

**T.C.**  
**DİCLE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**WAVELET ( DALGACIK DÖNÜŞÜMÜ ) VE YAPAY SİNİR  
AĞI KULLANARAK SES SİNYALİNDEN KONUŞMACI  
TESPİTİ**

**Murat İKİZ**

**YÜKSEK LİSANS TEZİ**

**( ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI )**

**DIYARBAKIR**  
**AĞUSTOS - 2006**

T.C  
DİCLE UNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜ  
DIYARBAKIR

Murat İKİZ, tarafından yapılan "Wavelet (Dalgacık Dönüşümü) ve Yapay Sinir Ağı Kullanarak Ses Sinyalinden Konuşmacı Tespiti" konulu bu çalışma , jürimiz tarafından Elektrik-Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir

Jüri Üyesinin

Ünvanı                      Adı Soyadı

Başkan: Doç. Dr. Mehmet AKIN

Üye : Yrd. Doç. Dr. M.Bahattin KURT

Üye : Yrd. Doç. Dr. Z.Fuat TOPRAK

*Mehmet Akın*  
*M. Bahattin Kurt*  
*Z. Fuat Toprak*

Tez Savunma Sınavı Tarihi: 28/08/2006

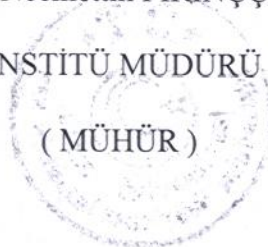
Yukarıdaki bilgilerin doğruluğunu onaylarım.

28/8/2006

*N. Pirinçcioğlu*  
Prof. Dr. Necmettin PİRİNÇÇİOĞLU

ENSTİTÜ MÜDÜRÜ

( MÜHÜR )



## TEŞEKKÜR

Tez çalışmamın gerçekleşmesi sürecinde yardımlarından dolayı tez danışmanı hocam Sayın Doç. Dr. Mehmet AKIN'a teşekkür ederim. Kıymetli vakitlerinden ayırarak çalışma boyunca yardımlarını esirgemeyen Muhittin BAYRAM hocama da ayrıca teşekkür ederim.

Tez çalışmam boyunca kendilerine ayıramadığım vakitler için daima sabretmiş olan eşim Zübeyde ve kızım Dilşah Gülay'a da çok teşekkür ederim.

Ayrıca, yüksek lisan boyunca anlayış birliği içinde olduğumuz diğer tez arkadaşlarıma da yardımlarından dolayı minnettarım.

Çalışmada kullanılmak üzere ses örneklerinin alınması esnasındaki yardımları ve destekleri için çok kıymetli arkadaşlarım; Y.Can ŞENCAN, Nail BÜLBÜL, Bilgehan BÜLBÜL, Haluk ERHAN, Eda ERHAN, Hasan KAPTANOĞLU, Hayal KAPTANOĞLU ve Mehmet ERŞİL'e de ayrıca çok teşekkür ederim.

## İÇİNDEKİLER

<b>ÖZET.....</b>	<b>VI</b>
<b>1. GİRİŞ .....</b>	<b>1</b>
1.1 GİRİŞ VE AMAÇ.....	1
1.2 SESİN ÖZELLİKLERİ.....	2
1.2.1 Ses Nedir?.....	3
1.2.1.1 Anatomi.....	3
1.2.2 Konuşma.....	6
1.2.2.1 Ses Yolunun İşlevi ve Akustik Analiz.....	6
1.2.2.2 Ses Siteminde Uyarım.....	6
1.2.2.3 İnsan Sesinin Özellikleri.....	7
1.2.3 İşitme.....	8
1.3. SES TANIMA.....	9
1.3.1 Genel Bilgiler.....	9
1.3.2 Ses Tanıma Süreci.....	12
1.3.2.1 İşlenmemiş Konuşma.....	14
1.3.2.2 Sesin Sayısal Kodlanması.....	14
1.3.2.3 Sinyal İşleme Teknikleri (Sinyal Analizi).....	16
1.3.2.4 Ses Sinyalinin Modellenmesi.....	18
1.3.3 Ses Tanıma Kullanıcıları ve Kullanım Alanları.....	18
1.3.4 Ses Tanımanın Avantajları ve Sınırlamaları.....	19
1.3.5 Ses Tanıma.....	21
1.3.5.1 Ses İşlemede Amaç.....	21
1.3.5.2 Ses İşleme.....	23
1.3.5.3 Konuşmacı Tanıma.....	24
1.3.5.4 Konuşma Tanıma İçin Ses Özellikleri.....	25
<b>2. MATERYAL VE METOD.....</b>	<b>27</b>
2.1 SAKLI MARKOV MODELLERİNİN SES TANIMADA KULLANILMASI...35	
2.1.1 Giriş.....	27
2.1.2 Bir SMM Yapısı.....	29
2.1.3 Olasılık Değerlendirmesi.....	30

2.1.3.1 İleriye-Dönük Algoritma (Forward Algorithm).....	32
2.1.3.2 Geriye-Dönük Algoritma (Backward Algorithm).....	32
2.1.4 “En uygun” Durum Dizisinin Hesaplanması.....	32
2.1.4.1 Viterbi Algoritması.....	33
2.1.5 Parametre Tahmini.....	33
<b>2.2 YAPAY SİNİR AĞLARININ SES TANIMADA KULLANILMASI.....</b>	<b>34</b>
2.2.1 Kavramlar ve Terimler.....	34
2.2.2 Yapay Sinir Ağlarının Sınıflandırılması.....	38
2.2.2.1 Tek katmanlı İleri Beslemeli Ağlar.....	39
2.2.2.2 Çok katmanlı İleri Beslemeli Ağlar.....	40
2.2.2.3 Geri Dönümlü Ağlar.....	41
2.2.2.4 Kafes Yapılı Ağlar.....	41
2.2.3 Yapay Sinir Ağlarında Öğrenme.....	42
2.2.3.1 Güdümlü Öğrenme.....	42
2.2.3.2 Güdümsüz Öğrenme.....	43
2.2.4 Çok Katmanlı Perceptron ve Geri Yayılımlı Öğrenme.....	43
2.2.5 Ses Tanımada Yapay Sinir Ağları.....	44
2.2.5.1 Nöron Ağları Yaklaşımı.....	47
2.2.5.2 Nöron Ağlarının Sesli İfade Tanımada Kullanılması.....	49
2.2.5.3 Zaman Gecikmeli Nöron Ağları(TimeDelay Neural Network).....	50
2.2.5.4 Yineli Nöron Ağları (Recurrent Neural Network).....	50
2.2.5.5 Öz Düzenleyici Özellik Haritası (Self Organizing Feature Map).....	51
<b>2.3 SES İŞLEMEDE ZAMAN DOMENİ METODLARI.....</b>	<b>53</b>
2.3.1 Sesin Zamana Bağlı Olarak İşlenmesi.....	53
2.3.2 Kısa Zaman Enerjisi ve Ortalama Genlik.....	54
2.3.3 Kısa Zamanlı Ortalama Sıfır Geçiş Hızı.....	57
2.3.4 Enerji ve Sıfır Geçiş Hızı Kullanarak Ses İle Sessizliği Ayırt Etme.....	59
2.3.5 Kısa zamanlı Özilinti İşlevi.....	61
2.3.6 Özilinti İşlevini Kullanarak Perde Peryodunu Kestirme.....	64
<b>2.4 DOĞRUSAL ÖNGÖRÜM KODLAMASI.....</b>	<b>66</b>
2.4.1 Kanal Ses Kodlayıcıları.....	66
2.4.2 Ötümlü/Ötümsüz Kararı.....	68

2.5 SESLİ İFADE TANIMA.....	69
2.5.1 Sesli İfade Tanıma Sistemlerinin Sınıflandırılması.....	74
2.5.2 Sesli İfade Tanıma Teknikleri.....	76
2.5.2.1 Önişleme ve Özellik Vektörü Çıkarma Teknikleri .....	77
2.5.2.1.1 Parametrik Yaklaşım.....	79
2.5.2.1.2 Parametrik Olmayan Yaklaşım.....	79
2.5.2.2 Özellik Vektörlerinin Sınıflandırması Teknikleri.....	79
2.5.3 Özellik Vektörlerinin Çıkarılması.....	82
2.5.3.1 Kısa Süreli Özellik Vektörleri.....	82
2.5.3.1.1 Ham Veriler.....	82
2.5.3.1.2 Spektrum.....	84
2.5.3.1.3 Cepstrum.....	88
2.5.3.1.4 LPC (Linear Predictive Coding) Tabanlı Cepstrum Değerleri.....	91
2.5.3.1.5 PLP (Perceptual Linear Prediction).....	94
2.5.3.2 Uzun Süreli Özellik Vektörleri.....	94
2.5.3.3 Sesli İfadenin Kesimlenmesi.....	95
2.5.3.4 Temel Frekansın Belirlenmesi(Pitch Estimation).....	97
2.5.3.4.1 Sesli İfade Dalgasının İşlenmesine yönelik Yöntemler.....	97
2.5.3.4.2 Korelasyon Kullanımına Dayalı Yöntemler.....	98
2.5.3.4.3 Spekturumun İşlenmesine Dayalı Yöntemler.....	98.
2.5.3.5 Sıfır Noktasını Geçme Sayısı(Zero Crossing Rate).....	108
2.5.3.6 Kısa Süreli Enerji Ve Ortalama Genlik.....	100
2.6 DALGACIKLAR TEORİSİ.....	102
2.6.1 Giriş.....	102
2.6.2 Hilbert Uzayı Analizi.....	103
2.6.3 Dalgacıklara Genel Bakış.....	105
2.6.4 Tarihsel Bakış.....	106
2.6.4.1 Baz Fonksiyonları Nedir?.....	108
2.6.4.2 Ölçeği Değişen Baz Fonksiyonları Nedir?.....	109
2.6.5 Fourier Analizi.....	111
2.6.5.1 Fourier Dönüşümleri.....	111
2.6.5.1.1 Ayrık Fourier Dönüşümleri.....	111

2.6.5.1.2 Pencerelemiş Fourier Dönüşümleri.....	111
2.6.5.1.3 Hızlı Fourier Dönüşümleri.....	112
2.6.6 Fourier İle Dalgacık Dönüşümlerinin Karşılaştırılması.....	112
2.6.6.1 Fourier ve Dalgacık Dönüşümleri Arasındaki Benzerlikler.....	112
2.6.6.2 Fourier ve Dalgacık Dönüşümleri Arasındaki Farklılıklar.....	113
2.6.7 Sürekli Dalgacık Dönüşümü.....	115
2.6.7.1 Tanım ve Basit Özellikler.....	115
2.6.7.2 Dalgacıklar Neye Benzer?.....	115
2.6.7.3 Dalgacık Analiz Teknikleri.....	121
<b>3. BULGULAR.....</b>	<b>125</b>
3.1 Önceki Çalışmalar.....	125
3.2 Seslerin Dosyalara Kaydedilmesi.....	126
3.3 Sesin Karakteristik Değerinin Bulunması.....	130
3.4 YSA 'nın Eğitilmesi.....	135
3.5 Verilerin Test Sonuçları.....	143
3.6 Ek Çalışmalar.....	148
<b>4. SONUÇLAR VE TARTIŞMA.....</b>	<b>157</b>
<b>KAYNAKLAR.....</b>	<b>159</b>
<b>ÖZGEÇMİŞ.....</b>	<b>161</b>

## ÖZET

Bu çalışmada amaçlanan; mikrofon yardımı ile kayıt edilen ses sinyallerinden wavelet ve yapay sinir ağı yardımı ile konuşmacı kimliğinin tespit edilmesidir. İlk önce 10 farklı kişiden ( 6 erkek ve 4 kadın ) ses örnekleri alınmıştır. Shareware yazılımlar olan Waveflow ve Wavepad programlarını kullanarak ses sinyalleri gürültüden temizlenmiştir. Matlab Simulink ile hazırlanan model yardımı ile seslere ait kullanılabilir veriler elde edilmiştir. Veri elde edilmesi esnasında Wavelet ayrımı kullanılmıştır. Elde edilen bu veriler Matlab ortamında hazırlanan Yapay Sinir Ağı'na giriş veri sinyalleri olarak uygulanmıştır. Bu sayede YSA'nın farklı kişilerin konuşmalarını sınıflandırması sağlanmıştır.

Bölüm 1'de; sinyal işleme yaklaşımı ve kullanım alanları geniş olarak anlatılmıştır. Ayrıca, kişi tanıma yönteminin avantajları ve dezavantajları ile kısıtlamaları hakkında kritikler yapılmıştır. Bu bölümle ilave olarak kişi tanıma yönteminin safhaları hakkında bazı temel bilgiler de verilmiştir. Bu safhalar; sesin kaydedilmesi, kodlanması, sinyal işleme teknikleri ve sinyal modellemesidir.

Bölüm 2'de; Yeni bir yaklaşım sunan bu çalışmanın daha iyi anlaşılabilmesi için Saklı Markov Modelleri yönteminin teori ve kullanımı, ses sinyalinin modellenmesinde yapay sinir ağları, ses işlemeye zaman domeni yaklaşımı, ses kodlama ve modelleme için Doğrusal Öngörüm Kodlaması, Wavelet teorisi ve uygulama alanları konuları hakkında bilgiler sunulmuştur.

Bölüm 3'de; Bu çalışma esnasında yapılmış olan tüm çalışmalardan bahsedilmiştir. Bu çalışmalar; sesin kaydedilmesi, ses işleme, Wavelet ayrımı yardımı ile sabitlerin elde edilmesi, YSA modelinin hazırlanması, sabitleri YSA modeline giriş verisi olarak uygulanması ve YSA çıkış verilerinin yorumlanması çalışmalarıdır.

Bölüm 4'de; Çıkan sonuçlar değerlendirilmiştir ve gelecek çalışmalar hakkında düşünceler elde edilmiştir.



Anahtar Kelimeler : Ses Tanıma, Yapay Sinir Ağları, Hızlı Fourier Dönüşümü, Saklı Markov Modelleri, Konuşmacı Bağımlı Ses Tanıma, Wavelet, Dalgacık Teoremi.

## ABSTRACT

The purpose of this study is to recognize the speaker identity by means of wavelet analysis and neural network approach. Firstly, sampling the voice signal generated from 10 different person ( 6 males and 4 females ), we extracted noise from voice signals by using Waveflow and Wavepad shareware programs. With the help of a Matlab Simulink model we generated the useable data from this voice signals. Wavelet approach has been used for getting these datas. These datas have been used as an input signal for Matlab based Neural Network. This neural network classified the voice data for different speakers.

In Section-1 ; The signal processing concept and it's using area has been largely explained. Additionally, the advantages, disadvantages and the limitations of the speaker recognition process have been criticized in the same section. Furthermore, some basic acknowledgements about the phases of speaker identification process have been presented in this section. These phases are recording the voice signal, coding the signal, signal processing technics ( signal processing ) and signal modeling.

In Section-2 ; For making easier the understanding of this study, new approach to speaker recognition, The process of HMM ( Hidden Markov Models ), theory and usage, Neural Networks used in modeling the voice signal, Time domain approach to signal processing , LPC ( Linear Prediction Coding ) for signal modeling and coding, Wavelet Theory and application areas have been underlined in this section.

In Section-3 ; All the work done for this study; signal recording, signal processing, getting wavelet constants from the voice signal, neural network model preperation, working with the model and interpreting the output datas are briefly explained.

In Section-4 ; The result and feature projects are explained.

Key Words:

Voice recognition, Neural Networks, Fast Fourier Transformation, Hidden Markov Models, Speaker Dependent Speaker Recognition, Wavelet.

## 1. GİRİŞ

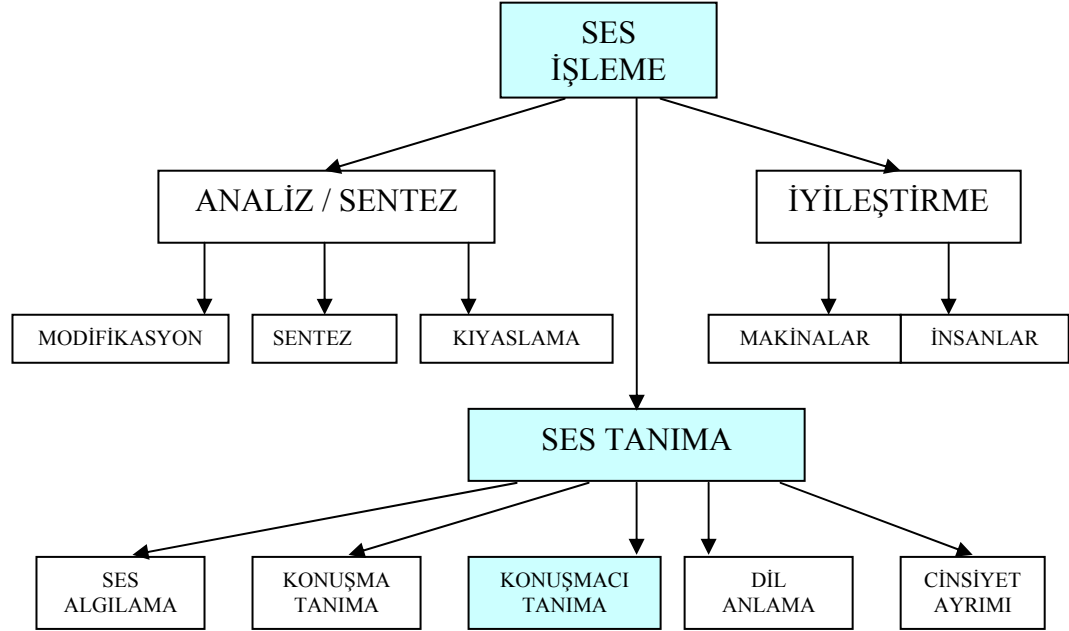
### 1.1 GİRİŞ VE AMAÇ

Ses doğadaki canlılar arasında kullanılan en temel iletişim araçlarından biridir. Sesin insanlar arasında iletişim amacıyla kullanılmasıyla konuşma ortaya çıkmıştır. Dolayısıyla konuşma insanların sahip olduğu en temel iletişim araçlarından biri olmuştur. Başka bir ifadeyle, konuşma insanlar arasında bilgi taşıyan ve bunun için başka bir aracı gerektirmeyen en önemli iletişim aracıdır.

Konuşma sesi, dil bilgisini, konuşmacının özelliklerini, konuşmacının duygularını taşıyabilir. Konuşarak iletişim yaşamımızda çok önemli bir rol oynamaktadır. Konuşmanın ses içeriği ve dilsel içeriği insanın kültürel yapısına göre değişiklik göstermektedir. Konuşmanın temeli içerdiği sesli ifadelerdir. Ancak sesli ifadenin etki sınırları ortamla kısıtlı olduğundan sesli ifadenin elektriksel sinyallere dönüştürülmesi gereklidir. Bu amaca yönelik ilk gerçekleştirim telefon ile sağlanmıştır. Telefon seslerin elektriksel sinyallere dönüştürülmesine dayanır. Telefonun icadı iletişimin tarihsel gelişimi içinde en önemli adımdır. Sesli ifade üzerinde araştırmalar telefonun icadı ile başlamıştır. Bu araştırmalar sonucunda ilk sesli ifade kodlama tekniği olarak 1938’de Genlik kodlaması modülasyon (*PCM, Pulse Code Modulation*) tekniği bulunmuştur. Bu yöntemle sesli ifade örneksel olarak kodlanmaktadır. 1960’dan sonra sayısal elektronik devrelerin ve bilgisayarların gelişmesiyle sesli ifadenin sayısal olarak kodlanması da mümkün olabilmıştır. Sesli ifade tanıma, bir sesli ifade sinyalinin sınırlı sayıda simgelerle ifade edilmesi olarak tanımlanabilir. Amaç ses sinyalinin yazılı eşdeğerinin bulunmasıdır. Sesli ifadenin yazılı karşılığının bulunması insan-bilgisayar arası iletişimi çok daha kolay bir hale getireceğinden, bilgisayarın daha kullanışlı bir duruma gelmesini sağlar. Yani insan-bilgisayar arası iletişim doğal bir olay haline gelir. Bu nedenle, bu konu çok rağbet görmüştür.

Bu güne kadar ses işleme ve ses tanıma konularında yapılmış olan ulusal çalışmalar incelenmiş ve “ Bölüm 3.5 Ek Çalışmalar” kısmında açıklanmıştır.

Tez çalışması konusu olarak ele alınan ses tanıma, ses işlemenin bir alt basamağı olarak düşünülebilir. Ses işleme ve ses tanıma konularını düzenleyen tipik bir sınıflandırma Şekil 1.1’ de verilmiştir.



Şekil 1.1 Ses İşlemenin Sınıflandırılması

## 1.2 SESİN ÖZELLİKLERİ

Ses tanıma, bilgisayar bilimi ve matematiği de içerisine alan birçok alanın ilgisi olmuş büyüleyici bir konudur. Güvenilir bir ses tanıma birçok tekniğin birlikte kullanılmasını zorunlu kılan oldukça zor bir iştir ancak modern yöntemler sayesinde daha yüksek seviyelerde sonuçlar alınmaya başlanmıştır. Genel olarak ses tanımada kullanılan yöntemleri tanımak ve bu yöntemler arasından doğruluk seviyesi ve kullanım kolaylığı esas alınmak üzere bir yöntem seçimi yapılarak başlanılan bu çalışmada öncelikle kullanım materyali olan ‘ses’ sinyalini tanımak ve uygun yöntemler için bilinmesi gerekenler üzerinde durulacaktır.

## 1.2.1 Ses Nedir?

### 1.2.1.1 Anatomi

Ses dalgası, ses üretim sistemini meydana getiren anatomik yapıların istemli hareketleri sonucunda oluşan, akustik bir basınç dalgasıdır. Bu sistemin ana bölümleri ciğerler, nefes borusu, gırtlak, boğaz, ağız boşluğu ve burun boşluğudur. Teknik terim olarak boğaz ve ağız boşluğu 'ses yolu' olarak tanımlanır. Dolayısıyla ses yolu, gırtlak çıkışından başlayıp, dudaklarda sona erer. Burun yolu ise damaktan başlar burun deliklerinde sona erer. Ses üretimi için kritik olan anatomik yapılar, ses telleri, damak, dil, dişler ve dudaklardır. Ağız iyice açıldığı zaman ağız boşluğunun arka tarafında duran damağın yumuşak uzantısına 'küçük dil' denir. Ses yolunu oluşturan bu anatomik yapılar, farklı pozisyonlar alarak değişik sesleri oluştururlar. Çene de ses yolunun şekil değişimini etkilediği için bu yapıların arasında yer alabilir.

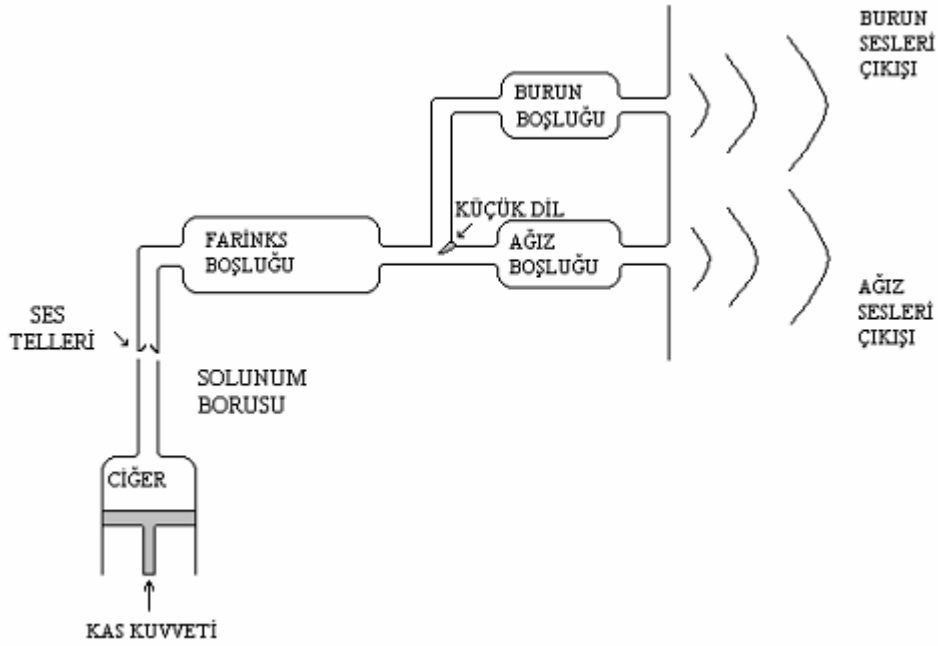
Ses üretimi bir akustik filtreleme işlemi olarak düşünülebilir.. Akustik filtre, ses üretim yollarının özelliklerini gösterir [1].

Ses yolu uzunluğu, erişkin bir kadında yaklaşık ortalama 14 cm, erkekte 17cm civarındadır. Ses yolunu oluşturan yapıların, değişik konumlar alarak sesi oluşturma sırasında, ses yolunun kesit alanı; 0-20  $cm^2$  arasında değişir. Ses yolunun temel yapısı Şekil 1.2'de görülmektedir. Burun boşluğu, ses iletiminde işlevi olan bir yapı olup, uzunluğu erkeklerde yaklaşık 12 cm'dir. Burun boşluğunun ses üretiminde ve iletimindeki kontrolü, damak, dolayısıyla küçük dil tarafından gerçekleştirilir. Bu sebeple burun yolu, ağızdan yayılan ses dalgası sıklık karakteristiğini etkiler. Açılıp kapanan küçük dil, burun seslerinin oluşumunu kontrol eder. Ses yolunun yapısı Şekil 1.2'de gösterilmiştir [2].

Gırtlak, ses üretiminde basit ama önemli bir role sahiptir. Ötümlü denilen seslerin üretilebilmesi için, ses üretim sistemine periyodik bir uyarım sağlar. Ötümlü

sesler, ses tellerinin periyodik titreşimleri sayesinde oluşur. Anatomik açıdan gırtlak oldukça karmaşık bir organdır [2].

Gırtlakta bulunan kıkırdak dokular birbirlerine zar ve bağlarla bağlıdır. Bunlar gırtlığın işlevlerini yerine getirmesi için yapılara hareket sağlarlar. Ses telleri, kaslardan ve zarlardan oluşan elastik bant çifti şeklindedir.



Şekil 1.2 Ses yolunun yapısı

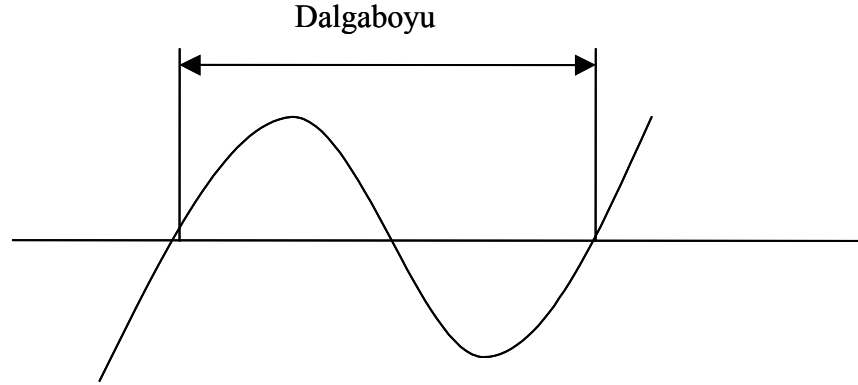
Ses, insan kulağını etkileyerek işitme duyusu oluşturan hava molekülleri titreşimleri, ya da bunların neden olduğu ufak hava basınç değişimleri gibi, ya da bu fiziksel olayın neden olduğu işitsel izlenim gibi tanımlanır.

Ses fizyolojisi ile ilgili bazı önemli kavramları şöyle açıklanmaktadır:

*Ses Dalgası* : Ses sıvı, katı, gaz ortamlarında 20 Hz ile 20 KHz arasındaki insan kulağının algılayabileceği basınç değişiklikleri olarak tanımlanmaktadır. Bu frekans aralığındaki mekanik dalgalar işitme duyumuzu uyardıklarından, bizim için özellikle önemli olan ses dalgalarını oluştururlar.

İnsan kulağına bir ses dalgası geldiğinde kulak ses dalgasındaki basınç değişikliklerini sinirlerdeki vuruşlara çevirir ve bunlar beyinde duyulan sesler olarak yorumlanır.

En basit ses dalgasının sadece bir frekansı ve sabit bir genliği vardır. Buna sinüs dalgası adı verilir. Şekil 1.3’de basit bir sinüs dalgası grafiği görülmektedir.



Şekil 1.3 Basit bir sinüs dalgası

*Frekans* : İki tepe arasındaki uzaklık dalga boyudur. Bir saniyede gözlenen dalga tepesi sayısına frekans denir. Frekans sesin tizliğini belirler. Saniyedeki çevrim (Cycle Per Second-CPS) veya Hertz (Hz) ile gösterilir. Düşük frekanslar bas sesler, yüksek frekanslar ise yüksek seviyeli seslerdir.

*Genlik* : Sesin diğer bir karakteristik özelliği genliğidir. Sesler yumuşak veya yüksek olurlar. Bu havada, havayı sıkıştırmak için kullanılan güce bağlı olan bir tür küçük veya büyük basınca karşılık gelir. Ses gücü veya seviyesi için desibel (dB) birimi kullanılmaktadır. Kulağın algılama özelliği ile ilgili yapılan çalışmalarda ses gücünün artması ile hissedilen ses artışının doğrusal olmadığı ve logaritmik bir ses şiddeti ile duyma olduğu anlaşılmıştır. Bu nedenle algılanan ses logaritmik bir büyüklüktür. Haberleşme sistem ve cihazlarının yapısı ve ölçü birimleri de bu sebepten logaritmik olarak düzenlenmiştir.



*Gürültü* : Periyodik olmayan titreşimlerdir. Kulağın, teknik duyumu bakımından sınırları zorlayan ve psikolojik rahatsızlık doğuran seslere gürültü diyoruz [3].

## 1.2.2 Konuşma

### 1.2.2.1 Ses Yolunun İşlevi ve Akustik Analiz

İnsan sesinin dalga şeklini incelediğimizde, fiziksel sistem zamana bağlı olarak değiştiği için, dalga şeklinin de zamanla değiştiğini görürüz. Konuşma sesleri, kısa süreler boyunca benzer akustik özellikler gösteren ses parçalarına ayrılabilir.

Ses sinyallerinin zamana bağlı dalga şekillerine bakılarak, sinyal periyotları, yoğunlukları, süreleri ve her bir ses parçasının sınırları tespit edilebilir. Ancak, arka arkaya gelen sesler birbirlerini etkilerse bu sınırları belirlemek mümkün olmayabilir. Buna, ‘birleşik telaffuz’ (coarticulation) denir. İnsanların ses üretme ve sesi algılama sistemlerindeki organların yapılarından kaynaklanan bazı sınırlamalardan dolayı, ses dalga şeklinde değişime sebep olan yapıların pozisyonlarının kısa zaman aralıklarında değişmediği farz edilebilir [4].

Sistem modelleme yönünden, ses filtresinin özelliklerini ses filtresini oluşturan yapılar belirler. Konuşma sinyali yaklaşık 7-8 KHz’lik bir bant genişliğine sahiptir ancak çok daha düşük bant genişliklerinde de anlaşılabilirliğini korur. Konuşma sinyalinin izgesi incelendiğinde, bazı seslerde, ses yolunun rezonans sıklıklarına karşı gelen sıklıklarda yüksek genlikli tepelikler görülür. Bu sıklıklara ‘formant sıklıkları’ ya da kısaca ‘formant’ adı verilir. Bir seste sonsuz tane formant sıklığı olmasına karşın, örneklemeden sonra Nyquist bandında görülen ilk 3 veya 4 sıklık sistemi karakterize eder [4].

### 1.2.2.2 Ses Sisteminde Uyarım

Sesin başlıca özelliği, uyarım şeklidir. İki temel uyarım şekli vardır. Bunlar ötümlü ve ötümsüz uyarımlardır. Diğer dört tip uyarım ise; ötümlü uyarım, ötümsüz

uyarım ve sessizliğin birleşimi sonucunda oluşan karışık, patlamalı, fısıltı ve sessiz uyarıdır.

Ötümlü sesler, havanın nefes borusundan veya ses telleri arasından geçmeye zorlanmasıyla gırtlakta üretilir. Ses tellerinin gerginliği, salınım sıklığını belirler. Ses telleri ortalama, erkeklerde dakikada 110 kez, kadınlarda ise yaklaşık bunun iki katı olmak üzere yarı periyodik olarak açılıp kapanırlar. Hava akımının periyodik olarak kapatılması, ses yolunu uyaran yarı periyodik hava akımının üretilmesini sağlar. Gırtlak tarafından üretilen sinyal, ötümlü ses sinyalidir.

Ötümsüz sesler, hava akımının sıkıştırılmış ses yolundan geçmeye zorlanması ile üretilirler ('sistem' deki 's' gibi).

Birbiri ardınca, ötümlü- ötümsüz olarak üretilen seslere 'karışık sesler' denir.

Patlamalı seslerde ses yolunun son kısmı kapatılarak içeride basınç oluşması sağlanır. Basıncılı hava, aniden bırakılarak bu sesler üretilir ('bot'taki 't' ve 'b', buna örnektir).

Uyarım şekli ne olursa olsun ses yolu, belli sıklıkları çok bazılarını ise az bastıran bir filtre gibi davranmaktadır [3].

### 1.2.2.3 İnsan Sesinin Özellikleri

Ses (konuşma) sinyalleri durağan olmayan sinyallerdir. Eğer bu ses parçalara bölünürse 5-20 milisaniyelik temel elemanlardan oluştukları görülebilir. Konuşma sinyalleri sesli (voiced), sessiz (unvoiced) ya da ikisinin karışımı olabilir. Burada sesli diye bahsedilen bildiğimiz sesli harfler, sessiz diye bahsedilen de geriye kalan harflerin telâffuzudur. Sesli sinyalin enerjisi normalde sessiz sinyale göre oldukça yüksektir.

Sesli konuşma gırtlığın, titreşen ses telleri tarafından ürettiği hava vuruşlarıyla tahrik edilmesi sonucu oluşur. Ses telleri periyodik salınımlar oluşturur

ve bu salınımların frekanslarına Temel Frekans adı verilir. Sessiz konuşma ise gırtlaktaki bir boşluktan havanın burun bölgesine zorlanmasıyla oluşturulur. “N” gibi burunsal sesler, gırtlığın akustik kaplinlenmesiyle (sürekli titreşim) oluşturulur. “P” gibi darbeli sesler ise ağız boşluğundan bir anda hava bırakılmak suretiyle oluşturulur. Konuşma üreten ve kodlayan sistemler bu karakteristik modelleri göz önünde bulundurarak hazırlanırlar.

### 1.2.3 İşitme

İşitme sistemi bizim işitme duyumuzdan sorumludur. Bu sistem akustik ses dalgalarını alır ve onları beyin tarafından yorumlanan sinir kodlarına dönüştürür. İnsanın dış dünya ile ilişki kurmasını sağlayan beş duyu organından biri de kulaktır. Kulak ses titreşimlerini sezer ve sinir uyarılarına çevirerek beyne yollar. Gerçek ses algılaması beyinde olur. Kulak ve beyin arasındaki ilişki, seslerin algılanmasını, işlemlerden geçmesini ve seslerin taklit edilebilecek şekilde öğrenilmesini sağlar. İşitme sistemi dört ana bölümden oluşur: *dış kulak*, *orta kulak*, *iç kulak* ve *beyine giden sinir yolları*.

İşitmenin gerçekleşebilmesi için;

- a) Sesin olması,
- b) Sesin kulağa ulaşması,
- c) O sesin insan kulağının alabileceği frekans ve şiddet sınırları içinde olması,
- d) Sesin kulaktaki dış, orta ve iç bölümleri aşması,
- e) Sesin işitme merkezine ulaşması ve merkezce algılanması, gerekmektedir.

## 1.3 SES TANIMA

### 1.3.1 Genel Bilgiler

Ses tanıma bir akustik sinyali bir karakter setine dönüştüren bir işlemdir. Son yıllarda geniş bir kullanım alanı bulmuştur. Tanıma işlemi bazı farklı teknolojiler ve uygulamalar gerektirir. Ses tanıma alanında 1950'li yıllardan günümüze kadar yapılan araştırmalarda en yüksek doğruluk yüzdesi her zaman için gerçekleştirilmek istenilen hedeflerin başında gelmiştir. Bu amaca ulaşmak için gerçekleştirilen ses tanıma sistemleri sinyal işleme, akustik, model tanıma, haberleşme ve bilgi teorisi, dilbilim, psikoloji ve bilgisayar bilimi gibi bir çok disiplini kendi doğası içinde barındırmaktadır. Ses tanıma alanında yapılan çalışmalarda bu disiplinler son derece iyi bir şekilde kavranmalı, bir başka deyişle ses tanıma alanına uygulanmalıdır [5]. Ses tanıma bu disiplinlerle bağlantılıdır, ama araştırmacılar kendi alanları ile doğru eşleşme yapma yöntemini uyguluyorlar. Bu disiplinler ve ses tanımadaki uygulamalarının ne şekilde olduğu aşağıda verilmiştir.

Sinyal İşleme: Bir ses sinyalinden gerekli bilgiyi en etkili ve en uygun şekilde çıkarma işlemidir.

Fizik: İnsan sesinin akustik ve fiziksel mekanizması ile ilgilidir (konuşma sistemi ve duyma mekanizması gibi).

Model Tanıma: Veriyi prototip olan modellerde gruplamak ve bir çift modelin özelliklerini temel alarak eşlemek için kullanılır.

Haberleşme ve Bilgi Teorisi: Bazı ses modellerini belirleyen yöntemlerdir.

Dilbilim: Dil içinde kodlanan ifadelerin sessel, morfolojik, sentaktik, semantik ve pragmatik düzeylerdeki yapısını ve işlevlerini ve bu düzeyler arasındaki ilişkileri inceler.

Fizyoloji: Ses tanıma çalışmalarında insan sinir sisteminde sesi anlama ve sesi üretmedeki mekanizmayı anlamak için kullanılır.

Bilgisayar Bilimi: Yazılım ve donanımda çeşitli yöntemlerin uygulanması için etkili algoritmaların geliştirilmesidir.

Psikoloji: Basit görevlerde, insanlar tarafından kullanılan teknolojiyi etkin hale getiren faktörleri anlama.

Ses tanıma işlemi için kullanılan bazı teknikler vardır. Peki bu tekniklerin geçerlilik sınırları nelerdir? Bu soru karmaşık bir sorudur. Çünkü bu durum bir sistemin değerlendirilmesinde kullanılan bazı koşullara dayanır. Bu koşullar aşağıdaki gibi sıralanabilir:

Konuşmacı Bağımlı/Bağımsız Sistemler (Speaker Dependent/Independent Systems):

Konuşmacı bağımlı sistemler için belirli bir kullanıcı tarafından daha önce tanımlanmış bir kelime ya da cümle ele alınır. Bu tür sistemler olanakların ve zamanın sistemin bir konuşmacıya bağımlı olarak eğitilmesi için yeterli olduğu masaüstü uygulamaları için kullanılırlar.

Konuşmacı bağımsız sistemler konuşmacılardan alınan çok miktarda ses örnekleriyle bir ön öğrenmeden geçirilir. Kullanıcı böyle bir sistemi hemen kullanmaya başlayabilir. Böylece işlemler çok daha kolaylaşır. Bu sistemin dezavantajı, bir dil için bütün konuşmacı varyasyonlarını modellemenin imkansız olmasıdır. Konuşmacı bağımsız sistemler özel konuşmacı eğitimine gerek duymazlar. Bu durum bir avantaj olarak görülse de daha düşük kalitede performansa sahiptirler.

Konuşmacı bağımlı sistemlerin performansı konuşmacı bağımsız sistemlere göre çok daha yüksektir. Ancak bir sistemin kullanım alanını arttırmak için amaç

konuşmacıdan bağımsız bir sistem olmasını sağlamaktır. Fakat, tahmin edilebileceği gibi, bunu başarmak konuşmacıya bağımlı bir sistem geliştirmekten daha zordur.

*Ayrışık Kelime Tanıma (Isolated Word Recognition):*

Ayrışık kelime, tanıma kısa aralıklarla seslendirilen kelimelerin tanınmasıdır. Doğal konuşmada, bütün kelimeler arasında duraklama olmaz. Duraklamalar bazı kelime grupları arasında vardır. Fakat, bu tür sistemlerde kelimeler arasında duraklamalar söz konusudur. Bu avantaj sayesinde sistem kelimeler arası sınırları bulmakla uğraşmaz. Kelimeler analiz edilir ve daha önceden hazırlanmış modellerle karşılaştırılırlar.

*Sürekli Ses Tanıma (Continuous Speech Recognition):*

Sürekli ses tanımada ara verilmeden seslendirilen kelimelerin tanınması amaçlanır. Bu sistemler bir tanıma işleminin asıl hedefidir. Bir kelimenin ne zaman ya da nasıl sonlandırıldığı sorun değildir. Kelimeler gerçek zamanlı olarak tanınırlar ve sonrasında bir aksiyona önderlik ederler. Konuşmadaki deęişkenlikler, telaffuzlar ve gerçek zamanlı işlem sorunu bu tanıma modu için başlıca sorunlardır.

*Fonem Tabanlı Ses Tanıma (Phoneme Based Speech Recognition):*

Fonların fonemlere dönüştürülmesi ve tanıma işleminin buna dayandırılmasına fonem tabanlı ses tanıma denir. Fon dildeki anlam ayırıcı en küçük olan öğedir. Yani; konuşma organlarının anlamlı kelimeler ve cümleler oluşturmak için düzenli olarak çalışması sonucu çıkan birimlerdir. Fonların simgesel olarak ifade edilmesi sonucu oluşan simgeler fonem (phoneme) olarak adlandırılır. Ters bir tanımlama yapılacak olursa fonemlerin seslendirilişyle fonlar oluşur. Tanıma sırasında seçilen birimler ikili fonem (diphone), üçlü fonem (triphone), hece veya kelimenin tamamı olabilir [24]. Türkçe'de her bir fon alfabetik bir simge ile ifade edilebildiğinden dolayı alfabetik simgeler aynı zamanda fonem olarak da adlandırılabilir. Fonem tabanlı bir sistemde sözcük sınırı yoktur. Bu tür sistemlerin

karşı karşıya bulunduğu önemli bir konu da fonemlerin ardarda sıralanması sırasında ortaya çıkan geçişlerdir. Bu geçişler sırasında insanın gırtlak yapısının bir sonucu olarak bir fonemin seslendirilmesi, bir sonraki fonem başladığında hala bitmemiş olur. Bu sebeple kesin bir sınır konulamaz. Bu durum sesli ifadenin kesimlenmesinde bir dezavantaj oluşturmaktadır. Yani sesli ifadenin fonemlere ayrılması sırasında fonem sınırlarının belirlenmesinde zorluklar görülür.

### Kelime Tabanlı Ses Tanıma (Word Based Speech Recognition):

Tanıma için öngörülen en küçük birimin kelime kabul edildiği sistemlerdir. Doğruluk yüzdesi fonem tabanlı ses tanıma sistemlerine göre daha yüksektir. Çünkü fonemler arası geçişlerin olumsuz etkisi burada gözlenmez. Bu tür sistemlerde kelime sayısı sınırlı tutulmak zorundadır.

### **1.3.2 Ses Tanıma Süreci**

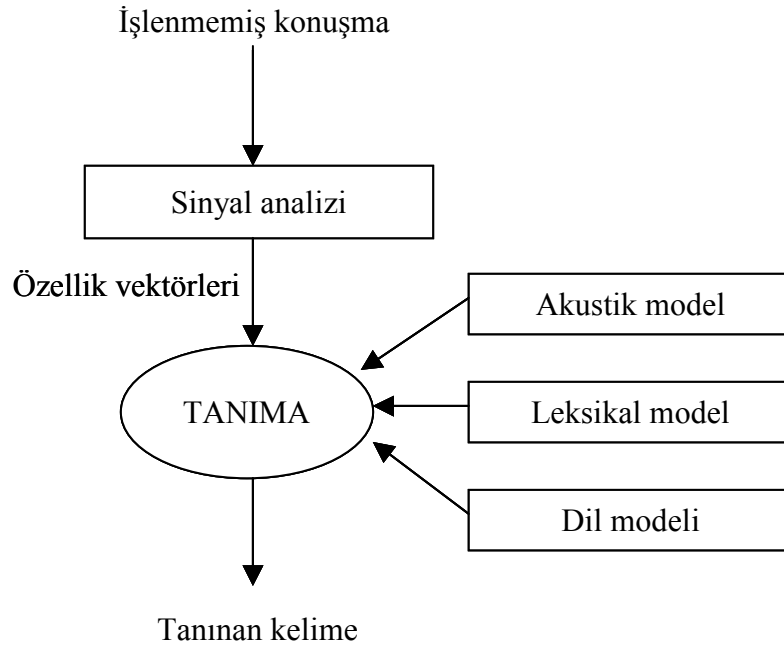
Ses tanıma işlemi sözlükteki bir kelimenin söylenişine karşılık gelen özellik vektörleri dizisinin haritalanması olarak düşünülebilir. Konuşmadaki özellik parametreleri belirlendikten sonra bu parametreler için istatistiksel bir model bulunur. Buna akustik modelleme denir [2]. Konuşmacıların konuşma sinyallerinin akustik modelleri veritabanını oluşturur. Bütün bu veri tabanında arama ve verilen bir sinyal için sözlükteki en iyi eşleşmeyi seçme tanıma işlemi gerçekleştirir.

Bir tanıma sürecinin çözümünde kullanılan denklem şudur:

$$W = \arg \max P(W | A) = \frac{P(W)P(A | W)}{P(A)}$$

Yani, akustik sinyal A'nın verdiği en uygun W kelimesini bulmalıyız. P(W) kelimenin olasılığını belirler ve dil ile ilgilidir. Yani, dil modelleri P(W)'nin bulunması için geliştirilmiştir. P(W|A), W kelimesinin verdiği akustik sinyalin durum olasılığıdır. P(A) kaydedilen sinyalin olasılığıdır, kaydedildikten sonra tüm

kayıtlar için benzer olur. Sonraki problem,  $P(W|A)$  ile  $P(W)$  ürünlerinin maksimum değerini bulmaktır. Şekil 1.4’de tipik bir ses tanıma sistemi yapısı şematik olarak verilmiştir.



Şekil 1.4 Tipik bir ses tanıma sistemi yapısı

Ses tanıma sürecinde konuşmacının söyledikleri sisteme verildiğinde öncelikle kaydedilir. Böylelikle, elimizde işlenmemiş konuşma verisi olur. Sonraki aşama ses sinyalinin sayısallaştırılmasıdır. Sayısallaştırılan ses sinyali üzerinde sinyal işleme teknikleri (sinyal analizi) uygulanılarak sesin ayırt edici özellikleri ortaya çıkarılır. Sonra sinyal işleme sonucunda oluşan özellik vektörleri bazı teknikler ile modellenir. Modellemenin sonucunda kelimenin söylenişine karşılık gelen özellik vektörleri dizisinden bir kelime modeli elde edilir ve bu model veritabanında bulunan şablonlarla karşılaştırılır. Bütün bu veritabanında arama ve verilen sinyal için en iyi eşleşmeyi seçme ile ses tanıma işlemi gerçekleştirilir. Bütün bu aşamalar sonraki bölümlerde ayrıntılı olarak açıklanacak.



### 1.3.2.1 İşlenmemiş Konuşma

Ses girişini alma işlemi farklı teknolojiler ve uygulamalar gerektirir. Konuşma girdi cihazı genellikle bir mikrofon veya bir telefondur. Konuşma çoğunlukla yüksek bir frekansta örneklenir, (örneğin bir mikrofondaki 16 KHz veya telefondaki 8 KHz olarak). Bu, bize zaman üzerindeki bir dizi genlik değerini verir.

### 1.3.2.2 Sesin Sayısal Kodlanması

Ses analogdur ve işlenebilmesi için öncelikle analog formdan sayısal forma dönüştürülmesi gerekir. Bunu yerine getirmek için geliştirilmiş olan farklı kodlama yöntemleri vardır. Kodlama yöntemlerinin çoğu kodlayıcıya girilen ve kod çözücünde görülen ses arasındaki farkedilir gecikmeden kaçınmak için yüklenmesi gereken, sayısal hatlar üzerindeki gerçek zamanlı ses iletimi için geliştirilmiştir. Bu gerekliliğin sayısal kodlamada mesajı saklamak için kullanımı uygulanamaz, bu nedenle bu uygulama için değişken fazlalığını kullanmak gibi daha büyük bir özgürlük vardır [6][7][8][9].

Rasgele seçilmiş bir ses sinyali üretmek için bant genişliği terimlerinden gerekli bilgi oranının (bits/s) ve sinyalin bant genişliğinde belirlenmesi gereken doğruluk derecesinin hesaplanması gerekir. Tipik telefon kalitesi için bant genişliği 3 KHz ve ses-sinyal oranı 40 dB olabilir. Bu durumda bilgi oranı 40.000 bit/s'dir. Yüksek bir aslına uygunluk için tek sesli ses üreten sistemde bant genişliği beş defa daha iyi olabilir ve ses sinyalinin en tepe noktasının altında 60-70 dB ses sinyal oranında olabilir. Bu durumda 30.000 bits/s civarındaki bir oran herhangi bir sistem tarafından üretilmiş olabilecek olası ayrık sinyallerden birini açıkça belirlemeyi gerektirir.

Bu ifadeler karşın, bilinen şudur ki insandaki kavramsal süreçler dakikadaki onlarca bit fazlalığında bir bilgi oranını hesaba katmaz. Böylece iletilen bir bilgi oranının uygulanmasında 1,000 ve 10,000 arasında bilgiyi kullanılır. Bu büyük oran gösteriyor ki bir ses kanalındaki tüm bilgi kapasitesi ses iletimi için gerekli

olmayabilir. Maalesef, bir haberleşme mühendisi için, insan dinleyici, kavramsal süreçler için uygun olan onlarca bit dikkate alınarak seçilen sinyal durumlarına karar vermede çok seçici olmalıdır. Genelde dinleyici mesaja odaklanır, yüksek derecede gereğinden fazlalığı olması normaldir. Bununla birlikte dinleyici, konuşmacının ses kalitesine özellikle dikkat etmelidir.

Burada ses kodlamada ağırlıklı olarak durulabilecek ses iletiminin iki özelliği vardır: Birincisi insan işitsel sisteminin sınırlı kapasitesidir. İşitsel sınırlar konuşmacının sesin yeniden üretimindeki çeşitli eksikliklerden etkilenmemesini sağlar. Ses kodlama sistemi tasarlanırken bu durum aynı zamanda avantajlı da olabilir. Şöyle ki, konuşma mekanizması fizyolojisi meydana gelebilecek sinyal tipleri üzerinde güçlü kısıtlamalar koyar ve bu gerçek bir ses hatından alınan insan ses üretimindeki bazı durumları modellemede kullanılabilir [7][9].

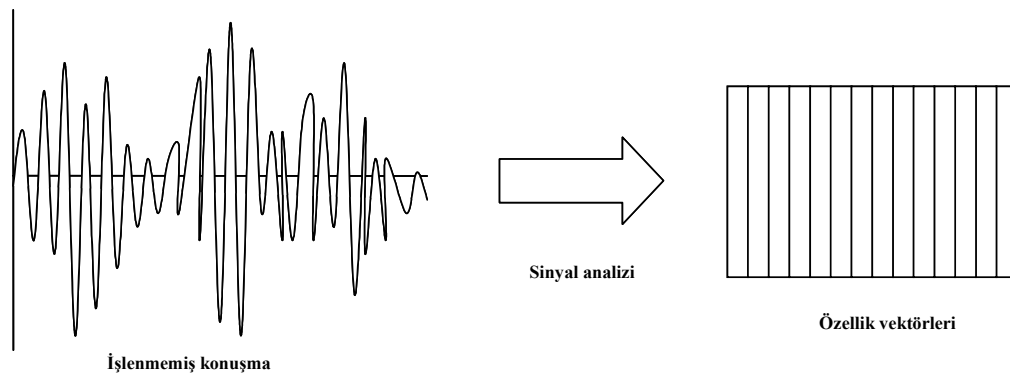
Kodlama metotları üç genel sınıfa bölünebilir:

1. Basit dalga formu kodlayıcıları, veri oranı 16 kbits/s üzerinde işlem yapar:
  - a. Darbe Kodu Modülasyonu (Pulse Code Modulation-PCM)
  - b. Uyarlanabilir Diferansiyel Darbe Kodu Modülasyonu (Adaptive Differential Pulse Code Modulation-ADPCM)
  - c. Delta Modülasyonu (Delta Modulation-DM)
2. Analiz/Sentez sistemleri şunlardır:
  - a. Kanal Ses Kodlayıcılar (Channel Vocoders)
  - b. Sinüsoid Kodlayıcılar (Sinusoidal Coders)
  - c. LPC Ses Kodlayıcılar (LPC Vocoders)
  - d. Biçimlendirici Ses Kodlayıcılar (Formant Vocoders)
  - e. Etkin Parametre Kodlama (Efficient Parameter Coding)
  - f. Parçasal/Fonetik Yapıdaki Ses Kodlayıcılar (Vocoders based on segmental/phonetic structure)

3. Orta düzey sistemler, yukarıdaki iki kategorinin bazı özelliklerine sahiptir ve 4-32 kbits/s bölgesindeki geniş bir alanı kapsar:
  - a. Alt-Bant Kodlama (Sub-band Coding)
  - b. Kalanı basit kodlama ile doğrusal tahmin (Linear prediction with simple coding of the residual)
  - c. Uyarlanabilir Kestirimci Kodlama (Adaptive predictive coding)
  - d. Çoklu-sinyal LPC (Multipulse LPC)
  - e. Kod-uyarımli Lineer Tahmin (Code-excited linear prediction-CELP)

### 1.3.2.3 Sinyal İşleme Teknikleri (Sinyal Analizi)

Ses sinyaline, tanımda gerekli olan faydalı özelliklerini ortaya çıkarmak için sinyal işleme teknikleri uygulanır. Amaç sinyaldeki fonetik bilgisini kötü telefon hatlarının içerdiği gürültü, konuşmacı farklılıkları veya konuşmacının duygusal durumu gibi faktörlerin tesiri olmayacak şekilde saklamaktır. Ses tanıma sistemleri genelde ses sinyallerinin analizinden faydalanmaktadır. Ses sinyalinin analizi sonucunda bu frekans ve genlik bilgilerini içeren özellik vektörleri oluşur. Bir özellik vektörü genelde her bir kısa zaman aralığındaki bir ses sinyali penceresinden (20~30 milisaniye) hesaplanır. Söylenen kelime bu özellik vektörlerinin bir dizisi olarak gösterilir. Sonraki aşamada bu özellik vektörleri tanıyıcıya giriş olarak verilir. Ses sinyallerinin özellik vektörlerine dönüştürülmesi işlemi şematik olarak Şekil 1.5'de verilmiştir.



Şekil 1.5 Ses sinyallerinin özellik vektörlerine dönüştürülmesi

Bazı sinyal işleme teknikleri herhangi bir önemli veri kaybı olmadan ayırıcı özellikleri bulduğu ve sıkışmış veriyi açabildiği için daha kullanışlıdır [10]. En popülerleri arasında:

- Hızlı Fourier Dönüşümü (Fast Fourier Transform-FFT) gözle görülebilecek biçimde yorumlanabilen zamandaki ayırık frekansları verir. Frekanslar düşük düzeyde lineer olan ama yüksek düzeyde logaritmik olan *Mel* scale kullanılarak ve insan kulağının fiziksel karakteristiklerine uygun olarak sınıflandırılırlar. Fourier dönüşümü ile genlik-zaman boyutu, frekans-genlik boyutuna dönüştürülür [10].
- Algısal Lineer Kestirimci Kodlama (Perceptual Linear Predictive-PLP) aynı zamanda fizyolojik olarak harekete geçirilir, ama katsayılar açıkça yorumlanamaz. Algısal Lineer Kestirimci Kodlama, Ayırık Fourier Dönüşümü (Discrete Fourier Transform-DFT) ve Lineer Kestirimci Kodlama (Lineer Predictive Coding-LPC) tekniklerinin birleştirilmesi ile sesin parametrelerinin hesaplanmasıdır. Bu yöntem insan kulağının duyma sistemini Lineer Kestirimci Kodlama yönteminden daha iyi modellemeye yöneliktir.
- Lineer Kestirimci Kodlama (Linear Predictive Coding-LPC) Bu kodlamada temel fikir, bir ses örneğinin kendisinden önceki ses örneklerinin doğrusal kombinasyonu kullanılarak tahmin edilmesidir. Gerçek ses örnekleri ile tahmin edilen örnekler arasındaki hata minimumlaştırılarak öngörü katsayılarından oluşan parametre değerleri elde edilir [11]. Kodlama yöntemleri içinde belki de en yaygın olanı ve en çok kullanılanı bu kodlama olmuştur. Gırtlak yapısını örnek alan diğer modellerin aksine bir filtre grubu değil de tek girişli ve tek çıkışlı bir filtre modeli benimsemiştir.

- Kepstrum (Cepstral Analysis) güçlü sinyal spektrumu logaritmasında Fourier dönüşümün tersi bir işlem yapar.

#### 1.3.2.4 Ses Sinyalinin Modellenmesi

Ses tanımda başlıca iki yaklaşım vardır: bilgi-tabanlı yaklaşım (*knowledge-based approach*) ve veri-tabanlı yaklaşım (*data-based approach*). Bilgi-tabanlı yaklaşımda, amaç insanın ses bilgisini şu kesin kurallarla anlatmaktır: Akustik-fonetik kurallar, leksikondaki kelimeleri tanımlayan kurallar, dilin sentaks yapısını tanımlayan kurallar ve diğerleri [12].

Veri-tabanlı istatistiksel yaklaşım son zamanlarda önemli başarı kazanmıştır. Bu yaklaşımlarda, ses verisinden bilgi çıkarabilen algoritmalarla ses sinyali modellenir. Buradaki iki yaklaşım Saklı Markov Modelleri (SMM) ve Yapay Sinir Ağları (YSA)'dır. Bilgi-tabanlı yaklaşımda sezgisel kurallar uzmanlar tarafından elde edilmektedir. SMM ve YSA yaklaşımlarında ise, öğrenme, ses verisi algoritmaya verilerek ve modellerin gelişerek veriyi elde etmesi ile başarılmaktadır. Genelde modele ne kadar fazla veri sunulursa, tanıyıcıdan o kadar iyi sonuç elde edilmektedir.

Son yıllarda çoğu ses tanıma araştırmalarında istatistiksel yaklaşımlarla ilgilenilmektedir. Bunun sonucu olarak performansta dikkate değer gelişmeler gözlenmektedir. SMM modelleri geniş bir kullanım alanı olan istatistiksel bir yöntemdir.

SMM ve YSA yaklaşımlarının teorisinden ve ses tanımadaki kullanımının nasıl gerçekleştiğinden sonraki bölümlerde ayrıntılı olarak bahsedilecektir.

#### 1.3.3 Ses Tanıma Kullanıcıları ve Kullanım Alanları

Ses tanıma yazılımı konuşulan kelimeleri yazılı metne çevirmekte bilgisayar kullanıcılarına yardımcı olmaktadır. Yazılım birçok kullanıcı tarafından farklı ihtiyaç ve amaçlarla kullanılır. Üç tip kullanıcı vardır ve bunlar;

1) *Ellerini kullanma problemi olan kullanıcılar*: Yazı yazmak için ellerini kullanmakta zorlanan insanlardır, ama doğru bir şekilde konuşabilirler. Bunlar bazı problemleri olan kullanıcılarıdır: Sinirsel problemi olan veya bir kaza geçirmiş zayıf durumda olanlar veya iyi yazamayanlar vb.

2) *Profesyoneller*: Yazı yazmak için zamanı olmayanlar veya iyi yazamayanlardır. Bu grup bazı çalışan insanları, tıbbi ve kanuni alanlarda çalışanları veya yazılmış yayınları takip etme ihtiyacı olanları içerir. Yazılım, zaman ve para kazandırır ve kullanıcılara rapor alabilmelerini sağlar. Tipik kullanıcıları doktorlar, avukatlar, psikologlar, satış sorumluları ve diğerleridir.

3) *Öğrenme zorluğu olan kullanıcılar*: Doğru olarak yazmalarını önleyen öğrenme yetersizliği olan insanlardır [2][13].

#### **1.3.4 Ses Tanımının Avantajları ve Sınırlamaları**

Ses tanıma metodunun avantajları ve dezavantajları bulunur. Hala çözümü olmayan problemler teknikte bazı sınırlamalara neden olur. Teknik, kontrollü ve sessiz ortamda iyi çalışır. Yüksek ses seviyeleri tekniğin faydalı olan avantajlarını bulmayı zorlaştırır.

Eski basit ses tanıyıcılarda içerik düşüncesi yoktur. Bir çıktıdaki sonuçlar sadece bir kelime topluluğudur ve kullanıcı için anlamı vardır. Aynı zamanda tanıyıcı büyük bir sözlük kullanıyorsa sistem tanımada zorlanır. Çünkü sözlüğü genişletmek sistemi daha karışık yapar.

Tanıyıcı daha fazla kullanılabilir işlem gücü gerektirdiği için uygulamanın kalanı bundan olumsuz etkilenir. Bir tanıyıcı için diğer bir problem, araştırmalar en normal sesin kısa kelimeler içerdiğini gösterirken uzun kelimeleri ayırmanın daha kolay olmasıdır.

Ses tanıma konusunda halihazırda uygulamadaki sıkıntı, her insanın konuşma tarzının ve ağzının farklı olması, yani bir standardın olmamasıdır. Tanınabilir bir lehçeye rağmen, bir ses tanıma aracı herkes için çalışmayı garanti etmez. Bazı insanlar, diğerleri en sessiz ortamlarda bile tanıma yapamıyorken, en gürültülü ortamlarda bile bu aracı kullanabilirler.

Bazı konuşma tanıma araçları frekans ölçüsünün tamamını kullanamayan donanıma sahiptir. Bu çok kapsamlı olmayan donanım özellikle yüksek frekanslı girişi kapsamaz ve çıktı sonuçları güvenilir değildir. Erkek sesinin kadın sesine tercih edilme nedeni budur.

Sekizinci ve son sınırlama tanıyıcının kapasite seviyesi ile ilgilidir. Mükemmel şartları olan bir iyi-fonksiyonlu tanıyıcı bile sürekli olarak çeşitli hatalar yapar. Tanıyıcı, hatalı kelimeler duyabilir, söylenen kelimeleri atlayabilir ve kelimeleri yanlış anlayabilir. Kelimenin doğruluk oranı %95'dir. Ama unutmamalıyız bir doğruluk oranının %95 olması istatistiksel bakış açısıyla 8-10 kelimeli cümlelerin yarı zamandan daha fazlasında en az bir hata ile tanınmasıdır. Ses tanıyıcılar için tüm bu kısıtlamaları maddeler halinde özetleyecek olursak:

1. Sessiz, kontrollü ortamda en iyi çalışır.
2. Eski basit ses tanıyıcılar dilsizdir, kelimeleri duygusuz söylerler.
3. Büyük sözlük kullanıldığında karmaşıklık artar.
4. Yüksek miktarda işlem gücü ister.
5. Kısa kelimeleri ayırt etmenin zorlukları vardır.
6. Herkes için çalışma garantisi yoktur.
7. Bazı tanıyıcıların diğerlerine göre daha fazla zahmetli işleme tarzı vardır.
8. En iyi durumda kelime doğruluğu %95'e ulaşır.

Bütün bunlardan başka, ses tanıma bir iş yeri için çok büyük potansiyellere sahiptir ve yetersizlikleriyle insanlar için eğitimsel uyum süreci vardır. Çoğu durumda bir ses tanıma aracı çalışmazsa, bu kullanıcının davranışına ve bilgisine

bağlıdır. Bu gibi yetersiz bilgiyi önlemek için iki ölçüm vardır. Kullanıcılar sistemi ses ile çalıştırmak için iyi hazırlanmış olmalıdır ve güncel teknolojileri bilmelidirler.

### 1.3.5 Ses Tanıma

#### 1.3.5.1 Ses İşlemede Amaç

Yapay zekanın alanına giren ses uygulamaları içinde ses tanıma ile beraber aynı zamanda konuşmacı tanımlama ve ses üretme de vardır. Bu üç başlık konuşma işleme (speech processing) veya ses işleme başlığı altında toplanır. Ses tanıma bu üç konudan en zor olanıdır, zira konuşmacı tanıma çok fazla sayıda tanınacak insan olmadığı takdirde, insan sesinin özelliklerinden dolayı kısmen çeşitli kolaylıklar içermektedir. Hatta bu özelliklerden dolayı ses tanıma daha zorlaşmaktadır. Burada anlatılmak istenen ses tanıma uygulamalarının konuşmacıdan bağımsız olması için ayrıca çaba harcanmaktadır. Ses üretme (speech synthesis) ise kısmen de olsa üzerinde belli başlı bazı algoritmalar geliştirilmiş ve oturmuş bir konudur. Özellikle İngilizce için bu konuda çok iyi uygulamalar vardır. Hatta son günlerde Türkçe için de bir iki uygulama çıkmıştır, ancak bunlar çok başarılı değiller [3]. Ses tanıma teknolojisi ile Türkiye'de ticari manada ciddi olarak ilgilenen pek fazla firma yoktur. Bu konuda daha çok çeşitli üniversitelerde doktora tezleri şeklinde çalışmalar yapılmaktadır. Ancak bu çalışmalardan şanslı olanların dışardan mali destek gördükleri de olmuştur. Yurt dışında ise, özellikle de Amerika'da bu konuda çalışan pek çok firma var. Amerika'da 1994 den beri 1250 civarında kuruluş bu konu ile ilgili çalışmalarda bulunmuştur, bunların 30'a yakını üniversite diğerleri ise ticari ve askeri kuruluşlardır. Bu konu ile ilgili çalışmalar yapan kuruluşların içinde US Army ve US Navy de yer almıştır. Bunların haricinde telefon şirketlerinde meşrubat şirketlerine pek çok kuruluş bu çalışmalara katılmıştır [13][14].

Ses tanıma ve doğal dil işleme, Microsoft'un hesaplarına göre Dos'tan Windows'a geçişten sonraki en büyük atılım olacak. Onlara göre bu teknoloji normalde cansızmış gibi görünen bir objeyle olan (bilgisayar) ilişkinizi köklü biçimde değiştirecek. Ancak Microsoft'a göre bu teknoloji birden ortaya çıkabilecek



bir teknoloji deęil. Sonraki on yıl içinde ortak çalışmalar sonucunda yavaş-yavaş gelişip yerine oturacak.

Bu teknolojiyi 4 başlık altında incelemek mümkündür. Bunlar;

1. Telefonda ses (konuşma) tanıma: Komutları anlayan bilgisayarlar aracılığı ile, bu konuda ciddi mesafeler alınmıştır. Telefonla servis veren veya verebilecek olan şirketler için bu konu büyük önem arz etmektedir.

2. Dikte ettirme: Sürekli konuşmaya kısıtlı olarak izin veren, mevcut yazılımlar vardır. Örnek olarak "Microsoft Dictation" ve "Dragon Dictate" verilebilir. Bu programların doğruluk oranları %90-95'ler civarındadır. Ancak yeterli değildir. Nitekim bu hata oranı 3000 kelimelik bir makalede pek çok boşluk kalmasına sebep olmaktadır. Bu yüzden konu ile ilgili çalışmalar devam etmektedir.

3. Konuyu anlayan tanıyıcılar: Bu alandaki çalışmalar sadece söylenen kelimeyi anlamayı deęil ne demek istendiğini yani söylenen cümlenin anlamını çıkarmayı da hedeflemektedir. Bu hedefe ulaşmak öncelikle uzmanlaşmış uygulamalar yapmayı ve sınırları belirli alanlar içinde kalmayı, bunu başardıktan sonra genel kullanıma geçmeyi gerektirmektedir.

4. Doğal dil anlama: Bilgisayarlar, sürekli konuşmayı ve diyalogları anlayabildiğinde, bu teknolojiye büyük bir devrim olacağı beklenmektedir. Henüz bu teknolojinin gelişmesi için en az 15 yıl kadar bir süreye ihtiyaç olduğu belirtilmektedir.. Ama bu teknoloji yayıldığında hayatımızda büyük deęişikliklere neden olacaktır. Bilgisayarınıza yapmasını istediğiniz şeyi normal bir cümle şeklinde söylediğinizde istediğiniz anında yerine getirilecektir. Microsoft ve IBM beraber bu hedefe ulaşmak için çalışıyorlar. Özellikle Microsoft, işletim sistemine bu teknolojiyi yerleştirmenin yollarını arıyor [13].

Ayrıca ses tanıma problemine getirilen farklı çözüm tarzları vardır. Bunlar tanınması gereken konuşmanın kesikli mi yoksa sürekli olduğundan etkilenirler.

Yani iki konuşmayı da aynı teknikle tanımak zordur. Kesikli bir konuşmanın tanınması daha kolaydır ve kelime-kelime yapılması gayet uygundur. Sürekli bir konuşmanın ise kelime-kelime tanınması daha zordur çünkü kelimelerin nerede başlayıp nerede bittiği bilinmemektedir. Dolayısıyla sürekli tanıma genelde fonem bazında yapılmaktadır. Fonem anlam içeren en küçük ses demektir. Yani fonem bir heceden daha kısa bir sestir.

Normal bir hecede başlangıç-orta-bitiş olmak üzere genelde üç fonem bulunur. Ancak fonemleri de birbirinde kesin hatlarla ayırmak pek mümkün değildir. Bu nedenle fonemleri tanıyacak ve temsil edecek çeşitli sistemler geliştirilmiştir. Bunların başında Hidden Markov Modeli (HMM) gelmektedir. Ses tanıma problemi kişiye bağımlı, kişiden bağımsız ya da kişiye uyum sağlayan tarzlarda çözülebilir. Ses tanıma probleminde önemli olan diğer bir nokta da tanınacak kelimelerin (kelime haznesi) sayısıdır [5].

Küçük kelime haznesi -	10-100 kelime
Orta kelime haznesi -	100-1000 kelime
Geniş kelime haznesi -	1000-10000 kelime
Çok geniş kelime haznesi -	10000 ve daha fazla kelime

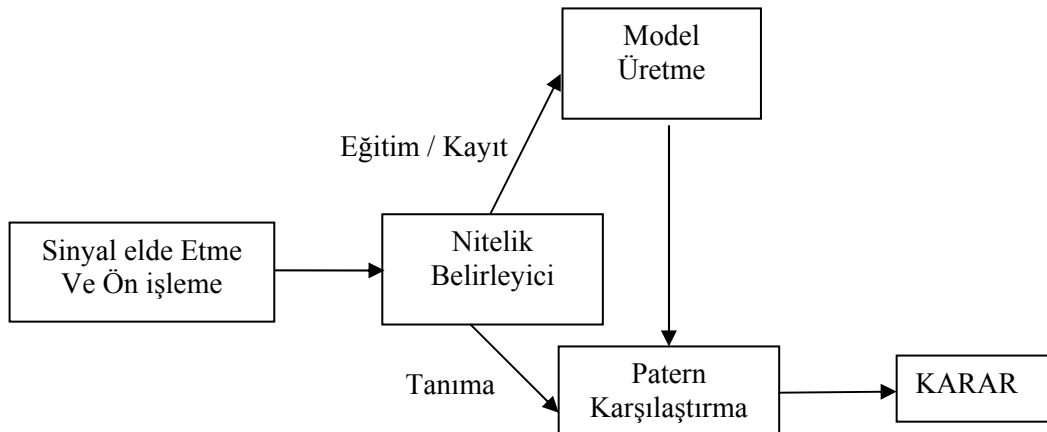
### 1.3.5.2 Ses İşleme

Ses işlemede, konuşmacı tanımadaki problem derken herkes için ayırt edici akustik ses sinyalinin tanımlanmasından bahsedilmektedir. Konuşmacı tanıma kabiliyetinin birçok alanda yararlı tarafları vardır. Çünkü asla diğer anahtarlarınız ya da şifreleriniz gibi bir yerlerde unutma ve çalınma tehlikesi söz konusu değildir. Bireylerin seslerini konuşmacı tanımada kullanmak üzere konuşmanın belirli kısımlarını örnekleyerek temel olarak başlangıç yapılır.

### 1.3.5.3 Konuşmacı Tanıma

Konuşmacı tanımadaki temel problemler onaylama ve tanımlama olarak iki kısımda ele alınabilir. Konuşmacı onaylama esnasında önceden belirlenmiş bir konuşma kalıbının konuşmacı tarafından söylenmesi gerekir. Sistem tanımlanan metnin akustik özelliklerinin doğru olup olmadığına bakarak karar verir. Diğer taraftan konuşmacı tanımlamada sistem bir seri kelime grubu içerisinde akustik özellikleri karşılaştırarak sonuca ulaşır. Ama esas olarak konuşmacı onaylama ve konuşmacı tanıma işlemlerinde benzer sinyal işleme yöntemleri ve konuşmacı modelleri kullanılır.

Kişileri kendi öz nitelikli ses sinyalleri ile ayırabilmek için her şeyden önce herkes için kendi sesinin akustik öz niteliklerinin tanımlanması gerekir. Burada problem olan konu akustik bir ses sinyali içerisinde kişiye ait karakteristiklerin çıkarılabilmesidir. Bu amaç doğrultusunda bir ses modeli hazırlanarak ‘ öğrenme ‘ olarak adlandırılan işlem sayesinde oluşturulan modelin özellikleri ile konuşmacının özellikleri karşılaştırılarak karakter olarak nitelenebilecek özellikler öğrenme dönemi içerisinde ortaya çıkarılır. Öğrenme gerçekleştikten sonra yeni konuşma örneklerinin özellikleri hesaplanarak tanıma amacı için modelin verileri ile karşılaştırılır. Konuşmacı Tanımda İzlenen Yöntemi gösteren tipik bir model yapısı Şekil 1.6’da gösterilmektedir.



Şekil 1.6 Konuşmacı Tanımda İzlenen Yöntem

#### 1.3.5.4 Konuşmacı tanıma için ses özellikleri

Konuşmacı tanıma için ilk yapılması gereken işlem örnek konuşmacı için ses örneklerinden konuşmacı tanımada kullanılacak parametreleri seçmek ve tanımlamaktır. Her insan için konuşma şeklinden ve fiziksel ses yapısından dolayı bir takım ayrılıklar söz konusudur. Ses yolunun ve ses tellerinin titreşimleri her ne kadar zamanla değişse de fiziksel olarak ses yolu ve ses tellerinin yapısı limitlidir.

Fiziksel olarak ses yolu yapısının ve ses tellerinin yapısının herkes için farklı olduğu ve çok genel olarak limitli olduğu kabulünü yaparak ses tanıma için ve öz nitelikli ses oluşturma için oldukça faydalı bir yöntemle işe başlamak uygun olacaktır

Ses özelliklerini konuşmacı tanıma için seçerken kişileri ayırt edebilecek kadar değişken ancak tanıma için kullanılacak kadar sabit özelliklerin belirlenmesi gerekir. Herhangi birine ait bir konuşma, konuşma modu, içerik, sağlık durumu vb. gibi nedenlerle değişken olabilir, ancak bu durumlara ait değişkenler iyi tespit edilmeli ve tanımlanan parametrelerden ayrılmalı. Arzu edilen özelliklerin ölçülmesi kolay olmalı ancak başka konuşmacı ile karışmalara olanak vermemek için yeteri kadar öz nitelikli olmalı.

Yukarıda bahsedilen kabullerden yola çıkarak, konuşmacı tanımada kullanılacak özel ses karakteristiklerinden söz edilebilir. Ses perdesi ses yollarındaki titreşimlerin temel frekansı ile ilgilidir. Ses perdesi tanımada çok değerli bilgiler sağlar, ancak ölçmek ve değişik tonlamalar esnasında belirleyici olarak kullanmak uygun değildir. Ses yolunun bu yankılanan frekansları formant frekanslar olarak bilinir, ayrıca konuşmacı tanımada konuşmacıyı ayırt eden bir özellik olarak kullanılır. Herhangi bir konuşma sinyalindeki formantlara bakarak aynı konuşmacıdan alınan iki farklı konuşmanın karşılaştırılması ile konuşmacının aynı olup olmadığını karar vermek mümkündür ancak direkt olarak ölçüm yapmak zordur.

Eğer ses sinyalinin spektrum analizini yapmak istersek göreceğiz ki konuşmacıya ait ses özellikleri spektrumun yapısında doğal olarak vardır. Verilen bir ses sinyalinin dalga biçimi için spektrumu tahmin edebilecek birçok metot geliştirilebilir. Eğer sinyal spektrumundaki parametrik tahminlere bakacak olursak parametrelerin kendilerinin konuşmacı tanıma için konuyla ilgili özellikler sağladığını göreceğiz. Doğrusal Tahmin Kodlaması (LPC) ses yolu ile ilgili çok güçlü tahmin nitelikli ‘ transfer fonksiyonları ‘ sağlar ki böylelikle elde edilen sabitler konuşmacı tanıma amacı için kullanılabilir [8][15].

## 2. MATERYAL VE METOD

### 2.1 SAKLI MARKOV MODELLERİNİN SES TANIMADA KULLANILMASI

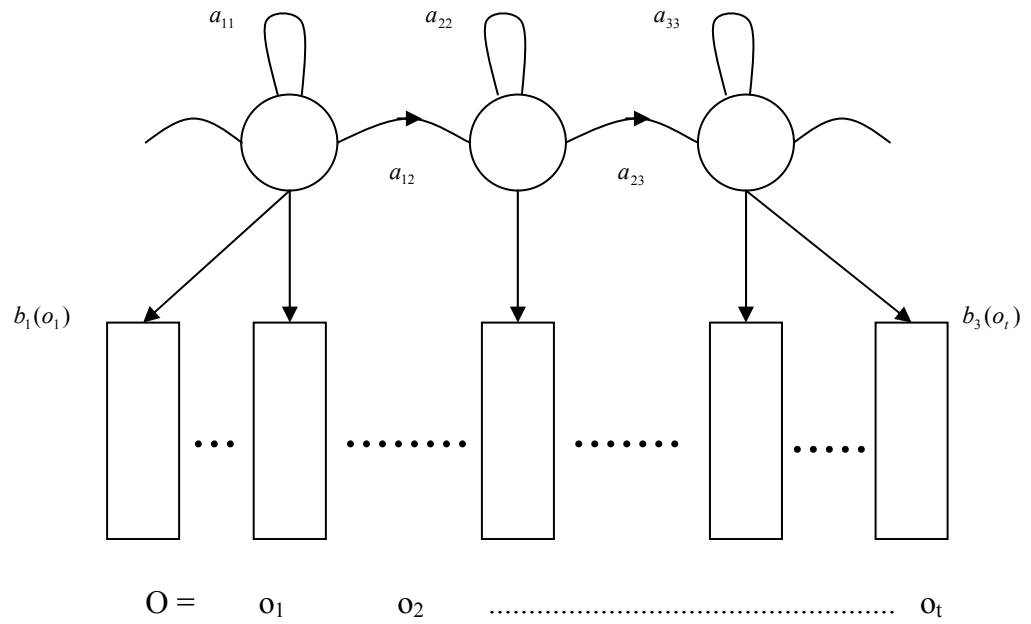
#### 2.1.1 Giriş

Bu yöntemdeki temel fikir, ses sinyalinin parametrik bir rasgele işlem olarak ifade edilebilmesidir. Saklı Markov Modeli (SMM) iki skotastik süreç içerir. Birincisi; Markov süreci zaman ile ilgili değişikliklerde kullanılır ve durumları içeren bir Markov zinciri üretir. Diğer süreç gözlemlenebilir ve özellik parametreleri veya gözlemler denilen rasgele değişkenler içerir [16].

Aslında her söylem ideal durumda bir SMM'e sahip olmalıdır. Bazen bu mümkün olmaz, bu yüzden kelime düzeyinde SMM'lerimiz olmalıdır. Söylenen kelime ile en uygun kelimeyi eşlemek istendiği için, bir SMM bir veritabanındaki tüm kelimeler için en iyisini yapabilmelidir. Ama bu durumda sözlük zaman kısıtlamalarının üstesinden gelmek için yeterince küçük olmalıdır. Bu yüzden geniş kelime tanıma sistemlerinde, konuşma birimi kelimededen fonem azaltmalıdır. Bu özellikle veritabanına yeni bir kelime eklemek zorunda kalındığında faydalıdır. Fonemler genel olarak fonemleri akustik gerçekleştiren konuşma birimlerinde kullanılır ve kolaylıkla değiştirilebilir.

SMM'in yapısı (Şekil 2.1) bir durumlar zincirinden meydana gelir. SMM zinciri üzerindeki her durum kelimenin bir parçasına karşılık gelir. Her durum bir diğerine geçişlerle bağlıdır. Geçişler, geçiş olasılıklarına ( $a_{ij}$ ) bağlı olarak durum değiştirmeye imkan verir. Durumlara iliştilen emisyon olasılıkları ( $b_j$ ) bir özellik vektörünün, referansın belirli bir zaman aralığıyla olan spektral benzerliğini gösterir. Sistem girdisine göre oluşturulan özellik vektörleri dizisine bağlı olarak, model üzerinde birinci durumdan başlayan farklı yollar izlenebilir. Bazı durumların tekrarı veya atlanması kullanıcının konuşma hızındaki değişimlere sistemin adaptasyonunu sağlar. Bir kelimenin tanınabilmesi için referans olarak alınan durumdan itibaren izlenen yolun en son duruma, kabul edilebilir bir olasılığa ulaşması gereklidir [17].

Bir SMM modeli her anda durumu deęişen birimleri olan bir sonlu durum makinesidir. Her  $t$  ayrık zaman anında,  $i$  durumundan  $j$  durumuna geçiř gerçekleřir ve gözlem vektörü  $o_t$ , yoğunluk vektörü  $b_j(o_t)$  ile dıřarı verilir. Bundan bařka  $i$  durumundan  $j$  durumuna geçiř aynı zamanda rasgeledir ve  $a_{ij}$  yoğunluęu ile olur. řekil 2.1'de, üç durumlu soldan saęa SMM atlamasız olarak verilmiřtir.



řekil 2.1 Gözlem Vektörlerinin Ver Biri Bir Durum Tarafından Üretilen Soldan Saęa Üçlü Bir SMM

Ses tanıma probleminde bir SMM'in altında yatan düşünce bir ses sinyalinin en iyi parametrik bir rasgele süreç olarak karakterize edilebilmesidir. Böylece, stokastik süreç parametreleri kesin ve iyi tanımlanmış bir şekilde hesaplanabilir.

SMM teorisi herhangi bir sonlu durum otomasyonuna uygulanırken lineer bir durumlar dizisi kullanılır. Durumlar normalize edilmiş bir zaman eksenindeki zaman noktaları olarak açıklanabilir. Konuşma hızı deęişimlerini hesaplamak için her durumun solda olduęu, genel olarak üç tip olası geçiř vardır: Sonraki duruma git, aynı duruma geri dön veya bir durumla ileri atla.

Uygulamada, sadece gözlem dizisi bilinir ve temelde olan durum dizisi bilinmez. Bu yapıya *Saklı Markov Modeli* denmesinin nedeni budur. Takip eden bölümlerde, kısaca ses tanıma amacı açısından SMM modellerinin teorisi açıklanacaktır.

### 2.1.2 Bir SMM Yapısı

Tam bir SMM modeli belirlenmesi iki model parametresi  $N$  ve  $M$ 'in, gözlem sembollerinin ve üç set olasılık ölçümleri  $A$ ,  $B$ ,  $\pi$ 'in belirlenmesini gerektirir. Bu parametrelerin tanımı şöyledir:

1.  $N$  parametresi, SMM'deki durum sayısıdır. Ayrı durumlar  $\{1, 2, \dots, N\}$  olarak tanımlanır,  $t$  anındaki durum  $q_t$  olarak gösterilir.
2.  $M$  parametresi her durumda bulunan farklı gözlem sembollerinin sayısıdır. Gözlem sembolleri modellenen sistemin fiziksel çıktısı olarak gösterilir. Ayrı gözlem sembolleri  $O = \{o_1, o_2, \dots, o_m\}$  ile gösterilir. SMM modellerinde sadece ayrı gözlem sembolleri için  $M$  parametresi tanımlanmıştır.
3.  $A = \{a_{ij}\}$  matrisi durum geçiş olasılık dağılımıdır. Burada  $a_{ij}$ ,  $i$  durumundan  $j$  durumuna geçiş olasılığıdır.

$$a_{ij} = P(q_{t+1} = j | q_t = i); 1 \leq i, j \leq N \quad (2.1)$$

$i$ 'den  $j$ 'ye tek bir geçişle ulaşılamıyorsa tüm  $i, j$  ler için  $a_{ij} = 0$  olur.

4.  $O = (o_1, o_2, \dots, o_T)$  gözlem sembolleri seti olsun.  $B = \{b_j(o_t)\}$  matrisi gözlem sembol olasılık dağılımıdır.

$$b_j(o_t) = P(o_t | q_t = j); 1 \leq t \leq T \quad (2.2)$$



$i, j = 1, 2, \dots, N$  durumunda sembol dağılımını tanımlar. Ses tanıma probleminde, gözlem sembolleri özellik vektörleridir.

5.  $\pi = \{\pi_i\}$  vektörü başlangıç durum dağılımıdır,

$$\pi_i = P(q_1 = i); \quad 1 \leq i \leq N \quad (2.3)$$

Parametre setini kısaltırsak, bir Saklı Markov Modelinde tam parametre seti göstermek için yoğunluk gösterimini  $\lambda = (A, B, \pi)$  olarak ifade edebiliriz. Bu parametre seti, verilen  $O$  gözlem dizisi için bir olasılık ölçümü tanımlar; örneğin:  $P(O | \lambda)$ . SMM modelini  $\lambda$  parametre setini göstermek ve ilgili gözlem ölçüm değiştirilebilirliği için kullanırız.

### 2.1.3 Olasılık Değerlendirmesi

SMM modellerindeki temel problem, SMM modeli  $\lambda = (A, B, \pi)$  ile verilen  $O = (o_1, o_2, \dots, o_T)$  gözlem dizisinin olasılığını hesaplamaktır. Bu problemin çözümündeki en sezgisel yol,  $T$  uzunluğundaki olası her durum dizisini baştan sona kadar saymaktır. Açıkçası, en çok  $N^T$  kadar durum dizisi vardır. Sabit bir durum dizisi için,

$$q = (q_1 q_2 \dots q_T) \quad (2.4)$$

$q_1$  ve  $q_T$  sırasıyla başlangıç ve bitiş durumlarıdır. Gözlem dizisi  $O$ ' nun olasılığı şudur:

$$P(O | q, \lambda) = \prod_{t=1}^T P(o_t | q_t, \lambda) \quad (2.5)$$

Yukarıdaki denkleme göre, gözlemlerin istatistiksel olarak bağımsız oldukları kabul edilir. Diğer bir deyişle,

$$P(O | q, \lambda) = b_{q_1}(o_1) \cdot b_{q_2}(o_2) \dots b_{q_T}(o_T) \quad (2.6)$$

Bundan başka, durum dizisi olasılığı  $q$  şu şekildedir:

$$P(q | \lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \dots a_{q_{T-2} q_{T-1}} \quad (2.7)$$

O ve q'nun aynı zamanda gerçekleşme olasılığı basit olarak yukarıdaki iki terimdir. Bu, model ile verilen gözlem dizisi O'nun olasılığı (2.8)'deki denklem üzerinde tüm olası durum dizileri q'nun toplamı ile elde edilir:

$$P(O, q | \lambda) = P(O | q, \lambda) P(q | \lambda) \quad (2.8)$$

$$\begin{aligned} P(O | \lambda) &= \sum_{q_1, q_2, \dots, q_T} P(O | q, \lambda) P(q | \lambda) \\ &= \sum_{q_1, q_2, \dots, q_T} \pi_{q_1} b_{q_1}(o_1) a_{q_1 q_2} b_{q_2}(o_2) a_{q_2 q_3} \dots a_{q_{T-2} q_{T-1}} b_{q_{T-1}} \end{aligned} \quad (2.9)$$

Yukarıdaki denklemi yorumlarsak  $t=1$  anında  $\pi_{q_1}$  olasılığı ile  $q_1$  durumundayız ve  $o_1$  sembolü  $b_{q_1}(o_1)$  olasılığı ile üretilir. Zaman  $t$  anından  $t+1$  anına kadar değişir ve  $q_1$  den  $q_2$  durumuna  $a_{q_1 q_2}$  olasılığı ile bir geçiş yaparız ve  $b_{q_2}(o_2)$  olasılığı ile  $o_2$  sembolünü üretir. Bu hesaplama son geçişe kadar,  $T$  anında,  $q_{T-1}$  durumundan  $q_T$  durumuna kadar devam eder ve  $o_T$  çıktı sembolünü üretir.

Anlaşılabacağı üzere, doğrudan tanım kullanarak  $P(O | \lambda)$  hesaplama,  $2TN^T$  hesaplama gerektirir. Bu hesaplamaların karmaşıklığı  $N$  ve  $T$ 'nin küçük değerleri için bile uygun değildir. Şöyle ki,  $N=3$  ve  $T=100$  olursa,  $2 \cdot 100 \cdot 3^{100} \approx 10^{40}$  hesaplama vardır. Bu yüzden, daha fazla etkili algoritmalara ihtiyac vardır.

Geriye-dönük Algoritma (*Backward Algorithm*) ve İleriye-dönük Algoritma (*Forward Algorithm*) bu hesaplamayı yerine getiren özyinelemeli (recursive) metotlardır. Bu algoritmaların en önemli özelliği her birinin verilen bir andaki bir durumun olasılığını hesaplayabilmesidir.

### 2.1.3.1 İleriye-Dönük Algoritma (Forward Algorithm)

İleriye-dönük algoritma değişkeni,  $\alpha_t(i)$ , şu şekilde tanımlanır:

$$\alpha_t(i) = P(o_1 o_2 \dots o_t, q_t = i | \lambda) \quad (2.10)$$

Örneğin,  $t$  anında  $i$  durumunda  $\lambda$  modeli verilen kısmi gözlem dizisi  $o_1 o_2 \dots o_t$ 'nin olasılığını verebiliriz. İleriye-dönük algoritma ayrışık kelime tanımada faydalıdır.

### 2.1.3.2 Geriye-Dönük Algoritma (Backward Algorithm)

Benzer olarak, geriye-dönük algoritma değişkeni  $\beta_t(i)$  şöyle tanımlanabilir:

$$\beta_t(i) = P(o_{t+1} o_{t+2} \dots o_T | q_t = i, \lambda) \quad (2.11)$$

Örneğin,  $t$  anında  $i$  durumunda  $t+1$  den son duruma kadar  $\lambda$  modeli verilen kısmi gözlem dizisi olasılığını verebiliriz. İleriye-dönük ve geriye-dönük algoritma SMM'in eğitiminde faydalıdır.

Geriye-dönük algoritma ve ileriye-dönük algoritmalarının her ikisi de “en uygun” durum dizisi hesaplamada kullanılır ve parametre tahmin algoritması sonraki bölümlerde de ayrı ayrı açıklanmıştır.

### 2.1.4 “En uygun” Durum Dizisinin Hesaplanması

SMM formülasyonunda en önemli problem en uygun durum dizileri tahminidir. Verilen gözlem dizisi ile ilgili en uygun durum dizisini bulmak için birkaç yol vardır. Çeşitli uygunluk kriterleri tanımlanabilir. Bizim uygunluk kriterimiz  $P(q | O, \lambda)$ 'yi maksimize eden durum dizisini bulmaya dayanır. Bu

$P(q, O | \lambda)$ 'yı maksimize etme ile eşdeğerdir. Çözümü aslında dinamik bir programlama metodu olan Viterbi algoritması verir.

#### 2.1.4.1 Viterbi Algoritması

Viterbi değişkeni  $\delta_t(i)$ 'nin şu şekilde tanımlandığını kabul edelim:

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P(q_1 q_2 \dots q_{t-1}, q_t = i, o_1 o_2 \dots o_t | \lambda) \quad (2.12)$$

$q = (q_1 q_2 \dots q_T)$  gözlem dizisi  $O = (o_1 o_2 \dots o_T)$  ile verilen en iyi durum dizisidir. Diğer bir deyişle,  $\delta_t(i)$  sadece tek bir yolla en yüksek olasılıktır: t anında, ilk t gözlem için ve i durumdaki sonlar için hesaplama yapılırsa.  $\delta_t(i)$ 'nin tekrarlamalı versiyonu şu şekilde yazılabilir:

$$\delta_{t+1}(j) = \left[ \max_i \delta_t(i) a_{ij} \right] b_j(o_{t+1}) \quad (2.13)$$

#### 2.1.5. Parametre Tahmini

Parametre tahmini hesaplanması Saklı Markov Modellerindeki en güç problemdir. Model parametreleri A, B, ve  $\pi$  bir uygunluk kriteri sağlamak amacıyla hesaplanır. Çoğu zaman uygunluk kriteri  $\mathbf{O}$ 'nun eğitilmiş gözlemleri gösterdiği yerde  $P(O | \lambda)$ 'nin maksimize edilmesine dayanır. Bunu yapmak için, Beklenti-Maksimizasyonu (*Expectation-Maximization-EM*) metodu olarak da bilinen *Baum-Welch Metodu* kullanılır.

## 2.2 YAPAY SİNİR AĞLARININ SES TANIMADA KULLANILMASI

Yapay sinir ağları (YSA) günümüzde bilgi sınıflama ve bilgi yorumlamann içinde bulunduğu değişik problemlerin çözümünde kullanılmaktadır [33]. Özel olarak ses tanıma problemi ele alındığında özellik vektörü çıkarılmış bir ses sinyalinin tanınması da bu kapsama girmektedir. Ses tanıma sürecinde ses sinyalinin modellenmesi aşamasında yapay sinir ağlarını kullanabiliriz.

Yapay sinir ağlarının ses tanımadaki uygulaması üzerine bilgi vermeden önce ses tanıma amacımız için YSA teorisinden biraz bahsedelim.

### 2.2.1 Kavramlar ve Terimler

YSA biyolojik sinir ağlarından esinlenerek geliştirilmiş, bilgi işlem sistemleridir.

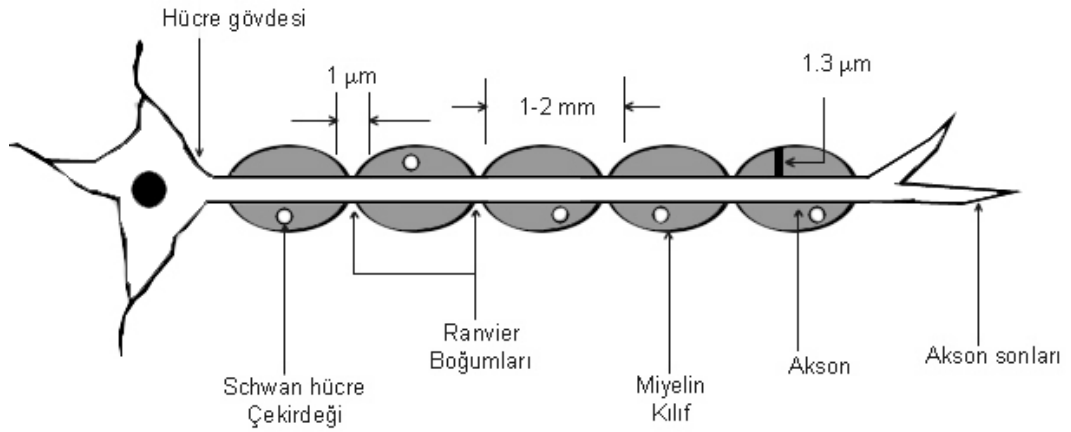
Bir sinir ağı, bilgiyi depolamak için doğal eğilimi olan basit birimlerden oluşan paralel dağıtılmış bir işlemcidir. Bu ağlar sinaptik ağırlıklar olarak bilinen nöronlar arası bağlantı kuvvetlerini, bilgiyi depolamak için kullanır. Benzer olarak YSA da birçok işlemciden oluşur ve farklı biçimlerde ifade edilebilen nümerik verileri taşıyan bağlantı yada ağırlıklarla birbirine bağlı yapılardır.

Biyolojik sinir ağlarında girdi sinyallerini alan, yorumlayan ve uygun çıktıyı ileten temel işlemci nöron olarak adlandırılır. Bir nöron, gövde (cell body), gövdeye giren sinyal alıcıları (dentrıt) ve gövdeden çıkan sinyal iletici (akson) olmak üzere üç kısımdan oluşur [13].

Dentritler, nörona bilgiyi alan ve sayısal olarak birden fazla olabilen yapılardır ve iç yapıları nöronla aynıdır.

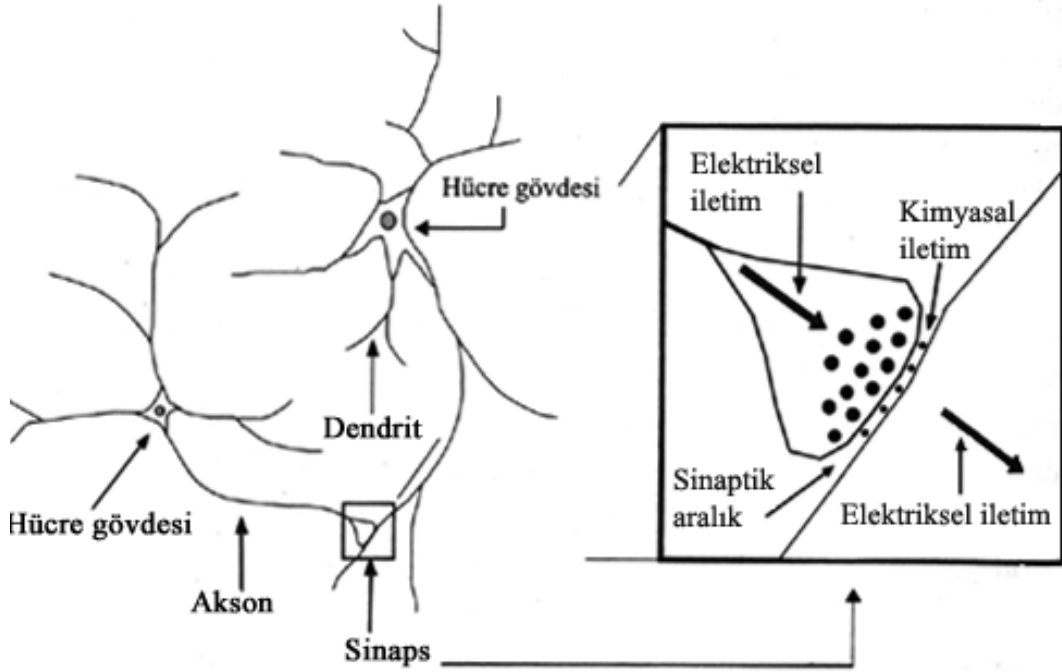
Aksonlar, dentritten aldığı bilgiyi diğer hücrelere aktaran uzantılardır. Uzunlukları birkaç mikrondan, 1-2 metreye kadar değişebilir. Her nöronun yalnızca

bir aksonu vardır. Aksonlar akson kesecikleri denilen ve içlerinde çok miktarda vezikül bulunan bir çok yapı ile sonlanırlar. Bu keseciklerde sinaptik geçişte rol alan kimyasal taşıyıcılar (nörotransmitter) bulunur. Aksonlar özel bir örtüye sahip olmalarına göre miyalinli yada miyalinsiz olarak sınıflandırılabilirler. Akson üzerini örten miyalin kılıfın, yalıtım ve darbe hızını arttırmak gibi iki önemli görevi vardır. Şekil 2.2’de miyalinli bir nöronun yapısı gösterilmiştir.



Şekil 2.2 Miyalinli Bir Nöron Yapısı

Sinir hücreleri, dış kısmı pozitif, iç kısmı negatif yüklü ve ortasında çift lipid tabakasıyla yalıtılmış, kutuplanmış bir kondansatöre benzer. Bir uyarıcı ile bu kondansatör nötrlenebilir hatta anlık olarak ters kutuplanabilir. Zarın belirli bir bölgesinde 1ms kadar süren bu potansiyel değişiminden sonra zar bu bölgede eski haline dönerken, potansiyel değişikliği zar boyunca yayılmaya devam eder. Akson boyunca iletilen bilginin temel birim olan bu potansiyel değişimine aksiyon potansiyeli denir.

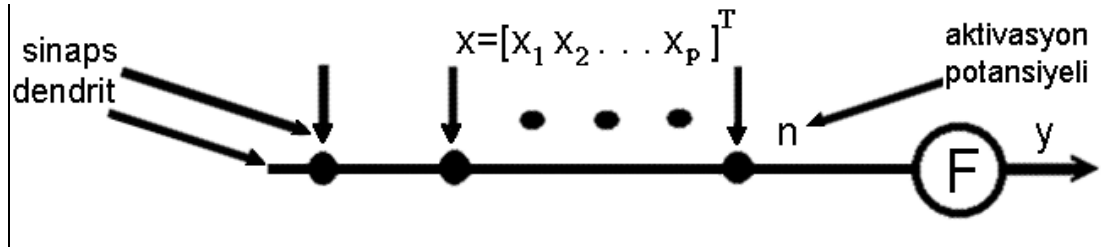


Şekil 2.3 Biyolojik Nöron

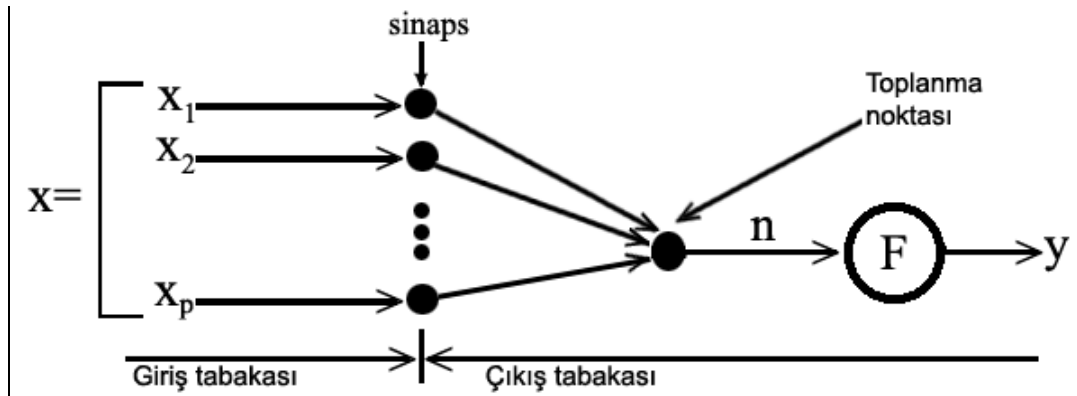
Sinir hücreleri arasında iletişimin gerçekleştiği, yapısal ve fonksiyonel olarak özelleşmiş bölgelere sinaps adı verilir. Mesajı gönderen ve presinaptik hücre olarak adlandırılan hücre ile mesajı alan ve postsinaptik hücre bu kavşak bölgelerinde bir birlerine oldukça yaklaşırlar.

Yapay sinir ağları, biyolojik sinir ağlarından esinlenerek, bir birine bağlı doğrusal ve/veya doğrusal olmayan bir çok işlemleri elemandan oluşur. Biyolojik nöron yapısı şematik olarak Şekil 2.3’de gösterilmiştir. Bir yapay nöron temel olarak girişler, ağırlıklar, toplam fonksiyonu, aktarım fonksiyonu ve çıkış olmak üzere beş kısımdan oluşur.

P girişli ve n çıkışlı bir yapay nöronun farklı grafik gösterimleri Şekil 2.4, Şekil 2.5, Şekil 2.6 ve Şekil 2.7’de gösterilmektedir.

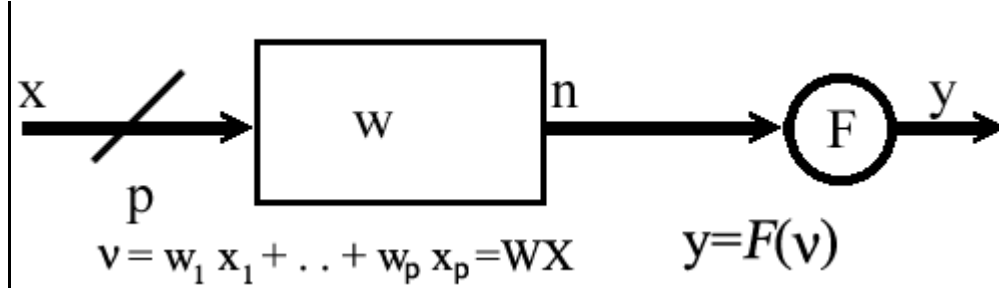


Şekil 2.4 Dendrit gösterimi

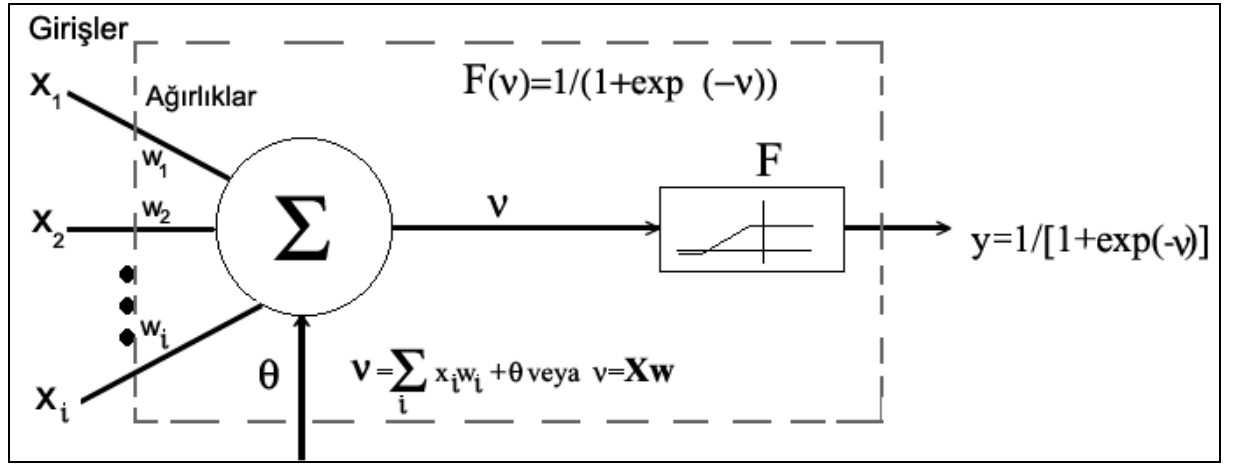


Şekil 2.5 İşaret akış gösterimi





Şekil 2.6 Blok diyagram



Şekil 2.7 Ayrıntılı gösterim

Bu gösterimlerde  $x$  girişleri,  $y$  çıkışı,  $F$  aktivasyon fonksiyonunu,  $p$  nöron giriş sayısını,  $n$  çıkış sayısını,  $w$  ise ağırlıkları ifade eder.

Dentrit gösteriminde presinaptik aktiviteleri giriş işaretlerinin  $p$  elemanlı sütun vektörü olarak gösterilir.  $x = [x_1, x_2, \dots, x_p]^T$  giriş desenlerinin uzayı  $p$  boyutludur.

Sinapslar ağırlıklar olarak adlandırılan ayarlanabilir parametreler ile karakterize edilirler. Ağırlıklar,  $p$  elemanlı satır vektörü olarak düzenlenir.

$$W = [w_1, w_2, \dots, w_p] \quad (2.14)$$

İşaret akış gösteriminde, p tane ağırlığı olan bir nöron giriş noktalarının bir katmanı şeklinde düzenlenir. Ağırlıklar, giriş ile toplama noktası arasındaki bağlantılara karşılık gelir.

Sinapslardan ve dentritlerden geçen giriş işaretleri, ‘toplam post-sinaptik aktiviteyi tanımlayan’ aktivasyon potansiyeli olarak toplanır.

Aktivasyon potansiyeli giriş işaretlerinin ve ağırlıklarının lineer toplamı olarak şekillenmiştir. Yani ağırlıklar ile geçiş vektörlerinin çarpımıdır:

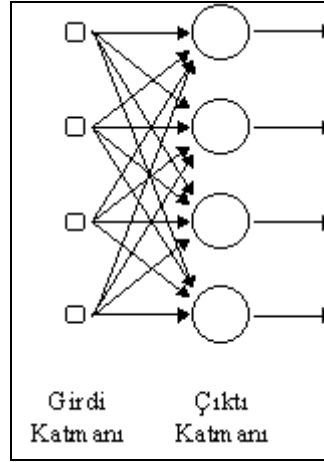
$$v = \sum_{i=1}^p w_i x_i = [w_1 w_2 \dots w_p] \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix} = WX^T \quad (2.15)$$

## 2.2.2 Yapay Sinir Ağlarının Sınıflandırılması

Yapay sinir ağlarında, ağ mimarisini belirleyen önemli etmenlerden biri de öğrenme algoritmasıdır. Genel olarak ağ mimarileri dört ana grupta toplanabilir. [21][6]. Bu alt-bölümde bu ağ mimarilerinin her biri hakkında kısa bilgi verilecektir.

### 2.2.2.1 Tek Katmanlı İleri Beslemeli Ağlar

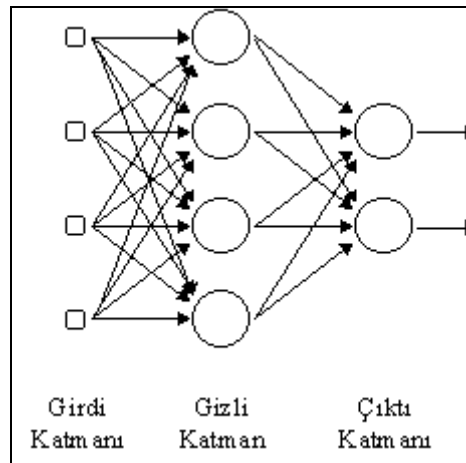
Katmanlı modellerdeki en basit ağ tipi olup bir çıktı katmanı ve buna bağlı bir girdi katmanı bulunmaktadır, Şekil 2.8’de gösterilmiştir.



Şekil 2.8 Tek Katmanlı İleri Beslemeli Ağ Modeli

### 2.2.2.2 Çok Katmanlı İleri Beslemeli Ağlar

Tek katmanlı ağlardaki girdi ve çıktı katmanından başka, bir yada daha fazla sayıda gizli katman içeren ağlara çok katmanlı ağ denir. Dış dünya tarafından doğrudan müdahale edilmediği için gizli katman adı verilen katmanda bulunan birimlere de gizli birimler adı verilir, Şekil 2.2’de gösterilmiştir.

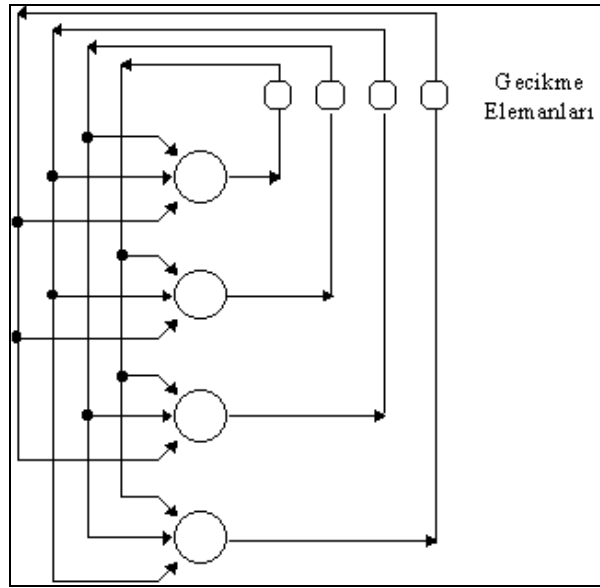


Şekil 2.9 Çok Katmanlı İleri Beslemeli Ağ Modeli

Çok katmanlı ağlar tek katmanlı ağlara göre daha karmaşık problemlere çözüm getirebilmektedir. Ancak eğitilmesi zordur.

### 2.2.2.3 Geri Dönümlü Ağlar

Geri dönümlü ağların ileri beslemeli ağlardan farkı, bu ağlarda en az bir tane geri-besleme döngüsünün bulunmasıdır, Şekil 2.10'da gösterilmiştir.

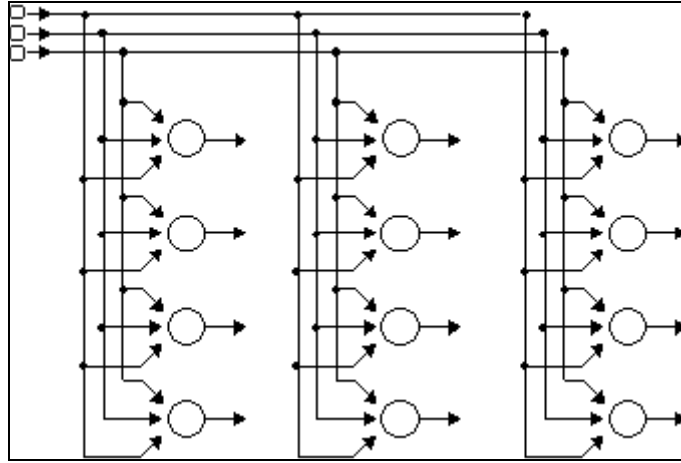


Şekil 2.10 Geri Dönümlü Ağlar

Bu gibi yapıların bazılarında senkronizasyonun sağlanması için gecikme elemanlarının kullanılması gerekebilmektedir. Bu yapıya sahip ağlarda geri-besleme sinyali aynı katmandaki nöronlara gönderilebileceği gibi, alt katmanlardaki nöronlara da gönderilebilir.

### 2.2.2.4 Kafes yapılı ağlar

Kafes yapılı ağlar bir, iki veya çok-boyutlu nöron dizinlerinden meydana gelmekte olup girdi katmanı, dizinlerin hepsine girdi bilgisi sağlamaktadır, Şekil 2.11'de gösterilmiştir.



Şekil 2.11 Kafes yapılı ağ modeli

### 2.2.3 Yapay Sinir Ağlarında Öğrenme

Yapay sinir ağlarında bilgi, nöronlar arasındaki bağlantılar üzerindeki ağırlık değerleri üzerinde tutulur. Bu yüzden yapay sinir ağlarının eğitimi ve öğrenme bu ağırlıklara değerlerin verilmesi ve değiştirilmesi anlamına gelmektedir. Öğrenme yaklaşımı, ağ yapıları için ayırt edici bir özelliktir. En genel anlamda öğrenme, güdümlü öğrenme ve güdümsüz öğrenme olmak üzere iki grupta toplanabilir. [17]

#### 2.2.3.1 Güdümlü Öğrenme

Güdümlü öğrenme, yapay sinir ağlarında yaygın olarak kullanılan bir yöntemdir. Eğitim sırasında sisteme bir girdi ve hedef çıktı vektörü çifti verilir. Giriş bilgisinden üretilen çıktı değeri ile hedef çıktı değeri kıyaslanarak ağırlıkların değiştirilmesi için gerekli bilgi hesaplanır. Bu kıyaslamadan elde edilen fark, hata olarak belirlenen değerden küçük oluncaya kadar eğitim devam eder. Hata değeri istenen değer altına düştüğünde ağırlık değerleri sabitlenir ve eğitim biter. Yapay sinir ağlarının eğitiminde kullanılan hedef çıktı 1 veya 0 gibi ikili olabileceği gibi bir örüntü de olabilir. Belli bir girdi kümesine karşılık ilgili çıktı kümesini hatırlayacak şekilde eğitilmiş sistemlere çağrışimli bellek adı verilir. Eğer girdi vektörü ile çıktı vektörü

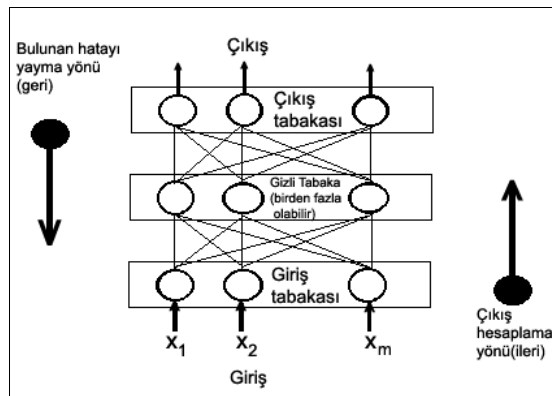
aynı ise buna öz-çağrışimli bellek, çıktı vektörü farklı ise buna da karşıt-çağrışimli bellek adı verilir.

### 2.2.3.2 GÜDÜMSÜZ ÖĞRENME

Kullanılan girdi vektörlerinin hangi sınıfa ait olduğu verilmiyorsa, bu sınıflama işlemini girdileri birbirleriyle karşılaştırarak sistemin kendisi yerine getiriyorsa, bu tür sistemlerdeki öğrenme yöntemine *güdümsüz öğrenme* adı verilir. Bu sistemlerde bir grup girdi vektörü sisteme verilir, ancak hedef çıktılar belirtilmez. Sistem girdiler içinde birbirine en çok benzeyenleri gruplar ve her bir grup için farklı bir örüntü tanımlar. Özdüzenlemeli özellik haritaları bu yöntemi kullanarak sınıflama işlemini yerine getirir.

### 2.2.4 Çok Katmanlı Perseptron ve Geri Yayımlı Öğrenme

Çok katmanlı perseptronlar, ileri beslemeli, tam bağlantılı, bir girdi katmanı, bir veya birden fazla gizli katman ve bir çıktı katmanından oluşan hiyerarşik yapıya sahip yapay sinir ağı mimarileridir. Böyle bir ağdaki herhangi bir nöronun çıkışı  $y_k = f(\sum w_k x)$  ile hesaplanır. Giriş katında herhangi bir bilgi işleme yapılmaz. Buradaki işlem elemanı sayısı problemin giriş sayısına bağlıdır. Ara katman sayısı deneme yanılma yoluyla bulunur. Çıkış katmanı eleman sayısı ise uygulanan probleme bağlıdır [17].



Şekil 2.12 Çok Katmanlı Perseptron Yapısı

Geri yayımlı öğrenme metodu olarak isimlendirilen eğitim algoritmasının temeli ilk olarak Werbos'un Harvard Üniversitesi'nde verdiği doktora tezinde ileri sürülmüş, daha sonra Parker tarafından MIT'nin bir teknik raporunda ele alınmış ve Rumelhart ve arkadaşları tarafından da popüler hale getirilip uygulanabilir şekle dönüştürülmüştür [10].

Daha önce Adaline mimarisi ile kullanılan LMS veya Delta kuralının geliştirilmiş ve çok katmanlı yapıya uyarlanmış hali olan geriyayımlı öğrenme metodu, çok katmanlı perseptron yapısı üzerinde çok çeşitli problemlerin çözümünde kullanılmaktadır. Burada geri yayılım kavramı, oluşan hatalar için kullanılmaktadır.

Geri yayımlı öğrenme metodu ile çok katmanlı perseptron ağı üzerinden iki geçiş söz konusudur. İleri besleme aşamasında eğitim için kullanılan girdi, sisteme beslenir ve bunun sonuçları her bir katmanı geçerek çıktı katmanına kadar gelir ve girdiye karşılık bir çıktı elde edilir. Bu aşama sırasında ağ üzerindeki ağırlık değerleri sabit tutulur. İkinci aşama olan geri besleme aşamasında, elde edilen çıktı ile hedef çıktı arasındaki farktan hata sinyali elde edilir ve bu sinyal ağ yapısında geriye doğru yayılır. Bu geri yayılım aşamasında, oluşan hatayı minimuma indirecek şekilde ağırlık değerleri güncellenir.

Bu güdümlü öğrenme işleminin tekrarlanması ile sistem, kendisine beslenen girdi örneklerini öğrenir. Eğitim bittikten sonra sistemin uygulama sürecinde sadece ileri beslemeli ilk aşama yeterli olmaktadır. Bu durumda sistemin eğitilmesi uzun süre almakla beraber, eğitilmiş bir sistemden bilgi alınması çok hızlı olmaktadır

### **2.2.5 Ses Tanımda Yapay Sinir Ağları**

Ses olduğu takdirde ağı giriş sesin bazı bilgilerini içeren kısa süreli bir spektral gösterim yani özellik vektörü olabilir.

YSA'lar herhangi bir eşleme için eğitilebildiğinden ses tanımada ses sinyalinin sınıflandırılması (modellenmesi) aşamasında kullanılabilir. Özellikle bundan sorumludur, çünkü diğer bazı sınıflandırma tekniklerinin karmaşıklık düzeyleri fazladır.

Ses tanımada kullanılacak farklı ağ tipleri vardır. Geri beslemeli bir ağda, ağa verilen bir yapı sınıflayıcı gibi kullanılarak bazı çıktı birimleri elde edilir. Çıktı birimlerinden bazılarının hareketi kısıtlanırken, her sınıf için istenilen çıktı birimi etkin tutulur. Bu sınıflandırma fonksiyonunu daha da açarsak, nesnelerin gözlemlenebilir sınıflardaki olasılık yoğunluk fonksiyonunu tahmin etmek için YSA'lar kullanılmalıdır. Bir ağ böylece SMM sistemi gibi bir probabilistik olayı içine alan bir sistemin bileşeni de olabilir [9][19].

İleri beslemeli veya geriyayılım algoritmaları ile eğitilen ağalara ek olarak sınıflandırmada kullanılacak farklı YSA yapıları vardır. Burada bahsedilebilecek iki yaklaşım vardır: Öğrenme Vektör Nicelendirmesi Ağı (Learning Vector Quantization-LVQ) sınıflar arasındaki farkı belirleyebilen iki katmanlı bir ağı eğiten bir algoritmadır [9][27]. Diğer yaklaşım ise Radyal Tabanlı Fonksiyon Ağıdır (Radial Basis Function-RBF)

Bundan başka, ses verisi için kompleks veri gösterimlerinin yeni bir türünü oluşturmak için kullanılan YSA yapıları vardır. Bu tür ağlara bir örnek Kohonen ağları olarak bilinen Özörgütlemeli Harita Ağlarıdır (Self Organizing Maps-SOMs) [5][6]. Bu ağlar Rekabete Dayalı Öğrenme (Competitive Learning) ile giriş verisinin yapısına göre kendilerini organize eder. Ayrıca SOM'lar konuşmanın gösterimi, analiz edilmesi ve karakterize edilmesinde faydalıdır. Örneğin, seslendirme bozukluklarını belirlemek için kullanılırlar.

YSA metotları farklı tipteki yapıları öğrenebilir. Bağlantılardaki ağırlık değerlerini değiştirerek öncelik verdiği yapıyı değiştirebilir. Bu nedenle YSA'daki başarı oranı fazladır, özellikle ses tanıma ve yorumlanmasında etkilidir. Bir YSA



sistemi yüksek sayıda birim gerektirir ve eğitimi zordur. Dile ait işlemlerde insan yeteneğinin büyük bölümü beynimizde doğuştan gelen sinirsel bağlantılardan ortaya çıkmıştır. Doğuştan gelme ve öğrenme arasındaki bağlantı için modeller geliştirmek istenmiştir, ancak bu yıllardır makinalarla sağlanamamıştır. YSA sistemleri geliştirilirken ortaya çıkan problem bütün modellerin ağda gizli olmasıdır. Bu da ağda gerçekte ne olduğunu anlamamızı ve böylece ses tanıma işleminin nasıl gerçekleştiğini kavrayabilmemizi zorlaştırır

Günümüze kadar, ses tanıma sistemlerinin büyük bölümünde yapay sinir ağları modelleri başarıyla uygulanmıştır. YSA'daki önemli özellikler şunlardır:

YSA'daki öğrenme algoritmaları tanıma için eğitilen ağın ayrı sınıfları arasındaki farkları yapısı gereği görebilir. Gerçekte bu gösterir ki, çeşitli YSA'ların çıktısı girişin verdiği çıktı sınıflarının bir sonraki gelenin olasılıklarını kesin tahmin etmesi olarak yorumlanabilir. Yapay sinir ağları doğru çıktı sınıfı ve alternatif hatalı sınıflar arasındaki farklılığı maksimize ederek eğitilir ve bundan dolayı sınıflandırma hatası minimize edilir.

Ağa giriş olarak verilen istatistiksel dağılımın özellikleri hakkında güçlü varsayımlarda bulunmaya gerek yoktur. Aynı zamanda, YSA'lar sınıflandırmayı gerçekleştirmek için çeşitli kısıtları birleştirebildiğine göre, özelliklerin bağımsız olduğunu farz etmek gereksizdir.

Girişte birkaç çerçeve bulunmasıyla veya geri besleme bağlantılarını kullanarak, ağ ses çerçeveleri arasındaki içerik bağıllığını modeller.

Çoğu YSA sistemleri geleneksel Von Neumann bilgisayar yapısında simule edilmesine rağmen, YSA yapılarının paralel ve kurallı doğası onları kolay donanım uygulamalarına uyumlu yapar.

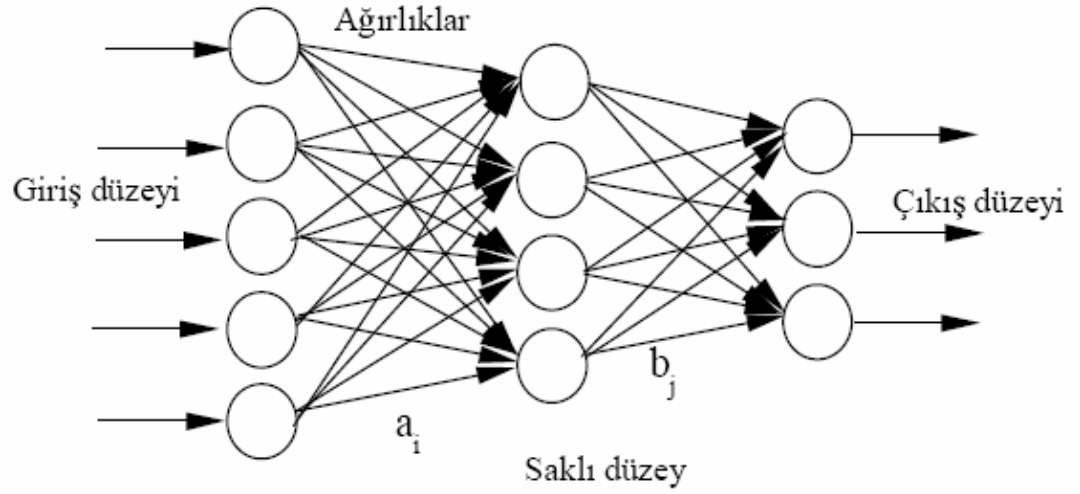
Ayrışık kelime tanıma ve ses fonem sınıflandırılmasında fon sınırlarının belirlenmesi için ses ön-parçalara ayrılır. YSA sistemleri bu konuda SMM ile

karşılaştırılabilir ve bazen SMM'lerden daha iyi sonuçlar verir. [14]. YSA'lar aynı zamanda bozuk ses sinyallerindeki gürültüyü azaltan ön uç uygulamaları (front-end applications) için faydalıdır. Bazen, YSA'lar kendi başlarına bazı sürekli ses tanıma görevleri için henüz etkili değildir. Bir bağlantıcı ağı eğitmek amacı ile giriş (özellik vektörleri) ve çıkış (tanınan sınıflar) arasındaki uygunluğu belirlemek gereklidir. Bu bilgi sürekli ses için uygulanamaz olma eğilimindedir, çünkü kelime sınırlarının yerleri ilerlerken genelde bilinmez. Daha genel olan, tekrarlanan yapılar ve Zaman Gecikmeli Yapay Sinir Ağları (Time Delay Neural Network-TDNN) sesin zamanla değişen yapısını yakalar. Bağlantıcı modeller zaman ölçeği değişimlerinin üstesinden kolaylıkla gelemeyebilir. Buna karşı SMM'lerin güçlü olan bir yanı spektral parçalama ve geçici özellikleri olmasıdır. Bundan dolayı, sürekli sesi tanıma ve parçalamada YSA ve SMM birlikte daha verimlidir [14].

### 2.2.5.1 Nöron Ağları Yaklaşımı

Nöron ağları, sesli ifade tanıma yöntemi olarak son zamanlarda daha çok kullanılmaktadır. Nöron ağı, birlikte işlem yaptıklarında bir işi başarıyla sonuca ulaştıran bir dizi işlemci birimin (düğüm, birim, yada nöron modeli) oluşturduğu bir ağ olarak tanımlanabilir. Her bir işlemci birim başka birimlerden gelen çıkış verilerini bir fonksiyondan geçirerek çıkışta üretilen etkinlik değerini başka birimlere iletir. Buradaki fonksiyon genellikle girişlerin ağırlıklı bir toplamıdır ve sigmoid benzeri doğrusal olmayan bir fonksiyon üzerinden sonuç, sonraki birimlere aktarılır. Nöron ağları alanı, nöron ağının içerdiği bu birimlerin yani kısaca nöronların birbirine bağlanma yollarını ve ağ içindeki ağırlıkların kararlaştırılması konusu ile ilgilidir. Ağ içindeki nöronların ağırlıkları hesaplanırken elde bulunan eğitim veri kümesi kullanılır. Bu veri kümesi kullanılarak nöronların öğrenmesi sağlanır. Nöron ağı yöntemiyle hesaplama, klasik hesaplamadan aşağıdaki özellikleri ile ayrılır;

- Basit ve benzer işlemci elemanların kullanılması,
- Bu elemanların önemli ölçüde koşut olarak işleyebilme özelliği,
- Bilgi, nöronlarda depolanmadığı gibi kural olarak da saklanmaz, bunun yerine nöronlar arası bağlantılar biçiminde ağ üzerinde dağıtılır.



Şekil 2.13 Basit İleriye Doğru Beslemeli (Feed Forward) Bir Nöron Ağı

Nöron ağlarının özelliği insan beyninin işleyişinin modellenmeye çalışılmasıdır. Bu amaç doğrultusunda nöron ağlarının kullanılış alanı çok geniş olmakla birlikte burada sadece nöron ağlarının sesli ifade tanımada kullanılışı konu edilecektir. Şekil 2.13’de daireler nöron benzeri hesaplama birimlerini temsil etmektedir. Her birim girişindeki değerlerin ağırlıklı toplamını doğrusal olmayan bir fonksiyondan geçirip sonraki birimlere iletir.

Bu tür nöron ağlarına çok düzeyli algılayıcı (Multi-layer perceptron -MLP) denir. Bu tür nöron ağlarının belirtilen bir giriş kümesi için istenen çıkış değerlerini üretebileceği denemeler sonucunda gösterilmiştir. Ağdaki ağırlıklar başlangıçta rasgele değerlerle güncellenmektedir. Bir giriş örneği nöron ağına giriş olarak verilir ve çıkış hesaplanır. Daha sonra bu çıkış istenen çıkış ile karşılaştırılır. Nöron ağının çıkışı ile istenen çıkış arasındaki uzaklık, ağırlıkların bir fonksiyonu olarak belirlenir ve ağırlıklar hata miktarının azaltılması için değiştirilir. Ağırlıkların değiştirilmesi ile ilgili hesaplamalar nöron ağının çıkış düzeyinde başlar ve daha sonra bir önceki düzey göz önüne alınır. Böylece ağırlık güncelleme çıkış düzeyinden başlayan ve giriş düzeyine doğru ilerleyen bir işlem olarak devam ettirilir. Bu şekildeki eğitme

algoritmaları, geriye doğru hata yayılması (error back propagation) olarak tanımlanır. Bu temel algortmada bir dizi değışiklik yapılarak düşük hata oranlarına daha hızlı ulaşma amaçlanır. eğitme aşaması bir çok tekrarlı işlemin yürütülmesini gerektirir. Temel olarak nöron ağları için iki türlü öğrenme biçimi vardır. Gözetimli ve gözetimsiz (supervised-unsupervised) öğrenme. Geriye doğru yayılma algoritması gözetimli öğrenme olarak tanımlanır. Çünkü, eğitim için kullanılan veriler verilen girişler için istenen çıkışları da içerir. Gözetimsiz öğrenmede istenen çıkışlarla ilgili veriler eğitim verilerinde yer almaz. Bunun yerine giriş verilerin genel özelliklerine dayalı ya da genel bazı prensiplere dayalı fonksiyonlar kullanılarak ağırlıklar giriş verilerine göre ayarlanır.

### **2.2.5.2 Nöron Ağlarının Sesli ifade Tanımda Kullanılması**

İlke olarak nöron ağları giriş ve çıkışlar arasında bir ilişki oluşturmak için eğitilebildiklerinden dolayı örüntü tanıma (pattern recognition) ya da sınıflandırma için rahatlıkla kullanılabilirler. Hatta, sınıflandırmanın çok karmaşık karar düzeylerinin oluşturulmasını gerektirdiği göz önüne alınırsa, nöron ağlarının çok etkili bir şekilde bu alanda kullanılabileceğini söylemek yanlış olmaz. İleri doğru beslemeli (feed forward) nöron ağlarının sınıflandırma için muhtemel bir kullanım biçimi, örüntü sınıflarının her biri için nöron ağında bir çıkış birimi tanımlamakla mümkün olabilir. Bir örüntü sınıfına ait verilerin giriş olarak nöron ağına verilmesiyle çıkış nöronlarından bir tanesinin değeri en büyük olurken diğerleri de olabildiğince düşük tutulmaya çalışılır. Sesli ifade tanıma bağlamında giriş verileri özellik vektörleri olabilir. Kısa süreli spektral değeri kullanılması sesli ifade içeriğinde kayıplara neden olabilir. Bu değeri kullanılması, sabit uzunluklu bir pencerenin sesli ifade üzerinde belirli zaman aralıklarıyla kaydırılması sonucu oluşan verilerin bir nöron ağına giriş olarak verilmesiyle bir sınıflandırma yapma şeklinde olabilir.

### 2.2.5.3 Zaman Gecikmeli Nöron Ağları (Time Delay Neural Network)

Zaman gecikmeli nöron ağı (Time Delay Neural Network-TDNN) kısa süreli spektral değerlerin kullanılmasından doğan içerik kayıplarını önlemenin bir yolu olarak ortaya çıkmış bir nöron ağı yaklaşımıdır. Bu yaklaşım 1989 yılında Weibel ve arkadaşları tarafından oluşturulmuştur. Bu yapıda 10 ms aralıklarla sesli ifadeden çıkarılan 16 spektral değer bir özellik vektörü olarak alınmaktadır. Bu özellik vektörlerinden 7 tanesi nöron ağına giriş verisi olarak verilmektedir. Bir sonraki giriş verisi 7\*10ms uzunluğundaki bu pencerenin 10ms kaydırılması ile elde edilmektedir. Amaç her bir giriş çerçevesi için saklı düzeydeki nöronların ağırlıklarına eklemeler yaparak önceki verilerin de çıkış üzerinde etkili olmasını sağlamaktır. TDNN yapısına sahip nöron ağları, seyrek bağlı ileri doğru beslemeli ve paylaşımlı ağırlıklara sahip olarak tanımlanabilir. Kullanım amaçları büyük bir örüntü içindeki özelliklerin konumdan bağımsız olarak tanınmasıdır. Bu özellikleri sayesinde belirli sayıda özellik vektörü içinde belirli bir fonemin geçim geçmediğini önemli bir doğrulukla tespit edebilirler.

### 2.2.5.4 Yineli Nöron Ağları (Recurrent Neural Network)

Sabit süreli zaman pencerelerinden kaçınmak için yineli nöron ağı (Recurrent Neural Network) kavramı ortaya çıkmıştır. Bu yapıda ağın geçmiş veriler hakkındaki bilgilerin daha önceden belirlenmemiş bir süre boyunca ağda tutulması öngörülür. Bu süre nöronların ağırlıklarına ya da giriş verilerine bağlı olabilir ama önceden belirlenmiş sabit bir süre değildir. 1989 yılında Robinson tarafından bu tür ağların eğitilmesi için geriye doğru yayılma algoritmasının bir türevi geliştirilmiştir. Bir sürekli nöron ağı, geriye doğru besleme özelliğinin eklendiği, bir dizi standart hata yayılmalı ağın birleşimi şeklinde düşünülebilir [1]. Giriş ve çıkış vektörleri iç ve dış olmak üzere iki kesime ayrılır. İç çıkış verileri bir sonraki vektörün iç giriş kesimine geri besleme olarak verilir. Böylece zaman içinde bir süreklilik sağlanır. Yukarıda bahsedilen, geriye doğru yayılma algoritması ile eğitilen ileri doğru beslemeli ve yineli nöron ağlarının dışında sınıflandırma amacıyla kullanılan farklı bir yapıya sahip nöron ağları da vardır. Bunlara örnek olarak Öğrenen Vektör Ağırlıklandırma

(Learning Vector Quantization-LVQ) verilebilir [9]. Bu yaklaşımda amaç örüntü sınıflarını, olabildiğince, birbirinden ayırmaya çalışmaktır. Bunun için, örneğin, örüntü sınıflarını temsil eden vektörlerin birbirine uzaklıkları en büyük yapılmaya çalışılırken, aynı sınıfa ait olan özellik vektörleri de birbirlerine yakınlaştırılmaya çalışılır.

#### 2.2.5.5 Öz Düzenleyici Özellik Haritası (Self Organizing Feature Map)

Su ana kadar söz konusu edilen nöron ağı yapılarının dışında karmaşık verilerin, örneğin sesli ifade verilerinin, uygun bir biçimde gösterimlerini sağlamak amacıyla yönelik bir nöron ağı yaklaşımı olarak Öz Düzenleyici Özellik Haritası (Self Organizing Feature Map- SOM) verilebilir [9]. Bu ağ yapısı kendi kendisini düzenleyen ve bunun için yarışmalı(competitive) öğrenme biçimini kullanır. Bu öğrenme biçimi gözetimsizdir ve öğrenme giriş verilerine göre yapılır. Temel ilke iki boyutlu bir nöron haritası üzerinde birbirine yakın olan özellik vektörlerini belirten nöronların birbirine yakın noktalara getirilmesidir. SOM, giriş verisi olarak verilen çok sayıdaki vektörün iki boyutlu bir nöron dizisi üzerine dağıtılması olarak kısaca tanımlanabilir. Bu iki boyutlu dizideki her bir eleman giriş verisinde bulunan ve aynı özellikleri gösteren vektörleri temsil eden bir referans vektörünü belirtir. Dizinin kafes şeklindeki yapısı dikdörtgen ya da sekizgen olabilir. Herhangi bir giriş vektörü dizinin her bir elemanının temsil ettiği vektörle karşılaştırılır ve en yakın değer üreten elemanın bulunduğu konum yeni vektörün haritadaki yeri olarak belirlenir.

SOM, daha başka bir tanımlama ile çok boyutlu vektörlerden oluşan giriş verisinin iki boyutlu bir harita üzerine izdüşümü olarak ifade edilebilir. Gecikmeli ve sürekli bağlantıları olan nöron ağları her ne kadar kuramsal olarak sesli ifade şablonlarını modellemede başarı sağlasa da, su anda sesli ifadenin bazı özelliklerinin bu şablonlarda kayıpsız olarak temsil edilebilmesinde zorluklar söz konusudur. Sadece nöron ağı kullanmak sesli ifadenin alt birimlerini tanımakta önemli başarılar sağlamaktadır. Ancak sesli ifade tanımada nöron ağlarının yanı sıra HMM ve DTW yaklaşımlarının kullanımı ya da nöron ağı ile elde edilen sonuçların bir uzman sistem(Expert System) yardımıyla değerlendirilmesi gerekli olmaktadır. Nöron

ağlarının bir diğere dezavantajı ise gereken uzun süreli eğitim aşamasıdır. Nöron ağları üzerinde yapılan arařtırmaların büyük bir çoğunluđu günümüzde bu eğitim süresinin azaltılması için daha etkin algoritmaların bulunmasına yöneliktir.

## 2.3 SES İŞLEMEDE ZAMAN-ALANI METODLARI

### 2.3.1 Sesin Zamana Bağlı Olarak İşlenmesi

Saniyede 8000 kez örneklenmiş tipik bir ses sinyali incelendiğinde, ses sinyalinin özelliklerinin zamana bağlı olarak değiştiği gözlenir. Mesela ötümlü/ötümsüz uyarımı zamana bağlı olarak değişmektedir, ayrıca, sinyalin tepe genliğinde ve ötümlü bölgelerde temel sıklıkta değişiklikler gözlenir.

Burada önemli bir konu, ses sinyalinin özelliklerinin zamanla çok yavaş değiştiğinin farz edilmesidir. Kısa zamanlı işlem metodları, incelenen ses sabit özelliklere sahip sürekli bir ses sinyalinin kısa bir parçasıymış gibi uygulanır [7] Bu kısa bölümlere analiz çerçeveleri denir. Genellikle üst üste biner ve periyodik olarak tekrar edilirler.

Kısa zamanlı işleme teknikleri, matematiksel olarak şu formda ifade edilir:

$$Q_n = \sum_{m=-\infty}^{\infty} T[x(m)]w(n-m) \quad (2.16)$$

Ses sinyali, bazı ayarlanabilir parametreler kümesine bağlı olabilecek, doğrusal veya doğrusal olmayan bir dönüşümden ( $T[\ ]$ ), den geçirilir. Daha sonra, n örneklik bir pencere dizisiyle çarpılır. Sıfır olmayan çarpım sonuçları toplanır. Daima olmamasına karşın, genellikle pencere dizisi sonlu sayıdadır.  $Q_n$  değerleri bölgesel ağırlıklı  $T[x(m)]$  değerlerinden oluşur.

Bu bilgileri gösteren bir örnek, bir sinyalin kısa zaman enerjisidir. Enerji,

$$E = \sum_{m=-\infty}^{\infty} x^2(m) \quad (2.17)$$

ifadesi ile verilir.

Bu büyüklüğün, ses sinyalinin zamana bağlı özellikleriyle ilgili fazla bir bilgi vermemesinden dolayı fazla bir anlamı yoktur. Kısa zamanlı enerjinin bir gösterimi;

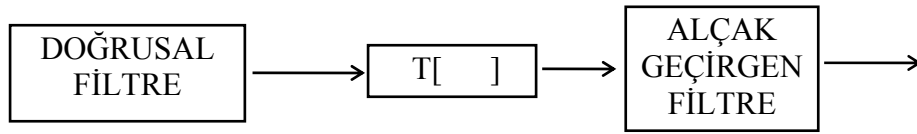
$$E_n = \sum_{m=n-N+1}^n x^2(m) \quad (2.18)$$



Bu,  $n-N+1$ 'den  $n$ 'e kadar olan  $N$  adet örneğin karesinin toplamıyla ifade edilen kısa zaman enerjisini gösterir.  $T[ ]$  ise bir kare alma işlemidir ve  $w(n)$  şöyle ifade edilir:

$$\begin{aligned} w(n) &= 1 & 0 \leq n \leq N-1 & & (2.19) \\ &= 0 & \text{diğerlerinde} & & \end{aligned}$$

$Q_n$  eşitliđi,  $T[x(n)]$  dizisiyle  $w(n)$  penceresinin ayrık evrişim şeklini göstermektedir. Dolayısıyla  $Q_n$ , doğrusal, zamandan bağımsız ve darbe cevabı  $h(n)=w(n)$  olan bir sistemin çıktısı olarak düşünülebilir.



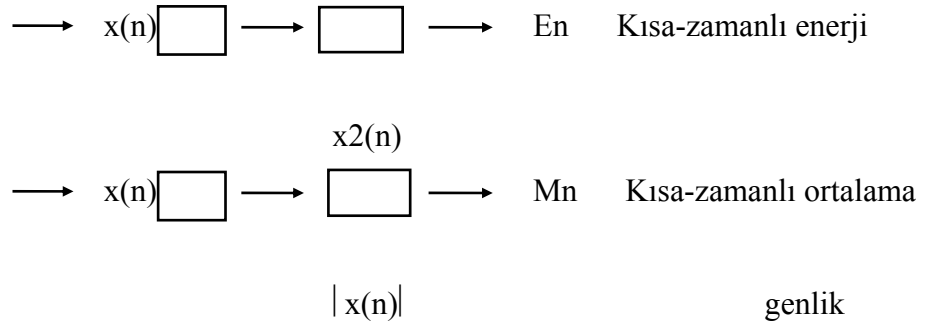
Şekil 2.2 Kısa Zamanlı Analizin Genel Gösterimi

### 2.3.2 Kısa Zaman Enerjisi ve Ortalama Genlik

Ses sinyalinin genliği zamana bağı olarak deđişir. Ötümsüz ses bölümleri, ötümlü seslerden daha düşük genliğe sahiptirler. Ses sinyalinin kısa zaman enerjisi, bu genlik deđişimlerini yansıtan uygun bir gösterim sağlar.

$$E_n = \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2 \quad \text{veya başka bir ifadeyle;} \quad (2.20)$$

$$E_n = \sum_{m=-\infty}^{\infty} x^2(m).h(n-m) \quad \text{burada } h(n)=w^2(n) \quad (2.21)$$



Şekil 2.3 Kısa zaman enerji ve ortalama genliđin blok şeması gösterimi

Darbe cevabı olan  $h(n)$ 'in seçimi, veya benzer şekilde, pencere işlevinin seçimi, kısa-zamanlı enerji gösteriminin doğasını tanımlar. Eğer  $h(n)$  çok uzun ve genliği sabit bir sinyal ise,  $E_n$ , zamanla çok az değişir. Böyle bir pencere, çok dar bir banda sahip alçak geçiren filtreye denktir. Aslında istenen  $h(n)$ , alçak geçiren bir filtre özelliğinde olmalı fakat, çıkışı sabite yakın olmamalıdır [6] Çünkü, kısa zaman enerjisi, genlik değişikliklerini yansıttığı için incelenmektedir. Zamana bağlı enerji gösterimi üzerinde pencerenin etkisi, dikdörtgen ve Hamming pencereleri için incelenmiştir.

$$\begin{aligned} \text{Dikdörtgen pencere; } h(n) &= 1 & 0 \leq n \leq N-1 & \quad (2.22) \\ &= 0 & \text{diğerlerinde} & \end{aligned}$$

$$\begin{aligned} \text{Hamming pencere; } h(n) &= 0.54 - 0.46 \cos(2\pi n / (N-1)) & 0 \leq n \leq N-1 & \quad (2.23) \\ &= 0 & \text{diğerlerinde} & \end{aligned}$$

Dikdörtgen pencere  $n-N+1$ 'den  $n$ 'e kadar  $N$  adet örneği eşit ağırlıklarla çarpmaktadır.

$$H(e^{j\Omega T}) = \frac{\sin(\Omega NT / 2)}{\sin(\Omega T / 2)} \cdot e^{-j\Omega T(N-1)/2} \quad (2.24)$$

Bu denklemde ilk sıfır,  $F = F_s / N$  olan analog sıklıkta bulunur. Burada  $F_s = 1/T$  örnekleme sıklığıdır. Bu, aynı zamanda dikdörtgen pencereye karşılık gelen alçak geçiren filtrenin kesim sıklığıdır. Aynı uzunluktaki Hamming penceresinin bant genişliği, dikdörtgen pencerenin bant genişliğinin iki katıdır. Hamming penceresi, dikdörtgen pencereye göre, geçiş bandı dışında daha fazla bir düşüş gösterir. Pencerelerdeki azalma oranı, pencerenin boyundan bağımsızdır.  $N$  örnek sayısını artırmak, bant genişliğini düşürür.  $N$  değeri çok küçükse, perde periyodu cinsinden, ya da dalga şeklinin ayrıntılarına daha az bağlı olarak  $E_n$ , çok hızlı dalgalanacaktır. Eğer  $N$  örnek sayısı çok fazla ise, bir kaç perde periyodu cinsinden,  $E_n$  ses sinyalinin değişken özelliklerini yansıtamayacak kadar yavaş değişecektir. Perde periyodu 10 kHz örnekleme hızında 20 ile 250 arasında değişebildiği için tek bir  $N$  değeri yeterli

olmayacaktır. 10 kHz örnekleme hızı için, uygun N değeri, 100 ile 200 arasındadır [2]. En, ötümlü/ötümsüz ses ayrımı için temel ayrıçlardan biridir. Ötümsüz seslerde, ötümlü seslere göre oldukça düşüktür. Ötümsüz-ötümlü, ötümlü-ötümsüz geçişlerinin yerini tespit etmede, yüksek kalitede sesler için, konuşma sesini ayırt edebilmede En kullanılır.

Kısa-zamanlı enerji işlevinin bir dezavantajı, yüksek sinyal seviyelerine karşı duyarlı olmasıdır. Bu problemi ortadan kaldırmanın bir yolu, ortalama genlik işlevidir.

$$M_n = \sum_{m=-\infty}^{\infty} |x(m)|w(n-m) \quad (2.25)$$

Burada, sinyalin karelerinin toplanması yerine, sinyalin mutlak değerinin ağırlıklı ortalamaları toplanmaktadır. Yukarıdaki eşitlikteki ortalama genlik hesaplamasındaki dinamik oran, (en büyüğün en küçüğe oranı) enerji hesaplamasındaki dinamik oranın yaklaşık olarak kareköküne eşittir.

Enerji ve ortalama genlik işlevinin bant genişlikleri, alçak geçiren filtreninki kadar olduğundan, ses sinyali kadar hızlı örneklenmelerine gerek yoktur. Mesela, 20 ms.lik bir pencere için, saniyede 100 örneklilik bir örnekleme hızı yeterlidir. 20 ms.lik bir pencerede;  $20 \cdot 10^{-3} \cdot 100 = 2$  örnek vardır.

Kısa zamanlı enerji ve kısa zamanlı ortalama genlik işlevlerinin özelliklerini yorumlamak için kullanılacak filtrenin, dikdörtgen filtre veya Hamming filtre gibi bir sınırlaması yoktur. Yeterince düzgünlük sağlayan etkili her filtre burada kullanılabilir. Dolayısıyla, kendimiz herhangi bir standart filtre tasarım metoduyla bir alçak geçiren filtre tasarımı yapabiliriz. Bu filtre SSDC (sonsuz darbe cevaplı) ya da SLDC (sonlu darbe cevaplı) filtre de olabilir. SLDC filtrelerin (Hamming veya dikdörtgen filtreler gibi) bir avantajı, çıkışın, düşük örnekleme hızında, hareketli pencere yöntemine göre daha kolay hesaplanabilmesidir [6].

Z-dönüşümü sonsuz uzunlukta bir pencere ile filtreleme yapılabilir. Basit bir örnek;

$$\begin{aligned} h(n) &= an & n \geq 0 \\ &= 0 & n < 0 \end{aligned} \quad (2.26)$$

$0 < a < 1$  şartı sağlanırsa, etki süresi istendiği gibi ayarlanabilen bir pencere elde edilmiş olur. Bu işlevin z-dönüşümü;

$$H(z) = \frac{1}{1 - az^{-1}} \quad |z| > |a| \quad (2.27)$$

Z dönüşümünden görülebileceği gibi,  $H(e^{j\Omega T})$  istenen alçak geçiren filtre özelliğindedir. Böyle bir filtre, basit bir fark eşitliği ile elde edilebilir. Kısa zamanlı enerji, tekrarlı formülü doğrular :

$$E_n = a \cdot E_{n-1} + x^2(n) \quad (2.28)$$

ve ortalama genlik de şu eşitliği doğrular :

$$M_n = a \cdot M_{n-1} + |x(n)| \quad (2.29)$$

$E_n$  ve  $M_n$ , daha düşük bir örnekleme hızı yeterli olmasına rağmen, giriş ses sinyalinin her örneği için hesaplanır. Eğer düşük örnekleme hızı yeterliyse, tekrarlı olmayan metotta daha az hesaplama olacaktır. Alçak geçiren filtrelerdeki diğer bir faktör de, gecikme faktörüdür. Dikdörtgen ve Hamming pencereleri, simetrik, doğrusal fazlı,  $(N-1)/2$  örneklik gecikmeli filtrelerdir.

### 2.3.3 Kısa-Zamanlı Ortalama Sıfır Geçiş Hızı

Ayrık-zaman sinyallerinde, ardarda gelen örnekler farklı işaretlere sahipse, sıfır geçişleri meydana gelir. Sıfır geçişlerinin oluşma hızı, sinyal sıklığının basit bir ölçümüdür. Bu ölçüm, dar bantlı sinyaller için doğrudur. Örnekleme hızı  $F_s$  olan,  $F_0$  sıklıklı bir sinüs dalgası, her döngüsünde  $F_s/F_0$  adet örnek içerir. Her döngü 2 adet sıfır geçişi içerdiği için, sıfır geçişlerin uzun zaman için ortalama hızı,

$$z = 2 \frac{F_0}{F_s} \frac{\text{gecis}}{\text{örnek}} \quad (2.30)$$

Dolayısıyla, sinyalin sıfır geçiş hızı, sıklığı hakkında bilgi vermektedir.

Ses sinyalleri, geniş bantlı sinyallerdir. Geniş bantlı sinyaller için sıfır geçiş hızı yeterince kesin değildir. Bununla beraber, izgesel özelliklerin kaba tahminleri, kısa-zamanlı sıfır geçiş hızları kullanılarak yapılabilir.

$$z_n = \sum_{m=-\infty}^{\infty} |\text{sgn}(x(m)) - \text{sgn}(x(m-1))| w(n-m) \quad (2.31)$$

$$\begin{aligned} \text{sgn}[x(n)] &= 1 & x(n) &\geq 0 \\ &= -1 & x(n) &< 0 \end{aligned}$$

$$\begin{aligned} \text{ve} \quad w(n) &= \frac{1}{2N} & 0 \leq n \leq N-1 \\ &= 0 & \text{diğerlerinde} \end{aligned} \quad (2.32)$$

Sinyalin sıfır geçiş hızı, genel olarak, kısa-zamanlı enerji ve kısa zamanlı ortalama genlik özelliklerine benzer.

Sıfır geçiş hızını hesaplamak için bütün gerekli olan şey, örnekleri çiftler çiftler kontrol etmek ve N adet ardarda gelen örnek için ortalama sıfır geçiş hızını hesaplamaktır.

Ses üretim modeli, ses dalgasının izgesindeki düşmeden dolayı, ötümlü seslerin enerjisinin 3 kHz'e kadar olan bölgede yoğunlaştığını ileri sürmektedir. Ötümsüz seslerde ise enerjinin çoğu, yüksek sıklıklarda bulunmaktadır. Yüksek sıklıklarda sıfır geçiş hızının fazla olması ve düşük sıklıklarda da bu hızın oldukça düşük olması, sıfır geçiş hızının ve enerji dağılımının sıklıkla oldukça ilintili olduğunu ortaya koymaktadır. Dolayısıyla şöyle bir genelleme yapılabilir; eğer sıfır geçi hızı yüksek ise ses ötümsüz, sıfır geçiş hızı düşükse, ses ötümlüdür. Aslında bu cümle oldukça belirsiz bir cümledir, çünkü, hangi seviyenin düşük ve hangi seviyenin yüksek olduğu söylenmemiştir ve bunu kesin olarak belirtmek de mümkün değildir.

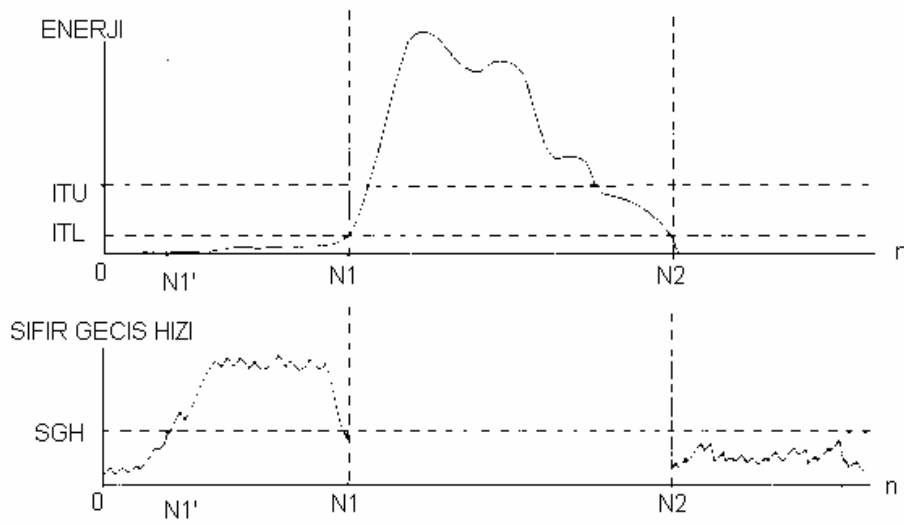
Sıfır geiş hızını hesaplamak için gerekli olan basit algoritma, sadece örnek çiftleri arasında işaret farklılıkları olup olmadığını tespit etmek olmasına karşın, örnekleme yapılırken oldukça dikkatli olmak gerekmektedir. Analogdan sayısala çeviricideki doğru akım etkisi, 60 Hz civarındaki gürültü ve sayısallaştırma sistemindeki gürültü, sıfır geiş hızını etkiler. Bu etkilerden korunmak için analog sinyal, dikkatli bir şekilde örneklenmelidir. Mesela, alçak geirgen filtreler yerine bant geirgen filtreler tercih edilmelidir. Böylelikle, sinyalin doğru akım ve 60 Hz bileşenleri yok edilebilir.

Sıfır geiş hızı ölçümü için diğere önemli noktalar ise, örnekleme periyodu  $T$  ve ortalama aralığı  $N$ 'dir. Örnekleme periyodu, sıfır geiş gösteriminin zaman veya sıklık çözünürlüğünü belirler, iyi çözünürlük için yüksek sıklık gereklidir.

#### **2.3.4 Enerji ve Sıfır Geiş Hızı Kullanılarak Ses İle Sessizliğı Ayırt Etme**

Arka plan gürültüsü içeren bir ses sinyalinde, sesin sinyalin hangi noktasından başlayıp hangi noktasında bittiğini tespit etmek, ses işleme alanlarında önemli bir problem olarak görülür. Sesi, arka plan gürültüsünden ayırmak, sinyalin gürültüye oranı düşük olduğu zaman önemlidir. Sesin gürültüye oranının yüksek olduğu durumlarda en düşük ses seviyesinin enerjisi dahi arka plan enerjisini geçtiğı için, basit bir enerji ölçümü yeterlidir.

Bir kelimenin başını ve sonunu en iyi belirleyebilecek şeyler, enerji ve sıfır-geiş hızıdır. Kullanılan temel gösterimler, 10ms'lik çerçeveler için kullanılan sıfır geiş sayıları ve hesaplanan ortalama genliktir. Sinyalin ilk 100ms'lik kısmında ses olmadığı düşünülür. Bu bölge için ortalama, standart sapma, ortalama genlik ve sıfır geiş hızı gibi hesaplamalar yapılarak geri plan gürültüsünün karakterleri tespit edilir. Bu istatistiksel değerleri ve bu aralıktaki en büyük genliğı kullanarak, sıfır geiş hızı ve enerji için eşik değerleri hesaplanır. Ortalama genlik grafiğı, daima geilen bir eşik değeri aralığını tespit etmek için incelenir. Başlama ve bitiş noktalarının, bu aralığın dışında olduğu varsayılır.



Şekil 2.4 Enerji ve Sıfır Geçiş Hızı

Algoritma, Şekil 2.4'e göre tanımlanabilir. Şekil 2.4'te ITU (eşik aralığı üst sınırı - Interval of Threshold Upper) ile gösterilir.  $M_n$ 'nin (enerji grafiğinde) ITU eşik değerini ilk geçtiği noktadan geriye doğru gidersek,  $M_n$ 'nin ITL (eşik aralığı alt sınırı)'den aşağıya ilk düştüğü nokta, başlama noktası olarak seçilir. Benzer şekilde  $N_2$  noktası da bitiş noktası olarak seçilir. Bu çift eşik değerli işlem ile, ortalama genlikteki düşüşlerin sinyal bitiş noktasını yanlış etkilememesi sağlanır. Bu aşamada, başlama ve bitiş noktalarının  $N_1$ ,  $N_2$  aralığı içinde olmadığını varsaymak daha güvenlidir. Diğer adım, sinyalin sıfır geçiş hızı ile gürültünün sıfır geçiş hızını karşılaştırarak,  $N_1$ 'i  $N_2$ 'ye doğru kaydırmaktır. Bu işlem  $N_1$ 'den itibaren 25 pencere kadar yapılır. Eğer sıfır geçiş hızı, eşik değerini 3 veya daha fazla oranda geçiyorsa  $N_1$  noktası, sinyalin sıfır geçişleri eşik değeri noktasını ilk geçtiği noktaya geri kaydırılır. Bu şartlar da sağlanmazsa  $N_1$  başlangıçtaki yerinde kalır. Aynı işlem, bitiş noktası için de tekrar edilir [6].

### 2.3.5 Kısa Zamanlı Özilinti İşlevi

Gerekirci bir ayrık sinyalin özilinti işlevi;

$$\Phi(k) = \sum_{m=-\infty}^{\infty} x(m).x(m+k) \quad (2.33)$$

Eğer sinyal rasgele veya periyodik ise özilinti işlevi;

$$\Phi(k) = \lim_{N \rightarrow \infty} \frac{1}{(2N+1)} \sum_{m=-N}^N x(m).x(m+k) \quad (2.34)$$

ile ifade edilir.

Her iki durumda da, özilinti işlevi gösterimi, sinyalin belli özelliklerini göstermek için uygun bir yoldur. Mesela, sinyal P örnek ile periyodik ise,

$$\Phi(k) = \Phi(k+P) \quad (2.35)$$

olarak gösterilebilir.

Özilinti işlevinin özellikleri;

1. Çift işlevdir.  $\Phi(k) = \Phi(-k)$
2. En büyük değerine,  $k=0$ 'da ulaşır.
3.  $|\Phi(k)| \leq \Phi(0)$  tüm  $k$ 'lar için.  $\Phi(0)$  değeri gerekirci sinyaller için enerjiye, rasgele veya periyodik sinyaller için ise ortalama güce eşittir.

Özilinti işlevi, özel bir durum olarak enerjiyi içerir. Bundan da önemlisi, periyot özelliğinin gösterildiği uygun bir yoldur. Periyodik sinyaller için özilinti işlevi,  $0, \pm P, \pm 2P \dots$  örneklerinde en büyük değere sahiptir. Sinyalin, başlangıç zamanı göz önüne alınmadan, özilinti işlevinin ilk en büyük değerlerinin aralığı, perde periyodunun değerini verir. Sesin, kısa-zamanlı özilinti işlev gösteriminin elde edilmesi için, özilinti işlevinin nasıl uyarlanabileceğinin tanımlanması oldukça önemlidir [6].



Kısa zamanlı özilinti işlevi;

$$R_n(k) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)x(m+k)w(n-k-m) \quad (2.36)$$

Bu eşitlik şöyle yazılır: ses bölümü, bir pencereyle çarpılarak seçilir. Daha sonra bu bölüme özilinti tanımı uygulanır:

$$\begin{aligned} R_n(-k) &= R_n(k) \\ R_n(k) &= R_n(-k) \\ &= \sum_{m=-\infty}^{\infty} x(m)x(m-k)[w(n-m)w(n+k-m)] \end{aligned} \quad (2.37)$$

eğer  $h_k(n)$ 'i tanımlarsak;

$$\begin{aligned} h_k(n) &= w(n).w(n+k) \\ R_n(k) &= \sum_{m=-\infty}^{\infty} x(m).x(m-k).h_k(n-m) \end{aligned} \quad \text{olur.} \quad (2.38)$$

$k$ 'inci özilinti gecikmesinin  $n$  anındaki değeri,  $x(n).x(n-k)$  dizisinin,  $h_k(n)$  darbe cevaplı bir filtreden geçirilmesiyle elde edilir.

$$R_n(k) = \sum_{m=-\infty}^{\infty} [x(n+m).w'(m)][x(n+m+k)w'(k+m)] \quad (2.39)$$

Burada  $w'(n)=w(-n)$ 'dir. Yukarıdaki eşitlik, giriş dizisinin  $n$  örnek kadar kaydırıldığını ve kısa bir ses bölümü seçmek için de,  $w'$  ile çarpıldığını gösterir. Eğer  $w'$  penceresi, sonlu süreli ise,  $x(n+m).w'(n)$  de sonlu sürelidir. Eşitlik;

$$R_n(k) = \sum_{m=0}^{N-1-k} [x(n+m)w'(m)][x(n+m+k)w'(k+m)] \quad \text{olur.} \quad (2.40)$$

Ötümsüz ses grafiğinde ise herhangi bir periyodik duruma rastlanmamaktadır. Ötümsüz seslerin özilinti işlevi, ötümsüz seslerin sinyalinde olduğu gibi, yüksek sıklıklı gürültü benzeri görünümündedir.

Önemli bir nokta, periyodikliğin elde edilmesi için N'nin nasıl seçileceğidir. Ses sinyalinin değişen özelliklerinden dolayı, N olabildiğince küçük seçilmelidir. Diğer yandan da, özilinti işlevinin periyodikliği gösterebilmesi için, seçilen pencerenin, dalganın periyodunun en az iki katı uzunluğunda olması gerekir.

$R_n(k)$ 'da sonlu uzunlukta, bir pencere işlevinden geçirilmiş ses sinyali olduğu için, k arttıkça daha az veriler arasında işlem yapılmaktadır. Yukarıdaki eşitliğin üst sınırından dolayı :  $N-1-k$  bu en büyük değerli noktalarda k arttıkça genliğin düşmesine sebep olur.

Geliştirilmiş kısa zamanlı özilinti işlevi:

$$\hat{R}_n(k) = \sum_{m=-\infty}^{\infty} x(m) \cdot w_1(n-m) \cdot x(m+k) \cdot w_2(n-m-k) \quad (2.41)$$

veya başka bir yazış ile;

$$\hat{R}_n(k) = \sum_{m=-\infty}^{\infty} x(n+m) \cdot \hat{w}_1(m) x(n+m+k) \hat{w}_2(m+k) \quad (2.42)$$

Burada;

$$\begin{aligned} \hat{w}_1(m) &= w_1(-m) \\ \hat{w}_2(m) &= w_2(-m) \end{aligned} \quad (2.43)$$

Pencere tanımlarsak :

$$\begin{aligned} \hat{w}_1(m) &= 1 & 0 \leq m \leq N-1 \\ &= 0 & \text{diğerlerinde} \\ \hat{w}_2(m) &= 1 & 0 \leq m \leq N-1+K \\ &= 0 & \text{diğerlerinde} \end{aligned} \quad (2.44)$$

K, mevcut en büyük gecikmedir.

Eşitliği tekrar düzenlersek;

$$\hat{R}_n(k) = \sum_{m=0}^{N-1} x(n+m)x(n+m+k) \quad 0 \leq k \leq K \quad (2.45)$$

Ortalama,  $N$  örnek üzerinden alınmaktadır,  $n$ 'den  $n+N-1$ 'e kadar olan örnekler de bu hesaplama dahilidir. Bu bir çapraz korelasyondur ve  $\hat{R}_n(-k) \neq \hat{R}_n(k)$  'dır.  $\hat{R}_n(k)$  'nın tepe değerleri, periyotların katlarında oluşmaktadır ve  $k$ 'nın büyük değerleri için grafikte bir düşüş görülmemektedir.

### 2.3.6 Kısa-Zamanlı Ortalama Genlik Farkı İşlevi

$P$  periyoduna sahip bir sinyalin  $d(n)=x(n)-x(n-k)$   $k=0, \pm P, \pm 2P, \dots$  için alacağı değer sıfırdır. Ses için de bunun geçerli olduğu düşünülür.

OGFİ işlevi şöyle tanımlanır :

$$x_n(k) = \sum_{m=-\infty}^{\infty} |x(n+m)w_1(m) - x(n+m-k)w_2(m-k)| \quad (2.46)$$

$x_n(k)$  işlevi,  $k=P, 2P, \dots$  değerlerinde hızlı bir düşüş gösterir.

### 2.3.7 Özilinti İşlevini Kullanarak Perde Periyodunu Kestirme

Sesin özilinti işlevinde en büyük genlik, perde periyodunda oluşur. Bu durumda genelde, seçilen pencere uzunluğu, perde periyoduna oranla daha kısadır. Ayrıca, ses yolu cevabına bağlı olan özilinti işlevinin tepe değerleri, ses uyarım periyodundan büyük ise, en yüksek tepe değerini bulma işlemi doğru sonuç vermez. Bu tür dezavantajları gidermek için, ses sinyalinin periyodikliğini ön plana çıkarıp, diğer özelliklerini bastıran işlevler kullanılmalıdır. Bu işlemi yapan izge düzelticilerin amacı, ses yolu aktarım işlevinin etkilerini ortadan kaldırmaktır.

Bunlardan biri, merkezi kırpma tekniğidir. Bir ses sinyalinin en büyük genliği  $A_{max}$  ise,  $CL$  kırpma değeri, bu değere göre sabitlenir. Sinyalin çıkışında, sinyalden  $CL$  genliğinin çıkarılmış hali elde edilir. Eğer sinyal genliği,  $CL$ 'den küçükse, çıkışta sıfır görülür.

Ses sinyalinin doğrudan özilintinin alınması ile çok yüksek tepe değerleri elde edilebilir. Burada ise, ses sinyali, önce merkezi kırpma tekniği ile indirgenmiş,

daha sonra özilintisi alınmıştır. Böylelikle karışıklığı önleyecek, periyotlarda, daha düşük değerlerde tepe noktalarının oluşması sağlanır.

Yapılan araştırmalar sonucu en büyük genliğin %30'unun CL olarak alınması önerilmiştir [6].

Perde periyodunun, kısa-zamanlı özilinti işlevinin gösteriminden bulunabileceği ileri sürülmüştür. Şöyle bir akış şeması izlenmektedir :

1. Ses 900Hz alçak geçirgen filtreden geçirilip 10kHz'de örneklenir.
2. 10ms'lik aralıklarda 30ms'lik ses bölümleri seçilir.
3. 100 örnekli dikdörtgen pencere ile ortalama genlik hesaplanır. Sinyalin tepe değeri, 50ms'lik arka plan gürültü seviyesiyle karşılaştırılır. Eğer gürültüden fazlaysa ses ötümlüdür, değilse ötümsüzdür.
4. Kırpma seviyesi CL, ses bölümünün ilk ve son 100 örneğinin en büyük genlik seviyelerinin en küçüğünün sabit bir yüzdesi olarak belirlenir. Mesela %68'i gibi.
5. Merkezi kırpma işlemi yapılarak korelasyonu alınır.
6. Özilinti işlevinin en yüksek tepe değeri bulunarak, sabit eşik değeri ile karşılaştırılır ( $R_n(0)$ 'nin %30'u). Bunun altında ise ötümsüz ses, üstünde ise ötümlü ses olarak saptanır. Bulunan nokta, perde periyodu olarak tanımlanır.

## 2.4 DOĞRUSAL ÖNGÖRÜM KODLAMASI

Gönderici, ses çıktısını analiz eder ve model parametrelerini elde eder. Bu parametreler benzer ses dalgası üretmek için sentezlemede kullanılacağı alıcıya gönderilir. Bu yaklaşımın doğru olması için iyi bir kaynak modellenmesinin yapılmış olması gerekir. Günümüzde ses üretimi için iyi modeller mevcuttur.

Hangi lisanın kullanıldığına bağlı olmaksızın her insanda ses üretimi için aynı mekanizma kullanılır. Bununla birlikte bu mekanizma, çıktıların durumunu kısıtlayan bir takım fiziksel kurallara sahiptir. Dolayısıyla ses, bir model olarak analiz edilebilir. Bu modelin parametreleri hesaplanarak alıcıya gönderilir. Alıcıda ise bu parametreler kullanılarak ses sinyali tekrar üretilir. Bu analiz/sentez yaklaşımı ilk kez Homer DUDLEY tarafından Bell laboratuvarlarında ortaya çıkmıştır. Homer DUDLEY kanal ses kodlayıcılarını da ilk geliştiren kişidir.

En basit ses sentezleme modeli Şekil 2.5’de gösterildiği gibidir.



Şekil 2.5 Ses sentezleme modeli

Göndericide ses çerçevelere ayrılır. Her bölüm için ses sinyali, uyarım sinyali ve ses yolu filtresinin parametreleri hesaplanır. Bazı prosedürlerde, uyarım sinyali için bir model alıcıya gönderilir. Uyarım sinyali daha sonra alıcıda sentezlenir ve ses yolunun uyarım sinyali olarak kullanılır. Diğer prosedürlerde ise uyarım sinyalinin kendisi analiz ve sentezde kullanılır.

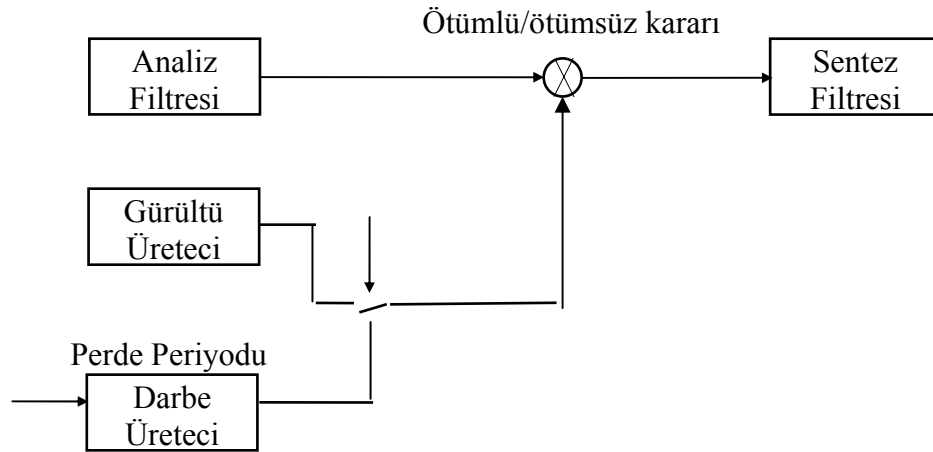
### 2.4.1 Kanal Ses Kodlayıcıları

Kanal ses kodlayıcılarında, giriş ses parçalarının herbiri bant geçiren filtre bankaları kullanılarak analiz edilir. Bu filtreler, analiz filtreleridir. Her filtrenin çıkışı sabit aralıklar için ölçülür ve alıcıya gönderilir. Sayısal kestirimlerde, enerji

kestirimi, filtre çıktısının karesinin ortalama değeri olarak ölçülebilir. Analog uygulamada ise zarf dedektörünün örneklenmiş çıktısından kestirim yapılır. Genellikle saniyede 50 kez kestirim yapılır. Filtre çıktısının kestirimine, ötümlü, ötümsüz bölgelerde karar verilir. Gönderici alıcıya aynı zamanda perde periyodu ile ilgili bilgileri de gönderir.

Alıcı ise ses yolu filtresi bant geçirgen filtre grubu ile ifade edilir. Bu filtrelere sentez filtresi denilip, analiz filtresine oldukça benzerdir.

Sesin ötümlü ve ötümsüz olma durumlarına bağlı olarak, sentez filtresinin girişine, periyodik darbe sinyali veya gürültü üretici bağlanır. Darbe sinyalinin periyodu, analiz bölümünde tahmin edilen perde periyodudur. Giriş, tahmin edilen enerji değerine göre ölçeklendirilir. Sentez blok şeması Şekil 2.6'da gösterilmiştir.



Şekil 2.6 Kanal Ses Kodlayıcısı

İlk dört formant ses analiz ve sentezinde yeterlidir. Her formantın bant genişlikleri tahmin edilir. Alıcıda uyarım sinyalleri, formant sıklıkları ve bant genişlikleri ayarı yapan filtrelerden geçirilir.

Uyarım sinyalinin öneminin anlaşılması ses kodlamada önemli bir gelişme olmuştur. Ötüm bilgisine böylece daha çok önem verilmeye başlanmıştır. Ötümlü olarak uyarılacak kanal ses kodlayıcısında, ses önce dar bantlı alçak geçirgen

filtreden geçirilir. Alçak geçirgen filtrenin çıkışı örneklenip alıcıya gönderilir. Alıcıda, bu sinyalin, yüksek dereceli harmoniklerini üretilerek, uyarım sinyali olarak kullanılır. Ötümlü uyarımda, hem perde periyodu kestirimi hem de ötümlü/ötümsüz ayrımı yapılmaz. Ötümlü veya ötümsüz ayrımı kesin olarak yapılamayan bölümler oldukça az olduğundan, kalite oldukça iyidir.

#### **2.4.2 Ötümlü/Ötümsüz Kararı**

Ötümlü seslerin örneklerinin büyüklüğü, ötümsüz seslere göre daha fazladır. Dolayısıyla, ötümlü seslerin enerjisi büyüktür. Ötümsüz sesler yüksek sıklık bileşenlerini içerirler. Her iki grup sesin de ortalaması sıfır civarındadır. Sıklık bileşenlerinden dolayı ötümsüz sesler zaman eksenini üzerinden ötümlülere göre çok daha fazla geçiş yaparlar. Bu özellikleri kullanılarak, ötümlü ve ötümsüz sesler ayırt edilebilir.

## 2.5 SESLİ İFADE TANIMA

Ses, doğadaki canlılar arasında kullanılan en temel iletişim araçlarından biridir. Sesin insanlar arasında iletişim amacıyla kullanılmasıyla konuşma ortaya çıkmıştır. Dolayısıyla konuşma insanların sahip olduğu en temel iletişim araçlarından biri olmuştur. Başka bir ifadeyle, konuşma insanlar arasında bilgi taşıyan ve bunun için başka bir aracı gerektirmeyen en önemli iletişim aracıdır.

Sesli ifade tanıma insan-bilgisayar arası iletişim için önemli yararlar sağlamaktadır. Sesli ifade verilerinin elde edilmesi çok kolaydır. Klavye veya diğer veri giriş yöntemlerini kullanmak için olduğu gibi özel bir yetenek gerektirmez. Sesli ifade kullanarak metinlerin elektronik ortama yazı olarak aktarılması, el yazısından ve en hızlı klavye kullanan birinin yazmasıyla klavye girişinden oldukça hızlı olabilmektedir. Kullanıcıya hareket serbestliği ve ellerini kullanabilme olanağı sağlamaktadır. Bir mikrofon yada telefon veri giriş aracı olarak kullanılabilir. Bu nedenle veri girişi ekonomik olmakta ve uzak mesafelerden telefon aracılığıyla veri girişi mümkün olmaktadır.

Konuşma sesi, dil bilgisini, konuşmacının özelliklerini, konuşmacının duygularını taşıyabilir. Konuşma iletişim yaşamımızda çok önemli bir rol oynamaktadır. Konuşmanın ses içeriği ve dilsel içeriği insanın kültürel yapısına göre değişiklik göstermektedir.

Konuşmanın temeli içerdiği sesli ifadelerdir. Ancak sesli ifadenin etki sınırları ortamla kısıtlı olduğundan sesli ifadenin elektriksel sinyallere dönüştürülmesi gereklidir. Bu amaca yönelik ilk gerçekleştirim telefon ile sağlanmıştır. Telefon, seslerin elektriksel sinyallere dönüştürülmesine dayanır. Telefonun icadı iletişimin tarihsel gelişimi içinde en önemli adımdır. Sesli ifade üzerinde araştırmalar telefonun icadı ile başlamıştır. Bu araştırmalar sonucunda ilk sesli ifade kodlama tekniği olarak 1938'de Genlik kodlaması modülasyon (PCM, Pulse Code Modulation) tekniği bulunmuştur. Bu yöntemle sesli ifade örneksel olarak kodlanmaktadır. [20].



1960'dan sonra sayısal elektronik devrelerin ve bilgisayarların gelişmesiyle sesli ifadenin sayısal olarak kodlanması da mümkün olabilmiştir. Sesli ifade tanıma, bir sesli ifade sinyalinin sınırlı sayıda simgelerle ifade edilmesi olarak tanımlanabilir. Amaç ses sinyalinin yazılı eşdeğerinin bulunmasıdır. Sesli ifadenin yazılı karşılığının bulunması insan-bilgisayar arası iletişimi çok daha kolay bir hale getireceğinden, bilgisayarın daha kullanışlı bir duruma gelmesini sağlar. Yani insan-bilgisayar arası iletişim doğal bir olay haline gelir. Bu nedenle, bu konu çok rağbet görmüştür.

Sesli ifade tanıma insan-bilgisayar arası iletişim için önemli yararlar sağlamaktadır. Sesli ifade verilerinin elde edilmesi çok kolaydır. Klavye kullanma, program kullanıcı ara yüzlerindeki butonları tıklayarak veri girme gibi işlemlerde olduğu gibi özel bir yetenek gerektirmez. Sesli ifade kullanarak metinlerin elektronik ortama yazı olarak aktarılması, el yazısından 8-10 kat, en hızlı klavye kullanan birine göre 4-5 kat daha hızlı olabilmektedir. Kullanıcı hareket halindeyken yada ellerini kullanması gereken bir iş yaparken aynı anda metin girişi yapabilir. Bir mikrofon yada bir telefon veri giriş aracı olarak kullanılabilirdiği için veri girişi ekonomik olmakta, uzak mesafeden telefon aracılığıyla veri girişi mümkün olmaktadır.

Sağladığı bu olanaklara karşın sesli ifade tanımada bazı zorluklar da vardır. Bu zorluklar sesli ifade tanımının kullanım amaçlarına göre ve kullanılan yöntemlere göre değişiklik göstermektedir. Giriş verisi yazılı olmadığından bazı gürültülerden çok kolay etkilenebilir. Bu nedenle gürültü filtreleme yöntemleri kullanılarak sesli ifade gürültüden arındırılmalıdır.

Sesli iletişim süreci;

- Konuşmacıda oluşan bir düşüncenin sözcüklere dönüştürülmesi,
- Ses telleri ve konuşma sistemi aracılığıyla sözcüklerin seslendirilmesi,
- Seslerin titreşimler halinde dinleyicinin kulağına ulaşması
- Dinleyicinin işitsel sınırları aracılığıyla sesin beyne iletilmesi ve beynin bu titreşimleri dilsel karşılıklarına çevirmesi,

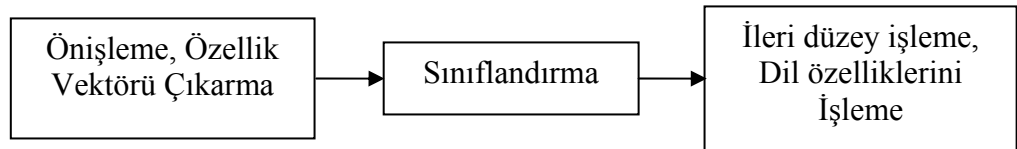
- Dilsel kodların, yani sözcüklerin yan yana getirilerek anlam çıkarılması biçiminde özetlenebilir.

İnsanlar yapay olarak üretilen sesleri, makinelerin insan konuşmalarını anlamasından daha etkin bir şekilde anlamaktadır. Bunun sebebi insanların ses konusunda sahip olduğu deneyim ve anlayış olarak gösterilebilir. Bu deneyim insanların yanlış olan sesleri daha rahat tolere edebilmesini sağlamaktadır. İnsanlar sahip oldukları dil bilgisini ve zamanla kazandıkları, değişik konuşmacıların konuşmalarını anlayabilme yeteneklerini kullanarak yanlış sesleri düzeltebilmektedir.

Yapay bir sesli ifade tanıma sisteminin amacı dinleyicinin yerini almaktır. Amaç her ne kadar dinleyicinin yerini almak ise de insan kulağının ve beyninin sunduğu esnekliğe ulaşmak yapay bir sistem için çok zordur. Bu sebeple sesli ifade tanıma sistemleri bazı sınırlamalar getirmek zorunda kalmaktadır. Örneğin sözcük bazında tanıma yapan bir sistem için kelime sayısı bir sınır olarak konulmaktadır. Tanımadaki başarıyı artırmak için süreç parçalar halinde ele alınmakta ve araştırmalar bu parçalar üzerinde yoğunlaştırılmaktadır. Sürecin parçalanması yaklaşımı, her bir parça için daha kolay başarıya ulaşmayı sağlamaktadır. Bu yaklaşımla sistemin tüm başarıyı artmaktadır.

Sesli ifade tanıma, konuşma sinyallerinden otomatik olarak dilsel bilgilerin çıkarılmasıdır. Sesli ifadede en önemli öge olan dilsel bilgi fonetik bilgi olarak da adlandırılır.

Klasik bir sesli ifade tanıma sistemi Şekil 2.7'deki gibi modellenenir.



Şekil 2.7 Sesli İfade Tanıma Süreci

Bu modelde blok sınırlarında tanıma süreci içinde deęişiklik yapılmaz, fakat bloklar arasında sık sık geri dönüşler gereklidir. Bu geri dönüşler sistemin daha esnek ve daha tutarlı olmasını sağlamak içindir.

Sesli ifade tanıma sistemlerinin çalışma ilkesi kabaca, giriş verisinin daha önce kaydedilmiş şablonlarla karşılaştırılmasına dayanır. Bu şablonlar fonem düzeyinde ya da sözcük düzeyinde düzenlenmiş olabilir. Bu karşılaştırma sonucunda verinin en çok benzediği şablon o verinin simgesel gösterimi olarak kabul edilir. Ses sinyallerinin doğrudan karşılaştırılması çok zordur. Sesin şiddeti farklı olabileceğinden dolayı ses sinyalleri üzerinde bir ön işlem yapılması gerekmektedir. Bu ön işlem özellik çıkarımı olarak adlandırılır. Alınan ses verisinden kısa süreli özellik vektörleri çıkarılarak daha önce sınıflandırılmış olan şablonlarla karşılaştırma yapılır. Burada elde edilen özellik vektörlerinin ses verilerini en iyi şekilde temsil etmesi, islenebilir oran ve büyüklükte olması ve ayırt edici özelliklere sahip olması gerekmektedir.

Doğal seslerin oldukça karmaşık bir yapısı vardır. Bu nedenle sesli ifade tanıma sistemlerinde bir çok kısıtlayıcı kullanılmakta ve ancak kısmi gerçekleştirmeler yapılabilmektedir. Bu kısıtlayıcılar, sistemin kişiye bağımlı olması, sürekli ya da kesikli sesli ifadeler kullanma gibi örneklenebilir.

Sesli ifade tanımada karşılaşılan en önemli sorun sesli ifadenin devingen yapısıdır. İçerik olarak aynı olan bir sesli ifade farklı kişiler tarafından seslendirildiğinde önemli farklılıklar göstermektedir. Bunun yanı sıra sesli ifadelerde sözcüklerin bazı kesimlerinde yapılan vurgular, sesin şiddeti, tizlik derecesi çok büyük farklılıklar göstermektedir. Sesli ifade genel olarak durağan olmayan bir özelliğe sahiptir. İçerik olarak aynı olan bir sesli ifadenin aynı kişi tarafından bile ayrı biçimde seslendirilmesi mümkündür. Bu farklılıklar sebebiyle sesli ifadede fonemler arasındaki geçimler durağan olan kesimlerden daha fazla bilgi taşır.

Özellik vektörleri çıkarılırken seslerin çok iyi bir biçimde ayırt edilmesi amaçlanmalıdır. Temel olarak sesli ifadenin yazılı ifadeye benzediği söylenebilir.

Sesli ifade ardı ardına gelen seslerden oluşmaktadır. Yazılı ifadede bu kesimler yerini dilin simgelerine yani harflere bırakmaktadır. Sesli ifadenin simgesel gösterimindeki en küçük birim fonem(phoneme) olarak adlandırılır. Fonemin tek basına bir anlamı yoktur. Fakat sözcükleri ayıkmaya yardımcı olur. Fonemlerin ayırıcı özellikleri aşağıdaki gibi örneklenebilir.

KAN KİN

KULE KALE

FULAR YULAR

Bu örnekte KAN ve KİN sözcüklerinin ayırt edilmesi A ve İ simgeleri ile sağlanmaktadır. Bu simgeler anlam farklılaşmasına yol açtıkları için fonem olarak adlandırılır. Fonem sesli ifade için kullanılan bir birim olduğundan harfle aynı değildir. Türkçe fonem tabanlı bir dildir. Bu sebeple Türkçe için her harfin bir fonem'e karşılık geldiğini söyleyebiliriz, fakat diğer dillerde durum farklıdır. Örneğin; İngilizce'de 26 tane harf olmasına rağmen 40 kadar fonem vardır. Bir fonemin seslendirilmesinde kullanılan ses birimine fon (phone) denir. Fon, fonemin gerçekleştirilmesinde kullanılır, ayırt ediciliği yoktur. Bir fonemin farklı şekillerde seslendirilmesi mümkündür. Aynı fonemin seslendirilmesiyle oluşan farklı fonlara allophone denir. Sesli ifade tanıma, bu durumda, fonların fonemlere dönüştürülmesi olarak da tanımlanabilir. Böyle bir dönüştürme sonucu sesli ifadenin fonetik gösterimi yapılmış olur. Bu fonetik gösterimden sözcük ve cümlelere geçim bir dizi dilbilgisi kuralının uygulanmasıyla sağlanabilir. Fonların fonemlere dönüştürülmesi ve tanıma işleminin buna dayandırılmasına fonem tabanlı sesli ifade tanıma denir. Bunun dışında tanıma sırasında seçilen birimler diphone (iki foneme), triphone (3 fonem), hece ve sözcüğün tamamı da olabilir. Seçilen birimin daha büyük alınması gereken işlem miktarını artırır fakat daha büyük bir doğrulukla tanıma sağlanır. Yani seçilen birimin büyüklüğü arttıkça tanımadaki başarı artarken sistemin işlem yükü de gittikçe artar. Buradan çıkarılabilecek başka bir sonuç ise tanınacak birimin uzunluğu arttıkça sisteme bazı sınırlar koyma gereğidir. Örneğin sözcük tabanlı bir sesli ifade tanıma sisteminde sözcük sayısı sınırlı tutulmak zorundadır. Fakat fonem tabanlı bir sesli ifade tanıma sisteminde sözcük sayısı sınırsızdır. Fonem tabanlı bir sistemde

sözcük sınırı yoktur. Ayrıca dilden bağımsız olarak da düşünülebilir. Yeni bir dil için tek yapılması gereken şablon fonemlerin değiştirilmesidir. Bu sistemin başarısı sinyalden simgeye dönüştürmede sağlanan başarıya bağlıdır. Fonem tabanlı bir sesli ifade tanıma sisteminin karşı karşıya bulunduğu önemli bir konu da fonemlerin ardı ardına sıralanması sırasında ortaya çıkan geçimlerdir. Bu geçimler sırasında insanın gırtlak yapısının bir sonucu olarak bir fonemin seslendirilmesi, bir sonraki fonem başladığında hala bitmemiş olur. Bu sebeple kesin bir sınır konulamaz. Bu durum sesli ifadenin kesimlenmesinde bir dezavantaj oluşturmaktadır. Yani sesli ifadenin fonemlere ayrılması sırasında fonem sınırlarının belirlenmesinde zorluklar görülür. Sesli ifade tanımada karşılaşılan bir diğer sorun da sözcüklerin seslendirilişindeki farklılıklardır.

Sözcüklerin seslendiriliş biçimi kişiden kişiye değişebileceği gibi bir kişinin konuşması da konuşmanın içeriğine, kişinin içinde bulunduğu ortamın gürültü düzeyine göre değişiklikler göstermektedir. Sesli ifadede dilbilgisi kurallarına tam olarak uyulmaması bir diğer sorundur. İnsanlar arası karşılıklı konuşmada dilbilgisi kurallarına göre bir sonraki sözcüğü tahmin etmek mümkün olabilmektedir. [3]

### **2.5.1 Sesli İfade Tanıma Sistemlerinin Sınıflandırılması**

Sesli ifade tanıma sistemlerini farklı seviyelerde sınıflandırmak mümkündür olmaktadır [7].

Sesli ifadenin sürekliliğine göre ;

- Ayrışık sesli ifade tanıma sistemlerinde, kısa aralıklarla seslendirilen sözcüklerin tanınması, (Isolated Word Recognition)
- Sürekli sesli ifade tanıma sistemlerinde, ara verilmeden seslendirilen sözcüklerin tanınması amaçlanır. (Continuous Speech Recognition)

Sürekli sesli ifade tanıma sistemleri, bağlı sözcük tanıma sistemleri (Connected Word Recognition), ve karşılıklı konuşma tanıma sistemleri

(Conversational Speech Recognition) olmak üzere iki gruba ayrılabilir. Bunlardan ilki tanıma işlemini sözcük bazında yapmayı hedeflerken, ikincisi cümlenin anlamının da anlaşılmasını hedefler. Bu sebeple karşılıklı konuşma tanıma sistemleri sesli ifade anlama sistemleri (speech understanding) olarak da adlandırılabilir ve karmaşık dilbilgisi kurallarının da sistemde yer almasını gerektirir. Bu alan Doğal Dil İşleme ya da Anlama (Natural Language Processing/Understanding-NLP-) olarak adlandırılmaktadır.

Sürekli ya da ayrışık olmalarının dışında sesli ifade tanıma sistemleri konuşmacıya bağımlılığına göre de ikiye ayrılır.

- Kişiyeye bağımlı sesli ifade tanıma sistemleri, (Speaker Dependent)
- Kişiden bağımsız sesli ifade tanıma sistemleri.(Speaker Independent)

Bunlardan ilki tek bir kişi için referans şablonlarının oluşturulmasını öngörür. Yeni kişilerin konuşmalarının tanınabilmesi için referans alınan şablonların günlenmesi gerekir. İkincisinde ise sistem herhangi bir kişi tarafından seslendirilen bir sesli ifadeyi tanıyabilir. Doğal olarak bir sistemin kullanılış alanını artırmak için amaç kişiden bağımsız bir sesli ifade tanıma sistemi olmalıdır. Fakat bunu başarmak kişiyeye bağımlı bir sistem geliştirmekten daha zordur. Bir diğer sınıflandırma da yukarıda söz konusu edildiği gibi tanıma için seçilen birimin büyüklüğüne göre yapılabilir. Bunun için de sesli ifade tanıma sistemlerini ikiye ayırmak mümkündür.

- Sözcük tabanlı sesli ifade tanıma sistemleri, tanıma için öngörülen en küçük birim olarak sözcüğün kabul edildiği sistemlerdir. (word based speech recognition)
- Fonem tabanlı sesli ifade tanıma sistemleri, tanıma için fonemlerin en küçük birim olarak kabul edildiği sistemlerdir (phoneme based speech recognition).

Bu sistemlerden ilkinin doğruluk derecesi daha yüksek olmaktadır. Çünkü fonemler arası geçimlerin olumsuz etkileri burada gözlenmez. Ancak, bunun yanında sürekli sesli ifade tanıma söz konusu olduğunda sözcükler arası geçimler yine sorun olacaktır.

Ayrıca sözcük tabanlı sesli ifade tanıma sistemlerinde referans şablonu olarak sözcüğün tamamı alındığından ve bir dilde çok sayıda sözcük olduğundan sistemin gereksinim duyduğu isleyici zamanı ve bellek gereksinimi çok daha fazla olacaktır. Fonem tabanlı tanımda ise doğruluk yüzdesi bir miktar düşerken, çok az olan fonem sayısı, hızlı sonuç üretme olanağı sayesinde, hata azaltma amaçlı geri dönüşleri mümkün hale getirmektedir. Sözcük ve fonem tabanlı sistemlerden başka bu ikisi arasında sözcük altı birimleri temel alan sesli ifade tanıma sistemlerini de söz konusu etmek mümkündür.

### 2.5.2 Sesli İfade Tanıma Teknikleri

Sesli ifade tanımanın ilk aşaması, sesli ifadenin sayısallaştırılmasıdır (Analog/Digital Conversion). Bu asamadan sonra sesli ifadenin işlenmesi mümkün olmaktadır. Sesli ifadeyi oluşturan ses dalgasının, sıklık(frequency) ve genlik(amplitude) özellikleri vardır. Genlik sesin şiddetini, taşıdığı enerjiyi belirler. Sıklık ise sesin tizlik ve peslik özelliklerini belirler. Sesli ifadenin, farklı sıklıkta çok sayıda sinüsel sinyalin üst üste binmiş biçimi olduğu söylenebilir. Sesli ifade tanıma sistemleri de genelde sesli ifade sinyallerinin analizinden, ayrıştırılmasından yararlanmaktadır. Sesli ifadenin içerdiği seslerdeki sıklık ve genlik değerleri her bir fonem için farklılık göstermektedir. Tanımanın temelinde bu ayırtıcı özellikleri bulmak ve sınıflandırmak yatmaktadır. insanın duyabileceği ses için bazı alt ve üst sınırlar vardır. İnsan kulağı kabaca 20 Hz ile 20 KHz sıklıklar arasındaki seslere duyarlıdır. Bu aralığın dışında kalan sesler algılanamaz. Genlik için de bir sınır söz konusudur. İnsan kulağı en az 20db şiddetindeki sesleri duyabilir.

120db değerinden daha fazla şiddetteki sesler ise insan kullarında hasara yol açmaktadır. Ses dalgası periyodik bir dalgadır. Periyodik bir dalgayı bir dizi sinüsel dalganın birleşimi şeklinde ifade etmek mümkündür. Sesli ifade için de bu durum geçerlidir. Periyodik bir sinyali oluşturan sinüsel dalgalar harmonim olarak adlandırılır. Bu dalgalardan periyodik sinyalin sıklığıyla aynı sıklığa sahip olanına temel harmonim veya temel sıklık (Fundamentals Frequency) denir. Sesli ifade

tanıma sistemlerinin çoğu sesli ifadenin harmoniklerinden yararlanmaktadır. Temel harmonik ise sesli ifadedeki sözcüklerin sınırlarını bulmak için sıkça kullanılan bir yöntemdir. Sesli ifade tanıma teknikleri kapsamında, sesli ifadenin farklı aşamalarında kullanılan yöntemlerin ayrı ele alınması gerekir.

### 2.5.2.1 Önişleme ve Özellik Vektörü Çıkarma Teknikleri

Özellik vektörleri çıkarma, önişleme aşamasından sonra gelir. Farklı özellik vektörü çıkarma yöntemleri için farklı önişleme aşamaları söz konusu olabilir. Burada önce önişleme teknikleri anlatılacaktır. Önişleminin ilk aşaması sesli ifade sinyallerinin sayısallaştırılmasıdır. Bu işlem ses sinyalinin temel özellikleri korunarak yapılmalıdır. Bunun için örnekleme işlemi, dönüştürülecek ses sinyalinin içerdiği en yüksek sıklıktaki harmonicin en az iki katı sıklıkta yapılmalıdır. Örnekleme işleminden sonra sesli ifadenin sayısal olarak kodlanması gerekir. Kodlama yöntemleri [13]'da;

- PCM-Pulse Code Modulation.
- Log-PCM-Logarithmic Pulse Code Modulation
- APCM-Adaptive Pulse Code Modulation
- DPCM-Differential Pulse Code Modulation
- ADPCM-Adapted Differential Pulse Code Modulation
- DM-Delta Modulation

olarak belirlenmiştir. Sayısallaştırma aşamasından sonra sesli ifade verileri üzerinde bazı ön işlemler uygulanarak özellik vektörlerinin ayılcı özellikleri artırılmaya çalışılır. Bu yöntemlerin başlıcaları, pencereleme, filtreleme, sıfır noktasını geççe sayısı hesaplama, enerji hesaplama, çenter clipping işlemleridir. Pencereleme işlemi, özellik vektörleri çıkarımı sırasında kullanılan parametre hesaplamalarında kullanılacak sınırlı uzunluktaki sesli ifade verileri üzerinde uygulanır. Bu verilerin doğrudan bir parametre hesaplama işlemine giriş verisi olarak verilmesi, dikdörtgen pencere kullanımı olarak tanımlanır. Bu pencereleme yönteminin dışında verilerin işlenmeden önce bir takım katsayılarla çarpılması da söz



konusudur. Bu katsayıların hesaplanması pencerelere yöntemi ile sağlanır. Sesli ifade verilerinin ön işleme aşamasında kullanılan belli başlı pencerelere yöntemleri aşağıda yer almaktadır.

- Rectangular Window
- Barlett Window
- Hanning Window
- Hamming Window
- Blackman Window
- Kaiser Window.

Burada belirtilen pencerelere yöntemlerinden pratikte en çok kullanılanları rectangular ve hamming pencereleridir. Hamming pencerelere yöntemine ilerde değinilecektir. Filtreleme işlemi, sinyallerin içerdiği istenmeyen sıklıkların sinyalden ayrılmasının sağlanmasıdır. Sıfırı geççe sayıları sinyalin sıklık değişimlerinin gözlenmesi, enerji hesaplama ise sesli ifade verilerinden sesli ifadenin bazı özelliklerinin önceden belirlenmesi için kullanılan yöntemlerdir. Bu yöntemlerin sesli ifade kesimleme için kullanılışı ilerde anlatılacaktır. Center Clipping işlemi sesli ifadenin gürültüden ayrılması için kullanılan bir diğer yöntemdir. Bu yöntem gürültü sinyallerinin sıklıklarının yüksek olduğu ve genliklerinin düşük olduğu ilkesi ile birlikte etkili olmaktadır. Sinyaldeki sıfır noktasına yakın kesimlerin sinyalden çıkarılması biçiminde kullanılır. Sesli ifade tanımanın ilk aşamasında kullanılan tekniklerin amacı sınıflandırma için özellik vektörleri çıkarmaktır. Amaç çok fazla olan sesli ifade verilerinin, sesli ifadeyi tanımlayabilecek vektörlere indirgenmesi ve bir sonraki asama olan sınıflandırma için kullanışlı veriler üretmektir. Özellik vektörü çıkarma için kullanılan yöntemler genel olarak iki grupta incelenir. Bunlar parametrik ve parametrik olmayan yaklaşımlardır.

### 2.5.2.1.1 Parametrik Yaklaşım

Sesli ifadenin üretiliş mekanizmasının tahmin edilmesine yönelik bir modeldir. Bir sesli ifade üretim sistemi öngörülür. Bu yöntemde giriş (kesin olarak bilinmez fakat tahmin edilir), ve çıkış (sesli ifadenin kendisi) arasında bir sesli ifade üretim fonksiyonu oluşturulur. Bu fonksiyonun parametreleri sesli ifade tanıma sisteminde özellik vektörü olarak kullanılır. Parametrik teknikler böyle bir modelin (ARMA-Autoregressive, moving average) varlığını kabul edip modeli tahmin etme temeline dayanır. Doğrusal tahmin (Linear Prediction- LP) bu model oluşturma yöntemlerinin bir alt kümesidir [7].

### 2.5.2.1.2 Parametrik Olmayan Yaklaşım

Parametrik olmayan yöntemler, sesli ifade sinyali üzerinde pencereler halinde ilerleyerek sinyal üzerinde bazı dönüşümlerin uygulanması temeline dayanır. Yöntemin başarısı, kullanılan pencerenin türünün ve uzunluğunun üzerinde yorum yapılabilecek nitelikte olmasına bağlıdır. Bu niteliğe sahip bir pencere türü ve uzunluğu için ayarlamalar yapmak bu tekniğin ilk aşamasını oluşturur. Pencere üzerinde daha sonra bir boyut dönüştürme işlemi yapılır. Örneğin Fourier dönüşümü ile genlik-zaman boyutu, sıklık zaman boyutuna dönüştürülür. Daha sonra bu dönüşüm sonucu elde edilen veriler bazı iyileştirme yöntemleriyle sınıflandırma aşamasına hazır hale getirilir.

### 2.5.2.2 Özellik Vektörlerinin Sınıflandırması Teknikleri

Önişleme ve özellik vektörü çıkarma yöntemleri sonucu ortaya çıkan ve sesli ifadeyi tanımlayıcı özelliği olan özellik vektörleri sınıflandırma kesiminde sesli ifade için şablon oluşturmak için kullanılır. Sınıflandırma için kullanılan tekniklerin dayandığı ortak kesim, sesli ifade ile referans alınan şablonların benzerliğin hesaplanmasıdır. Bunun için spektral çözümleme kullanılır. Spektral çözümleme genelde aşağıdaki beş yöntemle yapılır.

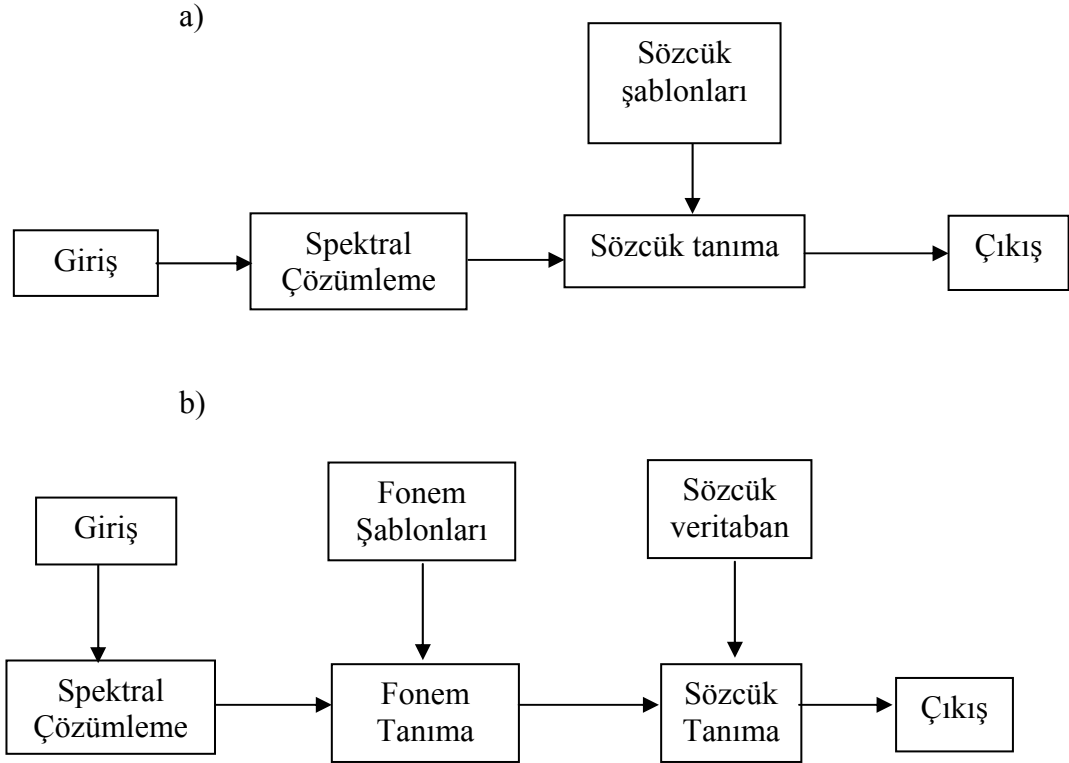
1. Band-pass filtrelerin kullanımı
2. FFT (Fast Fourier Transformation)
3. Cepstral katsayıların kullanılması.
4. Autocorrelation fonksiyonlarının kullanılması
5. LPC çözümlemesinden yararlanarak spektral zarf oluşturmak (en yüksek benzerlik tahmini)

Bu yöntemler bölüm 3.2’de anlatılmaktadır. Birçok uzaklık ölçümü bu yöntemlerle elde edilen kısa süreli spektral değerler ile yapılır.  $x$  ve  $y$  vektörleri arasındaki uzaklık  $d(x,y)$  biçiminde tanımlandığında, sesli ifade tanıma bağlamında aşağıdaki eşitliklerden söz edilebilir.

a. Simetri:  $d(x,y)=d(y,x)$

b. Pozitif tanımlılık:  $d(x,y)>0, x \neq y$   
 $d(x,y)=0, x=y$

Sınıflandırmadaki rolünden dolayı bu uzaklığın hesaplanması önem kazanmaktadır. Bu sebeple bu uzaklığı formüle etmek için etkili algoritmalar gerekmektedir. Her ne kadar standart öklid uzaklığı uzaklık hesaplamada kullanılıyorsa da bu mantıkta değişiklikler yapmak denenmektedir. İşitme duyarlılığına bağlı ağırlıklı uzaklıklar, istatistiksel çözümlemelere ve temel bileşen çözümlemesine dayanan indirgenmiş çok boyutlu uzay kullanımı bu denemelerden bazılarıdır. Sesli ifadenin önemli bazı özelliklerini taşıyan temel harmonikler burada çok seyrek kullanılmaktadır. Çünkü hesaplanmaları ve otomatik olarak çıkarılmaları çok zor olmaktadır. Burada belirtilen uzaklık ölçme yöntemleri kullanılarak gerçekleştirilebilecek bir sözcük tanıma sisteminin genel yapısı Şekil 2.8’de verilmiştir.



Şekil 2.8 Sözcük tanıma sisteminin yapısı:

(a) Sözcük tabanlı sesli ifade tanıma;

(b) Fonem tabanlı sesli ifade tanıma [7].

Şekil 2.8'deki sözcük tanıma sistemlerinin ikisinde de amaç tüm sözcüğün tanınmasıdır.

Birinci yaklaşımda, sesli ifadeden doğrudan sözcük tanımaya geçilirken ikinci yaklaşımda önce fonem tanıma işlemi yapılmakta, daha sonra bir sözcük veritabanı kullanılarak bu fonemler sözcüklere dönüştürülmektedir. Daha önce de söz konusu edildiği gibi (a) seklindeki sistemin doğruluğu daha fazla fakat tanıyabildiği sözcük sayısı az olacaktır. (b) seklindeki sistemde ise sözcük sayısı için başlangıçta bir sınır yoktur. Bir sözcük veritabanına sözcük eklemek doğal olarak o sözcük için sesli ifade şablonunun sisteme öğretilmesinden daha kolay olduğu için (b) sekli genişlemeye daha elverişlidir. Uzaklık ölçüleri her ne kadar ortak olsa bile

sesli ifade özellik vektörlerinin sınıflandırılmasında bir çok farklı yöntem kullanılmaktadır. Bunlar;

- Hidden Markov Model(HMM)
  - Time Warping-Dynamic Time Warping(DTW)
  - Nöron Ağları(Neural Network-NN)
- yaklaşımlarıdır.

### 2.5.3 Özellik Vektörlerinin Çıkarılması

Özellik vektörü, sesli ifadenin bilgi taşıyan en küçük birim zamanını temsil eden ve boyut sayısı genellikle 15-25 arasında değişen bir vektördür. Özellik vektörlerini genel olarak ikiye ayırmak mümkündür. Bunlardan ilki kısa süreli özellik vektörleridir. Bunlar sesli ifade tanınmanın temelleri olarak kabul edilebilirler. Diğeri ise uzun süreli özellik vektörleridir. Bunlar içerik vektörleri olarak da adlandırılırlar ve ardı ardına gelen birçok kısa süreli özellik vektörünün birleşiminden oluşurlar. Özellik vektörleri için geniş bir tanımlama 1993 yılında Picone tarafından yapılmıştır [21].

#### 2.5.3.1 Kısa Süreli Özellik Vektörleri

Kısa süreli özellik vektörleri sesli ifadenin kısa bir süresinde alınan veriler için hesaplanan vektörlerdir. Burada kısa süreli özellik vektörlerinin türleri açıklanacaktır.

##### 2.5.3.1.1 Ham Veriler

Sesli ifade önce sayısal olarak kaydedilir. Kaydedilmiş bu sesli ifadenin içeriği belirli bir süreyle olduğu gibi özellik vektörü olarak kullanılır. Ham verilerin daha iyi sonuç vermesi için üzerlerinde bazı işlemlerin yapılması gerekir. Örnek bir kullanım aşağıdaki gibi olabilir. Sesli ifade 16 bit genlik düzeyi ve 16 kHz sıklığında örneklenerek, sayısal ortama kaydedilir. Yüksek frekanslar basit bir birinci derece

filtre ile iyileştirilir. Bu filtrenin transfer fonksiyonu  $H(z)=1-cz^{-1}$  olabilir. Buradaki  $c$  parametresi iyileştirme katsayısı olarak tanımlanabilir. Örnek olarak  $c=0.95$  verilebilir. Burada basit bir  $z$  dönüşümü yapılmaktadır.  $c$  katsayısı sıfır noktasının  $z$ -düzlemindeki konumunu belirlemek için kullanılmaktadır. Zaman ekseninde filtrenin karşılığı (Es. 2.26)'da belirtilmiştir.

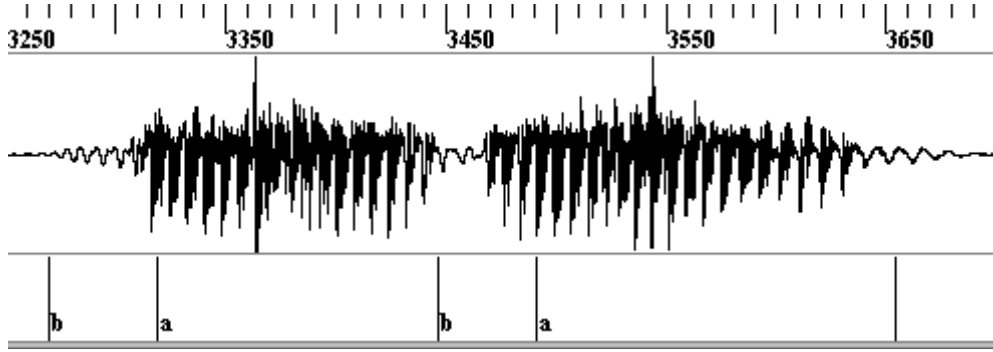
$$y(n) = x(n) - cx(n-1) \quad (2.26)$$

$y(n)$  filtreden çıkan sinyali temsil etmektedir.  $x(n)$  ise giriş sinyalini temsil etmektedir. Sonuçta oluşan veri çerçeveler şeklinde alınır ve Hamming penceresinden geçirilir. Bu pencerelere yöntemi bir sinyalin merkezinin daha belirgin hale getirilmesini sağlar. Pencerelere fonksiyonu (Es. 2.27) ile tanımlanır.

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1 \quad (2.27)$$

Burada  $N$  pencerelere yönteminin uygulanacağı çerçevenin uzunluğunu belirtir. Çerçeve dışında fonksiyonun değeri sıfırdır.  $n$  ise çerçevenin sırasıyla tüm değerlerinin indisini belirtir. Çerçevenin tüm değerleri bu fonksiyonun ürettiği değerlerle çarpıldıktan sonra orta kesiminde bulunan değerler baskınlaştırılmış olur. Çerçeve bu şekilde ağırlıklandırıldıktan sonra özellik vektörü olarak kullanılabilir. Bir sonraki çerçeve ilk çerçevenin tam ortasından baslarsa yani çerçeveler bir miktar (burada yarı yarıya) örtüşürse özellik vektörünün taşıdığı bilgi daha fazla olmaktadır. Böylelikle çerçeve sonundaki sesli ifade verilerinin önemlerini kaybetmemesi sağlanır. Çerçevelerin üst üste bindirilmesi işlemi genellikle tüm özellik vektörü çıkarma yöntemlerinde kullanılan bir tekniktir.

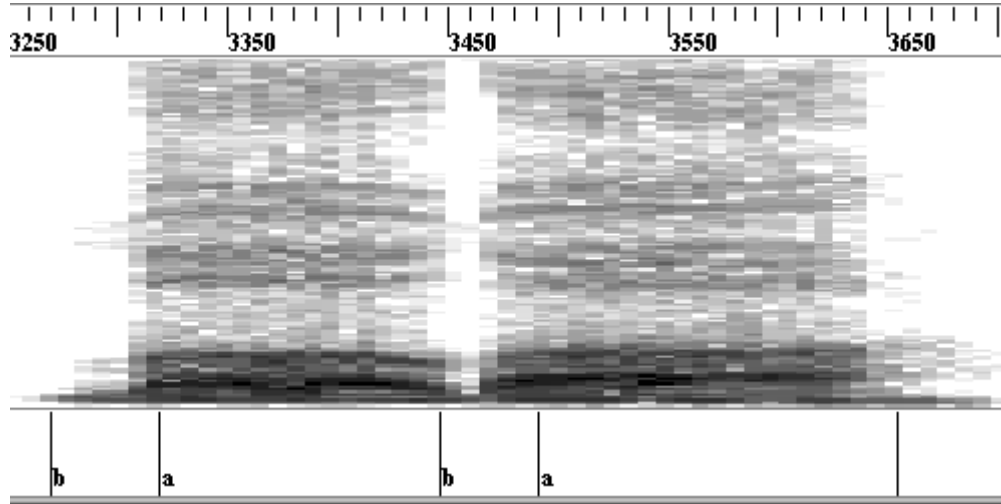
Burada çerçeve boyu 16ms yani 256 değer olarak alınırsa bir sonraki çerçevenin başlangıç noktası ilk çerçevenin bitişinden 128 değer yani 8ms önce olmalıdır.



Şekil 2.9 Örnek Sesli İfade Sinyali.

### 2.5.3.1.2 Spektrum

Sesli ifade tanımada genelde en sık kullanılan yöntem sesli ifadenin spektrum içeriğinin kullanılmasıdır. FFT, spektrum hesaplamak için kullanılan etkili yöntemlerden biridir. Şekil 2.10'da sesli ifadenin spektrumu örneklenmiştir.

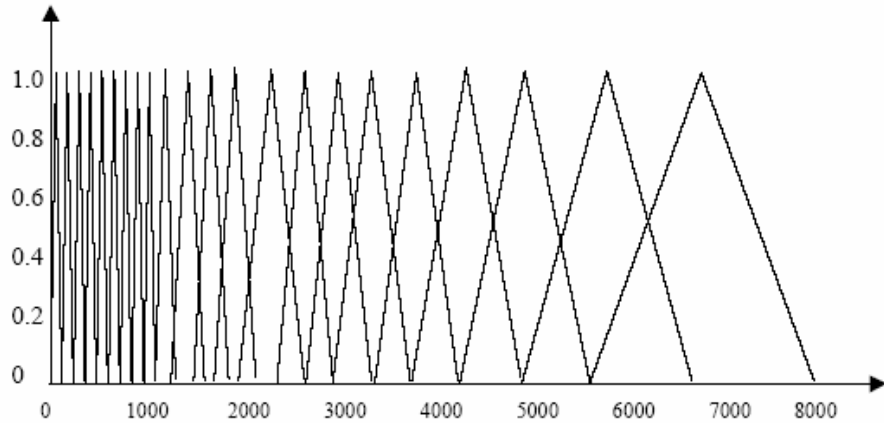


Şekil 2.10 Sesli İfadenin Spektrumu.

Spektrum sesin kısa süreli çerçeveler arasında değişimine duyarlıdır. Sesin belirli aralıklarla frekans yoğunluğunun alınması ile hesaplanır. Bunun için fourier dönüşümünün yanı sıra bir dizi filtre bankasının kullanılması da mümkündür.

Örneğin 256 değer için alınan güç spektrumunda 256 boyutlu bir özellik vektörü kullanılmaz. Bunun yerine, bu değerler belirli sayıda kanallara ayrılır. Bu kanalların her biri filtre bankasındaki birer bant geçimli filtre olarak düşünülür. Bu şekilde filtrelenmiş değerler o andaki çerçevenin spektrumu olarak alınır. Burada bahsedilen bant geçimli filtrelerin bant genişliklerinin eşit olması zorunlu değildir. Değişken uzunluklu bant genişliklerine sahip filtrelerin kullanılması mümkündür. Ardışık olmayan, yani üst üste bindirilmiş filtre kullanımı da mümkündür.

Filtre seçimi yapılırken amaç sesli ifadenin özelliklerinin en iyi biçimde bir vektör ile ifade edilmesidir. Bunun için günümüzde sık kullanılan ve iyi sonuç verdiği gözlenen Mel-scale filtre bankası iyi bir çözümdür. (Şekil 2.11). Bu filtre bankasında üçgen seklinde filtreler vardır. 0-1000 Hz arasında 100 Hz uzunluğunda ardışık, 1000-8000 Hz arasında ise logaritmik olarak genişliği artan filtreler içermektedir. Bu durumda 8 kHz sıklığındaki bir ses için toplam 22 tane filtre kullanılmakta ve her bir çerçeve için 22 elemanlı bir özellik vektörü kullanılmaktadır.



Şekil 2.11 Mel-Scale Filtre Bankası

Mel frekansı için aşağıdaki yaklaşım kullanılabilir;

$$\text{Mel frekansı} = 2595 \log_{10}(1 + f/700) \quad (2.28)$$



Mel skalası dışında bir başka filtre bankası oluşturma yöntemi de Bark scale filtreler kullanmaktır. Ses frekansından belirli bir frekans aralığına bir eşleştirme yöntemi olan bark skalası (Es. 2.28)'deki formülle açıklanabilir.

$$Bark = 13a \tan\left(\frac{0.76f}{1000}\right) + 3.5a \tan\left(\frac{f^2}{7500^2}\right) \quad (8.3.4)$$

Buradaki frekans skalasının birimi kritik bant genişliği oranı ya da bark olarak adlandırılır. Yukarıda belirtilen mel ve bark skalası formülleriyle filtre oluşturmak için kritik bant genişliklerinin belirlenmesi gerekir. Kritik bant genişlikleri (Es. 2.29) formülüyle belirlenir.

$$BW_{critical} = 25 + 75\left[1 + 1.4(f/1000)^2\right]^{0.69} \quad (2.29)$$

Mel skalası için kritik bant genişlikleri Şekil 2.11'de verildiği gibidir. Mel ve Bark skalalarının her ikisi de frekans aralığının algısal (perceptual) olarak anlamlı ve doğrusal olan bir aralığa dönüştürülmesi için kullanılır. Buradaki tekniklerin birleştirilmesiyle elde edilen sinyal çözümü tekniği kritik bant filtre bankası (critical band filter bank) olarak adlandırılır. Bu filtreler doğrusal fazlı FIR bant geçimli filtrelerdir ve yukarıda belirtilen mel ve bark skalalarında doğrusal sıralanmışlardır. Filtre bankası çözümlemesinin ürettiği veriler her bir veri çerçevesi için bir dizi güç değeridir. Spektrum oluşturma bir diğer yöntemi de Fourier dönüşümüne dayalı filtre bankasının kullanılmasıdır. Bu yöntem kritik bant genişliğine dayalı filtre bankasına göre gerçekleştirimi daha kolay olan bir yöntemdir. Yapılan işlem sesli ifade sinyali üzerinde bir fourier dönüşümü yapıp, sinyalden belirli değerler için örnekler almaktır. Bir sinyalin kesikli fourier dönüşümü (discrete fourier transformation- DFT) (Es. 2.30) ile hesaplanır;

$$S(f) = \sum_{n=0}^{N_s-1} s(n)e^{-j(2\pi f / f_s)n} \quad (2.30)$$

Bu formülde  $f$  hertz cinsinden frekans,  $f_s$  örnek olarak alınan frekans,  $N_s$  örneklerin alındığı pencerenin uzunluğudur. Bu denklem kullanılarak sinyalin belirli frekanslardaki spektrumu hesaplanabilir. Bu basit olarak fourier dönüşümünün kullanılmasıdır. Bunun dışında, daha iyi bir spektral yoğunluk elde etmek için her bir spektral değer elde edilirken komşu değerlerin ağırlıklı bir toplamı alınmaktadır. Bu ağırlıklı toplam alınırken kullanılan komşuların belirlenmesi için mel skalası kullanılmaktadır. DFT dışında fourier dönüşümü için FFT (fast fourier transformation) yöntemi de kullanılmaktadır. Bu sadece DFT yönteminin daha etkinleştirilmiş bir biçimidir.

### 2.5.3.1.3 Cepstrum

Cepstrum kavramı ilk olarak 1963'de Bogert, Healy ve Tukey tarafından kullanılmıştır. Sözcük olarak spektrumdan türetilmiştir. Cepstrum, homomorfik sinyal işleme teknikleri içinde yer alır. Homomorfik sistemler doğrusal olmayan sistemlerin bir sınıfı olarak kabul edilirler. Doğrusal sistemler homomorfik sistemlerin özel bir durumudur. Sesli ifade bağlamında kullanılan homomorfik sistem,  $S(f)=V(f).G(f)$  şeklinde ifade edilebilir. Burada  $s(n)$  sesli ifadeyi,  $v(n)$  gırtlığı, yani, sesin izlediği yolu,  $g(n)$  ise asıl ses sinyalini, yani ses telleri tarafından üretilen ve değişime uğramamış ses sinyalini temsil eder. Bu şekilde ifade edildiğinde gırtlığın etkisini asıl sesteki homomorfik sinyal işleme teknikleriyle ayıkmak mümkün olmaktadır. Bu yöntem ses yolunun üst kısımlarında oluşan sesleri belirlemede zorlanacaktır. Çünkü burada ses telleri tarafından üretilen bir asıl ses söz konusu değildir. Homomorfik sistemlerin sesli ifade işlemede kullanışlı olarak kabul edilmelerinin bir nedeni de gırtlığın özelliklerinin özellik vektöründe temsil edilmesinin sağlanmasıdır. Cepstrum değerlerinin hesaplanması için (Es. 2.31) eşitliği kullanılır.

$$c(n) = \frac{1}{N_s} \sum_{k=0}^{N_s-1} \log_{10} |S(k)| e^{(2\pi / N_s)kn} \quad , 0 \leq n \leq N_s - 1. \quad (2.31)$$

Burada  $c(n)$ ,  $n$ . Cepstrum olarak adlandırılır.  $S(k)$ , Cepstrum değerinin ait olduğu frekans aralığı için alınan fourier dönüşümünü belirtir.  $N_s$  o andaki çerçevenin boyunu gösterir. Burada dikkat edileceği gibi  $c(0)$  doğrudan o andaki DFT spektrum değerini gösterir. Sesli ifadenin gürültüden ayırt edilmesi için önce spektrumun logaritması alınır ve ters fourier dönüşümü yapılır. Bu şekilde belirlenen Cepstrum değerleri fourier dönüşümünden türetilmiş cepstral katsayılar (fourier transform derived cepstral coefficients) olarak adlandırılır. Cepstrum hesaplamak için bir çok başka yöntem kullanmak mümkündür. Bunlardan birisi de doğrusal tahmin (Linear Prediction) yöntemidir. Burada hesaplanan Cepstrum değeri sesli ifade tanımada kullanılan önemli bilgileri elde etmede etkilidir. Fourier dönüşümünde kullanılan frekans mel skalasında örneklenirse elde edilen Cepstrum değerleri mel Cepstrum değerleri olarak adlandırılır. Genelde Cepstrum değerlerinin ilk 20 tanesi özellik vektörü olarak kullanılmaktadır. Yani sesli ifade hakkında önemli bilgiler içeren Cepstrum değerleri ilk 20 tanesidir. Cepstrum değerleri doğrusal olmayan işlemlerle hesaplandığından dolayı gürültüye duyarlı oldukları söylenebilir. Bu sebeple gürültülü ortamlarda sesli ifade tanıma uygulamalarında tercih edilmezler. Mel Cepstrum değerlerini hesaplamak için kullanılan bir diğer yöntemde fourier dönüşümü yerine mel skalasındaki filtre dizisi de kullanılabilir.

$$melc(i) = \sum_{k=1}^K \log |X_k| \cos \left[ i \left( k - \frac{1}{2} \right) \frac{\pi}{K} \right] \quad (2.32)$$

Mel filtre bankası yöntemi (Es. 2.32) formülüyle uygulanır. (Davis and Mermelstein). Burada  $X_k$  mel skalasında  $k$ . bant geçimli filtrenin uygulanmasıyla elde edilen spektrum değeridir.

#### 2.5.3.1.4 LPC (Linear Predictive Coding) Tabanlı Cepstrum Değerleri

Sesli ifade çözümleme tekniklerinden en güçlüsü doğrusal tahmin (Linear Prediction) yöntemidir. Bu yöntem sesli ifade parametrelerinin tahmininde kullanılan en etkin yöntemlerden biridir. Bu parametreler, temel frekans, formant frekansları, spektrum, gırtlığı modelleyen fonksiyonlar ve sesli ifadenin sıkıştırılmasında kullanılan kodlama teknikleridir. Bu metodun önemi ürettiği doğru sonuçlara ve bu

sonuçlara ulaşırken görece olarak daha hızlı çalışmaktadır. LP çözümlemesinin dayandığı temel fikir, bir sesli ifade örneğinin geçmiş sesli ifade örneklerine dayanarak yaklaşık olarak elde edilebileceğidir. Su andaki örnek sesli ifade ile doğrusal olarak tahmin edilen sesli ifade arasındaki farkların karelerinin toplamı en aza indirilmeye çalışılarak, sesli ifadenin tahminini sağlayacak bir dizi birim katsayı bulunabilir.

Bu katsayılara tahmin edici katsayılar (Linear Predictor Coefficients) denir ve tahmin edilene sesli ifadenin doğrusal olarak birleştirilmesi sırasında kullanılan ağırlıklandırma katsayıları olarak da tanımlanabilirler. LP yöntemi sesli ifadenin doğrusal, zamana bağlı değişen bir sistem olarak modellenmesine dayanır. Sesli ifade çözümleme bağlamında LP ses dalgasının formüle edilmesi olarak düşünülebilir. Temel olarak, all-pole türünden doğrusal filtrelerle sesli ifadenin kaynak filtreleme modeli oluşturulur. Bir sonraki sesli ifade örneğinin doğrusal olarak tahmini geçmiş örneklerin ağırlıklı toplamı (Es. 2.33) ile yapılabilir.

$$\hat{s}_n = \sum_{i=1}^p a_i s_{n-i} \quad (2.33)$$

Bu doğrusal filtrenin transfer fonksiyonu ise (Es. 2.34)'daki gibidir.

$$H(z) = \frac{1}{1 - \sum_{i=1}^p a_i z^{-i}} \quad (2.34)$$

Kayıpsız bir borunun transfer fonksiyonu all-pole modeli kullanılarak tanımlanabilmektedir. Sesli ifade bağlamında gırtlak, kayıpsız bir boru olarak kabul edilirse, ses telleri tarafından oluşturulan seslerin bu boru içinden geçimi de all-pole modeli ile modellenebilir. Ancak,

- Gırtlak silindirler biçiminde değildir,
- Gırtlak kayıpsız değildir,
- Gırtlak içinde odacıklar bulunan bir yapıya sahiptir.
- Bazı sessiz harfler dudaklara yakın bir bölgede ve ses tellerinin katkısı olmadan oluşmaktadır.

Bütün bu olumsuz yönlerine rağmen yeterince LP katsayısı kullanılarak sesli ifadedeki seslerin yaklaşık bir değerini elde edecek yaklaşımlar mümkündür.

### *LP parametrelerinin tahmin edilmesi*

N değerden oluşan bir sesli ifade örneği verilmiş olsun. Amaç en uygun sonucu üretecek olan  $a_i$  katsayılarını tahmin etmek için hesaplamalar yapmaktır. En uygun sonucu elde etme testi için farkların karesini en aza indirme yöntemi kullanılır. Herhangi bir anda asıl sesli ifade ile tahmin edilen arasındaki hata (Es. 2.35) formülüyle hesaplanabilir.

$$e_n = s_n - \hat{s}_n = s_n - \sum_{i=1}^p a_i s_{n-i} \quad (2.35)$$

Bu durumda farkların kareleri toplamı, (Es. 2.36) formülüyle hesaplanır.

$$E = \sum_{n=0}^{N-1} e_n^2 = \sum_{n=0}^{N-1} \left( s_n - \sum_{k=1}^p a_k s_{n-k} \right)^2 \quad (2.36)$$

Burada E'nin en küçük değerini aldığı an türevinin sıfır olduğu andır. Yani yukarıdaki formülün ak için türevini alıp sifira eşitlersek gerçek sesli ifade ile tahmin edilen arasında en az hata olduğu durumu belirlemiş oluruz. Buna göre;

$$\frac{\partial E}{\partial a_j} = 0 = - \sum_{n=0}^{N-1} \left( 2(s_n - \sum_{k=1}^p a_k s_{n-k}) s_{n-j} \right) = -2 \sum_{n=0}^{N-1} s_n s_{n-j} + 2 \sum_{n=0}^{N-1} \sum_{k=1}^p a_k a_{k-j} s_{n-j}$$

eşitliğinde yeniden bir düzenleme yapılarak (Es. 2.37) formülü elde edilir.

$$\sum_{n=0}^{N-1} s_n s_{n-j} = \sum_{n=0}^{N-1} \sum_{k=1}^p a_k a_{k-j} s_{n-j} \quad (2.37)$$

Bu eşitlik sesli ifade örneği s-p..s-1 noktaları için LP katsayılarını (ak) bulmayı sağlar. Bu eşitliğin çözümü için üç yol vardır.[9]

- Otokorelasyon (autocorrelation) yöntemi
- Covariance yöntemi.
- Kafes (lattice) yöntemi.

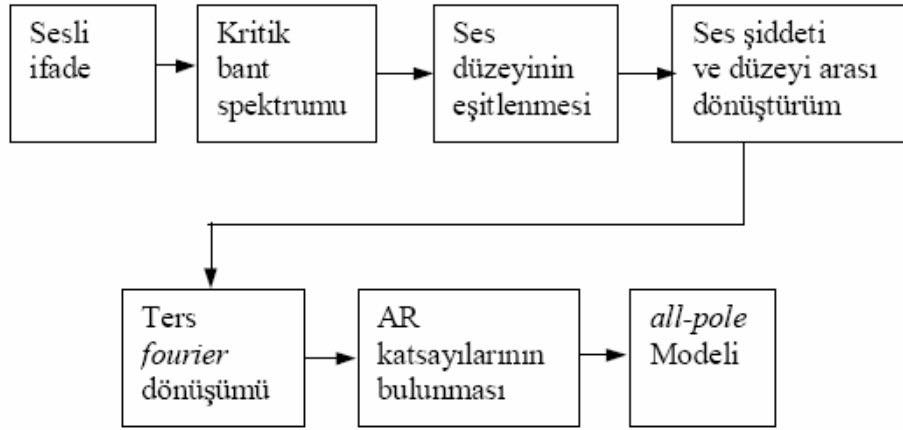
Bu yöntemlerin ayrıntıları burada verilmeyecektir. Sesli ifade tanımada genellikle otokorelasyon yöntemi kullanılır. Bunun sebebi kullanılan etkin hesaplama yöntemi ve ürettiği daha durağan sonuçlardır. Bu yöntemlerle amaçlanan LP katsayılarının tahmini için yukarıda belirtilen doğrusal denklemin çözümünü yapmaktır. Bu denklemin çözümü sonucunda elde edilen LP katsayıları (ak) sesli ifade tanımada kullanılabilir. LPC Cepstrum değerleri, LP katsayılarının(parametrelerinin) doğrudan kullanılmasıyla elde edilir. Bunun için aşağıdaki özyineli yaklaşım kullanılır.( Es. 2.38).

$$c_k = a_k + \frac{1}{k} \sum_{i=1}^{k-1} i c_i a_{k-i} \quad (2.38)$$

Burada ck Cepstrum değerlerinden k indisine sahip olanını temsil etmektedir. ak-i ise ilgili LP katsayısını belirtir.

### 2.5.3.1.5 PLP (Perceptual Linear Prediction)

PLP yöntemi, DFT (kesikli fourier dönüşümü) ve LP tekniklerinin birleştirilmesi ile sesli ifade parametrelerinin hesaplanmasıdır. Bu yöntem insan kulağının duyma sistemini LP yönteminden daha iyi modellemeye yöneliktir. LP tekniğinde sesli ifade modellenirken tüm frekanslardaki sesler eşdeğer tutulmaktadır. Bu durum insan kulağıyla uyumlu değildir. 800 Hz değerinden daha düşük frekanslarda duyma miktarı frekansla birlikte düşer. İnsan kulağı daha çok duyma frekans aralığının ortasındaki frekanslara duyarlıdır. Bu sorunu çözmek için birçok çalışma yapılmıştır. Bu çalışmalardan biri de bulunan LP katsayılarının mel skalasına uyarlanması olmuştur. Bir başka yaklaşım da LP tekniği uygulamadan önce sesli ifadenin güç spektrumunun alınmasıdır [22]. PLP yöntemi de bu yaklaşımı kullanmaktadır. PLP yönteminin aşamalarını gösteren blok gösterimi Şekil 2.12'de gösterilmiştir.



Şekil 2.12 PLP yönteminin aşamaları

### ***a. Kritik Bant Spektrumu***

Spektrum hesaplanırken ilk olarak, uygun bir çerçeve boyutu belirlenip bu çerçeve üzerinde DFT uygulanmaktadır. Elde edilen sonuçlar bark skalasında filtrelenerek kritik bant çözümü yapılmaktadır.

### ***b. Ses Düzeyinin Eşitlenmesi***

Spektrum hesaplandıktan sonra, kulaktaki algılamanın tüm ses düzeyleri için aynı olmadığı gerçeğine dayanarak ses düzeyinin kulağın algılama durumuna göre eşitlenmesi için elde edilen spektrum, es-düzeylilik eğrisi adı verilen bir eğri kullanılarak ön iyileştirme yapılır.

### ***c. Ses Şiddeti-Düzeyi Arası dönüşüm***

Ses şiddeti ve algılanan duyma düzeyi arasındaki doğrusal olmayan bir ilişkinin varlığına dayanılarak bu ilişkinin modellenmesi yapılır. Şiddet ve algılama düzeyi arasındaki bu ilişki  $L = l^{0.33}$  formülüyle modellenmiştir. Yani algılanan sesin düzeyi sesin şiddetinin küp köküne eşittir. Bu iyileştirme de önceki aşamada yapılan

düzey eşitleme işleminde olduğu gibi, daha sonra yapılacak olan LP işleminde daha az parametrenin yeterli modelleme için yeterli olmasına olanak sağlamaktadır.

#### ***d. Ters fourier Dönüşümü (Inverse Fourier Transformation)***

Yukarıdaki iyileştirmelerden sonra ters fourier dönüşümü (IDFT) yapılarak üzerinde iyileştirmeler yapılmış sesli ifadenin tekrar elde edilmesi sağlanır.

#### ***e. Autoregressive(AR) Modelleme***

Burada son aşamada elde edilmiş olan, üzerinde kulağın algılamasına dayalı iyileştirmelerin yapıldığı sesli ifade sinyali için bir LP modeli oluşturulur. Yapılan deneyler oluşturulan bu modelin daha az parametre ile doğrudan oluşturulan LP modelinden daha sonuç verdiğini göstermektedir [22].

#### ***f. RASTA (RelAtive SpekTrA)***

Özellik vektörü oluşturmada kullanılan RASTA yönteminde, sesli ifade içindeki çevresel etkilerin, yani gürültünün, modellenmesine dayalı bir sesli ifade modelleme yöntemi kullanılır. RASTA ismi, görelî spektrum (RelAtive SpekTrA) ifadesinin kısaltılmasıyla elde edilmiştir. Yukarıda belirtilen PLP yöntemi üzerine gürültü modelleme tekniği eklenerek elde edilen bir yöntemdir. RASTA yönteminin dayandığı temel insan kulağının sesli ifadeyi algılamasının daha önceki seslerden önemli derecede etkilendiğidir. Yani sesli ifadenin algılanması daha önce duyulan seslere bağlıdır. Daha değişik bir ifadeyle algılama su andaki ses ile önceki ses arasındaki spektral farka bağlıdır. Bu durumda insan kulağı yavaş değişen seslere daha az duyarlıdır denebilir. Yapılan sesli ifade çözümlemesinin yavaş değişen seslere daha az duyarlı yapılması insan kulağının bu özelliğinin de modellenmesini sağlar. Bunu yapmak için daha önce belirtilen PLP yönteminde kullanılan filtreleme yönteminde değişiklikler yapılmıştır. Kullanılan filtreler spektral sıfır değeri keskinleştirilmiş, yani sıfır frekans düzeyine aniden inen filtrelerle değiştirilmiştir. Böylece frekanslardaki yavaş değişimlerin etkisi azaltılmıştır [22][23].



### 2.5.3.2 Uzun Süreli Özellik Vektörleri

Uzun süreli özellik vektörleri sesli ifadenin görelisi olarak daha uzun bir kesimini temsil eden özellik vektörleridir. Bu süre kısaca bir veri çerçevesinden (örneğin 16 ms) daha uzun bir süredir. Bu vektörler içerik vektörü diye de adlandırılabilir. Uzun süreli özellik vektörleri birden fazla kısa süreli özellik vektörünün birleştirilmesi ile elde edilir. Bu birleştirme işlemi ortalama alma, ardı ardına ekleme ya da fark alma yöntemleri kullanılarak yapılır. Ortalama alma sesli ifadede bulunabilecek gürültünün etkisini azaltmaya yönelik bir yaklaşımdır. Ortalama alma sonucu elde edilen özellik vektörlerinin oluşturduğu spektrum daha durağan ve daha açıklayıcıdır. Burada belirtilen üç yöntemi de değişik şekillerde, bir arada kullanmak mümkündür [7]. Birleştirme (concatenation) işlemi birden fazla özellik vektörünün yan yana eklenmesiyle gerçekleştirilir. Birleştirilen özellik vektörleri farklı yöntemlerle de elde edilmiş olabilir. Ayrıca sadece bir çerçeve üzerinden hesaplanan özellik vektörleri birleştirilebileceği gibi, farklı zamanlarda hesaplanan özellik vektörlerini de birleştirmek mümkündür. Örneğin enerji değerini bir cepstrum vektörünün başına eklemek ya da ardı ardına iki çerçeve için hesaplanmış iki Cepstrum vektörünü yan yana eklemek mümkündür. Fark alma işlemi (delta), özellik vektörlerinin herhangi bir biçimde farklarının alınmasıdır. Fark vektörleri sesli ifadenin kısa süreli değişimlerinin açığa çıkmasını sağlayan özellik vektörleridir. Fark alma işlemi sonucu elde edilen vektörler zaman boyutunda özellik vektörlerinin benzerliğini açığa çıkarır.

### 2.5.3.3 Sesli İfadenin Kesimlenmesi

Sesli ifade tanınmanın ilk aşaması olan ön işleme ve özellik vektörü çıkarma sırasında yapılması gereken önemli bir işlem de kesimlemedir (segmentation). Kesimlemenin sonuçları sesli ifadenin değişik aşamalarında kullanılabilir. Sınıflandırma yapılırken etiketlenmiş sesli ifade şablonlarının (özellik vektörleri) kullanılması gerekli olduğundan, öncelikli olarak yukarıda belirtilen yöntemlerle elde edilen özellik vektörlerinin etiketlenmesi gereklidir. Bu işlem sesli ifade tanıma

sisteminin eğitim aşaması için temel oluşturmaktadır. Etiketleme işlemi el ile (manual) yapılan bir işlemdir. Ancak etiketlerin doğru yerlere konması için sesli ifade kesimleme yöntemlerinden yararlanır. Kesimleme işlemini de el ile yapmak mümkündür. Belirli bir algoritmaya dayalı sesli ifade kesimleme yöntemi otokesimleme (auto-segmentation) olarak adlandırılır. Kesimleme işlemi farklı düzeylerde olabilir. Bunlardan başlıcaları;

- Sesli ifadenin olmadığı kesimlerin, yani duraklama anlarının sesli ifadeden ayrılmasına yönelik kesimle,
- Fonem sınırlarının belirlenmesine yönelik kesimleme,
- Sözcük sınırlarını belirlenmesine yönelik kesimleme, olarak sıralanabilir.

Kesimlemede sesli ifadeyi oluşturan temel frekansın (pitch-fundamental frequency) belirlenmesi önemli bir yer tutar. Fakat temel frekansın belirlenmesi zor bir işlem olduğundan ve bu işlem için kesin sonuçlar üreten belirli bir yöntem olmadığından, üst düzeyde kesimleme yapılırken, yani hece ya da sözcük sınırları belirlenirken pek sık kullanılan bir yöntem değildir. Daha çok ünlü ve ünsüz seslerin sınırlarının belirlenmesi için kullanılır.

#### **2.5.3.4 Temel Frekansın Belirlenmesi (Pitch Estimation)**

Sesli ifade ses tellerindeki periyodik titreşimlerin gırtlakta ve ağız boşluğunda biçimlenmesiyle oluşmaktadır. Ses tellerindeki bu titreşimin frekansı, konuşmacıya bağlı ve değişim aralığı küçük olan bir frekanstır. Bu frekansa sesli ifadenin temel frekansı yada pitch denir. Temel frekans sesli ifadedeki sesli ve sessiz kesimlerin belirlenmesinde etkili olarak kullanılabilir. Sesli ifade içinde böyle bir temel frekansın olduğu kesimler sesli, olmadığı kesimler ise sessiz olarak etiketlenebilir. Pitch periyodunun belirlenmesi için kullanılan bir çok yöntem vardır. Bu yöntemleri şu şekilde gruplayabiliriz;

- Sesli ifadenin zaman boyutundaki özelliklerinin kullanılması,

- Sesli ifadenin frekans boyutundaki özelliklerin kullanılması,
- Hem zaman, hem de frekans özelliklerinin aynı anda kullanılması.

Zaman boyutundaki yöntemler, doğrudan sesli ifade dalgası üzerinde işlem yaparak pitch periyodunun belirlenmesine yöneliktir. Genellikle sesli ifade dalgasındaki tepe ve vadilerin sayılarını yada dalganın sıfır noktasını geçme sayılarını kullanırlar. Bu tür pitch belirleyiciler gerçekleştirimi en basit olanlardır. Frekans boyutundaki yöntemler genellikle zaman boyutunu da kullanır. Sesli ifadenin hem spektrumu kullanılır hem de oto korelasyon yöntemi ile zaman boyutunda işlemler yapılır. Pitch belirleme yönteminin seçiminde etkili olabilecek faktörler şunlardır;

- Pitch değerinin doğru olarak tahmini,
- Sesli ifadedeki sesli-sessiz kesimlerin doğru olarak belirlenmesi,
- Farklı kişilerin sesi için etkili olması,
- Hızlı işlem yapma olanağı,
- Algoritma karmaşıklığının yüksek olmaması,
- Donanımsal olarak kolay gerçekleştirilebilme olanağı,

Pitch belirleme tarihsel olarak, sesli ifade üzerinde yapılan çözümlerlerden ilkidir, fakat yukarıda da belirtildiği gibi hala kesin bir çözümü bulunamamıştır. Pitch belirlemedeki zorluklar genel olarak üç etkene dayandırılabilir. İlk olarak, ses tellerindeki titreşimin tam olarak periyodik olmamasıdır, özellikle seslerin basındaki titreşim ile sonlara doğru olan titreşim farklılık göstermektedir. İkincisi, sesli ifadeden gırtlaktaki farklılaşmaların etkisini çıkarıp ses tellerinin titresim sinyalinin yakalanması çok zordur. Sonuncusu ise pitch değer aralığının çok geniş olmasıdır. Bu sebepler pitch belirlemelerini zor hale getirmektedir. Pitch periyodunun belirlenmesi için kullanılan yöntemler aşağıdaki gibi sınıflandırılabilir [7].

#### 2.5.3.4.1 Sesli İfade Dalgasının İşlenmesine Yönelik Yöntemler;

a) *Koşut isleme dayalı pitch belirleme*: Bu yöntemde pitch belirlemek için, bir çok tepe noktası belirleyici ögenin koşut olarak islemesine dayalı kurallar tanımlanır.

b) *Veri azaltma yöntemi*: Pitch değeri olamayacak periyodların mantıksal bir işlem sonucu sesli ifade sinyalinden arındırılmasına dayanır.

c) *Sıfırı geçme sayısının kullanılması*: Tekrarlı sıfırı geçme oranlarını tespit edip buna dayalı bir pitch tahmini yapmaya dayanır.

#### 2.5.3.4.2 Korelasyon Kullanımına Dayalı Yöntemler;

a) *Oto korelasyon yöntemi*: Sesli ifade dalgasının oto korelasyon fonksiyonu kullanılır. Hesaplama işlemlerinin kolaylaştırılması için sesli ifade dalgası üzerine merkezi kesme (center clipping) ve tepe kesme yöntemleri uygulanır. Bunlar sesli ifade dalgasının daha düzgün bir spektruma sahip olmasını sağlar.

b) *Değiştirilmiş korelasyon yöntemi*: LPC çözümlemesinin artık (residual) sinyali üzerinde oto korelasyon fonksiyonu kullanımı ile pitch belirleme yöntemidir. Hesaplanması LP filtreleme ve kutuplaştırma kullanılarak basitleştirilmiştir.

c) *SIFT (Simplified Inverse Filtering Technique - basitleştirilmiş, ters filtreleme tekniği) yöntemi*: Spektrum düzenleme için, düşük frekanslarda tekrar örneklenen sesli ifade verilerine LPC modelinin uygulanması yöntemidir.

d) *AMDF (Average Magnitude Difference Function - ortalama büyüklük farkları fonksiyonu) yöntemi*: Periyodikliğin tespiti için verilerin büyüklüklerinin farklarını almaya dayalı bir fonksiyon kullanma yöntemidir.

### 2.5.3.4.3 Spektrogramun İşlenmesine Dayalı Yöntemler;

a) Cepstrum yöntemi: FFT'ye dayalı Cepstrum değerleri kullanarak spektral yapıdan faydalanır.

b) Periyot histogramı hesaplama yöntemi: Spektral boyuttaki harmonik bileşenleri için bir histogram hesaplar. Pitch bu histogramdaki değerlerin ortak böleni olarak hesaplanır. Burada sınıflama amacıyla pitch belirleme yöntemleri kısaca tanımlanmıştır. Daha geniş açıklama için [7][20] referanslarına bakılabilir. Sesli ifadenin kesimlenmesi için kullanılan bir başka yöntem de sesli ifade dalgasının enerjisi ve sıfırı geççe sayısıdır. Bu tez kapsamında sözcük sınırlarının ve sözcükte bulunan hece sayısının bulunması için bu iki yöntem kullanılmıştır.

### 2.5.3.4.5 Sıfır Noktasını Geçme Sayısı (Zero Crossing Rate)

Kesikli sinyal işleme bağlamında sıfır noktasını geçme, sinyalin ardışık iki değerinin farklı işaretlere sahip olması durumunda oluşur. Bir sinyalin sıfırı geçme sayısı dar banda sahip sinyallerde sinyalin frekansını belirlemenin en basit yollarından biridir. Örneğin sinüsel bir dalganın frekansı  $F_0$  ve örnekleme frekansı (sıklığı)  $F_s$  ise sinüsel dalganın her bir periyodu için  $F_s / F_0$  tane örnek alınmış demektir. Her bir periyot için sıfırı geçme sayısı ikidir. Bu durumda bu sinüsel dalga için sıfırı geçme sayısı (Eş. 2.39) formülüyle hesaplanır.

$$Z = 2 \frac{F_0}{F_s} \frac{\text{geçiş}}{\text{örnek sayısı}} \quad (2.39)$$

Örneğin 1 kHz frekansına sahip bir sinüsel dalga 16 kHz frekansıyla örneklenirse sıfır noktasını geçiş sayısı 1/8 olur. Bunun anlamı, sekiz örnek için bir sıfırı geçme durumunun oluşacağıdır. Buradan da anlaşılabilceği gibi ortalama sıfırı geçme sayısı sinüsel sinyalin frekansının belirlenmesi için etkin bir yöntemdir. Sesli ifade sinyalleri geniş bant sinyallerdir. Yani içerdikleri frekanslar geniş bir aralıkta değişim gösterir. Bu sebeple sıfırı geçme sayısı sesli ifade verileri için daha az önem

taşıır. Ancak kısa süreli sıfırı geme sayısı sinyalin spektral özelliklerinin tahmin edilmesinde faydalı olabilir. Sıfırı geme sayısını sayısallaştırılmıř bir sinyal için (Es. 2.40) ile hesaplamak mümkündür.

$$Z_n = \sum_{m=-\infty}^{\infty} |\text{sgn}[x(m)] - \text{sgn}[x(m-1)]| w(n-m)$$

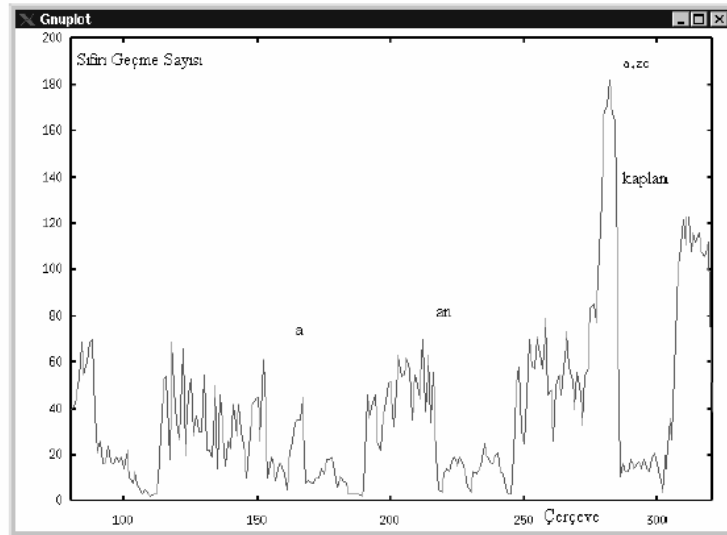
Burada,

$$\begin{aligned} \text{sgn}[x(n)] &= 1 & x(n) \geq 0, \\ &= -1 & x(n) < 0 \end{aligned}$$

ve

$$\begin{aligned} w(n) &= \frac{1}{2N} & 0 \leq n \leq N-1 \\ &= 0 & \text{Diğer durumlarda} \end{aligned} \quad (2.40)$$

Burada sesli ifade verilerinden n. örnek için daha önceki N örnek kullanılarak bir sıfırı geme oranı hesaplanmaktadır.



řekil 2.13 Çereve sayısı - sıfır noktasını geme sayısı grafiđi

Sıfırı geçme sayısı sesli ifade tanıma sırasında genellikle sesli ve sessiz kesimlerin belirlenmesinde kullanılmaktadır. Sesli ifade üretim modeline göre sesli kesimler 3 kHz frekans değerinden daha büyük frekanslara sahip olamazlar. Frekansın yükselmesi sıfırı geçme sayısının da yükselmesi anlamına geleceğinden sıfırı geçme sayısına bakarak sesli ve sessiz kesimleri belirlemek mümkün olmaktadır. Yani sesli ifadenin belirli bir kesimini, sıfırı geçme sayısı yüksek ise sessiz, sıfırı geçme sayısı düşük ise sesli olarak etiketlemek mümkündür. Fakat burada yüksek ve düşük sınırları belirsizdir. Bu yüzden kesin bir sınır koymak mümkün olamaz. Ancak, eşik değerleri belirlenerek ya da sesli ifade üzerinde yapılan başka ölçümler (enerji gibi) kullanılarak sınırların gerçeğe yakın olması sağlanabilir. Örnek bir sıfırı geçme sayısı hesaplaması Şekil 2.13'da verilmiştir. Burada sıfırı geçme sayıları hesaplanırken 16 kHz sıklığında örneklenmiş bir sesli ifade verisi üzerinde 15 ms uzunluğunda bir çerçeve 10 ms çerçeve kaydırması yapılmıştır.

### 2.5.3.6 Kısa Süreli Enerji Ve Ortalama Genlik(Short Time Energy And Average Magnitude)

Sesli ifade sinyalinin genliği zamanla değişim göstermektedir. Gerçekte sesli ifadedeki sessiz kesimlerin genliği sesli kesimlere göre çok daha düşüktür. Sesli ifade sinyalinin kısa bir süresi için hesaplanan enerji sesli ifade sinyalindeki genlik değişimlerini belirgin bir şekilde ortaya çıkarır. Genel olarak kısa süreli enerji (Eş. 2.41)'deki formülle açıklanabilir.

$$E_n = \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2 \quad (2.41)$$

Bu formül (Eş. 2.42)'deki şekilde de yazılabilir.

$$E_n = \sum_{m=-\infty}^{\infty} x^2(m)h(n-m) \quad (2.42)$$

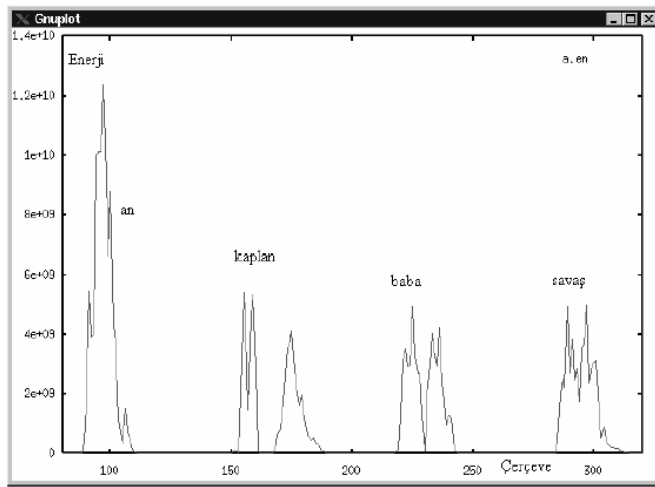
Burada  $h(n) = w^2(n)$  olarak düşünülebilir. Yukarıdaki formüle göre enerji kısa süreli ortalama genlik değerinin karesinin alınmasıyla bulunmaktadır. Bu işlem

yapılırken pencerelere yöntemi kullanılarak elde edilen sonuçların daha belirleyici olması sağlanmaktadır. Kısa süreli ortalama genlik sesli-sessiz kesimlerin ayırt edilmesinde kullanıldığında yeterince ayırtıcı olmayabilir, enerji hesaplanırken kare alma yöntemi kullanıldığından elde edilen sonuçlar küçük değişimlere daha duyarlı olacaktır. Sesli ifade sinyalinin hamming penceresinden geçirildikten sonraki enerjisi Şekil 2.14’de örneklenmiştir. Enerji hesaplanırken iki türlü pencerelere yöntemi kullanılabilir. Bunlardan biri dikdörtgen pencerelerdir. Bu pencerelere yönteminde sinyal belirli bir sabit değerle çarpılır. Bir diğer pencere de hamming penceresi kullanma yöntemidir.

$$h(n) = 0.54 - 0.46 \cos(2\pi n / (N - 1)), \quad 0 \leq n \leq N - 1$$

$$= 0 \quad \text{diğer durumlarda} \quad (2.43)$$

Hamming penceresi ortası tümsek bir eğri seklindedir. O andaki çerçevede bulunan verilerden çerçeve ortasında bulunan değerlerinin önemi artarken çerçeve kenarlarındaki veriler daha önemsiz hale getirilmektedir. Hamming penceresi sesli ifade sinyallerinin islenmesinde, çeşitli aşamalarda sıklıkla kullanılan bir pencere yöntemi. Hamming filtresinin etkisi Şekil 2.14’de görülmektedir. Çizimdeki enerji eğrileri karşılaştırıldığında hamming pencere kullanıldığında oluşan enerji eğrisinin daha düzgün olduğu gözlenmektedir. Bu sebeple üzerinde yorum yapmaya, sesli-sessiz kesimlerin ayırt edilmesinde kullanılmaya daha elverişlidir.



Şekil 2.14 Çerçeve sayısı - sesli ifade sinyalinin enerjisi. (Hamming penceresinden geçirilmiş sesli ifade kullanılmıştır.)



## 2.6 DALGACIKLAR TEORİSİ

### 2.6.1 Giriş

Dalgacıklar, veriyi farklı frekans bileşenlerine ayıran ve sonra kendi ölçekleriyle eşleştirilmiş bir çözünürlüğe sahip bileşenler üzerinde çalışan matematiksel fonksiyonlardır. İşaretin (sinyalin) süreksizliklere ve keskin, sivri uçlara sahip olduğu fiziksel durumları incelemede, geleneksel Fourier metotları üzerine avantajlara sahiptir. Dalgacıklar matematik, kuantum fiziği, elektrik mühendisliği ve sismik jeoloji alanlarından bağımsız olarak geliştirildiler. Son on yıl içerisinde bu alanlar arasındaki yer değiştirmeler resim sıkıştırma, türbülans, insanın görme gücü, radar ve depremi önceden tahmin etme gibi birçok yeni dalgacık uygulamalarına yol açtı.

Burada öncelikle Hilbert uzayı kısaca gözden geçirilecek, Fourier ile başlayan dalgacıkların tarihi anlatılacaktır. Fourier dönüşümleri ile dalgacık dönüşümleri karşılaştırılıp, dalgacıkların diğer özel yönleri ve özellikleri ifade edilecektir ve bazı ilginç uygulamalar verilecektir.

Dalgacık dönüşümü, fonksiyonları, operatörleri veya veriyi farklı frekanstaki bileşenlerine ayıran ve ayrı ayrı her bileşen üzerinde çalışmamıza izin veren bir araçtır. Dalgacık terimi ilk kez 1982 yılında Daubechies tarafından kullanılmıştır. İlgilendiğimiz uzayın ortogonal bazını Hilbert uzayı metodu ile bulabiliriz. İşte dalgacık analizi bu metod ile yapılan analizin bir genelleştirilmesi olarak düşünülebilir. O zaman ilgilendiğimiz uzaydaki denklemler baz cinsinden çözülebilir. Hilbert uzayı teknikleri özellikle lineer adi diferansiyel denklemlerin (ODEs) çözümünde yararlıdır ve bu teknikler belirli kısmi türevli diferansiyel denklemleri (PDEs) değişkenlere ayrılması yöntemiyle iki veya daha fazla adi diferansiyel denkleme (ODEs) indirgememize izin verir.

## 2.6.2 Hilbert Uzayı Analizi

Aşağıda açıklanan Hilbert uzayı analizinin basit ifadesi dalgacık dönüşümünü anlamamıza yardım eder:

### İç (skaler) Çarpım Uzayını Belirlemek

Bir iç (skaler) çarpım uzayı kapalı bir vektör uzayı ve o uzayda tanımlanan bir iç (skaler) çarpımdan oluşur. Örneğin,  $V$  tanım bölgesi  $R$  reel ekseni olan reel değerli fonksiyonlar uzayı olsun ve  $V$  'deki bütün  $f, g$  'ler için  $\langle f, g \rangle$  iç çarpımı

$$\langle f, g \rangle \equiv \int_R f(x) \overline{g(x)} dx$$

şeklinde tanımlansın. Ayrıca  $\langle f, g \rangle = 0$  olduğunda  $f$  ve  $g$ 'ye birbirlerine ortogonaldir denir.

### Hilbert Uzayını Tanımlamak

Bir  $f$  fonksiyonunun normu, iç çarpım cinsinden  $\|f\| = \langle f, f \rangle^{1/2}$  ifadesi ile verilir. Böyle bir iç çarpım verildiğinde, norm doğrudan Öklid uzayındaki uzunluğa benzerdir. Normu kullanarak bir  $L_p(V)$  Hilbert uzayı

$$L_p(V) \equiv \{ f \in V \mid \|f\|_p < \infty \}$$

şeklinde tanımlanır.  $p = 2$  durumu çok doğaldır. Bu durumda ilgilendiğimiz uzayın,  $V$  üzerindeki bütün karesi integre edilebilir  $f$  fonksiyonlarını içerdiği söylenir.

### Hilbert Uzayı Üzerinde Lineer Bir Operatör Belirtmek

$H$  Hilbert uzayındaki bir  $L$  lineer operatörü,  $H$  'deki bütün  $f, g$  fonksiyonları ve bütün  $c$  sabitleri için

$$L(f + g) = L(f) + L(g)$$

ve

$$L(cf) = c L(f)$$

özelliklerini sağlar.

Genellikle bu operatör iki kısımda belirtilir. İlk kısım  $\ell(y)$  lineer diferansiyel ifadesidir. İkinci kısım  $\text{Dom}(\ell)$  operatörü şeklinde gösterilen,  $\ell$  işlemi altındaki kapalıktan ortaya çıkan ve ayrı olarak belirtilen tanım bölgesidir. Bu kapalılık, ilave olarak belirli sınır koşullarının gelmesiyle sınırlandırılabilen maksimal operatörü tanımlar.

Yukarıda kullanılan iç çarpım, tanımdan dolayı, bilinear, simetrik ve pozitif definit olmalıdır. Yani bilinearlik için

$$\langle af + bg, h \rangle = a \langle f, h \rangle + b \langle g, h \rangle$$

$$\langle f, ag + bh \rangle = \bar{a} \langle f, g \rangle + \bar{b} \langle f, h \rangle,$$

Simetriklik için

$$\langle f, g \rangle = \overline{\langle g, f \rangle},$$

ve pozitif definitlik için

$$\langle f, f \rangle \geq 0 \quad \forall f \in V, \text{ ve } (\langle f, f \rangle = 0) \Rightarrow f = 0$$

şartları sağlanmalıdır.

İç çarpımın bu özellikleri, herhangi bir iç çarpım uzayı için Schwarz eşitsizliğini:

$$|\langle f, g \rangle| \leq \|f\| \|g\|$$

ve üçgen eşitsizliğini:

$$\|f + g\| \leq \|f\| + \|g\|$$

ispatlamakta kullanılabilir.

Metrik, uzayda iki nokta arasındaki geometrik uzaklığa benzer şekilde tanımlanabilir. Bir iç çarpım uzayındaki  $f$  ve  $g$  gibi iki fonksiyon arasındaki uzaklık, basitçe  $d(f, g) = \|f - g\|$  ifadesine eşittir. Bu metrik, terim sayısı arttığında ortogonal baz cinsinden keyfi bir fonksiyonun açılımının yakınsaklığını kanıtlamada kullanılır.

Genellikle bir  $L$  lineer operatörünün mümkün olan  $\lambda_1, \lambda_2, \dots, \lambda_n$  özdeğerlerini belirlemek amacıyla

$$Ly = \lambda y \quad y \in \text{Dom}(L)$$

denklemini çözülür ve sonra yukarıdaki denklemi sağlayan, özdeğerlere karşı gelen  $\varphi_1, \varphi_2, \dots, \varphi_n$  özfonksiyonları bulunur.

$L_2(\mathbb{R})$  Hilbert uzayı birden fazla ortonormal baza sahiptir. Örneğin, Fourier dönüşümü Hilbert uzayının dört ayrı ortonormal bazını üretmek için kullanılabilir. Özel operatörlerin özuzaylarını belirleyerek de ortogonal bazlar bulunabilir. Bu özuzayların doğrudan toplamı operatörün tanım bölgesi olacaktır. Eğer operatörün tanım bölgesi tüm Hilbert uzayı ise, o zaman uzay için bir baz bulunabilir. Ancak böyle iddialar aşikar olarak ispat edilemez.

### 2.6.3 Dalgacıklara Genel Bakış

Dalgacıkların arkasındaki temel fikir, ölçeğe göre analiz etmektir. Veri işlemlerinde dalgacıkları kullanarak tamamen yeni bir düşünce yapısı veya bakış açısı kazanılabilir.

Dalgacıklar, belirli matematiksel ihtiyaçları karşılayan ve veri veya diğer fonksiyonları temsil etmede kullanılan fonksiyonlardır. Bu fikir yeni değildir. Fonksiyonları yaklaşık olarak temsil etme, 1800'lü yılların başından beri mevcuttur. Bu yıllarda Joseph Fourier, fonksiyonları temsil etmesi için, sinüs ve kosinüs fonksiyonlarının kullanılabileceğini bulmuştur. Bununla birlikte, veriye bakmak için kullandığımız ölçek de dalgacık analizinde özel bir rol oynar. Dalgacık algoritmaları, veriyi farklı ölçek veya çözünürlüklerde işler. Eğer geniş bir pencereden bir işarete bakarsak, büyük özellikleri farkedebiliriz. Benzer şekilde, küçük bir pencereden bir işarete bakarsak, küçük özellikleri farkedebiliriz.

Dalgacıkları ilginç ve aynı zamanda yararlı kılan bu özelliktir. Bilim adamları yıllarca çok sayıda kısa ve düzensiz dalgalara sahip olan işaretleri yaklaşık olarak temsil etmesi için, Fourier analizinin bazlarını oluşturan sinüs ve kosinüs fonksiyonlarından çok daha uygun fonksiyonlar aramışlardır. Tanımlarına göre bu fonksiyonlar lokal değildir (ve sonsuza doğru uzanırlar). O yüzden bu fonksiyonlar

sivri uçları yaklaşık olarak temsil etme konusunda çok yetersiz kalırlar. Ama dalgacık analiziyle, sonlu tanım bölgelerinde düzgün olarak bulunan, yaklaşık olarak temsil edilmiş fonksiyonlar kullanılabilir. Dalgacıklar sivri süreksizliklere sahip olan veriyi yaklaşık olarak temsil etmek için oldukça uygunlardır.

Dalgacık analizinde amaç, “analiz eden dalgacık” veya “ana dalgacık” diye isimlendirilen bir dalgacık örnek fonksiyonunu elde etmektir. Zaman analizi, örnek dalgacığın daraltılmış, yüksek frekans versiyonu ile yapılır. Buna karşılık frekans analizi ise, aynı dalgacığın genişletilmiş, düşük frekans versiyonu ile yapılır. Orijinal işaret veya fonksiyon, dalgacık açılımı (dalgacık fonksiyonlarının lineer kombinasyonundaki katsayıları kullanarak) cinsinden temsil edilebildiği için, veri işlemleri yalnız karşı gelen dalgacık katsayılarını kullanarak yapılabilir. İlave olarak eğer veriye uygun en iyi dalgacıklar seçilirse veya bir başlangıç değerinin altında kalan katsayılar atılırsa, veri kısmen temsil edilmiş olur. Bu kısmi kodlama, veri sıkıştırma alanında dalgacıkları mükemmel bir araç yapar.

Dalgacıkların kullanıldığı diğer uygulamalı alanlar astronomi, ses bilimi, nükleer mühendisliği, alt-band kodlama, işaret ve görüntü işlemleri, nörofizyoloji, müzik, manyetik rezonans görüntüleme, konuşmayı ayırtma, optik, fraktallar, türbülans, depremin önceden tahmini, radar, insan görüşü ve kısmi türevli diferansiyel denklemleri çözme gibi teorik matematik uygulamalarıdır.

#### **2.6.4 Tarihsel Bakış**

Matematik tarihinde, dalgacık analizi çok farklı kökenlere sahiptir. İşin çoğu 1930’lu yıllarda yapılmıştır. Ancak aynı zamanlarda yapılan başka çalışmalar tutarlı bir teorinin parçaları şeklinde ortaya çıkmamışlardır.

##### **a. 1930 Öncesi**

1930 yılından önce dalgacıkların ortaya çıkmasına sebep olan dal Fourier analizidir. Matematikğin bu dalı, günümüzde Fourier sentezi olarak da isimlendirilen

frekans analizi teorilerine sahip olan Joseph Fourier (1807) ile başladı. Fourier, periyodu  $2\pi$  olan herhangi bir  $f(x)$  fonksiyonunun, o fonksiyonun Fourier serisi denilen

$$a_0 + \sum_{k=1}^{\infty} (a_k \cos kx + b_k \sin kx)$$

toplamı ile temsil edilebileceğini iddia etti. Buradaki  $a_0$ ,  $a_k$  ve  $b_k$  katsayıları

$$a_0 = \frac{1}{2\pi} \int_0^{2\pi} f(x) dx, \quad a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos(kx) dx,$$

$$b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin(kx) dx$$

formülleri ile hesaplanır.

Fourier'in bu iddiası, matematikçilerin fonksiyonlar hakkında sahip oldukları fikirlerin evriminde temel rol oynadı. Bu şekilde Fourier, yeni bir fonksiyonel evrene giden kapıyı açmış oldu.

1807 sonrası, fonksiyonların Fourier serisinin yakınsaklığının ve ortogonal sistemlerin anlamının keşfedilmesiyle matematikçiler giderek önceki frekans analizi fikirlerinden ölçek analizi fikrine, yani ölçeği değişen matematiksel yapılar oluşturarak  $f(x)$  fonksiyonunu analiz etmeye yöneldiler. Bunu yapmak için bir fonksiyon oluşturup onu bir miktar ötelediler ve ölçeğini değiştirdiler. Bu yapıyı bir işareti (sinyali) yaklaşık olarak göstermede uyguladılar ve yöntemi tekrar ettiler. Temel yapıyı alıp onu ötelediler ve yeniden ölçeklediler. Yeni bir yaklaşık gösterim elde etmek için aynı işarete bu yöntemi uygulayıp bu şekilde devam ettiler. Bu tür ölçek analizinin dış etkilere karşı daha az duyarlı olduğu ortaya çıktı. Çünkü bu analiz, farklı ölçek değerlerinde işaretin ortalama dalgalanmalarını ölçüyordu.

Dalgacıkların ilk bahsi A. Haar (1909)'ın tezinde bulunan bir ekte geçti. Haar dalgacığının bir özelliği, kompakt desteğe sahip olmasıdır. Yani Haar dalgacığı sonlu bir aralık dışında sıfır olur. Ne yazık ki, Haar dalgacıklarının türevi sürekli değildir ve bu özellik onların uygulamalarını biraz sınırlar.

### **b. 1930'lu Yıllar**

1930'lu yıllarda, birbirlerinden bağımsız olarak çalışan birkaç grup, ölçeği değişen baz fonksiyonlarını kullanarak fonksiyonları temsil etme üzerine araştırma yaptılar.

Baz fonksiyonları ve ölçeği değişen baz fonksiyonları kavramlarını anlama, dalgacıkları anlamının temelidir. Şimdi baz fonksiyonunu ve ölçeği değişen baz fonksiyonunu açıklayalım:

#### **2.6.4.1 Baz Fonksiyonları Nedir?**

Analog (fonksiyonlar) alanından dijital (vektörler) alanına geçerse bir baz fonksiyonunu açıklamak daha basit olur.

Her iki boyutlu  $(x,y)$  vektörü,  $(1,0)$  ve  $(0,1)$  vektörlerinin lineer kombinasyonu şeklinde yazılabilir. Bu iki vektör  $(x,y)$  için baz vektörleridir. Çünkü  $(1,0)$  vektörünün  $x$  ile çarpılması sonucu oluşan vektör  $(x,0)$ ,  $(0,1)$  vektörünün  $y$  ile çarpılması sonucu oluşan vektör  $(0,y)$  ve toplam  $(x,y)$  vektörüdür.

En iyi baz vektörleri, vektörlerin birbirine dik veya başka bir deyişle ortogonal olması gibi ekstra özelliğe sahiptir.  $(1,0)$  ve  $(0,1)$  bazları için bu kriter sağlanır.

Şimdi analog dünyaya geri dönelim ve bu kavramlarla baz fonksiyonları arasında nasıl ilişki kurulduğuna bakalım.  $(x,y)$  vektörü yerine  $f(x)$  fonksiyonuna sahibiz.  $f(x)$  fonksiyonunun bir müzik ezgisi, örneğin özel bir oktavda A notası olduğunu düşünelim. Genliklerin ve frekansların kombinasyonlarını kullanıp, sinüs ve kosinüsleri toplayarak A notasını oluşturabiliriz. Sinüs ve kosinüsler bu örnekte baz fonksiyonlarıdır ve Fourier sentezinin elemanlarıdır. Seçilen sinüs ve kosinüsler için, onların ortogonal olması gibi ek koşul koyabiliriz. İç çarpımları (skaler

çarpımları) sıfır olan uygun sinüs ve kosinüs fonksiyon terimlerinin kombinasyonunu seçerek bu yapılabilir.  $f(x)$  fonksiyonunu oluşturan ve ortogonal olan özel fonksiyonlar kümesi bu problem için bizim ortogonal baz fonksiyonlarımızdır.

#### 2.6.4.2 Ölçeği Değişen Baz Fonksiyonları Nedir?

Bir baz fonksiyonu, farklı ölçek boyutları kullanmak suretiyle aynı fonksiyon veya veri uzayını parçalara ayırarak ölçekte değişim sağlar. Örneğin 0'dan 1'e kadarlık bölge üzerinde tanımlı bir işarete (sinyale) sahip olduğumuzu düşünelim. İşareti, 0'dan  $\frac{1}{2}$ 'ye ve  $\frac{1}{2}$ 'den 1'e kadar uzanan iki adım fonksiyonu ile bölebiliriz. Sonra orijinal işareti 0'dan  $\frac{1}{4}$ 'e,  $\frac{1}{4}$ 'ten  $\frac{1}{2}$ 'ye,  $\frac{1}{2}$ 'den  $\frac{3}{4}$ 'e ve  $\frac{3}{4}$ 'ten 1'e kadar uzanan dört adım fonksiyonu kullanarak tekrar bölebiliriz. Ve bu şekilde devam ederiz. Her temsilciler kümesi, orijinal işareti özel bir çözünürlük veya ölçek ile kodlar.

##### a. 1960 – 1980

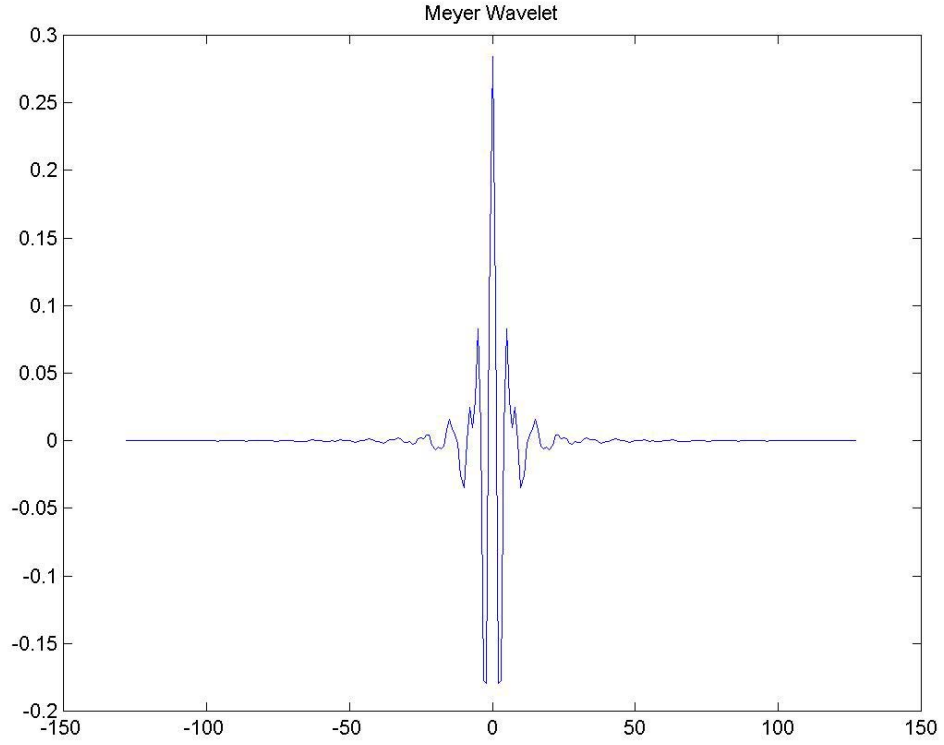
1960 ve 1980 yılları arasında, Guido Weiss ve Ronald R. Coifman isimli matematikçiler, atomlar diye isimlendirilen bir fonksiyon uzayının en basit elemanları üzerinde çalıştılar. Amaçları ortak bir fonksiyon için atomları ve bu atomları kullanarak fonksiyon uzayının tüm elemanlarını yeniden yapmaya izin veren “toplanma kuralları”nı bulmaktı. 1980 yılında, fizikçi Grossman ve mühendis Morlet kuantum fiziği bağlamında dalgacıkları genel olarak tanımladılar. Bu iki araştırmacı, dalgacıklar hakkında fiziksel sezgiye dayanan bir düşünce biçimi sağladılar.

##### b. 1980 Sonrası

1985 yılında, Stephane Mallat dijital işaret işleme konusundaki çalışmaları sonucunda dalgacıkların gelişimine ek bir hız kazandırdı. İkinci derece ayna filtreleri, piramit algoritmaları ve ortonormal dalgacık bazları arasında bazı ilişkiler



olduđunu keřfetti. Y. Meyer, bu sonuların bir kısmından esinlenerek ilk ařık olmıyan dalgacıkları oluřturdu. Haar dalgacıklarının turevi serekli olmadıđı halde, Meyer dalgacıklarının turevi sereklidir; ama buna rađmen Meyer dalgacıkları kompakt desteđe sahip deđildir.



řekil 2.15 Meyer dalgacıđı

İki yıl sonra, Ingrid Daubechies belki řimdiye kadarkilerin en dzenlisi olan, dalgacık uygulamalarının kilometretařı olmuř ve bugun hala olmıya devam eden dalgacık ortonormal baz fonksiyonları kumesi oluřturmak iin Mallat'ın alıřmasını kullandı [23].

### 2.6.5 Fourier Analizi

Fourier'in fonksiyonları sinüs ve kosinüsün bir lineer kombinasyonu olarak temsil etmesi, hem diferansiyel denklemlerin analitik ve sayısal çözümlerinde hem de haberleşme işaretlerinin analizi ve düzeltilmesinde yaygın olarak kullanıldı. Fourier ve dalgacık analizi arasında çok güçlü bağlar vardır [24].

### **2.6.5.1 Fourier Dönüşümleri**

Fourier dönüşümünün faydası, zaman tanım kümesindeki bir işaretin frekans içeriğini analiz etme kabiliyetinde yatar. Dönüşüm, ilk olarak tanım kümesi zaman olan bir fonksiyonu, tanım kümesi frekans olan bir fonksiyona çevirmek suretiyle çalışır. O zaman sinyalin frekans içeriği incelenebilir. Çünkü dönüştürülen fonksiyonun Fourier katsayıları, her frekans değerinde sinüs ve kosinüs fonksiyonlarının her birinin katkısını temsil eder. Ters Fourier dönüşümü de, verinin frekans tanım kümesinden zaman tanım kümesine dönüştürülmesini gerçekleştirir.

#### **2.6.5.1.1 Ayrık Fourier Dönüşümleri**

Ayrık Fourier dönüşümü (DFT) fonksiyonun sonlu sayıdaki örnek noktasından Fourier dönüşümünü tahmin eder. Örnek noktaların, diğer zamanlarda işaretin neye benzediğini tipik olarak gösterdiği farzedilir.

Ayrık Fourier dönüşümü (DFT), sürekli Fourier dönüşümünün sahip olduğu simetri özelliklerinin hemen hemen aynısına sahiptir. Ayrıca, ters ayrık Fourier dönüşümü, ayrık Fourier dönüşümü için olan formülü kullanarak kolayca hesaplanır. Çünkü iki formül hemen hemen özdeştir.

#### **2.6.5.1.2 Pencerelemiş Fourier Dönüşümleri**

Eğer  $f(t)$  periyodik olmayan bir işaretse, periyodik fonksiyonlar olan sinüs ve kosinüsün toplamı işareti doğru olarak temsil etmez. Sinyali periyodik yapmak için onu yapay olarak uzatabiliriz. Fakat bu durum uç noktalarda ek süreklilik gerektirir. Pencerelemiş Fourier dönüşümü (WFT), periyodik olmayan işaret daha iyi nasıl

temsil edilir problemine bir çözümdür. Pencerelemiş Fourier dönüşümü (WFT), tanım kümesi aynı anda zaman ve frekans olan işaretler hakkında bilgi vermek için kullanılabilir.

Pencerelemiş Fourier dönüşümü (WFT) ile, giriş işareti olan  $f(t)$  bölümlere ayrılır ve her bölüm frekans açısından ayrı ayrı analiz edilir. Eğer işaret keskin, sivri uçlu değişimlere sahipse, uç noktalarda kesitler sifira yakınsayacak şekilde giriş verisi pencereleir. Bu pencereleme işlemi, ortasından ziyade aralığın uç noktalarına yakın kısmına daha az önem veren bir ağırlık fonksiyonu aracılığı ile gerçekleştirilir. Pencerenin etkisi, işareti zamanla sınırlamaktır.

### 2.6.5.1.3 Hızlı Fourier Dönüşümleri

Bir fonksiyonu örneklerle yaklaşık olarak temsil etmek ve Fourier integralini ayrık Fourier dönüşümü ile temsil etmek için, mertebesi örnek nokta sayısı olan  $n$ 'e eşit bir matrisi uygulamak gerekir. Bir  $n \times n$  matrisi bir vektörle çarpmak,  $n^2$  mertebesinde aritmetik işleme neden olduğu için, örnek nokta sayısı arttığında problem çabucak daha kötü bir hale gelir. Ama eğer örnekler düzgün yerleştirilmişse, o zaman Fourier matrisi birkaç matrisin çarpımı şeklinde çarpanlara ayrılabilir ve sonuçta oluşan bu çarpanlar, toplam  $n \log n$  mertebesinde aritmetik işleme neden olacak şekilde bir vektöre uygulanabilir. Bu hızlı Fourier dönüşümü (FFT) diye isimlendirilir.

### 2.6.6 Fourier İle Dalgacık Dönüşümlerinin Karşılaştırılması

#### 2.6.6.1 Fourier ve Dalgacık Dönüşümleri Arasındaki Benzerlikler

Hızlı Fourier dönüşümü (FFT) ve ayrık dalgacık dönüşümünün (DWT) her ikisi de lineer işlemlerdir. Bu işlemler, genellikle doldurarak ve  $2n$  uzunluğundaki farklı bir veri vektörüne dönüştürerek bir veri yapısı üretir. Bu veri yapısı çeşitli uzunluklara sahip  $\log 2^n$  sayıda parça içerir.

Dönüşümlerde kullanılan matrislerin matematiksel özellikleri de benzerdir. Hızlı Fourier dönüşümü (FFT) ve ayrık Fourier dönüşümünün (DWT) her ikisi için

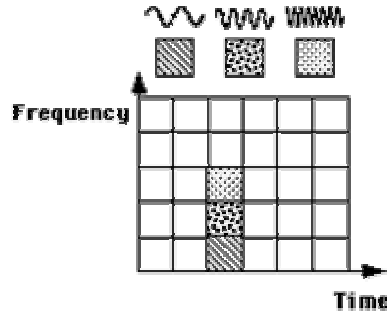
ters dönüşüm matrisi orijinal matrisin transpozudur. Sonuç olarak, her iki dönüşüm, fonksiyon uzayında farklı tanım bölgesine dönüştüren hareket gibi görülebilir. Hızlı Fourier dönüşümü (FFT) için, bu yeni tanım bölgesi sinüs ve kosinüsten oluşan baz fonksiyonlarını içerir. Dalgacık dönüşümü için, bu yeni tanım bölgesi dalgacıklar, ana dalgacıklar veya analiz eden dalgacıklar diye isimlendirilen daha karmaşık baz fonksiyonlarını içerir.

İki dönüşümün başka bir benzerliği de vardır. Baz fonksiyonları frekans ile sınırlandırılırlar. Bu, güç tayfi (bir frekans aralığında ne kadar güç bulunduğu) ve güç dağılımlarını hesaplamada yararlı olan matematiksel araçları oluşturur.

#### **2.6.6.2 Fourier ve Dalgacık Dönüşümleri Arasındaki Farklılıklar**

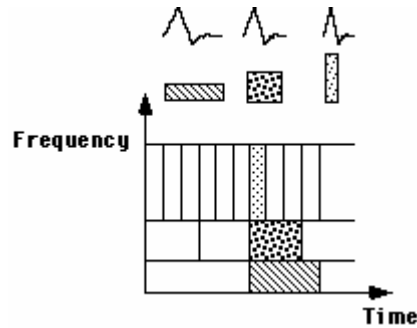
Bu iki tür dönüşüm arasındaki en ilginç farklılık her dalgacık fonksiyonunun uzayda sınırlandırılmasıdır. Fourier sinüs ve kosinüs fonksiyonları böyle değildir. Dalgacıkların frekans sınırlaması düşünüldüğünde, bu sınırlandırma özelliği, dalgacık tanım bölgesine dönüştürüldüğü zaman seyrek dalgacıklar kullanan birçok fonksiyon ve operatör oluşturur. Tersten bakıldığında bu seyreklik, veri sıkıştırma, resimlerdeki özellikleri bulma ve zaman serilerindeki sorunları ortadan kaldırma gibi yararlı bir sürü uygulamayla sonuçlanır.

Fourier dönüşümü ve dalgacık dönüşümü arasındaki zaman – frekans çözünürlük farklarını görmenin bir yolu, zaman – frekans düzleminde baz fonksiyonunun kapladığı alana bakmaktır. Şekil 2.16 penceresinin basit olarak kare dalga olduğu bir pencerelenmiş Fourier dönüşümünü gösteriyor. Kare dalga penceresi, pencereyi belirli bir genişliğe ayarlayacak şekilde sinüs veya kosinüs fonksiyonunu keser. Pencerelenmiş Fourier dönüşümünde (WFT) bütün frekanslar için tek bir pencere kullanılması nedeniyle, çözünürlük analizi zaman – frekans düzlemindeki bütün bölgelerde aynıdır.



Şekil 2.16 Fourier baz fonksiyonlarının zaman-frekans düzlemini kaplaması

Dalgacık dönüşümlerinin bir avantajı pencerelerin değişebilir olmasıdır. İşaret (sinyal) süreksizliklerini ayırmak için, bazı çok kısa baz fonksiyonlarına sahip olmak istenir. Bunu gerçekleştirmenin bir yolu kısa yüksek-frekans baz fonksiyonları ve uzun düşük-frekans baz fonksiyonlarına sahip olmaktır. Bu ortam kesinlikle dalgacık dönüşümleri ile elde ettiğimiz durumdur. Şekil 2.17, Daubechies dalgacığı diye isimlendirilen bir dalgacık fonksiyonunun zaman-frekans düzleminde kapladığı alanı gösteriyor.



Şekil 2.17 Daubechies dalgacık baz fonksiyonunun zaman-frekans düzlemini kaplaması

Hatırlamamız gereken şey, dalgacık dönüşümlerinin, sadece sinüs ve kosinüs fonksiyonlarını kullanan Fourier dönüşümü gibi tek bir baz fonksiyonları kümesine sahip olmadığıdır. Onun yerine, dalgacık dönüşümleri sonsuz sayıda mümkün baz fonksiyonları kümesine sahiptir. Bundan dolayı dalgacık analizi, Fourier analizi gibi diğer zaman-frekans metotları tarafından saklı bırakılan bilgiye doğrudan doğruya erişim sağlar.

## 2.6.7 Sürekli Dalgacık Dönüşümü

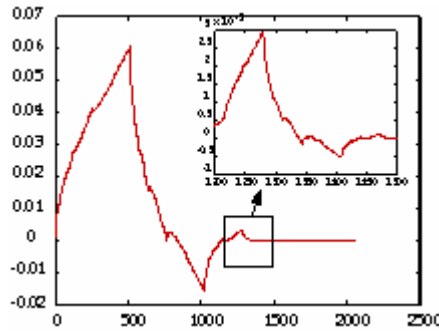
### 2.6.7.1 Tanım ve Basit Özellikler

Bu bölümde amaç tanımlar vermek ve dalgacık dönüşümünün hangi özelliklere sahip olduğunu ve bu dönüşümün hangi alanlarda başarılı bir şekilde uygulanabileceğini incelemektir.

### 2.6.7.2 Dalgacıklar neye benzer?

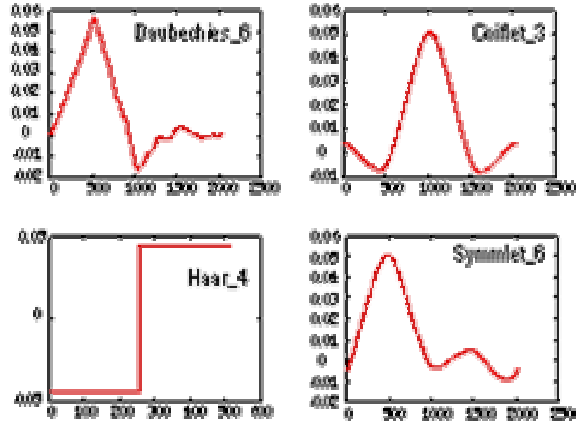
Dalgacık dönüşümleri sonsuz bir kümeden oluşur. Farklı dalgacık aileleri, baz fonksiyonlarının uzayda ne kadar kompakt bir şekilde yerleştirildiği ve bu baz fonksiyonlarının ne kadar düzgün olduğu arasında farklı alternatifler meydana getirir.

Dalgacık bazlarından birkaçı fraktal yapısına sahiptir. Daubechies dalgacık ailesi buna bir örnektir (Şekil 2.18).



Şekil 2.18 Daubechies ana dalgacığının fraktal diye isimlendirilen kendine benzerliği

Her dalgacık ailesi içerisinde (örneğin Daubechies ailesi), katsayıların sayısı ve tekrarlama seviyesi tarafından ayırte edilen dalgacık alt sınıfları vardır. Dalgacıklar çoğu zaman sıfırlandığı anların sayısına göre bir aile içinde sınıflandırılırlar. Bu sağlanması gereken katsayılar için ekstra bir matematiksel ilişkiler kümesidir ve doğrudan katsayıların sayısı ile alakalıdır. Örneğin, Coiflet dalgacık ailesi içerisinde iki sıfır anı olan Coifletler ve üç sıfır anı olan Coifletler vardır. Şekil 2.19'te birkaç farklı dalgacık ailesi örneklerle gösterilmiştir.



Şekil 2.19 Dalgacık ailesi örnekleri

$f$  işareti,  $\varphi$  diye isimlendirilen dalgacık yardımıyla birlikte dalgacık dönüşümü tarafından kontrol ediliyor; daha kesin söylersek,  $f$  ile  $\varphi$ 'nin ötelenmiş ve büyütülmüş versiyonlarının  $L^2$  – skaler çarpımını (iç çarpım) oluşturuyoruz.

$$\tilde{L}_\varphi f(a,b) = |a|^{-1/2} \int f(t) \varphi\left(\frac{t-b}{a}\right) dt$$

Burada kabaca oluşturulmuş olan bu taslağın yardımıyla dalgacık dönüşümünün bazı özellikleri önceden görülebilir. Dalgacık dönüşümünün etkisini örnekle açıklamak için, kompakt desteğine sahip bir  $\varphi$  dalgacığını gözönüne alıyoruz.  $b$  parametresi,  $t = b$  anında  $f$  hakkındaki yerel bilgi  $L_\varphi f(a,b)$ 'de içerilecek şekilde dalgacığı öteler.  $a$  parametresi,  $a \rightarrow 0$  için dalgacık dönüşümü  $t = b$ 'yi yakından gösterecek şekilde, etki küresinin boyutunu kontrol eder.

Aşağıdaki dalgacık dönüşümü incelemesi için Fourier dönüşümü denen başka bir integral dönüşüme ihtiyaç duyuyoruz.  $f \in L^2(\mathbb{R})$  fonksiyonları için Fourier dönüşümü

$$\tilde{f}(w) = \lim_{n \rightarrow \infty} (2\pi)^{-1/2} \int_{-n}^n f(x) e^{-ixw} dx$$

şeklinde tanımlanır. Burada limit  $L^2$  uzayında alınıyor.

Sentez problemini çözebilmek için  $\varphi$  üzerinde teknik bir koşula ihtiyaç duyulur. Bu koşul aşağıdaki tanımda ifade edilecektir:

Kabul edilebilirlik koşulu olan

$$0 < c_\Psi = 2\pi \int_R \frac{|\hat{\Psi}(w)|^2}{|w|} dw < \infty$$

eşitsizliğini sağlayan bir  $\Psi \in L^2(R)$  fonksiyonu dalgacık diye isimlendirilir. Dalgacık dönüşümü,  $L_\Psi$  ( $\Psi$  dalgacığına göre  $f \in L^2(R)$ 'nin fonksiyonu),

$$L_\Psi f(a, b) = \frac{1}{\sqrt{c_\Psi}} |a|^{-1/2} \int_R f(t) \Psi\left(\frac{t-b}{a}\right) dt, \quad a \in R \setminus \{0\}, \quad b \in R$$

şeklinde tanımlanır.

$\Psi$  üzerindeki kabul edilebilirlik koşulundan hemen dalgacık üzerindeki bir gerek koşulu türetebiliriz:  $\Psi \in L^1(R)$  bir dalgacık olsun. Riemann-Lebesgue teoremine göre Fourier dönüşümü  $\Psi$   $R$ 'de süreklidir. Ortalama değeri bu yüzden sıfırdır:

$$0 = \hat{\Psi}(0) = (2\pi)^{-1/2} \int_R \Psi(t) dt$$

Şimdi çeşitli dalgacıkları oluşturmamızı sağlayan bir metot verelim.  $k \geq 1$  olmak üzere  $k$ 'ninci mertebeye kadar türetilen  $\Psi$  fonksiyonu verilsin. Burada  $\Psi, \Psi^{(k)} \in L^2(R)$  ve  $\Psi \neq 0$  olsun. Bu durumda  $\Psi(x) = \varphi^{(k)}(x)$  bir dalgacıktır.

İspat: Fourier dönüşümünün özelliklerinden  $|\hat{\Psi}(w)| = |w|^k |\hat{\varphi}(w)|$  eşitliğini buluruz. Kabul edilebilirlik sabiti  $c_\Psi$ 'nin hesaplanması aşağıdaki sonucu verir:

$$c_\Psi = 2\pi \int_R \frac{|\hat{\Psi}(w)|^2}{|w|} dw = 2\pi \int_R \frac{|w|^{2k} |\hat{\varphi}(w)|^2}{|w|} dw$$



$$\begin{aligned}
&= 2\pi \int_{-1}^1 |w|^{2k-1} |\hat{\varphi}(w)|^2 dw + 2\pi \int_{|w|>1} \frac{|w|^{2k} |\hat{\varphi}(w)|^2}{|w|} dw \\
&\leq 2\pi \left( \|\varphi\|_{L^2}^2 + \|\varphi^{(k)}\|_{L^2}^2 \right) < \infty.
\end{aligned}$$

Dalgacık fonksiyonu

$$\Psi(t) = \begin{cases} 1 & : 0 \leq t < 1/2 \\ -1 & : 1/2 \leq t \leq 1 \\ 0 & : \text{bunların dışında} \end{cases}$$

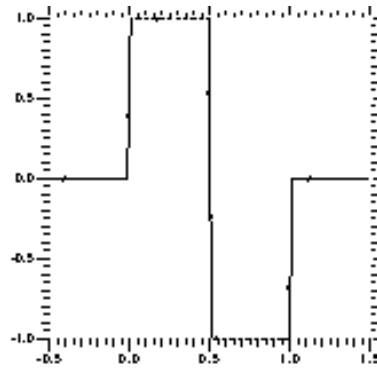
şeklinde verilsin. O zaman  $\Psi$ 'nin Fourier dönüşümü,

$\text{Sinc}(x) = \frac{\text{Sin}(x)}{x}$ , şeklinde tanımlanan Sinc(x) fonksiyonu yardımıyla

$$\hat{\Psi}(w) = \frac{ie^{-iw/2} \text{Sin}(w/4) \text{Sinc}(w/4)}{\sqrt{2\pi}}$$

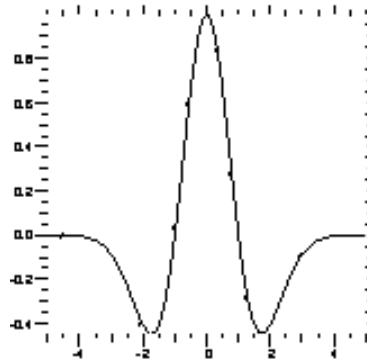
ile verilir. Böylece  $|\hat{\Psi}|$  çift fonksiyondur,  $c_\Psi = 2\ln^2$  olur ve  $w_0 = \pm 4.6622$  noktasında mutlak maksimuma sahiptir.

Şekil 2.20'de verilen bu  $\Psi$  fonksiyonu Haar dalgacıdır.



Şekil 2.20 Haar dalgacı

$\varphi \in L^1(\mathbb{R})$  türevi sürekli bir fonksiyon ve  $\Psi = \varphi' \in L^2$  olsun. O zaman  $\Psi$  kabul edilebilirlik koşulunu sağlar. Bu tipin önemli bir örneği de Şekil 2.21'de görülen Meksika şapkasıdır.



Şekil 2.21 Meksika şapkası fonksiyonu

$$\Psi(x) = -\frac{d^2}{dx^2} e^{-x^2/2} = (1-x^2)e^{-x^2/2}$$

Burada  $c_\Psi = 1$  olmak kaydıyla  $\hat{\Psi}(w) = \frac{w^2 e^{-w^2/2}}{\sqrt{2\pi}}$  eşitliği geçerlidir ve

$w_0 = \sqrt{2}$  'de maksimum  $\hat{\Psi}(\sqrt{2}) = \frac{2}{(\sqrt{2\pi}e)}$  eşitliği ile verilir. Haar dalgacına zıt

olarak, Meksika şapkası her mertebeden türevi sürekli olan bir fonksiyondur. (Yani

Meksika şapkası  $C^\infty$  sınıfından bir fonksiyondur.) Bu, Fourier tanım bölgesinde hızlı bir bozulmaya veya  $w_0$  frekansı civarında daha ayrıntılı bir yerleşime neden olur.

$$f(x) = \begin{cases} 1 & : -1 \leq x \leq 0 \quad \text{or} \quad 1 \leq x \leq \frac{3}{2} \\ 2+x & : -2 \leq x \leq -1 \\ 0 & : \text{bunların dışında} \end{cases}$$

işaretinin dalgacık dönüşümünü Haar dalgacığına ve Meksika şapkasına göre bulabiliriz. Ancak  $a \rightarrow 0$  iken  $f$ 'in süreksizlik noktalarında tekillikler vardır.

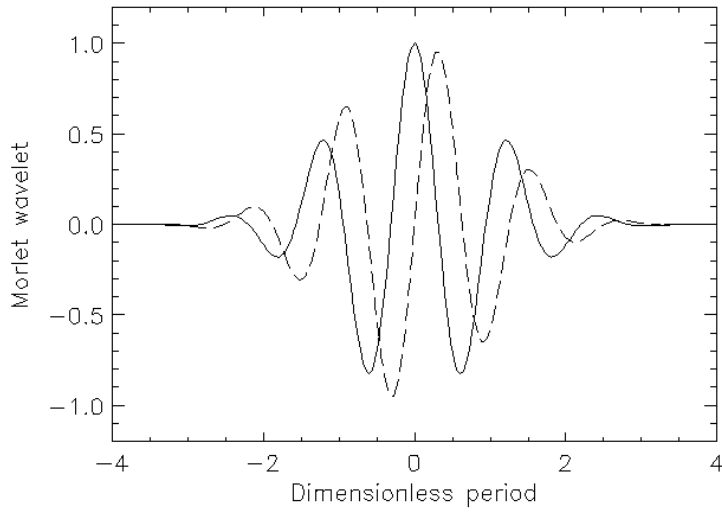
Kabul edilebilirlik koşulu zayıf bir koşuldur. Çünkü dalgacıklar kümesi  $L^2(R)$ 'de yoğundur [2].

Şimdiye kadar sürekli dalgacık dönüşümü ile ilgilendik. Ancak bunun dışında, ayrık dalgacık dönüşümü, hızlı dalgacık dönüşümü ve dalgacık paketlerinin de var olduğunu belirtmeliyiz.

Pencerelenmiş Fourier analizine benzer şekilde, Morlet dalgacığı da yerel olarak periyodik dalgalardan oluşur. Bu dalgacık, kompleks sinus dalgalarının lokalize edilmesiyle elde edilir ve

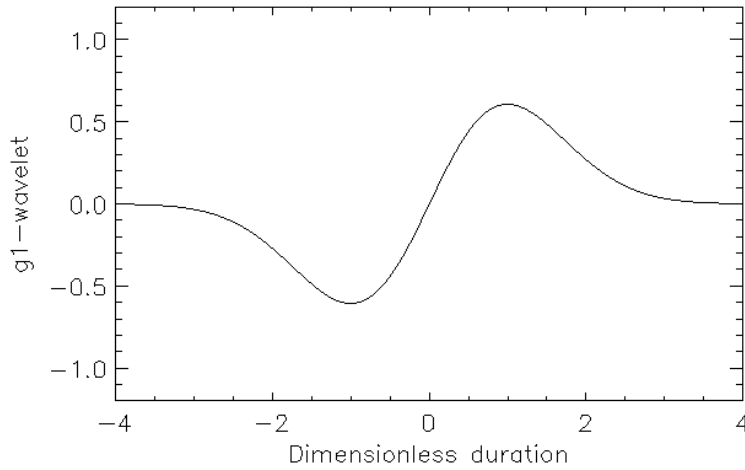
$$\Psi(x, z_0) = (\cos 2\pi x + \sin 2\pi x) e^{(-2x^2\pi^2/z_0^2)} - e^{(-z_0^2/2 - 2x^2\pi^2/z_0^2)}$$

şeklinde tanımlanır. Şekil 2.22'de düz çizgi ile Morlet dalgacığının reel kısmı, kesikli çizgi ile de sanal kısmı gösteriliyor



Şekil 2.22 Morlet dalgacı

Son olarak simetrik olmayan bir dalgacıya örnek verelim:



Şekil 2.23 Antisimetrik dalgacık

### 2.6.7.3 Dalgacık Analiz Teknikleri

Haar bazı, 1910 yılından beri bilinen bir ortogonal bazdır. Aynı zamanda bu baz, dalgacık bazının bilinen ilk ve belki de en basit örneğidir. Bu nedenlerden ötürü biz onu dalgacık analizinin temellerinin anahatlarını göstermede kullanacağız.

Hilbert uzay analizinde, bir lineer operatörün özuzaylarının birleşimini buluyoruz ve o uzayın bazını belirliyoruz. Operatörün kendisi, problemin tanım bölgesinin bazı yönlerini yansıtacak şekilde seçilir. Örneğin, basit harmonik salınım ünitesinin bir çeşidini tarif etmek için sabit katsayılı ikinci mertebeden bir adi diferansiyel denklem (ODE) seçebiliriz. Öz uzayın bazının, belirli periyotlara sahip sinüs ve kosinüs fonksiyonlarından oluştuğu gösterilir.

Dalgacık analizi, problemin tanım bölgesine uygun hale getirilmiş bir baz üretmek için başka yollar sağlar. Örneğin, Haar bazı formülasyonu biçiminde “ana fonksiyon” diye isimlendirilen bir fonksiyondan üretilebilir. Bu strateji genel biçimde dalgacık ailelerini yaratmak için kullanılır. Uygun bir  $\Psi$  fonksiyonundan bir  $\Psi_{m,n}$  dalgacık ailesini üretmek için kullanılan formül  $\Psi_{m,n}(x) \equiv a_0^{-m/2} \Psi(a_0^{-m} x - b_0 n)$  eşitliği ile verilir.  $\Psi$  ve  $a_0, b_0$ 'ın bazı çok özel seçimleri için,  $\Psi_{m,n} L^2(R)$  uzayı için bir ortonormal baz oluşturur. Özel olarak  $a_0 = 2$ ,  $b_0 = 1$  değerleri için öyle  $\Psi$  vardır ki,  $\Psi_{m,n}(x) \equiv 2^{-m/2} \Psi(2^{-m} x - n)$  ( $m, n \in Z$ ) eşitliği  $L^2(R)$  uzayının bir ortonormal bazını oluşturur. Bu, kullanacağımız üreteç fonksiyonudur.

Haar bazı,

$$\Psi(x) = \begin{cases} 1 & : 0 \leq x < \frac{1}{2} \\ -1 & : \frac{1}{2} \leq x < 1 \\ 0 & : \text{bunların dışında} \end{cases}$$

Haar fonksiyonundan elde edilen bu yöntemler tarafından üretilebilir.

$\Psi_{m,n}(x)$  Haar ailesinin gerçekten bir ortonormal baz oluşturduğunu tespit etmek için  $\Psi_{m,n}$ 'in ortogonal ve normalleştirilmiş olduğunu, ve herhangi  $f \in L^2(R)$  fonksiyonunun  $\Psi_{m,n}$  cinsinden keyfi hassasiyette bir kısmi açılım ile yaklaşık olarak temsil edilebileceğini göstermek zorundayız.

İlk önce ortogonalliği kontrol ediyoruz.  $\Psi_{m,n}$ 'in desteğinin  $[2^m n, 2^m(n+1)]$  olduğu hazır olarak görülür. Bu bizi hemen  $n \neq n'$  olduğu zaman  $\langle \Psi_{m,n}, \Psi_{m,n'} \rangle = 0$  sonucuna götürür. Başka bir şekilde ifade edersek, aynı m ölçeğine sahip Haar dalgacıkları, aynı öteleme indeksi n'i paylaşmadıkça üstüste gelmez. Bu bizi

$$\int_{2^m n}^{2^m(n+1)} \left( 2^{-m/2} \Psi(2^{-m} x - n) \right)^2 dx = 2^{-m} \int_0^{2^m} \Psi^2(2^{-m} x) dx = 1$$

integraline götürür. Böylece  $\langle \Psi_{m,n}, \Psi_{m,n'} \rangle = \delta_{n,n'}$  eşitliğini buluruz. Şimdi

keyfi derecede küçük  $\varepsilon$  için,  $\left\| f - \sum_{m,n} c_{m,n} \Psi_{m,n} \right\| < \varepsilon$  olacak şekilde,  $\Psi_{m,n}$ 'in sonlu sayıdaki terimi cinsinden, herhangi  $f \in L^2(R)$  fonksiyonunun açılımını yapmanın mümkün olduğunu göstermek istiyoruz. Başka bir deyişle, ortalama kare metrikte yakınsak olan kısmi toplamı arıyoruz. İspat notasyon olarak aralıklar kullanıldığında daha uzundur ve Daubechies tarafından verilmiştir [5]. Şimdi kısaca ispatı özetleyelim.

Herhangi  $f \in L^2(R)$  fonksiyonunun Lebesgue anlamında integrallenebilir olması nedeniyle,  $f$  fonksiyonunun yeterince geniş desteği ve j için  $[l2^{-j}, (l+1)2^{-j}]$  aralıkları üzerinde parçalı sabit olan kompakt desteğine sahip bir fonksiyon tarafından, yaklaşık olarak temsil edilebileceğini biliyoruz. Haar dalgacığı  $\Psi_{1-j,n}$ 'in aralığıyla kesinlikle üst üste gelen ve farklarını bir  $c_{1-j,n}$  sabiti ile gösterebileceğimiz iki komşu aralık seçebiliriz. Bu işlem,  $f$ 'in desteği  $-j + 1$  ölçeğine sahip olan

dalgacıklar tarafından içerilene kadar tekrar ettirilir. Sonra  $c_{2^{-j}}$  sabitlerini seçmek üzere geri döneriz ve  $f$ 'in desteğinin  $-j + 2$  ölçeğine sahip olan dalgacıklar tarafından içerilmesini sağlarız.

Bu işlem, dalgacığın desteği  $f$ 'in desteğinin tamamını yok edinceye kadar tekrar ettirilir. Bu noktada ardışık dalgacıklar, kısmi toplamın toplam hatasını (azaltarak) göstermeye başlar. Bu durum, verilen hata eşiğine ulaşılan kadar belirsiz olarak sürdürülebilir. Böylece  $f$ , Haar dalgacıklarının kısmi seri açılımı ile keyfi derecede iyi şekilde yaklaşık olarak temsil edilebilir.

### 3. BULGULAR

#### 3.1 ÖNCEKİ ÇALIŞMALAR

İNAL, tarafından DÖK tabanlı özellik çıkartım algoritmalarından kepsral sayıların , ÇKA sınıflandırıcılarında yapılan çalışmada % 91 verimle diğer katsayılara göre % 10 civarında daha iyi sonuç verdiği görülmüştür. İNAL, SOM sınıflandırıcısı kullandığı çalışmasında Türkçe veritabanı ile yapılan metne bağlı kapalı set konuşmacı tanıma uygulaması yapmıştır. Konuşmacı saptama ve doğrulama uygulamalarına ilişkin sonuçlar sırasıyla; ortalama verim % 97.455 ve tüm konuşmacılar için ayrı ayrı tanımlanan SOM ağları yerine tek bir SOM ağı kullanılarak ortalama sistem verimi % 84.029 olarak bulunmuştur [14].

DEMİR, çalışmasında dalgacık dönüşümünü zaman frekans gösterilimi ve çoklu-çözünürlük açısından incelemiştir. Dalgacık dönüşümünün bir uygulaması olarak doku tanıma problemi ele alınmıştır. Bu amaçla; ağaç yapılı dalgacık dönüşümü ile elde edilen alt bant görüntülerine ait enerji değerleri göz önünde bulundurulmuştur. Bu enerji değerlerinden belli bir eşik seviyesini geçen enerjilerin elde edildiği alt bantlar belirlenerek tanıma işlemi gerçekleştirilmeye çalışılmıştır. Çalışmada ağaç yapılı dalgacık dönüşümü ile doku yapıları için enerji değerlerine göre baskın frekans kanalları belirlenmiş ve bunun her doku yapısı için farklı olduğu görülmüştür. Bu baskın frekans kanalları kullanılarak eğitilen sinir ağı sayesinde dokuların tanınması sağlanmıştır [24].

SEVEN , çalışmasında sesin ayırt edici özelliklerini bulma işlemi sırasında ilk olarak Sessizlik Ayırıştırma Algoritmasından yararlanmıştır. Bu yöntemle örneklerin ses bilgisi taşımayan kısımları ayırt edilir ve dikkate alınmaz. Ses örnekleri Hamming penceresi ile parçalara bölünür ve her parça için DÖK katsayıları hesaplanır [25].

ERDEM, yapay sinir ağlarının kullanıldığı ses teknolojisi ile yapılan uygulamalarda sesli ve sessiz sesler ile kelimelerin, cümlelerin ve sürekli konuşmanın tanınması için önceden yapılan çalışmaları esas almış, ses örneklerinin YSA da kullanılması için bu referans bilgilerden yararlanılarak sesin analizini bir algoritma yardımı ile elde etmiştir. Sesin matematiksel modelini elde ederken DÖK analizi ve

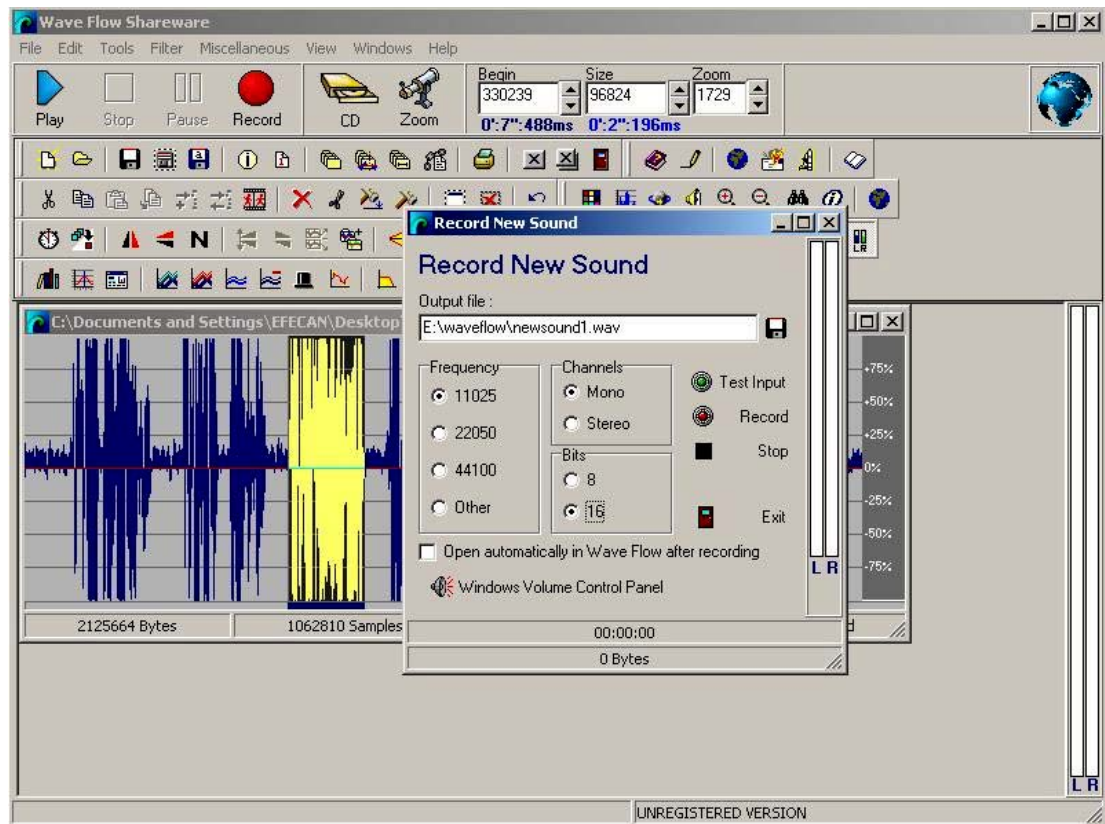


DFT (Discrete Fourier Transform) metotlarından faydalanarak bu analizler için de sesin bilgisayarda bulunan analizlerinden yararlanmıştır [18].

### 3.2 SESLERİN DOSYALARA KAYDEDİLMESİ

Bu bölümde verilerin hazırlanması, eğitimin yapılması, test aşamasının gerçekleştirilmesi aşamalarında yapılan çalışmalardan bahsedilmektedir. 6 erkek ve 4 kadın olmak üzere 10 kişiden oluşan konuşmacı grubundan “ Merhaba” , “Günaydın” , “ Deneme” sözcüklerini 10 kez seslendirmeleri istenmiştir.

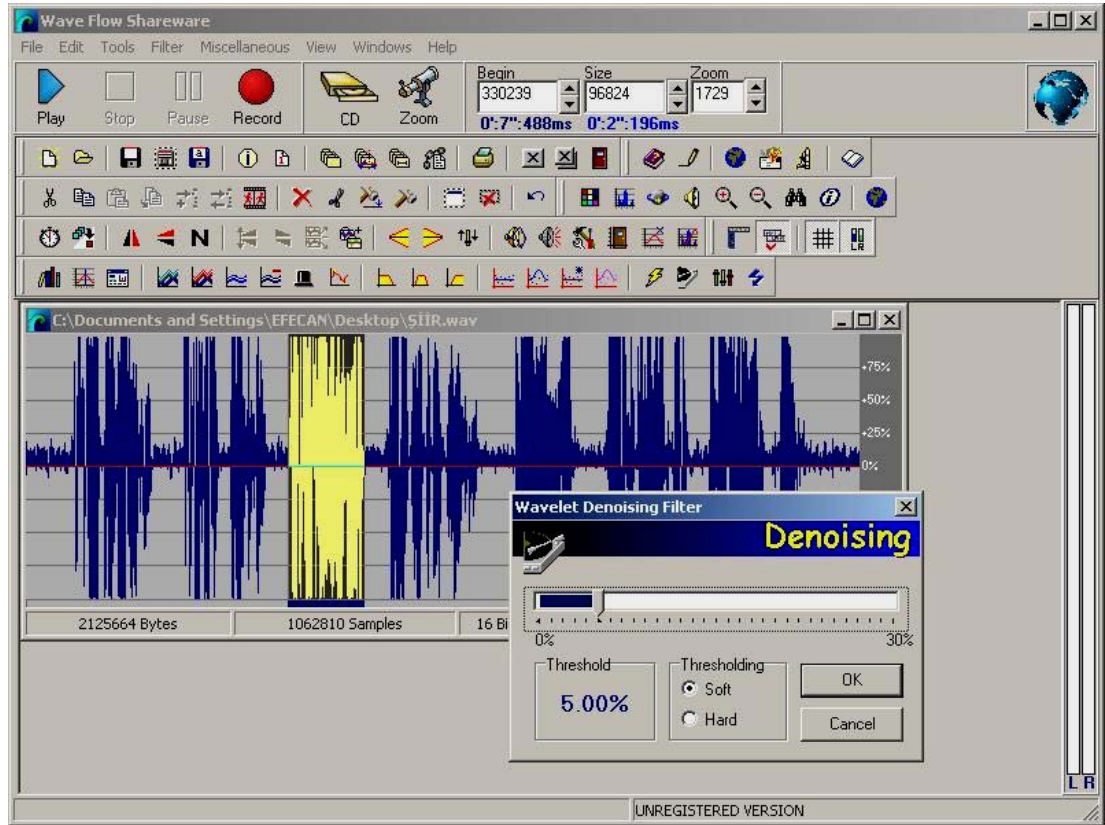
Bilgisayar ve sound blaster pro 16 bit stereo ses kartı ve dinamik mikrofon yardımı ile sesler mono olarak kaydedilmiştir. Mono olarak kaydedilmesinin nedeni kayıt dosyalarının daha az yer tutması içindir. Kayıt frekansı 11025 Hz olarak kullanılmıştır. Kayıt için Wave Flow programı kullanılmıştır (Şekil 3.1).



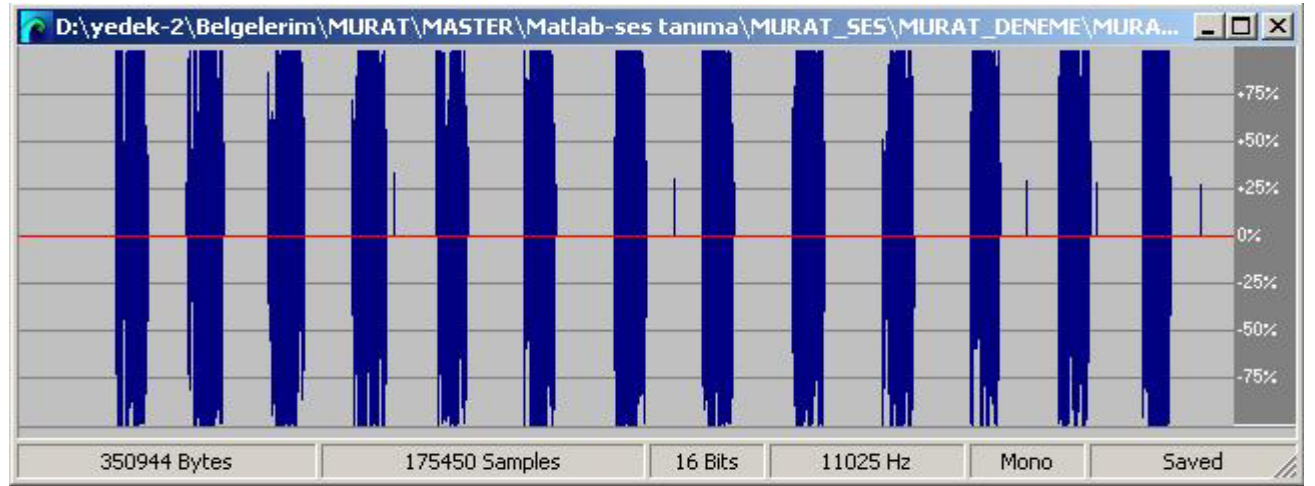
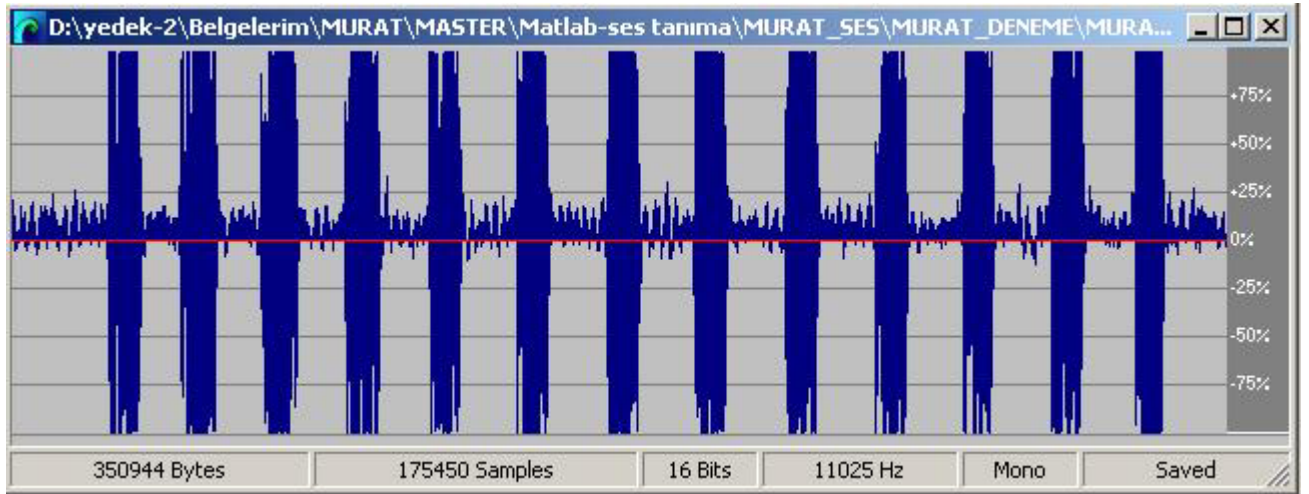
Şekil 3.1 Seslerin Wave Flow programı yardımı ile kaydedilmesi

Kayıt frekansı olarak 22 kHz yada 44 kHz seçilebilirdi ancak ihtiyaç duyulan kayıt alanı miktarı artacağı için 11 KHz olarak seçildi. Telefon hatlarında 3-4 KHz kullanıldığı ve ses tanıma anlamında aşırı kayıplar yaşanmadan konuştuğumuz kişinin kimliğini hala anlayabildiğimiz düşünülürse 11 KHz değerinin oldukça iyi bir değer olduğunu söylemek mümkün olmaktadır. Genel olarak ses ve konuşmacı tanıma sistemleri 8 ila 11 KHz arasında kayıt değerleri kullanmaktadırlar.

Daha sonra bu program yardımı ile ses dosyaları gürültüden temizlenmiş ve kayıt olmayan boş kısımlar kayıtlardan çıkarılmıştır. Seslerin kayıt ortamı normal ev ortamıdır. Tamamıyla olmasa bile bilgisayar fanının sesi hariç sessiz kabul edilebilir. Gürültüden arındırma işlemi esnasında % 5 eşik değeri kullanılmıştır. % 5 eşik değeri denemeler sonunda yeterli olarak görülmüştür. Gürültüden arındırma için kullanılan program ekranı Şekil 3.2’ de verilmiştir. Gürültülü ve gürültüsüz ses dosyası Şekil 3.3 ‘de verilmiştir.



Şekil 3.2 Ses Dosyalarının Gürültüden Temizlenmesi



Şekil 3.3 Gürültülü ve Gürültüsüz Ses Örnekleri

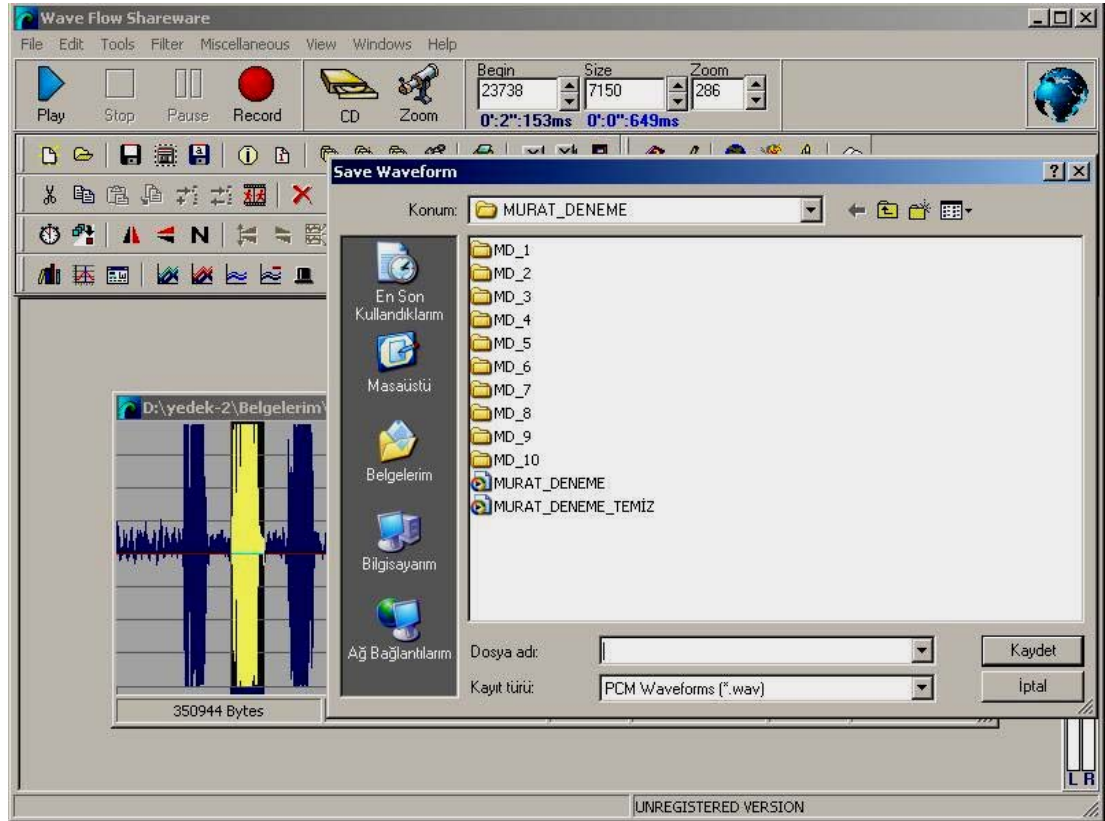
Her konuşmacıya ait ard arda seslendirilen 10 sözcük içeren esas ses kaydı için ses kaydının ilk harfinden itibaren ayırım için gerekli olan kadarı ve seslendirilen sözcük kullanılarak ses dosyalarının adları oluşturulmuştur.(Örneğin; Murat tarafından seslendirilen “ Merhaba “ sözcüğü için dosya adı “ mme”, “ Günaydın “

sözcüğü için dosya adı “mgü” olmuştur. Hasan için “ hasme “ , haluk için “ halme “ adları dosya adları olarak kullanılmıştır. Tüm dosya adları Tablo 3.1 ‘ de verilmiştir.

Tablo 3.1 Ses Dosyaları Adları

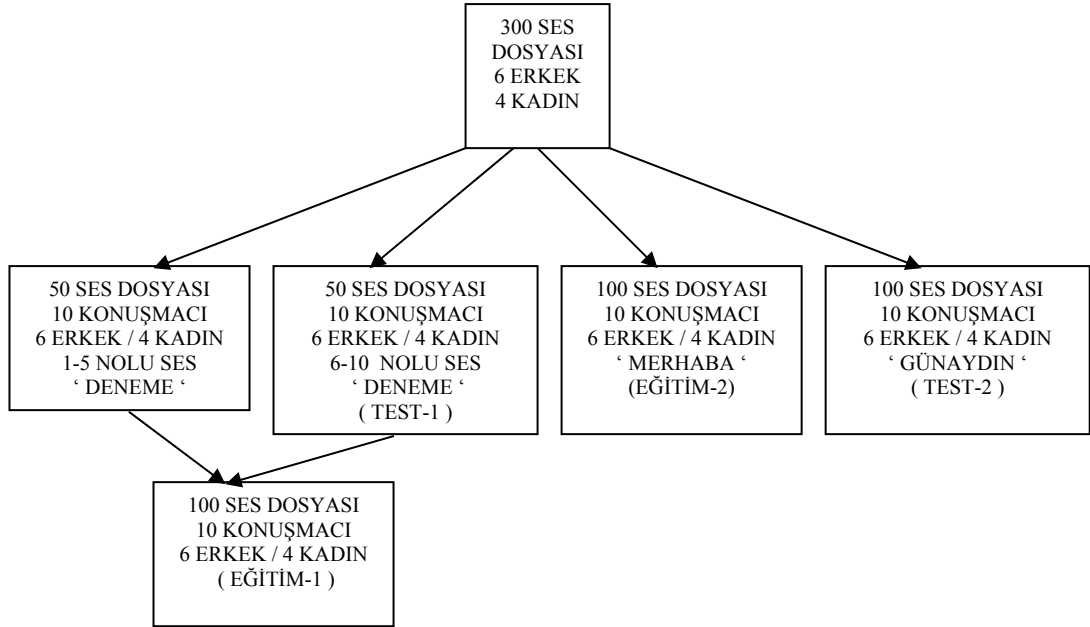
Konuşmacı	DENEME	MERHABA	GÜNAYDIN
MURAT	mde1...mde10	mme1...mme10	mgü1...mgü10
ZÜBEYDE	zde1...zde10	zme1...zme10	zgü1...zgü10
HALUK	hde1...hde10	hme1...hme10	hgü1...hgü10
EDA	ede1...ede10	eme1...eme10	egü1...egü10
HASAN	hasde1...hasde10	hasme1...hasme10	hasgü1...hasgü10
HAYAL	hayde1...hayde10	hayme1...hayme10	haygü1...haygü10
NAİL	nde1...nde10	nmer1...nme10	ngü1...ngü10
BİLGEHAN	bde1...bde10	bme1...bme10	bgü1...bgü10
SENCAN	sde1...sde10	sme1...sme10	sgü1...sgü10
MEHMET	mede1...mede10	meme1...meme10	megü1...megü10

10 adet sözcük içeren her bir dosya tekrar 10 dosya olacak şekilde yine aynı program yardımı ile kesimlenmiştir (Şekil 3.4).



Şekil 3.4 Ses Dosyalarının Elle Kesimlenmesi

Bütün bu işlemler sonunda 10 kişiden 3 farklı 10'ar adet veri ile  $10*10*3= 300$  adet veri dosyası oluşturulmuştur. Eğitim ve test için kullanılacak veri grupları Şekil 3.5'de gösterildiği gibi oluşturulmuştur.



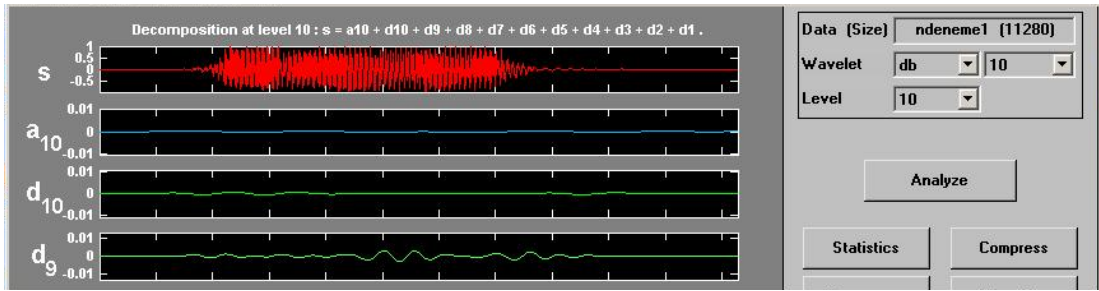
Şekil 3.5 Veri Dosyalarının Gruplandırılması

### 3.3 SESİN KARAKTERİSTİK DEĞERLERİNİN BULUNMASI

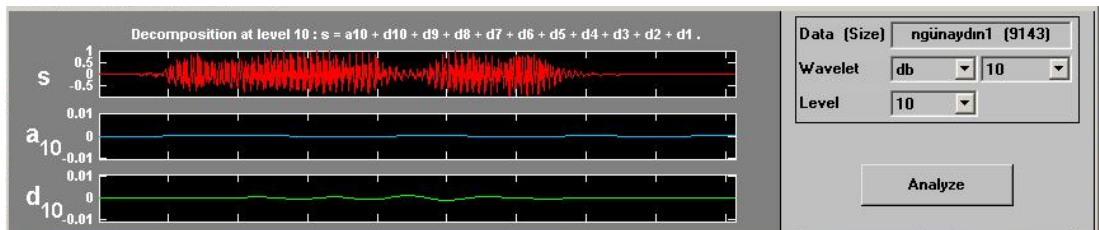
Dosyalanmış ve örneklenmiş ses bilgisinin YSA ile tanıma işlemine geçilebilmesi için veri haline dönüştürülmesi gerekir. Windows altında kayıt yapılan ses örnekleri .wav uzantılı dosyalar haline gelmiştir. Daha sonra bu dosyalar Matlab programı yardımı ile veri dosyaları haline dönüştürülmüştür.

Veri grupları oluşturulmadan önce Matlab Wavelet Toolbox yardımı ile ses sinyalinde analiz çalışmaları yapılmış ve orijinal sinyali en iyi temsil eden sinyalin **Db2 ve 10ncu seviye yaklaşımı** olduğu görülmüştür. Analiz çalışması esnasında sesler kişilere göre ve ayrı gruplar halinde incelenmiş ve en uygun ayrıştırma parametresi ve seviyesi tespit edilmeye çalışılmıştır.

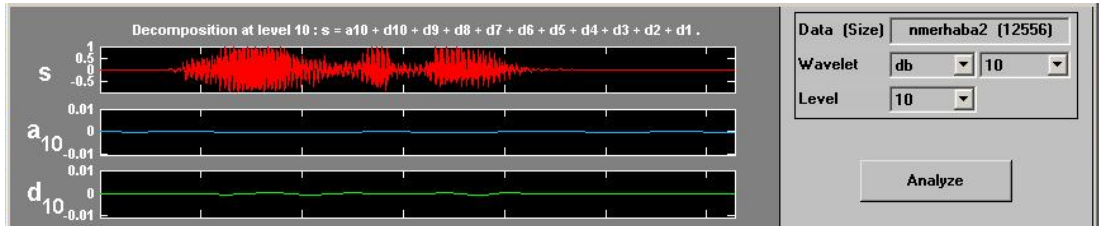
Wavelet toolbox ile yapılan analiz çalışmaları ile ekran görüntüleri aşağıdaki örneklerde verilmiştir (Şekil 3.6-3.10).



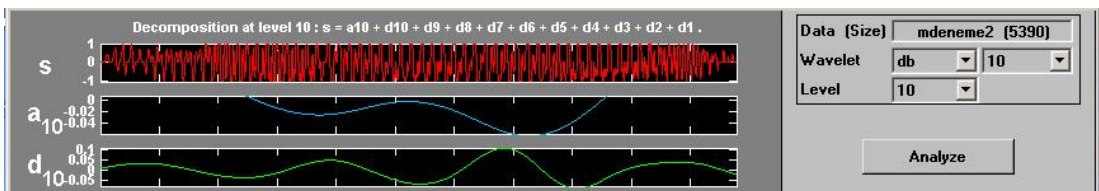
Şekil 3.6.a Nail-Deneme Db10 Level 10



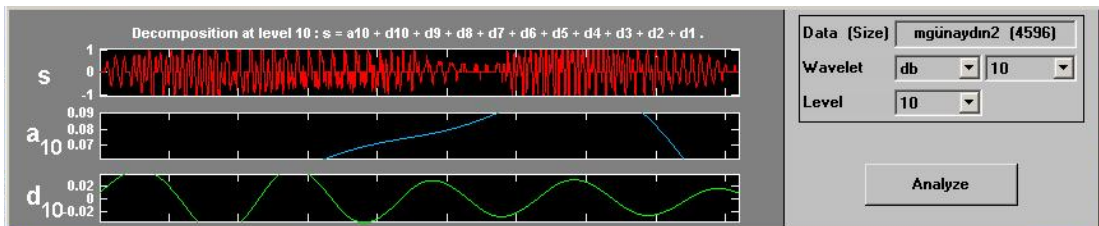
Şekil 3.6.b Nail-Günaydın Db10 Level 10



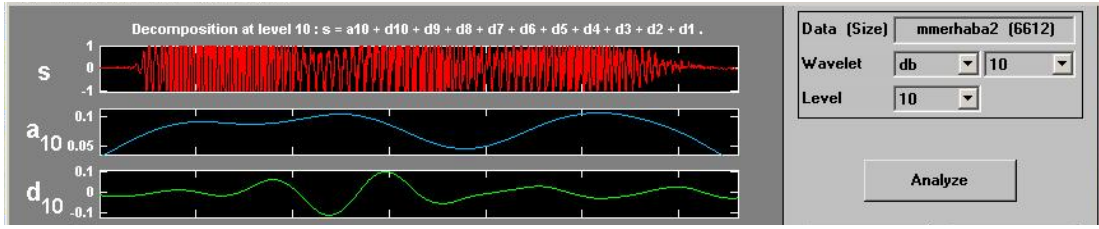
Şekil 3.6.c Nail-Merhaba Db10 Level 10



Şekil 3.7.a Murat-Deneme Db10 Level 10

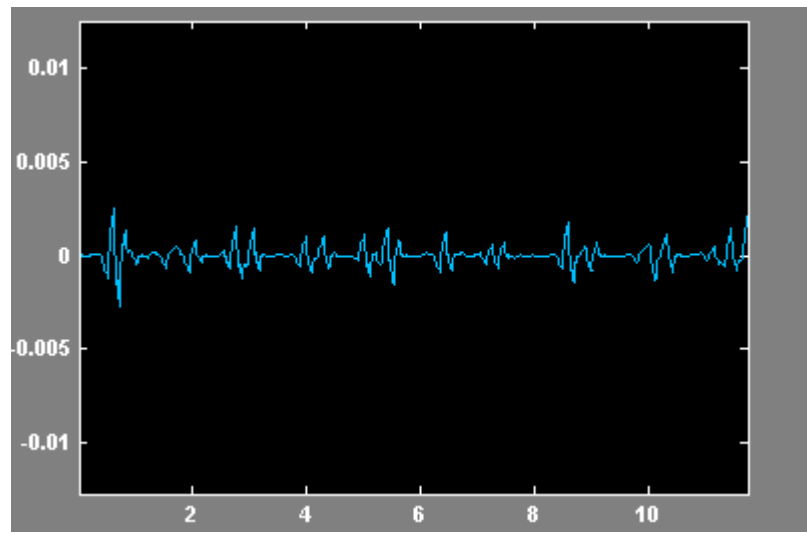


Şekil 3.7.b Murat-Günaydın Db10 Level 10

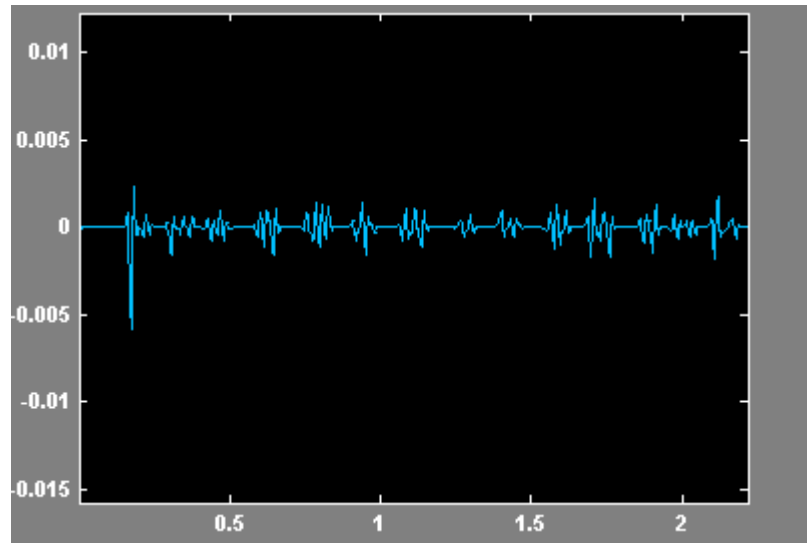


Şekil 3.7.c Murat-Merhaba Db10 Level 10

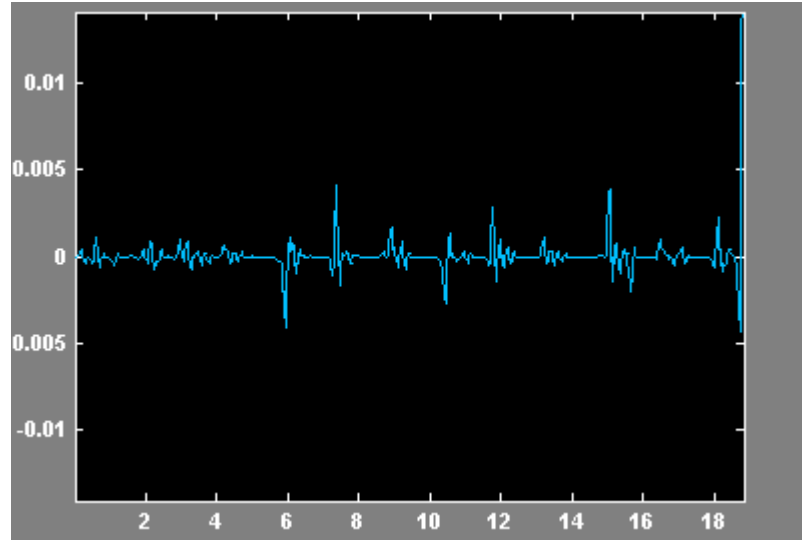
Şekil 3.6 ve Şekil 3.7’den de görüldüğü gibi Db 10 10ncu seviye wavelet yaklaşımı ile aynı kişi tarafından seslendirilen farklı kelimelerin birbirine benzediği ancak farklı kişiler için birbirine benzemediği görülmektedir.



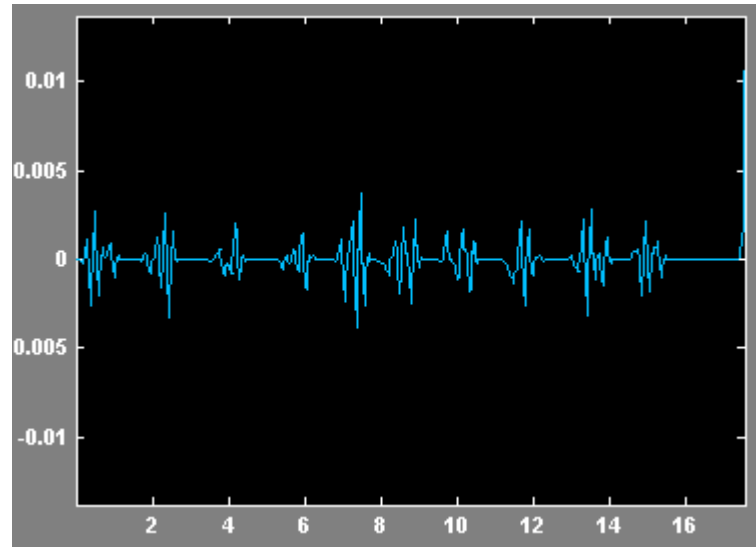
Şekil 3.8.a Bilgehan-Deneme Db2 a10



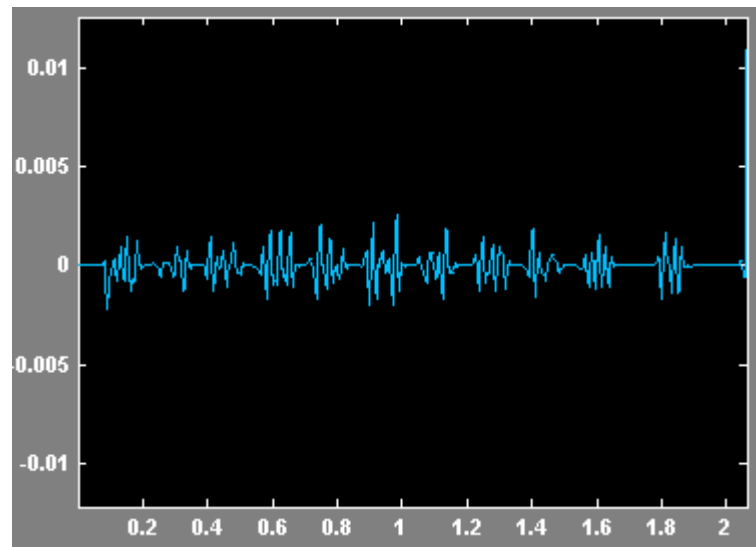
Şekil 3.8.b Bilgehan-Günaydın Db2 a10



Şekil 3.8.c Bilgehan-Merhaba Db2 a10

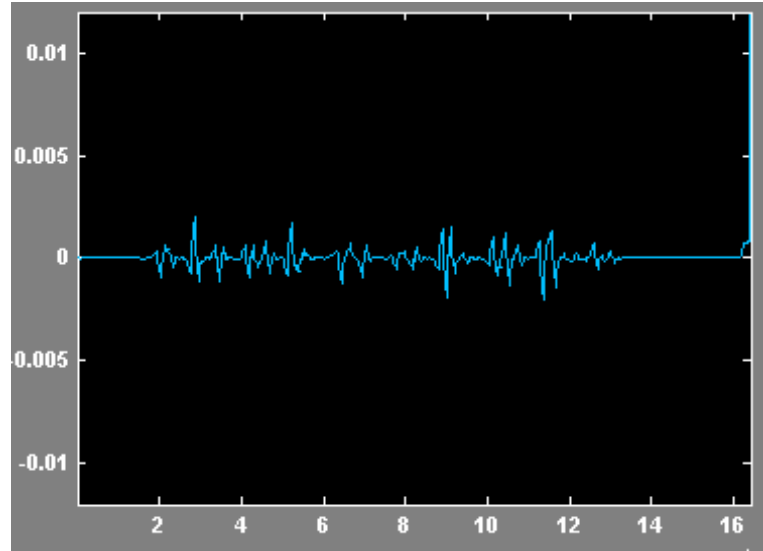


Şekil 3.9.a Zübeyde-Deneme Db2 a10

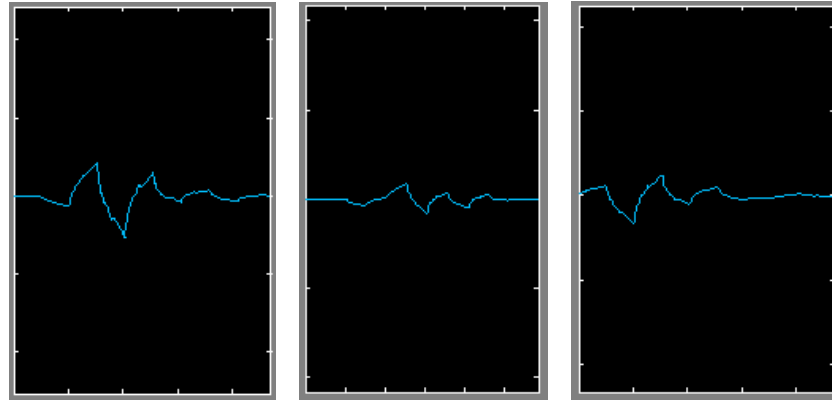


Şekil 3.9.b Zübeyde-Günaydın Db2 a10





Şekil 3.9.c Zübeyde-Merhaba Db2 a10

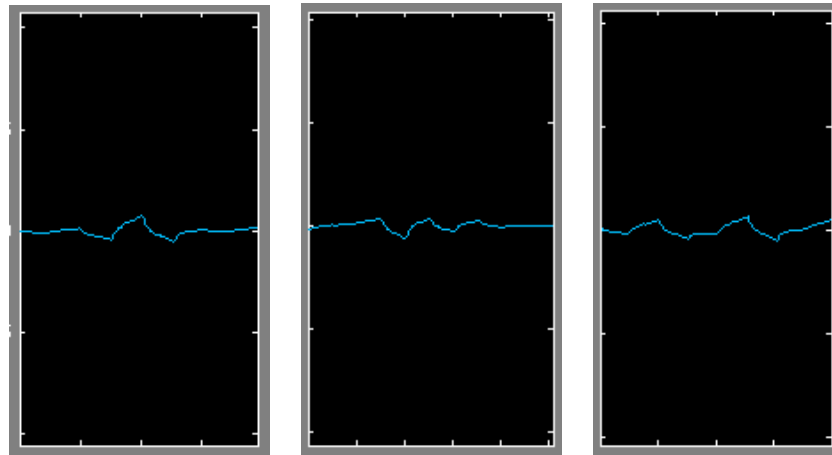


Merhaba

Günaydın

Deneme

Şekil 3.10.a Mehmet'e ait Db2 a10 karakteristiği



Merhaba

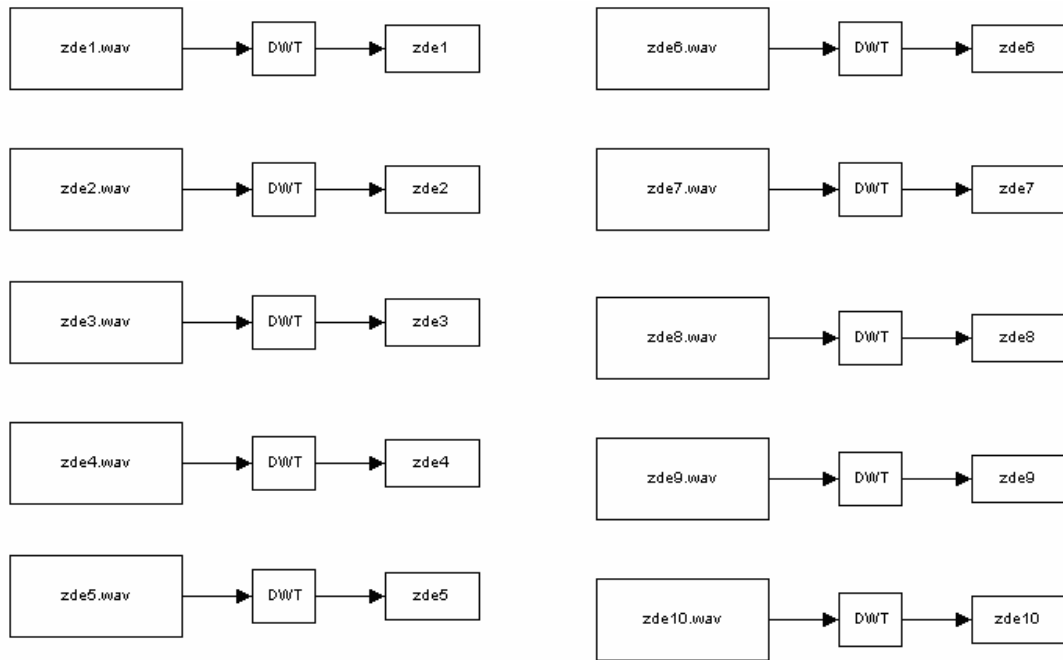
Günaydın

Deneme

Şekil 3.10.b Murat'a ait Db2 a10 karakteristiği

Şekil 3.8, Şekil 3.9 ve Şekil 3.10'dan da anlaşıldığı üzere ses örneklerine ait Db2 ve a10 (Level 10 Approximation) sinyalinin seslerin karakteristiği taşıdığı ve konuşmacı tanımada ses karakteristiği olarak kullanılabilceğini söylemek teorik olarak mümkün görülmektedir.

Bu ön çalışmalardan yola çıkarak tüm ses örnekleri için Db2 a10 sinyali ses örneklerinden elde edilmiş ve çalışmada sınıflandırıcı olarak kullanılacak YSA'nın giriş sinyalleri olarak kullanılmıştır. Wavelet Toolbox ile ses örneklerinden veri hazırlanması Şekil 3.11'de verilmiştir.

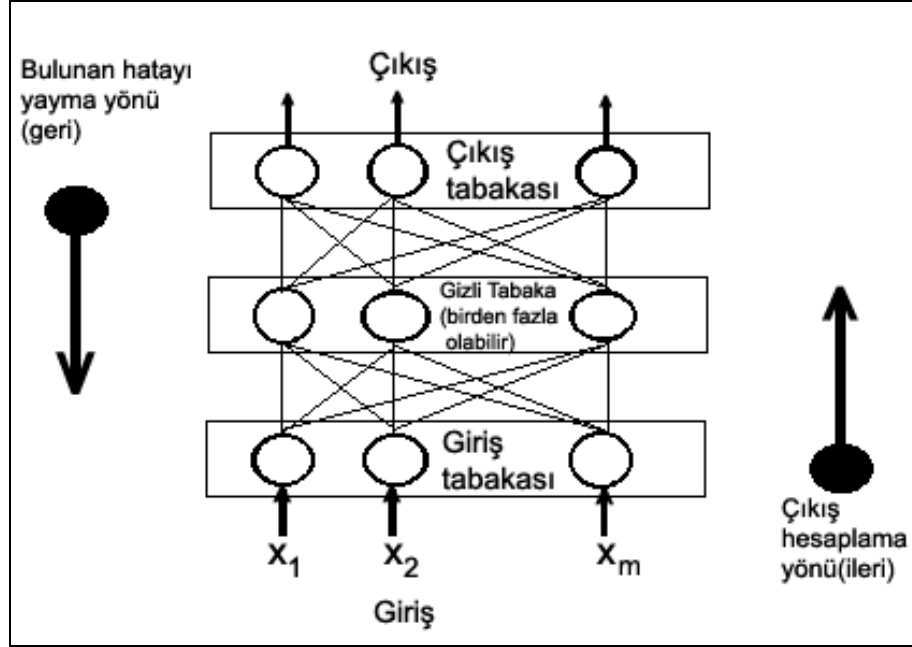


Şekil 3.11 Wavelet Toolbox İle Ses Örneklerinden Veri Hazırlanması

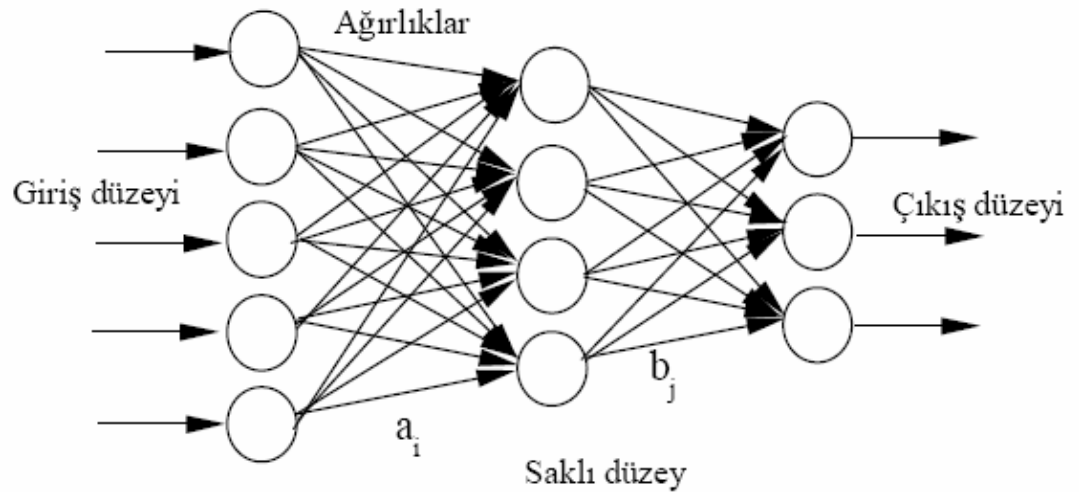
### 3.4 YSA'NIN EĞİTİLMESİ

Hata geri yayımlı yapay sinir ağının Geri yayımlı öğrenme yöntemi ileri besleme aşamasında eğitim için kullanılan girdi, sisteme beslenir ve bunun sonuçları her bir katmanı geçerek çıktı katmanına kadar gelir ve girdiye karşılık bir çıktı elde edilir. Bu aşama sırasında ağ üzerindeki ağırlık değerleri sabit tutulur. İkinci aşama olan geri besleme aşamasında, elde edilen çıktı ile hedef çıktı arasındaki farktan hata sinyali elde edilir ve bu sinyal ağ yapısında geriye doğru yayılır. Bu geri yayılım aşamasında, oluşan hatayı minimuma indirecek şekilde ağırlık değerleri güncellenir.

Bu güdümlü öğrenme işleminin tekrarlanması ile sistem, kendisine beslenen girdi örneklerini öğrenir. Eğitim bittikten sonra sistemin uygulama sürecinde sadece ileri beslemeli ilk aşama yeterli olmaktadır. Bu durumda sistemin eğitilmesi uzun süre almakla beraber, eğitilmiş bir sistemden bilgi alınması çok hızlı olmaktadır



Şekil 3.12 Hata Geri Yayılımlı Yapay Sinir Ağı



Şekil 3.13 Basit İleriye Doğru Beslemeli (Feed Forward) Bir Nöron Ağı

Giriş örnek sayısı = 10

1. ara katmandaki nöron sayısı = 36

2. ara katmandaki nöron sayısı = 9

Çıkış katmandaki nöron sayısı = 10

Yukarıda açıklanan değerlere göre YSA modeli oluşturulmuştur. 1nci ve 2nci ara katmanlardaki nöron sayısının ne olması gerektiği denmeler sonucunda bulunmuştur. Farklı miktarlardaki nöron sayıları ve farklı miktarlardaki ara katmanlarla denemeler sonucunda daha doğru neticeler almak mümkün olabilir.

Dalgacık dönüşümü sonrası elde edilen a10 sinyalleri YSA programını eğitmek üzere giriş verileri olarak kullanılmıştır.

Bu çalışmada kullanılan çok katmanlı YSA programı aşağıda verilmiştir.

% YSA programi

% P: giriş vektörü, T :hedef vektör

```
load zde1.mat;A1 = ans;%ZÜBEYDE
load zde2.mat;A2 = ans;%ZÜBEYDE
load zde3.mat;A3 = ans;%ZÜBEYDE
load zde4.mat;A4 = ans;%ZÜBEYDE
load zde5.mat;A5 = ans;%ZÜBEYDE
load zde6.mat;A6 = ans;%ZÜBEYDE
load zde7.mat;A7 = ans;%ZÜBEYDE
load zde8.mat;A8 = ans;%ZÜBEYDE
load zde9.mat;A9 = ans;%ZÜBEYDE
load zde10.mat;A10 = ans;%ZÜBEYDE
load bde1;B1 = ans;%BILGEHAN
load bde2;B2 = ans;%BILGEHAN
load bde3;B3 = ans;%BILGEHAN
load bde4;B4 = ans;%BILGEHAN
load bde5;B5 = ans;%BILGEHAN
load bde6;B6 = ans;%BILGEHAN
load bde7;B7 = ans;%BILGEHAN
```

load bde8;B8 = ans;%BILGEHAN  
load bde9;B9 = ans;%BILGEHAN  
load bde10;B10 = ans;%BILGEHAN  
load sde1;C1 = ans;%SENCAN  
load sde2;C2 = ans;%SENCAN  
load sde3;C3 = ans;%SENCAN  
load sde4;C4 = ans;%SENCAN  
load sde5;C5 = ans;%SENCAN  
load sde6;C6 = ans;%SENCAN  
load sde7;C7 = ans;%SENCAN  
load sde8;C8 = ans;%SENCAN  
load sde9;C9 = ans;%SENCAN  
load sde10;C10 = ans;%SENCAN  
load mde1;D1 = ans;%MURAT  
load mde2;D2 = ans;%MURAT  
load mde3;D3 = ans;%MURAT  
load mde4;D4 = ans;%MURAT  
load mde5;D5 = ans;%MURAT  
load mde6;D6 = ans;%MURAT  
load mde7;D7 = ans;%MURAT  
load mde8;D8 = ans;%MURAT  
load mde9;D9 = ans;%MURAT  
load mde10;D10 = ans;%MURAT  
load ede1;E1 = ans;%EDA  
load ede2;E2 = ans;%EDA  
load ede3;E3 = ans;%EDA  
load ede4;E4 = ans;%EDA  
load ede5;E5 = ans;%EDA  
load ede6;E6 = ans;%EDA  
load ede7;E7 = ans;%EDA  
load ede8;E8 = ans;%EDA  
load ede9;E9 = ans;%EDA  
load ede10;E10 = ans;%EDA  
load hayde1;F1 = ans;%HAYAL  
load hayde2;F2 = ans;%HAYAL

load hayde3;F3 = ans;%HAYAL  
load hayde4;F4 = ans;%HAYAL  
load hayde5;F5 = ans;%HAYAL  
load hayde6;F6 = ans;%HAYAL  
load hayde7;F7 = ans;%HAYAL  
load hayde8;F8 = ans;%HAYAL  
load hayde9;F9 = ans;%HAYAL  
load hayde10;F10 = ans;%HAYAL  
load hde1;G1 = ans;%HALUK  
load hde2;G2 = ans;%HALUK  
load hde3;G3 = ans;%HALUK  
load hde4;G4 = ans;%HALUK  
load hde5;G5 = ans;%HALUK  
load hde6;G6 = ans;%HALUK  
load hde7;G7 = ans;%HALUK  
load hde8;G8 = ans;%HALUK  
load hde9;G9 = ans;%HALUK  
load hde10;G10 = ans;%HALUK  
load nde1;H1 = ans;%NAIL  
load nde2;H2 = ans;%NAIL  
load nde3;H3 = ans;%NAIL  
load nde4;H4 = ans;%NAIL  
load nde5;H5 = ans;%NAIL  
load nde6;H6 = ans;%NAIL  
load nde7;H7 = ans;%NAIL  
load nde8;H8 = ans;%NAIL  
load nde9;H9 = ans;%NAIL  
load nde10;H10 = ans;%NAIL  
load hasde1;K1 = ans;%HASAN  
load hasde2;K2 = ans;%HASAN  
load hasde3;K3 = ans;%HASAN  
load hasde4;K4 = ans;%HASAN  
load hasde5;K5 = ans;%HASAN  
load hasde6;K6 = ans;%HASAN  
load hasde7;K7 = ans;%HASAN

```

load hasde8;K8 = ans;%HASAN
load hasde9;K9 = ans;%HASAN
load hasde10;K10 = ans;%HASAN
load mede1;L1 = ans; % MEHMET
load mede2;L2 = ans; % MEHMET
load mede3;L3 = ans; % MEHMET
load mede4;L4 = ans; % MEHMET
load mede5;L5 = ans; % MEHMET
load mede6;L6 = ans; % MEHMET
load mede7;L7 = ans; % MEHMET
load mede8;L8 = ans; % MEHMET
load mede9;L9 = ans; % MEHMET
load mede10;L10 = ans; % MEHMET

```

```

P=[A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 B1 B2 B3 B4 B5 B6 B7 B8 B9
B1 C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 D1 D2 D3 D4 D5 D6 D7 D8 D9
D10 E1 E2 E3 E4 E5 E6 E7 E8 E9 E10 F1 F2 F3 F4 F5 F6 F7 F8 F9 F10
G1 G2 G3 G4 G5 G6 G7 G8 G9 G10 H1 H2 H3 H4 H5 H6 H7 H8 H9
H10 K1 K2 K3 K4 K5 K6 K7 K8 K9 K10 L1 L2 L3 L4 L5 L6 L7 L8 L9
L10];

```

```

T=[1 0 0 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0
0;1 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0
0;1 0 0 0 0 0 0 0;1 0 0 0 0 0 0 0;
0 1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0
;0 1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0;0
1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0 0;
0 0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0
;0 0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0;0
0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0 0;
0 0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0
;0 0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0;0
0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0 0;

```

```

0000100000;0000100000;0000100000;0000100000
;0000100000;0000100000;0000100000;0000100000;0
000100000;0000100000;
00000100000;0000010000;0000010000;0000010000
;0000010000;0000010000;0000010000;0000010000;0
000010000;0000010000;
00000010000;0000001000;0000001000;0000001000
;0000001000;0000001000;0000001000;0000001000;0
000001000;0000001000;
00000001000;0000000100;0000000100;0000000100
;0000000100;0000000100;0000000100;0000000100;0
0000000100;0000000100;
00000000100;0000000010;0000000010;0000000010
;0000000010;0000000010;0000000010;0000000010;0
000000001;000000001;
0000000001;0000000001;0000000001;0000000001
;0000000001;0000000001;0000000001;0000000001;0
000000001;0000000001];

```

T=T';

#### % YSA Ağının Kurulması

[R,Q]=size(P); % R : giriş sayısı, Q : giriş örnek sayısı

S1=36; % S1= 1. ara katmandaki nöron sayısı

S2=9; % S2= 2. ara katmandaki nöron sayısı

[S3,Q]=size(T); % S3= çıkış katmandaki nöron sayısı

[w1,b1]=nwtan(S1,R);

[w2,b2]=nwtan(S2,S1);

[w3,b3]=nwtan(S3,S2);

#### % Eğitim Parametreleri

df=100; % tekrar sayısı

sse=0.005; % minimum hata

mitr=80000; % maksimum iterasyon

lr=0.05; % öğrenme oranı (0.01,.....,1----- 0.005)



```

lr_inc=1.05; % öğrenme oranının artırımı
lr_dec=0.005; % öğrenme oranının azaltımı
mr=0.95; % momentum katsayısı (0...1)
er=1.04; % hata oranı -adaptif lr için gerekli- 1.04
tp=[df,mitr,sse,lr,lr_inc,lr_dec,mr,er];

```

YSA'nın eğitilmesi için kullanılan program aşağıda verilmiştir.

%Eğitim Yapılıyor

```

[w1,b1,w2,b2,w3,b3,epocherrors]= trainbpx (w1,b1,'tansig', w2,b2,'tansig', w3 ,
b3 , 'tansig',P,T,tp);

```

TRAINBPX: 0/80000 epochs, lr = 0.05, SSE = 127.501.

TRAINBPX: 100/80000 epochs, lr = 0.0313098, SSE = 0.3438.

TRAINBPX: 200/80000 epochs, lr = 8.46824e-005, SSE = 0.0348938.

TRAINBPX: 300/80000 epochs, lr = 0.0111358, SSE = 0.0283002.

TRAINBPX: 400/80000 epochs, lr = 0.00262814, SSE = 0.0184255.

TRAINBPX: 500/80000 epochs, lr = 0.000791624, SSE = 0.014524.

TRAINBPX: 600/80000 epochs, lr = 0.0188722, SSE = 0.0136209.

TRAINBPX: 700/80000 epochs, lr = 4.86124e-005, SSE = 0.0114102.

TRAINBPX: 800/80000 epochs, lr = 0.00355963, SSE = 0.011115.

TRAINBPX: 900/80000 epochs, lr = 0.00130327, SSE = 0.00940993.

TRAINBPX: 1000/80000 epochs, lr = 0.0557968, SSE = 0.00885104.

TRAINBPX: 1100/80000 epochs, lr = 0.00603261, SSE = 0.00811788.

TRAINBPX: 1200/80000 epochs, lr = 0.00164815, SSE = 0.00725532.

TRAINBPX: 1300/80000 epochs, lr = 0.0580519, SSE = 0.00689142.

TRAINBPX: 1400/80000 epochs, lr = 0.00726575, SSE = 0.00644817.

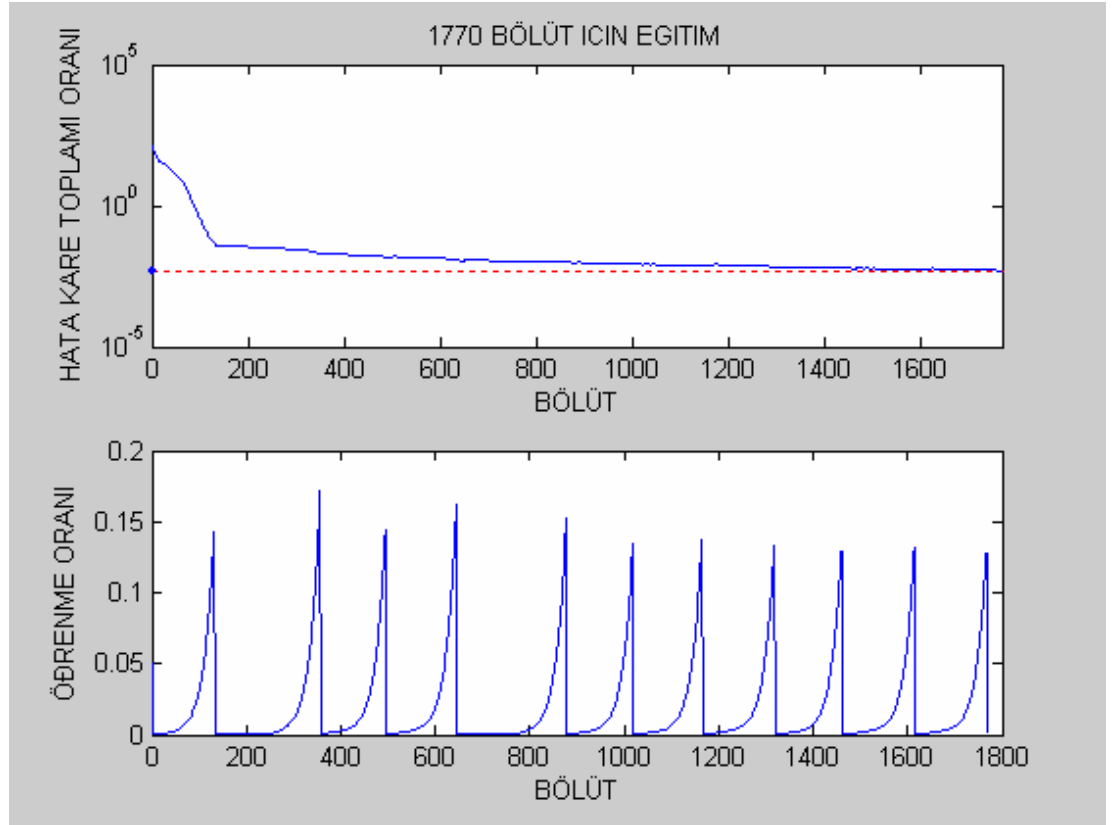
TRAINBPX: 1500/80000 epochs, lr = 0.00163311, SSE = 0.00594342.

TRAINBPX: 1600/80000 epochs, lr = 0.066589, SSE = 0.00563292.

TRAINBPX: 1700/80000 epochs, lr = 0.00592299, SSE = 0.00536523.

TRAINBPX: 1770/80000 epochs, lr = 0.000706018, SSE = 0.00498783.

Hata geri yayımlı yapay sinir ağının eğitim sürecindeki öğrenme iterasyonlarına göre sistem hatasının değişimi Şekil 3.14'de gösterilmiştir. Ağın eğitim süresince herhangi bir kararsızlık ve pürüz olmamıştır.



Şekil 3.14 1770 Bölüt İçin Hata Karelerinin ve Öğrenme Oranının Değişimi

Eğitilen YSA'ya ait parametreler tog.mat dosyasında saklanmaktadır.

save tog.mat w1 b1 w2 b2 w3 b3

Test etmek istediğimiz sinyal tekrar kayıt alanından çağrılmak sureti ile aşağıdaki test programı yardımı ile test edilmektedir.

Test etme;

load mude01.mat;

out=simuff(P,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');

### 3.5 VERİLERİN TEST SONUÇLARI

YSA eğitildikten sonra aşağıdaki program yardımı ile denek gurubundaki veriler test edilmiştir. Örnek olarak Bilgehan tarafından seslendirilen “ GÜNAYDIN “ sözcüğü verilmiştir.

```

% Test etme
% load tog.mat;
load bgü1.mat;
out1=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü2.mat;
out2=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü3.mat;
out3=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü4.mat;
out4=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü5.mat;
out5=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü6.mat;
out6=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü7.mat;
out7=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü8.mat;
out8=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü9.mat;
out9=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');
load bgü10.mat;
out10=simuff(ans,w1,b1,'tansig',w2,b2,'tansig',w3,b3,'tansig');

```

Test sonucunda oluşan çıkış tabloları aşağıda verilmiştir (Tablo 3.2.a-1)

Her konuşmacı için elde edilen 10 adet konuşma örneği Tablolarda 1-10 ile numaralandırılmış olarak toplam 10 sütun halinde verilmiştir. Her sütuna ait değerler test edilen konuşma sinyalinin yukardan aşağıya doğru Zübeyde, Bilgehan, Şencan, Murat, Eda, Hayal, Haluk, Nail, Hasan ve Mehmet'e ait ses örneklerine benzerlik oranını vermektedir. Örneğin Tablo 3.2.a'da birinci satır birinci sütun değeri olan 0.9938 değeri test edilen ses sinyalinin Zübeyde'nin ses sinyaline benzerlik oranını verdiği gibi, birinci satır üçüncü sütun değeri olan 0.0002 değeri test edilen ses sinyalinin Şencan'ın ses sinyaline ne kadar benzediğini göstermektedir. Tablo 3.2'nin oluşmasını sağlayan test sinyalleri Zübeyde'ye ait olduğu için Test sinyallerinin Zübeyde'nin ses sinyalleri ile açık bir şekilde benzerlik

gösterirken diğer ses sinyalleri ile çok düşük bir oranda benzerlik gösterdikleri tablo üzerinde görülmektedir.

Tablo 3.2.a-ı ‘ DENEME ‘ Bağımlı Çıkış Tabloları

Tablo 3.2.a <b>ZÜBEYDE</b>									
1	2	3	4	5	6	7	8	9	10
0.9938	0.9939	0.9938	0.9939	0.9938	0.9939	0.9939	0.9938	0.9938	0.9939
0.0003	-0.0004	0.0002	-0.0008	0.0002	-0.0000	-0.0002	0.0004	0.0004	-0.0001
0.0002	-0.0002	0.0001	-0.0004	0.0002	0.0000	-0.0001	0.0003	0.0003	-0.0001
0.0001	-0.0001	0.0001	-0.0002	0.0001	0.0000	-0.0001	0.0001	0.0002	-0.0001
0.0001	-0.0001	0.0001	-0.0002	0.0001	0.0000	-0.0000	0.0002	0.0002	-0.0000
0.0001	-0.0001	0.0001	-0.0002	0.0001	-0.0000	-0.0000	0.0001	0.0002	-0.0001
0.0001	-0.0001	0.0000	-0.0002	0.0001	-0.0000	-0.0001	0.0001	0.0001	-0.0000
0.0004	-0.0005	0.0002	-0.0010	0.0002	-0.0000	-0.0002	0.0006	0.0006	-0.0001
0.0002	-0.0002	0.0001	-0.0004	0.0001	0.0000	-0.0001	0.0002	0.0003	-0.0001
0.0003	-0.0004	0.0002	-0.0008	0.0002	-0.0000	-0.0002	0.0005	0.0005	-0.0001

Tablo 3.2.b <b>BİLGEHAN</b>									
1	2	3	4	5	6	7	8	9	10
0.0000	-0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.3637
0.9937	0.9937	0.9937	0.9937	0.9937	0.9937	0.9937	0.9937	0.9937	0.3588
-0.0001	-0.0001	0.0000	0.0001	0.0001	-0.0000	0.0000	0.0002	0.0001	0.1645
0.0000	-0.0001	0.0000	0.0000	0.0001	0.0000	0.0001	0.0001	0.0001	0.9230
-0.0001	-0.0001	0.0000	0.0000	0.0000	-0.0000	0.0000	0.0001	0.0000	-0.0338
-0.0000	-0.0001	0.0000	0.0000	0.0000	-0.0000	0.0000	0.0001	0.0001	-0.0027
-0.0000	-0.0001	0.0000	0.0000	0.0000	-0.0000	0.0000	0.0000	0.0000	0.4242
-0.0000	-0.0001	0.0001	0.0001	0.0001	0.0000	0.0001	0.0002	0.0001	-0.0212
-0.0001	-0.0001	0.0000	0.0001	0.0000	-0.0000	0.0001	0.0001	0.0001	0.0829
-0.0001	-0.0001	0.0000	0.0000	0.0000	-0.0000	0.0000	0.0001	0.0000	-0.1626

Tablo 3.2.c <b>ŞENCAN</b>									
1	2	3	4	5	6	7	8	9	10
0.0001	-0.0001	-0.0000	0.0002	0.0001	-0.0002	-0.0002	-0.0001	0.0000	-0.0000
0.0002	-0.0000	0.0000	0.0003	0.0002	-0.0004	-0.0002	-0.0001	0.0001	0.0000
0.9917	0.9918	0.9917	0.9917	0.9917	0.9918	0.9918	0.9918	0.9917	0.9918
0.0001	-0.0000	0.0000	0.0002	0.0001	-0.0002	-0.0002	-0.0000	0.0001	-0.0000
0.0001	-0.0000	0.0000	0.0002	0.0001	-0.0002	-0.0002	-0.0001	0.0000	-0.0000
0.0002	-0.0000	0.0000	0.0004	0.0002	-0.0004	-0.0003	-0.0001	0.0001	0.0000
0.0001	-0.0000	0.0000	0.0002	0.0001	-0.0002	-0.0002	-0.0001	0.0001	-0.0000
0.0001	-0.0000	0.0000	0.0002	0.0001	-0.0002	-0.0002	-0.0000	0.0001	-0.0000
0.0001	-0.0000	0.0000	0.0003	0.0002	-0.0003	-0.0002	-0.0001	0.0001	0.0000
0.0001	-0.0000	0.0000	0.0003	0.0001	-0.0003	-0.0002	-0.0001	0.0001	-0.0000

Tablo 3.2.ç **MURAT**

1	2	3	4	5	6	7	8	9	10
-0.0000	0.0005	0.0002	-0.0003	0.0001	0.0001	-0.0004	-0.0000	-0.0000	-0.0000
-0.0001	0.0003	0.0002	-0.0002	0.0000	0.0001	-0.0003	-0.0000	-0.0000	-0.0001
-0.0001	0.0012	0.0006	-0.0006	0.0001	0.0002	-0.0008	-0.0001	-0.0001	-0.0002
<b>0.9933</b>	<b>0.9932</b>	<b>0.9933</b>	<b>0.9934</b>	<b>0.9933</b>	<b>0.9933</b>	<b>0.9934</b>	<b>0.9933</b>	<b>0.9933</b>	<b>0.9933</b>
-0.0002	0.0016	0.0008	-0.0009	0.0003	0.0003	-0.0011	-0.0002	-0.0003	-0.0003
-0.0001	0.0008	0.0003	-0.0004	0.0001	0.0002	-0.0006	-0.0001	-0.0001	-0.0001
-0.0002	0.0014	0.0007	-0.0007	0.0002	0.0003	-0.0009	-0.0002	-0.0002	-0.0002
-0.0001	0.0005	0.0003	-0.0003	0.0001	0.0001	-0.0004	-0.0000	-0.0001	-0.0001
-0.0002	0.0016	0.0009	-0.0009	0.0002	0.0003	-0.0011	-0.0002	-0.0003	-0.0003
-0.0000	0.0008	0.0004	-0.0004	0.0001	0.0002	-0.0006	-0.0000	-0.0000	-0.0001

Tablo 3.2.d **EDA**

1	2	3	4	5	6	7	8	9	10
-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0001	-0.0001	0.0001	0.0001	0.0001
-0.0000	-0.0001	-0.0000	-0.0000	-0.0000	-0.0001	-0.0001	0.0001	0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	-0.0001	-0.0001	0.0001	0.0000	0.0001
-0.0000	0.0000	-0.0001	-0.0001	0.0001	-0.0001	-0.0001	0.0001	0.0000	0.0001
<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>	<b>0.9924</b>
-0.0000	-0.0000	-0.0001	-0.0000	-0.0000	-0.0001	-0.0001	0.0001	0.0000	0.0000
0.0000	0.0001	0.0000	-0.0000	0.0001	-0.0000	-0.0001	0.0002	0.0001	0.0002
-0.0000	0.0000	-0.0000	-0.0000	0.0001	-0.0001	-0.0001	0.0001	0.0001	0.0001
-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0001	-0.0001	0.0001	0.0001	0.0001
-0.0000	0.0000	-0.0000	-0.0000	0.0001	-0.0001	-0.0001	0.0001	0.0000	0.0001

Tablo 3.2.e **HAYAL**

1	2	3	4	5	6	7	8	9	10
-0.0000	0.0000	0.0001	0.0000	-0.0000	0.0001	0.0000	0.0000	0.0000	0.0000
-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000	0.0000	0.0000	0.0000
-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000	0.0000
-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	-0.0000	0.0000
-0.0000	0.0000	0.0001	-0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000
<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>	<b>0.9925</b>
-0.0000	0.0000	0.0001	-0.0000	-0.0000	0.0001	0.0000	0.0000	0.0000	0.0000
-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000	0.0000
-0.0001	-0.0000	-0.0000	-0.0000	-0.0001	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000
-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000	0.0000

Tablo 3.2.f **HALUK**

1	2	3	4	5	6	7	8	9	10
-0.0000	-0.0000	0.0001	0.0000	-0.0000	0.0004	-0.0007	0.0002	0.0003	-0.0000
-0.0000	-0.0001	0.0000	0.0000	-0.0001	0.0004	-0.0005	0.0002	0.0002	0.0000
-0.0000	-0.0001	0.0000	0.0000	-0.0001	0.0004	-0.0007	0.0002	0.0002	-0.0000
-0.0000	-0.0001	0.0000	0.0000	-0.0001	0.0003	-0.0005	0.0001	0.0001	-0.0001
-0.0000	-0.0001	0.0001	0.0000	-0.0001	0.0005	-0.0007	0.0002	0.0003	0.0000
-0.0000	-0.0001	0.0000	0.0001	-0.0001	0.0006	-0.0010	0.0003	0.0003	-0.0000
<b>0.9930</b>	<b>0.9930</b>	<b>0.9930</b>	<b>0.9930</b>	<b>0.9930</b>	<b>0.9929</b>	<b>0.9931</b>	<b>0.9930</b>	<b>0.9930</b>	<b>0.9930</b>
-0.0000	-0.0001	0.0000	0.0000	-0.0001	0.0004	-0.0007	0.0002	0.0002	-0.0000
-0.0000	-0.0001	0.0001	0.0001	-0.0001	0.0007	-0.0010	0.0003	0.0004	-0.0000
-0.0001	-0.0000	0.0001	-0.0000	-0.0001	0.0003	-0.0005	0.0001	0.0001	-0.0000

1	2	3	4	5	6	7	8	9	10
-0.0000	0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000
-0.0001	-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	0.0000
-0.0000	0.0000	0.0001	0.0001	0.0000	-0.0000	0.0000	0.0000	0.0000	0.0000
-0.0000	0.0000	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	-0.0000	0.0000
-0.0000	0.0000	0.0000	0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000
-0.0000	0.0000	0.0000	0.0001	-0.0000	-0.0000	0.0000	-0.0000	0.0000	0.0000
-0.0000	0.0000	0.0000	0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000
0.9929	0.9929	0.9929	0.9929	0.9929	0.9929	0.9929	0.9929	0.9929	0.9929
-0.0000	0.0000	0.0001	0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	-0.0000
-0.0001	-0.0000	0.0000	0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000

1	2	3	4	5	6	7	8	9	10
-0.0001	-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	-0.0001	-0.0000	-0.0001	-0.0000
-0.0001	-0.0000	0.0000	-0.0000	0.0000	0.0000	-0.0001	-0.0000	-0.0000	0.0000
-0.0001	-0.0000	0.0000	0.0000	0.0000	-0.0000	-0.0001	-0.0000	-0.0000	0.0000
-0.0001	-0.0000	0.0000	0.0000	0.0001	0.0000	-0.0000	0.0000	0.0000	0.0001
-0.0001	-0.0000	0.0000	0.0000	0.0001	0.0000	-0.0001	-0.0000	0.0000	0.0000
-0.0001	-0.0000	0.0000	0.0000	0.0001	0.0000	-0.0000	-0.0000	0.0000	0.0000
-0.0001	-0.0001	-0.0000	-0.0000	-0.0000	-0.0001	-0.0001	-0.0001	-0.0001	-0.0000
-0.0001	-0.0000	0.0000	0.0000	0.0000	0.0000	-0.0001	-0.0000	-0.0000	0.0000
0.9927	0.9927	0.9927	0.9927	0.9927	0.9927	0.9927	0.9927	0.9927	0.9927
-0.0001	-0.0000	0.0000	0.0000	0.0001	0.0000	-0.0000	-0.0000	0.0000	0.0001

1	2	3	4	5	6	7	8	9	10
-0.0001	-0.0001	-0.0000	-0.0001	-0.0000	-0.0000	-0.0001	-0.0000	-0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0001	-0.0000	0.0000	0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	-0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	-0.0000	-0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	0.0000	0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	-0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	0.0000	0.0000	-0.0000
-0.0000	-0.0000	-0.0000	-0.0000	0.0000	0.0000	-0.0000	-0.0000	0.0000	-0.0000
0.9938	0.9938	0.9938	0.9938	0.9938	0.9938	0.9938	0.9938	0.9938	0.9938

Her konuşmacıya ait toplam 10 adet ses sinyalinin test sonucundaki başarı oranları toplanarak aritmetik ortalaması bulundu. Aritmetik ortalama 100 ile çarpılarak ortalama başarı oranından yüzde başarı oranına geçiş yapılmış oldu. Yüzde başarı sonuçları Tablo 3.3'de ayrıca verilmiştir.

Tablo 3.3 Test Sonuçları / Başarı Yüzdeleri

ZÜBEYDE	BİLGEHAN	ŞENCAN	MURAT	EDA	HAYAL	HALUK	NAİL	HASAN	MEHMET
99,38	99,37	99,17	99,33	99,25	99,25	99,30	99,29	99,27	99,38

Test sonucunda Eğitim-1 Grubu ile eğitilen YSA'na Test-1 grubundaki veriler giriş olarak uygulandı ve yukarıda Tablo 3.2 'de verilen sonuçlara ulaşıldı. Bu sonuçlar neticesinde YSA'nın konuşmacıların ' DENEME ' kelimesi temel alınarak eğitilmesi sonucunda, **kelime bağımlı** bir ortam başarısı Tablo 3.3'ün aritmetik ortalaması esas alınmak sureti ile **% 99,30** olarak oluştu.

Tablo 3.4 Kelimededen Bağımsız Test-2 Çıktı Sonuçları

TEST-2									
ZÜBEYDE	BİLGEHAN	ŞENCAN	MURAT	EDA	HAYAL	HALUK	NAİL	HASAN	MEHMET
96,25	87,27	67,57	95,81	94,38	30,79	97,24	81,46	99,38	84,78
85,92	90,27	98,06	97,21	97,78	16,66	36,14	97,46	73,30	91,82
99,38	70,89	66,86	96,37	99,34	21,04	96,08	98,74	99,38	95,67
98,99	78,23	96,67	93,16	97,62	71,41	81,99	97,40	97,79	93,54
95,14	98,67	77,39	98,72	77,50	62,21	82,58	53,33	97,63	78,08
97,69	87,65	95,07	96,21	75,57	39,62	87,75	98,59	98,64	86,88
88,40	98,28	90,68	98,12	72,71	52,40	78,35	49,35	97,18	91,53
88,60	78,77	66,91	98,61	88,52	98,81	98,77	98,80	95,66	97,79
78,20	85,39	94,46	99,15	78,23	42,29	78,25	99,30	96,53	97,64
91,55	99,22	99,60	98,62	99,03	39,27	97,48	99,59	87,79	96,09

Tablo 3.4'de Test-2 Grubu ile yapılan testler neticesinde her konuşmacıya ait toplam 10 adet ses sinyalinin benzerlik oranları her konuşmacı için 10 satırdan oluşan sütunlar halinde verilmiştir. Her konuşmacı için kendisini temsil eden sütuna ait aritmetik ortalamaların yüzde başarı olarak ifadesi yine aynı yolla yapılmış ve sonuçlar Tablo 3.5'de verilmiştir.

Tablo 3.5 Test Sonuçları / Başarı Yüzdeleri

ZÜBEYDE	BİLGEHAN	ŞENCAN	MURAT	EDA	HAYAL	HALUK	NAİL	HASAN	MEHMET
92,01	87,46	85,33	97,20	88,07	47,45	83,46	87,40	94,33	91,38

Kelimededen bağımsız olarak YSA'nın başarı yüzdesinin test edilebilmesi için oluşturulan Test-2 grubu veri YSA' nın girişine veri olarak uygulanmıştır. Sonuçlar Tablo 3. 4 'de gösterilmiştir. Tablo 3.5 esas alınarak

kelimedен bağımsız olarak YSA ‘nın başarı yüzdesi Tablo 3.5’in aritmetik ortalaması esas alınmak sureti ile **% 85.41** olarak bulunmuştur.

Tablo 3.6 Veri Grubuna Dahil Olmayan İçin YSA ‘nın Başarı Yüzdeleri

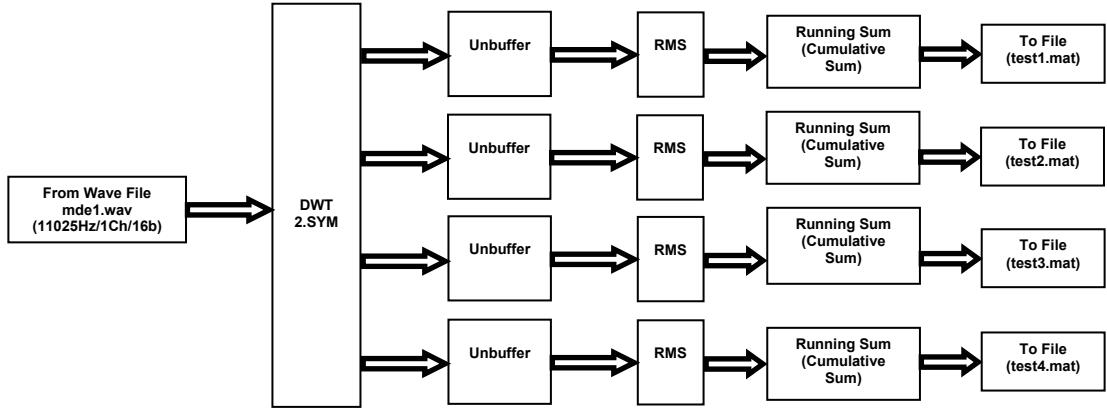
KONUŞMACI-1	KONUŞMACI-2	KONUŞMACI-3
0.3431	0.4170	0.0632
0.2911	0.6226	0.5460
-0.3957	-0.0981	-0.3300
0.9609	-0.0288	0.9567
-0.2899	-0.7202	-0.6770
-0.1938	0.3598	0.3376
0.8842	0.9236	0.9315
-0.0576	0.9182	0.2848
-0.2573	0.0414	-0.2891
0.1035	-0.7122	-0.5152

Tablo 3.6’da herhangi bir veri grubuna dahil olmayan üç farklı konuşmacıya ait ses sinyalleri YSA’ya test verisi olarak verilmiştir. Hem eğitime hem de test gruplarına katılmayan Konuşmacı-1, Konuşmacı-2, Konuşmacı-3 ‘den ‘DENEME‘ kelimesini seslendirmeleri istenmiştir. Bu grup ile yapılan testler neticesinde her konuşmacıya ait toplam 10 adet ses sinyalinin benzerlik oranları her konuşmacı için 10 satırdan oluşan sütunlar halinde verilmiştir. Tablo 3.6 ‘dan çıkan sonuç; YSA’nın kişi bağımlı sistemler için uygun sonuçları üretebildiğidir. Kişi bağımsız olarak değerlendirildiğinde çıkan sonuçların değerlendirilemeyecek seviyede çıktılar olduğu gözlenmiştir.

### 3.6 EK ÇALIŞMALAR

Wavelet analiz çalışmaları esnasında wavelet alt bantlarına ayırma yapılırken; “ Acaba alt bantlardaki herhangi bir değer üzerinden bir sınıflandırma yapmak mümkün olabilir mi? “ sorusu akla gelmiş ve bu yönde bu çalışmadan bağımsız olarak başka bir çalışma da yapılmıştır. Bu yolla da kelime bağımlı sistemler için % 99, kelimedен bağımsız sistemler için % 83 başarı ile sınıflandırma yapılmıştır. Konuşmacı tanımadaki sonraki çalışmayı alt bantlardaki toplam RMS değerleri esas alarak YSA’nın girişine veri olarak vermek bir sonraki çalışmanın fikir alt yapısını oluşturmuştur.





Şekil 3.15 RMS değerlerinin Matlab Sümülük ile elde edilmesi

Wavelet ile 4 alt banda ayrılan ses örnekleri Resim 3.15 ‘daki Matlab Simulink yardımı ile hazırlanmıştır. Tüm konuşmacılara ait ses örneklerinin her alt bant için toplam RMS (Root Mean Square) değerleri Tablo 3.7.a-1’de verilmiştir. Tablolar için satır değerleri olan A,B,C,D ifadeleri 4 farklı alt bandı temsil etmektedir. Her bantın genişliği ses sinyalinin toplam bant genişliğinin  $\frac{1}{4}$  ‘üne karşılık gelmektedir. Örneğin Tablo 3.7.a’da ilk satırda verilmiş olan 115,49,208 ve 1 değerleri aynı şekilde A,B,C,D alt bantlarına ait toplam RMS yani karelerinin kökü değerlerini vermektedir. Burada RMS değerlerini kullanmaktaki amaç sinyalin negatif olan kısımlarının toplama işlemi esnasında işlem hatası oluşmasına neden olmasının engellenmesi içindir.

Tablo 3.7.a-1 Alt Bant RMS Değerleri

Tablo 3.7.a

KAYIT	ZÜBEYDE-DENEME				ZÜBEYDE-MERHABA				ZÜBEYDE-GÜNAYDIN			
	A	B	C	D	A	B	C	D	A	B	C	D
1	115	49	208	1	61	28	216	1	86	45	217	1
2	79	37	189	1	78	31	258	1	73	36	209	1
3	84	38	190	1	64	30	242	1	71	37	209	1
4	86	39	197	1	69	31	268	1	88	44	216	1
5	84	43	210	1	61	30	248	1	78	45	223	1
6	78	42	195	1	79	36	271	1	93	55	234	1
7	92	43	223	1	84	39	300	1	88	52	233	1
8	88	42	192	1	87	35	295	1	97	59	240	1
9	85	41	217	1	83	37	296	1	89	54	237	1
10	95	49	208	1	96	42	316	1	78	44	223	1

Tablo 3.7.b

KAYIT	BİLGEHAN-DENEME			
	A	B	C	D
1	72	36	186	746
2	78	35	214	867
3	72	34	196	808
4	82	44	220	871
5	83	43	216	896
6	79	41	187	779
7	92	44	229	857
8	91	54	209	796
9	55	24	159	738
10	75	39	185	766

BİLGEHAN-MERHABA			
A	B	C	D
72	35	292	964
75	33	297	968
66	29	307	814
67	29	279	772
75	37	331	871
69	31	310	852
65	32	294	913
65	29	276	926
65	28	269	916
85	39	296	819

BİLGEHAN-GÜNAYDIN			
A	B	C	D
56	24	197	963
76	34	243	764
76	35	244	816
67	31	212	819
63	27	214	832
40	18	177	719
47	20	178	970
37	17	174	963
58	27	176	874
37	17	174	892

Tablo 3.7.c

KAYIT	SENCAN-DENEME			
	A	B	C	D
1	16	10	93	652
2	11	7	73	490
3	15	10	92	893
4	12	8	78	607
5	11	7	73	514
6	20	10	79	519
7	18	10	110	926
8	14	8	82	687
9	14	8	90	591
10	12	7	83	598

SENCAN-MERHABA			
A	B	C	D
22	11	101	599
23	13	112	675
27	13	120	765
20	11	119	671
18	10	102	711
18	11	108	831
21	11	83	682
28	12	120	772
33	13	129	758
15	9	100	733

SENCAN-GÜNAYDIN			
A	B	C	D
12	7	79	677
17	8	90	671
16	9	85	710
13	8	74	646
15	8	83	703
14	8	78	646
13	7	80	679
10	7	71	646
11	6	76	658
13	7	76	657

Tablo 3.7.ç

KAYIT	MURAT-DENEME			
	A	B	C	D
1	4	1	6	152
2	6	2	9	219
3	3	1	5	125
4	3	1	4	116
5	4	2	6	154
6	4	1	7	141
7	6	2	9	139
8	6	2	9	154
9	9	2	11	146
10	4	1	6	155

MURAT-MERHABA			
A	B	C	D
4	3	17	188
8	4	23	226
5	3	16	207
4	2	15	225
4	3	14	182
4	3	13	161
4	3	14	170
4	3	13	153
2	2	8	135
3	2	8	171

MURAT-GÜNAYDIN			
A	B	C	D
17	4	20	271
5	2	10	206
4	1	5	174
4	1	5	166
5	2	8	196
9	2	14	255
11	3	13	255
14	4	16	244
11	3	15	232
10	3	15	245

Tablo 3.7.d

KAYIT	EDA-DENEME			
	A	B	C	D
1	33	10	73	351
2	41	7	66	560
3	26	10	78	470
4	30	11	78	444
5	23	11	81	453
6	25	4	37	420
7	46	15	106	657
8	36	6	58	502
9	11	8	50	442
10	9	6	37	406

EDA-MERHABA			
A	B	C	D
57	21	155	647
102	57	313	687
79	60	219	708
78	41	222	627
64	20	176	602
66	44	176	605
54	28	166	514
46	40	124	555
56	19	145	637
55	28	176	534

EDA-GÜNAYDIN			
A	B	C	D
24	12	83	333
36	20	127	516
21	9	73	406
18	4	30	388
28	12	84	409
24	12	78	445
20	9	69	445
23	16	64	427
23	11	75	371
43	16	119	368

Tablo 3.7.e

KAYIT	HAYAL-DENEME			
	A	B	C	D
1	40	14	101	437
2	68	23	172	562
3	58	20	144	443
4	61	22	152	475
5	50	21	151	467
6	69	25	178	537
7	54	19	145	454
8	52	18	140	466
9	46	16	118	447
10	45	14	115	441

HAYAL-MERHABA			
A	B	C	D
33	19	154	551
60	31	248	762
45	24	174	683
41	21	185	623
37	20	167	632
33	18	157	509
34	20	171	586
33	18	143	514
28	16	138	528
36	16	163	522

HAYAL-GÜNAYDIN			
A	B	C	D
41	17	137	336
63	32	223	593
62	31	201	539
63	28	170	466
48	20	133	412
68	34	210	456
47	23	149	404
42	20	145	377
45	21	154	363
41	17	142	394

Tablo 3.7.f

KAYIT	HALUK-DENEME			
	A	B	C	D
1	12	7	33	473
2	21	11	51	722
3	9	5	26	605
4	8	4	24	395
5	16	7	36	509
6	15	6	32	510
7	12	3	17	355
8	11	5	30	464
9	12	6	32	438
10	11	4	21	422

HALUK-MERHABA			
A	B	C	D
16	9	51	429
11	5	26	364
4	4	20	298
9	6	27	357
14	7	40	475
14	7	44	501
12	7	38	426
13	5	31	468
15	6	37	468
12	8	42	478

HALUK-GÜNAYDIN			
A	B	C	D
9	4	23	424
11	5	3	419
8	4	20	341
9	4	25	405
7	3	18	391
6	3	15	394
12	6	33	491
9	4	22	456
9	5	25	487
11	6	31	471

Tablo 3.7.g

KAYIT	NAİL-DENEME			
	A	B	C	D
1	55	16	27	767
2	77	24	204	973
3	54	16	143	860
4	62	64	144	834
5	64	20	155	826
6	53	17	130	867
7	53	16	128	911
8	46	15	215	756
9	51	16	134	850
10	75	21	175	936

NAİL-MERHABA			
A	B	C	D
78	28	199	859
92	29	202	840
69	22	188	838
67	21	166	834
81	25	192	833
70	23	182	829
86	27	208	895
71	23	162	778
89	27	181	844
89	27	181	844

NAİL-GÜNAYDIN			
A	B	C	D
59	18	143	804
37	14	117	799
42	15	116	734
51	16	134	719
51	17	132	791
50	16	131	725
41	15	124	707
48	16	121	778
43	16	126	775
48	15	130	759

Tablo 3.7.h

KAYIT	HASAN-DENEME			
	A	B	C	D
1	11	6	42	408
2	24	11	94	678
3	12	6	42	527
4	14	7	48	478
5	16	7	53	476
6	6	3	23	373
7	16	7	53	442
8	13	7	45	545
9	16	8	55	510
10	15	8	61	512

HASAN-MERHABA			
A	B	C	D
33	19	154	551
60	31	248	762
45	24	174	683
41	21	185	623
37	20	167	632
33	18	157	509
34	20	171	586
33	18	143	514
28	16	138	528
36	18	163	522

HASAN-GÜNAYDIN			
A	B	C	D
41	17	137	336
63	32	223	593
62	31	201	539
63	28	170	466
48	20	175	412
68	34	210	456
47	23	149	404
42	20	145	377
45	12	154	363
41	17	142	394

Tablo 3.7.1

KAYIT	MEHMET-DENEME			
	A	B	C	D
1	34	11	61	797
2	19	7	48	907
3	29	10	51	757
4	24	9	53	921
5	25	9	55	912
6	31	9	59	791
7	30	10	54	811
8	31	9	55	860
9	34	11	57	807
10	27	9	48	823

MEHMET-MERHABA			
A	B	C	D
28	14	100	973
29	14	101	947
33	18	99	980
33	17	101	967
26	16	101	954
33	17	104	965
32	18	110	953
34	18	120	956
29	16	103	964
28	15	101	973

MEHMET-GÜNAYDIN			
A	B	C	D
32	9	60	955
26	11	85	841
32	12	79	834
23	9	53	735
21	10	48	825
24	9	57	873
27	8	55	912
26	9	65	982
18	8	54	964
90	9	57	796

Wavelet ile 4 alt banda ayrılan ses örnekleri her banddaki toplam RMS değerlerinin alt ve üst değerleri sınır koşul değerler olarak kullanılmış ve algoritma ile sınıflandırma başarısı değerlendirilmiştir. Her konuşmacının 10 farklı ses sinyalinden elde edilen dört farklı alt bant için toplam RMS değerlerinden her konuşmacı için en düşük ve en yüksek değer esas alınarak her dört bant için de ayrı ayrı sınır değerler oluşturulmuştur. Böylece her konuşmacı için farklı değer aralığına sahip dört farklı alt bant sınır koşulları elde edilmiştir. Test edilecek ses sinyalinin bu sınır değerler arasında olup olmadığına bakılarak her konuşmacı için benzerlik durumları araştırılmıştır. Örneğin sırası ile 90,40,210,730 değerlerinden oluşan alt bant RMS değerleri Bilgehan için oluşturulmuş sınır değerler içindedir ve bu ses sinyalinin Bilgehan'a ait olduğunu söylemek mümkündür. Ancak bu değerler A,B ve C alt bantları için Zübeyde'nin sınır koşullarını sağlarken Zübeyde için tanımlı D alt bandının sınır aralığında yer almamaktadır. Dolayısı ile bu sesin Zübeyde'ye ait olmadığını söylemek mümkün olmaktadır. Bu genel algoritma mantığı içinde tüm ses örnekleri değerlendirilmiş ve ortaya çıkan sonuçlar Tablo 3.8'de verilmiştir.

Tablo 3.8 Kelime Bağımlı Algoritma Sonuçları

	TOPLAM RMS				TOPLAM HATA	% TOPLAM HATA	% BAŞARI
	TANIMLI ALT BAND KOŞULLARI						
	A BANDI	B BANDI	C BANDI	D BANDI			
BİLGEHAN	72-92	34-44	185-229	746-896	1	10	90
EDA	9-36	6-11	37-81	351-502	1	10	90
HALUK	8-16	3-7	21-36	355-605	2	20	80
HASAN	6-16	6-8	23-55	373-527	5	50	50
HAYAL	40-68	14-25	101-178	441-562		0	100
MURAT	4-9	1-2	4-11	116-219		0	100
NAİL	53-75	15-24	115-204	756-973		0	100
SENCAN	11-20	7-10	73-110	514-893		0	100
ZÜBEYDE	79-115	37-49	189-223	1000-		0	100
MEHMET	19-34	7-11	48-61	791-912		0	100
KİŞİLERİN 10 ADET " DENEME SÖZCÜĞÜNÜN HATA SAYISI "							<b>91</b>
" DENEME " SÖZCÜĞÜ ESAS ALINARAK HAZIRLANMIŞTIR							
** ALT BANTLARIN EN BÜYÜK VE EN KÜÇÜK DEĞERLERİ ALINARAK DEĞER ARALIĞI İLGİLİ ALT BAND İÇİN BELİRLENMİŞTİR.							

Tablo 3.8 ‘de yapılan çalışma ile örnek olarak Bilgehan için tanımlı koşul alt bant RMS değerleri diğer tüm konuşmacıların ‘ DENEME ‘ esas alınarak hazırlanmış tanımlı alt bant koşulları ile karşılaştırılmış ve Bilgehan’ının ses örneği haricinde kalan toplam 90 ses örneğinden sadece birine benzediği görülmüştür. Bu şekilde diğer örnekler de değerlendirilmiştir. **Kelime bağımlı** olarak sınıflandırma başarısı **% 91** olmuştur.

Tablo 3.9 Kelime Bağımlı Algoritma Sonuçları

	TOPLAM EN AZ VE EN ÇOK RMS				İÇ HATA ( ADET )	% İÇ HATA	DIŞ HATA ( ADET )	% DIŞ HATA	TOPLAM HATA ( ADET )	% TOPLAM HATA	% BAŞARI
	TANIMLI ALT BAND KOŞULLARI										
	A BANDI	B BANDI	C BANDI	D BANDI							
BİLGEHAN	37-92	17-54	185-229	738-968	4	13	4	5	8	3	73
EDA	20-79	4-16	50-222	333-687	6	20	3	7	9	3	70
HALUK	11-16	3-8	15-45	341-722	3	10	3	4	6	2	80
HASAN	11-19	6-11	42-104	373-820		0	7	0	7	2	77
HAYAL	33-68	14-34	133-185	363-683	5	17	4	6	9	3	70
MURAT	3-10	1-4	4-20	116-271	2	7		2	2	1	93
NAİL	43-89	14-29	114-208	707-936	3	10	1	4	4	1	87
SENCAN	11-28	7-13	73-129	514-926	3	10	4	4	7	2	77
ZÜBEYDE	80-95	37-49	189-316	1000-1350		0		0	0	0	100
MEHMET	21-34	7-18	51-120	735-982	2	7	3	2	5	2	83
TÜM SÖZCÜKLER ESAS ALINARAK HAZIRLANMIŞTIR										<b>83</b>	
** ALT BANTLARDAKİ TOPLAM EN BÜYÜK VE EN KÜÇÜK RMS DEĞERLERİ ALINARAK DEĞER ARALIĞI İLGİLİ ALT BAND İÇİN BELİRLENMİŞTİR.											

Tablo 3.9 'da yapılan çalışma ile örnek olarak Bilgehan için tanımlı koşul alt bant RMS değerleri tüm ses örnekleri esas alınarak hazırlanmış diğer tüm konuşmacıların tanımlı alt bant koşulları ile karşılaştırılmış ve Bilgehan'nın ses örneği haricinde kalan toplam 290 ses örneğinden % 73 oranında başarı elde edilmiştir. Bu şekilde diğer örnekler de değerlendirilmiştir. **Kelime bağımsız** olarak sınıflandırma başarısı **% 83** olmuştur.

#### 4.SONUÇLAR VE TARTIŞMA

Ses işleminin alt basamaklarından biri olan Konuşmacı tanıma için birçok yöntem kullanılmaktadır. Bunlardan en fazla kullanılanı DÖK ( Doğrusal Öngörüm Kodlaması)‘dır. YSA‘nın ses işleme ve Konuşmacı tanımda kullanılması ile çok daha hızlı ve doğru sonuçlara ulaşmak mümkün olabilmiştir. Tüm bu çalışmalar insan sesinin ses oluşumunda etkin rolü olan gırtlak ve ses yolu yapısının farklı olduğunu ve kişi için zaman zaman çeşitli nedenlerden dolayı ( hastalık gibi.) yapısında değişiklik olsa bile esas itibarı ile değişmediği ve her kişi için ifade edilebilecek bir karakter taşıdığıdır.

Bu çalışmada amaçlanan; mikrofon yardımı ile kayıt edilen ses sinyallerinden wavelet ve yapay sinir ağı yardımı ile konuşmacı kimliğinin tespit edilmesi olmuştur. Bu amaç doğrultusunda ses sinyalleri pc ortamına mikrofon yardımı ile alınmış, ön işleme ve gürültüden temizleme yapıldıktan sonra Matlab tabanlı hazırlanan model yardımı ile hazırlanan ses dosyalarından veriler elde edilmiş ve yine Matlab ortamında hazırlanan YSA programı ile sınıflandırma işlemi gerçekleştirilmiştir.

YSA‘nın konuşmacıların‘ DENEME ‘ kelimesi temel alınarak eğitilmesi sonucunda, **kelime bağımlı** bir ortam başarısı **% 99,30** olarak oluşmuştur.

**Kelimededen bağımsız** olarak YSA ‘nın başarı yüzdesi **% 85.41** olarak bulunmuştur.

Hem eğitime hem de test gruplarına katılmayan Konuşmacı-1, Konuşmacı-2, Konuşmacı-3 ‘den ‘ DENEME ‘ kelimesini seslendirmeleri istenmiştir. Bu konuşmacılara ait ses örneklerinden elde edilen veriler YSA‘a giriş olarak uygulanmış ve sonuçlar Tablo 3.4 ‘de verilmiştir. Tablo 3.4 ‘den çıkan sonuç; hazırlanmış olan YSA modelinin kişi bağımlı sistemler için uygun sonuçları üretebildiğidir. Kişi bağımsız olarak değerlendirildiğinde çıkan sonuçların değerlendirilemeyecek seviyede çıktılar olduğu gözlenmiştir.



Wavelet analiz çalışmaları esnasında wavelet alt bantlarına ayırıştırma yapılırken; acaba alt bantlardaki herhangi bir değer üzerinden bir sınıflandırma yapmak mümkün olabilir mi? Sorusu akla gelmiş ve bu yönde bu çalışmadan bağımsız olarak başka bir çalışma da yapılmıştır. Sonuçlar Bölüm 3' de ayrıntılı olarak anlatılmıştır. Bu yolla da kelime bağımlı sistemler için % 99, kelimedan bağımsız sistemler için % 83 başarı ile sınıflandırma yapılmıştır. Konuşmacı tanımadaki sonraki çalışmayı alt bantlardaki toplam RMS değerleri esas alarak YSA'nın girişine veri olarak vermek bir sonraki çalışmanın fikir alt yapısını oluşturmuştur.

Yapılan bu çalışma ile oldukça başarılı bir oranda Wavelet ve YSA kullanarak ses sinyallerinden Konuşmacı Kimliğinin Tespit edilmesi sağlanmıştır.

## KAYNAKLAR

- [1] **ROBINSON**, T., 1995. Speech Analysis. Lecture Notes.
- [2] **SELEN**, N., 1979, Söyleyis sesbilimi, akustik sesbilim ve Türkiye türkçesi, Türk Dil Kurum Yayınları.
- [3] **MENGÜŞOĞLU**, E., 1999 “Bir Türkçe Sesli İfade Tanıma Sisteminin Kural Tabanlı Tasarımı ve Gerçekleştirimi”, Master Tezi, Hacettepe Üniversitesi.
- [4] **KUŞ**, P., 1998, “Ses Sinyallerinin Düşük Hızda İletimi”, Master Tezi, Hacettepe Üniversitesi.
- [5] **RABINER** L. R., **JUANG** B. H., 1993, “Fundamentals of Speech Recognition”, Prentice Hall, Englewood Cliffs.
- [6] **RABINER**, L. R., 1989. A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedins of IEEE, vol. 77, no. 2, pp. 257-286.
- [7] **FURUI**, S., 1989. Digital Speech Processing, Synthesis, and Recognition. Marcel Dekker Inc.
- [8] **ROBINSON**, T., Fallside, F. 1991. A recurrent error propagation network speech recognition system. Computer Speech and Language, vol 5, no. 3, pp. 259-274.
- [9] **KOHONEN**, T., 1988. An introduction to neural computing. Neural Networks, vol 1, pp.3-16.
- [10] **TEBELSKIS**, J., 1995, “Speech Recognition Using Neural Networks”, PhD. Thesis, Carnegie Mellon University, Pittsburgh Pennsylvania.
- [11] **ERTAŞ** F., **ESKİDERE** Ö., 2001, “Yazılım Tabanlı Sözcük Sentezleyici”, DEÜ Mühendislik Fakültesi Fen ve Mühendislik Dergisi, Cilt:3, **Sayı:1**.
- [12] **SOMERVUO**, P., 1996, Speech Recognition using context vectors and multiple feature streams, Master’s Thesis.
- [13] **ARTUNER**, H., 1994, Bir Türkçe Fonem Kümeleme Sistemi Tasarımı ve Gerçekleştirimi.
- [14] **İNAL**, M, 2001, “Yapay Sinir Ağları Tabanlı Konuşmacı Tanıma “, Doktora Tezi, Kocaeli Üniversitesi.
- [15] **KOHONEN**, T., et. al., 1996, LVQ\_PAK: The Learning Vector Quantization Program Package, Report A30.

- [16] **KOÇ**, A., 1999, “Acoustic Feature Analysis for Robust Speech Recognition”, MSc. Thesis, Boğaziçi University.
- [17] **YÜCETÜRK**, A. C., 2000, “Yapay Sinir Ağları Kullanılarak Örüntü Sınıflandırma ve Tanıma”, Doktora Tezi, Ege Üniversitesi.
- [18] **ERDEM** , T., 2004, “ Yapay Sinir Ağları Kullanarak Konuşmacı Tanıma Uygulaması “ , Yüksek Lisans Tezi, Niğde Üniversitesi.
- [19] **KOHONEN**, T., et. al., 1996, SOM\_PAK: The Self Organizing Map Program Package, Report A31.
- [20] **RABINER**, L. R., Schafer, R., W., 1978. Digital processing of speech signals. Prentice Hall Inc.
- [21] **PICONE**, J., W., 1993. Signal modelling techniques in speech recognition. Proceedings of the IEEE, vol. 81, no. 9, pp. 1215-1247.
- [22] **HERMANSKY**, H., 1990. Perceptual Linear Predictive (PLP) Analysis of Speech. Journal Of Acoustic Society of America. vol. 87, no. 4. pp. 1738-1752.
- [23] **HERMANSKY**, H., Morgan, N., 1994. RASTA processing of speech. IEEE Transactions on Speech and Audio Processing, vol. 2, no. 4, pp. 578-589.
- [24] **LEE**, K. F., 1988. Large-vocabulary speaker-independent continuous speech recognition: The SPHINX system. Technical Report CS-88-148, Carnegie Mellon University.
- [25] **DEMİR**, H, 2000, “ Wavelet Dönüşümü Ve İşaret İşleme Uygulamaları”, Yüksek Lisans Tezi , İstanbul Üniversitesi.
- [26] **SYDRAL** A., BENNETT R., GREENSPAN S., 1995, “Applied Speech Technology”, CRC Press.
- [25] **SEVEN**, A, 1997, “ Small Vocabulary Word And Speaker Recognition Using Artificial Neural Networks”, Master Tezi, Boğaziçi Üniversitesi.
- [27] **KOHONEN**, T., 1990. The self-organizing map. Proceedings of IEEE, vol 78, pp. 1464-1480.
- [28] **ELMAS**, Ç., 2003, “Yapay Sinir Ağları (Kuram,Mimari,Uygulama)”, Seçkin Yayınları.
- [29] **SAĞIROĞLU** Ş., BEŞDOK E., ERLER M., 2003 “Mühendislikte Yapay Zeka Uygulamaları Yapay Sinir Ağları”, Ufuk Yayınları.

## ÖZGEÇMİŞ

Murat İKİZ 1976 yılında Ordu'da doğdu. İlk ve Orta öğrenimini Ordu'da tamamladıktan sonra 1994 yılında girdiği Hava Harp Okulu Mühendislik Fakültesi Elektronik Mühendisliği Bölümü'nden 1998 yılında mezun oldu. 2004 yılında Dicle Üniversitesi Fen Bilimleri Enstitüsü'ne bağlı olarak Elektrik ve Elektronik Mühendisliği bölümünde Yüksek Lisans eğitimine başladı. Murat İKİZ halen Hava Kuvvetleri Komutanlığı bünyesinde pilot olarak görev yapmaktadır.