T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KONUŞMACI TANIMA YÖNTEMLERİNİN KARŞILAŞTIRMALI ANALİZİ

Cemal HANİLÇİ

YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

BURSA-2007

T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KONUŞMACI TANIMA YÖNTEMLERİNİN KARŞILAŞTIRMALI ANALİZİ

Cemal HANİLÇİ

Yrd. Doç. Dr. Figen ERTAŞ
(Danışman)

YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

BURSA-2007

T.C.
ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KONUŞMACI TANIMA YÖNTEMLERİNİN KARŞILAŞTIRMALI ANALİZİ

Cemal HANİLÇİ

YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Bu Tez ..../...../200... tarihinde aşağıdaki jüri tarafından oybirliği/oy çokluğu ile kabul edilmiştir.

Yrd Doç Dr. Figen ERTAŞ .....
.....
Danışman

....

ÖZET

Son yıllarda kişinin sesinden kim olduğunun belirlenebildiği uygulamalar yoğun ilgi odağı olmuştur. Kimlik belirleme ya da doğrulama, güvenlik ve erişim kontrolü gibi uygulamalarda en önemli işlevlerden biridir. Gizli kaynaklara (bilgi, bilgisayar, özel saha) kontrollü erişimi sağlamanın yöntemlerinden olan anahtar, şifre, kimlik kartı kolaylıkla kaybolabilir, çalınabilir veya taklit edilebilirken, başkalarınca taklit edilemeyen kişiye has eşsiz özellikler yani biyometriklerin kullanımı rağbet görmeye başlamıştır. Biyometrikler parmak izi, el geometrisi ve retina örüntüsü gibi fiziksel özellikleri ya da el yazısı ve sesizi (voiceprint) gibi kişisel özellikleri kullanır. Her ne kadar parmak izi ve retina örüntüsü kişinin kimliğini belirlemede daha güvenilir olsa da telefon hattı üzerinden bilgi toplama gibi pratik uygulanabilirliğinden dolayı ses örneğinden kişinin kimliğinin belirlendiği uygulamalar son yıllarda ön plana çıkmıştır.

Bu tezde metinden bağımsız konuşmacı belirleme konusunda sıkça kullanılan yöntemlerden *Saklı Markov Modelleri* ve *Vektör Nicemle* algoritmaları incelenmiştir. Birinci bölümde, konuşmacı tanıma uygulamalarında bugüne kadar kullanılmış kişinin sesini temsil eden özellikler ve bu özelliklerin modellenmesinde kullanılan yöntemlerden bahsedilmektedir. İkinci bölümde bu tezde yapılan deneyler sırasında kişinin sesini temsil eden parametrelerden *mel ölçekli kepstrum katsayıları* (mfcc) ve bu katsayıların çıkarımı sırasında izlenen adımlar detaylı bir şekilde anlatılmaktadır. Konuşmacı tanıma sisteminde özellik çıkarımından sonraki adım olan modelleme tekniklerinden Saklı Markov Modelleri (SMM) ve Vektör Nicemleme (VN) algoritmaları da detaylı bir şekilde ikinci bölümde anlatılmaktadır. Son bölümde ise mfcc özellikleri ile SMM ve/veya VN kullanılarak elde edilen deneysel sonuçlar verilmektedir.

Bu tezin iki temel amacı vardır. Bunlardan ilki, konuşmacı tanıma sistemlerinin yapı taşlarından olan özellik vektörleri boyutunun optimum değerinin belirlenmesidir. İkincisi ise konuşmacı tanıma uygulamalarında en çok kullanılan iki yöntem olan SMM ve VN algoritmalarının karşılaştırmalı analizlerinin yapılmasıdır. Ayrıca SMM yöntemi ile en fazla konuşmacı sayısının kullanıldığı metinden bağımsız konuşmacı tanıma uygulaması olması nedeniyle de bu tez ayrı bir önem taşımaktadır.

Deneyler sırasında 630 kişilik TIMIT veritabanı kullanılmıştır. VN ile yapılan deneylerde 21 sn eğitim (7 cümle) ve 9 sn test verisi (3 cümle) için 32 kod kitabı ile 630 kişi için %100 tanıma oranı elde edilmiştir. Yine aynı şartlarda 32 karışım ve 1 durumlu SMM kullanılarak 630 kişi için %100 tanıma oranı elde edilmiştir. Her iki test sonucu da deneysel sonuçlar ve tartışma bölümünde de belirtileceği gibi literatürde yapılan çalışmalardan yüksektir.

Anahtar kelimeler: metinden bağımsız konuşmacı tanıma, mel ölçekli kepstrum katsayıları, saklı Markov Modelleri, Vektör Nicemleme

ABSTRACT

Nowadays identifying people from their voices has become one of the most popular applications. Personal identification is an essential requirement for controlling access to protected resources. Personal identity can be claimed by a key, a password or a badge, all of which can be easily stolen, lost or faked. However, there are some unique (biometrics) features of individuals which cannot be imitated by someone else. Biometrics uses physical characteristics such as fingerprints, hand geometry and retinal patterns, and personal traits such as handwriting and voiceprint. Although fingerprints or retinal pattern are usually more reliable ways of verifying that a person is who he claims to be, identity verification based on person's voice has special advantages for practical deployment such as the convenience of easy data collection over the telephone.

In this thesis, two most common techniques, Hidden Markov Models (HMM) and Vector quantization (VQ), which are used in text-independent speaker identification, are analyzed from the view point of performance analysis. First chapter of this thesis describes the parameters which represent speakers' and the modeling techniques that are used for modeling of these parameters. In the second chapter we describe the Mel Frequency Cepstral Coefficients (mfcc), that is used during experiments as the parameters that represent speaker, and the steps of extraction these features from a given voice sample. It is also described in the second chapter, modeling of these features, HMM and VQ, which is the second step of a speaker identification system. Finally it is given that the text-independent speaker identification results using both HMM and VQ in the last chapter of this thesis.

This thesis has two main purposes. First, making a decision about the optimum number of mfcc which is going to be used in the system and the second is, comparing two popular approaches to perform speaker identification, HMM and VQ, according to identification rates. The other importance of this thesis is, it is the largest population text-independent speaker identification study using HMM.

The TIMIT database which contains 630 speakers was used during experiments. 100% speaker identification rate was achieved with the speaker identification system that uses VQ with 32 codebooks for 630 speakers when the 7 sentences (approximately 21 seconds) of each speaker were used to create codebook and the remaining 3 sentences (approximately 9 seconds) for testing. Under the same conditions but using a 1 state HMM with 32 mixtures for modeling the speakers instead of VQ, % 100 speaker identification rate was achieved. It will be shown that these are the highest identification rates of the earlier studies in the last chapter.

Key Words: text-independent speaker identification, mel frequency cepstrum coefficients, Hidden Markov Models, Vector Quantization.

İÇİNDEKİLER

KISALTMALAR DİZİNİ

DTW – Dinamik zaman eğirme
GKM - Gauss karışım modeli
SMM – Saklı Markov Modelleri
VN – Vektör Nicemleme
MFCC – Mel ölçekli kepstrum katsayıları
LPC – doğrusal öngörülü kodlama katsayıları
NN – Yapay sinir ağları

ÇİZELGELER DİZİNİ

Sayfa

ŞEKİLLER DİZİNİ

Sayfa

SİMGELER DİZİNİ

$x = (x_1, x_2, ..., x_n)$ - özellik vektörü

w_i - i. sınıf

μ - Ortalama

$\sum$ - Ortak değişinti matrisi

$P(w_i)$ - w_i sınıfının önsel olasılığı

$P(w_i|x)$ - w_i sınıfının şartlı olasılık yoğunluk işlevi

$P(x|w_i)$ - w_i sınıfında x'in olasılık yoğunluk işlevi

λ - Saklı Markov Model

C_j - j. kod vektörü

GİRİŞ

Konuşma işareti pek çok bilgi taşımaktadır. İçerdiği bilgiler arasında en önemlileri kelime veya konuşulan mesajın içeriği ve konuşmacının kimliğidir. Konuşma tanıma söylenen kelime veya cümlenin içeriği ile ilgilenirken konuşmacı tanıma kelime veya cümleyi söyleyenin kimliği ile ilgilenmektedir. Konuşmacı tanıma sistemi özellikle son yıllarda oldukça ilgi çeken konular arasında yerini almıştır.

Konuşmacı tanıma, kişiye özgü bilgilerin bulunduğu konuşma işaretleri aracılığı ile otomatik olarak kimin konuştuğunun belirlenmesidir (Doddington 1985, O'Shaugnessy 1986, Furui 1997, Campbell 1994, Gısh ve Schmıdt 1994). Konuşmacı tanıma günümüzde sesli arama, telefon bankacılığı, telefonla alışveriş, veritabanı erişim servisleri, güvenlik kontrolü, bilgisayarların sesle kontrolü ve adli uygulamalar gibi alanlarda kullanılmaktadır.

Konuşmacı tanıma işlemi, konuşmacı doğrulama ve konuşmacı belirleme olmak üzere iki gruba ayrılır. Bu iki yöntemin ortak noktası, her iki yöntemin de referans konuşmacılara ait bir veritabanı, benzer analiz ve karar tekniklerini kullanmasıdır. Konuşmacı belirleme, sistemde kayıtlı konuşmacılardan hangisinin konuştuğunun belirlenmesi, konuşmacı doğrulama ise kim olduğunu iddia eden kişinin kabul veya reddedilmesidir. İki yöntem arasındaki temel fark, karar aşamasında ortaya çıkmaktadır. Konuşmacı belirlemede sistemin ürettiği karar sayısı sistemde kayıtlı bulunan konuşmacı sayısına eşittir. Buna karşılık konuşmacı doğrulamada kişi sayısından bağımsız olarak karar açısından sadece iki seçenek vardır: Kabul veya Ret. Bundan dolayı konuşmacı belirlemede kişi sayısı arttıkça tanıma oranı azalırken, konuşmacı doğrulamada tanıma oranı kişi sayısından bağımsız olarak bir sabite yakınsayacaktır.

Konuşmacı tanıma sistemleri "*Açık Küme*" ve "*Kapalı Küme*" olmak üzere ikiye ayrılır. İkisi arasındaki tek fark, sisteme kayıtlı olmayan kişilerin de sisteme giriş yapabilip yapamadığının önceden bilinmesidir. Kapalı Kümede sistem sadece kayıtlı

kişiler ile çalışırken, Açık Küme ise sisteme bilinmeyen hatta yanıltmak isteyen kişilerin de erişebileceğinin kabul edildiği sistemdir. Açık küme konuşmacı tanıma sisteminde, bilinmeyen konuşmacıya ait girilen test cümlesi, sistemde kayıtlı bulunan referans modellerden hiç biri ile uyuşmuyor olabilir. Bu tür sistemlerin karar aşamasında ek bir sonuç daha ortaya çıkmaktadır. Bu sonuç "*Bilinmeyen konuşmacı hiçbir modelle uyuşmadı*" şeklindedir. Doğrulama veya belirleme işlemlerinin her ikisinde de açık küme tanıma işlemi yapılacaksa karar aşamasında ek olarak bir eşik değer testi yapılmalıdır. Bu eşik değer testi sayesinde bilinmeyen konuşmacıya ait işaretin kabul edilebilir olup olmadığına karar verilir (Furui 1997).

Genel olarak, bir konuşmacı tanıma sistemi *Eğitim* ve *Test* olmak üzere iki aşamadan oluşmaktadır. Eğitim aşamasında, bilinen konuşmacılar eğitim cümleleri ile sisteme tanıtılırlar. Bu işleme *modelleme* denilmektedir. Test aşamasında ise bilinmeyen konuşmacıya ait test cümlesi, eğitim aşamasında oluşturulan her bir modelle karşılaştırılır ve benzerlik ölçütü kullanılarak test cümlesinin hangi modele ait olduğuna karar verilir. Bu işleme ise *sınıflandırma* adı verilmektedir.

Konuşmacı tanıma sistemleri "*metinden bağımsız*" veya "metine bağımlı" olabilir. Metine bağımlı sistemlerde, eğitim ve test aşamalarında aynı cümleler kullanılır. Bu tür sistemlerde genellikle şablon eşleştirmeye dayalı sınıflandırma yöntemleri kullanılmaktadır (Furui 1981, Naik ve ark. 1989, Rosenberg ve ark. 1991, Zheng ve Yuan 1988). Metine bağımlı tanıma sistemleri genelde yüksek tanıma oranları vermektedir. Ancak adli uygulamalar gibi güvenlik açısından önem taşıyan durumlarda önceden tanımlanmış sabit cümleler kullanılmaz. Üstelik insan kulağı konuşmacıları konuşulan sözün içeriğinden bağımsız olarak tanıyabilmektedir. Bu nedenle son yıllarda metinden bağımsız konuşmacı tanıma uygulamaları daha da ilgi çekmeye başlamıştır.

Metine bağlı ve metinden bağımsız yöntemlerin her ikisinin de çok önemli bir zayıf noktası vardır. Bu zayıflık, bu tarz sistemlerin sisteme kayıtlı birinin konuşmalarının kaydedilmesi ve bu kayıtlar ile kolayca aldatılabileceğidir. Bu problemi çözmek için, bazı yöntemler küçük kelime kümeleri, sayılar, anahtar kelimeler ve hatta bazı sistemler o an istatistiksel seçilen kelimeler kullanmaktadır (Higgins ve ark. 1991, Rosenberg ve ark. 1991).

Konuşmacı tanıma sisteminde, bilinmeyen konuşmacıya ait ses işareti, bilinen konuşmacılara ait modellerle karşılaştırılır. Bilinmeyen konuşmacı, giriş işareti en iyi eşleşmeyi hangi modelle sağlıyorsa o modelin ait olduğu kişiye atanır. Konuşmacı doğrulamada ise bilinmeyen konuşmacı tarafından bir kimlik iddiası ortaya atılır ve sistem iddia sahibi konuşmacıyı iddia ettiği kişinin modeli karşılaştırır. Eğer yeterli eşleşme sağlanırsa (eşik değerin üstünde) kimlik iddiası kabul edilir. Yüksek eşik değeri, sistemde olmayıp da sisteme girmeye çalışan taklitçilerin kabul edilmesini zorlaştırır ancak bu durumda da sistemde kayıtlı kullanıcıların kabul edilmeme riski ortaya çıkmaktadır. Bu nedenle optimum bir eşik değerinin belirlenmesi gerekmektedir.

Bu tezde TIMIT veritabanı kullanılarak kapalı küme, metinden bağımsız konuşmacı tanıma sistemi geliştirilmiştir. Bu sistem gerçekleştirilirken Saklı Markov Modelleri (Rabiner 1989) ve Vektör Nicemleme (Linde ve ark. 1980) olmak üzere iki ayrı modelleme tekniği kullanılmış ve bunların karşılaştırmalı analizleri verilmiştir.

1. KURAMSAL TEMELLER

Konuşmacı tanıma problemi, Özellik Çıkarımı ve Sınıflandırma olmak üzere iki kısımdan oluşur (Atal 1976). Özellik Çıkarımı kısmında konuşma işaretinden kişiyi temsil eden parametreler elde edilir. Sınıflandırma aşamasında ise Özellik Çıkarımı kısmında elde edilen parametreler kullanılarak bilinmeyen test verisinin kime ait olduğunun bulunması için değişik sınıflandırma algoritmaları kullanılır.

1.1. Özellik Seçimi

Kişiyi temsil eden özellik vektörleri zamanla değişen ve zamanla değişmeyen olarak iki gruba ayrılır (Atal 1976). Zamanla değişmeyen özellikler, zaman geçtikçe değişiklik gösteren özelliklerin ortalamalarının alınması ile ya da ses yolunun değişmez anatomik yapısının ölçülmesi ile elde edilir. Bu özelliklerin en önemli avantajı konuşmanın içeriğinden bağımsız oluşu ve bundan dolayı da metinden bağımsız konuşmacı tanıma uygulamaları için uygun olmasıdır. Zamanla değişen özellikler ise, zamanın sürekli bir fonksiyonu olan parametrelere karşı seçici olarak tanımlanmış parametrelerin ayırt edilmesi sonucu elde edilir. Zamanın sürekli bir fonksiyonu olarak tanımlanan özelliklerin elde edilmesi kolay olmasına karşın çok sayıda gereksiz bilgi içermektedir.

Konuşmacı tanıma konusunda başlangıçta yapılan çalışmalarda genellikle zaman, frekans, enerji gibi özellikler kullanılmıştır. Günümüze kadar frekans ve zaman ortamında analize dayanan birçok ek özellikler üzerinde çalışılmıştır. Bunlardan sıklıkla kullanılanları aşağıda detaylı bir şekilde verilmiştir.

1.1.1. Kısa zaman Spektrumu

Kısa zaman spektrumu ses işaretinin üç boyutlu olarak temsil edilmesine dayanır. Koordinatlar, zaman, frekans ve enerjidir. Kısa zaman spektrumu ses işaretinin özelliklerinin tamamını tanımlamaktadır.

1.1.2. Pitch (Perde) Frekansı:

Pitch frekansı ses tellerinin titreşimlerinin temel frekansıdır. Pitch frekansı hem zaman ortamında direkt olarak ses sinyalinden periyotların ölçülmesi ile hem de frekans ortamında spektral tepe değerlerinin hesaplanması ile elde edilebilir. Pitch frekansı konuşmacı tanıma uygulamalarında önemli bir etkiye sahiptir. Pitch frekansı tek başına kullanıldığında ayırt edici olmasa bile diğer ses özellikleri ile birlikte konuşmacı tanıma uygulamalarında sıklıkla, ama ses tanımada nadiren kullanılmaktadır (Rosenberg ve Sambur 1975, Rosenberg 1976, Markel ve ark. 1977, Jankowski ve diğ. 1994).

1.1.3. Formant Frekansları

Formant frekansları ses yolunun rezonans frekansları olarak tanımlanmaktadır. Formant frekansları konuşmacıya bağımlıdır. Formant frekanslarının ölçülmesine ilişkin bir çok yöntem literatürde tanımlanmıştır (Schaefer ve Rabiner 1970). Ancak bayan ve erkek konuşmacılar için formant frekansları ile ilgili güvenilir bir ölçüm yapılabilmesi hala temel problemlerden biridir.

1.1.4. Öngörücü katsayıları

Doğrusal öngörü analizi, zaman ortamında ses işaretinin spektral özelliklerini karakterize etmede önemli bir yere sahiptir. Bu yöntemde, ses işaretinin her bir örneği, geçmiş p adet örneğin doğrusal ağırlıklandırılmış toplamı şeklinde öngörülür. Ortalama karasel öngörü hatasını minumum yapan ağırlık katsayıları öngörücü katsayıları olarak tanımlanır. Genellikle 5 kHz band sınırlı bir ses işareti için 12 adet katsayı kullanmak yeterlidir. Öngörücü katsayıları zamanın bir fonksiyonu olarak değişir ve genellikle 20 ms'lik periyotlar halinde hesaplanması uygundur. Doğrusal öngörücü tabanlı katsayılar (LPC) ses yolunu modellemektedir. Bu katsayılar konuşmacı tanımada sıklıkla kullanılmasına rağmen gürültüden oldukça etkilenmektedirler (Tierney 1980). Bu nedenle gürültülü ses içeren uygulamalarda süzgeç takımından elde edilen özellikler daha gürbüzdürler (Van Alphen ve Pols 1991, Paliwal 1992, Reynolds ve Rose 1995).

1.1.5. Mel Ölçekli Kepstrum Katsayıları

Mel ölçekli kepstrum katsayıları günümüzde en çok kullanılan özellik vektörleridir. Doğrusal öngörü analizine gerek kalmadan hesaplanabilmektedir. Çünkü doğrusal öngörü analizi ses yolunu modellerken, mfcc özellikleri insan kulağını modellemektedir. Süzgeç takımı kullanılarak elde edildiğinden dolayı gürültülü seslerde öngörücü katsayılarına göre daha iyi performans göstermektedir. İnsanın sesi algılama karakteristiğine dayanmaktadır. Genellikle mfcc özelliklerine ek olarak bunların türevleri de kullanılmaktadır. Mfcc katsayıları hem konuşma tanıma hem de konuşmacı tanıma uygulamalarında başarılı sonuçlar vermektedir (Reynolds 1995).

1.2. Sınıflandırma Yöntemleri

Konuşmacı tanıma basitçe bir örüntü sınıflandırma problemidir. Verilen bir test cümlesine ait özellik vektörlerini kullanarak bu test cümlesini hangi konuşmacının söylediğini bulmak sınıflandırıcının görevidir. Bu görevi yerine getirmek için her konuşmacının eğitim verileri ile akustik modeller oluşturulur. Sınıflama aşamasında test cümlesine ait özellik vektörlerinin eğitim kümesindeki konuşmacılara ait şablonlarla olan benzerliğine bakılır. Bu benzerlik ölçütü yardımı ile konuşmacı tanıma sistemi test cümlesinin kim tarafından söylendiğini belirler.

Konuşmacı tanıma uygulamalarında çeşitli sınıflandırıcı teknikleri kullanılmaktadır. Bu teknikler genel olarak şablon tabanlı ve istatistiksel olmak üzere iki gruba ayrılabilir. Bu bölümde bu sınıflandırıcılar hakkında genel bilgilere yer verilecektir.

1.2.1. Şablon Modeller

Şablon model tabanlı sınıflandırıcılar en basit sınıflandırıcılardandır. Bu nedenle konuşmacı tanıma uygulamalarında ilk kullanılmaya başlanan yöntemler genellikle şablon modeller grubuna ait sınıflandırıcılardan oluşmuştur. En yaygın şablon modeller Dinamik Zaman Eğirme ve Vektör Nicemlemedir.

1.2.1.1. Dinamik Zaman Eğirme

Dinamik Zaman Eğirme konuşmacı tanıma uygulamalarının ilk dönemlerinde kullanılmaya başlanan bir sınıflandırma tekniğidir. Metinden bağımsız konuşmacı tanıma uygulamaları için kullanışlı ve iyi sonuçlar veren bir yöntemdir. 1980'li yıllarda oldukça popüler bir yöntem olmasına karşın günümüzde yerini istatistiksel modellere bırakmıştır (Furui 1994).

Konuşma hızındaki değişikliklerden dolayı bir konuşmacının farklı zamanlarda söylediği aynı cümleler arasında zamanlama açısından farklılıklar ortaya çıkmaktadır. Zamanlamadaki bu problem test cümlesi ile eğitim cümlesi arasındaki önemli benzerlikleri bulmak için Dinamik zaman eğirme algoritması ile çözülür (Doddington 1985).

DTW, test cümlesi ile eğitim şablonunu karşılaştırırken muhtemel yollardan optimum olanı bulabilmektedir. Verilen bir referans (eğitim) şablonu *R* ve test cümlesi *T* için N_R ve N_T sırasıyla eğitim ve test cümlelerindeki çerçeve sayıları olsun. DTW *T*'nin zaman ekseni *n* ' yi *R* ' nin *zaman* eksenine eşleştiren bir $m = w(n)$ fonksiyonu bulmaktadır.

DTW, *T* cümlesini çerçeve çerçeve, *R* cümlesindeki en iyi çerçeveyi bularak aşağıdaki karşılaştırmayı yapabilmek için tarar.

$$D = \min_{w(n)} \left[\sum_{n=1}^{T} d(T(n), R(w(n))) \right] \tag{1.1}$$

Denklem (1.1)'de d, *T* cümlesine ait *n.* çerçeve ile *R* cümlesine ait w(n). çerçeve arasındaki bir uzaklık ölçütü ve D, en iyi yolu veya en iyi eşleşmeyi temsil eden uzaklık ölçütüdür.

Verilen bir test cümlesine ait özellik vektörü dizisi için, DTW bütün referans şablonlar arasından en iyi eşleşme uzaklıklarını bulmaktadır. Sistem bu uzaklıkları saklar ve test cümlesinin en küçük uzaklık veren şablona ait olduğu kararını verir. DTW genellikle metine bağımlı konuşmacı tanıma uygulamalarında kullanılmaktadır (Campbell 1997).

1.2.1.2. Vektör Nicemleme

DTW genellikle metine bağımlı konuşmacı tanıma uygulamalarında kullanılmaktadır. Eğer amaç metinden bağımsız konuşmacı tanıma gerçekleştirmek ise muhtemel yöntemlerden birisi konuşmacıyı modellemek için konuşmacıya ait tüm özellik vektörlerini kullanmaktır. Fakat özellik vektörü boyutunun yüksek olduğu durumlarda bu yaklaşım pratik değildir. Bu nedenle bu tür yaklaşımlarda genellikle özellik vektörü boyutunu azaltan/sıkıştıran yöntemler kullanılmaktadır.

Kişiye özgü kod kitabı kullanan VN yöntemi hem konuşma hem de konuşmacı tanımaya başarıyla uygulanan ve en bilinen yöntemlerden biridir (Li ve Wrench 1983, Soong ve diğ. 1985, Rosenberg ve Soong 1987, Matsui ve Furui 1990, Matsui ve Furui 1991). VN' de, her bir konuşmacıya ait özellik vektörlerinden, bu vektörleri temsil eden az sayıda vektör elde edilir. Her konuşmacı bir kod kitabı ile temsil edilir. Kod kitabı özellik vektörlerinin ortalamalarından oluşan kod vektörlerinden oluşmaktadır.

1.2.2. İstatistiksel Modeller

Günümüzde çoğu konuşmacı tanıma sisteminde istatistiksel modeller kullanılmaktadır. İstatistiksel modeller istatistiksel skorlarla uygun ve anlamlı sonuçlar ortaya çıkarmaktadır. Bir istatistiksel model tabanlı sınıflandırıcıda, özellik eşleştirme işleminde verilen bir konuşmacı modeli kullanılarak test cümlesine ait bir olabilirlik hesabı yapılır.

Bir konuşmacıya ait eğitim verilerinin eğitilmesi sonucu oluşan bir istatistiksel model λ^s olsun. Sistem N adet kişiyle eğitildiğinde sistemde N adet istatistiksel model olacaktır. Bir test cümlesine ilişkin özellik vektörü $Y = (y_1, y_2, ..., y_L)$ olsun. Amacımız bu test cümlesini sistemde kayıtlı olan N adet kişiden hangisinin söylediğini bulmaktır. Bu işlem olasılık hesabı ile yapılmaktadır.

$$P(Y|\lambda^s) = p(y_1, y_2, ..., y_L|\lambda^s) \qquad s = 1,2,...,N_s \qquad (1.2)$$

Bu olasılıkların tamamı hesaplandıktan sonra karar aşağıdaki kurala göre verilir.

$$s^* = \arg\max_{1 \le s \le N_s} p(Y|\lambda^s) \tag{1.3}$$

Eğer ardışık çerçevelere ait özellikler arasında ilişki yoksa (bağımsızlarsa) denklem (1.2) şu şekilde düzenlenebilir.

$$P(Y|\lambda^s) = \prod_{i=1}^{L} p(y_i|\lambda^s) \tag{1.4}$$

İstatistiksel modellerde Denklem (1.4)'den her sınıf için elde edilen olasılık değerine göre sınıflama yapılır.

Literatürde bu olasılığı hesaplayan bir çok yöntem vardır. En önemlileri Gauss Karışım Modeli (Reynolds 1992), Saklı Markov Modelleri (Rabiner ve Juang 1993) ve Yapay Sinir Ağlarıdır (Chester 1993). Bu kısımda kısaca bu yöntemlerden bahsedilecektir.

1.2.2.1. Saklı Markov Modelleri

Dizilerin modellenmesinde kullanılan diğer bir istatistiksel model türü Saklı Markov Modelleridir (SMM) (Rabiner 1989). SMM her bir gözlem vektörünün bir durumun istatistiksel fonksiyounu olduğu istatistiksel bir süreçtir. Bu istatistiksel fonksiyon direkt olarak gözlenemez ancak başka bir istatistiksel süreç tarafından gözlenebilir bu nedenle Saklı Markov Modelleri adını almaktadır (Rabiner ve Juang 1993). SMM sonlu sayıda durumdan oluşan ve her durumun özellik vektörüne ait olasılık yoğunluk fonksiyonunu içerdiği bir süreçtir. SMM'de durumlar birbirlerine bir durum geçiş işlevi aracılığı ile bağlıdır. Durum geçiş olasılıkları, a_{ij}, bir durumdan diğer bir duruma geçiş olasılıklarını belirtmektedir. SMM tabanlı sınıflandırıcılar genellikle metine bağımlı konuşmacı tanıma yöntemleri için uygundur.

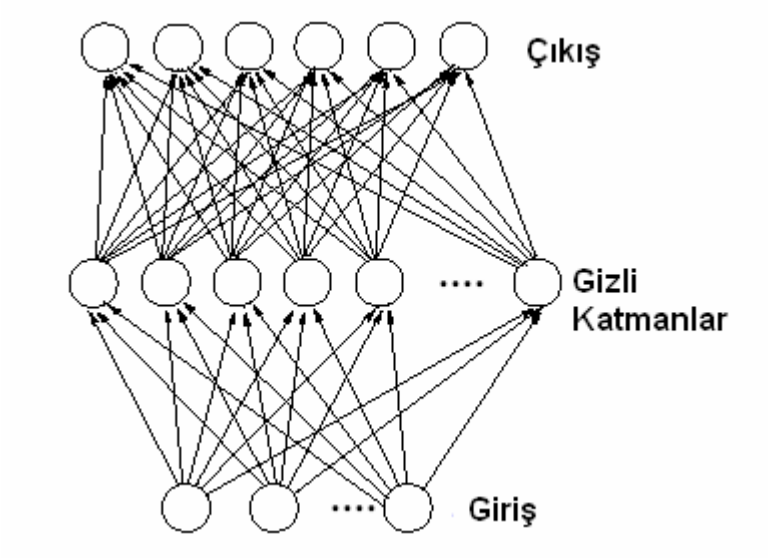
Saklı Markov Modelleme çeşitli ses tanıma uygulamalarında başarı ile kullanılmış olup (Juang ve diğ. 1985, Rabiner ve diğ. 1988, Rabiner ve Juang 2005) konuşmacı tanıma uygulamalarında da ses tanımada olduğu gibi yüksek başarı göstermiştir. SMM

tabanlı yöntemler metinden bağımsız uygulamalarda VN yöntemi ile kıyaslanabilecek başarım göstermiş olup (Thisby 1991), metne bağımlı uygulamalarda ise diğer yöntemlere kıyasla daha iyi sonuçlar vermiştir (Reynolds and Carlson 1995).

1.2.2.2. Yapay Sinir Ağları (NN)

Yapay sinir ağları temelli sınıflandırıcılar hem metine bağımlı hem de metinden bağımsız uygulamalarda kullanılmaktadır. NN giriş ve çıkış arasında eşleştirme yapmakta oldukça başarılıdır ve eğitilmiş sınıflar için sonsal olasılıkları tahmin edebilmektedir. NN lineer olmayan karar yüzeylerini tahmin edebilmektedir. NN, sinir ağının arzu edilen transfer fonksiyonunu oluşturmak için birbirine bağlı az sayıda fonksiyonel birimlerden (nöron) oluşmaktadır. Yapay Sinir Ağlarının birçok türü vardır. Bunlardan bazıları

- Çok katmanlı algılayıcılar (Multi-Layer Perception) (Oglesby ve Mason 1990, Rudasi ve Zahorian 1991),
- Radyal Tabanlı Fonksiyon (Radial basis function) (Oglesby ve Mason 1991),
- Öğretici Vektör Nicemleyici (Learning Vector Quantizer) (Bennani ve Gallinari 1991).



Şekil 1.1. İki katmanlı bir yapay sinir ağı

Bu türlerden en yaygın kullanılanı MLP'dir. MLP bir giriş katmanı, belirli sayıda gizli katman ve bir çıkış katmanından oluşur (Şekil 1.1). Giriş katmanı girişleri bütün gizli nöronlara dağıtan lineer olmayan bir katmandır. Çıkış katmanındaki her bir nöron direkt

olarak bir sınıfla ilişkilidir. Giriş işareti, giriş nöronları tarafından MLP'ye iletilir ve her bir çıkış nöronu ilgili sınıf için sonsal olasılığı içerir. Giriş işareti, hangi nöron en yüksek olasılığa sahip ise o sınıfa atanır.

Yapay sinir ağları tekniğinin en önemli dezavantajı sisteme yeni bir kişi eklendiğinde tüm sistemin tekrar eğitilmesi gerekliliğidir (Reynolds ve Rose 1995).

1.2.2.3. Gauss Karışım Modeli

Gauss Karışım Modeli (GKM) tabanlı yöntemler metinden bağımsız konuşmacı tanıma uygulamalarında kullanılmaktadır. GKM, konuşmacı tanımda ilk defa 1990 yılında Reynolds kullanılmıştır (Reynolds 1992). Bu yöntem metinden bağımsız uygulamalarda oldukça iyi sonuçlar vermektedir. GKM'de *n*. çerçeveye ait özellik vektörünün olasılığı, $p(y_n|\lambda^s)$, M adet çok boyutlu Gauss olasılık yoğunluk fonksiyonunun ağırlıklandırılmış toplamından elde edilir.

$$p(y_n|\lambda^s) = \sum p_i^s b_i^s(y_n) \tag{1.5}$$

Bu ifadede $b_i^s(y_n)$, ortalaması μ_i ve ortak değişinti matrisi $\sum_i^s$ olan i. karışım bileşenini göstermektedir. Ağırlık katsayıları aşağıdaki şartı sağlamaktadır.

$$\sum_{i=1}^{M} c_i^s = 1 \tag{1.6}$$

Herhangi bir konuşmacıya ait bir GKM modeli şu şekilde ifade edilir:

$$\lambda^s = \left\{c_i^s, \mu_i^s, \sum_i^s\right\}, \qquad 1 \le i \le M \tag{1.7}$$

GKM yöntemi tek durumlu sürekli ergodik SMM' ye karşılık gelmektedir (Matsui ve Furui 1994). Bozuk ve kısıtlanmamış ses kullanan konuşmacı tanıma sistemlerinde yüksek başarım sağladığı kanıtlanmıştır (Reynolds ve Rose 1995). Bu yöntem işlem maliyeti az

ve gerçek zamanlı uygulanabilirliği kolay olan bir yöntemdir (Reynolds 1992, Reynolds ve diğ. 1992).

Bu tezde kişiyi temsil eden özellikler olarak mfcc katsayıları, sınıflandırıcı model olarak ise vektör nicemleme (VQ) ve Saklı Markov Modelleri kullanılmıştır. Bu yöntemlerle ilgili ayrıntılı açıklamalar 2. bölümde verilmiştir.

2. MATERYAL VE YÖNTEM

Genel olarak, bir konuşmacı tanıma sistemi iki ana kısımdan oluşmaktadır. Bunlardan ilki, kişinin ses örneklerinden o kişiyi en iyi şekilde ayırt edebilecek ses özelliklerini çıkartma ve daha sonraki ise bu özellikleri kullanarak o kişiye ait model oluşturmaktır. Özellik çıkarma ve model oluşturma için öncelikle sistemin kullanacağı bir veritabanına ihtiyaç vardır. Veritabanı seçiminde dikkat edilmesi gereken en önemli noktalar; herkesin erişebileceği, yaygın olarak kullanılan ve dünyaca kabul edilir özelliklere sahip olmasıdır. Ancak bu özelliklere sahip bir veritabanı ile yapılan çalışmalar birbirleriyle kıyaslanabilir. Bu nedenlerden dolayı bu tezde bu şartlara uygun, oldukça sık kullanılan ve iyi bilinen bir veritabanı olan TIMIT (Jankowski ve ark. 1990) kullanılmıştır. TIMIT veritabanındaki kişilere ait ses örnekleri işlenerek, kişileri en iyi karakterize eden mel ölçekli kepstrum katsayıları elde edilmiştir. Daha sonra bu özellikler sınıflanarak kişilere ait modeller oluşturulmuştur. Modelleme aşamasında vektör nicemleme ve saklı markov modeli olmak üzere iki ayrı yöntem kullanılarak sonuçlar elde edilmiştir. Bu bölümde, özellik vektörlerinin elde edilmesi, sınıflandırılması ve konuşmacı tanıma sisteminde kullanılması aşamaları detaylı olarak verilmiştir.

2.1. Veritabanı

TIMIT veritabanında Amerikan İngilizcesinin 8 ana lehçesine sahip bölgelerden seçilmiş 438 erkek, 192 kadın olmak üzere toplam 630 konuşmacıya ait 10’ ar fonetik olarak zengin cümle bulunmaktadır. Konuşmalar sessiz ortamda karbon mikrofon kullanarak kaydedilmiş ve 16 kHz de örneklenmiştir. Bu tezde, her konuşmacının 10 cümlesinden 7 tanesi eğitim, kalan 3 cümle ise test için kullanılmıştır.

2.2. Özellik Seçimi

Ses işareti konuşmacı ile ilgili değişik bilgiler içermektedir. Ses işaretinin içerdiği bilgiler arasında kullanılan dil, lehçe, konuşmanın içeriği ve konuşmacının ruhsal

durumu gibi önemli özellikler vardır. Ses işareti konuşmacının fiziksel özelliklerinin (ses yolu boyutu, çevresel etkenler ve iletim kanalı) ve ruhsal durumunun bir fonksiyonu olarak düşünülebilir (Naik 1990). Bu nedenle farklı konuşmacıların ses örnekleri arasında ve hatta aynı konuşmacıdan değişik zamanlarda alınmış ses örnekleri arasında farklılıklar vardır. Özellik çıkarma işlemi iki nedenden dolayı önemlidir. Bunlardan ilki, konuşmacılara ait istatistiksel modellerin gürbüz olması için, ikincisi ise eğitim verilerinin ölçülebilir boyutlarda olması, bu sayede de işlem fazlalığını azaltmak içindir (Kinnunen 2003). İdeal özellik vektörlerinin sağlaması gereken belirli şartlar vardır Bu şartlar,

- Konuşma sırasında doğal olarak ve sıklıkla ortaya çıkmalı,
- Kolay ölçülebilir olmalı,
- Zamanla değişmemeli veya kişinin sağlık durumundan etkilenmemeli,
- Gürültüden veya iletim hattından etkilenmemeli,
- Taklitlelere karşı hassas olmamalı,

şeklinde sıralanabilir (Wolf 1972).

Pratikte bu şartların tümünü aynı anda sağlayan özellikleri bulmak oldukça zordur. Uygulamanın türüne göre gerekli şartları sağlayan özellikler kullanılmalıdır.

Ses işaretinin akustik parametreleri zamanla değişen ve zamanla değişmeyen parametreler olmak üzere iki gruba ayrılabilir. Zamanla değişmeyen parametrelerin en önemli avantajı konuşmanın içeriğinden çok konuşmacıyı temsil etmesidir ve bu yüzden de metinden bağımsız konuşmacı tanıma uygulamaları için uygundur (Atal 1976) .

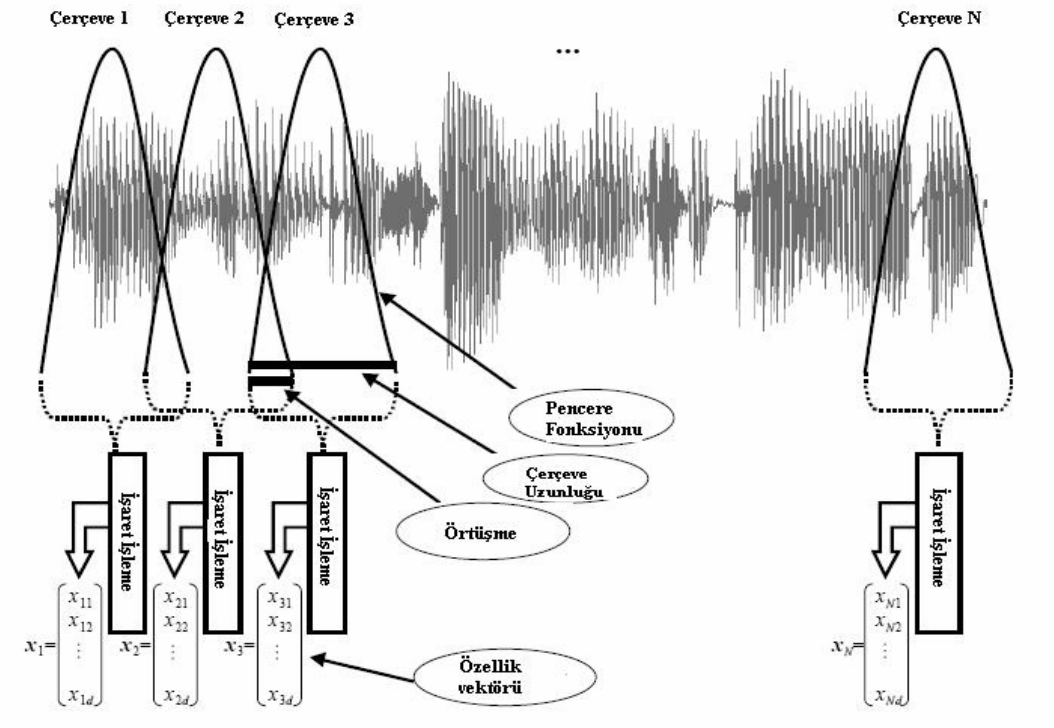
Konuşmacı tanıma uygulamalarında özellik seçiminde üzerinde durulan temel noktalardan birisi kullanılan özelliklerin sesin perde ve spektrum özelliklerini temsil edecek şekilde seçilmesidir (Reynolds 1992). Sesin spektrumunu temsil eden özelliklerden doğrusal öngörü katsayıları ve bunların değişik türevleri (PARCOR katsayıları, Kepstrum katsayıları) ile süzgeç dizisi enerjileri ve bunların kepstral dönüşümleri en yaygın olarak kullanılanlardır.

2.2.1. Kısa Dönem Analizi

Ses işareti, ses yolunun yapısı itibariyle sıkça değiştiği için işaret kısa bölümler halinde işlenmelidir. Böylece işaret yeterince küçük parçalar halinde işlendiği zaman daha kararlı sessel özellik gösterecektir (Deller ve ark. 1993) Böylece işaretin kısa dönemli bir bölümünden özellikler çıkarılmış olacaktır ki, bu işleme kısa-dönem analizi denilmektedir. Şekil 1 de kısa dönem analizinin adımları gösterilmektedir. Kısa dönem analizinde işaret belirli kısımları örtüşen küçük parçalara (çerçeve) ayrılır. Örtüşmenin sebebi bilgi kaybını engellemektir. Her bir çerçeve uzunluğu önceden tanımlanmış bir pencere fonksiyonu ile çarpılır. Pencere fonksiyonu ile çarpılan çerçevelere pencerelenmiş çerçeveler de denilmektedir. Konuşmacı tanıma sistemlerinde kullanılan birçok pencere fonksiyonu vardır ancak bunlardan en yaygın olanı *Hamming* Penceresidir. *Hamming* penceresinin matematiksel gösterimi;

$$w(n) = 0.54 - 0.46\cos(\frac{2n\pi}{N-1}) \tag{2.1}$$

şeklindedir. Denklem (2.1) de *N*, pencere boyutu ya da çerçeve boyutunu ifade etmektedir.

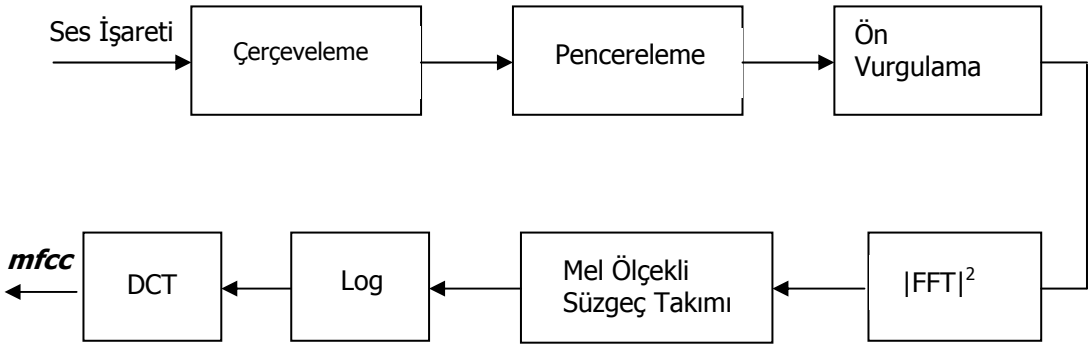


Şekil 2.1. Kısa-Dönem Analizi

Bir çerçeveden elde edilen özellikler kümesine özellik vektörü adı verilir. Kısa dönem analizinden sonra özellik vektörlerinin elde edilmesinde değişik yöntemler kullanılmaktadır. Bu çalışmada kişinin sesini karakterize eden özellikler olarak Mel Ölçekli Kepstrum Katsayıları kullanılmaktadır.

2.2.2. Mel Ölçekli Kepstrum Katsayıları (MFCC)

Mel ölçekli kepstrum katsayıları (mfcc), ses işaretini temsil eden özellikler arasında en çok bilinen özelliklerdir. Mfcc özellikleri, ses işaretinin düşük frekans bileşenlerinin taşıdığı bilgi miktarının insanlar açısından yüksek frekans bileşenlerinin taşıdığı bilgi miktarına göre daha önemli olduğu temeline dayanır (Deller ve ark. 1993). Mfcc özellikleri kısa dönem analizinden sonra her bir çerçeveden bu özelliklerin elde edilmesi şeklinde olur. Mfcc özelliklerinin elde edilmesi sırasında izlenen adımlar Şekil 2.2 de gösterilmektedir.



Şekil 2.2. Mel ölçekli kepstrum katsayıları akış diyagramı

2.2.2.1. Çerçeveleme

Giriş işareti, M örnekten oluşan kısımları örtüşen N örnek uzunluğunda konuşma parçalarına bölünür (M<N). İlk çerçeve N örnekten oluşurken sonraki çerçeve, ilk çerçeveden M örnek sonra başlar ve böylece N-M kadar örnek örtüşür. Deneyler sırasında TIMIT veritabanı için 10 ms'lik kısımları örtüşen 20 ms uzunluklu çerçeveler kullanılmıştır.

2.2.2.2. Pencereleme

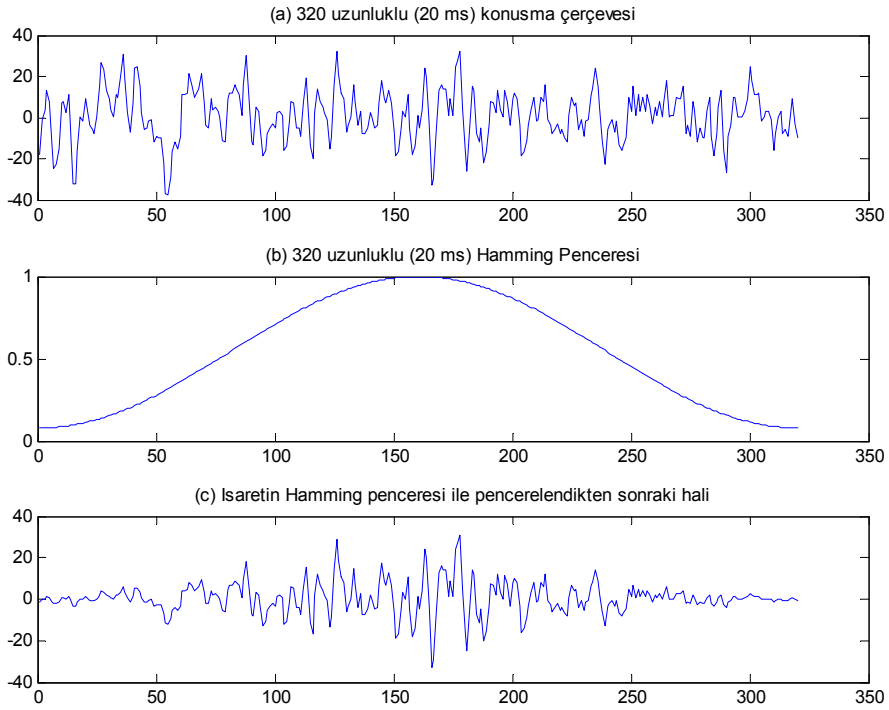
Çerçeveleme işleminden sonraki adım olan pencereleme işleminde amaç sinyalin başındaki ve sonundaki süreksiz kısımları azaltmak, dolayısıyla sinyalin başındaki ve sonundaki bilgi içermeyen bölümleri bastırarak spektral bozulmayı engellemektir. Giriş işaretimizi *x(n)*, pencere fonksiyonunu *w(n)* ve çıkış işaretimizi ise *y(n)* ile ifade edecek olursak, çıkış işaretimiz,

$$y(n) = x(n)w(n) \tag{2.2}$$

şeklinde olacaktır. Genellikle pencere fonksiyonu olarak Hamming penceresi kullanılır ve Hamming penceresinin matematiksel ifadesi Denklem (2.1)'de verilmektedir.

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1}) \quad , \qquad 1 \le n \le N \tag{2.3}$$

şeklindedir.



Şekil 2.3. (a) 20 ms uzunluklu konuşma çerçevesi, (b) 20 ms uzunluklu Hamming penceresi, (c) konuşma işaretinin pencerelenmiş hali

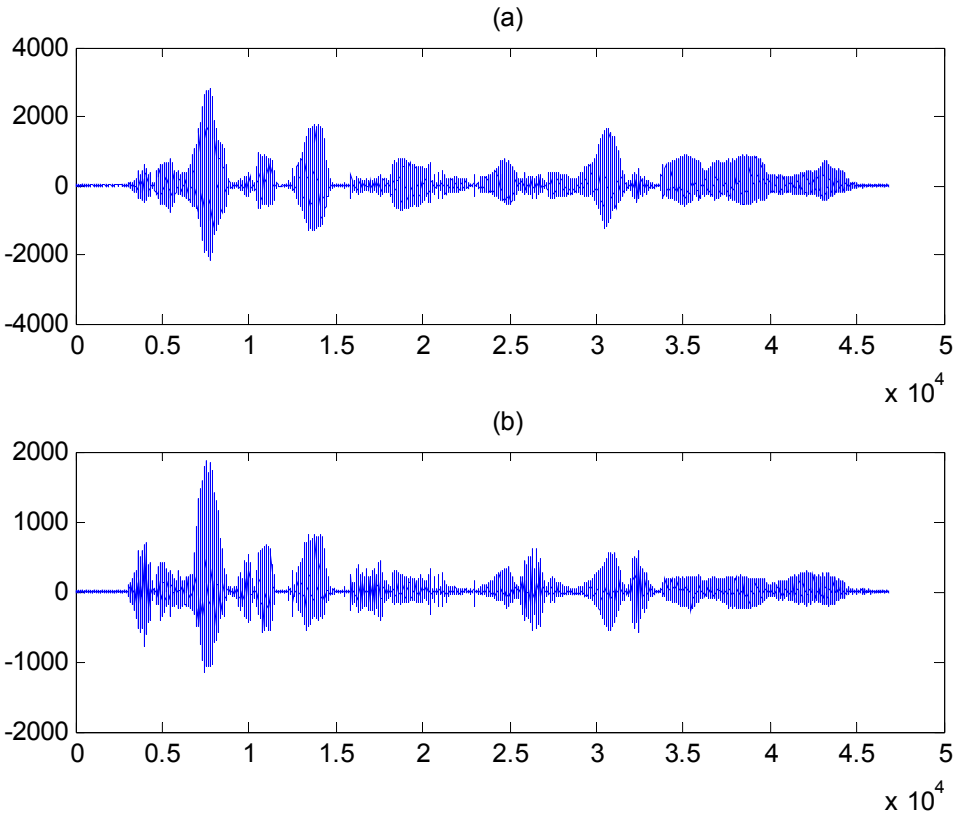
Denklem (2.3) de *N*, çerçeve uzunluğunu ifade etmektedir. Şekil 2.3 (a)'da 20 ms uzunluklu bir Hamming Penceresi, (b)'de 20 ms uzunluklu bir konuşma işareti ve (c)'de işaretin Hamming penceresi ile pencerelenmiş hali görülmektedir.

2.2.2.3. Ön Vurgulama

Ön vurgulama işleminde giriş işareti birinci dereceden bir FIR süzgeç girişine uygulanır. Birinci dereceden süzgecin transfer fonksiyonu,

$$H(z) = 1 - 0.95z^{-1} \tag{2.4}$$

şeklindedir. Ön vurgulama işleminin amacı sinyalin yüksek frekans bileşenlerini daha baskın hale getirmektir. Şekil 2.4 (a) orijinal ses işaretini ve (b) ön vurgulama işlemi yapıldıktan sonra süzgeç çıkışında elde edilen işareti göstermektedir.



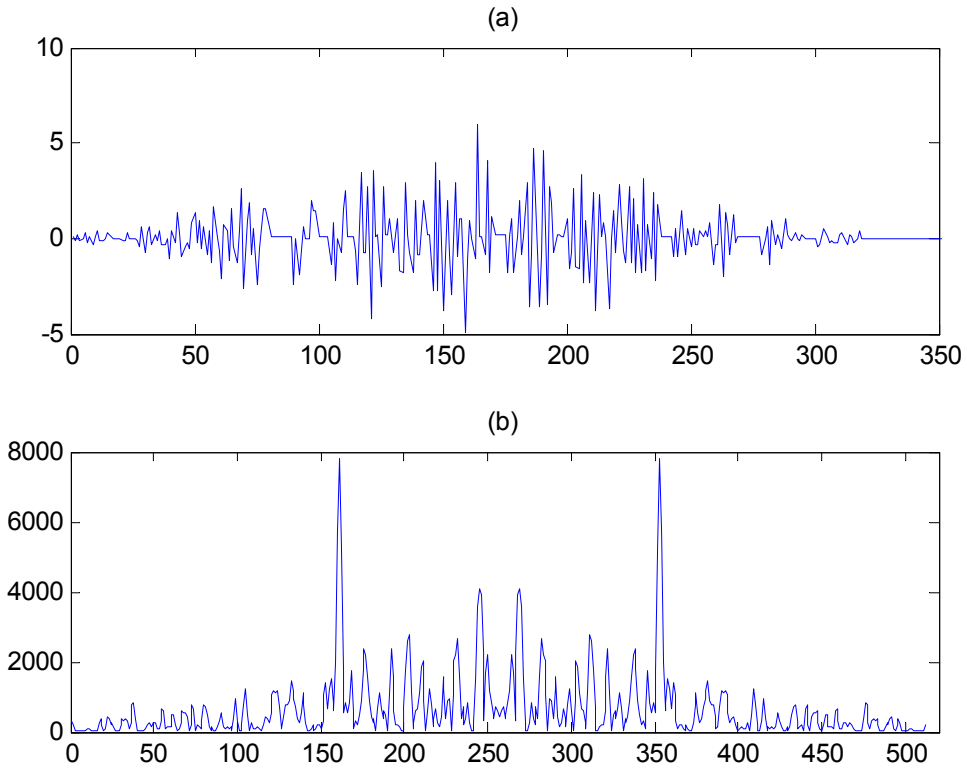
Şekil 2.4. (a) Orijinal ses işareti, (b) Ön vurgulama işleminden sonra elde edilen işaret

2.2.2.4. Hızlı Fourier Dönüşümü (HFD)

N örnekten oluşan konuşma parçasını zaman domeninden, frekans domenine çevirmek için Hızlı Fourier Dönüşümü uygulanır. HFD, Ayrık Fourier Dönüşümünü (AFD) hızlandırmak için uygulanan bir algoritmadır. N, örnekli bir set için AFD'nin matematiksel ifadesi,

$$X_n = \sum_{k=0}^{N-1} x_k e^{-2\pi jkn/N}, \qquad n = 0,1,2,\ldots\ldots\ldots,N-1 \tag{2.5}$$

şeklindedir. Şekil 2.5 (a)'da ön vurgulama işlemi yapılmış ve daha sonra da Hamming penceresi ile pencerelenmiş konuşma çerçevesi, (b)'de ise Hızlı Fourier Dönüşümü alınarak elde edilen genlik spektrumu görülmektedir.



Şekil 2.5. (a) Konuşma çerçevesi, (b) konuşma çerçevesinin genlik spektrumu

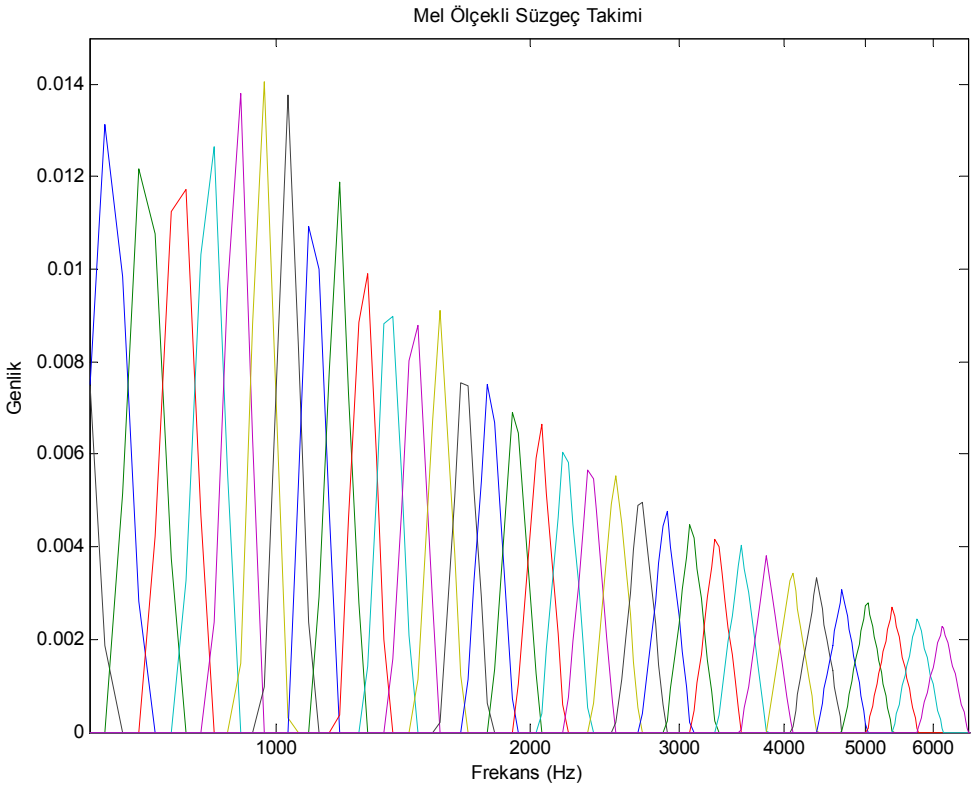
2.2.2.5. Mel Ölçekli Süzgeç Takımı

Akustik çalışmalar sonucunda konuşma sinyallerinin frekans ortamındaki içeriklerinin doğrusal ölçekli olmadığı sonucuna varılmıştır (Rabiner ve Juang 1993). Bu

sonuç yeni bir ölçeğin tanımlanmasına sebep olmuştur. Böylece gerçek frekansı *f* (*Hz*) olan bir işaret *mel* adında bir ölçeklendirme ile ifade edilir. 1 *kHz* frekanslı bir sesin, insan kulağının algısal duyma eşiğinin 40 dB yukarısı 1000 *mel* olarak tanımlanır. Diğer değerler referans sese göre ayarlanır (Rabiner ve Juang 1993). Mel ölçeği 1 kHz'e kadar doğrusal, 1 kHz'den sonra ise logaritmik olarak değişen aralıklarla ifade edilen bir ölçektir. Verilen bir *f* (*Hz*) frekansını mel frekansı ölçeğinde ifade etmek için,

$$mel(f) = 2595\log(1+\frac{f}{700}) \tag{2.6}$$

denklemi kullanılır. Genlik spektrumu hesaplanan işaret bir sonraki adımda mel ölçekli süzgeç takımından geçirilir. Mel ölçekli süzgeç takımı, 1 kHz'e kadar doğrusal, 1 kHz'den yüksek frekanslarda ise logaritmik olarak yerleştirilmiş üçgen süzgeçlerden oluşmaktadır. Şekil 2.6' da mel ölçekte yerleştirilmiş süzgeç takımı görülmektedir.



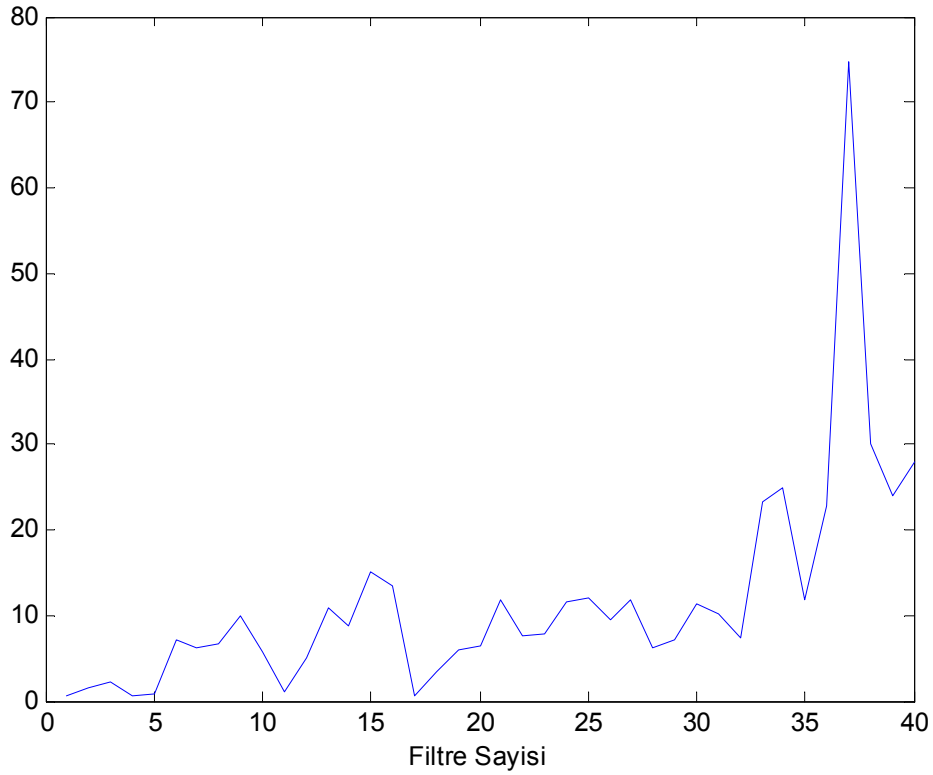
Şekil 2.6. Mel ölçekli süzgeç takımı.

Mel ölçekli süzgeç takımında kullanılacak süzgeç sayısı (FS) işaretin band genişliğini kapsayacak şekilde seçilmektedir. İşaretin örnekleme frekansı f_s ise süzgeç sayısı [0,

$f_s/2$] frekans aralığını kapsayacak şekilde seçilmelidir. *l* süzgeç takımında bulunan süzgeçlerden biri olsun. Bu filtrenin merkez frekansı f_{c_l}, alt ve üst frekansları ise sırasıyla $f_{c_{l-1}}$ ve $f_{c_{l+1}}$ olur. *N* noktalı ayrık Fourier dönüşümü ile genlik spektrumu için üçgen filtreler ayrık Fourier dönüşümü frekans indisi *k* ile tanımlanır, $k \in [0, N/2]$.

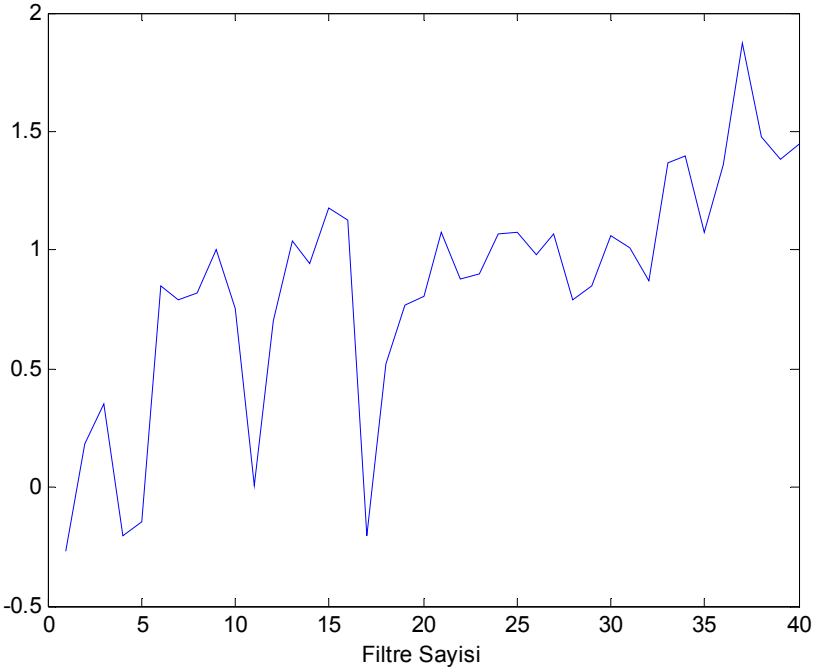
$$F_l[k] = \begin{cases} ((\frac{k}{N})f_s - f_{c_{l-1}})/(f_{c_l} - f_{c_{l-1}}) & L_l \le k \le C_l \\ (f_{c_{l+1}} - (\frac{k}{N})f_s)/(f_{c_l+1} - f_{c_l}) & C_l < k \le U_l \end{cases} \quad (2.7)$$

Burada $C_l = \frac{f_{c_l}}{f_s} N$, $U_l = \frac{f_{c_l+1}}{f_s} N$ ve $L_l = \frac{f_{c_{l-1}}}{f_s} N$ sırasıyla l. filtreye ait merkez, alt ve üst frekans indislerini belirtmektedir (Reynolds 1992).



Şekil 2.7. Süzgeç takımı çıkışında elde edilen işaret

Süzgeç takımında 40 adet süzgeç kullanılması durumunda, genlik spektrumu elde edilmiş işaret süzgeç takımının girişine uygulandığında çıkışta 40x1 uzunluklu bir vektör, yani her bir filtreden vektörün bir elemanı elde edilmektedir. Şekil 2.7 genlik spektrumu alınmış işaretin süzgeç takımından geçirilmesi ile çıkışta elde edilen işareti göstermektedir.



Şekil 2.8. Süzgeç çıkışında elde edilen işaretin logaritması alındıktan sonra elde edilen işaret

2.2.2.6. Logaritma Alma

Sonraki adımda ise süzgeç çıkışında elde edilen işaretin logaritması alınmaktadır. *l.* filtrenin logaritmik enerji çıkışı *c(l)* ile gösterecek olursak,

$$c(l) = \log(\frac{1}{A_l} \sum_{k=L_l}^{U_l} F_l[k]X[k]) \tag{2.8}$$

şeklinde hesaplanır. Denklem (2.8)'deki A_l filtrelerin band genişliğine bağlı olarak kullanılan normalizasyon katsayısı olup,

$$A_l = \sum_{k=L_l}^{U_l} F_k[k] \tag{2.9}$$

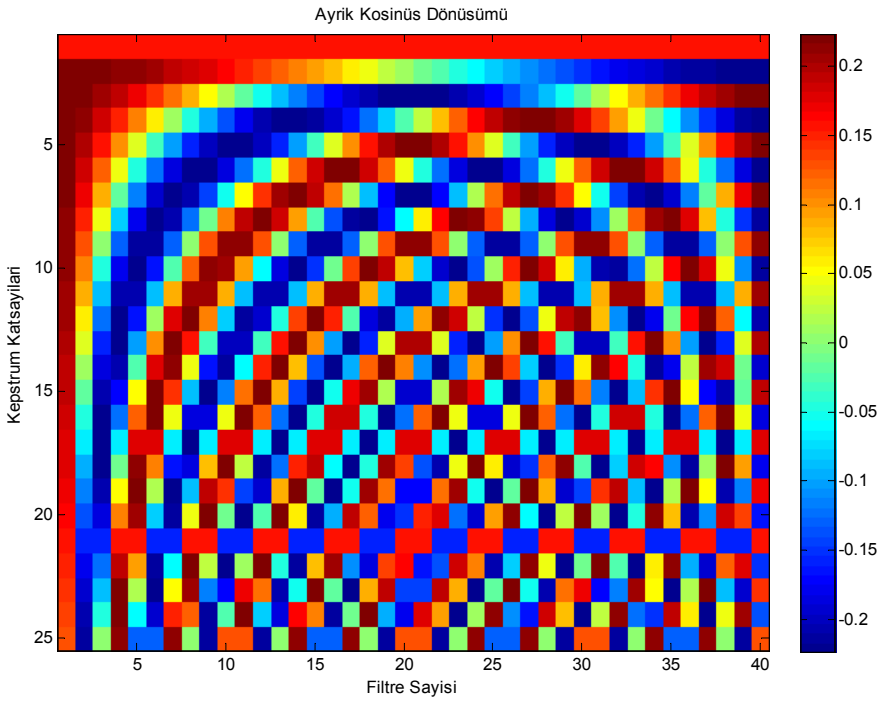
şeklinde hesaplanır. Logaritma alarak özellik vektörlerinin değişimlere karşı daha az hassas olmaları sağlanmaktadır. Şekil 2.8' de işaretin logaritmasının alındıktan sonraki hali görülmektedir.

2.2.2.7. Ayrık Kosinüs Dönüşümü (AKD)

Mfcc çıkarma işleminin en son adımı olan Ayrık Kosinüs Dönüşümünde logaritmik mel ölçeğindeki veriler tekrar zaman ortamına dönüştürülür. Sonuç olarak da elde edilen veriler Mel frekansı kepstrum katsayıları (mfcc) olarak adlandırılır. Ses sinyalinin spektrumunun kepstral gösterimi ilgili çerçevedeki ses işaretini iyi bir şekilde temsil etmektedir. Mel spektrum katsayıları ve bunların logaritmaları reel sayılar olduğu için zaman ortamına geçmek için ayrık kosinüs dönüşümü kullanılabilir. Logaritma alma işleminden sonra elde edilen işareti c_l ile gösterirsek mfcc katsayıları,

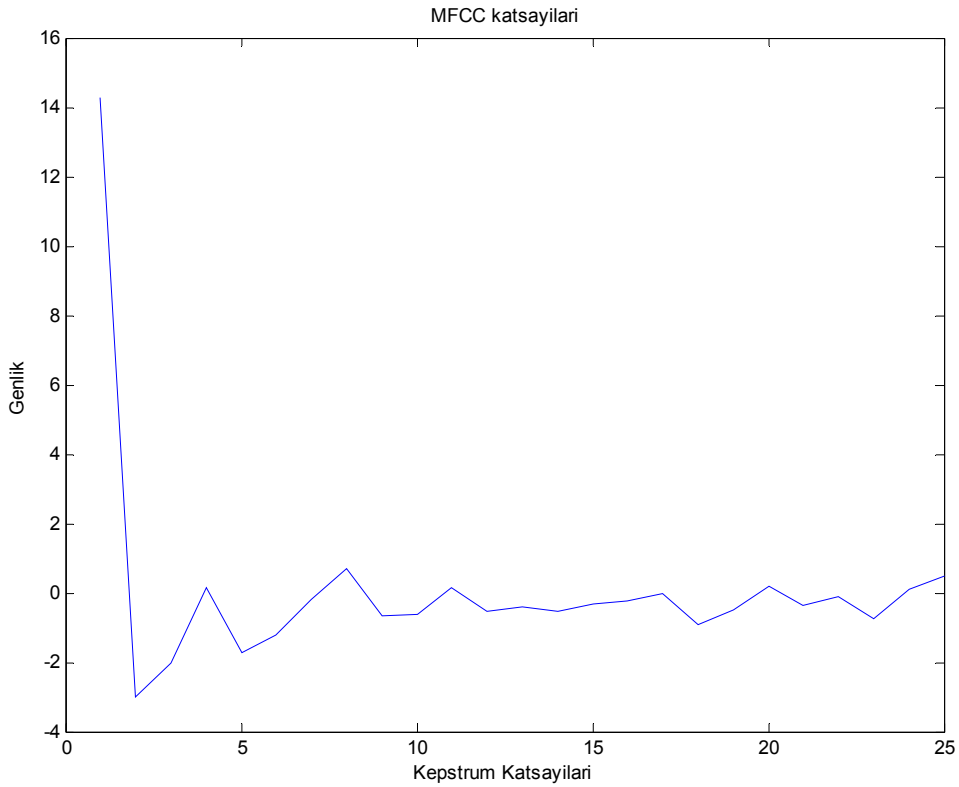
$$mfcc(i) = \frac{1}{FS}\sum_{l=1}^{FS} c_l \cos\left(i\left(l-\frac{1}{2}\right)\frac{\pi}{FS}\right), \qquad i = 1,.....FS-1 \tag{2.10}$$

şeklinde hesaplanır.



Şekil 2.9. Ayrık Kosinüs Dönüşümü Matrisi

Şekil 2.9' da 40 süzgeç ve 25 kepstrum katsayısı için Ayrık kosinüs dönüşümü görülmektedir. Şekildeki renk ölçeğine göre kepstrum katsayılarının aldığı değerler görülmektedir. Şekil 2.10' da ise logaritması alınmış işaretin ayrık kosinüs dönüşümü alındıktan sonra elde edilen veriler görülmektedir.



Şekil 2.10. AKD sonucunda mfcc'lerin elde edilmesi.

2.3. Özellik Eşleştirme ve Konuşmacı Modelleme

Önceki bölümde ses sinyalinden konuşmacıyı temsil eden özellik vektörlerinin elde edilmesi anlatılmıştı. Bu bölümde ise konuşmacı tanıma sisteminin adımlarından olan *sınıflandırma* aşaması anlatılacaktır. Sınıflandırma, verilen bir ses örneğinin sistemde kayıtlı olan kullanıcılardan hangisine ait olduğuna karar verilmesidir. Bu adım genellikle eşleştirme ve modelleme olmak üzere iki gruba ayrılır. Modelleme, kişiye ait ses sinyalinden elde edilen özellik vektörü kullanılarak oluşturulan modelin konuşmacı tanıma sistemine kayıt edilmesidir. Eşleştirme ise bilinen konuşmacı modelleri ile bilinmeyen konuşmacıya ait özellik vektörleri arasındaki benzerliğin ölçülmesidir (Campbell 1997).

Konuşmacı tanıma uygulamalarında sınıflandırma probleminin çözümünde iki temel yöntem kullanılmaktadır: Şablon Eşleştirme ve İstatistiksel Eşleştirme. Şablon yöntemi zamandan bağımsız ya da zamana bağımlı olabilir. Zamana bağımlı yöntemde konuşmacı modeli sabit bir cümleden elde edilen özellik vektörlerinden oluşmaktadır. Tanıma sırasında test cümlesi ile şablon model arasındaki benzerliğin bir ölçüsü olan eşleşme skoru Dinamik Zaman Eğirme yöntemi ile elde edilir. Bu yöntem metine bağımlı uygulamalar için ideal bir yöntem olabilir ancak metinden bağımsız uygulamalarda istenilen düzeyde performans göstermemektedir. Metinden bağımsız uygulamalarda ise özellik ortalama (feature averaging) (Gish ve Schmidt 1994) olarak bilinen yöntemler mevcuttur. Özellik ortalama, herhangi bir kişi için uzun zaman periyodu boyunca ortalama özelliğe olan uzaklık prensibine göre özellik vektörlerinin ortalamasını kullanır.

Diğer bir alternatif yöntem ise ses sinyalinin zamanla değişen karakteristiklerini ifade etmek için istatistiksel bir model oluşturmaktır (Naik 1990). Bu yöntem, konuşmacıların özellik vektörlerinin olasılık yoğunluk işlevi ile modellenmesi ve sınıflandırma ise olasılık veya benzeşime dayalı olarak yapılmasından oluşur. Bu bölümde en çok kullanan modelleme ve eşleştirme yöntemlerinden olan Saklı Markov Modelleri ve Vektör Nicemleme tekniklerinden bahsedilecektir.

2.3.1. Markov Modellleri

Sınıflama yöntemleri içerik bağımlı ve içerik bağımsız olmak üzere iki gruba ayrılır. İçerik bağımsız sınıflamada bir deney sonucunda ortaya çıkan özelliklerin bağımsız olduğu kabul edilir. Bu varsayım sınıflar arasında ilişki olmadığı anlamına gelir. Ses tanıma, konuşmacı tanıma gibi uygulamalarda ardışıl özellik vektörleri birbirinden bağımsız değildir. Bu nedenle sınıflama bütün özellik vektörleri aynı anda kullanılarak yapılmalıdır. Markov Modelleri içerik bağımlı sınıflandırıcılar grubuna girmektedir (Theodoridis ve Koutroumbas 2003). Bu türden sınıflandırıcıların temel başlangıç noktası *Bayes* sınıflandırıcılardır. Diğer bir deyişle bir *x* özellik vektörü,

$$P(w_i|x) > P(w_j|x) \qquad \forall j \neq i \tag{2.11}$$

şartını sağlıyorsa w_i sınıfına atanır. *K* adet gözlemden oluşan bir dizi (özellik vektörü) $X = x_1, x_2, ..., x_K$ ve bu gözlem vektörlerinin atanacağı sınıflar $w_i, (i = 1,2,.....M)$ olsun. $\Omega_i : w_{i_1}, w_{i_2}, ...w_{i_K}$ ise bir gözlem dizisine karşılık gelen muhtemel bir sınıf dizisi olsun. Bu türden sınıf dizilerinin toplam sayısı M^K kadardır. Sınıflandırma yapmadaki amacımız "*hangi* Ω_i *sınıf dizisi, X gözlem dizisi ile uyumludur?"* sorusuna cevap vermektir. Bu yaklaşım, x_1 dizisinin w_{i_1} sınıfına, x_2 dizisinin w_{i_2} sınıfına... atanmasına karşılık gelir. Bayes kuralına göre bir X gözlemi Ω_i sınıfına aşağıdaki şart gerçekleştiğinde atanır.

$$P(\Omega_i | X) > P(\Omega_j | X) \qquad \forall\, j \neq i \tag{2.12}$$

Bu ifade aşağıdaki ifadeye denktir:

$$P(\Omega_i)\, p(\Omega_i | X) > P(\Omega_j)\, p(\Omega_j | X) \quad \forall\, i \neq j \ . \tag{2.13}$$

Bayes sınıflandırıcısı ile bir *X* gözleminin ait olduğu sınıfa karar verebilmek için M^K adet sınıf dizisi üzerinden olasılık hesabı yapmak ve bunlardan maksimum olasılık veren sınıf dizisini elde etmek gerekmektedir. Birçok uygulama için bu tür bir yaklaşım hesaplama karmaşası yüzünden oldukça zordur. Bunun yerine sınıflar arası ilişkiyi kullanan modeller kullanılabilir.

Markov modelleri en çok kullanılan içerik bağımlı sınıflandırıcılardan biridir. Bir $w_{i_1}, w_{i_2}, ...$ sınıf dizisi için Markov modeli aşağıdaki varsayımı yapmaktadır.

$$P(w_{i_k} | w_{i_{k-1}}, w_{i_{k-2}}, ...w_{i_1}) = P(w_{i_k} | w_{i_{k-1}}) \tag{2.14}$$

Bu varsayımın anlamı sınıflar arası bağımlılık sadece birbirini takip eden iki sınıf ile sınırlıdır. Bu tip bir model birinci dereceden Markov model olarak adlandırılır. Diğer bir deyişle sırasıyla $w_{i_{k-1}}, w_{i_{k-2}}, ..., w_{i_1}$ sınıflarına ait olan $x_{k-1}, x_{k-2}, ..., x_1$ gözlemleri (özellik vektörleri) için, x_k gözleminin *k* anında w_{i_k} sınıfına ait olma olasılığı, *k-1* anında x_{k-1}

gözleminin ait olduğu sınıfa bağlıdır. (2.14) denklemini olasılığın zincir kuralı ile yeniden yazarsak aşağıdaki ifade elde edilir.

$$P(\Omega_i) \equiv P(w_{i_1}, w_{i_2}, ..., w_{i_N})$$

$$= P(w_{i_N} \left| w_{i_{N-1}}, ..., w_{i_1}) P(w_{i_{N-1}} \right| w_{i_{N-2}}, ..., w_{i_1}) ... P(w_{i_1})$$

Buradan

$$P(\Omega_i) = P(w_{,1}) \prod_{k=2}^{N} P(w_{i_k} \left| w_{i_{k-1}}) \right. \tag{2.15}$$

elde edilir. Gözlemlerin istatistiksel olarak bağımsız olduğu bir sınıf dizisi için

$$P(X \left| \Omega_i) P(\Omega_i) = P(w_{i_1}) p(x_1 \right| w_{i_1}) \prod P(w_{i_k} \left| w_{i_{k-1}}) p(x_k \right| w_{i_k}) \tag{2.16}$$

şeklinde ifade edilebilir.

2.3.2 Saklı Markov Modellleri

Saklı Markov Modelleri (SMM) bir örüntüye ait çerçevelerin spektral özelliklerini temsil etmede sıklıkla kullanılan ve en çok bilinen yöntemlerden biridir. SMM'nin (veya diğer istatistiksel yöntemler) temelinde yatan en önemli varsayım, ses sinyalinin parametrik bir rastsal süreç olarak iyi bir şekilde karakterize edilebilmesi ve bu rastsal süreç parametrelerinin doğru bir şekilde hesaplanabilmesidir (Rabiner ve Juang 1993).

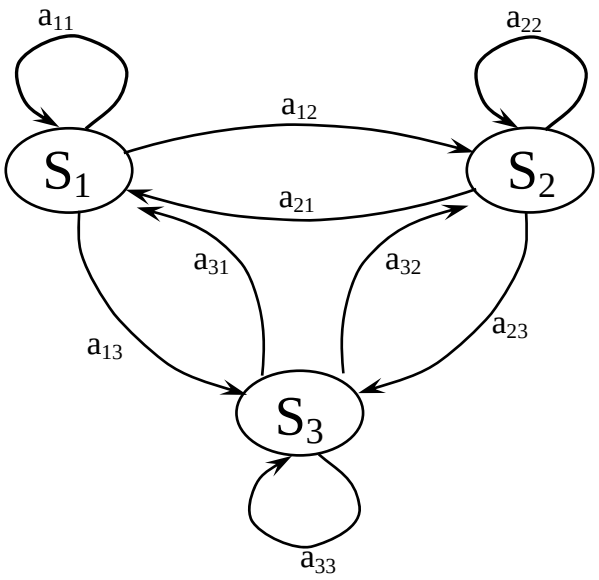
SMM'nin temel kuramı Baum ve arkadaşları tarafından 1960'ların sonlarında belirtilmiştir. Konuşma tanıma uygulamalarında ise 1970'lerde Baker, Jelinek ve arkadaşları tarafından kullanılmıştır. SMM, konuşma tanıma uygulamalarında en gelişmiş tekniktir (Juang ve diğ. 1985, Rabiner ve diğ. 1988, Rabiner ve Juang 2005). SMM ayrıca konuşmacı tanıma uygulamalarında da kullanılmaktadır. (Furui ve Matsui 1994, Zhu ve ark. 1994, Yu ve ark. 1995).

SMM sınıflar arası ilişkiyi tanımlamada kullanılan bir modelleme tekniğidir. SMM stokastik bir modeldir ve istatistiksel özellikleri zamanla değişen dizilerin modellenmesinde sıkça kullanılmaktadır. SMM'de durumlar doğrudan gözlenemez. Bir

durumda hangi sembolün/gözlemin üretileceği başka bir olasılık ifadesi ile belirlenir. Bu modelde bir takım eniyileme tekniklerinin gözlem dizilerine uygulanması sonucu en yüksek olasılıklı durum dizisi elde edilir.

Konuşma sinyali gerçekte istatistiksel özellikleri zamanla değişen bir sinyaldir. Konuştuğumuz zaman ses organlarımız hava basıncını ve hava akışını duyulabilecek ses dizileri üretecek şekilde modüle eder. Ses modelleme belirli seslerin kısa dönem spektral özelliklerinin analizini içerir ve bu modelleme farklı seslere karşılık gelen ses organ yapısının uzun dönem değişimini tanımlamamızı sağlar.

SMM basitçe gözlem dizisi $(x_1, x_2, ..., x_N)$ üreten ve sonlu sayıda durumdan oluşan bir rastsal süreçtir. Bu nedenle SMM belirli sayıda durumdan ve gözlem dizisinden oluşmaktadır. Gözlem dizisi bir durumdan diğer bir duruma geçişin bir sonucudur. Şekil 2.11'de 3 durumlu bir SMM gösterilmektedir.



Şekil 2.11. 3 durumlu Markov modeli

Bu şekildeki bir Markov model ergodik model olarak ifade edilmektedir. Ergodik modelin anlamı her durumdan her duruma geçişin olmasıdır. Şekilde gösterilen modelde a_{ij} 'ler durumlar arası geçiş olasılıklarını belirtmektedir. Matematiksel olarak durumlar arası geçiş olasılıkları şu şekilde ifade edilmektedir.

$$a_{ij} = P[q_t = S_j | q_{t-1} = S_i] \qquad 1 \le i, j \le N \tag{2.17}$$

Durum geçiş olasılıkları aşağıdaki şartı sağlamalıdır.

$$a_{ij} \ge 0 \tag{2.18a}$$

$$\sum_{j=1}^{N} a_{ij} = 1 \tag{2.18b}$$

Denklem (2.17) 'de *N*, durum sayısını göstermektedir. Durum geçiş olasılıkları durum geçiş matrisi ile ifade edilmektedir. Şekil 2.11 deki 3 durumlu Markov modeli için durum geçiş matrisi şu şekildedir:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

Bir SMM aşağıdaki parametrelerden oluşmaktadır:

- Modelde kullanılan durum sayısı *N*.
- Gözlem sembol sayısı *M*.
- Durumlar arası geçiş olasılık matrisi *A*.
- Gözlem sembol olasılık matrisi *B*.
- Başlangıç durum olasılıkları π.

Tüm bu parametreler yardımıyla bir SMM modeli şu şekilde belirtilmektedir:

$$\lambda = (A, B, \pi).$$

Bu sayede modelin tüm parametreleri belirtilmiş olur. Durum geçiş olasılık matrisi A ve başlangıç durum olasılıkları $\pi = [\pi_1, \pi_2, ..., \pi_N]$ şeklinde verilen bir sistemde her hangi bir $q = [q_0, q_1, ..., q_T]$ durum dizisinin bu model tarafından üretilme olasılığı aşağıdaki ifadeden elde edilir:

$$P(q | A, \pi) = \pi_{q0} a_{q0q1} a_{q1q2} ... a_{qT-1qT} \quad . \tag{2.19}$$

Durum dizisinin gözlenmediğini ve bunun yerine q_t durumlarında üretilen her bir O_{t_1} gözleminin (özellik vektörü) bilindiğini kabul edersek mümkün olan her S_i *(i=1,2,…N)* durumunda O_t gözleminin üretilme olasılığı gözlem olasılık matrisi $B = \{b_i(O_t)\}_{i=1}^{N}$ ile tanımlanır. Burada $b_i(O_t)$, S_i durumunda O_t gözleminin (özellik vektörü) üretilme olasılığıdır ve şu şekilde ifade edilir (Rabiner ve Juang 1993).

$$b_i(O_t) = P(O_t | q_t = S_i) \tag{2.20}$$

Her hangi bir $O = (O_1, O_2, ..., O_T)$ gözlem dizisini üreten *q* durum dizisi bilindiğinde sistem tarafından bu gözlem dizisinin üretilme olasılığı

$$P(O|q, B) = b_{q1}(O_1) b_{q2}(O_2) ... b_{qT}(O_T) \tag{2.21}$$

şeklinde olacaktır. Gözlem dizisi O ve durum dizisi *q* 'nun sistem tarafından üretilme olasılığı

$$P(O, q|\pi, A, B) = \pi_{q0} \prod_{t=1}^{T} a_{qt-1qt} b_{qt}(O_t) \tag{2.22}$$

olacaktır. O gözlem dizisinin verilen model tarafından üretilme olasılığı ise

$$P(O|\lambda) = \sum_{q} P(O, q|\lambda) \tag{2.23}$$

olur. SMM'de gözlem sembolleri sürekli veya ayrık olabilir. Sürekli olması durumunda gözlem sembol olasılıkları sürekli bir olasılık dağılım işlevi ile ifade edilir. Bu durumda model Sürekli Saklı Markov Model adını alır.

2.3.3 Saklı Markov Modelde Üç Temel Problem ve Çözümü

SMM'nin geliştirilmesinde çözülmesi gereken üç temel problem bulunmaktadır (Rabiner 1989). Bunlar:

1. Verilen bir O gözlem dizisi ve λ modeli için, bu gözlem dizisinin model tarafından üretilme olasılığı $P(O|\lambda)$ etkin bir şekilde nasıl hesaplanır?
2. Verilen bir O gözlem dizisi ve λ modeli için en yüksek olasılığı verecek q durum dizisi nasıl bulunur?
3. $P(O|\lambda)$ olasılığını maksimum yapacak model parametreleri nasıl ayarlanır?

Bu üç problem sırasıyla değerlendirme (evaluation), model yapısını öğrenme (learn structure) problemi ve tahmin (estimation) problemi olarak adlandırılmaktadır.

2.3.3.1. Değerlendirme Problemi

Olasılık hesabının nasıl yapılacağı ile ilgilenmektedir. Denklem (2.23) kullanılarak $P(O|\lambda)$'yı hesaplamak mümkündür ancak bunun için gereken işlem sayısı $2T.N^T$ kadardır. Bu durum sayısı ve gözlem sembol sayısının yüksek olduğu sistemler için oldukça karmaşık bir hal almaktadır. Bu nedenle olasılık hesabındaki işlem sayısının azaltmak için ileri-geri işlemi kullanılmaktadır. İleri-geri işlemi ile $P(O|\lambda)$ olasılığı hesaplanırken $N^2.T$ kadar işlem yapmak yeterli olmaktadır (Rabiner ve Juang 1993). İleri-geri işlemi sayesinde olasılık hesabı etkin bir şekilde yapılabilmektedir (Bkz. Ek A).

2.3.3.2. Tahmin Problemi

Verilen bir gözlem dizisi için tahmin probleminin çözümü ile bu diziyi en yüksek olasılıkla üretecek model parametreleri bulunmuş olur. Konuşmacı tanıma sisteminde bu işlem eğitim aşamasına karşılık gelmektedir. Model parametrelerinin elde edilmesinde kullanılan gözlem dizisi eğitim dizisi olarak adlandırılır ve konuşmacı tanımada bu dizi özellik vektörüdür.

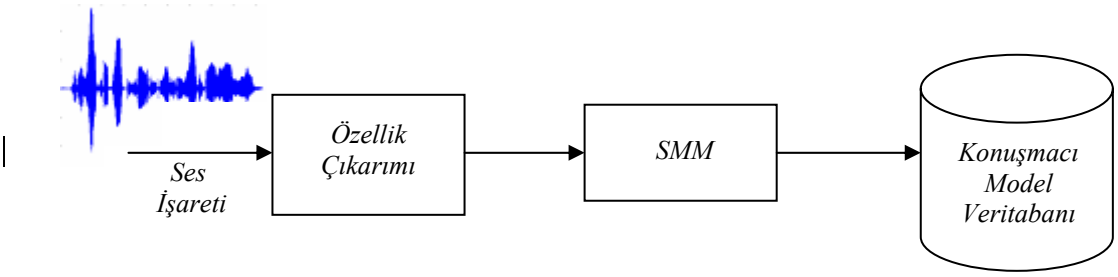
SMM 'de tahmin probleminin çözümü için Baum-Welch yöntemi kullanılmaktadır (Rabiner 1989). Problemin çözümünde en büyük olabilirlik (Maximum Likelihood) yöntemi kullanılır.

2.3.3.3. Model Yapısını Öğrenme Problemi

O gözlem dizisini üreten en yüksek olasılıklı durum dizisinin ne olduğu ile ilgilenir. Bu problemin çözümünden $P(q|O,\lambda)$ olasılığını en büyük yapan q dizisi elde edilir. $P(q|O,\lambda)$ olasılığını en büyük yapan dizi Viterbi algoritması ile elde edilir (Bkz. Ek B).

2.3.4. Saklı Markov Modellerinin Konuşmacı Tanımaya Uygulanması

Saklı Markov Modelleri ile konuşmacı tanıma sisteminin eğitim aşaması şekil 2.12'de gösterilmektedir. Şekil 2.12'den de görüldüğü gibi eğitim aşamasında ilk olarak eğitilecek ses işaretinden özellik vektörleri elde edilir. Elde edilen özellik vektörleri SMM'de anlatılan gözlem dizilerine karşılık gelmektedir. Özellik vektörleri ile SMM'in üç temel probleminden ikincisi olan tahmin probleminin çözümü gerçekleştirilir. Bu işlem Baum-Welch algoritması ile yapılır ve bu işlem sonucunda SMM parametreleri (A,π,B) ayarlanmış olur. Böylece ilgili konuşmacıya ait SMM modeli elde edilmiş olur ve bu model veritabanına kaydedilir. Tüm eğitim konuşmacıları için bu adımlar tekrarlanır ve böylece eğitim aşaması tamamlanmış olur.



Şekil 2.12 SMM ile konuşmacı tanıma sisteminin eğitim aşaması

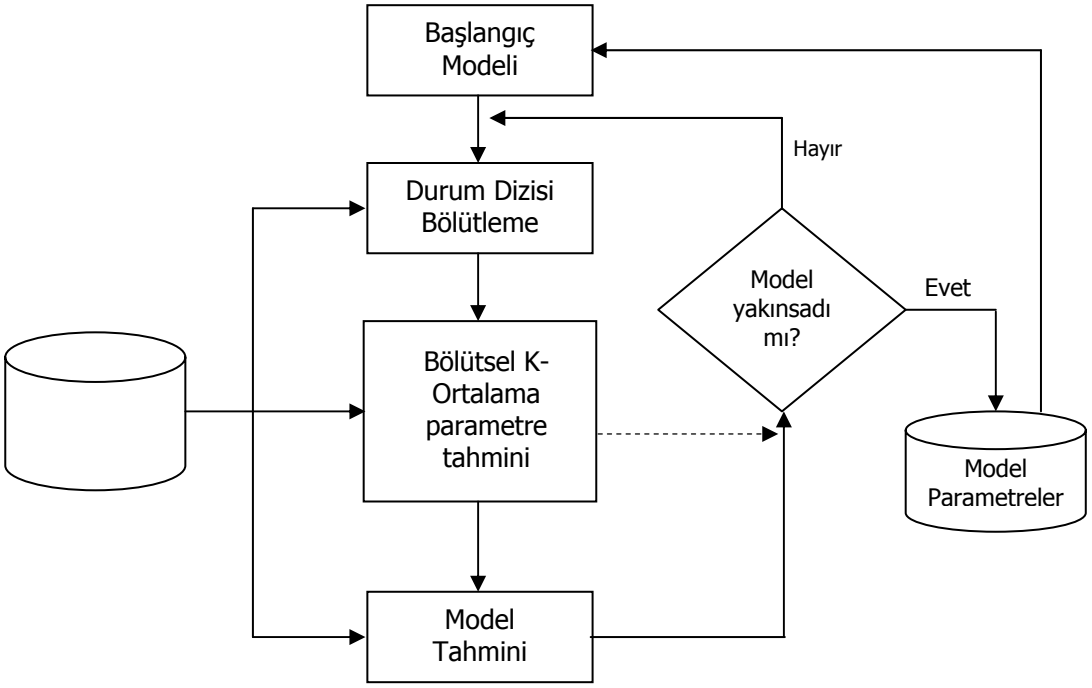
Saklı Markov Modellerinde tahmin probleminin çözümü için çeşitli özyineleme işlemleri kullanılmaktadır. Baum-Welch algoritması $P(O|\lambda)$ olasılığının λ modeli üzerinden maksimum yapılması probleminin çözümüdür. Verilen bir O gözlem dizisi için $P(O|\lambda)$ olasılığını maksimum yapacak farklı eniyileme ölçütleri kullanılabilir. SMM'de $b_j(o_t)$ başlangıç olasılık yoğunluklarının iyi bir şekilde hesaplanması, tahmin probleminin çözümünde parametre değerlerinin tahmin edilmesini ve yakınsamasını

hızlandıracaktır. Bundan dolayı sembol olasılıklarının başlangıç değerlerinin iyi bir şekilde tahmin edilmesi için bölütsel k-ortalama algoritması kullanılmaktadır (Juang ve Rabiner 1990).

Parametre değerlerinin tahmin edilmesi için de kullanacağımız eğitim gözlem sembollerinin (özellik vektörleri) ve bütün parametrelerinin başlangıç değerlerine de sahip olduğumuzu varsayalım. Model başlangıcında, gözlem vektörleri ilgili λ modeline göre bölütlenerek durumlara dağıtılır. Bu bölütleme Viterbi algoritması ile optimum durum dizisi bulunarak yapılır. N adet durumdan her biri için yapılan bu bölütlemenin sonucu, λ modeli kullanılarak her bir j durumu boyunca gözlem vektörlerinin maksimum oluşma olasılığıdır. Sürekli gözlem yoğunlukları kullandığımız için, gözlem vektörlerinin her bir j durumu boyunca M adet kümeye bölünmesi işlemi öklit mesafesi ölçütüne göre yapılır. Bölünen her bir parça (toplam M adet) $b_j(o_t)$ yoğunluğunun M adet karışımından bir tanesini temsil eder. Bu bölme işlemi sonucunda aşağıdaki model parametrelerinin tahmin değerleri elde edilir:

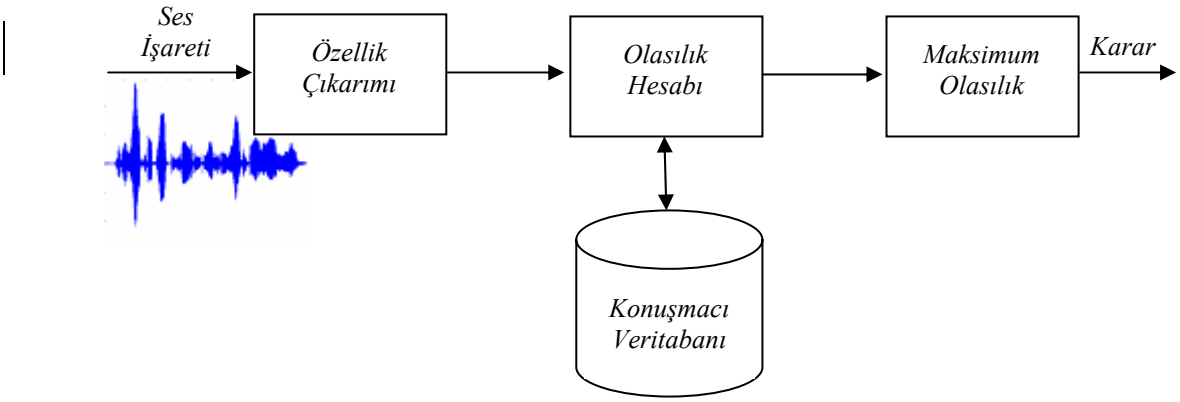
- c_{jm} = j durumunda m kümesine atanan vektör sayısı,
- μ_{jm} = j durumunda m kümesine atanan vektörlerin ortalaması,
- U_{jm} = j durumunda m kümesine atanan vektörlerin ortak değişinti matrisi.

Tahmin edilen model parametre değerleri yeni modelin, $\overline{\lambda}$, yakınsamasına bakılabilir. Bunun yanında tahmin edilen parametreler Baum-Welch'de kullanılarak yeni model parametreleri de tahmin edilebilir. Tahmin edilen model bir önceki modele yakınsadığında eğitime son verilir ve model parametreleri kaydedilir. . Bölütsel K-ortalama algoritması Şekil 2.13' de gösterilmektedir.



Şekil 2.13. Bölütsel K-ortalama algoritması akış diyagramı (Rabiner ve Juang 1993)

Test aşamasında bilinmeyen konuşmacıya ait ses işaretinden özellik vektörleri elde edilir. Elde edilen özellik vektörleri SMM'de gözlem dizisine karşılık gelmektedir. Veritabanında kayıtlı bulunan tüm modeller için Viterbi algoritması kullanılarak en yüksek olasılıklı durum dizisi ve gözlemlerin bu dizinin üzerinde üretilme olasılıkları elde edilir. Elde edilen olasılıklardan maksimum olanı hangi model tarafından elde edilmiş ise bilinmeyen konuşmacı o modele atanır. Bu adımlar Şekil 2.14 de gösterilmektedir.



Şekil 2.14. SMM ile konuşmacı tanıma sisteminin test aşaması

SMM'de gözlem sembollerinin sürekli olması halinde gözlem sembol olasılıkları sürekli bir olasılık yoğunluk fonksiyonu tarafından elde edilmelidir. Sürekli gözlem sembolleri kullanılması halinde gözlem sembol olasılıkları şu şekilde ifade edilmektedir:

$$b_j(O) = \sum_{m=1}^{M} c_{jm} N(O, \mu_{jm}, U_{jm}), \quad 1 \le j \le N \quad . \tag{2.24}$$

Denklem (2.24) de O modellenecek olan gözlem vektörünü; c_{jm}, *j.* durum ve *m.* karışım için karışım katsayısını; *N,* ortalaması μ_{jm} ve ortak değişinti matrisi U_{jm} olan Gauss olasılık yoğunluk fonksiyonunu belirtmektedir. Karışım katsayıları aşağıdaki şartı sağlamalıdır (Rabiner 1989):

$$\sum_{m=1}^{M} c_{jm} = 1, \quad 1 \le j \le N \tag{2.25a}$$

$$c_{jm} \ge 0, \quad 1 \le j \le N, \quad 1 \le m \le M \quad . \tag{2.25b}$$

Konuşmacı tanıma sisteminde gözlemler (özellik vektörleri) sürekli olduğundan dolayı gözlem sembol olasılıkları Denklem (2.24)'de verildiği şekilde hesaplanmaktadır. Bu şekilde kullanılan SMM'ye sürekli Saklı Markov Modelleri denir. Özellik vektörleri vektör nicemleme gibi yöntemlerle ayrık hale getirilebilir. Bu durumda ise model Ayrık Saklı Markov Modelleri adını alır.

2.3.5. Vektör Nicemleme

Metinden bağımsız konuşmacı tanıma sisteminde kullanılabilecek bir diğer yol ise bütün özellik vektörlerinin kullanılmasıdır. Ancak bu durum özellik vektörü boyutunun ve konuşmacı sayısının yüksek olduğu durumlarda pratik bir çözüm değildir. Bu nedenle özellik vektör boyutunun azaltılabileceği (sıkıştırılabileceği) yöntemler geliştirilmiştir. Vektör nicemleme algoritması ile eğitim vektörleri sıkıştırılarak, az sayıda fakat yüksek ifade etme gücüne sahip vektörler elde edilebilir.

Literatürde yüksek boyutlu verilerin sıkıştırılarak daha az sayıda elemanla ifade edilebilmesini sağlayan birçok sıkıştırma algoritması mevcuttur (Kinnunen ve diğ.

2000). Bu tezde en çok kullanılan iki yöntem olan LBG (Linde ve diğ. 1980) ve K-ortalama algoritmaları incelenmiştir.

2.3.5.1 LBG Algoritması

Vektör nicemleme algoritmasının matematiksel ifadesi Linde ve arkadaşları tarafından detaylı bir şekilde anlatılmıştır (Linde ve diğ. 1980). Algoritmanın işleyiş şekli Şekil 4'de gösterilmiştir. Bir konuşmacıya ait *l* adet eğitim vektörü, $(x_1, x_2, ..., x_l)$, verilmiş olsun. Bu vektörlerin iki boyutlu uzayda gösterimi Şekil 2.15 (a)' da verilmiştir. Bu eğitim vektörlerini N_c adet kod vektörden oluşan kod kitabı ile ifade etmek istediğimizi düşünelim $(Nc < l)$. Bu işlem şu şekilde yapılır:

İlk olarak özellik vektörleri, sınırları Şekil 2.15 (b)' de gösterildiği gibi tek bir grup olarak alınır. Bu grup içerisindeki vektör elemanlarının tamamı bu vektörün merkezi C_p (ortalama değer) ile ifade edilir (Şekil 2.15 (c)).

$$C_p = \frac{\sum_{i=1}^{l} x_i}{l} \tag{2.26}$$

Merkez, aşağıdaki kurala göre iki yeni kod vektöre bölünür ($C_{s1}\, ve\, C_{s2}$) (Şekil 2.15 (d)).

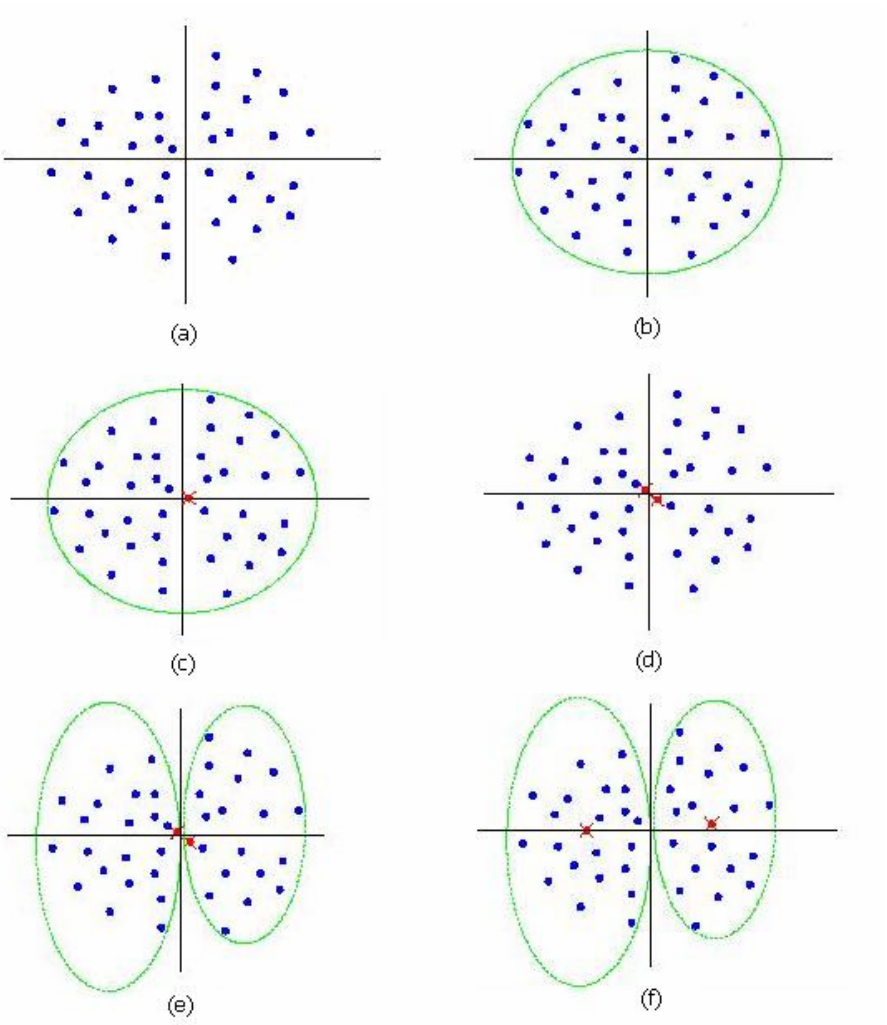
$$C_{s1} = (1-\varepsilon)C_p$$
$$C_{s2} = (1+\varepsilon)C_p$$

Buradaki ε değeri, bölme parametresi olarak adlandırılır ve genellikle 0,001 ile 0,0001 arasındadır. Özellik vektörleri öklit mesafesi ölçütü kullanılarak bu yeni iki kod vektöre yakınlıklarına göre iki grup olacak şekilde gruplandırılır. Bu iki grubun merkezleri (ortalama değerleri) hesaplanır ve yeni kod vektörleri olarak atanır (Şekil 2.15 (f)). Bu gruplandırma ve merkez hesaplama işlemi merkez değerler ile özellik vektörü arasındaki ortalama bozulma miktarı minimum olana kadar bu şekilde devam eder (Şekil 2.15 (e)). $d(x_i, C_j)$ *j.* kod vektörü C_j ile *i.* özellik vektörü x_i arasındaki

öklit mesafesini belirtsin. Bu durumda / adet özellik vektörünün ortalama bozulma miktarı şu şekilde ifade edilmektedir.

$$D = \frac{1}{l}\sum_{i=1}^{l} \min_{1 \le j \le M} d(x_i, C_j) \tag{2.27}$$

Bu iki kod vektörü yine aynı gruplandırma ve merkez hesaplama işlemi ile dört kod vektöre çıkarılır. Bu işlem istenen sayıda kod vektör, N_c, elde edilinceye kadar devam eder.



Şekil 2.15. Vektör Nicemleme algoritmasının işleyiş adımları

Eğitim aşamasında her konuşmacı için N_c adet kod vektörü içeren kod kitabı oluşturulur ve veritabanında bu kod kitapları saklanır. Test aşamasında ise bilinmeyen konuşmacının test cümlesine ait özellik vektörleri $y_1, y_2, \ldots, y_L$, ile veritabanında kayıtlı

bulunan bütün konuşmacıları kod kitapları ile tek tek uzaklık ölçümü yapılır. i. konuşmacıya ait toplam bozulma şu şekilde hesaplanır.

$$D^i = \sum_{l=1}^{L} \min_{1 \le j \le N_c} d(y_l, C_j^i) \tag{2.28}$$

(2.28) denklemindeki C_j^i *i.* konuşmacıya ait kod kitabının *j.* kod vektörünü belirtmektedir.

Veritabanındaki bütün konuşmacılarla bilinmeyen konuşmacıya ait test cümlesi arasındaki uzaklık ölçümü hesaplandıktan sonra sistem test cümlesinin en az bozulum sağlayan kod kitabının temsil ettiği kişiye ait olduğuna karar verir.

$$i^* = \arg\min_{1 \le i \le Ns} D^i \tag{2.29}$$

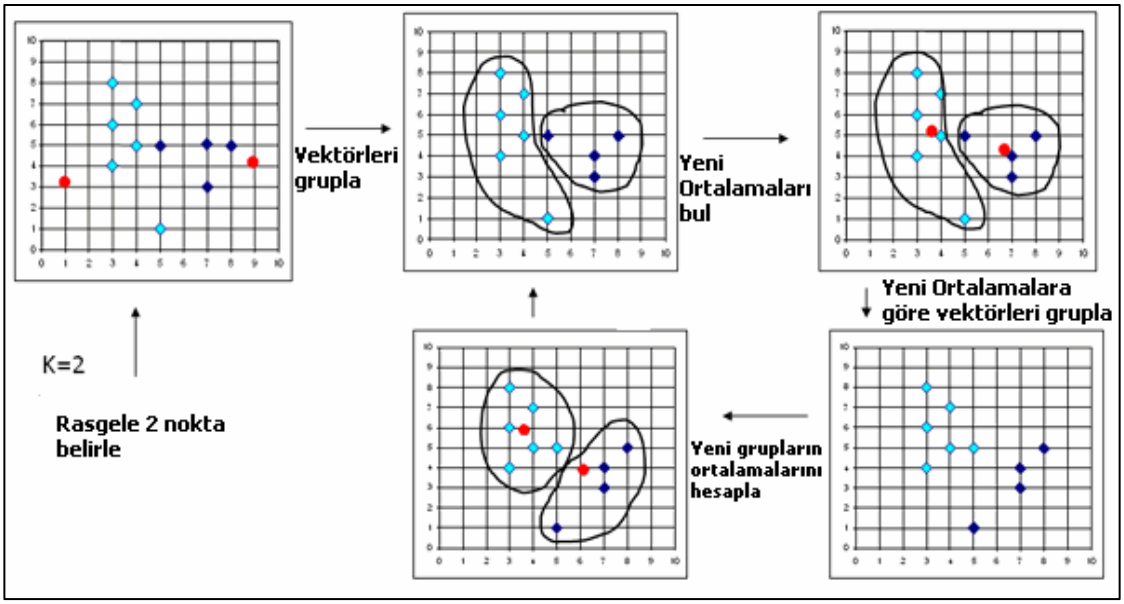
(2.29) denklemindeki *Ns,* veritabanında kayıtlı bulunan konuşmacı sayısını belirtmektedir.

2.3.5.2 K-ortalama Algoritması

K-ortalama algoritması LBG algoritmasından farklı olarak arzu edilen sayıda kod kitabını başlangıçta rasgele olarak belirler ve kod kitaplarını güncelleyerek sıkıştırma işlemi yapar. K-ortalama algoritmasının işleyiş adımları şu şekildedir:

- *Adım 1:* ragele olarak *K* adet nokta belirle (*K* arzu edilen kod kitabı boyutu)
- *Adım 2:* K adet noktanın her birine en yakın olan vektörleri grupla
- *Adım 3:* Her vektör grubu için kod vektörleri güncelle
- *Adım 4:* Ortalama bozulma miktarı belirli bir eşik değerin altına düşene kadar adım 2 ve adım 3' ü tekrarla.

K-ortalama algoritmasının 4. adımında belirtilen bozulma miktarının hesaplanması aynen LBG algoritmasında olduğu gibi Denklem (2.28) ile yapılır. K=2 için k-ortalama algoritmasının işleyişi Şekil 2.16' da verilmiştir.



Şekil 2.16 K-ortalama algoritmasının işleyiş adımları (K=2 için)

Deneysel çalışmada denklem (2.28)'de belirtilen ve hem LBG hem de K-ortalama algoritmalarının geliştirilirken bozulma miktarının hesaplanması sırasında değişik uzaklık ölçütleri kullanılmıştır. İki vektör arasındaki uzaklığın hesaplanması sırasında kullanılan ölçütler aşağıda belirtilmiştir.

- Öklit Mesafesi:

$$d(x,y) = \sqrt{\sum_{i=1}^{L}(x_i - y_i)^2} \tag{2.30}$$

- Manhattan Mesafesi:

$$d(x,y) = \sum_{i=1}^{L}\left|(x_i - y_i)\right| \tag{2.31}$$

- Chebychev Mesafesi

$$d(x,y) = \max_{i=1}^{L}\left|x_i - y_i\right| \tag{2.32}$$

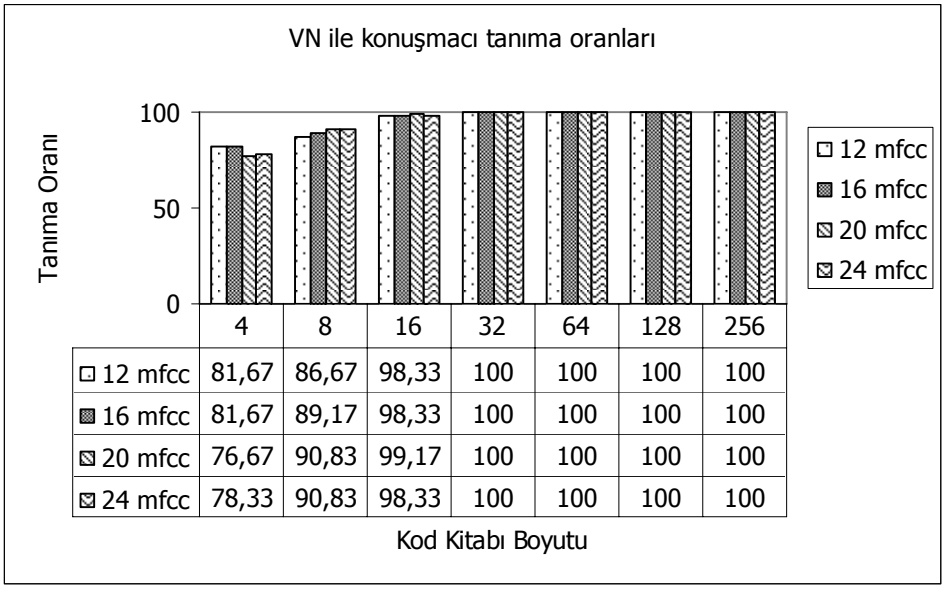
Denklem (2.30) – (2.32) de *L*, vektörlerin boyutunu belirtmektedir.

3. DENEYSEL SONUÇLAR VE TARTIŞMA

Deneysel çalışmanın ilk aşamasında hem ergodik SMM hem de VN için optimum sistem parametrelerinin belirlenmesine ilişkin deneyler yapılmıştır. Bu amaçla TIMIT veritabanının test dizininden 20' si erkek ve 20' si bayan olmak üzere toplam 40 kişilik bir konuşmacı kümesi seçilmiştir. Her konuşmacının 10 cümlesinden 7 tanesi eğitim, kalan 3 cümlesi ise test aşamasında (ayrı ayrı) kullanılmış olup, 40 kişi için toplam 120 test yapılmıştır. İlk aşamada SMM için optimum durum ve karışım sayılarının, VN için ise optimum kod kitabı boyutunun belirlenmesi amaçlanmıştır. Ayrıca kullanılan özellik vektörü boyutunun değişiminin konuşmacı tanıma performansına etkileri de incelenmiştir.

3.1. Vektör Nicemleme ile Deneysel Sonuçlar

Bu bölümde yapılan deneylerde eğitim algoritması olarak LBG, uzaklık ölçütü olarak da öklit mesafesi kullanılmış olup diğer eğitim algoritması ve mesafe ölçütleri ile elde edilen sonuçlar Bölüm 3.2' de verilmiştir.

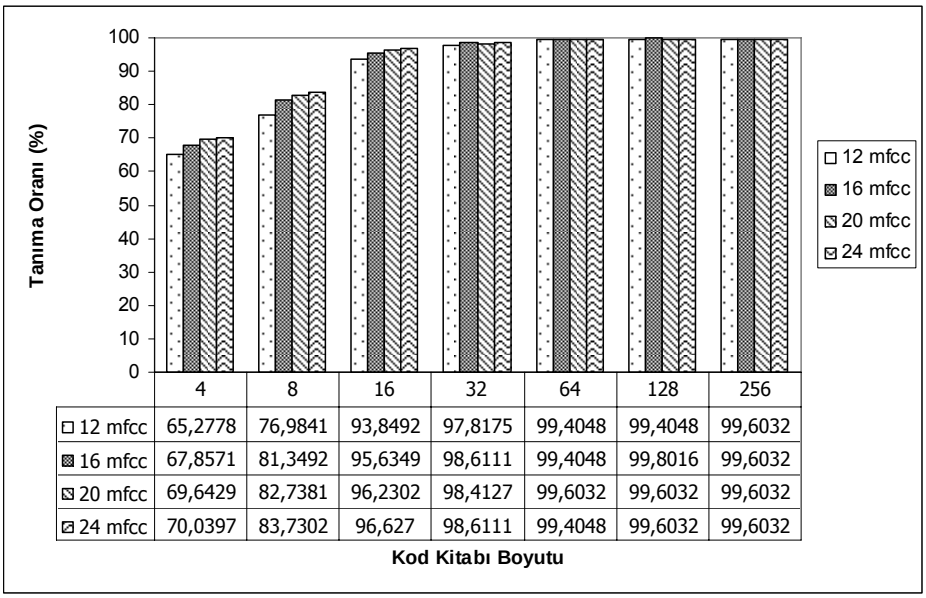


	4	8	16	32	64	128	256
12 mfcc	81,67	86,67	98,33	100	100	100	100
16 mfcc	81,67	89,17	98,33	100	100	100	100
20 mfcc	76,67	90,83	99,17	100	100	100	100
24 mfcc	78,33	90,83	98,33	100	100	100	100

Şekil 3.1. 40 kişi için VN ile konuşmacı tanıma oranlarının kod kitabı ve özellik vektörü boyutuna göre değişimi

Şekil 3.1 değişik kod kitabı ve özellik vektörü boyutları için, 40 kişilik konuşmacı kümesi kullanılarak VN ile elde edilen konuşmacı tanıma oranlarını(%) göstermektedir.

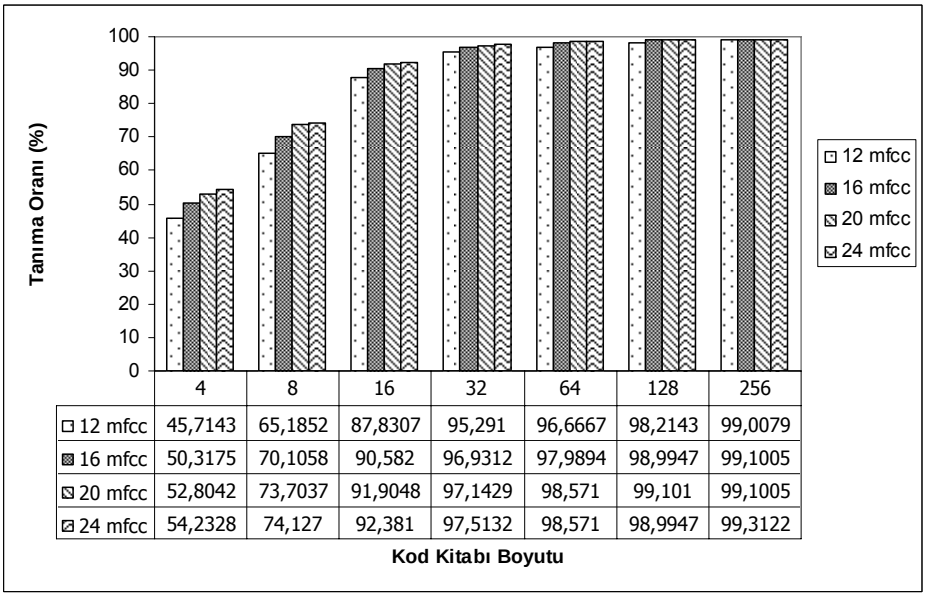
Konuşmacı sayısının 168 kişiye çıkarılması durumunda elde edilen tanıma oranları Şekil 3.2'de gösterilmektedir.



	4	8	16	32	64	128	256
12 mfcc	65,2778	76,9841	93,8492	97,8175	99,4048	99,4048	99,6032
16 mfcc	67,8571	81,3492	95,6349	98,6111	99,4048	99,8016	99,6032
20 mfcc	69,6429	82,7381	96,2302	98,4127	99,6032	99,6032	99,6032
24 mfcc	70,0397	83,7302	96,627	98,6111	99,4048	99,6032	99,6032

Şekil 3.2. 168 kişi için VN ile konuşmacı tanıma oranlarının kod kitabı ve özellik vektörü boyutuna göre değişimi

630 kişi kullanıldığında elde edilen tanıma oranları ise Şekil 3.3. de verilmektedir.



	4	8	16	32	64	128	256
12 mfcc	45,7143	65,1852	87,8307	95,291	96,6667	98,2143	99,0079
16 mfcc	50,3175	70,1058	90,582	96,9312	97,9894	98,9947	99,1005
20 mfcc	52,8042	73,7037	91,9048	97,1429	98,571	99,101	99,1005
24 mfcc	54,2328	74,127	92,381	97,5132	98,571	98,9947	99,3122

Şekil 3.3. 630 kişi için VN ile konuşmacı tanıma oranlarının kod kitabı ve özellik vektörü boyutuna göre değişimi konuşmacı kümesi için tanıma oranları

Şekil 3.1 - Şekil 3.3 den görüldüğü gibi, 40 kişi ile yapılan deneylerde 32 ve üzerindeki kod kitabı boyutlarında özellik vektörü boyutundan bağımsız olarak en iyi sonuçlar elde edilmiştir. 40 kişilik konuşmacı grubu için kod kitabının boyutunun 4 olduğu durum hariç olmak üzere diğer tüm kod kitabı boyutları için 20 mfcc kullanmak daha iyi tanıma oranları vermektedir. Kişi sayısı 168 kişiye çıkarıldığında beklendiği gibi tanıma oranları düşmektedir. 168 kişilik grup için en iyi tanıma oranı 128 boyutlu kod kitabı ile 16 mfcc katsayısı kullanıldığında elde edilmiştir (%99.80). 40 kişilik grup için 12 adet mfcc kullanarak 32 kod kitabı boyutunda dahi %100 tanıma oranı elde edilirken, bu parametrelerle 168 kişilik veri kümesinde tanıma oranı %97.81'e düşmüştür. En düşük tanıma oranları 12 mfcc kullanıldığında elde edilmiştir. Yapılan deneylerden de görüldüğü gibi kişi sayısı arttıkça, sistemin kişileri ayırt edebilmesi için gereken optimum sistem parametreleri değişmektedir. 630 kişi kullanılarak yapılan deneylerde bu durum açıkça görülmektedir. 168 kişi için en iyi başarımı 16 mfcc ve 128 boyutlu kod kitabı sağlarken, 630 kişi için en iyi başarım 24 mfcc ve 256 kod kitabı için elde edilmiştir.

3.2. VN ile Farklı Eğitim Algoritmaları ve Uzaklık Ölçütlerinin Karşılaştırılması

VN algoritması ile yapılan son testlerde en çok kullanılan iki yöntem olan LBG ve K-ortalama algoritmaları ile farklı uzaklık ölçütlerinin karşılaştırılması amaçlanmıştır. Çizelge 3.1 TIMIT veritabanının 168 kişilik test dizini kullanılarak değişik uzaklık ölçütleri için LBG algoritması ile elde edilen tanıma oranlarını göstermektedir.

Çizelge 3.1 TIMIT 168 kişi için LBG algoritmasıyla farklı uzaklık ölçütleri ile elde edilen tanıma oranları (%)

	Tanıma Oranı (%)		
Kod Kitabı	**Öklit**	**Manhattan**	**Chebychev**
8	83.73	86.70	59.33
16	96.62	94.04	87.89
32	98.61	98.41	95.43
64	99.40	99.40	98.41
128	99.60	99.40	99.00
256	99.60	99.80	99.60

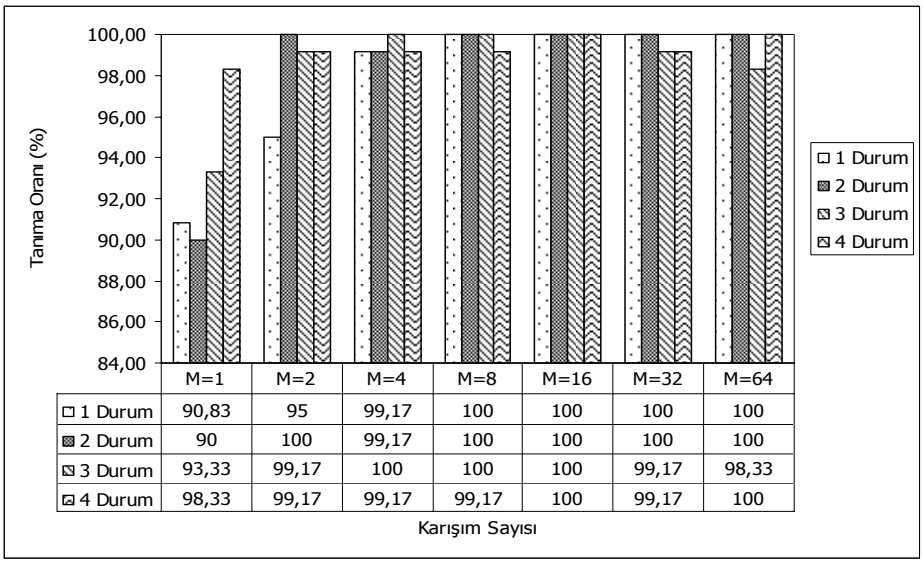
Aynı şartlarda K-ortalama algoritması kullanılarak farklı uzaklık ölçütleri için elde edilen tanıma oranları Çizelge 3.2 ' de verilmektedir.

Çizelge 3.2 TIMIT 168 kişi için K-ortalama algoritmasıyla farklı uzaklık ölçütleri ile elde edilen tanıma oranları (%)

	Tanıma Oranı (%)		
Kod Kitabı	**Öklit**	**Manhattan**	**Chebychev**
8	74.8	85.16	48.77
16	91.35	96.27	75.12
32	98.21	98.49	91.15
64	99.29	99.4	96.79
128	99.56	99.64	98.37
256	99.52	99.76	99.05

Çizelge 3.1 ve 3.2 den de görüldüğü gibi LBG algoritması, K-ortalama algoritmasına göre daha iyi tanıma oranı vermektedir. Her iki algoritma içinde Manhattan uzaklık ölçütü çoğu zaman daha yüksek başarım vermekte olup, aynı zamanda Manhattan ölçütü ile iki vektör arasındaki uzaklık hesaplanırken öklit ölçütünden daha az işlem yapılmaktadır. Yine her iki nicemleme yöntemi için Chebychev mesafesi en düşük başarımları vermektedir.

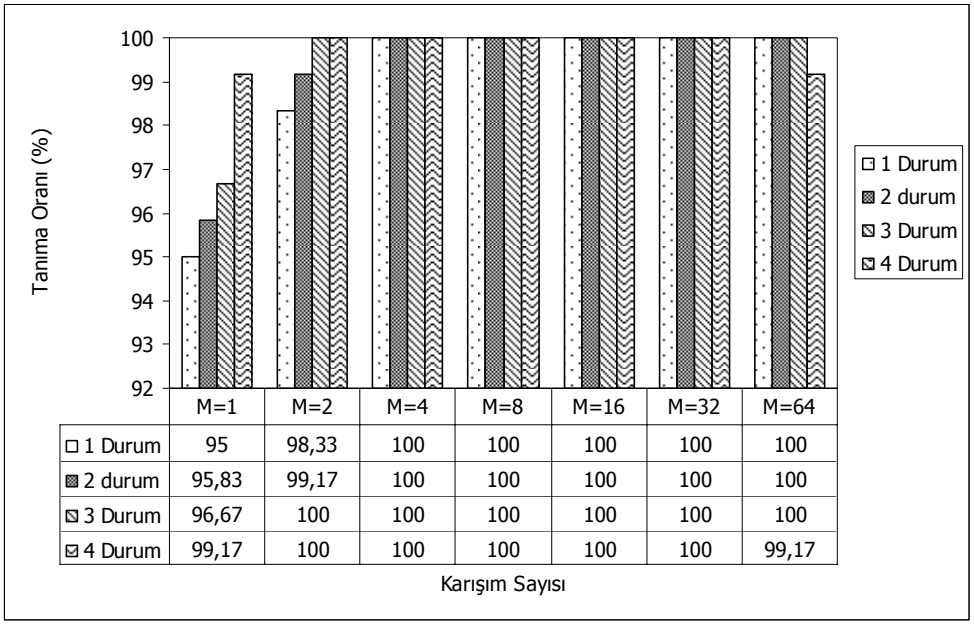
3.3 Saklı Markov Modelleri ile Deneysel Sonuçlar



	M=1	M=2	M=4	M=8	M=16	M=32	M=64
1 Durum	90,83	95	99,17	100	100	100	100
2 Durum	90	100	99,17	100	100	100	100
3 Durum	93,33	99,17	100	100	100	99,17	98,33
4 Durum	98,33	99,17	99,17	99,17	100	99,17	100

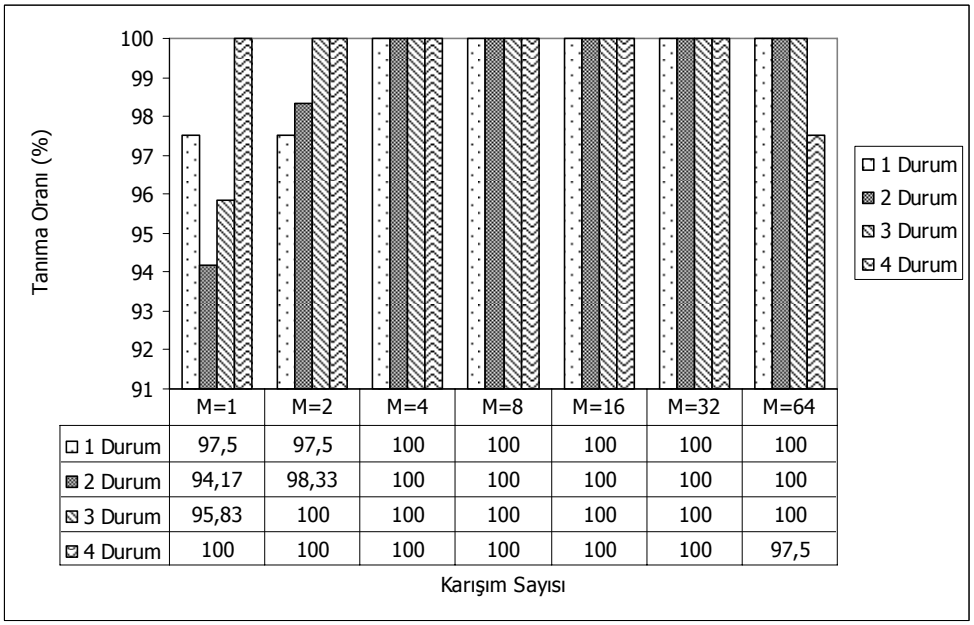
Şekil 3.4. SMM ile 12 mfcc kullanıldığında konuşmacı tanıma oranlarının durum ve karışım sayısına göre değişimi

İlk olarak, aynı 40 kişilik grup için SMM ile konuşmacı tanıma oranlarının, durum ve karışım sayısına göre değişimi incelenmiştir. Özellik vektörü olarak, 12, 16, 20 ve 24 tane mfcc kullanıldığı durumlarda elde edilen sonuçlar, sırası ile Şekil 3.4-Şekil 3.7 de verilmiştir.



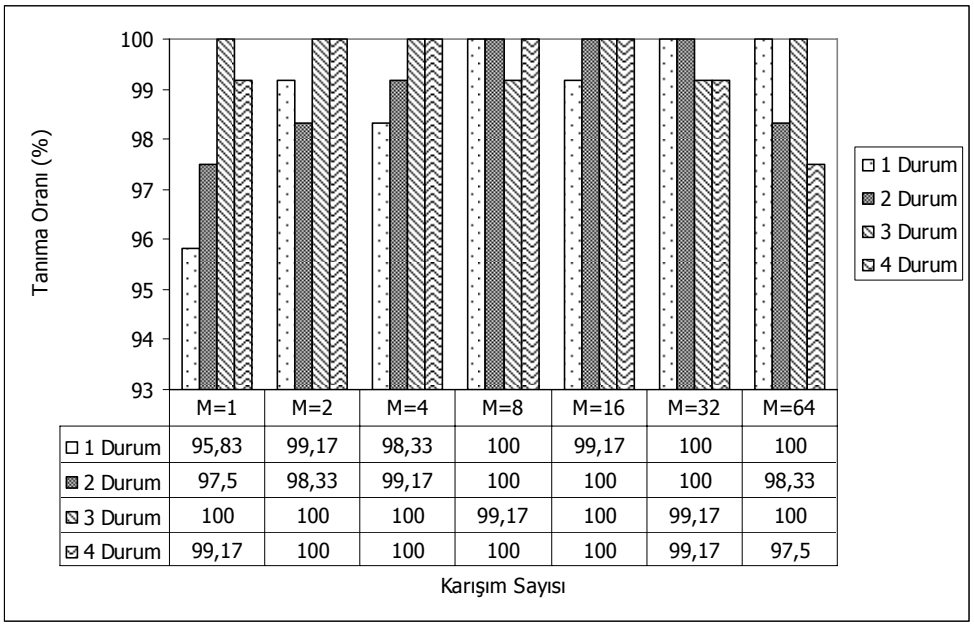
	M=1	M=2	M=4	M=8	M=16	M=32	M=64
1 Durum	95	98,33	100	100	100	100	100
2 durum	95,83	99,17	100	100	100	100	100
3 Durum	96,67	100	100	100	100	100	100
4 Durum	99,17	100	100	100	100	100	99,17

Şekil 3.5. SMM ile 16 mfcc kullanıldığında konuşmacı tanıma oranlarının durum ve karışım sayısına göre değişimi



	M=1	M=2	M=4	M=8	M=16	M=32	M=64
1 Durum	97,5	97,5	100	100	100	100	100
2 Durum	94,17	98,33	100	100	100	100	100
3 Durum	95,83	100	100	100	100	100	100
4 Durum	100	100	100	100	100	100	97,5

Şekil 3.6. SMM ile 20 mfcc kullanıldığında konuşmacı tanıma oranlarının durum ve karışım sayısına göre değişimi

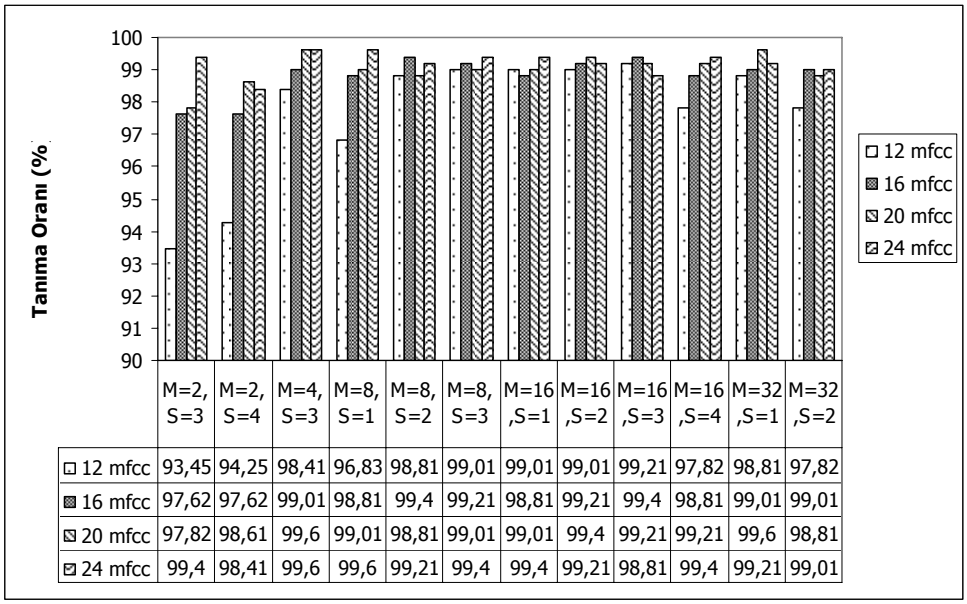


	M=1	M=2	M=4	M=8	M=16	M=32	M=64
1 Durum	95,83	99,17	98,33	100	99,17	100	100
2 Durum	97,5	98,33	99,17	100	100	100	98,33
3 Durum	100	100	100	99,17	100	99,17	100
4 Durum	99,17	100	100	100	100	99,17	97,5

Şekil 3.7. SMM ile 24 mfcc kullanıldığında konuşmacı tanıma oranlarının durum ve karışım sayısına göre değişimi

Şekil 3.4 - Şekil 3.7 den görüldüğü gibi 40 kişilik konuşmacı grubu için, SMM ile konuşmacı tanıma sisteminde optimum sistem parametrelerinin belirlenmesi zordur. Çünkü kullanılan tüm özellik vektörü boyutu, durum ve karışım sayısı değerleri 40 kişilik konuşmacı kümesi için yüksek tanıma oranları vermektedir. Daha iyi bir seçim yapabilmek için konuşmacı sayısının artırılması gerekmektedir. Ancak bu durumda optimum parametre değerlerinin seçilmesi kolaylaşacaktır.

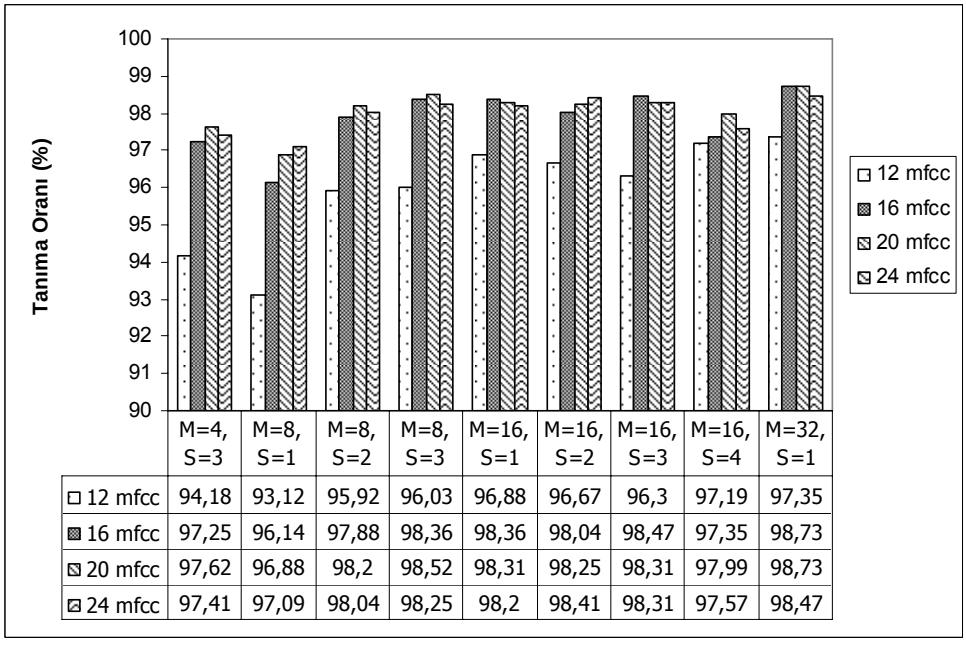
Deneysel çalışmanın ikinci aşamasında 40 kişilik alt grupla yapılan çalışmada beklenen düzeyde başarım gösteremeyen durum ve karışım sayısı kombinasyonları atılarak, istenilen düzeyde başarı gösteren kombinasyonlar 168 kişilik (52 kadın, 116 erkek) konuşmacı kümesine uygulanmıştır. Beklendiği üzere konuşmacı sayısındaki artış tanıma oranını düşürmüştür. Şekil 3.8 168 kişilik grup için SMM ile elde edilen ortalama tanıma oranlarını göstermektedir. Burada, *M* karışım sayısını, *S* ise durum sayısını belirtmektedir.



	M=2, S=3	M=2, S=4	M=4, S=3	M=8, S=1	M=8, S=2	M=8, S=3	M=16 ,S=1	M=16 ,S=2	M=16 ,S=3	M=16 ,S=4	M=32 ,S=1	M=32 ,S=2
12 mfcc	93,45	94,25	98,41	96,83	98,81	99,01	99,01	99,01	99,21	97,82	98,81	97,82
16 mfcc	97,62	97,62	99,01	98,81	99,4	99,21	98,81	99,21	99,4	98,81	99,01	99,01
20 mfcc	97,82	98,61	99,6	99,01	98,81	99,01	99,01	99,4	99,21	99,21	99,6	98,81
24 mfcc	99,4	98,41	99,6	99,6	99,21	99,4	99,4	99,21	98,81	99,4	99,21	99,01

Şekil 3.8. SMM ile 168 kişi için konuşmacı tanıma oranlarının durum, karışım ve mfcc sayısına göre değişimi

168 kişilik ikinci grupla yapılan çalışmada da yüksek başarım gösteremeyen bazı durum ve karışım sayısı kombinasyonları atılarak, istenilen düzeyde başarı gösteren kombinasyonlar 630 kişilik TIMIT veritabanının tamamına uygulanmış ve sonuçlar Şekil 3.9' da verilmiştir.



	M=4, S=3	M=8, S=1	M=8, S=2	M=8, S=3	M=16, S=1	M=16, S=2	M=16, S=3	M=16, S=4	M=32, S=1
12 mfcc	94,18	93,12	95,92	96,03	96,88	96,67	96,3	97,19	97,35
16 mfcc	97,25	96,14	97,88	98,36	98,36	98,04	98,47	97,35	98,73
20 mfcc	97,62	96,88	98,2	98,52	98,31	98,25	98,31	97,99	98,73
24 mfcc	97,41	97,09	98,04	98,25	98,2	98,41	98,31	97,57	98,47

Şekil 3.9. SMM ile 630 kişi için konuşmacı tanıma oranlarının durum, karışım ve mfcc sayısına göre değişimi

SMM ile yapılan çalışmalarda da VN yönteminde olduğu gibi genel olarak en düşük tanıma oranları 12 mfcc kullanıldığında elde edilmiştir. Dolayısıyla, 168 kişilik konuşmacı grubu için 12 mfcc kullanmanın yeterli olmadığı sonucu çıkarılabilir. Genel olarak, SMM ile 20 veya 24 mfcc kullanıldığında daha yüksek tanıma oranları verdiği deney sonuçlarından görülmektedir. VN' de olduğu gibi SMM' de de kişi sayısı arttıkça tanıma oranları düştüğü gibi, sistemin kullandığı optimum parametre değerleri de değişmektedir. Ayrıca Şekil 3.7' de görüldüğü gibi durum sayısının artması ile tanıma başarımı her zaman artmamaktadır. Bu da tanıma oranının durum sayısından bağımsız olduğu (Rabiner 1989) ve metinden bağımsız konuşmacı tanıma uygulamalarında durumlar arası geçişin tanıma oranında etkili olmadığı (Matsui ve Furui 1994) teorik yaklaşımları doğrulamaktadır. Şekil 3.8' den de görüleceği üzere 168 kişi için en iyi tanıma oranını, %99.6 (2 yanlış) ile 20 mfcc kullanan, 32 karışım ve 1 durumlu SMM vermektedir. Kişi sayısı 630'a çıkarıldığında ise yine en iyi tanıma oranını, %98.73 (24 yanlış) ile 20 mfcc kullanan, 32 karışım ve 1 durumlu SMM vermektedir. Şekil 3.9' a bakıldığında, mfcc vektörü boyutundan bağımsız olarak 32 karışım ve 1 durumlu SMM' nin tüm kombinasyonlar arasından en yüksek tanıma oranı verdiği görülmektedir. Sonuç olarak, 630 kişilik TIMIT veritabanının tamamı ile yapılan deneylerden de görüldüğü gibi, 32 karışım ve 1 durumlu SMM kullanmak tüm mfcc sayılarında en iyi tanıma oranını vermektedir. Ayrıca SMM ile yapılan deneylerde görüldüğü gibi genel olarak 20 mfcc kullanmak diğer mfcc sayıları ile yapılan deneylere göre daha iyi tanıma oranı sağlamaktadır.

SMM ile VN yöntemleri karşılaştırıldığında, VN' nin SMM' ye göre daha iyi performans gösterdiğini söyleyebiliriz. Bu durum şekillerden açıkça görülmektedir. VN algoritması, işlem sayısı ve uygulamadaki kolaylığı bakımından SMM' ye göre daha avantajlıdır. VN algoritmasının bir diğer avantajı ise, özellik vektörü sayısının yüksek olduğu durumlarda aynı zamanda bir sıkıştırma işlemi yapmasıdır. Bu sayede de hesaplamadaki işlem sayısı azalmaktadır. Ayrıca gerçek zamanlı bir sistemde uygulanması en kolay algoritmalardan birisi VN algoritmasıdır. Her ne kadar bu tezde metne bağlı çalışılmasa da, literatürdeki uygulamalar incelendiğinde, SMM' nin metne bağımlı konuşma ve konuşmacı tanıma uygulamalarında metinden bağımsız olanlara göre daha iyi performans gösterdiği, bunun da SMM' nin temelinde yatan durumlar arası geçiş işlevinden kaynaklandığı görülmektedir (Yu ve diğ. 1995).

3.4. Karşılaştırmalı Sonuçlar

Bu kısımda literatürde yapılan çalışmalar ile bu tezde elde edilen sonuçların karşılaştırmaları yapılmıştır. Karşılaştırma yapmak için gereken en önemli şart, kullanılan veritabanının herkes tarafından kolay ulaşılabilir ve güvenilir olmasıdır. Hem SMM hem de VN algoritması ile yapılan çok sayıda çalışma olmasına rağmen, birçoğu ile hem kullanılan veritabanları hem de kullanılan özellik vektörleri bu çalışmadakilerle uyuşmamaktadır. Literatürde, genellikle az sayıda konuşmacının kullanıldığı çalışmalar yer almaktadır.

İlk olarak VN yöntemi ile TIMIT veritabanı kullanılarak son yıllarda yapılan iki çalışma ile bu tezin VN kullanılarak elde edilen sonuçları karşılaştırılacaktır. Öncelikle 2006 yılında Kinnunen tarafından yapılan çalışma ile karşılaştırma yapılacaktır (Kinnunen ve ark. 2006). Çizelge 3.3 de A sütunu Kinnunen ve arkadaşları tarafından elde edilen sonuçları, D sütunu bu tezde bulunan deneysel sonuçları verirken, B ve C sütunları, kıyaslamanın daha anlamlı olması için bu tezdeki sistem parametrelerinin Kinnunen ve arkadaşlarının deney şartlarına uydurmak için yapılmış aşamalara ait sonuçları göstermektedir.

Çizelge 3.3 VN algoritması ile TIMIT 630 kişi için karşılaştırmalı sonuçlar

	Tanıma Oranları (%)			
Kod Kitabı Boyutu	**A**	**B**	**C**	**D**
16	97.78	97.78	98.88	87.83
32	99.37	99.37	100	95.29
64	99.52	99.52	100	96.66
128	99.84	99.84	100	98.21

Çizelge 3.3 de verilen tüm (4 sütunun hepsi) sonuçların elde edildiği ortak sistem parametreleri aşağıdaki gibi tanımlanabilir.

- 630 kişilik TIMIT veritabanının tamamına uygulanmıştır,
- 12 adet mfcc katsayısı kullanılmıştır,

İlk 3 sütun (A, B ve C sütunları) ile son sütun (D) arasındaki tek fark ise test süreleridir. İlk 3 çalışmada her konuşmacının 10 cümlesinin 7 si (~ 21 s) eğitim, 3 ü (~ 9 s) test için kullanılırken, son çalışmada her konuşmacının 10 cümlesinin 7 si (~ 21 s) eğitim, 3 ü ayrı ayrı (~ 3 s) test için kullanılmıştır. Dolayısıyla ilk üç sütunda yapılan test sayısı 630 iken son sütunda 1890 test sonucu verilmektedir.

Çizelge 3.3' in ilk iki sütunundan (A ve B) görüldüğü üzere aynı parametreler ile elde edilen sonuçlar birebir aynıdır. Ancak Çizelge 3.3' in üçüncü sütununda (C) verilen sonuçlar her kod kitabı boyutu için daha yüksektir. İlk iki sütun ile üçüncü sütunun deney şartları arasında 2 önemli fark bulunmaktadır. Bunlardan ilki, 1. ve 2. sütundaki deneylerde, TIMIT veritabanındaki cümlelerin 16 kHz olan örnekleme frekansı 8 kHz' e düşürülmüştür. Diğeri ise mfcc katsayılarının elde edildiği süzgeç takımındaki süzgeç sayısındaki farktır. İlk iki sütundaki deneylerde 27 adet üçgen süzgeç kullanılırken bu tezde kullanılan üçgen süzgeç sayısı 40 dır. Çizelgenin son sütunundaki (D) deney şartlarındaki fark ise her konuşmacıya ait 3 test cümlesinin ayrı ayrı test edilmesidir. Aslında gerçek zamanlı sistemlerde de olması istenen budur. Çünkü gerçek zamanlı uygulamalarda sisteme girilen test işaretleri genellikle kelime, dijit ya da cümle olmaktadır. Gerçek zamanlı uygulamalarda 9 s' lik bir test işareti almak her zaman mümkün olmayabilir.

VN algoritması ile TIMIT veritabanının 168 kişilik test dizininin kullanıldığı diğer bir çalışma (Fan ve Roska 2003) ile bu tezde elde edilen sonuçlar karşılaştırılacaktır. Karşılaştırılacak olan çalışmada kullanılan sistem parametreleri şu şekilde sıralanabilir:

- TIMIT veritabanının 168 kişilik test dizini kullanılmıştır,
- VN algoritması ile 16,32 ve 64 kod kitabı boyutları için testler yapılmıştır,
- Özellik vektörü olarak 20 adet mfcc katsayısına ek olarak bunların birinci türevleri eklenmiştir. Dolayısıyla toplam 40 adet özellik kullanılmıştır,
- Her kişi 8 cümlesi ile eğitilmiş ve 2 cümlesi ile test edilmiştir. 168 kişi için toplam 336 test yapılmıştır.

TIMIT' in tek oturumda kaydedilmiş bir veritabanı olduğu, kayıtlarda kullanılan mikrofonlarda farklılık olmadığı ve tanıma oranını düşürdüğü için (Reynolds 1995), bu tezde yapılan testler sırasında özellik vektörlerinin birinci türevleri kullanılmamıştır. Fan

ve arkadaşları tarafından yapılan çalışmada elde edilen tanıma oranları ile bu tezde 20 mfcc katsayısı kullanılarak TIMIT veritabanının 168 kişilik test dizinindeki her bir konuşmacı 7 cümlesi ile eğitilip kalan 3 cümlesi ise ayrı ayrı test edildiğinde (168 kişi için toplam 504 test) elde edilen tanıma oranları Çizelge 3.4 de verilmektedir.

Çizelge 3.4 VN algoritması ile TIMIT 168 kişi için karşılaştırmalı sonuçlar

	Tanıma Oranları (%)	
Kod Kitabı Boyutu	**Fan ve Ark.**	**Bu Tezde**
16	79.8	96.23
32	91.4	98.41
64	93.8	99.60

Çizelge 3.4' den de görüldüğü gibi 20 adet özellik vektörüne bunların birinci türevleri eklenmeden ve daha fazla sayıda test yaparak daha yüksek tanıma oranları elde edilmiştir. Buradan da TIMIT veritabanı için dinamik özelliklerin (türevler) tanıma oranını düşürdüğü görülmektedir.

İkinci olarak bu tezde elde edilen SMM sonuçları literatürde verilen sonuçlar ile karşılaştırılacaktır. Ancak, hem TIMIT veritabanının tamamını kullanan hem de SMM kullanılarak yapılan çalışma mevcut olmadığından bu kısımda SMM' nin özel bir hali (1 durumlu SMM) olarak literatürde kullanılan Gauss Karışım Modeli (GKM) ile yapılan çalışmalarla karşılaştırma yapılacaktır. GKM, literatürde 1 durumlu SMM ile eşdeğer görülmektedir (Furui 1997).

Öncelikle GKM yöntemi ile TIMIT veritabanı kullanılarak Reynolds tarafından yapılan iki çalışma (Reynolds ve ark. 1995, Reynolds 1995) ile bu tezin SMM kullanılarak elde edilen sonuçları karşılaştırılacaktır. İlk olarak 1995 yılında Reynolds ve arkadaşları tarafından yapılan çalışma ile karşılaştırma yapılacaktır. Yapılan çalışmada Reynolds ve arkadaşları TIMIT veritabanının 168 kişilik test dizini ile GKM yöntemi ile metinden bağımsız konuşmacı tanıma sonuçları elde etmiştir. 1995 yılında yapılan bu çalışmada veritabanında bulunan her kişiyi 8 cümlesi ile eğitmiş, kalan 2 cümlesi ile test etmişlerdir. Özellik vektörü olarak 24 adet mfcc katsayısı kullanılmıştır. Çizelge 3.5

Reynolds ve arkadaşları tarafından elde edilen sonuçlar ile bu tezde bulunan deneysel sonuçları göstermektedir.

Çizelge 3.5. SMM ile TIMIT 168 kişi için karşılaştırmalı sonuçlar

	Tanıma Oranı (%)	
Model Parametresi	**Reynolds ve ark.**	**Bu Tezde**
32 karışım (1 durum)	99.1	99.2

Çizelge 3.5' de belirtilen iki çalışma arasındaki tek fark yapılan test sayısındadır. Reynolds ve arkadaşları tarafından her kişinin 2 cümlesi test olarak kullanılmış (toplam 336 test) olup, bu tezde yapılan deneylerde her kişinin 3 cümlesi test edilmiştir (toplam 504 test). Görüldüğü gibi SMM ile daha fazla sayıda test yapılmasına rağmen en az GKM kadar iyi sonuç vermektedir.

İkinci olarak Reynolds 1995 (Reynolds 1995) yılında yaptığı diğer bir çalışmada ise 630 kişilik TIMIT veritabanının tamamını kullanarak GKM ile tanıma oranını belirtmiştir. Yine bir önceki çalışmada olduğu gibi Reynolds bu çalışmasında da her kişiyi 8 cümlesi ile eğitip, kalan 2 cümlesi ile test etmiştir (toplam 1260 test). Ancak bu defa Reynolds 24 yerine 30 adet mfcc katsayısı kullanmıştır. Bu tezde yapılan çalışmalarda ise daha önce de belirtildiği gibi 24 mfcc kullanılarak her kişi 7 cümlesi ile eğitilip kalan 3 cümlesi ile test edilmiştir. Reynolds' un elde ettiği sonuç ile bu tezde elde edilen sonuç Çizelge 3.6' de verilmektedir.

Çizelge 3.6. SMM ile TIMIT 630 kişi için karşılaştırmalı sonuçlar

	Tanıma Oranı (%)	
Model Parametresi	**Reynolds ve ark.**	**Bu Tezde**
32 karışım (1 durum)	99.5	98.47

Reynolds' un yaptığı deneyler ile bu tezde yapılan deneyler arasında önemli bir fark bulunmaktadır. Bu fark, mfcc katsayılarının çıkarımı algoritmasının adımlarından biri olan ön-vurgulama süzgecinin uygulanış sırasıdır. Reynolds yaptığı tüm çalışmalarda ön-vurgulama filtresini işaretin spektrumuna uygulamaktadır, yani hızlı Fourier

dönüşümü alınan işarete uygulamaktadır. Oysa bu çalışmada kullanılan mfcc katsayıları çıkarılırken ön-vurgulama süzgeci işaretin spektrumuna değil kendisine uygulanmıştır.

2006 yılında Kinnunen ve arkadaşları tarafından yapılan çalışmada VN algoritması dışında GKM ile de test yapılmıştır (Kinnunen ve ark. 2006). Son karşılaştırma olarak Kinnunen ve arkadaşlarının elde ettiği tanıma oranı ile bu tezde elde edilen tanıma oranları verilecektir. Kinnunen ve arkadaşları çalışmalarında TIMIT veritabanı ile 32 karışımlı Gauss Karışım Modeli ve 12 mfcc katsayısı kullanarak 630 kişi için, her kişiyi 7 cümlesi ile eğitip (~ 21 sn) ve 3 cümlesiyle test (~9 sn) ederek, %99.68 (1 yanlış) tanıma oranı elde etmiştir. Gauss Karışım modeli tek durumlu SMM' ye karşılık geldiğinden (Furui 1997) bu tezde, 32 karışım ve 1 durumlu SMM ile yine 7 cümle eğitim 3 cümle test ve 12 mfcc kullanarak 630 kişi için yapılan test sonucunda %100 tanıma oranı elde edilmiştir.

Bunların dışında TIMIT veritabanı ile SMM kullanılarak yapılan çalışmalar literatürde mevcuttur. Ancak kullanılan kişi sayısının az olması ve elde edilen tanıma oranlarının bu tezde elde edilen tanıma oranlarından düşük olması nedeniyle bu çalışmalar ile karşılaştırma ihtiyacı duyulmamıştır.

ÖNERİLER

Bu tezde gürültü içermeyen bir veritabanı olan TIMIT kullanılmıştır. Ancak gerçek zamanlı uygulamalarda bu kadar iyimser sonuçlar alınamayacağı açıktır. Konuşmacı tanıma uygulamalarının en fazla kullanıldığı alanlardan biri olan telefon bankacılığı işlemlerinde gürültünün olmaması mümkün değildir. Telefon hatlarından kaynaklanan bozulma veya günümüzde yaygın olarak kullanılan cep telefonlarıyla yapılan konuşmalar sırasında meydana gelen bozulmalardan dolayı yüksek tanıma oranları elde edilmesi zordur. Bu nedenle, bu tezde kullanılan iki yöntemin yine herkes tarafından kolayca ulaşılabilecek, telefon hattı üzerinden konuşmaların olduğu bir veritabanı üzerinde test edilmesi faydalı olacaktır. VN algoritmasında kullanılan farklı uzaklık ölçütlerinin de gürültü içeren bir veri tabanı ile denenmesi, birbirlerine göre üstünlükleri konusunda daha kesin yargıya varılmasını kolaylaştıracaktır.

Aynı şekilde telefon hatlarından kaynaklanan bozulmayı engelleyici veya meydana gelen bozulmayı giderici algoritmalarda kullanılabilir. Ayrıca gerçek zamanlı bir sisteme benzetebilmek adına konuşmacılara ait eğitim cümleleri gürültü içermeyen veritabanından seçilip, test cümleleri ise telefon hattından geçirilmiş cümlelerden seçilebilir. Böylece kullanılan sınıflandırıcıların birbirlerine göre avantajları veya varsa dezavantajları tespit edilebilir.

Bu tezdeki yazılımlar MATLAB 7.1 ortamında gerçekleştirilmiştir. Özellikle 630 kişilik konuşmacının tamamı kullanıldığında eğitim ve test aşamaları bir hayli uzun sürmektedir. Bu nedenle, yüksek sayıda konuşmacı kullanacak sistemlerin, gerçek zamanlı sistemler ile kıyaslanabilir olabilmesi için yazılımların C dili ile yapılması önerilir.

KAYNAKLAR

Atal, S. B. 1976. Automatic Recognition of Speaker from Their Voices. Proc. of IEEE, 64 (4), p.460-475

Bennani, Y. and Gallinari, P. 1991. On the use of TDNN-Extracted Features Information in Talker Identification, IEEE Proc. ICASSP'91, p. 385-388

Campbell, J. P. 1997. Speaker recognition: A tutorial. Proc. IEEE 85 (9), p. 1437-1462

Chester, M. 1993. Neural Networks: A tutorial. Prentice Hall.

Deller, J. R., Proakis J. G. and Hansen J. H. L. 1993. Discrete Time Processing of Speech Signals. Macmillan Publishing Company.

Doddington, G. 1985. Speaker Recognition-Identifying People by Their Voices. Proc. IEEE 73 (11), p.1651-1664.

Fan, N. and Rosca, J. 2003. Enhanced VQ-based Algorithms for Speech Independent Speaker Identification. Audio and Video-Based Biometric Person Authentication.

Furui, S. 1994, An Overview of Speaker Recognition Technology, ESCA Workshop on Automatic Speaker Recognition, Identification and Verification, p. 1-9

Furui, S. 1997. Recent Advances in Speaker Recognition. Pattern Recognition Letters 18, p.859-872.

Furui, S. 1981. Cepstral Analysis Technique for Automatic Speaker Verification. IEEE Transactions on Acoustic Speech Signal Processing 29 (2), p.254-272.

Gish, H. and Schimidt, M. 1994. Text-Independent Speaker Identification. IEEE Signal Processing Magazine, p.18-31

Higgins, A., Bahler, L. and Porter, J. 1991. Speaker Verification Using Randomize Phrase Prompting. Digital Signal Processing 1, p.89-106

Jankowski, C., Kalyanswamy, A., Basson, S. and Spitz, J. 1990. NTIMIT: A Phonetically Balanced, Continuous Speech, Telephone Bandwidth Speech Database. IEEE International Conference on Acoustic, Speech and Signal Processing, ICASSP-90, p. 109-112

Jankowsi, C. R., Quatieri, T. F. and Reynolds, D. A. 1994. Formant AM-FM for speaker Identification. IEEE International Symposium on Time-Frequency and Time-Scale Analysis, p. 608-611

Juang, B. H., Rabiner, L. R., Levinson, S. E. and Sondhi, M. M. 1985. Recent Developments in the Application of Hidden Markov Modelst to Speaker Independent Isolated Word Recognition, IEEE Int. Conf. Acoustic, Speech and Signal Proc., p. 9-12

Juang, B. H. and Rabiner L. R. 1990. The Segmental K-Means Algorithm for Estimating Parameters of Hidden Markov Models. IEEE Transactions on Acoustic, Speech and Signal Processing 38 (9), p. 1639-164

Kinnunen, T., Kilpelainen, T. and Franti, P. 2000. Comparison of Clustering Algorithms in Speaker Identification, Int. Conf. Signal Processing and Communications, p. 222-227

Kinnunen, T. 2003. Spectral Features for Automatic Speaker Recognition. PhD Thesis, University of Joensuu, Finland.

Kinnunen, T., Karpov, E. and Franti, P. 2006. Real-Time Speaker Identification and Verification. IEEE Transactions on Audio, Speech and Language Processing 14 (1), p. 77-288

Li, K. and Wrench, E. 1983. An Approach to Text-Independent Speaker Recognition with Short Utterances, Proc. IEEE Conf. Acoustic, Speech and Signal Processing, p. 555-558

Linde, Y., Buzo, A. and Gray, R. M. 1980. An Algorithm for Vector Quantization. IEEE Transactions on Communications 28 (1), p. 84-95

Markel, J. D., Oshika, B. T. and Gray, A. H. 1977. Long-term Feature Averaging for Speaker Recognition. IEEE Transactions on Acoustic, Speech and Signal Processing 25 (4), p. 330-337

Matsui, T. and Furui, S. 1990. Text-Independent Speaker Recognition Using Vocal Tract and Pitch Information, Int. Conf. Spoken Language Processing, ICSLP, p. 137-140

Matsui, T. and Furui, S. 1991. A Text-Independent Speaker Recognition Method Robust Against Utterance Variations, IEEE Proc., p. 377-380

Matsui, T. and Furui, S. 1994. Comparison of Text-Independent Speaker Recognition Methods Using VQ-Distortion and Discrete/Continuous HMM's. IEEE Transactions on Speech and Audio Processing, p. 456-459

Naik J., Netsch M. and Doddington G. 1989. Speaker Verification over Long Distance Telephone Lines. IEEE International Conference on Acoustic Speech Signal Processing, p.524-527.

Naik, J. M. 1990. Speaker Verification: A Tutorial. IEEE Communication Magazine, p. 42-48

Oglesby, J. and Mason, J. S. 1990. Optimization and Neural Models for Speaker Identification, IEEE Proc., p.261-264

Oglesby, J and Mason, J. S. 1991. Radial Basis Function Networks for Speaker Recognition, IEEE Proc. ICASSP'91, p. 393-396

O'Shaughnessy, D. 1986. Speaker Recognition. IEEE ASSP Magazine 3 (4), p.4-17.

Paliwal, K. K. 1992. Dimensionality Reduction of the Enhanced Feature Set for HMM Speech Recognizer, Digital Signal Processing, 2, p. 157-173

Rabiner, L. R., Wilpon, J. G. and Soong, F. K. 1988. High Performance Connected Digit Recognition Using Hidden Markov Models, IEEE Int. Conf. Acoustic, Speech and Signal Proc., p. 119-122

Rabiner, L. R. 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc IEEE, 77, p. 257-286

Rabiner, L. R. and Juang, B. H., 1993. Fundamentals of Speech Recognition. Prentice Hall, New Jersey

Rabiner, L. R and Juang B. H., 2005. Statistical Methods of Speech Recognition, Elsevier Encyclopedia of Language and Linguistics, Second Edition

Reynolds, D. A. 1992. A Gaussian Mixture Modeling Approach to Text Independent Speaker Identification. PhD. Thesis, Georgia Institute of Technology.

Reynolds, D. A., Rose, R. C. and Smith, M. J. T. 1992. PC-based TMS320C30 Implementation of the Gaussian Mixture Model Text-Independent Speaker Recognition, Int. Conf. Signal Processing Applications, p. 967-973

Reynolds, D. A. and Rose, R. C. 1995. Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models, IEEE Trans. on Speech and Audio Processing, 3 (1), p. 72-83

Reynolds, D. A., Zissman, M. A., Quatieri, T.F., O'Leary, G. C. and Carlson, B. A. 1995. The Effects of Telephone Transmission Degradations on Speaker Recognition Performance. IEEE International Conference on Acoustic, Speech and Signal Processing, p. 329-332

Reynolds, D. A. 1995. Large Population Speaker Identification Using Clean and Telephone Speech. IEEE Signal Processing Letters 2 (3), p. 46-48

Reynolds, D. A. and Carlson, B.1995. Text-Dependent Speaker Verification Using Decoupled and Integrated Speaker and Speech Recognizers, Proc. EUROSPEECH, p. 647-650

Rosenberg, A. E. and Sambur, M. R. 1975. New Techniques for automatic speaker Verification. IEEE Transactions on Acoustic Speech and Signal Processing 23 (2), p. 169-175

Rosenberg, A. E. 1976. Automatic Speaker Verification: A Review. Proc. of IEEE 64 (4), p. 475-487

Rosenberg, A. E. and Soong, F. K. 1987. Evaluation of a Vector Quantization Talker Recognition System in Text-Independent and Text-Dependent Modes, Computer Speech and Language, p. 143-157

Rosenberg, A., Lee, C. and Gökcen, S. 1991. Connected Word Talker Verification Using Whole Word Hidden Markov Models. Proc. IEEE international conference on Acoustic Speech, Signal Processing, Toronto, p.381-384

Rudasi, L. and Zahorian, S. A. 1991. Text-Independent Talker Identification with Neural Networks, Proc. ICASSP'91, p. 389-392

Schaefer, R. W. and Rabiner, L. R. 1970. System for Automatic Formant Analysis of Voiced Speech. Journal of Acoustic Society of America, 47, p.634-648

Soong, F. K., Rosenberg, A. E., Rabiner, L. R. and Juang, B. H. 1985. A Vector Quantization Approach to Speaker Recognition, Proc. IEEE, p. 387-390

Tierney, J. 1980. A Study of LPC Analysis of Speech in Additive noise. IEEE Trans. on Acoustic, Speech and Signal Processing, 28, p. 389-397

Theodoridis, S. and Koutroumbas K. 2003. Pattern Recognition 2nd Edition. Academic Pres

Thisby, N. Z. 1991. On the Application of Mixture AR Hidden Markov Models to Text-Independent Speaker Recognition, IEEE Trans. Acoustic, Speech and Signal Proc., p. 563-570

Van Alpen, P. and Pols, L. 1991. Comparing Various Feature Vectors in Automatic Speech Recognition, Proc. EUROSPEECH, p. 533-536

Wolf, J. 1972. Efficient Acoustic Parameters for Speaker Recognition. Journal of Acoustic Society of America 51 (6), p. 2044-2056

Yu, K., Mason, J. and Oglesby, J. 1995. Speaker Recognition using Hidden Markov Models, Dynamic Time Warping and Vector Quantization. IEEE Proceedings of Vision, Image and Signal Processing 142 (5), p. 313-318

Zheng Y. and Yuan B. 1988. Text-Independent speaker identification using circular Hidden Markov Models. IEEE International Conference on Acoustic Speech Signal Processing, p.580-582.

Zhu, X., Gao, Y., Ran, Chen, F., Macheod, I., Millar, B. and Wagner, M. 1994. Text-Independent Speaker Recognition Using VQ, Mixture Gaussian VQ and Ergodic HMMs. Esca Workshop on Automatic Speaker Recognition, Identification and Verification, Martigny.

Zue V., Seneff S., and Glass J., 1990 Speech Database Development at MIT: TIMIT and beyond. Speech Communication, 9, p.351-356

EKLER

EK–1. İLERİ-GERİ İŞLEMİ

Verilen bir λ modeli için t anında i. durumda $o_1, o_2, ..., o_t$ gözlem dizisinin olasılığı aşağıdaki denklemde belirtildiği gibi tanımlansın.

$$\alpha_t(i) = P(o_1, o_2, ..., o_t, q_t = i|\lambda)$$

$\alpha_t(i)$ şu şekilde çözülebilir.

1. Başlangıç Değeri Atama

$$\alpha_1(i) = \pi_i b_{,}(o_1), \qquad 1 \le i \le N$$

2. Özyineleme

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(o_{t+1}), \qquad \begin{matrix} 1 \le t \le T-1 \\ 1 \le j \le N \end{matrix}$$

3. Sonlandırma

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

Geri İşlemi

Verilen bir λ modeli için t anında i. durumda $o_{t+1}, o_{t+2}, ..., o_T$ gözlem dizisinin olasılığı aşağıdaki gibi tanımlansın.

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

$\beta_t(i)$ şu şekilde çözülebilir.

1. Başlangıç Değeri Atama

$$\beta_T(i) = 1, \quad 1 \le i \le N$$

2. Tümevarım

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

EK–2. VITERBI ALGORİTMASI

1. Başlangıç Değeri Atama

$$\delta_0 = \pi_i, \quad 1 \le i \le N$$

2. Özyineleme

$$\delta_t = \max_{1 \le i \le N} \left[\delta_{t-1}(i) a_{ij} \right] b_j(O_t)$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} \left[\delta_{t-1}(i) a_{ij} \right]$$

$$1 \le t \le T, \quad 1 \le j \le N$$

3. Sonlandırma

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

$$i_T^* = \arg\max_{1 \le i \le N} \delta_T(i)$$

4. Geriye gidilerek en yüksek olasılık veren durum dizisinin elde edilmesi

$$t = T-1, T-2, \ldots, 1$$

$$i_t^* = \psi_{t+1}(i_{t+1}^*)$$

ÖZGEÇMİŞ

Cemal HANİLÇİ, 14 Şubat 1983 tarihinde Malatya'da doğdu. İlk, orta ve lise eğitimini burada tamamladı. 2001 yılında Uludağ Üniversitesi Elektronik Mühendisliği bölümünü kazandı ve 2005 yılında mezun oldu. Aynı yıl Uludağ Üniversitesi Fen Bilimleri Enstitüsü Elektronik Mühendisliği anabilim dalında yüksek lisans eğitimine başladı. Halen 2005 yılında başladığı yüksek lisans eğitimine devam etmektedir.

TEŞEKKÜR

Öğrenim hayatım boyunca ve özellikle de bu çalışma süresince desteklerini esirgemeyen, engin bilgileriyle yol gösteren danışman hocam Yrd. Doç. Dr. Figen ERTAŞ ve görüşleriyle ışık tutan hocam Prof. Dr. Tuncay ERTAŞ' a teşekkürü bir borç bilirim. Ayrıca bu çalışma boyunca her türlü desteğini esirgemeyen, sabrı ve bilgisiyle bana destek olan değerli hocam Yrd. Doç. Dr. Rıfat EDİZKAN' a teşekkür ederim. Ayrıca bu çalışma boyunca bana her konuda yardımcı olan Sayın Ömer ESKİDERE' ye, hayatımın her anında maddi ve manevi desteklerini devamlı hissettiren aileme, sabrı ve sevgisiyle devamlı bana destek veren sevgili Selver Başak MUTOĞLU' na sonsuz teşekkürlerimi sunuyorum.