

BURSA TEKNİK ÜNİVERSİTESİ ❖ FEN BİLİMLERİ ENSTİTÜSÜ

**İNSAN BİLGİSAYAR ETKİLEŞİMİ UYGULAMALARI İÇİN GÖZ MERKEZİNİN
BELİRLENMESİ VE KAFA POZU TAHMİNİ**



YÜKSEK LİSANS TEZİ

Fatih DURSUN

Mekatronik Mühendisliği Anabilim Dalı

OCAK 2020

BURSA TEKNİK ÜNİVERSİTESİ ❖ FEN BİLİMLERİ ENSTİTÜSÜ

**İNSAN BİLGİSAYAR ETKİLEŞİMİ UYGULAMALARI İÇİN GÖZ
MERKEZİNİN BELİRLENMESİ VE KAFA POZU TAHMİNİ**

YÜKSEK LİSANS TEZİ

**Fatih DURSUN
(181241782016)**

Mekatronik Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Gökhan GELEN

OCAK 2020

BTÜ, Fen Bilimleri Enstitüsü'nün 181241782016 numaralı Yüksek Lisans Öğrencisi Fatih DURSUN, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "İnsan Bilgisayar Etkileşimi Uygulamaları İçin Göz Merkezinin Belirlenmesi ve Kafa Pozu Tahmini" başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Doç. Dr. Gökhan GELEN**
Bursa Teknik Üniversitesi



Jüri Üyeleri : **Doç. Dr. Hakan GÜRKAN**
Bursa Teknik Üniversitesi



Doç. Dr. Özgür Turay KAYMAKÇI
Yıldız Teknik Üniversitesi



Savunma Tarihi : 2 Ocak 2020

FBE Müdürü: **Doç. Dr. Murat ERTAŞ**
Bursa Teknik Üniversitesi

.....
...../...../.....



Aileme ve sevdiklerime,

ÖNSÖZ

Tez yazım sürecindeki düzenlemeler ve değerlendirmeler konusunda desteklerinden dolayı değerli hocam Doç. Dr. Gökhan GELEN'e çok teşekkür ederim. Hayatımın her aşamasında olduğu gibi yüksek lisans sürecimde de bana olan inançlarımı ve desteklerini esirgemeyen aileme minnettarım.

Ocak 2020

Fatih DURSUN



İÇİNDEKİLER

Sayfa

ÖNSÖZ	v
İÇİNDEKİLER	vi
KISALTMALAR	vii
ÇİZELGE LİSTESİ	viii
ŞEKİL LİSTESİ	ix
ÖZET	x
SUMMARY	xi
1. GİRİŞ	1
1.1 Literatür Araştırması	2
1.1.1 Göz merkezinin belirlenmesi	2
1.1.2 Kafa pozunun belirlenmesi	4
1.2 Tezin Amacı ve Organizasyonu	6
2. GÖZ MERKEZİNİN KABACA BELİRLENMESİ	9
2.1 Yüzün Algılanması ve Göz Bölgelerinin Belirlenmesi.....	10
2.1.1 Haar-Benzeri özniteliklerin seçimi	10
2.1.2 İntegral resmin oluşturulması.....	11
2.1.3 Adaboost eğitimi	12
2.1.4 Basamaklı sınıflandırıcı	13
2.2 Göz Merkezini Kabaca Belirleme Algoritması	15
3. İYİLEŞTİRİLMİŞ GÖZ MERKEZİ BELİRLEME YAKLAŞIMI	18
3.1 John Daugman'ın Göz Merkezi Belirleme Yaklaşımı	19
3.2 Yeni Amaç Fonksiyonunun Oluşturulması	21
3.3 Önerilen Göz Merkezi Belirleme Yaklaşımının Deneysel Sonuçları	26
3.3.1 BioID veri seti	26
3.3.2 Deneysel Bulgular	27
3.3.3 Gerçek zamanlı sonuçlar	29
4. KAFA POZUNUN BELİRLENMESİ	32
4.1 Takip İçin Uygun Öznitelik Noktalarının Belirlenmesi	34
4.2 Pinhole Kamera Model.....	36
4.3 Modelin Başlatılması.....	39
4.4 Lucas-Kanade Optik Akış Algoritması ile Noktaların Takibi.....	42
4.5 Pozun Belirlenmesi	45
4.6 Kafa Pozu Tahmini Uygulamasının Deneysel Sonuçları	48
4.6.1 Kafa Pozu Veri Seti.....	48
4.6.2 Deneysel Bulgular ve Gerçek Zamanlı Sonuçlar	49
5. SONUÇ VE ÖNERİLER	52
KAYNAKLAR	53
ÖZGEÇMİŞ	58

KISALTMALAR

AGM	: Aktif Görünüm Modeli
AŞM	: Aktif Şekil Modeli
CHT	: Circular Hough Transform
EOG	: Electrooculography
HCI	: Human Computer Interaction
IR	: Infrared
PCA	: Principal Components Analysis
RANSAC	: Random Sample Consensus
3B	: 3 Boyutlu
2B	: 2 Boyutlu

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Yüz algılama aşamasında kullanılan Adaboost algoritması [45].	13
Çizelge 3.1 : Tez çalışmasında geliştirilen göz merkezi iyileştirme algoritması.	25
Çizelge 3.2 : En kötü göz, en iyi göz ve gözlerin ortalaması için doğruluk değerleri.	28
Çizelge 3.3 : Önerilen göz merkezi belirleme yönteminin diğer yöntemlerle karşılaştırılması.	28
Çizelge 3.4 : Göz belirleme algoritmasının tek ve dört çekirdek için işleme hızları.	30
Çizelge 3.5 : Farklı boyuttaki göz bölgeleri için işlem süreleri.....	31
Çizelge 4.1 : Bulunan doğruluk değerleri ve diğer metotlarla karşılaştırma sonuçları.	50
Çizelge 4.2 : Tasarlanan sistemin farklı aşamaları için geçen süreler.....	51

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Yüz algılama için belirlenen Haar-Benzeri öznitelik dikdörtgenleri [45].	11
Şekil 2.2 : İnsan yüzüne uygun seçilmiş öznitelik dikdörtgenleri [45].	11
Şekil 2.3 : İntegral resim üzerindeki bir öznitelik dikdörtgeni [45].	12
Şekil 2.4 : Basamaklı sınıflandırıcı [45].	14
Şekil 2.5 : Yüz ve göz bölgelerinin algılanmasıyla elde edilen sonuç resmi.	14
Şekil 2.6 : İris ve göz akını temsil eden yapay resim üzerinde gradyan ve mesafe vektörlerinin doğrultuları.	15
Şekil 3.1 : Kabaca belirlenmiş göz merkezleri etrafında oluşturulan arama bölgeleri.	19
Şekil 3.2 : Farklı merkez noktaları için oluşturulan daireler.	20
Şekil 3.3 : Faydalanılan göz merkezi belirleme yöntemlerinin ve tezde önerilen yöntemin tarama bölgeleri: (a) Timm ve Barth tarafından önerilen metotta, (b) J. Daugman tarafından önerilen metotta, (c) Tez çalışmasında geliştirilen göz merkezi belirleme metodunda.	22
Şekil 3.4 : Nokta çarpıma dahil edilen noktaları kapsayan açı aralığı.....	24
Şekil 3.5 : Göz merkezi belirleme işlemine ait bazı doğru sonuçlar.	27
Şekil 3.6 : Göz merkezi belirleme işlemine ait bazı yanlış sonuçlar.	28
Şekil 3.7 : Göz merkezi ve iris yarıçapı belirleme algoritmasıyla elde edilen bazı sonuçlar.	30
Şekil 4.1 : Harris köşe algılama algoritmasının yüz bölgesine uygulanmasıyla elde edilen öznitelik noktaları.	34
Şekil 4.2 : Köşe noktası, kenar noktası ve düz bölgedeki bir nokta için türev değerlerinin büyüklüğü.	36
Şekil 4.3 : Pinhole kamera modeli.	37
Şekil 4.4 : Resim düzlemi ve gerçek resim için referans noktaları.	38
Şekil 4.5 : Silindir kafa modelinin başlangıç durumu.	39
Şekil 4.6 : Özniteliklerin silindir kafa modeline aktarılması.	40
Şekil 4.7 : Ardışık resimlerle pozun belirlenmesi [9].	46
Şekil 4.8 : Kafa pozunu veri tabanıyla elde edilen bazı sonuç resimleri. Satırlar sırasıyla z eksenini etrafında dönme (Roll), x eksenini etrafında dönme (Pitch) ve y eksenini etrafında dönme (Yaw) durumlarını göstermektedir.	49
Şekil 4.9 : Elde edilen bazı gerçek zamanlı sonuç resimleri.	51

İNSAN BİLGİSAYAR ETKİLEŞİMİ UYGULAMALARI İÇİN GÖZ MERKEZİNİN BELİRLENMESİ VE KAFA POZU TAHMİNİ

ÖZET

Göz merkezinin belirlenmesi ve kafa pozunun tahmini, görüntü temelli insan bilgisayar etkileşimi uygulamaları için önemli bir problemdir. Bu tezde, düşük çözünürlüklü görüntülerden göz merkezi yerini ve baş pozunu tespit eden bir sistem önerilmiştir. Bu sistem kullanıcıya rahatsızlık verebilecek herhangi bir harici donanıma ihtiyaç duymadan, sadece tek kamera ile çalışabilmektedir. Göz merkezlerini belirlemek için, kameradan alınan görüntüye yüz algılama, göz alanı algılama, kaba göz merkezi lokalizasyonu ve hassas göz merkezi lokalizasyonundan oluşan çok aşamalı bir yaklaşım uygulanmaktadır. Tez kapsamında, göz merkezi belirleme çalışmalarında yaygın olarak kullanılan iki farklı yöntemin avantajlarından aynı anda faydalanacak yeni bir amaç fonksiyonu oluşturulmuştur. Kafa pozunun algılanması için kafanın rijit bir silindir olarak dikkate alındığı model tabanlı bir yaklaşım kullanılmıştır. Poz algılama işlemi, yüzdeki belirli öznitelik noktalarının ardışık resimler üzerinde takip edilmesine dayanmaktadır.

Göz merkezi tespiti ve kafa pozu tahmini için önerilen yöntemlerin her ikisi de, literatürde yaygın olarak kullanılan veri setleri üzerinde test edilmiştir. Benzer çalışmalar ile yapılan karşılaştırmalar, önerilen yöntemlerin doğruluk, hız ve uygulanabilirlik açısından önde olduğunu göstermektedir. Geliştirilen sistemin gerçek zamanlı çalışması hedeflendiği için veri setleri kullanılarak elde edilen sonuçların yanında, sistemin çalışma performansı gerçek zamanlı olarak da test edilmiştir. Önerilen sistem kullanıcıyı en az seviyede kısıtlamakta, ortam şartlarından en az seviyede etkilenmekte ve düşük çözünürlükteki resimlerde bile doğru ve hızlı sonuçlar üretebilmektedir.

Anahtar kelimeler: İnsan bilgisayar etkileşimi, Göz merkezi belirleme, Göz takibi, Kafa pozu tahmini, Silindir kafa modeli

EYE CENTER LOCALIZATION AND HEAD POSE ESTIMATION FOR HUMAN COMPUTER INTERACTION APPLICATIONS

SUMMARY

Detection of eye center and estimation of head pose is an important problem for image-based human-computer interaction applications. In this thesis, a system that detects eye center location and head pose from low-resolution images is proposed. The system can be operated with only one camera without any external equipment which may cause discomfort to the user. To determine eye centers, a multi-stage approach consisting of face detection, eye area detection, coarse eye center localization and precise eye center localization to the image taken from the camera is applied. Within the scope of the thesis, a new objective function has been proposed to benefit from the advantages of two different methods which are commonly used in eye center determination studies. A model-based approach is used for the detection of head pose where the head is considered as a rigid cylinder. Pose detection is based on tracking of specific feature points on the face from consecutive images.

Both of the proposed methods for eye center detection and head exposure estimation have been tested on data sets commonly used in the literature. Comparisons with similar studies show that the proposed methods are leading in terms of accuracy, speed, and applicability. Since the real-time operation of the developed system is aimed, in addition to the results obtained by using data sets, the operating performance of the system is also tested in real-time. The proposed system minimally restricts the users, is minimally influenced by the ambient conditions and can produce accurate and fast results even in low-resolution images.

Keywords: Human-Computer Interaction Eye center localization, Eye tracking, Head Pose Estimation, Cylinder Head Model

1. GİRİŞ

İnsan bilgisayar etkileşimi (Human Computer Interaction-HCI), insan ile bilgisayar arasındaki etkileşimi ve ilişkiyi ifade etmektedir. İnsan bilgisayar etkileşimi çalışmaları, geliştirilen uygulamaları daha kullanışlı, daha faydalı ve kullanıcı dostu hale getirmeyi hedeflemektedir. Bu çalışmaların amacı, başlarda kullanıcı dostu arayüzler tasarlamak olsa da bu durum zamanla değişerek grafik tasarımı, insan davranışları, yazılım mühendisliği, sosyoloji gibi birçok disiplinin bir arada bulunduğu, uygulama geliştirme alanı haline gelmiştir. İnsan bilgisayar etkileşimi uygulamalarının önemli amaçlarından birisi, herhangi bir ön bilgi gerektirmeden, bilgisayar arayüzlerinin tüm kullanıcılar tarafından kolayca kullanılabilmesini sağlamak ve bunun için gerekli etkileşim cihazlarını geliştirmektir. İnsan ile bilgisayar arasındaki etkileşimi kolaylaştırmak için fare ve klavye gibi cihazlar geliştirilmiş ve bu cihazların kullanıcı dostu olmasını sağlayan farklı özellikler eklenmiştir. Geliştirilen donanımlar, ampute olmayan bireyler için kolay ve kullanışlı olsa da ampute bireyler için insan ile bilgisayar arasındaki etkileşimi kolaylaştıracak çözümlerin ortaya konulması gerekmektedir. İnsanlar arası etkileşim için göz ve kafa hareketleri, el hareketleri ve ses çok önemlidir. Bu nedenle fare veya klavye gibi donanımları kullanamayacak haldeki ampute bireylerin bilgisayar ile etkileşimini kolaylaştırmak için gözlerden, ses sinyallerinden veya baş hareketlerinden alınacak verilerin kullanılması çok uygundur.

Göz merkezlerinin belirlenmesi ve kafa pozununun tahmin edilmesi, insan bilgisayar etkileşimi için kullanılan giriş cihazlarına alternatif oluşturacak ve daha geniş bir kitleyi kapsayacak yaklaşımların hayata geçirilmesini sağlamaktadır. Bazı sistemler, kafa pozunu algılama ve göz merkezi belirleme işlemlerinden sadece birisini kullanırken, her ikisinin de kullanıldığı sistemler mevcuttur. Sadece kafa pozunu kullanarak etkileşim sağlayan sistemler, genellikle kafa pozunu verisini imleç hareketlerini gerçekleştirmek için kullanmaktadır [1, 2]. Sadece göz merkezini veya tüm göz bölgesini kullanan yöntemler de benzer amaçla kullanılabilir [3–5]. Bakış noktası tahmini alanında, sadece gözden alınan verilerle etkileşim sağlayan

sistemler, kafa hareketlerini ihmal etmekte veya kafanın sabit tutulmasını sađlayan bir çene destek aparatı gerektirmektedir. Sadece kafa hareketlerini kullanan sistemler de göz hareketlerini ihmal etmektedir. Ancak sadece gözlerden veya sadece baş hareketlerinden alınan verilerin kullanılması, sistemlerin doğruluđunu azaltmaktadır. Bunun sebebi, ekrana bakılan noktanın bulunması veya ekranda imlecin hareket ettirilmesi uygulamalarında, göz hareketleri ve kafa pozunun birbirlerini etkilemesinin kaçınılmaz oluşudur. Bu sebeple bu iki ayrı problemin, insan bilgisayar etkileşimi uygulamalarında bir bütün olarak ele alınması gerekmektedir. Bu nedenle literatürde gerçekleştirilen bakış noktası tahmini gibi uygulamalarda, genellikle her iki problem de çözüme dahil edilmektedir [6–9].

1.1 Literatür Araştırması

Göz merkezinin belirlenmesi ve kafa pozunun tahmini, insan bilgisayar etkileşimi alanında çözülmesi gereken en önemli problemlerdendir. Bu nedenle literatürde her iki probleme de yoğun ilgi gösterilmiştir. Bu iki problemin çözümüne yönelik literatürde gerçekleştirilen çalışmalar ayrı başlıklar altında sunulmuştur.

1.1.1 Göz merkezinin belirlenmesi

Göz merkezinin belirlenmesi/takibi için yapılan çalışmalar, temel anlamda iki ana başlığa ayrılabilir. Göz kenarlarına yerleştirilen problar vasıtası ile bu bölgelerde gerçekleşen potansiyel farkı ölçen ve ölçüm verisini kullanarak göz hareketlerini takip eden Electrooculography (EOG) tabanlı yöntemler ilk grubu oluşturmaktadır [10–14]. Göz yuvası içerisinde gerçekleşen göz hareketleri, ölçüm probundan farklı ölçümler alınmasına sebep olmakta ve elde edilen ölçüm farklılıkları göz takibi için kullanılmaktadır. EOG tabanlı yöntemler, kullanıcının bir ölçüm probu takmasını gerektirdiđi için diđer yöntemlere göre daha az ilgi görmektedir. Göz merkezinin belirlenmesi konusunda daha yaygın olarak kullanılan ve ikinci grubu oluşturan yöntemler ise görüntü işleme algoritmaları kullanılarak video veya kameradan alınan görüntünün işlenmesi ile göz merkezinin belirlendiđi yöntemlerdir [15–19]. Göz merkezi belirlemek için önerilen çalışmaları, kullanıcının elektrot veya başa takılan cihazlar gibi ekstra bir aparat takmasını gerektiren (intrusive) yöntemler ve herhangi bir harici donanım gerektirmeyen (non-intrusive) yöntemler olmak üzere iki alt gruba ayırmak da mümkündür. Göz kenarlarına prob

takılmasını gerektiren yöntemler, kullanıcıya rahatsızlık verebileceği için genellikle tercih edilmemektedirler. Bu yöntemlerin yanında görüntü işleme tabanlı yöntemlerden olan ve görüntünün kafaya monte edilen bir kamera ile alındığı çalışmalar da kullanıcının rahatını olumsuz etkileyeceği için genellikle tercih edilmemektedir. Bu sebeple kullanıcıya rahatsızlık verebilecek bir aparat takılmadan, göz merkezinin algılandığı görüntü tabanlı yöntemlere olan eğilim çok daha fazladır.

Göz merkezinin doğru bir şekilde belirlenmesi, bakış noktası tahmini ve iris tanıma gibi alanlarda gerçekleştirilmesi gereken en önemli aşamalardan birisidir. İris tanıma için önerilen yöntemlerde yüksek çözünürlüklü resimler kullanılırken, bakış noktası tahmini gerçekleştirmek için önerilen çalışmaların çoğunlukla düşük çözünürlüklü resimler ile çalışması beklenmektedir. Yüksek çözünürlüklü resimlerde göz bebeği ve iris diğer göz bölgelerinden kolaylıkla ayırt edilebilirken, düşük çözünürlüklü resimlerde göz bebeğinin diğer göz bölgelerinde ayırt edilmesi genellikle mümkün olmamaktadır. Göz bebeği ve iris merkezleri eş olduğu için düşük çözünürlüklü resimlerde göz merkezi belirlenmesinde irisin kullanılması daha uygundur. Görüntü işleme algoritmaları kullanan yöntemleri görüntünün alınma şekline göre ikiye ayırmak mümkündür. Bunlardan ilki, görünür ışık altında elde edilen görüntüyü işleyerek göz merkezini belirleyen yöntemler [15, 16], diğeri ise Kızılötesi-IR (Infrared) aydınlatma altında alınan resimlerde göz merkezini belirleyen yöntemlerdir [20–22]. IR aydınlatma altında çalışan yöntemler, görünür ışık altında çalışan sistemlere göre daha başarılı sonuçlar vermektedir. Ancak bu sistemler ektsra bir IR aydınlatmaya ihtiyaç duymakta ve gün ışığı altında doğrulukları azalmaktadır. Bu nedenle görünür ışık altında çalışan yöntemler daha fazla dikkat çekmektedir. Görüntü işleme algoritmaları kullanılarak göz merkezinin belirlendiği yöntemler, genellikle göz resminden çıkarılan kenar ve gradyan verilerini kullanmaktadır. Iris (siyah bölge) ile sclera (göz akı) arasındaki gri seviye farkı ve irisin dairesel şekli, bu amaçla önerilen yöntemlerde başvurulan özelliklerdir [6, 7].

İris ile göz akı arasındaki yüksek gri seviye farkından faydalanılan oylama tabanlı yöntemlere sıklıkla başvurulmaktadır. Oylama tabanlı yöntemlerden birisi Dairesel Hough Dönüşümü (Circular Hough Transform-CHT) yöntemidir [23, 24]. Oylama işlemini uygulamak için resme ilk olarak kenar bulma işlemi uygulanmakta, belirli bir sınır değeri üzerinde kalan pikseller, oylamaya dahil edilecek pikseller olarak belirlenmektedir. Sabit yarıçaplı farklı merkezli dairelere ait denklemler

kullanılmakta ve bulunan noktaların en çok uyduğu denkleme sahip daire iris olarak seçilmektedir. Oylama tabanlı bir diğer yöntem Timm ve Barth tarafından önerilen gradyan tabanlı yöntemdir [15]. Bu yöntemde, iris dairesel olarak kabul edilmiş ve dairesel şekiller için daire üzerinde bir noktadaki normalize edilmiş gradyan vektörü ile daire merkezinden bu noktaya çizilen birim mesafe vektörünün aynı doğrultuda olması gerekliliği ortaya konulmuştur. İki vektör arasındaki nokta çarpım, aynı doğrultudaki vektörler için maksimum olacağından, nokta çarpımların karesinin toplamını maksimum yapacak bir amaç fonksiyonu oluşturulmuştur. Bu amaç fonksiyonunu kullanarak göz bölgesindeki tüm piksellerin merkez olma ihtimali değerlendirilmiştir. Özellikle iris tanıma alanında önemli bir yere sahip olan bir başka göz merkezi belirleme yöntemi de J. Daugman tarafından önerilmiştir [25]. J. Daugman'ın yaklaşımı, göz bölgesindeki noktaların gri seviye değerlerinin dairesel olarak toplanmasına ve farklı daireler için elde edilen toplam değerlerinin farklarının alınmasına dayanmaktadır. J. Daugman tarafından önerilen yöntem, düşük çözünürlüklü resimlerde iyi sonuç vermemekle birlikte, IR aydınlatma altında alınan yüksek çözünürlüklü resimlerde oldukça başarılı sonuçlar vermektedir.

Bazı oylama tabanlı yaklaşımlarda ise iris daire yerine elips olarak düşünülmekte ve bulunan noktalara en uygun elips uydurulmaya çalışılmaktadır. İris, farklı açılardan elips şekline daha çok benzediği için irisi elips olarak düşünmek, daha gerçekçi bir model oluşturmayı sağlamaktadır. Elips uydurma yaklaşımı, farklı boyutlardaki elipsleri de dikkate almaktadır. Bulunan noktalar elips denkleminde yerine yazılarak ve minimum kareler yöntemi kullanılarak elips denkleminin katsayıları belirlenmektedir [26, 27]. Elips denkleminde ait katsayıların hesaplanmasında kullanılan noktaların bazıları, iris ile göz akı arasındaki sınırın dışında bulunabilmektedir. Bu durumda, bu noktaların elips denklemi katsayılarının belirlenmesinde kullanılması, katsayıların yanlış bulunmasına sebep olmaktadır. Bu probleme çözüm üretmek için tüm noktaları kullanmak yerine Ransac (Random Sample Consensus) algoritması kullanılarak rastgele noktalar seçilmekte ve iteratif bir yaklaşımla en uygun elips katsayıları bulunmaktadır [5, 28, 29].

1.1.2 Kafa pozunun belirlenmesi

Kafa pozunu tahmini problemi, insan bilgisayar etkileşimi ve bilgisayarla görüş alanlarında yoğun olarak ilgilenilen konulardandır. Bu uygulama da diğer görüntü

işleme uygulamaları gibi kamera bozuklukları, ışık şartlarındaki değişim gibi faktörlerden en az seviyede etkilenecek şekilde kararlı bir şekilde çalışmalıdır. Bu amaçla ortaya konulan çalışmalar, detaylı olarak sekiz farklı başlık altında incelenebilir [30]. Ancak en genel anlamda görünüm tabanlı yöntemler, esnek model kullanılan yöntemler ve sabit modele dayalı yöntemler olarak üçe ayırmak mümkündür.

Görünüm tabanlı yöntemlerde, farklı pozlarda çekilen görüntülerden oluşan veri tabanları kullanılarak poz tahmini gerçekleştirilmektedir. Bu amaçla kullanılan en basit yöntem, işlenen resmin veri tabanında bulunan resimler ile karşılaştırılması ve en benzer resmin belirlenmesi yöntemidir. Resimlerin benzerliklerinin belirlenmesi için piksel değerleri farkı, belirli öznitelikler ve çapraz korelasyon gibi yaklaşımlar kullanılmaktadır [31–33]. Bahsedilen görüntü tabanlı yöntemlerin doğruluğu, kullanılan veritabanıyla sınırlıdır ve eğitim işlemi zordur. Bu nedenle daha gelişmiş görünüm tabanlı yöntemlerde, pozları sınıflandırmak yerine regresyon yaklaşımı kullanılarak resim uzayından kafa pozuna geçiş yapılmaktadır. Bu yöntemlerde poz verisi sürekli bir şekilde elde edilmektedir [34, 35].

Model tabanlı yöntemlerde, esnek veya sabit modeller kullanılarak insan kafası modellenmektedir. Esnek kafa modeli kullanılan yöntemler, kafayı bir bütün olarak değerlendirmenin yanında kaş, göz kenarı, ağız kenarı gibi yüzdeki yerel noktaların hareketini de temsil edebilmektedirler. Sabit modeller ise kafayı sadece silindir, elips gibi rijit nesnelere olarak modellemektedirler. Sıklıkla kullanılan esnek model tabanlı yöntemlerden birisi, Aktif Şekil Modeli (AŞM) yaklaşımıdır [36]. AŞM yaklaşımında, farklı kişilerden toplanmış yüzlerdeki belirli noktaların koordinatlarını gösteren bir şekil veri tabanı oluşturulmaktadır. Bu şekil veri tabanına PCA (Principal Component Analysis) algoritması [37] uygulanarak ortalama şekil ve şekil değişiminin en yüksek olduğu özvektörler hesaplanmaktadır. Yeni gelen yüze en iyi uyan şeklin hesaplanması için veritabanından elde edilen özvektörlerin katsayıları hesaplanmaktadır. Katsayıları hesaplamak için gelen resim ile hesaplanacak şekil noktaları etrafındaki gradyan değerlerinin uyumu gerekliliğinden faydalanılmaktadır. AŞM yaklaşımını temel alan bir diğer yöntem ise Aktif Görünüm Modeli (AGM) yaklaşımıdır [38]. AGM yaklaşımında, veri tabanındaki yüzlerin şekil verisinin yanında gri veya rgb değerleride dikkate alınmaktadır. Şekil veritabanına ait özvektörler ile birlikte gri veya rgb değerlerden oluşan veritabanına ait özvektörler de hesaplanmaktadır. Yeni gelen yüze en iyi uyan şeklin

hesaplanması için her iki veri tabanından elde edilen özvektörlerin katsayıları hesaplanmaktadır.

Bazı model tabanlı yöntemler, silindir, elips gibi kafa şekline benzer sabit modeller kullanarak kafanın pozunu elde etmektedir. Bu yöntemler, yüz bölgesinde belirli özniteliklerin veya tüm yüz bölgesinin takip edilmesine dayanmaktadır. Öznitelik noktalarının takip edildiği yöntemlerde, kameradan alınan başlangıç resmi üzerinde öznitelik noktalarının algılanması ve algılanan bu noktaların ardışık resimler arasında takip edilmesi veya eşleştirilmesi prensibi kullanılmaktadır [39, 40]. Takip edilen öznitelikler, sabit modelin hareketlerini temsil etmektedir. Öznitelik noktaları, resimler arasındaki poz değişimi, çevre şartlarının değişimi gibi bozucu etkilere rağmen takip edilebilecek özellikte olan köşe ve kenar noktaları olarak seçilmektedir. Başlangıçta belirlenen 2B resim noktalarının, 3B model koordinatlarına dönüştürülmesi için kamera dahili parametreleri ve doğru-obje kesişiminden yararlanılmaktadır. Başlangıçta elde edilen 3B öznitelik noktaları ile takip edilerek veya eşleştirme ile belirlenen, bir sonraki resimdeki 2B resim koordinatları arasındaki dönüşüm matrisi belirlenerek modelin ilk durumuna göre bağıl dönme ve öteleme miktarı hesaplanmaktadır. Sabit model kullanılan bazı çalışmalarda ise tüm yüz bölgesinin takibi gerçekleştirilerek poz verisi elde edilmektedir [41, 42]. Sabit modeller, esnek modellerin aksine kafayı bir bütün olarak ele almakta ve yüz bölgesinin karmaşıklığını modellememektedir. Bu nedenle sabit modeller esnek modellere göre çok daha basit yapıya sahiptir ve modelin bulunması gereken parametre sayısı çok daha azdır. Bu avantajları nedeniyle gerçek zamanlı uygulamalarda sabit model kullanımı daha çok tercih edilmektedir.

Kafa pozunu tahmini yaklaşımları, farklı senaryolarda birbirlerine karşı üstünlük göstermektedir. Ancak iki yaklaşımın birleştirildiği hibrit yöntemler daha yüksek doğruluk seviyelerine çıkabilmektedir [43, 44].

1.2 Tezin Amacı ve Organizasyonu

Bu tez çalışmasında, insan bilgisayar etkileşimi için büyük öneme sahip olan göz merkezi belirleme ve kafa pozunu tahmini probleminin iyileştirilmesine yönelik yaklaşımlar ortaya konmuş ve gerçek zamanlı olarak uygulanmıştır. Göz merkezinin belirlenmesi için şekil tabanlı bir yöntem kullanılarak, göz bölgesinin gradyan verisi ve iris dairesel şeklinden faydalanılmıştır. Bu amaçla daha önce önerilen iki

yöntemden [15, 25] faydalanılarak yeni bir amaç fonksiyonu geliştirilmiş ve iteratif bir yaklaşımla göz merkezi ve iris yarıçapı belirlenmiştir. Bu yöntemlerden ilki Timm ve Barth tarafından önerilmiştir [15]. Bu yöntemde, irisin dairesel şekline ve göz bölgesinden elde edilen gradyan verisine dayalı yeni bir amaç fonksiyonu geliştirilmiştir. Bu yöntem, düşük çözünürlüklü resimlerde tatmin edici sonuçlar vermektedir. Yöntemin dezavantajları ise amaç fonksiyonu hesaplanırken dahil edilen noktaların dairesel olarak seçilmemesi ve iris yarıçapının belirlenememesidir. Görüntü işleme tabanlı bir diğer önemli yöntem ise John Daugman tarafından ortaya konulmuştur [25]. Bu yöntemde, iris merkezi olma adayı noktaların etrafına farklı yarıçaplı daireler çizilmiş ve bu daireler üzerinde bulunan noktaların gri seviye değerleri toplanmıştır. Bu yöntem yüksek çözünürlüklü ve IR aydınlatma altında alınan resimlerin kullanıldığı, iris tanıma gibi çalışmalar için ideal bir çözüm olsa da düşük çözünürlüklü resimlerde aynı sonuçlar elde edilememektedir. Tez çalışmasında, Timm ve Barth tarafından önerilen nokta çarpım yaklaşımı ile J. Daugman tarafından önerilen dairesel tarama yaklaşımı birleştirilerek yeni bir amaç fonksiyonu elde edilmiştir. Böylece literatürde önemli yere sahip bu iki yöntem birleştirilerek, sundukları avantajlardan aynı anda faydalanacak yeni bir amaç fonksiyonu geliştirilmiştir.

Kafa pozunu algılamak için model tabanlı ve takibe dayalı bir yaklaşım kullanılmıştır. Kafa, sabit silindir olarak modellenmiş ve silindirin kamera koordinatlarına göre pozunu belirlenmiştir. Silindir kafa modelinin kameraya göre pozunun belirlenmesi için yüz bölgesinde belirlenen öznitelikler kullanılmıştır. Başlangıçta belirlenen öznitelik noktalarının silindir model üzerindeki 3B koordinatları bulunmuş ve kafa pozunu modelin başlangıç durumuna göre belirlenmiştir.

Tezde sunulan göz merkezi belirleme ve kafa pozunu tahmini yaklaşımlarının doğruluk değerleri, literatürde sıklıkla başvuru Boston Üniversitesi tarafından oluşturulmuş olan kafa pozunu veri seti [42] ve göz merkezi belirleme yöntemlerinde başvuru BioID [47] veri seti kullanılarak değerlendirilmiştir. Veri setleri kullanılarak elde edilen sonuçların yanında sistemin gerçek zamanlı performansı, işlem süreleri temel alınarak sunulmuştur.

Bu tez beş ana bölümden oluşmaktadır. İlk bölüm, tez çalışmasında gerçekleştirilen sistemin uygulama alanını, amacını ve bu alanda gerçekleştirilmiş diğer önemli çalışmaların özetlendiği, literatür taraması konularını kapsamaktadır. İkinci

bölümde, göz merkezinin kabaca belirlenmesi amacıyla kullanılan metot detaylı biçimde sunulmuştur. Üçüncü bölümde, tez çalışmasında geliştirilen göz merkezi ve iris yarıçapı belirleme yaklaşımı açıklanmış, elde edilen sonuçlar ve gerçek zamanlı uygulama sonuçları sunulmuştur. Dördüncü bölümde, kafa pozu tahmininin gerçekleştirilmesi anlatılarak elde edilen sonuçlar ve gerçek zamanlı uygulama sonuçları sunulmuştur. Beşinci bölümde, elde edilen sonuçlar ve öneriler sunulmuştur.



2. GÖZ MERKEZİNİN KABACA BELİRLENMESİ

Göz merkezinin belirlenmesine yönelik çalışmalar, insan bilgisayar etkileşimi uygulamalarında önemli bir yere sahiptir. Bunun en iyi örneklerinden biri, kolu olmayan kişilerin bilgisayar kullanmalarına olanak veren, bakış noktası tahmini uygulamalarıdır. Bu tür uygulamalarda, göz merkezi yüksek doğrulukta ve hızlı bir şekilde belirlenmelidir. Tez çalışmasında, doğruluk, hız ve kararlılık yönünden gereksinimleri karşılayacak nitelikte olan, göz merkezi ve iris yarıçapının belirlendiği bir yöntem önerilmiştir. Sunulan yöntem, sadece bir kameraya ihtiyaç duymaktadır ve kullanıcıyı rahatsız edebilecek harici herhangi bir donanım gerektirmemektedir. Tezde sunulan göz merkezi belirleme algoritması, göz merkezini doğru ve kararlı bir şekilde belirleyebilmek için iki aşamada gerçekleştirilmektedir. Bu bölümde detaylıca anlatılan ilk aşamada, göz merkezinin konumu kabaca belirlenmekte, ikinci aşamada ise tez çalışmasında önerilen amaç fonksiyonu kullanılarak kabaca belirlenen göz merkezi etrafında gerçek göz merkezi belirlenmektedir. Kabaca belirleme işlemi için Timm ve Barth tarafından önerilen, resim gradyanına dayalı yöntem kullanılmıştır [15]. Timm ve Barth tarafından önerilen yöntemde, iris şeklinin dairesel olduğu düşünülmekte ve bir dairenin kenar noktasından çizilen normal vektörü ile dairenin merkezinden normalin çizildiği noktaya olan mesafe vektörünün aynı doğrultuda olması gerektiği bilgisinden yararlanılmaktadır.

Gerçek zamanlı görüntü işleme uygulamalarında, tüm resim ile işlem yapmak yerine aranan niteliğin bulunduğu yerel bölgelerde işlem yapmak çok daha verimlidir. Bu yüzden tezde önerilen göz merkezi belirleme algoritması, kameradan alınan tüm resme uygulanmak yerine belirlenen göz bölgelerine uygulanmıştır. Böylece göz merkezi belirleme işlemi, daha kısa sürede ve daha yüksek doğrulukta gerçekleştirilmiştir. Göz merkezinin kabaca belirlenmesi için üç aşamalı bir yaklaşım kullanılmıştır. İlk aşamada, Viola ve Jones [45] tarafından önerilen ve basamaklı sınıflandırıcılar kullanılan, yüz algılama algoritması ile resimde yüzün bulunduğu

bölge belirlenmiştir. İkinci aşamada, yüz ile gözler arasındaki geometrik ilişki kullanılarak sağ ve sol göz bölgeleri belirlenmiştir. Son aşamada göz merkezini kabaca belirleme algoritması, elde edilen göz bölgelerine uygulanmıştır.

2.1 Yüzün Algılanması ve Göz Bölgelerinin Belirlenmesi

Viola ve Jones [45], resim içerisinde bulunan herhangi bir nesnenin algılanması amacıyla, eğitime dayalı ve ardışık sınıflandırıcıların kullanıldığı bir yöntem önermişlerdir. Önerdikleri algoritma, genel bir nesne algılama algoritması olmasına karşın algoritmanın birincil amacı yüz algılamaktır. Algoritma, çok fazla sayıda yüz resmi ile eğitilmiştir ve eğitim süresi uzundur. Ancak algılama işlemi çok hızlı bir şekilde gerçekleştirilmektedir. Algılama hızının yüksek olması nedeniyle, gerçek zamanlı uygulamalarda sıklıkla tercih edilmektedir. Algoritma, güçsüz sınıflandırıcıların birleştirilmesiyle oluşturulan, güçlü bir sınıflandırıcıdan elde edilen sonuçlara göre işlenen resim içerisinde yüz olup olmadığını ve olan yüzlerin konumunu belirlemektedir. Yüz algılama işlemi dört aşamada gerçekleştirilmektedir.

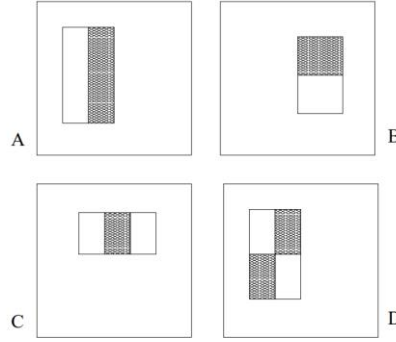
- i. Haar-Benzeri Özniteliklerin Seçimi
- ii. İntegral Resmin Oluşturulması
- iii. Adaboost Eğitimi
- iv. Basamaklı Sınıflandırıcı

2.1.1 Haar-Benzeri özniteliklerin seçimi

Nesne algılama çalışmalarında, resimdeki piksel ilişkilerini iyi ifade eden ve daha hızlı işlem yapılmasını sağlayan öznitelikler sıklıkla kullanılmaktadır. Bu özniteliklerden yaygın kullanılanlardan biri, Papageorgiou tarafından önerilen Haar-Benzeri özniteliklerdir [46]. Haar-Benzeri öznitelikler, dikdörtgensel bir bölge içerisinde kalan bir grup pikselin toplamının, aynı dikdörtgensel bölge içerisindeki başka bir grup pikselin toplamından çıkarılması ile elde edilmektedir. Dikdörtgensel bölge içerisindeki grup sayıları değiştirilerek farklı öznitelikler elde edilmektedir. Haar-Benzeri öznitelikler basit bir toplama ve çıkarma işlemine dayandığı, resimdeki piksel ilişkilerini iyi ifade ettiği için sıklıkla kullanılmaktadır.

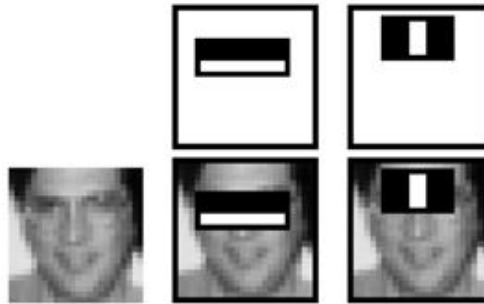
Viola ve Jones [45], yüz algılama işlemi için Haar-Benzeri özniteliklerden faydalanmış ve tüm insan yüzleri için benzer olan bölgeleri ifade edecek öznitelikler

belirlemişlerdir. Bu benzerliklere örnek olarak, insanların burnunun göz bölgelerine göre daha parlak bir yapıda olması gösterilebilir. Şekil 2.1’de Viola ve Jones tarafından önerilen nesne algılama algoritmasında kullanılan öznelik dikdörtgenleri görülmektedir.



Şekil 2.1 : Yüz algılama için belirlenen Haar-Benzeri öznelik dikdörtgenleri [45].

Her bir öznelik dikdörtgeni için beyaz bölgeler içerisinde kalan piksellerin toplamı, siyah bölgeler içerisinde kalan piksellerin toplamından çıkarılmaktadır. Böylece her bir öznelik dikdörtgeni için bir fark değeri hesaplanmaktadır. Şekil 2.2’de öznelik dikdörtgenlerinin insan yüzü ile uyuşma durumu gösterilmiştir.



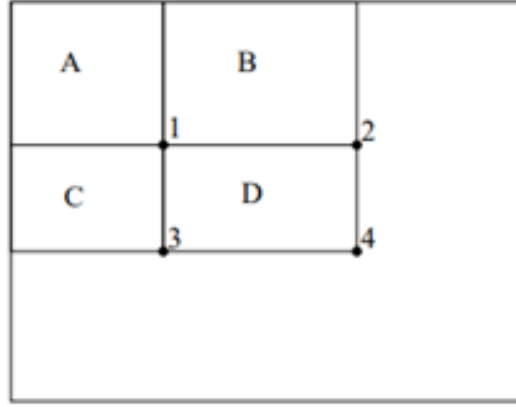
Şekil 2.2 : İnsan yüzüne uygun seçilmiş öznelik dikdörtgenleri [45].

2.1.2 İntegral resmin oluşturulması

Nesne algılama uygulamalarında, nesne boyutları değişiklik gösterebildiği için geliştirilen algoritmalar, nesne boyutlarından bağımsız olarak algılama işlemini gerçekleştirmelidir. Viola ve Jones [45] tarafından önerilen yüz algılama yöntemi, oluşturulan öznelik dikdörtgenlerinin boyutlarını değiştirerek nesne boyutları farklı olsa da algılama işleminde başarılı olmaktadır. Aynı öznelik dikdörtgenlerinin, farklı boyutlarda kullanılması ile 24x24 piksellik çözünürlüğe sahip bir resim için 180.000’den fazla öznelik dikdörtgeni elde edilmektedir. Bu kadar fazla sayıdaki öznelik dikdörtgeni için beyaz ve siyah bölgelerinin içinde kalan pikselleri

toplamak, işlem yükünü ciddi derecede arttırmaktadır. Bu nedenle Viola ve Jones tarafından, öznitelik dikdörtgenleri içerisinde kalan pikselleri toplamak için işlemsel olarak verimli bir yöntem olan ve integral resim olarak adlandırılan bir yöntem kullanılmıştır.

İntegral resim, orijinal resimle aynı boyutlara sahiptir. İntegral resmin herhangi bir pozisyonundaki değeri, üzerindeki ve solundaki tüm piksel değerlerinin toplamlarına eşittir. Şekil 2.3'te dört bölmeli bir öznitelik dikdörtgeni görülmektedir.



Şekil 2.3 : İntegral resim üzerindeki bir öznitelik dikdörtgeni [45].

Bu şekle göre 1 numaralı pozisyondaki toplam değeri, A dikdörtgeninin içerisindeki piksellerin toplamına eşittir. 2 numaralı pozisyondaki toplam değeri, A ve B dikdörtgenleri içerisinde kalan piksel değerlerinin toplamına eşittir. Bu hesaplama işlemi, diğer tüm pozisyonlar için benzer şekilde uygulanabilmektedir.

2.1.3 Adaboost eğitimi

Yüz algılama algoritmasının eğitilmesi ve işe yarar özniteliklerin seçimi için Adaboost algoritması kullanılmaktadır [45]. Boyutları 24x24 piksel olan bir resim için 180.000'den fazla öznitelik dikdörtgeni elde edilebilmektedir. Özniteliklerin hesaplanması sadece toplama gibi basit bir işleme dayansa bile bu kadar fazla sayıda öznitelik için işlem yükü çok fazla olacaktır. Adaboost algoritmasında, elde edilen her bir öznitelik dikdörtgeni bir sınıflandırıcı olarak ele alınmaktadır. Doğruluğu yüksek sınıflandırıcılar (öznitelik dikdörtgenleri) belirlenmekte ve güçsüz sınıflandırıcı olarak atanmaktadır. Seçilen güçsüz sınıflandırıcılar birleştirilerek sınıflandırma doğruluğu daha yüksek olan, güçlü bir sınıflandırıcı oluşturulmaktadır. Sınıflandırıcının eğitimi, pozitif ve negatif resimlerin sınıf etiketleriyle birlikte bulunduğu bir veri tabanı kullanılarak gerçekleştirilmektedir. Adaboost algoritması,

bir önceki güçsüz sınıflandırıcı tarafından yanlış sınıflandırılan örneklere daha büyük ağırlık vererek bir sonraki seçilecek olan güçsüz sınıflandırıcının, önceki aşamada yanlış sınıflandırılan örnekleri sınıflandırabilecek şekilde seçilmesini sağlamaktadır. Bu şekilde seçilen belirli sayıdaki güçsüz sınıflandırıcı birleştirilerek güçlü bir sınıflandırıcı elde edilmektedir. Adaboost algoritması aşağıda verilmiştir.

Çizelge 2.1 : Yüz algılama aşamasında kullanılan Adaboost algoritması [45].

Adaboost Algoritması

1. $(x_1, y_1), \dots, (x_n, y_n)$ verilen veri seti olmak üzere y_i pozitif örnekler için 1, negatif örnekler için sıfırdır.
2. Tüm örneklerin ağırlıklarını başlangıçta eşit olarak belirle $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$, burada m negatif örneklerin, l ise pozitif örneklerin sayısıdır.

3. $t = 1, \dots, T$

- Ağırlıkları normalize et

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$$
- Herbir öznitelik, j için bir sınıflandırıcı, h_j eğit ve sınıflandırıcı hatasını belirle.

$$\epsilon_j = \sum_i w_i (h_j(x_i) - y_i)$$

- En düşük hataya sahip sınıflandırıcıyı belirle.
- Ağırlıkları güncelle

$$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$

İlgili sınıflandırıcı doğru sınıflandırma yaptı ise $e_i = 0$, yanlış ise $e_i = 1$ ve $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$

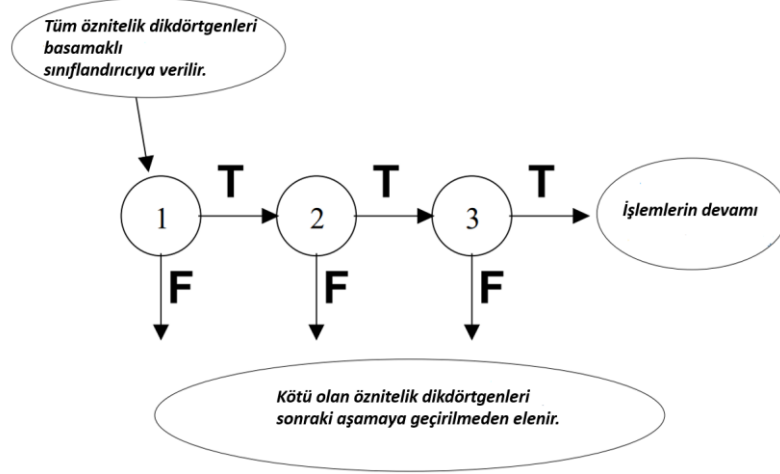
4. Sonuçta elde edilen güçlü sınıflandırıcı:

$$|x| = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{diğer} \end{cases}$$

2.1.4 Basamaklı sınıflandırıcı

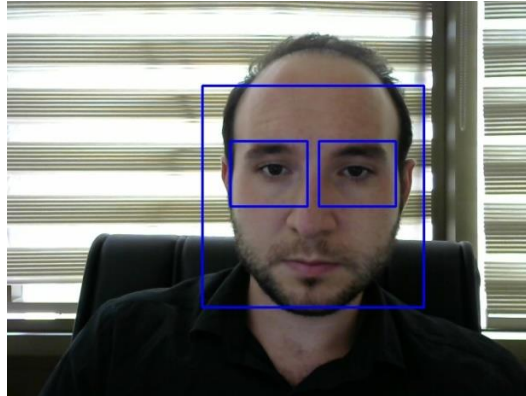
Yüzün algılanacağı resimden elde edilen tüm öznitelik dikdörtgenlerinin işleme alınması algoritmanın yavaş çalışmasına sebep olmaktadır. Bu yüzden tüm öznitelik dikdörtgenlerini aynı anda verip, sınıflandırma sonucuna bakmak yerine öznitelikler basamaklı bir sınıflandırıcıdan geçirilmiştir [45]. Bu yaklaşıma göre ilk gruptaki sınıflandırıcıları geçemeyen öznitelik dikdörtgeni, bir sonraki aşamaya geçirilmeden elenmektedir. Basamaklı sınıflandırıcı, ilk aşamalara yerleştirilen basit sınıflandırıcılardan ve sonraki aşamalara yerleştirilen daha karmaşık

sınıflandırıcılardan oluşmaktadır. Böylece çoğu resim, basit sınıflandırıcı çıkışına göre elenmekte ve karmaşık sınıflandırıcıların uygulanmasına gerek kalmamaktadır. Şekil 2.4'te basamaklı sınıflandırıcı yapısı görülmektedir.



Şekil 2.4 : Basamaklı sınıflandırıcı [45].

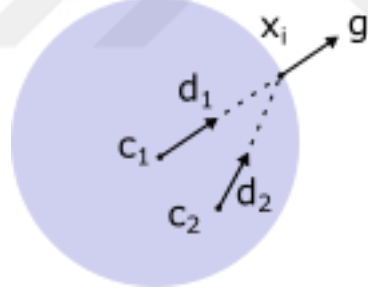
Tez çalışmasında, kameradan alınan resme, aşamaları verilen yüz belirleme algoritması uygulanmıştır. Yüz belirleme işleminde, açık kaynaklı olarak görüntü işleme algoritmalarının sunulduğu OpenCv kütüphanesi kullanılmıştır. OpenCv kütüphanesi, Viola ve Jones [45] tarafından önerilen metot için eğitilmiş olan bir model içermektedir. Bu model kullanılarak kameradan alınan resim içerisindeki yüz bölgesi belirlenmiştir. Göz merkezi belirleme algoritması, sağ ve sol göz bölgelerine ayrı ayrı uygulanacağı için belirlenen yüz bölgesinden göz bölgeleri elde edilmiştir. Göz bölgelerini elde etmek için yüz bölgesi ile göz bölgeleri arasındaki geometrik ilişki kullanılmıştır. Şekil 2.5'te yüz ve göz bölgelerinin algılanması işlemlerinden sonra elde edilen sonuç resmi görülmektedir.



Şekil 2.5 : Yüz ve göz bölgelerinin algılanmasıyla elde edilen sonuç resmi.

2.2 Göz Merkezini Kabaca Belirleme Algoritması

Göz merkezinin kabaca belirlenmesi için literatürde yaygın olarak kullanılan, gradyan tabanlı bir yaklaşımdan faydalanılmıştır [15]. Daha önceden belirlenmiş olan göz bölgelerine, bu yöntem uygulanarak göz merkezi kabaca belirlenmiştir. Yararlanılan yöntem, resmin gradyan vektör alanından ve irisin dairesel şeklinden faydalanmaktadır. İris, insan gözü içerisindeki en koyu bölümdür ve iris ile göz akı arasında güçlü bir gri seviye farkı bulunmaktadır. Bu nedenle iris ile göz akı arasındaki sınır bölgesinde (limbus) gradyan vektörlerinin büyüklükleri fazla olmaktadır. Herhangi bir resimdeki dairesel bir şeklin sınırlarından çizilen normal vektörleri, normalize edilmiş gradyan vektörleri olarak adlandırılmaktadır. Dairesel bir şeklin merkezinden, kenardaki bir noktaya çizilen mesafe vektörü ile bu kenar noktasındaki normal vektörü aynı doğrultudadır. Normal vektörü, gradyan vektörüne eşit olduğundan, merkezden herhangi bir kenar noktasına çizilen mesafe vektörü ile o noktadaki gradyan vektörü aynı doğrultuda olmaktadır [15]. Şekil 2.6'da görüldüğü gibi gerçek daire merkezinden çizilen mesafe vektörü ile gradyan vektörünün doğrultuları aynıdır.



Şekil 2.6 : İris ve göz akını temsil eden yapay resim üzerinde gradyan ve mesafe vektörlerinin doğrultuları.

Şekilde c iris merkezini, g_i resmin x_i noktasındaki normalize edilmiş gradyan vektörünü, d_i ise mesafe vektörünü ifade etmektedir.

Bu iki vektör aynı doğrultuda olduğu için nokta çarpımlarının değeri de yüksek olmaktadır. Kullanılan metot, bu nokta çarpımların toplamının yüksek olması gerekliliğine dayanmaktadır ve göz bölgesi içerisindeki her bir pikselin merkez olma ihtimalini, nokta çarpımların toplamının değeriyle hesaplamaktadır. Yöntemde, merkez arama işlemi, nokta çarpımların karelerinin toplamının maksimum yapılmaya çalışıldığı bir amaç fonksiyonu ile gerçekleştirilmektedir.

En uygun c_{opt} merkezi aşağıdaki amaç fonksiyonu ile bulunmaktadır.

$$c_{opt} = \underset{c}{argmax} \frac{1}{N} \sum_{i=1}^N (d_i^T g_i)^2 \quad (2.1)$$

Burada;

$$d_i = \frac{x_i - c}{\|x_i - c\|_2} \quad (2.2)$$

$$\|g_i\|_2 = 1 \quad (2.3)$$

dir.

Amaç fonksiyonunda verilen *argmax* ifadesi, kendisinden sonra kullanılan matematiksel ifadenin maksimum değerini almasını sağlayan c parametresinin arandığını ifade etmektedir. Burada c parametresi muhtemel merkez noktaları anlamına gelmektedir. Mesafe vektörü olan d_i birim vektöre dönüştürülerek mesafe vektörünün büyüklüğü ihmal edilmiştir. Aydınlatma şartlarının değişimine karşı sistemi daha kararlı hale getirmek için g_i gradyan vektörü normalize edilmiştir. Gradyan değeri, hesaplanan bir sınır değerden düşük olan noktalar, amaç fonksiyonundaki nokta çarpıma dahil edilmemiştir. Böylece gereksiz hesaplamalar önlenerek sistemin çalışma hızı artırılmıştır. Ayrıca göz bölgesini belirleyen sınırlar üzerinde olan pikseller de hesaplama dahil edilmemiştir [15]. Timm ve Barth tarafından önerilen, algoritmanın hız ve doğruluğunu arttırmak için yapılan bu eklemelere rağmen göz bölgesi içerisinde bulunan, amaç fonksiyonuna dahil edilecek fazla sayıda nokta bulunmaktadır. Bu nedenle bu işlemi göz bölgesinin orijinal boyutlarına uygulamak sistemin işlemsel yükünü çok arttırmakta ve gerçek zamanlı bir çalışma imkânı sunmamaktadır. Bu nedenle sistemin doğruluğu ile hızı arasındaki dengeyi sağlayacak bir göz bölgesi küçültme oranı belirlenmeli ve göz bölgesi bu oranda küçültülmelidir. Algoritmanın tez çalışmasındaki kullanılma amacı göz merkezini kabaca belirlemek olduğu için bulunan merkez noktasının iris sınırları içerisinde kalması yeterli olmaktadır. Bu durum kullanılan göz bölgesinin küçültülmesine olanak sağlamaktadır. Yapılan denemelerde göz bölgesi genişliğinin ve yüksekliğinin orijinal boyutlarının %60'ına düşürülmesinin en uygun doğruluk-hız dengesini sağladığı görülmüştür.

İris diğer göz bölümlerine göre daha koyudur. Bu nedenle daha siyah olan piksellerin merkez olma ihtimalini arttırmak için Timm ve Barth tarafından, amaç fonksiyonuna bir w_c ağırlık değeri eklenmiştir. Burada w_c orijinal resmin terslenmiş ve gauss filtresi uygulanmış halini gösteren matristeki ilgili merkez noktasına karşılık gelen değerdir. Ağırlığın denkleme eklenmesiyle yeni amaç fonksiyonu aşağıdaki gibi elde edilmektedir [15].

$$c_{opt} = \underset{c}{argmax} \frac{1}{N} \sum_{i=1}^N w_c (d_i^T g_i)^2 \quad (2.4)$$

Yapılan denemelerde, algoritmanın kaş bölgesinde bir noktayı göz merkezi olarak gösterdiği durumların fazla olduğu tespit edilmiştir. Timm ve Barth, bu durumu aşmak için göz bölgesi sınırlarında olan pikselleri ihmal etmiştir. Ancak bu yaklaşımın hatayı istenilen seviyede azaltmadığı görülmüştür. Bu nedenle tez çalışmasında, sadece kenar pikselleri değil kenara belirli mesafeden daha yakın olan piksellerin merkez olma olasılığı ihmal edilmiştir. Bu oran göz bölgesinin %20'si olarak belirlenmiştir. Eğer belirlenen nokta sağ veya sol göz kenarına göz genişliğinin %20'sinden daha yakın, alt veya üst kenara göz bölgesi yüksekliğinin %20'sinden daha yakınsa bu nokta göz merkezi olarak belirlense bile ihmal edilmiş ve bir sonraki en yüksek değere sahip nokta göz merkezi olarak atanmıştır. Bu iyileştirmenin algoritmaya eklenmesi, kaş üzerinde bir noktanın merkez olarak seçilme ihtimalini azaltmıştır.

3. İYİLEŞTİRİLMİŞ GÖZ MERKEZİ BELİRLEME YAKLAŞIMI

Göz merkezinin kabaca belirlenmesi için kullanılan algoritma [15], ışıklandırma şartlarındaki değişim ve diğer bozucu etmenlere rağmen kararlı bir şekilde çalışabilmektedir. Ancak bu algoritma kullanılarak belirlenen göz merkezi, çoğunlukla iris içerisinde olsa da her zaman tam göz merkezini ifade etmemektedir. Göz merkezi bulma işlemini hızlandırmak için göz bölgesi belirli oranda küçültüldüğünde ise sonuçlar daha da kötüleşmektedir. Küçültme işleminin ardından neredeyse aynı giriş resimleri için bile iris bölgesi içerisinde farklı noktalar, göz merkezi olarak belirlenebilmektedir. Bu sonuçlardan yola çıkarak göz merkezinin kabaca belirlenmesinin ardından tam yerinin belirlenmesi için bir iyileştirme algoritmasına ihtiyaç olduğu görülmüştür.

Tez çalışmasında, literatürde önemli yere sahip iki göz merkezi belirleme yaklaşımının [15, 25] güçlü yönleri birleştirilerek yeni bir amaç fonksiyonu geliştirilmiştir. Geliştirilecek sistemin, gerçek zamanlı olarak çalışması amaçlandığı için göz merkezinin tam konumunun belirlenmesi için uygulanacak olan iyileştirme aşamasının, göz merkezi bulma işlemini önemli derecede yavaşlatmaması gerekmektedir. Kabaca belirlenen göz merkezi, çoğunlukla iris sınırları içerisinde bulunmaktadır. Bu nedenle bulunması gereken gerçek merkez noktası, kabaca belirlenen göz merkezi noktası veya etrafında bulunan küçük bir bölgedeki noktalardan birisidir. Denemeler sonucunda kabaca belirlenen merkez noktası etrafındaki 10x10 boyutlarındaki karesel piksel bölgesinin, gerçek merkez noktasını arama işlemi için uygun olduğu görülmüştür. Sadece bu bölge içerisinde bulunan noktaların merkez olma durumu değerlendirileceği için iyileştirme aşaması çok kısa sürede tamamlanabilmektedir. Böylece işlem süresi fazla uzatılmadan göz merkezi belirleme işleminin doğruluk değeri arttırılmaktadır. Şekil 3.1'de kabaca belirlenmiş göz merkezi etrafında oluşturulan arama bölgesi, farklı merkez noktaları için gösterilmiştir.



Şekil 3.1 : Kabaca belirlenmiş göz merkezleri etrafında oluşturulan arama bölgeleri. Şekilde görüldüğü gibi göz merkezinin kabaca bulunması ile elde edilen farklı noktalar etrafında çizilen 10x10 boyutlarındaki piksel bölgesi, gerçek merkezi kapsamaktadır.

İyileştirme aşamasında, göz merkezinin kabaca belirlenmesi için kullanılan, Timm ve Barth [15] tarafından önerilen yöntem ile John Daugman [25] tarafından önerilen yöntemin güçlü yönleri birleştirilerek yeni bir amaç fonksiyonu önerilmiştir. Göz merkezinin kabaca belirlenmesi aşamasında, göz bölgesi sınırlarına yakın olmayan tüm noktaların merkez olma ihtimali değerlendirilmiştir. İyileştirme aşamasında ise sadece 10x10 boyutlarındaki piksel bölgesindeki 100 noktanın merkez olma ihtimali değerlendirilmiştir. Timm ve Barth tarafından önerilen yöntemde, gradyan vektörleri ile mesafe vektörleri arasındaki nokta çarpıma dayalı bir amaç fonksiyonu kullanılmıştır. Önerdikleri yöntemin diğer yöntemlere göre avantajı, irisin dairesel şeklinden faydalanan basit bir nokta çarpıma dayalı olmasıdır. Yöntemin dezavantajları ise nokta çarpıma dahil edilecek noktaların seçiminde irisin olabilecek maksimum ve minimum yarıçaplarının dikkate alınmaması ve seçim işleminin iris şekline uygun şekilde dairesel olarak gerçekleştirilmemesidir. Bu dezavantajların ortadan kaldırılması için J. Daugman tarafından önerilen, dairesel tarama işlemine dayalı yaklaşımdan faydalanılmıştır. Böylece bilinen iki göz merkezi belirleme yaklaşımının avantajlarından aynı anda faydalanan, daha doğru sonuçlar üreten yeni bir amaç fonksiyonu elde edilmiştir.

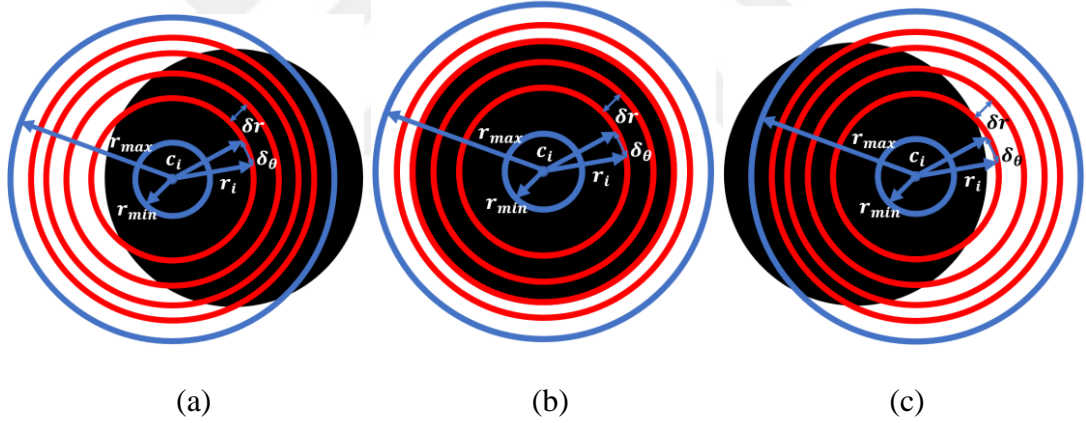
3.1 John Daugman'ın Göz Merkezi Belirleme Yaklaşımı

John Daugman, piksel değerlerinin toplamının türevine dayalı bir yaklaşım kullanarak göz merkezini bulan bir yöntem geliştirmiştir [25]. Bu yöntem iris tanıma

alanında kullanılmak üzere sunulmuştur. Bu nedenle yöntem, yüksek çözünürlükte ve IR aydınlatma altında alınan göz resimleri üzerinde yüksek doğrulukla çalışabilmektedir. Ancak düşük çözünürlükte başarı oranı ciddi oranda düşmektedir. John Daugman tarafından bulunan amaç fonksiyonu aşağıdaki gibidir.

$$\underset{(r, x_c, y_c)}{\operatorname{argmax}} \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r, x_c, y_c} \frac{I(x, y)}{2\pi r} ds \right| \quad (3.1)$$

Denklemden $I(x, y)$ resmin (x, y) noktasındaki gri seviye değerini, (x_c, y_c) muhtemel göz merkezi noktasını ve G_σ gauss filtresini ifade etmektedir. Kullanılan amaç fonksiyonu, iris ile göz akı arasındaki gri seviye farkından ve iris dairesel şeklinden faydalanmaktadır. Amaç fonksiyonu, muhtemel merkez noktaları etrafında oluşturulan, farklı yarıçaplardaki daireler üzerinde bulunan piksellerin gri seviye değerlerini toplamakta ve farklı yarıçaplı daireler için elde edilmiş olan toplam değerlerinin farkını almaktadır. Şekil 3.2’de farklı merkez noktaları için çizilmiş, farklı yarıçaplı daireler görülmektedir.



Şekil 3.2 : Farklı merkez noktaları için oluşturulan daireler: (a) ve (c) gerçek göz merkezi dışındaki noktalar için elde edilen daireler, (b) gerçek merkez noktası için elde edilen daireler.

Şekilde görüldüğü gibi çemberler üzerinde kalan piksel değerleri, yarıçap ve merkeze göre farklılık göstermektedir. Şekilde, r_{min} ve r_{max} bir merkezden çizilecek olan dairelerin olabilecek maksimum ve minimum yarıçaplarını ifade etmektedir. Her bir merkez noktası için bu aralıkta çizilecek dairelerin yarıçapları, δr yarıçap adımı ile belirlenmektedir. Çizilen daireler üzerinde bulunan noktalar, o andaki r_i yarıçapı ve δ_θ açı adımı kullanılarak belirlenmektedir. Daire üzerinde olan noktaların $I(x, y)$ gri seviye değerleri toplanmakta ve o anda ele alınan merkez noktası için kaydedilmektedir. Bir merkez noktası etrafına çizilen her bir daire için toplam değeri

hesaplanmaktadır. Son olarak ardışık dairelere ait toplam değerleri arasındaki fark hesaplanmakta ve en büyük fark değeri, ilgili merkez noktasının gerçek merkez olma skoru olarak değerlendirilmektedir. Bu işlem merkez olma ihtimali değerlendirilen tüm noktalar için uygulanmakta ve merkez olma skoru en yüksek olan nokta, gerçek merkez noktası olarak belirlenmektedir. Gerçek merkez noktası için elde edilmiş olan skoru sağlayan yarıçap ise gerçek yarıçap olmaktadır.

Şekil 3.2 (a) ve (c)'ye bakıldığında, merkez olma durumuna bakılan noktanın gerçek merkez noktasından uzakta olduğu görülmektedir. Bu noktalar için çizilen dairelerin bir kısmı koyu bölgede (iris veya göz bebeği), diğer kısmı ise beyaz bölgede (göz akı) bulunmaktadır. Bu durumda farklı daireler için elde edilen toplam değerleri arasındaki farkın düşük olacağı görülmektedir. Şekil 3.2 (b)'ye bakıldığında ise ele alınan noktanın merkez noktası olduğu görülmektedir. Beyaz bölgeye çizilen bir daireye ait toplam değeri ile siyah bölgeye çizilen bir daireye ait toplam değeri arasındaki farkın yüksek olacağı görülmektedir. Fark değeri ardışık çizilen iki daire arasında hesaplandığı için beyaz bölgenin içerisine çizilen ilk daire ile siyah bölgeye çizilen son daire arasındaki fark değeri en yüksek olacaktır.

J. Daugman tarafından önerilen yöntem yüksek çözünürlüklü resimlerde tatmin edici sonuçlar vermektedir. Ancak düşük çözünürlükteki resimlerde aynı başarıyı gösterememektedir. Tarama işleminde irisin alt ve üst kısımlarının göz kapaklarıyla kapanması durumu dikkate alınmamakta ve daire üzerindeki tüm pikseller hesaba dahil edilmektedir. Bu durum algoritmanın doğruluğunu azaltmaktadır.

3.2 Yeni Amaç Fonksiyonunun Oluşturulması

Göz merkezinin hassas bir şekilde belirlenmesi için Timm ve Barth tarafından önerilen ve tez çalışmasında göz merkezinin kabaca belirlenmesi amacıyla kullanılan yöntem ile J. Daugman tarafından önerilen yöntem birleştirilerek yeni bir amaç fonksiyonu geliştirilmiştir.

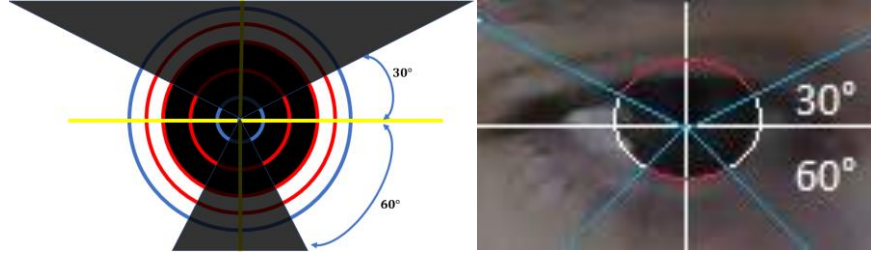
Geliştirilen amaç fonksiyonu, birleştirilen iki yöntemin güçlü yönlerinden faydalanmakta ve bu yöntemleri birbirlerini tamamlayacak şekilde kullanmaktadır. Timm ve Barth tarafından sunulan ve göz merkezini kabaca bulmak için kullanılan yöntemde, göz bölgesi sınırlarında bulunan pikseller hariç diğer tüm pikseller nokta çarpım işlemine dahil edilmektedir. Geliştirilen yeni amaç fonksiyonunda, göz

hesaplanmaktadır. Şekil 3.3 (a)'da gösterildiği gibi bir merkez için hesaba dahil edilen noktaların belirlenmesinde herhangi bir dairesel seçim stratejisi kullanılmamış, göz bölgesi sınırları üzerinde olmayan tüm noktalar hesaba dahil edilmiştir. Şekil 3.3 (b)'de J. Daugman tarafından önerilen yöntemde kullanılan dairesel tarama yaklaşımı gösterilmiştir. Yöntemde her bir dairenin üzerindeki noktaların $I(x,y)$ gri seviye değerleri toplanmaktadır. Çizilen dairelerin yarıçap değerleri, belirlenen r_{min} minimum yarıçapı ve r_{max} maksimum yarıçapı aralığında seçilmektedir. Şekil 3.3(c)'de tez çalışmasında önerilen iyileştirme yaklaşımı gösterilmiştir. Sunulan yöntemde, Timm ve Barth tarafından önerilen nokta çarpım yaklaşımı, J. Daugman tarafından önerilen amaç fonksiyonuna dahil edilmiştir. Bunun için J. Daugman tarafından ortaya konulan amaç fonksiyonundaki gri seviye değeri yerine Timm ve Barth tarafından önerilen, nokta çarpımın karesi ifadesi yazılmıştır.

$$I(x,y) \rightarrow (d_{(x,y)}^T g_{(x,y)})^2 \quad (3.2)$$

$I(x,y)$ muhtemel merkez noktası etrafında çizilen bir daire üzerindeki noktanın gri seviye değerini, $d_{(x,y)}$ muhtemel merkezden bu noktaya çizilen mesafe birim vektörünü, $g_{(x,y)}$ ise daire üzerindeki noktanın normalize edilmiş gradyan vektörünü ifade etmektedir.

İris geometrik olarak daire şeklinde olsa da genellikle göz kapağı, kirpik gibi gözün diğer bölümleri tarafından kapatıldığı için işlenen göz resimlerinde tam daire şeklinde görünmemektedir. Göz kapakları tarafından kapatılan alt ve üst iris bölümleri, görünen iris bölümleri ile göz akı arasındaki yüksek gri seviye farkını göstermemektedir. Bu nedenle kapanan kısımlardan elde edilen gradyan vektörünün büyüklüğü ve doğrultusu yanıltıcı sonuçlara sebep olabilmekte ve algoritmanın doğruluğunu azaltmaktadır. Tez çalışmasında, bu durumun önüne geçmek için çizilen daireler üzerinde bulunan noktalardan sadece belirlenen açı aralığında kalanları nokta çarpıma dahil edilmiştir. Belirlenen açı aralığı irisin genellikle görünür olan kısımlarını kapsamaktadır. İrisin görünür kısmını ifade eden açı değerleri, saatin tersi yönünde 30 derece ve saat yönünde 60 derece olarak seçilmiştir. Bu açılar için elde edilen noktalar ve bu noktaların y eksenine göre simetrikleri amaç fonksiyonuna dahil edilmiştir. Şekil 3.4'te nokta çarpıma dahil edilecek noktaları kapsayan açı aralığı görülmektedir.



Şekil 3.4 : Nokta çarpıma dahil edilen noktaları kapsayan açı aralığı.

Denklem 3.3'te J. Daugman tarafından önerilen amaç fonksiyonu ile Timm ve Barth tarafından önerilen yöntemin birleştirilmesi ile elde edilen yeni amaç fonksiyonu verilmiştir.

$$\operatorname{argmax}_{(r,x_c,y_c)} \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r,x_c,y_c} \frac{(d_{(x,y)}^T g_{(x,y)})^2}{\theta r} ds \right| \quad (3.3)$$

Denklemde $d(x,y)$, muhtemel merkez noktasını ifade eden (x_c,y_c) ile o anda ele alınan daire üzerindeki (x,y) noktası arasındaki mesafe vektörünü, $g(x,y)$ daire üzerindeki noktanın gradyan vektörünü, G_σ gauss filtresini ve θ nokta çarpıma dahil edilecek noktaları kapsayan açı aralığını ifade etmektedir.

Timm ve Barth tarafından önerilen yöntemde kullanılan ve iris bölgesindeki noktaların merkez olma ihtimallerini arttıran w_c ağırlığı geliştirilen amaç fonksiyonuna da eklenmiştir. Denklem 3.4'te önerilen amaç fonksiyonunun ağırlık eklenmiş hali görülmektedir.

$$\operatorname{argmax}_{(r,x_c,y_c)} \left| G_\sigma(r) * \frac{\partial}{\partial r} \oint_{r,x_c,y_c} w_c \frac{(d_{(x,y)}^T g_{(x,y)})^2}{\theta r} ds \right| \quad (3.4)$$

Tez çalışmasında geliştirilen göz merkezi belirleme yöntemi, birleştirilen iki yöntemle göre de üstünlük göstermektedir. Timm ve Barth, önerdikleri yöntemde nokta çarpıma dahil edilecek noktaları dairesel tarama ile belirlememekte ve iris yarıçapının olabilecek maksimum ve minimum değerlerini dikkate almamaktadır. Bu nedenle uygulanması gereken işlem sayısı artmakta ve göz merkezi belirleme doğruluğu olumsuz etkilenmektedir. Ayrıca önerdikleri yöntemde sadece göz merkezi belirlenmekte, iris yarıçapının değeri elde edilememektedir. Birleştirilen diğer yöntem olan J. Daugman tarafından önerilen yöntem, dairesel bir tarama işlemi uygulayarak daire üzerindeki noktaların gri seviye değerlerini kullanmaktadır.

Yöntem yüksek çözünürlüklü ve IR aydınlatma altında elde edilen göz resimlerde başarılı olsa da düşük çözünürlüklü resimlerde doğruluk değeri azalmaktadır. Ayrıca irisin alt ve üst kısımlarının göz kapaklarıyla kapatılmış olabileceği durumu dikkate alınmadan, daire üzerindeki tüm noktalar hesaba dahil edilmektedir. Tez çalışmasında önerilen yöntemde, J. Daugman tarafından önerilen yöntemde hesaplanan, daire üzerindeki noktaların gri seviye değerlerinin toplamı işlemi yerine Timm ve Barth tarafından önerilen nokta çarpımların karelerinin toplamı işlemi uygulanmıştır. Yapılan bu değişiklikle elde edilen yeni amaç fonksiyonu, J. Daugman tarafından önerilen amaç fonksiyonunun aksine düşük çözünürlüklü resimlerde de göz merkezini yüksek doğrulukla belirleyebilmektedir. Tez çalışmasında önerilen yöntem, daire üzerindeki irisin görünen bölümünü temsil eden açı aralığındaki noktaları hesaba dahil etmektedir. Böylece göz kapaklarının, irisin alt ve üst kısımlarını kapatması durumunda da yöntemin doğruluğu korunmaktadır. Tez çalışmasında geliştirilen göz merkezi bulma algoritması aşağıda verilmiştir.

Çizelge 3.1 : Tez çalışmasında geliştirilen göz merkezi iyileştirme algoritması.

Geliştirilen İyileştirme Algoritması

```

1: Radım=1
2: AçıSayısı=100
3: Tadım= 2*PI/AçıSayısı
4: for c ∈ {1, ... OlasıMerkezSay} do
5:   for r ∈ {1, ... YarıçapSayısı} do
6:     yarıçap=minYarıçap+r*Radım;
7:     for theta ∈ {1, ... AçıSayısı} do
8:       if th > AçıSayısı / 12 and th < 5 * AçıSayısı /12 then
9:         continue
10:      end if
11:      if th >4* AçıSayısı / 6 and th < 5 * AçıSayısı /6 then
12:        continue
13:      end if
14:      Pi=NoktayıHesapla(c,theta)
15:      di = (Pi - c)/|Pi - c|
16:      gi=(∂I(xi,yi)/∂xi , ∂I(xi,yi)/∂yi)
17:      gi= gi/|gi|
18:      val=val+(diTgi)2
19:    end for
20:    rTürev=TürevAl(rSum)
21:    rGauss=GaussFiltresiUygula(rDiff)
22:    MaxMerkez(c),MaxDeğer(c)=MaxBul(rGauss)
23:  end for
24: end for
25: EniyiIndex,EniyiMerkez= MaxBul(MaxDeğer)
26: EniyiYarıçap= (minYarıçap + yarıçap * Radım)
27: Çıkış: EniyiMerkez,EniyiYarıçap

```

3.3 Önerilen Göz Merkezi Belirleme Yaklaşımının Deneysel Sonuçları

3.3.1 BioID veri seti

Tez çalışmasında geliştirilen göz merkezini belirleme yaklaşımı, bu alanda yaygın olarak kullanılan BioID [47] veri seti kullanılarak test edilmiştir. Veri seti 23 farklı kişiden alınmış olan 1521 adet resimden oluşmaktadır. Resimlerdeki kişilerin kameraya uzaklıkları, pozları ve ortamın aydınlatma şartları farklılık göstermektedir. Veri setindeki tüm resimler 286x384 çözünürlüğe sahiptir. Bazı kişiler gözlük takmaktadır, bazı resimlerde ise gözlerin tamamı veya çoğunluğu kapalı durumdadır. Bu nedenle BioID veri seti göz merkezi belirleme alanında kullanılan en zorlu veri setlerinden biri olarak kabul edilmektedir.

Denklem 3.5'te hata değerlerinin hesaplanması görülmektedir. Hata değeri, veri setinde verilen gerçek merkez noktası ile bulunan merkez noktası arasındaki öklid mesafesi kullanılarak hesaplanmaktadır.

$$e = |P_{gerçek} - P_{bulunan}| \quad (3.5)$$

Her iki göz içinde ayrı ayrı hata değerleri hesaplanmaktadır. Literatürdeki çalışmalarda, sağ ve sol göz için bulunan hata değerlerinden daha yüksek olanı karşılaştırma amacıyla kullanılmaktadır. Karşılaştırma için kullanılacak ölçüt, hata değeri daha yüksek olan gözün hatasının gözler arası mesafeye bölünmesi ile elde edilmektedir.

$$e_{enkötü} \leq \frac{1}{d} \max(e_{sağ}, e_{sol}) \quad (3.6)$$

Denklem 3.6'da $e_{sağ}$ ve e_{sol} sırasıyla sağ ve sol göz için elde edilen hata değerini, d ise gerçek göz merkezleri arasındaki mesafeyi ifade etmektedir. Bu karşılaştırma ölçütüne bakılarak belirlenen noktanın göz içerisinde hangi bölgede olduğunu tespiti yapılabilmektedir. (i) $e \leq 0.25$ bulunan noktanın göz bölgesi içerisinde olduğunu ifade etmektedir. Bu hata değeri göz merkezi ile göz kenarları arasındaki mesafeye eşittir. (ii) $e \leq 0.1$ iris çapına eşittir. Bulunan noktanın iris sınırları içerisinde olduğunu ifade etmektedir. (iii) $e \leq 0.05$ göz bebeği çapına eşittir. Bulunan noktanın göz bebeği sınırları içerisinde olduğunu ifade etmektedir. Tez çalışmasında kötü göze göre verilen hata değerinin yanında, en iyi göze göre ve ortalamaya göre

sonular da elde edilmiřtir. Denklem 3.7’de hatası dřk olan gze gre, denklem 3.8’de ise ortalama hata deęerine gre belirlenen karřılařtırma ltrleri grlmektedir.

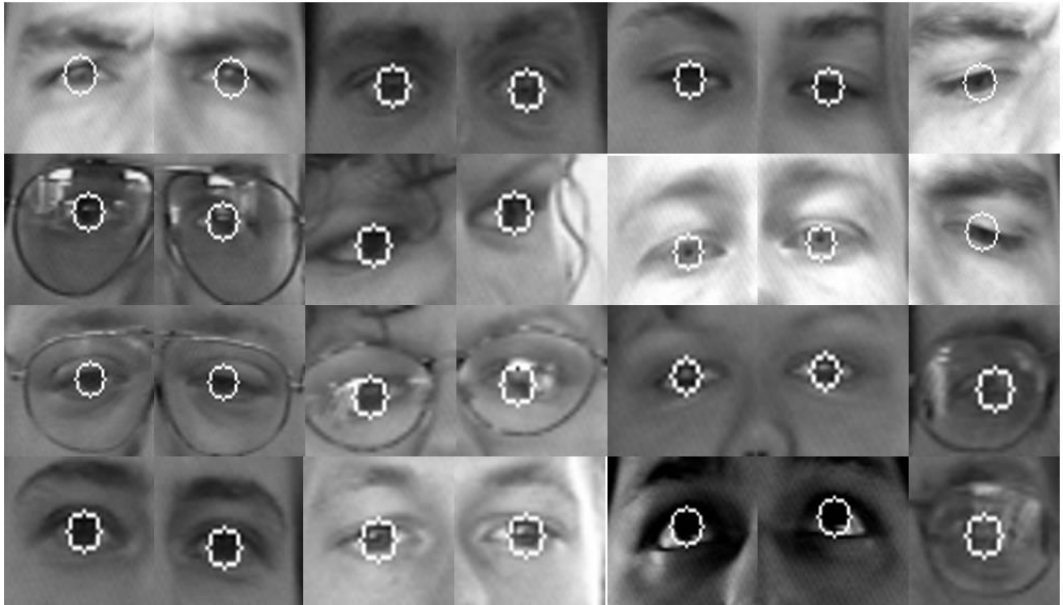
$$e_{eni} \leq \frac{1}{d} \min(e_{saę}, e_{sol}) \quad (3.7)$$

$$e_{ort} \leq \frac{1}{2d} (e_{saę} + e_{sol}) \quad (3.8)$$

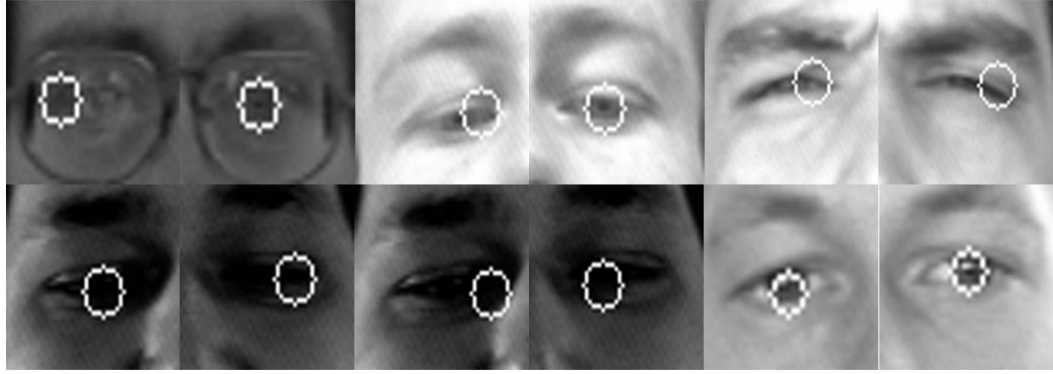
3.3.2 Deneysel Bulgular

BioID veri setindeki resimlere, geliřtirilen ok ařamalı gz merkezi belirleme yntemi uygulanmıřtır. İlk olarak resimlerdeki kiřilerin yz blgeleri belirlenmiřtir. Yz algılama iřlemi iin Viola ve Jones tarafından nerilen yntem kullanılmıřtır [45]. Belirlenen yz blgesi ierisindeki saę ve sol gz blgeleri, yz blgesi ile gz blgeleri arasındaki geometrik iliřki kullanılarak elde edilmiřtir. Elde edilen gz blgeleri ierisindeki gz merkezleri, Timm ve Barth tarafından nerilen yntem kullanılarak kabaca belirlenmiřtir [15]. Son olarak, kabaca belirlenen gz merkezleri etrafındaki 10x10 boyutlarındaki piksel blgesinde kalan 100 nokta iin tez alıřmasında nerilen iyileřtirme algoritması uygulanmıřtır.

řekil 3.5’te elde edilen bazı doęru sonular, řekil 3.6’da ise bazı yanlış sonulara ait resimler grlmektedir.



řekil 3.5 : Gz merkezi belirleme iřlemine ait bazı doęru sonular.



Şekil 3.6 : Göz merkezi belirleme işlemine ait bazı yanlış sonuçlar.

Çizelge 3.2’de en kötü göz, en iyi göz için doğruluk değerleri ve ortalama doğruluk değerleri görülmektedir.

Çizelge 3.2 : En kötü göz, en iyi göz ve gözlerin ortalaması için doğruluk değerleri.

	Doğruluk ($e \leq 0.05$)	Doğruluk ($e \leq 0.10$)	Doğruluk ($e \leq 0.25$)
En Kötü Göz	88,70%	94.00%	98.00%
Ortalama	92.90%	96.69%	98,83%
En İyi Göz	97,27%	98,58%	98.93%

Çizelge 3.3’te tez çalışmasında önerilen yöntemin literatürdeki diğer yöntemlerle karşılaştırılma sonuçları görülmektedir.

Çizelge 3.3 : Önerilen göz merkezi belirleme yönteminin diğer yöntemlerle karşılaştırılması.

Metod	Doğruluk ($e \leq 0.05$)	Doğruluk ($e \leq 0.10$)	Doğruluk ($e \leq 0.25$)
Cristinacce [48]	56.00%	96.00%	98.00%
Bai [49]	37.00%	64.00%	96.00%
Campadelli [50]	62.00%	85.20%	96.10%
Timm [15]	82.50%	93.40%	98.00%
Valenti [16]	86.09%	91.67%	97.87%
Garg [19]	77.60%	88.70%	98.60%
Turkan [51]	18.60%	73.70%	99.60%
Önerilen Metot	88,70% (1)	94.00% (2)	98.00% (3)

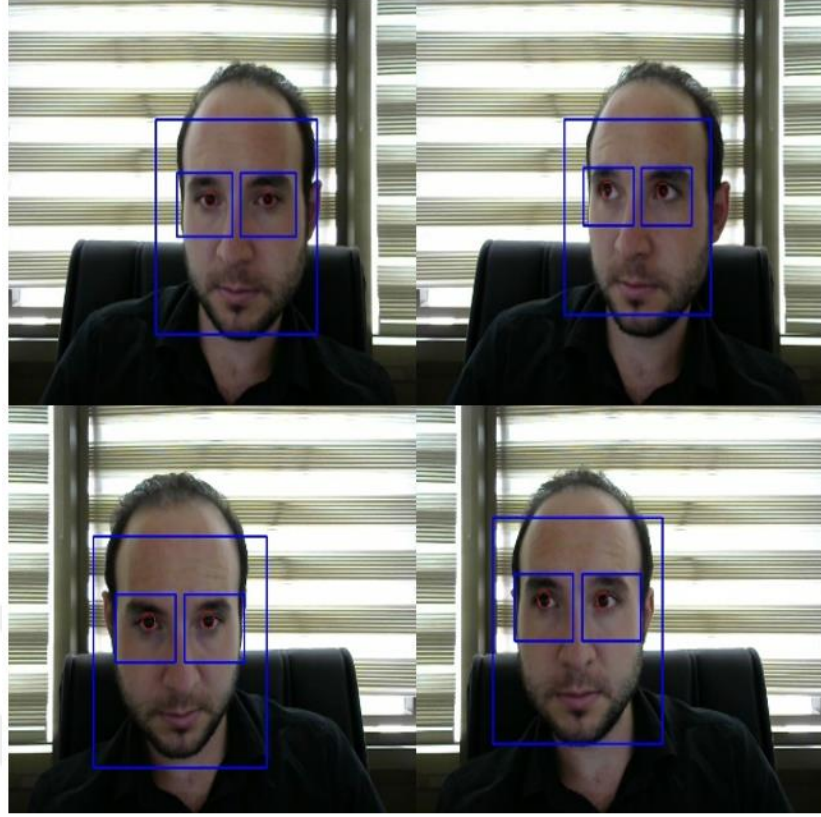
Çizelge 3.3’te görüldüğü gibi tez çalışmasında önerilen göz merkezi belirleme yöntemi, tüm hata aralıkları için tatmin edici sonuçlar vermektedir. Göz merkezini göz bebeği sınırları içerisinde bir nokta olarak belirlemede ($e \leq 0.05$) ilk sırada, iris içerisinde bir nokta olarak belirlemede ($e \leq 0.10$) ikinci sırada, göz bölgesi içerisinde bir nokta olarak belirlemede ($e \leq 0.25$) ise üçüncü sırada yer almaktadır. Tez çalışmasında hedeflenen, göz bebeği sınırları içerisinde olacak şekilde göz merkezini

belirlemektir. Elde edilen sonuçlar önerilen yöntemin, istenilen amaca ulaştığını göstermektedir. Çizelge 3.2 incelendiğinde tüm hata aralıklarında en doğru sonucu veren herhangi bir yöntem bulunmadığı görülmektedir. Örneğin, Turkan [51] tarafından önerilen yöntem, göz içerisinde bir nokta belirleme konusunda ilk sırada olmasına rağmen, iris veya göz bebeği içerisinde bir nokta bulmada kötü sonuçlar vermektedir. En zorlu hata aralığında ($e \leq 0.05$) ilk sırada olan yöntemimize en yakın doğruluk değerine sahip yöntem, %86.09 doğruluk değerine sahip, Valenti tarafından önerilen yöntemdir. Valenti [16] tarafından önerilen yöntemde, kümeleme, sınıflandırma ve SIFT özneliklerinin çıkarımı gibi algoritmalara ihtiyaç duyulduğu için işlem karmaşıklığı tezde önerilen yöntemle göre daha fazladır. Tez çalışmasında, göz merkezinin kabaca belirlenmesi amacıyla kullanılan, Timm ve Barth tarafından önerilen yöntem [15], göz bebeği sınırları içerisinde bir nokta belirlemede ($e \leq 0.05$) % 82.5 doğruluk değeri ile üçüncü sırada yer almaktadır. Bu yöntemle elde edilen kabaca belirlenmiş göz merkezi etrafına, tezde önerilen iyileştirme yaklaşımının uygulanması ile doğruluk değeri %6.5 arttırılarak %88.70'e çıkarılmıştır.

3.3.3 Gerçek zamanlı sonuçlar

İnsan bilgisayar etkileşimi uygulamaları, genellikle gerçek zamanlı olarak çalışacak şekilde geliştirilmektedir. Bu nedenle tezde geliştirilen yöntemin gerçek zamanlı çalışma performansının değerlendirilmesine de ihtiyaç vardır. Gerçek zamanlı uygulama için donanım olarak sadece bir bilgisayar ve kamera kullanılmıştır. Kamera olarak Logitech C310 kullanılmıştır. Gerçek zamanlı sonuçlar Intel i7-2.70 GHz 4 çekirdekli işlemciye ve 8 Gb ram'e sahip bir bilgisayar kullanılarak elde edilmiştir.

Gerçek zamanlı sonuçların elde edilmesi için ilk olarak Viola ve Jones tarafından önerilen metot [45] kullanılarak yüz belirleme işlemi gerçekleştirilmiştir. Belirlenen yüz bölgesi içerisinde sağ ve sol göz bölgeleri, yüz ile göz bölgeleri arasındaki geometrik ilişkiden faydalanılarak belirlenmiştir. Elde edilen sağ ve sol göz bölgelerine, Timm ve Barth tarafından önerilen göz merkezi belirleme yaklaşımı [15] uygulanarak göz merkezlerinin konumu kabaca belirlenmiştir. Kabaca belirlenen göz merkezleri etrafına, tez çalışmasında geliştirilen iyileştirme metodu uygulanarak göz merkezlerinin tam konumu ve iris yarıçapları belirlenmiştir. Şekil 3.7'de gerçek zamanlı olarak elde edilen bazı sonuçlar görülmektedir.



Şekil 3.7 : Göz merkezi ve iris yarıçapı belirleme algoritmasıyla elde edilen bazı sonuç resimleri.

Önerilen göz merkezi belirleme yöntemi, iteratif bir yöntem olduğu için paralel olarak gerçekleştirilmeye uygun yapıdadır. Bu nedenle algoritma, tek çekirdek ve dört çekirdek kullanılarak ayrı ayrı test edilmiştir. Çizelge 3.3'te işlemcinin tek çekirdeği ve dört çekirdeği kullanıldığında göz merkezi belirleme işlemi için geçen süreler görülmektedir.

Çizelge 3.4 : Göz belirleme algoritmasının tek ve dört çekirdek için işleme hızları.

	Tek Çekirdek	Dört Çekirdek
Kabaca Belirleme Aşaması	32ms	7.2 ms
İyileştirme Aşaması	6.4ms	2.3 ms
Toplam Geçen Süre	38.4ms	9.5 ms

Çizelge 3.4'te görüldüğü gibi göz merkezi belirleme algoritmasının paralel bir şekilde uygulanması işlem süresini yaklaşık dört kat kısaltmıştır. Göz merkezinin kabaca belirlenmesinde, göz bölgesindeki piksellerin çoğu işleme dahil edildiği için bu aşamadaki iterasyon sayısı iyileştirme aşamasına göre çok daha fazladır. Bu nedenle iterasyonların paralel bir şekilde uygulanması, göz merkezinin kabaca belirlenmesi için geçen süreyi kayda değer bir şekilde azaltmıştır.

Göz merkezinin kabaca belirlenmesi aşamasında, hız-doğruluk dengesinin sağlanması için göz bölgesinin belirli oranda küçültülmesi gerekmektedir. Çizelge 3.4'te sunulan sonuçlar, göz bölgesinin %60'ına kadar küçültülmesi ile elde edilen sonuçlardır. Daha yüksek doğruluk değerleri elde etmek için işlem yapılan göz bölgesi, orjinal boyutlarında veya yakın boyutlarda seçilmelidir. Bu durumda iterasyon sayısı ciddi derecede artacağı için tek çekirdek kullanımı ile dört çekirdek kullanımı arasındaki süre farkı çok daha fazla olacaktır. Çizelge 3.5'te farklı boyutlarda alınan göz bölgeleri için elde edilen işlem süreleri gösterilmektedir.

Çizelge 3.5 : Farklı boyuttaki göz bölgeleri için işlem süreleri.

Çekirdek Sayısı	Kullanılan Göz Bölgesi Boyutu			
	% 40	% 60	% 80	% 100
Tek Çekirdek	9.5 ms	32 ms	55 ms	175 ms
4 Çekirdek	1.5 ms	7.2 ms	11 ms	50 ms

4. KAFA POZUNUN BELİRLENMESİ

Kafanın üç boyutlu (3B) uzaydaki konumu ve oryantasyonunun belirlenmesi, insan bilgisayar etkileşimi için önerilen yöntemlerde çözülmesi gereken başlıca problemlerdendir. Bu probleme çözüm getirmek için önerilen yöntemlerin amacı, kamera tarafından görüntülenen insan kafasının, üç adet dönme ve üç adet öteleme olmak üzere toplam altı serbestlik derecesine sahip hareketlerinin tahminini gerçekleştirmektir.

Kafa pozunu algılanması amacıyla kullanılan yöntemler, genel olarak kafanın bir model ile temsiline dayanmaktadır. AGM [38] veya AŞM [36] gibi esnek modeller, yüzdeki kaş, ağız kenarı gibi yerel bölgelerin hareketlerini temsil edebilirken, silindir, elips gibi sabit modeller [9, 41, 42] kafayı bir bütün olarak ele almaktadır. Esnek modeller için hesaplanması gereken parametre sayısı, sabit modellere göre çok daha fazladır. Bu nedenle sabit modellerde model parametrelerinin hesaplanması, esnek modellere göre daha hızlı yapılabilmektedir. Eğer yüzdeki yerel hareketlerle ilgilenilmeden, sadece kafa pozunu tahmini yapılacak ise sabit bir modelin kullanılması daha avantajlıdır.

Tez çalışmasında amaçlanan yerel hareketlere bakmaksızın sadece kafa pozunun tahminini gerçekleştirmektedir. Bu nedenle kafa modeli olarak sabit bir silindir modeli kullanılmıştır. Kullanılan silindir modeli, kafanın 3B hareketlerini temsil etmektedir. Silindirin 3B uzaydaki pozunun tahmini için öznitelik takibine dayalı bir yaklaşım kullanılmıştır. Takibe dayalı yöntemlerde, başlangıç resmi üzerinde öznitelik noktaları belirlenmekte ve bu noktaların ardışık resimlerde takibi gerçekleştirilmektedir. Takip edilecek noktaların belirlenmesi, kafa pozunun tahmininin doğruluğunu etkileyen en önemli faktörlerdendir. Bu nedenle takip için seçilecek olan noktaların, farklı poz ve aydınlatma koşullarında bile bulunabilir olması çok önemlidir. Başlangıç resminde takip edilecek noktalar belirlendikten sonra bu noktaların, kafayı temsil eden 3B silindir model üzerindeki koordinatları belirlenmektedir. Daha sonra, başlangıç resminde belirlenen noktalar, Lucas-Kanade optik akış algoritması [52] kullanılarak ardışık resimler arasında takip edilmektedir.

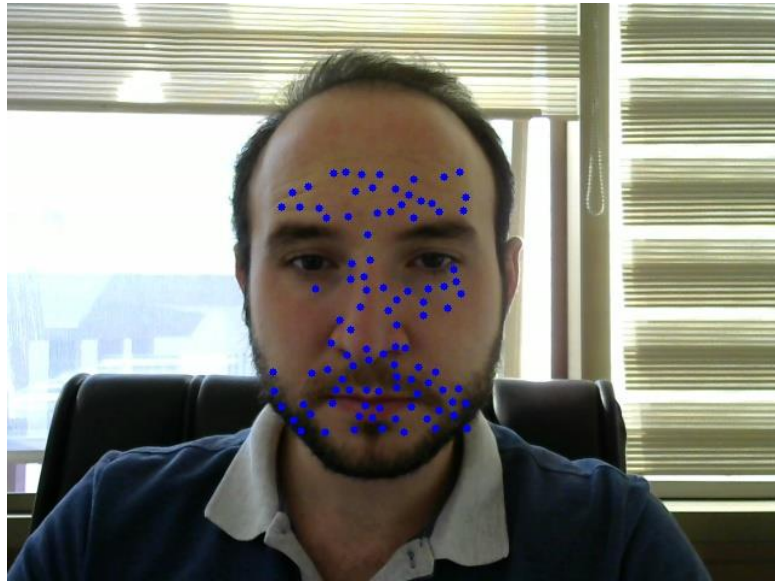
Öznelik noktalarının model üzerindeki 3B başlangıç koordinatları ile herhangi bir anda alınan resim üzerindeki iki boyutlu (2B) koordinatları arasındaki dönüşüm matrisi bulunarak modelin bir sonraki poza geçmesi için gerekli olan dönme ve öteleme miktarı belirlenmektedir. Takip edilen noktalar, tüm resimlerde aynı noktalar olduğu için başlangıçta elde edilen 3B nokta koordinatları ile herhangi bir anda elde edilen 2B resim koordinatları arasındaki 3B-2B dönüşüm matrisinin bulunması ile modelin başlangıç haline göre pozunu elde edilebilmektedir. Tez çalışmasında kullanılan takip tabanlı poz tahmini yaklaşımı aşağıdaki adımlardan oluşmaktadır;

- i. Yüz bölgesinde takip edilmeye uygun öznelikler Harris köşe bulma algoritması [53] kullanılarak belirlenir.
- ii. Daha önce belirlenen göz merkezleri arasındaki mesafe kullanılarak kafanın kameraya olan başlangıç mesafesi belirlenir. Silindir denklemi, pinhole kamera modeli ve belirlenen mesafe verisi kullanılarak noktaların model üzerindeki koordinatlarını belirlenir.
- iii. Lucas-Kanade optik akış algoritması kullanılarak özneliklerin bir sonraki resimdeki koordinatları belirlenir [52].
- iv. Optik akış ile belirlenen yeni noktalardan sadece iyi takip edilebilmiş olanları poz tahmini işlemine dahil edilir.
- v. Noktaların model üzerindeki 3B başlangıç koordinatları ile takip edilerek bulunan, yeni resimdeki 2B koordinatlar arasındaki dönüşüm matrisi belirlenir. Bu dönüşüm matrisi, silindir kafa modelinin başlangıç durumuna göre ne kadar döndürülmesi ve ötelenmesi gerektiği verisini tutmaktadır.
- vi. Optik akış algoritması ile doğru bir şekilde takip edilemeyen noktaların yeni resim üzerindeki koordinatları, bu noktaların başlangıçtaki 3B koordinatları ve iyi takip edilen noktalardan elde edilen dönüşüm matrisi kullanılarak belirlenir.
- vii. Eğer optik akış algoritması sonucu iyi takip edilebilmiş nokta sayısı belirli bir değerin altında ise elde edilecek olan dönüşüm matrisi güvenilir olmayacaktır. Bu nedenle, takip edilebilen nokta sayısı belirli bir sınır değerden daha az ise ilk adıma gidilir ve model yeniden başlatılır. Eğer bu sınır değerin üzerinde ise adım 3'e gidilir.

4.1 Takip İçin Uygun Öznitelik Noktalarının Belirlenmesi

Kameradan alınan resim içerisindeki herhangi bir nesnenin takibi için nesnenin hareketlerini modelleyebilecek bazı özniteliklerin kullanılması, takip çalışmalarındaki yaygın olarak kullanılan bir yaklaşımdır. Başlangıç resminden çıkarılan bu özniteliklerin, ardışık resimler üzerinde takip edilebilmesi için bu öznitelik noktalarının farklı pozlarda bile algılanabilmeleri gerekmektedir. Bu öznitelikler, resmin geneline göre farklı özellikler göstermeli ve farklı pozlarda bile resimdeki diğer bölgelerden ayırt edilebilmelidir. Bu özelliklere sahip öznitelikler, genellikle köşe noktaları ve/veya kenar noktaları olmaktadır. Özniteliklerin çıkarılması için literatürde önerilmiş birçok yöntem bulunmaktadır [53–55]. Tez çalışmasında, öznitelik noktalarının belirlenmesi için basitliği, hızı ve doğruluğu nedeniyle Harris köşe algılama yöntemi [53] tercih edilmiştir.

Bir resimdeki köşe noktaları, hem x hem de y doğrultusunda yüksek gri seviye farkı, kenar noktaları yalnızca x veya y doğrultusunda yüksek gri seviye farkı, çeşitliliğin az olduğu diğer bölgeler ise her iki yönde de düşük gri seviye farkı göstermektedir. Harris köşe algılama yöntemi, türeve dayalı bir matris oluşturarak noktaları yukarıda bahsedilen özelliklerine göre sınıflandırmayı amaçlamaktadır. Şekil 4.1’de Harris köşe algılama algoritmasının yüz bölgesine uygulanmasıyla elde edilen öznitelik noktaları görülmektedir.



Şekil 4.1 : Harris köşe algılama algoritmasının yüz bölgesine uygulanmasıyla elde edilen öznitelik noktaları.

Köşe noktalarının belirlenmesi için belirli bir w penceresi tüm resim üzerinde kaydırılmakta ve bu pencerenin kaydırılması ile elde edilen pikseller arasındaki gri seviye farkına bakılmaktadır. Buna göre oluşturulan amaç fonksiyonu;

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \quad (4.1)$$

şeklinde elde edilir.

Denklemden, $w(x, y)$ ilgili bölgeyi temsil etmektedir. Amaç fonksiyonu, bir noktanın gri seviye değeri ile bu noktanın x yönünde u , y yönünde v kadar ötelenmesiyle elde edilen yeni noktanın gri seviye değeri arasındaki farka bakılmaktadır.

Amaç fonksiyonu içerisindeki ifadeyi lineer hale getirmek için ifade Taylor serisine açılır ise;

$$E(u, v) = w(x, y) \sum_{x,y} u^2 I_x^2 + 2uv I_x I_y + v^2 I_y^2 \quad (4.2)$$

Matris şeklinde yazılır ise;

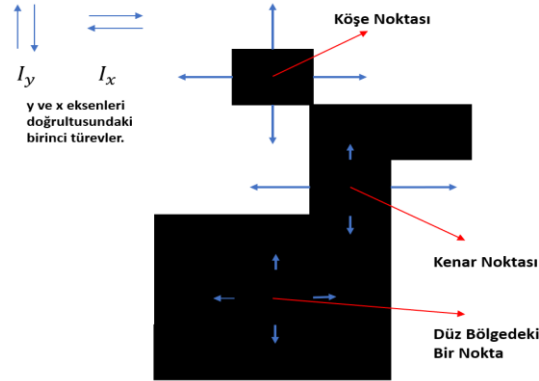
$$E(u, v) = w(x, y) \sum_{x,y} [u \ v] \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \quad (4.3)$$

$$E(u, v) = \sum_{x,y} [u \ v] M \begin{bmatrix} u \\ v \end{bmatrix} \quad (4.4)$$

elde edilir.

Buna göre, x ve y yönündeki türevler köşe ve kenar olma özelliği hakkında bilgi vermektedir [53]. Denklem 4.4'te verilen M matrisi, birinci ve ikinci türevler hakkındaki bilgiyi vermektedir. M matrisinin en yüksek ilk iki özdeğeri olan λ_1 ve λ_2 değerlerine bakılarak noktanın köşe veya kenar noktası olup olmadığına dair yorum yapılabilmektedir. Köşe noktaları için türev değeri her iki yönde de yüksek değerli olacağından, λ_1 ve λ_2 değerlerinin her ikisi de yüksek değere sahip olacaktır. Kenar noktaları için tek yönde türev değeri yüksek olacağı için λ_1 değeri yüksek, λ_2 değeri düşük olacaktır. Çeşitliliğin olmadığı, köşe ve kenar noktaları dışında kalan bölgelerdeki noktalar için ise her iki özdeğer de küçük değerler olacaktır.

Şekil 4.2’de köşe noktası, kenar noktası ve düz bölgede bulunan bir nokta için x ve y doğrultularında görülen türev değeri büyüklükleri görülmektedir.



Şekil 4.2 : Köşe noktası, kenar noktası ve düz bölgedeki bir nokta için türev değerlerinin büyüklüğü.

Denklem 4.5’te özdeğerler kullanılarak belirlenen R skorunun denklemi görülmektedir.

$$R = \det M - k(\text{trace} M)^2 \quad (4.5)$$

Burada;

$$\det M = \lambda_1 \lambda_2 \quad (4.6)$$

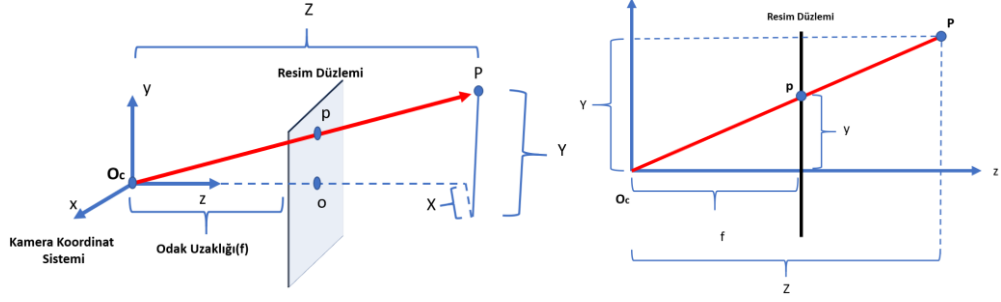
$$\text{trace} M = \lambda_1 + \lambda_2 \quad (4.7)$$

dir.

Burada, k değeri deneysel olarak belirlenmektedir ve genellikle 0.04-0.06 değerleri arasında alınmaktadır [53]. Skor değeri köşeler için pozitif ve büyük, kenarlar için negatif ve büyük, düz bölgelerde ise küçük değerdedir.

4.2 Pinhole Kamera Model

Günümüzde kullanılan kameraların, 3B nesnelere 2B resme dönüştürmesi işlemini modellemek için farklı kamera modelleri kullanılmaktadır. Şekil 4.3’te görülen Pinhole kamera modeli yaygın olarak kullanılan bir modeldir.



Şekil 4.3 : Pinhole kamera modeli.

Şekilde görüldüğü gibi 3B bir P noktasının resim düzlemindeki 2B karşılığı, 3B noktanın resim düzlemi üzerine izdüşürülmesi ile bulunmaktadır. 3B noktanın resim düzlemi üzerine izdüşümü olan iki boyutlu p noktası, şekilde görülen üçgenler arasındaki benzerlik kullanılarak hesaplanmaktadır [56].

$P = [X, Y, Z]^T$ nesnenin kamera koordinatlarına göre 3B konumu ve p resim düzlemindeki izdüşümü olmak üzere, p noktanın x ve y koordinatları aşağıdaki denklemler kullanılarak hesaplanmaktadır .

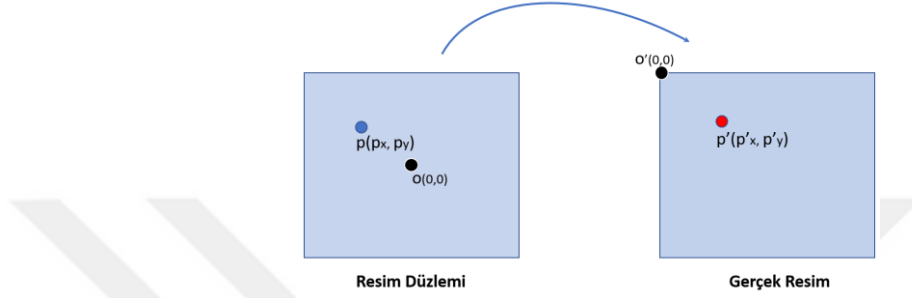
$$p_x = \frac{P_x f_x}{P_z} \quad (4.8)$$

$$p_y = \frac{P_z f_y}{P_z} \quad (4.9)$$

Denklemlerde f_x ve f_y sırasıyla, kameranın x ve y doğrultularındaki odak uzaklığını temsil etmektedir. Genellikle, x ve y doğrultularındaki odak uzaklıklarının eşit olduğu kabul edilmektedir. Odak uzaklığı kameranın dahili bir parametresidir ve önceden belirlenebilmektedir.

3B noktaların izdüşürüldüğü resim düzlemi ile bilgisayarda işlenen gerçek resim arasında farklar bulunmaktadır. Bu farklardan birisi, kullanılan ölçü birimidir. Resim düzlemi kameralar içerisinde bulunan fiziksel bir düzlemdir. Bu nedenle resim düzlemindeki bir noktanın koordinatları cm veya mm cinsinden elde edilmektedir. Bilgisayarda işlenen resimlerde ise ölçü birimi olarak piksel kullanılmaktadır. Bu yüzden resim düzleminde elde edilen bir noktanın, gerçek resimdeki karşılığı belirlenirken nokta bir katsayı ile çarpılmalıdır. Odak uzaklığı f 'nin piksel cinsinden değerinin kullanılması ile hesaplamalara bu katsayı dahil edilmektedir. İki resim

arasındaki diğer fark ise nokta koordinatlarının hesaplandığı referans noktasının farklı olmasıdır. Resim düzleminde koordinatları bulunan noktalar, merkez noktası referans alınarak belirlenmektedir. Bilgisayarda işlenen resimlerde ise piksel koordinatları, resmin sol üst köşe noktasına göre belirlenmektedir. Bu nedenle resim düzleminde elde edilen noktaların, gerçek resim üzerindeki karşılıklarının bulunması için bir dönüşüme ihtiyaç vardır. Şekil 4.4'te resim düzlemi ve gerçek resim için referans noktaları görülmektedir [57].



Şekil 4.4: Resim düzlemi ve gerçek resim için referans noktaları.

İki resim arasındaki oranlama işlemi için odak uzaklığı piksel cinsinden kullanırken, referans noktaları arasındaki farkın giderilmesi için bir öteleme işlemine ihtiyaç vardır. Bunun için elde edilen koordinatlar o_x ve o_y kadar ötelenmelidir. o_x ve o_y öteleme miktarları, yaklaşık olarak gerçek resmin orta noktası olarak alınmaktadır.

o_x : Resim genişliğinin yarısı

o_y : Resim yüksekliğinin yarısı

Ötelemenin eklenmesi ile elde edilen yeni noktalar;

$$p'_x = p_x + o_x \quad (4.10)$$

$$p'_y = p_y + o_y \quad (4.11)$$

şeklinde elde edilir. Bu işlemler matris haline getirilir ise;

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (4.12)$$

$$\vec{p} = K \vec{P} \quad (4.13)$$

elde edilir.

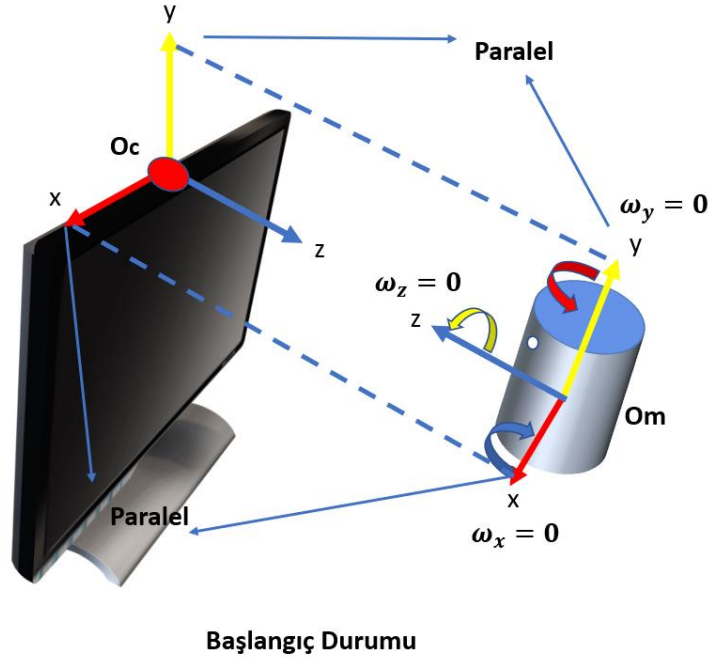
K matrisi, kamera matrisi olarak adlandırılmaktadır. Noktanın z derinliği biliniyor ise ters işlem uygulanarak 2B resim koordinatlarından, 3B koordinatlara geçiş yapılabilmektedir. Bunun için aşağıdaki eşitlikler kullanılır.

$$P_x = \frac{(p_x - o_x)}{f_x} P_z \quad (4.14)$$

$$P_y = \frac{(p_y - o_y)}{f_y} P_z \quad (4.15)$$

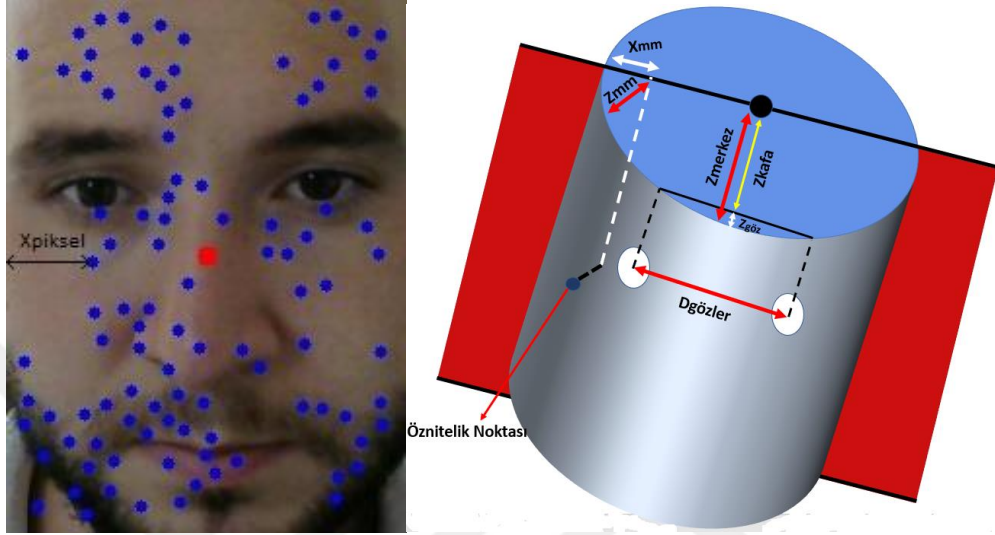
4.3 Modelin Başlatılması

Tez çalışmasında, insan kafası bir silindir olarak modellenmiştir. Silindir kafa modeli, insan kafasının karmaşıklığını tamamen modelleyememekle birlikte sadece kafa pozunun algılanması için kullanıldığında, tatmin edici ve işlemsel olarak basit bir çözüm sunmaktadır. Modelin başlatılması aşamasında, daha önce belirlenen öznitelik noktalarının, model üzerindeki 3B koordinatları belirlenmektedir. Bu işlem için kişinin kameraya düz bir şekilde baktığı kabul edilmektedir. Şekil 4.5'te başlangıç durumunda silindir kafa modelinin pozu görülmektedir.



Şekil 4.5 : Silindir kafa modelinin başlangıç durumu.

Kullanıcı ile kamera arasındaki başlangıç mesafesi, önceden belirlenen göz merkezleri arasındaki mesafe kullanılarak bulunmaktadır. Bu bilgiler ve silindir denklemini kullanarak resimdeki bir öznitelik noktasının 3B koordinatını elde etmek mümkündür. Şekil 4.6’da bir öznitelik noktasının silindir model üzerindeki konumunu görülmektedir.



Şekil 4.6 : Özniteliklerin silindir kafa modeline aktarılması.

Kullanıcının ekrana düz baktığı varsayımı yapıldığı için modelin x , y ve z eksenleri etrafındaki dönme miktarı sıfırdır. Silindirin yarıçap değeri ve gözler arasındaki mesafe, insanlar için ölçülen ortalama değerlere yakın seçilmiştir [58, 59].

Göz merkezleri arasındaki ortalama mesafe mm cinsinden sabit bir değer kabul edildiği için, pinhole kamera modeline göre gözleri birleştiren doğrunun kameraya olan uzaklığı aşağıdaki denklemle bulunabilir.

$$Z_{gözler} = \frac{f_l D_{mm}}{D_{piksel}} \quad (4.16)$$

Denklemde, f_l kameranın odak uzaklığını, D_{mm} milimetre cinsinden gözler arasındaki mesafeyi, D_{piksel} ise resimdeki gözler arası mesafeyi ifade etmektedir. $Z_{gözler}$ gözleri birleştiren doğru merkezinin kameraya olan uzaklığını ifade etmektedir. Silindir merkezinin kameraya olan uzaklığını belirlemek için Z_{kafa} değerinin belirlenmesi gerekmektedir. Z_{kafa} mesafesi, göz merkezlerini birleştiren doğrunun merkezinden silindir merkezine olan mesafedir. Bu mesafe değeri, silindirin yarıçapı ve gözler arasındaki mesafe değerlerine bağlı olarak bulunmaktadır [60].

Buna göre;

$$Z_{kafa} = \sqrt{r^2 - \left(\frac{D_{gözler}}{2}\right)^2} \quad (4.17)$$

elde edilir.

Şekil 4.6'da görüldüğü gibi silindir merkezinin kameraya olan uzaklığı, gözlerin uzaklığı ile Z_{kafa} değerinin toplanması ile bulunmaktadır.

$$Z_{merkez} = Z_{kafa} + Z_{gözler} \quad (4.18)$$

Silindir merkezinin kameraya olan uzaklığı belirlendikten sonra silindir üzerindeki herhangi bir noktanın 3B koordinatlarını belirlemek mümkündür. Şekil 4.6'da gösterildiği gibi yüz bölgesindeki herhangi bir öznitelik noktasının piksel cinsinden yüz çerçevesine olan x doğrultusundaki uzaklığı X_{piksel} ise bu noktaya karşılık gelen silindir üzerindeki X_{mm} değeri aşağıdaki denklemle bulunur [60].

$$X_{mm} = X_{piksel} \frac{D_{mm}}{D_{piksel}} \quad (4.19)$$

Silindir üzerinde bulunan ve x koordinatı bilinen bir noktanın z değeri, silindir denklemleri kullanılarak aşağıdaki gibi elde edilir.

$$z = r \sqrt{1 - \left(\frac{x}{r}\right)^2} \quad (4.20)$$

Denklemlerde bulunan z değeri, herhangi bir noktanın silindir merkezine göre derinlik verisidir. Bu noktanın kamera merkezine olan uzaklığının belirlenmesi için silindir merkezinin kameraya olan uzaklığından, z değerinin çıkarılması gerekmektedir.

$$Z_{nokta} = Z_{merkez} - z \quad (4.21)$$

Kameraya olan uzaklığı bilinen bir noktanın x ve y koordinatları, pinhole kamera modeli kullanılarak aşağıdaki gibi bulunur.

$$x_{nokta} = (x_{piksel} - o_x) \frac{Z_{nokta}}{f_l} \quad (4.22)$$

$$y_{nokta} = (y_{piksel} - o_y) \frac{z_{nokta}}{f_l} \quad (4.23)$$

Başlangıçta bulunan tüm öznitelik noktalarına bu işlemler uygulanarak noktaların kamera koordinatına göre 3B konumu belirlenmektedir. Tüm öznitelik noktalarının 3B koordinatlarının belirlenmesi ile modelin başlatılma işlemi tamamlanmaktadır.

4.4 Lucas-Kanade Optik Akış Algoritması ile Noktaların Takibi

Optik akış algoritması, öznitelik noktalarının ardışık resimler arasında takip edilmesi amacıyla kullanılmaktadır. Lucas-Kanade optik akış yöntemi [52], bir pikselin başlangıçtaki gri seviye değerinin, bir sonraki resimde de aynı kalacağı kabulüne dayanmaktadır. Optik akış algoritmasının sağlıklı sonuçlar üretebilmesi, aşağıda verilen şartların sağlanmasıyla mümkündür.

- i. Takip edilmek istenen noktanın bulunduğu ardışık iki resim arasındaki zaman farkı az olmalıdır.
- ii. Resimler arasında gerçekleşen gri seviye değişimi, yumuşak ve yavaş bir şekilde olmalıdır.
- iii. Bir pikselin komşu pikselleri, bu pikselle aynı hareketi gerçekleştirmelidir.

Optik akış yöntemine göre t anında alınan resimdeki bir noktanın gri seviye değeri, $t+dt$ anında alınan resimdeki değerine eşit olmalıdır.

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \quad (4.24)$$

Denklemden, d_t ardışık resimler arasında geçen zamanı, d_x ve d_y ise noktanın sırasıyla x ve y doğrultularında, t anından $t+dt$ anına kadar olan hareket miktarını göstermektedir. Optik akış algoritmasının başarılı olabilmesi için gerekli olan şartlardan ilkinin sağlanması için ardışık resimler arasında geçen süre olan dt yeterince küçük olmalıdır.

Denklem 4.24'te verilen ifade lineer değildir. Bu lineer olmayan denklem, Taylor serisine açılarak lineer hale getirilir ise aşağıdaki denklem elde edilir.

$$I(x, y, t) = I(x, y, t) + \frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt \quad (4.25)$$

+ "Yüksek Derece Terimler"

$$f_x dx + f_y dy + f_t dt = 0 \quad (4.26)$$

Denklem dt'ye bölünür ise;

$$u = \frac{dx}{dt} \quad , \quad v = \frac{dy}{dt} \quad (4.27)$$

$$f_x u + f_y v + f_t = 0 \quad (4.28)$$

elde edilir.

Burada f_x ve f_y , t anında alınan resimdeki noktanın sırasıyla x ve y doğrultularındaki türevini ifade etmekte, f_t ise $t+dt$ anında alınan resimdeki bir noktanın gri seviye değeri ile t anında alınan resimdeki aynı noktanın gri seviye değeri arasındaki farkı ifade etmektedir. Denklem, tek bir nokta için uygulanırsa bilinmeyenler, u ve v olmak üzere iki tane, denklem sayısı ise bir tane olacaktır. Problemin bu haliyle çözülmesi mümkün değildir. Bu nedenle belirli bir pencere içerisinde kalan noktaların aynı hareketi yaptığı ve bu noktaların da denklemi sağladığı kabul edilmiştir. Buna göre 3x3 piksellik bir penceredeki noktaların aynı şekilde hareket ettikleri varsayılır ise bu penceredeki tüm noktalar için hareket miktarları, u ve v eşit olacaktır. Denklem 4.28'de verilen ifade, pencere içerisindeki dokuz noktaya uygulanır ise;

$$\begin{aligned} f_{x1}u + f_{y1}v &= -f_{t1} \\ f_{x2}u + f_{y2}v &= -f_{t2} \\ \dots & \\ f_{x9}u + f_{y9}v &= -f_{t9} \end{aligned} \quad (4.29)$$

elde edilir. Denklemler matris vektör formunda yazılır ise;

$$\begin{bmatrix} f_{x1} & f_{y1} \\ f_{x2} & f_{y2} \\ \vdots & \vdots \\ f_{x9} & f_{y9} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -f_{t1} \\ -f_{t2} \\ \vdots \\ -f_{t9} \end{bmatrix} \quad (4.30)$$

$$A \vec{u} = -f_t \quad (4.31)$$

elde edilir.

Dokuz denklem olmasına karşın, iki adet bilinmeyen olduğu için bu denklem sisteminin kesin çözümü mümkün değildir. Bu nedenle en yakın çözümü verecek

olan, u ve v değerleri, en küçük kareler yöntemi kullanılarak elde edilmelidir. Denklemde f_t , olması gereken sonuçları ifade etmektedir. Çözüm için amaçlanan $A\vec{u}$ ile elde edilen sonucu, gerçek sonuçlara olabildiğince yaklaştıracak \vec{u} vektörünü belirlemektir. Bu nedenle elde edilen sonuçlar ile gerçek sonuçlar arasındaki farkı minimum hale getirecek u vektörü hesaplanmalıdır. Bu amaçla oluşturulan amaç fonksiyonu aşağıdaki gibidir.

$$\underset{u}{\text{minimum}} \|Au + f_t\|_2 \quad (4.32)$$

Bu ifade A matrisi içerisinde her bir terim için yazılır ise;

$$\underset{(x_i, y_i)}{\text{minimum}} \sum_i (f_{xi}u + f_{yi}v + f_{ti})^2 \quad (4.33)$$

elde edilir.

Herhangi bir fonksiyonun minimum noktası, fonksiyonun bağlı olduğu değişkenlere göre birinci türevlerinin sıfıra eşit olduğu noktadır. Buna göre denklem 4.33'te verilen ifadenin minimum değeri, u ve v değişkenlerine göre türevlerinin sıfıra eşit olduğu noktadadır.

u değişkenine göre türev;

$$\sum_i 2(f_{xi}u + f_{yi}v + f_{ti})f_{xi} = 0 \quad (4.34)$$

$$\sum f_{xi}^2 u + \sum f_{xi}f_{yi}v = -\sum f_{xi}f_{ti} \quad (4.35)$$

v değişkenine göre türev;

$$\sum_i 2(f_{xi}u + f_{yi}v + f_{ti})f_{yi} = 0 \quad (4.36)$$

$$\sum f_{yi}^2 v + \sum f_{xi}f_{yi}u = -\sum f_{yi}f_{ti} \quad (4.37)$$

İfadenin u ve v değişkenlerine göre türevleri sonucunda iki denklem elde edilmektedir.

Elde edilen iki denklem matris formunda yazılır ise;

$$\begin{bmatrix} \sum f_{xi}^2 & \sum f_{xi}f_{yi} \\ \sum f_{xi}f_{yi} & \sum f_{yi}^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -\sum f_{xi}f_{ti} \\ -\sum f_{yi}f_{ti} \end{bmatrix} \quad (4.38)$$

elde edilir.

Denklemden u ve v değişkenleri yalnız bırakılır ise;

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \sum f_{xi}^2 & \sum f_{xi}f_{yi} \\ \sum f_{xi}f_{yi} & \sum f_{yi}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum f_{xi}f_{ti} \\ -\sum f_{yi}f_{ti} \end{bmatrix} \quad (4.39)$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{\sum f_{xi}^2 \sum f_{yi}^2 - (\sum f_{xi}f_{yi})^2} \begin{bmatrix} \sum f_{xi}^2 & \sum f_{xi}f_{yi} \\ \sum f_{xi}f_{yi} & \sum f_{yi}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum f_{xi}f_{ti} \\ -\sum f_{yi}f_{ti} \end{bmatrix} \quad (4.40)$$

olur.

Denklemin 4.40'ın çözülmesi ile u ve v değerleri elde edilmektedir. Elde edilen u ve v değerleri, noktanın resimler arasındaki hareketini ifade etmektedir. Ancak denklemler elde edilirken Taylor seri açılımı kullanıldığı için bulunan u ve v değerleri tam sonuçları vermemektedir. Tam sonuçların elde edilmesi için iteratif bir yaklaşım kullanılmalıdır. İteratif yaklaşım aşamaları aşağıdaki gibidir [52];

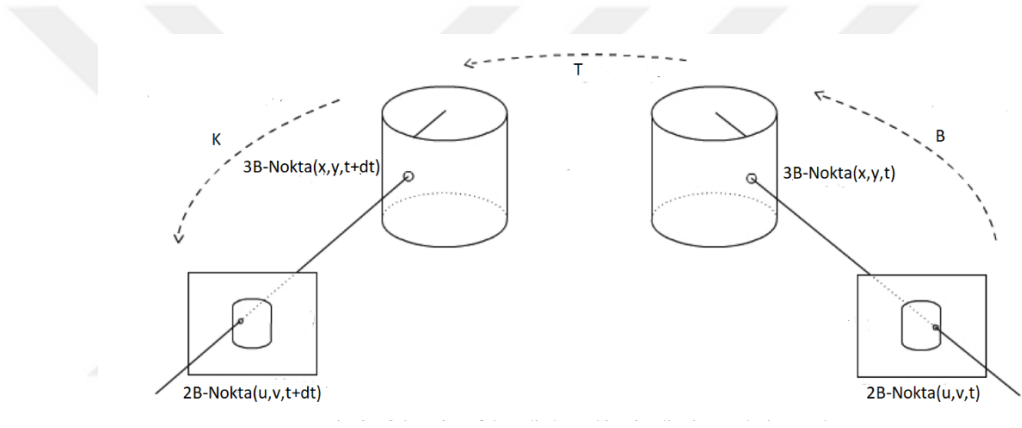
- i. İlk resimdeki, belirlenen pencere içerisindeki noktalar için f_x , f_y ve f_i değerlerini hesapla.
- ii. Denklem 4.40'ı kullanarak u ve v değerlerini hesapla.
- iii. Pencere içerisindeki tüm noktalara hesaplanan u ve v değerlerini ekle.
- iv. f_i değerini tekrar hesapla.
- v. u ve v değerleri çok küçük oluncaya kadar 2-4 arasındaki aşamaları tekrarla.

4.5 Pozun Belirlenmesi

Kafa pozunun tahmini, kafayı modelleyen silindirin kamera koordinat sistemine göre dönme ve yer değiştirme miktarını ifade etmektedir. Tez çalışmasında, silindir modelin belirli bir andaki pozunu, modelin başlangıç durumuna göre belirlenmektedir. Başlangıçta, kişinin kameraya baktığı ve kafasının yaklaşık olarak düz konumda olduğu varsayımı kullanıldığı için modelin başlangıçtaki dönme miktarı sıfır alınmış ve sadece z doğrultusunda bir başlangıç mesafesi olduğu varsayımında bulunulmuştur.

Modelin başlangıç durumunun bilinmesi, başlangıç resminde belirlenen herhangi bir öznitelik noktasının, silindir yüzeyindeki 3B karşılığının belirlenmesine imkân

vermektedir. Başlangıçta belirlenen öznitelik noktalarının 3B koordinatları, modelin başlangıç durumunu oluşturmakta ve poz verisi bu 3B noktalar referans alınarak bulunmaktadır. Başlangıçta 3B karşılığı elde edilen 2B öznitelik noktaları ardışık resimlerde takip edilmektedir. Takip edilen noktalar başlangıçta belirlenen noktalar olduğu için başlangıçta bulunan 3B koordinatlar ile ardışık resimlerde bulunan 2B koordinatlar arasındaki dönüşüm matrisinin bulunması, modelin ilk duruma göre dönme ve öteleme miktarlarının belirlenmesini sağlamaktadır. Şekil 4.7’de başlangıçta elde edilen 3B koordinatlar ile ilk resimde bulunan 2B öznitelik noktalarının bir sonraki resimde takip edilmesiyle elde edilen yeni 2B noktaların koordinatları arasında dönüşümü ifade eden şekil görülmektedir.



Şekil 4.7 : Ardışık resimlerle pozun belirlenmesi [9].

Şekil 4.7’de (u,v,t) 2B resim noktası, başlangıçta bulunan öznitelik noktalarından herhangi birisini temsil etmektedir. Bu noktanın kamera koordinatlarına göre 3B konumu, modelin başlangıç durumu ve silindir denklemi vasıtası ile belirlenmektedir. Burada B matrisi başlangıç durumunu ifade etmektedir. Bir sonraki resimde elde edilen aynı 2B öznitelik noktası, $(u,v,t+dt)$ ile ifade edilmektedir. K matrisi kamera koordinatlarına göre konumu bilinen 3B bir noktanın, resimdeki karşılığını elde etmek için gerekli olan kamera matrisidir. Pozun belirlenmesi için modelin başlatılması aşamasında elde edilen 3B noktalar ile yeni resimde elde edilen 2B noktaların 3B karşılıkları arasındaki dönüşüm matrisi olan T matrisinin belirlenmesi gerekmektedir.

Başlangıçta bulunan öznitelik noktalarının yeni resimdeki karşılıkları $p_{yeni}=[u_{yeni},v_{yeni}]^T$ ve bu noktaların kamera koordinat sistemine göre 3B koordinatları $P_{yeni}=[X_{yeni}, Y_{yeni}, Z_{yeni}]^T$ olsun.

Bu durumda 2B noktalar;

$$p_{yeni} = K P_{yeni} \quad , \quad K = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \quad (4.41)$$

olur.

K matrisi kameranın dahili parametrelerini ifade etmektedir ve önceden bilinmektedir. 2B öznitelik noktasını ifade eden p_{yeni} noktası, ilk resimde elde edilen özniteliklerin takip edilmesi ile bulunmaktadır. Denklemden bilinmeyen 3B P_{yeni} noktasıdır. Modelin başlatılması aşamasında elde edilen eski 3B noktalar ile yeni 3B noktalar arasındaki ilişki bir dönüşüm matrisi ile aşağıdaki gibi ifade edilebilir.

$$P_{yeni} = T P_{eski} \quad , \quad T = [R|t] \quad (4.42)$$

Denklem 3.42 ile denklem 3.41 birleştirilerek 3B başlangıç noktaları ile 2B yeni noktalar arasındaki ilişki tanımlanabilir. Buna göre;

$$p_{yeni} = K T P_{eski} \quad (4.43)$$

$$\begin{bmatrix} u_{yeni} \\ v_{yeni} \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & o_x & 0 \\ 0 & f_y & o_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{ilk} \\ Y_{ilk} \\ Z_{ilk} \\ 1 \end{bmatrix} \quad (4.44)$$

elde edilir.

Denklem 3.44'te özniteliklerin yeni resimdeki 2B koordinatları, kamera matrisi ve başlangıçtaki 3B koordinatlar bilindiği için tek bilinmeyen dönüşüm matrisidir. Dönüşüm matrisi üç dönme, üç öteleme olmak üzere altı bilinmeyen parametreye sahiptir. Dönüşüm matrisi, bilinen 3B başlangıç noktaları ile takip edilen 2B noktalar kullanılarak belirlenmektedir. Bunun için başlangıçta elde edilen 3B koordinatların 2B izdüşümleri ile takip edilen noktaların yeni resimdeki 2B koordinatları arasındaki farkı minimuma indirecek dönüşüm matrisi parametreleri hesaplanmaktadır. Tez çalışmasında, bu matrisin hesaplanması için OpenCv kütüphanesi içerisinde bulunan, direkt lineer dönüşüm ve Levenberg-Marquardt [61] optimizasyon yöntemlerini kullanan bir fonksiyondan yararlanılmıştır.

4.6 Kafa Pozu Tahmini Uygulamasının Deneysel Sonuçları

4.6.1 Kafa Pozu Veri Seti

Tezde uygulanan kafa pozu tahmini algoritmasının sonuçlarını değerlendirmek için Boston Üniversitesine ait, literatürde yaygın olarak kullanılan bir veri seti kullanılmıştır [42]. Veri seti içerisinde 5 farklı kişiye ait 45 adet video bulunmaktadır. Videolar 30 fps kamera ile kaydedilmiştir ve her bir video 320x240 çözünürlüğe sahip 200 resimden oluşmaktadır. 200 adet resmin her biri için karşılık geldiği poz değeri, kafaya yerleştirilmiş olan manyetik sensör vasıtası ile ölçülmüş ve kaydedilmiştir. Karşılaştırma işlemi için bulunan oryantasyon verileriyle gerçek veriler arasındaki öklid mesafelerinin ortalama değerlerine bakılmaktadır.

$$A_{ölçülen} = [\omega_x, \omega_y, \omega_z]$$

$$A_{gerçek} = [\tilde{\omega}_x, \tilde{\omega}_y, \tilde{\omega}_z]$$

Burada $\omega_x, \omega_y, \omega_z$ sırasıyla x, y ve z eksenleri etrafında ölçülen dönme miktarını, $\tilde{\omega}_x, \tilde{\omega}_y, \tilde{\omega}_z$ ise gerçek dönme miktarlarını ifade etmektedir. Her bir resimdeki hata değerini hesaplamak için aşağıdaki denklem kullanılmaktadır.

$$\vec{e}_i = [A_{gerçek} - A_{ölçülen}]^T [A_{gerçek} - A_{ölçülen}] \quad (4.45)$$

Her bir video için hata değeri;

$$\vec{e}_{video} = \sqrt{\frac{1}{N} \sum_{i=1}^N \vec{e}_i} \quad (4.46)$$

olarak elde edilir.

Burada N, videodaki resim sayısını ifade etmektedir. Son olarak tüm videolar için ortalama hata değeri;

$$\vec{e}_{ort} = \frac{1}{T} \sum_{i=1}^T \vec{e}_{video}(i), \quad \vec{e}_{ort} = [e_x, e_y, e_z]^T \quad (4.47)$$

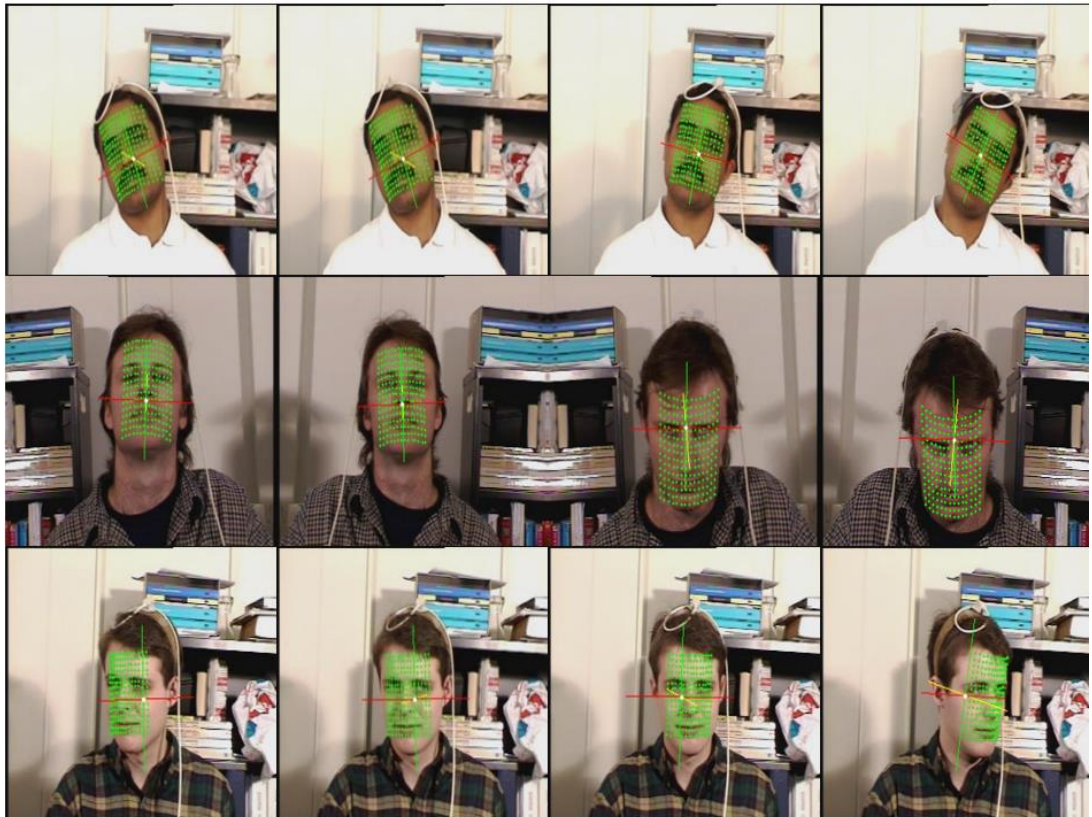
olarak elde edilir.

Denklem 4.47'de T, veri setindeki video sayısını ifade etmektedir. Bulunan ortalama hata değeri, 45 video için x, y ve z eksenleri etrafındaki ortalama dönme hata

değerlerini ifade etmektedir. Veri seti üzerinde elde edilen doğruluk değerleri, her bir eksen için ayrı ayrı hesaplanmaktadır.

4.6.2 Deneysel Bulgular ve Gerçek Zamanlı Sonuçlar

Kafa pozunu tahmini işlemi için ilk olarak veri tabanında bulunan videoların ilk resimlerine, yüz algılama ve göz merkezi belirleme işlemleri uygulanmış ve gözler arası mesafe kullanılarak modelin kameraya olan başlangıç mesafesi belirlenmiştir. İlk resimde, yüz bölgesi içerisindeki takip edilecek öznitelikler belirlenmiş ve 200 resimlik video boyunca bu noktalar takip edilerek kafa pozunu tahmini gerçekleştirilmiştir. Her bir resim için veri setinde bulunan poz değeri ile tez çalışmasında elde edilen poz değerleri karşılaştırılmıştır. Poz verisi x ekseninde dönme (Pitch), y ekseninde dönme (Yaw) ve z ekseninde dönme (Roll) miktarı ile ifade edilmiştir. Şekil 4.8'de x , y ve z eksenleri etrafında dönmeyi gösteren bazı pozlarda kafa pozunu belirleme algoritmasının uygulanmasıyla elde edilen sonuçlar gösterilmiştir.



Şekil 4.8 : Kafa pozunu veri tabanı ile elde edilen bazı sonuç resimleri. Satırlar sırasıyla z ekseninde dönme (Roll), x ekseninde dönme (Pitch) ve y ekseninde dönme (Yaw) durumlarını göstermektedir.

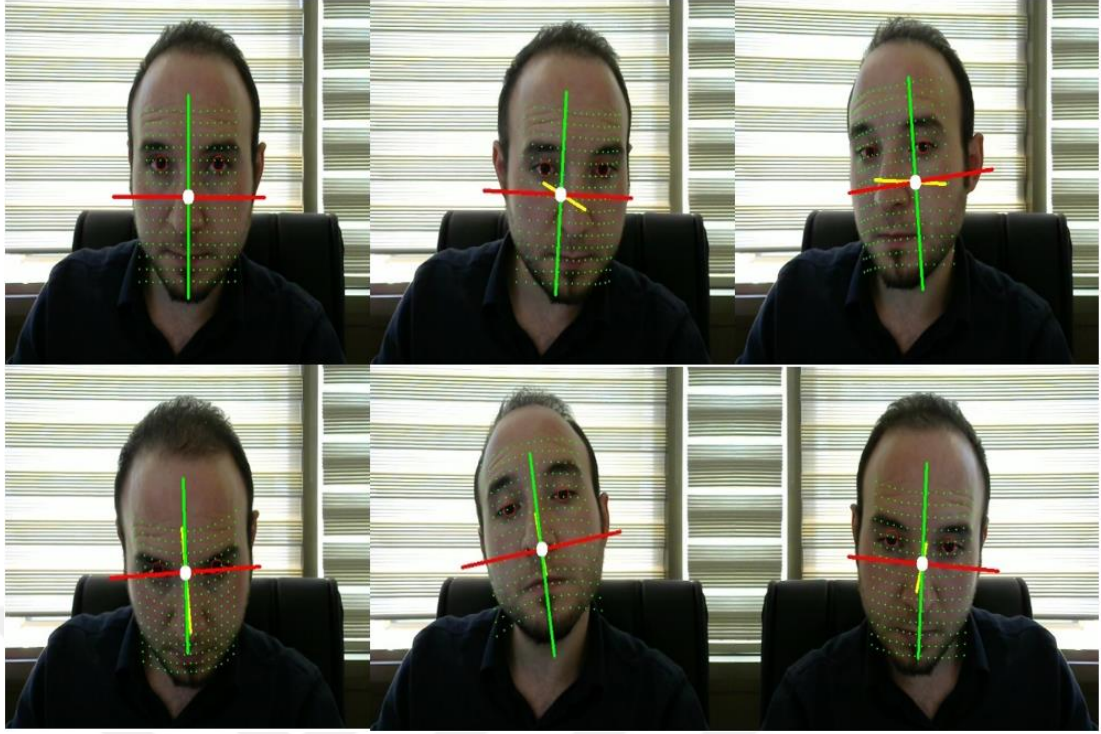
Çizelge 4.1’de tezde sunulan kafa pozunu tahmini uygulamasının veri setine uygulanmasıyla elde edilen doğruluk değerleri ve diğer metotlarla karşılaştırılması ile elde edilen sonuçlar görülmektedir.

Çizelge 4.1 : Bulunan doğruluk değerleri ve diğer metotlarla karşılaştırma sonuçları.

Metot	Roll (Dönme)(ω_z)	Yaw (Sapma)(ω_y)	Pitch (Eğilme) (ω_x)
An [62]	3.22°	5.33°	7.22°
Sung [44]	3.1°	5.4°	5.6°
Valenti [9]	3.00°	6.1°	5.26°
Cascia [42]	9.8°	3.3°	6.1°
Önerilen Metot	3.05° (2)	5.23°(1)	5.84°(3)

Sunulan kafa pozunu belirleme yöntemi literatürdeki diğer yöntemler ile karşılaştırıldığında, yöntemin tatmin edici sonuçlar verdiği görülmektedir. Modelin başlangıç mesafesini hesaplamak için gözler arasındaki mesafenin belirlenmesi doğruluğu arttırmıştır. Yöntem, z eksenini etrafındaki kafa dönmesini 3.05 derecelik bir hatayla hesaplamaktadır ve bu değer karşılaştırılan metotlar arasında ikinci sırada yer almaktadır. Bu dönme değeri için en düşük hata değeri 3,00 derece olarak Valenti tarafından elde edilmiştir. Valenti tez çalışmasında olduğu gibi silindirik bir model kullanmıştır ve göz konumlarını modelin başlatılması için kullanmıştır. Tezde silindir model belirli öznitelik noktaları ile temsil edilirken, Valenti tarafından kullanılan yöntemde tüm yüz bölgesi takip edilmektedir. Sunulan yöntem, y eksenini etrafındaki dönme hatasına bakıldığında ilk sırada yer almaktadır. Son olarak x eksenini etrafındaki dönme hatasında üçüncü sırada yer almaktadır.

Veri seti kullanılarak elde edilen sonuçların yanında, kafa pozunu tahmini yaklaşımının gerçek zamanlı sonuçları da değerlendirilmiştir. Şekil 4.9’da kafa pozunu tahmini uygulamasına ait gerçek zamanlı bazı sonuç resimleri görülmektedir.



Şekil 4.9 : Elde edilen bazı gerçek zamanlı sonuç resimleri.

Sunulan yöntem, sabit bir modele ve özneliklerin takibine dayalı olduğu için çalışma hızı yüksektir. Bu nedenle tez çalışmasında gerçekleştirilen göz merkezi belirleme uygulamasında olduğu gibi işlemcinin dört çekirdeğini kullanmak, kafa pozunu tahmini uygulamasının performansını arttırmamaktadır.

Çizelge 4.2’de göz merkezi belirleme ve kafa pozunu tahmini uygulamalarının birleştirilmesi ile elde edilen sistemin genel performansını değerlendirmek için sistem aşamaları için gerekli işlem süreleri verilmiştir.

Çizelge 4.2 : Tasarlanan sistemin farklı aşamaları için geçen süreler.

Sistem Aşaması				
Yüz Algılama	Göz Merkezi Belirleme	Kafa Pozu Tahmini	Diğer İşlemler	Toplam Geçen Süre
8ms	9.5ms	2.5 ms	4 ms	24ms

Elde edilen gerçek zamanlı sonuçlar, tasarlanan sistemin yaklaşık 42 fps ile çalışabileceğini ortaya koymuştur. Sistemin çalışma hızı gerçek zamanlı uygulamalar için tatmin edici seviyededir.

5. SONUÇ VE ÖNERİLER

Tez çalışmasında, insan bilgisayar etkileşimi uygulamalarında büyük öneme sahip olan, göz merkezi belirleme ve kafa pozunu tahmini uygulamalarının birlikte bulunduğu gerçek zamanlı bir sistem tasarlanmıştır. Göz merkezi belirlenmesi uygulamasında amaçlanan, göz bebeği içerisinde bir noktayı göz merkezi olarak belirleyebilecek doğruluğa sahip bir yaklaşım sunmaktır. Bu amaçla, göz merkezinin kabaca belirlenmesi ve belirlenen merkez noktasının iyileştirilmesi aşamalarından oluşan, iki aşamalı bir yaklaşım kullanılmıştır. Göz merkezinin kabaca belirlenmesi için literatürdeki yaygın bir yöntem kullanılmıştır. İyileştirme aşaması için ise literatürdeki yöntemlerin avantajlarından aynı anda faydalanmayı sağlayacak yeni bir amaç fonksiyonu geliştirilmiştir. Tez çalışmasında geliştirilen iyileştirme aşamasının literatüre sağladığı katkıyı göstermek için elde edilen sonuçlar literatürdeki diğer sonuçlarla karşılaştırılmıştır. Karşılaştırma sonucunda sunulan yaklaşımın, doğruluk ve çalışma hızı bakımından diğer metotların önünde olduğu görülmüştür. Göz merkezini, göz bebeği sınırları içerisinde belirleme konusunda, diğer metotları geride bırakarak ilk sırada yer alan metodumuz, diğer hata değerleri için de tatmin edici sonuçlar vermiştir. Tasarlanan sistemin gerçek zamanlı olarak çalışması amaçlandığı için sistemin çalışma hızı ile ilgili sonuçlarda verilmiştir. Bu amaçla çalışmada kullanılan iteratif göz merkezi belirleme yaklaşımı, seri ve paralel bir şekilde ayrı ayrı uygulanmıştır. Paralel çalışma ile elde edilen performans artışının gösterilmesi amacıyla, tek çekirdek ve dört çekirdek kullanılarak elde edilen işlem süreleri karşılaştırılmıştır. Karşılaştırma sonuçlarına göre dört çekirdek kullanımı sistem performansını yaklaşık dört kat arttırmış ve ortalama 9.5 ms'de göz merkezi hassas bir şekilde belirlenebilmektedir.

Kafa pozunu tahmini için takibe ve sabit bir modele dayalı bir yaklaşım tercih edilmiştir. Kafa pozunu tahmini için elde edilen sonuçlar, literatürdeki yöntemlerle karşılaştırılmış ve tatmin edici sonuçlar elde edilmiştir. Özellikle y ekseninde dönmeyi belirlemede, diğer metotlara göre üstünlük sağlanmıştır. Özetle

noktalarının takibine dayalı bir yaklaşım olduđu için oldukça hızlı bir şekilde çalışmaktadır. Gerçek zamanlı sonuçlara göre, yaklaşık olarak 2.5 ms'de poz algılama işlemi gerçekleştirilmektedir.

Sunulan sistem, saniyede 42 resim işleyebilmektedir. Bu nedenle gerçek zamanlı uygulamalar için çok uygundur. Ancak sistem performansının daha da artırılması için her bir resim için yüz bölgesinin algılanması işlemi yapılmayabilir. Bunu gerçekleştirmek için kafa pozunu aşamasında takip edilen öznitelik noktalarını çevreleyen dikdörtgensel bölge, yüz bölgesi olarak kullanılabilir. Sisteme eklenebilecek bir diğer iyileştirme, göz merkezi belirlemede kullanılan iyileştirme aşamasındadır. İyileştirme aşamasında, iris üzerinde işleme alınacak noktaları belirleyen ve irisin yoğunlukla görünen kısımlarını temsil eden açı aralığı sabit olarak seçilmiştir. Bu açı aralığını kafa pozuna ve göz kapaklarının durumuna göre belirlemek elde edilen doğruluk değerini daha da arttırabilir.

KAYNAKLAR

- [1] LoPresti, E. F., Brienza, D. M., & Angelo, J. (2002). Head-operated computer controls: Effect of control method on performance for subjects with and without disability, *Interacting with Computers*, 14(4), 359–377.
- [2] Evans, D. G., Drew, R., & Blenkhorn, P. (2000). Controlling mouse pointer position using an infrared head-operated joystick, *IEEE Transactions on Rehabilitation Engineering*, 8(1), 107–117.
- [3] Zhang, X., Sugano, Y., Fritz, M., & Bulling, A. (2015). Appearance-Based Gaze Estimation in the Wild, 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (ss. 4511-4520). Boston, ABD, Haziran 8-10.
- [4] Wang, J.-G., & Sung, E. (2001). Gaze Determination via Images of Irises, *Image and Vision Computing*, 19(12), 891- 911.
- [5] Wood, E., & Bulling, A. (2014). Eyetab: Model-based gaze estimation on unmodified tablet computers, *Eye Tracking Research and Applications Symposium (ETRA)*, (ss. 207-210). Florida, ABD, Mart 26-28.
- [6] Chen, J., & Ji, Q. (2008). 3D gaze estimation with a single camera without IR illumination, *19th International Conference on Pattern Recognition*, (ss. 1-4). Florida, ABD, Aralık 8-11
- [7] Xiong, X., Liu, Z., Cai, Q., & Zhang, Z. (2014). Eye Gaze Tracking Using an RGBD Camera: A Comparison with a RGB Solution. *International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, (ss. 1113–1121). Washington, ABD, Eylül 13-17.
- [8] Kang, W., & Ji, Q. (2016). Real time eye gaze tracking with Kinect. *23rd International Conference on Pattern Recognition*, (ss. 2752–2757). Cancun, Meksika, Aralık 4-8.
- [9] Valenti, R., Sebe, N., & Gevers, T. (2012). Combining head pose and eye location information for gaze estimation. *IEEE Transactions on Image Processing*, 21(2), 802–815.
- [10] Bulling, A., Ward, J. A., Gellersen, H., & Tröster, G. (2011). Eye movement analysis for activity recognition using electrooculography. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4), 741–753.
- [11] Dhillon, H. S., Singla, R., Rekhi, N. S., & Jha, R. (2009). EOG and EMG based virtual keyboard: A brain-computer interface, *2nd IEEE International Conference on Computer Science and Information Technology*, (ss. 259–262). Beijing, Çin, Ağustos 8-11.
- [12] Acuna, O. V., Aqueveque, P., & Pino, E. J. (2014). Eye-tracking capabilities of low-cost EOG system, *36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, (ss. 610–613). Chicago, ABD, Ağustos 26-30.
- [13] Kirbiš, M., & Kramberger, I. (2009). Multi channel EOG signal recognition for an embedded eye movement tracking device, *16th International Conference on Systems, Signals and Image Processing*, (ss. 1-4). Chalkida, unanistan, Haziran 18-20.
- [14] Noor, N. M. M., & Bin Kamarudin, M. Q. (2016). Study the different level of

eye movement based on electrooculography (EOG) technique, *2016 IEEE-EMBS Conference on Biomedical Engineering and Sciences*, (ss. 792–796). Kuala Lumpur, Malezya, Aralık 4-7.

[15] **Timm, F., & Barth, E.** (2011). Accurate Eye Centre Localisation By Means of Gradients, *International Conference on Computer Vision Theory and Applications*, (ss. 125–130). Vilamoura, Portekiz, Mart 5-7.

[16] **Valenti, R., & Gevers, T.** (2008). Accurate eye center location and tracking using isophote curvature, *26th IEEE Conference on Computer Vision and Pattern Recognition*, (ss. 1-8). Portland, ABD, Haziran 23-28.

[17] **Rurainsky, J., & Eisert, P.** (2004). Eye center localization using adaptive templates, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, (ss. 67-67). Washington, ABD, Haziran 27- Temmuz 2.

[18] **Poulopoulos, N., & Psarakis, E. Z.** (2018). A new high precision eye center localization technique, *International Conference on Image Processing*, (ss. 2806–2810). Beijing, Çin, Eylül 17-20.

[19] **Garg, S., Tripathi, A., & Cutrell, E.** (2016). Accurate eye center localization using Snakuscul, *2016 IEEE Winter Conference on Applications of Computer Vision*, (ss. 1-8). New York, ABD, Mart 7-9.

[20] **Lee, K.** (2002). Combining Kalman Tracking Filtering Under and Mean Shift for Real Active IR Illumination Eye, *Object recognition supported by user interaction for service robots*, 4, 318-321.

[21] **Shin, Y. G., Choi, K. A., Kim, S. T., Yoo, C. H., & Ko, S. J.** (2015). A novel 2-D mapping-based remote eye gaze tracking method using two IR light sources, *2015 IEEE International Conference on Consumer Electronics*, (ss. 190–191). Taiwan, Ocak.

[22] **Boumbarov, O., Panev, S., Sokolov, S., & Kanchev, V.** (2009). IR based pupil tracking using optimized particle filter, *5th IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, (ss. 404–408). Rende, İtalya, Eylül 21-23.

[23] **Young, D., Tunley, H., & Samuels, R.** (1995). Specialised Hough Transform and Active Contour Methods for Real-Time Eye Tracking.

[24] **Toennies, K., Behrens, F., & Aurnhammer, M.** (2002). Feasibility of hough-transform-based Iris localisation for real-time-application, *Object recognition supported by user interaction for service robots*, 16(2), 1053–1056.

[25] **Daugman, J.** (2009), How Iris Recognition Works, *The Essential Guide to Image Processing*, (ss. 715–739). Boston: Academic Press.

[26] **Fitzgibbon, A. W., Fisher, R. B., Hill, F., & Eh, E.** (1999). Direct Least Squares Fitting of Ellipses, *IEEE Transactions on pattern analysis and machine intelligence*, 21(5), 476–480.

[27] **Yahya, A. E., & Nordin, M. J.** (2008). A new technique for iris localization in iris recognition systems, *Information Technology Journal*, 7(6), 924–929.

[28] **Fischler, M. A., & Bolles, R. C.** (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography, 24(6), 381–395.

[29] **Świrski, L., Bulling, A., & Dodgson, N.** (2012). Robust real-time pupil tracking in highly off-axis images, *Eye Tracking Research and Applications Symposium*, (ss. 173-1760). Santa Barbara, ABD, Mart 28-30.

[30] **Murphy-Chutorian, E., & Trivedi, M. M.** (2009). Head pose estimation in computer vision: A survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4), 607–626.

- [31] **Beymer, D. J.** (1994). Face recognition under varying pose, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (ss. 756–761), Washington, ABD, Temmuz 21-23.
- [32] **Niyogi, S., & Freeman, W. T.** (1996). Example-based head tracking, *International Conference on Automatic Face and Gesture Recognition*, (ss. 374–378). Killington, ABD, Ekim 13-16.
- [33] **Sherrah, J., Gong, S., & Ong, E. J.** (2001). Face distributions in similarity space under varying head pose, *Image and Vision Computing*, 19(12), 807–819.
- [34] **Ma, Y., Konishi, Y., Kinoshita, K., Lao, S., & Kawade, M.** (2006). Sparse Bayesian regression for head pose estimation, *18th International Conference on Pattern Recognition*, (ss. 507–510). Hong Kong, Çin, Ağustos 20-24.
- [35] **Voit, M., Nickel, K., & Stiefelhagen, R.** (2006). A bayesian approach for multi-view head pose estimation, *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, (ss. 31–34). Heidelberg, Almanya, Eylül 3-6.
- [36] **Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J.** (1995). Active Shape Models-Their Training and Application, *Computer Vision and Image Understanding*, 61(1), 38–59.
- [37] **Wold, S., Esbensen, K., & Geladi, P.** (1987). Principal component analysis, *Chemometrics and Intelligent Laboratory Systems*, 2(1), 37–52.
- [38] **Cootes, T. F., Edwards, G. J., & Taylor, C. J.** (2001). Active appearance models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23, 681-685.
- [39] **Zhao, G., Chen, L., Song, J., & Chen, G.** (2007). Large Head Movement Tracking Using Sift-based Registration, *15th ACM International Conference on Multimedia*, (ss. 807–810). Augsburg, Almanya, Eylül 23-28.
- [40] **Vacchetti, L., Lepetit, V., & Fua, P.** (2004). Stable real-time 3D tracking using online and offline information, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10), 1385–1391.
- [41] **Xiao, J., Kanade, T., & Cohn, J. F.** (2002). Robust full-motion recovery of head by dynamic templates and re-registration techniques, *5th IEEE International Conference on Automatic Face Gesture Recognition*, (ss. 163–169). Washington, ABD, Mayıs 21.
- [42] **La Cascia, M., Sclaroff, S., & Athitsos, V.** (2002). Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3D models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(4), 322–336.
- [43] **Medioni, G., Liao, W. K., & Fidaleo, D.** (2010). Robust, real-time 3D face tracking from a monocular view, *Eurasip Journal on Image and Video Processing*, 1-15.
- [44] **Sung, J., Kanade, T., & Kim, D.** (2008). Pose robust face tracking by combining active appearance models and cylinder head models, *International Journal of Computer Vision*, 80(2), 260–274.
- [45] **Viola, P., & Jones, M.** (2001). Rapid Object Detection using a Boosted Cascade of Simple Features, *2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (ss. I-511). Hawaii, ABD, Aralık 8-14.
- [46] **Papageorgiou, C. P., Oren, M., & Poggio, T.** (1998). General framework for object detection, *6th IEEE International Conference on Computer Vision*, (ss. 555–562). Bombay, Hindistan, Ocak 7.
- [47] **Cristinacce, D., Cootes, T., & Scott, I.** (2004). A Multi-Stage Approach to

- Facial Feature Detection, *15th British Machine Vision Conference*, (ss. 231-240). Londra, İngiltere, Eylül 7-9.
- [48] **Bai, L., Shen, L., & Wang, Y.** (2006). A Novel Eye Location Algorithm based on Radial Symmetry Transform, *18th International Conference on Pattern Recognition*, (ss. 511–514). Hong-Kong, Çin, Ağustos 20-24.
- [49] **Campadelli, P., Lanzarotti, R., & Lipori, G.** (2006). Precise eye localization through a general-to-specific model definition, *2006 British Machine Vision Conference*, (ss. 187–196). Edinburgh, İngiltere, Eylül 4-7.
- [50] **Türkan, M., Pardàs, M. M., & Çetin, A. E.** (2007). Human eye localization using edge projections, *2nd International Conference on Computer Vision Theory and Applications*, (ss. 410–415). Barselona, İspanya, Mart 8-11.
- [51] **Lucas, B. D., & Kanade, T.** (1981). An iterative image registration technique with an application to stereo vision, *7th International Joint Conference on Artificial Intelligence*, (ss. 674–679). Kanada, ABD, Ağustos 24-28.
- [52] **Harris, C., & Stephens, M.** (2013). A Combined Corner and Edge Detector, *4th Alvey Vision Conference*, (ss. 147-151). Manchester, İngiltere, Ağustos 31- Eylül 2.
- [53] **Lowe, D. G.** (2004). Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision*, 60(2), 91–110.
- [54] **Murray, J. A. H.** (1887). BRISK: Binary Robust invariant scalable keypoints, *13th International Conference on Computer Vision*, (ss. 2548-2555). Barselona, İspanya, Kasım 6-13.
- [55] **Zinke, R., Hollingsworth, J., Dolan, J. F., Dissen, R. Van, Angeles, L., Ayoub, F., ... Farag, A. A.** (2009). CS231A Course Notes 1 : Camera Models, *ISPRS Journal of Photogrammetry and Remote Sensing*, 3(6), 1541–1576.
- [56] **Model, M.** (2013). Computer Vision Lecture 1: The Pinhole Camera Model 1 Mathematical Model, (2), 2–5.
- [57] **Dodgson, N. A.** (2004). Variation and extrema of human interpupillary distance, *Stereoscopic Displays and Virtual Reality Systems XI*, 5291, 36–46.
- [58] **Gordon, C. C., Blackwell, C. L., Bradtmiller, B., Parham, J. L., Barrientos, P., Paquette, S. P., ... Kristensen, S.** (2014). 2012 Anthropometric Survey of U.S. Army Personnel: Methods and Summary Statistics.
- [59] **Barros, J. M. D., Mirbach, B., Garcia, F., Varanasi, K., & Stricker, D.** (2018). Fusion of Keypoint Tracking and Facial Landmark Detection for Real-Time Head Pose Estimation, *2018 IEEE Winter Conference on Applications of Computer Vision*, (ss. 2028–2037). Nevada, ABD, Mart 12-14.
- [60] **Ranganathan, A.** (2004). The Levenberg-Marquardt Algorithm 3 LM as a blend of Gradient descent and Gauss-Newton itera, Retrieved from <http://twiki.cis.rit.edu/twiki/pub/Main/AdvancedDipTeamB/the-levenberg-marquardt-algorithm.pdf>
- [61] **An, K. H., & Chung, M. J.** (2008). 3D head tracking and pose-robust 2D texture map-based face recognition using a simple ellipsoid model, *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, (ss. 307–312). Nice, Fransa, Eylül 22-26.

ÖZGEÇMİŞ



Ad-Soyad : Fatih Dursun
Doğum Tarihi ve Yeri : 26.10.1993 Kadıköy/İstanbul
E-posta : fatih.dursun@btu.edu.tr

ÖĞRENİM DURUMU:

- **Lisans** : 2017, Erciyes Üniversitesi, Mühendislik Fakültesi, Mekatronik Mühendisliği
- **Lisans** : 2017, Erciyes Üniversitesi, Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği
- **Yüksek Lisans** : 2020, Bursa Teknik Üniversitesi, Mekatronik Mühendisliği Anabilim Dalı

MESLEKİ DENEYİM VE ÖDÜLLER:

- Araştırma Görevlisi (Bursa Teknik Üniversitesi 2018-2020)