

T.C.
KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRİK – ELEKTRONİK ANA BİLİM DALI
YÜKSEK LİSANS TEZİ

GÖRÜNTÜ DİZİLERİNDEN İNSAN YÜZ İFADE ANALİZİ

TURAN GÜNEŞ

HAZİRAN – 2009

Fen Bilimleri Enstitü Müdürünün onayı.

30/06/2009

Doç. Dr. Burak BİRGÖREN

Müdür V.

Bu tezin Yüksek Lisans tezi olarak Elektrik–Elektronik Anabilim Dalı standartlarına uygun olduğunu onaylarım.

Doç. Dr. Ediz POLAT

Anabilim Dalı Başkanı

Bu tezi okuduğumuzu ve Yüksek Lisans tezi olarak bütün gerekliliklerini yerine getirdiğini onaylarız.

Doç. Dr. Ediz POLAT

Danışman

Jüri Üyeleri

Prof. Dr. Şerafettin EREL

Doç. Dr. Ediz POLAT

Yrd. Doç. Dr. Tolga EREN

ÖZET

GÖRÜNTÜ DİZİLERİNDEN İNSAN YÜZ İFADE ANALİZİ

GÜNEŞ, Turan

Kırıkkale Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik – Elektronik Anabilim Dalı, Yüksek Lisans Tezi

Danışman : Doç. Dr. Ediz Polat

Haziran 2009, 74 sayfa

Yüz ifadeleri, insanlar arası iletişimde anlam bütünlüğünün sağlanması için büyük rol oynayan, sözlü olmayan işaretlerdir. İnsanoğlu yüz ifadelerini kavramada herhangi bir zorluk çekmezken, bu durum makineler için geçerli olmayıp, halen güvenilir ifade tanıma sistemleri üzerinde araştırmalar yapılmaktadır.

Bu çalışmada, insanın içinde bulunabileceği 6 ifade durumunun (öfke, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık) analizi gerçekleştirilmiştir. Bu amaçla, her bir ifade için alınan görüntülerin öznitelikleri Gabor dalgacıkları kullanılarak çıkartılmıştır.

Gabor özniteliklerinin 6 farklı ifadeye göre sınıflandırılmasında k-NN, SVM ve AdaBoost sınıflandırıcıları kullanılmıştır. Ayrıca sınıflandırma başarımlarını

arttırmak ve hız kazancı sağlamak için, çeşitli öznitelik dönüştürme ve öznitelik seçme teknikleri kullanılarak Gabor özniteliklerinin sayısı azaltılmıştır.

Kullanılan yöntemlerin ifade sınıflandırılması üzerindeki etkileri incelenerek detaylı bir karşılaştırma yapılmıştır.

Anahtar Kelimeler: Yüz İfade Analizi, İfade Sınıflandırması, Boyut İndirgemesi

ABSTRACT

HUMAN FACIAL EXPRESSION ANALYSIS FROM IMAGE SEQUENCES

GÜNEŞ, Turan

Kırıkkale University

Graduate School of Natural and Applied Sciences

Department of Electrical and Electronics Eng., M. Sc. Thesis

Supervisor : Assoc. Prof. Dr. Ediz Polat

June 2009, 74 pages

Facial expressions are non-verbal signs that play important role to provide complete meaning in human communication. While humans can easily comprehend the facial expressions, it is not valid for the computers, thus the researchers are still working on developing reliable facial expression recognition systems.

In this research, the analysis of 6 different human facial expressions (anger, disgust, fear, happiness, sadness and surprise) is performed from human facial images. For this purpose, the features for every facial expression are extracted using Gabor wavelets.

To classify the Gabor features, k-NN, SVM, and AdaBoost classifiers are carried out. The dimension of the Gabor feature space is also reduced by using

different feature extraction and feature selection techniques to improve the classification performances and to save time.

A detailed comparison is made to analyze the effect of the methods and techniques used for expression classification.

Key Words: Facial Expression Analysis, Expression Classification, Dimension Reduction

Babam, Annem, Ablam
ve
Kendim
iin

TEŐEKKÜR

Bu alıőmada desteklerini benden esirgemeyen, önerileri ile beni yönlendiren, büyük ilgi ve yardımlarını gördüğüm danışman hocam Sayın Do. Dr. Ediz POLAT'a;

alıőma süresince gösterdiği ilgi ve desteğinden dolayı Almanya Tuebingen Üniversitesi, Max-Planck Enstitüsü' nden Dr. Gökhan BAKIR'a;

Ve sonsuz sevgileriyle her zaman yanımda olan sevgili babam, annem ve ablama en içten teşekkürlerimi sunarım.

İÇİNDEKİLER

ÖZET	i
ABSTRACT	iii
TEŞEKKÜR	vi
İÇİNDEKİLER	vii
ÇİZELGELER DİZİNİ	ix
ŞEKİLLER DİZİNİ	xi
KISALTMALAR DİZİNİ	xiii
1. GİRİŞ	1
1.1. Yüz İfade Analizi	4
1.1.1. Yüz Belirleme	6
1.1.2. Yüz İfadelerine Ait Veri Çıkartılması	8
1.1.3. Yüz İfadelerinin Sınıflandırılması	8
1.2. Literatürde Yapılış Çalışmalar	9
1.3. Çalışmanın Amacı	11
2. MATERYAL VE YÖNTEM	12
2.1. Gabor Dalgacıları Kullanılarak Öznitelik Çıkartılması	12
2.1.1. Gabor Dalgacıları	13
2.1.1.1. Yüz Görüntülerinin Gabor Dalgacık Gösterimi	17
2.1.2. Öznitelik Çıkartılması	20
2.2. Öznitelik Boyutunun İndirgenmesi	21
2.2.1. Öznitelik Dönüşümü	23
2.2.1.1. Temel Bileşenler Analizi	23
2.2.2. Öznitelik Seçimi	25

2.2.2.1. Fisher Ölçüt Skoru	27
2.2.2.2. Öznitelik Seçimi İçin Sıfıncı-Norm Kullanılması (L0) ..	27
2.2.2.3. Ortak Bilgi Temelli Öznitelik Seçimi (MutInf)	28
2.2.2.4. Yinelemeli Öznitelik Eliminasyonu (RFE)	30
2.3. Gabor Özniteliklerinin Sınıflandırılması	31
2.3.1. k-En Yakın Komşuluk Sınıflandırma Algoritması (k-NN)	32
2.3.2. Destek Vektör Makineleri (SVMs)	34
2.3.2.1. İkili Sınıflandırma İçin SVM	34
2.3.2.1.1. Lineer SVM (Ayrılabilir Veri Durumu)	35
2.3.2.1.2. Lineer SVM (Ayrılamayan Veri Durumu)	38
2.3.2.1.3. Lineer Olmayan SVM	40
2.3.2.2. Çoklu Sınıflandırma İçin SVM	42
2.3.3. Adaptive Boosting (AdaBoost)	43
3. ARAŞTIRMA BULGULARI VE TARTIŞMA	45
3.1. Giriş	45
3.2. Yüz İfade Görüntüleri Veri Seti	45
3.3. Ön İşlemler	47
3.4. Deney 1	49
3.5. Deney 2	54
3.6. Deney 3	56
3.6.1. İki Sınıflı Öznitelik Seçimi	56
3.6.2. Çok Sınıflı Öznitelik Seçimi	64
3.7. Deney 4	65
4. SONUÇ	66
KAYNAKLAR	68

ÇİZELGELER DİZİNİ

ÇİZELGE

2.1	Dört Temel Kernel Tipi ve Fonksiyonları	42
2.2	Çoklu Sınıflandırma İçin AdaBoost.M1 Algoritması	44
3.1	Her bir ifadeye ait görüntü sayıları	47
3.2	Öznitelik indirgemesi yapılmadan elde edilen SVM sınıflandırıcı başarımları	53
3.3	TBA ile boyut indirgemesi yapıldıktan sonra elde edilen SVM sınıflandırıcı başarımları	55
3.4	One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümeleri	57
3.5	One-vs-One yaklaşımına göre oluşturulan öznitelik kümeleri	58
3.6	One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümelerinden %1 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları	60
3.7	One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümelerinden %3 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları	60
3.8	One-Vs-One yaklaşımına göre oluşturulan öznitelik kümelerinden %1 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları	61
3.9	One-Vs-One yaklaşımına göre oluşturulan öznitelik kümelerinden %3 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları	61
3.10	One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümelerinden %1 ve %3 oranlarında seçilen öznitelikler için alınan ortalama SVM sınıflandırıcı başarımları	62
3.11	One-Vs-One yaklaşımına göre oluşturulan öznitelik kümelerinden %1 ve %3 oranlarında seçilen öznitelikler için alınan ortalama SVM sınıflandırıcı başarımları	62

3.12 “One-Vs-Rest” öznitelik kümesi oluşturma yaklaşımına göre öznitelik seçme algoritmaları için ortalama SVM sınıflandırıcı başarımları	63
3.13 “One-Vs-One” öznitelik kümesi oluşturma yaklaşımına göre öznitelik seçme algoritmaları için ortalama SVM sınıflandırıcı başarımları	63
3.14 İki ve çok sınıflı kümelerden Fisher ile seçilen öznitelikler için SVM başarımları	64

ŞEKİLLER DİZİNİ

ŞEKİL

1.1	Yüz ifadelerinin oluşmasına neden olan kaynaklar	2
1.2	“Makine Öğrenme” problemleri için genel çözüm yöntemi	5
1.3	Yüz İfade Analizi Sistemi	7
2.1	Uzay bölgesinde karmaşık Gabor fonksiyonu ($v = 2, \mu = 0, f = \sqrt{2}, k_{max} = \frac{\pi}{2}, \sigma = \pi$) (a) Fonksiyonunun gerçek kısmı (çift simetrik) (b) Fonksiyonunun sanal kısmı (tek simetrik)	16
2.2	Gabor Filtre Bankasının Reel Kısımları (3 Farklı Frekans ve 4 Farklı Yönelim İçin)	17
2.3	Gabor filtrelerinin herhangi bir görüntüye uygulanması durumunda elde edilen Gabor görüntüleri (a) Orijinal görüntü (b) Gabor görüntülerinin büyüklük cevapları	19
2.4	Denetimli ve Denetimsiz Öğrenme Algoritmalarına Göre Boyut İndirgeme Teknikleri	22
2.5	k-NN Sınıflandırma Örneği	33
2.6	SVM, maksimum marjin γ olmak üzere, pozitif (beyaz daireler) ve negatif (siyah daireler) örnekleri ayıran $\langle w, x \rangle + b = 0$ aşırı düzlemini bulur	35
2.7	Geçici Değişkenlerin Gösterimi	39
2.8	Lineer olmayan SVM gösterimi. $\phi = \mathbb{R}^2 \rightarrow \mathbb{R}^3$ şeklindeki eşleştirme $\phi([x_1, x_2]) = [z_1, z_2, z_3]$ ile tanımlanır ($z_1 = x_1^2, z_2 = x_2^2$ ve $z_3 = \sqrt{2}x_1x_2$)	40
3.1	6 adet yüz ifadesine bir örnek (Cohn and Kanade DFAT-504 veri seti)	46
3.2	İki farklı görüntünün yüz kısımlarının ayrıştırılmış halleri	48

3.3	Farklı kişilere ait göz hizalı 6 yüz ifadesi	49
3.4	Öznitelik Kümesi Genel Yapısı	50
3.5	Kodlama Matrisi ve İfade Görüntü Sayılarına Göre Boyutu	50
3.6	Öznitelik indirgemesi yapılmadan elde edilen k-NN sınıflandırıcı başarımları	54
3.7	TBA ile boyut indirgemesi yapıldıktan sonra elde edilen k-NN sınıflandırıcı başarımları	55
3.8	AdaBoost Sınıflandırıcısı Başarımı	65

KISALTMALAR DİZİNİ

k-NN	k-Nearest Neighbor
SVM	Support Vector Machine
SVM 1	SVM (One-Vs-One) Classifier
SVM 2	SVM (One-Vs-Rest) Classifier
SVM 3	SVM (Multi-Class) Classifier
AdaBoost	Adaptive Boosting
TBA	Temel Bileşenler Analizi
Fisher	Fisher Criterion Score
L0	Zero-Norm Feature Selection
MutInf	Mutual Information Based Feature Selection
RFE	Recursive Feature Elimination

1. GİRİŞ

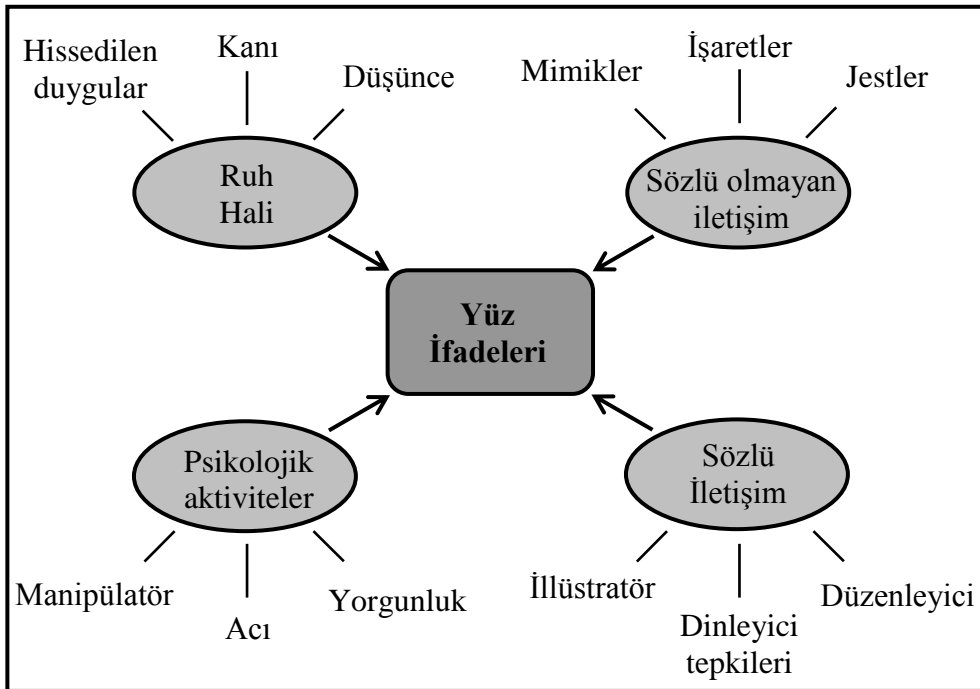
İnsanlar arasında gerçekleştirilen yüz-yüze iletişim, gelişmiş insan-bilgisayar ara yüzlerinin tasarlanmasında ideal bir model teşkil etmektedir⁽¹⁻³⁾. İnsan iletişiminin temel özelliği, iletişim kanallarının çok çeşitli olması ve yapılan iletişimin farklı şekillerde gerçekleşmesidir. Kanal, iletişimin yapıldığı ortamı belirtirken, iletişim şekli, dış dünyadan sinyallerin alınma biçimini ifade etmektedir. İnsanlardaki iletişim kanalları, konuşulan seslerin ve seslere ait vurgulamaların taşındığı işitsel kanal ile yüz ifadeleri ve beden hareketlerinin taşındığı görsel kanaldan oluşmaktadır. İnsanlardaki iletişim yapısı çok gelişmiş olup, herhangi bir iletişim kanalındaki problem, mesajın bir diğer iletişim kanalı tarafından iletilmesi ile telafi edilmektedir.

İnsanlardaki iletişimin bu özelliğinden yola çıkılarak gürbüz ve güvenilir insan-bilgisayar ara yüzlerinin geliştirilmesi için birçok çalışma yapılmıştır. Örneğin, Chen vd.⁽⁴⁾ ve De Silva⁽⁵⁾ yüz ve ses ifadelerinin birleştirilmesi üzerine çalışmışlardır. Fakat çalışmalar çoğunlukla her bir iletişim kanalı için ayrı ayrı yapılmıştır⁽⁶⁾.

Yüz yüze yapılan iletişim, insanlar arasındaki iletişimde çok büyük rol oynamaktadır. Çünkü insan yüzü diğer iletişim şekillerinin bir göstergesi ve bu göstergenin kaynağı olan herhangi bir yüz ifadesi de, bir insanın hissi durumunun, anlayış tavrının, karakterinin ve içinde bulunduğu psikolojik durumunun görünen belirtisidir⁽⁷⁾. Yapılan birçok sosyo-psikolojik çalışma, karşılıklı yapılan konuşmalarda yüz ifadelerinin büyük etkisinin olduğunu göstermiştir^(8,9). Mehrabian⁽¹⁰⁾, yüz ifadelerinin karşımızdaki dinleyiciler üzerinde çok büyük etkisinin olduğunu belirtmiştir ki, bu etkilerin yaklaşık %55' i konuşmacının

açıklamalarındaki yüz ifadelerinden, %38' i konuşmadaki kelime tonlamalarından ve %7' si de konuşulan kelimelerden kaynaklanmaktadır. Bu açıklamalar, insanlar arasındaki iletişimde yüz ifadelerinin en baskın özellik olduğunu açıkça belirtmektedir.

Yüz ifadeleri biçimsel olarak yüzdeki belirli bölgelerin şekil değiştirmesi ile ortaya çıkmaktadır. Özellikle yüz kaslarının kasılarak göz kapakları, kaşlar, burun, dudaklar ve yüz derisi gibi yüz özniteliklerinde geçici değişimler oluşturması sonucu ortaya çıkan, zamana bağımlı ve birkaç saniye süren bu değişimlerin yeri, yoğunluğu ve dinamiğinin ölçülmesi, yüz ifadelerinin tanınarak sınıflandırılması için önem taşıyan niteliklerdir. Bu değişimlerin kişiye, yaşa, ırka, cinsiyete göre miktar ve zaman olarak farklılık göstermesi, problemi çok boyutlu hale getirip zorlaştırmaktadır⁽¹¹⁾. Şekil 1.1'de yüz ifadelerinin oluşmasına neden olan kaynaklar gösterilmiştir.



Şekil 1.1 Yüz ifadelerinin oluşmasına neden olan kaynaklar

Sabit görüntülerin ve görüntü dizilerinin kullanılmasıyla birlikte yüz ifade analizi, sadece psikoloji alanında yapılan bir araştırma konusu olmaktan çıkmıştır. Bununla birlikte görüntülerde yüz bulma (face detection), yüz takip etme (face tracking) ve yüz tanıma (face recognition) alanındaki ilerlemeler, yüz ifade analizi üzerine yapılan çalışmaların artmasına sebep olmuştur⁽¹²⁾.

Taşıdıkları bilginin öneminden dolayı yüz ifadeleri, bir insan-makine ara yüzü olarak örneğin, bir müşteri veya kullanıcının pazarlık içeren alım-satım işlemi gibi durumlarda iken içinde bulunduğu duygusal durumun düzgün değerlendirilmesinin gereksinim duyulduğu servislerde, otomatikleştirilmiş önlem ve hazırlıkları mümkün kılacaktır. Bununla birlikte yüz ifade analizi sistemleri sözlü olmayan haberleşmede, davranış biliminde, tıpta, psikolojide, psikiyatride, nörolojide, acı tespitlerinde, yalan belirlemede, çeşitli robotik uygulamalarında ve insan-bilgisayar ara yüzlerinin gerekli olduğu birçok alanda kullanılmaktadır⁽¹³⁻¹⁶⁾.

Bu çalışmada “Yüz İfade Analizi”, Ekman ve Friesen⁽¹⁷⁾ tarafından tanımlanan 6 temel duyguya ait yüz ifadelerinin (“öfke”, “iğrenme”, “korku”, “mutluluk”, “üzüntü” ve “şaşkınlık”) sınıflandırılması çerçevesinde gerçekleştirilmiştir. Çalışmanın ilk bölümünde “Yüz İfade Analizi” nin genel olarak nasıl gerçekleştirildiği anlatılmış, analizin hangi bölümlerden oluştuğu ve bu bölümlere ait bilgiler verilmiş, geçmişten günümüze “Yüz İfade Analizi” üzerine literatürde yapılmış olan çalışmalar derlenmiş ve bu çalışmanın amacından bahsedilmiştir.

Çalışmanın ikinci bölümünde, yüz ifade analizinde kullanılmış olan yöntemlere ilişkin bilgiler sunulmuştur. Öncelikle yüz görüntülerine ait özniteliklerin çıkartılmasında kullanılan “Gabor dalgacıkları” anlatılmıştır. Daha sonra, çok

boyutlu Gabor öznitelik boyutunun azaltılması amacıyla, bir öznitelik dönüşüm tekniği olan “Temel Bileşenler Analizi (Principle Component Analysis – PCA)” ne ilişkin bilgi verilmiştir. Hemen ardından, yine öznitelik boyutunun azaltılmasında kullanılan öznitelik seçme algoritmaları “Fisher Ölçüt Skoru (Fisher Criterion Score – Fisher)”, “Öznitelik Seçiminde Sıfıncı Norm Kullanılması (Zero Norm Feature Selection – L0)”, “Ortak Bilgi Temelli Öznitelik Seçilmesi (Mutual Information Based Feature Selection – MutInf)” ve “Yinelemeli Öznitelik Eliminasyonu (Recursive Feature Elimination – RFE)” hakkında açıklamalar yapılmıştır. Son olarak Gabor özniteliklerinin ilgili ifadelerle göre sınıflandırılmasında kullanılan “k-En Yakın Komşuluk (k-Nearest Neighbor – k-NN)”, “Destek Vektör Makineleri (Support Vector Machines – SVMs)” ve “AdaBoost (Adaptive Boosting)” algoritmalarına değinilmiştir.

Üçüncü bölümde, sözü edilen algoritmalar ve teknikler kullanılarak yüz ifadelerinin sınıflandırılmasına yönelik deneyler yapılmıştır. Dördüncü ve son bölümde, elde edilen sonuçlarla ilgili genel bir değerlendirme yapılmıştır. Hangi teknik ve algoritmaların doğru ve hızlı yüz ifade analizi sistemi için daha uygun olduğu, hangilerinin ise olmadığı nedenleri ile birlikte anlatılmaya çalışılmıştır.

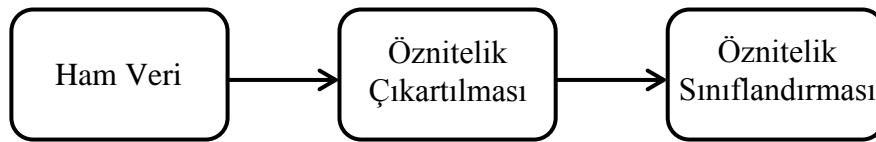
1.1. Yüz İfade Analizi

İnsanlarda öğrenmenin nasıl gerçekleştiği psikologlar tarafından hala incelenmekte olan bir konudur. Genel olarak öğrenme; geçmiş deneyimler, örnekler ve merakla gelen bir işlemler bütünü olup, kesin olarak açıklanabilmesi pek mümkün değildir. Yüz tanıma örneğini ele alalım. Hemen her gün aile bireylerini ve arkadaşlarımızı yüzlerine bakarak veya fotoğraflarından, saç stillerindeki

farklılıklara, makyaj yapıp yapmadıklarına ve duruşlarındaki değişikliklere rağmen hiç zorluk çekmeden tanıyabiliriz. Bunu bilinçsizce ve nasıl gerçekleştiğini açıklayamadan gerçekleştiririz.

Yapay Sinir Ağlarının (YSA) bir altkümesi olan “Makine Öğrenmesi” nin amaçlarından biri de, insanlardaki öğrenmenin nasıl gerçekleştiğini anlamamıza yardımcı olmaktır. Makine öğrenmesi tanımsal olarak, “bir problemin çözümünde örnekler veya geçmiş deneyimlere ait verilerin kullanılarak bilgisayarların programlanmasıdır” şeklinde ifade edilir.

Makine Öğrenme uygulamaları genellikle farklı nesnelerin sınıflandırılmasını amaçlar. Yüz tanıma, ses tanıma, karakter tanıma, hava durumu tahmini, yüz ifade analizi çalışmaları Makine Öğrenme uygulamalarından sadece bir kaçıdır. Farklı nesnelerin sınıflandırılması işlemi, bir başka deyişle Makine Öğrenme problemlerinin çözümü klasik olarak Şekil 1.2’ de gösterildiği gibi gerçekleştirilmektedir^(18,19).



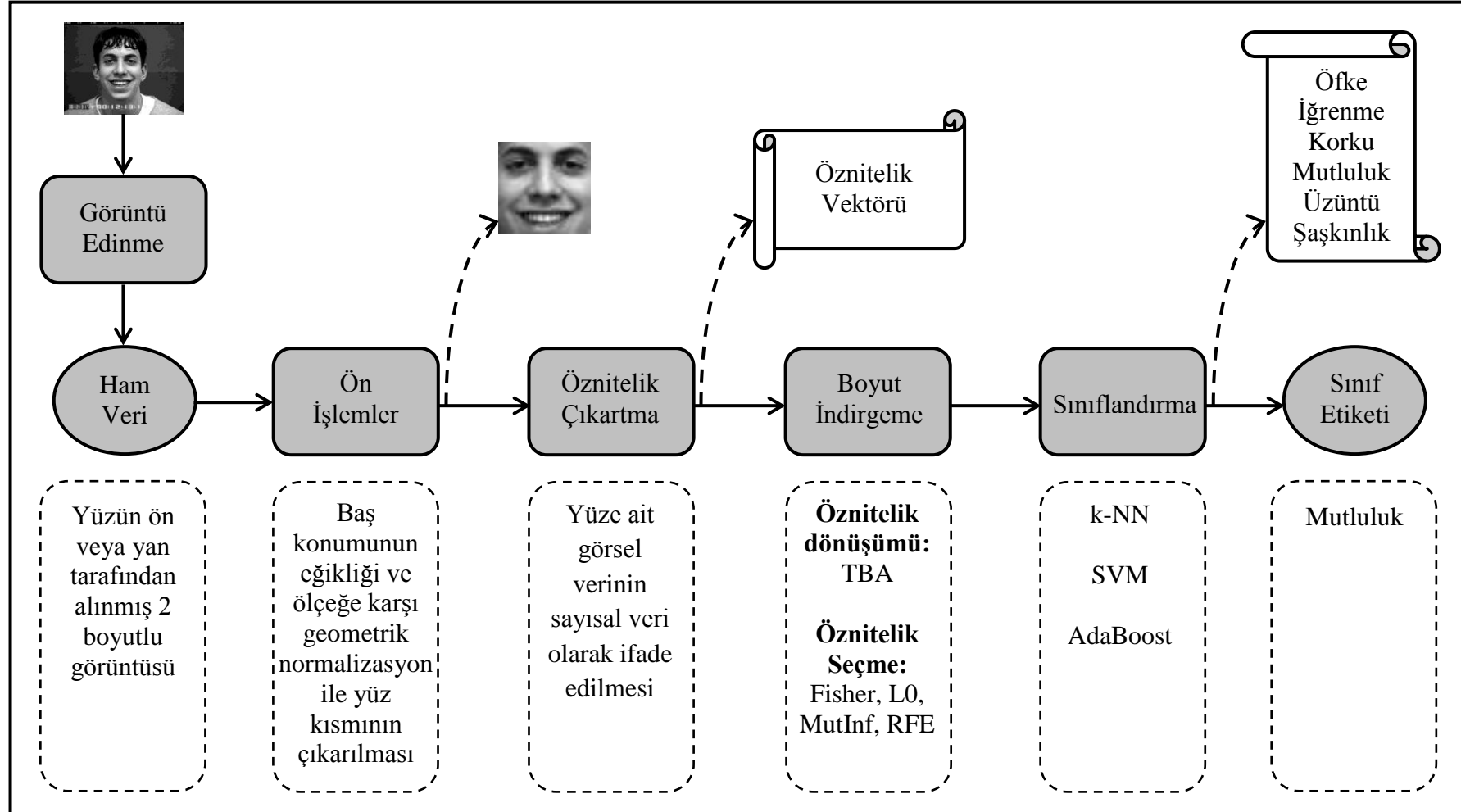
Şekil 1.2 “Makine Öğrenme” problemleri için genel çözüm yöntemi

Bir makine öğrenme problemi olarak “Yüz İfade Analizi” karmaşık bir sisteme sahiptir. Cinsiyet, etnik yapı ve yaş farklılıklarının yanı sıra, yüzde olması muhtemel sakal, kozmetik ürünler, gözlükler ve saçlar nedeniyle kişiden kişiye değişen yüz ifadelerini tanıyan, genel ve geçerli bir sistem oluşturmak bu karmaşıklığın temel nedenidir. Ayrıca yüz görünümleri duruş ve ışıktaki değişimlere göre farklılıklar gösterebilir⁽¹⁶⁾.

Pantic ve Rothkrantz⁽²⁰⁾ yüz ifade analizinde 3 temel problem bulunduğunu belirtmişlerdir: Bir görüntüdeki yüz kısmının belirlenmesi, yüz ifadelerine ait özniteliklerin çıkartılması ve yüz ifadelerinin sınıflandırılması. Pantic ve Rothkrantz⁽²⁰⁾, in işaret ettiği problemler ve makine öğrenme yaklaşımı çerçevesinde, yüz ifade analizi üzerine bir çalışma yapılması için öncelikle, resimlerdeki ifadelerin taşındığı kısım, yani yüz görüntüleri çıkarılır. Ardından yüz görüntülerinde bulunan ifadeleri temsil eden bilgiler (öznitelikler) çıkartılır. Çıkartılan öznitelikler, her bir ifade için sınıflandırılırlar. Ayrıca hızlı ve doğru bir analiz yapılması amacıyla, sınıflandırmadan hemen önce gereksiz öznitelikler atılarak öznitelik boyutu azaltılabilir. “Yüz İfade Analizi” için genel sistem yapısı Şekil 1.3’ teki gibi gösterilirken, sistemin temel bileşenlerine ilişkin bilgiler sonraki bölümlerde verilmiştir.

1.1.1. Yüz Belirleme

Bir yüz ifadesinin sınıflandırılabilmesi için öncelikle görüntüdeki “yüz” kısmının belirlenmesi gerekmektedir. Görüntüler genellikle kişilerin ön yüzlerinden çekilmiş şekildedir. Buradaki en önemli problem, görüntüdeki yüz bölgesine ait kısmın doğru bir şekilde çıkarılmasıdır. Çünkü yüz bölgesinin konumu ve büyüklüğü her bir görüntü için farklılık gösterebildiğinden, görüntülerden yüz kısmının çıkarılmasında aynı türden bir şablon kullanılması pek mümkün olmamaktadır⁽²⁰⁾. Bu nedenle yüz kısımları, çeşitli yüz belirleme teknikleri^(21,22) veya el ile yapılan çalışmalar sonucu çıkarılmaktadır.



Şekil 1.3 Yüz İfade Analizi Sistemi

1.1.2. Yüz İfadelerine Ait Veri Çıkartılması

Görüntülerde ifadeleri taşıyan yüz kısımlarının belirlenmesinden sonra her bir ifadeye ait veri, yani öznitelikler çıkartılır. Bu işlem görsel verinin sayısal veriye dönüştürülmesi şeklinde açıklanabilir. Yüz bileşenlerine (ağız, kaşlar, burun) ait olan noktalardan elde edilen “Geometrik Öznitelikler” ile yüz bölgesindeki değişimlerden (kırışıklık, izler) kaynaklanan “Görüntüye Dayalı Öznitelikler” olmak üzere 2 farklı tipte öznitelik bulunmaktadır⁽¹⁶⁾.

Öznitelik çıkartılması aynı zamanda öznitelik uzayının boyutunun azaltılmasını da kapsamaktadır. Çünkü bir çok örüntü tanıma sisteminde bulunan büyük boyutlardaki veriler, problem çözümünü neredeyse imkansız hale getirmektedir. Boyut azalması ile verideki esas bilgi kaybolmamalı, bunun yanında boyutu azalan veri, orijinal veriye göre sınıf ayrımı çok daha fazla olan öznitelikleri içermelidir.

1.1.3. Yüz İfadelerinin Sınıflandırılması

Bir yüz görüntüsüne ait öznitelikler çıkartıldıktan sonra yüz görüntüsünün taşımış olduğu ifade belirlenir. Bu işlem bir sınıflandırıcı tarafından gerçekleştirilir. Makine, ifadelere ait belirli sayıda örnekle eğitilir ve bir karar kuralı oluşturulur. Diğer örnekler bu kurala göre her bir ifade için sınıflandırılırlar.

İfade sınıflandırılmasında literatürde k-En Yakın Komşuluk, Destek Vektör Makineleri, Saklı Markov Modelleri ve Karar Ağaçları gibi birçok sınıflandırıcı kullanılmaktadır⁽¹⁶⁾.

1.2. Literatürde Yapılmış Çalışmalar

1872 yılında Darwin⁽²³⁾'in insanlarda doğuştan gelen bazı duyguları olduğunu belirtmesi, “Yüz İfade Analizi” üzerine yapılan çalışmaların temelini oluşturmuş ve “Yüz İfade Analizi”, birçok davranış bilimci için popüler bir araştırma konusu olmuştur.

Darwin⁽²³⁾'in yapmış olduğu çalışmadan yola çıkarak, 1971’ de Ekman ve Friesen⁽¹⁷⁾ ve 1983’te Izard vd.⁽²⁴⁾ insanlarda 6 temel duygu bulunduğunu ve bu duygusal ifadelerin her birine karşılık gelen yüz ifadelerinin olduğunu belirtmişlerdir. Farklı kültür ve milletler için geçerli kabul edilmiş olan bu duygular “öfke”, “iğrenme”, “korku”, “mutluluk”, “üzüntü” ve “şaşkınlık” olarak gösterilmiştir.

Önceleri psikoloji alanında bir araştırma konusu olan yüz ifade analizi üzerine yapılan ilk çalışma, 1978’ de Suwa vd.⁽²⁵⁾ tarafından bir görüntü dizisi üzerinde 20 nokta belirlenerek gerçekleştirilmiştir.

1980’ lerde Marcelja⁽²⁶⁾ ve Daugman⁽²⁷⁾ tarafından insan görsel sisteminin 2-boyutlu karmaşık Gabor fonksiyonları ile modellenilebildiğinin gösterilmesi, örüntülere ait özniteliklerin çıkartılmasında önemli bir adım olmuştur.

Suwa vd.⁽²⁵⁾ nin 1978’ de yapmış oldukları çalışmadan sonra, bilgisayar temelli yüz ifade analizi 1990’ lara kadar yapılmamıştır. Mase⁽²⁸⁾ yüz ifadelerinin tanınmasında optik akış tekniğini kullanmıştır.

Black ve Yacoob⁽²⁹⁾ görüntüdeki hareketlerden yola çıkarak elde ettikleri verileri 6 temel yüz ifadesinin sınıflandırmasında kullanmışlardır.

Rosenblum, Yacoob ve Davis⁽³⁰⁾ yüz üzerindeki bölgelerde optik akış hesabı yapmışlar ve Radial Basis Fonksiyon Ağı kullanarak ifade sınıflandırmasını gerçekleştirmişlerdir.

Essa ve Pentland⁽³¹⁾ bölgesel optik akış metodunu kullanarak ifadeleri sınıflandırmışlardır.

Otsuka⁽³²⁾ 6 farklı yüz ifadesini sınıflandırmak amacıyla Saklı Markov Model (Hidden Markov Model – HMM) temelli sınıflandırıcılar kullanmıştır.

Donato vd.⁽⁷⁾ yüz ifadelerinin tanınmasında farklı tiplerdeki öznitelikleri karşılaştırmışlardır.

Martinez⁽³³⁾ yüz görüntülerinin, farklı aydınlık ortamlar, yüz ifadeleri ve görüntüyü kapayan engelleyici durumlar altında tanınması üzerine çalışmıştır. Bu amaçla, Saklı Markov Modelleri kullanılmıştır.

Oliver vd.⁽³⁴⁾ ağız şeklinin değişiminden kaynaklanan öznitelikleri Saklı Markov Model temelli tanıma sisteminde kullanmışlar ve “ifadesizlik”, “mutluluk” ve “üzüntü” gibi ifadelerin sınıflandırması üzerine çalışmışlardır.

Kotsia ve Pitas⁽³⁵⁾ görüntü dizilerinden yüz ifade analizini Destek Vektör Makineleri kullanarak gerçekleştirmiştir. Çalışmalarında, dizideki ilk görüntü üzerine yerleştirilen noktalar referans alınıyor ve sonraki görüntülerle olan yer değiştirmeye göre ifade sınıflandırması yapılıyordu.

Bartlett vd.⁽³⁶⁾, Cohn and Kanade veri setindeki yüz görüntülerinden farklı ifadeler için elde ettikleri Gabor özniteliklerini sınıflandırmak için, yeni geliştirdikleri AdaSVM metodunu kullanmışlardır.

1.3. Çalışmanın Amacı

Bu çalışmada, 6 farklı yüz ifadesinin sınıflandırılması hedeflenmiştir. İnsanların herhangi bir zorluk çekmeden tanıyabildikleri bu ifadelerin makinelerce olabildiğince doğru ve hızlı bir şekilde sınıflandırılması ancak, hızlı ve güvenilir algoritmalar kullanılarak, makinelerin önceden doğru örneklerle eğitilmesi ile mümkün olacağı inkar edilemez bir gerçektir.

Bu amaç doğrultusunda, yapılan çalışmada, literatürde sık rastlanılan birçok algoritma kullanılarak, hızlı ve doğru yüz ifade sınıflandırılması için kapsamlı bir inceleme gerçekleştirilmiştir. Çalışmanın temeli, farklı ifadelerle ait yüz görüntülerine ilişkin özniteliklerin farklı Gabor dalgacıkları kullanılarak elde edilmesi üzerine kurulmuştur. En iyi ifade analizi sistemine sahip olan insan örneğinden yola çıkıldığında, insanlardaki görme merkezindeki hücrelerin tepki işaretlerinin Gabor dalgacık fonksiyonları ile modellenmesi, öznitelik çıkartılmasında Gabor dalgacıklarının kullanılmasının en önemli nedeni olmuştur.

Çıkarılan Gabor öznitelikleri k-NN, SVM ve AdaBoost algoritmaları kullanılarak sınıflandırılmıştır. Ayrıca, çeşitli öznitelik dönüştürme ve öznitelik seçme algoritmaları kullanılarak yapılan boyut azaltılması ile sınıflandırıcı başarımlarının artırılması amaçlanmıştır. Kullanılan boyut indirgeme algoritmalarının temel özelliği hızlı sonuçlar ürettiği olmalarıdır.

İfade sınıflandırması üzerindeki etkileri bakımından kullanılan algoritmalar ve teknikler arasında kapsamlı bir karşılaştırma yapılmıştır. Ayrıca, yüz ifadelerinin doğru bir şekilde sınıflandırılmasına yönelik çeşitli yaklaşımlar kullanılarak sınıflandırma başarımları arttırmaya çalışılmıştır.

2. MATERYAL VE YÖNTEM

2.1. Gabor Dalgacıkları Kullanılarak Öznitelik Çıkartılması

Öznitelik çıkartılması, yüz görüntülerinde farklı ifadeler sonucu oluşan değişikliklerin sayısal gösterimidir. Yüz ifadelerinin neden olduğu değişikliklere ait bilgilerin, yani özniteliklerin çıkartılmasında genel olarak;

- Geometrik tabanlı öznitelik çıkartım metotları
- Görüntü tabanlı öznitelik çıkartım metotları

olmak üzere 2 tür yaklaşım vardır. Geometrik öznitelikler, yüz bileşenlerinin (ağız, göz, kaşlar, burun) yerleri ve şekillerini ifade ederler. Yüz bileşenlerine ait öznitelik noktaları, yüz geometrisini belirten bir öznitelik vektörü oluşturmak için çıkartılır. Görüntüye dayalı metotlarda ise öznitelik vektörü, Gabor dalgacıkları gibi bir görüntü filtresinin yüz resminin tümüne veya belli bir bölgesine uygulanması suretiyle elde edilir⁽³⁷⁾.

Zhang vd.⁽³⁸⁾, öznitelik çıkartılmasındaki bu iki yaklaşımı yüz ifadelerinin tanınması için karşılaştırmışlardır. Bir yüz görüntüsünden alınan 34 adet referans noktası ile bu noktalara uygulanan farklı Gabor filtreleri ile elde edilen 612 Gabor dalgacık katsayısı öznitelik vektörleri olarak kullanılmış ve 6 farklı yüz ifadesinin tanınmasında Gabor filtreli elde edilen özniteliklerin daha üstün başarımlar gösterdiği görülmüştür. Donato vd.⁽⁷⁾ de ifade tanınmasında optik akış, temel bileşenler analizi, bağımsız bileşenler analizi, yerel öznitelik analizi ve Gabor filtre tekniklerini karşılaştırmışlar ve yine en iyi sonucu Gabor filtreleri ile elde edilen öznitelikleri kullandıkları durumda almışlardır. Çünkü Gabor filtreleri ile görüntüde var olan ışık

dağılımının homojen olmamasının oluşturduğu negatif etkiler yok olmakla birlikte, bu filtrelemenin küçük miktardaki yer değiştirme ve deformasyonlara karşı daha az duyarlı olması da öznelik çıkartılmasında diğer yöntemlere üstünlük sağlamasına neden olmaktadır.

Literatürde yapılmış olan bu çalışmalar göstermiştir ki; ifade analizinde görüntüye dayalı öznelik çıkartma metotları (özellikle Gabor filtreleri) ile daha hassas ve kesin sonuçlar alınmıştır. Bu yüzden, yapılan çalışmada yüz görüntülerine ait öznelikler Gabor filtreleri kullanılarak elde edilmiştir.

2.1.1. Gabor Dalgacıkları

1946 yılında Dennis Gabor⁽³⁹⁾ sinyallerdeki karakteristik bilgilerin elde edilebilmesi amacıyla, sinyale uygulanacak bir fonksiyonun zaman ve frekans eksenlerindeki bant genişliğinin, Fourier analizi belirsizlik ilkesine göre minimum olması gerektiğini belirtmiştir. Gabor⁽³⁹⁾, bu fonksiyonun bir Gauss sinyali ile modüle edilmiş karmaşık üstel bir sinyal ile elde edilebileceğini göstermiş ve bir boyutlu Gabor fonksiyonunu üretmiştir.

Memeli canlılardaki görüntü sisteminin Gabor filtreleri ile modellenebileceğini gösteren Marcelja⁽²⁶⁾, görüntü sisteminde bulunan hücrelerdeki alıcı alanların bir görüntü karşısında verdikleri sinyallerin çift veya tek simetrik yapıda olduğunu belirtmiştir. Ayrıca, memeli canlılardaki görüntü merkezinde bulunan hücreler üzerine yapılan fizyolojik çalışmalar bu hücrelerin farklı yönelim ve uzaysal frekanslarda sinyal verdiklerini göstermiştir. Bu bilgiler ışığında Daugman⁽²⁷⁾, görüntü merkezindeki alıcı hücrelerin verdikleri tepki sinyallerinin teorik olarak 2 boyutlu karmaşık Gabor fonksiyonları şeklinde ifade edilebileceğini göstermiştir.

Daugman'ın görüntü sistemini 2 boyutlu karmaşık Gabor fonksiyonları ile matematiksel olarak göstermesi, Gabor filtrelerinin doku bölütlemesi⁽⁴⁰⁾, parmak izi tanıma⁽⁴¹⁾, yüz tanıma⁽⁴²⁾ ve iris tanıma⁽⁴³⁾ gibi birçok bilgisayarlı görü ve görüntü tanıma uygulamasında öznelik çıkartılmasında yaygın olarak kullanılmasına imkan sağlamıştır. Bunun yanı sıra Hubel ve Wiesel'in⁽⁴⁴⁾ görüntü sistemindeki alıcı alanların birer kenar detektörü işlevi gördüklerini göstermesi, Gabor filtrelerinin de iyi birer kenar detektörü olduklarını belirtmektedir. Bu özellik sayesinde, yüz ifadelerine ait öznelikler çıkartılırken göz, ağız, kaşlar, burun gibi ifadeyi temsil eden önemli noktaların belirlenmesi gerçekleşmektedir.

Gabor filtresi, impuls cevabı, bir Gauss fonksiyonuyla harmonik bir fonksiyonun çarpılmasıyla tanımlanan lineer bir filtredir. Konvolüsyon işleminin çarpma özelliğinden dolayı, Gabor filtresinin impuls cevabının Fourier dönüşümü, harmonik fonksiyonun Fourier dönüşümü ile Gauss fonksiyonunun Fourier dönüşümünün konvolüsyonudur⁽⁴⁵⁾. Buradan, memeli canlılarda görme sisteminin modellendiği ve Daugman tarafından 2 boyutlu hale getirilen karmaşık Gabor dalgacık fonksiyonu $\psi_{\mu,\nu}(z)$;

$$\psi_{\mu,\nu}(z) = \frac{\|k_{\mu,\nu}\|^2}{\sigma^2} e^{-\frac{\|k_{\mu,\nu}\|^2 \|z\|^2}{2\sigma^2}} \left[e^{ik_{\mu,\nu}z} - e^{-\frac{\sigma^2}{2}} \right] \quad 2.1$$

olarak tanımlanır. 2.1 eşitliği ile verilen 2 boyutlu karmaşık Gabor fonksiyonda

$\frac{\|k_{\mu,\nu}\|^2}{\sigma^2}$ terimi görüntülerdeki frekansa bağımlı güç spektrumunun azalmasını telafi

ederken⁽⁴⁶⁾, $e^{-\frac{\|k_{\mu,\nu}\|^2 \|z\|^2}{2\sigma^2}}$ terimi Gauss dalgasını, $e^{ik_{\mu,\nu}z}$ terimi de karmaşık üstel sinyali

ifade etmektedir. $e^{-\frac{\sigma^2}{2}}$ terimi ise Gabor filtresinin görüntüdeki farklı ışık

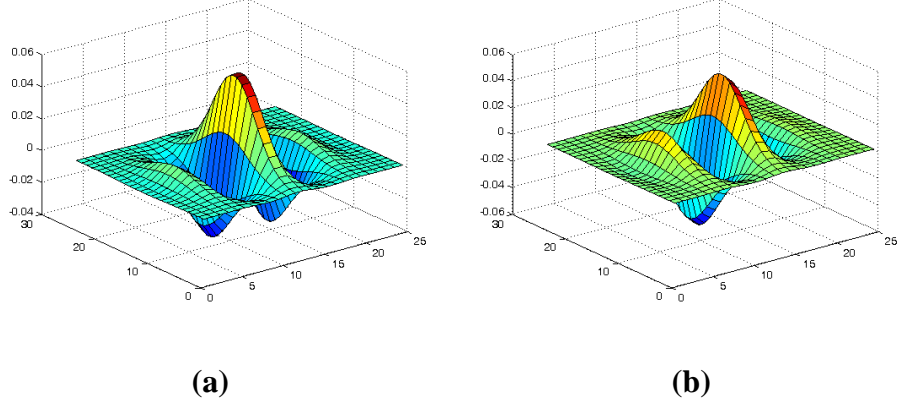
dağılımlarına karşı etkilenmeyecek bir yapıda olduğunu göstermektedir. 2.1 fonksiyonunda $z=(x, y)$, yatay eksen x ve dikey eksen y üzerinde bulunan noktayı belirtir. μ ve ν parametreleri sırasıyla, Gabor kernel' inin yönelim ve ölçeğini belirtirken, Gauss penceresinin standart sapmasını belirten σ parametresi, Gauss pencere genişliğinin dalga boyuna oranı şeklinde tanımlanır. Dalga vektörü $k_{\mu,\nu}$ ise;

$$k_{\mu,\nu} = k_{\nu} e^{i\phi_{\mu}} \quad 2.2$$

olarak tanımlanır. 2.2 fonksiyonunda

$$k_{\nu} = \frac{k_{max}}{f^{\nu}} \quad \text{ve} \quad \phi_{\mu} = \frac{\pi\mu}{n} \quad 2.3$$

biçiminde tanımlanır. k_{max} maksimum frekansı, f^{ν} kernel' ler arasındaki uzaysal frekansı ve n de yönelim sayısını ifade eder. Bu çalışmada $\nu \in \{0,1,2\}$ ve $\mu \in \{0,1,2,3\}$ olmak üzere 3 farklı frekans ve 4 farklı yönelimde toplam 12 farklı Gabor dalgacığı yüz görüntülerinden öznetelik çıkartmak amacıyla kullanılmıştır. Gabor dalgacıklarının diğer parametreleri ise $\sigma = \pi$, $k_{max} = \frac{\pi}{2}$ ve $f = \sqrt{2}$ olarak seçilmiştir. Şekil 2.1' de uzay bölgesinde karmaşık Gabor fonksiyonunun sanal ve gerçek kısımları gösterilmektedir.



Şekil 2.1 Uzay bölgesinde karmaşık Gabor fonksiyonu. $(\nu = 2, \mu = 0, f = \sqrt{2}, k_{max} = \frac{\pi}{2}, \sigma = \pi)$ (a) Fonksiyonunun gerçek kısmı (çift simetrik) (b) Fonksiyonunun sanal kısmı (tek simetrik)

Gabor filtresinin bir diğer önemli parametresi de yüz görüntülerine uygulanacak olan Gabor filtre maskesinin boyutudur. Filtre maskesinin boyutu Gabor dalgacıklarını belirtecek kadar büyük olmalı, fakat işlemsel zorluklar yaşanmaması adına çok fazla da büyük olmamalıdır. Dunn vd.⁽⁴⁷⁾, ne göre Gabor dalgacığının boyutu, Gauss fonksiyonunun yayılma alanının 6 katı olarak belirlenmiştir. Gauss fonksiyonunun yayılma alanı $\frac{\sigma}{\|k_v\|}$ ve $k_v = \frac{k_{max}}{f^\nu}$ olmak üzere

Gabor filtre maskesinin boyutu W ;

$$W = \frac{6\sigma}{\|k_v\|} + 1 = 6f^\nu \frac{\sigma}{k_{max}} + 1 \quad 2.4$$

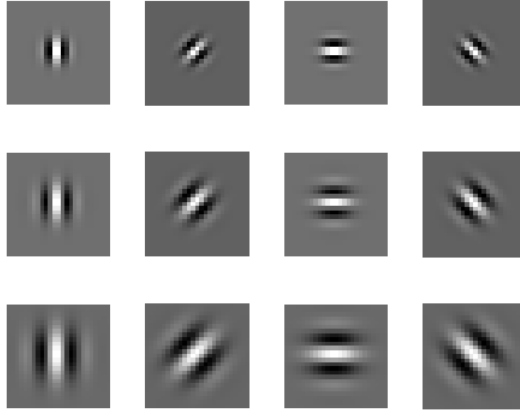
olarak belirlenir. Bu çalışmada $k_{max} = \frac{\pi}{2}$, $f = \sqrt{2}$ ve Gauss fonksiyonunun standart sapması $\sigma = \pi$ olarak belirlendiğinden Gabor filtre maskesinin boyutu;

$$W = 6\sqrt{2}^\nu \frac{\pi}{\pi/2} + 1 = 12\sqrt{2}^\nu + 1 \quad 2.5$$

olacaktır. Bu durumda, $\nu \in \{0,1,2\}$ olmak üzere 3 farklı frekans için Gabor filtre maske boyutları 13x13, 18x18 ve 25x25 olup, bu çalışmada 25x25 maske boyutu kullanılmıştır.

2.1.1.1. Yüz Görüntülerinin Gabor Dalgacık Gösterimi

Son zamanlarda yapılan nörofizyolojik çalışmalar görüntü merkezi hücrelerindeki alıcı alanların farklı uzaysal yapıda olduklarını göstermiştir⁽⁴⁸⁾. Daugman⁽⁴⁹⁾ bu hücre topluluklarının en iyi, farklı yönelim ve frekanslardaki Gabor filtrelerinden oluşan bir filtre bankasıyla modellenebileceğini göstermiştir. Bu çalışmada $\nu \in \{0,1,2\}$ ve $\mu \in \{0,1,2,3\}$ olmak üzere, 3 farklı frekans ve 4 farklı yönelimde toplam 12 farklı Gabor filtresinden oluşan bir filtre bankası oluşturularak, yüz görüntülerinden farklı özelliklerde özniteliklerin çıkartılması amaçlanmıştır. Oluşturulan Gabor filtre bankasının reel kısımları Şekil 2.2’ de gösterilmiştir.



Şekil 2.2 Gabor filtre bankasının reel kısımları (3 farklı frekans ve 4 farklı yönelim için)

Bir görüntünün Gabor dalgacık gösterimi, görüntünün Gabor filtre bankasında bulunan her bir filtre ile ayrı ayrı konvolüsyonuyla elde edilir. I görüntüsü ile $\psi_{\mu,\nu}(z)$ gibi herhangi bir Gabor dalgacığının konvolüsyonu;

$$O_{\mu,\nu}(z) = I(z) * \psi_{\mu,\nu}(z) \quad 2.6$$

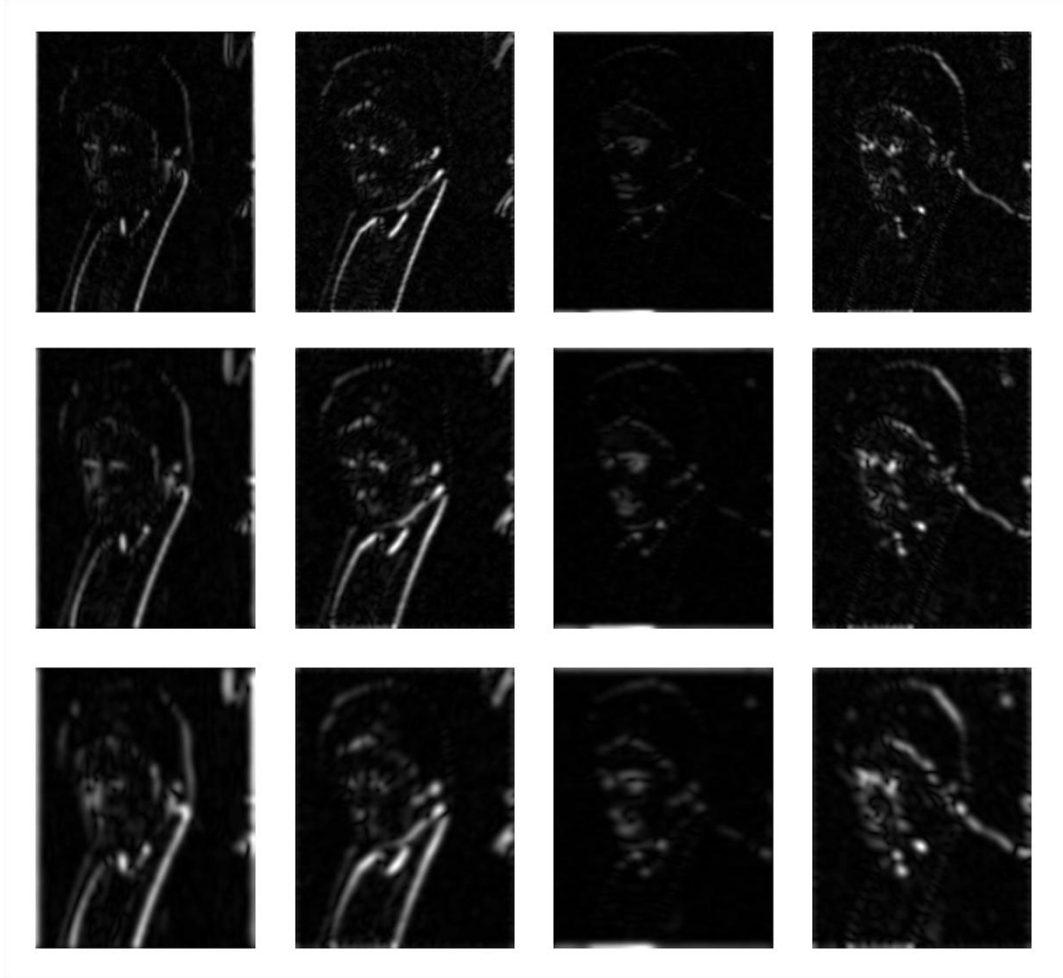
biçiminde tanımlanır. $I(z)$ görüntüsünün tüm Gabor dalgacıkları (3 frekans ve 4 yönelim olmak üzere 12 farklı Gabor dalgacığı) için gösterimi;

$$S = \{O_{\mu,\nu}(z) : \mu \in \{0, \dots, 2\}, \nu \in \{0, \dots, 3\}, z = (x, y)\} \quad 2.7$$

olarak ifade edilebilir⁽⁵⁰⁾. Şekil 2.3' te bir görüntü ve bu görüntüye karmaşık Gabor dalgacıklarının uygulanması sonucu elde edilen Gabor görüntülerinin büyüklük cevapları gösterilmektedir.



(a)



(b)

Şekil 2.3 Gabor filtrelerinin herhangi bir görüntüye uygulanması durumunda elde edilen Gabor görüntüleri (a) Orijinal görüntü (b) Gabor görüntülerinin büyüklük cevapları

2.1.2. Öznitelik Çıkartılması

Filtrelenen görüntülerden özniteliklerin çıkartılması için literatürde birçok yöntem kullanılmaktadır. Clausi ve Jernigan⁽⁵¹⁾, a göre filtre çıkışlarından özniteliklerin elde edilmesinde kullanılan bu yöntemlerden bazıları;

- Büyüklük cevabını kullanma
- Uzaysal düzeltme uygulanması
- Sadece reel bileşenlerin kullanılması
- Lineer olmayan sigmoidal fonksiyonun kullanılması
- Piksel komşuluğu bilgisini kullanma
- Tam dalga doğrultma uygulanması

olarak sıralanmıştır. Yapılan bu tezde öznitelik vektörü filtrelenmiş görüntülerin büyüklük cevaplarından elde edilmiştir. $z = (x, y)$ olmak üzere, herhangi bir giriş görüntüsü $I(z)$ ile herhangi bir kompleks Gabor fonksiyonu $\psi_{\mu, \nu}(z)$ ' nin konvolüsyonundan elde edilen sonuç olan $O_{\mu, \nu}(z)$, reel ve sanal olmak üzere iki kısımdan oluşur. O halde öznitelik vektörünün elde edilmesi için bu sonucun büyüklüğü $\|O_{\mu, \nu}(z)\|$;

$$\|O_{\mu, \nu}(z)\| = \sqrt{\Re^2 \{O_{\mu, \nu}(z)\} + \Im^2 \{O_{\mu, \nu}(z)\}} \quad \mathbf{2.8}$$

olarak elde edilir.

2.2. Öznitelik Boyutunun İndirgenmesi

Yüz görüntülerine Gabor filtrelerinin uygulanması sonucu çok sayıda öznitelik elde edilmektedir. Bu durum veri analizi, makine öğrenmesi, örüntü tanıma, metin sınıflandırma, web madenciliği ve biyolojik gen analizi uygulamalarında karşılaşılan en büyük problemlerden biridir. Bu nedenle, yapılan analize ilişkin etkili ve doğru sonuçlar alınması amacıyla veri boyutunun indirgenmesi çoğu zaman yararlı olmaktadır.

Birçok makine öğrenme ve örüntü tanıma uygulamalarında kullanılan algoritmalar literatürde “boyutun laneti”⁽⁵²⁾ olarak bilinen probleme karşı hassas olduklarından, öznitelik sayısının artması kullanılan öğrenme algoritmasının performansının azalmasına neden olmaktadır. Zira Gabor filtreleme sonucu elde edilen özniteliklerin tamamı yüz ifade analizi için gerekli olmayıp ifade sınıflandırması için önemi ve yararı olmayan bilgiler içermektedirler. Bu nedenle, yapılan çalışmalarda çoğu zaman büyük boyutlu veri kümelerinden çeşitli boyut indirgeme teknikleri kullanılarak daha küçük boyutta azaltılmış yeni veri kümeleri oluşturulmuştur. Sınıflandırma ve kümeleme üzerine yapılan çalışmalarda boyutu indirgenmiş verilerin kullanılması ile daha doğru ve anlaşılabilir sonuçlar elde edilmiştir⁽⁵³⁾. Boyut indirgeme yapılmasının nedenleri ve avantajları;

- Boyutu indirgenmiş verideki öznitelikler daha fazla ayırt edici olabilmektedirler.
- Birçok öğrenme algoritması için eğitim ve sınıflandırma zamanı boyutun indirgenmesiyle önemli ölçüde azalır.
- Önemsiz ve yararı olmayan öznitelikler sınıflandırma performansını olumsuz yönde etkilemektedirler.

- Kullanılan algoritmalar daha hızlı çalışır ve sistem belleğinin gereksiz yere kullanılması önlenmiş olur.

biçiminde sıralanabilir⁽⁵⁴⁾. Yapılan çalışmalar boyut indirgemenin genel olarak Şekil 2.4’ te gösterildiği üzere 2 boyutlu bir yapısının bulunduğunu göstermiştir.

		Denetimli	Denetimsiz
Öznitelik Dönüşümü		DAA	TBA
	Öznitelik Seçimi	Öznitelik Kümesi Seçimi (Filtre) (Sarmalama)	Laplas Skoru

Şekil 2.4 Denetimli ve denetimsiz öğrenme algoritmalarına göre boyut indirgeme teknikleri

Birinci boyut, elde bulunan özniteliklerden belli bir küme “seçilmesini” veya azaltılmış öznitelik kümesine “dönüştürülmesini” tanımlarken, ikinci boyut öğrenme algoritmasının “denetimli” veya “denetimsiz” olması ilkesine dayanmaktadır. Denetimli boyut indirgeme teknikleri farklı sınıf etiketlerine göre ayrılmış verinin boyutunu azaltır. Kullanılan algoritmanın niceliği, bir sınıfı diğer sınıflardan en çok ayıran özniteliklerin bulunmasında kullanılır. Denetimsiz boyut indirgeme teknikleri ise herhangi bir sınıf etiketine ihtiyaç duymadan, öznitelikler arasındaki benzerliklere göre hangi özniteliklerin veri kümesinden çıkarılacağına karar verirler. Benzer öznitelikler bir arada gruplandırılırken diğer öznitelikler atılır. Örneğin Temel

Bileşenler Analizi bir denetimsiz öznitelik dönüşüm tekniği iken, Fisher Ölçüt Skoru bir denetimli öznitelik seçme tekniğidir⁽⁵⁴⁾.

2.2.1. Öznitelik Dönüşümü

Öznitelik dönüşümü, orijinal öznitelik kümesinin çeşitli teknikler kullanılarak daha küçük boyutta ve gerekli bilgiyi olabildiğince koruyan alternatif bir kümenin oluşturulması şeklinde gerçekleştirilir. Bu tekniklerden en önemlisi orijinal öznitelik kümesine uygulanacak bazı dönüştürme uygulamalarıyla azaltılmış veri kümesi oluşturulmasıdır. Literatürde denetimli öznitelik dönüşüm tekniği Doğrusal Ayırma Analizi – DAA (Linear Discriminant Analysis) ile denetimsiz öznitelik dönüşüm tekniği Temel Bileşenler Analizi kullanılmakta olup, bu çalışmada öznitelik dönüşüm tekniği olarak Temel Bileşenler Analizi kullanılmıştır.

2.2.1.1. Temel Bileşenler Analizi

İstatistiksel bir yöntem olan Temel Bileşenler Analizi'nin (TBA) amacı boyut indirgemesi yapmaktır. TBA, orijinal t vektörün varyans yapısını daha az sayıda ve bu t vektörün doğrusal bileşenleri olan yeni vektörlerle ifade etme yöntemidir. Aralarında korelasyon (ilişki) bulunan t tane vektörün açıkladığı yapıyı, aralarında korelasyon bulunmayan ve sayıca orijinal vektör sayısından daha az sayıda ($k \leq t$) orijinal değişkenlerin doğrusal bileşenleri olan vektörlerle ifade etme yöntemi Temel Bileşenler Analizi olarak tanımlanır.

N tane görüntü, $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t]$ olmak üzere t boyutlu Gabor öznitelik vektörü ile gösterilmiş olsun. Temel Bileşenler Analizi^(55,56) $k \leq t$ olmak üzere orijinal t

boyutlu öznitelik uzayını k boyutlu öznitelik alt uzayına dönüştüren lineer bir dönüşüm bulmak için kullanılmaktadır. Yeni öznitelik vektörü y_i ;

$$\mathbf{y}_i = W^T \mathbf{x}_i, \quad (i = 1, 2, \dots, t) \quad \mathbf{2.9}$$

biçiminde tanımlanır. Burada W aradaki lineer dönüşümü göstermekte olup, bu lineer dönüşüm $[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_t]^T$ in istatistiksel olarak elde edilen kovaryans matrisinin özdeğer ve özvektörlerinin bulunması ile elde edilir. Buna göre kovaryans matrisinin özdeğer ve özvektör çiftleri $(\lambda_1, e_1), (\lambda_2, e_2), \dots, (\lambda_t, e_t)$ olmak üzere i . temel bileşen;

$$\mathbf{y}_i = W^T \mathbf{x}_i = e_{i1} \mathbf{x}_1 + e_{i2} \mathbf{x}_2 + \dots + e_{it} \mathbf{x}_t \quad \mathbf{2.10}$$

biçiminde elde edilir. i . temel bileşenin varyansı ise;

$$\text{Var}(\mathbf{y}_i) = \lambda_i \quad \mathbf{2.11}$$

olur. Burada $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_t \geq 0$ dır. Temel bileşenler birbirleriyle ilişkisizdir ve varyansları kovaryans matrisinin özdeğerlerine eşittir.

t tane temel bileşen oluşturulduktan sonra, verilerin kaç tane temel bileşen ile ifade edilmesi gerektiği en önemli konudur. Bununla ilgili olarak literatürde çeşitli kurallar mevcuttur. Ancak bunlardan en yaygın kullanılanı “Genel varyansın en az %67’sini açıklayan sayıda temel bileşen” seçilmesidir. Buna göre, ilk k tane özdeğerin toplamı, tüm özdeğerlerin toplamına oranlandığında elde edilen %67 oranı, orijinal veri setinin temsil edilebilmesi için ilk k tane temel bileşenin kullanılmasının yeterli olduğunu ifade etmektedir⁽⁵⁷⁾.

2.2.2. Öznitelik Seçimi

Boyut indirgemenin, öznitelik verisinin yeni bir öznitelik kümesine dönüşümü şeklinde gerçekleştirildiği öznitelik dönüşümüne nazaran öznitelik seçme algoritmaları, orijinal öznitelik kümesinden en iyi öznitelik alt kümesini seçmeyi amaçlarlar. Öznitelik dönüşümü ile orijinal öznitelik verisinden farklı yeni bir öznitelik kümesi oluşturulmakta ve bu işlem bazı veri kayıplarına neden olabilmektedir. Tam tersine, öznitelik seçimi ile orijinal öznitelik kümesinden seçilen öznitelikler kullanılarak yeni azaltılmış veri kümesi oluşturulmaktadır. Dolayısıyla öznitelik dönüşümüne göre daha anlaşılır ve yorumlanabilir veri kümesi elde edilmiş olur.⁽⁵⁴⁾

Öznitelik seçilmesi ile esasen ilgisiz ve alakasız öznitelikler ile fazla ve ihtiyaç duyulmayan, yani gereksiz özniteliklerin atılması gerçekleştirilmektedir. Örneğin denetimli öznitelik seçme algoritmalarından Entropi (Bilgi Kazancı) tekniğine göre öznitelik seçimi şu şekilde gerçekleştirilmektedir: E entropi'yi göstermek üzere F_1 ve F_2 gibi iki öznitelik olsun. F_i özniteliğinin bilgi kazancı $E_0 - E_i$ şeklinde hesaplanacaktır. E_0 , F_i özniteliğinin sınıf bazında bölünmeden önce elde edilen entropi değeridir ve

$$E_0 = \sum_c p_c \log p_c \quad \mathbf{2.12}$$

biçiminde hesaplanır. Burada p , c sınıfının ($c = 1, 2, \dots, C$) hesaplanan olasılığıdır. E_i ise F_i özniteliğinin sınıf bazında bölündükten sonra elde edilen entropi değeridir. Sonuç olarak, entropi tekniğine göre, büyük bilgi kazancının elde edildiği öznitelik

seçilecektir. Denetimsiz öznitelik seçiminde ise (örneğin kümeleme) öznitelikler arasındaki uzaklığa göre benzer öznitelikler gruplandırılır ve diğerleri atılır^(54,58).

Öznitelik seçilmesinde genel olarak “Filtre Modeli” ve “Sarmalama Modeli” olmak üzere 2 farklı model kullanılmaktadır. Filtre modeline göre öznitelik seçme işlemi, herhangi bir öğrenme algoritması hesaba katılmaksızın önemsiz ve gereksiz özniteliklerin atılması şeklinde gerçekleştirilir. Bu işlemde genellikle, bir değerlendirme fonksiyonunu maksimum yapan bir küme seçmek için bir arama metodu kullanılır. Buna göre, bir özneliğin incelenen problemdeki önemi belirlenir ve önemsiz olan öznitelikler atılarak öznitelik sayısı azaltılmış olur. Sarmalama modeline göre ise algoritma, seçilecek gerekli öznitelikleri belirlemek için bir öğrenme algoritması kullanır. Bunun için sarmalama algoritması, seçilen her öznitelik alt kümesi için ilgili sınıflandırıcının başarımlarını ölçer ve en iyi başarımları veren öznitelik kümesi seçilir^(58,59).

Çoğu örüntü tanıma uygulamasında sarmalama modeli algoritmaları filtre modeli algoritmalarına göre üstün performans göstermiş olmasına karşın hesaplama bakımından çok daha zahmetlidir⁽⁵⁹⁾. Yapılan bu çalışmada öznitelik veri kümesinin çok büyük boyutta olması nedeniyle işlem yükü fazla olmayan ve hızlı sonuçlar üreten öznitelik seçme algoritmaları tercih edilmiştir. Öznitelik seçiminde;

1. Fisher Ölçüt Skoru
2. Öznitelik Seçimi İçin Sıfırcı-Norm Kullanılması
3. Ortak Bilgi Temelli Öznitelik Seçimi
4. Yinelemeli Öznitelik Eliminasyonu

algoritmaları kullanılmıştır. Bu algoritmalara ilişkin bilgiler sonraki bölümlerde sıralı bir şekilde verilmiştir.

2.2.2.1. Fisher Ölçüt Skoru

Fisher ölçüt skoru (Fisher Criterion Score)⁽⁶⁰⁾, öznitelik seçimi için kullanılan en basit filtre algoritmalarından biridir. İki sınıf durumunda, her bir t özniteliği için;

$$F_t = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \quad 2.13$$

değeri hesaplanır. Burada μ_1 , Sınıf1' e ait t değerlerinin ortalaması; μ_2 , Sınıf2' ye ait t değerlerinin ortalaması ve σ_1 ile σ_2 ' de sırasıyla Sınıf1 ve Sınıf2' nin standart sapmalarıdır. Fisher ölçüt skoruna göre bir öznitelik, Sınıf1 ve Sınıf2 dağılımlarını ne kadar çok ayırabiliyorsa o kadar iyidir.

İki sınıftan daha fazla sınıf varsa Fisher ölçüt skoru;

$$F_t = \frac{(\mu_1 - \sum_{i=2}^M \mu_i)^2 + \dots + (\mu_j - \sum_{\substack{i=1 \\ i \neq j}}^M \mu_i)^2 + \dots + (\mu_M - \sum_{i=1}^{M-1} \mu_i)^2}{\sum_{i=1}^M \sigma_i^2} \quad 2.14$$

biçiminde kolayca genelleştirilebilir. Burada M sınıf sayısıdır. Fisher ölçüt skoru aracılığıyla tüm öznitelikler için bir sıralama hesaplanır. Metodu karakterize eden, öznitelikler arasındaki tam bağımsızlık varsayımıdır. Her F_t katsayısı tek bir öznitelik hakkındaki bilgiyle hesaplanır ve öznitelikler arasındaki ortak bilgi hesaba katılmaz⁽⁶¹⁾.

2.2.2.2. Öznitelik Seçimi İçin Sıfıncı-Norm Kullanılması (L0)

Weston vd.⁽⁶²⁾ standart destek vektör makinelerinde ℓ_1 -norm veya ℓ_2 -norm' un minimize edilmesi yerine $\|w\|_0 = |\{w_j : w_j \neq 0\}|$ biçiminde tanımlanan

sıfıncı normun minimize edilmesini önermişlerdir. Dolayısıyla destek vektör makinesini tanımlayan optimizasyon problemi, sıfıncı norm için;

$$\begin{aligned} \min_w \|w\|_0 \\ y_i(w \cdot x_i + b) \geq 1 \quad (i = 1, 2, \dots, l) \end{aligned} \quad 2.15$$

olarak ifade edilir. Bu optimizasyon probleminin çözümü standart destek vektör makineleri için olan problemin çözümünden daha kısadır. Buradan, seçilecek öznitelik sayısı belirlenemiyor olmasına rağmen öznitelik seçimi dolaylı olarak gerçekleştirilebilir. r adet öznitelik;

$$\begin{aligned} \min_{w \in \mathbb{R}^n} \|w\|_2 \\ y_i(w \cdot x_i + b) \geq 1 \quad (i = 1, 2, \dots, l) \text{ ve } \|w\|_0 \leq r \end{aligned} \quad 2.16$$

optimizasyon probleminin çözülmesiyle seçilebilir. Bu denklem, $\|w\|_0 \leq r$ kısıtından başka herhangi bir kısıt olmaksızın standart destek vektör makinesi optimizasyon problemine eşittir. $\|w\|_0$ kısıtı, çözüm vektörü w^* in sadece r adet ve sıfır olmayan öğeden oluştuğunu garanti eder. $w_j^* \neq 0$ durumunu sağlayan j indisleri seçilen öznitelikleri gösterir.

2.2.2.3. Ortak Bilgi Temelli Öznitelik Seçimi (MutInf)

Ortak bilgi (I), ayrık rastgele değişkenlerin stokastik (olasılıksal) bağımlılık bilgisinin ölçümünde yaygın olarak kullanılan bir teknik olup aynı zamanda sınıflandırma problemlerinde sınıf değişkenlerinin öznitelik kümesinin seçiminde kullanılır⁽⁶³⁾. X ve Y gibi iki ayrık rastgele değişkenin ortak bilgisi;

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \quad 2.17$$

olarak tanımlanır. 2.17 denkleminde $p(x,y)$, X ve Y 'nin birleşik olasılık dağılım fonksiyonu, $p(x)$ ve $p(y)$ 'de sırasıyla X ve Y 'nin marjinal olasılık dağılım fonksiyonlarıdır. Yukarıda verilen toplam, sürekli zamanda iki katlı belirli integral ile;

$$I(X;Y) = \int_Y \int_X p(x,y) \log \frac{p(x,y)}{p(x)p(y)} dx dy \quad 2.18$$

biçiminde tanımlanır. Burada, $p(x,y)$, X ve Y 'nin birleşik olasılık yoğunluk fonksiyonu, $p(x)$ ve $p(y)$ 'de sırasıyla X ve Y 'nin marjinal olasılık yoğunluk fonksiyonlarıdır.

\log fonksiyonunun tabanı belirtilmediğinden verilen bu tanımlamalar belirsizdir. Bu belirsizliği ortadan kaldırmak için $I(\)$ fonksiyonu, b logaritma tabanı olmak üzere $I(X,Y,b)$ şeklinde yeniden parametrelendirilebilir. Fakat ortak bilgi ölçümlerinin çoğunun ortak birimi "bit" olduğundan, logaritma tabanı 2 olarak belirtilebilir. Bu durumda tanımlamalar;

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)} \quad 2.19$$

ve

$$I(X;Y) = \int_Y \int_X p(x,y) \log_2 \frac{p(x,y)}{p(x)p(y)} dx dy \quad 2.20$$

haline gelir. Ortak bilgi, X ve Y ' nin paylaştıkları bilgiyi ölçer. Yani bu değişkenlerden herhangi birinin bilinmesinin, bir diğeri hakkındaki belirsizliği ne kadar düşürdüğü ölçülür. Örneğin, eğer X ve Y bağımsız iseler, X ' in bilinmesi Y hakkında hiçbir bilgi vermez (veya tersi). Buradan bu iki değişken arasındaki ortak bilgi sıfırdır. Öte yandan, eğer X ve Y özdeş iseler X tarafından taşınan tüm bilgiler Y ile paylaşılır, bilinen X , Y değerini belirler (veya tersi).

Ortak bilgi, X ve Y ' nin birleşik dağılımları ile X ve Y bağımsız oldukları durumdaki birleşik dağılımın arasındaki farkın miktarını belirtmektedir ve;

$$I(X;Y)=0 \Leftrightarrow X \text{ ve } Y \text{ bağımlı değilse (} X \text{ ve } Y \text{ bağımsız rastgele değişkenler)}$$

şeklindeki bağımlılığın bir ölçümüdür. Yani X ve Y bağımsız iseler $p(x, y) = p(x) \times p(y)$ olacaktır ve buradan;

$$\log \frac{p(x, y)}{p(x) p(y)} = \log 1 = 0 \quad \mathbf{2.21}$$

olur. Ayrıca ortak bilgi negatif değildir (yani; $I(X;Y) \geq 0$) ve simetriktir (yani; $I(X;Y) = I(Y;X)$)⁽⁶⁴⁾.

2.2.2.4. Yinelemeli Öznitelik Eliminasyonu (RFE)

Yinelemeli öznitelik eliminasyonu (Recursive Feature Elimination – RFE), Guyon vd.⁽⁶⁵⁾ tarafından son yıllarda geliştirilmiş bir öznitelik seçme algoritmasıdır. Algoritma, n toplam öznitelik sayısını göstermek üzere, $r < n$ koşulunu sağlayan en iyi r boyutlu öznitelik altkümesini bulmaya çalışır. Bu işlem, bir SVM

sınıflandırıcısı kullanılarak sınıflar arası marjini en büyük yapan r özneliğin seçilmesi ile gerçekleştirilir. r boyutlu öznelik kümesi kalıncaya kadar her iterasyonda giriş öznelik kümesi azaltılır. Algoritma özellikle mikro dizi veri analizi için gen seçilmesinde iyi performans göstermiştir^(62,66).

SVM' ler için marjin, $W^2(\alpha) = \sum \alpha_i \alpha_j y_i y_j k(x_i, x_j)$ değeri ile ters orantılıdır. Algoritma, bu değeri küçük tutan öznelikleri çıkarmakta olup, bu işlem aşağıdaki yinelemeli süreç ile yapılmaktadır:

α çözümü ve her bir öznelik p için

$$W_{(-p)}^2(\alpha) = \sum \alpha_i \alpha_j y_i y_j k(x_i^{-p}, x_j^{-p}) \quad 2.22$$

değeri hesaplanır. (Burada x_i^{-p} , p özneliği atılan i noktasının eğitimi anlamına gelir). Daha sonra $|W^2(\alpha) - W_{(-p)}^2(\alpha)|$ ifadesinin en küçük değerli olan özneliği çıkarılır. Eğer sınıflandırıcı lineer ise ($g(x) = w \cdot x + b$ biçiminde) bu algoritma, her yinelemede en küçük $|w_i|$ değerli özneliği atar. Özneliklerin sayısı büyük olduğu zaman hesaplamaları hızlandırmak için her yinelemede özneliklerin yarısının çıkarılması önerilmektedir. Çok-sınıflı öznelik seçimi “bire karşı diğerleri” gibi bir yaklaşım için kolaylıkla elde edilebilmesine rağmen, RFE iki-sınıflı problemler için tasarlanmıştır⁽⁶²⁾.

2.3. Gabor Özneliklerinin Sınıflandırılması

Yüz ifade analizinde son adım, çıkarılan özneliklerin ifadelere göre sınıflandırılmasıdır. İfade sınıflandırması, bir karar işlemi ile bağlantılı örüntü

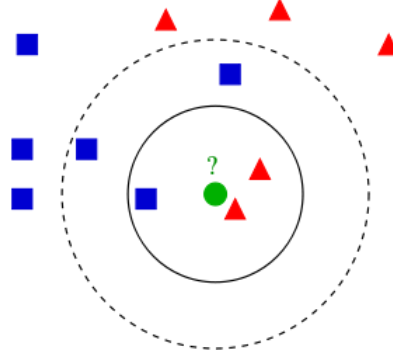
dağılımı modelleri içeren bir sınıflandırıcı ile gerçekleştirilmektedir. Bu işlem için öncelikle makine, sınıflandırması istenen öge kümesinden belli sayıda örnekle eğitilir ve diğer ögeler eğitim örneklerine göre sınıflandırılırlar⁽²⁰⁾.

Literatürde yüz ifade analizinde Destek Vektör Makineleri⁽⁶⁷⁾, Bayes Ağları⁽⁶⁸⁾, En Yakın Komşu⁽⁶⁹⁾ ve Saklı Markov Modeli⁽⁷⁰⁾ gibi birçok sınıflandırma algoritması kullanılmıştır. Bu çalışmada yüz ifadelerinin sınıflandırılmasında en basit sınıflandırma algoritmalarından k-NN (k-Nearest Neighbors – En Yakın Komşu) ile en gelişmiş sınıflandırma algoritmalarından SVM (Support Vector Machines – Destek Vektör Makineleri) ve AdaBoost (Adaptive Boosting) kullanılmıştır. Algoritmalarla ilgili bilgiler sonraki bölümlerde verilmiştir.

2.3.1. k – En Yakın Komşuluk Sınıflandırma Algoritması (k-NN)

k-NN sınıflandırıcı (k-Nearest Neighbor – k-NN), bir örnek tabanlı öğrenme algoritması olup, gözlem çiftlerinin Öklidyen uzaklık gibi bir fonksiyonuna dayalıdır. Bu sınıflandırma örneğinde, önce eğitim kümesinin k yakın komşuları hesaplanır. Sonra test verisinden, bir örneğin k en yakın komşuluklarına olan benzerlikler, komşulukların sınıflarına göre kümelenir ve test örneği en benzer sınıfa atanır. k-NN' in avantajı, sınıflandırma kararının benzer nesnelere küçük komşuluklarına dayalı olmasından ötürü, çoklu sınıflar için uygun olmasıdır. Buradan hedef sınıfı çoklu (örneğin, içerdiği nesnelere bağımsız değişkenleri farklı alt kümeler için farklı karakteristiklere sahip) ise k-NN doğruluğunu yine de devam ettirir. k-NN' de kullanılan benzer değerlerin en büyük dezavantajı, tüm öznitelikleri eşit hesaplama benzerlikleri içinde kullanmasıdır. Bu, sınıflandırma için sadece küçük bir öznitelik kümesi faydalı iken, zayıf ölçümler ve sınıflandırma hatalarına neden olur⁽⁷¹⁾.

Genel olarak en iyi k parametresinin seçimi veriye bağlı olup k 'nin büyük değerleri sınıflandırma üzerindeki gürültüyü azaltır, fakat sınıflar arasındaki sınırların belirginliğini en aza indirger. İyi bir k değerinin seçimi, çapraz geçerlilik (cross-validation) gibi bir teknikle gerçekleştirilebilir.



Şekil 2.5 k-NN sınıflandırma örneği

Şekil 2.5' te gösterilen k-NN sınıflandırma örneğine göre test örneği (yeşil daire), Sınıf1 (mavi kareler) veya Sınıf2 (kırmızı üçgenler)' den birine sınıflandırılmalıdır. Eğer $k = 3$ ise yeşil daire Sınıf2' ye sınıflandırılır. Çünkü içteki dairenin içinde 2 tane üçgen ve sadece 1 tane kare bulunmaktadır. Eğer $k = 5$ ise yeşil daire Sınıf1' e sınıflandırılır. Çünkü dıştaki dairenin içinde 3 tane kare, 2 tane üçgen var.

Bir sınıfın en yakın eğitim örneğinin sınıfı olarak tahmin edildiği özel durum ($k = 1$ olduğu durum) en yakın komşuluk algoritması olarak bilinir. k-NN algoritmasının doğruluğu, gürültülü ve ilgisiz özniteliklerin bulunmasından veya özniteliklerin önemlerine uygun bir şekilde ölçeklendirilmemesinden dolayı ciddi miktarda düşebilir⁽⁷²⁾.

2.3.2. Destek Vektör Makineleri (SVMs)

Destek Vektör Makineleri (Support Vector Machines – SVMs) veri sınıflandırma için çok kullanışlı bir tekniktir. Bu tekniğe ait sınıflandırma algoritması ilk olarak 1992 yılında Vapnik ve çalışma arkadaşları tarafından ikili sınıflandırma çerçevesi içinde ileri sürülmüştür^(73,74). Son yıllarda bu metot, kernel metotları^(75,76) olarak bilinen tüm istatistiksel öğrenme algoritma ailesinin doğmasına yol açan makine öğrenme çerçevesinde büyük bir ilgi kazanmıştır. SVM, optik karakter tanıma⁽⁷⁷⁾, metin madenciliği⁽⁷⁸⁾, yüz tanıma^(79,80), yüz ifade analizi⁽³⁶⁾ ve biyotıp⁽⁸¹⁾ gibi birçok uygulamada başarılı olmuş ve çoğu yaklaşıma üstünlük sağlamıştır.

2.3.2.1. İkili Sınıflandırma İçin SVM

En basit SVM biçimi, örnekler ve bu örneklere ait etiketlerden oluşan kümenin bir ikili sınıflandırma kuralı öğrenme algoritmasıdır. Biçimsel olarak ifade edilirse, her bir örneğe ilişkin ikili olarak etiketlenmiş bir küme olsun. Örneğin bu küme $S = \{(x_1, y_1), \dots, (x_l, y_l)\}$ olmak üzere, $(x_i, y_i) \in X \times \{-1, +1\}$ ve $i = 1, 2, \dots, l$ biçimindedir.

Burada X , \mathbb{R}^d biçiminde olup, analiz edilecek veri uzayını belirtir. Örneğin d boyutlu yüz görüntülerine ait ifadeler veri uzayını, etiketler $+1$ ve -1 ' de ifadelerin ikili sınıflarını belirtmektedirler. SVM' nin amacı, S ' den $f: X \rightarrow \{-1, +1\}$ şeklinde bir sınıflandırma fonksiyonu öğrenmektir. SVM sınıflandırıcı çıkışı, lineer bir fonksiyonun işaretine dayalıdır:

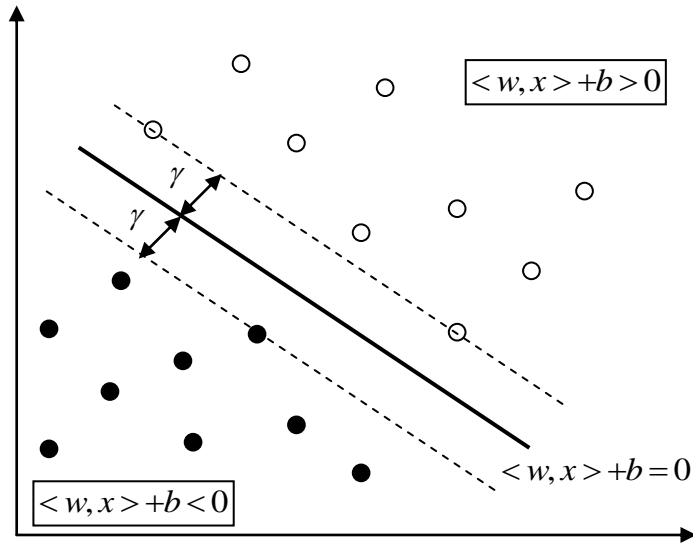
$$f(x) = \text{sign}(\langle w, x \rangle + b)$$

2.23

Geometrik olarak, bir $(w, x) \in X \times \mathbb{R}$ ikilisi $H_{w,b} = \{x \in X : \langle w, x \rangle + b = 0\}$ biçiminde bir aşırı düzlem tanımlar ki bu düzlem, giriş uzayı X 'i iki yarım uzaya böler ve böylece yeni bir noktanın sınıf tahmini, bu noktanın düzlemin hangi tarafında bulunduğuyla bağlı olacaktır.

2.3.2.1.1. Lineer SVM (Ayrılabilir Veri Durumu)

Eğitim seti lineer bir şekilde ayrılabilir olduğunda, başka bir deyişle pozitif ve negatif örneklerin aşırı düzlemin farklı taraflarında bulunduğu bu durum, SVM'nin en basit çeşidi olan Hard Marjın SVM'dir. Hard Marjın SVM, sonsuz sayıdaki aşırı düzlem arasından, S 'den en yakın veri noktasına en geniş aralıklı olanı seçer. Bu aralık, aşırı düzlemin marjini olarak bilinir ve genellikle γ ile belirtilir. Aşırı düzlem ise Şekil 2.6'da gösterildiği gibi maksimum marjın sınıflandırıcısını tanımlar⁽⁷⁴⁾.



Şekil 2.6 SVM, maksimum marjın γ olmak üzere, pozitif (beyaz daireler) ve negatif (siyah daireler) örnekleri ayıran $\langle w, x \rangle + b = 0$ aşırı düzlemini bulur.

$(x_i, y_i) \in S$ noktasından $H_{w,b} = \{x \in X : \langle w, x \rangle + b = 0\}$ aşırı düzlemine olan uzaklık $\frac{|\langle w, x_i \rangle + b|}{\|w\|}$ ile verilir. Eğer $H_{w,b}$ bir ayırma aşırı düzlemi ise, bu aralık $y_i \frac{|\langle w, x_i \rangle + b|}{\|w\|}$ 'ya eşit olur ve buna ilişkin marjin;

$$\gamma = \min_{i=1, \dots, l} y_i \frac{|\langle w, x_i \rangle + b|}{\|w\|}, \quad (x_i, y_i) \in S \quad 2.24$$

ile verilir. Aşırı düzlemler, önlerine bir ölçeklendirme sabiti alarak tanımlandıkları için $H_{w,b}$ tanımına;

$$\min_{i=1, \dots, l} y_i (\langle w, x_i \rangle + b) = 1, \quad (x_i, y_i) \in S \quad 2.25$$

kısıtı eklenebilir. Buradan $H_{w,b}$ aşırı düzleminin marjini $\frac{1}{\|w\|}$ olarak elde edilir.

SVM algoritması, maksimum marjine sahip olan H_{w^*, b^*} aşırı düzlemini arar ve sözü edilen maksimum marjin,

$$(w^*, b^*) = \underset{\substack{w \in X \\ b \in \mathbb{R}}}{\operatorname{argmin}} \frac{1}{2} \|w\|^2 \quad 2.26$$

$$y_i (\langle w, x_i \rangle + b) \geq 1, \quad i = 1, \dots, l \quad 2.27$$

biçiminde bir optimizasyon problemi olarak formülize edilebilir⁽⁸²⁾. Optimizasyon problemleri pratikte Lagrange duality çerçevesi içinde çözülebilir. 2.26 ve 2.27 ile belirtilen primal optimizasyon problemlerine bağlı olan Lagrange;

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i (y_i (\langle w, x_i \rangle + b) - 1) \quad 2.28$$

biçiminde ifade edilir. Burada $\alpha_i \geq 0$ katsayıları Lagrange çarpanları olup $y_i(\langle w, x_i \rangle + b) \geq 1$ kısıtıyla ilişkilendirilir. Bu optimizasyon probleminin çözümü Lagrange'ın eyer noktasında, yani $L(w, b, \alpha)$ 'nin (w, b) 'ye göre minimumunda ve α 'ya göre bir maksimumunda bulunur. Sabit bir α için, w ve b 'ye göre Lagrange minimize edilir. Bu işlem $L(w, b, \alpha)$ 'nin w ve b 'ye göre kısmi türevleri alınarak gerçekleştirilir ve buradan sırasıyla;

$$\frac{\partial}{\partial w} L(w, b, \alpha) = 0 \quad \Rightarrow \quad w = \sum_{i=1}^l \alpha_i y_i x_i \quad \mathbf{2.29}$$

$$\frac{\partial}{\partial b} L(w, b, \alpha) = 0 \quad \Rightarrow \quad \sum_{i=1}^l \alpha_i y_i = 0 \quad \mathbf{2.30}$$

elde edilir. Bu ifadeler (2.28) ile verilen Lagrange denkleminde yerine konulursa orijinal problem şu şekildeki dual probleme dönüşür:

$$\alpha^* = \operatorname{argmax}_{\alpha \in \mathbb{R}^l} \quad \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \quad \mathbf{2.31}$$

$$\alpha_i \geq 0, \quad i = 1, \dots, l \quad \mathbf{2.32}$$

$$\sum_{i=1}^l \alpha_i y_i = 0 \quad \mathbf{2.33}$$

Bu problem Sequential Minimal Optimization (SMO)⁽⁸³⁾ algoritması ile çözülür ve optimum α^* ile karşılaştığında karar fonksiyonu $f(x) = \operatorname{sign}(\langle w^*, x \rangle + b^*)$ şu şekilde olur:

$$f(x) = \operatorname{sign} \left(\sum_{i=1}^l \alpha_i^* y_i \langle x, x_i \rangle + b^* \right) \quad \mathbf{2.34}$$

b^* kayması da optimumda elde edilir ve noktalar Karush-Kuhn-Tucker durumunu doğrular:

$$\alpha_i^* \left(y_i (\langle w^*, x_i \rangle + b^*) - 1 \right) = 0, \quad i = 1, \dots, l \quad 2.35$$

Bu duruma göre sıfır olmayan α_i^* Lagrange çarpanları sadece $y_i (\langle w^*, x_i \rangle + b^*) = 1$ olduğu durumdaki (x_i, y_i) noktalarına karşılık gelir. Bu noktalar aşırı düzlemin marjini üzerinde bulunurlar ve sınıflandırıcının “destek vektörleri” olarak adlandırılırlar.

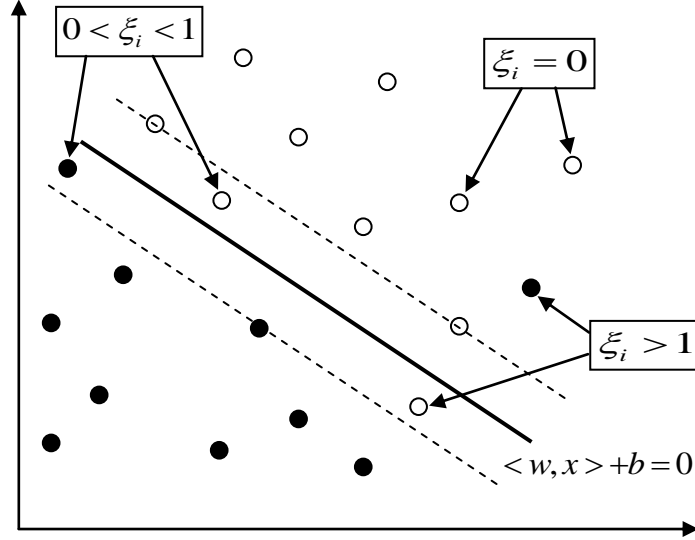
2.3.2.1.2. Lineer SVM (Ayrılamayan Veri Durumu)

Ayrılabilir veri için olan durumun aksine, genel olarak veriyi ayırabilen lineer bir aşırı düzlem bulunmayabilir. Ayrılamayan veri kümeleri, pratikte her $(x_i, y_i) \in S$ noktası için primal problemin, 2.27 ile verilen ayrılabilirlik kısıtlarını sağlamak amacıyla, $x_i^3 \geq 0$ geçici değişkenlerin kullanılması şeklinde düşünülebilir:

$$y_i (\langle w, x_i \rangle) + b^3 \geq 1 - x_i \quad 2.36$$

x_i terimi, (x_i, y_i) noktasının sınıflandırılmasında yapılan hatayı açıklamaktadır. Şekil 2.7’ den görüleceği üzere, eğer $x_i = 0$ ise, (x_i, y_i) noktası doğru bir şekilde sınıflandırılır ve marjinin dışında bulunur; $0 < x_i < 1$ ise, (x_i, y_i) noktası yine doğru sınıflandırılır, fakat marjinin içinde bulunur ve eğer $x_i > 1$ ise, (x_i, y_i) noktası

sınıflandırılmaz. Aşırı düzlemin marjininin $x_i = 0$ için $(x_i, y_i) \in S$ noktalarında tanımlandığına dikkat edilmelidir.



Şekil 2.7 Geçici değişkenlerin gösterimi

Geçici değişkenlerin toplamı, ayrılabilirlik kısıtlarının miktarının eğitim seti üzerindeki ihlalini açıklar, dolayısıyla bu toplam öğrenme işlemi süresince kontrol edilmelidir. Bu işlem, geçici değişken niceliğinin SVM optimizasyon problemi objektif fonksiyonuna eklenmesiyle yapılır:

$$(w^*, b^*, x^*) = \underset{\substack{w \in X, \\ b \in \mathbb{R}, \\ x \in \mathbb{R}^l}}{\operatorname{argmin}} \quad \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l x_i \quad 2.37$$

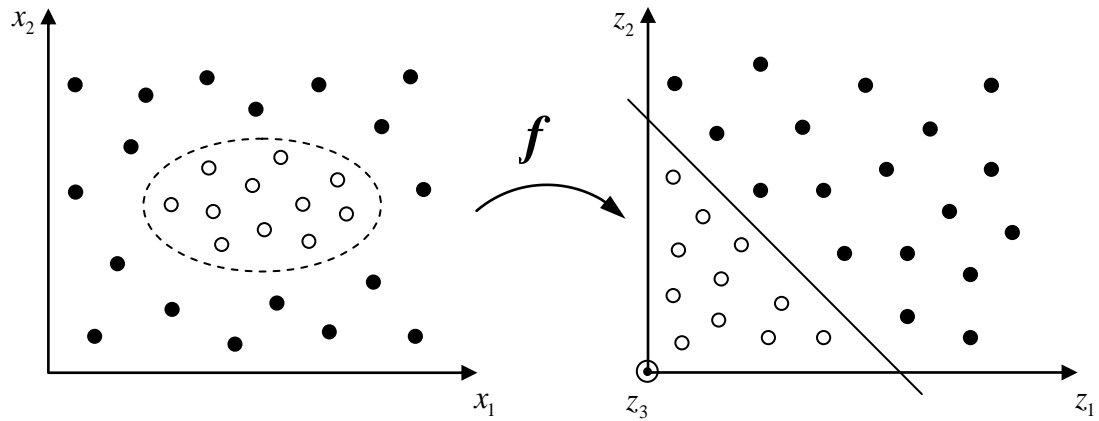
$$y_i (\langle w, x_i \rangle) + b \leq 1 - x_i, \quad i = 1, \dots, l \quad 2.38$$

$$x_i \geq 0, \quad i = 1, \dots, l \quad 2.39$$

2.37 ile belirtilen amaç fonksiyondaki C sabiti, $\frac{1}{2}\|w\|^2$ terimi ile ifade edilen marjin maksimizasyonu arasındaki ilişkiyi gösterme anlamındadır. C 'nin azaltılması, artan marjinin yararı için eğitim kümesi üzerindeki büyük sayıdaki sınıflandırılmayın tolere edilmesini sağlar. “Soft Marjin SVM” olarak bilinen bu formülasyon, ayrılabilir durumdaki Hard Marjin formülasyonunun tam tersidir. Hard Marjin formülasyonu, eğitim seti lineer olarak ayrılabilir olmasına rağmen, $C = +\infty$ için düzelme gösterirken, Soft Marjin formülasyonu ise gürültü ve olabilecek aykırı değerlere karşı uyumu önler.

2.3.2.1.3. Lineer Olmayan SVM

Şekil 2.8' in sol tarafında görüldüğü gibi lineer olarak ayıramayan problemlerde, lineer sınıflandırıcılar kümesi düzgün bir sınıflandırma fonksiyonu sağlamak için yeterli olmayabilir.



Şekil 2.8 Lineer olmayan SVM gösterimi. $\phi = \mathbb{R}^2 \rightarrow \mathbb{R}^3$ şeklinde şeklindeki eşleştirme $\phi([x_1, x_2]) = [z_1, z_2, z_3]$ ile tanımlanır. ($z_1 = x_1^2, z_2 = x_2^2$ ve $z_3 = \sqrt{2}x_1x_2$)

Lineer olmayan durumda destek vektör makineleri, orijinal veri yerine dönüştürülmüş veri $\phi(x_1), \dots, \phi(x_l)$ ' ye lineer yaklaşım kullanılarak genelleştirilebilir. Burada ϕ , giriş uzayı X ' ten yeni öznitelik uzayı H ' ye lineer olmayan dönüşümü gösterir. 2.31 denklemindeki $\langle x_i, x_j \rangle$ yerine $\langle \phi(x_i), \phi(x_j) \rangle_H$ konulmasıyla yeni öznitelik uzayında lineer bir aşırı düzlem hesaplanır ve buradan giriş uzayındaki lineer olmayan ayırma fonksiyonu f ;

$$f(x) = \text{sign} \left(\sum_{i=1}^l \alpha_i \langle \phi(x_i), \phi(x) \rangle_H + b \right) \quad 2.40$$

biçiminde gösterilir. Şekil 2.8' de giriş uzayı $X = \mathbb{R}^2$ ' de elipsoidal ayırma fonksiyonu, yeni öznitelik uzayındaki ($H = \mathbb{R}^3$) aşırı düzlemine göre eşleştirildiğinde;

$$\begin{aligned} \phi: \quad \mathbb{R}^2 &\rightarrow \mathbb{R}^3 \\ (x_1, x_2) &\mapsto (z_1, z_2, z_3) = (x_1^2, x_2^2, \sqrt{2}x_1x_2) \end{aligned} \quad 2.41$$

haline gelir.

Bununla birlikte, eğitim kümesindeki vektörlerin daha yüksek boyutlu uzaya dönüştürülmesinde bazı sayısal problemler meydana gelir. Yeni yüksek boyutlu öznitelik uzayı, eğitim vektörü x_i ' lere karşılık gelen $\phi(x_i)$ öznitelik vektörlerinin ifade edilmesini hafıza ve zaman bakımından çok zahmetli kılar. Ayrıca, dönüştürülen veriyi lineer olarak ayıran ϕ dönüşümünün bulunması çok zor olabilir.

Dikkat edilirse daha önce bahsedilen amaç fonksiyon ve aşırı düzlemin tanımı vektörler arasındaki iç çarpıma bağlıdır. Eğer x_i ' den dönüştürülmüş $\phi(x_i)$

öznitelik vektörleri arasındaki iç çarpımlar, $\phi(x_i)$ ’ yi hesaplayabilen bir fonksiyon bulursa, hesap karmaşıklığı büyük ölçüde giderilebilir. Böyle bir fonksiyon “Kernel Fonksiyonu” ile gerçekleştirilebilir ve Kernel Fonksiyonu;

$$K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j) \quad 2.42$$

biçimindedir⁽⁸⁴⁾. Araştırmacılar tarafından yeni kernel çeşitleri geliştirilmesine rağmen, literatürde bulunan dört temel kernel Çizelge 2.1’ de gösterilmektedir:

Çizelge 2.1 Dört Temel Kernel Tipi ve Fonksiyonları

KERNEL TİPİ	FONKSİYON
Lineer	$K(x_i, x_j) = x_i^T x_j$
Polinomal	$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$
Radial Basis Function (RBF)	$K(x_i, x_j) = \exp(-\gamma \ x_i - x_j\ ^2), \gamma > 0$
Sigmoid	$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

2.3.2.2. Çoklu Sınıflandırma İçin SVM

Destek vektör makineleri aslında ikili sınıflandırma için tasarlanmışlardır. Bunların etkin bir şekilde çoklu sınıflandırma için kullanılabilmesi hala devam eden bir araştırma konusudur. Buradan çoklu sınıflandırma problemlerinin (sınıf sayısı k olmak üzere, $k > 2$) SVM kullanılarak çözülmesinde genel olarak 3 metot vardır:

- **Multi-Class Ranking SVM (Çoklu Sınıflandırma)** : Bu yaklaşımda tüm sınıfların sınıflandırılması için tek bir SVM karar fonksiyonu vardır.

- **One-Against-Rest (One-Vs-Rest) Classification (Bir Sınıfın Diğerlerinden Ayrılması)** : Bu yaklaşım k tane ikili sınıflandırıcının düzenlenmesine göre çalışır. i ' nci sınıflandırıcı, i ' nci sınıftaki pozitif etiketli tüm örnekler ile negatif etiketli diğer tüm örneklerle eğitilir. Final çıkışı, en yüksek değerli sınıflandırıcıya karşılık gelen sınıftır (Winner Takes All).
- **One-Against-One (One-Vs-One) Classification (Bir Sınıfın Diğer Bir Sınıftan Ayrılması)** : Bu yaklaşım k adet sınıfın eğitim setinden olabilecek tüm ikili sınıflandırıcıları düzenler. Dolayısıyla k adet sınıfın ikili kombinasyonu kadar sınıflandırıcı oluşacaktır ($\binom{k}{2} = k(k-1)/2$). Final çıkışında ise her sınıflandırıcı tercih ettiği sınıf için bir oy verir ve en çok oy alan sınıfa göre çıkış belirlenir (Voting).

2.3.3. Adaptive Boosting (AdaBoost)

Boosting herhangi bir öğrenme algoritmasının performansını artıran genel bir metot olup, öğrenme algoritması tarafından üretilen zayıf sınıflandırıcılardan iyi bir öğrenme algoritması oluşturmayı amaçlar. Schapire⁽⁸⁵⁾ ilk boosting algoritmasının kendisi tarafından üretilmiş olduğunu ve Freund⁽⁸⁶⁾ tarafından geliştirildiğini belirtmiştir.

Boosting teorik ve deneysel olarak birçok araştırmacı tarafından incelenmiştir. En popüler boosting algoritması olan AdaBoost, Freund ve Schapire⁽⁸⁷⁾ tarafından ileri sürülmüş ve önceki boosting yaklaşımlarına göre üstünlük sağlamıştır. AdaBoost aynı zamanda çoklu sınıflandırma ve regresyon problemlerinin çözümü için de geliştirilmiştir^(87,88). AdaBoost algoritmasının çoklu

sınıflandırma problemlerinin çözümü için geliştirilen ilk versiyonu olan AdaBoost.M1, sınıflandırma işlemini Çizelge 2.2’ de gösterildiği şekilde yürütmektedir.

Çizelge 2.2 Çoklu Sınıflandırma İçin AdaBoost.M1 Algoritması

Verilen m örnekten oluşan eğitim seti $S = \langle (x_1, y_1), \dots, (x_m, y_m) \rangle$, $x_i \in X$ ve $y_i \in Y = \{1, \dots, k\}$ sınıf etiketlerini göstermek üzere üzere, ağırlık değerlerine başlangıç değerleri olarak $w_i^1 = D(i) = 1/m$, ($i = 1, \dots, m$) verilir.

Her $t = 1, 2, \dots, T$ için; (T iterasyon sayısını gösterir)

1. p^t bir olasılık dağılımını göstermek üzere ağırlıklar normalize edilir:

$$p_i^t = \frac{w_i^t}{\sum_{i=1}^m w_i^t}$$

2. S eğitim kümesi üzerinde p^t dağılımını sağlayan bir zayıf sınıflandırıcı $h_t : X \rightarrow Y$ şeklinde bir hipotez hesaplar. Zayıf sınıflandırıcının amacı eğitim hatasını minimuma indiren bir hipotez bulmaktır.
3. $h_t : \varepsilon_t = \sum_{i=1}^m p_i^t h_t(x_i) \neq y_i$ olarak ifade edilen eğitim hatası hesaplanır. Bu hata zayıf sınıflandırıcı tarafından sağlanan p^t dağılımına göre hesaplanır. Eğer hata $\varepsilon_t > 1/2$ ise, $T = t - 1$ olacak şekilde ayarlanır ve döngüden çıkılır.
4. $\beta_t = \varepsilon_t / (1 - \varepsilon_t)$ değeri hesaplanır.
5. Ağırlıklar $w_i^{t+1} = w_i^t \beta_t^{1 - h_t(x_i) \neq y_i}$ ile hesaplanan yeni değerlerine güncellenir.

Bu işlem T iterasyon sayısı boyunca devam eder ve h_1, \dots, h_T zayıf sınıflandırıcıları birleştirilerek kuvvetli sınıflandırıcı bulunmuş olur⁽⁸⁷⁾.

$$h_f(x) = \arg \max_{y \in Y} \sum_{t=1}^T \left(\log \frac{1}{\beta_t} \right) h_t(x_i) = y_i$$

3. ARAŞTIRMA BULGULARI VE TARTIŞMA

3.1. Giriş

Yüz ifade analizinde, yüz görüntülerinin taşıdıkları ifadelere göre sınıflandırılması için 4 farklı deney yapılmıştır. Yapılan deneylerde daha önce açıklanan k-NN (k en yakın komşuluk), SVM (Destek Vektör Makineleri) ve AdaBoost sınıflandırıcılarının başarımları gözlenmiştir. Ayrıca öznitelik dönüşümü ve öznitelik seçimi gibi çeşitli algoritmalar kullanılarak boyutu azaltılan öznitelik kümesinin, sınıflandırıcılar üzerindeki etkisi incelenmiş ve kullanılan algoritmalar arasında çeşitli karşılaştırmalar yapılmıştır.

Yapılan deneyler, çeşitli paket programlar^(89,90) kullanılarak AMD Athlon 3000+ 1.8GHz 64-bit işlemcili ve 1GB bellek birimine sahip bir bilgisayar ile MATLAB ortamında gerçekleştirilmiştir.

3.2. Yüz İfade Görüntüleri Veri Seti

Bu çalışmada, Cohn and Kanade DFAT-504⁽¹⁵⁾ veri setinden alınan görüntüler kullanılmıştır. Bu veri seti, yaşları 18-30 arasında değişen 100 üniversite öğrencisine ait görüntülerden oluşmaktadır. Öğrencilerin %65' i bayan, %15' i zenci ve %3' ü Uzak Doğu ve Latin kökenlidir. Görüntüler, analog S-video kamera ile kişilerin ön taraflarından kaydedilmiştir. Kişilerden 23 tane yüz ifadesi yapmaları istenmiştir. Görüntüler 640x480 veya 640x490 piksel boyutlarında ve 8-bit gri tona sahiptir.

Bu veri setinden alınan görüntüler her biri farklı duygusal içerik taşıyan 6 temel yüz ifadesine göre sınıflandırılmıştır. Bu ifadeler temel duygular olarak adlandırılır ve tüm etnik ve kültürlerde evrenseldir. Buradan duygusal görünümlere göre 6 yüz ifadesi **öfke**, **iğrenme**, **korku**, **mutluluk**, **üzüntü** ve **şaşkınlık** olarak sıralanır. Cohn and Kanade DFAT-504⁽¹⁵⁾ veri setinden alınan ve 6 yüz ifadesini gösteren bir örnek Şekil 3.1’ de gösterilmiştir.



Şekil 3.1 6 adet yüz ifadesine bir örnek (Cohn and Kanade DFAT-504 veri seti)

Cohn and Kanade DFAT-504⁽¹⁵⁾ veri setinde farklı kişilere ait görüntülerin bulunduğu klasörlerdeki son resim 6 yüz ifadesinden herhangi birinin en çok belirgin olduğu durumu nitelermektedir. Dolayısıyla, yüz ifadeleri sınıflandırılırken her bir kişinin içinde bulunduğu durumun son görüntüsü kullanılmıştır. Buradan 6 temel yüz ifadesine göre görüntüler ayrı ayrı klasörlere ayrılmış ve meydana gelen her bir ifadeye ait klasördeki görüntü sayıları Çizelge 3.1’ de gösterilmiştir.

Çizelge 3.1 Her bir ifadeye ait görüntü sayıları

Yüz ifade klasörü	Görüntü sayısı
Öfke	47
İğrenme	42
Korku	50
Mutluluk	54
Üzüntü	50
Şaşkınlık	51
TOPLAM	294

Yüz ifade klasörleri oluşturulurken farklı kişilere ait görüntüler kullanılmıştır. Çünkü aynı ifade sınıfı için farklı kişilerden alınan örnekler o ifadeye ait özellikleri daha belirgin yapacaktır. Ayrıca Çizelge 3.1’ de görüldüğü gibi, her bir ifade klasöründeki görüntü sayıları farklıdır. Bunun nedeni, veri setini oluşturan görüntüleri veren öğrencilerin her bir duyguyu profesyonelce ifade edememeleridir.

Her bir görüntünün hangi yüz ifadesine ait olduğunun belirlenmesi esnasında, farklı kişilere danışılarak sınıflandırmanın daha doğru bir şekilde yapılması sağlanmıştır. Görüntülerin yüz ifadelerine göre sınıflandırılması tüm çalışmanın sonucunu doğrudan etkileyeceği için üzerinde titizlikle durulmuştur.

3.3. Ön İşlemler

Yüz ifade analizinde bir görüntünün sadece ifadeyi taşıyan kısmı, yani yüz bölgesi önemlidir. Buna göre, daha önce 6 adet ifade sınıfı klasörüne göre sınıflandırılan görüntülerin yüz kısımları ayrıştırılmış ve elde edilen yeni yüz görüntüleri 48x48 piksel boyutlarına indirgenmiştir. Böylelikle, işlem fazlalığı

azaltılmış ve belleğin gereksiz yere kullanılmaması sağlanmıştır. Şekil 3.2’ de iki görüntünün, yüz kısımlarının ayrıştırılmış halleri gösterilmiştir.



Şekil 3.2 İki farklı görüntünün yüz kısımlarının ayrıştırılmış halleri

Bu işlemlerin hepsi herhangi bir programlama tekniği kullanılmadan el ile yapılmıştır. Çünkü literatürde bulunan otomatik yüz belirleme teknikleri^(11,22) herhangi bir görüntüye ait yüz bölgesinin ifade analizinde hiç bir önemi olmayan kısımlarını da içerecek şekilde belirlediğinden çalışmada kullanılmamıştır.

İfade tasviri, bir görüntünün baş kısmının konumunun yer değiştirmesi, ölçeklendirilmesi ve döndürülmesine karşı hassas olabilir. Bu tür istenmeyen durumların etkisini ortadan kaldırmak için yüz görüntüsü geometrik olarak standartlaştırılabilir. Bu normalizasyon genellikle gözler veya burun delikleri referans alınarak yapılır. Bu yüzden yüz görüntülerinin elde edilmesi esnasında eğik duran görüntüler düzeltilmiş ve tüm yüz görüntüleri olabildiğince göz hizalı olacak biçimde Şekil 3.3’ te gösterildiği gibi ayarlanmıştır.



Şekil 3.3 Farklı kişilere ait göz hizalı 6 yüz ifadesi

3.4. Deney 1

Deney 1’ de, k-NN (k – En Yakın Komşuluk) ve SVM (Destek Vektör Makineleri) sınıflandırıcılarının, herhangi bir öznitelik indirgemesi yapılmadan 6 farklı yüz ifadesinin sınıflandırmasındaki başarımlarına değinilmiştir. Yapılan işlemler 5 adımda gerçekleştirilmiş olup, sırasıyla sonraki bölümlerde açıklanmıştır.

Adım 1: Sınıflandırılmak Üzere Öznitelik Veri Kümesinin Oluşturulması

Uygulamanın ilk aşamasında 6 farklı yüz ifadesinin sınıflandırılması için öncelikle, geleneksel örüntü tanıma problemlerinin çözümünde olduğu gibi, tüm ifadelerine ait özniteliklerin bulunduğu bir öznitelik kümesi oluşturulmuştur. Öznitelik kümesi, yüz ifadelerine ait her bir görüntünün (öfke için 47, iğrenme için 42, korku için 50, mutluluk için 54, üzüntü için 50 ve şaşkınlık için 51 adet olmak üzere toplam 294 adet) farklı Gabor dalgacık filtreleri ile işleme tabi tutulmasıyla elde edilmiştir.

Bu çalışmada 48x48 boyutlarında görüntüler 3 farklı frekans ($\nu \in \{0,1,2\}$) ve 4 farklı yönelimde ($\mu \in \{0,1,2,3\}$) olmak üzere toplam 12 farklı Gabor dalgacığı ile işleme tabi tutulduğundan, her bir görüntü için toplam $48 \times 48 \times 3 \times 4 = 27648$ tane öznitelik elde edilmiştir. Bu durumda, sözü edilen öznitelik kümesinin genel yapısı Şekil 3.4 ile gösterildiği gibi olacaktır.

Öfke	⇒	47 görüntü	294x27648
İğrenme	⇒	42 görüntü	
Korku	⇒	50 görüntü	
Mutluluk	⇒	54 görüntü	
Üzüntü	⇒	50 görüntü	
Şaşkınlık	⇒	51 görüntü	
27648 adet öznitelik			

Şekil 3.4 Öznitelik Kümesi Genel Yapısı

Öznitelik veri kümesinde bulunan 6 sınıfın birbirinden ayırt edilmesi amacıyla kodlama yapılmıştır. Bu kodlama yapısı Şekil 3.5’ te gösterilmektedir.

+1	-1	-1	-1	-1	-1	⇒	47 satır	294x6
-1	+1	-1	-1	-1	-1	⇒	42 satır	
-1	-1	+1	-1	-1	-1	⇒	50 satır	
-1	-1	-1	+1	-1	-1	⇒	54 satır	
-1	-1	-1	-1	+1	-1	⇒	50 satır	
-1	-1	-1	-1	-1	+1	⇒	51 satır	

Şekil 3.5 Kodlama Matrisi ve İfade Görüntü Sayılarına Göre Boyutu

Kodlama matrisindeki +1’ ler verinin ilgili sınıfa ait olduğunu belirlerken, geriye kalan 5 adet -1’ de verinin diğer sınıflarla herhangi bir ilgisinin bulunmadığını ifade eder. Kodlama matrisinde, her satır bir ifadeye ait görüntüyü temsil etmektedir. Örneğin, +1 -1 -1 -1 -1 -1 vektörü “öfke” ifadesini

belirtmektedir. 47 tane “öfke” görüntüsü olduğu için, matriste bu vektörden 47 tane satır olarak bulunmaktadır. Bir diğer deyişle yukarıdaki kodlama matrisinde “47 satır” ile anlatılmak istenen, ilgili vektörden 47 tane olduğu ve bunların her birinin de birer “öfke” görüntüsünü ifade ettiğiidir. Benzer şekilde 42 satır ile ilgili vektörden 42 adet olduğu ve bunların her birinin birer “iğrenme” görüntüsü olduğu sonucuna varılabilir. Diğer vektörlerle de sırasıyla belirtilen sayılarda diğer ifadeler temsil edilmektedir.

Adım 2: Öznitelik Veri Kümesi Üzerinde Yapılan Ölçeklendirme

Öznitelik kümesi oluşturulduktan sonra ölçeklendirme yapılmıştır. Öznitelik kümesinin sınıflandırma işleminden önce ölçeklendirilmesi çok önemli olup, Sarle⁽⁹¹⁾ herhangi bir sınıflandırıcı kullanılırken neden ölçeklendirme yapılması gerektiğini açıklamıştır.

Ölçeklendirme yapılmasının en büyük avantajı, öznitelikleri daha küçük sayısal değer aralığına indirgeyerek çok büyük sayısal aralıklardan kaçınmaktadır. Diğer bir avantaj ise hesaplamalar boyunca meydana gelen sayısal zorluklardan kurtulmaktır. Çünkü kernel değerleri genellikle öznitelik vektörlerinin iç çarpımlarına bağlıdır ve büyük öznitelik değerleri sayısal problemler doğurabilir. Hsu vd.⁽⁹²⁾, öznitelik değerlerinin $[-1,+1]$ veya $[0,+1]$ aralığına gelecek biçimde ölçeklendirilmesini önermektedir.

Yapılan çalışmada veri kümesi $[0,+1]$ aralığına ölçeklendirilmiş ve bu işlem;

$$\tilde{X}(i) = \left[\frac{X(i) - \min A}{\max A - \min A} \right] (\text{yeni_max } A - \text{yeni_min } A) + \text{yeni_min } A \quad 3.1$$

formülasyonu ile gerçekleştirilmiştir. O halde N öznitelik sayısını göstermek üzere, A veri setinde bulunan her bir $X(i)$ özneliği $[0,+1]$ aralığına ölçeklendirilmiş olur.

Adım 3: Sınıflandırıcıların Kullanılmasındaki Yaklaşımlar

Öznitelik kümesi sınıflandırıcılar için uygun hale getirildikten sonra bu deneyde kullanılan sınıflandırıcılardan k-NN, direkt çoklu sınıflandırma için kullanılmıştır. SVM ise hem çoklu sınıflandırma (Multi-Class SVM – MC-SVM), hem de çoklu sınıflandırma probleminin çözümünde kullanılan, “bir sınıfın diğer bir sınıftan ayrılması” (One-Vs-One SVM) ve “bir sınıfın diğer sınıflardan ayrılması” (One-Vs-Rest SVM) biçiminde belirtilen ikili sınıflandırma problemi çözüm yaklaşımları olmak üzere 3 şekilde kullanılmıştır.

Adım 4: Sınıflandırıcı Algoritmalarının Parametre Seçimleri

En iyi sınıflandırıcı başarımlarının elde edilmesi amacıyla, öznitelik veri kümesi için en iyi parametrelerin seçilmesine dikkat edilmiştir. k-NN sınıflandırıcısında k parametresinin $k = \{1,2,3,\dots,15\}$ değerleri için denemeler yapılmış ve en iyi sonucun $k=9$ en yakın komşuluk değeri için alındığı gözlenmiştir.

SVM sınıflandırıcısı da, daha önce açıklanmış olan kernel tiplerinden birisi olan RBF kernel ile birlikte kullanılmıştır. Çünkü RBF kernel, örneklerin daha yüksek boyutlu bir uzaya eşleştirilmesini lineer olmayan bir biçimde yapar ve böylece lineer kernelin aksine, sınıf etiketleri ile öznitelikler arasındaki ilişkinin lineer olmadığı durum için çözüm sunabilir⁽⁹²⁾. RBF kernel kullanılmasının ikinci nedeni ise, RBF kernel’ in polinomal kernel’ e göre daha az parametreye sahip

olmasıdır. Çünkü sınıflandırma performansı, artan parametre sayısından olumsuz yönde etkilenmektedir.

RBF kernel parametreleri C ve γ 'nın öznitelik kümesi için en uygun değerlerinin seçimi amacıyla çapraz geçerlilik (cross-validation - CV) tekniği kullanılmıştır. v -fold çapraz geçerlilik tekniğinde, önce eğitim kümesi v tane eşit boyuttaki alt kümeye bölünür. Sırasıyla her bir alt küme, geri kalan $v-1$ alt küme üzerinde eğitilen sınıflandırıcı kullanılarak test edilir. Buradan tüm eğitim kümesinin her mesafesi bir defa tahmin edilir ve böylece çapraz geçerlilik doğruluğu, doğru bir biçimde sınıflandırılan verinin yüzdesel olarak oranı olacaktır⁽⁹²⁾.

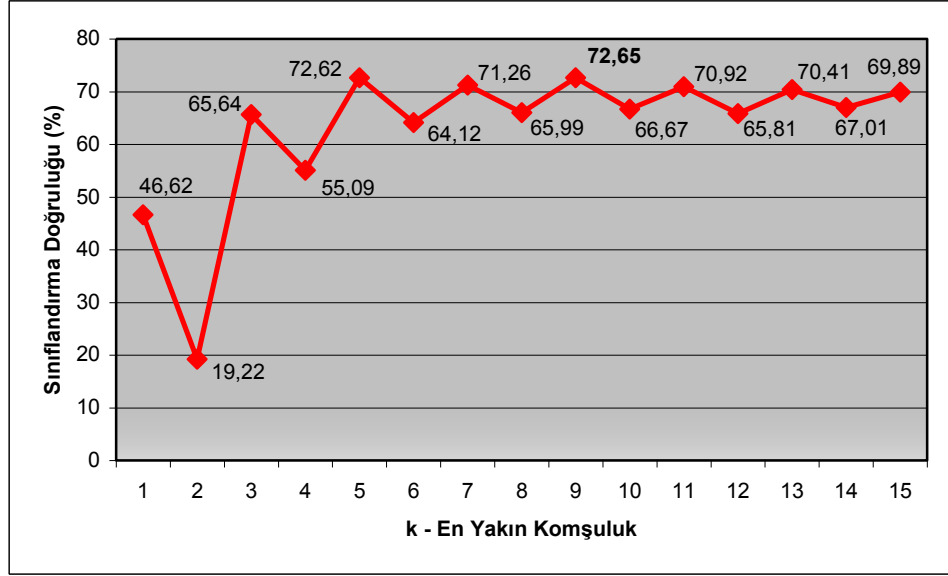
Bu çalışmada 5-fold için çapraz geçerlilik tekniği kullanılmış ve SVM sınıflandırıcısında en düşük sınıflandırma hatasını veren C ve γ parametrelerinin değerleri sırasıyla ∞ ve 16 olarak belirlenmiştir.

Adım 5: Alınan Sonuçlar ve Değerlendirme

SVM ve k-NN sınıflandırıcıları ile elde edilen sınıflandırma başarımları sırasıyla Çizelge 3.2 ve Şekil 3.6' da gösterilmiştir.

Çizelge 3.2 Öznitelik indirgemesi yapılmadan elde edilen SVM sınıflandırıcı başarımları

Boyut İndirgeme	SVM (One-Vs-One)	SVM (One-Vs-Rest)	SVM (Multi-Class)
Uygulanmadı	% 85,38	% 85,72	% 84,03



Şekil 3.6 Öznitelik indirgemesi yapılmadan elde edilen k-NN sınıflandırıcı başarımları

Çizelge 3.2’ den de görülebileceği üzere en iyi sınıflandırıcı başarımları %85,72 ile SVM’ nin One-Vs-Rest yaklaşımı ile kullanıldığı durumda elde edilmiştir. Literatürde bu sonuçları destekleyecek yönde, One-Vs-Rest yaklaşımı kullanılarak yapılan sınıflandırma ile çok daha doğru sonuçların alındığı görülebilir⁽⁹³⁻⁹⁵⁾. En basit sınıflandırıcılardan biri olarak gösterilen k-NN ile alınan en iyi sınıflandırma başarımları ise, k=9 en yakın komşuluk değeri için %72,65 olarak gözlenmiştir.

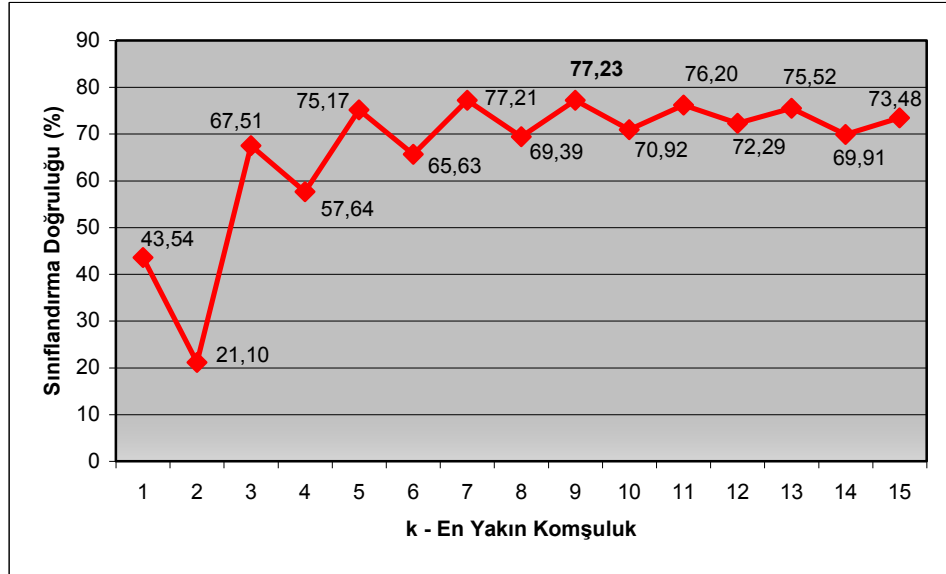
3.5. Deney 2

Bu deneyde k-NN ve SVM sınıflandırma algoritma başarımlarını artırmak ve gereksiz işlem yükünden kurtulmak amacıyla, “Temel Bileşenler Analizi (TBA)” kullanılarak Deney 1’ de elde edilen öznitelik kümesinin boyutu indirgenmiştir. Literatürde “genel varyansın en az %67’sini açıklayan sayıda temel bileşen” seçilmesinin yeterli olduğu belirtilmiş olmasına rağmen, bu çalışmada bu oran %95’e

çıkarılarak, daha fazla sayıda temel bileşenle çalışılmış, böylece çok daha doğru sonuçlar elde edilmiştir. Boyutu indirgenen öznitelik kümesi için alınan SVM ve k-NN sınıflandırıcı performansları sırasıyla Çizelge 3.3 ve Şekil 3.7’ de gösterilmiştir.

Çizelge 3.3 TBA ile boyut indirgemesi yapıldıktan sonra elde edilen SVM sınıflandırıcı başarımları

Boyut İndirgeme	SVM (One-Vs-One)	SVM (One-Vs-Rest)	SVM (Multi-Class)
TBA	% 86,40	% 87,77	% 83,34



Şekil 3.7 TBA ile boyut indirgemesi yapıldıktan sonra elde edilen k-NN sınıflandırıcı başarımları

Görülebileceği üzere, Temel Bileşenler Analizi ile boyutu indirgenen öznitelik kümesi her iki sınıflandırıcının da performansını az da olsa olumlu yönde etkilemiştir. En iyi sınıflandırma başarımları, SVM(One-Vs-Rest) sınıflandırıcısı ile %87,77 ve yine k=9 en yakın komşuluk değerinde k-NN sınıflandırıcısı ile %77,23

olarak alınmıştır. Boyut azalmasının sınıflandırma sürelerinde, Deney 1 ile karşılaştırma yapıldığında önemli miktarlarda kazanç sağladığı görülmüştür.

3.6. Deney 3

Bu uygulamada yüz ifade sınıflandırılmasında öznitelik seçiminin nasıl bir etki yaptığı incelenmiştir. Öznitelik seçimi, “iki sınıflı” ve “çok sınıflı” öznitelik seçimi olmak üzere 2 farklı biçimde gerçekleştirilmiştir. Öznitelik seçme algoritmalarından L0, MutInf ve RFE sadece “iki sınıflı” öznitelik seçimi için, Fisher ise hem “iki sınıflı” hem de “çok sınıflı” öznitelik seçimi için kullanılmıştır. Deney 1 ve Deney 2’ de en iyi sınıflandırma başarımını SVM algoritması verdiği için bu uygulamada tercih edilmiştir.

3.6.1. İki Sınıflı Öznitelik Seçimi

İki sınıflı öznitelik seçiminde, öznitelik seçiminin yapılması için kullanılacak olan öznitelik kümeleri iki farklı yaklaşım ile oluşturulmuştur. Bu yaklaşımlar çoklu sınıflandırma probleminin literatürde “One-Vs-Rest” ve “One-Vs-One” biçiminde ifade edilen ikili sınıflandırma problemi olarak formülize edilmesine dayanmaktadır.

İki sınıflı öznitelik seçiminde yapılan işlemler 4 adımda gerçekleştirilmiş olup, sonraki bölümlerde bu işlemler sırasıyla ayrıntılı bir biçimde anlatılmıştır.

Adım 1: Öznitelik Seçiminde “One-Vs-Rest” Yaklaşımı

Bu yaklaşıma göre oluşturulan öznitelik kümeleri, sınıflandırma probleminin literatürde One-Vs-Rest biçiminde ifade edilen ikili sınıflandırma problemi olarak gösterilmesine dayanılarak oluşturulmuştur. One-Vs-Rest yaklaşımına göre, ikili

sınıflandırma problemi yüz ifade sınıflarından birinin diğerlerinden ayrıştırılması biçiminde ifade edilir. Çalışmada 6 yüz ifadesinin analizi yapıldığından meydana gelen yüz ifadeleri kümesi;

$$\text{Yüz İfadeleri} = \{\text{Öfke, İğrenme, Korku, Mutluluk, Üzüntü, Şaşkınlık}\}$$

biçiminde olacaktır. One-Vs-Rest yaklaşımının yapısı gereği meydana gelebilecek öznitelik küme sayısı da yüz ifadeleri kümesinin eleman sayısı kadar olacaktır. Sınıflandırma probleminin çözümü için gereken öznitelik kümeleri Çizelge 3.4' te gösterildiği üzere 6 adet ikili sınıflandırma problemi olarak ifade edilir.

Çizelge 3.4 One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümeleri

Öfke	–	Diğerleri
İğrenme	–	Diğerleri
Korku	–	Diğerleri
Mutluluk	–	Diğerleri
Üzüntü	–	Diğerleri
Şaşkınlık	–	Diğerleri

Öznitelik kümelerinden her bir öznitelik seçme algoritması için toplam 6 adet öznitelik altkümesi elde edilir. Herhangi bir öznitelik seçme algoritması için seçilen öznitelik altkümeleri sırasıyla;

$$S1 = \{ \text{Öfke-Diğerleri veri dosyasından seçilen öznitelik alt kümesi} \}$$

$$S2 = \{ \text{İğrenme-Diğerleri veri dosyasından seçilen öznitelik alt kümesi} \}$$

$$S3 = \{ \text{Korku-Diğerleri veri dosyasından seçilen öznitelik alt kümesi} \}$$

$$S4 = \{ \text{Mutluluk-Diğerleri veri dosyasından seçilen öznitelik alt kümesi} \}$$

$$S5 = \{ \text{Üzüntü-Diğerleri veri dosyasından seçilen öznitelik alt kümesi} \}$$

$$S6 = \{ \text{Şaşkınlık-Diğerleri veri dosyasından seçilen öznitelik alt kümesi} \}$$

biçiminde olacaktır.

Adım 2: Öznitelik Seçiminde “One-Vs-One” Yaklaşımı

Öznitelik seçiminde kullanılan bu yaklaşıma göre ise, oluşturulan öznitelik kümeleri, sınıflandırma probleminin literatürde One-Vs-One biçiminde ifade edilen ikili sınıflandırma problemi biçiminde gösterilmesine dayanılarak oluşturulmuştur. One-Vs-One yaklaşımına göre, ikili sınıflandırma problemi yüz ifade sınıflarından birinin diğerinden ayrıştırılması biçiminde ifade edilir. Bu durumda k ifade sınıf sayısını göstermek üzere, ikili sınıflandırma problemi sayısı k ' nin ikili kombinasyonu kadar ($\binom{k}{2} = k(k-1)/2$) olacaktır. Yapılan bu çalışmada 6 ifade sınıfı olduğundan toplam 15 adet ikili sınıflandırma problemi meydana gelir ve bunlara karşılık gelen öznitelik kümeleri de Çizelge 3.5' te gösterilmiştir.

Çizelge 3.5 One-vs-One yaklaşımına göre oluşturulan öznitelik kümeleri

Öfke – İğrenme	İğrenme – Korku	Korku – Üzüntü
Öfke – Korku	İğrenme – Mutluluk	Korku – Şaşkınlık
Öfke – Mutluluk	İğrenme – Üzüntü	Mutluluk – Üzüntü
Öfke – Üzüntü	İğrenme – Şaşkınlık	Mutluluk – Şaşkınlık
Öfke – Şaşkınlık	Korku – Mutluluk	Üzüntü – Şaşkınlık

One-Vs-Rest yaklaşımına benzer şekilde her bir öznitelik seçme algoritması için toplam 15 adet öznitelik altkümüsi elde edilir.

Adım 3: Öznitelik Final Kümesinin Oluşturulması

Her iki yaklaşıma göre elde edilen öznitelik kümelerinden, her bir öznitelik seçme algoritması için meydana gelen öznitelik altkümeleri kendi aralarında birleştirilerek sınıflandırmak üzere kullanılacak öznitelik kümesi oluşturulur. S_i

öznitelik alt kümesini, T ilgili yaklaşıma göre oluşturulan öznitelik altküme sayısını ve “*algo*” kullanılan öznitelik seçme algoritmalarından birini göstermek üzere, sınıflandırmak üzere kullanılacak öznitelik final kümesi F , matematiksel olarak;

$$F_{algo} = \bigcup_{i=1}^T S_i \quad 3.2$$

biçiminde ifade edilir.

Adım 4: Alınan Sonuçlar ve Değerlendirme

İkili öznitelik seçiminde, “One-Vs-Rest” ve “One-Vs-One” yaklaşımlarına göre oluşturulan öznitelik kümelerinden her bir öznitelik seçme algoritmasıyla, %1 ve %3 oranlarında öznitelik seçilmiş ve kendi aralarında birleştirilerek öznitelik final kümesi elde edilmiştir. Öznitelik final kümeleri SVM sınıflandırıcıları ile sınıflandırılmıştır. Deney 1’ de olduğu gibi, SVM sınıflandırıcıları RBF (Radial Basis Function) kernel ile birlikte kullanılmış ve kernel parametreleri C ve γ ’ nın değerleri sırasıyla ∞ ve 16 olarak belirlenmiştir.

Çizelge 3.6 ve 3.7’ de “One-Vs-Rest” yaklaşımına göre oluşturulan öznitelik kümelerinden %1 ve %3 oranlarında, 4 farklı öznitelik seçme algoritması ile seçilen öznitelikler için sınıflandırıcı başarımları bulunmaktadır. Öznitelik sayıları arasındaki farklılık, ilgili öznitelik seçme algoritmasının yapısına göre değişkenlik göstermektedir. Benzer şekilde Çizelge 3.8 ve 3.9’ da, “One-Vs-One” yaklaşımına göre oluşturulan öznitelik kümelerinden seçilen öznitelikler için sınıflandırıcı başarımları bulunmaktadır.

Çizelge 3.6 One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümelerinden %1 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik Seçme Algoritması	Öznitelik sayısı	SVM 1	SVM 2	SVM 3
One-Vs-Rest (%1)	Fisher	2220	% 87,18	% 86,75	% 86,76
	L0	2515	% 90,16	% 92,53	% 90,48
	MutInf	2318	% 87,42	% 85,38	% 86,07
	RFE	2674	% 91,85	% 92,52	% 92,86

Çizelge 3.7 One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümelerinden %3 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik Seçme Algoritması	Öznitelik sayısı	SVM 1	SVM 2	SVM 3
One-Vs-Rest (%3)	Fisher	5253	% 85,05	% 87,76	% 87,77
	L0	6080	% 91,51	% 90,15	% 89,12
	MutInf	5494	% 87,77	% 87,76	% 87,08
	RFE	6343	% 91,85	% 91,50	% 89,46

Çizelge 3.8 One-Vs-One yaklaşımına göre oluşturulan öznitelik kümelerinden %1 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik Seçme Algoritması	Öznitelik sayısı	SVM 1	SVM 2	SVM 3
One-Vs-One (%1)	Fisher	2301	% 88,45	% 86,07	% 87,43
	L0	2538	% 89,47	% 90,49	% 87,43
	MutInf	2301	% 87,76	% 88,11	% 88,78
	RFE	2633	% 90,82	% 93,19	% 88,78

Çizelge 3.9 One-Vs-One yaklaşımına göre oluşturulan öznitelik kümelerinden %3 oranında seçilen öznitelikler için alınan SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik Seçme Algoritması	Öznitelik Sayısı	SVM 1	SVM 2	SVM 3
One-Vs-One (%3)	Fisher	5642	% 87,43	% 88,44	% 86,41
	L0	5990	% 89,47	% 88,11	% 88,78
	MutInf	5562	% 89,12	% 89,46	% 86,73
	RFE	6198	% 90,49	% 89,13	% 88,79

Alınan sonuçlar, öznitelik kümesi oluşturma yaklaşımlarının, seçilen öznitelik oranlarının ve SVM sınıflandırıcılarının kendi aralarında karşılaştırılabilmesi amacıyla Çizelge 3.10 ve 3.11’ de gösterildiği gibi özetlenebilir.

Çizelge 3.10 One-Vs-Rest yaklaşımına göre oluşturulan öznitelik kümelerinden %1 ve %3 oranlarında seçilen öznitelikler için alınan ortalama SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	SVM 1	SVM 2	SVM 3
One-Vs-Rest (%1)	% 89,15	% 89,30	% 89,04
One-Vs-Rest (%3)	% 89,05	% 89,29	% 88,36

Çizelge 3.11 One-Vs-One yaklaşımına göre oluşturulan öznitelik kümelerinden %1 ve %3 oranlarında seçilen öznitelikler için alınan ortalama SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	SVM 1	SVM 2	SVM 3
One-Vs-One (%1)	% 89,13	% 89,47	% 88,11
One-Vs-One (%3)	% 88,79	% 89,13	% 87,68

Çizelge 3.10 ve 3.11’ den görüleceği üzere “One-Vs-Rest” yaklaşımına göre oluşturulan öznitelik kümelerinden seçilen öznitelikler ile daha yüksek sınıflandırma doğruluğu elde edilmiştir. Dolayısıyla One-Vs-Rest” yaklaşımına göre oluşturulan öznitelik kümelerinden sınıf ayırımı daha fazla olan öznitelikler seçilebilmiştir.

Seçilen öznitelik oranlarının sınıflandırma başarımı üzerindeki etkileri incelendiğinde ise, daha az sayıda öznitelik ile daha doğru sonuçların alındığı gözlenmiştir. Bu durum, seçilen öznitelik sayısı arttıkça, ilgisiz ve işe yaramayan özniteliklerin seçildiğinin bir göstergesidir. Ayrıca, Deney 1 ve Deney 2’ de alınan sonuçlara paralel olarak, en iyi sınıflandırma başarımı yine SVM(One-Vs-Rest) sınıflandırıcısı ile elde edilmiştir. Kullanılan öznitelik seçme algoritmalarının SVM

sınıflandırıcıları üzerindeki etkileri ise, öznitelik kümesi oluşturma yaklaşımlarına göre Çizelge 3.12 ve 3.13’ de gösterilmiştir.

Çizelge 3.12 “One-Vs-Rest” öznitelik kümesi oluşturma yaklaşımına göre öznitelik seçme algoritmaları için ortalama SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik Seçme Algoritması	SVM 1	SVM 2	SVM 3
One-Vs-Rest	Fisher	% 86,12	% 87,26	% 87,27
	L0	% 90,84	% 91,34	% 89,80
	MutInf	% 87,59	% 86,57	% 86,58
	RFE	% 91,85	% 92,01	% 91,16

Çizelge 3.13 “One-Vs-One” öznitelik kümesi oluşturma yaklaşımına göre öznitelik seçme algoritmaları için ortalama SVM sınıflandırıcı başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik Seçme Algoritması	SVM 1	SVM 2	SVM 3
One-Vs-One	Fisher	% 87,94	% 87,26	% 86,92
	L0	% 89,47	% 89,30	% 88,11
	MutInf	% 88,44	% 88,79	% 87,76
	RFE	% 90,66	% 91,16	% 88,79

Her iki çizelgeden de görüleceği üzere, yüz ifade sınıflandırılmasında RFE algoritması ile seçilen öznitelikler ile en iyi sınıflandırma performansı elde edildiğinden RFE, kullanılan diğer öznitelik seçme algoritmaları içinde en etkili olanıdır.

3.6.2. Çok Sınıflı Öznitelik Seçimi

Çok sınıflı öznitelik seçimi, farklı ifadelerle ait öznitelikler içeren “çok sınıflı öznitelik kümesi” nin boyutunun azaltılması amacıyla uygulanmıştır. Bu çalışmada “Fisher Ölçüt Skoru” algoritması bu amaç için kullanılmıştır. Bu deneyde, ikili öznitelik seçilmesinde “One-Vs-Rest” ve “One-Vs-One” yaklaşımlarına göre oluşturulan öznitelik kümelerinden, %1 ve %3 oranlarında Fisher ile seçilen özniteliklerin final kümesindeki toplam sayıları kadar öznitelik seçilmiştir. Elde edilen sınıflandırma başarımları Çizelge 3.14’ te gösterilmiştir.

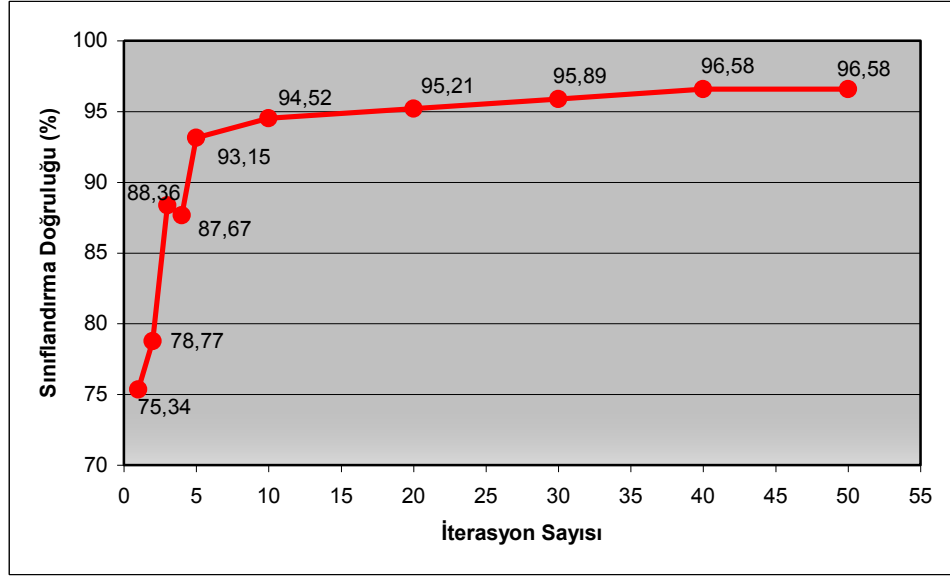
Çizelge 3.14 İki ve çok sınıflı kümelerden Fisher ile seçilen öznitelikler için SVM başarımları

Öznitelik kümesi oluşturma yaklaşımı	Öznitelik sayısı	SVM 1	SVM 2	SVM 3
Fisher (One-Vs-One - %1)	2301	% 88,45	% 86,07	% 87,43
Fisher (Çok sınıflı)	2301	% 86,07	% 85,04	% 84,02
Fisher (One-Vs-Rest - %1)	2220	% 87,18	% 86,75	% 86,76
Fisher (Çok sınıflı)	2220	% 86,07	% 86,06	% 84,35
Fisher (One-Vs-One - %3)	5642	% 87,43	% 88,44	% 86,41
Fisher (Çok sınıflı)	5642	% 85,46	% 86,42	% 86,38
Fisher (One-Vs-Rest - %3)	5253	% 85,05	% 87,76	% 87,77
Fisher (Çok sınıflı)	5253	% 84,56	% 86,41	% 86,75

Çizelge 3.14’ ten elde edilen sonuçlar öznitelik seçiminde, iki sınıflı öznitelik kümelerinden seçilen özniteliklerin, çok sınıflı öznitelik kümesinden seçilen özniteliklere nazaran daha çok bilgi taşıdıklarını göstermektedir.

3.7. Deney 4

Bu deneyde, 6 farklı yüz ifadesine ait özniteliklerin bulunduğu küme üzerinde, çoklu sınıflandırma için tasarlanan AdaBoost algoritmasının başarımı izlenmiştir. Zayıf sınıflandırıcı olarak 1-NN (en yakın komşuluk) algoritması kullanılmıştır. AdaBoost sınıflandırıcısı ile alınan sonuçlar Şekil 3.8’ de gösterilmiştir.



Şekil 3.8 AdaBoost Sınıflandırıcısı Başarımı

Şimdiye kadar gerçekleştirilen deneylerde gözlenen sınıflandırma başarımlarından çok daha iyi sonuçların alındığı AdaBoost sınıflandırıcısı ile sadece 5 iterasyon sonunda %93,15 seviyesinde sınıflandırma başarımı elde edilmiştir. 50 iterasyon sonunda %96,58 oranında çok yüksek bir sınıflandırma doğruluğu elde edilmiş olmasına rağmen, iterasyon sayısı arttıkça sınıflandırma süresi de oldukça uzamaktadır.

4. SONUÇ

Bu çalışmada, bir makine öğrenme problemi olan yüz ifadelerinin sınıflandırılmasında, çeşitli sınıflandırma ve öznitelik boyut indirgeme algoritmaları kullanılarak karşılaştırmalı bir analiz yapılmıştır.

Yapılan ilk deneyde, geleneksel örüntü tanıma problemlerinde olduğu gibi, öznitelik kümesinde herhangi bir boyut indirgemesi yapılmadan k-NN ve SVM sınıflandırıcıları ile ifade sınıflandırılması gerçekleştirilmiştir. Ardından literatürde oldukça sık kullanılan ve bir denetimsiz öznitelik dönüşüm tekniği olan TBA ile öznitelik kümesinin boyutu, sınıflandırılma yapılmadan önce azaltılmıştır. Boyut azalmasının sınıflandırma başarımları ve süreleri açısından, k-NN ve SVM sınıflandırıcıları üzerinde olumlu etkisinin olduğu gözlenmiştir. Sonuçlar, k-NN sınıflandırıcısına göre SVM sınıflandırıcıları ile çok daha doğru sonuçlar alındığını göstermiştir. İfade sınıflandırması probleminin çözümünde 3 farklı yaklaşım ile birlikte kullanılan SVM' nin, çok sınıflı sınıflandırma problemini çözmek amacıyla geliştirilmiş olan SVM (Multi-Class) yaklaşımının, çoklu sınıflandırma probleminin ikili sınıflandırma problemine dönüştürülmesi yoluyla çözülen SVM (One-Vs-One) ve SVM (One-Vs-Rest) yaklaşımlarına göre daha kötü sonuç verdiği görülmüştür. Buradan, çoklu sınıflandırma problemlerinin SVM kullanılarak ikili sınıflandırma problemi biçiminde çözümlerinin daha etkili olduğu sonucu çıkarılabilir.

Öznitelik seçiminin, önceki deneylerde en iyi sınıflandırma başarımını sağlayan SVM sınıflandırıcıları üzerindeki etkisi gözlemlendiğinde, RFE algoritması ile her bir sınıf için daha özgün özniteliklerin seçildiği görülmüştür. Bir öznitelik kümesinden seçilecek optimal öznitelik sayısının belirlenemiyor olmasından ötürü,

farklı oranlarda seçilen öznitelikler için alınan sınıflandırma başarımları incelendiğinde, seçilen öznitelik sayısının artmasıyla sınıflandırma performanslarında düşüş olduğu görülmüştür. Bu durum, seçilen öznitelik sayısının artması ile gereksiz ve sınıf ayırımında herhangi bir yararı olmayan özniteliklerin seçilebileceği sonucunu ifade etmektedir. İki sınıflı öznitelik seçiminde, One-Vs-Rest yaklaşımı ile oluşturulan öznitelik kümelerinden seçilen özniteliklerin, One-Vs-One yaklaşımı ile oluşturulan öznitelik kümelerinden seçilen özniteliklere kıyasla daha etkili olduğu gözlenmiştir. Ayrıca, çok sınıflı öznitelik seçimi yapılarak elde edilen sınıflandırma başarımları incelendiğinde, çok sınıflı problemlerin önce iki sınıflı öznitelik kümelerine dönüştürülüp, ardından öznitelik seçimi yapıldıktan sonra birleştirilerek sınıflandırması ile daha doğru sonuçlar alınmıştır. Bununla birlikte boyut azalmasında, öznitelik seçimi ile kullanılan öznitelik dönüşüm tekniği arasında bir karşılaştırma yapılırsa, sonuçlar öznitelik seçimi ile daha iyi başarımlar alındığını göstermektedir.

Son deneyde, 6 farklı yüz ifadesine ait Gabor öznitelik kümesi AdaBoost ile sınıflandırılmış ve en iyi sınıflandırma başarımları elde edilmiştir. Sınıflandırma parametresi olarak tek değişkenin (iterasyon sayısı) olması nedeniyle AdaBoost algoritması, kullanılan diğer iki sınıflandırıcıya göre önemli bir üstünlük sağlamaktadır. Çünkü SVM sınıflandırıcısı için, RBF kernel parametreleri C ve γ için uygun değerler seçilmeli iken, k-NN sınıflandırıcısı içinde en iyi başarımları sağlayan k-en yakın komşuluk değeri bulunmalıdır. Ayrıca AdaBoost sınıflandırıcısının performansı, artan iterasyon sayısından etkilenmediğinden dolayı diğer iki sınıflandırıcıya göre daha gürbüz bir algoritmadır.

KAYNAKLAR

1. V. Bruce, *Advanced Robotics*, **8**, 341(1993).
2. F. Hara and H. Kobayashi, *Advanced Robotics*, **11**, 585(1997).
3. A. Takeuchi and K. Nagao, *Proceedings of the INTERCHI '93 conference on Human factors in computing systems*, **1**, 187(1993).
4. L.S. Chen, T.S. Huang, T. Miyasato, and R. Nakatsu, *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition (FG'98)*, IEEE, **1**, 366(1998).
5. L.C. De Silva, T. Miyasato, and R. Nakatsu, *Information, Comm. and Signal Processing Conf.*, **1**, 397(1997).
6. R. Nakatsu, *Proc. IEEE*, **86**, 825(1998).
7. G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **21**, 974(1999).
8. E. Boyle, A.H. Anderson, and A. Newlands, *Language and Speech*, **37**, 1(1994).
9. G.M. Stephenson, K. Ayling and D.R. Rutter, *Britain J. Social Clinical Psychology*, **15**, 113(1976).
10. A. Mehrabian, *Psychology Today*, **2**, 53(1968).
11. C. C. Chibelushi and F. Bourel, *Facial Expression Recognition: A Brief Tutorial Overview*, School of Computing, Staffordshire University, Staffordshire, 2002.
12. T. Güneş ve E. Polat, *Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi*, **24**, 7(2009).
13. A. Van Dam, *IEEE Computer Graphics and Applications*, **20**, 50(2000).
14. B. Fasel and J. Luetttin, *Pattern Recognition*, **36**, 259(2003).

15. T. Kanade, J. F. Cohn and Y. Tian, 4th Int. Conf. on Automatic Face and Gesture Recognition, **1**, 46(2000).
16. Z. L. Stan and K. J. Anil, Handbook of Face Recognition, Springer Inc., New York, 2005.
17. P. Ekman and W. V. Friesen, Journal of Personality and Social Psychology, **17**, 124(1971).
18. R. Schapire, Foundations of Machine Learning, Lecture Notes COS 511, 2003.
19. N. J. Nilsson, Introduction to Machine Learning, Robotics Laboratory, Department of Computer Science, Stanford University, 2005.
20. M. Pantic, and L. J. M. Rothkrantz, IEEE Trans. on PAMI, **22**, 1424(2000).
21. P. Viola and M. Jones. Robust real-time object detection, Technical Report CRL 20001/01, Cambridge Research Laboratory, Cambridge Inc., 2001.
22. M. Yang, D. Kriegman, and N. Ahuja, IEEE Trans. on Pattern Analysis and Machine Intell., **24**, 34(2002).
23. C. Darwin, The Expression of the Emotions in Man and Animals, J. Murray Inc., London, 1872.
24. C. Izard, L. Dougherty, and E. A. Hembree. A System for Identifying Affect Expressions by Holistic Judgments, In Unpublished Manuscript, University of Delaware, 1983.
25. M. Suwa, N. Sugie, and K. Fujimora, Proc. Int'l Joint Conf. Pattern Recognition, **41**, 408(1978).
26. S. Marcelja, Journal of the Optical Society of America, **70**, 1297(1980).
27. J. G. Daugman, Journal of the Optical Society of America, **2**, 1160(1985).
28. K. Mase, IEICE Transactions, **74**, 3474(1991).
29. M.J. Black and Y. Yacoob, Fifth International Conference on Computer Vision (ICCV'95), Cambridge, **1**, 374(1995).
30. M. Rosenblum, Y. Yacoob, and L. S. Davis, IEEE Transactions on Neural Network, **7**, 1121(1996).

31. I.A. Essa and A.P. Pentland, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 757(1997).
32. T. Otsuka, and J. Ohya, *IEEE Proc. International Conf. on FG*, **1**, 442(1998).
33. A. Martinez, In *IEEE Workshop on Content-based Access of Images and Video Libraries*, **1**, 35(1999).
34. N. Oliver, A. Pentland, and F. B´erard, *Pattern Recognition*, **33**, 1369(2000).
35. I. Kotsia and I. Pitas, , *IEEE International Conference on Image Processing*, **2**, 966(2005).
36. G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind and J. Movellan, *Image and Vision Computing*, **24**, 615(2006).
37. Y. L. Tian, T. Kanade, and J. F. Cohn, Facial Expression Analysis. In: Li, S.Z., Jain, A.K. (eds.) *Handbook of Face Recognition*, Springer Inc., New York, 2005.
38. Z. Zhang, M. Lyons, M. Schuster, and S. Akamatsu, In *International Workshop on Automatic Face and Gesture Recognition*, **1**, 454(1998).
39. D. Gabor, *J. Inst.Electr. Eng.*, **93**, 429(1946).
40. A. Jain, and F. Farrokhnia, *Pattern Recogn.*, **24**, 1167(1991).
41. Y. Hammamoto, *Intelligent Biometric Techniques in Fingerprint and Face Recognition* (L.C. Jain, U. Halici et al., Eds.), CRC Press, 1999.
42. C. Liu, and H. Wechsler, *IEEE Int. Conf. Computer Vision*, **2**, 270(2001).
43. J. G. Daugman, *Biometrics: Personal Identification in Networked Society* (A.K. Jain, R. Bolle and S. Pankanti, Eds.), Kluwer Academic Publishers, 1998.
44. D. H. Hubel, and T. N. Wiesel, *Royal Soc. B (London)*, **198**, 1(1978).
45. http://en.wikipedia.org/wiki/Gabor_filter
46. David J. Field, *Optical Society of America A*, **4**, 2379(1987).
47. D. Dunn, W. E. Higgins, and J. Wakeley, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**, 130(1994).

48. R. L. De Valois, and K. K. De Valois, *Spatial Vision*, Oxford Univ. Press, New York, 1988.
49. J. G. Daugman, *IEEE Trans. Acoustics, Speech, and Signal Processing*, **36** 169(1988).
50. M. Zhou, and H. Wei, 18th International Conference on Pattern Recognition (ICPR'06), **1**, 404(2006).
51. D. Clausi, and M. Jernigan, *Pattern Recognition*, **33**, 1835(2000).
52. A. K. Jain, R. P. W. Duin, and J. Mao, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **22**, 4(2000).
53. R. Gilad-Bachrach, A. Navot, N. Tishby, 21st International Conference on Machine Learning (ICML04), **69**, 43(2004).
54. P. Cunningham, M. Cord, P. Cunningham (eds). *Machine Learning Techniques for Multimedia: Case Studies on Organization and Retrieval*, Springer Inc., Berlin, 2008.
55. A. M. Martinez, and A. C. Kak, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **23**, 228(2001).
56. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Wiley, New York, 2001.
57. K. Özdamar, *Paket Programlar İle İstatistiksel Veri Analizi (Çok Değişkenli Analizler)-2*, Kaan Kitabevi, 1999.
58. H. Liu, and H. Motoda, *Computational Methods of Feature Selection*, Chapman & Hall/Crc Data Mining and Knowledge Discovery Series, 2008.
59. T. Gunes, and E. Polat, 23rd Symposium on Computer and Information Sciences-Abstract Book, **1**, 54(2008).
60. R. O. Duda, and P. E. Hart, *Pattern classification and scene analysis*, Wiley, 1973.
61. H. Frohlich, *Feature Selection for Support Vector Machines by Means of Genetic Algorithms*, Diploma Thesis in Computer Science, 2002.

62. J. Weston, A. Elisseeff, B. Scholkopf, and M. Tipping, *Journal of Machine Learning Research*, **3**, 1439(2003).
63. M. Zaffalon and M. Hutter, *Proceedings of the 18th International Conference on Uncertainty in Artificial Intelligence*, **1**, 577(2002).
64. http://en.wikipedia.org/wiki/Mutual_information
65. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, *Machine Learning*, **46**, 389(2002).
66. A. Rakotomamonjy, *The Journal of Machine Learning Research*, **3**, 1357(2003).
67. R. El Kaliouby, and P. Michel, *Fifth International Conference on Multimodal Interfaces*, **1**, 258(2003).
68. I. Cohen, N. Sebe, L. Chen, A. Garg, and T. Huang, *Computer Vision and Image Understanding*, **91**, 160(2003).
69. N. Sebe, M. Lew, I. Cohen, Y. Sun, T. Gevers, and T. Huang, *Image and Vision Computing*, **25**, 1856(2007).
70. J. J. Lien, T. Kanade, J. F. Cohn, C-C. Li, *Proc. Third IEEE Int. Conf. Automatic Face and Gesture Recognition*, **1**, 390(1998)
71. <http://www.cs.utexas.edu/users/hyukcho/bioinformaticsProject.html>
72. [http://en.wikipedia.org/wiki/Nearest_neighbor_\(pattern_recognition\)](http://en.wikipedia.org/wiki/Nearest_neighbor_(pattern_recognition))
73. B. E. Boser, I. M. Guyon, and V. N. Vapnik, *Proceedings of the 5th annual ACM workshop on Computational Learning Theory*, **1**, 144(1992).
74. V. N. Vapnik, *Statistical Learning Theory*, John Wiley and Sons Inc., New York, 1998.
75. B. Scholkopf, and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, 2002.
76. J. Shawe-Taylor, and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.

77. B. Scholkopf, C. Burges and V. Vapnik, Proceedings of the First International Conference on Knowledge Discovery & Data Mining, AAAI Press, **1**, 252(1995).
78. T. Joachims, Learning to Classify Text Using Support Vector Machines, Kluwer Academic Publishers, 2002.
79. O. Deniz, M. Castrillon and M. Hernandez, Pattern Recognition Lett. **24**, 2153(2003).
80. P. Shaoning, K. Daijin and Y. B. Sung, Pattern Recognition Lett. **24**, 215(2003).
81. H. Kashima, K. Tsuda, and A. Inokuchi, In B. Schölkopf, K. Tsuda, and J. P. Vert, editors, Kernel Methods in Computational Biology, MIT Press, 2004.
82. S. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, 2004.
83. J. Platt, In B. Schölkopf, C. Burges, and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning, MIT Press, Cambridge, MA, USA, 1999.
84. B. Vanschoenwinkel, and B. Manderick, Deterministic and Statistical Methods in Machine Learning, **3635**, 256(2005).
85. R.E. Schapire. The boosting approach to machine learning: an overview. Princeton University, New Jersey, 2002.
86. Y. Freund, *Information and Computation*, **121**, 256(1995).
87. Y. Freund, and R. Schapire, Journal of Computer and System Sciences, **55**, 119(1997).
88. R. Schapire, and Y. Singer, *Machine Learning*, **39**, 297(1999).
89. J. Weston, A. Elisseeff, G. Bakır and F. Sinz, The Spider, <http://www.kyb.tuebingen.mpg.de/bs/people/spider>, 2004.
90. Jia Sheng, A Study of AdaBoost in 3 D Gesture Recognition, Department of Computer Science. University of Toronto. Toronto, 2005.

91. W. S. Sarle, Neural Network FAQ, Periodic posting to the Usenet news group comp.ai.neural-nets, 1997.
92. C. W. Hsu, C. C. Chang and C. J. Lin, A practical guide to support vector classification, Department of Computer Science and Information Engineering, National Taiwan University, 2003.
93. R. Rifkin and A. Klautau, *Journal of Machine Learning Research*, **5**, 141(2004).
94. C. H Yeang, S. Ramaswamy, P. Tamayo, S. Mukherjee, R. Rifkin, M. Angelo, M. Reich, E. Lander, J. Mesirov and T. Golub, *In Proceedings, 11th international conference on intelligent systems for molecular biology*, **17**, 316(2001).
95. A. Ben-Hur and D. Brutlag, *Studies in Fuzziness and Soft Computing*, Springer Berlin / Heidelberg Inc., 2006.