ÖZET

Yüksek Lisans Tezi

OTOMATİK KONUŞMA TANIMA ALGORİTMALARININ UYGULAMALARI

Köksal ÖCAL
Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik Mühendisliği Anabilim Dalı
Danışman : Yrd. Doç. Dr. H. Gökhan İLK

Bu çalışmada, SMM (Saklı Markov Model) tabanlı izole bir kelime tanıma sistemi geliştirilerek, sesin akustik parametreleri LPC (Linear Predictive Coding), LPCC (LPC Cepstrum), CEPS (Ayrık Fourier dönüşümü tabanlı cepstrum) ve MFCC (Mel Frequency Cepstral Coefficients) 'nin konuşmacıdan bağımsız konuşma tanıma sistemlerindeki performansları değerlendirilmiştir. Değişik akustik parametrelerle birlikte değişik SMM tipleri de (ergodik, Bakis vb.) kullanılarak bu modellerin konuşmacıdan bağımsız konuşma tanıma sistemlerindeki başarılarını karşılaştırılmıştır. Konuşma tanıma sistemi MATLAB ortamında geliştirilmiş ve sözlük olarak sadece rakamlar kullanılmıştır. Rakamların 20 adet konuşmacıya üçer adet tekrarlatılması sonucu her bir rakam için 60 adet eğitim verisi toplanmıştır. Eğitim verileri kullanılarak sesin farklı akustik parametreleri ve farklı SMM tipleriyle (ergodik, Bakis) her bir rakam için model hesaplamaları yapılmıştır. Eğitim verileriyle sistemin doğruluğu incelendikten sonra, test verisi olarak eğitim aşamasına katılmamış 20 adet konuşmacının bir veya birkaç rakamı tekrarlaması sonucu elde edilen veriler kullanılarak, farklı akustik parametrelerin ve model tiplerinin performansları incelenmiştir. Yapılan çalışmalar sonucunda en iyi performansı, Bakis tipi SMM'lerin ve MFCC akustik parametrelerinin verdiği tespit edilmiştir.

2005 , 81 sayfa

ANAHTAR KELİMELER : SMM, konuşmacı bağımsız izole kelime tanıma, öznitelik vektörleri.

ABSTRACT

Master Thesis

APPLICATION OF AUTOMATIC SPEECH RECOGNITION ALGORITHMS

Köksal ÖCAL
Ankara University
Graduate School of Natural and Applied Sciences
Department of Electronics Engineering

Supervisor : Asst. Prof. Dr. H. Gökhan İLK

In this study, the performance of the acustic parameters of speech, LPC (Linear Predictive Coding), LPCC (LPC derived cepstrum), CEPS (Discrete Fourier Transform based cepstrum), MFCC (Mel Frequency Cepstral Coefficients) and different HMM (Hidden Markov Model) types (ergodic, Bakis) were compared in a speaker independent isolated word recognition system. The system was devoloped in the MATLAB environment. Only digits were used as the recognition dictionary. A trainning set that consists 60 occurrences of each digit by 20 talkers was used (three occurrences of each digit per talker). Different HMM model parameters were calculated for digits using different acustic parameters of the training set and different HMM types. After testing the algorithm using the training set, a testing set that consists 60 occurrences of digits by another 20 talkers was used (one or more occurrences of each digit per talker) in order to compare different acustic parameters of speech and HMM types in a speaker independent manner. The results revealed that Bakis type HMM models and MFCC acustic parameters give better performances than the others.

2005 , 81 pages
KEY WORDS : HMM, speaker independent isolated speech recognition, feature vectors.

TEŞEKKÜR

Çalışmanın her aşamasında önerileri ile beni yönlendiren ve ihtiyaç duyduğum her an yanımda olarak bana bilimsel danışmanlıktan çok daha fazlasını veren değerli danışmanım Sayın Yrd. Doç. Dr. H. Gökhan İLK' e teşekkürü bir borç bilirim.

Tez çalışması ve tezin yazımı süresince dostluklarını ve sevgilerini bir an bile esirgemeyen, bu süre boyunca bana tahammül eden aileme ve iş arkadaşlarıma sonsuz teşekkürlerimi sunuyorum.

Köksal ÖCAL
Ankara, Temmuz 2005

İÇİNDEKİLER

SİMGELER DİZİNİ

ARPA	Advanced Research Projects Agency
CEPS	Ayrık Fourier dönüşümü tabanlı cepstrum
DARPA	Defense Advanced Research Projects Agency
DFT	Discrete Fourier Transform
HMM	Hidden Markov Model
IDFT	Inverse Discrete Fourier Transform
ISP	Internet Service Provider
ISRTK	Isolated Speech Recognition Toolkit
LPC	Lineer Predictive Coding
LPCC	LPC Cepstrum
MFCC	Mel Frequency Cepstral Coefficients
SMM	Saklı Markov Modeli
SUR	Speech Understanding Research

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

1.1. Sinyal Modelleme

Günlük hayattaki prosesler ölçülebilir çıktılara sahiptir ve bunlar genel olarak sinyaller olarak karakterize edilebilir. Bu sinyaller yapıları itibariyle kesikli (sonlu bir alfabeden karakterler, her hangi bir kod defterinden nicemlenmiş vektörler) veya süreklidir (konuşma, ısı ölçümleri). Sinyal kaynağı durağan (istatiksel özellikleri zamanla değişmeyen) veya durağan değildir (sinyal özelikleri zamanla değişen). Sinyal saf (tek bir sinyal kaynağından oluşan) veya bozulmuş (gürültülü) olabilir.

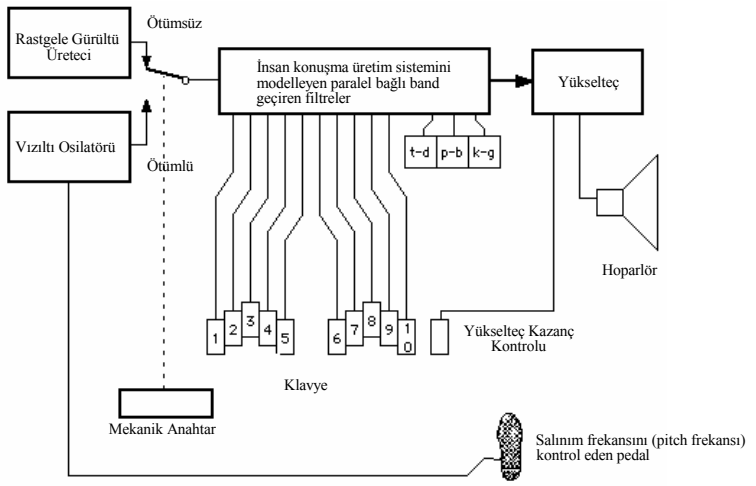
Bu tip sinyallerin, sinyal modelleriyle nasıl karakterize edilebileceği ilgi duyulan temel problemlerden biridir. Sinyal modelleri kullanmanın çeşitli sebepleri vardır. İlk olarak, bu modeller sinyal işleme sistemlerinin teorik yapılarının temelini oluşturur. Örneğin gürültüyle veya iletim hatları nedeniyle bozulmuş konuşma sinyali göz önüne alındığında, gürültüyü uzaklaştırmak ve iletim hatlarından kaynaklanan bozulmayı yok etmek amacıyla bir sinyal modeli tasarlanabilir. Sinyal modelleri kullanmanın diğer bir sebebi ise, bu modellerin sinyalin gerçek kaynağını bilmeden sinyal kaynağı hakkında çok önemli bilgiler verebilmesidir. Bu durumda iyi bir sinyal modeli ile sinyal kaynağına benzetim yapılabilir. Sonuç olarak sinyal modelleri kullanmanın en önemli sebebi, bu modellerin pratikte çok iyi çalışması ve bir çok önemli pratik sistemin gerçeklenmesini sağlayabilmesidir. Bu sistemlere örnek olarak öngörüm sistemleri ve tanıma sistemleri sayılabilir.

Genel olarak sinyal modelleri deterministik ve istatistik modeller olarak ikiye ayrılabilir. Deterministik modeller sinyallerin bilinen bir takım özellikleri üzerinde dururlar. Örnek olarak sinyalin bir sinüs dalgası veya eksponensiyellerin toplamı olduğunu düşünelim. Bu durumda sinyalin modellenmesi için genlik, frekans, faz vb. kullanacağımız bir takım belirli özellikler vardır. İstatiksel modeller sinyallerin istatiksel özelliklerini modellerler. Bu modellere örnek olarak Gausian prosesleri, Poision prosesleri, Markov prosesleri ve Saklı Markov prosesleri sayılabilir.

Bu çalışmada konuşma sinyalinin SMM (Saklı Markov Model) ile modellenmesi üzerinde durulacaktır.

1.2. Konuşma Tanımanın Kısa Tarihçesi

Kronolojik olarak konuşma alanında yapılan çalışmalar incelendiğinde göze çarpan en eski çalışmanın 1936 yılında AT&T Bell Labs tarafından üretilen ve Voder olarak adlandırılan ilk elektronik konuşma sentezleyicisi olduğu görülmektedir (Dudley, Riezs ve Watkins). Voder'in çalışmasına ait blok diyagram **Şekil 1.1.**'de verilmiştir.



Şekil 1.1. Voder çalışma ilkesi

Konuşma sinyali genel olarak ötümlü ve ötümsüz olmak üzere iki gruba ayrılabilir. Ötümlü sesler hemen hemen periyodik bir yapıya, ötümsüz sesler ise gürültüye benzerler. Ötümlü seslere örnek olarak a, e, i, ötümsüz seslere ise ğ, j, k verilebilir. **Şekil 1.1.** 'de bahsedilen band geçiren filtreler üretilecek sesin ötümlü veya ötümsüz olmasına bağlı olarak ya vızıltı osilatöründen yada rasgele gürültü üreticinden beslenir. Bu işlem Voder'ı kullanan operatör tarafından mekanik bir anahtar aracılığıyla gerçekleştirilir. Vızıltı osilatörünün frekansı, pedal yardımıyla ayarlanır. Band geçiren filtreler üretilecek sesin formantlarını belirler ve dolayısı ile

insan konuşma sistemindeki artikülatörleri (dil, dudak, çene vb..) modeller. Formantlar konuşma sinyalinin spektrumunun zarfındaki rezonans frekanslarıdır (spektrumun zarfındaki tepe noktaları) ve yerleri artikülatörler tarafından belirlenir. Voder'da bu işlem şekilde de görülen klavye aracılığıyla gerçekleştirilir. Son olarak paralel bağlı filtrelerin çıkışları toplanarak kazancı operatör tarafından ayarlanabilen bir yükseltece girilir. Üretilen bu sinyal ses çıkış aygıtına verilerek konuşma üretilir.

Voder ilk kez eğitilmiş operatörler tarafından 1939 yılında Newyork World's Fairs' de tanıtılmıştır. Uzun bir eğitim sürecinden sonra, Voder, operatörler tarafından bir piyano gibi çalınabilmiştir.

1970'lerin başlarında konuşma tanımaya, Saklı Markov Modelleme (SMM) yaklaşımı Princeton Üniversitesinde Lenny Baum tarafından keşfedilmiş ve içinde IBM' inde bulunduğu bir çok ARPA (Advanced Research Projects Agency) tarafından paylaşılmıştır. SMM, karmaşık bir matematiksel örüntü eşleme stratejisi olarak tanımlanabilir ve içinde Dragon Systems, IBM, Philips ve AT&T'nin de bulunduğu bir çok konuşma tanıma şirketi tarafından kullanılmıştır. SMM ilerleyen bölümlerde ayrıntılı olarak anlatılacaktır.

1971 yılında DARPA (Defense Advanced Research Projects Agency) tarafından, sürekli konuşmayı anlayabilecek bir bilgisayar sistemi geliştirmek için SUR (Speech Understanding Research) kuruldu. Programı başlatan Lawrance Roberts 5 yıl boyunca her bir yıl için 3 milyon dolar olmak üzere hükümet fonundan harcama yaptı. Buna ek olarak CMU, SRI, MIT Lincoln Laboratory, Systems Devolopment Corporation (SDC) ve BBN (Bolt, Berenak and Newman) 'da kapsamlı SUR projeleri kurulmuştur.

1978 yılında Texas Instruments tarafından, popüler oyuncak "Speak and Spell" geliştirilmiştir. "Speak and Spell" için bir konuşma çipi tasarlanmış ve bu çip daha doğal (insana yakın) konuşma sentezleme alanında büyük adımların atılmasını sağlamıştır.

1984 yılında SpeechWorks isminde bir şirket kurulmuş ve telefon üzerinden otomatik konuşma tanıma sistemleri (ASR) üretmiştir.

1995 yılında, ilk kez Dragon Systems tarafından üretilen kelime tabanlı dikte yazılımı piyasaya sürülmüştür. Bunun ardından, benzer yazılımlar IBM ve Kurzweil tarafından da üretilmeye başlanmıştır.

1996'da Charles Schwab ve Nuance tarafından Voice Broker isminde bir konuşma tanıma sistemi geliştirilmiş ve bu sistemle 360 adet müşteri telefon üzerinden aynı anda borsa işlemi yapmıştır. Bu sistem, her gün 50000 adet isteği yerine getirebilmiştir. Sistemin doğrulunun %95 civarında olduğu belirlenmiştir. Yine aynı yıl Dragon Systems "Naturally Speaking" 'i geliştirmiş ve bu ürün ilk sürekli dikte yazılımı olmuştur.

1.3. Konuşma Tanımanın Günümüzdeki Uygulamaları ve Geleceği

Konuşma tanıma uygulamaları günlük hayatta yavaş yavaş yerini almaya başlamıştır. Bunlara örnek olarak dikte paketleri (dictation packages), sesli tarama (voice browsing), telefon üzerinden konuşma tanıma tabanlı otomatik sistemler, görme hareket vb. engelleri olan insanlara makinelerle iletişimi sağlamak için bir takım alternatifler sunan sistemler verilebilir.

Dikte paketleriyle, bilgisayarlara en hızlı daktilo kullananlarla karşılaştırıldığında bile çok hızlı ve kolay metin girişi yapılabilmektedir. Bu tip dikte sistemleri satın alınabilecek uygun fiyatlarla piyasaya sürülmüştür. Lernout and Hauspie'dan Voice Xpress, Dragon Systems'den Naturally Speaking, Philips'den FreeSpeech, SpeechPro ve IBM'den ViaVoice günümüzdeki dikte paketlerine örnek olarak verilebilir.

Günümüzdeki diğer bir uygulama ise sesli web tarama sistemleridir. Bu tip sistemlerle doğrudan bir telefon üzerinden daha önce mümkün olmayan koşullarda web taraması gerçekleştirilmektedir. TellMe buna benzer bir sistemdir. Bu sistemle kullanıcılar telefon üzerinden sesli olarak borsa bilgilerinden restoranlara kadar her türlü bilgiye, bilgisayar ve ISP bağlantısı olmadan ulaşabilmektedir.

Konuşma alanındaki yaygın uygulamalardan biride telefon üzerinden bankacılık rezervasyon gibi işlemlerdir. Daha önce telefon tuşları kullanılarak uzun zaman alan işlemler konuşma tanıma tabanlı sistemlerle daha hızlı, ucuz ve kolay hale getirilmiştir.

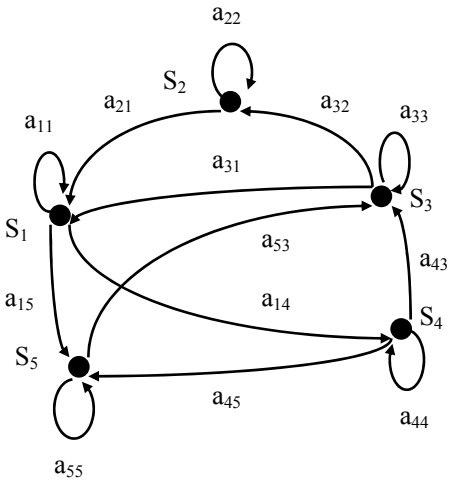
Yakından tanıdığımız sesli arama yapan cep telefonları, sese duyarlı ev eşyaları daha önce saydığımız örnekler gibi hayatımızı kolaylaştırmaktadır.

Konuşma tanıma alanında şu an yapılan çalışmalara bakıldığında sistemlerin doğruluğunun artırılması, bağımlılıkların ortadan kaldırılması (konuşmacı, dil, konuşulan ortam vb.) yönünde olduğu görülmektedir. Doğruluğun artması ve bağımlılıkların ortadan kalkmasıyla ilerde daha pratik ve önemli sistemlerin ortaya çıkacağı açıktır.

2. KURAMSAL TEMELLER

2.1. Ayrık Markov Prosesleri

Her hangi bir anda S_1, S_2, … S_N gibi N farklı durumdan herhangi birinde bulunan bir sistem düşünelim. N = 5 için böyle bir sistem **Şekil 2.1.**'de verilmiştir.



Şekil 2.1. Bazı durum geçişleri verilmiş 5 durumlu bir Markov zinciri

Sistem eşit aralıklı ayrık zamanlarda durumlarla ilgili bir olasılık setine göre bir durumdan diğer duruma veya aynı duruma geçiş yapar. Geçişlerin yapıldığı ayrık zaman anları t = 1,2,… olarak ve t anındaki durumu q_t olarak gösterelim. **Şekil 2.1**'deki sistemin olasılıksal olarak tam bir şekilde ifade edilebilmesi için her hangi bir durumun, ondan önce gelen durumlarla birlikte olasılıksal olarak ifade edilebilmesi gerekir. Birinci dereceden bir Markov zincirinde bu olasılıksal ifade şimdiki ve bir önceki duruma indirgenir.

$$P[q_t = S_j \mid q_{t-1} = S_i, q_{t-2} = S_k, ...] = P[q_t = S_j \mid q_{t-1} = S_i] \qquad (2.1)$$

(2.1)'de verilen eşitliğin zamanla değişmediği proseslerde durum geçiş olasılıkları, a_{ij}'ler eşitlik (2.2)'deki gibi ifade edilir ve eşitlik (2.3) ve (2.4)'de verilen istatiksel kısıtlamaları sağlarlar.

$$a_{ij} = P[q_t = S_j \mid q_{t-1} = S_i) \qquad 1 \le i,j \le N \tag{2.2}$$

$$a_{ij} \ge 0 \tag{2.3}$$

$$\sum_{j=1}^{N} a_{ij} = 1 \tag{2.4}$$

Şimdi, hava durumunun üç durumlu Markov modelini düşünelim. Her hangi bir günde hava durumu aşağıdaki durumlardan biri olarak gözlensin.

Durum 1: Yağmurlu veya karlı
Durum 2: Bulutlu
Durum 3: Güneşli

A durum geçiş matrisi, $\{a_{ij}\}$ eşitlik (2.5)'deki gibi olsun.

$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix} \tag{2.5}$$

Birinci günde (t=1) hava durumunun güneşli (Durum 3) olduğu verilsin. Bu durumda şöyle bir soru sorulabilir. Modele göre gelecek 7 günün "güneşli-güneşli-yağmurlu-yağmurlu-güneşli-bulutlu-güneşli" olma olasılığı nedir? Daha formal olarak, bir Markov modeli verildiğinde gözlem sırası $O=\{O_1, O_2, O_3, O_4, O_5, O_6, O_7, O_8\}=\{S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3\}$ olma olasılığı nedir. Bu problem aşağıdaki gibi çözülebilir.

$$P(O \mid Model) = P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3 \mid Model)$$

$$= \pi_i \cdot a_{33} \cdot a_{33} \cdot a_{31} \cdot a_{11} \cdot a_{13} \cdot a_{32} \cdot a_{23}$$

$$= 1 \cdot (0.8) \cdot (0.8) \cdot (0.1) \cdot (0.4) \cdot (0.3) \cdot (0.1) \cdot (0.2)$$

$$= 1.536 \times 10^{-4}$$

Burada π_i, t=0 anında (başlangıç) S_i durumundan başlama olasılığıdır. Modelin (zincirin) hangi durumla hangi olasılıkta başlangıç yapacağını belirleyen parametredir.

$$\pi_i = P[q_1 = S_i], \qquad 1 \le i \le N \tag{2.6}$$

Bunun dışında modeli kullanarak hava durumu her hangi bir durumda iken bu durumda d gün boyunca kalma olasılığı nasıl hesaplanır? gibi bir soruda sorulabilir.

$$O = \{\underset{1}{S_i}, \underset{2}{S_i}, \underset{3}{S_i}, ..., \underset{d}{S_i}, \underset{d+1}{S_j} \neq S_i\}$$
$$P(O \mid Model, q_1 = S_i) = (a_{ii})^{d-1}(1 - a_{ii}) = p_i(d) \tag{2.7}$$

Eşitlik (2.7)'de verilen $p_i(d)$, i. durumun d gün boyunca devam etmesinin olasılık dağılımıdır. Bu üstsel yoğunluk fonksiyonu Markov zincirinin durum kalma süresi karakteristiğidir. Bu olasılık yoğunluk fonksiyonunu kullanılarak i. durumun devam etme sayısının beklenen değeri eşitlik (2.8)'deki gibi hesaplanabilir.

$$\begin{aligned} \bar{d}_i &= \sum_{d=1}^{\infty} dp_i \\ &= \sum_{d=1}^{\infty} d(a_{ii})^{d-1}(1 - a_{ii}) = \frac{1}{1 - a_{ii}} \end{aligned} \tag{2.8}$$

Hava durumu modeli için durum kalma sayılarının beklenen değerleri **Çizelge 2.1.**'de verilmiştir.

Çizelge 2.1. Hava durumu modeli için durum kalma sayılarının beklenen değerleri

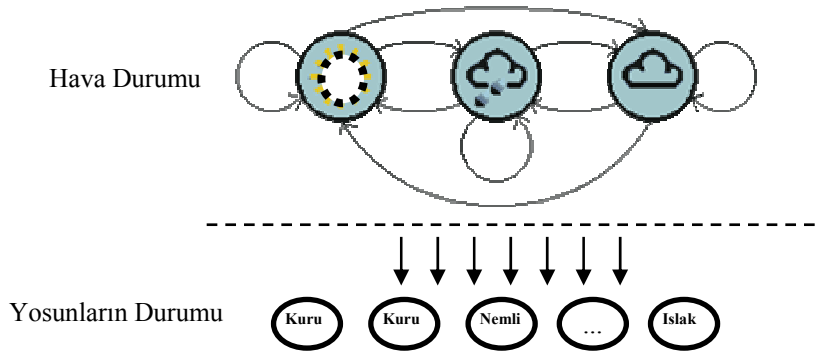
Durum	**Durum kalma sayısının beklenen değeri**
Yağmurlu	$\frac{1}{1-0.4} = 1.67$
Bulutlu	$\frac{1}{1-0.6} = 2.5$
Güneşli	$\frac{1}{1-0.8} = 5$

2.2. Saklı Markov Modellerine Geçiş

Bölüm 2.1.'de anlatılan Markov modellerinde her bir durum gözlemlenebilir fiziksel bir olaya karşılık geliyordu. Bu tip bir modelleme bir takım problemlerin modellenmesi için gerekli yapıya sahip olmayabilir. Bu bölümde gözlemlerin, durumların birer olasılıksal fonksiyonları olduğu Markov modelleri incelenecektir. Sonuç olarak oluşan model, arka planda işleyen, gizli ve gözlemlenemeyen bir istatiksel proses ile bu istatiksel proses sonucunda oluşan ve ölçülebilen gözlem sıralarını oluşturan başka istatiksel proseslerin birleşiminden oluşan çift katlı istatiksel bir süreç halini alır ve Saklı Markov Modeli (SMM) olarak bilinir. Bu bilgiler ışığında Bölüm 2.2.1'de verilen hava durumu yosun modeli ve Bölüm 2.2.2'de verilen çuval içinde renkli bilye modellerini inceleyelim.

2.2.1. Hava durumu yosun modeli

İçinde yosunlarının yetiştiği ve hava durumunun doğrudan izlenemediği bir zindanın içinde olduğumuzu düşünelim. Zindandaki yosunların fiziksel durumları zamanla değişmekte ve "Kuru","Nemli" ve "Islak" şeklinde doğrudan izlenebilmektedir. Yosunlardaki fiziksel değişimlerin sebebinin doğrudan gözlemlenemeyen havanın farklı durumları olduğu tahmin edilmektedir. Böyle bir durumda yosunların fiziksel hallerindeki değişimi modelleyen bir SMM nasıl oluşturulabilir?



Şekil 2.2. Hava durumu yosun modeli

Bu sorunun cevaplanması için ilk yapılması gereken modeldeki durumların neler olduğunu ve kaç tane olabileceğini belirlemektir. **Şekil 2.2.**'de örnek bir SMM gösterilmiştir. Bu modelde üç adet durum vardır. Bu durumlar havanın "Güneşli","Yağmurlu" ve "Bulutlu" durumlarıdır. Hava durumunun değişimi istatiksel bir prosestir ve bu istatiksel proses direkt olarak gözlemlenememektedir. Fakat arka planda işleyen bu gizli prosesin sonucunda oluşan ve gözlemlenebilen üç adet istatiksel proses daha vardır. Bu prosesler havanın her bir farklı durumu için yosunların fiziksel değişimidir.

2.2.2. Çuval içinde renkli bilye modeli

Şimdi SMM ile modellenebilecek başka bir senaryo üzerinde düşünelim. Kapalı bir odanın içinde birden altıya kadar numaralandırılmış 6 adet çuval ve her bir çuvalın içinde M değişik renkte, değişik sayıda bilyeler olsun. Bu odanın içindeki bir adam, yazı tura atarak bir başlangıç çuvalı belirlesin ve bu çuvaldan bir bilye çeksin. Çektiği bilyenin rengini kaydettikten sonra bilyeyi aldığı yere bıraksın. (Parayla başlangıç çuvalı seçme işleminde 'yazı' ilk çuvala 'tura' ise ikinci çuvala karşılık gelecektir. Diğer çuvallarla başlama olasılığı ise sıfır olacaktır). Bu işlemden sonra zar atarak geçiş yapacağı çuvalı belirlesin ve bilye çekme işlemini bu çuval için tekrarlasın.(Geçiş çuvalı belirlerken zar atma işleminin olası sonuçları 1, 2, 3, 4, 5, 6 geçiş yapılacak çuvalın numarasına karşılık gelecektir). Bu şekilde T adet bilye çekme işleminden sonra, çektiği bilyelerin rengini, çekme sırasına göre, odanın dışında bulunan ve olup bitenlerden haberi olmayan başka bir adama söylesin. Bu işlem belli sayıda tekrar edildiğinde T uzunluğunda çekilen bilyelerin rengini gösteren gözlem dizileri oluşacaktır. Bu durumda gözlem sıralarının oluşmasının açıklayan **Şekil 2.3.** 'deki gibi bir model oluşturulabilir. Bu modelde 6 adet durum vardır ve her bir durum bir çuvala karşılık gelmektedir. Durum başlangıç olasılıkları ve durum geçiş olasılık matrisi anlatılan senaryoya göre sırasıyla eşitlik (2.9) ve (2.10)'daki gibi verilebilir.

$$\pi = \{\pi_i\} = \{\frac{1}{2}, \frac{1}{2}, 0, 0, 0, 0, 0\} \qquad (2.9)$$

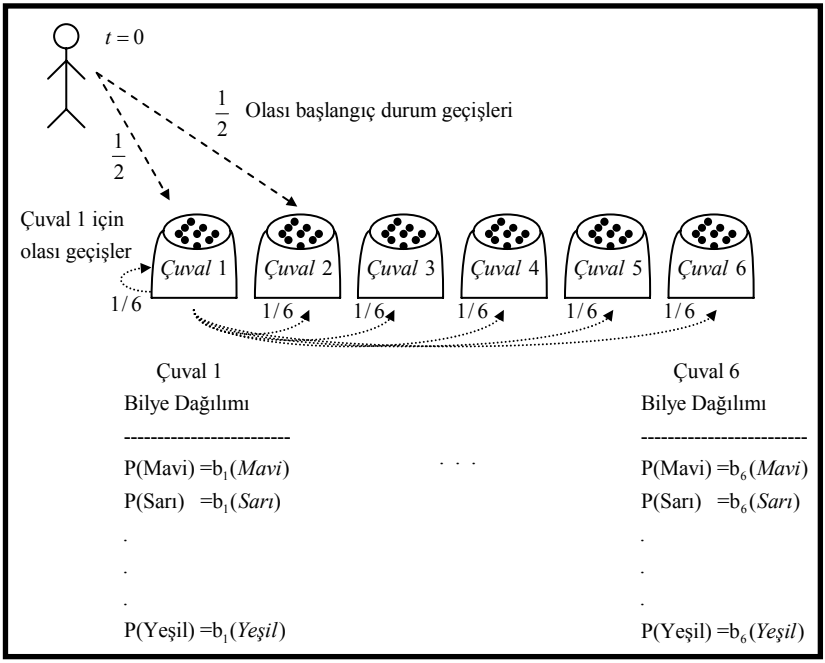
$$a_{ij} = \begin{bmatrix} \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6} \\ \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6} \\ \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6} \end{bmatrix} \tag{2.10}$$

Başlangıç durum olasılıkları ve durum geçiş olasılıklarından başka belirlenmesi gereken bir parametre daha vardır. Bu parametre modeldeki durumların hangi gözlemleri hangi olasılıkla ürettiğini belirleyen durum gözlem olasılık dağılımlarıdır ve (2.11)'deki gibi tanımlanır.

$$b_j(k) = P[t \text{ anında } v_k \mid q_t = S_j], \quad \begin{matrix} 1 \le \text{j} \le \text{N} \\ 1 \le \text{k} \le \text{M} \end{matrix} \tag{2.11}$$

Burada j durum numarasını, k ise gözlemin gözlem alfabesindeki indisini gösterir. $b_j(k)$ j. durumun alfabedeki k indisli gözlemi oluşturma olasılığıdır. Gözlem alfabesi her bir durum için izlenen farklı gözlemlerin bir alfabesidir. Bu alfabe her bir durum için farklı seçilebilir. Bu bölümde anlatılan modeldeki gözlem alfabeleri her bir durum için aynıdır ve (2.12)'deki gibi tanımlanabilir.

$$\begin{aligned} V &= \{v_k\} = \{v_1, v_2, ..., v_M\} \\ &= \{Mavi, Sarı, ..., Yeşil\} \end{aligned} \tag{2.12}$$



Şekil 2.3. Çuval içinde renkli bilye modeli

2.2.3. SMM'nin bileşenleri

Bölüm 2.2.1 ve Bölüm 2.2.2'de verilen SMM örneklerinde arka planda işleyen gizli bir Markov süreci olduğunu ve bu Markov sürecindeki durumların belli bir olasılık dağılımına göre gözlem vektörlerini ürettiği görüldü. Bu gözlem vektörleri SMM 'nin çıktısı olan gözlem dizilerini oluşturur. Şimdi bütün bunları bir araya getirerek daha formal bir biçimde bir SMM 'nin bileşenlerini inceleyelim.

Bir SMM aşağıdaki bileşenlerle karakterize edilir.

1) N, modeldeki durum sayısıdır. Modellemedeki durumlar gizli olmasına ve dolayısı ile sayısının tam olarak bilinmemesine rağmen bazı sistemlerdeki durumlar daha önce verilen örneklerde

olduğu gibi fiziksel bir olguyla ilişkilendirilebilir. Örnek olarak hava durumu-yosun modelinde durumlar "Güneşli", "Yağmurlu" ve "Bulutlu" hava koşullarına , çuval içinde renkli bilye modelinde ise durumlar çuvallara karşılık geliyordu. Genel olarak her bir duruma her hangi bir durumdan geçiş yapılabilir. Daha sonra ayrıntılı olarak inceleneceği gibi bu tip bir SMM 'ye ergodik SMM denir. Bazı modellerde durumlar arasındaki bazı geçişler sınırlandırılabilir. Modeldeki durumlar $S=\{S_1, S_2,\ldots, S_N\}$ ve t anındaki durum q_t şeklinde gösterilecektir.

2) M, her bir durum için gözlemlenen farklı gözlem sayısı veya sonlu alfabe büyüklüğü. Gözlemler sistemin çıktılarıdır. Hava durumu – yosun modelinde gözlemler yosunun "Kuru", "Nemli" ve "Islak" halleri, çuval içinde renkli bilye modelinde ise bilyelerinin renkleriydi. Alfabe büyüklüğü durum sayısına eşit olmak zorunda değildir.

$$V = \{v_k\} = \{v_1, v_2, ..., v_M\} \tag{2.13}$$

3) Durum geçiş olasılık dağılımları $A=\{a_{ij}\}$

$$a_{ij} = P[q_{t+1} = S_j \mid q_t = S_i], \qquad 1 \le i,j \le N \tag{2.14}$$

4) Durumların gözlem olasılık dağılımları, $B=\{b_j(k)\}$

$$b_j(k) = P[t \text{ anında } v_k \mid q_t = S_j], \qquad \begin{matrix} 1 \le j \le N \\ 1 \le k \le M \end{matrix} \tag{2.15}$$

5) Başlangıç durum dağılımı

$$\begin{matrix} \pi = \{\pi_i\} \\ \pi_i = P[q_1 = S_i], \qquad 1 \le i \le N \end{matrix} \tag{2.16}$$

Şimdi tüm bu parametreleri kullanarak bir SMM 'nin

$$O = O_1, O_1, ..., O_T \tag{2.17}$$

gibi bir gözlem dizisini nasıl oluşturduğunu aşama aşama inceleyelim. Burada O_t t anında her hangi bir durumdayken üretilen gözleme karşılık gelmektedir.

1) Başlangıç durum dağılımı π ye göre bir $q_1 = S_i$ başlangıç durumu belirle.
2) t =1
3) $q_t = S_i$ durumundaki gözlem olasılık yoğunluk dağılımı $b_i(k)$ 'ya göre bir gözlem vektörü $O_t = v_k$ seç.
4) S_i durumunun durum geçiş olasılık dağılımına göre (a_{ij}) $q_{t+1} = S_j$ durumuna geçiş yap.
5) t=t+1. Şayet t<T ise 3. basamağa geri dön. Aksi takdirde işlemi bitir.

Bir SMM 'nin tam olarak belirtilebilmesi için N,M, gözlem vektörleri ile A, B, π olasılık setlerinin belirlenmesi gerekir $\lambda = \{A, B, \pi\}$ bir SMM 'yi gösterir.

2.2.4. SMM için üç temel problem

Problem 1: Bir $O = O_1, O_2, ..., O_T$ gözlem dizisi ve $\lambda = \{A, B, \pi\}$ modeli verildiğinde, gözlem dizisinin olasılığı, $P(O \mid \lambda)$ nasıl hesaplanır?

Problem 2: Bir $O = O_1, O_2, ..., O_T$ gözlem dizisi ve $\lambda = \{A, B, \pi\}$ modeli verildiğinde, bu gözlem dizisine karşılık gelen durum dizisi $Q = q_1, q_2, ..., q_T$ (bu gözlemleri oluşturan durumların dizisi) her hangi bir kritere göre optimal olacak şekilde nasıl hesaplanır?

Problem 3: Belli sayıda gözlem dizisi verildiğinde SMM parametreleri, $\lambda = \{A, B, \pi\}$ $P(O \mid \lambda)$ 'lar maksimum olacak şekilde nasıl hesaplanır?

Problem 1 bir değerlendirme problemidir. Bir gözlem dizisi ve bir model verildiği zaman, bu gözlem dizisinin bu model tarafından hangi olasılıkla üretildiğinin bulunmasıdır. Daha başka bir bakış açısıyla bu gözlem dizisinin hangi ölçüde bu modele uygun olduğunun hesabıdır. Problem 1 'in

çözümü ile, bir gözlem dizisi ve birden fazla model verildiğinde bu gözlem dizisinin hangi modele daha yakın olduğu bulunabilir.

Problem 2, bir model ve bir gözlem dizisi verildiğinde, bu gözlem dizisine karşılık gelen optimum durum dizisini belli bir kritere göre ortaya çıkarma problemidir.

Problem 3 gözlem dizileri verildiği zaman bu gözlem dizilerinin oluşmasını en iyi şekilde açıklayan model parametrelerinin bulunması problemidir. Bu gözlem dizileri eğitim dizileri, bu işlem ise SMM eğitim işlemi olarak adlandırılır.

Bu fikirler ışığında içinde W adet kelime bulunan küçük sözlüklü bir konuşma tanıma sistemini düşünelim. Örneğin bir rakam tanıma sisteminde W=10 ve kelimeler {'sıfır','bir',...,'dokuz'} olur. Konuşma sinyalinin kodlanmış spektral vektörler olarak düşünelim. Bu vektörlerin M büyüklüğündeki bir kod defteri aracılığıyla kodlansınlar. Bu durumda her bir gözlem vektörü kod defterindeki bir indise karşılık gelir. Her bir kelime için belli sayıda tekrarlar sonucu oluşturulan eğitim dataları olsun. Bu eğitim datalarını kullanarak SMM 'lerin tasarımı Problem 3'ün çözümüdür. Yani $\lambda = \{A, B, \pi\}$ hesaplanır. Modeldeki durumların fiziksel anlamlarını bulmak ve eğitim datalarını durumlara ayırarak model üzerinde değişiklikler yapmanın çözümü Problem 2'dir. Yani gözlemlere göre durum sıralıları bulunur. Bu işlemler sonucunda W adet SMM bulunduğu zaman bir test dizisi verildiğinde bu dizinin hangi modele daha yakın olduğunun çözümü ise Problem 1 dir. Yani $P(O \mid W_i)$ 'yi maksimum yapan W_i kelimesidir. Burada W_i sözlükteki i. kelime O ise test gözlem dizisidir.

2.2.5. Problem 1'in çözümü

Problem 1 $O = O_1, O_2, ..., O_T$ gözlem dizisi ve $\lambda = \{A, B, \pi\}$ SMM 'si verildiğinde $P(O \mid \lambda)$ olasılığının hesaplanmasıdır. Bu olasılığı hesaplamanın bir yolu T uzunluğundaki tüm durum sıralıları için olasılıkların hesaplanıp bu olasılıkların toplanması olabilir.

$Q = q_1, q_2, ..., q_T$

Burada q_1 başlangıç durumudur. Bu durum sıralısı için gözlem sıralısının olasılığı

$$P(O \mid Q, \lambda) = \prod_{t=1}^{T} P(O_t \mid q_t, \lambda) \tag{2.18}$$

olarak hesaplanır. Bu eşitlik yazılırken gözlemlerin istatiksel olarak bağımsız oldukları varsayılmıştır. Bu eşitlik açılırsa

$$P(O \mid Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \cdots b_{q_T}(O_T) \tag{2.19}$$

olarak elde edilir.

Durum sıralısının olasılığı ise eşitlik (2.20)'deki gibi yazılabilir.

$$P(Q \mid \lambda) = \pi_{q_1} \cdot a_{q_1 q_2} \cdot a_{q_2 q_3} \cdots a_{q_{T-1} q_T} \tag{2.20}$$

$\lambda = \{A, B, \pi\}$ verildiğinde O gözlem dizisinin ve Q durum dizisinin birleşik olasılığı (2.19)'da ve (2.20)'de ayrı ayrı hesaplanan bu iki olasılığın çarpımıdır.

$$P(O, Q \mid \lambda) = P(O \mid Q, \lambda) P(Q, \lambda) \tag{2.21}$$

O gözlem dizisinin $\lambda = \{A, B, \pi\}$ modeli altında oluşma olasılığı tüm olası durum dizileri için (2.21)'de verilen olasılığın hesaplanıp toplanması ile bulunabilir. $P(O \mid \lambda)$'nın eşitlik (2.22) verilen şekilde hesaplanmasına doğrudan hesaplama denir.

$$\begin{aligned} P(O \mid \lambda) &= \sum_{all\ Q} P(O \mid Q, \lambda) P(Q \mid \lambda) \\ &= \sum_{q_1, q_2, ..., q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \cdots a_{q_{T-1} q_T} b_{q_T}(O_T) \end{aligned} \tag{2.22}$$

Eşitlik (2.18)-(2.21)'de verilen hesaplamalar sözle şu şekilde anlatılabilir. Başlangıç anında (t=1) q_1 durumundan π_{q_1} olasılığı ile başlanıyor ve bu durumda $b_{q_1}(O_1)$ olasılıkla O_1 gözlemi oluşturuluyor. Bunun ardından zamanın t anından t+1(t=2) anına gelmesiyle birlikte q_1 durumundan q_2 durumuna $a_{q_1 q_2}$ olasılığıyla geçiş yapılıyor ve yeni geçilen bu durumda

$b_{q_2}(O_2)$ olasılığıyla O_2 gözlemi oluşturuluyor. Bu işlem bu şekilde q_{T-1} durumundan q_T durumuna $a_{q_{T-1}q_T}$ olasılığıyla geçip $b_{q_T}(O_T)$ olasılığıyla O_T gözlemi oluşturuluncaya kadar (t=T) devam ediyor.

$P(O|\lambda)$ 'nın doğrudan hesaplama yöntemiyle hesaplanması halinde $(2T-1)\cdot N^T$ çarpma ve N^T-1 toplama yapılması gerekir, çünkü N farklı durum için toplam T uzunluğunda N^T adet farklı durum sıralısı vardır. Her bir durum sıralısı için (2T-1) adet çarpma yapıldığından, toplamda $(2T-1)\cdot N^T$ tane çarpma yapılır. Daha sonra her bir durum sıralısı için bulunan sonuçlar topladığından ise N^T-1 adet toplama gerekir. $P(O|\lambda)$ 'nın doğrudan (2.18)-(2.22)'de verilen eşitliklerle hesaplanması uygulanabilir değildir. Çünkü küçük N ve T değerlerinde bile çok fazla sayıda işlem yapılması gerekir. Örnek olarak N=6 ve T=100 olması durumunda 199×6^{100} adet çarpma ve $(6^{100}-1)$ adet toplama yapılması gerekir. Bu sebepden dolayı daha etkili bir hesaplama yönteminin kullanılması zorunludur. Bu amaçla geliştirilmiş ve ileri-geri değişken yöntemi (The Forward-Backward Procedure) olarak bilinen çok etkili bir hesaplama yöntemi vardır.

İleri – Geri Değişken Yöntemleri:

İleri ve geri değişken yöntemleri $\lambda=\{A,B,\pi\}$ SMM parametrelerinin zamanla değişmezliği ve gözlem vektörlerinin birbirlerinden bağımsız oldukları varsayımı göz önüne alınarak geliştirilmiş yöntemlerdir. İleri değişken yöntemine geçmeden önce ileri değişken $\alpha_t(i)$ değişkenini tanımlayalım.

$$\alpha_t(i)=P(O_1O_2...O_t,q_t=S_i\,|\,\lambda) \qquad (2.23)$$

$\alpha_t(i)$ değişkeni (2.23)'deki eşitlikten görüldüğü gibi t anına S_i durumunda iken kısmi gözlem sıralısı $O_1O_2...O_t$ 'nin üretilme olasılığıdır. $\alpha_t(i)$ değişkeni özyineli bir yapıda aşağıdaki gibi hesaplanabilir.

1) Başlama

$$\alpha_1(i) = \pi_i b_i(O_1), \qquad 1 \le \text{i} \le \text{N}. \tag{2.24}$$

2) Özyineleme

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] \cdot b_j(O_{t+1}), \quad 1 \le \text{t} \le \text{T-1} \quad 1 \le \text{j} \le \text{N}. \tag{2.25}$$

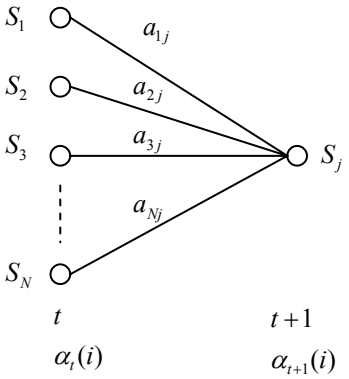
3) Sonlandırma

$$P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{2.26}$$

Yöntemin ilk basamağında ileri değişkenler ($\alpha_1(i), \quad 1 \le \text{i} \le \text{N}$) t=1 anında başlangıç durumu S_i ve başlangıç gözlemi O_1'in birleşik olasılığına set edilir. Daha sonra özyineleme aşamasına geçilir. Bu aşama yöntemin en önemli noktasıdır. **Şekil 2.4.** 'de t+1 anındaki S_j durumuna, t anındaki N adet olası durumda (S_i, $1 \le \text{i} \le \text{N}$) geçişler gösterilmiştir. $\alpha_t(i)$, t anında S_i durumunda iken, $O_1 O_2 ... O_t$ kısmi gözlem dizisinin üretilmesi olasılığı olduğundan $\alpha_t(i) a_{ij}$ olasılığı t anında S_i durumunda iken $O_1 O_2 ... O_t$ gözlem dizisinin üretilme ve t+1 anında S_j durumuna geçme olasılığını verir. Tüm i değerleri ($1 \le \text{i} \le \text{N}$) için bu olasılık hesaplanıp toplanması halinde kısmi gözlem dizisi $O_1 O_2 ... O_t$'nin üretilip, t+1 anında S_j durumunda olma olasılığı bulunur. Bu işlem sonunda S_j durumu bilindiğinden bu olasılık değeri $b_j(O_{t+1})$ ile çarpılarak $\alpha_{t+1}(j)$ hesaplanır. Özyinelemeye tüm t=1,2,…,T-1 anları için devam edilerek elde edilen N adet ileri değişkenin toplamı $P(O \mid \lambda)$ olasılığını verir.

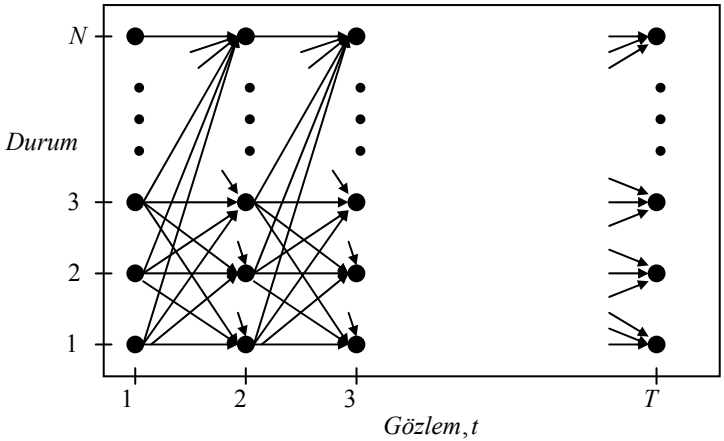
$P(O \mid \lambda)$'nın ileri değişken yöntemi ile hesaplanmasını işlem yükü bakımından doğrudan hesaplama yöntemiyle karşılaştıralım. $P(O \mid \lambda)$ doğrudan hesaplama yöntemiyle hesaplandığında $(2T-1) \cdot N^T$ adet çarpma ve $N^T - 1$ toplama gerektirir. Bu iki büyüklüğü topladığımızda yaklaşık olarak $2T \cdot N^T$ adet işlem yapılması gerektiği ortaya çıkmaktadır. İleri değişken yönteminde her bir özyinelemede N^2 işlem yapıldığından ve özyineleme T kere tekrarlandığından toplam yapılan işlem yaklaşık olarak

N^2T adettir. N=5 ve T=100 için doğrudan hesaplama yönteminin ve ileri değişken yönteminin işlem yükü hesaplanırsa yaklaşık olarak sırasıyla 10^{72} ve 3000 değerleri bulunur. Bu iki büyüklüğü oranladığımızda doğrudan hesaplama yönteminin 10^{69} kat daha fazla işlem gerektirdiği ve ileri değişken yönteminin çok etkili bir yöntem olduğu sonucuna varılabilir.



Şekil 2.4. t anından t+1 anında S_j durumuna olası tüm geçişler

İleri değişken ile olasılık hesaplama yöntemi **Şekil 2.5.** 'de verilen kafes yapısına dayalıdır. Bu yapıdaki anahtar nokta modelde sadece N adet durum olması ve gözlem dizisinin büyüklüğünden bağımsız olarak tüm olası durum dizilerinin her hangi bir anda anda bu N adet noktadan birine kavuşmasıdır. t=1 anında $1 \le i \le N$ için sadece $\alpha_1(i)$ değerlerini hesaplamamız gerekir. t=2,3,…,T anlarında ise sadece $\alpha_t(j), \quad 1 \le j \le N$ değişkenlerini hesaplamamız gerekir. Bu hesaplamalarda sadece $\alpha_t(i)$ 'nin bir önceki andaki N adet değeri $\alpha_{t-1}(i), \quad 1 \le i \le N$ kullanılır. Çünkü t anında N adet noktaya sadece t-1 anıdaki N adet noktadan geçiş yapılabilir.



Şekil 2.5. İleri değişken yönteminin grafiksel anlatımı

İleri değişken yöntemine benzer mantıkla geri değişken yöntemiyle de $P(O|\lambda)$ hesaplanabilir. Bunun için öncelikle geri değişken $\beta_t(i)$ tanımlansın.

$$\beta_t(i) = P(O_{t+1}O_{t+2}\cdots O_T \mid q_t = S_i, \lambda) \qquad (2.27)$$

Bu değişken t anında S_i durumunda iken ,bu andan sonra kısmi gözlem dizisi $O_{t+1}O_{t+2}\cdots O_T$ üretilme olasılığıdır. $\beta_t(i)$ değişkeni özyineli bir yapıda aşağıdaki gibi hesaplanabilir (Eşitlik (2.28)-(2.30)).

1) Başlama

$$\beta_T(i) = 1, \qquad 1 \le i \le N \qquad (2.28)$$

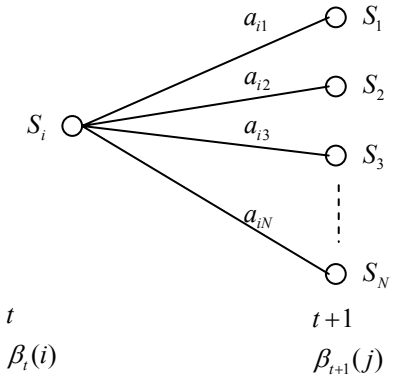
2) Özyineleme

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j),$$
$$t = T-1, T-2, \ldots, 1 \qquad (2.29)$$
$$1 \le i \le N$$

3) Sonlandırma

$$P(O|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_t(i) \qquad (2.30)$$

Geri değişken ile olasılık hesaplama yönteminde ilk basamakta $\beta_T(i)$ geri değişkenleri 1'e eşitlenir ve her bir özyinelemede **Şekil 2.6.** 'da gösterilen tüm olası yolların geçiş olasılıkları ve gözlem üretme olasılıkları bu başlangıç değerinden düşülerek bir önceki andaki geri değişken değerleri hesaplanır.



Şekil 2.6. Geri değişken hesaplama yönteminin grafiksel gösterimi

2.2.6. Problem 2'nin çözümü

Problem 2'nin problem 1'den farklı olarak tam bir çözümü yoktur. Çünkü problem bir gözlem dizisi ve model verildiğinde bu gözlem dizisiyle ilgili optimum durum dizisini bulmak olduğundan, problemin değişik optimumluk kriterlerine göre farklı çözümleri vardır. Örnek olarak optimumluk kriterini, bir gözlem dizisi ve model verildiğinde t anında olması en olası durum olarak belirleyelim. Bu durumda Problem 2'in çözümü nedir? Bunun için öncelikle eşitlik (2.31)'de verilen değişkeni tanımlayalım.

$$\gamma_t(i) = P(q_t = S_i \mid O, \lambda) \qquad (2.31)$$

Bu değişken bir model ve bir gözlem dizisi verildiğinde t anında S_i durumunda olma olasılığıdır. Bu olasılık değeri Bölüm 2.2.5'de incelenen ileri ve geri değişkenler cinsinden eşitlik (2.32)'deki gibi yazılabilir.

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O \mid \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)} \qquad (2.32)$$

Bölüm 2.2.5'de anlatıldığı gibi $\alpha_t(i)$ öngörülen modelin t anında S_i durumunda iken $O_1O_2...O_t$ gözlem dizisini oluşturma olasılığıdır. $\beta_t(i)$ ise t anında S_i durumunda iken, bu andan sonra kısmi gözlem dizisi $O_{t+1}O_{t+2}\cdots O_T$ 'nin üretilme olasılığıdır. Bu iki olasılığın çarpımının $P(O \mid \lambda)$ olasılığı ile normalizasyonu ile öngörülen modelin O gözlem dizisini üretirken t anında S_i durumunda olma olasılığı bulunur.

$$\sum_{i=1}^{N} \gamma_t(i) = 1 \qquad (2.33)$$

$\gamma_t(i)$ değişkenini kullanılarak her hangi bir andaki en olası durum, eşitlik (2.34) verilen şekilde bulunabilir.

$$q_t = \arg\max_{1 \le i \le N}[\gamma_t(i)], \qquad 1 \le t \le T. \qquad (2.34)$$

(2.34)'de verilen çözümde her hangi bir t anındaki optimum durum bulunurken o anda olabilecek en olası durum seçildi. Bu şekildeki bir kritere göre yapılan çözümün sonucunda bazı problemler ortaya çıkabilir. Örnek olarak bazı durum geçiş olasılıklarının sıfır olduğu modellerde, (2.34)'de verilen yöntemle bulunan durum dizisi, bu sıfır geçiş olasılıklarıyla çelişkili olabilir. Çünkü bu yöntem durum dizilerinin oluşma olasılıklarını göz ardı ederek her hangi bir t anındaki en olası durumu bulur. Bu problemi çözmenin bir yolu optimumluk kriterini değiştirmektir. Örnek olarak optimumluk kriteri en olası ardışık durum çiftleri (q_t, q_{t+1}) veya üçlüleri (q_t, q_{t+1}, q_{t+2}) olabilir yada $P(Q \mid O, \lambda)$ maksimum olacak şekilde bir durum dizisi de bulunabilir. Bu örnekler içinde en çok kullanılanı en son bahsedilendir. Bu kriter sözle şu şekilde ifade edilebilir. Bir model ve bir gözlem dizisi verildiğinde öyle bir durum dizisi kullanılsın ki, bu durum dizisiyle gözlem dizisinin oluşturulma olasılığı maksimum olsun. Bu problemi çözmek için kullanılan, dinamik programlama yöntemlerine dayalı ve Viterbi algoritması olarak bilinen bir yöntem vardır.

Viterbi algoritmasına geçilmeden önce $\delta_t(i)$ değişkeni tanımlansın.

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{t-1}} P[q_1 q_2 \ldots q_t = i, O_1, O_1, \ldots, O_t \mid \lambda] \quad (2.35)$$

$\delta_t(i)$ değişkeni t anında S_i durumuyla biten ve kısmi gözlem dizisi $O_1, O_1, \ldots, O_t$ 'yi en yüksek olasılıkla üreten durum dizisinin olasılığıdır. Bu değişken özyineli bir yapıda aşağıdaki gibi ifade edilebilir (Eşitlik (2.37)-(2.42)).

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] \cdot b_j(O_{t+1}) \quad (2.36)$$

$\delta_t(i)$ yardımıyla Viterbi algoritması aşağıdaki gibi çalışır.

1) Başlangıç

$$\delta_1(i) = \pi_i b_i(O_t), \qquad 1 \le i \le N$$
$$\psi_1(i) = 0 \quad (2.37)$$

2) Özyineleme

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(O_t), \quad 2 \le t \le T, \; 1 \le j \le N \quad (2.38)$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) \cdot a_{ij}], \quad 2 \le t \le T, \; 1 \le j \le N \quad (2.39)$$

3) Sonlandırma

$$P^* = \max_{1 \le i \le N} [\delta_T(i)] \quad (2.40)$$

$$q_T{}^* = \arg\max_{1 \le i \le N} [\delta_T(i)] \quad (2.41)$$

4) Geri dönüş ve optimum durum dizisinin bulunması

$$q_t^* = \psi_{t+1}[q_{t+1}{}^*], \qquad t = T-1, T-2, \ldots, 1. \quad (2.42)$$

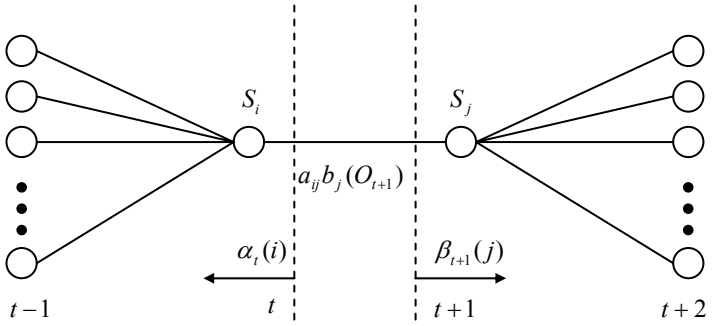
2.2.7. Problem 3'ün çözümü

Bu problem şimdiye kadar incelenen problemlerin en zorudur. Gözlem dizileri verildiği zaman , bu dizilerin olasılıkları maksimum olacak şekilde SMM parametrelerinin ayarlanmasıdır. Bu problemin analitik bir çözümü yoktur. Fakat $P(O \mid \lambda)$ lokal maksimum olacak şekilde $\lambda = (A, B, \pi)$ parametreleri Baum-Welch tahmin algoritması, EM (expectation-modification) algoritmaları (Dempster *et al.* 1977) veya gradient teknikleri (Levinson *et al.* 1973) kullanılarak çözülebilir. Bu bölümde Baum-Welch tahmin algoritmasına dayalı çözümü göz önüne alınacaktır.

Bu algoritmaya geçmeden önce $\xi_t(i,j)$ değişkenini tanımlayalım.

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \lambda). \tag{2.43}$$

$\xi_t(i,j)$ model ve gözlem dizisi verildiğinde t anında S_i ve t+1 anıda S_j durumunda olma olasılığıdır. Değişkene ait grafiksel gösterim **Şekil 2.7.**'de verilmiştir.



Şekil 2.7. $\xi_t(i,j)$ değişkeninin grafiksel gösterimi

$\xi_t(i,j)$ değişkenini daha önce incelenen ileri ve geri değişkenler cinsinden eşitlik (2.44)'deki gibi yazılabilir.

$$\begin{aligned}\xi_t(i,j) &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(O \mid \lambda)} \\ &= \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}\end{aligned} \tag{2.44}$$

(2.44) eşitliğindeki paydadaki terim $P(O \mid \lambda)$ istenilen olasılık ölçüsünü verir.

Bölüm 2.2.6.'da anlatılan Problem 2'nin çözümünde tanımlanan $\gamma_t(i)$ değişkeni, $\xi_t(i,j)$ değişkeninin j üzerinden toplanması yolu ile eşitlik (2.45)'deki gibi elde edilebilir.

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{2.45}$$

t anında S_i durumunda olma olasılığı $\gamma_t(i)$ t üzerinden toplanırsa, S_i durumundan geçiş sayısının beklenen değerini elde edilir. Yine benzer mantıkla $\xi_t(i,j)$ değişkeni t üzerinden toplanırsa S_i durumundan S_j durumunda geçişin beklenen değeri bulunur.

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{ S}_\text{i} \text{ durumundan geçişin beklenen değeri} \tag{2.46}$$

$$\sum_{t=1}^{T-1} \xi_t(i,j) = \text{S}_\text{i} \text{ durumundan S}_\text{j} \text{ durumuna geçişinin bekleneni} \tag{2.47}$$

(2.46) ve (2.47) eşitliğinde T anında geçiş söz konusu olmadığından toplamlardan çıkarılmıştır. Bu eşitlikler kullanılarak SMM parametreleri $\lambda = (A, B, \pi)$ (2.48)-(2.50)'deki gibi yeniden tahmin edilir.

$$\overline{\pi}_i = (t=1) \text{ anında S}_\text{i} \text{ durumunun beklenen frekansı} = \gamma_1(i) \tag{2.48}$$

$$\overline{a}_{ij} = \frac{S_i \text{ durumundan } S_j \text{ durumuna geçişin beklenen değeri}}{\text{S}_\text{i} \text{ durumundan geçişin beklenen değeri}} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{2.49}$$

$$\overline{b}_j(k) = \frac{j \text{ durumunda v}_\text{k} \text{ gözleminin oluşmasının bekleneni}}{\text{j durumundan geçişin bekleneni}} = \frac{\sum_{\substack{t=1 \\ O_t = v_k}}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{2.50}$$

$\lambda = (A, B, \pi)$ modelini şu anki model ve (2.48), (2.49) ve (2.50) eşitlikleri kullanılarak yeniden tahmin edilen modeli ise $\overline{\lambda} = (\overline{A}, \overline{B}, \overline{\pi})$ olarak tanımlayalım. Şu anki model $\lambda = (A, B, \pi)$ $P(O \mid \lambda)$ olasılığının kritik bir noktasını tanımlar ve bu noktada $\lambda = \overline{\lambda}$ 'dir. Bununla birlikte $P(O \mid \overline{\lambda}) \geq P(O \mid \lambda)$ 'dır (Rabiner 1989).

(2.48)-(2.50)'de verilen algoritmaya dayalı olarak, $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$ modeli özyinelemeli bir şekilde hesaplanırsa O gözlem dizisinin olasılığı belli bir limit noktasına kadar artırılabilir ve sonuç olarak Problem 3'ün çözümünü elde edilir.

Baum-Welch yeniden algoritması Baum'un yardımcı fonksiyonu $Q(\lambda, \bar{\lambda})$ 'nın $\bar{\lambda}$ üzerinde enbüyütülmesi ile türetilebilir (Rabiner 1989).

$$Q(\lambda, \bar{\lambda}) = \sum_{Q} P(Q \mid O, \lambda) \log[P(O, Q \mid \bar{\lambda})] \qquad (2.51)$$

$$\max_{\bar{\lambda}}[Q(\lambda, \bar{\lambda})] \Rightarrow P(O \mid \bar{\lambda}) \geq P(O \mid \lambda) \qquad (2.52)$$

2.2.8. Bölüm Özeti

Bölüm 2.2.5, Bölüm 2.2.6 ve Bölüm 2.2.7'de verilen problemler ve çözümleri özet olarak Çizelge 2.2'de verilmiştir.

Çizelge 2.2. SMM'deki problemler ve çözümleri

No:	Problem:	Çözüm:
1	Bir $O = O_1, O_2, ..., O_T$ gözlem dizisi ve $\lambda = \{A, B, \pi\}$ modeli verildiğinde, gözlem dizisinin olasılığı, $P(O \mid \lambda)$ nasıl hesaplanır?	İleri ve geri değişken yöntemleri
2	**Problem 2:** Bir $O = O_1, O_2, ..., O_T$ gözlem dizisi ve $\lambda = \{A, B, \pi\}$ modeli verildiğinde $P(O, Q \mid \lambda)$ olasılığını maksimum yapan $Q = q_1, q_2, ..., q_T$ durum dizisi nasıl hesaplanır?	Viterbi algoritması
3	**Problem 3:** Belli sayıda gözlem dizisi verildiğinde SMM parametreleri, $\lambda = \{A, B, \pi\}$ $P(O \mid \lambda)$ 'lar maksimum olacak şekilde nasıl hesaplanır?	Baum-Welch yeniden tahmin algoritması

2.3. SMM Çeşitleri

Şimdiye kadar incelenen SMM 'lerde her bir duruma diğer durumlardan hiçbir kısıtlama olmadan geçiş yapılabiliyordu. **Şekil 2.8.**'de de gösterilen bu tip modellere tam bağlı yada ergodik SMM denir. Bu şekildeki modellerde geçiş katsayılarının tümü pozitiftir ve geçiş olasılık matrisi eşitlik (2.54) verilen şekildedir.

$$a_{ij} > 0, \quad 1 \le \mathrm{i} \le \mathrm{N} \text{ ve } 1 \le j \le \mathrm{N} \tag{2.53}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{2.54}$$

Durum geçişlerine bir takım kısıtlamalar getirilerek türetilen bazı SMM çeşitlerinin sinyalleri ergodik modellerden daha iyi modelleyebildiği görülmüştür. Böyle bir model **Şekil 2.9.**'de verilmiştir. Bu tip modellere soldan-sağa model (left-right model) yada Bakis model denir. Bakis tipi modellerin özelliği zaman ilerledikçe durum indisinin yerinde sayması veya artmasıdır. Bu yüzden Bakis tipi modeller, özellikleri zamanla değişen sinyallerin modellenmesi için istenilen özelliğe sahiptir (Rabiner 1986). Soldan-sağa SMM 'lerin durum geçiş olasılıklarındaki temel kısıtlama (2.55) eşitliğinde verilmiştir.

$$a_{ij} = 0, \qquad \mathrm{j} < \mathrm{i} \tag{2.55}$$

(2.55) 'den de anlaşıldığı gibi indisi büyük olan durumlardan indisi küçük olan durumlara geçiş olasılığı sıfırdır. Bununla birlikte başlangıç durum olasılıklarıda (2.56) eşitliğinde verilen özelliğe sahiptir.

$$\pi_i = \begin{cases} 0, & \mathrm{i} \neq 1 \\ 1, & \mathrm{i} = 1 \end{cases} \tag{2.56}$$

(2.55) ve (2.56) eşitliklerinde de anlaşıldığı gibi, Bakis tipi modeller her zaman aynı durumla başlar ve aynı durumla biter. Bunlara ek olarak Bakis

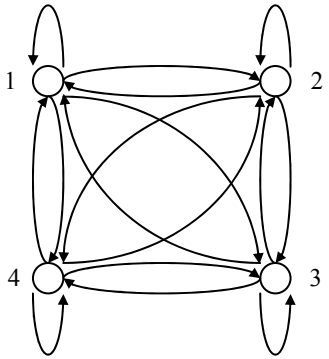
tipi modellere durum indisinin çok fazla değişememesi için (2.57)'de verilen kısıtlama da getirilebilir.

$$a_{ij} = 0, \qquad j > i + \Delta \tag{2.57}$$

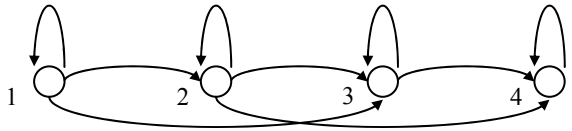
Şekil 2.9.'da gösterilen soldan-sağa SMM modelinde Δ değeri 2'dir ve bu modele ait durum geçiş matrisi eşitlik (2.58) ve (2.59)'da verilmiştir.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix} \tag{2.58}$$

$$\begin{aligned} a_{NN} &= 1 \\ a_{Ni} &= 0, \qquad i < N \end{aligned} \tag{2.59}$$



Şekil 2.8. 4 durum ergodik (tam bağlı) SMM



Şekil 2.9. 4 durum soldan-sağa (Bakis) SMM ($\Delta = 2$)

SMM çeşitlerindeki durum geçiş kısıtlamaları, Bölüm 2.2.7.'de bahsedilen Baum-Welch yeniden tahmin algoritmasında hiçbir değişikliğe yol açmaz. Yeniden tahmin algoritması, başlangıç değerlerinin kısıtlamalar göz önüne

alınarak seçilmesi ve özyineleme boyunca korunması yoluyla aynen gerçekleştirilebilir (Rabiner 1986).

2.3.1. Sürekli Gözlem Olasılık Yoğunluk Fonksiyonuna Sahip SMM'ler

Şu ana kadar bahsedilen modellerde gözlemlerin sonlu bir alfabeden alınan ayrık semboller olduğu ve her bir durum için gözlem olasılık dağılımlarının ayrık olduğu düşünüldü. Bazı uygulamalarda (konuşma tanıma) gözlemler süreklidir. Bu tip sinyaller kod defterleriyle nicemlenebilir ve bununla birlikte nicemlemeden kaynaklanan hatalar ortaya çıkabilir. Bu tip durumlarda SMM 'lerde sürekli olasılık yoğunlukları kullanmak avantajlı olabilir.

SMM 'deki durumların gözlem olasılık yoğunluklarında kullanılacak fonksiyonların, parametrelerinin sürekli ve tutarlı bir biçimde yeniden tahmin edilmeye uygun bir formda olması gerekir. Yeniden tahmin yöntemi, Liporace (1982) tarafından, (2.60) eşitliğinde verilen, sonlu karışım halindeki pdf için formulize edilmiştir.

$$b_j(O) = \sum_{m=1}^{M} c_{jm} \Re[O, \mu_{jm}, U_{jm}], \qquad 1 \le \mathrm{j} \le \mathrm{N} \qquad (2.60)$$

(2.60)'da verilen eşitlikte O gözlem vektörü, c_{jm} j. durumdaki m. karışımın katsayısı ve $\Re$, μ_{jm} ortalama değerine ve U_{jm} kovaryans matrisine sahip log-konkav eliptik simetrik yoğunluk fonksiyonudur (Liporace 1982). Konuşma tanıma uygulamalarında Gausian yoğunluk dağılımları yaygın olarak kullanılır. c_{jm} (2.61) ve (2.62) eşitliklerindeki istatiksel kısıtlamayı sağlar ve pdf (2.63) eşitliğinde verilen şekilde normalize edilmiştir.

$$\sum_{m=1}^{M} c_{jm} = 1, \quad 1 \le j \le N \qquad (2.61)$$

$$c_{jm} \ge 0, \quad 1 \le j \le N, 1 \le m \le M \qquad (2.62)$$

$$\int_{-\infty}^{\infty} b_j(x)dx = 1, \quad 1 \le \mathrm{j} \le \mathrm{N}. \qquad (2.63)$$

Liporace (1982) ve Juang (1986) tarafından c_{jk}, μ_{jk} ve U_{jk} için yeniden tahmin formülleri (2.64), (2.65) ve (2.66) eşitliklerinde verilmiştir.

$$\overline{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{k=1}^{M} \gamma_t(j,k)} \tag{2.64}$$

$$\overline{\mu}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k).O_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{2.65}$$

$$\overline{U}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k).(O_t - \mu_{jk})(O_t - \mu_{jk})^T}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{2.66}$$

Burada T vektör transpozu, $\gamma_t(j,k)$ ise t anında j. durumda iken O_t gözlem vektörünün k. karışımdan üretilme olasılığıdır.

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jk}\Re(O_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^{M} c_{jm}\Re(O_t, \mu_{jm}, U_{jm})}\right] \tag{2.67}$$

$\gamma_t(j,k)$ tek bir karışım veya ayrık yoğunluk kullanıldığında (2.45) eşitliğinden verilen $\gamma_t(j)$'ye eşit olduğu görülmektedir. (2.64), (2.65) ve (2.66) eşitliklerinde verilen yeniden tahmin formülleri sözle şu şekilde açıklanabilir. $\overline{c}_{jk}$ sistemin j. durumda k. karışımı kullanmasının beklenen değerinin j. durumda olmasının beklenen değerine oranıdır. Benzer şekilde μ_{jk} k. karışımın ortalama değeri, U_{jk} ise kovaryans matrisidir.

2.4. SMM'nin Konuşma Tanıma Uygulamaları

2.4.1. Genel hatlarıyla konuşma tanıma sistemi

Şekil 2.10'da sürekli konuşma tanıma sistemine ait blok diyagram verilmiştir. Bu bloklara ait açıklamalar ise aşağıda verilmiştir.

1) Öznitelik Analizi (Feature Analysis)

   Konuşma sinyalinin spektral analizi yapılarak gözlem vektörleri elde edilebilir. Bu konuyla ilgili ayrıntılı açıklama ilerleyen bölümlerde verilecektir.

2) Birim Eşleme Sistemi (Unit Matching System)

   Birim eşleme sistemini tasarlamadan önce yapılması gereken ilk işlem eşlemede kullanılacak konuşma tanıma biriminin seçilmesidir. Konuşma tanıma birimlerine örnek olarak fonem, fonem ikilileri (diphones), fonem üçlüleri (triphones), hece, kelime ,kelime öbekleri verilebilir. Konuşmanın en küçük (parçalanamaz, tek) ses birimine fonem (phoneme) denir. İki adet fonemin bir araya gelmesiyle diphone, üç adet fonemin bir araya gelmesiyle ise triphone oluşur. Fonemler konuşma esnasında kendilerinden önce ve sonra gelen fonemlerden etkilenir. Bunun sebebi konuşma sistemindeki artikülatörlerin (dil, dudak, çene vb.) bir pozisyondan diğer pozisyona aninden geçememesidir. Bu olay literatürde co-articulation effect olarak bilinir. Bu durumdan dolayı birim eşleme sistemlerinde fonemlerden ziyade diphone ve triphone kullanılır. Konuşma tanıma birimi karmaşıklaştıkça dildeki sayısı da artar. Örnek olarak İngilizce'de yaklaşık olarak 40-50 foneme karşın yüz binlerce kelime vardır. Konuşma tanıma birimi konuşma tanıma siteminin özelliklerine ve yapısına uygun olarak seçilir. Örnek olarak geniş sözlük kapasitesine sahip konuşma tanıma sistemlerinde (large vocabulary speech recognition systems) kelime tabanlı birimlerden ziyade fonem tabanlı birimler seçmek zorunlu olacaktır çünkü kelime tabanlı geniş sözlük kapasitesine sahip bu tip sistemlerin (1000 veya daha fazla kelime) eğitimi için yeterli datayı sağlamak ve saklamak zordur. Seçilen konuşma

tanıma biriminden bağımsız olarak bu birimlerin bir envanteri eğitim aşamasında elde edilir. Tipik olarak, bu eğitim dataları kullanılarak her bir birim, daha önce gördüğümüz SMM çeşitlerinden biriyle karakterize edilir. Bu SMM'lerin parametreleri eğitim dataları kullanılarak hesaplanır. Birim eşleme sisteminin görevi, bilinmeyen konuşma sinyaline karşılık gelebilecek , konuşma tanıma birimlerinin sıralılarının olasılıklarının hesaplamaktır.

3) Sözcüksel Kodlama (Lexical Decoding)

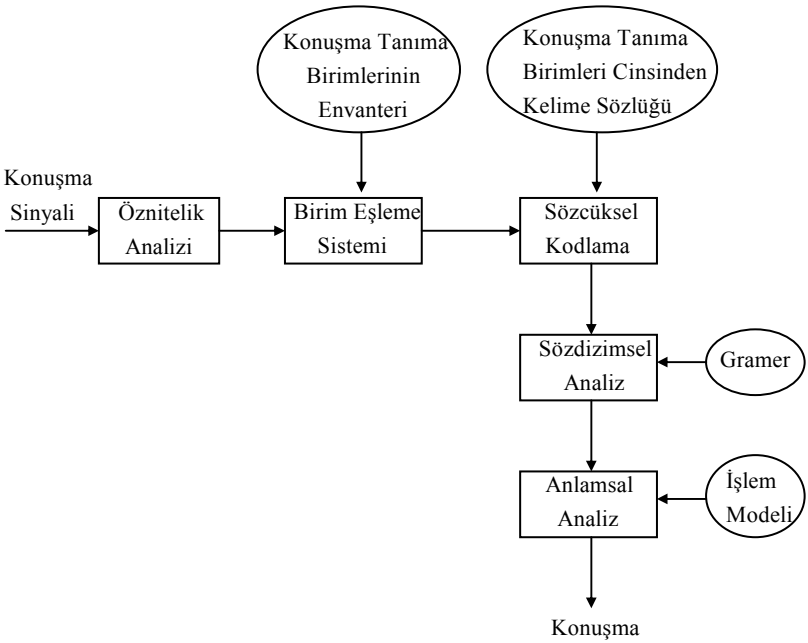
Sözcüksel kodlama işlemi, birim eşleme sisteminin bulduğu konuşma tanıma birimleri sıralılarının sözlükte olup olmadığını kontrol ederek bu sıralılar için bir filtre teşkil etmektir. Burada bahsedilen sözlük (lexicon) konuşma tanıma birimleri cinsinden kelime sözlüğüdür. Kelime tabanlı konuşma tanıma sistemlerinde bu işlemin yapılmasına gerek yoktur.

4) Sözdizimsel Analiz (Syntactic Analysis)

Sözdizimsel analiz (gramer) sözcüksel kodlama gibi birim eşleme sisteminin sonucunda elde edilen konuşma tanıma birimleri sıralıların, konuşma tanıma sisteminin kelime gramerine uygun olup olmadığını kontrol ederek bir filtre görevi yapar. Kelime gramerleri deterministik sonlu durum ağları (Bu tip gramerlerde kabul edilebilecek tüm kelime kombinasyonları sıralanır) veya istatiksel gramer olarak ifade edilebilir. İstatiksel gramerlere örnek olarak n-gram verilebilir. Bazı basit komut ve kontrol sistemlerinde sonlu sayıdaki bir sözlükten sadece tek bir kelimenin tanınması gerekir. Bu tip konuşma tanıma sistemlerine yalıtık konuşma tanıma sistemleri (isolated speech recognition systems) denir ve bu sistemlerde gramer gereksizdir. Bazı konuşma tanıma sistemlerinde ise çok basit gramer kuralları yeterlidir. Örnek olarak bir sürekli rakam tanıma sisteminde bir rakamdan sonra sadece diğer bir rakam gelebilir. Son olarak gramerin baskın bir faktör olduğu konuşma tanıma sistemleri de vardır. Bu tip sistemlere örnek olarak geniş sözlüklü sürekli konuşma tanıma sistemleri verilebilir.

5) Anlamsal Analiz (Semantic Analysis)

Daha önce bahsettiğimiz işlemler esnasında elde edilen kelime ve kelime gruplarının anlamsal olarak analiz edildiği aşamadır. Anlamsal analiz, sözcüksel kodlama ve sözdizimsel kodlama gibi konuşma tanıma arama yolları üzerinde başka bir filtre daha teşkil eder.



Şekil 2.10. Sürekli konuşma tanıma sisteminin genel hatlarıyla blok diyagramı

Konuşma tanıma uygulamalarında karşılaşılan en büyük problemlerden biri konuşma sinyalini, arka plan sessizliğinden (background silence) ayırmaktır. Bu işlem için kullanılan bir takım yöntemler vardır.

1) Sinyal enerjisi, sinyal süresi gibi bir takım özelliklere bakılarak, doğrudan konuşma sinyalinin varlığını belirlemek.
2) Arka plan için istatiksel bir model belirlenerek gelen sinyal "sinyal = (silence) + speech + (silence) " şeklinde ifade edilir.

3) Konuşma tanıma birimlerinin modelleri, opsiyonel olarak ilk ve son durumları arka plan sessizliği içerecek şekilde genişletilebilir.

Konuşma tanıma sistemlerinde bu üç yöntemden de faydalanılır.

2.4.2. Yalıtık kelime tanıma

V adet kelimeden oluşan bir sözlük ve sadece bu kelimeleri tanıyan bir konuşma tanıma sistemi düşünelim. Sözlükteki her bir kelime bir SMM ile modellensin. Eğitim verisi olarak sözlükteki her bir kelime için farklı veya aynı konuşmacılar tarafından tekrarlanmış K adet örnek bulunsun. Bu K adet örnek öznitelik analizi (feature analysis) yöntemleriyle konuşma vektörlerine dönüştürülür ve bunun sonucunda O gözlem sıraları elde edilir. Bu gözlem sıraları kullanılarak her kelime için bir SMM oluşturulur ve model parametreleri hesaplanır.

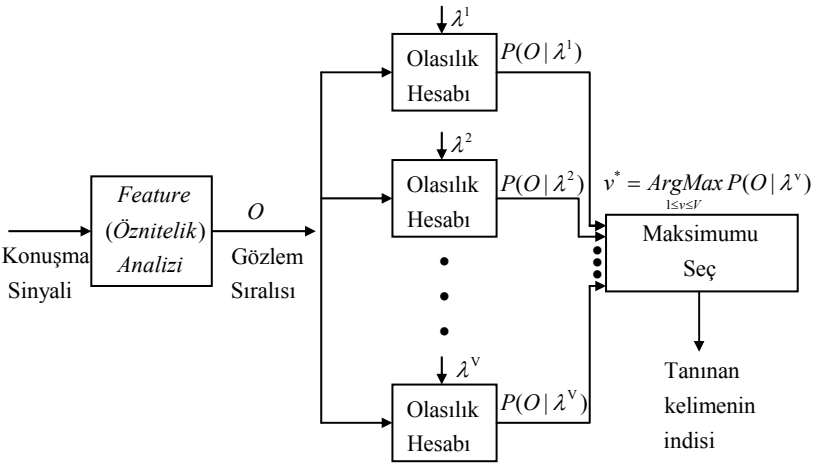
Şekil 2.11. 'de basit bir yalıtık kelime tanıma sistemi için blok diyagram verilmiştir. Şekilde görülen $\lambda^1, \lambda^2, ..., \lambda^V$ sözlükte kelimelere ait SMM'lerdir. Bilinmeyen bir konuşma sinyali verildiğinde, yapılması gereken bu sinyali V adet kelimeden birine sınıflandırmaktır. $O = O_1, O_2, ..., O_T$ bilinmeyen konuşma sinyaline ait gözlem vektörleridir. $\lambda^v, \quad 1 \le v \le V$ kelime modelleri ve O gözlem vektörü verildiğinde izole kelime tanıma sistemi (2.68) 'de verilen eşitliğe indirgenebilir. Burada v^*, $\lambda^v, \quad 1 \le v \le V$ ve O verildiğinde konuşma tanıma sisteminin tanıdığı kelimenin indisidir.

$$v^* = \arg\max_{1 \le v \le V} [P(O \mid \lambda^v)] \qquad (2.68)$$

Bu bölümde bahsedilen kelime tanıma sistemi basit bir Bayesian sınıflandırıcı olarak da düşünülebilir. V adet sınıf düşünelim. Her bir sınıf sözlükteki kelimelere karşılık gelsin. Bilinmeyen bir O gözlem sırasını basit bir Bayesian sınıflandırıcı kullanarak bu sınıflardan biriyle ilişkilendirelim.

Sınıfları $W = \{w_1, w_2, ..., w_v\}$ ile gösterirsek sınıflandırma problemi (2.69)'de verilen ifadenin çözümüdür.

$$\arg\max_i \{P(w_i \mid O)\} \tag{2.69}$$



Şekil 2.11. SMM ile izole kelime tanıma sisteminin blok diyagramı

(2.69) ifadesindeki $P(w_i \mid O)$ direkt olarak hesaplanabilir değildir. Fakat Bayes kuralından faydalanılarak hesaplama (2.70) eşitliğindeki gibi yapılabilir.

$$P(w_i \mid O) = \frac{P(O \mid w_i)P(w_i)}{P(O)} \tag{2.70}$$

(2.70) ifadesinde verilen $P(O)$ ifadesi (evidence) tüm sınıflar için ortak olduğundan hesaplamadan çıkarılabilir. Aynı zamanda önsel olasılık (prior probability) tüm kelimelerin eşit olasılıklarla konuşulacağı varsayılırsa eşitlikten atılabilir. Bunun sonucunda sonsal olasılık (posterior probability) $P(O \mid w_i)$ 'nin sadece benzerlik (likelihood) ifadesi $P(O \mid w_i)$ 'ye bağlı olduğu bulunur. $P(O \mid w_i) = P(O \mid \lambda^i)$ olarak kabul edilirse

Şekil 2.11. 'de verilen kelime tanıma sisteminin bir Bayesian sınıflandırıcı olduğu ortaya çıkmış olur.

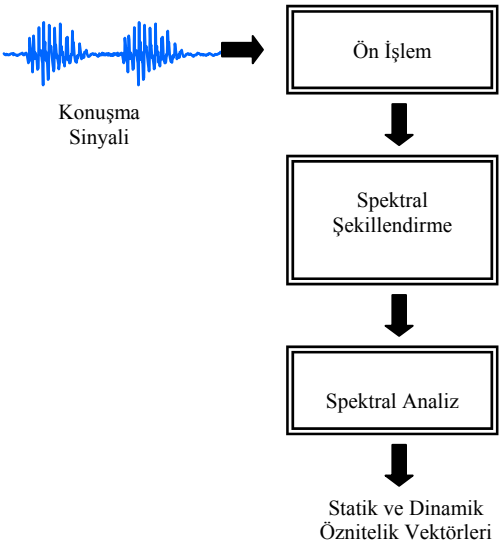
2.5. Konuşma Öznitelik Vektörlerinin Çıkarımı

Havanın ciğerlerden çıkarılması sonucunda, gerilen ses telleri hava akışı sebebiyle titreşmeye başlar. Bu titreşmeyle oluşan hemen hemen periyodik (quasi-periodic) darbeler ses yolundan geçerek filtrelenilir ve ötümlü sesleri üretirler. Çene, dil, dudak ve damak gibi çeşitli artikülatörlerin değişik pozisyonları değişik seslerin oluşturulmasını sağlar. Ötümlü seslere örnek olarak 'a', 'e' vb. verilebilir. Ses telleri serbest durumdayken, bir sıkışma bölgesinden geçen veya bir kapanma noktasının ardında oluşan basınçla aniden bırakılan hava akışı ötümsüz sesleri oluşturur. Sıkışma veya kapanma bölgesinin yeri değişik sesleri meydana getirir. 'b', 'c' ve 'd' ötümsüz seslere örnek olarak verilebilir. Konuşma sinyali ötümlü ve ötümsüz sesler ile bu iki genel ses türünün birbirleri arasındaki geçişlerinden oluşan bir dizidir. Bu sinyalin zamanla değişimi artikülatörlerin fiziksel yapılarından dolayı yavaştır (5 – 100 ms).

İdeal olarak öznitelik vektörleri benzer sesleri bir birinden ayırt edebilecek bilgi taşımalı, çok fazla eğitim datasına ihtiyaç duyulmadan akustik modellerin oluşturulmasını sağlamalı, konuşmacıdan ve konuşulan ortamdan bağımsız olarak aynı konuşma sinyali için aynı sonuçları verebilmelidir. Bu vektörleri çıkarmak için bilinen optimum bir yöntem yoktur.

Konuşma öznitelik (feature) vektörlerinin çıkarılması konuşma tanıma sistemlerinin ilk işlem bloğudur. Bu işlemle ilgili literatürde geçen bir çok algoritma vardır. Bu algoritmalara örnek olarak LPC (Linear Predictive Coding), MFCC (Mel Frequency Cepstral Coefficients) , PLP (Perceptual Linear Prediction) 'yi verebiliriz. Bu yöntemlerin çoğu konuşma sinyalini insanın duyma sistemini emule ederek duyum olarak anlamlı (perceptually meaningful) parametre vektörlerine çevirir.

Konuşma öznitelik vektörlerinin çıkarılması genel olarak üç ana bloktan oluşur. Bunlar sırasıyla ön işlem bloğu, spektral şekillendirme, spektral analizdir. Bu işlemlere ait blok diyagram **Şekil 2.12**.'de gösterilmiştir.



Şekil 2.12. Öznitelik vektörlerinin çıkarımı

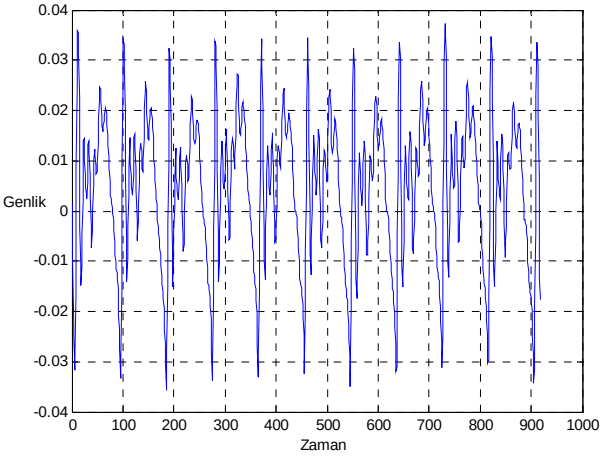
2.5.1. Ön işlem bloğu

Konuşma sinyalinin geldiği kanaldan veya A/D çevirimden kaynaklanan DC ofset, gürültü gibi bir takım istenmeyen etkilerin giderilmesi amacıyla kullanılır. A/D çevirim sırasında oluşan DC ofset, sinyal ortalamasının sinyalden çıkarılmasıyla giderilebilir. Kanaldan kaynaklanan yüksek frekanslı gürültü bileşenleri konuşma sinyalinin 20 Hz – 20 kHz arasında değiştiği göz önüne alınarak ve uygun filtreler kullanılarak uzaklaştırılabilir.

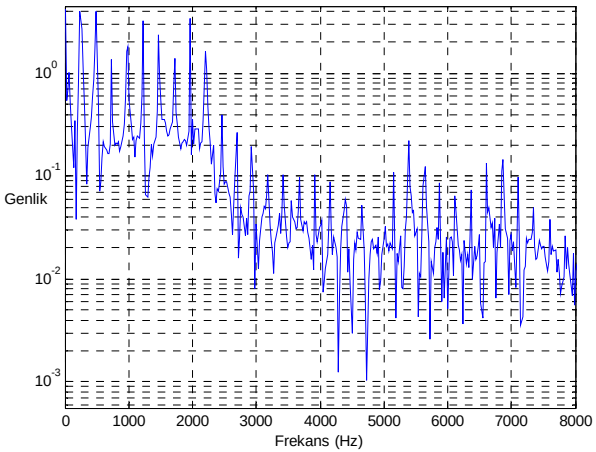
2.5.2. Spektral şekillendirme

İnsanın ses üretim sistemi alçak geçiren bir yapıya sahiptir. Konuşma sisteminin fiziksel özelliğinden dolayı, konuşma sinyalinin ötümlü bölümlerinde -20 dB/decade 'lik spektral negatif bir eğim vardır (Deller *et*

al. 1993, Markel *et al.* 1980). **Şekil 2.14.** 'de verilen 'a' ötümlü sesine ait frekans bölgesi eğrisinden bu durum açıkça görülmektedir.

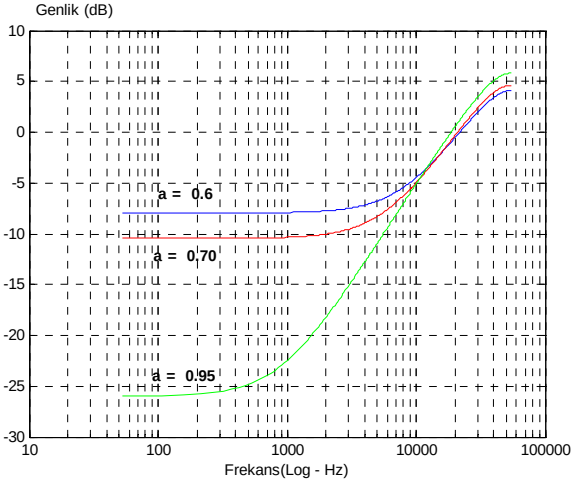


Şekil 2.13. 'a' sesinin zaman bölgesi eğrisi



Şekil 2.14. 'a' sesinin frekans bölgesi eğrisi

Bu etkiyi ortadan kaldırmak için konuşma sinyali 1. dereceden 20 dB/decade 'lik eğimli frekans tepkisine sahip FIR bir filtre ile filtrelenir.



Şekil 2.15. Önvurgu filtresinin frekans tepkisi

Bu filtreye önvurgu (pre-emphasis) filtresi denir. Genel olarak transfer fonksiyonu (2.71) eşitliğindeki gibidir.

$H(z) = 1 - az^{-1}$,$0.95 \le a \le 0.97$ (2.71)

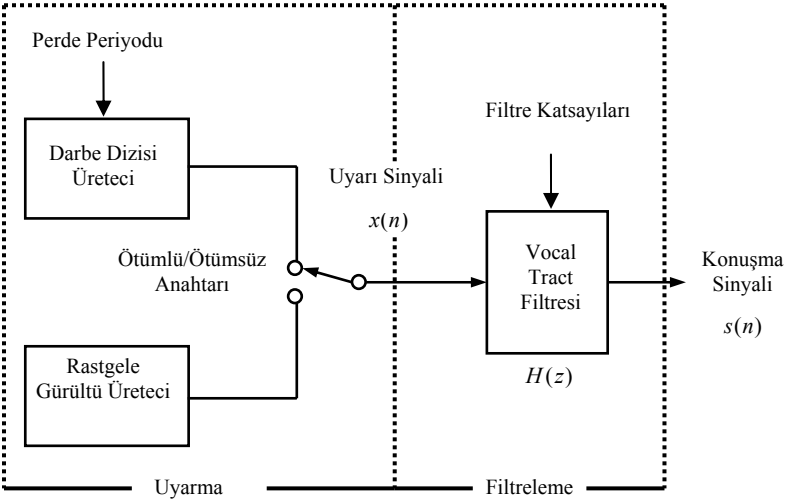
Değişik a değerleri için H(z) filtresinin frekans tepkisi Şekil 2.15'de verilmiştir.

2.5.3. Spektral analiz

Konuşma sinyali, ötümlü, ötümsüz ve bunların birbirleri arasındaki geçişlerin birleşiminden oluşan bir sinyaldir. Ötümlü sesler hemen hemen periyodik bir yapıya sahiptir. **Şekil 2.13.**'deki 'a' ötümlü sesinin durağan haldeki zaman bölgesi eğrisinden bu durum açıkça görülmektedir. Ötümlü seslerin periyodik yapısı ses tellerinin titreşim frekansıyla doğrudan ilgilidir. Ötümlü seslerdeki bu periyoda perde periyodu (Pitch Period) denir. Ötümsüz sesler gürültüye benzer, düşük genlikli, anlam içeren ancak

periyodik bir yapısı olmayan (ses tellerinin titreşmediği) seslerdir. Bunlara ek olarak konuşma sırasında ses etkinliğinin olmadığı aralıklar vardır.**Şekil 2.14.** 'de ki 'a' ötümlü sesinin frekans bölgesi eğrisinden spektrumdaki rezonans noktalarını net olarak izlenmektedir. Şekildeki 500 Hz, 1200 Hz, 2000 Hz ve 3500 Hz noktaları rezonans noktalarıdır. Bu noktalarda spektrumun zarfı tepe yapmıştır. Bu rezonanslar ses yolundaki değişik akustik çukur bölgelerinden oluşan artikülatörlerin bir sonucudur. Rezonansların yeri ses yolunun fiziksel boyutlarına ve şekline bağlıdır. Yani artikülatörlerin değişik fiziksel hallerinde ses yolu değişir ve bunun sonucunda rezonans frekanslarının yerleri de değişir. Rezonans frekansları aynı zamanda tüm spektrumun formunu da belirler ve konuşma bilimcileri tarafından formant olarak adlandırılırlar.

Bu bilgiler ışığında insan konuşma üretim mekanizması **Şekil 2.16.**'daki gibi modellenebilir.



Şekil 2.16. Konuşma sinyali üretim modeli

Şekil 2.16.'daki konuşma üretim modelinden konuşma sinyalinin uyarı sinyalinin, katsayıları zamanla değişen ses yolu filtresiyle filtrelenmesi sonucu oluştuğu görülmektedir. Modelden de anlaşıldığı gibi konuşma sinyalinin spektrumunun uyarı sinyalinin ve zamanla katsayıları değişen ses yolu filtresinin frekans tepkisinin özelliklerini yansıtması beklenir ve

dolayısıyla konuşma sinyalinin spektrumu zamanla değişen bir yapıya sahiptir. Periyodik veya durağan yapıya sahip sinyaller için uygun olan standart Fourier dönüşümü konuşma sinyaline doğrudan uygulanamaz. Bununla birlikte konuşma sinyalinin artikülatörlerin yapısından dolayı enerji, sıfır geçiş korelasyon gibi bir takım geçici özelliklerinin 15-30 ms 'lik aralıklarda sabit olduğu varsayılır. Bu varsayımla spektral analiz bu aralıklarda konuşma sinyalinin pencerelenmesi ile yapılır. Bu şekilde yapılan spektral analize kısa zamanlı spektral analiz (Short Time Spectral Analysis) denir.

2.5.4. Pencereleme

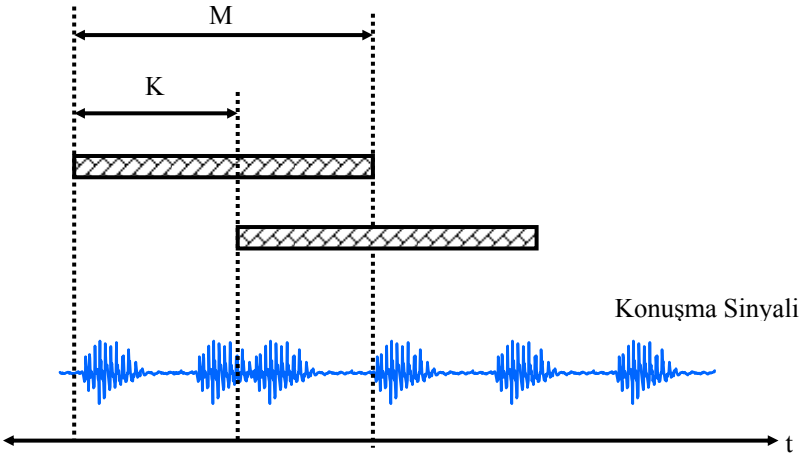
Pencereleme, sinyalin zaman bölgesinde bir pencere fonksiyonu $w(n)$ ile çarpılması işlemidir. Bu işlem ile sinyalin işlenmek istenilen bölümü alınır, diğer bölümleri ise sıfırlanır. İdeal olarak kullanılan pencerenin frekans tepkinin çok dar bir ana lobunun ve hiçbir yan lobunun olmaması gerekir. Ana lobun dar olması frekans çözünürlüğünü arttırır. Yan lobların olmaması ise frekans sızıntısının olmasını engeller. Böyle bir pencere pratikte mümkün değildir. Bu nedenle bu etkiler göz önüne alınarak uygulamaya göre uygun bir pencere seçilir. Dikdörtgen, Hanning, Hamming, Blackman, Kaiser vb. bir çok pencere vardır. Konuşmanın spektral analizinde yaygın olarak Hamming pencere kullanılır. Hamming penceresinin tanımı (2.72) eşitliğinde verilmiştir.

$$w(n) = \begin{cases} 0.54 - 0.46\cos(2\pi \dfrac{n}{N-1}) & ;0 \le n \le N-1 \\ 0 & ;\text{diğer} \end{cases} \qquad (2.72)$$

(2.72) eşitliğindeki N pencerenin uzunluğunu belirler. Pencere uzunluğu konuşmanın perde periyodu göz önüne alınarak seçilmeli ve yapılan kısa zamanlı spektral analizde harmonik yapı net olarak elde edilmelidir. N çok büyük seçilmesi durumunda (birkaç perde periyodu boyunca) sinyalin bu aralıktaki spektrumu yumuşatılacağından (smoothing, avareging) spektrumdaki değişimler izlenemeyecektir. N küçük seçilmesi halinde ise harmonikler net olarak elde edilmeyebilir. N pratik sistemlerde 15-30 ms 'ye karşılık gelecek şekilde seçilir.(Deller *et al.* 1993, Markel *et al.* 1980).

Pencereleme işleminde önemli olan diğer bir parametre ise pencereleme işleminin hangi sıklıkta yapılacağıdır. Diğer bir deyişle ardışık iki ardışık pencereleme arasında geçen sürenin büyüklüğünün ne olacağıdır. Pencere kalma süresi K aynı zamanda ardışık iki pencere arasındaki örtüşmenin oranını da belirler.

$$\%\text{Örtüşme} = \left(\frac{\text{N-K}}{\text{N}}\right)\cdot 100 \qquad (2.73)$$



Şekil 2.17. Konuşma sinyalinin pencerelenmesi

Bu parametrelere ait grafiksel açıklama **Şekil 2.17.**'de verilmiştir. K insan ses üretim mekanizmasındaki artikülatörlerin değişim hızı göz önüne alınarak seçilir. Genel olarak pratik sistemlerde K 10-20 ms' ye karşılık gelecek şekilde seçilir (Deller *et al.* 1993, Markel *et al.* 1980).

2.5.5. Doğrusal öngörümsel kodlama (Linear Predictive Coding, LPC)

LPC analizinde Şekil 2.16.'da verilen ses üretim modelindeki ses yolu filtresi $H(z)$ (2.74) eşitliğinde'de verilen yapıdaki bir filtreyle modellenir. Bu filtre eşitlikten de görüldüğü gibi sadece kutuplardan oluşmaktadır.

$$H(z) = \frac{G}{1 - \sum_{j=1}^{p} a_j z^{-j}} = \frac{G}{A(z)} \tag{2.74}$$

$$A(z) = 1 - \sum_{j=1}^{p} a_j z^{-j} \tag{2.75}$$

Burada p LPC analizinin derecesidir. (2.74)'de verilen $H(z)$ filtresi zaman bölgesinde ifade edilirse (2.76)'da verilen LPC fark eşitliği elde edilir.

$$s(m) = Gx(m) + \sum_{j=1}^{p} a_j s(m-j) \tag{2.76}$$

(2.76)' da ki LPC fark eşitliğinden her hangi bir andaki ses sinyalinin o andaki G ile ağırlıklandırılmış uyarı sinyali ile daha önceki anlardaki filtre katsayılarıyla ağırlıklandırılmış ses sinyalinin toplamı olarak ifade edildiği ortaya çıkar. Bu eşitlik göz önüne alındığında, LPC analizi bir $s(n)$ sinyali verildiğinde filtre katsayıların bulunmasıdır. Filtre katsayılarının sinyalin belli aralıklarında sabit olduğu varsayılır ve LPC analizi sinyalin durağan olduğu bu aralıklarda yapılır. Bu aralıklar Bölüm 2.5.4.'de anlatılan pencereleme yöntemiyle elde edilir.

$$s_n(m) = s(m+n)w(m) \tag{2.77}$$

Burada $w(m)$ (2.72)'de verilen pencere sinyalidir.

Tahmini filtre katsayıları α_j olarak gösterilirse hata veya artık (error, residual) (2.78) eşitliğindeki gibi verilebilir.

$$e(m) = s_n(m) - \sum_{j=1}^{p} \alpha_j s_n(m-j) \tag{2.78}$$

Filtre katsayıları, ortalama kare hatası minimum olacak şekilde aşağıdaki gibi hesaplanabilir.

$$E = \sum_m e^2(m) = \sum_m \left[s(m) - \sum_{j=1}^{p} \alpha_j s(m-j) \right]^2 \tag{2.79}$$

(2.78)'de verilen hata, sinyal $0 \le m \le N-1$ aralığında pencerelendiğinden $0 \le m \le N-1+p$ aralığında sıfırdan farklıdır. Dolayısı ile (2.79) 'da verilen eşitlikte m' in sınırları $0 \le m \le N-1+p$ alınır ve (2.80)'de verilen varsayım yapılır.

$$s_n(m) = \begin{cases} 0, & \text{m<0,m} \ge N \\ s(m)w(n+m), & 0 \le \text{m} \le \text{N-1} \end{cases} \tag{2.80}$$

(2.80) eşitliğinde verilen varsayımla sinyal $0 \le m \le p-1$ rasgele sıfıra eşitlendiğinden başlangıçta öngörüm hatası fazla olacaktır. Yine aynı şekilde $N \le m \le N+p-1$ aralığında sinyal sıfırdan farklı olmasına rağmen sıfıra eşitlendiğinden sonlardaki öngörüm hatası da fazla olacaktır.

$$\frac{dE}{d\alpha_i} = 0, \qquad \text{i=1,2,...,p} \tag{2.81}$$

(2.81) eşitliğndeverildiği gibi E 'nin tahmini filtre katsayılarına göre kısmi türevleri alınıp sıfıra eşitlenirse .(2.86) elde edilir.

$$2s_n(m-i)\sum_{m=0}^{N+p-1}\left[s_n(m)-\sum_{j=1}^{p}\alpha_j s_n(m-j)\right]=0, \quad 1\le \text{i} \le \text{p} \tag{2.82}$$

$$\sum_{m=0}^{N+p-1}\left[s_n(m)s_n(m-i)-s_n(m-i)\sum_{j=1}^{p}\alpha_j s_n(m-j)\right]=0, 1\le \text{i} \le \text{p} \tag{2.83}$$

$$\sum_{m=0}^{N+p-1} s_n(m)s_n(m-i)=\sum_{j=1}^{p}\sum_{\text{m=0}}^{N+p-1}\alpha_j s_n(m-j)s_n(m-i), 1\le \text{i} \le \text{p} \tag{2.84}$$

$$\phi_n(i,j)=\sum_{m=0}^{N+p-1} s_n(m-i)s_n(m-j), 1\le \text{i} \le \text{p},\ 0\le \text{j} \le \text{p} \tag{2.85}$$

$$\sum_{j=1}^{p}\alpha_j\phi_n(i,j)=\phi_n(i,0), \quad 1\le \text{ j} \le \text{p} \tag{2.86}$$

(2.85)'de verilen eşitlik $s_n(m)$ 'in $0 \le m \le N-1$ aralığı dışında sıfır olduğu göz önüne alınarak (2.87) eşitliğindeki gibi yazılabilir.

$$\phi_n(i,j)=\sum_{m=0}^{N-1-(i-j)} s_n(m)s_n(m+i-j), 1\le \text{i} \le \text{p},\ 0\le \text{j} \le \text{p} \tag{2.87}$$

(2.87) eşitliği kısa zamanlı oto korelasyon fonksiyonu cinsinden (2.88) eşitliğine dönüştürülebilir. Bu durumda (2.86) eşitliği (2.90) eşitliği şeklinde ifade edilebilir.

$$\phi_n(i,j)=R_n(|i-j|), \quad 1\le \text{i} \le \text{p},\ 0\le \text{j} \le \text{p} \tag{2.88}$$

$$R_n(j)=\sum_{m=0}^{N-1-j} s_n(m)s_n(m+j) \tag{2.89}$$

$$\sum_{j=1}^{p}\alpha_j R_n(|i-j|)=R_n(i), \quad 1\le \text{ j} \le \text{p} \tag{2.90}$$

(2.90) eşitliği matris formunda (2.91) 'de olduğu gibi yazılabilir.

$$\begin{bmatrix} R_n(0) & ,R_n(1), & \dots & ,R_n(p-1) \\ R_n(1) & ,R_n(0), & \dots & ,R_n(p-2) \\ . & . & . & \\ . & . & . & \\ . & . & . & \\ R_n(p-1), & R_n(p-2), & \dots & ,R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ . \\ . \\ . \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ . \\ . \\ . \\ R_n(p) \end{bmatrix} \quad (2.91)$$

(2.91) eşitliğindeki p x p boyutlu Toeplitz bir matristir çünkü simetrik ve her hangi köşegendeki tüm elemanlar bir birine eşittir. (2.91) eşitliğinde verilen denklem sisteminin çözümü p x p boyutlu matrisin tersi alınarak bulunabilir. Bu yola alternatif olarak matrisin Toeplitz karakteristiği kullanılarak daha etkili ve önyinelemeli bir yöntem olan Durbin 'nin algoritması kullanılması daha avantajlıdır.

1) $E_n^{(0)} = R_n^{(0)}$ (2.92)

2) $1 \le i \le p$

3) $$k_i = \frac{\left[R_n(i) - \sum_{j=1}^{i-1} \alpha_j^{i-1} R_n(i-j) \right]}{E_n^{i-1}} \quad (2.93)$$

4) $\alpha_i^{(i)} = k_i$ (2.94)

5) $\alpha_j^{(i)} = \alpha_j^{(i-1)} - k_i \alpha_{i-j}^{(i-1)}, \qquad 1 \le \text{j} \le \text{i-1}$ (2.95)

6) $E_n^{(i)} = (1-k_i^2) E_n^{(i-1)}$ (2.96)

(2.93) 'den (2.96) 'ya kadar olan eşitliklerin $1 \le i \le p$ aralığında önyinelenmesi sonucu elde edilen $\alpha_j^{(p)}$ değerleri aradığımız çözümlerdir.

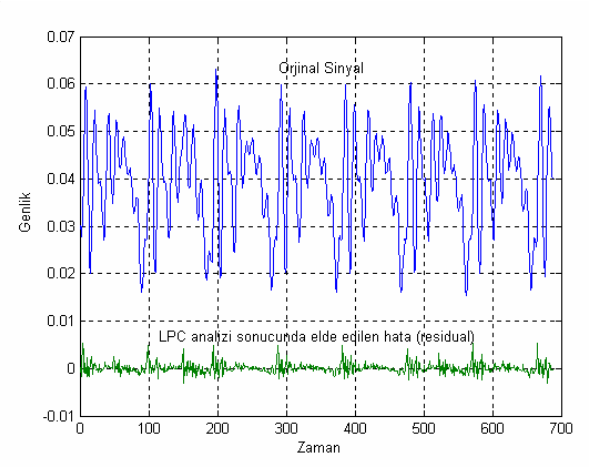
$\alpha_j = \alpha_j^{(p)}, \qquad 1 \le \text{j} \le \text{p}$ (2.97)

Hesaplanan bu sonuçları (2.76) eşitliğinde verilen LPC fark eşitliğiyle eşlenirse filtre katsayıları α_j iken, uyarı sinyalinin $e(n)$ olması gerektiğini bulunur.

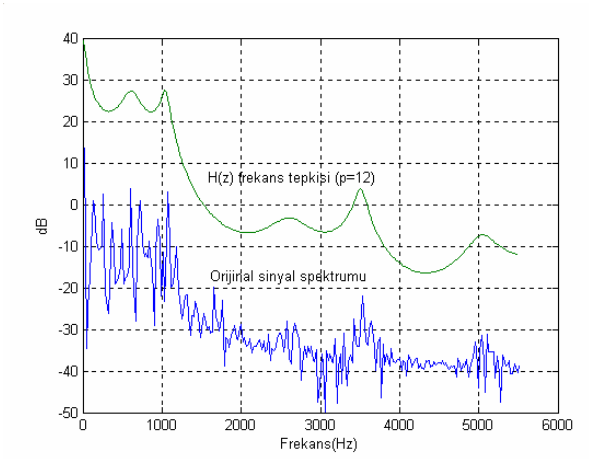
$\alpha_j = a_j$ ise $e(n) = Gx(n)' dir.$ (2.98)

Şekil 2.18.'deki grafikte 11025 Hz 'de örneklenmiş ötümlü orijinal ses sinyalini ve LPC analizi sonucunda elde edilen hata sinyalini görmekteyiz.

Şekil 2.19. 'deki orijinal ses sinyalinin kısa zamanlı spektrumu ve H(z) filtresinin frekans tepkisi verilmiştir. Orijinal spektrumla H(z) filtresinin



Şekil 2.18. Orijinal konuşma sinyali ve LPC analizi sonucunda elde edilen hata



Şekil 2.19. Orijinal sinyal spektrumu ve H(z) frekans tepkisi

frekans tepkisi arasında ki ilişki net olarak izlenmektedir. H(z) filtresinin frekans tepkisi orijinal sinyalin spektrumunun zarfının yumuşatılmış bir hali

olduğu net olarak görülmektedir. Bu nedenle LPC analizi kısa zamanlı spektrum tahmini olarak da düşünülebilir.

2.5.6. Cepstral analiz

Bölüm 2.1.3.' de bahsedilen konuşma üretim modelinde konuşma sinyali, uyarı sinyalinin ses yolu filtresi ile filtrelenmesi sonucu oluşmaktadır. Uyarı sinyali ile ses yolu filtresinin katlanması sonucunda oluşan bu sinyalden uyarı sinyalini ve ses yolu filtresinin dürtü tepkisini ayırmak konuşma kodlama ve tanıma sistemleri için önemlidir.

$$s(n) = e(n) * h(n) \tag{2.99}$$

Burada $s(n)$ konuşma sinyali, $e(n)$ uyarı sinyali ve $h(n)$ ses yolu filtresinin dürtü tepkisidir.

(2.99) eşitliğinde verilen konvolusyon halindeki ifade frekans bölgesine taşınırsa (2.100) eşitliği elde edilir.

$$S(\omega) = E(\omega)H(\omega) \tag{2.100}$$

(2.100) eşitliğinden de görüldüğü gibi iki sinyalin spektrumu çarpım halindedir. Bu nedenle standart lineer filtreleme yöntemleriyle bu iki sinyalin bir birinden ayrılması mümkün değildir. Lineer filtreleme yöntemiyle bu iki sinyali birbirinden ayırmak için (2.100)'de verilen çarpımsal ifade toplamsal bir ifadeye dönüştürülebilir. Bu işlemi yapmak için (2.100) eşitliğinin her iki tarafının logaritması alınırsa (2.101) eşitliği elde edilir.

$$\log(S(\omega)) = \log(E(\omega)) + \log(H(\omega)) \tag{2.101}$$

(2.101) eşitliğinden görüldüğü gibi (2.100) eşitliğindeki çarpımsal ifade toplamsal bir ifadeye dönüşmüştür.

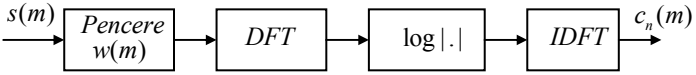
$$e(n) * h(n) = \log(\Im(e(n)) + \log(\Im(h(n)) \tag{2.102}$$

Burada $\Im$ DTFT (Discrete Time Fourier Transform) operatörüdür. (2.102) eşitliğinin sol tarafındaki katlama ifadesinin sağ tarafta toplamaya dönüştü görülebilir. Bu tip bir dönüşüme homomorphic dönüşüm denir. Cepstrum (real cepstrum) homomorphic bir dönüşümdür ve ayrık zamanda (2.103)'deki gibi tanımlanır.

$$c(m) = \Im^{-1}\left[\log\left|\Im\left[s(m)\right]\right|\right] = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log\left|S(\omega)\right| e^{j\omega m} d\omega \tag{2.103}$$

Konuşma sisteminde ses yolu filtresi ve uyarı sinyali zamanla değişir. Bu değişimden dolayı cepstral analiz konuşma sinyalinin durağan olduğu kısa

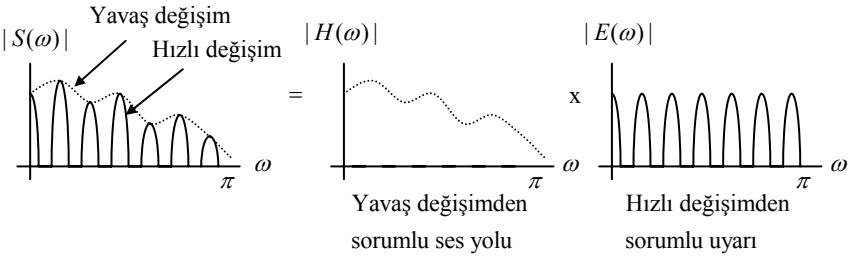
zamanlı aralıklarda yapılır. Bu işleme ait blok diyagram **Şekil 2.20.** 'de verilmiştir.



Şekil 2.20. Konuşma sinyalinin kısa zamanlı cepstral analizi

Şekil 2.20.'de verilen blok diyagram sözle şöyle anlatılabilir. Konuşma sinyali önce pencerelenir. Daha sonra pencerelenen bu sinyale DFT (Discrete Fourier Transform) uygulanır. Bunun sonucunda elde edilen frekans bileşenlerinin genliği hesaplanarak logaritması alınır ve daha sonra bu değerlerin IDFT'si (Inverse Discrete Time Transform) hesaplanır. Bunun sonucunda pencerelenmiş s(n) sinyalinin cepstrum değerleri elde edilir. Cepstral analiz sonucunda elde edilen yeni bölge literatürde quefrency bölgesi olarak adlandırılır.

Ses yolu filtresi konuşmanın spektrumunun yavaş değişen kısmını belirler. Yani formatların yerlerini belirleyerek spektrumun zarfını oluşturur. Uyarı sinyali ise spektrumun hızlı değişen kısmını maydana getirir. Bu durum **Şekil 2.21.** 'de anlatılmıştır.



Şekil 2.21. Ses yolu ve uyarı sinyalinin spektruma etkileri

Cepstral analizde **Şekil 2.21.**'de bahsedilen sinyallerin spektrumlarının genlik değerlerinin logaritmları toplamının IDFT'si alınır. Bu işlem sonucunda elde edilen yeni bölgede bu iki sinyal ayrı bölgelere düşer. Çünkü IDFT'si alınan sinyal yavaş ve hızlı değişen iki parçadan

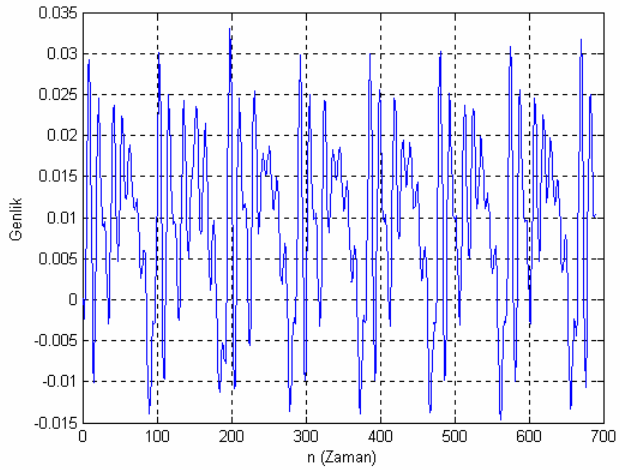
oluşmaktadır. Dolayısı ile düşük indisli quefrency bileşenleri ses yolu filtresine, büyük indisliler ise uyarı sinyaline ait olacaktır.

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{\frac{-j2\pi kn}{N}} \qquad (2.104)$$

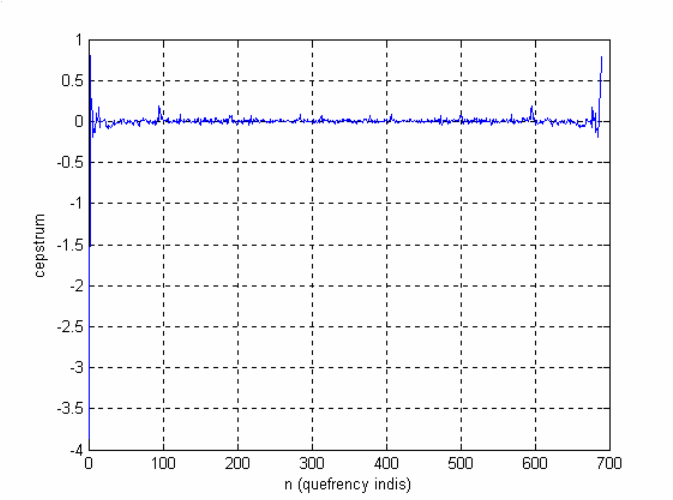
$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{\frac{j2\pi kn}{N}} \qquad (2.105)$$

(2.104) ve (2.105) eşitliklerinde bir x(n) sinyali için DFT ve IDFT eşitlikleri verilmiştir. Burada N pencerelenmiş x(n) sinyalinin uzunluğudur.

Şekil 2.22. ve **Şekil 2.23.** 'de durağan haldeki 'a' ötümlü sesine ait zaman ve cepstrum bölgesi grafikleri verilmiştir.

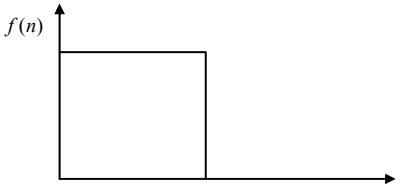


Şekil 2.22. 'a' ötümlü sesi zaman eğrisi



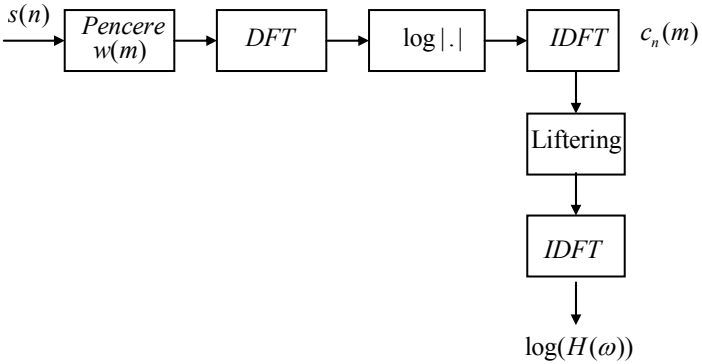
Şekil 2.23. 'a' ötümlü sesine ait cepstrum

Quefrency bölgesinde lineer filtreleme yöntemleriyle ses yolu filtesi ve uyarı sinyali bir birinden ayrılabilir. Düşük quefrency indisleri ses yolu filtresine bükük qufrency indisleri ise uyarı sinyaline aittir. Bu nedenle **Şekil 2.24.** verilen bir pencere kullanılarak düşük quefrency bileşenleri yani ses yolu filtresine ait cepstrum elde edilebilir. Bu işlem literatürde low-time liftering olarak bilinir. Liftering sonucunda elde edilen cepstrum değerlerinin DFT'si alınarak tahmini ses yolu filtresinin frekans spektrumunun logaritması elde elde edilir. **Şekil 2.26.**'da 'a' ötümlü sesine ait orijinal spektrum ve düşük indisli cepstrum değerlerinden elde edilen spektrum verilmiştir. Burada ilk 32 quefrency bileşini kullanılmıştır. Düşük indisli cepstrum değerlerinden elde edilen spektrum ile orijinal spekrumun zarfı arasındaki benzerlik **Şekil 2.26.**'de net olarak görülmektedir. Bu işlem ait blok diyagram **Şekil 2.25.** 'de verilmiştir.

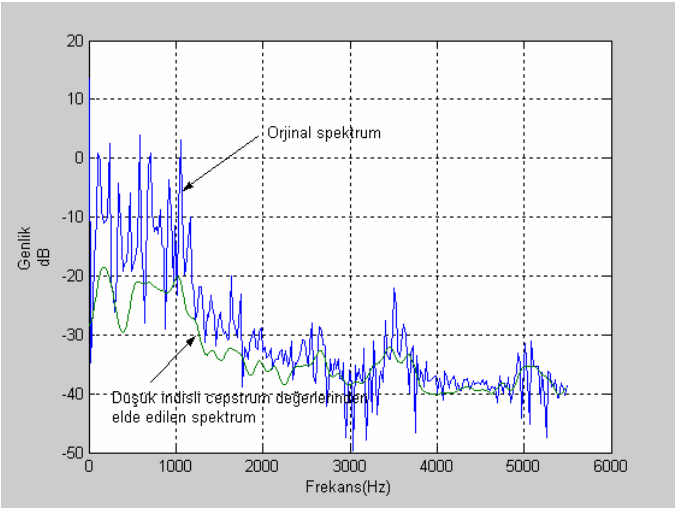


Şekil 2.24. Low-time lifter

Sonuç olarak düşük indisli cepstrum değerleri LPC analizindeki filtre katsayıları gibi konuşma spektrumunun zarfını temsil edebilir.



Şekil 2.25. Cepstrum yöntemiyle ses yolu frekans tepkisi, $H(\omega)$ 'nin elde edilmesi



Şekil 2.26. Cepstrum yöntemiyle formant analizi

2.5.7. LPC' den türetilen cepstrum

LPC analizi konuşma sinyalinden spektral bilginin elde edilmesi için uzun yıllardır kullanılan en popüler yöntemlerden biri olmuştur ve konuşma tanıma sistemlerinde yaygın olarak kullanılmıştır. Fakat pratikteki konuşma tanıma sistemlerinden LPC analizinin bir takım dezavantajları olduğu ve konuşma tanıma sisteminin performansını düşürdüğü görülmüştür. Bu performans düşüklüğü sinyal spektral sıfırlara sahip olduğu zaman gerçekleşmektedir. Bu spektral sıfırlar sinyalin iletimi, filtrelenmesi ve uygun olmayan ön-vurgu kullanımı aşamasında meydana gelebilir. LPC, spektrumu sadece kutuplarla modellediği için spektrumdaki sıfırların olduğu bölgede spektrum iyi bir şekilde modellenememektedir. Diğer bir anlatımla LPC analizi spektrumdaki tepeleri iyi bir şekilde modelleyebilmekte fakat spektrumdaki çukur bölgeleri modelleyememektedir. (Juang *et al.* 1987). **Şekil 2.19.** 'de bu durum açıkça görülmektedir. Bununla birlikte (Deller *et al.* 1993)'un bahsettiği gibi LP parametreleri aynı konuşma için konuşmacıdan konuşmacıya büyük farklılıklar göstermektedir. Bu da konuşmacıdan bağımsız konuşma tanıma sistemleri için performans düşüklüğüne sebep olmaktadır.

LP parametrelerindeki varyasyonları ortadan kaldırmak için cepstrum metotları geliştirilmiştir. Bu metotlar kısa zamanlı LP parametrelerini cepstral parametrelere çevirmeye dayalı metotlardır.

LP 'den türetilen cepstrum, LPC analizi sonucunda elde edilen ses yolu filtresinin dürtü tepkisinin cepstrum değerlerinin hesaplanması sonucu bulunur fakat daha etkili bir şekilde bölüm 2.5.5.'de anlatılan LP filtre katsayıları $a_i, 1 \le i \le p$ ve $R(0)$ otokorelasyonu kullanılarak (2.106)'deki gibi özyinelemeli bir şekilde hesaplanabilir.

$c_0 = R(0)$ olmak üzere

$$c_m = \begin{cases} a_m + \sum_{k=1}^{m-1} \frac{k}{m} c_k a_{m-k}, 1 < m < p \\ \sum_{k=m-p}^{m-1} \frac{k}{m} c_k a_{m-k}, m > p \end{cases} \qquad (2.106)$$

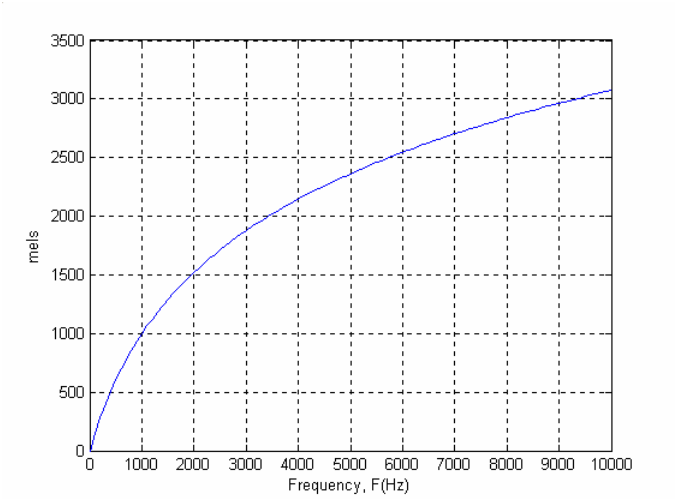
2.5.8. Mel-cepstrum

Mel – cepstrum anlatılmadan önce "mel" teriminin tanımının yapılması konunun daha iyi anlaşılmasını sağlayacaktır. Mel bir tonun algılanan frekansının bir ölçüsüdür. Bu ölçü tonun fiziksel frekansıyla lineer olarak örtüşmez çünkü insan duyma sistemi frekansları lineer olarak algılamaz. Mel skalası Stevens ve Volkman (1940) tarafından deneysel olarak açıkça ortaya çıkarılmıştır. Deney şu şekilde yapılmıştır. Keyfi olarak 1000 Hz' lik bir ton referans olarak seçilmiş ve buna 1000 mels denilmiştir. Daha sonra bu tonu dinleyen dinleyicilerden algıladıkları frekans iki katına çıkana kadar fiziksel frekansı değiştirmeleri istenmiştir ve bu frekans 2000 mels olarak adlandırılmıştır. Ardından bu deney referans frekansın 10, 100 ,0.5, 0.1 vb. katları için tekrarlanmış ve bu frekanslar sırasıyla 10.000 mels, 100.000 mels, 500 mels ve 100 mels vb. adlandırılmıştır. Bunun sonucunda gerçek fiziksel frekans (Hz) ile algılanan frekans (Mel) arasında bir eşleme yapma imkanı doğmuştur. Bu eşlemenin 1 kHz 'nin altında lineer, yukarısında ise logaritmik olduğu görülmüştür.

Bu eşleme konuşma tanıma sistemlerinde (2.107)'de verilen yaklaşımla kullanılmaktadır ve bu yaklaşımla gerçek fiziksel frekans (Hz) ve algılanan frekans (mel) arasındaki eğri **Şekil 2.27.**' de verilmiştir.

$$F_{mel}(f_{Hertz}) = 2595\log_{10}(1+\frac{f_{Hertz}}{700}) \text{ yada}$$

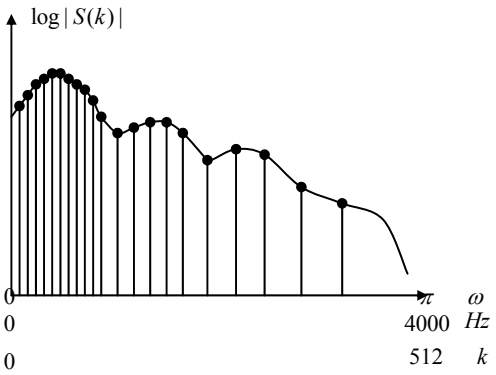
$$F_{mel}(f_{Hertz}) = 1127\ln(1+\frac{f_{Hertz}}{700}) \qquad (2.107)$$



Şekil 2.27. Frekans ve mel arasındaki ilişki

Mel skalası kısa zamanlı cepstruma kolaylıkla uygulunabilir. Çünkü cepstrum analizi doğrudan konuşma sinyalinin genlik spektrumuna uygulanır. Şimdi örnek olarak 8 kHz'de örneklenmiş 30 ms'lik bir konuşma sinyali için mel skalasında kısa zamanlı cepstrum analizi yapılsın. Öncelikle **Şekil 2.20.**'de verilen DFT bloğunda 1024 nokta kullanıldığı varsayılsın. Bu durumda frekans çözünürlüğü yaklaşık olarak 7.8 Hz olur. Şimdi mel skalasına geçmek için örnekleme sonucunda elde edilen 4 kHz lik bölümün mel karşılığı bulunsun. (2.107)'de verilen yaklaşımla bu değer 2840 meldir. Bu durumda 1024 noktada mel skalası hesaplanmak istenirse 5.5 mel çözünürlüğüne sahip olunur. Fakat bu işlemin yapılabilmesi için bu frekans bileşenlerinin bilinmesi gerekir. Örnek olarak 5.5, 11, 16.5, 22 vb. mel değerleri için 3.4, 6.9, 10.3, 13.8 vb. Hz bileşenlerinin bilinmesi gerekir.

Frekans çözünürlüğü 7.8 Hz olduğundan 5.5 mel çözünürlüğe geçilmesi mümkün değildir. Buna çözüm olarak mel çözünürlüğü düşürülebilir. Örnek olarak 20 noktada mel skalası hesaplanmaya çalışılsın. 20 noktada mel çözünürlüğü 142 meldir. Bilinmesi gereken bileşenler 142, 284, 426, 568, 710 vb. mel değerleri için 94, 200, 321, 459, 614 Hz vb. bileşenleridir. Bu bileşenler 7.8 Hz çözünürlükteki frekans bileşenlerinden yaklaşık olarak elde edilebilir. Bu işlem **Şekil 2.28.** 'de gösterilmiştir. Bu işlemden sonra kullanılmayan diğer bileşenler ya sıfıra eşitlenir yada şimdi bahsedilecek olan başka bir fizyoakustik prensible yaklaşılır.



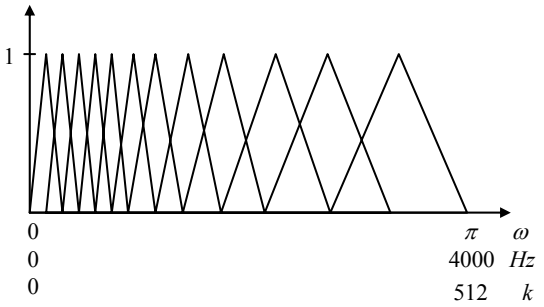
Istenilen Frekansı (Hz)	DFT Frekansı (Hz)	k değeri DFT
69.9	70.3	9
146.8	148.4	19
231.4	234.4	30
324.5	328.1	42
426.8	429.7	55
539.4	539.1	69
663.2	664.1	85
799.3	796.9	102
949.1	945.3	121
1113.8	1117.2	143
1295.0	1296.9	166
1494.3	1492.2	191
1713.5	1710.9	219
1954.6	1953.1	250
2219.8	2218.8	284
2511.4	2507.8	321
2832.2	2835.9	363
3185.1	3187.5	408
3573.1	3570.3	457
4000.0	4000	512

Şekil 2.28. Kısa zamanlı DFT kullanılarak mel-cepstral katsayıların hesaplanması

İnsan duyma sisteminin belli bir frekansı algılaması, bu frekans etrafındaki kiritik bir band içiresinde bulunan frekansların enerjisinden etkilenir (Schroeder 1977). Bununla birlikte bu kritik bandın genişliği frekansa göre değişir.1 kHz'in altındaki frekanslar için 100 Hz'den başlar ve 1 kHz'in üzerinde logaritmik olarak artar. Bu nedenle bazı araştırmacılar, mel-cepstrum hesaplarken mel dağılımlı logaritmik frekans genlik bileşenleri

kullanmak yerine, mel frekansları etrafındaki kritik bantlar içerisinde bulunan toplam enerjinin logaritmasını kullanmayı önermişlerdir. Bu işlem için genlik spektrumu, mel skalasında eşit aralıkla dağılmış ve bir birbiriyle %50 oranında kesişen L adet (genellikle 20 adet kullanılır) band geçiren üçgen filtre ile çarpılır ve çarpım sonucunda her bir filtre (band) içerisinde kalan enerjinin logaritması hesaplanarak bu enerjiler **Şekil 2.20.**'da verilen IDFT bloğuna sokulur. Bu L adet band geçiren filtre literatürde mel filtre bankası, yapılan işlem ise mel filtre bankası analizi olarak bilinir. **Şekil 2.29.**'de mel filtre bankası verilmiştir.

Şekil 2.29. Mel filtre bankası



2.5.9. LPC 'den türetilen cepstrum ile öznitelik vektörlerinin çıkarılması

Bu bölümde anlatılacak öznitelik vektörleri Bölüm 2.5.7.'de anlatılan LPC 'den türetilen cepstrum değerlerine dayanmaktadır.

1) Önvurgu: Örneklenmiş ses sinyali birinci dereceden FIR bir filtreden geçirilerek sinyal spektral olarak düzlenir (Bölüm 2.5.2.).
2) Çerçeveleme: Ardışık N_A örneklik kısım bir çerçeve olarak kullanılır(sinyalin 8 kHz de örneklendiği ve 45 ms 'lik kısmının alındığı düşünülürse 360 örnek olur). Ardından gelen çerçeve bir önceki çevreden K_A örnek öteye yerleştirilir. (K_A'yı 120 olarak seçilirse çerçeveler arasındaki aralık 15 ms olur ve ardışık iki çerçeve bir birbiriyle 30 ms örtüşür.)
3) Pencereleme: Çerçeveler N_A nokta hamming pencere kullanılarak pencerelenir (Bölüm 2.5.4.).

4) LPC analizi: 2. ve 3. aşama sonucunda elde ettiğimiz N_A noktalık pencerelenmiş sinyale p. derecen LPC analizi yapılır.
5) LPC 'den cepstrum değerlerinin türetilmesi: 4. aşama sonucunda elde edilen, LP filtre katsayıları $a_i, 1 \le i \le p$ ve $R(0)$ otokorelesyonu kullanılarak Q. bileşene kadar LPC cepstrum hesaplanır. (Q>p,Q=12)
6) Cepstral ağırlıklandırma: 5. aşamada elde edilen cepstrum değerlerinin düşük indisli bileşenlerinin genliği büyük, yüksek indislerin ise genliği küçüktür. Cepstrum değerlerindeki bu varyans farklılığını ortadan kaldırmak için cepstrum (2.108)'de verilen pencere ile ağırlıklandırılır.

$$W_c(m) = 1 + \frac{Q}{2}\sin(\frac{\pi m}{Q}), \quad 1 \le \mathrm{m} \le \mathrm{Q} \qquad (2.108)$$

$$\hat{c}_l(m) = c_l(m) \cdot W_c(m) \qquad (2.109)$$

Burada $c_l(m)$ cepstral katsayılar $W_c(m)$ ise ağırlıklandırma penceresidir.

7) Delta Cepstrum: Ağırlıklandırılmış cepstral katsayı dizisinin zaman türevi, sonlu uzunlukta çerçevelerden oluşan (2K+1) ve zaman türevi hesaplanan dizi etrafında merkezlenmiş pencere üzerinden birinci dereceden ortogonal polinom yaklaşımıyla hesaplanır.

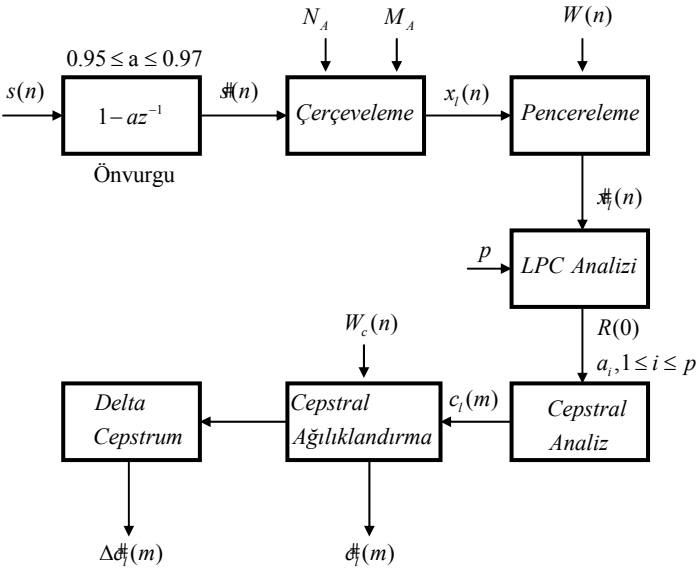
$$\Delta\hat{c}_l(m) = \left[\sum_{k=-K}^{k=K} k \cdot \hat{c}_{l-k}(m)\right].G, \qquad 1 \le \mathrm{m} \le \mathrm{Q} \qquad (2.110)$$

Burada G kazanç terimi $\hat{c}_{l-k}(m)$ ise $(l-k).$çerçevenin $m.$ cepstral katsayısıdır. G $\hat{c}_l(m)$ ve $\Delta\hat{c}_l(m)$'nin varyanslarını eşitlemek için kullanılır (genel olarak 0.375 olarak kullanılır).

8) Bu işlemler sonunda elde edilen Q adet ağırlıklandırılmış cepstral katsayı ve bunların Q adet delta cepstrum değeri birleştirilerek 2Q boyutlu bir vektöre dönüştürülür ve öznitelik vektörü olarak kullanılır (Q=12 durumunda 24 boyutlu bir vektör elde edilir).

$$O_l = \{\hat{c}_l(m), \Delta\hat{c}_l(m)\} \qquad (2.111)$$

Bu işleme ait blok diyagram **Şekil 2.30.** 'da verilmiştir.



Şekil 2.30. LPC'den türetilen cepstrum ve delta cepstrum ile öznitelik vektörlerinin çıkarılması

2.5.10. FFT tabanlı mel-cepstrum ile öznitelik vektörlerinin çıkarılması (MFCC)

LPC 'den türetilen cepstrum değerlerinden başka yaygın olarak kullanılan diğer bir yöntem ise MFCC (Mel Frequency Cepstral Coefficients) yöntemidir. Bu yöntem Bölüm 2.5.8' de anlatılan mel-cepstrum yöntemine dayanır.

1) Önvurgu: Örneklenmiş ses sinyali birinci dereceden FIR bir filtreden geçirilerek sinyal spektral olarak düzlenir (Bölüm 2.5.2.).
2) Çerçeveleme: Ardışık N_A örneklik kısım bir çerçeve olarak kullanılır(sinyal 8 kHz de örneklendiği ve 45 ms 'lik kısmının alındığı düşünülürse 360 örnek olur). Ardından gelen çerçeve bir önceki çevreden K_A örnek öteye yerleştirilir. (K_A 120 olarak seçilirse çerçeveler arasındaki aralık 15 ms olur ve ardışık iki çerçeve bir birbiriyle 30 ms örtüşür.)
3) Pencereleme: Çerçeveler N_A nokta hamming pencere kullanılarak pencerelenir (Bölüm 2.5.4.).

4) FFT : Pencerelenmiş N_A noktalık konuşma sinyalinin FFT 'si alınır ve FFT bileşenlerinin genlikleri hesaplanır.
5) Mel filtre bankası analizi: FFT sonucunda elde edilen genlik spektrumu mel skalasında eşit olarak dağıtılmış ve birbirini %50 oranında kesen üçgen filtre ile çarpılır. Çarpma işlemi sonucunda her bir filtre altında kalan enerji hesaplanır. Bu işlem filtre içerisinde kalan genlik değerlerinin, karşılık gelen filtre kazanclarıyla çarpılıp, bu değerlerin toplanması ile bulunur. Bu işlem sonucunda elde edilen L adet enerji değerinin logaritması alınır.
6) Cepstrum: 5. aşamada elde edilen L adet logaritmik enerji değerinin IDFT'si veya DCT'si alınarak Q. bileşene kadar cepstrum elde edilir.
7) Cepstral ağırlıklandırma: 6. aşamada elde edilen cepstrum değerlerinin düşük indisli bileşenlerinin genliği büyük, yüksek indislerin ise genliği küçüktür. Cepstrum değerlerindeki bu varyans farklılığını ortadan kaldırmak için cepstrum (2.112)'de verilen pencere ile ağırlıklandırılır.

$$W_c(m) = 1 + \frac{Q}{2}\sin(\frac{\pi m}{Q}), \quad 1 \le \mathrm{m} \le \mathrm{Q} \qquad (2.112)$$

$$\hat{c}_l(m) = c_l(m) \cdot W_c(m) \qquad (2.113)$$

Burada $c_l(m)$ cepstral katsayılar $W_c(m)$ ise ağırlıklandırma penceresidir.

8) Delta Cepstrum: Ağırlıklandırılmış cepstral katsayı dizisinin zaman türevi, sonlu uzunlukta çerçevelerden oluşan (2K+1) ve zaman türevi hesaplanan dizi etrafında merkezlenmiş pencere üzerinden birinci dereceden ortogonal polinom yaklaşımıyla hesaplanır.

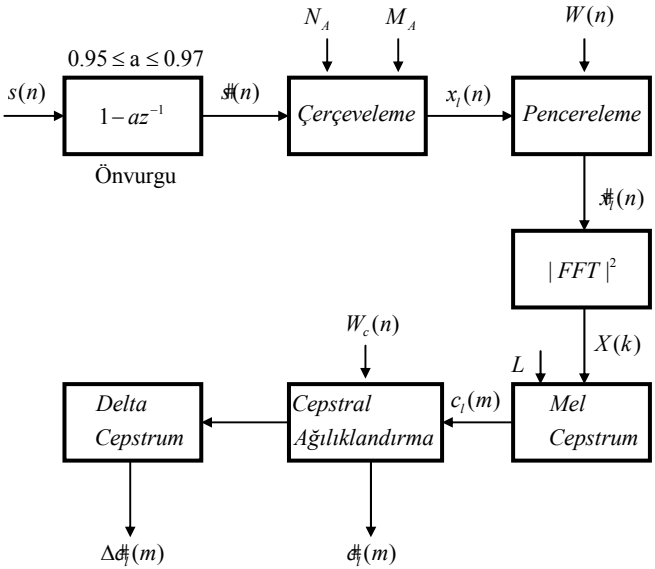
$$\Delta\hat{c}_l(m) = \left[\sum_{k=-K}^{k=K} k \cdot \hat{c}_{l-k}(m)\right].G, \qquad 1 \le \mathrm{m} \le \mathrm{Q} \qquad (2.114)$$

Burada G kazanç terimi $\hat{c}_{l-k}(m)$ ise $(l-k).$çerçevenin $m.$ cepstral katsayısıdır. G $\hat{c}_l(m)$ ve $\Delta\hat{c}_l(m)$'nin varyanslarının eşitlemek için kullanılır (genel olarak 0.375 olarak kullanılır).

9) Bu işlemler sonunda elde edilen Q adet ağırlıklandırılmış cepstral katsayı ve bunların Q adet delta cepstrum değeri birleştirilerek 2Q boyutlu bir vektöre dönüştürülür ve öznitelik vektörü olarak kullanılır(Q=12 durumunda 24 boyutlu bir vektör elde edilir).

$$O_l = \{\hat{c}_l(m), \Delta\hat{c}_l(m)\} \qquad (2.115)$$

Bu işleme ait blok diyagram **Şekil 2.31.** 'de verilmiştir.



Şekil 2.31. FFT'den türetilen mel cepstrum ve delta cepstrum ile öznitelik vektörlerinin çıkarılması

3. MATERYAL VE YÖNTEM

Bu çalışmada yöntem olarak kuramsal temeller bölümünde anlatılan yöntemlerin MATLAB ortamında gerçekleştirilmesi kullanılmıştır. Materyal olarak çeşitli konuşmacılardan alınan ses örnekleri kullanılmıştır.

3.1. MATLAB Ortamında Geliştirilen İzole Kelime Tanıma Yazılımı (ISRTK)

Program **Şekil 3.1.** ve **Şekil 3.3.** 'de verilen iki adet kullanıcı ara yüzüne (GUI) sahiptir. Bunlardan ilki ana GUI diğeri ise konuşma tanıma sistemine ait bir takım konfigürasyonların yapıldığı konfigürasyon GUI'sidir. Bu GUI'lere ait ayrıntılı açıklama ilerleyen bölümlerde verilmiştir. Yazılımın geliştirilmesinde Kevin Murphy tarafından geliştirilen Hidden Markov Model (HMM) Toolbox'dan faydalanılmıştır.

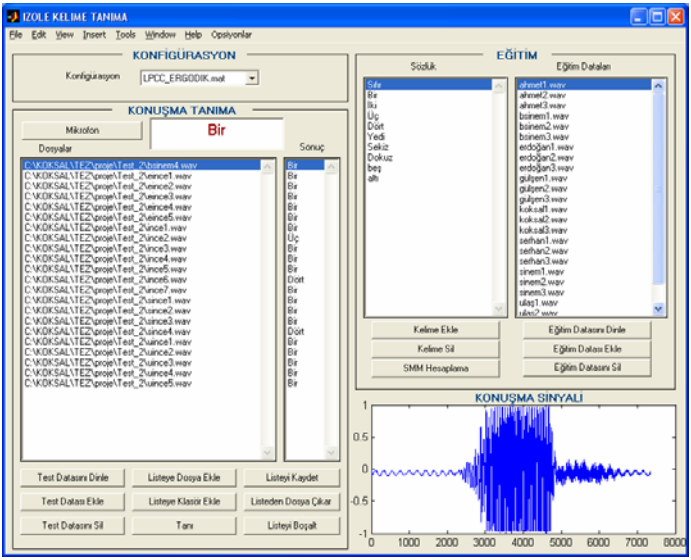
3.1.1. Ana GUI

Konuşma tanıma sistemin temel işlemlerinin yapıldığı GUI'dir. GUI aşağıdaki bölümlerden oluşmaktadır.

- Konfigürasyon seçim bölümü
- Tanıma bölümü
- Eğitim bölümü
- Konuşma sinyali izleme bölümü

GUI'ye ait ekran görünümü **Şekil 3.1.**'de verilmiştir. Konfigürasyon seçim bölümü "KONFİGÜRASYON" başlığı altındaki bölümü kapsar ve sadece bir adet popup menuden oluşmaktadır. Bu menude daha önce konfigürasyon GUI'sinde yapılıp kaydedilmiş konfigürasyon dosyaları listelenir. Kullanıcı istediği konfigürasyonu bu bölümden seçerek tanıma ve eğitim işlemlerini bu konfigürasyona göre yapabilir.

Ana GUI'deki diğer bir bölüm ise "EĞİTİM" başlığı altında bulunan ve sistemin eğitiminin yapıldığı bölümdür. Bu bölüm iki adet listeden ve altı adet komut butonundan oluşur.



Şekil 3.1. Ana GUI

- **Sözlük Listesi**
  Bu listede konuşma tanıma sisteminin sözlüğü listelenir. Sözlülüğe kelime ekleme ve çıkarma "Kelime Ekle" ve "Kelime Sil" butonları ile yapılabilir. Ayrıca istenilen kelime fare ile seçilerek bu kelimeye ait eğitim dosyaları eğitim listesinde listelenebilir.

- **Eğitim Listesi**
  Sözlük listesinden seçilen kelimeye ait eğitim dosyalarının listelendiği listedir. Kullanıcı bu listeden istediği kelimeyi seçerek ve bu listenin aşağısında bulunana butonları kullanarak dinleme, ekleme, silme işlemlerini gerçekleştirebilir. Bu işlemler "Eğitim Datası Dinle", "Eğitim Datası Ekle" ve "Eğitim Datası Sil" butonları kullanılarak yapılır.

- **Kelime Ekle Butonu**
  Sözlük listesine bir kelime ekler.

- **Kelime Sil Butonu**
  Sözlük listesinden seçilen kelimeyi siler.

- **SMM Hesapla Butonu**
  Konfigürasyon seçim bölümünde seçilen konfigürasyona göre sözlükteki her bir kelime için bir SMM hesaplar. Hesaplanan SMM'ler seçilen konfigürasyonun Parametre klasörüne kaydedilir ve daha sonra tanıma işlemlerinde kullanılır.

- **Eğitim Datası Dinle Butonu**
  Eğitim listesinde seçilen eğitim datasını standart ses çıktı aygıtına gönderir. Bununla birlikte seçilen eğitim datasının zaman eğrisi "KONUŞMA SİNYALİ" başlığı altında bulunan grafiğe çizilir.

- **Eğitim Datası Ekle Butonu**
  Sözlük listesinden seçilen kelime için bir eğitim datası ekler. Eğitim datası direkt olarak standart ses giriş aygıtından (mikrofon) eklenir. Bu tuşa basıldığında öncelikle kaydedilecek dosyanın ismi sorulur ve ardından **Şekil 3.2.** 'de verilen ses kayıt ekranı ortaya çıkar. Konuşmacı bu anda, bu kelimeye ait konuşma sinyalini seçilen konfigürasyondaki "Kayıt Süresi" parametresi süresi içinde üretmesi gerekir. Aksi takdirde kayıt yeniden tekrarlanmalıdır.

- **Eğitim Datası Sil Butonu**
  Eğitim listesinden seçilen eğitim datasını siler.

Şekil 3.2. Ses giriş ekranı

Ana GUI'de bulunan diğer bir bölüm ise "KONUŞMA TANIMA" başlığı altında bulunan bölümdür. Bu bölüm sistemde tanıma işlemlerinin yapıldığı bölümdür. Tanıma bölümüne konuşma sinyalinin girişi iki şekilde yapılabilir. Bunlardan birincisi doğrudan mikrofon kullanılarak, ikincisi ise daha önceden wav formatında kaydedilmiş dosyalarla yapılabilir. Bu bölüm 10 adet tuş, iki adet liste ve bir adet metin kutusundan oluşur.

- **Dosyalar Listesi**
  Bu listede daha önceden "Listeye Dosya Ekle" ve "Listeye Klasör Ekle" butonlarıyla eklenmiş test dosyaları listelenir. Bu liste "Listeyi Kaydet" butonu kullanılarak kaydedilebilir. Bununla birlikte "Listeden Dosya Çıkar" ve "Listeyi Boşalt" butonları kullanılarak listede istenildiği gibi değişiklikler yapılabilir. Bu listeden seçilen dosya için "Test Datasını Dinle" ve "Test Datasını Sil" butonları kullanılarak dinleme ve silme işlemleri gerçekleştirilebilir. "Test Datası Ekle" butonu ile listeye yeni bir test datası kaydedilip eklenebilir.

- **Sonuç Listesi**
  Dosyalar listesindeki test datalarına karşılık gelen tanıma sonuçları "Tanı" tuşuna basıldığında bu listede listelenir.

- **Mikrofon Butonu**
  Bu tuş tanıma işlemini doğrudan mikrofondan yapmak için kullanılır. Bu tuşa basıldığında **Şekil 3.2.** 'de verilen ses kayıt ekranı ortaya çıkar. Konuşmacı bu anda, konuşma sinyalini seçilen konfigürasyondaki "Kayıt Süresi" parametresi süresi içinde üretmesi gerekir. Aksi takdirde kayıt yeniden tekrarlanmalıdır. Konuşma sinyali alındıktan sonra bu sinyalle eşlenen kelime bu tuşun yanındaki metin kutusuna yazılır.

- **Test Datasını Dinle Butonu**
  Dosyalar listesinde seçilen test datasını standart ses çıktı aygıtına gönderir. Bununla birlikte seçilen test datasının zaman eğrisi "KONUŞMA SİNYALİ" başlığı altında bulunan grafiğe çizilir.

- **Test Datası Ekle Butonu**
  Dosya listesine bir test datası ekler. Test datası direkt olarak standart ses giriş aygıtından (mikrofon) eklenir. Bu tuşa basıldığında öncelikle kaydedilecek dosyanın ismi sorulur ve ardından **Şekil 3.2.** 'de verilen ses kayıt ekranı ortaya çıkar. Konuşmacı bu anda, konuşma sinyalini seçilen konfigürasyondaki "Kayıt Süresi" parametresi süresi içinde üretmesi gerekir. Aksi takdirde kayıt yeniden tekrarlanmalıdır. Test datası "Listeye Klasör Ekle" butonu ile açılan son klasöre kaydedilir.

- **Test Datasını Sil Butonu**
  Dosya listesinden seçilen test datasını siler.

- **Listeye Dosya Ekle Butonu**
  Dosya listesine varolan bir test dosyasını ekler. Bu butona basıldığında kullanıcı dosya seçme penceresinden istediği dosyayı seçebilir.

- **Listeye Klasör Ekle Butonu**
  Dosya listesine istenilen bir klasördeki tüm test dosyalarını ekler. Bu tuşa basıldığında ekrana gelen klasör seçme penceresinden kullanıcı istediği klasörü seçebilir.

- **Tanı Butonu**
  Dosya listesindeki test dosyaları için tanıma işlemini gerçekleştirir. Sonuçları sonuç listesinde görüntüler.

- **Listeyi Kaydet Butonu**
  Listeyi kaydeder.

- **Listeden Dosya Çıkar Butonu**
  Listeden seçilen dosyayı çıkarır.

- **Listeyi Boşalt Butonu**
  Listedeki tüm dosyaları çıkararak listeyi boşaltır.

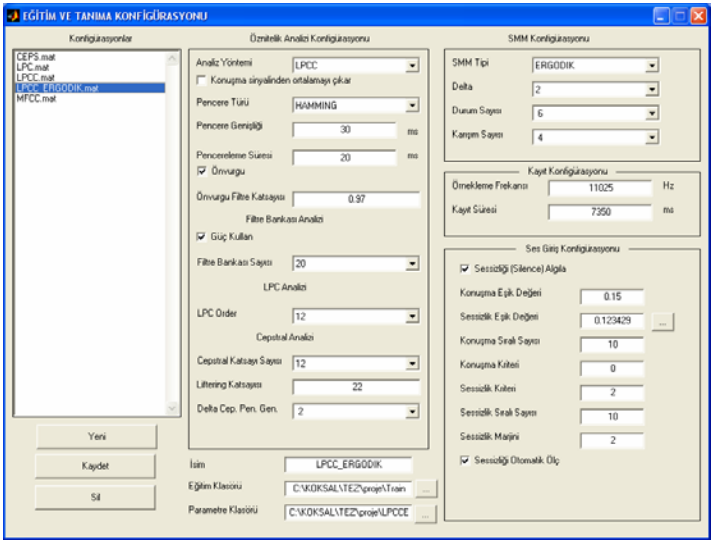
Ana GUI'de bahsedeceğimiz son bölüm ise "KONUŞMA SİNYALİ" başlığı altındaki konuşma sinyali izleme bölümüdür. Bu bölüm sadece bir grafikten oluşur. Bu grafikte eğitim ve test datalarının dinleme ve ekleme işlemleri sırasında konuşma sinyalinin zaman eğrisi görüntülenir. Bununla birlikte mikrofondan direkt olarak tanıma işlemi sırasında alınan test datasının zaman eğrisi de bu grafikte görüntülenir.

3.1.2. Konfigürasyon GUI'si

Konuşma tanıma sistemine ait bir takım konfigürasyonların yapıldığı GUI'dir. Yapılabilecek konfigürasyonları temel olarak 4 ana gruba ayrılabilir.

- Öznitelik analizi konfigürasyonu
- SMM konfigürasyonu
- Kayıt konfigürasyonu
- Ses giriş konfigürasyonu

GUI'ye ait ekran görünümü **Şekil 3.3.**'de verilmiştir. Ekranın solunda bulunan "Konfigürasyonlar" başlıklı listede daha önce yapılmış konfigürasyonlar listelenir. Bu listeden istenilen konfigürasyon seçilerek değişiklik yapılabilir. Bu listenin alt kısmında üç adet buton bulunmaktadır. "Yeni" butonu konfigürasyonu varsayılan değerlere set ederek ilgili yerlere yükler. Kullanıcı istediği değişiklikleri yaptıktan sonra "Kaydet" tuşunu kullanarak konfigürasyonu kaydedebilir. Konfigürasyonlar listesinden seçilen konfigürasyon "Sil" butonu kullanılarak silinebilir.



Şekil 3.3. Konfigürasyon GUI'si

Konfigürasyon GUI'si ile yapılabilecek konfigürasyonlar bölüm bölüm aşağıda anlatılmıştır.

1) Öznitelik analizi konfigürasyonu

- Analiz Yöntemi: Öznitelik analiz yöntemini belirler. Dört adet opsiyonu vardır.
    - LPC : Lineer öngörümsel kodlama
    - LPCC : LPC'den türetilen cepstrum
    - CEPS : Cepstral katsayılar
    - MFCC : Mel Frequency Cepstral Coefficients

- Konuşma sinyalinden ortalamayı çıkar: Sinyalden ortalamanın çıkarılıp çıkarılmayacağını belirler.
- Pencere Türü: Pencere türünü belirler. İki adet opsiyonu vardır.
    - RECTANGULAR Diktörtgen Pencere
    - HAMMING Hamming Pencere
- Pencere Genişliği: Pencere genişliğini ms cinsinden belirleyen parametredir.
- Pencereleme Süresi: Pencereleme süresini ms cinsinden belirleyen parametredir.
- Önvurgu: Önvurgu yapılıp yapılmayacağını belirler.
- Önvurgu Filtre Katsayısı: Önvurgu yapılması halinde önvurgu filtresinin katsayısının belirler.
- Güç Kullan: Filtre bankası analizinde FFT genlik değerlerinin karelerinin kullanılıp kullanılmayacağını belirler.
- Filtre Bankası Sayısı: Filtre bankası analizinde kullanılacak filtre sayısını belirler.
- LPC Order: LPC analizinin derecesini belirler.
- Cepstral Katsayı Sayısı: Kullanılacak cepstral katsayı sayısını belirler.
- Liftering Katsayısı: Lineer filtreleme sabitini belirler.
- Delta Cep. Pen. Gen: Delta parametre hesabında kullanılacak pencere genişliğini belirler.

2) SMM Konfigürasyonu
- SMM Tipi: SMM'lerin tiplerini belirler ve iki adet seçeneği vardır.
    - ERGODIK Ergodik tip SMM
    - BAKIS Bakis tip SMM
- Delta: Bakis tipi SMM' ler için geçerlidir ve Δ (atlama) değerini belirler.
- Durum Sayısı: Modeldeki durum sayısını belirler.
- Karışım Sayısı: Durum gözlem olasılık yoğunluklarında kullanılacak karışım sayısıdır.

3) Kayıt Konfigürasyonu
- Örnekleme Frekansı: Hz cinsinden örnekleme frekansını belirler.
- Kayıt Süresi: Eğitim ve tanıma esnasında kullanılacak kayıt süresinin ms cinsinden değeridir.

4) Ses Giriş Konfigürasyonu: Bu bölümdeki opsiyonlar bu çalışmada kullanılmamıştır.

3.2. ISRTK Çalışma İlkeleri ve Tez İçinde Kullanımı

ISRTK SMM tabanlı izole kelime tanıma sistemidir. Yazılım temel olarak üç ana bölümden oluşmaktadır.

- Eğitim
- Tanıma
- Konfigürasyon

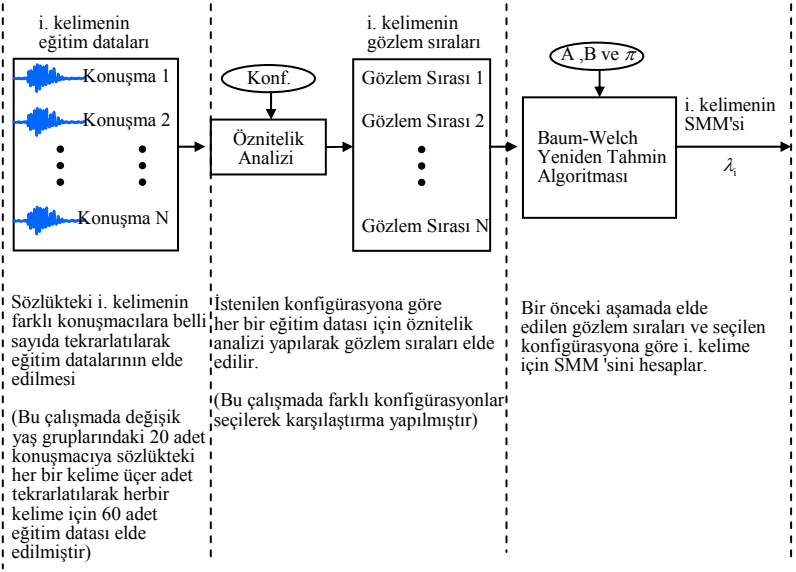
Bu bölümlere ait ayrıntılı açıklamalar ve tez içinde kullanım şekli ilerleyen bölümlerde verilmiştir.

3.2.1. ISRTK'nın Eğitilmesi

Sistemin eğitilmesi sözlükteki her bir kelime için eğitim verilerinin alınıp öznitelik analizlerinin yapılmasını ve analiz sonucunda elde edilen gözlem sıralarıyla SMM'lerin hesaplanmasını kapsar. Bu işleme ait blok diyagram **Şekil 3.4.**'de verilmiştir. Şekilde bahsedilen konfigürasyon öznitelik analizi ve SMM hesaplanması aşamasında kullanılacak bir takım opsiyonları içerir. Bu opsiyonlara ait açıklamalar 3.1.2. Konifigürasyon GUI'si bölümünde verilmiştir.

Bu çalışmada eğitim verisi olarak 20 adet, farklı yaş gruplarından alınan konuşmacılara ait veriler kullanılmıştır. Her bir konuşmacının sözlükteki her bir kelimeyi üçer adet tekrarlaması sonucu her bir kelime için 60 adet eğitim verisi elde edilmiştir. Elde edilen eğitim verilerine değişik konfigürasyonlarla öznitelik analizi yapılarak sözlükteki her bir kelime için değişik SMM modelleri hesaplanmıştır. Sekiz adet farklı konfigürasyon kullanılmıştır. Bu konfigürasyonlara ait bilgiler 3.2.3. ISRTK 'nın Konfigürasyonu bölümünde anlatılmıştır. Hesaplama öznitelik analizi sonucu elde edilen 60 adet gözlem sırası ve A, B ve π'nin başlangıç değerlerleriyle Baum-Welch yeniden tahmin algoritması kullanılarak yapılmıştır. A ve π için başlangıç değerleri modeldeki kısıtlamalar göz önüne alınarak düzgün dağılımlı olarak seçilmiştir. B'nin başlangıç değerleri için ise K-Means Clustering algotitması kullanılmıştır (Kondoz 1990). B'nin başlangıç değerlerinin hesaplanmasında model tipine göre

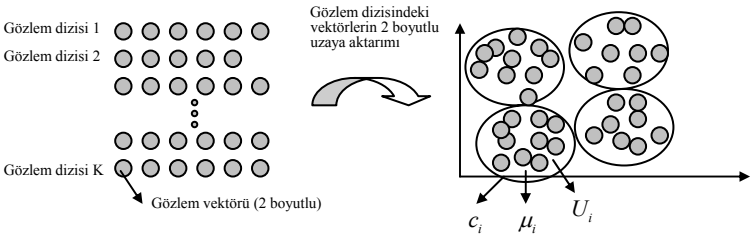
farklı yöntemler izlenmiştir. Ergodik modellerde K-Means Clustering algoritması kullanılarak



Şekil 3.4. ISRTK'nın eğitimi

tüm gözlem vektörleri N x M (N= Durum Sayısı, M=Karışım Sayısı) kadar cluster'a bölünmüştür. Bu cluster'ların merkez noktaları μ (centroid) karışımların ortalama değerleri olarak alınmıştır. Karışımların kovaryans matrisleri U , her bir cluster için, merkez noktalarına göre hesaplanarak bulunmuştur. Son olarak her bir karışımın ağırlığı c her bir cluster'daki vektör sayısının toplam vektör sayısına bölünmesi ile bulunmuştur.

Bu işlemler sonucunda elde edilen MxN adet karışım her bir duruma ardışık olarak eşit bir biçimde dağıtılarak, her bir durum için olasılık yoğunluk fonksiyonun başlangıç değerleri set edilmiştir. Bu işleme ait grafiksel anlatım 2 boyutlu gözlem vektörleri için **Şekil 3.5.**'de verilmiştir. Şekilden de görüldüğü gibi K adet gözlem dizisinde bulunan tüm vektörler 2 boyutlu uzayda belli sayıda cluster'a ayrılarak bu cluster'lardan karışımlara ait parametreler tahmin edilmiştir.



Şekil 3.5. Karışımların hesaplanması

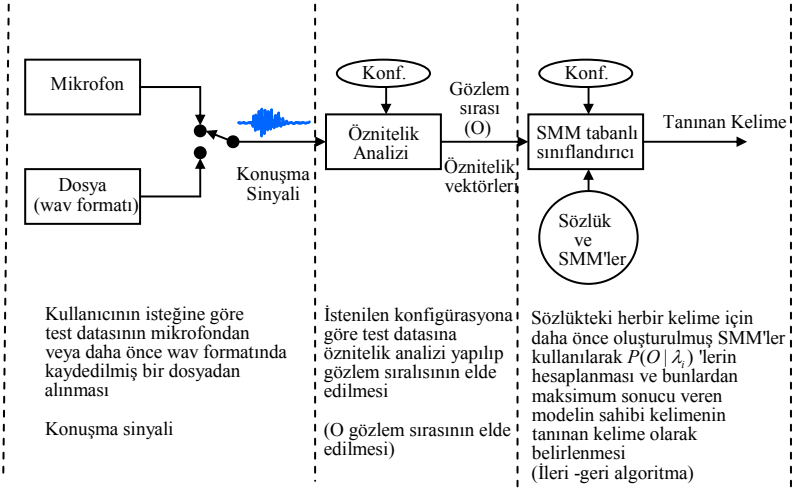
Bakis tipi modellerde modelin yapısı göz önüne alınarak farklı bir yöntem izlenmiştir. Öncelikle gözlem sıralarındaki vektörler düzgün olarak N eşit parçaya bölünmüştür. Gözlem sıralarını eşit bölme işlemi şu şekilde gerçekleştirilmiştir. Örnek olarak bir gözlem dizisinde 16 adet vektör olduğunu ve modelde 4 adet durum olduğunu düşünülürse, gözlem dizisindeki ilk 4 vektör ilk duruma, 5-8 arası vektörler ikinci duruma, 9-12 arası 3. duruma ve son olarak 13-16 arası vektörler 4. duruma atanmıştır. Bölümleme işleminin bu şekilde yapılmasının sebebi Bakis tipi modellerin her zaman ilk durumdan başlayarak, durum indisi aynı kalacak veya belli bir miktar (Δ) artacak şekilde ilerleyerek son durumda son bulmasıdır. Bu N adet bölüm K-Means Clustering algoritması kullanılarak M adet cluster'a ayrılmıştır. Bu cluster 'lardan daha önce ergodik modellerde uygulanan yöntemle her bir karışım için μ , U , c tahmin edilmiştir.

3.2.2. ISRTK ile konuşma tanıma

ISRTK' nın konuşma tanıma ilkesi

Şekil 2.11. verilen yalıtık kelime tanıma sistemi ile aynıdır. Mikrofon veya bir dosyadan alınan konuşma sinyalinin, istenilen konfigürasyona göre öznitelik analizi yapılır. Bu işlem sonunda elde edilen gözlem dizisi O kullanılarak sözlükteki her bir model için olasılık ileri-geri algoritma ile hesaplanır. Hesaplama sonucunda maksimum sonucu veren modelin sahibi olan kelime tanınan kelimedir. Bu işleme ait blok diyagram **Şekil 3.6.**'de verilmiştir.

Bu çalışmada test datası olarak, eğitim aşamasına katılmamış 20 adet konuşmacının sözlükteki bir veya birkaç kelimeyi tekrarlaması sonucu elde edilen 60 adet konuşma kullanılmıştır. Bu konuşmalar ISRTK ile 8 adet farklı kongifürasyon kullanılarak tanınmış ve sonuçlar karşılaştırılmıştır.



Şekil 3.6. ISRTK ile konuşma tanıma

3.2.3. ISRTK ile konfigürasyon

ISRTK ile konfigürasyon bölüm 3.1.2.'de anlatılan konfigürasyon GUI'si ile yapılır. Bu çalışmada konuşmanın değişik akustik parametrelerinin ve SMM tiplerinin konuşmacı bağımsız tanıma sistemlerindeki başarıları karşılaştırılacak şekilde 8 adet farklı konfigürasyon kullanılmıştır. Bu konfigürasyonlar CEPS_B, CEPS_E, LPC_B, LPC_E, LPCC_B, LPCC_E, MFCC_B ve MFCC_E'dir. Konfigürasyonlardan, isimlerinin sonlarında B bulunanlarda Bakis tipi modeller, E bulunanlarda ise ergodik modeller kullanılmıştır. İsimlerdeki alt çizgiye kadar olan kısımlar ise kullanılan akustik parametrelerin cinsini göstermektedir Çizelge 3.1'de konfigürasyonlara ait özellikler verilmiştir. Çizelgede "-" ile gösterilen bölümler ilgili konfigürasyon için uygulanabilir olmadığını göstermektedir.

Çizelge 3.1'den görüldüğü gibi tüm konfigürasyonlarda konuşma sinyali 11025 Hz'de örneklenmiş ve konuşmanın geldiği kanaldan kaynaklanan DC ofseti gidermek için sinyalin ortalaması kendisinden çıkarılmıştır. Öznitelik analizi, konuşma sinyalinin 30 ms (330 örnek) uzunluğundaki Hamming pencere ile pencerelenmesi sonucu elde edilen pencerelerde yapılmıştır. Pencereleme süresi olarak 20 ms (220 örnek) seçilerek ardışık pencerelerin birbirleriyle 10 ms (110 örnek) örtüşmesi sağlanmıştır.Tüm pencereler transfer fonksiyonu (2.71)'de verilen önvurgu filtresiyle a=0.97 alınarak filtrelenmiştir.

CEPS_B ve CEPS_E konfigürasyonlarında pencereleme sonucu elde edilen pencerelerin FFT tabanlı gerçel cepstrum değerleri hesaplanarak bunlardan ilk bileşen (DC değer) hariç ilk 12 tanesi kullanılmıştır.

LPC_ B ve LPC_E'de 12. dereceden LPC katsayıları, LPCC_ B ve LPCC_E'de ise bu katsayılardan elde edilen cepstrum (LPC cepstrum) kullanılmıştır.

MFCC_ B ve MFCC_ B konfigürasyonlarında mel skalasında eşit aralıklarla dağılmış bir birleriyle %50 oranında kesişen 22 adet üçgen filtre ve FFT spektrumunun karesi kullanılarak MFCC katsayıları hesaplanmıştır.

Bu işlemler sonucunda her bir pencereye karşılık gelen 12 boyutlu bir akustik parametre vektörü hesaplanmıştır. Bu vektörlere 12 boyutlu delta parametre vektörleri eklenerek 24 boyutlu vektörler elde edilmiştir.

Kayıt süresi olarak 7350 ms seçilmiştir. Bunun sonucunda her bir eğitim datası için (her hangi bir kelimenin tekrarlatılması sonucu elde edilen konuşma sinyali) 367 adet pencere ve bunlara karşılık olarak 367 adet 24 boyutlu parametre vektörü elde edilmiştir. Konuşma sinyalinin başında ve sonunda bulunan sessizlik atılmamıştır. Her bir kelime için 60 adet eğitim datası kullanıldığından SMM'lerin hesaplanması sırasından 60 adet gözlem sırası kullanılmıştır. Her bir gözlem sırası ise 367 adet 24 boyutlu vektörden oluşmaktadır.

Tüm konfigürasyonlarda SMM'ler için durum ve karışım sayısı 8 olarak belirlenmiştir. CEPS_E, LPC_E, LPCC_E, MFCC_E konfigürasyonlarında ergodik modeller CEPS_B, LPC_B, LPCC_M, MFCC_B konfigürasyonlarında ise atlama miktarı Δ 2 olan bakis tipi modeller kullanılmıştır.

Çizelge 3.1. ISRTK'da kullanılan konfigürasyonlar

	Konfigürasyon İsimleri			
	CEPS_B	LPC_B	MFCC_B	LPCC_B
Analiz Yöntemi	CEPS	LPC	MFCC	LPCC
Sinyalden ortalamayı çıkar	Evet	Evet	Evet	Evet
Pencere Türü	Hamming	Hamming	Hamming	Hamming
Pencere Genişliği	30	30	30	30
Pencere Süresi	20	20	20	20
Önvurgu	Evet	Evet	Evet	Evet
Önvurgu Filtre Katsayısı	0.97	0.97	0.97	0.97
Güç Kullan	-	-	Evet	-
Fitre Bankası Sayısı	-	-	22	-
LPC Derecesi	-	12	-	12
Cepstral Katsayı Sayısı	12	-	12	12
Liftering Katsayısı	22	-	22	22
Delta Parametre Pencere Gen.	2	-	2	2
SMM Tipi	Bakis	Bakis	Bakis	Bakis
Delta	2	2	2	2
Durum Sayısı	8	8	8	8
Karışım Sayısı	8	8	8	8
Örnekleme Frekansı	11025	11025	11025	11025
Kayıt Süresi	7350	7350	7350	7350

Çizelge 3.1. ISRTK'da kullanılan konfigürasyonların (devam)

	Konfigürasyon İsimleri			
	CEPS_E	LPC_E	MFCC_E	LPCC_E
Analiz Yöntemi	CEPS	LPC	MFCC	LPCC
Sinyalden ortalamayı çıkar	Evet	Evet	Evet	Evet
Pencere Türü	Hamming	Hamming	Hamming	Hamming
Pencere Genişliği	30	30	30	30
Pencere Süresi	20	20	20	20
Önvurgu	Evet	Evet	Evet	Evet
Önvurgu Filtre Katsayısı	0.97	0.97	0.97	0.97
Güç Kullan	-	-	Evet	-
Fitre Bankası Sayısı	-	-	22	-
LPC Derecesi	-	12	-	12
Cepstral Katsayı Sayısı	12	-	12	12
Liftering Katsayısı	22	-	22	22
Delta Parametre Pencere Gen.	2	-	2	2
SMM Tipi	Ergodik	Ergodik	Ergodik	Ergodik
Delta	-	-	-	-
Durum Sayısı	8	8	8	8
Karışım Sayısı	8	8	8	8
Örnekleme Frekansı	11025	11025	11025	11025
Kayıt Süresi	7350	7350	7350	7350

4. ARAŞTIRMA BULGULARI

Araştırma bulguları, CEPS_E, CEPS_B, LPC_E, LPC_B, LPCC_B, LPCC_E, MFCC_E ve MFCC_B konfigürasyonları kullanılarak test datalarının ISRTK'ya tanıtılması sonucu elde edilmiştir. Konfigürasyonlardan isimlerinin sonunda B harfi olanlarda Bakis tipi SMM'ler, E olanlarda ise ergodik SMM'ler kullanılmıştır. İsimlerdeki alt çizgiye kadar olan kısımlar ise kullanılan akustik parametrelerin cinsini göstermektedir. Bu konfigürasyonlarla ilgili ayrıntılar Çizelge 3.1.'de verilmiştir.

20 adet, farklı yaş grubundaki konuşmacının sözlükteki her bir kelimeyi üçer adet tekrarlaması sonucu her bir kelime için 60 adet eğitim datası elde edilmiştir. Bu eğitim dataları ve değişik konfigürasyonlar kullanılması sonucu, her bir kelime için değişik özellikteki gözlem vektörleri ve SMM tipleri kullanılarak 8 adet farklı model oluşturulmuştur. Eğitim aşamasına katılmamış 20 adet konuşmacının sözlükteki bir veya bir kaç kelimeyi tekrarlaması sonucu elde edilen 60 adet test datası, 8 adet farklı konfigürasyon kullanılarak ISRTK'ya tanıtılmıştır. Bu işlem sonucunda her bir konfigürasyona karşılık gelen doğruluk oranları Çizelge 4.1.'de verilmiştir.

Çizelge 4.1. ISRTK ile farklı konfigürasyonların doğruluk oranları

Konfigürasyonlar	Doğruluk
CEPS_B	%91,49
CEPS_E	%89,72
LPC_B	%80,86
LPC_E	%79,17
LPCC_B	%93,25
LPCC_E	%89,72
MFCC_B	%95,00
MFCC_E	%92,89

Çizelge 4.1.' e bakıldığında göze çarpan ilk bulgunun konuşmacı bağımsız konuşma tanıma sistemlerinde aynı akustik parametre vektörleri kullanıldığında bakis tipi modellerin ergodik modellerden daha başarılı

olduğudur. Örnek olarak MFCC_B nin doğruluk oranı %95,00 iken MFCC_E'nin doğruluk oranı %92,89'dur.

Çizelge 4.1.'den çıkarabileceğimiz diğer bir bulgu ise LPC'ye cepstrum yaklaşımının doğrudan LPC parametrelerinden daha başarılı olduğudur.

Çizelge 4.1.'ya toplu olarak bakıldığında ise MFCC parametrelerinin en yüksek başarı oranını yakaladığı görülmektedir.

5. TARTIŞMA VE SONUÇ

Çizelge 4.1'de verilen araştırma bulguları incelendiğinde kelime tabanlı konuşmacı bağımsız konuşma tanıma sistemlerinde ergodik modellerin başarılarının Bakis tipi modellerin başarılarından daha düşük olduğu görülmektedir. Kelimelerin farklı ses gruplarının yan yana gelmesiyle oluştuğu ve bu farklı ses grupları modeldeki durumlarla ilişkilendirilirse, kelime tabanlı tanıma sistemlerinde Bakis tipi modellerin Ergodik modellerden daha uygun olduğu rahatça görülebilir. Ergodik modellerde durumlar arasında geçiş sınırlaması yoktur. Yani her bir durum başka bir duruma geçiş yapabilir. Bununla birlikte model her hangi bir durumdan başlayabilir. Bu tip bir model kelime tabanlı konuşma tanıma sistemleri için uygun değildir. Çünkü bir kelimenin farklı ses gruplarının yan yana gelmesiyle oluştuğunu ve her bir ses grubunun (triphone, hece, vb.) bir duruma karşılık geldiğini düşünülürse, o kelimeyi üretecek modelin her zaman aynı durumdan başlaması gerekir. Aksi takdirde gereksiz olasılıklar hesaba katılarak gereksiz bir modelleme yapılacaktır. Bakis tipi modellerde her zaman ilk durumdan başlanır ve son durumla bitirilir. Diğer geçişler ise durum indisi yerinde sayacak veya belli bir miktar (Δ) artacak şekildedir. Kelimelerin hep aynı ses grubuyla başladığı, kelimenin söylenmesi sırasında bir takım bölgelerin uzatıldığı veya yutulduğu düşünülürse, kelimelerin üretim şekillerinin Bakis tipi modellerin özellikleriyle uyuştuğu açıkça görülebilir.

Çizelge 4.1.'den çıkarılabilecek diğer bir sonuç ise LPCC parametrelerinin LPC parametrelerinden daha başarılı olduğudur. Konuşmacı bağımsız konuşma tanıma sistemlerinde LPCC, LPC'den daha başarılıdır. Bunun sebebi aynı konuşma sinyali için LP katsayılarının konuşmacıdan konuşmacıya konuşma tanıma sisteminin performansını düşürecek şekilde farklılıklar göstermesidir (Deller 1993). Cepstrum yöntemleriyle bu farklılıklar azaltılarak konuşmacı bağımsız konuşma tanıma sistemlerinin performansı artırılabilir.

Çizelge 4.1.'de göze çarpan diğer bir sonuç ise MFCC'nin ham cepstral katsayılara (CEPS) göre ve LPCC'ye göre daha başarılı olmasıdır. MFCC Bölüm 2.4.8.' de bahsedildiği gibi insan duyma sisteminin fiziksel özelliklerini göz önüne alarak lineer bir frekans skalası yerine logaritmik bir frekans skalasında (mel skalası) analiz yapar. Bunun sonucunda

konuşmanın duyum olarak anlamlı özelliklerini daha iyi yakalar. Bununla birlikte MFCC analizinde kullanılan filtre bankaları insan duyma sisteminin band geçiren yapısını simule ederek duyum olarak anlamlı bölgeleri vurgular.

KAYNAKLAR

Baum, L.E., Petrie, T., Soules, G. and Weiss, N. 1970. A maximization technique occuring in the statistical analysis of probalilistic function of Markov chains. Ann. Math. Stat.,41(1); 164-171.

Deller, J.R. , Proakis, J.G. and Hansen, J.H.L. 1993. Discrete Time Processing of Speech Signals, MacMillian Publishing Co., New York.

Dempster, A.P., Laird, N.M. and Rubin, D.B. 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc.,39(1);1-38.

Juang, B.H., Levinson, S.E. and Sondhi, M.M. 1986 Maximum likelihood estimation for multivariate observations of Markov Chains. IEEE Transactions on Information Theory, IT-32(2);307-309

Juang, B.H., Rabiner, L.R. and Wilpon, J.G. 1987. On the use of bandpass liftering in speech recognition. AT&T System Tecnical Journal, 64;391-408

Kondoz, A.M. 1990. Digital Speech, JOHN WILEY & SONS, New York

Levinson, S.E., Rabiner, L.R. and Sondhi, M.M. 1983. An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition-A unified view. Bell Syst. Tech. J., 62(4);1035-1074.

Liporace, L.A. 1982. Maximum likelihood estimation for multivariate observation of Markov sources. IEEE Trans. Informat. Theory, IT-28(5);729-734

Markel, J. and Gray, A.H. 1980. Linear Prediction of Speech, Springer-Verlag, NewYork.

Rabiner, L.R. and Juang, B.H. 1986. An Introduction to Hidden Markov Model. IEEE ASSP Magazine

Rabiner, R. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE, 77(2);257-286.

Stevens, S. S. and Volkman, J. 1940. The relation of pitch to frequency. American Journal of Psychology, 53;329

ÖZGEÇMİŞ

1981 yılında Ankara'da doğdu. İlk öğrenimini Kırıkkale'de, orta ve lise öğrenimini Ankara'da tamamladı. 1997 yılında girdiği Ankara Üniversitesi Elektronik Mühendisliği Bölümü'nden 2002 yılında Elektronik Mühendisi ünvanı ile mezun oldu. 2002 yılının Eylül ayında Ankara Üniversitesi Fen Bilimleri Enstitüsü'nde yüksek lisans öğrenimine başladı.

ELIMKO Ltd. Şti AR-GE bölümü bünyesinde 2002 yılından beri AR-GE mühendisi olarak görev yapmaktadır.