

**T.C.
SÜLEYMAN DEMİREL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

MOBİL CİHAZLAR ÜZERİNDEN GÖZ HASTALIKLARI TAKİBİ

Ahmet Seçkin BİLGİ

**Danışman
Doç. Dr. Habil KALKAN**

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
ISPARTA - 2016**



© 2016 [Ahmet Seçkin BİLGİ]

TEZ ONAYI

Ahmet Seçkin BİLGİ tarafından hazırlanan "**Mobil Cihazlar Üzerinden Göz Hastalıkları Takibi**" adlı tez çalışması aşağıdaki jüri üyeleri önünde Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği Anabilim Dalı**'nda **YÜKSEK LİSANS TEZİ** olarak başarı ile savunulmuştur.

Danışman

Doç. Dr. Habil KALKAN
Süleyman Demirel Üniversitesi

Jüri Üyesi

Doç. Dr. Alime GÜNEŞ
Süleyman Demirel Üniversitesi

Jüri Üyesi

Yrd. Doç. Dr. Mahmut SİNECEN
Adnan Menderes Üniversitesi

Enstitü Müdürü

Doç. Dr. Yasin TUNCER

.....

TAAHHÜTNAME

Bu tezin akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

Ahmet Seçkin BİLGİ

Ahmet Seçkin

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER.....	i
ÖZET	ii
ABSTRACT	iii
TEŞEKKÜR.....	iv
ŞEKİLLER DİZİNİ	v
ÇİZELGELER DİZİNİ	vii
SİMGELER VE KISALTMALAR DİZİNİ.....	viii
1. GİRİŞ.....	1
1.1. Karar Destek Sistemi Nedir?	1
1.2. Klinik Karar Destek Sistemi Nedir?.....	1
1.3. Göz Hastalıkları Takibi	3
1.4. Önerilen Sistem	4
1.4.1. Kapsadığı platformlar.....	6
1.5. Tezin Bölümleri	6
2. KAYNAK ÖZETLERİ	8
3. MATERYAL VE YÖNTEM	11
3.1. Göz Hastalıkları.....	11
3.2. Materyal	12
3.2.1. UBIRIS.v1 veri seti	12
3.2.2. SDUeye veri seti	13
3.3. Göz Akı (Sklera) Bölütlemesi	15
3.3.1. Çevrimdışı yaklaşım.....	16
3.3.2. Çevrimiçi yaklaşım	26
3.4. Göz Hastalıkları Takip Sistemi	36
3.4.1. Model.....	36
3.4.2. Yazılım.....	40
3.4.3. Kabiliyetler.....	43
3.5. Başarı Hesaplama	48
3.5.1. Göz akı bölütlemesi başarı hesabı.....	49
3.5.2. Göz kızarıklığı başarı hesabı.....	49
4. ARAŞTIRMA BULGULARI VE TARTIŞMA	51
4.1. Çevrim Dışı Göz Akı Bölütleme	51
4.2. Göz Kızarıklığı Tespiti.....	56
4.2.1. Brien Holden ve algoritma karşılaştırması	57
4.2.2. SDUeye veri seti	57
5. SONUÇLAR VE ÖNERİLER	60
KAYNAKLAR.....	61
ÖZGEÇMİŞ.....	64

ÖZET

Yüksek Lisans Tezi

MOBİL CİHAZLAR ÜZERİNDEN GÖZ HASTALIKLARI TAKİBİ

Ahmet Seçkin BİLGİ

Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Habil KALKAN

Bu tez çalışmasında göz hastalıklarını uzaktan takip edebilecek mobil program geliştirilmiştir. Mobil program göz doktorlarının kullandığı görme anormallikleri, görünüm anormallikleri, ağrı ve rahatsızlık belirtilerini toplayarak doktora iletacaktır. Bu belirtilerden görünüm anormallikleri altında bulunan, göz kızarıklık seviyesi tespit edilerek doktorun nesnel bir yaklaşım sergilemesi sağlanmıştır. Ayrıca program vasıtası ile hastaya verilecek olan uyarılar ile ilaç zamanına uyumun artırılarak tedavinin düzgün yapılması sağlanmıştır. Sağladığı bu yöntemler ile sadece bir takip sistemi değil aynı zamanda klinik karar destek sistemi özelliği taşımaktadır.

Mobil cihazlar, gerekli olan verileri toplayarak web servise göndermektedir. Doktor ise istediği zaman mobil cihazından giriş yaparak kendi hastalarının belirtilerini takip edebilmektedir. Bu yolla hastalığın uzaktan, güvenilir sosyoekonomik sorunlara neden olmadan takip edilmesi sağlanmıştır.

Geliştirilen program daha fazla kullanıcıya ulaşabilmek amacıyla iki farklı işletim sistemi için yazılmıştır. Bunlar Android ve IOS işletim sistemleridir. Android işletim sistemini için olan program java dili ile, IOS için olan program ise swift diliyle programlanmıştır.

Anahtar Kelimeler: hastalık takibi, göz akı bölütlemesi, göz kızarıklığı tespiti, göz hastalıkları.

2016, 64 sayfa

ABSTRACT

M.Sc. Thesis

EYE DISEASE SURVEILLANCE SYSTEM ON MOBILE DEVICES

Ahmet Seçkin BİLGİ

**Süleyman Demirel University
Graduate School of Applied and Natural Sciences
Department of Computer Engineering**

Supervisor: Assoc. Prof. Dr. Habil KALKAN

In this thesis, a mobile program has been developed for surveillance eye diseases remotely. This mobile program collects vision abnormalities, appearance abnormalities, eye pain and the other disorder symptoms and, sends it to the doctor. These symptoms, eye redness level that also part of appearance abnormalities are evaluated by the developed program. Using the program, doctors can perform better on medical decisions. Also the treatment process of the patient is supported by the medicine time reminder function of the program. This program is not only a surveillance system, but also a clinical decision support system.

In the developed system, the required data is gathered by the user's smart mobile devices and the data is send to web service for computation. Doctors can observe their patient's symptoms by logging in with their phones remotely and can track the patient's treatment process at any time.

In order to reach more users, this program written in two different mobile operating systems which are Android and IOS. The program for Android operating system has been coded with JAVA and codes for IOS has been coded with Swift programming language.

Keywords: disease surveillance, sclera segmentation, eye redness level detection, eye disease.

2016, 64 pages

TEŞEKKÜR

Sadece bu tezde değil, tanıştığımız günden bu yana her daim beni yetiştirmeye çalışan, ufkumu açan, güvenen, desteğini hiç eksik etmeyen ve üniversitede güzel işler yapılabileceğini gösteren saygıdeğer hocam Doç. Dr. Habil KALKAN'a çok teşekkür ederim. Her daim soru sorabildiğimiz, verdiği cevaplar ile ben dahil pek çok öğrencinin ufkunu açan, yeniye ve iyiye sevk eden Ali GÜNEŞ hocama teşekkür ederim. Bilmediğim bir konu olan göz ile alakalı verdiği desteklerden dolayı Doç. Dr. Alime GÜNEŞ hocama teşekkür ederim. Çalışma arkadaşlarım Yusuf Erkan hocam başta olmak üzere, Efkan, Gizem, Hüseyin Orhan ve Begench'e, tezimde maddi manevi desteği olan arkadaşlarım Mehmet, Enes Altinkaya'ya teşekkür ederim.

Hayatımın her aşamasında yanımda olup desteklerini esirgemeyen benim kocaman ailemin karşısına hep hayırların çıkması dileği ile, sevgiler...

Ahmet Seçkin BİLGİ
ISPARTA, 2016

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 1.1. Kızarıklık derecelendirme ölçütü Brien Holden.....	5
Şekil 1.2. Mobil işletim sistemleri pazar payı.	6
Şekil 3.1. Gözün yapısı.....	11
Şekil 3.2. UBIRIS.v1 veri setinden örnek görüntüler, açık göz (a), kısmi kapalı göz (b), kapalı göz (c).....	13
Şekil 3.3. Program üzerinden verilen 6 adet (a, b, c, d, e, f) eğitim görüntüsü.	14
Şekil 3.4. Çevrimdışı sistem adımları.	17
Şekil 3.5. LOG filtre.	17
Şekil 3.6. Birinci satır; göz bebeği içindeki parlama (a, b), ikinci satır sırasıyla görüntülerin filtrelenmiş durumları (c, d).....	18
Şekil 3.7. 30 derece açıyla takip edilen çizgiler.....	19
Şekil 3.8. Çizilen yay.....	20
Şekil 3.9. Görüntü üzerine çizilen yaylar (a), sol yay üzerindeki 1. türev değerleri (b), sağ yay üzerindeki 1. türev değerleri (c). Şekil 2.9 (c)'de görüldüğü üzere sağ yayın 20-55. ile gözeğin 240-260 gözeğin arasında kirpik üzerine yerleşmiş olması türev değerleri yüksek çıkmıştır.....	21
Şekil 3.10. Çizilen yaylar üzerindeki potansiyel göz akı bölgeleri (2 siyah çizgi arası) (a), -x eksenini için 1. türev değerleri (b), +x eksenini için çizilen yay üzerindeki 1. türev değerleri (c). Şekil 2.10 (c)'de 75. gözeğin üstünde ve 160. gözeğin altında olmak üzere 2 bölge gözlenmiştir.	22
Şekil 3.11. Çizilen yaylar ve düzey kümeleri sıfır düzey çevriti (cyan dikdörtgenler).....	24
Şekil 3.12. KTDK Sıfır Düzey Çevriti (a) ve bölütleme sonucu (b).....	26
Şekil 3.13. Çevrimiçi sistem adımları.	27
Şekil 3.14. Sırasıyla en yakın komşu (a), blineer (b), bikübik (c) interpolasyonlar ile 72dpi düşürülüp daha sonra orjinal hale getirilen görüntüler.	28
Şekil 3.15. Roberts(a), Prewit (b) ve Sobel (c) filtre örnekleri.....	31
Şekil 3.16. Canny algoritma Adımları.	32
Şekil 3.17. Kenar tespiti işleminin uygulanacağı görüntü (a) ve sonuçları (b)	33
Şekil 3.18. Hough dönüşümü ρ , θ değeri.....	34
Şekil 3.19. Takip programının işleyişi.	38
Şekil 3.20. Kızarıklık tespiti algoritması.....	39
Şekil 3.21. Görme keskinliği testi.	40
Şekil 3.22. Göreceli bilgilerin alınması.....	40
Şekil 3.23. Xcode TGO.....	41
Şekil 3.24. Örnek StoryBoard.....	42
Şekil 3.25. Android Studio geliştirme ortamı.....	43
Şekil 3.26. Kızarıklık tespiti algoritma adımları.	44
Şekil 3.27. KSAHE algoritması çalışma mantığı.	45
Şekil 3.28. Görme keskinlik testi için kullanılan eşel örneği.....	47
Şekil 3.29. Keskinlik ve anma hesabında kullanılacak kümelerin gösterimi	

(a, b).....	49
Şekil 4.1. Sırasıyla birinci örnek ve sistem tarafından çıkartılmış maske (a, b), on dördüncü örnek ve sistem tarafından çıkartılan maske (c, d).	53
Şekil 4.2. Sırasıyla birinci örnek ve el ile çıkartılmış maske (a, b), sırasıyla on dördüncü örnek ve el ile çıkartılan maske (c, d).....	54
Şekil 4.3. Başarı hesabından çıkartılan görüntüler (a, b).....	55
Şekil 4.4. Sistem tarafından hazırlanan maskelerin doğruluk maskeleri ile karşılaştırılması. Beyaz renk sistem tarafından doğru tahmin edilen bölgeyi, kırmızı renk göz akı ama sistem tarafından tespit edilemeyen bölgeyi, yeşil renk ise göz akı olmayan ama sistemin göz akı olarak varsaydığı bölgeyi göstermektedir (a, b, c, d). Sırasıyla, görüntüler için kesinlik değerleri %83.36, %91.09, %90.72, %77.08; anma değerleri ise %89.99, %82,18, %92.28, %29.96 şeklindedir.....	55
Şekil 4.5. SDUeye veri setinden görüntü örnekleri (a, b, c, d). Görüntünün iris içermemesi sebebiyle çember tespiti gerçekleştirilememiştir (e). Hough çember tespit algoritması çemberi kaş üzerine yerleştirmiştir (f). Bu sebeple sistem göz akı tespit edememiştir.....	58
Şekil 4.6. İlgili bölge çıkarımı yapılmış görüntü (a, c) ve sırasıyla bu görüntülerin KSAHE algoritmasından ve kenar tespit algoritmasından geçirilmiş hali (b, d).....	59

ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 3.1. Belirti tipleri ve bu belirtilerin görülebileceği bazı hastalıklar .	12
Çizelge 3.2. Biometrik tanımlama sistemleri karşılaştırması.	16
Çizelge 3.3. Kullanıcı kayıt işlemi sırasında kullanılan bilgiler.	37
Çizelge 3.4. Objective-C ve Swift söz dizimi örnekleri.....	41
Çizelge 4.1. Brien Holden ölçütü ve algoritma çıktısı karşılaştırması.....	57



SİMGELER VE KISALTMALAR DİZİNİ

DD	Dirac Delta
DK	Düzey Kümeleri
KDS	Karar Destek Sistemi
KKDS	Klinik Karar Destek Sistemi
KTDK	Kenar Tabanlı Düzey Kümeleri
KYM	Kırmızı Yeşil Mavi
LOG	Laplacian of Gaussian
SDÇ	Sıfır Düzey Çevriti
TGO	Tümleşik Geliştirme Ortamı



1. GİRİŞ

1.1. Karar Destek Sistemi Nedir?

İnsan beyninin bilgi depolama ve işleme gücü sınırlıdır. Kimi durumlarda ise öznel düşüncelerin karışması kararı etkilemektedir. Bu gibi nedenler özellikle dikkatin az olduğu durumlarda kişilerin hatalı kararlar vermesine neden olur. Gelişen teknoloji ile beraber insan tabanlı bu hataları ortadan kaldırmak veya en aza indirmek mümkündür. Bu amaçla teknolojiden faydalanılarak bir program yazılması fikri, Little (1970) tarafından ortaya atılmıştır. Buna göre, yöneticilere ve kullanıcılara verecekleri kararlarda yardımcı olmak amacıyla oluşturulan programlara Karar Destek Sistemi (KDS) denir. KDS'lerin amacı, yapay zeka programları gibi karar verenleri ortadan kaldırmak değildir. Kullanıcıya gerekli her türlü veriyi sunarak daha başarılı karar vermesini sağlamaktır. Bu özelliği ile yapay zeka programlarından ayrılır.

1.2. Klinik Karar Destek Sistemi Nedir?

Sağlık alanında kullanılan KDS'lere Klinik Karar Destek Sistemi (KKDS) adı verilmektedir. KKDS'ler hastaya özgü klinik verilerin analizini yapar. Elde edilen klinik verilerin hasta bakımına ve tedavilere uygulanmasına yardım eder. Ayrıca sağlık personeline, vereceği kararlarda destek sağlar. Bazıları ise tıbbi hataları önleyerek hasta güvenliğini arttırmayı amaçlar.

Doktora öneriler veren sistemlerin geliştirilmesi için çalışmalar yapılmıştır. Doktorlar tarafından yapılan klinik hataların fazlalığını, bu hataları önlemek için yapılan çalışmaların önemini vurgulamak amacıyla Kawamoto vd. (2005) makale yayımlamışlardır. Makalede, Amerika Birleşik Devletleri'nde medikal hatalar sonucu ölen insan sayısının tahmini yıllık 98000 olduğu belirtilmiştir. Londra'da iki hastanede yapılan retrospektif çalışmada; hastalara uygulanan tedavilerin %11'inde yan etki olduğu belirtilmiştir. Bu tedavilerin %48'inin önlenebilir olduğu ve %8'inin ölümlle sonuçlandığı da belirtilmiştir. Bu gibi hataları önlemek için artık hastanelerin hastaya özgü değerlendirmeler yapan

ve doktora öneriler veren KKDS'ye geçtiği belirtilmiştir. Bu sistemlerin her zaman doktor performansını arttırmadığı fakat, bilgisayar tabanlı sistemlerin 66% oranında performans artışı sağladığının gözlemlendiği belirtilmiştir. Garg vd. (2005) yapmış oldukları çalışmada 100 tane KKDS'nin doktor ve hasta üzerindeki etkisini araştırmışlardır. Araştırmalarında KKDS'leri 4 farklı gruba ayırmışlardır. İlk grup tanı tahlil sistemlerini içermektedir. Bu grup altında toplam 10 çalışma incelenmiştir. İncelenen çalışmalar arasından 4 tanesi sistemin başarılı bir şekilde amacına ulaştığını raporlarken diğerleri herhangi bir gelişme olduğunu belirtmemiştir. İkinci grup önlem için uyarıcı çalışmaları kapsamaktadır. Bu grup altında toplam 21 çalışma incelenmiştir. Çalışmaların 16 tanesinin başarılı sonuç verdiği gözlemlenmiştir. Bu çalışmalarda tarama, sağlık rehberliği, aşılama, ilaç kullanımı ve risk tanımlaması kullanılarak başarı hesabı yapılmıştır. Üçüncü grup hastalık yönetimi sistemlerini içermektedir. Bu alanda toplam 40 çalışma incelenmiştir. Kırk çalışmanın 37 tanesi performans değerlendirmesi yapmıştır ve bunların yirmi üç tanesi doktor performansının arttığını raporlamıştır. Dördüncü grup ise ilaç dozu ve ilaç yazma sistemidir. Bu grup dahilinde 29 çalışma incelenmiştir. Çalışmaların 24 tanesi doktor performansını göz önüne almıştır. Performans değerlendirmesi yapan çalışmaların 15 tanesi performansta artış olduğunu bildirmiştir. İncelenen makalelerde de belirtildiği gibi KKDS her zaman performans artışı sağlamamaktadır. Khalifa (2014) yapmış olduğu derleme çalışmada iyi ve kullanılabilir bir KKDS'nin taşınması gereken özelliklerden bahsetmiştir. Özellikleri 10 başlık altında toplamıştır.

- Doğru içeriği kapsama; yazılan programın disiplinler arası bir çalışma gerektirdiğinden bahsetmiş, klinik kurallar ve doktorların isteklerine göre şekillenmesi gerekliliğini dile getirmiştir.
- Doğru ve güvenilir bilginin sağlanması; hastadan elde edilen verilerin iyi değerlendirilmesi, doktorlara verilecek alarmların bilgi kirliliği yaratmaması gerektiği bildirilmiştir. Doğru alarmların verilmemesi ve doktora sürekli alarm verilmesinin doktoru rahatsız ettiği ve destek sisteminin güvenilirliğini düşürerek kullanılmamasına neden olduğu da belirtilmiştir.

- Sade mesaj ulařtırılması; mesajların ham veri içermemesi ve elde edilen bilgilerin yorumlanarak doktora iletilmesinin KKDS'nin kullanımını artıracığına dikkat çekilmiştir.
- Kullanıcılara bilimsel referanslar sağlamak; eđer doktora yapılacak öneri bilimsel dayanakları ve mantığı ile beraber gönderilirse, KKDS'nin doktorun güvenini kazanacağı belirtilmiştir.
- Zaman kazandırma; KKDS'nin doktorların zamanını etkin bir şekilde kullanmalarına yardımcı olması gerektiğı, sistemin kullanımının kolay, anlaşılır olması ve doktorlar için gerekli olan yerlere kolay erişim sağlaması gerektiğı belirtilmiştir.
- Klinik iş akışına entegrasyon; doktorların iş akışlarını bölerek farklı bir alanda çalışma yapmak istemedikleri ve bu şekilde olan sistemleri kullanmamak için direnç gösterdikleri belirtilmiştir.
- Sistem hızı; KKDS'nin hızlı olmasının kullanıcı memnuniyetini arttırarak sistemin kullanışlı olmasını sağladığı saptanmıştır.
- Aktif ve pasif alarm mekanizması; durumun ciddiyetine göre alarmların kritik, orta, düşük olmak üzere 3 seviyeye ayrılması gerektiğı belirtilmiştir. Kritik alarmların iş akışını bozabilecek şekilde aktif olabileceğı, fakat orta ve düşük seviye alarmların pasif olması gerektiğine dikkat çekilmiştir.
- Hastane bilgi sistemine entegrasyon; KKDS'nin Hastane Bilgi Sistemine entegre olarak hem oradaki ilaç duyarlılığı, tahlil sonuçları gibi dijital verileri kullanması gerektiğı hem de doktorun tüm işlemleri tek bir sistem üzerinden yapabilmesi gerektiğı saptanmıştır.
- KKDS bilgilerinin yönetimi; hasta ile alakalı bilgilerin güncel tutulması gerektiğı vurgulanmıştır.

Oluřturulan KKDS'nin yukarıda bahsedilen 10 özelliğı taşımasının, doktor memnuniyetini arttırdığı belirtilmiştir.

1.3. Göz Hastalıkları Takibi

Hastalık, insan vücudunda meydana gelen bozukluklara verilen isimdir. Bu bozukluklar kimi zaman doğrudan insan gözlemi veya tıbbi aletler ile fark

edilebilen işaretler oluşturur. Bu işaretlere belirti adı verilir. Doktorlar hastalığın tanısını, belirtileri kontrol ederek koyar. Hastalığın ilerleyişi de yine belirtiler yardımı ile takip edilir. Bazı hastalıkların tedavi süreci uzun sürmektedir. Bu durumda bulunan hasta gruplarına tedavi verildikten sonra kontrol işlemi için ileri bir tarihe randevu verilmektedir. Sonraki zamana verilen randevular tedavi ile ilgili sorunlara neden olabilmektedir. Sorunlardan ötürü uzaktan takip sistemi geliştirilmesi ihtiyacı ortaya çıkmıştır. Önerdiğimiz program hasta takibi işleminin uzaktan, güvenli bir şekilde yapılmasını sağlamaktadır.

1.4. Önerilen Sistem

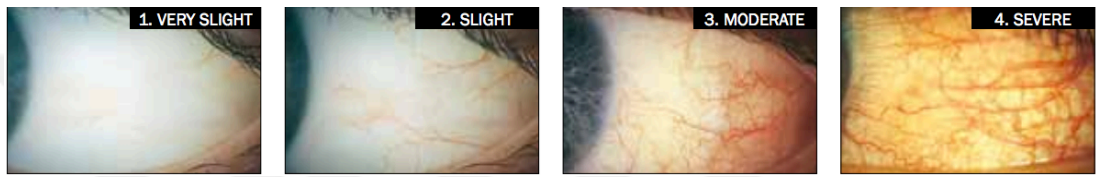
Günümüzde hemen hemen herkesin sahip olduğu akıllı telefonlar gelişmiş kameralara, yüksek işleme kabiliyetine ve internet bağlantısına sahiptir. Bu özelliklerinden dolayı mobil cihazlar artık sadece iletişim amaçlı kullanılmamaktadır. Dizdaroğlu vd. (2014) Android işletim sisteminde belirli görüntüler için tıbbi görüntü bölütleyici geliştirmiştir. Günay (2016) ise yaptığı tez çalışmasında mobil cihazları istemci olarak kullanarak imge bölütleyici geliştirmiştir. Bu iki yaygın mobil cihazların nerelerde kullanılabileceğine dair örnek teşkil etmektedir.

Sunulan tez çalışmasında mobil cihazlar göz hastalıklarının uzaktan takibi için kullanılmıştır. Hastadan belirtileri alarak doktora göndermektedir. Böylece doktorun hastayı uzaktan, sık ve güvenilir bir şekilde takip etmesi sağlanmıştır. Ayrıca kontrol işleminin ortaya çıkardığı sosyoekonomik sorunlar ve sağlık sorunları ortadan kaldırılmış olacaktır.

Belirtiler 3 başlık altında toplanacaktır. Bunlar kızarıklık seviyesi, görme keskinliği ve göreceli bilgilerdir.

Kızarıklık göz içerisindeki damarların genişleyerek görünür hale gelmesi sonucu oluşur. Kızarıklık pek çok hastalığın belirtisi olabilir (Bölüm 2.1) ve bu hastalıkların takibinde önemli bir belirti olarak kullanılır. Fakat kızarıklık

seviyesinin belirlenmesi doktorun inisiyatifine bırakılmış bir işlemdir. Kızarıklık algılaması doktordan doktora değişebileceği gibi aynı doktorun farklı zamanlarına göre de değişebilmektedir. Bu değişimi ortadan kaldırmak amacıyla bir takım ölçütler ortaya atılmıştır (Şekil 1.1) (BrienHoldenVision, 2011). Ölçütler, görsel olarak göz kızarıklığını farklı derecelere ayırıp doktorlara rehberlik etmektedir. Tez çalışmasında hastadan alınan göz görüntülerini işleyerek kızarıklık seviyesi tespitini yapmayı amaçlıyoruz. Böylece doktorların kararlarının güvenilir bir şekilde desteklenmesi amaçlanmıştır.



Şekil 1.1. Kızarıklık derecelendirme ölçütü Brien Holden (2011)

Görme keskinliği, geleneksel yöntemlerde hastaya kontrollü çevre altında gösterilen harf veya şekiller ile yapılmaktadır. Mobil cihazlar içinde benzer bir yaklaşım sergilenebileceği düşünülmüştür. Hastaya giderek küçülen harfler gösterilerek görebildiği en küçük harf tespit edilmeye çalışılmıştır.

Göreceli bilgiler hastaya sorular yöneltilerek alınan cevaplar sonucu elde edilmiştir. Alınan bilgiler doktora gösterilmek üzere sonucu üzerinde depolanmıştır.

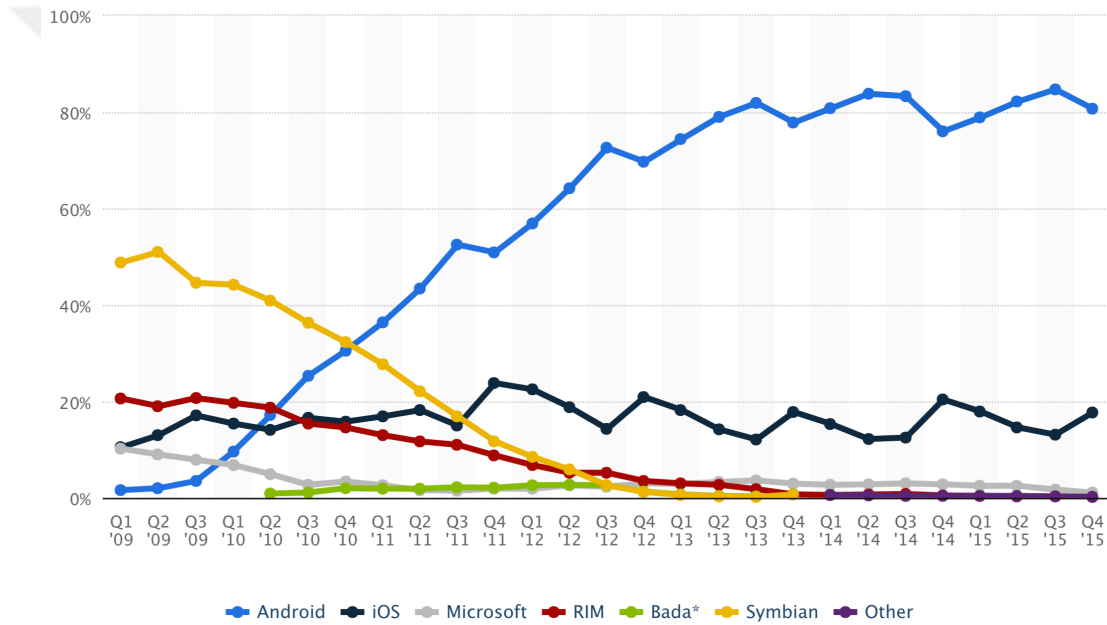
Üç belirtinin hastadan toplanması ile doktorun hastayı görmeden, güvenilir şekilde kontrol işlemini gerçekleştirebileceği bir ortam oluşturulmuştur. Aynı zamanda hastanın güvenliği için doktorun kararları da desteklenmiştir.

Hastaların ilaç uyumlarına dikkat etmediği yapılan çalışmalar sonucunda ortaya konmuştur. Budak vd. (2013) yaptıkları çalışmada Amerika Birleşik Devletleri'nde her yıl 100 milyar doların, hastaların ilaç kullanıma dikkat etmemesi nedeniyle boşa harcandığını belirtmişlerdir. Sunulan başka bir çalışmada, 80 hasta incelenmiş ve verilen ilaçların kullanımına uyumun %84

olduğu tespit edilmiştir. Yapılan çalışma, önerilen tez çalışmasında ilaç takip sistemine yer verilmesinin önemini göstermektedir. Oluşturulan hasta takip sistemine eklenecek ilaç takip kısmının ise hastaların ilaç uyumunu artırması beklenmektedir.

1.4.1. Kapsadığı platformlar

Geliştirilen uygulama 2 farklı işletim sistemi için geliştirilmiştir. Bu işletim sistemleri Android ve IOS'tur. Android ve IOS mobil cihazlar arasında en yüksek pazar payına sahip işletim sistemleridir (Şekil 1.2).



Şekil 1.2. Mobil işletim sistemleri pazar payı (Statistica, 2016).

2009 yılından 2015 yılı sonuna kadar olan her bir çeyrek (Q) için mobil işletim sistemleri kullanım yüzdesi Şekilde 1.2'de verilmiştir. Buna göre 2015 yılı son çeyreğine bakılırsa; Android işletim sistemi için % 80.7, IOS işletim sistemi için ise %17.7 olmak üzere, iki işletim sisteminin toplam pazar payı %98.4 olmaktadır. Böylece yazılacak olan program akıllı telefonlar arasında yüksek bir kullanıma sahip olabilecektir.

1.5. Tezin Bölümleri

Önerilen tez çalışmasının ikinci bölümünde kaynak özetleri üçüncü bölümü olan materyal metot bölümünde kullanılan veri setleri ve yöntem hakkında bilgi verilecektir. Dördüncü bölümde ortaya çıkan araştırma bulguları sunulacak ve son bölüm olan beşinci bölümde de sonuçlar verilecektir.



2. KAYNAK ÖZETLERİ

Sağlık alanında özellikle uzun süren tedavi süreçlerinde hastanın birden fazla zamanda doktor ile buluşması gerekebilmektedir. Bu sorun Akkaya ve Örnek'in (2015) Haziran 2012-2013 tarihlerini kapsayan, Ankara Eğitim ve Araştırma Hastanesi Göz polikliniğine başvuran hastaları inceledikleri çalışmalarında ortaya konmuştur. Yaptıkları çalışma sonucunda polikliniğe gelen hastaların %10.2'sini il dışından gelenlerin oluşturduğunu gözlemişlerdir. Özellikle il dışından gelen hastalara verilen kontrollerin gerçekleştirilmesinde sosyoekonomik nedenlerden (ulaşım giderleri, coğrafi ulaşım zorlukları, zaman, vs..) ötürü aksamaların olduğunu belirtmişlerdir. Akkaya ve Örnek (2015) bu tür sorunların önüne geçme için uzaktan tedavi sistemlerinin gerekliliğini öne sürmüşlerdir. Uzaktan hastalık takibi genellikle telefon haberleşmesi ile sağlanmaya çalışılmıştır. Gröne ve Garcia-Barbero (2001) yapmış oldukları çalışmada telefon vasıtasıyla tedavinin desteklenmesi gerektiğini belirtmişlerdir.

Uzaktan hastalık takibi internete bağlanabilen cihazların yaygınlaşmasıyla giderek popüler hale gelmiş ve bu alanda yapılan çalışmalar genel olarak "mHealth" ismi altında toplanmıştır. mHealth ile alakalı olarak yapılan çalışmalardan bir tanesi Xu vd. (2015) yapmış oldukları çalışmadır. Geliştirmiş oldukları bulut tabanlı sistemi, hastaların antibiyotik kullanımlarının takibi için kullanmışlardır. Böylece sağlık alanında giderek ciddi bir problem haline gelen antibiyotik direnci konusunda doktorların daha bilinçli davranabileceğini ifade etmişlerdir. Xu vd (2015) çalışmasında Çin'in Şangay şehrindeki birçok hastanenin tansiyon şeker gibi ölçümlerin takibi için artık mHealth'ten faydalandıklarını anlatmışlardır.

mHealth uygulamaları sağlık alanının hemen hemen her dalına girmiş bulunmaktadır. Bu dallardan birisi de hamile kadın takibidir. Hamilelerde görülebilen "preeclampsia" hastalığı hamile kadına zarar verebilmektedir. Bu hastalığın takibi için mobil bir program geliştirilmiş ve program vasıtasıyla hastaya; göğüs ağrısı, baş ağrısı, görme bozukluğu ile alakalı bir takım sorular

yöneltmiştir. Ayrıca akıllı cihaza bağlanabilen pulse oksimetre cihazı ile hastanın kanındaki oksijen seviyesi alınarak hastaya öneriler verilmiş ve sonuçlar doktora iletilmiştir (Dunsmuir vd., 2014). Yine sağlık içerisinde farklı bir uygulama ise hastanın yediklerinin mHealth ile takibi fikridir. Frumkin (2010) hastanın yediği yiyeceklerin fotoğrafını, sesli bir şekilde sorularını doktora iletebilmesi fikrini ortaya atmıştır. Fakat bu düşünce ile ilgili fiili bir çalışma yapılmamıştır. Tansiyon, şeker gibi hastalıklarının takibi için ise pek çok çalışma yapılmıştır. Bu çalışmalar hastanede yatan (Airstrip, 2016) veya evde tedavi olan hastaları kapsamaktadır (Bosch, 2011; WellDoc, 2016). Yapılan çalışmaların bir kısmı takip işlemi dışında hastaya ve doktora öneriler de vererek Klinik Karar Destek Sistemi (KKDS) niteliği de taşımaktadır (Georga vd., 2009). Yapılan bütün bu mHealth çalışmalarında tedavinin neden olduğu sosyoekonomik sorunların önüne geçilerek daha güvenilir ve kontrollü tedavi sunulması amaçlanmıştır.

Bilgisayarlı görünümün gelişmesi ile beraber göz tedavisi ile alakalı alanlarda (göz kızarıklığı tespiti, retina sarı nokta tespiti, vd..) da çalışmalar yapılmıştır. Görüntü işleme ile kızarıklık seviyesi tespiti yapan çalışmalardan biri Peterson ve Wolffsohn (2009)'a aittir. Peterson ve Wolffsohn yaptıkları çalışmada görüntü analizi ile tarafsız bir şekilde göz kızarıklık seviyesini bulmaya çalışmışlardır. Çalışma sonucunda geliştirilen tekniğin 16 kat daha güvenilir olduğunu tespit etmişlerdir. Amparo vd. (2013) makalelerinde kızarıklık seviyesi tespitinin insana dayalı olduğundan ve bu yüzden tutarsızlıklar içerdiğinden bahsetmişlerdir. Bunu ortadan kaldırmak için görüntü üzerinden göz kızarıklık seviyesini tespit etmeye çalışmışlardır. Geliştirilen objektif sistem ile başarılı sonuçlar elde etmişlerdir. Rodriguez vd. (2013) kuru göz sendromuna özel, gözden alınan görüntüler üzerinden kızarıklık seviyesini tespit etmeye çalışmışlardır. Çalışmalarında, kuru göz sendromunda yatay olarak irise doğru uzanan damarlar olduğuna dikkat çekilmiş ve bu damarlar dikey sobel filtre ile tespit edilmiştir. Zhao vd. (2014) çalışmalarında kızarıklık seviyesinin tespitinin iyi bir tedavi için önemli olduğunu belirtmiş ve bunun için görüntüler üzerinden objektif metot geliştirdiklerini söylemişlerdir. Abelson vd. (2010) makalelerinde, göz kızarıklık seviyesinin tespitinin hastayı görmeden

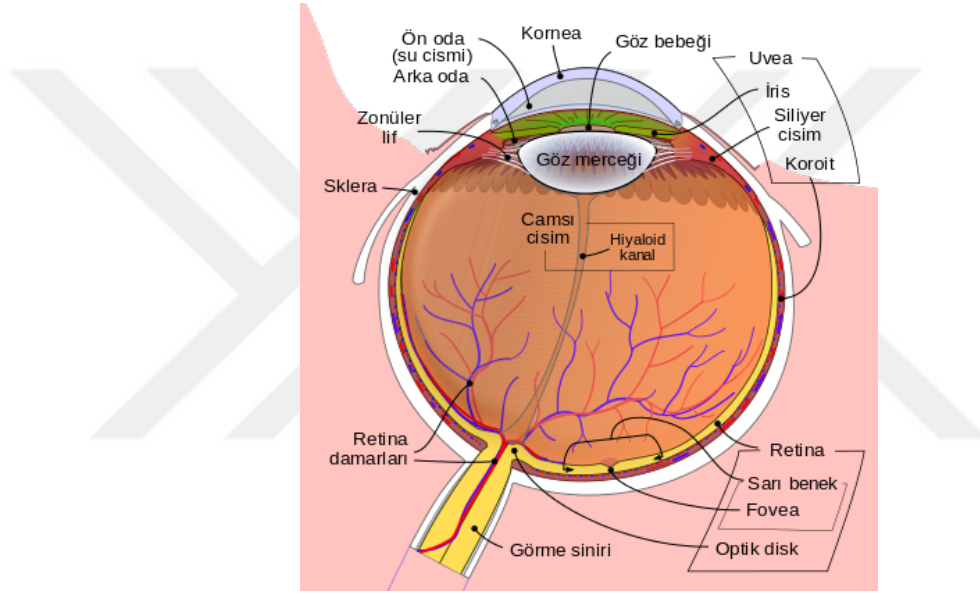
tanı koyacak doktorlar için önemli olduğunu belirtmişlerdir. Fakat kızarıklık seviyesinin mobil ortamlardan tespiti ile alakalı herhangi bir çalışma yapılmamıştır. Akıllı telefonların yaygınlaşması dikkate alındığında bu telefonlara yönelik yapılacak çalışmaların uzaktan göz kızarıklığı tespit uygulamalarının kullanımını yaygınlaştıracağı düşünülmektedir.



3. MATERYAL VE YÖNTEM

3.1. Göz Hastalıkları

Göz, kafatası üzerinde orbita boşluğunda yer alan, bağ doku yönünden zengin yağ yatağına yerleşmiş bir çift duyu organıdır (Şekil 3.1). Bir tanesi yaklaşık olarak 2.5 cm çapındadır. İris denilen renkli yapının ortasında bulunan göz bebeği (pupilla) içerisindeki mercek (lens) ve kornea vasıtası ile, ışığı kırıp arka bölümde bulunan retina (fovea) üstüne düşürerek beyne iletilmesini sağlar.



Şekil 3.1. Gözün yapısı (Wikipedia, 2016).

Göz, görme işlevini yerine getirebilmek için özelleşmiş yapılara sahiptir. Bunlardan bazıları kornea, iris, göz bebeği, mercek, optik sinir, sklera (göz akı), koroit, göz kaslarıdır. Bu yapılardan biri olan kornea, sert ve saydam bir yapıya sahiptir. Gözün ön yüzeyini örter. Korneanın devamında sklera vardır. Sklera saydam olmayan bir yapıya sahiptir. Korneanın olmadığı, göz yüzeyinin geri kalan kısmını örter. Koroit, skleranın hemen altında bulunur ve gözü besleyen damarları içerir. Retina ise gözün en iç kısmında bulunur ve üzerinde fovea bulunur (Gonzalez, 2008).

Gözü oluşturan dokuların işlevini yerine getiremez hale gelmesi veya bozukluk içermesi sonucu göz hastalıkları oluşur. Meydana gelen hastalıklar 3 ana madde

altında belirti vermektedir. Bunlar; görmek anormallikleri, görünüm anormallikleri ve ağrı, rahatsızlıktır. Oftalmologlar bu belirtiler üzerinden teşhis koyar ve hastalığın takibini bu belirtiler üzerinden gerçekleştirirler. Belirtilerin görülebileceği bazı hastalıklar çizelge 3.1’de verilmiştir (Berta, 2007; Riordan-Eva ve Witcher, 2010).

Çizelge 3.1. Belirti tipleri ve bu belirtilerin görülebileceği bazı hastalıklar.

Belirti Tipi	Görülebileceği Hastalıklar
Görme Anormallikleri	Kırma kusuru, pitozis, kornea ödemi, katarakt, aköz boşlukta kanama, vitreus veya retina bozuklukları
Görünüm Anormallikleri	Konjonktivit, keratit, iritis, akut glokom, sarılık, pterijyum, anizokori, pitozis, sklerit, episklerit
Ağrı ve Rahatsızlık	Akut glokom, endoftalmi, iritis, sklerit, kuru göz, keratit

Önerilen tez çalışmasında güvenli takip ortamı sunmak amacıyla, doktorların teşhis ve takipte kullandığı görmek anormallikleri, görünüm anormallikleri ve ağrı, rahatsızlık belirtileri toplanmaya çalışılmıştır. Göz kızarıklık seviyesi ve fotoğraf, görünüm anormallikleri belirtileri için, görme keskinlik testi görme anormallikleri belirtisi için ve göreceli bilgiler ağrı ve rahatsızlık belirtileri için toplanması amaçlanmıştır. Böylece hasta takibi uzaktan ve güvenilir bir şekilde yapılabilecektir.

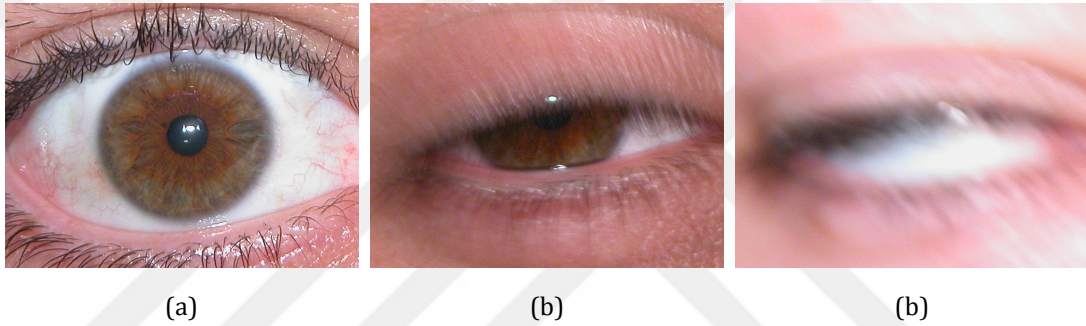
3.2. Materyal

Çalışmaların gerçekleştirilmesi aşamasında 2 farklı veri seti kullanılmıştır. Bunlardan ilki, gözün bölütlenmesi ve çevrimdışı deneyler için kullanılan UBIRIS.v1 görüntü seti (Proença ve Alexandre, 2005) olup; ikincisi ise kontrolsüz ortamda alınmış çevrimiçi çalışmalarda kullanılan SDUeye veri setidir.

3.2.1. UBIRIS.v1 veri seti

UBIRIS.v1 görüntü seti çoğunlukla iris veya sklera üzerinden insan tanımada kullanılmaktadır (Zhou vd. 2012). Görüntü seti içinde 241 farklı kişiden alınan

1877 görüntü bulunmaktadır. Görüntüler 2 farklı oturum halinde alınmış olup, birinci oturumda kısmen kontrollü çevre koşulları altında alım yapılmıştır. İkincisinde ise doğal aydınlatma koşulları altında görüntü alınmıştır. Set içerisinde deneklerden birden fazla alınan tekrar görüntüler bulunmaktadır. Fakat aynı kişiden alınan görüntüler eşlenik olmayıp arasında kalite farklılıkları görülebilmektedir. Bazı görüntülerde göz kapağı tamamen açık, bazılarında ise kısmen veya tamamen kapalıdır (Şekil 3.2). UBIRIS veri seti için alınan görüntüler görünür dalga boyunda ve 600x800 boyutunda alınarak JPEG dosya formatında kaydedilmiştir. Deneylerde kullanılan görüntüler kontrollü çevre koşulları altında alınan 1035 adet görüntüdür.

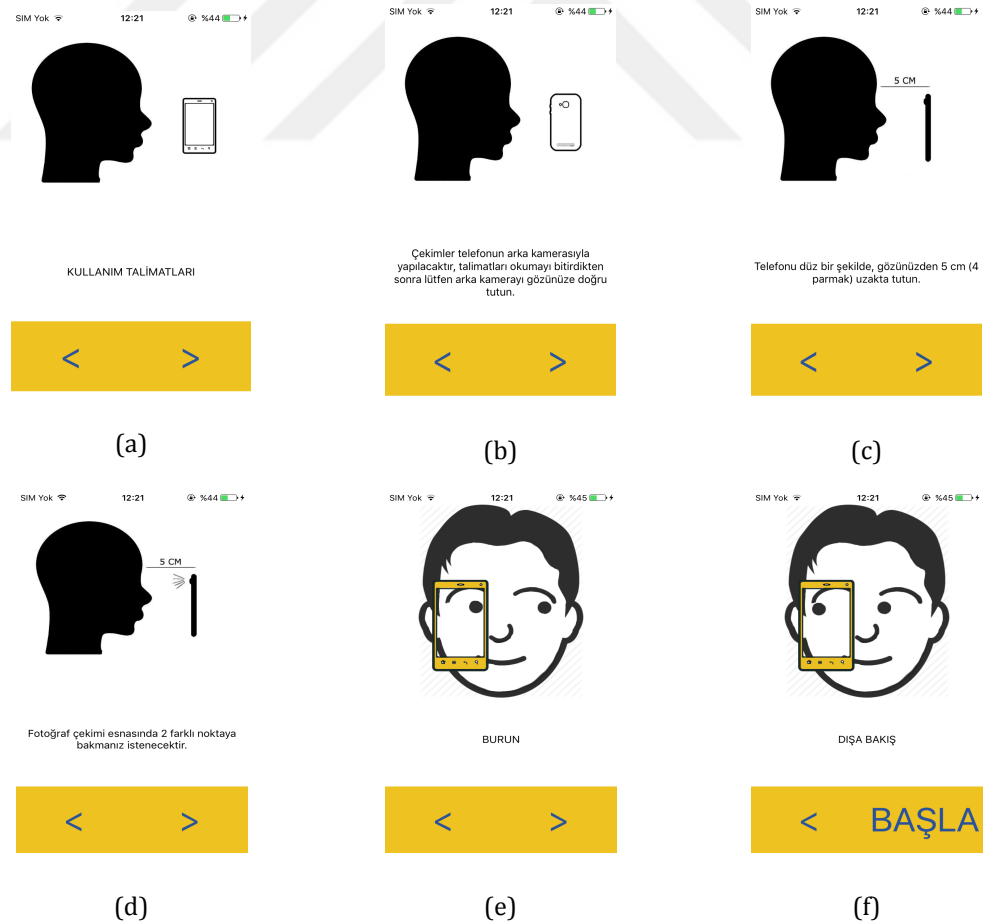


Şekil 3.2. UBIRIS.v1 veri setinden örnek görüntüler, açık göz (a), kısmi kapalı göz (b), kapalı göz (c)

3.2.2 SDUeye veri seti

Çalışmada kullanılan ikinci veri seti ise çevirim içi deneylerde kullanılmak üzere oluşturulan set olup; bu set Süleyman Demirel Araştırma ve Uygulama Hastanesine başvuran, deneylere katılmayı kabul eden gönüllülerden akıllı telefonlar aracılığı ile alınan görüntülerle oluşturulmuştur. Veri setinin kızarıklık seviyesinin otomatik olarak tespiti için de kullanılması planlanmıştır. Bu sebeple denekler özellikle göz kızarıklığı bulunan hastalardan arasından seçilmiştir. Piyasada bulunan eski cihazların ön kameralarının yeterli kaliteye, otomatik odaklama (autofocus) özelliğine sahip olmamaları nedeniyle çekimler cihazın arka kamerası ile yapılmıştır. Görüntülerin toplanması için Android ve IOS mobil platformlarında program yazılmıştır. Denek gözlerinin görüntüleri bu program vasıtası ile otomatik olarak alınmıştır. Görüntülerin doğru bir şekilde

alınabilmesi için kullanıcılardan bazı kurallara uyması istenmiştir. Çekimden önce programın nasıl kullanılacağını ve uyulması gereken kuralları anlatmak için, kullanıcılara görsel eğitim verilmiştir (Şekil 3.3). Bu kurallar şu şekilde sıralanabilir: Kullanıcının uyması beklenen ilk kural, telefonun arka kamerasını göz hizasında, gözünden yaklaşık 5-10cm (yaklaşık 5 parmak mesafesinde) uzaklıkta tutmasıdır. Sonrasında kullanıcının gözünden daha fazla veri alabilmek amacıyla çekim esnasında kafasını hareket ettirmeden gözünü 5 (orta, burun, dışa bakış, yukarı ve aşağı) farklı pozisyona hareket ettirmesi istenmiştir. Ayrıca çekim esnasında gözün hareket etmesi gereken noktalar sesli komut vasıtasıyla kullanıcıya hatırlatılmıştır. Cihazdan cihaza değişmekle birlikte, programın bir gözün 5 adet görüntüsünü alması ortalama olarak 45 saniye sürmektedir. Program görüntü alımını tamamladıktan sonra kullanıcıdan onay alırsa görüntüleri bakış pozisyonu bilgisiyle beraber, program için yazılmış web-servise yüklemiştir.



Şekil 3.3. Program üzerinden verilen 6 adet (a, b, c, d, e, f) eğitim görüntüsü.

Kızarıklık seviyesi göz akındaki damarların hacmi, dolayısıyla damarlardaki kan miktarı ile doğrudan ilintilidir. Bu ilintiden yola çıkarak, kızarıklık seviyesi tespitinde, göz akı içinde bulunan kanlanma tespit edilecektir. Kanlanmanın doğru bir şekilde tespit edilebilmesi için gelecek görüntünün kaliteli olması önemlidir. Cihazların farklı sensörlere sahip olması nedeniyle, kaliteli çekim yapılırken görüntülerin boyutlarında farklılıklar olabilmektedir.

Görüntü alımı yapılırken, gözün ortalama olarak 45 saniye kadar kırılmadan açık kalması nedeniyle giderek kızardığı gözlenmiştir. Bu durumun kızarıklık seviyesi tespitinde hatalara yol açacağı düşünülerek, alınan görüntü sayısı 5'den 2'ye (burun ve dışa bakış) düşürülmüştür. İki görüntünün alımı ise ortalama olarak 15 saniye sürmektedir. Eğer kullanıcıdan arka arkaya birden fazla cihazla görüntü alınacaksa, alımlar arasında en az bir dakika kadar gözünü dinlendirmesi ve kurumaya önlem olarak, gözlerini kırıp göz yüzeyini nemlendirmesi istenmiştir.

Veri seti içinde farklı cihazlar, farklı sensörler, farklı aydınlatma koşulları altında alınmış aynı kullanıcıya ait fotoğraflar bulunmaktadır. Bu yönüyle, bu veri seti, ileride göz akı üzerinden yapılacak insan tanıma algoritmalarının test edilmesinde de kullanılabilir.

3.3. Göz Akı (Sklera) Bölütlemesi

Göz akı içindeki damarlardan insan tanımanın uzaktan, görünür ışık altında, kullanıcıyla etkileşime girmeden ve yüksek başarılı bir şekilde yapılabilmesi; bu tanıma yöntemini iris, parmak izi, avuç içi taraması gibi diğer insan tanıma yöntemlerine karşı (Çizelge 3.2) avantajlı kılmaktadır.

Çizelge 3.2. Biometrik tanımlama sistemleri karşılaştırması (Zhou vd. 2012).

	Başarı	Güvenilirlik	Kararlılık	Uzaktan tanımlama	Kullanıcı işbirliği
Parmak izi	Yüksek	Çok Yüksek	Evet	Hayır	Evet
Yüz	Orta	Orta	Hayır	Bazen	Hayır
Göz Akı	Çok Yüksek	Çok Yüksek	Evet	Bazen	--
İris	Çok Yüksek	Çok Yüksek	Evet	Bazen	Evet
Ses	Düşük	Düşük	Hayır	--	Hayır
El geometrisi	Düşük	Düşük	Evet	--	Hayır
Kulak şekli	Orta	Orta	Evet	Bazen	Hayır
İmza	Düşük	Düşük	Hayır	--	Hayır

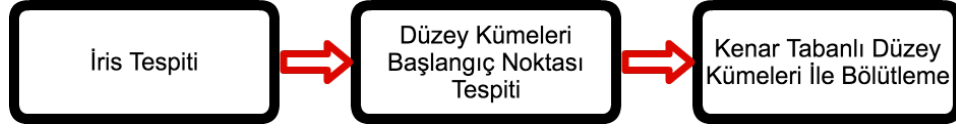
Bu alanda yapılan çalışmaların artması nedeniyle göz akının bölütlenmesi önemli bir problem haline almıştır. Fakat göz akının içindeki damarların renk değerlerinin deriye yakın olması, göz kapaklarının ne kadar açık olduğunun bilinmemesi, kirpikler gibi etmenler nedeniyle bu işlem bir takım zorluklar içermektedir. Bölütleme aşamasında yapılacak bir hata, örneğin kirpiklerin göz akına dahil edilmesi, farklı damar yapılarının varlığı olarak algılanıp tanıma işlemini hatalı duruma getirebilir. Bu bölümün devamında göz akı bölütlemesi için sunulan yöntem ve bu yöntem içerisinde kullanılan algoritmalar anlatılacaktır

Sunulan tez çalışmasında çevrimdışı ve çevrimiçi için farklı yaklaşımlar sergilenmiştir. Çevrimdışı yaklaşım görüntü işleme çözümleri için sıkça kullanılan MATLAB programında, çevrimiçi yaklaşım ise Python programlama dilinde OpenCV kütüphanesi kullanılarak geliştirilmiştir.

3.3.1. Çevrimdışı yaklaşım

Çevrimdışı yaklaşımında UBIRIS.v1 veri setindeki 1035 görüntünün göz akının bölütlenmesi için, Düzey Kümesi (DK) yöntemlerinden olan Kenar Tabanlı Düzey Kümeleri (KTDK) yöntemi kullanılmıştır (Li vd., 2007). Çevrimdışı yaklaşımda işlemler 3 ana adım altında gerçekleştirilmiştir (Şekil 3.4). İris

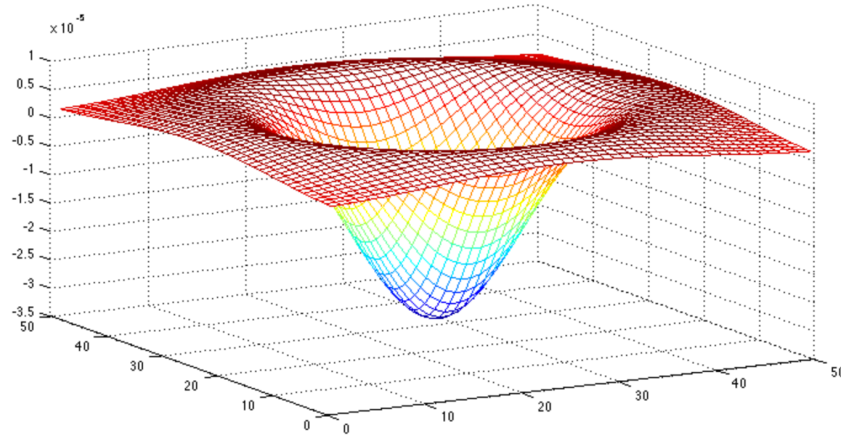
tespiti, Düzey Kümeleri başlangıç noktası tespiti ve KTDK ile bölütleme. Bu yöntemlerin kodlanması MATLAB programı üzerinde yapılmıştır.



Şekil 3.4. Çevrimdışı sistem adımları

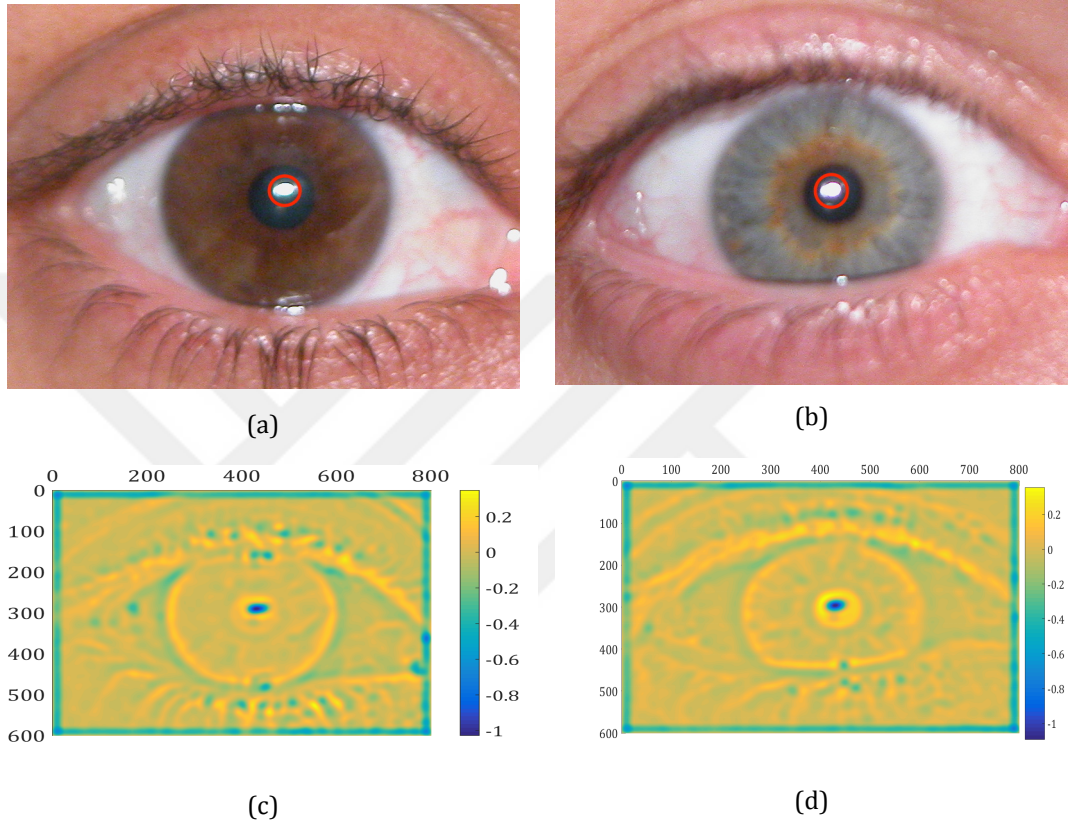
3.3.1.1. İris tespiti

Göz akı bölgesinin tespiti öncesi gözdeki iris bölgesinin sınırlarının tespiti, bu sınırlardan referans olarak da göz akı bölgesinin tespiti hedeflenmiştir. UBIRIS veri seti içerisindeki tüm resimlerde göz bebeğinin üst kısmında parlama olduğu gözlenmiştir. Parlama resimdeki yoğunluk değerinin en yüksek olduğu bölgeye tekabül etmekte olup bu bölge aynı zamanda iris bölgesinin sınırları içerisinde, çoğunlukla merkezinde yer almaktadır. Fakat tek parlama bu bölgede değildir. Göz akı içerisinde ve deri üzerinde de parlamalar gözlenmiştir. Göz bebeği içindeki parlama, göz bebeğinin siyah, parlamaların ise beyaz olması nedeniyle diğer parlamalardan ayrılır. Bu ayırmadan yola çıkılarak göz bebeğinin merkezinin tespit edilebilmesi amacıyla Laplacian of Gaussian (LOG) filtre kullanılmıştır (Şekil 3.6).



Şekil 3.5. LOG filtre.

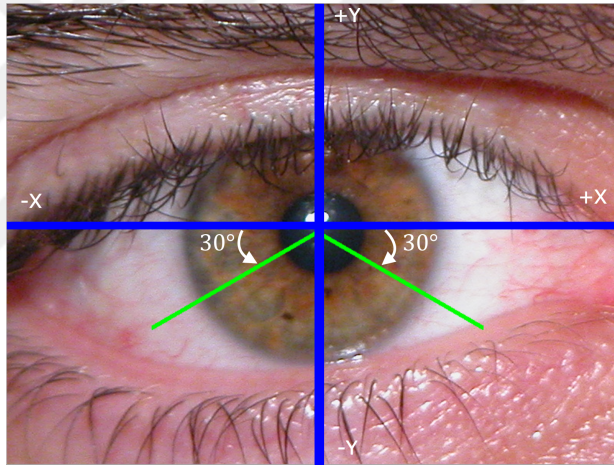
Filtre parlamasının olduğu bölgeye oturarak görüntü üzerindeki minimum değerin üretilmesini sağlamaktadır. Minimum değerin olduğu gözeğin bulunması ile göz bebeğinin bulunduğu nokta kısmen tespit edilmiştir (Şekil 3.6). Bu noktanın %10 oranında aşağı kaydırılması ile göz bebeğinin orta noktası tespit edilmektedir.



Şekil 3.6. Birinci satır; göz bebeği içindeki parlama (a, b), ikinci satır sırasıyla görüntülerin filtrelenmiş durumları (c, d).

İris göz bebeğini çevreler (Şekil 3.1). Mavi, yeşil, kahverengi ve siyah gibi renkler alabilir. Bu nedenle irisin KYM Kırmızı-Yeşil-Mavi (KYM) görüntülerindeki kırmızı bantta bulunan renk yoğunluk değerleri, irisi çevreleyen göz akı veya deriye göre daha düşüktür. Bu özellikten yola çıkılarak irisin merkezinden dışa doğru (+x, -x yönünde) ilerleyerek yoğunlukta yaşanacak değişime göre göz bebeğinin sınırı ve dolayısıyla yarı çapı tespit edilebilir.

UBIRIS veri setindeki bazı görüntülerde göz kapağı yukarıdan veya aşağıdan irisin bir bölümünü kapatmıştır (Şekil 3.2). İnsan gözünün fizyolojisi gereği gözün kapanması durumunda iris, en fazla üst göz kapağı tarafından kapatılmaktadır. Göz yarı açık konumda olsa bile üst göz kapağının alt kapağa göre gözü daha fazla kapattığı gözlenmiştir. İrisin sınırlarının doğru şekilde bulunabilmesi amacıyla göz kapanma karakteristiğine dayanıklı bir algoritma kullanılmalıdır. Bu amaçla, tespit edilen göz bebeğinin orta noktasından sola ve sağa (-x, +x) x eksenini ile 30 derece yapacak şekilde ilerlenmiştir (Şekil 3.7). İlerleme esnasında gözeklerdeki yoğunluk değerlerinin, belirlenen eşik değerini geçtiği tespit edilirse göz akına veya deriye geçildiği düşünülerek ilerleme işlemine son verilmiştir. Sol ve sağ için aynı işlemler uygulanmış iki tarafın ürettiği değerlerin ortalaması alınarak irisin yarı çapı tespit edilmiştir.

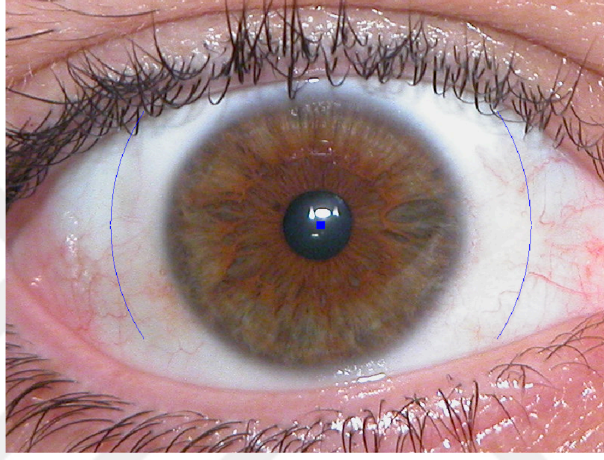


Şekil 3.7. 30 derece açıyla takip edilen çizgiler.

3.3.1.2. Düzey kümeleri başlangıç noktası

DK yönteminin ana prensibi, bir cisme ait olduğu bilinen noktadan, istenilen bölümleri tespit etmektir. Bunun için DK algoritmalarında aidiyeti kesin olarak bilinen bir bölüme Sıfır Düzey Çevriti (SDÇ) ataması yapılmalıdır. İşaretleme olarak tabir edebileceğimiz bu işlem kimi zaman el ile kullanıcı tarafından yapılmaktadır. Fakat bu, otomatik sistemler için olmaması gereken bir durumdur. Otomatik bir sistem geliştirebilmek amacıyla SDÇ tespit edebilecek bir algoritma geliştirilmiştir.

Önerilen sistemde tespit edilen göz bebeği noktasından iris çapının 0.65 katı kadar uzağa gidilerek, -y yönünde 150; +y yönünde 150 birim olmak üzere toplam uzunluğu 300 birim sürecek olan yay çizilir (Şekil 3.8). İşlem göz bebeğinin sağ ve sol tarafı olmak üzere her iki tarafı içinde gerçekleştirilir. Çalışmada her iki yay üzerinde en uygun noktaya (göz akı olduğunun bilinmesi ve merkezi bir bölge olması amaçlanmaktadır) birer adet SDÇ yerleştirilmesi hedeflenmiştir.



Şekil 3.8. Çizilen yay

Şekil 3.8’de görüldüğü gibi çizilen yaylar gözün kapanıklık durumuna göre alt veya üst göz kapağına, kirpiklere temas edebilmektedir. Aşırı kapanmış gözlerde ise yay derinin üstüne çıkmaktadır. Yay üzerinde bulunan göz akı bölgesinin tespit edilmesi gereklidir. Bu işlemde görüntünün kırmızı bileşenin göz akı ve deri bölgelerinde benzer renk yoğunluk değerlerine sahip olması yeşil bileşenin ise fazla gürültü taşıması nedeniyle görüntünün mavi bileşeni kullanılmıştır. Giriş görüntüsü $I(x, y) \in R^3$ ’nin mavi bandı $I_b(x, y) \in R$ alçak geçiren filtre $h(x, y)$ ile yumuşatılarak gürültüden ayıklanır (Denklem 3.1).

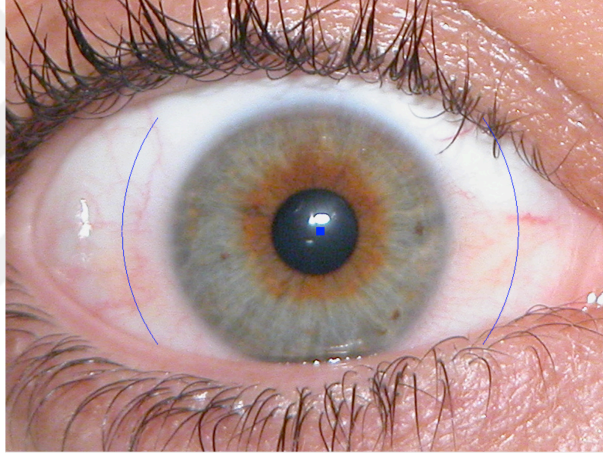
$$\tilde{I}_b(x, y) = I_b(x, y) * h(x, y) \quad (3.1)$$

Daha sonra görüntü $c(m, n)$ çekirdek fonksiyonu ile konvolüsyon işlemine ($*$) tabi tutularak y eksenine yönündeki 1. Türevi (Gradient) alınır. Yay boyunca var olan yoğunluk değişimi hesaplanır (Denklem 3.2, 3.3).

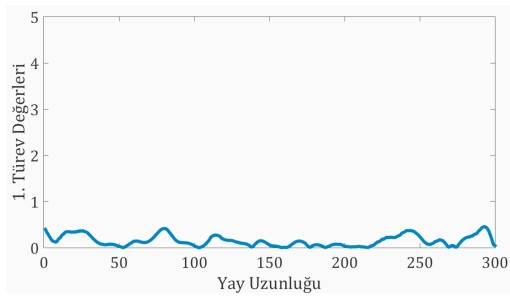
$$D_l(x, y) = \tilde{I}_{b_sol}(x, y) * c(m, n) \quad (3.2)$$

$$D_r(x, y) = \tilde{I}_{b_sağ}(x, y) * c(m, n) \quad (3.3)$$

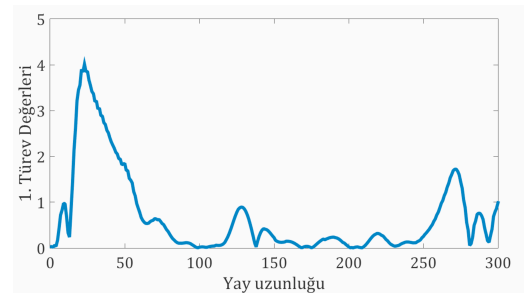
Değişim türevinin y eksini bileşeninde ve çizilen sol, sağ yay üzerinde bulunan $D_l(x, y)$ ve $D_r(x, y)$ değerleri, sırasıyla v_l, v_r vektörlerine atanır. Bu vektörler göz bebeğinin sol ve sağ yanındaki yayda gözlenen yoğunluk değişimini göstermektedir. İdeal bir göz akında bu değerler çok düşük çıkacaktır. Fakat göz akının homojen olmaması, göz akı içerisindeki damarlar ve yayın göz akı dışına çıkması durumları vektörlerin değerlerini yükseltmektedir (Şekil 3.9).



(a)



(b)



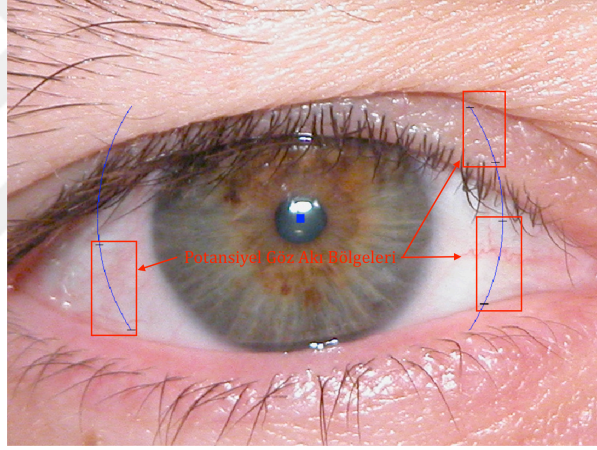
(c)

Şekil 3.9. Görüntü üzerine çizilen yaylar (a), sol yay üzerindeki 1. türev değerleri (b), sağ yay üzerindeki 1. türev değerleri (c). Şekil 3.9 (c)'de görüldüğü üzere sağ yayın 20-55. ile gözlek arası 240-260 gözlek arası yayın kırıpık üzerine yerleşmiş olması türev değerleri yüksek çıkmıştır.

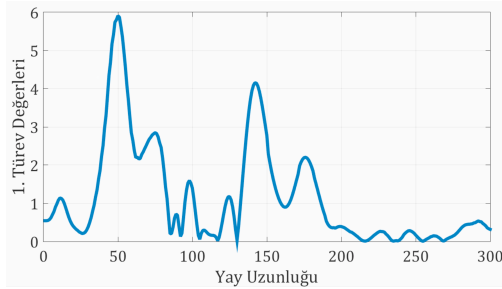
Atama işleminden sonra vektörler üzerinde değeri 1'in altında olan bölgeler tespit edilir (Denklem 3.4).

$$b_l = v_l < 1, b_r = v_r < 1 \quad (3.4)$$

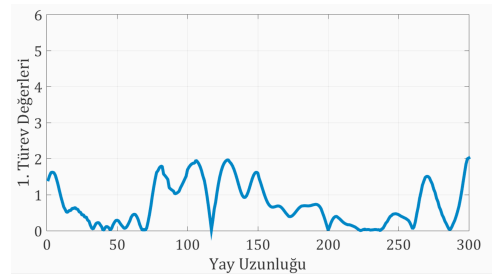
b_l, b_r bölgeleri değişimin az olduğu göz akı olarak varsayılabilecek bölgelerdir. Değişimin fazla olduğu yerlerin türev değerleri 1'den büyük olduğundan Denklem 3.4 ile bu bölgeler elenmiştir. Fakat bazı durumlarda yay derinin üstüne çıkabilmektedir. Deri üzerinde de türev değişimi birin altında olan ve göz akı gibi karakteristiğine sahip bölgeler bulunabilmektedir. Bunun gibi durumlarda b_l, b_r birden fazla potansiyel göz akı bölgesi içermektedir (Şekil 3.10).



(a)



(b)



(c)

Şekil 3.10. Çizilen yaylar üzerindeki potansiyel göz akı bölgeleri (2 siyah çizgi arası) (a), -x eksenini için 1. türev değerleri (b), +x eksenini için çizilen yay üzerindeki 1. türev değerleri (c). Şekil 3.10 (c)'de 75. gözeğin üstünde ve 160. gözeğin altında olmak üzere 2 bölge gözlenmiştir.

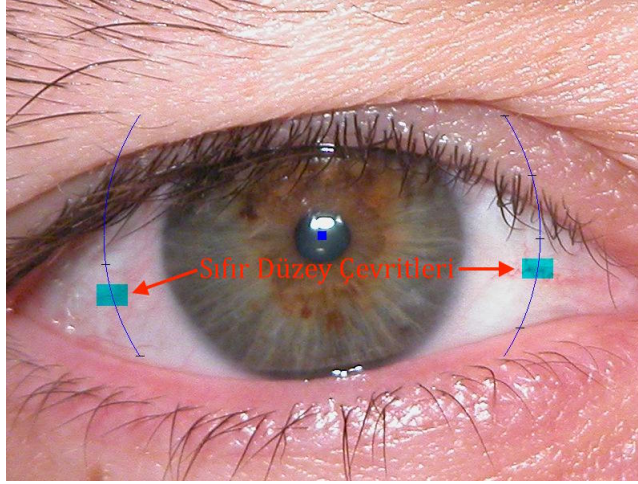
Başlangıç noktasını en iyi şekilde seçebilmek amacıyla bölgeler arasında tercih yapılması gerekmektedir. Bu tercih işlemi, bölgelerin merkez noktalarının göz bebeği merkezine olan dikey uzunluğuna göre yapılmıştır. Böylece gözün sol ve sağ kısmına çizilen yaylar üzerinde (Şekil 3.9) göz akı bölgesini içeren gözleklerin koordinatları sırasıyla B_l ve B_r vektörüne atanmıştır. SDÇ'nin yerleştirilmesi, B_l, B_r bölgelerinin içerisine denklem 3.5, 3.6 ve 3.7'de anlatılan şartları sağlayacak şekilde yapılmıştır.

$$bn_l = B_l(1) + \|B_l\| * \frac{1}{3}, \quad bn_r = B_r(1) + \|B_r\| * \frac{1}{3} \quad (3.5)$$

$$g_l = \|B_l\| * \frac{1}{3}, \quad g_r = \|B_r\| * \frac{1}{3} \quad (3.6)$$

$$u_l = \|B_l\| * \frac{1}{2}, \quad u_r = \|B_r\| * \frac{1}{3} \quad (3.7)$$

Burada $B_l(1)$ tespit edilen göz akı bölgesinin başlangıç koordinatını; $\|B_l\|$ ise B_l vektörünün uzunluğunu vermektedir. Denklem 3.5, 3.6, 3.7'de oluşturulan $bn_l, bn_r, g_l, g_r, u_l, u_r$ sırasıyla sol ve sağ yay üzerinde oluşturulacak olan düzey kümesi SDÇ (dikdörtgen bölge) başlangıç satırı, genişliğini ve uzunluğunu ifade etmektedir. Oluşturulan üç değer ile yay üzerinde kapalı bir dikdörtgen bölgesi alınabilmektedir. Bu değerler ile KTDK'nın SDÇ'sini oluşturan dikdörtgen alanlar belirlenir (Şekil 3.11).



Şekil 3.11. Çizilen yaylar ve düzey kümeleri sıfır düzey çevriti (cyan dikdörtgenler).

3.3.1.3. Kenar tabanlı düzey kümeleri

Düzey kümeleri yöntemi bölütleme işlemlerinde pek çok yerde kullanılmaktadır. En önemli avantajlarından birisi sabit bir şekle sahip kalmayarak içinde bulunduğu nesnenin şeklini alabilmesi, düzey kümelerinin bölünebilmesi, birleşebilmesidir (Li vd. 2010). Bu avantajlarının yanında bir takım sorunları da barındırmaktadır. Bu sorunların başında değerlendirme sırasında oluşan ve kararlılığı yok eden nümerik hatalar gelmektedir. Sorunu ortadan kaldırmak için yeniden başlatma denilen düzey kümeleri fonksiyonunu düzenleyen ve kararlı tutmaya çalışan işlem gerçekleştirilir. Bu işlem denklem 3.8'deki kısmi diferansiyel denklemin çözümü ile yapılır.

$$\frac{\partial \phi}{\partial t} = \text{sign}(\phi)(1 - |\nabla \phi|) \quad (3.8)$$

Burada, *sign* işaret fonksiyonu; ϕ ise yeniden başlatılacak olan zamana (t) bağlı düzey küme fonksiyonudur. Fakat yeniden başlatma bazı sorunları ortaya çıkarmaktadır. Bunların başında bu işlemin pratikte nasıl uygulanacağı yer almaktadır. Aynı zamanda sorunların çözülmesi için önerilen yöntemlerin karmaşıklığı çok yüksektir. Li vd. (2010) Kenar Tabanlı Düzey Kümeleri (KTDK) adıyla yeni bir yöntem önermiştir. KTDK yönteminde uzaklık düzenleme ve dışsal enerji terimleri ile başlangıç noktası olan SDÇ'nin istenilen bölgelere

doğru ilerletilmesi sağlanmıştır. Bu işlem gerçekleştirilirken denklem 3.9'da formülü verilen enerji fonksiyonundan faydalanmışlardır.

$$\mathcal{E}(\phi) = \mu\mathcal{R}_p + \lambda\mathcal{L}_g + \alpha\mathcal{A}_g \quad (3.9)$$

λ ve α enerji fonksiyonlarının katsayıları olup $\lambda > 0$ ve $\alpha \in \Omega$ şartlarını sağlamalıdır. Burada Ω görüntünün ait olduğu uzayı temsil etmektedir. Enerji fonksiyonun hesabı ise denklem 3.10, 3.11, ve 3.12'de olduğu gibi yapılmaktadır.

$$g \triangleq \frac{1}{1 + |\nabla G_\sigma * I|} \quad (3.10)$$

$$\mathcal{L}_g \triangleq \int_{\Omega} g \delta(\phi) |\nabla \phi| dx \quad (3.11)$$

$$\mathcal{A}_g \triangleq \int_{\Omega} g H(-\phi) dx \quad (2.12)$$

Denklem 3.10'da bulunan G_σ yumuşatma işleminde kullanılacak, σ standart sapmaya sahip gauss çekirdeği, g ise I giriş görüntüsünün değişim türevidir. Denklem 3.11 ve 3.12 de bulunan δ , H ise sırasıyla Dirac Delta (DD) ve Heaviside fonksiyonlarıdır.

Enerji terimi olan \mathcal{L}_g , DD fonksiyonu ile düzey kümeleri fonksiyonunun çizgisel integralidir. Bu fonksiyon minimum değerine ulaştığı zaman, düzey kümeleri fonksiyonu görüntü üzerinde cismin sınırlarına oturmuştur (Şekil 3.12). \mathcal{L}_g enerji teriminin minimuma yaklaştırılması işlemi döngüsel bir işlemdir. Kullanıcı tarafından belirlenen bir döngü sayısı kadar bu işlem tekrarlanır ve minimum değerine ulaştırılmaya çalışılır.



Şekil 3.12. KTDK Sıfır Düzey Çevriti (a) ve bölütleme sonucu (b) (Li vd., 2010).

3.3.2. Çevrimiçi yaklaşım

Çevrimiçi yaklaşımda bölütleme işleminin uygulanabilmesi için hastaneye başvuran gönüllü hastalardan alınan görüntülerden oluşan SDUeye veri seti kullanılmıştır. Çevrimiçi yaklaşımın algoritmasının kodlanması için Python dili ve OpenCV kütüphanesinden faydalanılmıştır.

3.3.2.1. Python programlama dili

Python dili basit söz dizimi, kolay kullanımı, nesne yönelimli olması, modüler , platform bağımsız, C ve C++ gibi temel seviye diller ile yazılan kodların kolayca aktarılabilmesi ve açık kaynaklı olması gibi avantajlardan dolayı çokça tercih edilen bir dildir. NumPy, SciPy OpenCV gibi matematiksel işlemler ve fonksiyonların kolay bir şekilde kullanımına izin veren kütüphaneler de destek vermektedir. Bunların yanında C, C++ gibi dillere nazaran daha kolay kodlama imkanı sağlaması, yorumlanan bir dil olması, kendi çöp toplama mekanizmasına sahip olması ve MATLAB gibi programlara nazaran hızlı yanıt vermesi bilimsel çalışmalar arasında da bu dili popüler hale getirmiştir. Anlatılan etmenlerden dolayı sunulan çalışmanın çevrimiçi bölümünde Python programlama dilinden faydalanılmıştır.

3.3.2.2. OpenCV görüntü işleme kütüphanesi

OpenCV (Open Source Computer Vision Library) kütüphanesi 1999 yılında Intel laboratuvarlarında C ve C++ dilleri için geliştirilmeye başlanmış ve ilk sürümü

2000 yılında yayınlanmış bilgisayarlı görü kütüphanesidir. 2016 yılı itibari ile Windows, Linux, Mac OS ve Android gibi işletim sistemlerine destek vermektedir. Ayrıca C, C++, Python, Java ve MATLAB üzerinden geliştirme yapılabilmektedir. OpenCV içerisinde 2500'den fazla optimize edilmiş algoritma bulundurmaktadır (OpenCV, 2016). Bu algoritmalar görüntü işleme ve görüntü tanımda çokça kullanılan, bilinen ve bazıları alanının en iyi algoritmalarıdır. OpenCV kütüphanesinin optimize edilmiş geniş algoritma havuzuna sahip olması nedeniyle, görüntü işleme tarafının geliştirilmesinde OpenCV kütüphanesinden faydalanılmıştır.

Çevrimiçi yaklaşımda göz akı tespiti 5 adımda gerçekleştirilmiştir (Şekil 3.13).



Şekil 3.13. Çevrimiçi sistem adımları

3.3.2.3. Görüntü boyutlandırma

Kullanılan SDUeye veri seti farklı boyutlarda, farklı aydınlatma koşullarında, farklı sensörler tarafından kayıt edilmiş görüntüleri içermesi nedeniyle, görüntüler ön işlemeden geçirilmiştir. Bu ön işleme değişik boyutlardaki görüntülerin tek bir boyuta indirilmesi amacıyla yapılmıştır. Bu işlem için bikübik interpolasyon yöntemi kullanılmıştır.

3.3.2.3.1. İnterpolasyon

Görüntü işlemede interpolasyon, döndürme, büzme ve büyütme gibi işlemlerde sıkça kullanılan bir yöntemdir. Bu işlem basitçe, bilinen bir veriyle bilinmeyen bir noktadaki veriyi tahmin etmektir diye ifade edilebilir. En bilinen yöntemler en yakın komşu, bilinear ve bikübik interpolasyon yöntemleridir. Bu yöntemlerden en yakın komşu interpolasyonu tahmin edilmek istenen $I(x, y)$ noktasının en yakınında bulunan gözeteki yoğunluk değerini alarak $I(x, y)$ noktasına yerleştirmektedir. Bilinear interpolasyon yöntemi ise $I(x, y)$

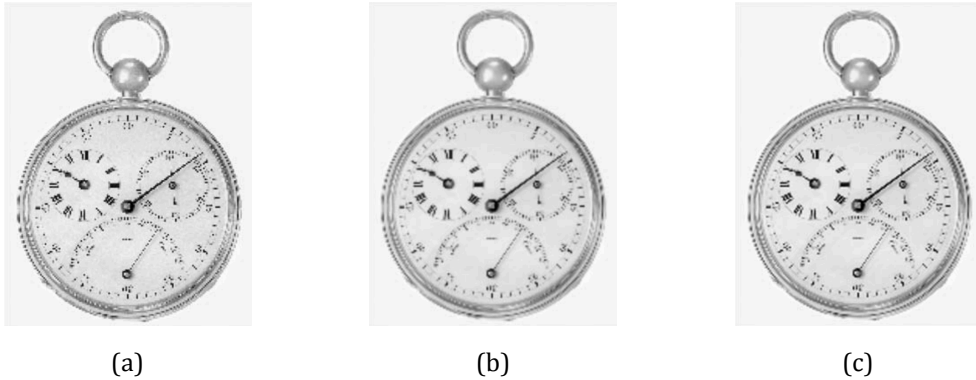
gözeğinin 4 komşusunun yoğunluk değerleri yardımıyla hesaplanır (Denklem 3.13).

$$I(x, y) = ax + by + cxy + d \quad (3.13)$$

Denklem 3.13'te bulunan 4 kat sayı (a, b, c, d), (x, y) gözeğinin komşuları yardımıyla hesaplanır. Diğer bir interpolasyon yöntemi olan bikübik interpolasyon ise (x, y) gözeğinin 16 komşusu yardımıyla hesaplanır (Denklem 3.14).

$$I(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} I(x^i, y^j) \quad (3.14)$$

Bu denklemde x^i, y^j komşu pixel lokasyonlarını; a_{ij} ise ij lokasyonundaki yoğunluk değerinin interpolasyona olan katkısını belirtmektedir. En yakın komşu algoritması görüntü üzerinde istenilmeyen değişikliklere neden olmaktadır. Bilineer interpolasyonda bir önceki yöntemin meydana getirdiği değişiklikler ortadan kalkmıştır. Fakat görüntüde bulanıklık meydana gelmektedir. Bikübik interpolasyon kendinden önce bahsedilen algoritmalarındaki sorunları ortadan kaldırmış ve daha keskin, detaylarını kaybetmemiş görüntülerin oluşturulmasına olanak vermiştir (Şekil 3.14).



Şekil 3.14. Sırasıyla en yakın komşu (a), bilineer (b), bikübik (c) interpolasyonlar ile 72dpi düşürülüp daha sonra orjinal hale getirilen görüntüler (Gonzalez ve Woods, 2008).

Bikübik yönteminin karmaşıklığı diğer yöntemler ile kıyaslandığında daha fazladır. O_{en}, O_{bl}, O_{bk} sırasıyla en yakın komşu, bilineer ve bikübik interpolasyonunun karmaşıklığı olsun. Bu durumda aralarındaki bağıntı denklem 3.15'deki eşitsizlik gibi olacaktır.

$$O_{en} < O_{bl} < O_{bk} \quad (3.15)$$

Bikübik interpolasyon karmaşıklığına rağmen, kalitesi nedeniyle ticari yazılımlar da dahil olmak üzere pek çok alanda kullanılmaktadır (Gonzalez ve Woods, 2008).

3.3.2.4. Görüntü yumuşatma

Görüntü yumuşatma uygulanacak kenar belirleme algoritması öncesinde gürültüleri yok etmek için bir ara adım niteliği taşımaktadır. Bu işlem için literatürde tercih edilen yöntemlerden birisi olan lineer filtreleme kullanılmıştır (Gonzalez ve Woods, 2008).

3.3.2.4.1. Lineer filtreleme

Lineer filtreleme yöntemi, literatürde ortalama filtresi olarak da geçmektedir. Ortalama filtresi $I(x, y)$ gözeğindeki yoğunluk değerini kendisinin ve komşularının ortalamasını alarak bulmaktadır (Denklem 3.16). Böylece görüntü, yumuşatılarak, gürültülerden kurtulmuş bir görüntü oluşturulmaktadır. Denklem 3.16'deki n ve m oluşturulan filtrenin boyutunu belirlemektedir. Filtrenin boyutu arttıkça gürültülerden kurtarma olasılığı artsa da aynı zamanda görüntüyü daha fazla bulanıklaştırmaktadır.

$$I(x, y) = \frac{1}{m * n} * \sum_{i=0}^m \sum_{j=0}^n x^i y^j \quad (3.16)$$

3.3.2.5. Kenar tespiti

Görüntüde kenar, ani renk değişikliklerinin meydana geldiği gözele verilen isimdir. Genellikle görüntülerin 1. dereceden türevleri alınarak elde edilirler.

Varsayalım $I(x, y)$ bizim giriş görüntümüzü, (x, y) 'de görüntünün içindeki gözeği ifade ediyor olsun. Bu durumda $I(x, y)$ görüntüsünün 1. türevi olan ∇I denklem 3.17'deki şekilde ifade edilir.

$$\nabla I \equiv \text{grad}(I) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \end{bmatrix} \quad (3.17)$$

Burada g_x , x yönündeki değişimi g_y , y yönündeki değişimi ifade etmektedir. Değişim vektörünün önemli özelliklerinden birisi büyüklük değeridir. ∇I değişim matrisinin büyüklüğü denklem 3.18'deki şekilde hesaplanır.

$$B(x, y) = \text{mag}(\nabla I) = \sqrt{g_x^2 + g_y^2} \quad (3.18)$$

Değişimin yönünü gösteren $\alpha(x, y)$ matrisinin ifadesi ise denklem 3.19'daki şekildedir.

$$\alpha(x, y) = \tan^{-1} \left[\frac{g_x}{g_y} \right] \quad (3.19)$$

$B(x, y)$, $\alpha(x, y)$ matrisleri I matrisi ile aynı boyuta sahiptir. Belirtilen türev alma yaklaşımı doğrudan görüntüler için uygulanamaz. Görüntülerde dijital değerler ile çalıştığımız için türeve dijital bir yaklaşım göstermemiz gerekecektir. Bunun için ise (x, y) gözeğinin komşuluklarından faydalanılmaktadır. Bu durumda gözeğin x ve y yönündeki türevleri olan g_x ve g_y aşağıdaki şekilde tanımlanır (Denklem 3.20, 3.21).

$$g_x = \frac{\partial I(x, y)}{\partial x} = I(x + 1, y) - I(x, y) \quad (3.20)$$

$$g_y = \frac{\partial I(x, y)}{\partial y} = I(x, y + 1) - I(x, y) \quad (3.21)$$

Bu yaklaşımı göz önünde bulundurarak bir takım filtreler oluşturulmuştur. Bunlardan ilki Roberts tarafından 1965 yılında ortaya atılmıştır. Roberts, bu filtresinde çapraz değişiklikleri göz önünde bulundurmıştır. 1970 yılında daha büyük bölgeyi ele alan ve daha iyi sonuçlar veren Prewit operatörü ortaya çıkmıştır. Yine aynı yıl Prewit operatörüne benzeyen Sobel geliştirilmiştir (Gonzalez ve Woods, 2008). (Şekil 3.15). Sobel operatörü 1. satır ve üçüncü satırda bulundurduğu 2 katsayıları nedeniyle daha yumuşak, gürültülerden arınmış bir çıktı vermektedir.

-1	0	0	-1
0	1	1	0

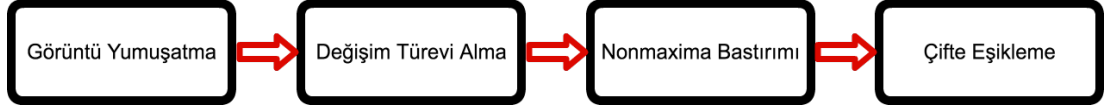
(a)

-1	-1	-1	-1	0	1	-1	-2	-1	-1	0	1
0	0	0	-1	0	1	0	0	0	-2	0	2
1	1	1	-1	0	1	1	2	1	-1	0	1

(b) (c)

Şekil 3.15. Roberts(a), Prewit (b) ve Sobel (c) filtre örnekleri.

Bu yaklaşımlardan farklı olarak sadece değişim türevi kullanmayan yaklaşımlarda ortaya atılmıştır. Bunlardan birisi de Canny kenar bulma algoritmasıdır. Canny algoritması diğerlerinden farklı olarak 4 adımda gerçekleştirilir. Bu adımlar şekil 3.16'de gösterilmiştir.



Şekil 3.16. Canny algoritma Adımları

İlk olarak görüntü Gauss filtresinden geçirilerek yumuşatılır. Varsayalım $I(x, y)$ yumuşatmak istediğimiz görüntümüz ve $G(x, y)$ de gauss filtremiz olsun. Bu durumda $G(x, y)$ gauss filtresi denklem 3.22’de gösterildiği şekilde ifade edilir.

$$G(x, y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (3.22)$$

* işleminin convolüsyon (filtreleme) işlemi olduğu varsayılır ise $I_s(x, y)$ filtrelenmiş görüntüsüne denklem 3.23’teki gibi ulaşılır.

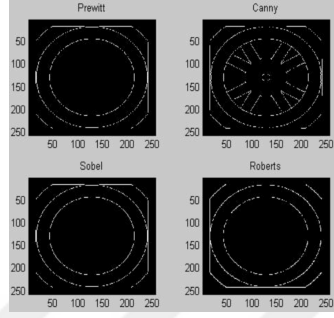
$$I_s(x, y) = G(x, y) * I(x, y) \quad (3.23)$$

Giriş görüntüsü gauss filtresi ile filtrelendikten sonra değişim türevinin büyüklüğü ve açısı denklem 3.18 ve 3.19’de olduğu gibi hesaplanır. Büyüklük matrisi olan B , yerel minimum etrafında geniş yükselteler içermektedir. Bir sonraki adım bu yükselteleri küçültmek için yapılacak olan düşük yoğunluk (nonmaxima), yüksek bölümleri bastırma adımıdır. 3x3 lük komşuluk düşünülecek olursa her biri 8 komşuya denk gelecek şekilde d_1, d_2, \dots, d_8 olmak üzere 8 adet yön belirlenir. Gözetim türevinin açısına göre bu yönlerden ait olduğuna atanır. Eğer (x, y) gözeği $I(x, y)$ ’nin d_k yönündeki komşularının birinden bile küçükse g_N ’e 0 ataması eğer 2 komşusundan büyükse g_N ’e $M(x, y)$ ataması yapılır. Böylece geniş yükselteler bastırılmış olur. Sonrasında ise çifte eşikleme ile Canny algoritması kenar bulmayı sonlandırır.

Canny algoritması diğer kenar bulma operatörlerine göre daha fazla karmaşıklığa sahip olsa da başarılı sonuçları (Şekil 3.17) nedeniyle diğer algoritmalara göre daha fazla tercih edilmektedir. Önerilen tez çalışmasında da Canny algoritmasından yararlanılmıştır.



(a)



(b)

Şekil 3.17. Kenar tespiti işleminin uygulanacağı görüntü (a) ve sonuçları (b) (Gonzalez ve Woods, 2008).

3.3.2.6. Çember tespiti

Kenarları tespit edilmiş, üzerinde n tane kenar noktası bulunan bir g_N görüntüsünün verildiğini varsayalım. Bu görüntünün içindeki çizgi üzerinde bulunan ve n kümesinin bir alt kümesini ifade eden n_c kümesini bulmaya çalışacağız. n kümesinin içinde bulunan ve aynı çizgi üzerinde bulunan n_c alt kümesini bulabilmek için n kümesinin her bir eleman için yaklaşık olarak n^3 lük bir işlem yapmak gerekmektedir. Yüksek karmaşıklıktan kurtarmak için Hough (1962), literatürde Hough dönüşümü olarak da bilinen yöntemi önermiştir.

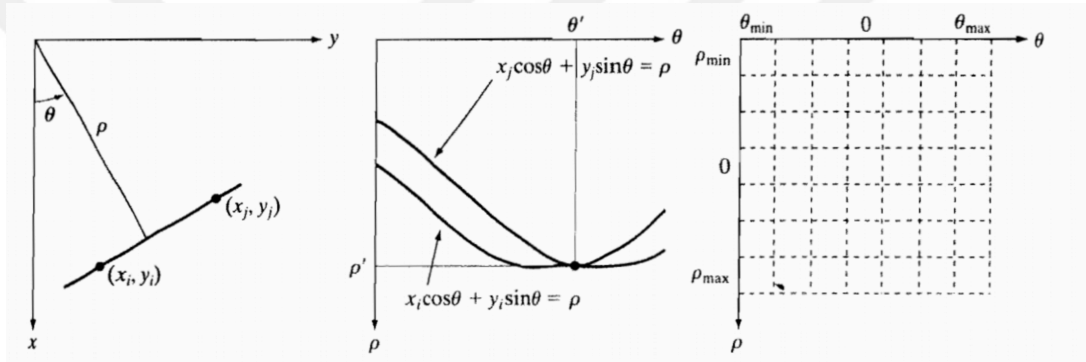
xy düzleminde g_N görüntüsüne ait (x, y) noktası olduğunu varsayalım. Bu noktadan sonsuz sayıda çizgi geçebilir. Fakat bütün x_i, y_i noktaları denklem 3.24'ü sağlamak zorundadır.

$$y_i = ax_i + b \quad (3.24)$$

Denklem 2.24'te a, b ifadeleri sabittir. Bilindiği üzere görüntüler matrislerden oluşmaktadır. Bu denklemi matrisler üzerinde ifade edebilmek için ve aynı zamanda eğimin sonsuza gitmesini engellemek için kutupsal koordinat sistemine dönüştüreceğ olursak denklem 3.25'i elde ederiz.

$$\rho = x \cos \theta + y \sin \theta \quad (3.25)$$

Denklem 3.25'de ρ , çizgiye dik olacak şekilde çizilen ve çizginin merkeze olan uzaklığını ifade edecek olan vektörün büyüklüğünü, θ ise çizginin x eksenine yaptığı açığı ifade etmektedir (Şekil 3.18).



Şekil 3.18. Hough dönüşümü ρ, θ değeri (Gonzalez ve Woods, 2008).

Hough, görüntü üzerinde tespit edilmiş tüm kenar noktalarının üzerinden geçebilecek tüm çizgilerin bu denklem vasıtasıyla hesaplanmasını önermiştir. Bir nokta düşünüldüğünde üzerinden sonsuz sayıda çizgi geçebilir. Fakat kutupsal koordinat sistemi üzerine yerleştirilmiş $m \times n$ boyutundaki bir görüntü için bu sayılar denklem 3.26 ve 3.27'deki eşitsizlikler ile sınırlandırılmıştır ($\rho_{min}, \rho_{max}; \theta_{min}, \theta_{max}$).

$$-90 \leq \theta \leq 90 \quad (3.26)$$

$$-\sqrt{n^2 + m^2} \leq \rho \leq \sqrt{n^2 + m^2} \quad (3.27)$$

Sınırlandırmalar ışığında her nokta bu sınırlar altında bir eğri oluşturur (Şekil 3.18). Hough, eğrilerin kesişim noktalarını ifade edebilecek, en az ve en çok ρ, θ

için bir akümülatör matrisi oluşturmuştur. Eğrilerin oluşturduğu kesişimlere ait ρ , θ değerleri üzerindeki kesişim sayısı ile ifade edilir. Akümülatör matrisi üzerinde en yüksek değere sahip noktanın ρ , θ karşılıkları, o görüntü üzerindeki en büyük çizgi olarak ifade edilir. Bu şekilde Hough, görüntü üzerinde meydana gelen tüm çizgileri ifade ederek büyükten küçüğe doğru sıralanmış bulunmaktadır.

Hough'un önerisi sadece çizgiler için geçerli değildir. Polar koordinat sisteminde ifade edilebilen çember, elips gibi cisimler için de uygulanabilmektedir. Özellikle görüntü üzerindeki çemberlerin bulunması amacıyla sıklıkla kullanılır. Çemberler, kutupsal koordinat sisteminde denklem 3.28'deki gibi ifade edilmektedir.

$$r = (x - x_i) + (y - y_i) \quad (3.28)$$

Denkleminde (x, y) ikilisi g_N görüntüsünde kenar üzerinde bulunan herhangi bir gözeği, (x_i, y_i) çemberin merkezini ve r ise çemberin yarı çapını ifade etmektedir. Çizgi tespitinden farklı olarak gözetler 2 değil, x_i, y_i, r olmak üzere toplamda 3 parametreyi akümülatör üzerinde oylayacaklardır. Akümülatör üzerinde en çok değere sahip olan üçlüler ise görüntü üzerindeki çemberler olarak algoritma tarafından ön plana çıkarılır. Böylece görüntüden istenilen miktardaki çember elde edilir.

Çember tespit edildikten sonra görüntünün, ilgili bölgesinin kesilmesi gerekmektedir. Kesilecek olan dikdörtgensel bölgenin uzunluğunu ve genişliğini q_g, q_u sabitleri ayarlanmaktadır (Denklem 3.29).

$$q_g = (r + 60) * S_1, q_u = S_2 \quad (3.29)$$

2.3.2.7. Eşikleme

Eşikleme işlemi KYM görüntünün görüntü bantları arasındaki yoğunluk farkına göre yapılmıştır (Denklem 3.31).

$$M(x, y) = \begin{cases} 1, & (I_r(i, j) - I_b(i, j)) > T \text{ OR } (I_r(i, j) - I_g(i, j)) > T \\ 0, & \text{else} \end{cases} \quad (3.31)$$

Yukarıdaki denklemde bulunan M oluşturulan maskeyi, I_r, I_g, I_b Giriş görüntüsü olan I 'nin bantlarını T ise koyulan eşik değerini temsil etmektedir. Eşikleme yapıldıktan görüntü içerisindeki en büyük bölge tespit edilmiştir. Tespit edilen bu bölge göz akı olarak varsayılmıştır.

2.4. Göz Hastalıkları Takip Sistemi

Hastalık takibi, hastalığın seyrinin, hastada görülen belirtilerin, doktor tarafından gözlenmesi ile yapılmaktadır. Bu işlemler tedavisi uzun süren ve yatarak tedavi edilmeyen hastalar için problemlerin ortaya çıkmasına neden olmaktadır. Bu tip hastalardan, verilen tedaviden bir müddet sonra hastaneye tekrar gelmesi istenmektedir. İleri bir zamana verilen randevu ise hastaya iş gücü ve maddi kayıp olarak yansımaktadır. Ortaya çıkan problemler göz önünde bulundurularak uzaktan takip sistemleri ortaya atılmıştır.

3.4.1, 3.4.2, 3.4.3 bölümlerinde sırasıyla programın işleyişi ile alakalı görsel bilgilendirme, yazılım ve yazılım geliştirme ortamları hakkında bilgilendirme ve yazılımın kabiliyetleri hakkında gerekli bilgiler sunulacaktır.

3.4.1. Model

Programın kullanıcıların verilerini, hastaların eklediği görüntüleri saklayabilmesi amacıyla Java programlama dilinde Spring kütüphanesinden yararlanılarak web servis yazılmıştır. Veri tabanı sistemi olarak NoSQL prensibine göre çalışan MongoDB kullanılmış ve web servis ile istemci arasındaki haberleşme JSON formatında yapılmıştır.

Web servis üzerinde 3 farklı kullanıcı tipi vardır. Bunlar yönetici, doktor ve hastadır. Yönetici sistemin yönetilmesinden sorumludur. Her türlü yetkiye

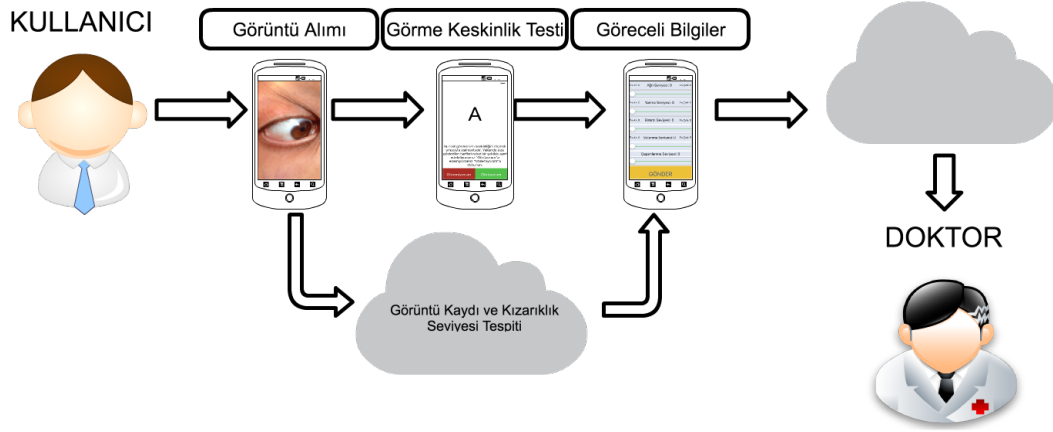
sahiptir. Doktor sisteme kendine ait hasta ekleyebilir. Hasta kullanıcı tipi ise kendi doktorunun kontrolü gerçekleştirilmesi için belirti ekleyebilir. Kullanıcı ekleme işlemi yapılırken bir takım bilgiler istenmektedir. Bu bilgiler çizelge 3.3'te verilmiştir.

Çizelge 3.3. Kullanıcı kayıt işlemi sırasında kullanılan bilgiler

Bilgi Adı	Girilebilirlik	Kullanıcı Tipi	Bilgi Adı	Girilebilirlik	Kullanıcı Tipi
ID	Girilemez	Hepsi	İlçe	Girilebilir	Hepsi
Kullanıcı Adı	Zorunlu	Hepsi	Mahalle	Girilebilir	Hepsi
Şifre	Zorunlu	Hepsi	Cadde	Girilebilir	Hepsi
Ad	Zorunlu	Hepsi	Tarif	Girilebilir	Hepsi
Soyad	Zorunlu	Hepsi	Posta Kodu	Girilebilir	Hepsi
Doktor ID	Zorunlu	Hasta	Telefon	Girilebilir	Hepsi
Boy	Zorunlu	Hasta	Fax	Girilebilir	Hepsi
İlaçlar	Girilebilir	Hasta	E-Posta	Girilebilir	Hepsi
il	Girilebilir	Hepsi	Web Adresi	Girilebilir	Hepsi

Hasta mahremiyetini koruyabilmek amacıyla doktorun kendisine ait olmayan bir hastanın bilgilerine ulaşması engellenmiştir. Bu amaçla hasta kullanıcı tipine hangi doktorun hastası olduğunu belirtmek amacıyla doktor ID bölümü eklenmiştir. Böylece doktorlar sadece kendi hastalarının belirtilerini görebileceklerdir.

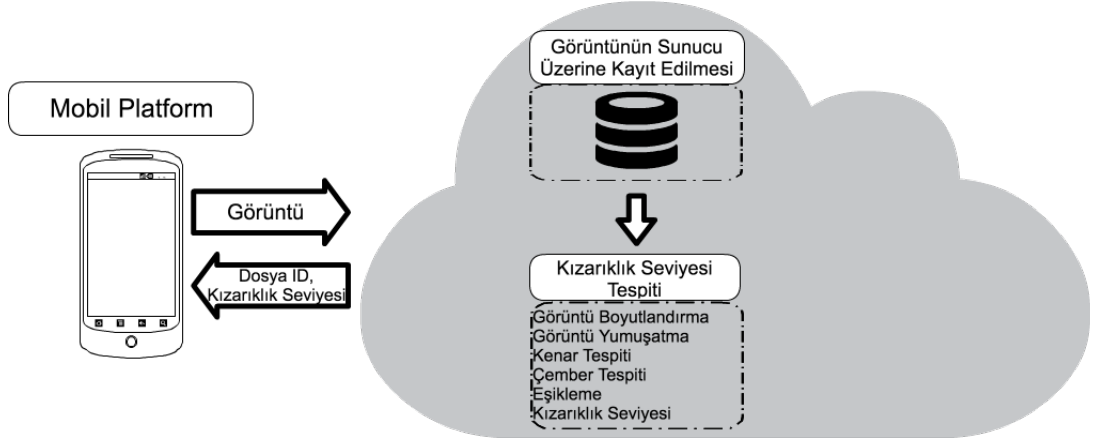
Doktor, hasta takip işlemine hastayı sisteme kayıt ettirerek başlar. Sonrasında hasta, doktorun istediği zamanlarda belirtileri ekleyerek doktorun program üzerinden takip edebilmesini sağlar. Bu işlemlerin genel tasviri Şekil 3.19'da verilmiştir.



Şekil 3.19. Takip programının işleyişi

Takip işlemi için hastanın yapması gereken 3 ana adım vardır. Bunlar sırasıyla görüntü alımı, görme keskinlik testinin yapılması ve göreceli bilgilerin alınmasıdır.

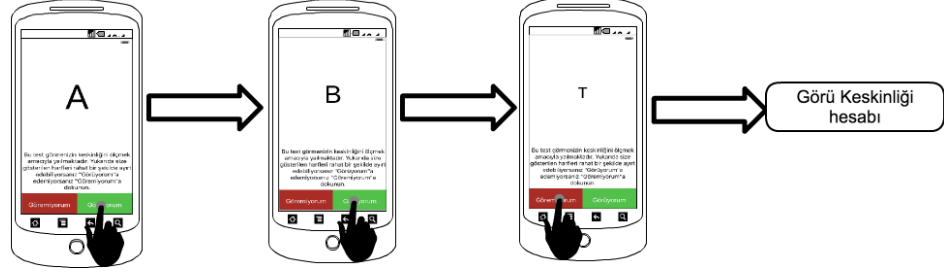
Görüntü alım işlemleri bölüm 3.2.2’de anlatıldığı gibi gerçekleştirilir. Görüntü alımı tamamlandıktan sonra kullanıcı görüntü içerisinde göz bulunduğunu onaylar. Onay alındıktan sonra program görüntüleri sunucu üzerine yükler. Sunucu ilk önce görüntünün kayıt işlemlerini gerçekleştirir. Daha sonra görüntüyü 3.4.3.1 de anlatılan algoritmaya göre Python dili OpenCV kütüphanesi yardımı ile işler. Java dili ile yazılmış web servisten Python dilinde yazılmış olan çalıştırılabilir dosya görüntünün yolu parametresi ile çalıştırılır. Görüntünün yolunu alan PY uzantılı, çalıştırılabilir dosya, görüntüyü işleyip kızarıklık seviyesinin tespit eder ve çıktı olarak dışarı verir. Servis bu çıktıyı yakalayarak kızarıklık seviyesini içe aktarmış olur. Kayıt edilen dosyaya ID ataması yapılır. Atanan ID ve kızarıklık seviyesi server tarafından JSON formatında istemciye geri döndürülür. Sunucu tarafında bu işlemler yapılıyorken program gerekli olan diğer bilgileri kullanıcıdan almaya devam eder (Şekil 3.20). Böylece zaman etkin bir şekilde kullanılmış olur.



Şekil 3.20. Kızarıklık tespiti algoritması

Görüntüler sunucuya gönderildikten sonra görüntülerin işlenmesi ile eş zamanlı olarak ikinci aşama olan görme keskinliği testine geçilmektedir. Test işlemleri kol mesafesinde tutulan mobil cihaz üzerinden, kullanıcıya harfler gösterilmesi ile gerçekleştirilir. Harflerin gösterilmesi ile beraber kullanıcıya harfi görüp göremediği sorulur. Eğer görüyor ise yeşil olan “görüyorum” butonuna göremiyor ise kırmızı olan “göremiyorum butonuna basması istenir. Görüyorum butonuna basan kullanıcılar için öncekinden küçük rastgele seçilmiş bir harf gösterilir. Program kullanıcıdan “göremiyorum” yanıtını alır ise gösterim işlemini sonlandırır (Şekil 3.21). Daha sonra keskinlik bilgisini hesaplar. Bu işlemlerin mobil cihazlar üzerinde ve farklı insanlar için kararlı bir şekilde geliştirilmesinin önünde bir takım engeller bulunmaktadır. Bu engeller; cihazların ekran boyutlarının birbirinden farklı olması, ekran parlaklıklarının ortam ışığının değişebilmesi ve insanların kol mesafelerinin farklılık göstermesidir. Ekran boyutlarının farklılık göstermesi probleminin üstesinden gelebilmek amacıyla tüm platformlar için ortak yazı fontu ve yazı boyutu kullanılmıştır. Parlaklık ve ışık değişimine önlem olarak programın bu bölümü açıldığı zaman ekran parlaklığı otomatik olarak son seviyeye getirmektedir. Bölümün çıkışında parlaklık eski haline getirilecektir. Son problem olan kol mesafesi problemi içinde doktor tarafından kaydedilmiş olan hastanın boyuna göre yazıların büyüklüğü değiştirilmiştir. Böylece ortam, cihaz ve kullanıcı değişiminden etkilenmeyen kararlı bir test elde edilmiştir.

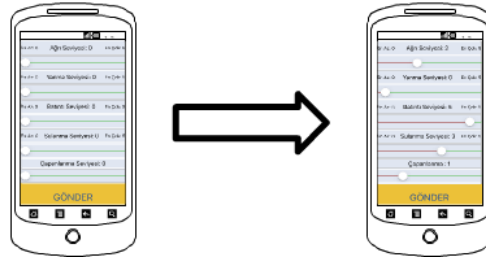
Görme Keskinlik Testi



Şekil 3.21. Görme keskinliği testi

Görme keskinliği testinin tamamlanması ile son aşama olan göreceli bilgilerin alınması aşamasına geçilmektedir. Bu aşamada hastadan gözünde meydana gelen doktorun da hastaya sorarak öğrendiği ağrı, yanma, batma, sulanma ve çapaklanma belirtilerini en az 0, en çok 5 değerini alan bir ölçek içerisinde sınıflandırması istenmiştir (Şekil 3.22). Kullanıcı gerekli bilgileri girdikten sonra gönder butonuna basarak belirtilerin toplanması işlemini sonlandırmaktadır.

Göreceli Bilgiler



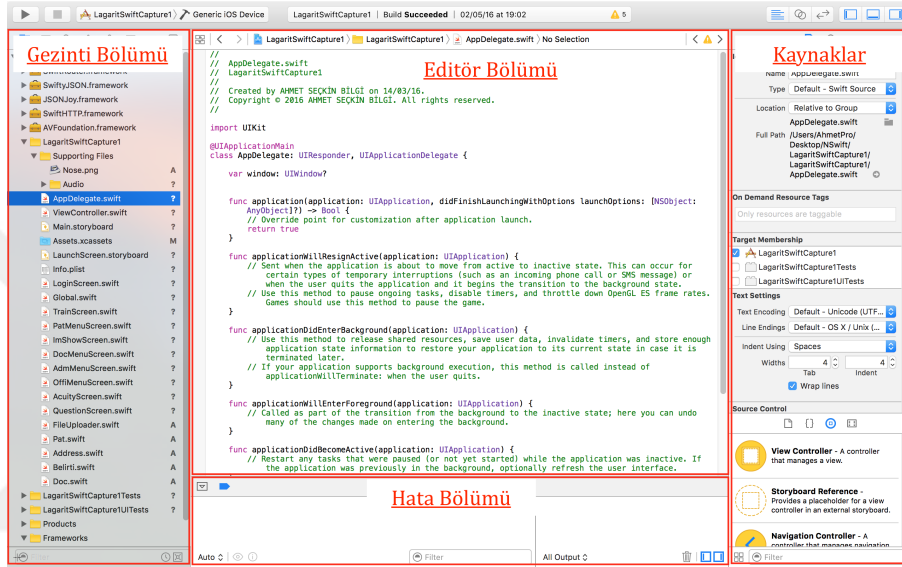
Şekil 3.22. Göreceli bilgilerin alınması

3.4.2. Yazılım

Göz hastalıkları takibi için geliştirilen mobil yazılım 2 farklı işletim sistemi için yazılmıştır. Bunlar Android ve IOS işletim sistemleridir.

3.4.2.1. IOS işletim sistemi için yazılım geliştirilmesi

IOS işletim sistemi 2007 yılında Apple tarafından duyurulmuştur. Arm mimarisine sahip işlemciler üzerinde çalışır. Apple firmasının ürettiği IPod, iPad ve iPhone cihazları için özelleştirilmiştir. Bu işletim sistemine yazılım geliştirmek için yine aynı firma tarafından geliştirilen XCode Tümleşik Geliştirme Ortamı (TGO) kullanılmaktadır (Şekil 3.22).



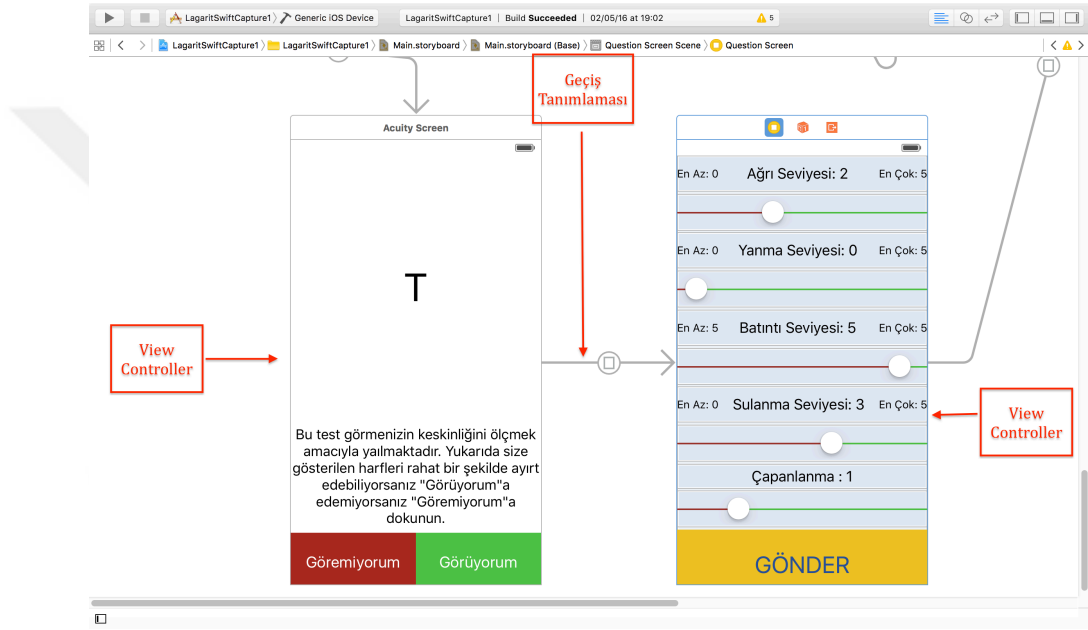
Şekil 3.23. Xcode TGO.

Apple IOS üzerine geliştirme yapmak için Objective-C veya Swift dillerini önermektedir. Programın yazımı esnasında daha sade bir söz dizimine sahip olan Swift programlama dili tercih edilmiştir (Çizelge 3.4).

Çizelge 3.4. Objective-C ve Swift söz dizimi örnekleri

Objective-C
(instancetype)initWithFrame:(CGRect)frame style:(UITableViewStyle)style;
Swift
init(frame: CGRect, style: UITableViewStyle) { /* ... */ }
Objective-C
UITableView *myTableView = [[UITableView alloc] initWithFrame:CGRectZero style:UITableViewStyleGrouped];
Swift
let myTableView: UITableView = UITableView(frame: CGRectZero, style: .Grouped)

Arayüz tasarımı “Controller” adı verilen ekranlar üzerinde yapılır. Ekranların üzerine istenilen bileşen eklenebilir. Ayrıca ekranlar ve ekran içinde bulunan bileşenler Swift veya Objective-C ile kontrol edilebilir. Ekranlar arası geçiş kontrolü için ise “Storyboard” adı verilen yapı kullanılmaktadır. StoryBoard içinde birden fazla ekran bulundurabilir. Aynı zamanda bu ekranlar arası geçişe olanak vererek olay akışını kontrol eder. Bunların arasında yapılacak geçişler ise StoryBoard içinde tanımlanır ve kod tarafında kullanılabilir (Şekil 3.24).



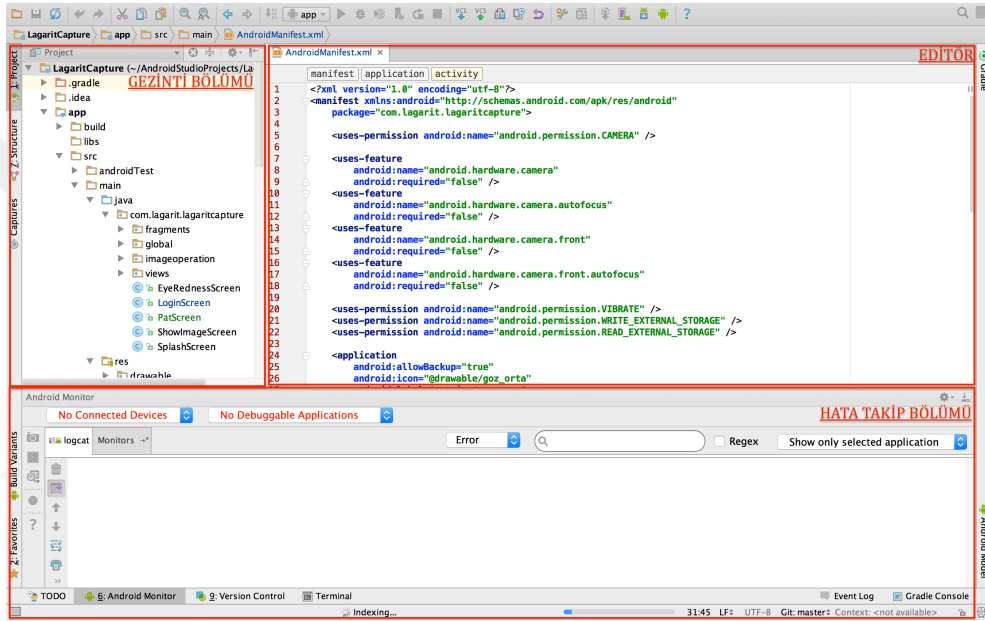
Şekil 3.24. Örnek StoryBoard.

3.4.2.2. Android işletim sistemi için yazılım geliştirilmesi

Android, Google firması tarafından geliştirilen Linux tabanlı işletim sistemidir. İlk kararlı sürümü 2008 yılında yayınlanmıştır. Mobil platformlar için özelleştirilmiş yapılara sahiptir.

Android işletim sistemine program geliştirebilmek için Google tarafından resmi olarak tanıtılmış iki programlama dili bulunmaktadır. Bunlar C++ ve Java programlama dilleridir. C++ hız gerektiren durumlarda tercih edilmektedir. Fakat geliştirme işlemleri genellikle Java programlama dili vasıtası ile yapılmaktadır.

Java ile program geliştirebilmek için Java Geliştirme Kiti ve Android'in özelliklerinin kullanılmasını sağlayan Android Yazılım Geliştirme Kiti kullanılması gerekmektedir. Ayrıca bir TGO'ya ihtiyaç duyulmaktadır. TGO olarak Google tarafından geliştirilen ve önerilen IntelliJ tabanlı Android Studio ve daha önce yine Google'ın geliştirdiği Eclipse tabanlı Android Bundle bulunmaktadır. Özellikle arayüz geliştirmede sunduğu avantajlar nedeniyle programın geliştirilmesi sırasında Android Studio tercih edilmiştir (Şekil 3.25).



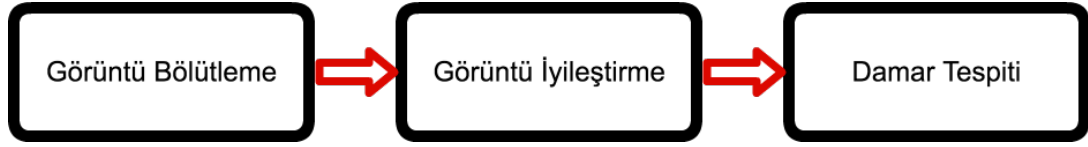
Şekil 3.25. Android Studio geliştirme ortamı.

3.4.3. Kabiliyetler

Program hastalık takibi için 3 şekilde belirti toplamaktadır. Kabiliyetler bölümü altında sırasıyla göz kızarıklığı, görme keskinliği testi göreceli bilgileri nasıl topladığı ve ilaç takip işleminin nasıl yapıldığı anlatılacaktır.

3.4.3.1. Göz kızarıklığı tespiti

Göz kızarıklığı pek çok hastalık için önemli bir belirtidir. Bunun için kızarıklık seviyesinin tespiti programın önemli adımlarından birisidir. Kızarıklık seviyesinin tespiti 3 ana adımdan oluşmaktadır (Şekil 3.26).



Şekil 3.26. Kızarıklık tespiti algoritma adımları

Göz akının bölütlenmesi işlemi bölüm 3.3.2 de anlatıldığı gibi çevrimiçi bölütleme prensibine göre yapılarak ilgili alana sahip (I_r) ulaşılmıştır. Fakat bölütme işlemi sırasında göz akı içerisinde bulunan parlama bölgeleri daha önce yapılan eşikleme tarafından tespit edilememektedir. Bu bölgeler beyaza yakın renge sahip olmakta ve parlama nedeniyle üzerinden hiçbir bilgi alınamamaktadır. Parlamalardan kurtulmak amacıyla görüntüye eşikleme işlemi uygulanmıştır (Denklem 3.39).

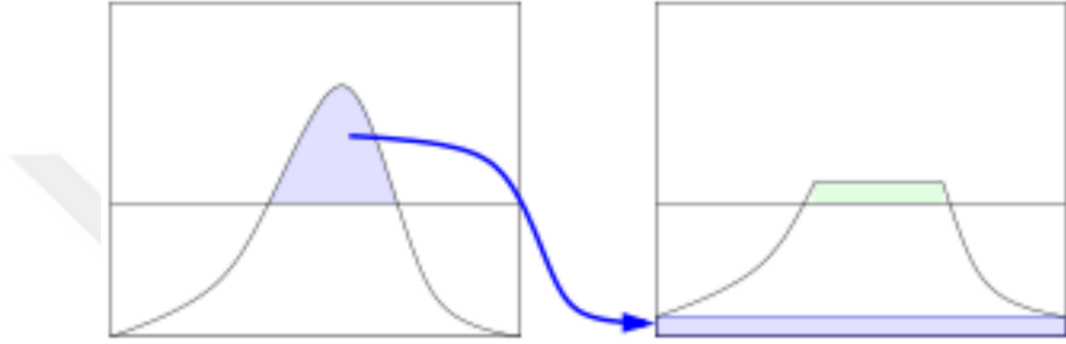
$$I_{rg} = I_r < T \quad (3.32)$$

Göz akı içindeki damarlar, kimi zaman bulanık görüntü, kimi zaman damarların boyutunun küçük olması sebebiyle net olarak görülememektedir. Görüntüdeki küçük detayların ortaya çıkarılması amacıyla histogram eşitleme algoritmalarından yararlanılmıştır.

3.4.3.1.1. Histogram eşitleme

Görüntü üzerinde histogram eşitleme, yoğunluk değerleri belirli bir bölgeye toplanmış görüntünün, yoğunluk değerlerinin yayılmasını sağlamaktadır. Bu işlem aynı zamanda bastırılmış bazı detayların ön plana çıkarılmasını da sağlar. Bu yüzden görüntü işlemede sıkça kullanılan yöntemler arasında yer alır. Fakat görüntünün tamamının üzerinde histogram eşitleme işleminin yapılması, detayların tam olarak ortaya çıkarılamaması gibi sorunlara neden olmaktadır. Bunu ortadan kaldırma amacıyla Adaptif Histogram Eşitleme (AHE) yöntemi ortaya atılmıştır. Bu yöntemde görüntü dikdörtgensel bölgelere ayrılarak bu bölgelerin birbirinden bağımsız olarak histogramlarının eşitlenmesi gerçekleştirilir. Bu sayede görüntü içindeki detaylar daha fazla ön plana

çıkarılır. AHE algoritmasının istenilmeyen özelliği fazla gürültü oluşturmastır. Bu problemi ortadan kaldırmak amacıyla Karşıtlık Sınırlamalı Histogram Eşitleme Yöntemi önerilmiştir. Bu yöntem AHE gibi görüntünün parçaları üzerinde işlem yaparken, kullanıcının verdiği sınırın üstüne çıkan bölgeleri sınırlandırarak, fazlalık olan kısmı tüm histogram kutucuklarına dağıtır (Şekil 3.27). Böylece hem detay yönünden zengin bir görüntü elde edilir, hem de gürültüler azaltılmış olur.



Şekil 3.27. KSAHE algoritması çalışma mantığı (Konyar ve Ertürk, 2015).

Kızarıklık seviyesi tespiti için detay ortaya çıkarması ve gürültüleri azaltması sebebiyle KSAHE algoritması tercih edilmiştir.

KSAHE'den geçirilerek küçük ve bulanık damalar ön plana çıkarılmıştır. Bu işlemden sonra damarların tespit edilmesi gerekmektedir. Damarlar göz akından farklı renk yoğunluk değerlerine sahiptir. Damar tespit işlemi bu fark üzerinden yapılacaktır. Görüntü işlemede yoğunluk farkını tespit edebilmek için bölüm 3.3.2.5 anlatılan değişim türevi tabanlı yaklaşımlar kullanılmaktadır. Değişim türevini ve eşiklemeyi birlikte kullanan Canny algoritması diğer değişim türevi yaklaşımlarına göre yüksek başarı elde etmektedir (Bölüm 3.3.2.5). Tez çalışmasında da Canny kullanılmıştır. Algoritma damar geniş değilse damarın kendisi, eğer geniş bir damar ise damarın sınırlarını bulmaktadır (I_v).

Kızarıklık, göz akı içerisindeki damarların genişleyip görünür hale gelmesi ile

oluşmaktadır. Bu bilgiden yola çıkılarak kızarıklık seviyesi tespiti denklem 2.33, 3.34 ve 3.35'de anlatıldığı gibi yapılmıştır.

$$A_s = \sum_{i=1}^m \sum_{j=0}^n c \begin{cases} c = 1, & I_{rg}(i, j) > 0 \\ c = 0, & I_{rg}(i, j) = 0 \end{cases} \quad (3.33)$$

$$A_v = \sum_{i=1}^m \sum_{j=0}^n c \begin{cases} c = 1, & I_v(i, j) > 0 \\ c = 0, & I_v(i, j) = 0 \end{cases} \quad (3.34)$$

$$S = \frac{A_v}{A_s} * 2 \quad (3.35)$$

Yukarıdaki denklemlerde m , n , A_s , A_v S sırasıyla görüntünün satır sayısını, sütun sayısını, görüntüdeki göz akı bölgesinin toplam büyüklüğünü (gözek sayısı), görüntüdeki damar bölgesinin büyüklüğünü (gözek sayısı) ve göz kızarıklık seviyesini (0-100) ifade etmektedir.

3.4.3.2. Görme keskinliği testi

Geleneksel yöntemde görme keskinliği testi eşeller (Şekil 3.28) üzerinden hastaya küçülen harf, rakam veya şekil gösterimi ile yapılmaktadır. Bu testin mobil ortamlar üzerinden geliştirilmesinde farklı ekran boyutları, farklı parlaklık seviyeleri ve mesafenin değişmesi gibi etmenler vardır. Bu etmenleri göz önünde bulundurarak kararlı bir program yazılması gerekmektedir.



Şekil 3.28. Görme keskinlik testi için kullanılan eşel örneği Plainis vd. (2012).

İlk çalışma çevrenin veya telefonun ışığının (ışık seviyesinin düşük olması) etkisinin azaltılması amacıyla yapılmıştır. Program otomatik olarak mobil cihazın ekranının ışık seviyesini en üst düzeye çekecektir. Bunun amacı çevredeki ışığın etkisini en aza indirmektir. İşlemler sırasında telefonun sabit uzaklıkta tutulması gerekmektedir. Bu uzaklık bir sonraki belirti toplama esnasında da aynı olmalıdır. Fakat kullanıcıya cm gibi bir ölçek cinsinden uzaklık vermek ve buna kullanıcının uymasını beklemek gerçekçi bir yaklaşım değildir. Bunun yerine kullanıcıdan cihazı kol mesafesinde tutması istenmiştir. Fakat insanların kol mesafeleri de değişebilmektedir. Bu sorunu çözmek için kayıt esnasında doktor tarafından girilen boy bilgisi kullanılmıştır. Boy da meydana gelen her 10cm'lik artış yazının fontunu 1 birim artırmıştır (Denklem 3.36).

$$f = \begin{cases} 101, & 0 < b < 150 \\ 102, & 150 \leq b < 160 \\ 103, & 160 \leq b < 170 \\ 104, & 170 \leq b < 180 \\ 105, & 180 \leq b < 190 \\ 106, & 190 \leq b \end{cases} \quad (3.36)$$

Denklem 3.36'te b kullanıcının boyunu, f ise yazının boyutunu ifade etmektedir. Kullanıcı görüyorum butonuna her dokunduğunda yazının boyutu 10 birim düşürülmekte ve rastgele olarak belirlenen yeni harf gösterilmektedir. Eğer kullanıcı göremiyorum butonuna basarsa veya tüm harfleri tanımlayabilirse denklem 3.37'te verilen şekilde görme keskinliği seviyesi (gs) hesaplanır.

$$gs = \left\| \frac{f}{10} \right\| \quad (3.37)$$

3.4.3.3. Göreceli bilgileri

Göreceli bilgiler doktorun hastaya sorarak veya bir kısmını gözlem yoluyla öğrenebildiği bilgilerdir. Bu bilgiler ağrı, yanma, sulanma, batma, ve çapaklanma olmak üzere 5 adet olarak belirlenmiştir. Bilgilerin hastadan toplanabilmesi amacıyla bir ekran düzenlenmiştir. Ekranda bulunan "slider" vasıtası ile hastalar en az 0, en çok 5 arasında değerler vermektedir. Alınan bu değerler doğrudan web servise yüklenerek doktorun kullanıcıyı takip etmesi beklenmektedir.

3.4.3.4. İlaç takip sistemi

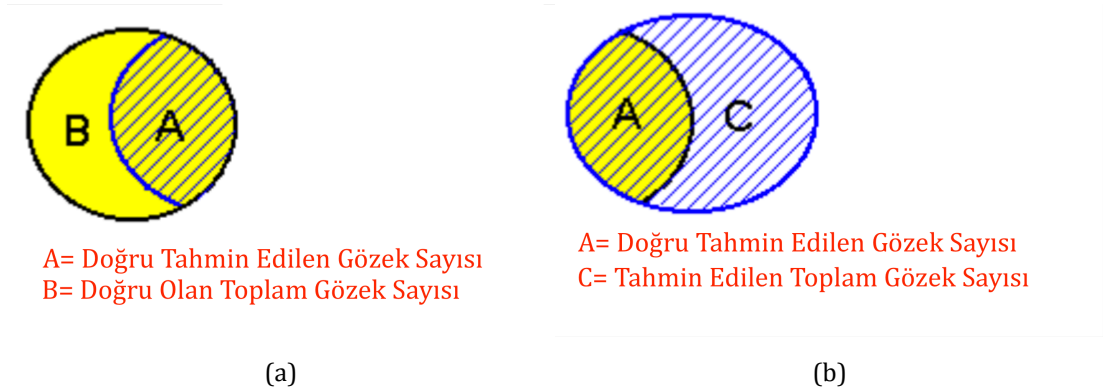
İlaç takip sistemi ilaçların zamanının hatırlatılması amacıyla oluşturulmuştur. Hasta kaydı yapılıyorken kullanacağı ilaçlar ve sıklık durumu (günde 1, 2 vb.) belirtilmektedir. Mobil platform üzerinde program ise sabah 9 akşam 11 arasında sıklık durumuna bölerek gerekli zamanlarda ilacın alınmasına dair sesli uyarılar vermektedir. Böylece ilacın düzgün bir şekilde kullanılması beklenmektedir.

3.5. Başarı Hesaplama

Önerilen tez çalışmasında iki bölümün başarı hesabı yapılmıştır. Bunlar çevrimdışı göz bölütlemesi ve göz kızarıklığı tespittir.

3.5.1. Göz akı bölütlemesi başarı hesabı

UBIRIS.v1 veri seti daha önce göz akı üzerinden insan tanımada kullanılmış bir veri setidir. Göz akı üzerinden insan tanımının adımlarından birisi olan bölütleme işlemi de veri seti üzerinde gerçekleştirilmiştir. Yayınlanan makaleler incelendiğinde bölütleme işleminin başarısından genelde öznel ifadeler ile bahsedilmiştir (Zhou vd. 2012). Fakat bu alanda yapılan çalışmaların ve yayınların artması ile nesnel değerlendirme işleminin gerekliliği hissedilmiştir. Bu amaçla ilki 2015 yılında “Biometrics: Theory, Applications and Systems” konferansının bir parçası olarak “Sclera Segmentation Benchmarking Competitions” etkinliği düzenlenmiştir. Etkinliğe kayıt olan gruplar algoritmalarını etkinliği düzenleyicilerine göndermektedir. Bu grup algoritmaları çalıştırarak kendi veri seti ve bu veri setinin göz akı maskeleri ile karşılaştırmaktadır. Karşılaştırma sonucunda her algoritmanın kesinlik (precision) ve anma (recall) hesabı yapılır. Kesinlik sistem tarafından tahmin edilen verilerin yüzde kaçının doğru olduğunu hesaplar. Anma ise doğru verilerin yüzde kaçının tahmin edildiğini hesaplar (Şekil 3.29, Denklem 3.38).



Şekil 3.29. Kesinlik ve anma hesabında kullanılacak kümelerin gösterimi (a, b) (Creighton University, 2016).

$$\text{Kesinlik} = \frac{A}{B}, \quad \text{Anma} = \frac{A}{C} \quad (3.38)$$

3.5.2. Göz kızarıklığı başarı hesabı

Göz kızarıklığı tespiti algoritmasının başarısını hesaplanması, Brienholden tarafından sağlanan görüntüler ile karşılaştırma yapılacaktır. Görüntüler göz akı dışında kısmi olarak iris, deri ve kirpik içermektedir (Şekil 1.1). Bu nedenle bölütleme yapılmasına gerek yoktur. Eşikleme ile göz akı dışındaki bölgeler elenerek sadece göz elde edilebilmektedir. Görüntülerin kızarıklık seviyesi hesaplanarak Brien Holden tarafından sağlanan dereceler ile tutarlılığına dikkat edilecektir.



4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu bölümde, bir önceki bölümde anlatılan çevrimdışı yaklaşım için göz akı bölütleme, göz kızarıklık seviyesi için kullanılan parametreler ve elde edilmiş sonuçlardan bahsedilecektir.

4.1. Çevrim Dışı Göz Akı Bölütleme

Çevrimdışı yaklaşım için geliştirilmiş olan algoritma DK tabanlı bir algoritmadır. DK algoritmasının doğru sonuçlar verebilmesi için başta SDÇ olmak üzere bazı parametrelerin doğru olarak girilmesi gereklidir. Otomatik, insandan destek almayan bir sistem kurmak için SDÇ'yi otomatik olarak tespit etmemiz gerekmektedir. Bu amaçla bir takım ön işlemler uygulanmıştır. Bunlardan ilki göz bebeği merkezinin tahminidir. Bu amaçla LOG filtre kullanılmıştır. Filtre 50x50 boyutunda ve 10 değışintiye sahip olarak oluşturulmuştur. Filtreleme sonucu tespit edilen nokta göz bebeği içerisindeki parlamanın bulunduğu noktadır. Bu nokta genelde göz bebeğinin üst bölgesinde bulunmaktadır. Göz bebeğinin merkezini tam olarak tespit edebilmek amacıyla, nokta %10 aşağıya kaydırılmıştır. Böylece göz bebeği merkezinin tam olarak elde edildiği varsayılmıştır. Kullanılan değerler deneysel olarak tespit edilmiştir.

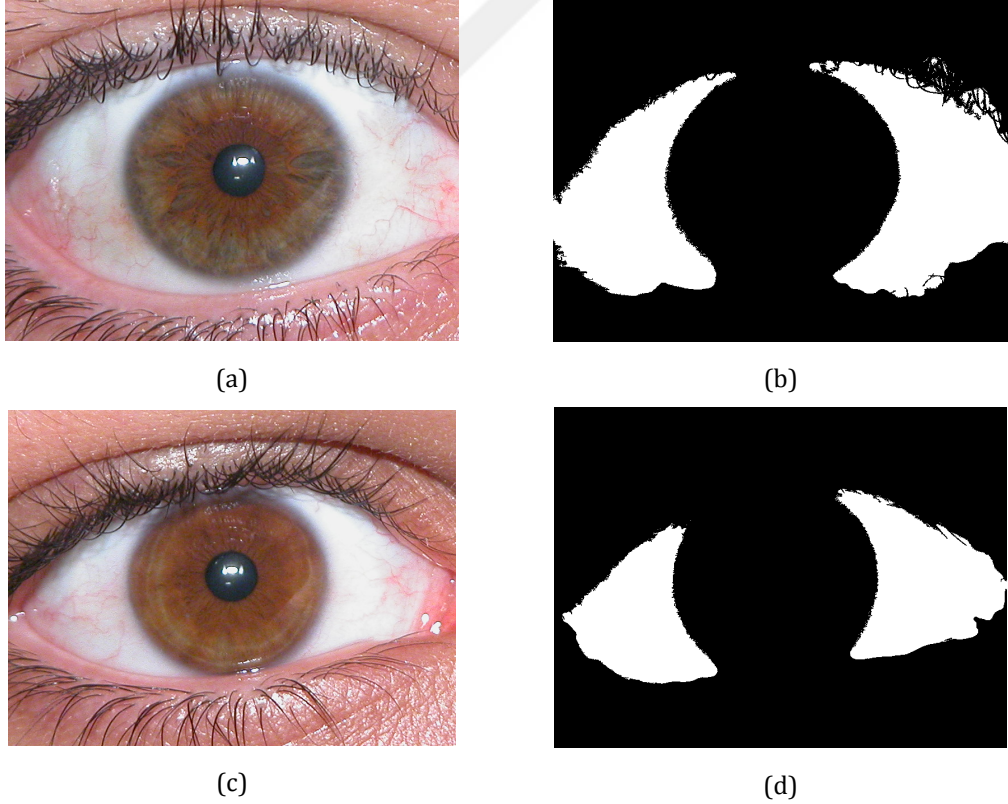
Göz bebeği merkezinin tespit edilmesinden sonra iris çapı tespit edilmeye çalışılmıştır. İris, göz akından ve deriden farklı renkte bir yapıya sahiptir. İrise ait gözeklerin kırmızı bantta aldığı değerler, göz akı ve deri ile karşılaştırıldığı zaman daha düşüktür. Bu özellikten faydalanılarak irisin yarı çapı tespit edilmiştir. Göz bebeğinden $-x$, $+x$ yönlerinde x eksenini ile 30 derece yapacak şekilde ilerlenmiştir. Her iki yön için de, yoğunluk değerlerinin 175 eşliğini geçtiği noktanın merkeze olan uzaklığı tespit edilmiştir. İki yönden gelen uzaklık verilerinin ortalaması alınarak irisin yarı çapı tespit edilmiştir. Kullanılan 30 derece parametresi gözün yapısı nedeniyle tercih edilmiştir. Göz kapakları dikkate alındığı zaman üst göz kapağının alta göre daha fazla kapandığı fark edilmiştir. Kapalı göz durumlarına dayanıklı bir algoritması

nedeniyle 30 derece olmasına karar verilmiştir. Eşik değer olan 175'e ise deneysel olarak karar verilmiştir.

SDÇ'nin doğru sonuç üretmesi için göz akı içerisine yerleştirilmesi gerekmektedir. Bu amaçla için göz bebeğinin orta noktasından iris çapının 0.65 katı kadar $-x$ ve $+x$ yönünde ilerlenmiştir. İrisin renk yoğunluk değerleri göz akından çok farklıdır. Bu yüzden iris-göz akı sınırındaki türev değerleri yüksek çıkmaktadır. Bu da KTDK'nın buraya ilerlemesini zorlaştırmaktadır. Katsayı yayın göz bebeğine yakın olması hatalı bir durum oluşturmayacağı için 0.65 olarak seçilmiştir. Göz bebeğini merkez kabul ederek $+y$, $-y$ yönünde toplam 300 birim uzunluğunda bir yay çizilmiştir. Çizilen yay üzerindeki 1. türev değerlerinin oluşturmuş olduğu vektör içerisinde, 1 eşiğinin altındaki parçalar tespit edilmiştir. Burada kullanılan 1 eşik değeri deneysel olarak elde edilmiştir. Gözün kapalılık durumuna göre göz akı gibi davranan, türev değerleri 1'in altında olan deri bölgeleri olabilmektedir. Bu bölgelerden kurtulmak amacıyla; eşikleme ardından geriye kalan bölgelerin, merkez noktası hesaplanmış ve bu noktaların göz bebeğine olan dikey uzunluğuna bakılmıştır. En yakın bölge göz akı olarak var sayılmış ve SDÇ'nin buraya yerleştirilmesine karar verilmiştir. Yerleştirilecek olan SDÇ'nin boyutu hızlı sonuç elde edebilmek için küçük olmamalıdır. Çok büyük SDÇ yerleştirilmesi ise SDÇ'nin göz akı dışındaki bölgelere temas etmesine veya çok yaklaşarak DK'nın göz akı dışarısına taşmasına neden olmaktadır. Hataya götürebilecek olan durumlardan kurtulmak için, elde edilen göz akı gölgesinin boyutuna bağlı bir yaklaşım sergilenmiştir. Göz içerisine yerleştirilecek olan SDÇ'nin boyutu denklem 3.5, 3.6 ve 3.7'de ifade edilen şekilde yapılmıştır. Başlangıç noktası varsayılan göz akı bölgesinin üst sınırına yakın olarak tespit edilmiştir. Bunun nedeni gözün yapısıdır. Göz akının üst bölümü kaş ve kirpik gibi renk olarak göz akına benzemeyen dokular vardır. Fakat alt bölümde, göz akından hemen sonra deri ve seyrek kirpikler başlar (Şekil 3.6). Alt bölgede türev değerleri düşük olduğu için KTDK üst bölgeye nazaran alt bölge sınırlarını daha kolay aşmaktadır. Bu nedenden dolayı SDÇ bölgenin üst sınırına yakın yerleştirilmiştir.

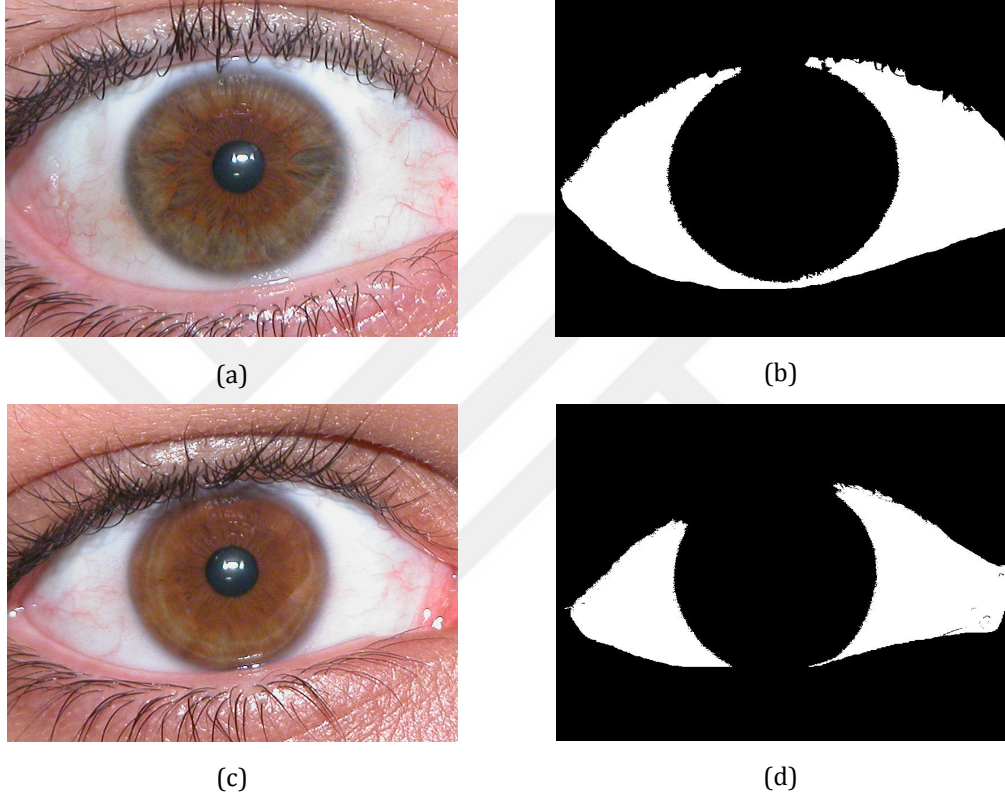
KTDK'nın ilerleme hızı parametre olarak sisteme verilebilmektedir. Geliştirilen programda bu parametre tüm veri seti için sabit 3.25 olarak kullanılmıştır.

KTDK'nın çalışması dögüsel bir işlemdir. Bunun için dögü sayısı verilmelidir. Elde edilen göz akı bölgelerinin boyutlarının farklı olması nedeniyle dögü miktarının da farklı olması gerekmektedir. Dögü miktarının belirlenmesi için göz akı bölgesinin büyüklüğüne bağlı olarak gerçekleştirilmiştir. Varsayılan göz akı bölgesinin büyüklüğünün 2 katı dögü miktarı olarak tespit edilmiştir. Fakat gözün sağ ve sol tarafından elde edilen bölgelerin büyüklüğü farklı olmaktadır. İki tarafa da aynı dögü miktarının uygulanması, göz akının sadece bir kısmının tahmin edilmesine veya göz akının dışına çıkılarak göz akı dışındaki dokuların dahil edilmesine neden olmaktadır. Bu nedenden dolayı KTDK gözün sağ ve sol tarafları için ayrı dögü miktarlarına göre 2 defa çalıştırılmıştır. Sonrasında elde edilen 2 bölge birleştirilerek 175 eşiğinin altında bulunan noktalar çıkarılmıştır (Şekil 4.1). Eşik deneysel olarak tespit edilmiştir.



Şekil 4.1. Sırasıyla birinci örnek ve sistem tarafından çıkartılmış maske (a, b), on dördüncü örnek ve sistem tarafından çıkartılan maske (c, d).

Sistem tarafından maskeler çıkartıldıktan sonra başarı hesaplaması yapılacaktır. UBIRIS.v1 veri seti göz akından insan tanımada, dolayısıyla göz akı bölütlemesinde kullanılan bir veri seti olmasına rağmen, literatürde göz akı için doğruluk bilgisi içeren bir veriye rastlanamamıştır. Bunun için “Süleyman Demirel Üniversitesi Bilgisayar Mühendisliği” bünyesinde bulunan “VIPLab” grubu tarafından el ile göz akına sahip bölgelerin maskesi çıkartılmıştır.

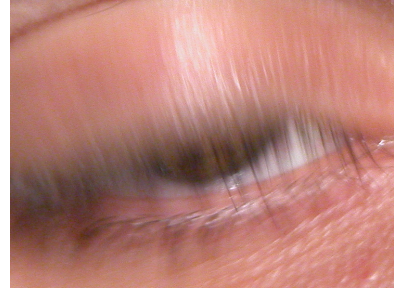


Şekil 4.2. Sırasıyla birinci örnek ve el ile çıkartılmış maske (a, b), sırasıyla on dördüncü örnek ve el ile çıkartılan maske (c, d).

Başarı, sistem tarafından elde edilen maskelerin, doğruluk maskeleri ile karşılaştırılarak kesinlik ve anma sonuçlarının elde edilmesi ile hesaplanmıştır. Veri seti içerisinde bazı görüntülerde gözler aşırı kapanmış durumdadır (Şekil 4.3). Bu görüntüler bölütleme için uygun olmadıklarından dolayı başarı hesaplamasından çıkarılmıştır. Çıkarma işlemi doğruluk maskelerinde bulunan göz akına ait olduğu kesin olarak bilinen toplam gözek sayısına göre yapılmıştır. Toplam gözek sayısı 30000'nin altında ise yeterli göz akına sahip olmadığı düşünülerek 1035 görüntü içerisinde toplam 67 tanesi çıkartılmıştır.



(c)



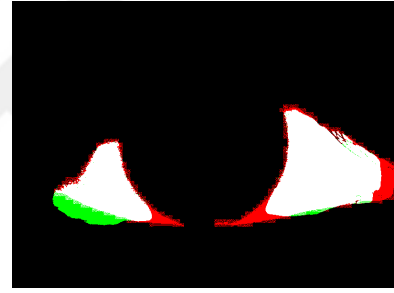
(d)

Şekil 4.3. Başarı hesabından çıkartılan görüntüler (a, b)

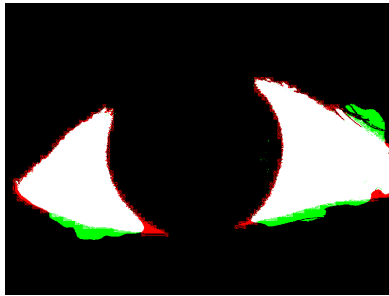
Oluşturulan otomatik göz akı bölütleme sisteminin %82.49 anma ve %85.84 kesinlik değerine sahip olduğu hesaplanmıştır. Sistem tarafından oluşturulan maskeler ile doğruluk maskelerinin bazılarının karşılaştırması Şekil 4.4'te verilmiştir.



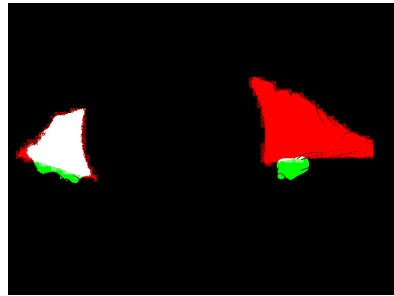
(a)



(b)



(c)



(d)

Şekil 4.4. Sistem tarafından hazırlanan maskelerin doğruluk maskeleri ile karşılaştırılması. Beyaz renk sistem tarafından doğru tahmin edilen bölgeyi, kırmızı renk göz akı ama sistem tarafından tespit edilemeyen bölgeyi, yeşil renk ise göz akı olmayan ama sistemin göz akı olarak varsaydığı bölgeyi göstermektedir (a, b, c, d). Sırasıyla, görüntüler için kesinlik değerleri %83.36, %91.09, %90.72, %77.08; anma değerleri ise %89.99, %82,18, %92.28, %29.96 şeklindedir.

Şekil 4.4 (d) görüntüsünde SDÇ gözün alt sınırında göz kapağına yerleştirildiği için bu görüntüde anma değeri düşük çıkmıştır.

4.2. Göz Kızarıklığı Tespiti

Göz kızarıklığı tespiti için önce çevrimiçi göz akı tespit algoritması daha sonra ise kızarıklık tespit algoritması çalıştırılmıştır. Göz akı tespit algoritmasında ilk önce görüntü boyutlandırma işlemi yapılmıştır. Farklı boyutlarda olan görüntülerin hepsi 604x480 boyutuna indirilmiştir. Daha sonra Canny kenar tespit algoritmasından geçirilmiştir. Burada Canny'nin alt eşik değeri 20 olarak belirlenmiştir. Kenarları tespit edilmiş görüntü üzerinde Hough'ın çember tespit algoritması çalıştırılmıştır. Algoritma için yarı çap değeri en az 10, en çok 50 olarak belirlenmiştir. Tüm bu parametreler deneysel olarak tespit edilmiştir. Tespit edilen çemberin göz bebeği veya iris olduğu varsayılarak ilgili bölge kesme işlemi gerçekleştirilmiştir. Çemberin merkezinden q_g, q_u değişkenleri kadar uzaklaşarak bir dikdörtgen çıkarılmıştır. Değişkenlerin hesaplanması denklem 3.29'da anlatıldığı şekilde yapılmaktadır. Bu denklemde kullanılan S_1, S_2 değerleri sırasıyla 60 ve 200 olarak kullanılmıştır. Dikdörtgen ilgili bölge elde edildikten sonra eşikleme işlemi yapılır. Bu işlemde kullanılan T eşiği ise 50 olarak belirlenmiştir. Eşiklemeden sonra tespit edilen bölgelerin en büyüğü nihai göz akı olarak varsayılmış bu parça üzerinde göz kızarıklığı tespit algoritması çalıştırılmıştır.

Göz kızarıklığı tespit algoritmasında ilk olarak elde edilen göz akı bölgesindeki damarların görünür hale getirilmesi amacıyla KSAHE algoritması kullanılmıştır. Algoritma da kırpma limiti ve parça genişliği sırasıyla 5 ve 4x4 olarak seçilmiştir. Damar tespiti için kullanılacak olan Canny algoritmasının alt limiti olarak 24 seçilmiştir.

4.2.1. Brien Holden ve algoritma karşılaştırması

Brien Holden tarafından sağlanan görüntüler kızarıklık tespit algoritmasına verilerek kızarıklık seviyesi tespit edilmiştir (Şekil 1.1). Kullanılan görüntüler sadece göz akını içerdiği için ilgili bölge tespitine gerek kalmadan kızarıklık tespit algoritması çalıştırılmıştır. Çalışma sonucu elde edilen veriler çizelge 4.1'te verilmiştir.

Çizelge 4.1. Brien Holden ölçütü ve algoritma çıktısı karşılaştırması.

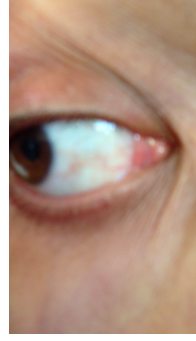
Brien Holden seviyesi	Brien Holden 1-100	Algoritma Çıktısı	Genişletilmiş Çıktı (2)
1	0-25	15,57	31,14
2	26-50	18,22	36,44
3	51-75	33,47	66,94
4	76-100	47,52	95,04

Brien Holden derecelendirmeyi 1-4 arası yaparken, sunulan algoritma derecelendirme işlemini 0-50 arası yapmaktadır. Aradaki uyumsuzluğu ortadan kaldırmak amacıyla Brien Holden ölçütünün derecelendirmesi 0-100 arasına çekilmiştir. Program çıktıları da 2 ile genişletilerek 0-100 arasına çekilmiştir. Brien Holden ölçütü ve program arasındaki uyum sağlandıktan sonra görülmektedir ki program 4 derecenin sadece birinde yapmıştır. Bu hatanın nedeni, görüntü boyutunun düşük olması olarak tespit edilmiştir. Boyut küçüklüğü sebebiyle gözetler arası geçiş sert olmakta ve bu durum kenar tespit eden algoritma olan Canny tarafından kenar olarak algılanmaktadır.

4.2.2. SDUeye veri seti

SDUeye veri seti içerisindeki görüntüler üzerinde bölütleme ve kızarıklık seviyesi tespit algoritması çalıştırılmıştır. Bölütleme aşamasında 64 görüntü içerisinde 2 tanesinde çember tespit edilememesi 6 tanesinde de çemberin iris dışında, kaş gibi noktalara konumlandırılması nedeniyle sistem tarafından reddedilmiştir (Şekil 4.5). Sistem ret işlemini eşikleme işleminden sonra kalan toplam gözet sayısı göre yapmaktadır. Gözet sayısı 7000'in altında ise yeterli

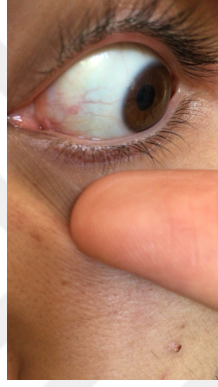
göz akı yoktur diyerek görüntüyü reddetmektedir. Bunun nedeni ise çemberin iris dışarısına yerleştirilmesidir.



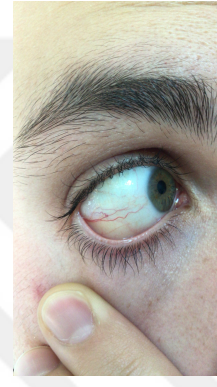
(a)



(b)



(c)



(d)



(e)

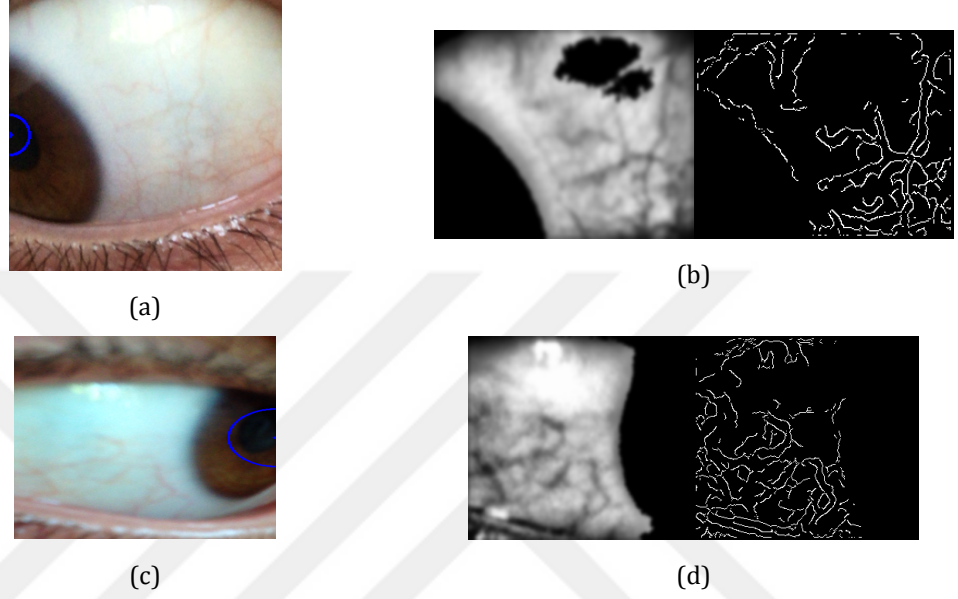


(f)

Şekil 4.5. SDUeye veri setinden görüntü örnekleri (a, b, c, d). Görüntünün iris içermemesi sebebiyle çember tespiti gerçekleştirilememiştir (e). Hough çember tespit algoritması çemberi kaş üzerine yerleştirmiştir (f). Bu sebeple sistem göz akı tespit edememiştir.

Sistem tarafından değerlendirilen görüntülerin bir kısmı oftalmolog tarafından da değerlendirilmiştir. Değerlendirme sonunda tutarsız bazı sonuçlar olduğu göze çarpmıştır. Bu durumun nedeni olarak, göz akı içerisinde meydana gelen

parlamalar saptanmıştır. Parlamalar bazen göz akını kapatarak, göz akı bölgesinin küçük olarak tespit edilmesine, bazen ise damarların üzerini örterek damar sayısının düşük olarak tespit edilmesine neden olmuştur. Parlamanın olmadığı veya az olduğu zamanlarda ise sistemin tahmini ve oftalmolog tarafından belirlenmiş dereceler birbirini tutmaktadır (Şekil 4.6).



Şekil 4.6. İlgili bölge çıkarımı yapılmış görüntü (a, c) ve sırasıyla bu görüntülerin KSAHE algoritmasından ve kenar tespit algoritmasından geçirilmiş hali (b, d).

Parlama olan görüntülerde oftalmologlar görüntü üzerindeki damarların çapına göre yorum yapabilmektedir. Fakat bu işlem sistem tarafından gerçekleştirilememektedir. Bu sebeple bazı hatalar oluşmuştur. Böyle durumlarda doktoru yanıltmamak amacıyla kızarıklık seviyesinin yanında doktora görüntülerin de gösterilmesine karar verilmiştir. Böylece doktor hastasını çok daha güvenli bir şekilde takip edebilecektir.

5. SONUÇLAR VE ÖNERİLER

Sunulan tez çalışmasında göz hastalıkları için uzaktan takip işlemine olanak veren mobil program geliştirilmiştir. Program doktorların muayene esnasında kullandığı 3 belirti tipi olan görme anormallikleri, görünüm anormallikleri, ağrı ve rahatsızlık belirti tiplerinin hepsini kapsayarak doktora mümkün olduğunca fazla veri iletilmesini sağlamıştır. Böylece hastanın takibi, uzaktan güvenilir bir şekilde sosyoekonomik sorunları ortadan kaldırarak yapılabilecektir. Ayrıca oluşturulan ilaç takip sistemi ile hastanın ilaca uyumluluğu artırılarak tedavinin daha düzgün bir şekilde ilerlemesi gerçekleştirilebilecektir.

Tez çalışmasında göz akı bölütleme işlemlerinin çevrimdışı ayağı UBIRIS.v1 veri seti üzerinde yapılmıştır. Bu veri seti göz akından insan tespiti için kullanılan bir veri seti olmasına rağmen, literatürde göz akı bölütleme işlemleri için doğruluk bilgisi içeren veriye rastlanmamıştır. Doğruluk bilgisi içeren veriler el ile oluşturulup literatüre kazandırılmıştır. UBIRIS.v1 veri seti için bölütleme başarısı %82.49 anma ve %85.84 kesinlik olarak hesaplanmıştır. Çevrimiçi bölütleme işlemlerinde kullanılan SDUeye veri seti oluşturularak literatüre kazandırılmıştır. Bunun yanında göz kızarıklığı tespiti için ilk defa mobil verilerden yararlanılarak bir yaklaşım sergilenmiştir. Kontrolsüz çevre koşulları altında alınan görüntülerin bazılarında parlama olması sorun çıkarsa da bu tür bir çalışma yapılmasının mümkün olduğu gösterilmiştir.

KAYNAKLAR

- Abelson MB, Lane K, Maffei C. 2010. Code red: The key features of hyperemia. Rev Ophthalmol. 17(4):92-94.
- Airstrip, 2016. Erişim Tarihi: 29.05.2016. <http://www.airstrip.com/>
- Akkaya, Z. Y., Örnek, F. 2015. Ankara Eğitim ve Araştırma Hastanesi Göz Hastalıkları Polikliniği'nin Hizmet Haritası. Türkiye Klinikleri Journal of Medical Ethics-Law and History, 23(1), 1-5.
- Amparo, F., Wang, H., Emami-Naeini, P., Karimian, P., & Dana, R. 2013. The Ocular Redness Index: A Novel Automated Method for Measuring Ocular Injection A Novel Automated System to Measure Redness. Investigative ophthalmology & visual science, 54(7), 4821-4826.
- Berta, A. (Ed.), 2007. Kırmızı Göz Ayırıcı Tanı Tedavi. Alcon Laboratories, İstanbul.
- BrienHoldenVission, 2011. Erişim Tarihi: 29.05.2016. <https://academy.brienholdenvision.org/browse/resources/courses/grading-scales>
- Bosch, 2011. Erişim Tarihi: 29.05.2016. <http://www.bosch-press.com/tbwebdb/bosch-usa/en-US/Presstext.cfm?id=598>
- Budak, B. A., Toka, F., Gelişken, Ö., Baykara, M., 2013. Göz Polikliniğine Başvuran ve İlaç Reçetesi Verilmiş Hastaların Tedavi Uyumlarının İncelenmesi. Uludağ Üniversitesi Tıp Fakültesi Dergisi, 6(14), 11-13.
- Creighton University, 2016. Erişim Tarihi: 29.05.2016. https://www.creighton.edu/fileadmin/user/HSL/docs/ref/Searching_-_Recall_Precision.pdf
- Dizdaroğlu ,B., Kengil, A., Balkıza, T., 2014. Kenar Tabanlı Düzey Kümeleriyle Tıbbi İmge Bölütleme İçin Mobil Uygulama. Tıp Teknolojileri Ulusal Kongresi, 25-27 Eylül, Kapadokya, 31-34.
- Dunsmuir, D. T., Payne, B. A., Cloete, G., Petersen, C. L., Görges, M., Lim, J., Ansermino, J. M., 2014. Development of mHealth applications for pre-eclampsia triage. IEEE journal of biomedical and health informatics, 18(6), 1857-1864.
- Frumkin, Howard (ed.), 2010. Environmental health: from global to local. John Wiley & Sons, 1221s San Francisco.
- Garg, A. X., Adhikari, N. K., McDonald, H., Rosas-Arellano, M. P., Devereaux, P. J., Beyene, J., ... & Haynes, R. B., 2005. Effects of computerized clinical

decision support systems on practitioner performance and patient outcomes: a systematic review. *Jama*, 293(10), 1223-1238.

Georga, E., Protopappas, V., Guillen, A., Fico, G., Ardigo, D., Arredondo, M. T., Fotiadis, D. I., 2009. Data mining for blood glucose prediction and knowledge discovery in diabetic patients: The METABO diabetes modeling and management system. Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 3-6 Eylül, Minneapolis, Amerika 5633-5636.

Gonzalez, R.C., Woods, R.E., 2008. Digital Image Processing. Pearson Education, 954s, New Jersey.

Gröne, O., Garcia-Barbero, M. 2001. Integrated care. *International journal of integrated care*, 1(2), 1-10.

Günay, M., 1980. Tıbbi Görüntülerin Bölütlenmesi Ve Karar Destek Sistemi İçin Mobil İstemci Uygulama Geliştirme. Kahramanmaraş Sütçü İmam Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, 118s, Kahramanmaraş.

Hough, P. V., 1962. Method and means for recognizing complex patterns. No. US 3069654, 6.

Kawamoto, K., Houlihan, C. A., Balas, E. A., & Lobach, D. F. 2005. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. *Bmj*, 330(7494), 765-772.

Khalifa, M. 2014. Clinical decision support: strategies for success. *Procedia Computer Science*, 37, 422-427.

Konyar, M. Z., Ertürk, S., 2015. Ultrason Görüntülerinin Çift Yönlü Süzgeç ve Rayleigh KLAHE ile Pekiştirilmesi. *Sinyal İşleme Ve İletişim Uygulamaları Kurultayı*, 16-19 Mayıs, Malatya, 1861-1864.

Peterson, R. C., Wolffsohn, J. S. 2009. Objective grading of the anterior eye. *Optometry & Vision Science*, 86(3), 273-278.

Li, C., Xu, C., Gui, C., Fox, M.D., 2010. Distance regularized level set evolution and its application to image segmentation. *Image Processing, IEEE Transactions on*, 19(12), 3243-3254.

Little, J. D. C., 1970. Models and Managers: The Concept of a Decision Calculus. *Management Science*, 16, B466-485.

OpenCV, 2016. Erişim Tarihi: 29.05.2016. <http://opencv.org/about.html>

- Plainis, S., Moschandreas, J., Giannakopoulou, T., Vitanova, V., Nikolitsa, P., Rozema, J. J., Pallikaris, I. G. 2013. Validation of a modified ETDRS chart for European-wide use in populations that use the Cyrillic, Latin or Greek alphabet. *Journal of optometry*, 6(1), 18-24.
- Proença, H., Alexandre, L.A., 2005. UBIRIS: A noisy iris image database, *International Conference on Image Analysis and Processing*, 6-8 Eylül, Cagliari, İtalya, 970-977.
- Riordan-Eva, P., Witcher, J.P. (Ed.), 2010. Vaughan ve Asbury Genel Oftalmoloji. Çev. Akova, Y.A. (Ed.). Güneş Tıp Kitapları, 461s, Ankara.
- Rodriguez, J. D., Johnston, P. R., Ousler, G. W., Smith, L. M., & Abelson, M. B. 2013. Automated grading system for evaluation of ocular redness associated with dry eye. *Clinical Ophtalmology*, 2013(7), 1197-1204.
- Statisticia, 2016. Erişim Tarihi: 29.05.2016. <http://www.statista.com/statistics/266136/global-market-share-held-by-smartphone-operating-systems/>
- WellDoc, 2016. Erişim Tarihi: 29.05.2016. <http://www.welldoc.com/>
- Wikipedia, 2016. Erişim Tarihi: 29.05.2016. <https://tr.wikipedia.org/wiki/G%C3%B6z>
- Xu, B., Xu, L., Cai, H., Jiang, L., Luo, Y., Gu, Y. 2015. The design of an m-Health monitoring system based on a cloud computing platform. *Enterprise Information Systems*, 1-20.
- Zhao, W. J., Duan, F., Li, Z. T., Yang, H. J., Huang, Q., & Wu, K. L. 2014. Evaluation of regional bulbar redness using an image-based objective method. *International journal of ophthalmology*, 7(1), 71-76.
- Zhou, Z., Du, E.Y., Thomas, N.L., Delp, E.J., 2012. A new human identification method: sclera recognition. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 42(3), 571-583.

ÖZGEÇMİŞ

Adı Soyadı : Ahmet Seçkin BİLGİ

Doğum Yeri, Yılı : Isparta, 1991

Medeni Hali : Bekar

Yabancı Dili : İngilizce

E-posta : a.seckin.bilgi@gmail.com



Eğitim Durumu

Lisans : Süleyman Demirel Üniversitesi, Mühendislik Fakültesi,
Bilgisayar Mühendisliği

Mesleki Deneyim

Nikel Ar-Ge	Şubat 2014-Ağustos 2014
Tübitak 1003	Nisan 2014-Mayıs 2016
Lagarit Yazılım	Haziran 2015-Halen

Yayınlar

Saygılı, A. S. Bilgi, H. Kalkan, E. A. Hendriks, Güven-Tabanlı Uyarlanırlı Doğrusal Tarama Kullanarak Stereo Karşılaştırma (245), G. Proceedings of IEEE SIU 2014, Trabzon.

A. S. Bilgi, E. Durmuş, H. Kalkan, G. Ortaç, K. Taşdemir, "Hiperspektral görüntü analizi ile bozulmuş incirlerin otomatik olarak tespiti", IEEE Signal Processing and Communications Applications Conference (SIU), 18-21 May 2015, Malatya, 2015

A. Güneş, E. Durmuş, H. Kalkan, A. S. Bilgi, Fusing the RGB Channels of Images for Maximizing the Between Class Distance, Proceedings of ICMV 2014, Milano.

E. Durmuş, A. S. Bilgi, G. Ortaç, H. Kalkan, K. Taşdemir, "Detection of black mold infected figs by using transmittance spectroscopy", Proceedings of WHISPERS 2015, 2-5 June, Tokyo, Japan, 2015

G. Ortaç, K. Taşdemir, A. S. Bilgi, E. Durmuş, H. Kalkan, "A hyperspectral imaging system for detection of dried figs with black mold", Proceedings of WHISPERS 2015, 2-5 June, Tokyo, Japan, 2015

G. Ortaç, A. S. Bilgi, Y.E. Görgülü, A. Güneş, H. Kalkan, K. Taşdemir, Classification of Black Mold Contaminated Figs by Hyperspectral Imaging, 2015 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), At Abu Dhabi, UAE