bir kodlamadan ziyade, sadece örnekleme ve kuantalama işlemleri ile müzik ya da konuşma içeriğinin kendine has özelliklerini göz önüne almamasından kaynaklanmaktadır. Buna rağmen, kodlama bit hızı 64 kbps’den 32 kbps’ye düştüğünde, PCM kodeği müzik veri kümesi üzerinde daha rahat tespit edilebilmekte, çünkü müzik üzerinde konuşmaya göre daha fazla

bozucu etki bırakmaktadır.

5.2 İkili Kodlama Testleri

Herhangi iki grup, ses iletişimine başlayacağı zaman, yoğunlukla ses farklı iletişim ortamlarını dolaşmak durumundadır. Örneğin, bir VoIP kullanıcısı bir GSM kullanıcısını aradığında, ses verisi ilk önce uygun VoIP kodu ile kodlanır, daha sonra ise bir GSM kodu ile ikinci defa kodlanmaktadır. Bu ve benzeri durumlarda, yani sesin birden fazla kodlandığı durumlarda, ilk kodlayıcıyı tespit edebilmek aramanın başlatıldığı ağı belirleyebilme ya da ikiden fazla kodlama varsa ara ağları tanımlayabilme imkanı sunmaktadır. Bu yetenek kısaca ikili kodlanmış sesler ile tek kodlanmış sesleri ayırt edebilmek olarak da özetlenebilir. Önerilen yöntemin bu tip durumlardaki başarımını ölçmek adına 3 farklı ikili kodlama senaryosu göz önüne alındı. Testlerde sadece Müzik-I ve Konuşma-I veri kümeleri 4 KB örnekleme pencere boyutu ile kullanıldığı için bundan sonra bu veri kümelerine sırası ile müzik ve konuşma veri kümeleri şeklinde atıfta bulunulacaktır.

Göz önüne aldığımız ilk durum, bir GSM ağından başlatılıp bir PSTN kullanıcısında sonlanan senaryodur. Bu tip durumlarda, ses ilk önce dört farklı GSM kodlayıcısından bir tanesi ile, daha sonra PSTN ağına geçtiğinde ise olası üç PSTN kodu ile kodlanmıştır. Üç farklı PSTN kodu olduğu için, üç farklı test, ilki a-law kodu, ikincisi u-law kodu ve son test de PCM kodu için, yapıldı. Testlerde, kodlanmamış ses dosyaları ilk önce dört farklı GSM kodu ile kodlandı, daha sonra bu kodlanmış ses örneklerinin kod çözümü yapıldı ve son olarak ise her test için uygun PSTN kodu ile kodlandılar. Tüm bu testlerde, yöntemin tek ve ikili kodlanmış sesleri ayırt edebildiğini göstermek için tek kodlanmış ses örnekleri de ayrı bir sınıf olarak testlere eklendi. Bu durum 5 sınıflı sınıflandırma problemini ortaya çıkardı. GSM ağından PSTN ağına geçiş senaryosunda ikinci kodlayıcı olarak a-law kodu kullanılan test için müzik veri kümesinde %92.28, konuşma veri kümesinde ise %89.78 başarımler elde edildi. Bu teste ait doğruluk matrisi Tablo 5.3'de verilmiştir. İkinci kodlayıcının u-law olduğu senaryoya ait başarımler ise sırası ile müzik ve konuşma veri kümeleri için %85.2 ve %90.66 olarak hesaplanmıştır. Son olarak PSTN kodu

PCM olarak seçildiğindeki duruma ait başarımlar aynı sıra için %84.48 ve %90.2 şeklinde bulunmuştur. Tüm bu sonuçlar Tablo 5.4'de verilmiştir.

Çizelge 5.3: GSM'den PSTN ağına geçiş senaryosunda a-law kodeği kullanılan teste ait doğruluk matrisi (Müzik veri kümesi)

			Sınıflandırılan				
			ikili kodlanmış				tek kodlanmış
			amr	awb	gsm	gsm(wav)	a-law
Gerçek	ikili kodlanmış	amr	99	1	0	0	0
		awb	4.8	94	0	1.2	0
		gsm	0	0.2	98.6	0	1.2
		gsm(wav)	0.6	26.2	0	72.6	0.6
	tek kodlanmış	a-law	0.8	0.6	0.6	0.8	97.2

Çizelge 5.4: GSM'den PSTN ağına geçiş senaryosu test sonuçları

İkinci Kodek	Başarım (%)	
	Müzik-I	Konuşma-I
a-law	92.28	89.78
pcm	84.48	90.20
u-law	85.20	90.66

Göz önüne aldığımız ikinci senaryo ise GSM ağından VoIP ağına geçiş durumudur. Bu testlerde de kodlanmamış ses dosyaları ilk önce dört farklı GSM kodeği ile kodlanmış, daha sonra ise dört farklı VoIP kodeği ile tekrar kodlanmıştır. Dört farklı VoIP kodeği olduğu için dört grup test yapıldı. Önceki senaryodaki gibi tek kodlanmış ses örnekleri de testlere ayrı bir sınıf olarak eklenmiş, ve tüm testler 5 sınıf sınıflandırma problemi şeklinde ifade edilmiştir. Müzik veri kümesi ile yapılan testlere ait başarımlar oranları ikinci kodek olarak sırası ile G.729, G.726, iLBC ve Speex kullanıldığı durumlar için %62, %87.44, %95 ve %91.2 olarak hesaplanmıştır. Konuşma veri kümesi için ise aynı kodek sırası için %45.46, %81.30, %87.54 ve %79.22 olarak ölçülmüştür. Bu testlere ait sonuçlar Tablo 5.5'de verilmiştir. Test sonuçlarına göre ikinci kodek olarak G.729 kullanıldığı durumlarda başarımlar son derece düşük çıkmıştır. Bu teste ait doğruluk matrisi ise Tablo 5.6'da verilmiştir. Sonuçlar incelendiğinde G.729 kodeğine ait başarımların düşük çıkmasının sebebi, bu kodeğin diğer tüm GSM kodeklerinden daha düşük bit hızına sahip olması olarak açıklanabilir. İkili kodlama senaryolarında, eğer ikinci kodlayıcı ilk kodlayıcıdan daha düşük bir bit

hızına sahip ise ikinci kodlama işlemi ilk kodlamaya ait tüm izleri kaldırmaktadır. Bu sebeple, bu tip durumlarda önerilen yöntem ilk kodeği tespit edememektedir.

Çizelge 5.5: GSM'den VoIP ağına geçiş senaryosu test sonuçları

İkinci Kodek	Başarım (%)	
	Müzik-I	Konuşma-I
g.729	62.00	45.46
g.726	87.44	81.30
ilbc	95.00	87.54
speex	91.20	79.22

Çizelge 5.6: GSM'den PSTN ağına geçiş senaryosunda G.729 kodeği kullanılan teste ait doğruluk matrisi (Konuşma veri kümesi)

			Sınıflandırılan				
			ikili kodlanmış				tek kodlanmış
			amr	awb	gsm	gsm(wav)	g.729
Gerçek	ikili kodlanmış	amr	33.4	11.5	6.1	36.2	12.8
		awb	0.2	89.5	0.6	2.5	7.2
		gsm	4.7	10.8	20.3	27.4	36.8
		gsm(wav)	9	12.4	13.4	35.2	30
	tek kodlanmış	g.729	8.2	7.2	9.9	26.8	47.9

Son deney grubu ise VoIP ağından PSTN ağına geçiş senaryosunu kapsamaktadır. Üç farklı PSTN kodeği olduğu için üç test yapılmıştır. Tablo 5.7 VoIP'den PSTN'e geçişte ikinci kodek olarak u-law kullanıldığı duruma ait doğruluk matrisini göstermektedir. Bu sonuçlar daha öncekiler ile benzerlik göstermektedir. Müzik veri kümesi üzerinde yapılan testlere ait sonuçlar ikinci kodek olarak a-law, PCM ve u-law kullanıldığında sırası ile %97.15, %99.2 ve %96.8 olarak hesaplanmıştır. Konuşma veri kümesi için ise başarımlar aynı sıra için %88.52, %88.67 ve %98.52 şeklinde ölçülmüştür. Tüm bu sonuçlar Tablo 5.8'de sunulmuştur.

Tablo 5.5'deki sonuçlar incelendiğinde, ilk kodek ya da ikinci kodek olarak konuşma kodlayıcıları kullanıldığında, ilk kodek, müzik veri kümesi üzerindeki testlerde konuşma veri kümesindeki testlere göre daha yüksek başarımla tespit edilmiştir. Bu sonuç, konuşma için optimize edilmiş kodekler müzik üzerinde kullanıldıklarında, ses üzerinde daha fazla ayırıcı iz bırakmakta ve bu

Çizelge 5.7: VoIP'den PSTN ağına geçiş senaryosunda u-law kodeği kullanılan teste ait doğruluk matrisi (Müzik veri kümesi)

			Sınıflandırılan			
			ikili kodlanmış			tek kodlanmış
			g.729	g.726	speex	u-law
Gerçek	ikili kodlanmış	g.729	99.2	0	0	0.8
		g.726	0	98.4	1.6	0
		speex	0	10	90	0
	tek kodlanmış	u-law	0.4	0	0	99.6

Çizelge 5.8: VoIP'ten PSTN ağına geçiş senaryosu test sonuçları

İkinci Kodek	Başarım (%)	
	Müzik-I	Konuşma-I
a-law	97.15	88.52
pcm	99.20	88.67
u-law	96.80	98.52

sayede sınıflandırma işleminin daha başarılı olmasını sağlamaktadır. Önceki gözlemlerimize benzer şekilde, Tablo 5.4 ve Tablo 5.8'de verilen sonuçlara bakıldığında, ikinci kodek olarak PSTN kodekleri kullanıldığında, ilk kodlayıcı bir konuşma kodeği de olsa, müzik veri kümesi her zaman daha başarılı sonuçlar vermemektedir. Bunun sebebi PSTN kodeklerinin dizayn prensiplerinin herhangi bir ses içeriğini göz önüne almaması olarak açıklanabilir.

Bu kısımda yapılan testler, önerilen yöntemin iki kere kodlanmış ses örneklerinin ilk kodeğini yeterince yüksek bir başarımla tespit edebildiğini göstermektedir. Buradaki tek kısıt, ikinci kodeğin ilk kodekten daha yüksek bir bit hızına sahip olması gerekmektedir. Bu kısıttan dolayı diğer senaryolar olan PSTN ağından VoIP ve GSM ağına geçiş ile VoIP ağından GSM ağına geçiş testleri göz önüne alınmamıştır.

5.2.1 Karşılaştırma

Literatürde bir verinin kodlanmasında kullanılan algoritmayı belirleyebilmek adına, kodlanmış veriden çıkartılan istatistiksel nitelikleri kullanan iki yöntem

daha mevcuttur. Bu iki yöntemin hedefledikleri uygulama alanları birbirlerinden farklıdır. Referans çalışmalardan [35], bir ağdaki akışları sınıflandırarak içerdiği veri tipini (ses, video, metin, resim) tespit etmeyi amaçlamaktadır. Buna karşın [12], MP3 kodlaması üzerine yoğunlaşmış ve MP3 formatının farklı kodlama şekillerini ayırt etmeyi hedeflemiştir. Önerilen yöntem ile kıyaslandığında, [35] farklı veri tiplerini ayırt etmeyi çalışmakta, [12] ise MP3 kodlama standartına ait detay bilgileri kullanarak formatın uygulama seviyesindeki niteliklerine bakmaktadır. Bizim yöntemimiz bu iki referans çalışmanın ortasında yer almakta, veri tipi olarak sadece ses dosya tipini göz önüne almakta ve herhangi bir kodlama formatından kendisini bağımsız tutmaktadır.

Ses kodlayıcısı tespit etme açısından bakıldığında, [12] çalışmada kullanılan yaklaşım ve belirlenen nitelikler, bizim çalışmamızın içeriğine taban tabana zıttır. Bunun sebebi belirtilen çalışmadaki niteliklerin tamamen MP3 kodlama formatına bağlı olması, ve diğer tüm formatlar için benzer şekilde kodlamaya özgün nitelikler çıkartılmadıkça diğer formatların tespitinin mümkün olmayacağıdır. Bu şekilde her formata özgü nitelikler belirlemek bizim çalışmamızın amacının tamamen dışındadır. Burada tekrar belirtilmelidir ki bizim çalışmamızın temeli, farklı kodlama yöntemlerinin kalite, sıkıştırma ve tekrar miktarı gibi dizayn prensiplerinde oluşan farklılıklardır. Bir kodlama formatının farklı gerçekleştirme türlerinde, yukarıda belirtilen farklılıklar tespit edilemeyecek kadar ufak olacaktır ve bu da bizim yöntemimizi etkisiz kılacaktır.

Kullandığımız rasgelelik nitelikleri ise, [35] nolu çalışmadan esinlenerek oluşturuldu ve kaotik niteliklerin ses kodlayıcısı ayırt etmedeki etkisini tespit etmek adına birçok deney yapıldı. Yapılan testler hem tekli kodlama hem de ikili kodlama senaryolarını kapsamaktadır.

Tekli kodlama testlerinde, tüm testleri sadece [35] nolu çalışmada kullanılan nitelikleri (rasgelelik nitelikleri) kullanarak tekrar ettik. Tablo 5.9 bu testlere ait sonuçları, ve bizim yöntemin başarımı ile olan kıyaslamaları içermektedir. Tüm testler öncekiler ile aynı veri kümeleri üzerinde, aynı örnekleme boyutu ile ve aynı kodlama parametreleri ile yapılmıştır. Hesaplanan ortalama başarımlara göre her iki yöntem de %90'dan yüksek bir başarıma sahiptir ve bizim yöntemimiz tüm testlerde %2 - %4 arasında daha yüksek bir başarıma sahiptir.

Çizelge 5.9: [35] ile önerilen yöntemin tüm veri kümeleri üzerinde yapılan 17 sınıf sınıflandırma testleri ile ortalama başarımlar açısından karşılaştırılması

Yöntemler	Başarımlar (%)			
	Müzik-I (4KB)	Konuşma-I (2KB)	Müzik-II (4KB)	Konuşma-II (4KB)
[35]	91.87	96.72	96.49	96.46
Önerilen yöntem	95.88	97.34	97.91	97.88

Bizim yöntemimizin üstünlüğü ikili kodlama testlerinde daha net gözlemlenebilmektedir. Kısım 5.2'de belirtilen GSM'den PSTN'e, GSM'den VoIP'e ve VoIP'ten GSM ağlarına geçiş senaryoları için her senaryodan rastgele bir deney seçilerek nitelik kümesi dışındaki diğer tüm özellikler aynı şekilde tutularak test edilmiştir. Tablo 5.10 ve 5.11 sırasıyla Müzik-I ve Konuşma-I üzerinde yapılan testlere ait sonuçları içermektedir.

Çizelge 5.10: [35] ile önerilen yöntemin Müzik-I veri kümesi üzerinde yapılan ikili kodlama testleri ile karşılaştırılması

Yöntemler	Başarımlar (%)		
	GSM'den PSTN'e (gsm'den a-law testi)	GSM'den VoIP'e (gsm'den speex testi)	VoIP'den PSTN'e (voip'den pcm testi)
[35]	61.8	73.84	79.85
Önerilen yöntem	92.28	91.20	99.20

Çizelge 5.11: [35] ile önerilen yöntemin Konuşma-I veri kümesi üzerinde yapılan ikili kodlama testleri ile karşılaştırılması

Yöntemler	Başarımlar (%)		
	GSM'den PSTN'e (gsm'den a-law testi)	GSM'den VoIP'e (gsm'den G.726 testi)	VoIP'den PSTN'e (voip'den u-law testi)
[35]	87.98	79.42	88.22
Önerilen yöntem	89.78	81.30	98.52

Sonuçlar incelendiğinde, bizim yöntemimiz Müzik-I veri kümesinde %30'a kadar, Konuşma-I veri kümesindeki testlerde ise %2 ile %10 daha yüksek başarımlar elde etmiştir. Bu sonuçlardan iki önemli karara varılabilir. Bunlardan ilki, kaotik nitelikler ikili kodlamalardan (daha yüksek bit hızı ile yapılan) daha az etkilenmekte ve ikili kodlamalardaki ilk kodeğin tespitinde daha başarılı olmaktadır. İkincisi ise kaotik nitelikler ses içeriğine daha az bağlıdır.

Müzik veri kümesi üzerindeki performans farkının büyük olmasının sebebi, müzik sinyallerinin rasgelelik özelliklerinin ikili kodlamaya karşı daha duyarlı (dayanıksız) olmasıdır.

Çizelge 5.12: Konuşma-I veri kümesi üzerinde yapılan 17 sınıf sınıflandırma testine ait doğruluk matrisi

	Sınıflandırılan																	
	PSTN				GSM				VoIP				Yüksek Kaliteli Sıkıştırma					
	A-law	U-law	PCM	AMR	AWB	GSM	GSM	GSM	G.729	G.726	iLBC	Speex	AAC	MP3	OGG	FLAC	WAV	WMA
PSTN	A-law	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	U-law	6.7	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	PCM	0	0	96	0	0	0	0	3.7	0	0.1	0	0.2	0	0	0	0	0
	AMR	0	0	0	96.7	2	0	0	1.3	0	0	0	0	0	0	0	0	0
	AWB	0	0	0	1	99	0	0	0	0	0	0	0	0	0	0	0	0
GSM	GSM	0	0	0	0	0	97.5	2.5	0	0	0	0	0	0	0	0	0	0
	GSM	0	0	0	0	0	14.2	85.8	0	0	0	0	0	0	0	0	0	0
	(WAV)																	
	G.729	0	0	0	2.3	0.4	0	0	96.2	0	0	0	1.1	0	0	0	0	0
VoIP	G.726	0	0	0	0	0	0	0	0	96	4	0	0	0	0	0	0	0
	iLBC	0	0	0	0	0	0	0	0	0	99.9	0	0	0.1	0	0	0	0
	Speex	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0
	AAC	0	0	0	0.3	0	0	0	0	0	0	0	99.7	0	0	0	0	0
Yüksek Kaliteli Sıkıştırma	MP3	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0
	OGG	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0
	FLAC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0
	WAV	0.5	2.1	0.1	0	0	0	0	0	2.3	0	0	0	0	0	0	0	95
	WMA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Çizelge 5.13: Müzik-I veri kümesi üzerinde yapılan 17 sınıf sınıflandırma testine ait doğruluk matrisi

		Sınıflandırılan																			
		PSTN			GSM				VoIP			Yüksek Kaliteli Sıkıştırma									
		A-law	U-law	PCM	AMR	AWB	GSM	GSM	GSM	GSM	G.729	G.726	iLBC	Speex	AAC	MP3	OGG	FLAC	WAV	WMA	
PSTN	A-law	93.6	4.6	1	0	0	0	0	0	0	0	0.2	0	0	0	0	0	0	0	0.6	0
	U-law	3	96	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	PCM	0	0.4	99.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GSM	AMR	0	0	0	99.4	0	0	0	0	0	0	0	0	0	0	0	0	0.6	0	0	0
	AWB	0	0	0	96.4	0	0	0.2	0	0	0.4	0	0	0	0	0	0	1.4	0	0	0
	GSM	0	0	0	0	92	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VoIP	GSM	0	0	0	0.2	0	96.6	0.8	0	0	0.4	0	0	0	0	0	0	1	0	0	0
	(WAV)																				
	G.729	0	0	0	0	0	0.8	96.8	0	0	0	0	0	0	0	0	0	2.4	0	0	0
Yüksek Kaliteli Sıkıştırma	G.726	0.2	0	0.2	0	0	0	0	0	99.6	0	0	0	0	0	0	0	0	0	0	0
	iLBC	0	0	0	0	0	0.2	0	0	0	98.6	0	0	0	1	0	0.2	0	0	0	0
	Speex	0	0	0	0	0	0.2	0.2	0	0	0	0	0	99	0.2	0	0.2	0.2	0	0	0
Yüksek Kaliteli Sıkıştırma	AAC	0	0	0	0	0	0	0	0	0	0	0	0	0	97.4	1.4	0.4	0.6	0	0	0
	MP3	0	0	0	0	0	0	0	0	0	0	0	0	0.2	2	88.4	6.4	3	0	0	0
	OGG	0	0	0	0	0	0	0	0	0	0	0	0.8	0.2	0.2	7.8	91	0	0	0	0.2
	FLAC	0	0	0	0	0	0.2	0.6	0	0	0	0	0	0	0	2	0	97	0	0	0.2
	WAV	0	0	1.6	0	0	0	0	0	0	0.6	0	0	0	0	0	0	0	0	0	97.8
WMA	0	0	0.8	0	0	0	0	0	0	0	0	0	0	0	6.8	1.6	0	0	0	0	90.8

6. SONUÇLAR VE ANALİZ DEĞERLENDİRME

Bir ses verisinin kodlamasında kullanılan kodeği belirleyebilmek amacıyla hızlı ve başarılı bir yöntem anlatılmıştır. Önerilen yöntem, kodlanmış veriye ait rasgelelik ve kaotik nitelikleri kullanan çok sınıflı bir sınıflandırıcı kullanmaktadır. Yöntem iki grup deney düzeneği ile test edilmiştir. İlk grup testlerde 16 farklı kodeğin birbirlerinden ayırt edilmesi amaçlanmıştır. Sonuçlar incelendiğinde birçok kodlayıcı %95'ten yüksek bir başarımla belirlenmiş olup, birbirleri ile karıştırılan kodeklerin aynı ya da benzer kodlama yöntemleri kullandıkları gözlemlenmiştir.

İkinci grup testlerde ise, kodlanmış ses örnekleri ikinci defa başka bir kodek ile kodlanmış ve yöntemin bu iki kere kodlanmış seslerin ilk kodeğini tespit edebilme yeteneği test edilmiştir. Bu grup testlerde tek kodlanmış sesler ile iki kere kodlanmış sesler birbirlerinden %100'e yakın başarımla ayrılmışlardır. Aynı zamanda iki kere kodlanmış ses örneklerinin ilk kodeği ise %80 civarında başarımla tespit edilmiştir. Bu ikili kodlama testleri için tek kısıt, ikinci kodlayıcı eğer ilk kodeğe göre daha yüksek bir sıkıştırma uyguluyorsa (örn. daha düşük bit hızı), yöntem ilk kodeği yüksek başarımla tespit edememektedir.

Benzer çalışmalardan bazıları testlere aynı kodlayıcının farklı bit hızı opsiyonlarını da dahil etmiştir. Bu çalışmada ise her kodlayıcının sadece bir bit hızı opsiyonu sınıflandırmalarda yer almıştır. Bunun sebebi çalışmanın aynı kodlayıcıların farklı bit hızı opsiyonlarını ayırmaktan ziyade kodeklerin kullandığı farklı tasarım prensiplerinden faydalanarak farklı kodekleri tespit etmektir. Bunun yanında önerilen yöntem aynı kolayıcının farklı bit hızı opsiyonlarını tespit etmek amacıyla

kullanıldığında başarısız olacaktır. Bu durum yapılan testler ile de gözlemlenmiştir. Şu anda kullanımdan kaldırılmış olan ve G.726 kodeğinin içinde ayrı bir bit hızı opsiyonu olarak yer alan G.723 kodeği, testlerin bir kısmında ayrı bir sınıf olarak kullanılmıştı. Ancak bu kodeğin G.726 ile neredeyse hiç ayrıştırılamaması sonucu bu kodlayıcı testlerden tamamen çıkartılmıştır.

Sonuç olarak önerilen yöntem tek ya da çift kodlanmış ses verilerinin kodeğini belirli sayıdaki kodlayıcılar arasından tespit etmede başarılı bir şekilde kullanılabilir. Bu kodeklerin aynı bit hızında çalışan fakat farklı kodlama yöntemleri kullanıyor olmaları ya da aynı kodlama yöntemine sahip fakat birbirlerinden farklı bit hızlarına sahip olmaları yöntemin başarımı açısından önem arz etmemektedir.

7. GELECEK ÇALIŞMALAR

Çok sınıflı sınıflandırma işlemi yapılan tekli kodlama testlerinin, özellikle 17 sınıf sınıflandırma için, çevrim dışı evredeki çalışma zamanları son derece uzun sürmektedir. Bunun sebebi SVM'in iki hiper-parametresi (hyperparameters) olan C (regularization constant) ve γ (kernel hyperparameter) değerlerinin doğru seçilebilmesi için yapılan ızgara arama (grid search) algoritmasının çalışma zamanıdır. Sınıf sayısı arttıkça bu algoritmanın çalışma zamanı ve aynı zamanda SVM'in sınıflandırma zamanı logaritmik olarak artmaktadır. Burada yapılabilecek ilk iyileştirme, sınıflandırma işlemi tüm 17 sınıf için tek adımda yapmak yerine, kodlayıcıları gruplara ayırarak adım adım sınıflandırma işlemi yapmak olabilir. Kodekler öncelikle gruplarına göre etiketlenerek (örn. PSTN, GSM) ilk adımda hangi gruba ait oldukları tespit edilebilir. Şu anda dört kodek grubu olduğu düşünüldüğünde, ilk adımda bir kodeğin bu dört gruptan hangi gruba ait olduğu belirlenir. Daha sonra ise zaten testlerde başarımları verilen kodek grup içi testleri ile kodek tespit edilebilir. Bu sayede daha düşük çalışma zamanı ve kaynak tüketimi ile sınıflandırma işlemi yapılabilir.

Mevcut sınıflandırma süreci, kodek sayısının arttırılmasını güçleştirmektedir. Yeni bir kodek sisteme eklenmek istendiğinde çevrim dışı evredeki sınıflandırıcının eğitilmesi adımı baştan tekrarlanmalıdır. Bu sebeple sınıflandırma işlemleri için öğrenmeli sınıflandırıcılar (örn. yapay sinir ağları) kullanılabilir. Bu sayede sisteme yeni bir kodeğin eklenmesi çok daha kısa zamanda yapılabilecektir. Benzer çalışmalar arasında bu çalışmadaki kadar fazla kodeği sınıflandırmaya çalışması yoktur. Ancak piyasada yüzden fazla kodek olduğu düşünüldüğünde öğrenmeli bir sınıflandırıcı kullanımı daha fazla kodeği kapsamak adına kritiktir.

KAYNAKLAR

- [1] Kurtosis(basiklik), url:<http://en.wikipedia.org/wiki/kurtosis>.
- [2] Psiko-akustik model, url:<http://en.wikipedia.org/wiki/psychoacoustics>.
- [3] Skewness(carpiklik), url:<http://en.wikipedia.org/wiki/skewness>.
- [4] Speex: a free codec for free speech, url:<http://www.speex.org/>.
- [5] Voxforge speech corpus, url:<http://voxforge.org/>.
- [6] ITU-T recommendation G.711. Pulse code modulation (PCM) of voice frequencies, 1988.
- [7] H. Abarbanel. *Analysis of observed chaotic data*. Springer, 1996.
- [8] D. W. Abu-Al-Saud. Sampling pulse code mod. lecture 25. Technical report, 1997.
- [9] D. Alley. Automatic identification of voice band telephony coding schemes using neural networks. *Electronics Letters*, 29(13):1156 –1157, june 1993.
- [10] S. Andersen, A. Duric, H. Astrom, R. Hagen, W. Kleijn, and J. Linden. Internet low bit rate codec (iLBC), 2004.
- [11] V. A. Balasubramaniyan, A. Poonawalla, M. Ahamad, M. T. Hunter, and P. Traynor. Pindr0p: using single-ended audio features to determine call provenance. In *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS 2010, Chicago, Illinois, USA, October 4-8, 2010*, pages 109–120, 2010.

- [12] R. Böhme and A. Westfeld. Statistical characterisation of mp3 encoders for steganalysis. In *Proceedings of the 2004 workshop on Multimedia and security*, pages 25–34, 2004.
- [13] M. Bosi and R. E. Goldberg. *Introduction to Digital Audio Coding and Standards*. Kluwer Academic Publishers, Norwell, MA, USA, 2002.
- [14] J. Bradbury. Linear predictive coding. 2000.
- [15] K. Brandenburg. Mp3 and aac explained. In *Audio Engineering Society Conference: 17th International Conference: High-Quality Audio Coding*, 1999.
- [16] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.
- [17] Cisco. Waveform coding techniques. Technical report, 2006.
- [18] D. Cişmpeanu and A. Cişmpeanu. Peaq i£i an objective method to assess the perceptual quality of audio compressed files. In *Proceedings of the International Symposium on System Theory, SINTES 12*, pages 487 – 492, 2005.
- [19] E. Ekudden, R. Hagen, I. Johansson, and J. Svedberg. The adaptive multi-rate speech coder. In *Speech Coding Proceedings, 1999 IEEE Workshop on*, pages 117 – 119, 1999.
- [20] ETSI. Digital cellular telecommunications system (phase 2+), full rate speech, transcoding, (gsm 06.10 version 7.0.0 release 1998). Technical report.
- [21] D. Griffin and J. Lim. Multiband excitation vocoder. In *IEEE Transactions on Acoustics, Speech and Signal Processing*, volume 36, pages 1123–1235, 1988.
- [22] C. R. Hilborn. *Chaos and Nonlinear Dynamics*. Oxford University Press, 2000.
- [23] T. Instruments. A-law and mu-law companding implementations using the tms320c54x. Technical report, 1997.

- [24] ITU-T Recommendation G.722.2. Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB). Technical report, International Telecommunication Union, 2003.
- [25] F. Jenner. *Non-intrusive Identification of Speech Codecs in Digital Audio Signals*. 2011.
- [26] F. Jenner and A. Kwasinski. Highly accurate non-intrusive speech forensics for codec identifications from observed decoded signals. In *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 1737–1740, march 2012.
- [27] L. Juan, L. Biqin, and F. Qiuliang. An 8-kb/s conjugate-structure algebraic celp (cs-acelp) speech coding. In *Signal Processing Proceedings, 1998. ICSP '98. 1998 Fourth International Conference on*, volume 2, pages 1729–1732, 1998.
- [28] W. B. Kleijn and K. K. Paliwal, editors. *Speech Coding and Synthesis*. 1995.
- [29] O. Kocal, E. Yuruklu, and I. Avcibas. Chaotic-type features for speech steganalysis. *Information Forensics and Security, IEEE Transactions on*, 3(4):651–661, 2008.
- [30] J. M. Nichols, C. C. Olson, J. V. Michalowicz, and F. Bucholtz. The bispectrum and bicoherence for quadratically nonlinear systems subject to non-gaussian inputs. *Trans. Sig. Proc.*, 57(10):3879–3890, Oct. 2009.
- [31] H. K. R. Hegger and T. Schreiber. Practical implementation of nonlinear time series methods: The tisean package, *chaos* 9, 413, 1999.
- [32] A. Rukhin, J. Soto, J. Nechvatal, E. Barker, S. Leigh, M. Levenson, D. Banks, A. Heckert, J. Dray, S. Vo, A. Rukhin, J. Soto, M. Smid, S. Leigh, M. Vangel, A. Heckert, J. Dray, and L. E. B. Iii. A statistical test suite for random and pseudorandom number generators for cryptographic applications, 2001.
- [33] K. Scholz, L. Leutelt, and U. Heute. Speech-codec detection by spectral harmonic-plus-noise decomposition. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, volume 2, pages 2295–2299 Vol.2, nov. 2004.

- [34] M. Schroeder and B. Atal. Code-excited linear prediction(celp): High-quality speech at very low bit rates. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '85.*, volume 10, pages 937 – 940, 1985.
- [35] K. Shanmugasundaram, M. Kharrazi, and N. Memon. Nabs: a system for detecting resource abuses via characterization of flow content type. In *Computer Security Applications Conference, 2004. 20th Annual*, pages 316 – 325, 2004.
- [36] B. F. Takens. *Dynamical Systems and Turbulance, Lecture Notes in Mathematics*, volume 898. Springer, 1981.
- [37] I. T. Union. G.726: 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM). *SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS; General Aspects of Digital Transmission Systems: Terminal Equipments*, 1990.
- [38] D. Zongker and A. Jain. Algorithms for feature selection: An evaluation. In *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, volume 2, pages 18 –22 vol.2, aug 1996.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : HİÇSÖNMEZ, Samet
Uyruğu : T.C.
Doğum tarihi ve yeri : 19.10.1987 İstanbul
Medeni hali : Bekar
Telefon : +90505 714 90 54
Faks :
e-mail : shicsönmez@etu.edu.tr

Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Y. Lisans	TOBB Ekonomi ve Teknoloji Üniversitesi	2013
Lisans	İstanbul Teknik Üniversitesi	2009

İş Deneyimi

Yıl	Yer	Görev
2012-halen	TÜBİTAK SAGE	Araştırmacı

Yabancı Dil

İngilizce (Çok iyi)

Yayınlar

Hicsönmez, S., Sencar, H., Avcibas, I., “Audio codec identification through payload sampling”. Information Forensics and Security (WIFS), 2011.

Hicsönmez, S., Sencar, H., Avcibas, I., “Classification of audio codecs based on coding characteristics”. Signal Processing and Communications Applications

Conference (SIU), 2012.

Hicsonmez, S., Sencar, H., Avcibas, I., “Audio codec identification from coded and transcoded audios”. *Digital Signal Processing*, 2013.

Hicsonmez, S., Uzun, E., Sencar, H., “Methods for identifying traces of compression in audio”. *Communications, Signal Processing, and their Applications (ICCSPA)*, 2013.