

OVY ve GMM ile Metinden Bağımsız Konuşmacı Tanıma

Selami Sadıç

DOKTORA TEZİ

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Eylül 2007

Text-Independent Speaker Recognition Using CVA and GMM

Selami Sadıç

DOCTORAL DISSERTATION

Department of Electrical and Electronics Engineering

September 2007

OVY ve GMM ile Metinden Bağımsız Konuşmacı Tanıma

Selami Sadıç

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Elektronik Bilim Dalında
DOKTORA TEZİ
Olarak Hazırlanmıştır

Danışman: Doç.Dr. M.Bilginer Gülmezoğlu

Eylül 2007

Selami Sadıç' ın DOKTORA tezi olarak hazırladığı “OVY ve GMM ile Metinden Bağımsız Konuşmacı Tanıma” başlıklı bu çalışma, jürimizce lisansüstü yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

Üye : Doç.Dr. M.Bilginer Gülmezoğlu (Danışman)

Üye : Prof.Dr. Atalay Barkana

Üye : Yrd.Doç.Dr. Rifat Edizkan

Üye : Yrd.Doç.Dr. Atıf Çay

Üye : Yrd.Doç.Dr. Erol Seke

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun tarih ve sayılı kararıyla onaylanmıştır.

Prof. Dr. Abdurrahman KARAMANCIOĞLU

Enstitü Müdürü

ÖZET

Bu tez çalışmasında Ortak Vektör Yaklaşımı (OVY), yeni bir uygulama olarak metinden bağımsız konuşmacı tanıma kullanılmıştır. Ovy'nin başarısı Fisher'in doğrusal ayırtaç analizi (FLDA) ve Gauss katışım modelleri (GMM) yöntemleriyle karşılaştırılmıştır. TIMIT veri tabanı için elde edilen tanıma oranları, Ovy ve GMM'in FLDA'dan üstün olduğunu göstermiştir. Ovy yöntemiyle bulunan tanıma sonuçları GMM'den elde edilen sonuçlar ile aynı olmakla birlikte, işlem hızı ve bellek ihtiyacı açısından Ovy daha avantajlıdır. Bu tez çalışmasında ayrıca GMM'den elde edilen sonuçların iyileştirilmesi için Ovy ve GMM'in birleşiminden oluşan yeni bir yöntem önerilmiştir.

Anahtar Kelimeler: Konuşmacı tanıma, Öznitelik çıkarma, Ortak vektör yaklaşımı, Gauss katışım modelleri, Fisher'in doğrusal ayırtaç analizi.

SUMMARY

In this doctoral dissertation, the common vector approach (CVA) is newly used for text-independent speaker recognition. The performance of CVA is compared with those of Fisher's linear discriminant analysis (FLDA) and Gaussian mixture models (GMM). The recognition rates obtained for the TIMIT database indicate that CVA and GMM are superior to FLDA. However, while the recognition rates obtained from CVA and GMM are identical, CVA has the advantages in terms of processing power and memory requirement. In order to obtain better results than those achieved with GMM, a new method which is a combination of CVA and GMM is proposed in this study.

Keywords: Speaker recognition, Feature extraction, Common vector approach, Gaussian mixture models, Fisher's linear discriminant analysis.

TEŐEKKÜR

Doktora alıŐmalarında, gerek derslerimde ve gerekse tez alıŐmalarında, bana danıŐmanlık ederek, beni ynlendiren ve her trl olanađı sađlayan danıŐmanım Do.Dr. M.Bilginer Glmezođlu'na teŐekkr ederim.

Tez alıŐmalarında gstermiŐ olduđu katkılarında dolayı deđerli arkadaŐım Hikmet Ycel'e de ayrıca teŐekkr ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	V
SUMMARY	VI
TEŞEKKÜR	VII
ŞEKİLLER DİZİNİ	X
ÇİZELGELER DİZİNİ	XI
SİMGELER VE KISALTMALAR DİZİNİ	XII
1. GİRİŞ	1
2. KONUŞMACI TANIMA	5
2.1 Giriş	5
2.2 Konuşmacı Tanıma Veri Tabanları.....	5
2.3 Konuşmacı Tanıma Sistemi.....	7
2.3.1 Konuşmacı tanıma türleri.....	7
2.3.2 Metine bağımlı ve metinden bağımsız tanıma	8
2.3.3 Konuşmacı tanıma sisteminin genel yapısı.....	8
2.3.4 Parametre seçimi ve öznitelik çıkarma	9
2.3.5 Örüntü karşılaştırma ve sınıflandırma.....	11
2.4 Konuşmacı Tanımanın Kullanıldığı Alanlar.....	11
3. KONUŞMACI TANIMADA KULLANILAN YÖNTEMLER	12
3.1 Giriş	12
3.2 Fisher'in Doğrusal Ayırtaç Analizi (FLDA)	12
3.3 Gauss Katılım Modelleri (GMM)	13
3.4 Ortak Vektör Yaklaşımı (OVY)	16
3.4.1 Yetersiz veri durumunda OVY	16
3.4.2 Yeterli veri durumunda OVY	18
3.5 Ayırteci Ortak Vektör Yaklaşımı (DCV)	19
3.6 OVY Tabanlı GMM	22

İÇİNDEKİLER (devam)

	<u>Sayfa</u>
4. DENEYSEL ÇALIŞMA	28
4.1 Giriş	28
4.2 TIMIT Veri Tabanı	28
4.3 Deneysel Çalışmada Kullanılacak Şekilde Veri Tabanının Düzenlenmesi	29
4.4 Öznitelik Vektörlerinin Elde Edilmesi	31
4.5 Karar Kriterleri ve Tanıma Oranları	33
4.5.1 Yeterli veri durumu ($N > d$) için yapılan çalışmalar	33
4.5.2 Yetersiz veri durumu ($N < d$) için yapılan çalışmalar	39
4.6 Uygulanan yöntemlerin hesaplama maliyetleri	41
5. SONUÇLAR	42
6. KAYNAKLAR DİZİNİ	45
EKLER.....	49
ÖZGEÇMİŞ.....	62

ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
Şekil 2.1 Konuşmacı tanıma sistemi genel yapısı.....	9
Şekil 3.1 Bir konuşmacıya ait tüm Gauss yoğunluk işlevleri.....	14
Şekil 3.2 GMM yönteminde bir konuşmacının modellenmesi.....	15
Şekil 3.3 GMM ve OVY tabanlı GMM yöntemlerinde (a) GMM yönteminde elde edilen gauss dağılımları (b) OVY tabanlı GMM yönteminde elde edilen gauss dağılımlarının karşılaştırılması	27
Şekil 4.1 MFCC algoritması akış diyagramı	32
Şekil 4.2 OVY, GMM ve OVY tabanlı GMM yöntemlerinin test seti için tanıma oranları.....	39

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
Çizelge 4.1 TIMIT konuşma materyali.....	29
Çizelge 4.2 Veri tabanındaki okunuşlar ile ilgili dosya tipleri	29
Çizelge 4.3 Düzenlenmiş TIMIT veri tabanındaki konuşmacı bilgileri	30
Çizelge 4.4 TIMIT SA, SI, SX cümlelerinin ortalama örnek sayıları ve uzunlukları	31
Çizelge 4.5 Eğitim seti tanıma oranları	34
Çizelge 4.6 Test seti tanıma oranları	35
Çizelge 4.7 GMM ve OVY tabanlı GMM yöntemlerinin eğitim seti tanıma oranları ...	37
Çizelge 4.8 GMM ve OVY tabanlı GMM yöntemlerinin test seti tanıma oranları	38
Çizelge 4.9 ($N < d$) ve ($N > d$) durumunda uygulanan yöntemlerin tanıma oranlarının karşılaştırılması	40
Çizelge 4.10 Yöntemlerin eğitim ve test süreleri	41
Çizelge 4.11 Yöntemlerin bellek ihtiyaçlarının karşılaştırılması	41

SİMGELER VE KISALTMALAR DİZİNİ

<u>Simgeler</u>	<u>Açıklamalar</u>
X_j	: j 'ninci sınıf
N_T	: Bütün konuşmacıların toplam öznitelik vektör sayısı
N_j	: X_j sınıfındaki öznitelik vektör sayısı
N	: Her sınıftaki öznitelik vektör sayısı (yetersiz veri durumu)
c	: Konuşmacı (sınıf) sayısı
μ	: Bütün sınıfların ortalama vektörü
μ_j	: X_j sınıfının ortalama vektörü
d	: Öznitelik vektör boyutu
M	: Gauss bileşen (katılım) sayısı
W	: İzdüşüm matrisi
Φ_j	: X_j sınıfının saçılım matrisi
S_B	: Sınıflar arası saçılım matrisi
S_w	: Toplam sınıf içi saçılım matrisi
λ	: Özdeğer
u, w	: Özvektör
Q	: Farklılık altuzayı
\bar{Q}	: Farksızlık altuzayı

<u>Kısaltmalar</u>	<u>Açıklamalar</u>
CVA	: Common Vector Approach / Ortak vektör yaklaşımı
DCT	: Discrete Cosine Transform / Kesikli kosinüs dönüşümü
DCV	: Discriminative Common Vectors / Ayırteci ortak vektörler
EM	: Expectation-Maximization algorithm / Beklenti-embüyütme algoritması
FLDA	: Fisher's Linear Discriminant Analysis / Fisher'in doğrusal ayırtaç analizi
GMM	: Gaussian Mixture Models / Gauss katılım modelleri
MFCC	: Mel-Frequency Cepstral Coefficients / Mel-frekans keprstral katsayıları
PCA	: Principal Component Analysis / Ana bileşenler analizi

1. BÖLÜM

GİRİŞ

Konuşmacı tanıma, geçtiğimiz 20 yılda gittikçe daha fazla ilgi çeken bir konu haline gelmiştir. Konuşmacı tanıma, bilinmeyen bir konuşmacıya ait konuşma örneği ile daha önceden bütün konuşmacılardan alınan konuşma örneklerinin karşılaştırılarak, giriş konuşma örneğinin hangi konuşmacıya ait olduğunu bulma işlemidir. Konuşmacı tanıma, klasik örüntü tanıma problemine bir örnek teşkil ettiğinden, konuşmacı tanımanın ana amacı alınan konuşma örneklerinden bireysel konuşmacıların ayırt edici karakteristiklerini çıkartmak ve modellemektir. Bu nedenle konuşma örüntülerinden konuşmacı tanımak için, konuşmacılar arası değişimi yüksek, konuşmacı içi değişimi düşük, yüksek konuşmacı ayırt etme gücüne sahip özelliklere ihtiyaç vardır.

Metinden bağımsız konuşmacı tanıma için çeşitli konuşmacı modelleri ve uzaklık ölçümleri önerilmiştir (Hayakawa and Itakura 1994; Griffin et al., 1994; Ariki et al., 1996; Quatieri et al., 2000). Konuşmacı tanıma sistemleri, test girdisi ile ilgili karar vermekte dikkate alınan toplam skoru hesaplamak için genellikle çerçeve skorlarının ortalamasını kullanır. Bununla birlikte, Besacier ve Bonastre (1998), metinden bağımsız konuşmacı tanıma için yeni bir çerçeve seçme yöntemi önermiştir. Çalışmalarında test girdisinin tamamına ait ortalama çerçeve olabilirliğini kullanmak yerine, sonuç skorunu bazı çerçeveleri atarak belli sayıda çerçeveyi dikkate alarak hesaplamışlardır.

Örüntü eşleme ve buna karşı gelen modellerin birçok türü olabilir. Örüntü eşleme yöntemleri dinamik zaman bükmesi (DTW), gizli Markov modelleme (HMM) (Roch and Hurtig, 2002; Chaudhari et al., 2003), vektör nicemleme (VQ) (Alonso-Martinez et al., 2000; Liu et al., 1996) ve yapay sinir ağları (ANN) (Rodriguez-Porcheron et al., 1999; Campbell et al., 2002) yöntemlerinden oluşur. DTW’de şablon modelleri, HMM’de istatistiksel modeller ve VQ’da kod tablosu modelleri kullanılır. Yapay sinir ağları, sinirsel ağaç ağları ve çok katmanlı algılayıcılarda olduğu gibi

bağlantısal sistemlerde kullanılır (Campbell et al., 2002). En başarılı yöntemler VQ, HMM and GMM'dir (Lamel and Gauvain, 1997; Thyges et al., 2000; Wan and Renals, 2002; Borah and DeLeon, 2004). Rodriguez-Porcheron and Faundez-Zanuy (1999), VQ konuşmacı tanıma sistemi ve çok katmanlı algılayıcı (MLP) sınıflandırıcının birleşiminden oluşan ve her iki yöntemin tek başına verdiği sonuçları iyileştiren yeni bir yöntem önermişlerdir.

HMM-tabanlı konuşmacı tanıma sisteminde tanınacak her bir konuşmacı bir HMM setiyle modellenir. Geleneksel konuşmacıdan bağımsız HMM'ler konuşmacı altuzayı ile ilgilenmez ve gözlem uzayında bulunan konuşma verisini dikkate alır (Ariki, 1996). HMM seti en yüksek olabirliği veren konuşmacı aranan konuşmacı kabul edilir. Konuşmacıdan bağımsız HMM genellikle birçok konuşmacının söylediği değişik tip konuşmalar kullanılarak oluşturulur. Bu durum HMM'in olasılık dağılımının düz olmasına ve tanıma hatalarına yol açan bir sorun teşkil eder (Ariki, 1996).

GMM, konuşmacı modelinin eğitiminden önce, konuşmanın fonetik ses sınıflarına bölüdüğü olasılıksal bir yaklaşımdır. GMM tabanlı konuşmacı tanıma sistemleri, yeterli uzunlukta yüksek kaliteli test konuşma girdileri kullanıldığında konuşmacı sayısının çok fazla olduğu uygulamalarda bile oldukça iyi sonuçlar vermektedir (Borah and DeLeon, 2004). GMM, sadece konuşmacıları temsil eden büyük miktarda veri olduğu durumlarda konuşmacı tanıma ve onaylamaya etkin bir şekilde uygulanabilir. Eğitim kümesinde veri sayısının az olması durumunda ise GMM'in performansı önemli ölçüde düşmektedir (Thyges et al., 2000).

Konuşmacı tanıma ve konuşmacı onaylama görevleri için destek vektör makineleri (SVM) da kullanılmaktadır (Wan and Campbell, 2000; Shriberg et al., 2004). SVM'ler yapısal risk enküçültme prensibini temel alan sınıflandırıcılardır ve sınıflandırmaya dayalı konuşmacı tanıma ve onaylama görevlerine çok uygun görünmektedirler (Wan and Renals, 2002). Bugüne kadar zaman gecikmeli sinirsel ağlar (TDNNs), radyal tabanlı işlevler (RBF), bilgi vektörü nicemleme (LVQ) ve kendini örgütleyen harita (SOM) gibi çeşitli yöntemler ileri sürülmüştür (Rodriguez-

Porcheron et al., 1999). Campbell (1997) otomatik konuşmacı tanıma sistemleri tasarımı ve geliştirilmesi konusunda bir eğitme sunmuştur. Geçtiğimiz yıllarda literatürde konuşmacı tanıma amaçlı farklı yaklaşımlar ortaya atılmıştır (Siohan et al., 1998; Quatieri et al., 2000; Campbell et al., 2002; Roberts et al., 2005).

Bu tez çalışmasında, metinden bağımsız konuşmacı tanıma için “Ortak Vektör Yaklaşımı (OVY) tabanlı GMM” olarak adlandırılan ve OVY ile GMM’in birleşiminden oluşan bir metot önerilmiştir. OVY konuşmayı tanımada (Gülmezoğlu et al., 1999; Gülmezoğlu et al., 2001) ve görüntü tanımada (Çevikalp et al., 2005; Çevikalp et al., 2006) sıkça kullanılan altuzay tabanlı bir örüntü tanıma yöntemidir. OVY’de her bir sınıfa ait eğitim setindeki öznitelik vektörleri¹ kullanılarak o sınıfa ait ortak bir vektör bulunur. Bu vektör tek bir sınıfın ortak veya değişmez özelliklerini temsil eder. OVY tabanlı GMM yönteminde öznitelik vektörleri önce OVY ile bulunan farksızlık/konuşmacı altuzayına atılır, daha sonra bu vektörler GMM’e giriş olarak uygulanır.

Konuşmacı tanıma sistemini eğitmek ve test etmek için yeterli miktarda konuşma kaydı içeren konuşma veri tabanlarına ihtiyaç vardır. Konuşmadaki değişkenlikler konuşmacı tanımayı zorlaştırır. Bazı yayınlar kayıt koşulları ile konuşmacı tanıma arasındaki ilgiyi ortaya koymuştur, ancak dilin konuşmacı tanımayla olan ilgisi iyi bilinmemektedir (Lamel and Gauvain, 1997; Alonso-Martinez et al., 2000). Standart ve kolayca temin edilebilen veri tabanları kullanmanın en görünür faydası aynı veri tabanı üzerinde farklı teknikler kullanılarak, böylece yöntemlerin ve konuşmacı tanıma protokollerinin nicel değerlendirilmesine izin verilerek elde edilen sistem başarımlarının karşılaştırılabilir olmasıdır (Campbell and Reynolds, 1999; Feng and Hansen, 2005). Buna ilave olarak standart veri tabanı kullanımı belirli görevler ve fazladan araştırma gerektiren en önemli eksiklikler için araştırma alanlarındaki mevcut en son teknolojinin başarımlarını ölçmede de kullanılabilir (Campbell and Reynolds, 1999). En gelişmiş konuşmacı tanıma ve onaylama sistemleri için gerekli konuşma verisinin asıl miktarı görevin doğasına göre değişmektedir.

¹ Herhangi bir titreşimli sinyalin ayırteci özelliklerini bu sinyalin öğeleri olarak taşıyan vektör öznitelik vektörü olarak adlandırılacaktır.

Bu tez çalışmasında önerilen yaklaşımın ve ele alınan diğer yaklaşımların başarımı TIMIT veri tabanı kullanılarak incelenmiştir. Deneysel çalışmada 20 konuşmacı tarafından söylenen konuşmalar kullanılmıştır. Konuşmacı tanıma sistemlerinin büyük bölümü uzun süreli ortalama spektrum veya öznitelik parametrelerinin dağılımı gibi statik özellikler kullanırlar (Griffin et al., 1994). LPCC ve MFCC parametrelerinin ikisi de az sayıdaki parametreler için etkin gösterimlerdir ancak, LPCC gösterimi daha iyi çalışmasına rağmen analiz derecesinin büyük olduğu durumlarda MFCC'ler daha üstündür (Liu et al., 1996).

Bu tezin ikinci bölümünde önce konuşmacı tanımada kullanılan başlıca veri tabanları açıklanmakta, sonra bir konuşmacı tanıma sisteminin yapısı, bileşenleri ve kullanım alanları konusunda genel bilgi verilmektedir. Üçüncü bölümde metinden bağımsız konuşmacı tanımada OVY yöntemi ve başarımının karşılaştırıldığı diğer yöntemler açıklanmıştır. Bu çalışmada önerilen OVY tabanlı GMM yöntemi bu bölümde detaylı olarak ele alınmıştır. Dördüncü bölümde, önce deneysel çalışmalarda oluşturulan sistemin altyapısı hakkında bilgi verilmiştir. Sonra uygulanan yöntemlerden elde edilen tanıma oranları ve hesaplama maliyetleri karşılaştırmalı olarak gösterilmiştir. Bu çalışmada elde edilen çıkarımlar sonuç bölümünde verilmektedir.

2. BÖLÜM

KONUŞMACI TANIMA

2.1 Giriş

Konuşmacı tanıma, genel konuşma işleme konusunun bir alt dalıdır. Konuşmacı belirleme ve konuşmacı onaylama, konuşmacı tanımanın kapsadığı ana konulardır.

Günümüzde güvenlik gerektiren sistemlere ulaşımın kontrol altında tutulmasında kişi belirleme ve/veya onaylama önemli bir ihtiyaçtır. Kimlik belirtme bir anahtar, şifre veya kart ile yapılabilir. Ancak bunların hepsi kaybedilebilir, çalınabilir veya kolayca kopyalanabilir. Bu nedenle biyometrik tanıma yöntemleri güvenlik gerektiren uygulamalarda öne çıkmış ve geniş araştırma alanları bulmuştur. Parmak izi tanıma, yüz tanıma ve retina tanıma başlıca biyometrik tanıma yöntemleri arasındadır. Konuşmacı tanıma da biyometrik tanıma yöntemlerinden birisidir.

Parmak izi ve retinal tanıma yöntemleri, mevcut kişiyi sesinden tanıma yöntemine göre daha güvenli gözükse de, özel donanım gerektirmeleri maliyeti arttırıcı bir unsur teşkil etmekte ve uygulama alanlarını kısıtlamaktadır.

2.2 Konuşmacı Tanıma Veri Tabanları

Konuşma veri tabanları incelenirken dikkate alınan kriterler şunlardır:

- Konuşmacı sayısı
- Konuşmacı özellikleri (Cinsiyet, yaş, vb.)
- Konuşma materyali (Sürekli konuşma, cümleler, tek kelime veya sesler)
- Kayıt ortamı

- Kayıtların alınışı arasındaki zaman dilimi

Konuşmacı belirleme ve konuşmacı onaylamada yaygın kullanılan veri tabanları şunlardır:

TIMIT : Konuşma tanıma amaçlı oluşturulmuş olmasına rağmen, konuşmacı tanımada da yaygın olarak kullanılmaktadır. Kayıtların telefon hattından geçirilmesi ile oluşturulan versiyonu NTIMIT adını alır.

KING : Konuşmacı onaylama için oluşturulmuş ilk veri tabanıdır. 51 erkek konuşmacının 10 oturumda söyledikleri 1 dakikalık monologlardan oluşur.

POLYCOST : Avrupa'nın COST250 projesi kapsamında oluşturulmuş çok dilli bir veri tabanıdır. Projede yer alan 13 ülkeden 10 konuşmacının (5 Erkek – 5 Bayan) 10 defa ISDN şebeke üzerinden telefonla aramaları sonucu elde edilen kayıtlardan oluşur. İngilizce rakamlar ve kısa cümleler ile ana dilde yapılan serbest konuşmaları içerir.

OGI : Konuşmacı tanıma araştırmaları için hazırlanmakta olan geniş kapsamlı bir veri tabanıdır. İlk sürümü 100 konuşmacıdan oluşmaktadır. Ancak bir sonraki sürümünde 600 konuşmacı olması beklenmektedir.

YOHO : Metin bağımlı konuşmacı onaylama araştırmalarını desteklemek amacıyla oluşturulmuş geniş ölçekli ve yüksek kaliteli konuşma veri tabanıdır. Veriler ABD hükümetinin isteği üzerine ITT tarafından toplanmıştır. 138 konuşmacıdan (106 Erkek, 32 Bayan) alınan üçlü rakamlardan oluşan ifadeleri içerir. Eğitim seti 4 ve test seti 10 farklı oturumda kaydedilmiştir.

TURTEL : TÜBİTAK tarafından oluşturulmuş milli konuşma veri tabanıdır. Oluşturulan veri tabanının içeriği toplam 373 kelime ve 15 cümleden meydana gelmektedir. Bu veri tabanında 65 konuşmacıdan toplanan eğitim kayıtları, 28 konuşmacıdan toplanan test kayıtları bulunmaktadır.

2.3 Konuşmacı Tanıma Sistemi

2.3.1 Konuşmacı tanıma türleri

Konuşma işleme konusunun bir alt dalı olan konuşmacı tanıma, konuşmayı tanıma kadar geniş uygulama alanları bulmuş ve özellikle GSM ve internet üzerindeki uygulamaları ayrı bir ticari önem kazanmıştır. Konuşmacı tanıma konusunda yapılan çalışmalar konuşmacı belirleme ve konuşmacı onaylama olmak üzere iki alanda toplanmıştır.

Konuşmacı belirleme, verilen bir konuşma örneğinden konuşanın kim olduğunun bulunmasıdır. “Kim konuştu?” sorusuna cevap verir. Kapalı küme ve açık küme konuşmacı belirleme şeklinde uygulanabilir. Kapalı küme kimlik belirlemede konuşma örneği, kayıtlı kullanıcılardan birine aittir. Açık kümede ise kayıtlı konuşmacılar dışında konuşmacılar da vardır.

Konuşmacı onaylama, söylenen bir sözün, kimliği verilen konuşmacıya ait olup olmadığının belirlenmesi işlemidir. “Konuşan şahıs kimliği verilen kişi mi?” sorusuna cevap verilir.

Konuşmacı belirleme ve onaylama, c adet bilinen kişinin referans örüntülerini içeren bir veri tabanı kullanır. Takip edilen analiz ve karar teknikleri de benzerdir. Ancak açıktır ki karar alternatiflerinin sayısı farklıdır. Konuşmacı onaylama daha basit bir işlemdir. Çünkü kimliğini belirten konuşmacının kabulü veya reddi söz konusudur. Dolayısı ile sistemin performansı konuşmacı sayısından etkilenmez. Konuşmacı belirlemede, bilinmeyen bir konuşmacının söylediği söz dikkate alınarak c konuşmacıdan oluşan bir küme içerisinde bilinmeyen konuşmacıya ait örüntünün en iyi uyduğu kişi bulunur. Konuşmacı onaylamada konuşan kişi aynı zamanda kimliğini belirtir ve amaç söylediği sözün iddia ettiği kişinin referans örüntüsüne yeterince benzediğinin belirlenerek kimliğinin onaylanmasıdır.

2.3.2 Metine bağımlı ve metinden bağımsız tanıma

Metine bağımlılık göz önüne alındığında konuşmacı tanıma iki alt bölüme daha ayrılır. *Metine bağımlı konuşmacı tanıma*, eğitim ve test aşamasında söylenen metinlerin aynı olmasını gerektirir. Konuşmacının zaten tanınmak istediği ve bu konuda yardımcı olduğu erişim kontrol sistemleri gibi alanlarda kullanılır. *Metinden bağımsız konuşmacı tanımada* ise konuşmacının eğitim ve test aşamasında aynı metni söyleme zorunluluğu yoktur. Adli uygulamalar gibi konuşmacının, tanıma algoritmasının eğitiminde kullanılan söylenmiş metinlerin aynısını söylemesinin garanti edilemeyeceği durumlarda kullanılır.

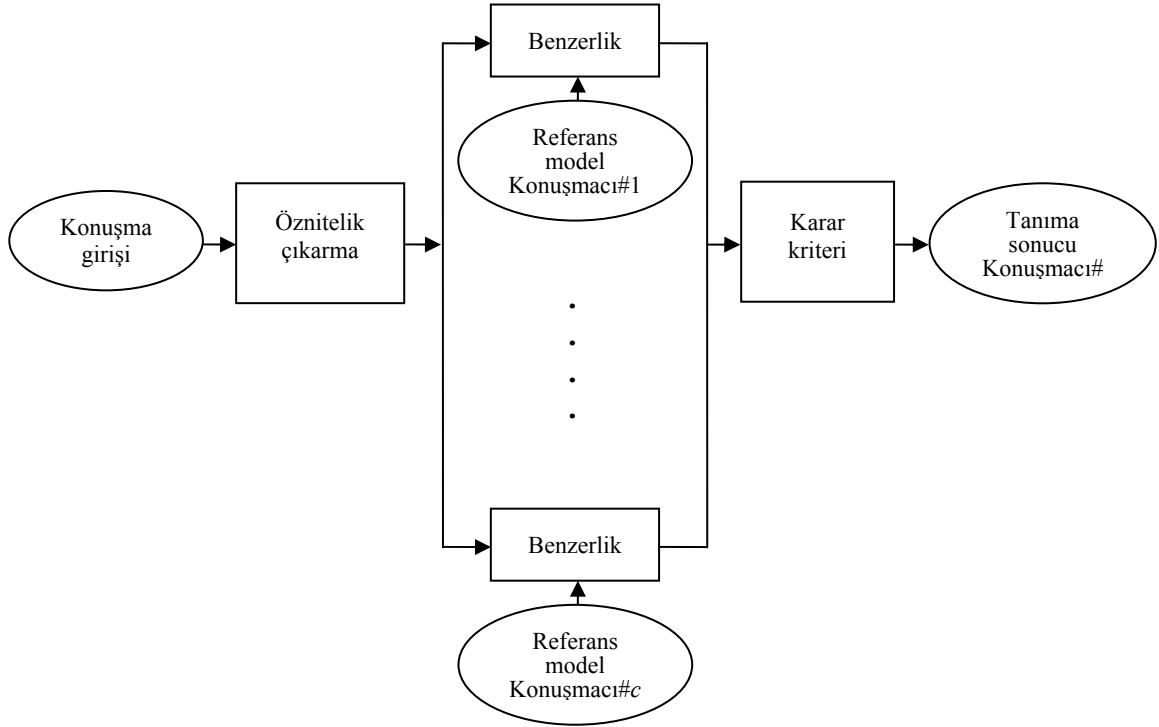
Metine bağımlı konuşmacı tanıma sistemlerinin tanıma oranı, özellikle kısa eğitim ve test söyleyişlerinde metinden bağımsız tanımaya göre daha yüksektir.

2.3.3 Konuşmacı tanıma sisteminin genel yapısı

Bilinmeyen bir konuşmacının sesini tanımak için yapılan işlemler aşağıdaki gibi özetlenebilir;

1. Eğitim kayıtlarının okunması ve özniteliklerin çıkartılması
2. Bilinen konuşmacıların öznitelik vektörlerinden bir model oluşturulması
3. Bilinmeyen konuşmacının test kümesindeki her bir konuşmasından elde edilen öznitelikler ile sistem tarafından bilinen konuşmacıların modellerinin karşılaştırılması
4. Test konuşmasının hangi konuşmacıya ait olduğuna karar verilmesi

Konuşmacı tanıma sisteminin genel yapısı Şekil 2.1’de gösterilmektedir.



Şekil 2.1 Konuşmacı tanıma sistemi genel yapısı

2.3.4 Parametre seçimi ve öznitelik çıkarma

Öznitelik çıkarma, konuşma dalga biçiminin belirli bir tipteki parametrik gösterime dönüştürülmesi işlemidir. Konuşmacı tanımada kullanılacak özniteliklerin, zamanla değişmemesi, gürültüden etkilenmemesi ve diğer konuşmacılardan kolay ayrılabilir olması istenir. Konuşmacı tanımada kullanılan başlıca parametre tipleri şunlardır;

- **Enerji / Formant frekansları / Perde frekansı** : Ses şiddeti veya enerjisi, en basit sinyal karakteristiğidir. Ancak sadece konuşma enerjisinin kullanılması başarılı olmamıştır. Formant frekanslarının ise özellikle yüksek formant bölgelerinde elde edilmesinde ve ölçülmesinde zorluklar vardır. Perde frekansı, bir sesin temel

frekansdır ve ünlü seslerin karakterize edilmesinde önemli bir parametredir. Kayıt ve iletim sistemlerinin frekans karakteristiğinden etkilenmez ve nispeten kolay elde edilir. Kötü tarafı ise kararsız oluşudur. Vurgu, entonasyon ve duygulara göre önemli ölçüde değişim gösterir.

- **Doğrusal Öngörü Katsayıları (LPC)** : Konuşma sinyalinin kısa-süreli spektral bilgisinin temsil edilmesinde en yaygın kullanılan katsayılardır ve genellikle bir filtre bankası, FFT veya LPC spektral analiz kullanılarak elde edilir. Kepstral katsayılar ve mel tabanlı kepsral katsayılar da kısa-süreli spektrum için diğer yaygın yaklaşımlardır.
- **LPC Kepstral (LPCC)** : Konuşmacı tanımda yaygın kullanılmalarına karşın LPC kepsral katsayıları ve yansıma katsayıları gürültüden oldukça etkilenebilmektedir.
- **Çizgi Spektral Çiftleri (LSP)** : Yüksek kaliteli konuşma için LSP katsayılarının konuşmacı belirlemedeki performansı kepsral katsayılarla yakın olsa da telefon kalitesindeki konuşma için kepsral katsayıların performansı çok daha iyidir.
- **Mel Frekans Kepstral Katsayıları (MFCC)** : İnsanın ses frekanslarını doğrusal olmayan bir şekilde algıladığı prensibine dayanır. Log spektrumun spektrumu olarak tanımlanabilir.

Delta Kepstrum Kepstral katsayıların zamana göre türevi alınarak elde edilir. Konuşmacının dinamik özelliklerini yansıttığı için konuşmacı tanıma ve onaylama araştırmalarında genellikle Kepstral katsayılar ile birlikte ilave özellik olarak kullanılır.

Delta Delta Kepstrum Kepstral katsayıların ikinci türevi alınarak elde edilir. Tanıma performansına katkıda bulunduğu görecelidir ve açıkça ortaya koyulamamıştır.

2.3.5 Örüntü karşılaştırma ve sınıflandırma

Girdi öznitelik vektörleri ile modeller arasındaki benzerliğin ölçülmesi ve bir uyum skoru hesaplanması amacı ile örüntü karşılaştırma yapılır. Bu skorun hesaplanması ile birlikte konuşmacı onaylama sistemlerinde kabul / ret kararı verilir veya yeni bir girdi istenebilir. Konuşmacı belirleme sistemlerinde ise skora bakılarak en yüksek skora sahip model konuşmacı seçilir veya skor belli bir eşik değerini geçmiyorsa konuşmacının grup dışından olduğuna karar verilebilir. Kullanılan sınıflayıcıya göre bazen en düşük skora bakmak gerekebilir.

2.4 Konuşmacı Tanımının Kullanıldığı Alanlar

Konuşmacı tanımının kullanıldığı başlıca alanlar aşağıda özetlenmiştir;

- Erişim kontrolü (İnternet bankacılığı, telefon bankacılığı, güvenli geçiş kontrolü)
- Çağrı merkezleri için ilave müşteri bilgisi
- Bilgisayarlara uzaktan erişim
- Sesli telefon numarası çevirme
- Adli soruşturmalarda şüpheli şahıs tespiti

3. BÖLÜM

KONUŞMACI TANIMADA KULLANILAN YÖNTEMLER

3.1 Giriş

Bu bölümde, bilinen yöntemler olan FLDA ve GMM, yakın zamanda ortaya çıkmış olan OVY ve DCV anlatılmıştır. Ayrıca yeni önerilmiş bir yöntem olan OVY tabanlı GMM açıklanmıştır.

3.2 Fisher'in Doğrusal Ayırtaç Analizi (FLDA)

FLDA, ayırt ediciliği en yüksek olan altuzay yöntemlerine bir örnektir. Bu yöntem, konuşmacıyı diğerlerinden ayıracak en iyi öznelik setini arar. Öznelik uzayının yüksek boyutlu olmasından dolayı, FLDA önce PCA kullanarak öznelik vektörlerini *konuşmacı uzayı* olarak adlandırılan daha düşük boyutlu uzaya atar ve sonra ayırt ediciliği arttırmak için LDA kullanır. Bu metot sınıflar arası saçılım matrisinin toplam sınıf içi saçılım matrisine oranını en fazla kılacak şekilde \mathbf{W} izdüşüm matrisini bulur (Belhumeur et al., 1997).

Sınıflar arası saçılım matrisi,

$$\mathbf{S}_B = \sum_{j=1}^c N_j (\boldsymbol{\mu}_j - \boldsymbol{\mu})(\boldsymbol{\mu}_j - \boldsymbol{\mu})^T, \quad (3.1)$$

olarak tanımlanır. Burada N_j , X_j sınıfındaki öznelik vektör sayısı, $\boldsymbol{\mu}_j$ bu sınıftaki öznelik vektörlerinin ortalaması ve $\boldsymbol{\mu}$ ise bütün sınıflardaki öznelik vektörlerinin ortalamasını gösterir. Toplam sınıf içi saçılım matrisi de aşağıdaki gibi tanımlanır;

$$\mathbf{S}_w = \sum_{j=1}^c \boldsymbol{\Phi}_j, \quad (3.2)$$

burada Φ_j , X_j sınıfına ait saçılım matrisidir:

$$\Phi_j = \sum_{\mathbf{x}_k \in X_j} (\mathbf{x}_k - \boldsymbol{\mu}_j)(\mathbf{x}_k - \boldsymbol{\mu}_j)^T, \quad (3.3)$$

\mathbf{x}_k lar ise X_j sınıfına ait d -boyutlu öznitelik vektörleridir.

Eğer \mathbf{S}_w 'nin tersi var ise, optimal izdüşüm matrisi \mathbf{W}_{opt}

$$\mathbf{W}_{\text{opt}} = \arg \max_{\mathbf{W}} \frac{|\mathbf{W}^T \mathbf{S}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_w \mathbf{W}|} = [\mathbf{w}_1 \ \mathbf{w}_2 \ \cdots \ \mathbf{w}_m], \quad (3.4)$$

olarak seçilebilir. Burada $\{\mathbf{w}_i | i=1,2,\dots,m\}$, $\mathbf{S}_w^{-1} \mathbf{S}_B$ 'in en büyük m genelleştirilmiş özdeğerine $\{\lambda_i | i=1,2,\dots,m\}$ karşılık gelen genelleştirilmiş özvektörler kümesidir.

$$\mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{S}_w \mathbf{w}_i \quad i=1,2,\dots,m. \quad (3.5)$$

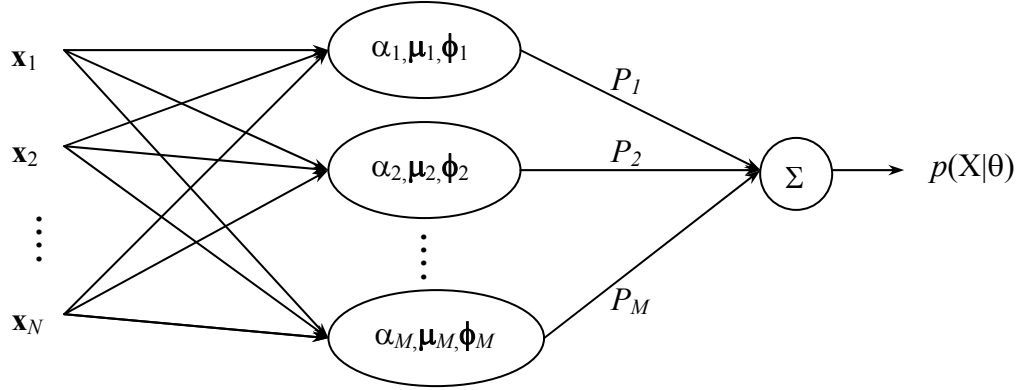
Konuşmacı (sınıf) sayısını c ile gösterirsek, sıfırdan farklı en fazla $(c-1)$ tane genelleştirilmiş özdeğer vardır. Bundan dolayı m in üst sınırı $(c-1)$ 'dir.

3.3 Gauss Katılım Modelleri (GMM)

Bu yöntemin temel prensibi, birden çok Gauss yoğunluk işlevi kullanılarak bir konuşmacıya ait öznitelik vektörlerinden kişinin akustik niteliklerini temsil eden olasılık yoğunluk işlevlerinin bulunmasıdır. Bu yöntemde her bir konuşmacı Şekil 3.1'de görüleceği gibi M adet Gauss yoğunluk işlevi ile tanımlanır.

Bir X_j konuşmacı sınıfına ait veri, aşağıda verilen çok değişkenli Gauss olasılık yoğunluk işlevi (pdf) ile modellenebilir,

$$b_i(\mathbf{x}_k | \boldsymbol{\mu}_i, \Phi_i) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Phi_i|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}_k - \boldsymbol{\mu}_i)^T \Phi_i^{-1} (\mathbf{x}_k - \boldsymbol{\mu}_i)\right). \quad (3.6)$$



Şekil 3.1 Bir konuşmacıya ait tüm Gauss yoğunluk işlevleri

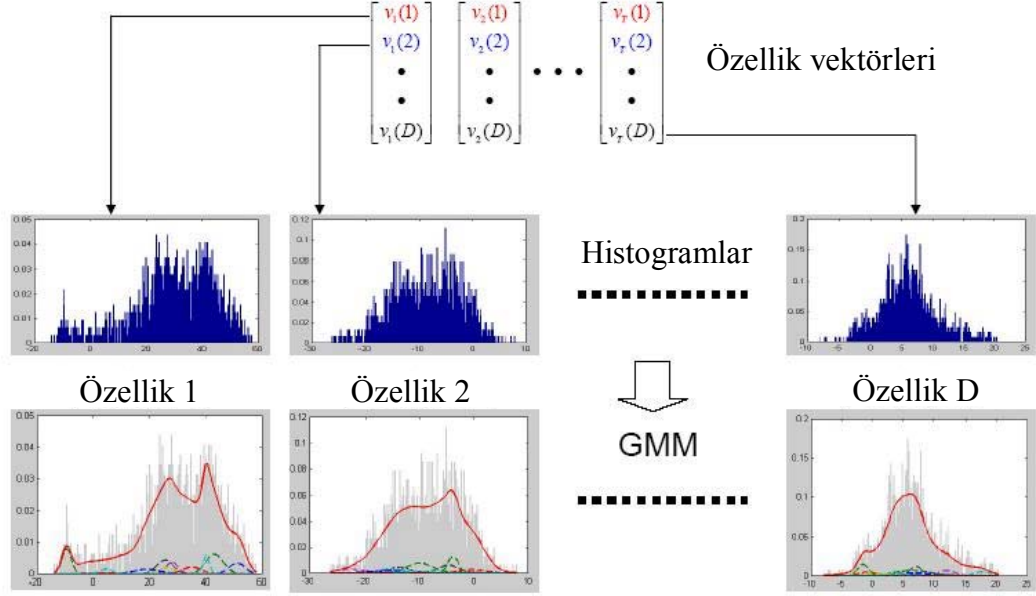
Gauss dağılımının en büyük avantajı basit oluşudur. Bununla beraber, Gauss dağılımı çok kipli bir dağılımı modelleyemez (genellikle pratikteki durum budur), bu nedenle Gauss katışımı daha fazla modelleme esnekliğine sahiptir (Lyu, 2005). Bu yolla yeterli sayıda bileşene sahip herhangi bir olasılık yoğunluğu herhangi bir derecedeki Gauss katışımına yakınlştırılabilir. Genellikle sonlu bir Gauss katışım yoğunluğu aşağıdaki gibi tanımlanır;

$$p(\mathbf{x}_k) = \sum_{i=1}^M \alpha_i b_i(\mathbf{x}_k | \boldsymbol{\mu}_i, \boldsymbol{\Phi}_i), \quad (3.7)$$

Burada M , bileşen sayısı ve $\alpha_1, \alpha_2, \dots, \alpha_M$, $\sum_{i=1}^M \alpha_i = 1$ koşulunu sağlayan katışım ağırlık katsayılarıdır. Ayrıca $i = 1, 2, \dots, M$ için $\alpha_i \geq 0$ 'dir. $\boldsymbol{\mu}_i$ ve $\boldsymbol{\Phi}_i$ ise i 'nci Gauss katışımının ortalaması ve saçılımıdır. *Logaritmik-olabilirlik* (L_L) bir modelin deneysel bir veriye ne kadar uyduğunu ölçmek için kullanılır ve aşağıdaki eşitlik ile ifade edilir:

$$L_L = \frac{1}{N} \log \prod_{k=1}^N p(\mathbf{x}_k) = \frac{1}{N} \sum_{k=1}^N \log p(\mathbf{x}_k) \quad (3.8)$$

Şekil 3.2'de GMM yöntemi ile bir konuşmacının nasıl modellendiği gösterilmektedir.



Şekil 3.2 GMM yönteminde bir konuşmacının modellenmesi

Katışımın ağırlık katsayıları, ortalamaları ve saçılım matrisleri, bileşen sayısı M 'in bilinmesi durumunda beklenti-embüytme (EM) algoritması kullanılarak X_j sınıfının öznelik vektör kümesinden bulunabilir. Bu parametrelerin başlangıç değerlerinden yola çıkılarak yakınsama sağlanıncaya kadar aşağıdaki adımlar izlenerek EM algoritması uygulanır. Bir başka deyişle E ve M adımları, iki döngü arasındaki kestirimlerdeki en büyük değişim yakınsama kriterinin altına düşünceye kadar tekrarlanır. Bu işlemin çıktıları ortalama vektör ve saçılım matrisidir.

Sonsal olasılık aşağıdaki eşitlik ile gösterildiğinde,

$$p_i(k) = \frac{\alpha_i b_i(\mathbf{x}_k | \boldsymbol{\mu}_i, \boldsymbol{\Phi}_i)}{\sum_{j=1}^M \alpha_j b_j(\mathbf{x}_k | \boldsymbol{\mu}_j, \boldsymbol{\Phi}_j)} \quad i = 1, \dots, M, \quad k = 1, \dots, N \quad (3.9)$$

i 'ninci bileşen (katılım) için bir sonraki döngüdeki model parametreleri

$$\alpha_i = \frac{1}{N} \sum_{k=1}^N p_i(k) \quad (3.10)$$

$$\boldsymbol{\mu}_i = \frac{\sum_{k=1}^N \mathbf{x}_k p_i(k)}{\sum_{k=1}^N p_i(k)} \quad (3.11)$$

$$\boldsymbol{\Phi}_i = \frac{\sum_{k=1}^N (\mathbf{x}_k - \boldsymbol{\mu}_i)(\mathbf{x}_k - \boldsymbol{\mu}_i)^T p_i(k)}{\sum_{k=1}^N p_i(k)} \quad (3.12)$$

olarak ifade edilebilir.

EM algoritması, bir sonraki adımda daha yüksek logaritmik-olabilirliğe sahip bir model bulmayı ve sonlu adımda, sağlanan verinin logaritmik-olabilirlik işlevinin yerel en büyüğe yakınsamasını garanti eder.

3.4 Ortak Vektör Yaklaşımı (OVY)

Ortak vektör yaklaşımı, konuşma ve örüntü tanımada tatmin edici sonuçlar veren altuzay tabanlı örüntü tanıma yöntemidir (Gülmezoğlu et al.,1999; Gülmezoğlu et al.,2001; Çevikalp et al., 2005). Ortak Vektör Yaklaşımı'nda hedef, bir öznelik vektörleri seti içerisinde, konuşmacıdan ve ortamdan kaynaklanabilecek farklılıkların uzaklaştırılarak her bir sınıfı temsil eden tek ve değişmeyen bir vektör bulmaktır. Ortak vektör yaklaşımının uygulanmasında eldeki öznelik vektörlerinin sayısının öznelik vektör boyutundan küçük veya büyük olduğu iki durum mevcuttur.

3.4.1 Yetersiz veri durumunda OVY

Eğitim setinin, her biri N adet vektör içeren, c farklı sınıftan oluştuğunu varsayalım. Sınıfı j olan k 'ninci öznelik vektörünü d -boyutlu uzayda \mathbf{x}_k ile

gösterelim. Eğer vektör sayısı N , öznitelik vektör boyutu d 'den küçükse buna *yetersiz veri durumu* denilir.

Yetersiz veri durumunda önce X_j sınıfa ait saçılım matrisi bulunur.

$$\Phi_j = \sum_{\mathbf{x}_k \in X_j} (\mathbf{x}_k - \boldsymbol{\mu}_j)(\mathbf{x}_k - \boldsymbol{\mu}_j)^T \quad k = 1, \dots, N, \quad j = 1, \dots, c \quad (3.13)$$

Φ_j saçılım matrisinin özdeğerleri ($\lambda_i | i = 1, 2, \dots, d$) büyükten küçüğe doğru sıralandığında, bunlara karşılık gelen özvektörler ($\mathbf{u}_i | i = 1, 2, \dots, d$) olsun. Bütün özvektörler tarafından gerilen d -boyutlu öznitelik uzayı, sıfırdan farklı özdeğerlere karşılık gelen özvektörler tarafından gerilen z -boyutlu farklılık altuzayı Q ve sıfır özdeğerlere karşılık gelen $(d-z)$ boyutlu farksızlık/konuşmacı altuzayı \bar{Q} olarak ikiye ayrılabilir. X_j sınıfındaki herhangi bir öznitelik vektörü \mathbf{x}_k 'nin farksızlık altuzayına izdüşümü o sınıfa ait *ortak vektörü* verir.

$$\mathbf{x}^{com} = \sum_{i=z+1}^d (\mathbf{x}_k^T \mathbf{u}_i) \mathbf{u}_i \quad (3.14)$$

Konuşmacı tanıma sürecinde, bilinmeyen bir konuşmacıya ait \mathbf{x}_t öznitelik vektörünün, her sınıfa ait farksızlık altuzayına izdüşümü alınarak o sınıfa ait *kalan vektör* bulunur.

$$\mathbf{x}_t^{rem} = \sum_{i=z+1}^d (\mathbf{x}_t^T \mathbf{u}_i) \mathbf{u}_i \quad (3.15)$$

Her sınıfa ait ortak vektör ile kalan vektör arasındaki öklid uzaklığı karar kriteri olarak kullanılır.

$$c^* = \arg \min_{1 \leq j \leq c} \|\mathbf{x}_t^{rem} - \mathbf{x}_j^{com}\| \quad (3.16)$$

Hangi X_j sınıfı için \mathbf{x}_i^{rem} ve \mathbf{x}^{com} arasındaki uzaklık en küçük ise, \mathbf{x}_i öznitelik vektörü o sınıfa aittir.

3.4.2 Yeterli veri durumunda OVY

OVY’de, eğitim için kullanılan öznitelik vektörü sayısı öznitelik vektörlerinin boyutundan büyük ise ($N > d$) buna *yeterli veri durumu* denilir. Yeterli veri durumunda da önce X_j sınıfına ait eğitim verisinin saçılım matrisine özdeğer-özvektör ayrıştırması yapılır. Buradan elde edilen özdeğerlerin hepsi sıfırdan büyüktür. Φ_j ’nin özdeğerlerinin ($\lambda_i | i = 1, 2, \dots, d$) büyükten küçüğe sıralandığını varsayalım. Bütün öznitelik vektörleri tarafından gerilen d boyutlu öznitelik uzayı, m -boyutlu farklılık altuzayı Q ve $(d-m)$ boyutlu dikgen farksızlık altuzayı \bar{Q} şeklinde ikiye ayrılabilir. Farklılık altuzayı Q , en büyük özdeğerlere karşılık gelen özvektörler ($\mathbf{u}_i, i=1, 2, \dots, m$) tarafından ve farksızlık altuzayı \bar{Q} ’de en küçük özdeğerlere karşılık gelen özvektörler ($\mathbf{u}_i, i=m+1, \dots, d$) tarafından gerilir (Gülmezoğlu et al., 2007). Bu iki altuzayın doğrudan toplamı bütün öznitelik uzayını içine alır (Gülmezoğlu et al., 2007). Öznitelik uzayının tamamını iki altuzaya ayrıştırmanın amacı, uzayın tamamının ortalamadan büyük sapmalara sahip kısımlarını uzaklaştırmaktır (Landgrebe, 2002).

Yeterli veri durumunda, en küçük özdeğerlerin toplamının bütün özdeğerlerin toplamına olan oranı belli bir L yüzdesinden küçük olacak şekilde m değerini belirleyebiliriz (Oja, 1983). Böylece m değeri aşağıdaki eşitsizlik kullanılarak elde edilebilir:

$$\frac{\sum_{i=m+1}^d \lambda_i}{\sum_{i=1}^d \lambda_i} < L. \quad (3.17)$$

Eğer $L=0.16$ seçilirse, orijinal öznitelik vektör uzayındaki değişimin büyük bir kısmını koruyarak, özniteliklerin sayısında makul bir indirim gidilebilir (Swets and Weng, 1996).

Eđitim verilerinin özdeđerleri azalan řekilde çizildiđinde özdeđerlerin yavař yavař deđiřmeye bařladıđı noktaya bakılarak da m deđeri belirlenebilir (Gölmezođlu et al., 2007).

X_j sınıfına ait $\boldsymbol{\mu}$ ortalama vektörünün \bar{Q} farksızlık altuzayına dikgen izdüřümü, bu sınıfın \mathbf{x}^{com} ortak vektörünü verir.

$$\mathbf{x}^{\text{com}} = \sum_{i=m+1}^d (\boldsymbol{\mu}^T \mathbf{u}_i) \mathbf{u}_i, \quad (3.18)$$

Burada \mathbf{u}_i 'ler öznitelik vektörlerindeki deđiřimin, farklılık altuzayındakinden daha küçük olduđu \bar{Q} farksızlık altuzayının özvektörlerini temsil eder.

Herhangi bir öznitelik vektörü $\mathbf{x}_k \in X_j$ 'nin Q farklılık uzayı üzerine izdüřümü hem konuşmacı içi ve konuşmacılar arası deđiřimleri hem de çevresel etkileri temsil eder. Herhangi bir sınıfın öznitelik vektörlerinin farksızlık altuzayına izdüřümü o sınıfın ortak vektörüne daha yakın olacaktır.

3.5 Ayırtedici Ortak Vektör Yaklařımı (DCV)

Ayırtedici Ortak Vektör yönteminde, her bir farklı sınıfı temsil eden ayırtedici ortak vektörler, toplam sınıf içi saçılım matrisinin sıfır altuzayından seçilen izdüřüm vektörleri kullanılarak elde edilmiř ve konuşmacıların sınıflandırılmasında kullanılmıřlardır (Çevikalp et al., 2005).

Eđitim setinin her biri N vektör içeren, c farklı sınıftan oluřtuđunu varsayalım. Bu durumda eđitim setinde toplam $N_T = N.c$ vektör olacaktır. Sınıfı j olan k 'ninci öznitelik vektörünü d -boyutlu uzayda x_k ile gösterirsek, \mathbf{S}_w matrisi ařađıdaki eřitlik kullanılarak bulunabilir:

$$\mathbf{S}_w = \sum_{j=1}^c \sum_{k=1}^N (\mathbf{x}_k - \boldsymbol{\mu}_j)(\mathbf{x}_k - \boldsymbol{\mu}_j)^T = \mathbf{A}\mathbf{A}^T \quad (3.19)$$

Bu eşitlikte $\boldsymbol{\mu}_j$, X_j sınıfa ait ortalama vektörü göstermektedir. \mathbf{A} ise dxN_T boyutlu matris olup, aşağıdaki eşitlikte verildiği gibidir.

$$\mathbf{A} = \begin{bmatrix} \mathbf{x}_{1,1} - \boldsymbol{\mu}_1 & \dots & \mathbf{x}_{N,1} - \boldsymbol{\mu}_1 & \mathbf{x}_{1,2} - \boldsymbol{\mu}_2 & \dots & \mathbf{x}_{N,c} - \boldsymbol{\mu}_c \end{bmatrix} \quad (3.20)$$

\mathbf{S}_w matrisinin sıfırdan farklı özdeğerlerine karşılık gelen özvektörlerini kullanarak,

$$\mathbf{Q} = [\mathbf{u}_1 \quad \dots \quad \mathbf{u}_m] \quad (3.21)$$

matrisini oluşturabiliriz. Bu eşitlikte \mathbf{u} , \mathbf{S}_w matrisinin erim altuzayını doğuran vektörleri, m ise \mathbf{S}_w matrisinin kertesini ifade etmektedir. \mathbf{P} ve $\bar{\mathbf{P}}$ matrisleri sırasıyla \mathbf{S}_w matrisinin erim ve sıfır altuzaylarının izdüşüm matrisleri olarak alınır, eğitim setindeki vektörlerin sıfır altuzayındaki izdüşümleri aşağıdaki gibi olacaktır:

$$\mathbf{x}^{com} = \mathbf{x}_k - \mathbf{Q}\mathbf{Q}^T \mathbf{x}_k, \quad k = 1, \dots, N \quad (3.22)$$

$$\mathbf{x}^{com} = \mathbf{x}_k - \mathbf{P}\mathbf{x}_k = \bar{\mathbf{P}}\mathbf{x}_k \quad (3.23)$$

Bu işlem sonucunda her sınıftaki herhangi bir örnek, o sınıfı temsil eden ortak bir vektör üretir.

Optimal izdüşüm vektörleri ortak vektörlerin toplam saçılımını enbüyüten vektörler olacaktır. Başka bir deyişle,

$$\mathbf{W}_{opt} = \arg \max_{|\mathbf{W}^T \mathbf{S}_w \mathbf{W}| \rightarrow 0} |\mathbf{W}^T \mathbf{S}_B \mathbf{W}| = \arg \max |\mathbf{W}^T \mathbf{S}_{com} \mathbf{W}| \quad (3.24)$$

Bu eşitlikte \mathbf{S}_{com} ortak vektörlere ait saçılım matrisi olup, aşağıdaki eşitlik kullanılarak bulunabilir:

$$\mathbf{S}_{\text{com}} = \sum_{j=1}^c (\mathbf{x}_j^{\text{com}} - \boldsymbol{\mu}^{\text{com}})(\mathbf{x}_j^{\text{com}} - \boldsymbol{\mu}^{\text{com}})^T, \quad j = 1, \dots, c \quad (3.25)$$

burada $\boldsymbol{\mu}^{\text{com}}$ ortak vektörlere ait ortalama vektördür.

\mathbf{S}_{com} matrisinin sıfırdan farklı özdeğerlerine karşılık gelen özvektörler, optimal izdüşüm vektörlerini verir. Optimal izdüşüm matrisi \mathbf{W} kullanılarak, ayırteci ortak vektörler;

$$\boldsymbol{\Omega} = [\langle \mathbf{x}_k, \mathbf{u}_1 \rangle \quad \dots \quad \langle \mathbf{x}_k, \mathbf{u}_m \rangle]^T \quad (3.26)$$

$$\boldsymbol{\Omega} = \mathbf{W}^T \mathbf{x}_k \quad (3.27)$$

olarak bulunur.

Test öznelik vektörleri ise,

$$\boldsymbol{\Omega}_{\text{test}} = \mathbf{W}^T \mathbf{x}_{\text{test}} \quad (3.28)$$

eşitliğinden bulunabilir.

Daha sonra $\boldsymbol{\Omega}_{\text{test}}$ ile eğitim setindeki sınıflara ait ayırteci ortak vektörlerin arasındaki Öklid uzaklığına bakılır.

$$c^* = \arg \min_j \|\boldsymbol{\Omega}_{\text{test}} - \boldsymbol{\Omega}_j\| \quad (3.29)$$

Test konuşması, en küçük uzaklığı veren konuşmacıya atanır.

3.6 OVY Tabanlı GMM

GMM yönteminin başarımı, eğitim ve test sürelerinin yetersiz olduğu durumlarda önemli ölçüde düşmektedir. Bu nedenle *OVY tabanlı GMM* adını verdiğimiz ve GMM'in başarısını bu gibi durumlarda iyileştirecek yeni bir yöntem önerdik. Önerilen yöntemde bir konuşmacıya ait konuşma verisinden, OVY uygulanarak zaman, çevresel koşullar ve fonetik bilgideki değişikliklerden kaynaklanan farklılıklar uzaklaştırılmakta ve konuşmacıya ait asıl öznitelikler elde edilmektedir.

OVY tabanlı GMM yöntemini uygulamak için önce OVY kullanılarak her sınıfın \bar{Q} farksızlık altuzayı oluşturulur. Sonra her \mathbf{X}_j sınıfının eğitim kümesindeki \mathbf{x}_k öznitelik vektörlerinin o sınıfın farksızlık altuzayına izdüşümü alınır.

$$\tilde{\mathbf{x}}_k = \mathbf{W}^T \mathbf{x}_k \quad (3.30)$$

Burada \mathbf{W} transformasyon matrisinin kolonları \bar{Q} farksızlık altuzayını oluşturan özvektörlerdir. Elde edilen izdüşüm vektörleri yeni öznitelik vektörleri olarak kabul edilir ve GMM'e giriş olarak uygulanır. Böylece izdüşüm alınarak elde edilen $\tilde{\mathbf{x}}_k$ öznitelik vektörleriyle GMM'den elde edilen yeni istatistiksel konuşmacı modeli, konuşmacıyı orijinal \mathbf{x}_k öznitelik vektörlerinden elde edilen modele göre çok daha iyi temsil eder.

GMM'de verilen bir test girdisi için her konuşmacının logaritmik-olabilirliği hesaplandığında, en büyük logaritmik-olabilirliğe sahip konuşmacı, aranan konuşmacı olacaktır. Referans konuşmacı kümesi, $\theta_1, \theta_2, \dots, \theta_c$ modelleri ile temsil edildiğinde öncelikli amaç, bir giriş öznitelik vektör dizisi için en büyük sonsal olasılığa ($P(\theta_j | X)$) sahip konuşmacı modelini bulmaktır (Reynolds, 1995). X 'deki öznitelik vektörlerinin, farksızlık altuzayına izdüşümü alınması ve \tilde{X} öznitelik vektör dizisinin elde edilmesi durumunda, bu problem aşağıdaki eşitlikle ifade edilebilir:

$$c^* = \arg \max_j P(\theta_j | \tilde{X}) \cong \arg \max_j P(\tilde{X} | \theta_j) \quad j=1, \dots, c \quad (3.31)$$

Burada $P(\tilde{X} | \theta_j)$, j 'nci sınıf şartlı olasılık yoğunluk işlevini göstermektedir. j 'nci konuşmacı modelinin parametreleri $\theta_j = \{\alpha_i, \mu_i, \Phi_i\}$, $i=1, \dots, M$ ile gösterilmiştir. Bu ifadenin logaritması alınır ve gözlemlerin (öznitelik vektörleri) birbirinden bağımsız olduğu kabul edilirse,

$$c^* = \arg \max_j \sum_{k=1}^N \log p(\tilde{\mathbf{x}}_k | \theta_j) \quad (3.32)$$

olur.

Burada $p(\tilde{\mathbf{x}}_k | \theta_j)$ ile ifade edilen ve daha önce Eş.3.7'de verilen Gauss katışım yoğunluğu, j 'nci konuşmacı için Gauss dağılımlarının bir katışımıdır. Bir başka deyişle $p(\tilde{\mathbf{x}}_k | \theta_j)$, M bileşenli $b_i(\tilde{\mathbf{x}}_k | \tilde{\mu}_i, \Phi_i)$ ile ifade edilen Gauss olasılık yoğunluk işlevlerinin (pdf) ağırlıklanmış doğrusal birleşiminden oluşur:

$$p(\tilde{\mathbf{x}}_k | \theta_j) = \sum_{i=1}^M \alpha_i b_{i,j}(\tilde{\mathbf{x}}_k | \tilde{\mu}_{i,j}, \Phi_{i,j}) \quad (3.33)$$

Eş.3.6'da verilen Gauss olasılık yoğunluk işlevi $b_i(\tilde{\mathbf{x}}_k | \tilde{\mu}_i, \Phi_i)$, Eş.3.33'de yerine konulursa ve elde edilen ifade de Eş.3.32'de kullanılırsa,

$$c^* = \arg \max_j \sum_{k=1}^N \log \left\{ \sum_{i=1}^M \alpha_{i,j} \frac{1}{(2\pi)^{\frac{d-m}{2}} |\Phi_{i,j}|^{\frac{1}{2}}} \exp \left(-\frac{1}{2} (\tilde{\mathbf{x}}_k - \tilde{\mu}_{i,j})^T \Phi_{i,j}^{-1} (\tilde{\mathbf{x}}_k - \tilde{\mu}_{i,j}) \right) \right\} \quad (3.34)$$

olur. Burada $\tilde{\boldsymbol{\mu}}_{i,j} = \mathbf{W}_j^T \boldsymbol{\mu}_{i,j}$ eşitliği ile ifade edilebilir ve ortak vektör \mathbf{x}^{com} ile aynı işleve sahip olduğu düşünülebilir. $\boldsymbol{\mu}_{i,j}$ ise, X_j sınıfının i 'nci katışımındaki ortalamasıdır. $\boldsymbol{\mu}_{i,j}$ 'nin M katışım için ortalaması, OVY'deki $\boldsymbol{\mu}_j$ ile aynı rolü oynar.

Exponensiyel ifadenin önündeki bölümde Φ 'lerin büyüklüğe etkisinin az olduğu kabullenmesi yapılırsa logaritmik-olabilirliğin enbüyütülmesi, exponensiyel ifadedeki $(\tilde{\mathbf{x}}_k - \tilde{\boldsymbol{\mu}}_j)$ 'nin normunun en küçük yapılmasına veya öklid uzaklığına karşılık gelir. Böylece metriğimiz;

$$F = \sum_{k=1}^N \|(\tilde{\mathbf{x}}_k - \tilde{\boldsymbol{\mu}}_j)\|^2 = \sum_{k=1}^N \|W^T (\mathbf{x}_k - \boldsymbol{\mu}_j)\|^2 \quad (3.35)$$

$$F = \sum_{k=1}^N \|W^T \mathbf{x}_k - W^T \boldsymbol{\mu}_j\|^2 \quad (3.36)$$

$$F = \sum_{k=1}^N \|\mathbf{x}_k - \mathbf{x}_{k,dif} - \mathbf{x}_j^{com}\|^2 \quad (3.37)$$

olur ki bu (Eş.3.37), yeterli veri durumunda OVY için kullanılan metrik ile aynıdır.

Bir sınıfın eğitim verisinin $X = \{\mathbf{x}_k \mid \mathbf{x}_k \in R^d, 1 \leq k \leq N\}$ kümesi olduğunu ve eğitim kümesindeki vektörlerin sayısının (N), öznelik vektörlerinin boyutundan (d) büyük olduğunu kabul edelim. R^d uzayı, biri \bar{Q} farksızlık altuzayı diğeri kertes m olan Q farklılık altuzayı olacak şekilde birbirine dik iki vektör kümesine ayrılabilir. X 'deki bütün vektörler, $\mathbf{x}_{k,dif} \in Q$ ve $\tilde{\mathbf{x}}_k \in \bar{Q}$ olmak üzere $\mathbf{x}_k = \tilde{\mathbf{x}}_k + \mathbf{x}_{k,dif}$ olarak yazılabilir. Konuşmacının değişmeyen karakteristiklerini ifade eden \mathbf{x}^{com} 'un, bütün $\tilde{\mathbf{x}}_k$ 'ların ortalaması olduğu varsayımı yapılırsa,

$$\mathbf{x}^{com} = W^T \boldsymbol{\mu} = W^T \left(\frac{1}{N} \sum_{k=1}^N \mathbf{x}_k \right) \quad (3.38)$$

F metriğini en küçük yapmış oluruz:

$$F = \sum_{k=1}^N \left\| \tilde{\mathbf{x}}_k - \mathbf{x}^{com} \right\|^2 \quad (3.39)$$

Q farklılık altuzayının, $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m\}$ birimlik taban vektörleri ve \bar{Q} farksızlık altuzayının $\{\mathbf{u}_{m+1}, \mathbf{u}_{m+2}, \dots, \mathbf{u}_d\}$ birimlik taban vektörleri tarafından gerilen altuzaylar olduğunu kabul edelim. Böylece $\tilde{\mathbf{x}}_k = \sum_{i=m+1}^d \langle \mathbf{x}_k, \mathbf{u}_i \rangle \mathbf{u}_i$ ve $\mathbf{x}^{com} = \sum_{i=m+1}^d \langle \boldsymbol{\mu}, \mathbf{u}_i \rangle \mathbf{u}_i$ olur.

F metriği tekrar yazılırsa:

$$F = \sum_{k=1}^N \sum_{i=m+1}^d \left\langle (\mathbf{x}_k - \boldsymbol{\mu}), \mathbf{u}_i \right\rangle^2 \quad (3.40)$$

olur. $i=m+1, \dots, d$ için $\|\mathbf{u}_i\|=1$ sınırlaması altında \mathbf{u}_i 'ler dikkate alınarak, F metriğinin en küçük yapılması ile farklılık ve farksızlık altuzaylarının \mathbf{u}_i özvektörleri, saçılım matrisi Φ 'nin özvektörlerine dönüşür. En küçük yapılmasından sonra F metriği:

$$F_{\min} = \sum_{i=m+1}^d \mathbf{u}_i^T \Phi \mathbf{u}_i = (\lambda_{m+1} + \lambda_{m+2} + \dots + \lambda_d) \quad (3.41)$$

olur. Burada $\lambda_{m+1}, \lambda_{m+2}, \dots, \lambda_d$, saçılım matrisi Φ 'nin en küçük özdeğerleri, $\mathbf{u}_{m+1}, \mathbf{u}_{m+2}, \dots, \mathbf{u}_d$ ise Φ 'nin bu özdeğerlere karşılık gelen özvektörleridir. Geleneksel GMM yöntemi kullanıldığında F metriği,

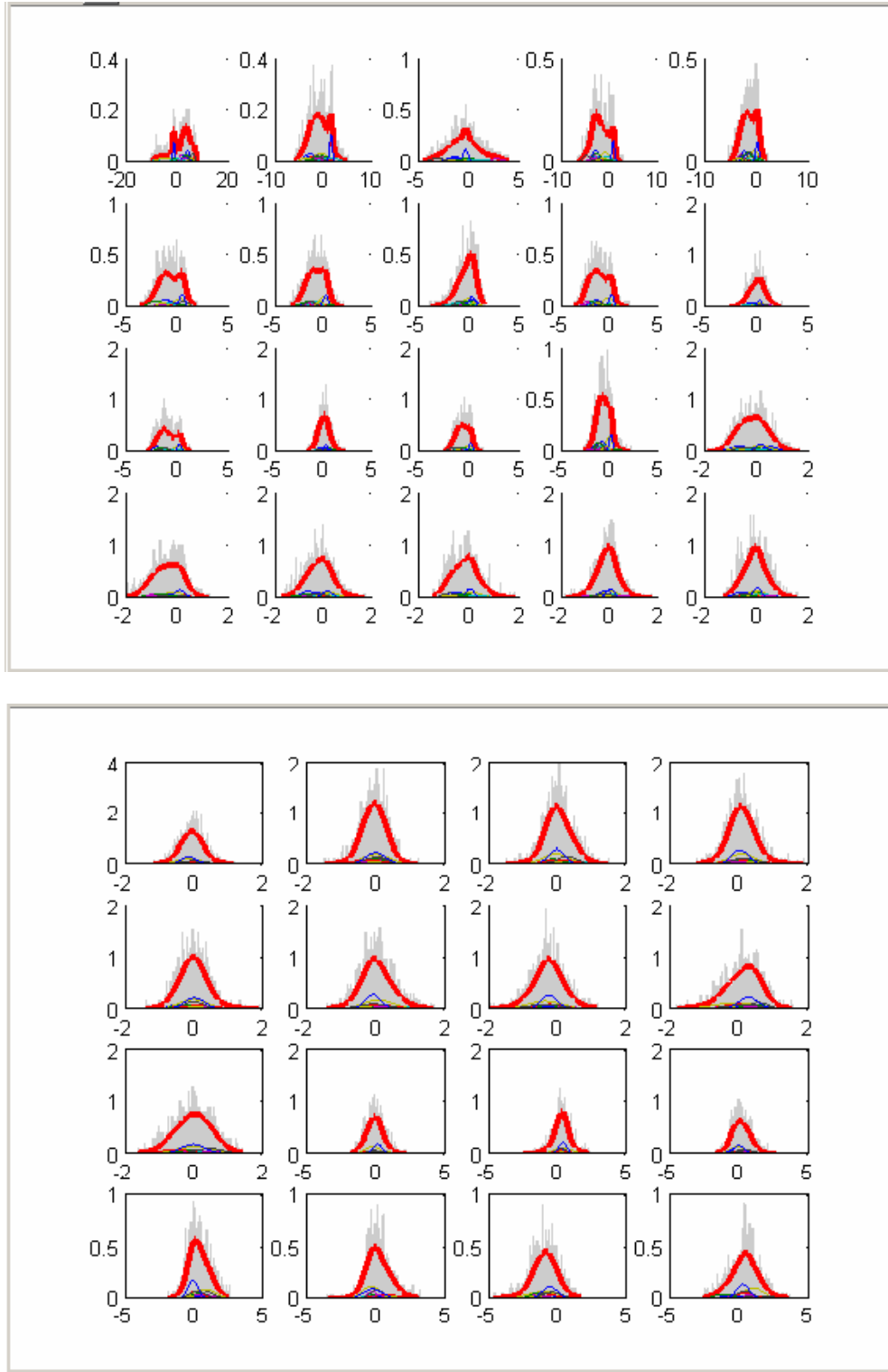
$$F_1 = \sum_{k=1}^N \left\| (\mathbf{x}_k - \boldsymbol{\mu}_j) \right\|^2 \quad (3.42)$$

olarak yazılabilir ve F_1 'in en küçük yapılması ile,

$$F_{1,\min} = \sum_{i=1}^d \mathbf{u}_i^T \Phi \mathbf{u}_i = (\lambda_1 + \lambda_2 + \dots + \lambda_d) \quad (3.43)$$

elde edilir. Böylece OVY tabanlı GMM'in F_{\min} 'i, GMM'in $F_{1,\min}$ 'inden daha küçük olur. Sonuç olarak OVY tabanlı GMM yönteminde daha büyük logaritmik-benzerliğe sahip olunacağı söylenebilir.

Şekil 3.3 OVY tabanlı GMM yöntemiyle bulunan Gauss dağılımlarının GMM'dekine göre nasıl daha düzgün hale geldiğini göstermektedir. Şekil 3-3(a)'da öznitelik vektörleri fonetik bilgileri de içerdiğinden spektrumları Gauss dağılımına fazla benzememektedir. Bu nedenle ancak çok sayıda katışım ile modellenenirler. Şekil 3-3(b)'de ise öznitelik vektörleri sadece konuşmacı bilgisini içerdiğinden spektrumları çok daha düzgündür ve tek bir katışım ile bile modellenenirler. Ayrıca EM algoritması asıl öznitelik vektörleriyle karşılaştırıldığında bu girişler için daha az adımda yerel en büyüğe yakınsar. Sonuç olarak OVY tabanlı GMM yöntemiyle geleneksel GMM'den daha iyi başarımlar elde edilecektir.



Şekil 3.3 GMM ve OVY tabanlı GMM yöntemlerinde (a) GMM yönteminde elde edilen gauss dağılımları (b) OVY tabanlı GMM yönteminde elde edilen gauss dağılımlarının karşılaştırılması

4. BÖLÜM

DENEYSEL ÇALIŞMA

4.1 Giriş

Bu bölümde önce, TIMIT veri tabanı ve deneysel çalışma veri tabanı konusunda daha detaylı bilgi verilmiş, sonra öznitelik vektörlerinin elde edilme yöntemi açıklanmıştır. Son olarak yukarıda bahsedilen yöntemlere ait tanıma performansı ve hesaplama maliyetlerinin değerlendirilmesi yapılmıştır.

4.2 TIMIT Veri Tabanı

Otomatik konuşma tanıma sistemlerinin geliştirilmesi ve değerlendirilmesi, aynı zamanda akustik fonetik bilgi edinilmesi için konuşma verisi sağlamak üzere oluşturulmuş bir veri tabanıdır (Garofolo et al., 1993). Öncelikli olarak konuşma tanıma için tasarlanmış olmasına rağmen oldukça fazla sayıda konuşmacıya sahip birkaç veri tabanından biri olduğu için konuşmacı tanıma çalışmalarında da yaygın olarak kullanılmaktadır. TIMIT (NIST, 1990), 630 konuşmacının (438 erkek/ 192 bayan), söylediği 10 cümlenin toplamını oluşturan 6300 cümleyi kapsar. Her cümle yaklaşık 3 sn'dir. Cümleler zengin fonetik değişkenliğe sahip olacak şekilde tasarlanmıştır (Reynolds, 1995). Konuşmacılar Amerika Birleşik Devletleri'ndeki farklı lehçelerin kullanıldığı 8 ana bölgeden seçilmiştir. Kayıtlar mikروفon ile tek kanallı ve 16 bit çözünürlükte yapılmıştır. Örnekleme frekansı 16 Khz'dir. TIMIT veri tabanını oluşturan 3 cümle tipiyle ilgili bilgiler Çizelge 4.1'de verilmiştir. Çizelge 4.2 ise TIMIT veri tabanında sağlanan dosya tiplerini açıklamaktadır.

Çizelge 4.1 TIMIT konuşma materyali

Cümle Tipi	Cümle sayısı	Konuşmacı sayısı	Toplam	Cümle/Konuşmacı
Lehçeli (SA)	2	630	1260	2
Kısa (SX)	450	7	3150	5
Çeşitli (SI)	1890	1	1890	3
Toplam	2342		6300	10

Çizelge 4.2 Veri tabanındaki okunuşlar ile ilgili dosya tipleri

Dosya Tipi	Tanımı
.wav	SPHERE-başlıklı konuşma dosyası
.txt	Kişilerin söyledikleri ifadelerin metin olarak gösterilmesi
.wrđ	Kelime geçişleri / cümle içindeki yerleri
.phn	Fonetik geçişler

4.3 Deneysel Çalışmada Kullanılacak Şekilde Veri Tabanının Düzenlenmesi

Yapılan deneysel çalışmalarda kullanılmak üzere TIMIT veri tabanından yeni bir veri tabanı oluşturulmuştur. Bu veri tabanı DR1 (New England) bölgesinden 12 erkek 8 bayan olmak üzere 20 konuşmacıdan oluşur. Seçilen 20 konuşmacının tanıtım kodları ve söyledikleri cümlelerin dosya indeksleri Çizelge 4.3’de verilmiştir.

Başlangıçta 10 cümle için iki farklı birleşimi denenmiştir. İlk durumda her konuşmacının 2 SA ve 3 SI cümlesi eğitim seti, 5 SX cümlesinin her biri test seti olarak kullanılmıştır. Bu durum için eğitim ve test devam süreleri sırasıyla yaklaşık 14 sn ve 3 sn’dir. İkinci durumda ise 2 SA, 3 SI ve ilk 3 SX cümlesi eğitim setinde, geriye kalan 2 SX cümlesinin her biri test setinde kullanılmıştır. Eğitim ve test setinin devam süreleri ikinci durum için yaklaşık olarak 24 sn ve 3 sn’dir.

OVY ve GMM'in bu iki durum için başarıları çok yüksek olduğundan dolayı, GMM ile Ovy tabanlı GMM'in tanıma başarısını daha iyi karşılaştırmak için veri tabanında yeni bir düzenleme yapılmıştır. Bu üçüncü durumda eğitim ve test süreleri kısaltılmıştır. Eğitim seti 2 SA ve 1 SI cümlesinden oluşur ve yaklaşık 9 sn'dir. Her SX cümlesinin ilk 1 sn'si test girdisi olarak kullanılmıştır.

Çizelge 4.3 Düzenlenmiş TIMIT veri tabanındaki konuşmacı bilgileri

Tanıtm kodu	SA indeksi	SX indeksi	SI indeksi
FCJF0	1 2	127 307 37 217 397	1027 1657 648
MCPM0	1 2	114 294 24 204 384	564 1194 1824
MDAC0	1 2	181 361 91 271 451	631 1261 1837
FDAW0	1 2	326 56 236 416 146	1271 1406 2036
FDML0	1 2	69 249 429 159 339	2075 1149 1779
MDPK0	1 2	153 333 63 243 423	1053 1683 552
FECD0	1 2	338 68 248 428 158	788 1418 2048
MEDR0	1 2	294 24 204 384 114	744 1374 2004
FETB0	1 2	68 248 428 158 338	518 1148 1778
MGRLO	1 2	417 147 327 57 237	867 1497 2127
MJEB1	1 2	387 117 297 27 207	837 1467 2097
FJSP0	1 2	354 84 264 444 174	804 1434 1763
MJWT0	1 2	301 31 211 391 121	751 1381 1291
FKFB0	1 2	78 258 438 168 348	978 1608 2238
MKLS0	1 2	357 87 267 447 177	1533 1437 2067
MKLW0	1 2	41 221 401 131 311	1844 1571 2201
FMEM0	1 2	297 333 207 387 117	747 1377 2007
MMGG0	1 2	179 359 89 269 449	1079 1709 2339
MMRP0	1 2	324 54 234 414 144	774 717 2034
MPGH0	1 2	24 204 384 114 294	924 1554 675

Yetersiz veri durumunda OVY yönteminin uygulanabilmesi için eğitim ve test süreleri her konuşmacı için aynı uzunlukta olması gerekmektedir. Dördüncü durumda oluşturulan veri tabanında, bu nedenle, ilk durumdaki eğitim süreleri bütün konuşmacılar için 13 sn ile ve test süreleri ise bütün SX cümleleri için 1 sn ile eşitlenmiştir.

Çizelge 4.4 oluşturulan veri tabanındaki konuşmacıların SA, SI ve SX söyleyişlerinin ortalama örnek sayısını ve süresini göstermektedir.

Çizelge 4.4 TIMIT SA, SI, SX cümlelerinin ortalama örnek sayıları ve uzunlukları

Kayıt Adı	Örnek Sayısı	Süre	Çerçeve
SA1	53192	3.3	414
SA2	44943	2.8	349
SI-1	55659	3.5	433
SI-2	56724	3.5	441
SI-3	49137	3.1	382
SX-1	47488	3.0	369
SX-2	45675	2.9	355
SX-3	48159	3.0	374
SX-4	41011	2.6	319
SX-5	46781	2.9	364

4.4 Öznitelik Vektörlerinin Elde Edilmesi

Yapılan çalışmalarda MFCC parametreleri kullanılmıştır. Şekil-4.1, MFCC parametrelerinin elde edilmesinde kullanılan algoritmayı blok şema olarak göstermektedir.

Çerçeveleme ve pencereleme: TIMIT veri tabanından alınan konuşma verisi önce 256 örneklilik çerçevelere ayrılır ve %50 örtüşme ile Hamming penceresi uygulanır. Böylece her bir çerçeve periyodu 8 ms'ye karşılık gelir. Pencereleme, çerçevenin başı

ve sonundaki süreksizliği azaltarak spektral bozulmayı engellemek amacıyla yapılır. Eğer pencereyi $w(n)$ ile gösterirsek pencerelenmiş sinyal

$$y_1(n) = x_1(n)w(n), \quad 0 \leq n \leq K-1 \quad (4.1)$$

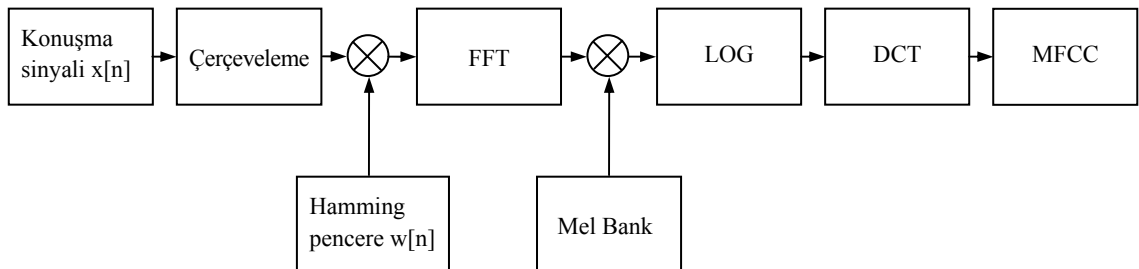
olur. Burada kullanılan Hamming penceresi

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{K-1}\right), \quad 0 \leq n \leq K-1 \quad (4.2)$$

ile ifade edilebilir.

Mel Filtreleme : FFT alınarak bulunan her çerçevenin spektrumu Mel-ölçekli filtre bankasından geçirilir. Bunun amacı insan kulağının doğrusal olmayan işitme spektrumunun benzetimini yapmaktır. Mel-ölçekli filtre bankası 39 üçgen şekilli filtreden oluşur.

Kepstrum : Spektrumun log-alanına dönüşümünden sonra kesikli kosinüs dönüşümü (DCT) ile mel-frekanslı kepsstral katsayıları (MFCC) bulunur. Her çerçeveden elde edilen 20 MFCC katsayısı, o çerçeveye karşılık gelen bir öznitelik vektörü gösterir.



Şekil 4.1 MFCC algoritması akış diyagramı

Sonuç olarak yöntemlerin performans değerlendirmesinde kullanılmasına karar verilen parametreler ile ilgili bilgiler topluca aşağıda verilmiştir:

Veri tabanı	:	TIMIT
Bölge	:	DR1
Örnekleme frekansı	:	16 Khz
Konuşmacı sayısı	:	20 (12 erkek + 8 bayan)
Parametre tipi	:	MFCC
Parametre sayısı	:	20
Çerçeve uzunluğu	:	256 örnek
Örtüşme	:	128 örnek
Pencere tipi	:	Hamming
Filtre sayısı	:	39 (üçgen)

MFCC katsayılarının oluşturulmasında “VOICEBOX : Speech Processing Toolbox for MATLAB” paketinden faydalanılmıştır.

4.5 Karar Kriterleri ve Tanıma Oranları

Metinden bağımsız konuşmacı tanımada FLDA, GMM, OVY, DCV ve OVY tabanlı GMM yöntemleri kullanılmış ve sonuçları karşılaştırılmıştır.

4.5.1 Yeterli veri durumu ($N > d$) için yapılan çalışmalar

FLDA yönteminde bilinmeyen vektörün genelleştirilmiş özvektörler üzerine izdüşümüyle her sınıfın ortalama vektörü arasındaki uzaklık karar kriteri olarak kullanılmıştır. 1. durumda en büyük 7 özdeğere karşılık gelen özvektörler ($L=16$) alındığında eğitim kümesi için %98, test kümesi için %87 tanıma oranı elde edilmiştir. 2. durumda en büyük 7 özdeğere karşılık gelen özvektörler alındığında eğitim ve test kümeleri için tanıma oranları %97.5 ve %95 olarak bulunmuştur. Her iki durumda

bulunan sonuçlar eğitim seti için Çizelge 4.5’de ve test seti için Çizelge 4.6’da yüzde olarak verilmiştir.

Çizelge 4.5 Eğitim seti tanıma oranları

Sıra No	Konuşmacı kodu	OVY		FLDA		GMM16		GMM32	
		Durum I	Durum II	Durum I	Durum II	Durum I	Durum II	Durum I	Durum II
1	FCJF0	100	100	100	100	100	100	100	100
2	MCPM0	100	100	100	100	100	100	100	100
3	MDAC0	100	100	100	100	100	100	100	100
4	FDAW0	100	100	100	100	100	100	100	100
5	FDML0	100	100	100	100	100	100	100	100
6	MDPK0	80	87.5	80	87.5	100	100	100	100
7	FECD0	100	100	80	100	100	100	100	100
8	MEDR0	100	100	100	100	100	100	100	100
9	FETB0	100	100	100	100	100	100	100	100
10	MGRL0	100	100	100	100	100	100	100	100
11	MJEB1	100	100	100	100	100	100	100	100
12	FJSP0	100	100	100	100	100	100	100	100
13	MJWT0	100	100	100	62.5	100	100	100	100
14	FKFB0	100	100	100	100	100	100	100	100
15	MKLS0	100	100	100	100	100	100	100	100
16	MKLW0	100	100	100	100	100	100	100	100
17	FMEM0	100	100	100	100	100	100	100	100
18	MMGG0	100	100	100	100	100	100	100	100
19	MMRP0	100	100	100	100	100	100	100	100
20	MPGH0	100	100	100	100	100	100	100	100
Ortalama tanıma oranı		99	99.3	98	97.5	100	100	100	100

GMM yönteminde tanıma oranlarının ve hesaplama maliyetlerinin katışım sayısı ile nasıl değiştiğini görmek için iki farklı katışım sayısı uygulanmıştır. Katışım sayısının $M=16$ olduğu durum GMM16 ile katışım sayısının $M=32$ olduğu durum da GMM32 ile ifade edilmiştir. Herhangi bir konuşmacıya ait öznitelik vektörleri dizisi verildiğinde eğitim aşamasında döngüsel EM algoritması kullanılarak en büyük olabilirlik için model parametreleri kestirilir (Dempster, 1977). EM algoritmasının yakınsaması için uygulanacak döngü sayısı en fazla 10 olarak belirlenmiştir. Ağırlık katsayılarının başlangıç değerleri $1/M$ olarak eşit alınmıştır. Bir konuşmacı verisinden (N_j/M) aralıklarla alınan öznitelik vektörleri ise katışım ortalamalarının başlangıç değerleri olarak kullanılmıştır. En iyileştirilecek parametre sayısını azaltmak amacı ile

köşegen saçılım matrisi tercih edilmiştir. Karar için olabilirlik testi kullanılır (Eş.3.8). Bilinmeyen bir konuşmacının bir test vektörü için her bir konuşmacı modeline karşılık gelen çoklu gauss olabilirliği hesaplanır. En büyük olabilirlik, bilinmeyen konuşmacının kimliğini belirler. İstatistiksel modelleme için Ek.1’de verilen “gmm_evaluate” MATLAB işlevi kullanılmıştır. GMM algoritmasının çıktısı; modelin ortalaması, sapması ve ağırlığıdır. Test aşamasında “Imultigauss” işlevi kullanılmıştır. Bu işlev test verisini ve model parametrelerini kullanarak çoklu Gauss logaritmik olabilirliğini hesaplar.

GMM16 ve GMM32 için birinci durumda elde edilen test seti tanıma oranları %99 ve %100’dür. İkinci durumdaki test seti tanıma oranları ise her iki katışım sayısı için %100’dür (Çizelge 4.6).

Çizelge 4.6 Test seti tanıma oranları

Sıra No	Konuşmacı kodu	OVY		FLDA		GMM16		GMM32	
		Durum I	Durum II	Durum I	Durum II	Durum I	Durum II	Durum I	Durum II
1	FCJF0	100	100	60	50	100	100	100	100
2	MCPM0	100	100	80	50	100	100	100	100
3	MDAC0	100	100	100	100	100	100	100	100
4	FDAW0	100	100	100	100	100	100	100	100
5	FDML0	100	100	100	100	100	100	100	100
6	MDPK0	100	100	60	100	100	100	100	100
7	FECD0	100	100	100	100	100	100	100	100
8	MEDR0	100	100	100	100	100	100	100	100
9	FETB0	100	100	100	100	100	100	100	100
10	MGRL0	100	100	100	100	100	100	100	100
11	MJEB1	100	100	100	100	80	100	100	100
12	FJSP0	100	100	100	100	100	100	100	100
13	MJWT0	100	100	60	100	100	100	100	100
14	FKFB0	100	100	100	100	100	100	100	100
15	MKLS0	100	100	100	100	100	100	100	100
16	MKLW0	100	100	80	100	100	100	100	100
17	FMEM0	100	100	60	100	100	100	100	100
18	MMGG0	100	100	100	100	100	100	100	100
19	MMRP0	100	100	60	100	100	100	100	100
20	MPGH0	100	100	80	100	100	100	100	100
Ortalama tanıma oranı		100	100	87	95	99	100	100	100

OVY yönteminin eğitim aşamasında, her konuşmacı için farksızlık altuzayı ve ortak vektör Bölüm 3.4.2’de anlatılan yöntem izlenerek bulunur. Farksızlık altuzayı belirlemede $L=16$ yüzdesi her sınıf için farklı sayıda özdeğere karşılık gelebilir. Deneysel çalışmada bütün sınıflar için hesaplanan özdeğer sayılarının ortalaması alınarak m değeri belirlenmiş ($m=4$) ve böylece en küçük 16 özdeğere karşılık gelen özvektörlerden farksızlık altuzayı oluşturulmuştur.

Tanıma aşamasında bilinmeyen bir \mathbf{x}_t öznitelik vektörü için her bir sınıfa ait *kalan vektör* olarak adlandırılan $\mathbf{x}_{t,j}^{\text{rem}}$ vektörü, \mathbf{x}_t ’nin bu sınıfa ait farksızlık altuzayına izdüşümü alınarak hesaplanır. Yeterli veri durumunda \mathbf{x}_t yerine test öznitelik vektörlerinin ortalaması $\boldsymbol{\mu}_t$ kullanılır.

$$\mathbf{x}_{t,j}^{\text{rem}} = (\mathbf{x}_t^T \mathbf{u}_j^{m+1}) \mathbf{u}_j^{m+1} + \dots + (\mathbf{x}_t^T \mathbf{u}_j^d) \mathbf{u}_j^d \quad (4.3)$$

Her sınıfın ortak ve kalan vektörleri arasındaki öklid uzaklığı OVY’de karar kriteri olarak kullanılır. Eğer $\mathbf{x}_{t,j}^{\text{rem}}$ ve $\mathbf{x}_j^{\text{com}}$ arasındaki uzaklık en az ise \mathbf{x}_t öznitelik vektörü j ’nci konuşmacıya aittir. Böylece karar kriteri, yetersiz veri durumunda

$$c^* = \arg \min_j \left\| \mathbf{x}_{t,j}^{\text{rem}} - \mathbf{x}_j^{\text{com}} \right\| = \arg \min_j \left\| \bar{\mathbf{P}}_j (\mathbf{x}_t - \mathbf{x}_{k,j}) \right\| \quad (4.4)$$

ve yeterli veri durumunda

$$c^* = \arg \min_j \left\| \boldsymbol{\mu}_{t,j}^{\text{rem}} - \mathbf{x}_j^{\text{com}} \right\| = \arg \min_j \left\| \bar{\mathbf{P}}_j (\boldsymbol{\mu}_t - \boldsymbol{\mu}_j) \right\| \quad (4.5)$$

eşitliği ile ifade edilebilir.

En küçük 16 özdeğere karşılık gelen özvektörler kullanıldığında ($L=16$), OVY’nin eğitim seti tanıma oranları Çizelge 4.5’de görüldüğü gibi birinci durum için %99, ikinci durum için %99.3 bulunmuştur. Test seti için aynı sayıda özvektör kullanıldığında her iki durumda da %100 tanıma oranı elde edilmektedir (Bkz. Çizelge

4.6). Ancak burada en küçük 14 özdeğere karşılık gelen özvektör kullanıldığında eğitim seti tanıma oranlarının %100 olduğunu belirtmekte yarar vardır.

GMM yönteminin başarısı 1. ve 2. durum için çok yüksek olduğu için OVY tabanlı GMM'in etkisi bu iki durum için test edilemez. Bu nedenle 3. durum adını verdiğimiz yeni bir veri tabanı oluşturduk. Bu yeni veri tabanında eğitim süresi 9 sn'ye ve test süreleri de 1 sn'ye indirilmiştir. GMM ve OVY tabanlı GMM yöntemleri için bulunan tanıma oranları eğitim seti için Çizelge 4.7'de ve test seti için Çizelge 4.8'de verilmiştir. Önerilen metodun test seti tanıma oranlarını GMM16'da %90'dan %93'a ve GMM32'de %86'dan %94'e çıkardığı görülmüştür.

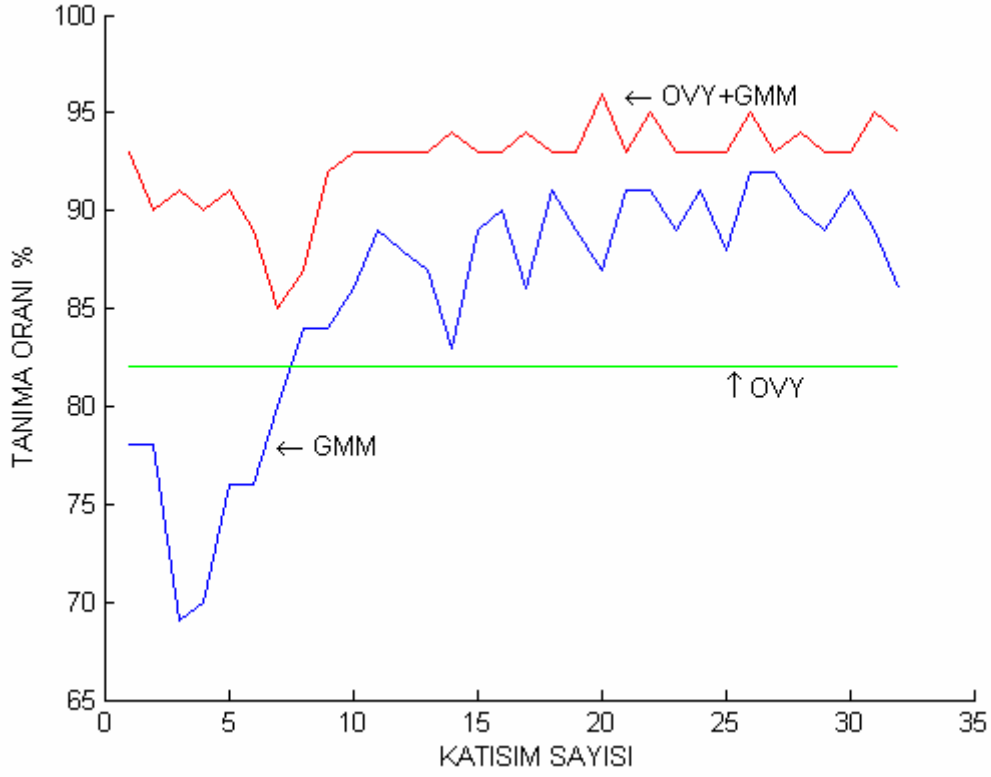
Çizelge 4.7 GMM ve OVY tabanlı GMM yöntemlerinin eğitim seti tanıma oranları

Konuşmacı Sıra No	Konuşmacı kodu	GMM16	GMM32	OVY+ GMM16	OVY+ GMM32
1	FCJF0	100	100	100	100
2	MCPM0	100	100	100	100
3	MDAC0	100	100	100	100
4	FDAW0	100	100	100	100
5	FDML0	100	100	100	100
6	MDPK0	66.6	66.6	66.6	66.6
7	FECD0	100	100	100	100
8	MEDR0	100	100	100	100
9	FETB0	100	100	100	100
10	MGRL0	100	100	100	100
11	MJEB1	100	100	100	100
12	FJSP0	100	100	100	100
13	MJWT0	100	100	100	100
14	FKFB0	100	100	100	100
15	MKLS0	100	100	100	100
16	MKLW0	100	100	100	100
17	FMEM0	100	100	100	100
18	MMGG0	100	100	100	100
19	MMRP0	66.6	66.6	66.6	66.6
20	MPGH0	100	100	100	100
Ortalama tanıma oranı		96.6	96.6	96.6	96.6

Çizelge 4.8 GMM ve OVY tabanlı GMM yöntemlerinin test seti tanıma oranları

Konuşmacı Sıra No	Konuşmacı kodu	GMM16	GMM32	OVY + GMM16	OVY + GMM32
1	FCJF0	80	80	80	80
2	MCPM0	100	100	100	100
3	MDAC0	100	80	100	100
4	FDAW0	100	60	100	100
5	FDML0	60	80	100	100
6	MDPK0	100	100	100	100
7	FECD0	100	100	100	100
8	MEDR0	100	100	100	100
9	FETB0	100	100	100	100
10	MGRL0	60	80	60	60
11	MJEB1	60	60	60	100
12	FJSP0	100	100	100	100
13	MJWT0	100	100	100	80
14	FKFB0	100	60	80	80
15	MKLS0	80	80	100	100
16	MKLW0	100	100	100	100
17	FMEM0	100	80	100	100
18	MMGG0	80	80	100	100
19	MMRP0	80	100	80	80
20	MPGH0	100	80	100	100
Ortalama tanıma oranı		90	86	93	94

OVY tabanlı GMM'in geleneksel GMM sonuçlarını iyileştirmedeki başarısı 1 ile 32 arasındaki bütün katışım sayısındaki sonuçlar dikkate alınarak incelendiğinde daha iyi anlaşılmaktadır. Şekil 4.2 OVY, GMM ve OVY tabanlı GMM'in test seti için bütün katışım sayılarındaki tanıma oranını göstermektedir. GMM yönteminde en iyi tanıma oranı 26 katışım sayısındaki %92'dir. OVY tabanlı GMM yöntemi ise tek bir katışım kullanılarak bile %93 ile bu oranı geçmektedir. OVY tabanlı GMM'in en iyi tanıma oranı ise 20 katışım sayısındaki %96'dır. Bu sonuçlar OVY tabanlı GMM'in tercih edilmesi için çok büyük iki avantaj sağlar. İlki düşük katışım sayılarında bile verdiği tanıma oranı GMM'in en iyi tanıma oranından daha yüksektir. İkincisi 26 katışım yerine tek bir katışım kullanılması hesaplama maliyetlerini büyük ölçüde düşürmektedir.



Şekil 4.2 OVY, GMM ve OVY tabanlı GMM yöntemlerinin test seti için tanıma oranları

4.5.2 Yetersiz veri durumu ($N < d$) için yapılan çalışmalar

OVY yönteminin ve diğer yöntemlerin metinden bağımsız konuşmacı tanımadaki başarımlarının yeterli veri durumunda olduğu kadar yetersiz veri durumunda da değerlendirilmesine ihtiyaç vardır. Ancak her çerçeveden elde edilen 20 MFCC katsayısı bir öznitelik vektörü olarak alındığında öznitelik vektörlerinin sayısı öznitelik uzayının boyutundan çok büyük olacağı için doğal olarak yetersiz veri durumu elde edilemeyecektir. Dolayısı ile her 1 sn yani 16000 örneklilik konuşma parçası için elde edilen MFCC katsayıları birbiri ardına sıralanarak 2480 boyutlu bir öznitelik vektörü haline getirilmiştir. Böylece her bir konuşmacı için eğitim seti 13 öznitelik vektöründen oluşur. Test vektörleri ise konuşmacılara ait SX cümlelerinin ilk 1 sn'lik bölümünden elde edilen öznitelik vektöründen oluşmaktadır.

OVY yönteminin yetersiz veri durumunda eğitim ve test seti için verdiği tanıma oranları Çizelge 4.9’da gösterilmektedir. Bu veri tabanı için Ayırteci Vektör Yaklaşımının konuşmacı tanımadaki başarımı da değerlendirilmiş ve aynı tabloda gösterilmiştir. Ovy yönteminin yetersiz veri durumundaki hesaplama süresi, öznelik vektörlerinin boyutundaki büyüme ile orantılı olarak çok artmaktadır. Bu dezavantajı ortadan kaldırmak amacıyla Turhal et al., (2005) tarafından ortaya atılan ve daha önce görüntü tanımda başarıyla uygulanan Ortak Matris Yaklaşımı (OMY) metinden bağımsız konuşmacı tanımda kullanılmıştır. OMY ile elde edilen tanıma oranları Ovy ile elde edilenlerle aynı olmakla birlikte hesaplama süresi yaklaşık 50 kat azalmıştır.

Çizelge 4.9’a ayrıca ($N > d$) durumu için uygulanan yöntemlerden elde edilen sonuçlar eklenmiştir. Bu sonuçlar Ovy (yeterli veri durumu), FLDA, GMM ve Ovy tabanlı GMM yöntemlerinin 4. durumda uygulanan eğitim ve test süreleri (13 sn eğitim, 1 sn test) için verdiği tanıma oranlarıdır. Eşit uzunlukta eğitim ve test süreleri için, deneysel çalışmada uygulanan bütün yöntemlerin tanıma oranlarının daha iyi karşılaştırılabilmesi amacı ile sonuçlar aynı çizelgede gösterilmiştir.

Çizelge 4.9 ($N < d$) ve ($N > d$) durumunda uygulanan yöntemlerin tanıma oranlarının karşılaştırılması

	$(N < d)$ durumu			$(N > d)$ durumu			
Yöntem	OVY	OMY	DCV	OVY	FLDA	GMM16	OVY+ GMM16
Eğitim	100	100	100	94	85	98	96
Test	26	26	44	85	83	91	94

4.6 Uygulanan yöntemlerin hesaplama maliyetleri

Deneysel çalışmalarda Pentium-4 tabanlı 3.4 GHz işlemcili ve 512 MB RAM'a sahip bir bilgisayar kullanılmıştır. İncelenen yöntemlerden her birisinin hesaplama süreleri ölçülmüş ve karşılaştırılmak amacıyla Çizelge 4.10'da verilmiştir. Çizelge 4.11'de ise yöntemler, bellek ihtiyacının miktarına göre küçükten büyüğe doğru sıralanmıştır. OVY yöntemi gerek eğitim ve test sürelerinin en kısa oluşu gerekse en az bellek ihtiyacı ile uygulanan diğer yöntemlerden daha üstündür.

Çizelge 4.10 Yöntemlerin eğitim ve test süreleri

YÖNTEM	DURUM I		DURUM II		DURUM III	
	Eğitim süresi (sn)	Test süresi (sn)	Eğitim süresi (sn)	Test süresi (sn)	Eğitim süresi (sn)	Test süresi (sn)
OVY	13.6250	0.0717	21.9380	0.0672	-	-
FLDA	14.3440	0.0800	22.9390	0.0703	-	-
GMM16	113.4840	0.9319	176.5470	0.8934	55.9530	0.2289
GMM32	219.6870	1.8720	341.0930	1.7950	118.5160	0.5351
OVY+GMM16	-	-	-	-	57.3750	0.2010
OVY+GMM32	-	-	-	-	119.4070	0.4120

Çizelge 4.11 Yöntemlerin bellek ihtiyaçlarının karşılaştırılması

Sıra	Bellek ihtiyacı
1	OVY
2	FLDA
3	OVY+GMM16
4	GMM16
5	OVY+GMM32
6	GMM32

5. BÖLÜM

SONUÇLAR

Konuşmacı tanıma, özellikle güvenlik uygulamaları başta olmak üzere birçok alanda büyük öneme sahiptir. Konuşmacı tanımaya yönelik birçok algoritma geliştirilmesine rağmen değişik koşullar altında konuşmacıların halen verimli bir şekilde tanınmaması yeni konuşmacı tanıma yöntemlerine ihtiyaç duyulmasına neden olmuştur.

Bu tez çalışmasında sınıflama amacıyla öncelikle iyi bilinen yöntemler olan FLDA ve GMM kullanılmıştır. Deneysel çalışmalar tanıma oranları açısından bakıldığında GMM'in FLDA'den daha üstün olduğunu ortaya koymuştur. Hesaplama süresi ve bellek ihtiyacı açısından bakıldığında FLDA GMM'den daha avantajlıdır. Diğer yandan GMM16 ve GMM32'nin tanıma oranları karşılaştırıldığında GMM32 daha iyi sonuçlar vermesine rağmen hesaplama maliyeti GMM16'dan çok daha büyüktür.

Bu tez çalışmasında ilk defa metinden bağımsız konuşmacı tanıma için OVY yöntemi kullanılmıştır. OVY konuşma ve görüntü tanıma başarıları sağlamış altuzay tabanlı yeni bir yöntemdir. Ortak vektör, bir sınıfın saçılım matrisinin en küçük özdeğerlerine karşılık gelen özvektörleri yönündeki vektördür ve sınıfının bütün ortak değişmez özelliklerini taşır. Ortak vektör her sınıf için tek olduğundan ve o sınıfın ortak özelliklerini taşıdığından sınıflama problemleri için uygun çözümdür. OVY'nin test seti için 1. durumdaki tanıma oranı FLDA'dan ve GMM16'dan daha iyi ancak GMM32 ile aynıdır. Bununla beraber OVY, hesaplama süresi ve bellek ihtiyacı açısından FLDA ve GMM yöntemlerinden daha avantajlıdır.

Ayırtedici ortak vektör yöntemi, yetersiz veri durumunda uygulanan OVY'den daha iyi sonuç vermektedir. Ancak diğer yöntemlere göre konuşmacı tanımadaki başarısının düşük olduğu görülmüştür. Bunun sebebi mevcut veri tabanında eğitim setindeki vektör sayısının, vektör boyutundan çok küçük oluşudur.

Bu tez çalışmasında GMM'in başarısını iyileştirmek amacıyla OVY tabanlı GMM yöntemi önerilmiştir. OVY tabanlı GMM yönteminin etkisini değerlendirmek üzere 3. durum diye tanımladığımız eğitim ve test sürelerinin kısaltılması ile oluşturulan farklı yapıda bir veri tabanı kullanılarak konuşmacı tanıma deneyi gerçekleştirilmiştir. Bu veri tabanı 1. ve 2. durumdaki veri tabanlarına göre daha düşük tanıma oranları vermektedir. Önerilen yöntem bu veri tabanında kullanıldığında, değişik sayıdaki katışımlar için 22 puana varan bir iyileşme sağlamaktadır. Aynı katışım sayısında OVY tabanlı GMM'in eğitim süresi GMM'dekine çok yakındır. Test süresi klasik GMM'den daha düşüktür. Bellek ihtiyacı açısından bakıldığında ise OVY tabanlı GMM yöntemi GMM'den daha az belleğe ihtiyaç duyar.

Düşük katışım sayılarında bile OVY tabanlı GMM yönteminin GMM'de elde edilen en iyi sonuçtan daha yüksek başarı oranı göstermesi GMM'in yüksek hesaplama yükünün düşürülmesinde büyük avantaj sağlar. Önerilen yöntem kullanılarak eğitim ve test süresinin yetersiz olduğu durumlarda GMM'in bir dezavantajı olarak ortaya çıkan tanıma oranı düşmesi iyileştirilmekte, aynı zamanda GMM'in hesaplama süresi ve bellek ihtiyacında da önemli bir azalma elde edilmektedir. Sonuç olarak tanıma oranları, bellek ihtiyacı, eğitim ve test süreleri birlikte değerlendirildiğinde önerilen OVY tabanlı GMM yönteminin GMM'e göre tercih edilen yöntem olduğu görülür.

Metinden bağımsız konuşmacı tanımada, yöntemlerin başarısı eğitim ve test sürelerinin uzunluğuna bağlıdır. Test girdilerinin 1 sn'nin altında olması durumunda tanıma yüzdesi önemli ölçüde düşmektedir. Yetersiz veri durumunda OVY'nin başarılı olabilmesi için en az 3 dk'lık bir eğitim süresi gerekmektedir. TIMIT buna uygun bir veri tabanı değildir. Daha uzun eğitim kayıtları sağlayabilecek bir veri tabanı üzerinde yöntemin tanıma oranları daha yüksek elde edilebilir.

Elde edilen sonuçlar TIMIT veri tabanında DR1 bölgesinden seçilen 20 kişilik konuşmacı kayıtları kullanılarak bulunmuştur. Uygulanan yöntemlerin lehçelerden nasıl etkilendiğini görmek için TIMIT'deki diğer bölgelere ait konuşmacı kayıtları kullanılabilir. Yöntemlerin değişik sınıf sayılarındaki tanıma oranlarının hesaplanması,

sınıf sayısı ile tanıma oranları arasındaki deęişimin ortaya koyulmasına yardımcı olacaktır. Özellikle telefon hattı üzerinden konuşmacı tanıma gibi gürültülü ortamda gerçekleştirilecek uygulamalarda, önerilen OVY tabanlı GMM yönteminin geleneksel GMM'in performansını çok daha yüksek oranda iyileştireceęi düşünülmektedir. Bu amaçla benzer bir çalışma NTIMIT veri tabanı kullanılarak tekrar edilebilir.

6. KAYNAKLAR DİZİNİ

- Alonso-Martinez C., Faundez-Zanuy M., 2000, Speaker identification in mismatch training and testing conditions. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 2, 1181-1184.
- Ariki Y., Tagashira S., Nishijima M., 1996, Speaker recognition and speaker normalization by projection to speaker subspace. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 1, 319-322.
- Belhumeur P.N., Hespanha J.P., Kriegman D.J., 1997, Eigensfaces vs. Fisherspaces: Recognition Using Class Specific Linear Projection. IEEE Trans. Pattern Analysis and Machine Intelligence 19.
- Besacier L., Bonastre J.F., 1998, Frame pruning for speaker recognition. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP), pp. 765-768.
- Borah D.K., DeLeon P., 2004, Speaker identification in the presence of packet losses. IEEE Digital Signal Processing Workshop, pp. 302-306.
- Campbell J.P., 1997, Speaker recognition: A tutorial. Proceedings of IEEE 85, 1437-1462.
- Campbell J.P., Reynolds D. A., 1999, Corpora for the evaluation of speaker recognition systems. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 2, 829-832.
- Campbell W.M., Assaleh K.T., Broun C.C., 2002, Speaker recognition with polynomial classifiers. IEEE Trans. On Speech and Audio Processing 10, 205-212.
- Chaudhari U.V., Navratil J., Maes S.H., 2003, Multigrained modeling with pattern specific maximum likelihood transformations for text-independent speaker recognition. IEEE Trans. On Speech and Audio Processing 11, 61-69.
- Çevikalp H., Neamtu M., Wilkes M., 2006, Discriminative common vector method with kernels. IEEE Trans. on Neural Networks 17, 1550-1565.

KAYNAKLAR DİZİNİ (devam ediyor)

- Çevikalp H., Neamtu M., Wilkes M., Barkana A., 2005, Discriminative common vectors for face recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 27, 1-10.
- Dempster A., Laird N. and Rubin D., 1977, Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc.*, 39, 1-38.
- Feng L., Hansen L.K., 2005. A new database for speaker recognition, Technical Report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU.
- Garofolo J.S., Lamel L.F., Fisher W.M., Fiscus J.G., Pallett D.S., Dahlgren N.L., 1993, DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM, NIST.
- Griffin C., Matsui T., Furui S., 1994, Distance measures for text-independent speaker recognition based on MAR model. In: *Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP)* 1, 309-312.
- Gülmezoğlu M.B., Dzhafarov V., Keskin M., Barkana A., 1999, A novel approach to isolated word recognition. *IEEE Trans. on Speech and Audio Processing* 7, 620-628.
- Gülmezoğlu M.B., Dzhafarov V., Barkana A., 2001, The common vector approach and its relation to the principal component analysis. *IEEE Trans. on Speech and Audio Processing* 9, 655-662.
- Gülmezoğlu M.B., Dzhafarov V., Edizkan R., Barkana A., 2007, The common vector approach and its comparison with other subspace methods in case of sufficient data. *Computer Speech and Language* 21, 266-281.
- Hayakawa S., Itakura F., 1994, Text-dependent speaker recognition using the information in the higher frequency band. In: *Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP)* 1, 137-140.

KAYNAKLAR DİZİNİ (devam ediyor)

- Lamel, L., Gauvain, J.L., 1997, Speaker recognition with the switchboard corpus. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 2, 1067-1070.
- Landgrebe D.A., 2002, Hyperspectral image data analysis. IEEE Signal Processing Magazine 19, 17-28.
- Liu L., He J., Palm G., 1996, Signal modeling for speaker identification. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 2, 665-668.
- Lyu S., 2005, Kernels for Unordered Sets: the Gaussian Mixture Approach. In: European Conference on Machine Learning (ECML). Porto, Portugal.
- NIST, 1990, Getting started with darpa TIMIT CD-ROM: an acoustic phonetic continuous speech database, National Institute of Standards and Technology (NIST), Gaithersburg, MD.
- Oja E., 1983, Subspace methods of pattern recognition. John Wiley and Sons Inc., New York.
- Quatieri T.F., Reynolds D.A., O'leary G.C., 2000, Estimation of handset nonlinearity with application to speaker recognition. IEEE Trans. on Speech and Audio Processing 8, 567-584.
- Quatieri T.F., Dunn R.B., Reynolds D.A., Campbell J.P., Singer E., 2000, Speaker recognition using G.729 speech codec parameters. In: Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 2, 1089-1092.
- Reynolds D.A., 1995, Speaker identification and verification using Gaussian mixture speaker models. Speech Communication 17, 91-108.
- Roberts W.J.J., Ephraim Y., Sabrin H., 2005, Speaker classification using composite hypothesis testing and list decoding. IEEE Trans. on Speech and Audio Processing 13, 211-219.

KAYNAKLAR DİZİNİ (devam ediyor)

- Roch M., Hurtig R.R., 2002, The integral decode: A smoothing technique for robust HMM-based speaker recognition. *IEEE Trans. On Speech and Audio Processing* 10, 315-324.
- Rodriguez-Porcheron D., Faundez-Zanuy M., 1999, Speaker recognition with a MLP classifier and LPCC codebook. In: *Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 2*, 1005-1008.
- Shriberg E., Ferrer L., Venkataraman A., Kajarekar S., 2004, SVM modeling of “SNERF-Grams” for speaker recognition. In: *Proc. Int. Conf. On Spoken Language Processing*, pp. 1409-1412.
- Siohan O., Rosenberg A.E., Parthasarathy S., 1998, Speaker identification using minimum classification error training. In: *Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing (ICASSP) 1*, 109-112.
- Swets D.L., Weng J., 1996, Using discriminant eigenfeatures for image retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18, 831-836.
- Thyes O., Kuhn R., Nguyen P., Jungua J.-C., 2000, Speaker identification and verification using eigenvoices. In: *Proc. Int. Conf. On Spoken Language Processing 2*, 242-246.
- Turhal Ü.Ç., Gülmezoğlu M.B., Barkana A., 2005, Face recognition using common matrix approach, *EUSIPCO*
- Wan V., Campbell W.M., 2000, Support vector machines for speaker verification and identification. In: *Proc. IEEE Int. Workshop on Neural Networks for Signal Processing*, pp. 775-784.
- Wan V., Renals S., 2002, Evaluation of kernel methods for speaker verification and identification. In: *Proc. IEEE Int. Conf. On Acoustics, Speech and Signal Processing 1*, 669-672.

EKLER

Ek.1. Matlab Programları

Ek.2. Sözlük

MATLAB PROGRAMLARI

```
%*****  
%  
% FILE NAME : CVA_TEST_3S  
%  
% Speaker Identification by Using CVA Method  
%  
%*****  
  
clear all;  
NoOfUserTrain      = 20;  
NoOfUserTest       = 20;  
NoOfUtteranceTrain = 3;  
NoOfUtteranceTest  = 5;  
NoOfMinEigenvalues = 16;  
  
%----- READ TRAINING AND TESTING DATA -----  
  
training_features = Read_Training_Data;  
testing_features  = Read_Testing_Data;  
  
%----- TRAINING PHASE -----  
  
for user=1:NoOfUserTrain  
    Data=training_features{user};  
    %[m,n]=size(Data);  
    C = cov(Data);    % Normalized covariance  
    [u,v] = eig(C);   % matrices of eigenvalues (v) and eigenvectors (u)  
    %e = eig(C);      % returns a vector of the eigenvalues of matrix C  
    A0{user}=mean(Data);  
    P{user}=u(:,1:NoOfMinEigenvalues); % indiff space projection matrix  
end  
  
%----- TESTING PHASE -----  
  
NoOfMatch=0;  
NoOfTry =0;  
S1="";  
for utterance=1:NoOfUtteranceTest  
    for user=1:NoOfUserTest  
        Testdata=testing_features{user}{utterance};
```

```

dt0=mean(Testdata);
for model=1:NoOfUserTrain
    p=P {model};
    a0=A0 {model};
    DV=p'*(dt0-a0)';
    % Compute Euclidean distance between common and remaining vectors
    diff(utterance, user, model)=norm(DV);
end
NoOfTry=NoOfTry+1;
mini=min(diff(utterance,user,:));
if (mini==diff(utterance,user,user))
    NoOfMatch=NoOfMatch+1;
end;
S1=strcat(S1,sprintf('\nNoOfTry=%d NoOfMatch=%d\n',NoOfTry,NoOfMatch));
end
end
%S1
result=100*(NoOfMatch/NoOfTry)

```

```

%*****
%
% FILE NAME : CVA_BASED_GMM_TEST_3S
%
% Speaker Identification by Using GMM and CVA-based-GMM Methods
%
%*****

```

```

clear all;
NoOfUserTrain      = 20;
NoOfUserTest       = 20;
NoOfUtteranceTrain = 3;
NoOfUtteranceTest  = 5;
NoOfMinEigenvalues = 16;

```

```

%----- READ TRAINING AND TESTING DATA -----

```

```

training_features = Read_Training_Data;
testing_features  = Read_Testing_Data;

```

```

%----- TRAINING PHASE OF GMM -----

```

```

for No_of_Gaussians=1:32
    tic
    for user=1:NoOfUserTrain
        Data=training_features {user};
        % Training models with the input data using GMM
    end
end

```

```

    [mu_1,sigma_1,c_1]=gmm_estimate(Data',No_of_Gaussians);
    m_mu(user)={mu_1};
    m_sigma(user)={sigma_1};
    m_c(user)={c_1};
end
t_training1=toc;

%----- TESTING PHASE OF GMM -----

NoOfTry=0;
NoOfMatch=0;
S1="";
tic
for utterance=1:NoOfUtteranceTest
    for user=1:NoOfUserTest
        Testdata=testing_features{user}{utterance};
        for model=1:NoOfUserTrain
            % Compute multigaussian log-likelihood of test data for each model
            [IYM,IY]=lmultigauss(Testdata',m_mu{model},m_sigma{model},m_c{model});
            diff(utterance,user,model)=abs(mean(IY));
        end
        NoOfTry=NoOfTry+1;
        mini=min(diff(utterance,user,:));
        if (mini==diff(utterance,user,user))
            NoOfMatch=NoOfMatch+1;
        end;
    S1=strcat(S1,sprintf('\nNoOfTry=%d NoOfMatch=%d\n',NoOfTry,NoOfMatch));
    clear Testdata;
    end %user
end %utterance
result1(No_of_Gaussians)=(NoOfMatch/NoOfTry)*100;
t_testing1=toc
clear m_mu m_sigma clear m_c clear diff;
end %No_of_Gaussians

%----- TRAINING PHASE OF CVA BASED GMM -----

for No_of_Gaussians=1:32
    tic
    for user=1:NoOfUserTrain
        Data=training_features{user};
        %[m,n]=size(Data);
        C = cov(Data);    % Normalized covariance
        [u,v] = eig(C);    % matrices of eigenvalues (v) and eigenvectors (u)
        %e = eig(C);
        w=u(:,1:NoOfMinEigenvalues);
        W{user}=w;    % indiff space transformation matrix
    end
end

```



```

    Dataproj=w'*Data';
    % Training models with the input data using CVA based GMM
    [mu_1,sigma_1,c_1]=gmm_estimate(Dataproj,No_of_Gaussians);
    m_mu(user)={mu_1};
    m_sigma(user)={sigma_1};
    m_c(user)={c_1};
end %user
t_training2=toc;

%----- TESTING PHASE OF CVA BASED GMM -----

NoOfTry=0;
NoOfMatch=0;
S2="";
tic
for utterance=1:NoOfUtteranceTest
    for user=1:NoOfUserTest
        Testdata=testing_features {user} {utterance};
        for model=1:NoOfUserTrain
            w=W {model};
            Testdataproj=w'*Testdata';
            [IYM,IY]=lmultigauss(Testdataproj,m_mu {model},m_sigma {model},m_c {model});
            diff(utterance,user,model)=abs(mean(IY));
        end
        NoOfTry=NoOfTry+1;
        mini=min(diff(utterance,user,:));
        if (mini==diff(utterance,user,user))
            NoOfMatch=NoOfMatch+1;
        end;
    S2=strcat(S2,sprintf('\nNoOfTry=%d NoOfMatch=%d\n',NoOfTry,NoOfMatch));
    clear Testdata;
    end %user
end %utterance
result2(No_of_Gaussians)=(NoOfMatch/NoOfTry)*100;
t_testing2=toc
clear model_mu model_sigma model_c diff Dataproj Testdataproj p;
end %No_of_Gaussians

result(1,:)=result1;
result(2,:)=result2;
result

t_training1 % training time of GMM
t_testing1 % testing time of GMM
t_training2 % training time of CVA based GMM
t_testing2 % testing time of CVA based GMM

```

```

% Plot CVA&GMM comparison graphic
hold on
YLABEL('TANIMA ORANI %')
XLABEL('KATISIM SAYISI')
%TITLE('COMPARISON OF RECOGNITION PERFORMANCE')
text(7,78,'\leftarrow GMM');plot(result1,'b');
text(25,96,'\leftarrow OVY+GMM');plot(result2,'r')
result3(1:32)=82;text(25,84,'\downarrow OVY');plot(result3,'g')
hold off
save 'CVA_GMM_9_1_39_16_.txt' result -ascii

```

```

%*****
function c = Read_Training_Data
%
% Read training data from TIMIT and compute MFCCs
%
% Outputs:   c   mel cepstrum output
%
%*****

```

```

NoOfParameter    = 20;

```

```

fpath='D:/TIMIT/TIMIT/DR1/';
folder=dir(fpath);
m=size(folder,1);
%pre-process loop for 20 speaker
for i=3:22 %m
    training_data=[1:0];
    file=[fpath,folder(i).name,'%SA*.WAV'];
    filen=dir(file);
    n1=size(filen,1); %Number of utterance starting with 'SA'
    for j=1:n1
        file=[fpath,folder(i).name,'/',filen(j).name];
        [d1,sr] = readsph(file);
        training_data=[training_data;d1];
    end

    file=[fpath,folder(i).name,'%SI*.WAV'];
    filen=dir(file);
    n2=size(filen,1); %Number of utterance starting with 'SI'
    for j=1:1 %n2
        file=[fpath,folder(i).name,'/',filen(j).name];
        [d1,sr] = readsph(file);
        training_data=[training_data;d1];
    end
end
MFCC= melcepst(training_data,sr,'M',NoOfParameter,39,256,128);

```

```

    c{i-2}=MFCC;
end %speaker counter

```

```

%*****
function c = Read_Testing_Data
%
% Read testing data from TIMIT and compute MFCCs
%
% Outputs:    c    mel cepstrum output
%
%*****

```

```

NoOfParameter    = 20;

```

```

fpath='D:/TIMIT/TIMIT/DR1/';
folder=dir(fpath);
m=size(folder,1);
%pre-process loop for 20 speaker
for i=3:22 %m
    file=[fpath, folder(i).name, '/SX*.WAV'];
    filen=dir(file);
    n=size(filen,1); %Number of utterance starting with 'SX'
    for j=1:n
        file=[fpath, folder(i).name, '/', filen(j).name];
        [d1, sr] = readsph(file);
        d1=d1(1:16000);
        MFCC= melcepst(d1, sr, 'M', NoOfParameter, 39, 256, 128);
        c{i-2}{j}=MFCC;
    end
end %speaker counter

```

```

% *****
function [mu, sigm, c]=gmm_estimate(X, M, iT, mu, sigm, c, Vm)
% [mu, sigma, c]=gmm_estimate(X, M, <iT, mu, sigm, c, Vm>)
%
% X  : the column by column data matrix (LxT)
% M  : number of gaussians
% iT : number of iterations, by default 10
% mu : initial means (LxM)
% sigm: initial diagonals for the diagonal covariance matrices (LxM)
% c  : initial weights (Mx1)
% Vm : minimal variance factor, by default 4 -> minsig=var/(M^2Vm^2)
%*****

```

```

DEBUG=0;
GRAPH=0;

% GENERAL PARAMETERS
[L,T]=size(X);    % data length
varL=var(X)';    % variance for each row data;

min_diff_LLH=0.001; % convergence criteria

% DEFAULTS
if nargin<3 iT=10; end % number of iterations, by default 10
% if nargin<4 mu=X(:,[fix((T-1).*rand(1,M))+1]); end % 1- mu def: M rand vect.
for i=1:M
    aa(i)= i*fix(T/M);
end;
if nargin<4
    mu=X(:,aa);
end

if nargin<5 sigm= repmat(varL./(M.^2),[1,M]); end % sigm def: same variance
if nargin<6 c=ones(M,1)/M; end % c def: same weight
if nargin<7 Vm=4; end % minimum variance factor

min_sigm=repmat(varL./(Vm.^2*M.^2),[1,M]); % MINIMUM sigma!

if DEBUG sqrt(devs),sqrt(sigm),pause;end

% VARIABLES
lgam_m=zeros(T,M); % prob of each (X:,t) to belong to the kth mixture
lB=zeros(T,1); % log-likelihood
lBM=zeros(T,M); % log-likelihood for separate mixtures

old_LLH=-9e99; % initial log-likelihood

% START ITERATIONS
for iter=1:iT

    %if GRAPH graph_gmm(X,mu,sigm,c),pause;end

    if DEBUG disp(['***** ',num2str(iter),' *****']);end

    % ESTIMATION STEP

    [lBM,lB]=lmultigauss(X,mu,sigm,c);

    if DEBUG lB,B=exp(lB),pause; end

```

```

LLH=mean(lB);

%disp(sprintf('log-likelihood : %f,LLH));

lgam_m=lBM-repmat(lB,[1,M]); % logarithmic version
gam_m=exp(lgam_m);          % linear version

% MAXIMIZATION STEP
sgam_m=sum(gam_m);          % sum of gam_m for all X(:,t)

% gaussian weights
new_c=mean(gam_m)';

% means
% (convert gam_m and X to (L,M,T) and .* and then sum over T)
mu_numerator=sum(permute(repmat(gam_m,[1,1,L]),[3,2,1]).*...
    permute(repmat(X,[1,1,M]),[1,3,2]),3);
% convert sgam_m(1,M,N) -> (L,M,N) and then ./
new_mu=mu_numerator./repmat(sgam_m,[L,1]);

% variances
sig_numerator=sum(permute(repmat(gam_m,[1,1,L]),[3,2,1]).*...
    permute(repmat(X.*X,[1,1,M]),[1,3,2]),3);

new_sigm=sig_numerator./repmat(sgam_m,[L,1])-new_mu.^2;

% the variance is limited to a minimum
new_sigm=max(new_sigm,min_sigm);

% UPDATE

if old_LLH>=LLH-min_diff_LLH
    disp('converge');
    break;
else

    old_LLH=LLH;

    mu=new_mu;
    sigm=new_sigm;
    c=new_c;
end

end

%graph_gmm(X,mu,sigm,c);

```

```

%*****
function [YM,Y]=lmultigauss(x,mus,sigm,c)
% [IYM,IY]=lmultigauss(X,mu,sigm,c)
%
% computes multigaussian log-likelihood
%
% X : (LxT) data (columnwise vectors)
% sigm: (LxM) variances vector (diagonal of the covariance matrix)
% mu : (LxM) means
% c : (Mx1) the weights
%*****

DEBUG=0;
DEBUG1=0;

[L,T]=size(x);
M=size(c,1);

if DEBUG [ size(x), size(mus), size(sigm), size(c)], end

% repeating, changing dimensions:
X=permute(repmat(x',[1,1,M]],[1,3,2])); % (T,L) -> (T,M,L) one per mixture

Sigm=permute(repmat(sigm,[1,1,T]],[3,2,1]); % (L,M) -> (T,M,L)

Mu=permute(repmat(mus,[1,1,T]],[3,2,1]); % (L,M) -> (T,M,L)

if DEBUG size(X), size(Sigm), size(Mu), end

%Y=squeeze(exp( 0.5.*dot(X-Mu,(X-Mu)./Sigm))) % L dissapears: (L,T,M) -> (T,M)
IY=-0.5.*dot(X-Mu,(X-Mu)./Sigm,3);
% c,const -> (T,M) and then multiply by old Y
lcoi=log(2.*pi).*(L./2)+0.5.*sum(log(sigm),1); % c,const -> (T,M)
lcoef=repmat(log(c')-lcoi,[T,1]);

if DEBUG1 lcoi,lcoef,IY,pause;end

YM=lcoef+IY; % ( T,M ) one mixture per column
Y=lsum(YM,2); % add mixtures

if DEBUG [ size(YM) NaN size(Y) ], end

```

```

%*****
function graph_gmm(X,mi,sig,c,coefs,ft)
%
% graph_gmm(X,mi,sig,c,<coefs,ft>)
%
% plots the distribution of coefficients
%*****

DEBUG=0;
PRINT=0;
[L,T]=size(X);

if (nargin<5), coefs=1:L; end
if (nargin<6), ft=0; end

LL=length(coefs);

li=fix(sqrt(LL));
co=ceil(LL/li);

figure;
clf;

for ll=1:LL
l=coefs(ll);
xm=min(X(l,:));
xM=max(X(l,:));
x=(-ft*(xM-xm)+xm):((ft+1)*(xM-xm)/100):(xM+ft*(xM-xm));

subplot(li,co,ll);

histn(X(l,:),300);
hold on;

if DEBUG size(x),end

[laux,lmulti]=lmultigauss(x,mi(l,:),sig(l,:),c);
aux=exp(laux);
multi=exp(lmulti);

if DEBUG size(x),size(multi'),pause,end

hp=plot(x,multi,'r','Linewidth',3);
%xlim([-xM xM]);

ha=get(gca,'Children');

```

%it seem that the bars are children number 4

```
set(ha(2),'FaceColor',[ 0.8 0.8 0.8 ]);  
set(ha(2),'EdgeColor',[ 0.8 0.8 0.8 ]);%*  
plot(x,aux);  
end
```


SÖZLÜK

cepstrum	kepstrum
classification-oriented	sınıflandırmaya dayalı
code-book	kod tablosu
combination	birleşim
covariance	ortak değişinti
discriminant	ayırtaç
discriminative	ayırt edici
feature	öznitelik
feature extraction	öznitelik çıkarma
frame	çerçeve
function	işlev
iterative	döngülü, döngüsel
likelihood	olabilirlik
matching	eşleme, eşleştirme
maximization	enbüyütme
mel-scaled	mel-ölçekli
minimization	enküçültme
mixture	katışım
multilayer	çok katmanlı
optimal	en iyi
orthogonal	dik, dikgen
orthonormal	birimdik
pattern	örüntü
perceptron	algılayıcı
posteriori	sonsal
priori	önsel
probabilistic	olasılıksal
project	izdüşürmek, izdüşüm almak
projection	izdüşüm
range space	erim uzayı
rank of a matrix	matrisin kertesı
quantization	nicemleme
scatter	saçılım
spectrum	spektrum
state-of-the-art	en son teknoloji, en gelişmiş teknik
template	şablon
tutorial	eğitmençe; eğitim kursu
variance	sapma
warping	çarpıtma, bükme
windowing	pencereleme

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Selami Sadıç
Doğum Tarihi : 1967
Doğum Yeri : ESKİŞEHİR
Medeni Hali : Evli
Yabancı Dil : İngilizce
İş Adresi : 1.HİBM.K.lığı, Teknoloji ve Silah Sistem Geliştirme Başkanlığı
ESKİŞEHİR
Tel : +90 222 2375940 / 4728
e-mail : selami.sadic@hvkk.mil.tr

EĞİTİM

İlk-Orta Öğrenim : Ülkü İlkokulu, 1979
Tepebaşı Ortaokulu, 1982
Yunusemre Teknik Lisesi, Elektronik Bölümü, 1986
Lisans : Anadolu Üniversitesi, Mühendislik Mimarlık Fakültesi,
Elektrik-Elektronik Mühendisliği Bölümü, 1991
Yüksek Lisans : Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü,
Elektronik Mühendisliği, 1994
Tez Konusu : Kısıtlamasız Türkçe Konuşma Sentezleyici

İŞ TECRÜBESİ

1992-1997 : Araştırma görevlisi, Anadolu Üniversitesi,
Sivil HavacılıkYüksekokulu, ESKİŞEHİR
1998- : Yazılım Mühendisi, 1.HİBM.K.lığı, ESKİŞEHİR