

**TÜRK HAVA KURUMU ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Göz Hareketlerine Dayalı Aktivite Sınıflandırma

YÜKSEK LİSANS TEZİ

Çağrı ŞENER

Elektrik ve Bilgisayar Mühendisliği Anabilim Dalı

Elektrik ve Bilgisayar Mühendisliği Programı

Ocak, 2017

**TÜRK HAVA KURUMU ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

Göz Hareketlerine Dayalı Aktivite Sınıflandırma



YÜKSEK LİSANS TEZİ

**Çağrı Şener
1406010018**

Elektrik ve Bilgisayar Mühendisliği Anabilim Dalı

Elektrik ve Bilgisayar Mühendisliği Programı

Tez Danışmanı: Yrd. Doç. Dr. Engin Demir

Türk Hava Kurumu Üniversitesi Fen Bilimleri Enstitüsü'nün 1406010018 numaralı Yüksek Lisans öğrencisi, Çağrı Şener, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “Göz Hareketlerine Dayalı Aktivite Sınıflandırma” başlıklı tezini, aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı: Yrd. Doç. Dr. Engin DEMİR

Çankaya Üniversitesi



Eş Danışman: Prof. Dr. Hasan OĞUL

Başkent Üniversitesi



Jüri Üyeleri: Yrd. Doç. Dr. Engin DEMİR

Çankaya Üniversitesi

Yrd. Doç. Dr. Erhan Mengüşoğlu

Türk Hava Kurumu Üniversitesi

Yrd. Doç. Dr. Tansel Dökeroğlu

Türk Hava Kurumu Üniversitesi



Tez Savunma Tarihi: 27 Ocak 2017

TÜRK HAVA KURUMU ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ'NE

Yüksek Lisans Tezi olarak sunduğum, “Göz Hareketlerine Dayalı Aktivite Sınıflandırma” adlı çalışmamın, tarafımdan akademik etik ve kurallara aykırı düşecek bir yardıma başvurmaksızın yazıldığını ve yararlandığım kaynakların kaynakçada gösterilenlerden oluştuğunu, bunlara atıf yapılarak yararlanılmış olduğunu belirtir ve bunu onurumla doğrularım.



Tarih: 23.02.2017

Ad Soyad: GÖĞRİ SENER

İmza:

TEŐEKKÖRLER

Yüksek Lisans tez alıŐma sürecinde beni yönlendiren, karşılaŐtıđım zorlukları bilgileri ve tecrübeleri ile aŐmamda yardımcı olan desteđini ve yardımını hiçbir zaman esirgemeyen tez danışmanım deđerli Yrd. Do. Dr. Engin DEMİR'e ve eŐ danışmanım Prof. Dr. Hasan OĐUL' a teşekkürlerimi sunarım.

Hayatım boyunca her zaman yanımda olan, maddi ve manevi desteklerini hiçbir zaman esirgemeyen aileme teşekkürlerimi, sevgi ve saygılarımı sunarım.

Ocak, 2017

ađrı Őener

İÇİNDEKİLER

TEŞEKKÜRLER	iii
İÇİNDEKİLER	iv
TABLolar	v
ŞEKİLLER	vi
ÖZET	vii
ABSTRACT	viii
BİRİNCİ BÖLÜM	1
GİRİŞ.....	1
1.1 Motivasyon	1
1.2 Temel Bilgiler.....	3
1.3 Tezin Katkıları.....	7
İKİNCİ BÖLÜM	8
LİTERATÜR.....	8
2.1 Göz Bebeği Bulma	8
2.1.1 Donanımsal Yardımla Göz Bulma Teknikleri.....	8
2.1.2 Düşük Çözünürlüklü İmgelerde Göz Bulma	11
2.2 Göz Hareketlerinin Yapılan Aktivitelerle İlişkilendirilmesi	14
2.3 Sıfır Hipotez Testi	15
ÜÇÜNCÜ BÖLÜM	17
MATERYAL ve YÖNTEM	17
3.1 Göz Bebeği Koordinatlarının Belirlenmesi	18
3.1.1 Kenar Filtreleri	19
3.1.2 En Uygun Elipsi Seçme.....	20
3.2 Özellik Çıkarımı	22
3.3 Özellik Seçimi	24
3.4 Sınıflandırma Algoritmaları	24
3.4.1 Bayesian Ağlar (Bayesian Networks)	25
3.4.2 Destek Vektör Makinası (Support Vector Machine).....	26
3.4.3 k- En yakın komşuluk Algoritması (k-NN).....	28
3.4.4 Rastgele Orman (Random Forests or Random Decision Trees)	29
DÖRDÜNCÜ BÖLÜM	30
DENEYSEL ÇALIŞMALAR	30
4.1 Veri seti	30
4.2 Değerlendirme Yöntemleri.....	31
4.3 Deneysel Sonuçlar	33
BEŞİNCİ BÖLÜM	44
TARTIŞMA VE ÖNERİLER	44
EKLER	52
ÖZGEÇMİŞ	61

TABLULAR

Tablo 3.1 BA Öğrenme algoritması için farklı durumlarda kullanılan yöntemler [21].....	26
Tablo 4.1 Bayesian Ağlar yönteminin sonuçları	35
Tablo 4.2 Bayesian Ağlar yönteminin Karışıklık (Confusion) matrisi	35
Tablo 4.3 SVM yönteminin sonuçları	36
Tablo 4.4 SVM yönteminin Karışıklık (Confusion) matrisi	36
Tablo 4.5 k-NN yönteminin sonuçları.....	37
Tablo 4.6 k-NN yönteminin Karışıklık (Confusion) matrisi	37
Tablo 4.7 Rastgele Orman yönteminin sonuçları.....	38
Tablo 4.8 Rastgele Orman yönteminin Karışıklık (Confusion) matrisi	38
Tablo 4.9 Tüm yöntemlerin doğruluk yüzdeleri	39
Tablo 4.10 Özellik seçim yöntemleri ile seçilen en etkili özellikler	40
Tablo 4.11 Algoritmaların özellik seçim yöntemlerinden sonraki sonuçları	42
Tablo 4.12 Rastgele orman algoritması için hipotez kabul testi sonuçları.....	43

ŞEKİLLER

Şekil 1.1 Gürültülü (sağda) ve gürültüsü giderilmiş (solda) imge	3
Şekil 1.2 Satır ve Sutunlarına ayrılmış resim	4
Şekil 1.3 Renk karışımları	4
Şekil 1.4 Orjinal imge ve histogramı	5
Şekil 1.5 Orijinal imge ve Median 5x5 filtresi uygulanmış imge [47]	5
Şekil 1.6 Orijinal resim ve kenarları belirlenmiş türevleri [44]	6
Şekil 2.1 Göz bebeğini takip etmek için kullanılan kontakt lens örneği [3]	9
Şekil 2.2 Göz bebeği takip Cihazı [5]	10
Şekil 2.3 Kızılötesi kameranın göz üzerindeki etkisi [6]	10
Şekil 2.4 Kızılötesi kamera ile göz bebeği takibi [7]	11
Şekil 2.5 İmge ve Isophote haritaları	12
Şekil 2.6 Yüzün özellik noktaları	13
Şekil 2.7 Göz bulma algoritmaları kıyaslaması [16]	14
Şekil 3.1 Geliştirilen sistemin genel görünümü	17
Şekil 3.2 Göz bebeği bulma algoritmasının adımları [35]	19
Şekil 3.3 Kenar manipülasyonu için morfolojik desenler [35]	20
Şekil 3.4 Elipsin Oluşturulması [35]	20
Şekil 3.5 Farklı göz bebeği bulma algoritmalarının başarı grafiği	21
Şekil 3.6 Sınıfların birbiriyle ilişkisi [21]	26
Şekil 3.7 Verinin belirgin şekilde gruplara ayrılması [39]	27
Şekil 3.8 En yakın komşuların belirlenmesi[46]	28
Şekil 4.1 Kayıt alımından örnek bir kesit	31
Şekil 4.2 Tüm denek göz bebeklerinin x1 ve x2 koordinatlarının minimum değerlerinin grafiği	34
Şekil 4.3 Tüm denek göz bebeklerinin y1 ve y2 koordinatlarının maksimum değerlerinin grafiği	34
Şekil 4.4 Farklı p değerleri için metotların doğruluk yüzdeleri	39

ÖZET

Göz Hareketlerine Dayalı Aktivite Sınıflandırma

ŞENER, Çağrı

Yüksek Lisans, Elektrik ve Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Yrd. Doç. Dr. Engin DEMİR

Eş Danışman: Prof. Dr. Hasan OĞUL

Ocak-2017, 61 sayfa

Bu çalışmada bilgisayar karşısında düşük çözünürlüklü kamera ile kayıt altına alınan kişinin göz hareketlerine bakılarak, belirlenen üç farklı aktivitenin (okuma, bir video izlenme, oyun oynama) farklı sınıflandırma algoritmaları ile tespiti hedeflenmektedir. Çalışmanın temel motivasyonu veri modeli oluşturulamayan hareketlerin, istatistiksel verilerin yardımı ile özellik uzayında temsil edilmesi ve sınıflandırılmasıdır.

Bilgisayar karşısında üç farklı aktiviteyi gerçekleştiren deneklerden alınan kayıtların sağ ve sol göz koordinatları çıkarılmış, bu koordinatlardan istatistiksel veriler elde edilmiştir. Elde edilen veriler Bayesian Ağlar, Destek Vektör Makinaları (Support Vector Machine-SVM), k-En Yakın Komşu (K-Nearest Neighbour-k-NN) ve Rastgele Orman (Random Forest) olmak üzere dört farklı denetimli sınıflandırma algoritması ile sınıflandırılmıştır. Algoritmaların performansları çeşitli sınıflandırma ölçütleri kullanılarak test edilmiştir. Elde edilen sonuçlara göre aktivite sınıflandırmasında SVM başarısız olmuş, Bayesian Ağlar ve k-NN algoritmaları iyi sonuçlar verirken en başarılı sonuç Rastgele Orman algoritması ile elde edilmiştir. Sonuçların istatistiksel olarak anlamlı olduğu ikili t-test yöntemi ile de doğrulanmıştır.

Anahtar Kelimeler: Aktivite tespiti ve sınıflandırması, İnsan bilgisayar etkileşimi, Bayesian Ağlar, Destek Vektör Makinası, SVM, k En Yakın Komşu Algoritması, k-NN, Rastgele Orman.

ABSTRACT

Activity Classification based on Eye Movement

ŞENER, Çağrı

Master, Department of Electrical and Computer Engineering

Thesis Supervisor: Assoc. Prof. Dr. Engin DEMİR

Co-Advisor: Prof. Dr. Hasan OĞUL

January–2017, 61 pages

This study is a comparative study which provides classifying human activities that are performed in front of the computer. It is aimed to detect activities (reading a textual document, watching a video, playing a computer game) with different classification techniques by using eye movements of a person recorded with the low resolution camera. The main contribution of the study is representing activities that do not have data model in the feature space and classifying these activities through the statistical data.

Statistical data are extracted from right and left eye coordinates of persons who perform three specified activities. The classification techniques implemented and compared in this study are; Bayesian Networks, Support Vector Machine, k Nearest Neighbor and Random Forest. Performance comparison of these techniques is provided in terms of different evaluation metrics. According to results, Bayesian and k-NN have good classification rates, while SVM failed to classify activities. The results also indicate that Random Forest has become more successful in classifying activities than other methods. Statistical significances of the results are validated by using a paired t-test.

Keywords: Activity recognition and classification, Human Computer Interaction, Bayesian Networks, Support Vector Machine, SVM, k-Nearest neighbor, k-NN, Random forest.

BİRİNCİ BÖLÜM

GİRİŞ

Bu çalışmada bilgisayar karşısında düşük çözünürlüklü kamera ile kayıt altına alınan kişinin ekrana baktığı andaki göz hareketlerine dayalı olarak, yaptığı aktivitenin farklı sınıflandırma algoritmaları ile tespiti hedeflenmektedir. Sınıflandırılacak eylemler okuma, video izleme ve oyun oynama olarak belirlenmiştir. Bu eylemlerden okuma sadece metinsel içerikten oluşan bir dökümanın okunması, video izleme altyazı gibi metinsel içeriğin olmadığı hareketli görsel içeriğin izlenmesi eylemidir. Oyun oynama ise belirli bir amaç için ekrandaki görüntü oluşumunun denek tarafından etkilendiği bir eylemdir.

Bu bölümün devamında çalışmanın motivasyonu, temel bilgileri, akademik katkıları anlatılacaktır.

1.1 Motivasyon

Son yıllarda internet kullanan kişi sayısı ve internetin kullanım alanları her geçen gün artmaktadır. Kişiler kendi aralarındaki iletişimden alışveriş, bankacılık işlemlerine kadar hatta eğitim hizmetini de internet ortamından uzaktan eğitimle sağlamaktadır. Kişilerin sanal ortamı bu kadar çok kullanması ve sanal ortamda faaliyet gösteren firmaların bu kişilere bir şekilde ulaşip hizmet veya ürün satmaya çalışması önemli bir çalışma alanı oluşturmaktadır. İnsan davranışlarına göre kişiye özel hizmet ve ürün sunmaya çalışan bu firmalar farklı teknikler kullanarak bu isteklerini gerçekleştirmektedir.

İnsanların en çok hizmet aldığı alan olan uzaktan eğitim alanı günümüzde hızla yaygınlaşmakta ve farklı sebeplerden dolayı daha fazla kişi için tercih sebebi olmaktadır. Sanal ortamda faaliyet gösteren diğer şirketler gibi uzaktan eğitim veren kurumlar da kişiye özel ve daha kaliteli eğitim hizmeti vermek için çalışmalar yapmaktadır. Kişilerin eğitimden daha verimli bir şekilde yararlanabilmesi ve eğitimin başarı yüzdesinin artırılması için bilgisayar yardımıyla kişinin dersi takip edip etmediğinin kontrolü yapılabilmektedir. Ekranın diğer ucunda bulunan eğitmenin bunu takip edebilmesi ya da bu kontrolü sürekli yapması mümkün değildir. Yapılan bu çalışma bu soruna çözüm üretmek amacıyla gerçekleştirilmiştir. Bilgisayar

karşısında bulunan kişinin eğitim esnasında eğitmenin gösterdiği bir videoyu izleyip izlemediğini ya da bir dokümanı okuyup okumadığını belirlemek eğitmenin kişiyi takip etmesine olanak verecektir. Bu çalışmanın amacı bilgisayar karşısındaki kişinin göz bebeklerini bir kamera yardımıyla belirleyerek takip etmek ve gözlerin belirli bir zaman aralığındaki hareketlerini öğrenme algoritmaları kullanarak sınıflandırmaktır. Bu çalışma sonucunda bilgisayar karşısındaki kişinin video izlemek, doküman okumak ve oyun oynamak olarak belirlenmiş üç farklı eylemden hangisini gerçekleştirdiğini belirleyen bir sistem tasarımı hedeflenmiştir.

Kişilerin göz bebeği koordinatlarından elde edilecek istatistiksel veriler ile kayıtlar özellik uzayında temsil edilmiştir. Kişilerin kayıtlarının özellik uzayında istatistiksel veriler ile temsil edilmesi çalışmanın temel motivasyonudur. Çünkü sınıflandırılan aktivitelerin matematiksel bir modeli yoktur, fakat bazı aktivitelerin modeli çıkarılabilmektedir. Örneğin bir yüzücünün sırt üstü yüzme esnasındaki kafa, kol ve bacak hareketlerinin bir sırası vardır. Uzunlardaki hareketlerde bu sıra sağlandığı zaman kişinin sırt üstü yüzdüğü anlaşılabilir. Fakat bir kişinin okuma sırasında ya da video izleme sırasındaki göz hareketlerinin belirgin bir modeli bulunmamaktadır. Çünkü kişinin okuduğu metnin dili ya da kişinin okuma tekniği bu modelin çıkarılmasını engelleyen faktörlerdir. Bu sebeple veri modeli oluşturmak yerine istatistiksel verilerin yardımı ile aktivite sınıflandırmasının yapılması hedeflenmektedir.

Yapılacak olan çalışmada kullanılacak materyalin çeşitli kısıtları olacaktır. Kullanılacak olan okuma dokümanının içinde sadece metin olacaktır. Bunun yanında kişilere izletilecek video içerisinde alt yazı ya da başka bir metin olmayacaktır. Oyun oynama esnasında kişi görsel ekranla klavye ya da fare kullanarak etkileşim halinde olacak ve oyunun bir metin içermemesine dikkat edilecektir. Bu şartlar sağlanarak kayıt altına alınacak kişilerin belirtilmiş aktivitelerden hangisini gerçekleştirdiğini belirleyebilmek için öğrenme algoritmaları yardımıyla aktiviteleri sınıflandırmaya yönelik bir çalışma olacaktır.

Bu çalışmada göz bebeklerini belirlemek için kamera dışında ek bir donanım kullanılmamıştır. Çalışmada kullanılmış olan verilerin hepsi aynı bilgisayar ile elde edilmiştir. Bu sayede farklı ekran büyüklüklerinden kaynaklanabilecek hatalar engellenmeye çalışılmıştır. Bilgisayar ekranı üzerinde takılı olan kamera ile yakalanan görüntüler üzerinde görüntü işleme algoritmaları kullanılarak göz bebekleri

belirlenmiştir. Göz bebeklerinin hareketleri takip edilmiş ve göz bebeklerinin yer değiştirme mesafeleri üzerinden çeşitli özellikler belirlenmiştir. Bu yaklaşımla oluşturulup etiketlenmiş verilerin öğrenme algoritmaları kullanılarak sınıflandırma işlemi yapılmıştır.

1.2 Temel Bilgiler

Bu bölümde tez kapsamında kullanılan görüntü işleme algoritmalarındaki adımların daha iyi anlaşılabilmesi için görüntü işleme ile ilgili temel bilgiler verilecektir.

İmgeler genellikle analog ortamdan dijital ortama geçirildiğinde Şekil 1.1’de görüldüğü gibi çeşitli bozukluklar (noise) içerir. İmge işleme tekniklerinin amacı girdi olarak alınan görüntü üzerinde bulunan hataya göre gürültü temizlemek (filtering), parlaklık (brightness) veya koyuluk (contrast) ayarlama, görüntü keskinleştirmek gibi farklı yöntemler uygulanarak yeni bir görüntünün ortaya çıkarılmasıdır.

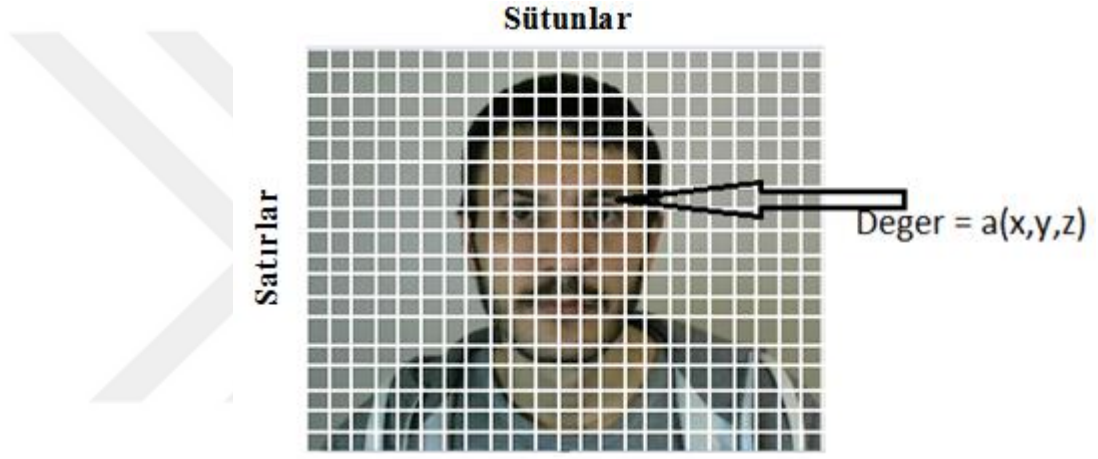


Şekil 1.1 Gürültülü (sağda) ve gürültüsü giderilmiş (solda) imge

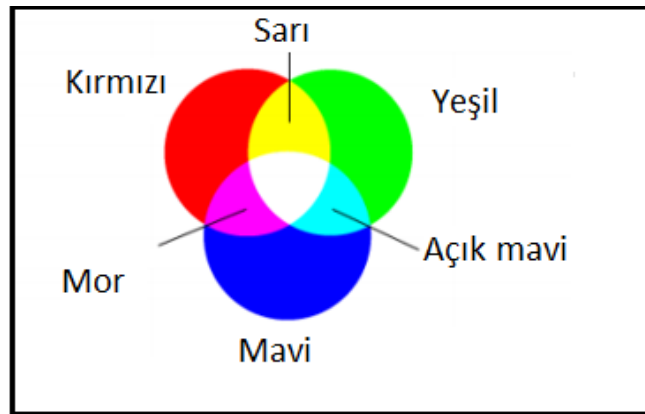
İmge belirli sayıda satır ve sütünün bir araya gelmesiyle oluşmuş, üzerinde sayısal değerleri barındıran bir matristir. Satır ve sütünün birleştiği yer ‘piksel’ olarak adlandırılır. Piksel aldığı değere göre bir renge karşılık gelir. Şekil 1.2’ de görülen imge 16 satır ve 16 sütundan oluşmuş bir imge örneğidir. Bu imge üzerinde bulunan 256 piksel bir araya gelerek bu görüntüyü oluşturmuştur.

İmgelerin arka arkaya kaydedilmesiyle oluşturulan dijital veriye video denir. Videoyu oluşturan her bir imge çerçeve (frame) olarak adlandırılır. Göz bir saniyede 24 çerçeve değişiminden daha fazlasını algılayamadığı için video üzerindeki imge geçişleri fark edilmeden akıcı bir görüntü sağlanmış olur.

Piksellerin aldığı değer aralığına göre imge farklı isimlerle adlandırılır. 8 bit gri seviye resimlerde her piksel 256 farklı renk alabilir. Bu imgeler siyah-beyaz imge olarak adlandırılır. 32 bit imgeler renkli imgelerdir. Bu imgelerde her piksel kırmızı, yeşil ve mavi rengin aldıkları değerler ile bir rengi oluştururlar. Gerçek renkli imge Şekil 1.3' de görüldüğü gibi kırmızı, yeşil ve mavi renklerin belirli değerler ile karışmasıyla farklı renkte piksellerin bir araya gelmesiyle oluşmaktadır. Bu imgeler her bir renk kanalı için 256 farklı değer ve bunların bir araya gelmesiyle yaklaşık 16 milyon farklı renk içermektedir. Eğer pikseller sadece 0 ya da 1 değerini içerirse bu imgelere ikil imge denilmektedir.



Şekil 1.2 Satır ve Sütunlarına ayrılmış resim

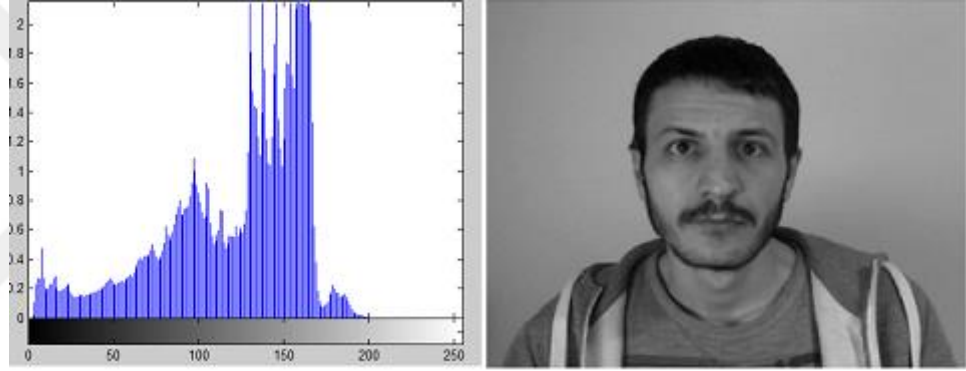


Şekil 1.3 Renk karışımları

İmge üzerinde bulunan piksellerdeki renk dağılımlarını gösteren grafiğe renk histogramı denir. Şekil 1.4' de görüldüğü gibi sağ tarafta orijinal imge sol tarafta bu

imge üzerindeki renklerin dağılımını gösteren histogram bulunur. Histogram grafiği sola dayalı çıkan imgeler koyu, sağa dayalı çıkan imgeler ise açık renkleri içermektedir.

Görüntü işlemede kullanılan tekniklerinden bir diğeri maskelemedir (filtreleme). Maskeleme görüntü üzerinde bulunan gürültülerin ortadan kaldırılması için kullanılmaktadır. İmge üzerinde 3x3, 5x5, 7x7 vs. gibi farklı büyüklükte ufak matrisler gezdirilerek piksel değerlerinin komşu piksel değerleri ile karıştırılıp yeniden elde edilmesi işlemidir. Bu işleme matris konvolusyon işlemi de denilmektedir. Maskelemede kullanılan maske türleri ve bu maskelerin büyüklük değerleri imge üzerindeki bulanma miktarını değiştirmektedir. Şekil 1.5' de bir imgenin orijinal hali ve Median 5x5 maskesi uygulandıktan sonraki durumu görülmektedir.



Şekil 1.4 Orjinal imge ve histogramı



a)Orjinal

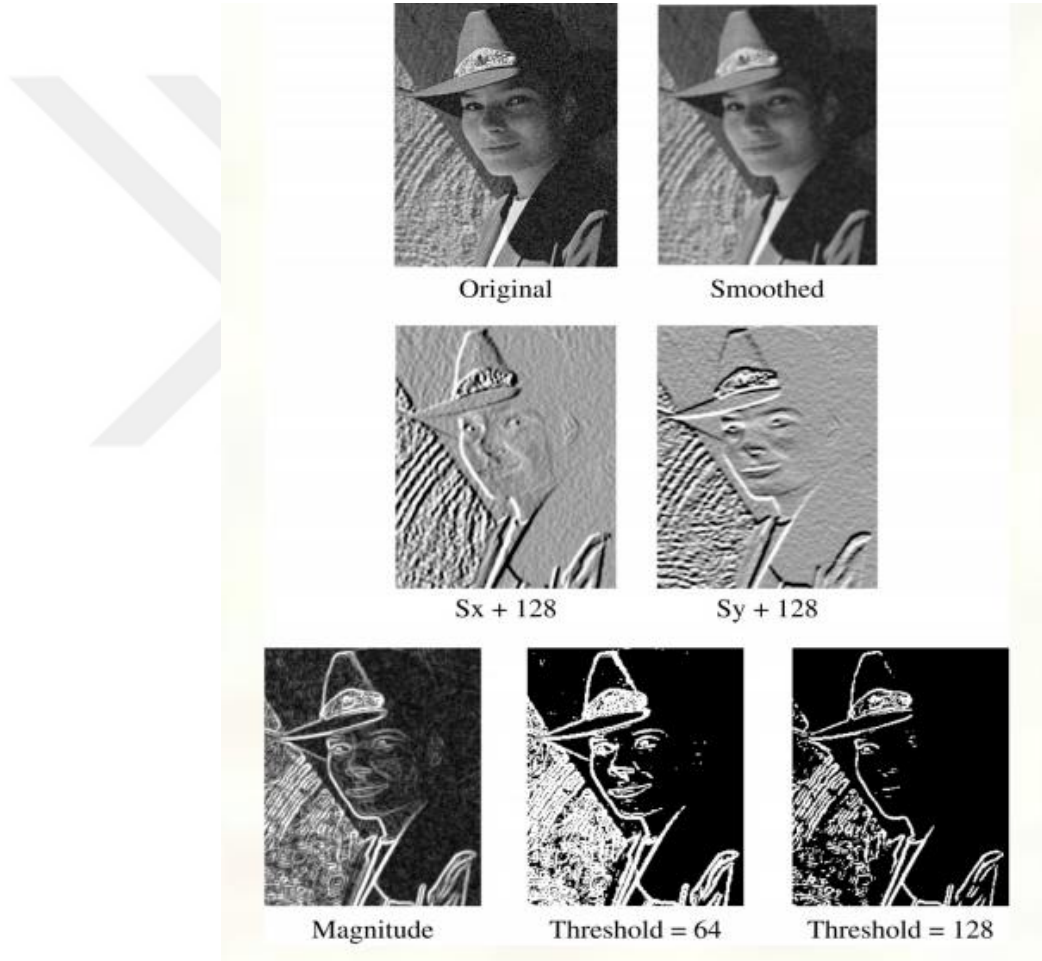
b)Median 5 x 5

Şekil 1.5 Orjinal imge ve Median 5x5 filtresi uygulanmış imge [47]

Görüntü işleme tekniklerinin kullanıldığı en önemli alanlardan birisi kenar belirlemedir (Edge Detection). İmge üzerinde bulunan nesnelere birbirinden ayırmak insanlar için çok kolay olmasına rağmen bunun bilgisayar tarafından belirlenmesi zor bir işlemdir. Bu sebeple imge üzerinde bulunan nesnelere arası geçişleri anlamak ve bu

kenarların yerlerin belirlemek için 3 adım uygulanır. Adımlardan ilki imge üzerinde bulunan gürültünün temizlenmesi için maskeleye tekniğinin kullanılmasıdır. İkinci adımda kenar ve kenar olmayan alanlar belirlenir. Son aşamada ise kenarların belirlenmesi yapılmaktadır.

Kenar belirlemek için farklı teknikler bulunmaktadır. Şekil 1.6' de görünen orijinal imgeye kenar belirleme tekniklerinden biri olan 'Sobel' algoritmasının farklı parametrelerle uygulanması ile ortaya çıkan farklı imgeler görülmektedir.



Şekil 1.6 Orijinal resim ve kenarları belirlenmiş türevleri [44]

1.3 Tezin Katkıları

Bu tez kapsamında göz hareketlerine dayalı aktivite sınıflandırılmasının mümkün olup olmadığı sorusuna cevap aranmıştır. Bu bağlamda aşağıdaki araştırma konuları üzerine çalışılmıştır.

- Düşük çözünürlüklü kamera ile alınan kişilerin yüz kayıtları üzerinde hangi algoritmanın daha düşük hata payı ile göz bebeği koordinatları belirlendiği incelenmiştir.
- Kişilerin ekrana bakarken oyun oynama, doküman okuma ve video izleme aktivitelerinin sadece birini gerçekleştirirken bu eylemlerin sınıflandırma algoritmaları yardımıyla sınıflandırılabilceği önermesine cevap aranmıştır.
- Sınıflandırma için bayesian ağlar, k-NN, SVM ve rastgele orman algoritmaları kullanılmıştır. Bu algoritmalarından rastgele orman algoritmasının en yüksek doğruluk payı ile aktiviteleri sınıflandırdığı belirlenmiştir.
- Sınıflandırma algoritmalarının kullandığı veri setlerinde konum bazlı özellikler ile sinyal bazlı bir özelliğin sınıflandırma için yeterli olduğu belirlenmiştir.
- Kullanılan özellikler üzerinde kullanılan özellik seçme yöntemlerinin (OneRAttribute, ReliefAttributeEval, CfsSubsetEval ve Principal) sınıflandırma sonuçları üzerinde kayda değer bir iyileştirme gerçekleştirmediği gözlenmiştir. (Özellik seçim yöntemleri üçüncü bölümde 3.3 başlığı altında anlatılmıştır.)
- Bu çalışma baz alınarak ileride yapılabilecek olası çalışmalar ile ilgili öneriler sunulmuştur.

Bu çalışma ile yukarıdaki bahsi geçen konular hakkında detaylı bilgi sonuçlar kısmında ifade edilmiştir.

İKİNCİ BÖLÜM

LİTERATÜR

Literatür özeti iki bölümden oluşmaktadır. Birinci bölümde göz bebeği belirlemek için kullanılan imge işleme algoritmalarından bahsedilecektir. İkinci bölümde ise göz hareketlerinin belirlenmesi, takip edilmesi ve gözün hareketlerinin bir aktiviteyle ilişkilendirilmesi üzerine yapılmış çalışmalar anlatılacaktır. Böylelikle tez kapsamında kullanılan algoritmaların daha iyi anlaşılacağı düşünülmektedir.

2.1 Göz Bebeği Bulma

İnsan-bilgisayar etkileşimi alanında geliştirilen görsel uygulamalarda göz bebeği bulma ve takip etme temel amaçtır. Bu amaç doğrultusunda göz bebeğini belirlemek için farklı yaklaşımlar kullanılmaktadır. Kullanılan tekniklerden bazılarında göz bebeği belirlemek için farklı donanımsal araçlardan faydalanırken bazılarında da sadece düşük çözünürlüklü imgelerden faydalanılmaktadır. Bu tekniklerden hangisinin kullanılacağı çalışmadaki göz bebeği bulma hassasiyetinin önemine göre tercih sebebi oluşturmaktadır. Genel olarak ek donanımsal araçların kullanıldığı teknikler daha hassas sonuçlara ulaşmaktadır. Fakat bu çözümler ek donanımsal cihazların ücretleri ile beraber daha pahalı çözümler sunmaktadır. Bu bilgiler doğrultusunda göz bebeği bulma teknikleri iki farklı başlık altında ele alınacaktır; donanımsal yardımla göz bulma teknikleri ve düşük çözünürlüklü imgeler üzerinde uygulanan göz bulma teknikleridir.

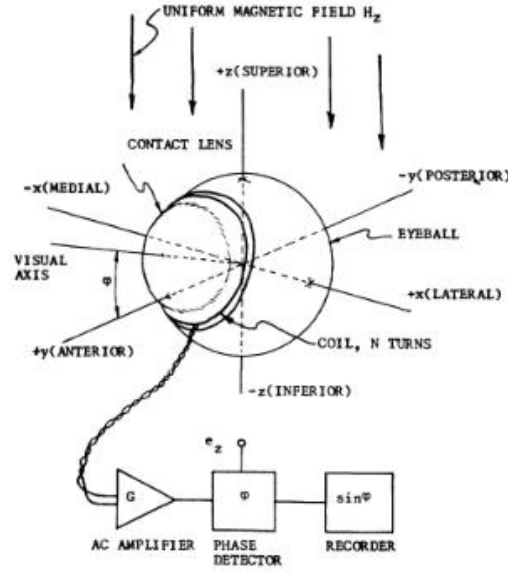
2.1.1 Donanımsal Yardımla Göz Bulma Teknikleri

Elektro okülografi (electro-oculography) ek donanımsal cihazlarla göz bulma tekniklerinden bir tanesidir [1]. Bu yöntem gözün elektriksel potansiyelinden yola çıkılarak geliştirilmiştir. Gözün ön tarafı pozitif, arka tarafı negatif olarak belirlenmiştir. Gözün ve göz bebeğinin ışık değişimlerine karşı verdiği tepkilerdeki dalgalanmalar cilde takılan gümüş veya altın elektrotlar yardımıyla tespit edilir. Bu teknik genel olarak sağlık alanında kullanılır [2].

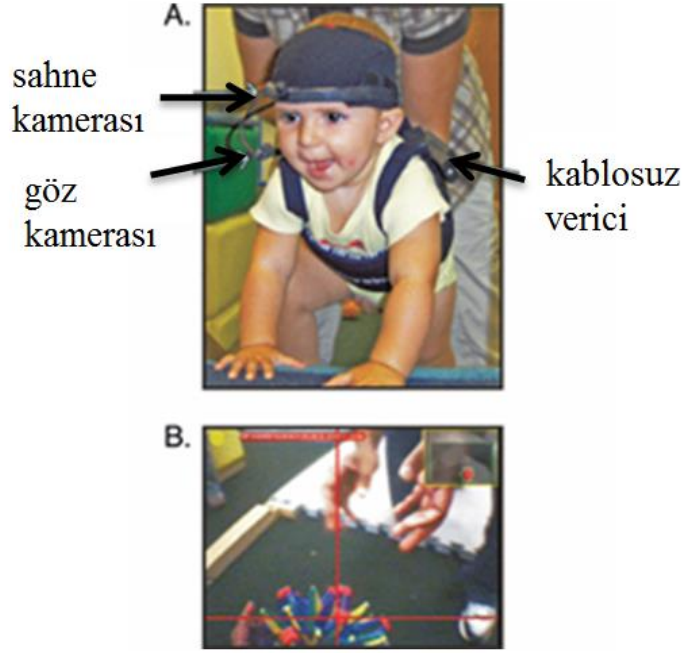
Sağlık alanında kullanılan tekniklerden bir diğeri kontakt lensler kullanarak göz bebeği belirlemektir. Bu teknikte Şekil 2.1' de görüldüğü gibi göze takılan kontakt

lensler göz ve göz bebeği hareketlerini takip eder. Topladığı sinyal bilgisini bir yükselticinin üzerinden faz detektörüne yollar. Bu veriler bir kayıt cihazı tarafından kayıt altına alınır. Bu işlemler yardımıyla kişinin göz ve göz bebeği koordinatları belirlenir [3].

Göz bulmak için kullanılan bir başka teknik kafaya monteli bir ekipmandan faydalanmaktadır [4]. Burada kullanıcının gözünü takip eden bir kamera ve çeşitli ışık kaynakları kullanılmaktadır. Göz bebeğini takip için yüzün farklı noktaları değişik miktarda aydınlatılarak göz bebeğinin yeri ve bakış açısı belirlenir. Kafaya monteli sistemler değişik teknikler kullanılarak Şekil 2.2' de görüldüğü gibi farklı alanlarda uygulanabilir [5]. Bu şekilde bebekler için geliştirilmiş bir donanım görülmektedir. Bebeğin göz bebeğini ışık kaynaklarıyla yardımıyla belirleyen bu donanım aynı zamanda bebeğin bakış açısını kayıt altına alınarak bebeğin odaklandığı cisim veya yer belirleniyor. Bu verilerin hepsi kablosuz bir verici yardımıyla ilgili kayıt cihazına yönlendiriliyor.

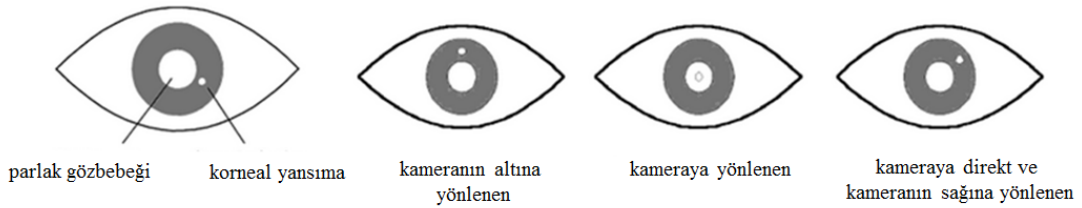


Şekil 2.1 Göz bebeğini takip etmek için kullanılan kontakt lens örneği [3]



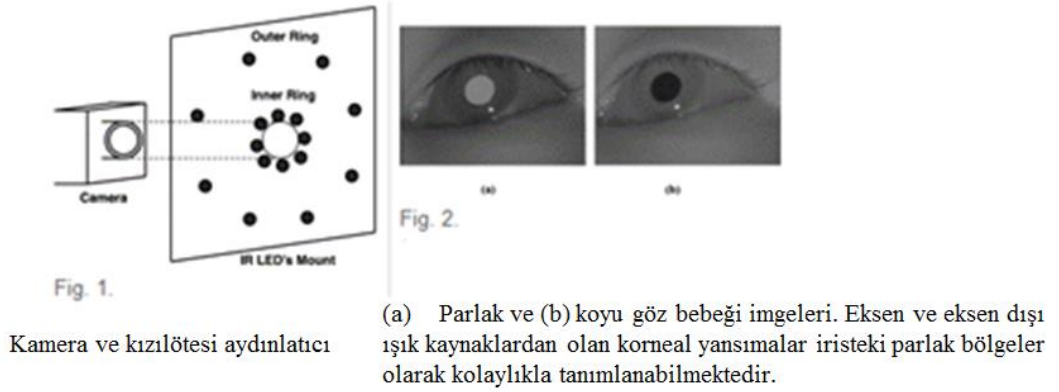
Şekil 2.2 Göz bebeği takip Cihazı [5]

Göz bulma tekniklerinden bir diğeri kızılötesi kameralar kullanmaktır [6]. Kızılötesi kameranın yüksek çözünürlükte kayıtlar alabilmesi imge üzerinde daha fazla detay ve daha az gürültü olmasını sağlamaktadır. Bu sayede imge işleme teknikleri daha net sonuçlar üretmesine olanak sağlamaktadır. Ancak bu yöntemin başarısının en temel sebebi ortamda parlak bir göz bebeği oluşturmaktır. Kamera üzerinde bulunan LED kaynaktan çıkan ışığın kornea üzerindeki yansımından faydalanılarak geliştirilen bu yöntem Şekil 2.3’ de görüldüğü gibi gözün üzerinde bir yansıma ve göz bebeğinin konumu hakkında bilgi verir. Bu teknik çok başarılı sonuçlar vermekle birlikte kullanıldığı ortamdaki ışık kaynaklarından etkilenmektedir. Bu yüzden açık havada çok başarılı değildir.



Şekil 2.3 Kızılötesi kameranın göz üzerindeki etkisi [6]

Kızılötesi kamera ile geliştirilen çözüm teknikleri farklı yaklaşımlar geliştirilerek de uygulanmaktadır. Bunlardan biri iki ışık kaynağı kullanılarak göz bebeğini tespit etmektir [7]. Bu yöntemde Şekil 2.4’ de görüldüğü gibi iki farklı ışık kaynağı yardımıyla göz bebeğini beyaz ve siyah gösteren iki tane imge elde edilir. Bu imge eşik değerlerine göre filtrelenerek aday göz bebeği noktası ve noktaları ortaya çıkarılır. Bu aday noktaların pozisyonuna göre aralarından en uygun olanı göz bebeği olarak belirlenir. Bu yöntemin dış ortamda başarısını artırmak için nesne tanıma metodu (SVM) ve Kalman filtreleme ile ortalama kaydırma teknikleriyle desteklenen bir çalışma mevcuttur [8].



Şekil 2.4 Kızılötesi kamera ile göz bebeği takibi [7]

2.1.2 Düşük Çözünürlüklü İmgelerde Göz Bulma

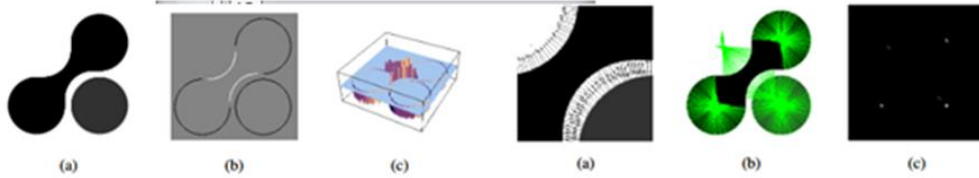
Bu başlık altında anlatılacak teknikler daha kolay uygulanabilmekle beraber başarı yüzdesi donanımsal tekniklere göre daha düşüktür. Burada yapılan bütün çalışmalarda kullanılan tüm yöntemlerde imge işleme teknikleri kullanılmıştır. Uygulanan işlemlerden sonra alınan sonuçlar belirli kurallar yardımıyla yorumlanarak nihai sonuca ulaşılmıştır.

Düşük çözünürlüklü imgelerde kullanılan göz belirleme algoritmaları genel olarak 3 farklı yöntemle tespit edilmeye çalışılmıştır. Bunlar özellik (feature) tabanlı yöntemler, model tabanlı yöntemler ve karma yöntemlerdir.

Asteriadis vd. [9] göz bulmak için geometrik özellikli bir yöntem kullanmıştır. Önce yüz bulma algoritmasıyla yüzün sınırlarını içeren bir dikdörtgen alan bulmuştur. Daha sonra “Canny” kenar bulma algoritmasını kullanarak bu alandaki kenar haritasını

çıkarmıştır. Buradaki her piksele en yakın kenarı gösteren bir vektör tanımlamıştır. Bu vektörlerin büyüklükleri ve eğimlerini hesaplayarak da gözün yerini belirlemiştir.

Roberto Valenti’ni [10] kullandığı “Isophote” eğrilik değerlendirme (Isophote Curvature Estimation) (IED) yaklaşımı karma bir göz bebeği bulma algoritmasıdır. Valenti çalışmasını gözlerin konumunun radyal olarak simetrik parlaklık desenlerine sahip olmasından ve merkez belirleme (center voting) yaklaşımından yola çıkarak yapmıştır. IED yaklaşımında resmin isophoteleri resmin eş yoğunlukta bulunan noktalarının birleşmesiyle meydana gelen eğriler oluşturur. Bu sebeple isophotelar birbirlerini hiçbir zaman kesmezler. Dahası isophotelerin şekli ışık değişiminden etkilenmezler. Bu özellikleri onları resmi bölümlendirmede ve resim üzerindeki nesnelerin belirlenmesinde başarılı kılmaktadır. Resim sınırlı büyüklükte, belirli bir koordinat sistemi ile tanımlanmış olduğundan resim üzerindeki maksimum yoğunluk değişimlerinin (gradient) hepsinin yönlerinin aynı olduğu kabul edilmiştir. Bu yaklaşımla belirlenen Şekil 2.5’de görülen isophote eğrilerinden olası göz bebekleri arasından doğru olanı tahmin etmeye çalışarak göz bebeklerini bulmuştur. Yaptığı çalışma sonucunda geliştirdiği algoritmayı BioID [48] veri kümesiyle denemiş ve başarılı sonuçlar elde etmiştir.



Şekil 1. Orijinal imge(a),
imgenin kenarlardaki isophote eğriliği (b),
ikincinin 3D çizimi (c)

Şekil 2. İmgenin kenarları altındaki gradient’ın yönü (a),
isophote merkezlerini işaret eden yerdeğişim vektörleri (b),
merkez haritası (c)

Şekil 2.5 İmge ve Isophote haritaları

Cristinacce [11] yüzün özelliklerini model bazlı bir şekilde belirleyen çalışma yapmıştır. Çalışmasını Şekil 2.6’ de görüldüğü gibi yüze ait 17 özelliği çıkararak bunları imgeler üzerinde işaretleyerek başlamıştır. İşaretlediği bu imgeleri kullanarak 17 tane özelliği belirleyici yapı oluşturmuştur. Oluşturduğu bu özellik belirleyicileri kullanarak resim üzerinde yüzü belirlediği alan içerisinde 17 farklı özelliğin her birini sarmalayan dikdörtgensel alanlar içerisinde özelliğin olası yerini belirlemektedir.

Daha sonra bu özellikleri histogram örnekleriyle tekrar kontrol ederek ve özelliğin x , y , z ve köşe yönlerini hesaplayarak gerçek yerini belirlemektedir. Yaptığı sistemi BioID [48] veri seti üzerinde denemiş ve başarılı sonuçlar elde etmiştir.



(a) Örnek eğitim imgesi



(b) Örnek eğitim eklemeleri özellik tespiti

Şekil 2.6 Yüzün özellik noktaları

Göz bebeğinin dairesel olmasından dolayı birçok kişi bu sorunu yüz üzerindeki dairesel nesnelere belirleyerek bulmaya çalışmıştır. İmge üzerindeki dairesel nesnelere belirlemek için en çok kullanılan yöntem Circular Hough Transform (CHT) [12] yöntemidir. Young [13] bu metodu kullanarak göz bebeğini bulmaya çalışmıştır. Bu metod yüksek çözünürlüklü resimlerde başarılı sonuçlar vermektedir. Smereka [14] ise dairesel nesnelere bulmak için CHT'yi değiştirmiş ve dairesel nesnelere yönlerini belirleyerek göz bebeğini bulmaya çalışmıştır.

Sewell ve Komogortsev [15] göz bebeği belirlemek için yapay sinir ağı tabanlı bir çözüm geliştirmişlerdir. Resim üzerindeki göz bebeği bölgelerini sisteme öğretmek başarılı bir sonuç elde etmişlerdir. Timm [16] göz bebeğinin kenarlara gittiği zaman bulmak sorun olduğu için göz bölgesinin yön vektörlerini kullanarak yeni bir metod geliştirmiştir. Tüm piksellerin normal yön ve uzaklık vektörlerinin iç çarpımlarından ağırlık fonksiyonlarının en büyüğünü alarak başarılı sonuçlar elde etmiştir.

D'Orazio [17] ise göz bebeği merkezini belirlemek için küçük matrislerle resim üzerinde gezinerek edindiği yeni özellikleri kullanmıştır. İmgelerin yön bilgileri ve piksellerin en tepe noktalarını kullanarak olası aday noktaları belirlemiştir. Wang [18]

sadece bir göz resmi üzerinde göz bebeği sınırlarını belirleyerek bu alan üzerine bir elips yerleştirerek göz bebeği merkezini bulmuştur.

2.2 Göz Hareketlerinin Yapılan Aktivitelerle İlişkilendirilmesi

Göz hareketlerini takip etmek için farklı yaklaşımlar kullanılarak çözümler geliştirilmiştir. Bazı çözümlerde Şekil 2.7’ de görüldüğü gibi belirli hata payları içerisinde yüzde yüze yakın oranda bir başarı elde edilmiştir. Fakat kişinin göz hareketlerinden bilgisayar başında hangi aktivite ile ilgilendiğini belirleyebilmek kolay bir işlem değildir. Bu sebeple yapılan aktiviteler genelde tek tek ele alınarak göz hareket desenleri oluşturmaya ve bu oluşan desenleri bir aktivite ile ilişkilendirilmeye çalışılmıştır.

Method	$e \leq 0.05$	$e \leq 0.10$	$e \leq 0.15$	$e \leq 0.20$	$e \leq 0.25$	Remarks
(Asadifard and Shanbezadeh, 2010)	47.0%	86.0%	89.0%	93.0%	96.0%	(*), (•)
(Kroon et al., 2008)	65.0%	87.0%	—	—	98.8%	
(Valenti and Gevers, 2008)	77.2%	82.1%	(86.2%)	(93.8%)	96.4%	MIC, (•)
(Valenti and Gevers, 2008)	84.1%	90.9%	(93.8%)	(97.0%)	98.5%	MIC+SIFT+kNN
(Türkan et al., 2007)	(18.6%)	73.7%	(94.2%)	(98.7%)	99.6%	
(Campadelli et al., 2006)	62.0%	85.2%	87.6%	91.6%	96.1%	
(Niu et al., 2006)	(75.0%)	93.0%	(95.8%)	(96.4%)	(97.0%)	
(Chen et al., 2006)	—	89.7%	—	—	95.7%	
(Asteriadis et al., 2006)	(44.0%)	81.7%	(92.6%)	(96.0%)	97.4%	(•)
(Hamouz et al., 2005)	(58.6%)	(75.0%)	(80.8%)	(87.6%)	(91.0%)	
(Zhou and Geng, 2004)	—	—	—	—	94.8%	(•)
(Cristinacce et al., 2004)	(57.0%)	96.0%	(96.5%)	(97.0%)	(97.1%)	
(Behnke, 2002)	(37.0%)	(86.0%)	(95.0%)	(97.5%)	(98.0%)	
(Jesorsky et al., 2001)	(38.0%)	(78.8%)	(84.7%)	(87.2%)	91.8%	
our method	82.5%	93.4%	95.2%	96.4%	98.0%	(•)

Şekil 2.7 Göz bulma algoritmaları kıyaslaması [16]

Yun [26] kişinin hangi aktivite ile uğraştığını belirleyebilmek için göz hareketlerine bakarak sınıflandırılmaya çalışmıştır. Çalışmasında yürüme, koşma, zıplama, ata binme, bisiklete binme, telefonla konuşma, fotoğraf çekme, bilgisayar kullanma, kitap okuma ve müzik aleti çalma olarak 10 tane aktivite belirlemiştir. Kişileri bu aktiviteleri gerçekleştirirken kayıt altına almıştır. Aynı zamanda göz hareketlerini de kayıt altına almıştır. Daha sonra iki tane destek vektör makinası kullanarak bunları eğitmiş ve sınıflandırma yapmaya çalışmıştır.

Yapılan bir başka çalışmada Rayner [27] okuma esnasında gözün nasıl hareket ettiğini analiz etmeye çalışmıştır. Kinsler [28] nota okurken gözün nasıl hareket ettiğini araştırmıştır. Yapılan çalışmalarda gözün sabitlenmesi, göz okuma yaparken yaptığı sıçramalardan yararlanılmıştır. Yapılan bu çalışmaların yanında Koornneef [32] okuma işinin göz hareket modelini çıkarmanın mümkün olup olmadığını

araştırmıştır. Çünkü okunan metnin yazıldığı dilin bile okuma davranışlarını değiştirdiğini göstermiştir. Bunun yanında kişilerin okuma alışkanlıklarının farklı olması bu işin yapılmasını zorlaştıran bir başka etken olarak ortaya çıkmıştır.

Buswell [29] ve Yarbus'un [30] insanların bir resme nasıl baktığını, onu nasıl yorumladığını sormuşlardır ve bu konu üzerine bir çalışma yapmışlardır. Yapılan bu çalışmadan yola çıkarak Castelano [31] ekranda bir video izlerken sahne geçişlerinin kişinin göz hareketlerini nasıl etkilediği üzerine bir çalışma yapmıştır. Yaptığı çalışmada gözün iki farklı davranışını belirlemiştir. Bunlardan birisi gözün ekran üzerinde bir şey araması esnasında yaptığı hareket, diğeri ise daha önce gördüğü ya da tanıdığı bir nesnenin ya da kişinin ekrana geldiğindeki gözün yaptığı harekettir. Bu konuda yapılmış bir diğerk çalışma gözün ekranda gördüğü aktiviteleri sınıflandırılmasıyla ilgilidir. Mathe [33] ve Sminchiescu [34] yaptıkları çalışmalarla film sahnelerindeki aktiviteleri ve izlenen spor türlerinin sınıflandırılmasıyla ilgili iki çalışma yapmışlardır.

2.3 Sıfır Hipotez Testi

Bilimsel yöntem aşamalarıyla tanımlanmış sistematik bir bilgi üretme biçimidir. Bilimsel yöntemin problemin belirlenmesi, olası çözüm önerilerin geliştirilmesi, çözümlerin test edilmesi, bulunan çözümlerin tekrarlanabilir olması ve raporlanması gibi birtakım aşamaları vardır. Bu aşamalardan çözümlerin üretilmesi hipotez ve araştırma sorularıyla ifade edilir. Hipotez sav demektir. Bir hipotez kurup bu hipotezi doğrulama veya yanlışlama ve sonunda bir karara varma işlemlerin bütününe hipotez testi denir [53]. Bu hipotezin test edilmesi aşamasında elde edilen sonuçların tesadüfe bağımlı olup olmadığını belirlemek için yapılan testlere istatistiksel önemlilik (significancy) testleri denir. Üzerinde çalışılan konuya bağılı olarak değişkenlik gösteren önemlilik miktarı genel olarak 0.05 olarak kabul edilir. Bunun anlamı elde edilen sonuçların yaklaşılmak istenilen sonuca göre 0.05'lik hata payı içerisinde kalması gerekmektedir.

Üstünde çalışılan veri setinin yapısına göre eşleştirilmiş T-test (paired samples T-test) , tek örneklem testi (one sample t-test) gibi farklı testler kullanılmaktadır [54]. Yapılan bu çalışmada tek örneklem testi kullanılmıştır. Bu test türünün hesaplanması aşağıdaki formül ile yapılmaktadır. Bu formülde z değeri normal dağılım değerini göstermektedir. X değeri sonuçların aritmetik ortalamasını göstermektedir. Δ değeri

test edilen evren deęeri, σ evren standart sapması, n ise örneklem büyüklüğünü göstermektedir.

$$Z = \frac{X - \Delta}{\frac{\sigma}{\sqrt{n}}} \quad (\text{Denklem 2.1})$$

Normal dağılım deęerinin hangi önemlilik deęerine karşılık geldiğini belirlemek için z tablosundan [55] faydalanılmaktadır.

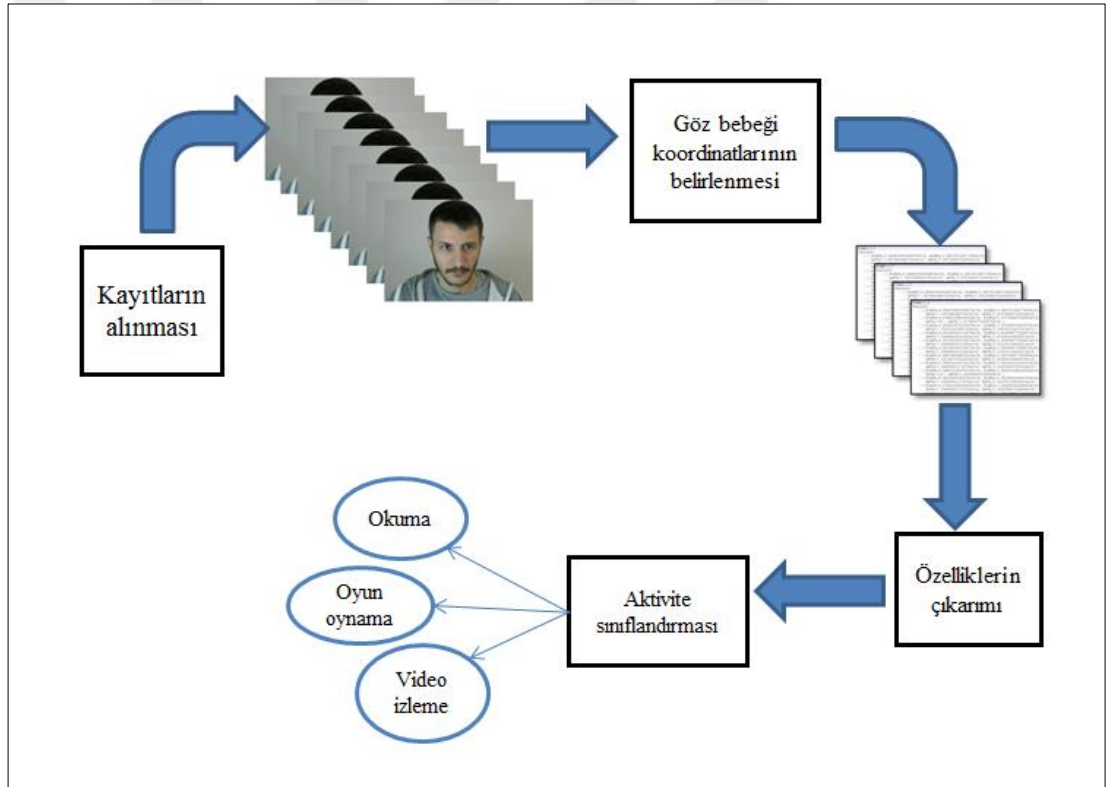


ÜÇÜNCÜ BÖLÜM

MATERYAL ve YÖNTEM

Bu çalışmanın ana amacı bilgisayar karşısındaki kişinin yaptığı eylem esnasında bir web kamera ile kayıt altına alınarak elde edilen düşük çözünürlüklü videolardan göz hareketlerini analiz etmek ve elde edilen özellikleri öğrenme algoritmaları kullanarak sınıflandırabilmektir. Geliştirilen sistemin genel görünümü Şekil 3.1' de verilmiştir.

Bu bölümde göz bebeğinin belirlenmesinde kullanılan yöntem, özelliklerin çıkarımı ve kullanılan sınıflandırma algoritmaları anlatılmaktadır.

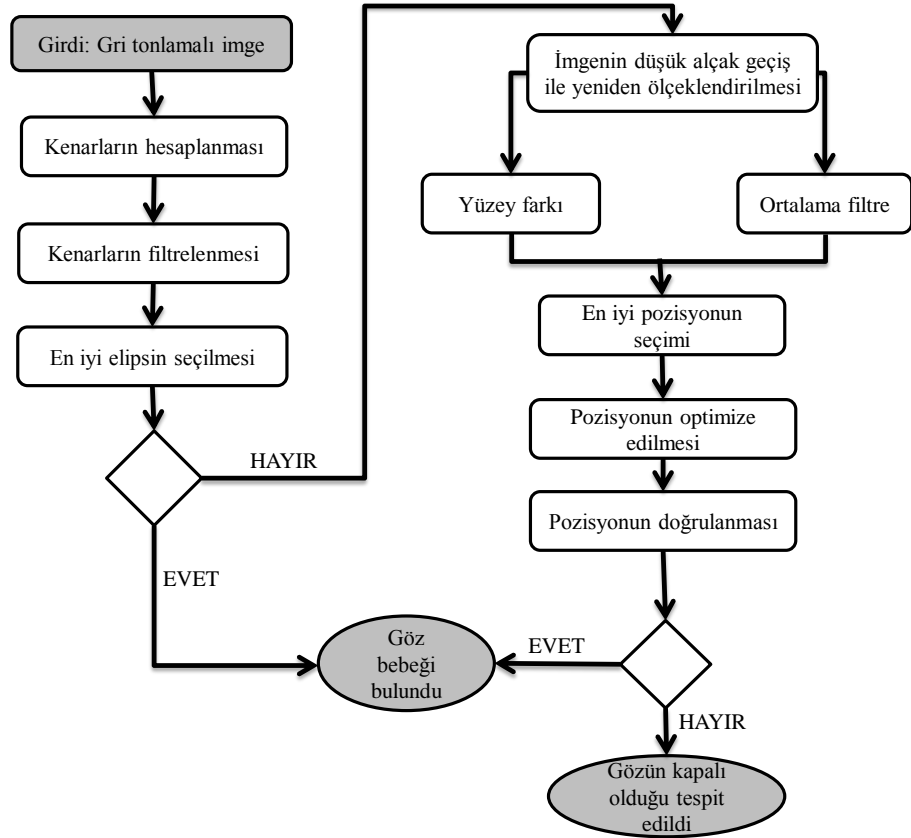


Şekil 3.1 Geliştirilen sistemin genel görünümü

3.1 Göz Bebeği Koordinatlarının Belirlenmesi

Çalışmada kullanılan algoritmanın adımları Şekil 3.2' de gösterilmektedir. Şekilde de görüldüğü gibi algoritma siyah beyaz resimler üzerinde uygulanmaktadır. Bu sebeple kamerada elde edilmiş renkli görüntüler öncelikle siyah beyaz resimlere dönüştürülmüştür. Algoritmanın çalışması için yapılan ön çalışmalardan bir diğeri ise önce imge üzerindeki yüz bölgesinin daha sonrada yüz üzerinde bulunan göz bölgesinin belirlenmesidir. Bu aşama için kullanılan yöntem ise Viola ve Jones' un [36] geliştirmiş olduğu Haar benzeri özellik tabanlı yöntemdir. Bu yöntem için daha önceden OpenCV Kütüphanesi içinde bulunan yüz bölgesi ve göz çiftini bölgesini bulmaya yarayan veri dosyalarından faydalanılmıştır. Bu yöntem en yüksek doğrulukla ve en hızlı şekilde istenilen bölgeyi bulduğu için tercih edilmiştir. Öncelikle yüz bölgesi bulunarak olası hatalı göz çifti bölgelerinin belirlenmesinin önüne geçinilmiştir. Daha sonra bulunan yüz bölgesi içinde göz çifti aranmıştır. Bulunan göz çiftini içeren dikdörtgen uygun parametrelerle sağ ve sol göz bölgesini barındıran iki farklı dikdörtgensel bölge içine alınmıştır. Bu aşamadan sonra Fuhl'un algoritmasına parametre olarak verilmeye hazır iki tane matris elde edilmiştir.

Şekil 3.2' de listelenen adımları sırasıyla şu şekildedir; öncelikle algoritmaya verilen göz bölgesi üzerine Canny kenar filtresi uygulanmıştır. Filtre uygulandıktan sonra göz bebeği etrafında kenar birleşmesini bozan noktalar temizlenmiştir. Bu adımdan sonra birleşik kenar bölgesi elde edilerek bu bölgenin doğruluk, iç bölge yoğunluk değeri, eliptik özellikleri, bu bölge üzerine bir elipsin yerleştirip yerleştirilemeyeceği ve son olarak buranın göz bebeği olma uygunluğu gibi değerleri belirlenmiştir. Eğer bu değerlerin sonuçları olumlu olursa göz bebeği tespit edilmiştir. Sonuçlar olumsuz olursa tekrar göz bölgesinin büyüklüğünü değiştirilir. Elde edilen ölçüleri değiştirilmiş göz bölgesi üzerinde ortanca filtre uygulanarak elde edilen olası göz bebeği bölgesi tekrar göz bebeği olmak için uygunluk testine girmektedir. Bu kontrolden sonra göz bebeği bulunur ya da gözün kapalı olduğuna karar verilmektedir. Algoritma adımlarında bahsedilen kenar filtreleri ve en uygun elipsin seçilmesi adımları sonraki başlıklarda detaylı anlatılmıştır.

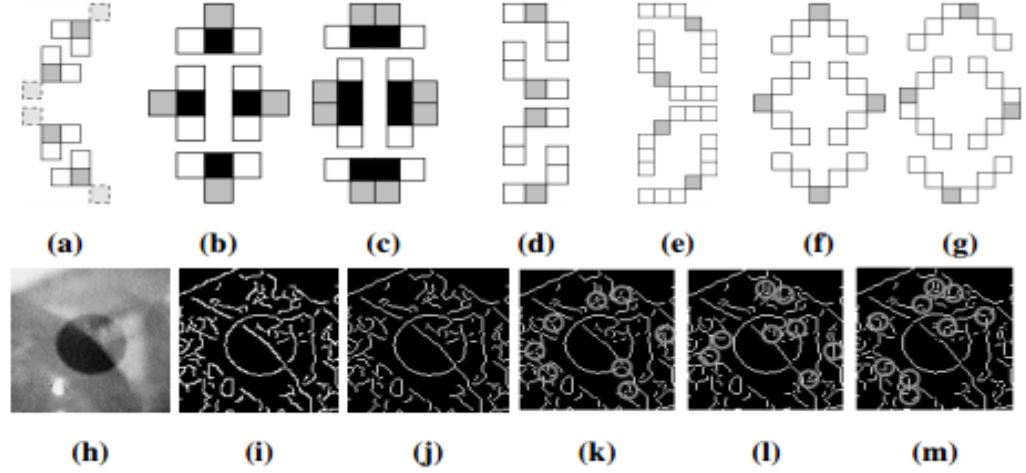


Şekil 3.2 Göz bebeği bulma algoritmasının adımları [35]

3.1.1 Kenar Filtreleri

Filtreleme sonucunda oluşan kenar elips oluşturmazsa parçalara ayrılır. Parçalama işlemi yapılırken kenar noktalarını inceltme ve doğrultma işlemi yapılarak parçalanabilir hale getirilirler. Bu işlem için morfolojik ve algoritmik iki farklı yaklaşım mevcuttur. Çalışmada kullanılan algoritma için Fuhr[35]'un morfolojik yöntemi tercih edilmiştir. Bu yöntem içerisinde daha az aritmetik işleme ihtiyaç duyulmaktadır. Şekil 3.3'de görülen morfolojik desenlerde elips oluşturmak için kenar bölgelerinde çeşitli manipülasyonlar yapılmaktadır. Şekil 3.3 (a)' da görülen 2 den daha fazla sayıda komşusu olan kenar noktalarının komşuları silinir. Şekil 3.3(i)'de Canny filtresiyle bulunan kenarlar inceltilerek Şekil 3.3(j)'deki imge elde edilir. Bu imge üzerinde iki komşu noktadan daha fazlası silinerek iki hattın çakışması engellenir. Bu işlem sonucunda elde edilen Şekil 3.3(k)'deki imge elde edilir. Bu imge Şekil 3.3 (b) ve Şekil 3.3(c) desenleri uygulanarak kenarları düzeltilmiş Şekil 3.3(l)

elde edilir. Bu imgede sırasıyla Şekil 3.3 d, Şekil 3.3.e, Şekil 3.3f ve Şekil 3.3g uygulanarak Şekil 3.3(m) 'de görülen imge elde edilir.

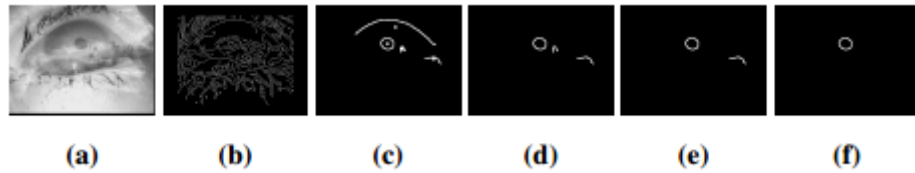


Şekil 3.3 Kenar manipülasyonu için morfolojik desenler [35]

3.1.2 En Uygun Elipsi Seçme

Bu adımda kenarları gösteren resimdeki tüm ardışık kenar pikselleri toplanır. Morfolojik yaklaşım bu adım için yapılan bir işlemdir. Bu sayede bağlantılı kenar pikselleri bir hat ile ifade edilebilir. Toplanan hat bilgileri kullanılarak Şekil 3.4' de görülen aşamalar sonunda olası göz bebeği bölgesi belirlenir.

Elips bulunurken düz hat oluşturan kenar noktaları çıkarılır. Geriye kalan eliptik hatlardan faydalanarak kapalı bir elips aranır. Elips filtreleme yapılarak geri kalan hatlarda yok edilir ve en son aşamada Şekil 3.4(f)'de görülen elips bulunur.

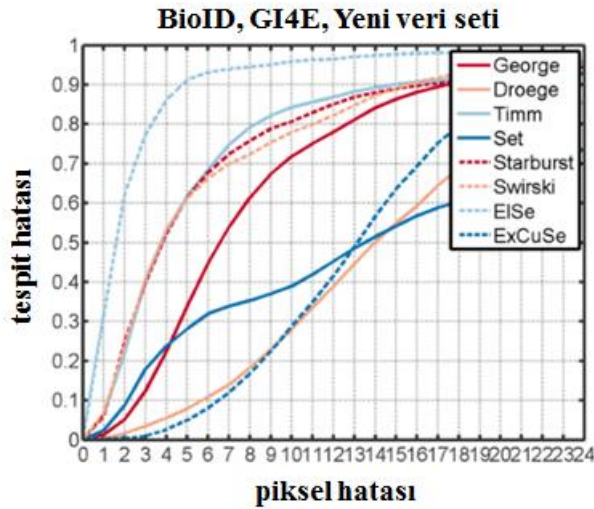


Şekil 3.4 Elipsin Oluşturulması [35]

İmge üzerinde bulunan elipsin iç sınır yarıçapı ve dış sınır yarıçapı hesaplanır. En az %5 en çok %10 hata payı kabul edilerek bu bölge üzerine bir elips yerleştirilmeye çalışılır. Elips yerleştirildikten sonra yerleşen elipsin merkezi göz bebeği olarak kabul edilir.

Fuhl [35] geliştirdiği tekniği BioID [48], GI4E ve kendi oluşturduğu veri kümesiyle test etmiş ve başarılı sonuçlar elde etmiştir. Şekil 3.5’ de Fuhl kendi geliştirdiği ExcuSe ve ElSe [35] algoritmaları ile diğer başarılı göz bulma algoritmalarını kıyaslayan bir tablo oluşturmuştur. Bu tabloda da görüldüğü gibi ElSe algoritmasının başarısı göz bebeği bulmada kullanılan parametrelerden hata payı miktarı arttıkça %100 ‘e yaklaşmıştır. Bunun yanında daha önce geliştirilen algoritmalarından farklı olarak düşük hata paylarında da diğer algoritmalara göre çok daha başarılı sonuçlar elde etmiştir.

Bu çalışma esnasında da kullanılan algoritma girdi olarak göz bölgesini almaktadır. Çalışmanın yapıldığı dokümanda algoritma başarı yüzdesi verilmiş olmakla birlikte bu çalışma esnasında kullanıcıların göz bebeğini belirlemede başarı oranı tekrar araştırılmıştır. BioID veri seti üzerinden opencv kütüphanesi yardımıyla belirlenen göz bölgeleri algoritmaya girdi olarak verilmiştir. Yapılan çalışma sonucunda algoritma hata payı %25 olarak belirlendiğinde %94 oranında başarılı sonuç alındığı gözlenmiştir.



Şekil 3.5 Farklı göz bebeği bulma algoritmalarının başarı grafiği

3.2 Özellik Çıkarımı

Bilgisayar karşısında alınan kayıtlardan göz bebeği koordinatları belirlendikten sonra her bir örneği özellik vektörleri ile temsil edebilmek için sağ ve sol göz bebeklerinin x ve y değerlerinden ve bu değerlerin birim zamanda yapmış oldukları yer değişim miktarlarından elde edilmiş istatistiksel değerler kullanılmıştır. Bu özelliklerden 9 tanesi zaman konum bazlı ve 1 tanesi frekans bazlı (frequency domain) olmak üzere iki farklı alandan alınmış toplam 10 özelliktir. Konum alanından alınan özellikler maksimum, minimum, ortalama ve standart sapma vb. istatistiksel özelliklerdir. Frekans alanından alınan özellik ise sinyal enerjisi özelliğidir. Göz koordinatları x ve y koordinatları için ayrı ayrı alındığında oluşturulan özellikler de her bir koordinat için ayrı oluşturulduğunda toplam 40 özellik elde edilmiştir.

Deneklerden alınan yaklaşık 5 dakikadan oluşan video kayıt örnekleri 15'er saniyelik alt videolara bölünmüştür. Bu alt videolar saniyede yaklaşık 15 çerçeve (frame)'den oluşan sağ ve sol göz koordinatlarının yazıldığı $(x_{sol}, y_{sol}, x_{sağ}, y_{sağ})$ boyutu $N \times 4$ olan S matrisi oluşturulmuştur. Tüm veri seti için oluşturulan bu matrislerin her biri sağ ve sol gözün koordinat değerleri için boyutları $1 \times N$ olan 4 farklı vektöre bölünmüştür; $S_k, k = 1, 2, 3, 4$. Özellik çıkarımı için bu vektörler kullanılmıştır. Öncelikle her bir vektörün minimum, maksimum ve dağılım (entropy) değerleri bulunmuştur, sonra vektör elemanlarının en küçük değeri 0 en büyük değeri 1 olacak şekilde normalizasyon yapılmıştır. Normalizasyon işleminin yapılma amacı deneklerin ekran karşısında kameraya farklı yüksekliklerden bakmasından kaynaklı oluşabilecek farklılıkları giderilmesidir. Bu işlemden sonra her bir vektörün, ortalama (mean), medyan (median), standart sapma (standard deviation), varyans (variance), çarpıklık (skewness), ortalama karekök (root mean square) ve sinyal enerji değerleri bulunarak 4 vektör için 10 ayrı özellik elde edilerek toplamda 36 farklı özellik oluşturulmuştur.

$$\text{ortalama}(S_k) = \mu_{S_k} = \frac{1}{N} \sum_{i=1}^N S_{k_i} \quad , k = 1,2,3,4 \quad (\text{Denklem 3.1})$$

$$\text{standart sapma}(S_k) = \sigma_{S_k} = \frac{1}{N} \sum_{i=1}^N (S_{k_i} - \mu_{S_k})^2 \quad , k = 1,2,3,4 \quad (\text{Denklem 3.2})$$

$$\text{varyans}(S_k) = \sigma_{S_k}^2 = \frac{1}{N} \sum_{i=1}^N S_{k_i} - \mu_{S_k} \quad , k = 1,2,3,4 \quad (\text{Denklem 3.3})$$

$$\text{çarpıklık}(S_k) = \frac{1}{N\sigma^3} \sum_{i=1}^N (S_{k_i} - \mu_{S_k})^3 \quad , k = 1,2,3,4 \quad (\text{Denklem 3.4})$$

$$\text{ort. kare kök}(S_k) = \sqrt{\frac{1}{N} \sum_{i=1}^N S_{k_i}^2} \quad , k = 1,2,3,4 \quad (\text{Denklem 3.5})$$

$$\text{dağınum}(S_k) = H(S_k) = E\{I(S_k)\} = E\{-\ln(P(x))\} \quad , k = 1,2,3,4 \quad (\text{Denklem 3.6})$$

$$H(S_k) = \sum_{i=1}^n P(S_{k_i}) I(S_{k_i}) = -\sum_{i=1}^N P(S_{k_i}) \log_b P(S_{k_i}) \quad , k = 1,2,3,4 \quad (\text{Denklem 3.7})$$

$$\text{sinyal enerji}(E_{S_k}) = \langle S_k, S_k \rangle = \int_{-\infty}^{\infty} |S_k|^2 d_{S_k} \quad (\text{Denklem 3.8})$$

Yukarıdaki formüllerde S_k , S matrisinden oluşturulan alt vektörü, S_{k_i} ise o vektörün i . elemanını temsil etmektedir. Dağınum hesabında Shannon's Entropy formülü kullanılmıştır. Formül (3.6) ve (3.7)'deki E beklenen değer operatörüdür (expected value operator), I ise S_k 'nin bilgi içeriğidir (information content), P ise olasılık kütle fonksiyonunu (probability mass function) temsil etmektedir. Formül (3.7)'de verilmiş olan dağınum hesabında b kullanılacak logaritma tabanını temsil etmektedir. Shannon's Entropy yaklaşımı kullanıldığı için b değeri 2 olarak alınmıştır. Buna ek olarak frekans bazlı özellik olan sinyal enerjisi hesaplayabilmek için öncelikle zaman alanında olan S_k sürekli sinyalini (vektörünü) frekans alanına dönüştürmek için Fast Fourier dönüşümü kullanılmıştır. Frekans alanına dönüştürülen sinyalin enerjisi formül (3.8) ile hesaplanmıştır. Sinyal işlemede, sinyal zamanın bir fonksiyonu olarak görülmektedir. Sinyal gücünün bir ölçümü olan sinyal enerji özelliği sıklıkla kullanılan bir özelliktir.

3.3 Özellik Seçimi

Özellik seçimi yöntemleri sınıflandırma uygulamalarında sıklıkla kullanılan yöntemlerdir. Sınıflandırıcı performansını yükseltmek, daha hızlı ve daha az hesaplama maliyetine sahip sınıflandırıcılar geliştirmek ve verinin üretim sürecinin daha iyi anlamak için kullanılan yöntemlerdir. Çalışmamızda OneRAttribute, ReliefAttributeEval, CfsSubsetEval ve Principal Component Analysis (PCA) özellik seçim teknikleri kullanılmıştır. OneRAttribute yöntemi özelliği değerlendirirken OneR sınıflandırıcısını kullanmaktadır. OneR sınıflandırıcısı verideki her bir sınıflandırıcı için bir kural belirleyen ve en küçük toplam hataya göre o sınıflandırıcının tek kuralını seçen bir yöntemdir. ReliefAttributeEval yöntemi bir örneği tekrar tekrar örnekleyerek, aynı ve farklı sınıfın en yakın örneği için verilen özelliğin değerini göz önünde bulundurarak bir değerlendirme yapmaktadır. CfsSubsetEval yöntemi ise özellik kümelerinin değerlendirmesini, her bir özelliğin tahmin edilebilirliğini aralarındaki fazlalık derecesi ile birlikte ele alarak yapmaktadır. Bu yöntemde sınıf ile yüksek korelasyonu olan, iç korelasyonu düşük olan özellik altkümeleri tercih edilmektedir. Bir diğer kullanılan yöntem olan PCA, örüntü tanıma ve sınıflandırma uygulamalarında sıklıkla kullanılan bir ön işleme adıdır. Bu yöntem boyut azaltmada (dimension reduction) kullanılan, verinin daha küçük boyutlarda temsil edildiği ve varyansının maksimize edildiği lineer bir yöntemdir. Boyut azaltmadaki temel amaç belirlenen optimalite kriterlerine göre orijinal verideki en önemli bilginin çıkarımını yapmaktır [49]. Sınıflandırma yöntemlerinin performansları PCA işleminden sonra nasıl değiştiği gözlemlenmiştir.

3.4 Sınıflandırma Algoritmaları

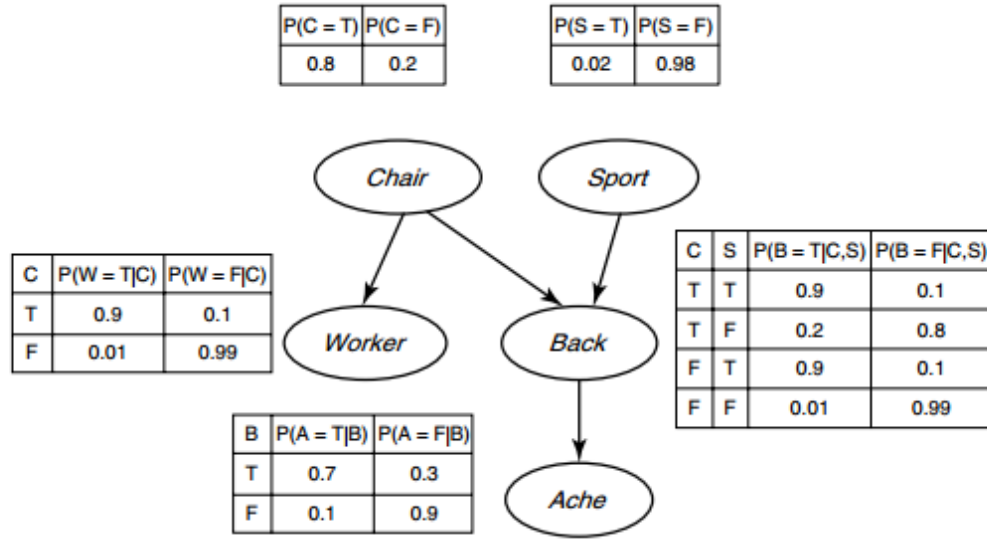
Çalışmanın temel amacı olan kişinin göz hareketlerine göre aktivite sınıflandırmasında Bayesian Ağlar, k-En yakın komşuluk algoritması (k-NN), Destek Vektör Makinası (Support vector machine-SVM) sınıflandırıcıları kullanılmıştır. Sınıflandırıcıların uygulanması WEKA yazılımı [37] kullanılarak gerçekleştirilmiştir. Oluşturulan farklı aktivite örnekleri ve üç farklı sınıf (aktivite) kullanılarak sınıflandırma yöntemlerinin performansları kıyaslanmıştır.

3.4.1 Bayesian Ağlar (Bayesian Networks)

Bayesian Ağ (BA) [19] ya da Bayesian Model olarak anılan algoritma grafik teorisi, olasılık teorisi, bilgisayar bilimleri ve istatistik dallarının beraber kullanılmasıyla ortaya çıkmıştır. BA grafik modeller içerisinde yönlü dairesel olmayan grafik (DAG) olarak bilinir. Bu model rastgele değişkenler kümesinin birleşik olasılık dağılımını hesaplamada [20] ve göstermede etkili bir yöntemdir.

DAG yapısı iki küme ile ifade edilir. Bunlardan birisi düğümler kümesi ikincisi ise yönlü kenarlar kümesidir. Düğümler rastgele değişkenleri simgeler ve bunlar daire içerisinde değişken ismi yazılarak gösterilir. Kenarlar değişkenler arasındaki yönlü ilişkiyi gösterirler ve ok ile ifade edilirler. Şekil 3.6' da görüldüğü gibi değişkenlerin belirlenen bir durum için olasılık dağılımları hesaplanarak birden fazla değişkenin bir durum üzerindeki birleşik olasılık dağılımı hesaplanır.

BA'nın öğrenmek için birisine ihtiyacı vardır. Ona sağlanan veriler değişkenler arasındaki ilişkiyi yani grafik modeli ve JPD'nin hesaplanması için gerekli parametreleri içerir. Bu durum bazı değişkenlerin eksik olmasında veya sağlanan verinin yanlış ya da eksik olması durumunda başka sıkıntılar oluşmasına sebep olur. Şekil 3.6' da BA yapısının ve gözlenebilirlik durumuna göre uygulanacak öğrenme yöntemleri mevcuttur. "Maximum-likelihood estimation" [22] en basit durum için uygulanan bir yöntemdir. İkinci durum için "Expectation Maximization"(EM) [23] ya da "Markov Chain Monte Carlo"(MCMC) [24] yerel en uygun çözümü bulmak için kullanılır. Üçüncü ve dördüncü durum "NP-hard" (nondeterministic polynomial) [25] problemlerdir. Bu durumlar için fayda yarar getirisi hesaplanarak en yakın çözüm bulunmaya çalışılır.



Şekil 3.6 Sınıfların birbiriyle ilişkisi [21]

Tablo 3.1 BA Öğrenme algoritması için farklı durumlarda kullanılan yöntemler [21]

Durum	BN yapısı	Gözlemlenebilirlik	Geliştirilen öğrenme yöntemi
1	Biliniyor	Tam	En büyük olasılık kestirimi
2	Biliniyor	Kısmi	EM (veya gradyan yükselişi)
3	Bilinmiyor	Tam	Model uzayında arama yapmak
4	Bilinmiyor	Kısmi	EM + model uzayında arama yapmak

3.4.2 Destek Vektör Makinası (Support Vector Machine)

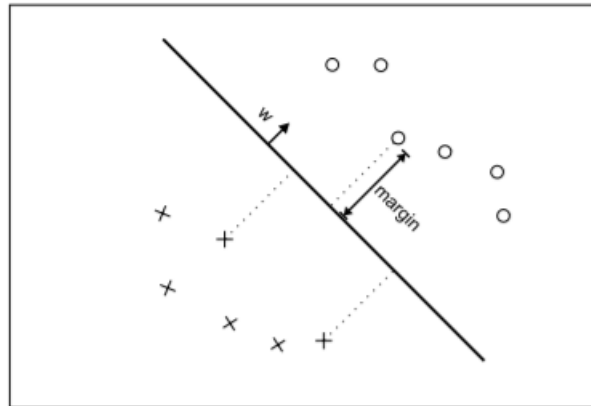
SVM ilk olarak Vapnik [38] tarafından tanıtılmış, günümüzde yaygın olarak kullanılan makine öğrenme algoritmalarından biridir. Veriyi sınıflandırmada ve regresyon analizinde kullanılır. Daha çok obje, ses, el yazısı tanıma vb. uygulamalarında kullanılan bir yöntemdir.

Bu algoritmaya verilen eğitim örnekleri iki kategori ile işaretlenir. SVM bir model çıkartırken eğitim için verilen veri kümesini iki kategoriden birinde olacak

şekilde işaretler, bir diğer ifadeyle veriyi yüksek boyutlu bir uzaya yerleştirerek bu veriyi uzayda en iyi ayıracak olan hiperdüzlemi (hyperplane) yapılandırır. SVM modeli noktalar uzayında düşünülürse bu iki kategoriye ait noktaları belirli bir mesafe üzerinden Şekil 3.7’ de görüldüğü gibi iki gruba böler. Burada oluşturulan hiperdüzlem iki farklı veri grubunu net bir şekilde birbirinden ayırmaktadır.

SVM yöntemi ilk olarak 2-sınıflı sınıflandırma problemleri için geliştirilmiştir. Daha sonraki çalışmalarda bu yöntemin çok sınıflı problemlere uygulanabileceği üzerinde çalışılmıştır [49; 50]. Bu çalışmalarda iki farklı yaklaşım üzerinde durulmuştur; bir-diğerleri (one-versus-all-OVA) ve birebir (one-versus-one-OVO) yaklaşımları. Birinci yaklaşımda k sınıflı bir problem için k adet ikili sınıflandırıcı oluşturulur. Burada ele alınan sınıftan olanlar pozitif sınıf, o sınıftan olmayanlar negatif sınıf olarak ele alınarak iki sınıf elde edilir. İkinci yaklaşımda ise tüm olası ikili sınıflandırıcılar oluşturulur bu durumda $\frac{k(k-1)}{2}$ adet sınıflandırıcı oluşturulur. Her bir sınıflandırıcının bir test örneğine uygulanması seçilen sınıfa bir puan kazandırır. Bir test örneği en çok puanı alan sınıf ile etiketlenir. OVO yaklaşımının sınıflandırıcıları boyutları OVA yaklaşımınınkinden büyüktür, bu da OVO yaklaşımının hesaplama maliyetini arttırmaktadır.

Çalışmamızda SVM yöntemi OVO yaklaşımı ile uygulamıştır. Diğer sınıflandırma algoritmalarında olduğu gibi ele alınan sınıftan olan örnek pozitif örnek olmayan örnekler ise negatif örnek olarak belirlenmiştir.

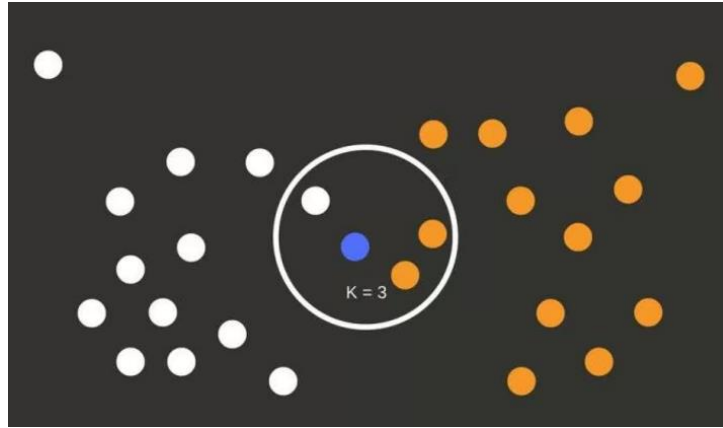


Şekil 3.7 Verinin belirgin şekilde gruplara ayrılması [39]

3.4.3 k- En yakın komşuluk Algoritması (k-NN)

En yakın komşuluk algoritması (k-NN) [40] algoritması sınıflandırma ve regresyonda kullanılan parametrik olmayan bir yöntemdir. k-NN yöntemi, örüntü tanıma, bilgisayarla görü, biyoenformatik gibi farklı çalışma alanlarında yaygın olarak kullanılan bir algoritmadır.

Bu yöntemde amaç çok boyutlu özellik uzayında her biri bir sınıfa sahip birer vektör olarak ifade edilen objeler içerisinde sınıfı bilinmeyen objenin sınıfını belirlemektir. Şekil 3.8’ da görüldüğü gibi sınıfı bilinmeyen bir objenin (mavi obje), kendisine en yakın k (k=3) komşusunun sınıf bilgilerini kullanarak sınıf belirlemesinin yapılmaktadır. Şekilde k=3 değerine göre objeye en yakın 3 obje belirlenmiş onların sınıf (renk) bilgisine göre (2 sarı, 1 beyaz) sınıfı bilinmeyen objenin sınıf ataması sarı olarak belirlenmektedir. Algoritmada k parametresi kullanıcı tanımlı bir parametredir ve tanımlanmamış objenin sınıf ataması kendisine en yakın k obje içerisinde en sık rastlanan sınıf bilgisine göre yapılmaktadır. Objelerin birbirlerine olan uzaklığı hesaplanırken genellikle “Euclidean” uzaklık metriği kullanılmaktadır. K-nn yönteminde k parametresinin seçimi önemli bir faktör olmakla birlikte, genellikle artan k değeri ile birlikte sınıflandırmadaki hata payının etkisini azaldığı bilinmektedir. K parametresini belirlemenin belirgin bir yolu bulunmamaktadır; sınıflandırma çalışmalarında deneysel olarak belirlenen bir parametredir.



Şekil 3.8 En yakın komşuların belirlenmesi[46]

3.4.4 Rastgele Orman (Random Forests or Random Decision Trees)

Rastgele Ormanlar (Random Forests-RF) algoritması Leo Breiman [45] tarafından sınıflandırma için geliştirilmiştir. Birden fazla sınıflandırıcının üretilip, sonuçların birleştirilerek bir sonuca varılması yaklaşımı olan “Ensemble Learning”, son zamanlarda ilgi çeken bir konu haline gelmiştir. Bu konuda iki farklı yaklaşım bulunmaktadır; sınıflandırma ağaçlarının “Boosting” [41] ve “Bagging” [42] yaklaşımı. Boosting yönteminde ardışık ağaçlar, önceki sınıflandırıcılar tarafından yanlış sınıflandırılan noktalara ekstra ağırlık (weight) vermekte, sonuçta ağırlıklı seçim (weighted vote) sınıflama için alınmaktadır. Bagging yönteminde ise ardışık ağaçlar önceki ağaçların kararlarına bağlı değildir, her biri bağımsız bir şekilde yapılandırılmakta ve sınıflandırma yapmaktadır. Breiman, Rastgele Orman yönteminde daha önce geliştirmiş olduğu “bagging” katmanına ek olarak “rastgellelik” katmanını eklenmiştir. Her bir ağacı farklı bir önyükleme örneği ile yapılandırmaya ek olarak rastgele ormanlar sınıflandırmanın veya regresyon ağaçlarının nasıl yapılacağını değiştirmektedir. Standart ağaçlarda her bir düğüm tüm değişkenler içindeki en iyi ayırıcı kullanılarak ayrılmaktadır. Rastgele orman ağaçlarında ise o düğümde rastgele seçilen alt sınıflandırıcı kümesi içinden seçilen en iyi sınıflandırıcı tarafından ayırım yapılmaktadır. Bu durum rastgele orman yöntemini diğer sınıflandırma metotlarına kıyasla daha iyi sonuçlar üretebilen ve aşırı uyuma (overfitting) probleminde karşı daha dayanıklı bir yöntem haline getirmektedir [43].

RF yöntemi birden fazla karar ağacı oluşturarak modeli eğiten, bu ağaçları kullanarak sınıflandırma yapmaktadır. Ağaçlardaki her iç düğüm ilgili örnekleme özelliği temsil ederken, her bir dal ise test sonucunu, yaprak düğüm ise sınıf etiketini temsil etmektedir. Her bir karar ağacı girdi verisinden rastgele seçilmiş değerler kullanılarak oluşturulmaktadır. Eğer orijinal öznitelik vektörü m adet öznitelige sahip ise, her bir ağaç n adet özniteliğin tüm özniteliklerden rastgele seçimini kullanmaktadır. Karar ağaçlarının kapasiteleri n' e ulaşana kadar büyümelerine izin verilir. Ormanı eğittikten sonra bir tahmine ulaşmak için her bir testin eğitilen ormandan geçmesi sağlanır. Bir sorgunun sınıflandırmasına yapılandırılan karar ağaçları arasından karar verilir.

DÖRDÜNCÜ BÖLÜM

DENEYSEL ÇALIŞMALAR

4.1 Veri seti

Yapılan çalışmada 10 farklı kişi denek olarak kullanılmıştır. Bu deneklerin dördü kadın altısı erkektir. Denekler arasında metin okuma teknikleri üzerine eğitim alan kimse bulunmamaktadır. Denekler aktiviteleri gerçekleştirmeleri için verilen materyallerin içeriğini ilk kez görmüşlerdir.

Deneklerin okuma eylemini gerçekleştirdiği doküman, içerisinde sadece metinsel ifadelerin olduğu altı sayfadan oluşmaktadır. Kullanıcılar sayfa geçişlerini fare üzerinde bulunan ekran kaydırma topunu kullanarak gerçekleştirmektedir. İzleme eylemi için içerisinde alt yazı gibi metinsel içeriğin olmadığı beş dakikalık bir video kullanılmıştır. Video tam ekran olarak izletilmiştir. Oyun oynama aktivitesi için kullanıcıların sadece fare ve üzerindeki tuşları kullanabileceği bir oyun seçilmiştir.

Denekler beyaz ışık ve gün ışığı ile aydınlatılmış odalarda aynı bilgisayar kullanılarak kayıt altına alınmışlardır. Kayıt cihazı bilgisayar ekranının üst kısmının tam ortasında bulunmaktadır. Bu şartlar altında deneklerden metin okuma, video izleme ve oyun oynama eylemlerini yukarıda bahsedilen materyalleri kullanarak yapmaları istenmiş ve her bir kullanıcıdan her bir eylem için yaklaşık beş dakikalık kayıtlar elde edilmiştir. Şekil 4.1’ de kayıt alımından örnek bir kesit verilmiştir. Görüldüğü üzere kullanıcı video izlerken ekran üzerinde yer alan kamera ile göz hareketleri kayıt altına alınmıştır. Alınan kayıtlar üzerinde bulunan kirli veriler temizlenerek 5 dakikalık her bir kayıt 15 saniyelik alt kayıtlara bölünmüştür. Bu işlemlerden sonra 192 izleme, 224 okuma, 172 oyun oynama olmak üzere toplam 588 aktivite kaydı elde edilmiştir. 15 saniyelik video parçalarından saniyede 30 tane çerçeve (frame) alınarak bu çerçevelerdeki sağ ve sol göz bebeği koordinatları “EISe” algoritmasının yardımıyla belirlenmiştir. Her bir çerçevede bulunan sağ ve sol göz bebeğinin koordinatlarının x ve y değeri bir dosyaya yazılmıştır. Bu işlemden sonra göz bebeklerinin birim zamanda yaptıkları değişimler belirlenmiştir. Sonraki adımda bu veriler kullanılarak maksimum, minimum, ortalama ve standart sapma gibi istatistiksel özellikler belirlenmiştir. Frekans alanından alınan özellik ise sinyal enerjisi özelliğidir. Göz koordinatları x ve y koordinatları için ayrı ayrı alındığında

oluşturulan özellikler de her bir koordinat için ayrı oluşturulduğunda toplam 40 farklı özellik elde edilerek her bir alt kayıt (örnek) özellik uzayında bu özellikler ile temsil edilmiştir. En son adımda her bir 15 saniyelik videoyu temsil eden örnekler videonun içerdiği eylem ile etiketlenmiştir.



Şekil 4.1 Kayıt alımından örnek bir kesit

4.2 Değerlendirme Yöntemleri

Sınıflandırma sonuçları p katlamalı çapraz doğrulama (p-fold cross validation) kullanılarak yapılmıştır. Uygulanan sınıflandırıcıların sonuçlarının değerlendirilmesi için kullanılan bu yöntemde veri seti eğitim ve test kümeleri olmak üzere ayrılmaktadır. P-katlamalı çapraz doğrulama yönteminde öncelikle bir p değeri belirlenir, veri seti p adet eşit parçaya ayrılır ve her bir parça sırasıyla hem eğitim kümesi hem test kümesi olmaktadır. Her bir test parçasından elde edilen sonucun en son ortalaması alınarak sistemin başarı yüzdesi hesaplanmaktadır. Bu yöntem ile veri集中的 her bir örnek hem eğitim kümesinin hem test kümesinde yer almaktadır.

Çalışmamızda p değerini belirlemede deneysel yöntem kullanılmıştır; farklı p değerleri için doğruluk yüzdesi hesaplanarak en iyi sonucun elde edildiği p değeri seçilmiştir. Doğruluk yüzdesi (accuracy) hesabı (Denklem 4.1) aşağıdaki formül kullanılarak hesaplanmıştır.

$$\text{doğruluk yüzdesi} = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (\text{Denklem 4.1})$$

Doğruluk yüzdesi hesabında; o sınıftan olanlar pozitif, olmayanlar negatif olarak isimlendirilirse formülde belirtilen oranlar aşağıdaki gibi hesaplanmaktadır;

TP (True positives): gerçekte pozitif olan, sistem tarafından da pozitif olarak işaretlenmiş örnek sayısı,

FP (False positive): gerçekte negatif olan, sistem tarafından da pozitif olarak işaretlenmiş örnek sayısı,

TN (True negatives): gerçekte negatif olan, sistem tarafından da negatif olarak işaretlenmiş örnek sayısı,

FN (False negative): gerçekte pozitif olan, sistem tarafından negatif olarak işaretlenmiş örnek sayısıdır.

Çalışmamızda üç farklı sınıf bulunduğu için; doğruluk yüzdesi her bir sınıf için ayrı ayrı hesaplanıp en son olarak ortalama bir doğruluk yüzdesi hesaplanmaktadır. Doğruluk yüzdesi makine öğrenme yöntemlerinin performanslarını değerlendirme yeterli olmayacağından TP ve FP oranları kullanılarak hesaplanan çeşitli ölçümler kullanılmıştır. Bunlar Precision (kesinlik) (10), Recall (hassasiyet) (11) ve F-measure (12) ROC (Receiver Operating Curve) alanı ve PRC (Precision-Recall Curve) alan değerleridir. Bu ölçümler sistem performansları değerlendirmede sıklıkla kullanılan ölçümlerdir. Kesinlik (Precision) bir dizi sonuç arasında anlaşmanın yakınlığını, güvenilirlik ölçüsünü, pozitif olarak tahmin edilen örnek sayısının oranını bularak elde etmektedir. Duyarlılık (Recall) ise gerçekte pozitif örneklerden doğru tahmin edilen örneklerin oranını hesaplayan bir ölçümdür [45]. F-ölçütü kesinlik ve duyarlılık değerlerinin harmonik ortalaması ile hesaplanan bir ölçümdür. Aynı zamanda kesinlik ve duyarlılık değerlerinin ağırlıklı ortalaması da denilmektedir; en başarılı performansın skoru 1 olurken en kötü performans skoru 0 olmaktadır.

$$kesinlik = \frac{TP}{TP+FP} \quad (\text{Denklem 4.2})$$

$$duyarlılık = \frac{TP}{TP+FN} \quad (\text{Denklem 4.3})$$

$$F - \text{ölçütü} = 2x \frac{kesinlik \times duyarlılık}{kesinlik + duyarlılık} \quad (\text{Denklem 4.4})$$

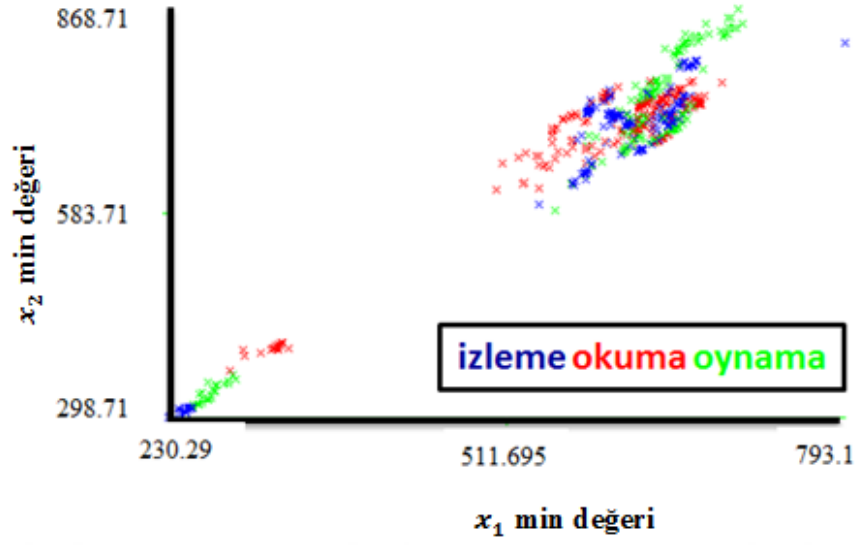
ROC alanı, ROC eğrisi altında kalan alanın değerini gösterirken, PRC (Precision Recall Curve) alanı kesinlik-duyarlılık eğrisi altında kalan alanın değeridir. ROC eğrileri doğru sınıflandırılan örnek sayısının yanlış sınıflandırılan negatif örnek sayısına göre nasıl değiştiğini gösteren bir egridir. Eğrinin alan değerinin 1' e yakın

olması iyi performansı göstermektedir. PRC eğrileri ise ROC eğrilerine alternatif olarak geliştirilmiş, Bilgi Çıkarımı alanında sıklıkla kullanılan bir yöntemdir. PR eğrileri kesinlik- duyarlılık değerlerinin grafiksel ölçütüdür [46].

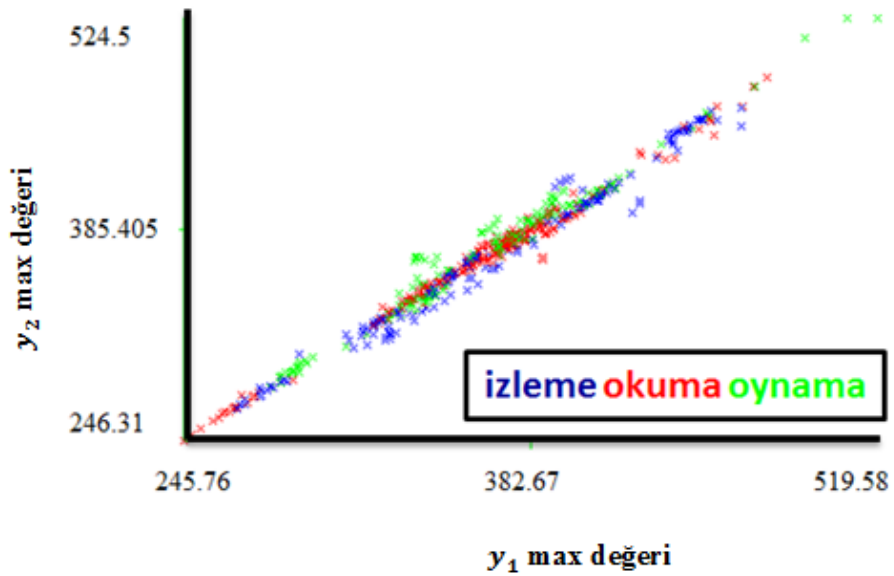
Elde edilen deneysel sonuçların istatistiksel olarak anlamlı olup olmadığını test etmek performans değerlendirmesinde önemli yöntemlerdendir. Sınıflandırma algoritmalarının performanslarını değerlendirmede en iyi sonucun elde edildiği yöntem referans yöntem olarak belirlenip, diğer üç yöntemin sonuçları bu referans yöntemin sonuçları ile ikili olarak İkili t-test (paired t-test) ile kıyaslanmıştır. Bağımlı örneklem t-testi (dependent sample t-test) olarak da bilinen ikili t-test, iki farklı küme gözlemlerinin arasındaki farkın anlamlı olup olmadığını ölçmede kullanılan istatistiksel bir testtir. Anlamlılık seviyesi (significance level) önemli parametrelerinden birisidir ve genellikle 0.05 olarak belirlenir. Elde edilen ikili t-test sonucunun p değeri belirlenen anlamlılık seviyesinden küçük ise farkın istatistiksel olarak anlamlı olduğu sonucuna varılmaktadır. P değeri (olasılık) ilgili h.potez testi sonucunda anlamlı bir fark denileceği durumda hatalı karar verme olasılığını temsil eden bir değerdir. Literatürde p değeri için 0.05 olarak belirlense de araştırmacının isteğine göre farklı değer (örneğin 0.01) de belirlenebilmektedir [51]. Çalışmamızda bu testi uygulamada WEKA' nın doğrulanmış ikili t-test (corrected paired t-test) paketinden faydalanılmıştır.

4.3 Deneysel Sonuçlar

Çalışmanın ilk adımı olan göz bebeği koordinatlarının belirlenmesinden sonra, elde edilen koordinat değerlerinin birbirleri ile tutarlı olduklarını göstermek amacıyla koordinatların minimum ve maksimum değerleri kontrol edilmiştir. Şekil 4.2' de sağ ve sol gözün x koordinatlarının minimum değerleri gösterilmektedir. Sağ ve sol gözün birbirleriyle uyumlu bir şekilde hareket ettiği bilgisi grafiğin doğrusallığından çıkarılabilmektedir. Bu grafikteki üç farklı renk üç farklı eylemi temsil etmektedir. Şekil 4.3 'de aynı amaçla sağ ve sol göz bebeklerinin y koordinat değerlerinin maksimum değerleri gösterilmiştir.



Şekil 4.2 Tüm denek göz bebeklerinin x_1 ve x_2 koordinatlarının minimum değerlerinin grafiği



Şekil 4.3 Tüm denek göz bebeklerinin y_1 ve y_2 koordinatlarının maksimum değerlerinin grafiği

Çalışmamızda kullanılan Bayesian Ağlar, SVM, k-NN ve Rastgele Orman sınıflandırma yöntemlerinin performansları aşağıdaki tablolarda verilmiştir. Tablo 4.1’ de sonuçları görülen Bayesian Ağ yönteminin kestirici parametresi Bayes Ağ yapısının koşullu olasılık tablosundan öğrenme değeri 0.5 olarak belirlenmiştir. Ayrıca arama algoritması parametresi TAN seçilmiştir. TAN arama algoritması öğrenme algoritması tarafından oluşturulan ağacın kapsadığı azami ağırlığı belirler ve bir ağaçla

zenginleştirilmiş Naive Bayes ağını döndürür. Bu parametrelerle çalıştırılan Bayesian Ağ sınıflandırma algoritması üç aktivite içinde yüksek sayılabilecek bir değer olan ~0.76 TP oranını üretmiştir. Aynı zamanda üç aktivite için FP oranları da birbirlerine yakın ve düşük değerlerde çıkmıştır. Kesinlik ve duyarlılık verilerine bakıldığı zamanda bu değerlerde 0.76 etrafında değişmektedir. Üç aktivitenin hepsine bakıldığında ise genel sınıflandırma başarısının %77 civarında olduğu gözlemlenmiştir. Aynı zamanda ROC alanı ve PRC alanı değerleri sistem başarısını destekler orandadır. Ayrıca Tablo 4.2’ de gösterilen karışıklık matrisine bakıldığı zaman da 192 video izleme aktivitesinin 145’i doğru olarak tahmin edilmiştir. Bunun yanında 224 okuma eyleminin 174’ü, 142 oyun oynama eyleminin ise 134’ü doğru tahmin edilmiştir. Tablodaki bu değerlere bakıldığı zaman genel olarak Bayesian Ağ yöntemi aktivite sınıflandırmada başarılı sayılabilecek bir algoritmadır.

Tablo 4.1 Bayesian Ağlar yönteminin sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.755	0.101	0.784	0.755	0.769	0.927	0.875	izleme
0.777	0.140	0.773	0.777	0.775	0.906	0.854	okuma
0.779	0.106	0.753	0.779	0.766	0.919	0.861	oyun oynama
0.770	0.177	0.771	0.770	0.770	0.917	0.863	ortalama

Tablo 4.2 Bayesian Ağlar yönteminin Karışıklık (Confusion) matrisi

Sistem tarafından tahmin edilen sınıf				
a	b	c		
145	27	20	a= video izleme	Gerçek sınıf
26	174	24	b= okuma	
14	24	134	c= oyun oynama	

SVM sınıflandırma algoritmasını uygulamada WEKA’nın SMO kütüphanesi [52] kullanılmıştır. Algoritmanın filtre parametresi standartlaştırılmış eğitim verisi seçilmiştir. Çekirdek parametresi “Pearson VII fonksiyonu” (PUK) seçilmiştir. Bu parametrelerle çalıştırılmış SVM algoritmasının sonuçları Tablo 4.3’ de gösterilmiştir. Algoritma okuma ve oyun oynama eylemleri için birbirlerine yakın TP ve FP değerleri

elde edilmiştir. İzleme aktivitesi için üretilen değer 0.792 ile diğer aktivitelere göre biraz daha aşağıda kalmıştır. FP değeri ise 0.061 ile en düşük izleme aktivitesi için elde edilmiştir. Algoritmanın kesinlik değerleri ortalama olarak 0.829 çıkmıştır. Bu algoritmanın oyun oynama ve okuma eylemi için duyarlılığı birbirlerine yakın ve izleme aktivitesinden daha fazladır. ROC alanı da en yüksek 0.895 ile izleme için çıkmıştır. SVM algoritmasının karışıklık matrisi ise Tablo 4.4’ de verilmiştir. Bu tabloda da görüldüğü gibi 192 izleme aktivitesinden 152’si doğru tahmin edilmiştir. Bu aktivitelere 20 tanesi okuma, 20 tanesi oyun oynama olarak etiketlenmiştir. 224 okuma aktivitesinden 188’i doğru olarak belirlenmiştir. 172 Oyun oynama aktivitesinin de 146’sı başarılı bir şekilde tahmin edilmiştir. Bu sonuçlara göre ortalama % 82.7 doğru sınıflandırma yapabilen SVM algoritması ile başarılı sonuçlar elde edilmiştir.

Tablo 4.3 SVM yönteminin sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.792	0.061	0.864	0.792	0.826	0.895	0.788	izleme
0.839	0.099	0.839	0.839	0.839	0.874	0.775	okuma
0.849	0.101	0.777	0.849	0.811	0.879	0.713	oyun oynama
0.827	0.087	0.829	0.827	0.827	0.882	0.761	ortalama

Tablo 4.4 SVM yönteminin Karışıklık (Confusion) matrisi

Sistem tarafından tahmin edilen sınıf				
a	b	c		
152	20	20	a= video izleme	Gerçek sınıf
14	188	22	b= okuma	
10	16	146	c= oyun oynama	

Çalışmada kullanılan bir diğer sınıflandırma algoritması k-NN algoritmasıdır. Bu algoritma kullanılırken uzaklık ağırlık parametresi 1/uzaklık olarak seçilmiştir. Mesafe belirleme algoritması olarak ise Öklid uzaklık yerine Manhattan uzaklık hesaplayan metot seçilmiştir. k-NN algoritması için belirlenen k=5 olarak belirlenmiştir. Bu parametrelerle çalıştırılan k-NN algoritmasının sonuçları Tablo 4.5’ de verilmiştir. Algoritmanın üç aktivite için de belirlediği TP oranları birbirlerine

yakın ve 0.87 civarında bir değerdir. 0.05 değeri ile oyun oynama aktivitesi için alınan FP değeri diğer iki aktivite içinde oldukça düşük bir değerdir. Algoritmanın aktiviteleri sınıflandırma konusundaki duyarlılık ve kesinlik değerleri de oldukça yüksek çıkmıştır. Üç aktivite için ROC alanı değerlerinin 1'e yakın çıkması algoritmanın sınıflandırma konusundaki başarısını desteklemektedir. k-NN sınıflandırma algoritması için Tablo 4.6' da karışıklık matrisi verilmiştir. Bu tablodan da görüldüğü gibi 224 okuma aktivitesinden 192'si, 192 video izleme aktivitesinden 169'u ve son olarak da 172 oyun oynama aktivitesinden 149'u doğru olarak sınıflandırılmıştır. Tüm bu sonuçlar Tablo 4.5' de görülen ortalama TP değerinin 0.867 olmasıyla beraber değerlendirildiğinde k-NN sınıflandırma algoritmasının oldukça yüksek bir başarı ile aktiviteleri sınıflandırdığı tespit edilmiştir.

Tablo 4.5 k-NN yönteminin sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.880	0.063	0.871	0.880	0.876	0.947	0.919	izleme
0.857	0.088	0.857	0.857	0.857	0.925	0.898	okuma
0.866	0.050	0.876	0.866	0.871	0.952	0.912	oyun oynama
0.867	0.069	0.867	0.867	0.867	0.940	0.909	ortalama

Tablo 4.6 k-NN yönteminin Karışıklık (Confusion) matrisi

Sistem tarafından tahmin edilen sınıf				Gerçek sınıf
a	b	c		
169	16	7	a= video izleme	
18	192	14	b= okuma	
7	16	149	c= oyun oynama	

Rastgele Orman sınıflandırma algoritması olağan parametreleri ile beraber çalıştırıldığı zaman en yüksek oranda başarılı sonuç üretmiştir. Bu sonuçlar Tablo 4.7' te verilmiştir. Bu yöntem aktiviteleri sınıflandırmada en başarılı sonuçların alındığı yöntemdir. Sistemin TP oranlarının ortalama %92 ve FP oranlarının %0.043 olması sistem başarısının en net göstergesidir. Bu değerler ile hesaplanan kesinlik ve duyarlılık değerleri ortalama sırasıyla %92,1 ve %92 olarak belirlenmiştir. ROC alanı

ve PRC alanı değerleri de sistemin aktivite sınıflandırılmasında kullanılabilir bir yöntem olduğunu desteklemektedir. Tablo 4.8'e bakıldığında 192 video izleme örneğinin 178'i doğru bir şekilde sınıflandırılmıştır. Algoritmanın 224 okuma aktivitesinden 211'ini ve 172 oyun oynama aktivitesinden 152'sini doğru sınıflandırarak kullanılan dört algoritma içindeki en yüksek başarı ile sınıflandırma yaptığı tespit edilmiştir.

Tablo 4.7 Rastgele Orman yönteminin sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.927	0.043	0.913	0.927	0.920	0.989	0.979	izleme
0.942	0.060	0.906	0.942	0.923	0.981	0.971	okuma
0.884	0.019	0.950	0.884	0.916	0.985	0.969	oyun oynama
0.920	0.043	0.921	0.920	0.920	0.985	0.973	ortalama

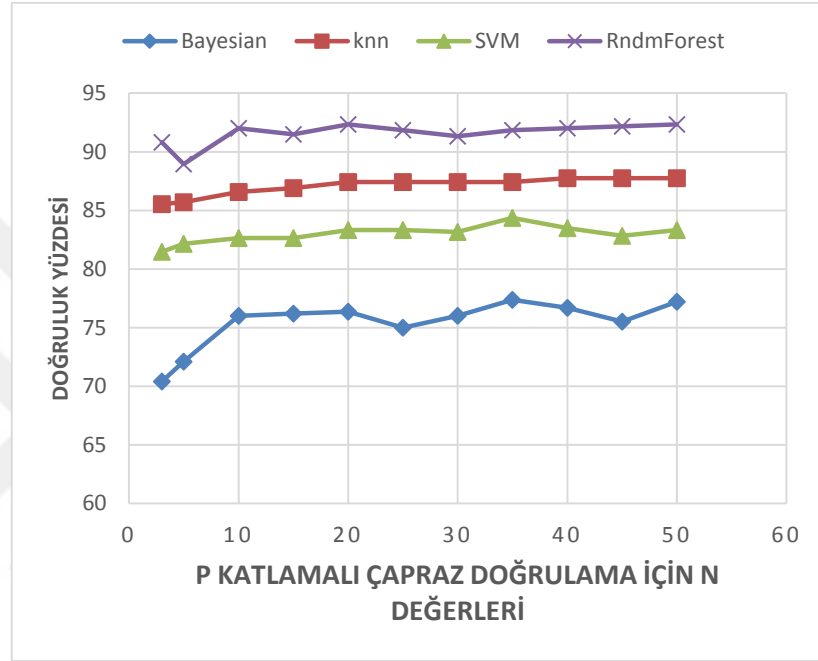
Tablo 4.8 Rastgele Orman yönteminin Karışıklık (Confusion) matrisi

Sistem tarafından tahmin edilen sınıf				
a	b	c		
178	11	3	a= video izleme	Gerçek sınıf
8	211	5	b= okuma	
9	11	152	c= oyun oynama	

Çalışmamızda kullanılan sınıflandırma yöntemlerinin doğruluk yüzdeleri p katlamalı çapraz doğrulama yöntemindeki farklı p değerleri için hesaplanmıştır. Farklı p değerleri için sınıflandırma yapılarak metotların hangi p değeri için daha iyi sonuç verdiği gözlemlenerek p değerinin en uygun değeri deneysel olarak elde edilmiştir. Elde edilen sonuçlar Şekil 4.4' de grafiksel olarak verilmiştir. Grafiğe göre p' nin artan değerlerine karşılık doğruluk yüzdesinin belirgin bir artış göstermediği söylenebilmektedir. P değeri aynı zamanda metotların çalıştırılmasında iterasyon sayısını temsil ettiği için büyük p değerleri özellikle SVM metodunun hesaplama

maliyetini arttırdığı için maliyet ve performans açısından iyi sonuç veren $k=10$ değeri seçilmiştir.

Tablo 4.9’ da kullanılan sınıflandırma algoritmalarının doğruluk yüzdeleri (accuracy) verilmiştir. Buna göre en iyi sonucun Rastgele Orman yönteminde elde edildiği görülürken, Bayesian ağ yönteminin sınıflandırmada diğer metotlara oranla başarılı olmadığı görülmektedir.



Şekil 4.4 Farklı p değerleri için metotların doğruluk yüzdeleri

Tablo 4.9 Tüm yöntemlerin doğruluk yüzdeleri

Sınıflandırıcı	Konum+Frekans bazlı özellikler
Bayesian Ağ	%77.040
SVM	%82.653
k-NN	%86.734
Rastgele Orman	%92.007

Tablo 4.10’ da ise çalışmada kullanılan farklı özellik seçim yöntemi ve bu yöntemlerin seçtiği en etkili özellikler sıralanmıştır. Aktiviteleri sınıflandırma işlemi

seçilen özellikler ile tekrar yapılmıştır. OneRAttribute ve ReliefAttributeEval yöntemleri sonucunda 40 tane özellik sınıflandırmadaki etki oranlarına göre sıralanmıştır. Bu sıralama sonucunda en etkili 10 özellik Tablo 4.10’ da gösterildiği gibi seçilmiştir. CfsSubsetEval ve PCA yöntemleri ise etkin sınıflandırma için kullanılması yeterli özellikleri seçmiştir.

Tüm sınıflandırma algoritmaları özellik seçim yöntemleri sonucunda belirlenen özellikler ile beraber çalıştırılıp Ek A, Ek B, Ek C ve Ek D içerisinde detaylı sonuçları ve karışıklık matrisleri verilmiştir. Aşağıda her bir algoritma için sonuçlarda meydana gelen dikkate değer görülen farklılıklardan bahsedilmiştir.

Bayesian Ağlar sınıflandırma algoritması her bir özellik seçim yönteminin sonucunda elde edilen özellikler ile tek tek çalıştırıldığı zaman elde edilen sonuçlar Ek A içerisinde bulunmaktadır. Tablo A.1 ‘de her bir özellik seçim yönteminin ve tüm özellik seçim yönteminin karışıklık matrislerini birlikte gösteren bir tablo bulunmaktadır. Bu tabloya bakıldığı zaman tüm aktiviteler için doğru tahmin edilme sayısında bir düşüş olduğu görülmektedir. OneRAttribute ve CfsSubsetEval için video izleme ve okuma aktiviteleri yaklaşık olarak aynı sayıda doğru tahmin edilmiştir. Tablo A.2 ve Tablo A.4’e bakıldığı zamanda bu aktiviteler için TP ve FP oranlarının birbirlerine yakın olduğunu görülmektedir. Oyun oynama aktivitesi ise CFSSubsetEval dışındaki tüm özellik seçim yöntemleri ile doğru tahmin sayısı 133 olarak belirlenmiştir. CFSSubsetEval yönteminde bu sayı 118’e kadar düşmektedir. Tablo A.4’de oyun oynama için TP oranınının 0.686 ya düşmesi de bunu göstermektedir.

Tablo 4.10 Özellik seçim yöntemleri ile seçilen en etkili özellikler

OneRAttribute	ReliefAttributeEval	CfsSubsetEval	PCA
X2min	Y2max	X1max	X1max
X2max	Y1max	X2max	Y1max
X1max	X2min	X1min	X2max
X1min	Y1min	X2min	Y2max
Y1min	X2max	Y2min	X1min
Y2min	Y2min	Y2entropy	Y1min
Y1max	X1min		X2min
Y2max	X1max		Y2min
Y2skewness	X2entropy		
Y1std	X1entropy		

SVM algoritmasının özellik seçim yöntemleri ile beraber çalıştırıldığı zaman elde edilen sonuçlar Ek B içerisinde yer almaktadır. Bu algoritma için Tablo B.1’de karışıklık matrisleri sonuçları bulunmaktadır. Bu tabloya bakıldığı zaman okuma ve video izleme aktiviteleri tüm özellik seçim yöntemlerine olumlu tepki verdiği görülmektedir. Oyun oynama aktivitesinde ise OneRAttribute ve ReliefAttributeEval yöntemlerine olumlu tepki verirken CfsSubsetEval ve PCA yöntemlerine olumsuz tepki verdiği görülmektedir. Tüm özellik seçim yöntemleri ile SVM algoritması daha iyi sonuç üretirken %89.5 değeri ile ReliefAttributeEval yönteminde en başarılı sonucu verdiği görülmektedir.

k-NN algoritmasının özellik seçim yöntemleri ile beraber çalıştırıldığı zaman elde edilen detaylı sonuçlar Ek C’de verilmiştir. Tablo C.1’ de görülen karışıklık matrisi sonuçlarına bakıldığı zaman tüm aktivitelerin özellik seçim yöntemlerine olumlu bir tepki verdiği görülmektedir. Tüm aktiviteler için elde edilen doğru tahmin sayısının tüm özellik seçim yöntemleri için yaklaşık olarak aynı olduğu görülmektedir. Tablo C.2, Tablo C.3, Tablo C.4 ve Tablo C.5 ‘e bakıldığı zaman buradaki TP ve FP oranlarının da tüm aktiviteler için birbirlerine yakın olduğu görülmektedir.

Son olarak Ek D’de rastgele orman algoritmasının özellik seçim yöntemleri ile beraber elde edilen detaylı sonuçları bulunmaktadır. Tablo D.1’e bakıldığı zaman video izleme ve okuma aktiviteleri için özellik seçim yöntemleri ile daha iyi sonuçlar alındığı görülmektedir. Tüm özellik seçim yöntemlerinin bu aktiviteler için birbirleri yakın tepkiler verdiği sonucuna ulaşılabilir. Fakat oyun oynama için OneRAttribute ve ReliefAttributeEval yöntemleri ile birbirlerine yakın ve daha iyi sonuçlar elde edilirken CfsSubsetEval ve PCA yöntemleri için birbirlerine yakın ve daha kötü sonuçlar elde edilmiştir. Tablo D.2, Tablo D.3, Tablo D.4 ve Tablo D.5’de de görüldüğü gibi OneRAttribute, ReliefAttribute ve CfsSubsetEval için elde edilen TP ve FP miktarları birbirlerine yakinken PCA için TP miktarının diğer yöntemlere göre bir miktar daha düşük ve FP miktarının bir miktar daha yüksek çıktığı görülmektedir.

Yapılan çalışmada eklerde gösterilen tablolara bakıldığı zaman kNN ve SVM algoritmaları için özellik seçim yöntemleri ile beraber daha iyi sonuç alındığı sonucuna ulaşılmaktadır. Bu sonuç kNN ve SVM algoritmaları için sınıflandırmada etkisi olmayan ya da çok az olan özelliklerin sınıflandırmayı zorlaştırdığı görülmektedir. Bayesian ağlar ve rastgele orman algoritmalarında ise sınıflandırma işlemi özellik

seçim yöntemlerinin etkilerinin çok daha az olduğu görülmektedir. Genel olarak özellik seçim yöntemlerinin bu algoritmalara etkisinin olumsuz olduğu görülmektedir.

Tablo 4.11’ de ise özellik seçim yöntemlerinin seçtikleri özellikleri kullanarak belirtilen sınıflandırma yöntemleri ile elde edilen ortalama doğruluk yüzdeleri verilmiştir. Elde edilen sonuçlarda görüldüğü gibi çalışmadaki doğruluk yüzdesi en yüksek olan Rastgele Orman sınıflandırma algoritmasının PCA dışındaki özellik seçim yöntemlerinin sonuçlarıyla beraber daha yüksek doğruluk yüzdesi verdiği görülmüştür. Bayesian ağlarda ise ReliefAttributeEval methodunda çok ufak bir iyileşme olmakla beraber diğer üç özellik seçim yöntemiyle daha düşük sonuçlar elde edilmiştir. k-NN ve SVM sınıflandırma algoritmalarının ise tüm özellik seçim yöntemlerine olumlu tepki verdiği görülmüştür. SVM algoritmasının ReliefAttributeEval özellik seçim yöntemiyle beraber başarı oranını yaklaşık %7 arttırdığı tespit edilmiştir. Bu değer algoritmaların özellik seçim yöntemlerine verdiği en yüksek tepki olduğu tabloda görülmektedir. Yapılan çalışmada rastgele orman algoritması ReliefAttributeEval yöntemi ile en başarılı sonuç elde edilmiştir. Fakat aradaki fark tüm özellikler ile elde edilen sonuca göre çok ufak kalmıştır. Bu durum da özellik seçim yönteminin kullanılmasının işlem maliyeti göz önüne alınarak uygulanmaması düşünülmüştür. Rastgele orman algoritmasının tüm özellik seçim yöntemleriyle elde edilen %92.007’lik sonucu çalışmada alınan en başarılı sonuç olarak kabul edilmiştir.

Üç aktiviteyi sınıflandırmak için kullanılan 4 farklı sınıflandırma algoritmasının sonuçları incelenmiş ve en başarısız sonucu veren algoritmanın Bayesian ağ algoritması olduğu sonucuna ulaşılmıştır. En başarılı algoritmanın ise

Tablo 4.11 Algoritmaların özellik seçim yöntemlerinden sonraki sonuçları

Sınıflandırıcı	Özellik Seçim Yöntemleri				Tüm Özellikler
	OneRAttribute	ReliefAttributeEval	CfsSubsetEval	PCA	
Bayesian Ağ	76.190	77.408	73.979	76.530	77.040
SVM	87.925	89.455	87.755	86.054	82.653
k-NN	90.816	89.115	88.095	89.795	86.734
Rastgele Orman	91.666	92.176	90.816	89.455	92.007

%92.007 ile Rastgele Orman algoritması olduğu görülmüştür. Yapılan bu çalışmada kullanılan Bayesian Ağlar, SVM, k-NN ve Rastgele Orman algoritmalarının kıyaslanmasında kullanılan doğruluk değerlerinin t-test ile istatistiksel olarak anlamlı olup olmadığı incelenmiştir.

Yapılan çalışmada sınıflandırma algoritmaları arasında en başarılı sonuçlar rastgele orman algoritmasıyla elde edilmiştir. Elde edilen bu sonucun istatistiksel olarak anlamlı olup olmadığını incelemek için hipotez kabul testi gerçekleştirilmiştir. Tablo 4.12’de rastgele orman algoritması verilerin %90’ı eğitim verisi %10’u test verisi olacak şekilde 10 kez çalıştırılmış ve elde edilen sonuçlar gösterilmiştir. Elde edilen bu sonuçlarla denklem 2.1’e göre standart olasılık değeri hesaplanmıştır. Bu işlem sonucunda elde edilen z değeri 0.285 çıkmıştır. Z tablosundaki 0.05 anlamlılık için elde edilmesi gereken değer olan 1.96 değerinden daha küçük bir değer elde edilmiştir. Bu sonuç yapılan çalışmada rastgele orman algoritması için elde edilen sonucun istatistiksel olarak anlamlı olduğunu göstermektedir.

Çalışmada kullanılan dört algoritmanın sonuçlarının her birinin diğer algoritmalarla aralarında oluşan farkların istatistiksel olarak anlamlı olup olmadığı incelenmiştir. Alınan sonuçlar farkların istatistiksel olarak anlamlı olduğunu göstermiştir.

Tablo 4.12 Rastgele orman algoritması için hipotez kabul testi sonuçları

Eğitim Verisi	Test Verisi	Doğru Bilinen	Yanlış Bilinen	Doğru Yüzdesi	Yanlış Yüzdesi
529	59	55	4	93.220	6.779
529	59	51	8	86.440	13.559
530	58	54	4	93.103	6.896
530	58	55	3	94.827	5.172
529	59	54	5	91.525	8.474
529	59	54	5	91.525	8.474
530	58	56	2	96.551	3.448
530	58	57	1	98.275	1.724
530	58	55	3	94.827	5.172
528	60	52	8	86.135	13.333

BEŞİNCİ BÖLÜM

TARTIŞMA VE ÖNERİLER

Yapılan bu çalışmada bilgisayar karşısındaki kişinin göz hareketlerine bakılarak belirlenen üç farklı aktivitenin (okuma, izleme oyun oynama) farklı sınıflandırma algoritmaları kullanılarak tahmini amaçlanmıştır.

Tezin başında belirlenen araştırma sonuçlarıyla ilgili şu sonuçlar elde edilmiştir:

- Düşük çözünürlük kamera ile kayıt altına alınan deneklerin göz bebeklerini en yüksek doğruluk payı ile belirlemek için Fuhl'un algoritması kullanılmıştır. Bu algoritma göz koordinatı hata miktarı arttıkça %100 yakın bir doğruluk ile sonuç dönmekte ve diğer algoritmalar ile kıyaslandığı zaman göz bebeği koordinatlarındaki düşük hata payları için daha başarılı sonuçlar dönmektedir.
- Deneklerin aktiviteleri gerçekleştirim esnasındaki belirli bir sürelik kayıt üzerinde belirlenen göz bebeği koordinatlarındaki değişim miktarları kullanılarak standart sapma, varyans, ortalama vs. gibi istatistiksel veriler sınıflandırma için kullanılacak özellikler olarak belirlenmiştir. Bunun yanında göz bebeğinin aktiviteyi gerçekleştirim esnasındaki uç nokta değerleri de özellik olarak seçilmiştir. Göz bebeği hareketlerindeki değişim miktarının Fourier dönüşümü yapılarak sinyalin enerji büyüklüğü de frekans alanında tanımlı özellik olarak seçilmiştir.
- Belirlenen bu özellikler ile çalıştırılan Bayesian ağlar, SVM, k-NN ve Rastgele Orman sınıflandırma algoritmalarından en iyi sonucu yaklaşık %92 ile Rastgele Orman algoritması vermiştir.
- Sınıflandırma için kullanılan özellikler üzerinde OneRAttribute, ReliefAttributeEval, CfsSubsetEval ve PCA özellik seçim yöntemleri çalıştırılmıştır. Özellik seçim yöntemlerinin sınıflandırma algoritmaları üzerindeki etkisi incelenmiştir. Algoritmalar arasında özellik seçim yöntemlerine en fazla tepkiyi SVM algoritması ReliefAttributeEval özellik seçim yöntemiyle beraber vermiştir.

- kNN ve SVM algoritmaları tüm özellik seçim yöntemleriyle beraber daha başarılı sonuçlar üretilirken, rastgele orman ve bayesian ağlar algoritmaları için ReliefAttributeEval özellik seçim yöntemi dışında genel olarak daha başarısız sonuçlar elde edilmiştir. ReliefAttributeEval yönteminin ise bu algoritmaların sonuçlarına olan olumlu etkisinin çok düşük kaldığı görülmüştür.
- Tüm özellik seçim yöntemlerinin sonuçları değerlendirerek hepsinde öne çıkan ortak özelliklerin uç noktalar olduğu gözlenmiştir. Bu özellikler kullanılarak çalıştırılan Bayesian Ağlar, SVM ve k-NN sınıflandırma algoritmalarında doğruluk yüzdelerinin arttığı gözlemlenmişken en yüksek doğruluk yüzdesinde bu oranın düştüğü gözlenmiştir. Aktivite sınıflandırmak için bu özellikler üzerinde özellik seçim yöntemlerinin çalıştırılmasına gerek olmadığı görülmüştür.
- Çalışmada en başarılı sınıflandırma algoritması olan rastgele orman algoritmasında alınan sonuçların istatistiksel olarak anlamlı olduğu belirlenmiştir.
- Deneysel sonuçların istatistiksel olarak anlamlılığı yapılan ikili t-testi ile doğrulanmıştır.

Gerçekleştirilen bu çalışma bilgisayar-insan etkileşimi alanında önemli bir temel oluşturmuştur. Bu çalışma ile kişilerin göz bebeği hareketlerindeki değişimin hangi aktivite ile gerçekleştiği sorusuna bir cevap aramıştır. İleride yapılacak çalışmalarda aktivite sınıflandırmanın sadece göz bebeği hareketleri ile değil diğer uzuvlarda takip edilerek gerçekleştirilebilir.

Aktiviteleri sınıflandırmak için kullanılan sınıflandırma algoritmaları dışındaki algoritmalarında ürettiği sonuçlar incelenebilir. Sınıflandırma için kullanılan özelliklere yeni özellikler belirlenerek çalışmanın doğruluk yüzdesini yukarı çekmeye çalışmak bu alanda yapılacak bir başka çalışma konusudur.

Bir başka çalışma alanı gerçekleştirilen aktivite sayısını arttırmak olabilir. Bunun yanında yapılan çalışmada kayıt altına alınan aktiviteler belirli işlemlerden geçirilerek sonuçlar elde edilmektedir. Bu sebeple çalışma gerçek zamanlı olarak çalışmamaktadır. Bu çalışmanın ileride gerçek zamanlı olarak sonuç üretmesi sağlanabilir.

Bu çalışma bir model çıkaramadığınız durumlar için kullanılacak bir yöntem sunmaktadır. Yukarıda bahsedilen çalışma konuları için bir temel teşkil eden bu çalışmada kullanılan yöntemler çeşitlendirilerek ve değiştirilerek önerilen çalışmalarda başarılı sonuçlar alınabilir.



KAYNAKÇA

- [1] Barea, R., Boquete, L., Bergasa, L. M., López, E., & Mazo, M. (2003). Electro-oculographic guidance of a wheelchair using eye movements codification. *The International Journal of Robotics Research*, 22 (7-8), 641-652.
- [2]<http://www.ozerkavalcioğlu.com/tanisaal-islemler/eog-elektrookulografi> (30.11.2016)
- [3] Robinson, D. A. (1963). A method of measuring eye movement using a scieral search coil in a magnetic field. *IEEE Transactions on bio-medical electronics*, 10(4), 137-145.
- [4] Kranz, Y., Ophir, Y., Shamir, H., & Yaeli, Y. (2008). *U.S. Patent No. 7,401,920*. Washington, DC: U.S. Patent and Trademark Office.
- [5] Franchak, J. M., Kretch, K. S., Soska, K. C., & Adolph, K. E. (2011). Head-mounted eye tracking: A new method to describe infant looking. *Child development*, 82(6), 1738-1750.
- [6] Poole, A., & Ball, L. J. (2006). Eye tracking in HCI and usability research. *Encyclopedia of human computer interaction*, 1, 211-219.
- [7] Morimoto, C. H., Koons, D., Amir, A., & Flickner, M. (2000). Pupil detection and tracking using multiple light sources. *Image and vision computing*, 18(4), 331-335.
- [8] Zhu, Z., Ji, Q., Fujimura, K., & Lee, K. (2002). Combining Kalman filtering and mean shift for real time eye tracking under active IR illumination. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (Vol. 4, pp. 318-321). IEEE.
- [9] Asteriadis, S., Nikolaidis, N., Hajdu, A., & Pitas, I. (2006, March). An eye detection algorithm using pixel to edge information. In *Int. Symp. on Control, Commun. and Sign. Proc.*
- [10] Valenti, R., & Gevers, T. (2008, June). Accurate eye center location and tracking using isophote curvature. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* (pp. 1-8). IEEE.
- [11] Cristinacce, D., Cootes, T. F., & Scott, I. M. (2004, September). A Multi-Stage Approach to Facial Feature Detection. In *BMVC* (pp. 1-10).
- [12] Illingworth, J., & Kittler, J. (1988). A survey of the Hough transform. *Computer vision, graphics, and image processing*, 44(1), 87-116.

- [13] Young, D., Tunley, H., & Samuels, R. (1995). *Specialised hough transform and active contour methods for real-time eye tracking*. University of Sussex, Cognitive & Computing Science.
- [14] Smereka, M., & Dulęba, I. (2008). Circular object detection using a modified Hough transform. *International Journal of Applied Mathematics and Computer Science*, 18(1), 85-91.
- [15] Sewell, W., & Komogortsev, O. (2010, April). Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems* (pp. 3739-3744). ACM.
- [16] Timm, F., & Barth, E. (2011). Accurate Eye Centre Localisation by Means of Gradients. *VISAPP*, 11, 125-130.
- [17] D'Orazio, T., Ancona, N., Cicirelli, G., & Nitti, M. (2002). A ball detection algorithm for real soccer image sequences. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (Vol. 1, pp. 210-213). IEEE.
- [18] Wang, J. G., Sung, E., & Venkateswarlu, R. (2003, October). Eye gaze estimation from a single image of one eye. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on* (pp. 136-143). IEEE.
- [19] Murphy, K. (1998). A brief introduction to graphical models and Bayesian networks.
- [20] Pearl, Judea. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 2014.
- [21] <http://www.eng.tau.ac.il/~bengal/BN.pdf> 06.12.2016
- [22] Aksoy, S. (2006). Parametric Models: Bayesian Belief Networks. Lecture Notes, Department of Computer Engineering Bilkent University.
- [23] Griffiths, T., & Yuille, A. (2008). A primer on probabilistic inference. The probabilistic mind: Prospects for Bayesian cognitive science, 33-57.
- [24] Pearl, J. (1987). Evidential reasoning using stochastic simulation of causal models. *Artificial Intelligence*, 32(2), 245-257.
- [25] Dagum, P., & Luby, M. (1993). Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial intelligence*, 60(1), 141-153.
- [26] Ge, G., Yun, K., Samaras, D., & Zelinsky, G. J. (2015, June). Action classification in still images using human eye movements. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 16-23). IEEE.

- [27] Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3), 372.
- [28] Kinsler, V., & Carpenter, R. H. S. (1995). Saccadic eye movements while reading music. *Vision Research*, 35(10), 1447-1458.
- [29] Buswell, G. T. (1935). How people look at pictures: a study of the psychology and perception in art.
- [30] Yarbus, A. L. (1967). *Eye movements during perception of complex objects* (pp. 171-211). Springer US.
- [31] Castelhana, M. S., Mack, M. L., & Henderson, J. M. (2009). Viewing task influences eye movement control during active scene perception. *Journal of Vision*, 9(3), 6-6.
- [32] Koornneef, A., & Mulders, I. (2016). Can We 'Read' the Eye-Movement Patterns of Readers? Unraveling the Relationship Between Reading Profiles and Processing Strategies. *Journal of psycholinguistic research*, 1-18.
- [33] Mathe, S., & Sminchisescu, C. (2012). Dynamic eye movement datasets and learnt saliency models for visual action recognition. In *Computer Vision—ECCV 2012* (pp. 842-856). Springer Berlin Heidelberg.
- [34] Mathe, S., & Sminchisescu, C. (2015). Actions in the eye: dynamic gaze datasets and learnt saliency models for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(7), 1408-1424.
- [35] Fuhl, W., Santini, T. C., Kübler, T., & Kasneci, E. (2016, March). Ellipse selection for robust pupil detection in real-world environments. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications* (pp. 123-130). ACM.
- [36] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on* (Vol. 1, pp. I-511). IEEE.
- [37] Eibe Frank, Mark A. Hall, and Ian H. Witten (2016). The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Fourth Edition, 2016.
- [38] Vapnik V., "The nature of statistical learning theory," Springer-Verlag, New-York, 1995

- [39] Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov), 45-66.
- [40] Breiman L: Random forests. *Machine Learning* 2001, 45: 5–32.
- [41] R. Shapire, Y. Freund, P. Bartlett, and W. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *Annals of Statistics*, 26(5):1651–1686, 1998.
- [42] L. Breiman. Bagging predictors. *Machine Learning*, 24 (2):123–140, 1996.
- [43] Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." *R news* 2.3 (2002): 18-22.
- [44] Lu, Yijuan, et al. "Feature selection using principal feature analysis." *Proceedings of the 15th ACM international conference on Multimedia*. ACM, 2007.
- [45] Powers, David Martin. "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation." (2011).
- [46] Davis, Jesse, and Mark Goadrich. "The relationship between Precision-Recall and ROC curves." *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006.
- [47] Young, I. T., Gerbrands, J. J., & Van Vliet, L. J. (1998). *Fundamentals of image processing*. The Netherlands: Delft University of Technology.
- [48] <https://www.bioid.com/About/BioID-Face-Database> (15.12.2016).
- [49] Hsu, Chih-Wei, and Chih-Jen Lin. "A comparison of methods for multiclass support vector machines." *IEEE transactions on Neural Networks* 13.2 (2002): 415-425.
- [50] Wang, Zhe, and Xiangyang Xue. "Multi-class support vector machine." *Support Vector Machines Applications*. Springer International Publishing, 2014. 23-48.
- [51] Kul, S. (2014). İSTATİSTİK SONUÇLARININ YORUMU: P DEĞERİ VE GÜVEN ARALIĞI NEDİR?. *Bulletin of Pleura/Plevra Bülteni*, 8(1).
- [52] Platt, J. C. (1999). 12 fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods*, 185-208.
- [53] http://acikders.ankara.edu.tr/pluginfile.php/1386/mod_resource/content/3/B13_Hipotez%20Testi.pdf 2.15.2017

[54] http://fbe.beun.edu.tr/fbe/images/ins729/sunu_4ci_hafta.pdf 2.15.2017

[55] eng.harran.edu.tr/~hbulut/ZTablosu.doc 2.15.2017



EKLER

- 1. Ek A - Özellik Seçim Yöntemleri ile Belirlenen Özellikler ile Bayesian Ağlar Sınıflandırma Algoritması Sonuçları**
- 2. Ek B - Özellik Seçim Yöntemleri ile Belirlenen Özellikler ile SVM Sınıflandırma Algoritması Sonuçları**
- 3. Ek C - Özellik Seçim Yöntemleri ile Belirlenen Özellikler ile kNN Sınıflandırma Algoritması Sonuçları**
- 4. Ek D - Özellik Seçim Yöntemleri ile Belirlenen Özellikler ile Rastgele Orman Sınıflandırma Algoritması Sonuçları**

EK A

Tablo A.1 Tüm Özellikler ile Özellik Seçim Yöntemlerinin Bayesian Ağlar Sınıflandırma Algoritması İçin Karışıklık Matrisleri Kıyaslaması

OneRAttribute			ReliefAttributeEval			CfsSubsetEval			PCA			Tüm Özellikler				
a	b	c	a	b	c	a	b	c	a	b	c	a	b	c		
140	30	22	144	27	21	140	30	22	142	28	22	169	16	7	izleme	Gerçek sınıf
25	175	24	24	176	24	23	177	24	25	175	24	18	192	14	okuma	
15	24	133	16	23	133	20	34	118	15	24	133	7	16	149	oyun oynama	

Tablo A.2 OneRAttribute Yöntemi ile Bayesian Ağların Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.729	0.101	0.778	0.729	0.753	0.926	0.879	izleme
0.781	0.148	0.764	0.781	0.773	0.910	0.859	okuma
0.773	0.111	0.743	0.773	0.758	0.918	0.859	oyun oynama
0.762	0.122	0.762	0.762	0.762	0.917	0.866	ortalama

Tablo A.3 ReliefAttributeEval Yöntemi ile Bayesian Ağların Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.750	0.101	0.783	0.750	0.766	0.928	0.882	izleme
0.786	0.137	0.779	0.786	0.782	0.912	0.864	okuma
0.773	0.108	0.747	0.773	0.760	0.918	0.859	oyun oynama
0.770	0.117	0.771	0.770	0.770	0.919	0.868	ortalama

Tablo A.4 CFSSubsetEval Yöntemi ile Bayesian Ağların Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.729	0.109	0.765	0.729	0.747	0.925	0.869	izleme
0.790	0.176	0.734	0.790	0.761	0.901	0.848	okuma
0.686	0.111	0.720	0.686	0.702	0.889	0.812	oyun oynama
0.740	0.135	0.740	0.740	0.739	0.905	0.844	ortalama

Tablo A.5 PCA Yöntemi ile Bayesian Ağların Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.740	0.101	0.780	0.740	0.759	0.926	0.880	izleme
0.781	0.143	0.771	0.781	0.776	0.910	0.860	okuma
0.773	0.111	0.743	0.773	0.758	0.918	0.859	oyun oynama
0.765	0.120	0.766	0.765	0.765	0.918	0.866	ortalama

EK B

Tablo B.1 Tüm Özellikler ile Özellik Seçim Yöntemlerinin SVM Sınıflandırma Algoritması İçin Karışıklık Matrisleri Kıyaslaması

OneRAttribute			ReliefAttributeEval			CfsSubsetEval			PCA			Tüm Özellikler				
a	b	c	a	b	c	a	b	c	a	b	c	a	b	c		
166	12	14	172	12	8	170	16	6	169	14	9	152	20	20	a = video	Gerçek sınıf
8	201	15	10	201	13	10	202	12	8	204	12	14	188	22	b = okuma	
7	15	150	7	12	153	9	19	144	15	24	133	10	16	146	c = oyun	

Tablo B.2 OneRAttribute Yöntemi ile SVM Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.865	0.038	0.917	0.917	0.890	0.929	0.859	izleme
0.897	0.074	0.882	0.882	0.889	0.916	0.837	okuma
0.872	0.070	0.838	0.872	0.855	0.909	0.778	oyun oynama
0.879	0.061	0.880	0.879	0.879	0.918	0.827	ortalama

Tablo B.3 ReliefAttributeEval Yöntemi ile SVM Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.896	0.043	0.910	0.896	0.903	0.948	0.869	izleme
0.897	0.066	0.893	0.897	0.895	0.919	0.847	okuma
0.890	0.050	0.879	0.890	0.884	0.929	0.825	oyun oynama
0.895	0.054	0.895	0.895	0.895	0.932	0.848	ortalama

Tablo B.4 CFSSubsetEval Yöntemi ile SVM Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.885	0.048	0.899	0.885	0.892	0.938	0.849	izleme
0.902	0.096	0.852	0.902	0.876	0.905	0.812	okuma
0.837	0.043	0.889	0.837	0.862	0.924	0.812	oyun oynama
0.878	0.065	0.878	0.878	0.877	0.921	0.824	ortalama

Tablo B.5 PCA Yöntemi ile SVM Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.880	0.058	0.880	0.880	0.880	0.936	0.836	izleme
0.911	0.104	0.843	0.911	0.876	0.911	0.810	okuma
0.773	0.050	0.864	0.773	0.816	0.886	0.753	oyun oynama
0.861	0.074	0.861	0.861	0.860	0.912	0.802	ortalama

EK C

Tablo C.1 Tüm Özellikler ile Özellik Seçim Yöntemlerinin kNN Sınıflandırma Algoritması İçin Karışıklık Matrisleri Kıyaslaması

OneRAttribute			ReliefAttributeEval			CfsSubsetEval			PCA			Tüm Özellikler				
a	b	c	a	b	c	a	b	c	a	b	c	a	b	c		
175	14	3	175	14	3	177	12	3	180	8	4	169	16	7	a= video izleme	Gerçek sınıf
8	204	12	19	197	8	17	197	10	12	197	15	18	192	14	b= okuma	
7	10	155	5	15	152	8	12	152	17	14	151	7	16	149	c= oyun oynama	

Tablo C.2 OneRAttribute ile kNN Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.911	0.038	0.921	0.911	0.916	0.985	0.973	izleme
0.911	0.066	0.895	0.911	0.903	0.970	0.958	okuma
0.901	0.036	0.912	0.901	0.906	0.975	0.946	oyun oynama
0.908	0.048	0.908	0.908	0.908	0.977	0.959	ortalama

Tablo C.3 ReliefAttributeEval ile kNN Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.911	0.061	0.879	0.911	0.895	0.963	0.943	izleme
0.879	0.080	0.872	0.879	0.876	0.948	0.929	okuma
0.884	0.026	0.933	0.884	0.907	0.968	0.939	oyun oynama
0.891	0.058	0.892	0.891	0.891	0.959	0.937	ortalama

Tablo C.4 CFSSubsetEval ile kNN Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.922	0.063	0.876	0.922	0.898	0.971	0.953	izleme
0.879	0.066	0.891	0.879	0.885	0.955	0.935	okuma
0.884	0.031	0.921	0.884	0.902	0.966	0.940	oyun oynama
0.895	0.055	0.895	0.895	0.895	0.963	0.942	ortalama

Tablo C.5 PCA ile kNN Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.938	0.048	0.905	0.938	0.921	0.987	0.981	izleme
0.879	0.060	0.900	0.879	0.889	0.967	0.952	okuma
0.878	0.046	0.888	0.878	0.883	0.971	0.944	oyun oynama
0.898	0.052	0.898	0.898	0.898	0.975	0.959	ortalama

EK D

Tablo D.1 Tüm Özellikler ile Özellik Seçim Yöntemlerinin Rastgele Orman Sınıflandırma Algoritması İçin Karışıklık Matrisleri Kıyaslaması

OneRAttribute			ReliefAttributeEval			CfsSubsetEval			PCA			Tüm Özellikler				
a	b	c	a	b	c	a	b	c	a	b	c	a	b	c		
178	10	4	180	9	3	178	11	3	179	6	7	175	13	4	a = video	Gerçek sınıf
8	211	5	9	211	4	9	210	5	11	205	8	9	208	7	b =	
6	14	152	4	14	154	5	19	148	12	16	144	4	17	151	c = oyun	

Tablo D.2 OneRAttribute ile Rastgele Orman Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.927	0.035	0.927	0.927	0.927	0.990	0.982	izleme
0.942	0.066	0.898	0.942	0.919	0.981	0.970	okuma
0.884	0.022	0.944	0.884	0.913	0.983	0.968	oyun oynama
0.920	0.043	0.921	0.920	0.920	0.985	0.973	ortalama

Tablo D.3 ReliefAttributeEval ile Rastgele Orman Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.938	0.033	0.933	0.938	0.935	0.989	0.982	izleme
0.942	0.063	0.902	0.942	0.921	0.983	0.973	okuma
0.895	0.017	0.957	0.895	0.925	0.987	0.973	oyun oynama
0.927	0.040	0.928	0.927	0.927	0.986	0.976	ortalama

Tablo D.4 CFSSubsetEval ile Rastgele Orman Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.927	0.035	0.927	0.927	0.927	0.983	0.976	izleme
0.938	0.082	0.875	0.938	0.905	0.978	0.964	okuma
0.860	0.019	0.949	0.860	0.902	0.979	0.959	oyun oynama
0.912	0.049	0.914	0.912	0.912	0.980	0.966	ortalama

Tablo D.5 PCA ile Rastgele Orman Sonuçları

TP oranı	FP oranı	Kesinlik	Duyarlılık	F-ölçütü	ROC alanı	PRC alanı	Sınıf
0.932	0.058	0.886	0.932	0.909	0.986	0.976	izleme
0.915	0.060	0.903	0.915	0.909	0.979	0.968	okuma
0.837	0.036	0.906	0.837	0.870	0.973	0.949	oyun oynama
0.898	0.053	0.898	0.898	0.898	0.980	0.965	ortalama

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı: Çağrı Şener

Uyruğu: T.C.

Doğum Yeri ve Tarihi: Ankara-15.02.1987

Medeni Hali: Evli

Adres: Bağlıca Mah. Mermeroğlu Cad. Seğmen Bağlıca Konutları 57/A No:28
Etimesgut/Ankara

E-Posta Adresi: csener@havelsan.com.tr

İletişim (Telefon) : 0533 139 75 18



EĞİTİM

Lise: 50.Yıl Lisesi - 2004

Lisans: Başkent Üniversitesi Bilgisayar Mühendisliği (%100 Burslu) - 2010

Yüksek Lisans: Türk Hava Kurumu Üniversitesi Elektrik ve Bilgisayar
Mühendisliği – Devam Ediyor

MESLEKİ DENEYİM

HAVELSAN, Ankara

Uzman Yazılım Mühendisi, Haziran 2013 – Devam Ediyor

KATRON

Uzman Yazılım Mühendisi, Eylül 2012 - Haziran 2013

SimBT

Yazılım Mühendisi, Şubat 2011 – Eylül 2012

YABANCI DİL

İngilizce