

KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

EL ŞEKİL, HAREKET VE POZİSYON BİLGİLERİ İLE SANAL MOUSE
KONTROLÜ

YÜKSEK LİSANS TEZİ

Bilgisayar Müh. Ramazan Özgür DOĞAN

ARALIK 2014

TRABZON

KARADENİZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**EL ŞEKİL, HAREKET VE POZİSYON BİLGİLERİ İLE SANAL MOUSE
KONTROLÜ**

Bilgisayar Müh. Ramazan Özgür DOĞAN

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"BİLGİSAYAR YÜKSEK MÜHENDİSİ"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

Tezin Enstitüye Verildiği Tarih : 05.12.2014

Tezin Savunma Tarihi : 29.12.2014

Tez Danışmanı: Prof. Dr. Cemal KÖSE

Trabzon 2014

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalında
Ramazan Özgür DOĞAN tarafından hazırlanan

EL ŞEKİL, HAREKET VE POZİSYON BİLGİLERİ İLE SANAL MOUSE
KONTROLÜ

başlıklı bu çalışma, Enstitü Yönetim Kurulunun 09/12/2014 gün ve 1580 sayılı kararıyla
oluşturulan jüri tarafından yapılan sınavda

YÜKSEK LİSANS TEZİ

olarak kabul edilmiştir.

Jüri Üyeleri

Başkan : Prof. Dr. Cemal KÖSE

Üye : Prof. Dr. Temel KAYIKÇIOĞLU

Üye : Yrd. Doç. Dr. Bekir DİZDAROĞLU

Prof. Dr. Sadettin KORKMAZ
Enstitü Müdürü

ÖNSÖZ

Günümüz taşınabilir bilgisayarların hemen hemen hepsi dâhili bir kameraya sahiptirler. Ancak bu kameralar sadece görüntüleme aygıtı olarak kullanılmaktadırlar. Bilgisayar kullanıcıları kısıtlı fiziksel alanda cihaza temasta bulunarak bilgisayar ile iletişime geçmektedirler. Özellikle sunum anında bilgisayar ekranından uzakta rahatça hareket etmek ve herhangi bir cihaza temas etmeden bilgisayarı kontrol etmek kullanıcı açısından oldukça önemlidir.

Yapılan çalışmada bilgisayarla görme ve görüntü işleme teknikleri kullanılarak kamera görüş alanına giren kullanıcının el bölgesinin yorumlanması ile hem bilgisayar kameraları daha fonksiyonel kullanılması hem de kullanıcının monitörden uzakta hiçbir fiziksel temasta bulunmadan bilgisayarı yönetmesi sağlanmaktadır.

Çalışmalarında danışmanlığımı üstlenen değerli hocam Prof. Dr. Cemal KÖSE'ye ilgi, destek ve tecrübelerinden dolayı teşekkürlerimi borç bilirim. Ayrıca projemize desteklerinden dolayı Bilim, Sanayi ve Teknoloji Bakanlığı'na, yüksek lisans eğitimim boyunca sabır, destek ve sevgileriyle yanımda olan eşim Hülya DOĞAN'a, sevgili aileme ve arkadaşlarıma çok teşekkür ederim.

Ramazan Özgür DOĞAN
Trabzon 2014

TEZ BEYANNAMESİ

Yüksek Lisans Tezi olarak sunduđum “El Őekil, Hareket ve Pozisyon Bilgileri ile Sanal Mouse Kontrolü” baŐlıklı bu alıŐmayı baŐtan sona kadar danıŐmanım Prof. Dr. Cemal KÖSE'nin sorumluluđunda tamamladıđımı, verileri/örnekleri kendim topladıđımı, deneyleri/analizleri ilgili laboratuvarlarda yaptıđımı/yaptırdıđımı, baŐka kaynaklardan aldıđım bilgileri metinde ve kaynakada eksiksiz olarak gösterdiđimi, alıŐma sürecinde bilimsel araŐtırma ve etik kurallara uygun olarak davrandıđımı ve aksinin ortaya ıkması durumunda her türlü yasal sonucu kabul ettiđimi beyan ederim.05/12/2014

Ramazan Özgür DOĐAN

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET.....	VIII
SUMMARY	IX
ŞEKİLLER DİZİNİ	X
TABLolar DİZİNİ.....	XII
SEMBOLLER DİZİNİ	XIII
1. GENEL BİLGİLER.....	1
1.1. Giriş.....	1
1.2. Literatür Araştırması	2
1.3. El Bölgesi Tespiti	4
1.3.1. Elektronik Eldiven Tabanlı Yaklaşımlar	5
1.3.2. Görsel Tabanlı Yaklaşımlar.....	6
1.3.2.1. El Bölgesi Özelliklerinin Çıkarılması	6
1.3.2.1.1. Model Tabanlı Yaklaşımlar	7
1.3.2.1.2. Görünüş Tabanlı Yaklaşımlar.....	7
1.3.2.1.3. Düşük Seviyeli Özellik Tabanlı Yaklaşımlar	7
1.3.2.2. El Bölgesi Özelliklerinin Sınıflandırılması	8
1.3.2.2.1. Kural Tabanlı Yaklaşımlar	8
1.3.2.2.2. Makine Öğrenmesi Tabanlı Yaklaşımlar.....	8
1.3.3. YCbCr Renk Uzayına Dayalı Ten Bölgesi Bulma Algoritması.....	8
1.3.4. Merkezsel Moment.....	9
1.3.5. Aktif Şekil Modeli.....	11
1.3.5.1. Teorik Altyapı	11
1.3.5.2. Model Oluşturma.....	11
1.3.5.3. Uygun İşaret Noktaları	11
1.3.5.3.1. Eğitim Setini Hizalama.....	13
1.3.5.4. Şeklin İstatistiksel Modeli	14
1.3.5.4.1. Temel Bileşen Analizi (TBA)	14

1.3.5.4.2.	TBA'nın İşlem Adımları	14
1.3.5.5.	Eğitimde Kullanılan Şekil Modelleri.....	18
1.3.5.6.	Modeli Yeni Pozisyona Yerleştirme.....	19
1.3.6.	TBA Öz El ile Özellik Çıkarma	21
1.3.7.	Görsel Kelime Çantaları ile Özellik Çıkarma	24
1.3.7.1.1.	Anahtar Noktası Belirleyici ve Yerel Tanımlayıcılar.....	24
1.3.7.1.1.1.	Anahtar Nokta Belirleyici.....	24
1.3.7.1.1.2.	Yerel Tanımlayıcılar.....	25
1.3.7.1.1.3.	SURF Yerel Tanımlayıcısı	25
1.3.7.1.2.	Görsel Sözlük Oluşturma ve Yerel Tanımlayıcıların Niceleştirilmesi.....	28
1.3.7.1.2.1.	Görsel Sözlük Oluşturma	28
1.3.7.1.2.2.	Yerel Tanımlayıcıların Niceleştirilmesi	30
1.4.	El Bölgesi Takibi	30
1.4.1.	Sensör Tabanlı El Bölgesi Takibi.....	31
1.4.2.	Bilgisayarla Görme Tabanlı El Bölgesi Takibi	31
1.5.	El Bölgesi Hareketlerinin Yorumlanması ile Bilgisayar Kontrolü.....	33
1.5.1.	El Bölgesi Hareketlerinin Sınıflandırılması	34
1.5.1.1.	Naive Bayes Sınıflandırıcı.....	34
1.5.1.2.	Yapay Sinir Ağları (YSA).....	35
1.5.1.2.1.	YSA'nın Genel Yapısı.....	36
1.5.1.2.2.	YSA Öğrenme Algoritmaları	37
1.5.1.2.3.	İleri Beslemeli Hata Geri Yayılımlı Öğrenme Algoritması.....	38
1.5.1.3.	Destek Vektör Makineleri (DVM)	39
1.5.1.3.1.	Doğrusal Destek Vektör Makineleri.....	40
1.5.1.3.1.1.	Hard Marjin.....	40
1.5.1.3.1.2.	Soft Marjin.....	42
1.5.1.3.2.	Doğrusal Olmayan Destek Vektör Makineleri	43
1.5.1.3.2.1.	Çekirdek Düzenlemesi	45
2.	YAPILAN ÇALIŞMALAR, BULGULAR VE İRDELEME	46
2.1.	Giriş.....	46
2.2.	Deneysel Sonuçlar ve Karşılaştırma.....	54
2.2.1.	Kullanılan Özellik Çıkarma Yöntemlerinin Deneysel Hız Sonuçları	54
2.2.1.1.	TBA Öz El Özellik Çıkarma Sonuçları	55

2.2.1.2.	Görsel Kelime Çantaları Özellik Çıkarma Sonuçları	56
2.2.2.	TBA ve GKÇ Özellik Çıkarma Sonuçlarının Karşılaştırılması	56
2.2.3.	Sınıflandırıcıların Sınıflandırma Başarılarının Karşılaştırılması.....	57
3.	SONUÇLAR VE TARTIŞMA.....	61
4.	ÖNERİLER	63
5.	KAYNAKLAR.....	64
ÖZGEÇMİŞ		

Yüksek Lisans Tezi

ÖZET

EL ŞEKİL, HAREKET VE POZİSYON BİLGİLERİ İLE SANAL MOUSE KONTROLÜ

Ramazan Özgür DOĞAN

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Cemal KÖSE
2014, 69 Sayfa

Bilgisayar-insan etkileşimi için kullanılan günümüz kontrol araçları bazı kısıtlamalara sahiptirler. Bilgisayar kullanıcıları bilgisayar kontrolü yapmak için fiziksel olarak bilgisayara temasta bulunmaları ve bu araçlar genelde kablo ile bağlandığından kullanıcının kısıtlı bir uzaklıkta bulunması gerekmektedir.

Yapılan tez çalışmasında kullanıcının hiçbir cihaza fiziksel temasta bulunmadan bilgisayar yönetimi ve kontrolü sağlayan bir sistem geliştirilmiştir. Görüş alanına giren bilgisayar kullanıcısının görüntülerinin kamera ile bilgisayara aktarımı sağlanmakta ve el hareketlerinin yorumlanması ile bilgisayarı yönetmesi ve kontrolü amaçlanmaktadır. Geliştirilen yaklaşımda ilk olarak bilgisayar kontrol edecek kişinin kamera görüş alanında belirlenen hedef çerçeveye girmesi beklenmektedir. Kamera görüş alanına giren kişinin hedef çerçevedeki ten bölgeleri ayırt edilmektedir. Hedef çerçevede bulunan ten bölgesinin merkezci momentler kullanılarak orta noktası bulunmaktadır. Orta noktası bulunan ten bölgesine aktif şekil modeli kullanılarak belli bir iterasyonda modelin oluşup oluşmadığı denetlenmektedir. El bölgesi modeli oluşturulduktan sonra tekrar aktif şekil modeli kullanılarak el bölgesi takibi başlatılmaktadır. El bölgesi takibi esnasında ilk olarak el bölgesi özellikleri çıkarılmakta, daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile bilgisayar kontrolü gerçekleştirilmektedir.

Anahtar Kelimeler: Ten Rengi Bulma Algoritması, Merkezci Moment, Aktif Şekil Modeli, Temel Bileşen Analizi, Görsel Kelime Çantaları, Öklid Uzaklığı, Naive Bayes Sınıflandırıcı, Yapay Sinir Ağları, Destek Vektör Makineleri, Çekirdek Destek Vektör Makineleri.

Master Thesis

SUMMARY

VIRTUAL MOUSE CONTROL WITH HAND SHAPE, GESTURE AND
POZITION INFORMATIONS

Ramazan Özgür DOĞAN

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering Graduate Program
Supervisor: Prof. Dr. Cemal KÖSE
2014, 69 Pages

Computer devices that used for computer-human interaction have various limitations. The computer users must make physical contact with the devices to control computer by using them, and have a limited distance because the connection is provided with cable.

In this study, a system that allows computer users to manage and control computer without physical contact is implemented. It is provided to transfer images of computer users within the field of view to a computer with the camera, and aimed to control and manage computers with the interpretation of hand gestures. In this proposed approach, firstly it is expected that the user enters target framework in the camera's field of view using computer camera. Skin regions of computer users within the field of view to a computer with the camera are recognized. Midpoint of skin regions in the target frame is found with central moment. Whether or not the model in the skin region midpoint of that found occurs at a particular iteration is controlled. After creation model of hand region, hand tracking is started by using active shape model again. During hand tracking, firstly the features of hand region are extracted, and then computer control is performed with interpretation of the movement of the hand by classifying the features of hand region.

Key Words: Skin Color Detection Algorithm, Central Moment, Active Shape Modeling, Principal Component Analysis, Bag of Visual Words, Euclidean Distance, Naive Bayes Classifier, Artificial Neural Network, Support Vector Machines, Kernel Support Vector Machines.

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1.1. Uygun işaret noktaları eli sınırlayan yüksek eğimin olduğu bölgelerden seçilmelidir. Ara noktalar eşit uzaklıklarla ayrılarak el bölgesinin tam belirginleştirmek için kullanılmaktadır.	12
Şekil 1.2. TBA nın iki boyutlu vektör setine uygulanması	16
Şekil 1.3. Eğitim görüntülerinden üretilmiş Şekli Modelleri	18
Şekil 1.4. ± 3 standart sapma ile parametrelerin değişmesi sonucu elde edilebilen varyasyonlar.....	19
Şekil 1.5. TBA Öz El akış diyagramı	23
Şekil 1.6. Tümlev görüntü elde edilmesi.....	26
Şekil 1.7. 9×9 boyutunda $\sigma = 1.2$ değerli Gauss filtresi kullanılarak elde edilmiş ölçek uzayın en alt seviyesi	27
Şekil 1.8. Hessian determinantları kullanılarak özellik noktaları çıkarılıyor	28
Şekil 1.9. Kümelenmiş özellik noktaları	29
Şekil 1.10. Özellik noktasının herhangi bir görsel kelimeye atanması	30
Şekil 1.11. Perseptron yapısı	36
Şekil 1.12. İleri beslemeli hata geri yayımlı öğrenme algoritması.....	38
Şekil 1.13. Doğrusal Ayrılabilen Veriler.....	41
Şekil 1.14. İki doğrusal ayrılabilen sınıf aracılığıyla hiper düzlemin gösterimi	41
Şekil 1.15. Doğrusal ayrılamayan verilerde hiper düzlemin gösterimi	43
Şekil 1.16. Doğrusal ayrılamayan veriler	44
Şekil 2.1. Sistemin temel işlem adımları	47
Şekil 2.2. Veritabanındaki 4 farklı sınıfa ait görüntüler. (a) Mouse serbest hareketi, (b) Mouse sol tıklama hareketi, (c) Mouse sağ tıklama hareketi, (d) Mouse taşıma hareketi	48
Şekil 2.3. Ten bölgesi ayırt edilen görüntü.....	49
Şekil 2.4. Merkezsel moment kullanılarak orta noktası bulunmuş görüntü	50
Şekil 2.5. AŞM başlangıç eşleme işlemi iterasyonları. (a) Model oturtma, (b) 5. iterasyon, (c) 10. iterasyon, (d) 15. iterasyon, (e) 20. iterasyon, (f) 22. iterasyon model hedeflenen bölgeye yakınsadı.	51
Şekil 2.6. TBA Öz El Özellik Vektörü Oluşturulması	52
Şekil 2.7. Görsel Kelime Çantaları yöntemi ile özellik vektörü oluşturulması.	53

Şekil 2.8. GKÇ-TBA Özellik Çıkarma Yöntemlerinin Farklı Boyutlardaki Görüntüler ile Eğitim ve Test Adınlarında Görüntü Boyutu(pixel) - Süre(saniye) karşılaştırmaları	57
Şekil 2.9. TBA Öz-El kullanılarak oluşturulan 20, 30, 40 ve 50 boyutlu özellik vektörlerinin sınıflandırılmasıyla elde edilen başarı grafiği.....	58
Şekil 2.10. GKÇ kullanılarak oluşturulan 20, 30, 40 ve 50 boyutlu özellik vektörlerinin sınıflandırılmasıyla elde edilen başarı grafiği	59
Şekil 2.11. TBA Öz-El özellik çıkarma yöntemi kullanılarak 100x100, 200x200 ve 300x300 boyutlu görüntülerin sınıflandırılmasıyla elde edilen başarı grafiği .	59
Şekil 2.12. GKÇ özellik çıkarma yöntemi kullanılarak 100x100, 200x200 ve 300x300 boyutlu görüntülerin sınıflandırılmasıyla elde edilen başarı grafiği.....	60

TABLolar DİZİNİ

	<u>Sayfa No</u>
Tablo 2.1. TBA Öz El Hesaplama Süreleri	55
Tablo 2.2. GKÇ Özellik Çıkarma Süreleri	56

SEMBOLLER DİZİNİ

YCbCr	luminance, blue-difference, red-difference <u>chroma</u>
RGB	Kırmızı, Yeşil, Mavi (Red, Green, Blue)
TBA	Temel Bileşen Analizi
GKÇ	Görsel Kelime Çantası
YSA	Yapay Sinir Ağları
DVM	Destek Vektör Makineleri
ÇDVM	Çekirdek Destek Vektör Makineleri
SIFT	Ölçek değişimsiz Özellik Dönüşümü (Scale Invariant Feature Transform)
SURF	Hızlandırılmış Gürbüz Öznitelikler (Speed up Robust Feature)
LoG	Laplacian of Gaussian
DoG	Difference of Gaussian
MSER	Maximally Stable Extremal Regions
LBP	Locally Binary Patterns
HSV	Hue, Saturation, Value
HSI	Hue, Saturation, Intensity

1. GENEL BİLGİLER

1.1. Giriş

Bilgi teknolojilerinin gelişmesi ile birlikte bilgisayar sistemleri tüm çalışma alanlarında kullanılmaya başlanmıştır. Geliştirilen bilgisayar sistemlerinin daha kolay ve etkin kullanımı için mouse klavye gibi bilgisayar-insan etkileşim araçları kullanılmaktadır. Ek olarak bu araçların kısıtlamalarını engellemek için günden güne yeni çalışmalar yapılmaktadır [1].

Kişisel bilgisayarların kullanıcı arayüzleri temel olarak klasik metin tabanlı komut satırından görsel tabanlı klavye mouse kontrolüne kadar geliştirilmiştir. Fakat bilgisayar-insan etkileşimi için kullanılan bu araçlar belirli kısıtlamalar sahiptirler. Bilgisayar kullanıcıları bu araçları kullanarak bilgisayar kontrolü yapmak için fiziksel olarak bilgisayara temasta bulunmaları ve bu araçlara genelde kablo ile bağlantı sağlandığı için kullanıcının kısıtlı bir uzaklıkta bulunması gerekmektedir. Klasik bilgisayar insan iletişim araçları kullanıcıların herhangi fiziksel temasta bulunmadan kablosuz bilgisayar kontrolünü sağlayamamaktadırlar [2].

El bölgesi tanıma, sanal gerçeklik, işaret dili tanıma ve bilgisayar oyunları gibi birçok alanda sıkça kullanıldığı için insan-bilgisayar etkileşiminde önemli araştırma alanlarından biridir. Literatürde bu alanla ilgili birçok çalışma yapılmasına rağmen, gerçek zamanda kullanılabilen dayanıklı bir el bölgesi bulma sistemi henüz geliştirilememiştir. Vücut dili insanlar arası iletişimde kullanılan önemli araçlardan biridir. El bölgesi tanıma, yüz tanıma, ses tanıma, göz izleme, dudak hareketlerini tanıma insan ve bilgisayar iletişiminde algısal bir kullanıcı arayüzü olarak kullanılmaktadırlar. El bölgesi tanıma sistemleri bilgisayar-insan arası etkileşimi daha kolaylaştırmakta ve daha doğal olmasını sağlamaktadır [3-4].

Günümüz taşınabilir bilgisayarların hemen hemen hepsi dâhili bir kameraya sahiptirler. Ancak bu kameralar sadece görüntüleme aygıtı olarak kullanılmaktadırlar. Bilgisayar kullanıcıları kısıtlı fiziksel alanda cihaza temasta bulunarak bilgisayar ile iletişime geçmektedirler. Özellikle sunum anında bilgisayar ekranından uzakta rahatça hareket etmek ve herhangi bir cihaza temas etmeden bilgisayarı kontrol etmek kullanıcı açısından oldukça önemlidir. Bilgisayarla görme ve görüntü işleme teknikleri kullanılarak kamera görüş alanına giren kullanıcının el bölgesinin yorumlanması ile hem bilgisayar

kameraları daha fonksiyonel kullanılabilmekte hem de kullanıcının moniterden uzakta hiçbir fiziksel temasta bulunmadan bilgisayarı yönetmesi sağlanmaktadır [5].

Bu tez çalışmasında kullanıcıya hiçbir cihaza fiziksel temasta bulunmadan bilgisayar yönetimi ve kontrolü sağlanmaktadır. Yapılan çalışmada görüş alınma giren bilgisayar kullanıcısının görüntülerinin kamera ile bilgisayara aktarımı sağlanarak bilgisayarı el hareketleri ile yönetmesi ve kontrolü amaçlanmaktadır. Tez çalışmasında geliştirilen yaklaşımda ilk olarak bilgisayar kontrol edecek kişinin kamera görüş alanında belirlenen hedef çerçeveye girmesi beklenmektedir. Kamera görüş alanına giren kişinin YCbCr renk uzayına dayalı ten bulma algoritması kullanılarak kullanıcının hedef çerçevedeki ten bölgeleri ayırt edilmektedir. Ten rengi bulma algoritmasında ilk olarak RGB renk uzayından YCbCr renk uzayına dönüşüm yapılmakta daha sonra ise Cb ve Cr bileşenlerinin maksimum ve minimum değerleri hesaplanarak bu değerler arasında kalan pikseller ten rengi olarak işaretlenmektedir. Hedef çerçevede bulunan ten bölgesinin merkezci momentler kullanılarak orta noktası bulunmaktadır. Orta noktası bulunan ten bölgesine aktif şekil modeli kullanılarak belli bir iterasyonda modelin oluşup oluşmadığı denetlenmektedir. El bölgesi modeli oluşturulduktan sonra tekrar aktif şekil modeli kullanılarak el bölgesi takibi başlatılmaktadır. El bölgesi takibi esnasında ilk olarak el bölgesi özellikleri çıkarılmakta, daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile bilgisayar kontrolü gerçekleştirilmektedir. Yapılan çalışmada el bölgesi özellikleri çıkarılması için temel bileşen analizi (TBA), görsel kelime çantası kullanılmaktadır. Temel bileşen analizi ile takip edilen el bölgesi çerçevesi buyut azaltımı amaçlanmaktadır. El bölgesi özelliklerinin sınıflandırılması için ise Naive Bayes, YSA, DVM ve çekirdek DVM gibi farklı algoritmalar kullanılmakta ve sonuçları karşılaştırılmaktadır.

1.2. Literatür Araştırması

El bölgesi tespiti ve takibi hareket modelleme, hareket analizi, tanıma ve makine öğrenmesi gibi birçok çalışmayı içeren karmaşık alanlardır. Literatürde bu alanlarla ilgili yapılmış birçok çalışma bulunmaktadır. El bölgesi kullanarak insan bilgisayar etkileşimi sağlayan birçok çalışma 1980 lerde Bolt tarafından gerçekleştirilmiş “Put-That-There” çalışmasından esinlenilerek geliştirilmiştir [6].

Starner ve arkadaşları 1998 yılında el bölgesi tanıma yaklaşımı olarak Hidden Markov modelini kullanmışlar ve gerçek zamanlı video tabanlı Amerikan işaret dili tanıma sistemi geliştirmişlerdir [7].

Ng ve Ranganath gerçek zamanda 14 farklı el bölgesini kullanarak görünüş tabanlı bir sistem geliştirmişlerdir. Bu çalışmada da Hidden markov modeli ve tekrarlayan sinir ağları kullanılmış el bölgesi tanımda %90 başarı elde edilmiştir [8].

Abe ve arkadaşları parmak özelliklerini kullanarak el bölgesini tanıyabilen bir sistem geliştirmişlerdir. Bu çalışmada parmakların kıvrımsal özellikleri elde edilerek özellik tabanlı bir yaklaşım önerilmiştir [9].

Hasanuzzaman ve arkadaşları ten rengi segmentasyonu ve çoklu özellik tabanlı eşleme algoritmalarını kullanarak gerçek zamanda el bölgesi tespit edebilen bir sistem geliştirmişlerdir. Bu çalışmada YIQ renk kanalları kullanılarak giriş görüntülerinden en büyük üç ten bölgesi elde edilmiş ve korelasyon, minimum uzaklık algoritmaları kullanılarak bu bölgeler karşılaştırılmıştır [10].

Segen el bölgesi tespiti için kenar tabanlı teknikler kullanarak bir sistem geliştirmiştir. Basit el silüetlerinden özellikler çıkarmış görüntü parametreleri elde etmiştir [11].

Kortenkamp ve arkadaşları gerçek zamanlı üç boyutlu el bölgesi tanıma sistemi geliştirmişlerdir. Sistemde karmaşık el bölgelerinin yorumlanabilmesi için akıllı kontrol yapıları birleştirilmiştir [12].

Nielsen ve arkadaşları gerçek zamanlı bir görme sistemi geliştirmişlerdir. Bu çalışmada görüntüden hareketli el bölgesini elde edebilmek için hızlı segmentasyon süreci kullanılmıştır. Segment edilen el bölgesi dizisinden tanıma işlemi gerçekleştirilmiştir [13].

Zhou ve arkadaşları yaptığı çalışmada SIFT ve TBA tabanlı sanal mouse gerçekleştirmeyi amaçlamıştır. Çalışmada mouse kontrolleri el hareketleri ile gerçekleştirmişlerdir. El bölgesi bulmak için SIFT (Scale Invariant Feature Transform) özellik çıkarma algoritması kullanılmış, el bölgesi takibi ise CAMShift algoritması ile gerçekleştirilmiştir [14].

Bao ve arkadaşları SURF (Speed Up Robust Features) özellik noktası çıkarma algoritması tabanlı dinamik el bulma metodu geliştirmişlerdir. Yapılan çalışmada 26 alfabetik el bölgesi kullanılmış, önerilen metodun başarı oranı %87 lere yükseltilmiştir [15].

Alsheakhali ve arkadaşları el bölgesi tanıma sistemi gerçekleştirmişlerdir. Çalışmalarında ilk olarak el bölgesinin merkezini bulmuş ve merkezi takip ederek el hareketlerini yorumlamışlardır [16].

Pansare ve arkadaşları karmaşık arka planlı yapılarda da çalışabilen gerçek zamanlı el bölgesi tanıma sistemi geliştirmişlerdir. Çalışma ön işlem, bölge çıkarma, özellik çıkarma ve özellik eşleme olmak üzere dört bölümden oluşmaktadır. İlk bölümde renkli olan görüntü ikili tabana çevrilmiştir. İkinci kısımda bloblar ve kenarlar kullanılarak bölgeler elde edilmiştir. Üçüncü kısımda bulunan kenarlar yardımıyla özellik vektörleri elde edilmiş, son kısımda ise vektörler Öklid uzaklığı ile karşılaştırılmıştır [17].

Asaari ve arkadaşları adaptif Kalman filtresini kullanarak el bölgesi takibi yapabailen bir sistem geliştirmişlerdir. Yapılan çalışmada ilk olarak ten rengi pikselleri kullanılarak el bölgesi ayırt edilmiş, sonraki aşamada ise Kalman takip algoritması ile el bölgesi takip işlemi gerçekleştirilmiştir [18].

Priyal ve arkadaşları kişi ve dokudan bağımsız statik el tanıma yapabilen bir sistem gerçekleştirmişlerdir. Çalışmada ilk olarak el bölgesi arka plandan ayrılmış, daha sonra ise elin geometrik özellikleri kullanılarak rotasyon normalizasyonu gerçekleştirilmiştir [19].

Abdalla ve arkadaşları işaret dilini el hareketleri ile tanıyabilen dinamik bir sistem geliştirmişlerdir. Önerilen çalışmada dinamik el bölgelerinden oluşan videolar giriş olarak kullanılmıştır. Elin ten rengi analizi için YCbCr uzayı kullanılmış, korelasyon ile karşılaştırma işlemi gerçekleştirilmiştir [20].

1.3. El Bölgesi Tespiti

El bölgesi tespiti, insan-bilgisayar arası etkileşimdeki önemli araştırma alanlarından biridir. İşaret dili tespiti, bilgisayar oyunları gibi birçok alanda kullanılmalarına rağmen tamamen performanslı çalışan el bölgesi tanıma sistemi geliştirilememiştir. Beden dili insanlar arasında etkileşimi sağlayan bir araçtır. El bölgesi tanıma ise beden hareketlerinin bilgisayar tarafından matematiksel olarak yorumlanmasıdır. El bölgesi tespiti birçok alanda tercih edilmekte olup kullanım alanlarından bazıları şu şekildedir:

- Sanal gerçeklik:

Sanal gerçeklik uygulamaları tek bir el bölgesini kullanarak sanal objelerin gerçekçi temsillerini elde etmektedirler. Bu uygulamalar 3D ekran etkileşimi ve 3D nin 2D de simülasyonu için de kullanılmaktadırlar.

- Robotik:

El bölgesinin kullanıldığı en etkin uygulamalarından biride robotik alanıdır. Robotik uygulamaları uzay keşfi ve askeri araştırmalar olmak üzere temel olarak iki alana ayrılmaktadır. Bu çalışmalarda el hareketleri sanal gerçeklik uygulamalarına benzer olarak robot kontrol ve etkileşimlerinde kullanılmaktadır.

- Masaüstü ve Tablet PC uygulamaları:

Masaüstü uygulamalarında el bölgeleri Mouse ve klavyeye alternatif etkileşim aracı olarak kullanılmaktadır. Bu tip uygulamalarda el bölgesi grafik çizme, doküman düzenleme ve kalem yerine yazma gibi çalışmalar yapabilmektedir.

- Bilgisayar oyunları:

El bölgesi tespiti sanal oyun uygulamalarında sıkça kullanılmaktadır. Bu uygulamalarda oyun objesinin etkileşimi ve hareket kontrolü için kullanıcının el bölgesi konumu ve hareketleri kullanılmaktadır.

- İşaret dili

İşaret dili el tespitinin kullanıldığı en etkin alanlardan biridir. İşaret dili yapısal olduğu için görsel algoritmalar için oldukça uygundur. Ek olarak bu uygulamalar engelli insanlar içinde oldukça faydalı olmaktadır.

Literatürde el bölgesi tespit yaklaşımları görsel ve elektronik eldiven tabanlı olmak üzere ikiye ayrılmaktadır.

1.3.1. Elektronik Eldiven Tabanlı Yaklaşımlar

Elektronik eldiven tabanlı yaklaşımlar bilgisayar ile iletişim kuracak kişinin ağır bir araç olan eldiven giymesini gerektirmektedir. Bu insan ile bilgisayar arasındaki iletişimin seviyesinin düşmesine neden olmaktadır (Mitra, 2007). Bu yaklaşımlar el hareketlerinin tespiti için parmak kıvrımlarını algılayan sensörler kullanmaktadırlar. Bilgisayara kablo ile bağlı oldukları için kullanıcı sınırlı bölgede hareket edebilmektedir. Bilgisayar ve insan etkileşiminin doğallığını bozmakta ve zorlaştırmaktadır. Elektronik eldiven teknolojisi oldukça pahalı olduğu için el bölgesi tespiti uygulamasında bu yaklaşımlar kullanıldığında oldukça pahalı sistemler ortaya çıkmaktadır. Güçlü ve etkin bir elektronik eldiven fiyatı oldukça yüksek olmasına rağmen görsel tabanlı çözümler oldukça ucuza mal olmaktadır. Elektronik eldiven tabanlı yaklaşımlarda kullanıcı bilgisayara bağlı bir eldiven giymek zorundadırlar. Bu aracı taşımak kullanıcıya oldukça külfet getirmekte hem de sınırlı

hareket alanına sahip olmasına neden olmaktadır. Görsel tabanlı yaklaşımlarda ise kullanıcının eldiven giymesi gerekse bile bilgisayara bağlı olmadığı için hareket alanı sınırlanmamaktadır. El bölgesi kişiden kişiye farklılık gösterebilmektedir. Çok büyük ya da küçük el bölgesine sahip olan insanlar elektronik eldiven tabanlı uygulamaları kullanmakta zorluk çekmektedirler.

1.3.2. Görsel Tabanlı Yaklaşımlar

Görsel tabanlı yaklaşımlar bilgisayarla görme, bilgisayar grafikleri, görüntü işleme, makine öğrenmesi, biometri gibi birçok alanda kullanılmaktadır. Başarılı bir sistem elde etmek için gerekli olan özellikler:

- Sağlıklik:

Gerçek hayatta uygulamaların çalışabilirliği için değişken aydınlatma, arka plan gibi olumsuzluklara karşı görsel bilgilerin oldukça kuvvetli olması gerekmektedir. Görsel tabanlı yaklaşımlar kullanıcıdan bağımsız ve tüm dış faktörlere karşı sağlam olmaktadır.

- Hesaplama etkinliği:

Görsel tabanlı yaklaşımlar genelde gerçek zamanlı sistemleri kullanılmaktadırlar. Bu yaklaşımlarda kullanılan görme ve öğrenme algoritmaları oldukça güçlü ve etkili olmalıdır.

- Kullanıcı toleransı:

Görsel tabanlı yaklaşımlarda hata toleranslı olmalıdır. Sistem çalışma anında gerçekleşen bir hata sistem performansını çok az etkilemelidir.

- Ölçeklenebilirlik:

Görsel tabanlı yaklaşımlar farklı ölçeklerdeki uygulamalara kolayca adapte edilebilmelidir.

Görsel tabanlı yaklaşımlar el bölgesinin özelliklerinin çıkarılması ve sınıflandırılması olmak üzere iki alt kısımda incelenmektedir.

1.3.2.1. El Bölgesi Özelliklerinin Çıkarılması

El bölgesinin kişiden kişiye farklılık göstermesi, farklı hareket ve dokulara sahip olması el bölgesi tanıma sürecinde özellik seçmeyi kritik hale getirmiştir. Parmak ucu ve

yönleri, el hatları gibi bazı geometrik özelliklerin çıkarımı ile statik el bölgesi tanıma işlemi gerçekleştirilebilmesine rağmen, aydınlanma koşulları gibi şartlardan dolayı bu işlem gerçek hayatta gerçekleştirilememektedir. El bölgesi özellik çıkartımı için renk, doku gibi birçok geometrik olmayan özellikler bulunmaktadır, fakat bu özellikler bu işlemler için yeterli olmamaktadır. Açıkça özelliklerini belirtmek kolay bir işlem olmadığı için tüm görüntü ve dönüşümü yapılmış görüntü giriş olarak alınmakta ve özellikler kullanıcı tarafından dolaylı olarak seçilmektedir. El özellikleri aşağıdaki yaklaşımlar kullanılarak türetilmektedir:

1.3.2.1.1. Model Tabanlı Yaklaşımlar

Model tabanlı yaklaşımlar el duruşunu ve eklem açısını çıkarmaya çalışmaktadırlar. Gerçekçi etkileşim için ideal bir yaklaşımdır. Genellikle bu yaklaşım el bölgesinin üç boyuttan iki boyuta izdüşürülmesi için gerekli olan parametre araştırmasını içermektedir. Model tabanlı yaklaşımlarda özellik çıkarma işlemi problemlidir. El içindeki dokular yetersiz kalmakta ve elin yapısından dolayı el içinde kenar bölgeleri tam net olmamaktadır. Elde edilen kenar bölgeleri genelde el bölgesi içinden olmamakta ve el bölgesi sınırlarından elde edilmektedir.

1.3.2.1.2. Görünüş Tabanlı Yaklaşımlar

Model tabanlı yaklaşımlarda karşılaşılan zorluklardan dolayı el bölgesi temsili elde etmek için birçok alternatif yaklaşım geliştirilmiştir. Son yıllarda geliştirilen alternatif yaklaşımlardan biride görünüş tabanlı yaklaşımdır. Görünüş tabanlı yaklaşım iki boyutlu imge dizisini kullanarak el bilgisini modellemektedir.

1.3.2.1.3. Düşük Seviyeli Özellik Tabanlı Yaklaşımlar

Birçok el izi uygulamalarında giriş görüntü dizileri ve el bölgesi arasında haritalama işlemi gerekmektedir. Oysaki birçok uygulamada el bölgesi yapısı gerekli olmamaktadır. Bu yaklaşımda gürültüye karşı dayanıklı düşük seviyeli görüntü hesaplamaları ile kolayca özellik çıkarılmaktadır. El bölgesi merkezi, eliptik sınırları belirleyen açılar literatürde geliştirilmiş düşük seviyeli yaklaşımlardandır.

1.3.2.2. El Bölgesi Özelliklerinin Sınıflandırılması

El bölgesi aşağıdaki yaklaşımlar kullanılarak sınıflandırılabilir.

1.3.2.2.1. Kural Tabanlı Yaklaşımlar

Kural tabanlı yaklaşımlar özellik girişleri arasında elle belirlenmiş kurallar dizisinden oluşmaktadır. Giriş olarak verilen el bölgesinin özellik seti çıkarılmakta ve belirlenen kurallara göre karşılaştırılmaktadır. Karşılaştırma kurallarına göre girişten verilen el bölgesi özelliklerinden çıkışlar üretilmektedir. Kural tabanlı yaklaşımlarda ana problem belirlenen kuralların kişiye bağımlı olmasıdır.

1.3.2.2.2. Makine Öğrenmesi Tabanlı Yaklaşımlar

Makine öğrenmesi verilerden öğrenmeye dayalı algoritma ve yapılarla ilgilenen bilimsel bir disiplindir. Bu yaklaşım el bölgesi tespiti için kullanılan en popüler yaklaşımlardandır. Hidden Markov Model literatürde kullanılan en göze çarpan algoritmalarındandır.

Kullanıcıya hiçbir cihaza fiziksel temasta bulunmadan bilgisayar yönetimi ve kontrolünü sağlayan bu tez çalışmasının el bölgesi tespiti aşamasında ilk olarak YCbCr renk uzayına dayalı ten bölgesi bulma algoritması kullanılarak kamera görüş alanına giren kişinin ten bölgesi ayırt edilir. Ayırt edilen ten bölgelerin merkezel momentler kullanılarak merkezi bulunur. Merkez noktaları bulunan ten bölgesine aktif şekil modeli kullanılarak belli bir iterasyonda modelin oluşup oluşmadığı denetlenir.

1.3.3. YCbCr Renk Uzayına Dayalı Ten Bölgesi Bulma Algoritması

YCbCr renk uzayı literatürde ten rengi algılamada sık kullanılan bir renk uzayıdır. YCbCr renk uzayında, Y bileşeni parlaklık bilgisini, Cb ve Cr bileşenleri ise renk bilgilerini temsil etmektedir. Bu yüzden parlaklık bilgisi kolaylıkla elde edilmektedir. YCbCr renk uzayına dayalı ten bölgesi bulma algoritmasında ilk olarak RGB renk uzayı aşağıdaki denklemler kullanılarak YCbCr renk uzayına dönüştürülmektedir.

$$Y = 0,299R + 0,587G + 0,114B \quad (1.1)$$

$$Cb = - 0,169R - 0,332G + 0,500B \quad (1.2)$$

$$Cr = 0,500R - 0,419G - 0,081B \quad (1.3)$$

Ten rengi bölgesi bulma işleminde Cb ve Cr değerleri kullanılmaktadır. Cb ve Cr bileşenlerinin maksimum ve minimum değerleri hesaplanarak bu değerler arasında kalan pikseller ten rengi olarak işaretlenmektedir. Cb ve Cr bileşenlerinin minimum ve maksimum değerleri aşağıdaki denklemler kullanılarak hesaplanmaktadır.

$$Cb_{min} = Cb_{ort} - Cb_{std} * f \quad (1.4)$$

$$Cb_{max} = Cb_{ort} + Cb_{std} * f \quad (1.5)$$

$$Cr_{min} = Cr_{ort} - Cr_{std} * f \quad (1.6)$$

$$Cr_{max} = Cr_{ort} + Cr_{std} * f \quad (1.7)$$

Cb ve Cr bileşenlerinin minimum ve maksimum değerlerine, bu bileşenlerin ortalama değerine (Cb_{ort}, Cr_{ort}) ve standartsapmalarına (Cb_{std}, Cr_{std}) göre karar verilmektedir.

1.3.4. Merkezsel Moment

2D uzaydaki sürekli bir $f(x, y)$ fonksiyonunun merkezsel momenti (1.8) ile hesaplanmaktadır.

$$M_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy \quad (1.8)$$

Bu sürekli fonksiyon $p, q = 0, 1, 2, \dots$ olarak değişirken programlamada integralin toplam fonksiyonuna dönüştürülebileceği düşünüldüğünde (1.8) formülü (1.9) formülündeki gibi kabul edilebilir.

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y) \quad (1.9)$$

$f(x, y)$ fonksiyonunun sıfırıncı momenti formül (1.10) ile hesaplanmakta ve görüntüde toplam bölgeyi temsil etmektedir.

$$M_{00} = \sum_x \sum_y I(x, y) \quad (1.10)$$

İkinci dereceden momentler ise formül (1.11) ve (1.12) kullanılarak hesaplanmaktadır.

$$M_{10} = \sum_x \sum_y x^1 I(x, y) \quad (1.11)$$

$$M_{01} = \sum_x \sum_y y^1 I(x, y) \quad (1.12)$$

Bu momentler görüntüde toplam bölgenin merkezini temsil etmektedirler. Bölgelerin merkez koordinatları formül (1.13) ve (1.14) de gösterilmekte olup görüntüyü tanımlayan bir özellik noktası olarak kullanılabilirler [21-22]. Toplam bölgenin merkezinin matamatiksel hesabında formül (1.13) ve formül (1.14) kullanılabilir.

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad (1.13)$$

$$\bar{y} = \frac{M_{01}}{M_{00}} \quad (1.14)$$

1.3.5. Aktif Şekil Modeli

Aktif şekil modeli ilk olarak 1995 yılında Tim Cootes ve Chris Taylor tarafından geliştirilmiştir. Bu modelde yeni görüntüde obje örneği oluşturmak için iteratif olarak obje şekil modelleri kullanılmaktadır. Şekil modeli kullanılan etiketlenmiş eğitim seti görüntülerinden alınmakta ve iteratif olarak değiştirilerek oluşturulmaktadır [23].

1.3.5.1. Teorik Altyapı

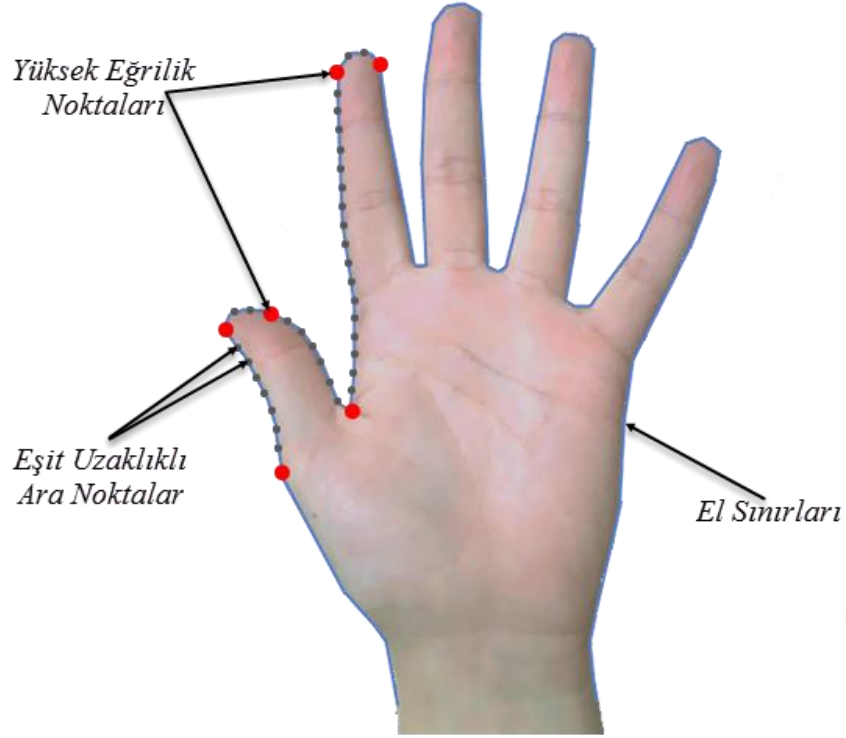
Bu bölümde görüntüdeki nesneyi gösterebilmek için kullanılan istatistiksel şekil modelinin teorik altyapısı açıklanmaktadır. Aktif şekil modellerinde danışmanlı eğitim kullanılarak girilen görüntü setinden elde edilen özellikler ile eğitim yapılmaktadır. Eğitim setinden elde edilen şekil varyasyonlarıyla oluşturulan model eğitimde kullanılan şekil varyasyonlarını taklit edebilir. Yeni bir görüntü yorumlamak için görüntüye en uygun model örneğinin parametreleri bulunmaktadır. Görüntüden model oluştururken parametreler ya da modelin görüntüdeki konum bilgileri sınıflandırma, ölçüm ve daha pek çok işlem için girdi olarak kullanılabilir. Bu kısımda bu işlemi en verimli şekilde gerçekleştiren algoritma anlatılmaktadır.

1.3.5.2. Model Oluşturma

Görüntüde ilgilendiğimiz nesnenin konumunu bulabilmek için öncelikle ilgili nesnenin modelini oluşturmamız gerekmektedir [24]. Bu istatistiksel şekil modelini inşa ederken ise nesnenin tipik örneklerinin bulunduğu görüntü setine ihtiyaç duyulmaktadır. Görüntü setindeki her bir görüntüde nesnenin şeklini tanımlayacak uygun işaret noktalarının belirlenmesi gerekmektedir.

1.3.5.3. Uygun İşaret Noktaları

Görüntüden nesnenin şekil modelini oluştururken seçilecek işaret noktaları için en iyi seçim nesneyi çevreleyen sınırların oluşturduğu köşe noktalarıdır [25]. İnsan eli fizyolojik yapısı nedeniyle işaret noktalarını belirlemek için yeterli kavisli yapıya sahiptir [26].



Şekil 1.1. Uygun işaret noktaları eli sınırlayan yüksek eğimin olduğu bölgelerden seçilmelidir. Ara noktalar eşit uzaklıklarla ayrılarak el bölgesinin tam belirginleştirmek için kullanılmaktadır.

Şekil modelini oluşturabilmek için uygun işaret noktaları belirlemek yeterli olmamaktadır. Aynı zamanda cisim sınırlarındaki işaret noktalarının ilişkilendirilebilirliğinin irdelenmesi gerekmektedir. İşaret noktaları ilişkilendirilerek cisimi çevreleyen sınırların yön bilgisi de elde edilmiş olmaktadır. Cisim sınırlarının oluşturduğu eğri boyunca belirlenen işaret noktalarını sırasıyla $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ olarak varsayılmaktadır.

İki boyutlu bir görüntü için $\{(x_i, y_i)\}$ şeklinde n adet işaret noktası belirlenebilmektedir, bu ifade $2n$ elemanlı tek boyutlu bir vektöre dönüştürüldüğünde x aşağıdaki gibi olmaktadır.

$$x = (x_1, \dots, x_n, y_1, \dots, y_n)^T \quad (1.15)$$

Eğer s adet eğitim örneği var ise s adet de formül (1.15) daki x_j vektörü oluşturmak gerekmektedir. İstatistiksel analizi başlatmadan önce bu vektörlerin normalize edilerek aynı koordinat sistemine oturtulmaları gerekmektedir. Nesnenin şekil modelinin bir karenin döndürüldüğünde, ölçeklendiğinde başka bir pozisyona ötelendiğinde hala bir kare olması gibi rotasyondan, ölçeklemeden ve ötelemeden bağımsız olması beklenmektedir.

1.3.5.3.1. Eğitim Setini Hizalama

Eğitim setinin hizalaması işlemi için farklı yöntemler bulunmakla birlikte en popüler yaklaşım Procrust Analizidir. Bu yöntem her örnek model için hizalama yapmaktadır. Böylece her modelin uzaklıklarının toplamı minimize edilmiş olmaktadır ($D = \sum |x_i - \bar{x}|^2$). Modellerden en az biri doğru konumlandırılmadıkça bu tanımlama kötü tanımlama olarak adlandırılır. Belki model doğru konumlandırılmış ve ölçeklenmiş olabilmekte fakat bu modelin dönmediğini garanti etmemektedir.

Bir eğitim setinin hizalanmasına dair analitik çözüm adımları aşağıdaki gibi sıralanabilmektedir:

Adım 1: Her bir eğitim örnek modeli ağırlık merkezi başlangıç noktasına gelecek şekilde konumlandırılmaktadır.

Adım 2: Modellerden biri ortalama model başlangıç tahmini olarak seçilmekte ve bu modele göre ölçeklendirme yapılmaktadır.

$$|\bar{x}| = \sqrt{\bar{x}_1^2 + \bar{y}_1^2 + \bar{x}_2^2 \dots} = 1 \quad (1.16)$$

Adım 3: Varsayılan yönelimi tanımlamak için ilk tahmin \bar{x}_0 olarak kaydedilmektedir.

Adım 4: Ortalama modelin geçerli tahmini ile bütün modeller hizalanmaktadır.

Adım 5: Hizalanan modellerle ortalama tahmini tekrar hesaplanmaktadır.

Adım 6: \bar{x}_0 ile hizalanmış ve ölçeklenmiş geçerli tahmin ortalamasına ölçekleme ve yönlendirme kısıtlamaları uygulanmakta böylece $|\bar{x}| = 1$ olmaktadır.

Adım 7: Yakınsama olmuyorsa Adım 4 e dönülmektedir.

Adım 8: Eğer ortalama tahmininde belirli bir iterasyon sonucunda önemsenecek bir değişim olmuyorsa yakınsama sonuçlandırılmaktadır.

İşlem hizalama boyunca final şekil modeli oluşturuluncaya kadar tekrarlanmaktadır. Temel yaklaşım bütün şekil modellerinin merkeze hizalanması sonrası tanjant uzayına ölçeklenerek ve döndürülerek ortalama şekil D ye benzetilmektedir.

1.3.5.4. Şeklin İstatistiksel Modeli

Ortak koordinat düzleminde hizalanmış olan x_i noktalar seti s olarak kabul edilmektedir. s setindeki vektörler gerçek zamanlı ve $2n$ boyutlu uzaydaki dağıtımlardır. Eğer s setindeki dağıtımlar modellenilebilirse bu model vasıtasıyla orijinal eğitim setindeki modeller gibi yeni modeller üretebilmek ve yeni şekillerin modele uygun olup olmadığını değerlendirebilmektedirler.

Problemi basitleştirmek için öncelikle veri setinin $2n$ olan boyut sayısı işlenebilir daha düşük boyuta düşürülebilmektedir. Bu işlem için kullanılacak en etkin çözüm Temel Bileşen Analizidir.

1.3.5.4.1. Temel Bileşen Analizi (TBA)

Temel bileşen analizi bilgisayarla görme ve görüntü işleme alanlarında çok boyutlu uzayda dağılan veri kümesinin ana eksenlerinin bulunmasını ve öz niteliklerinin çıkarılmasını sağlayan bir yöntemdir [35]. TBA yöntemi literatürde ilk defa Sirovich ve Kirby tarafından insan yüzlerinin temsili için kullanılmıştır [36]. Ayrıca TBA, çok değişkenli veri kümelerinin yorumlanmasında ve boyut indirgemesinde de kullanılmaktadır. Bu yöntem ile herhangi bir noktanın veri kümesindeki yaklaşık konumunu söyleyebilmek için daha az bilgiye ihtiyaç duyulmaktadır [37].

1.3.5.4.2. TBA'nın İşlem Adımları

$\{x_i\}$ şeklinde verilmiş olan bir vektör setine Temel Bileşen Analizi uygulandığında izlenecek adımlar şu şekildedir:

Adım 1: Formül (1.17) kullanılarak TBA uygulanacak verinin ortalama değeri hesaplanır.

$$\bar{x} = \frac{1}{s} \sum_{i=1}^s x_i \quad (1.17)$$

Adım 2: Formül (1.18) kullanılarak verinin kovaryansı hesaplanır.

$$S = \frac{1}{s-1} \sum_{i=1}^s (x_i - \bar{x})(x_i - \bar{x})^T \quad (1.18)$$

Adım 3: S veri kovaryansından özvektörler p_i ve ilişkili özdeğerler λ_i hesaplanır. (Özdeğerler $\lambda_i \geq \lambda_{i+1}$ olacak şekilde sıralanmalıdır).

Adım 4: Her özdeğer bize ilişkili özvektörünün yöne bağlı ortalaması hakkındaki verinin varyansını verir. Toplam varyans formül (1.19) ile hesaplanır.

$$V_T = \sum_t \lambda_i \quad (1.19)$$

Adım 5: f_v bütün varyansların oranıyken seçilecek ilk t adet özdeğer formül (1.20) ile bulunur.

$$\sum_{i=1}^t \lambda_i \geq f_v V_T \quad (1.20)$$

Bulunan $\{p_i\}$ özvektöründen x eğitim seti örneğinin yaklaşık konumu formül (1.21) ile hesaplanmaktadır.

$$x \approx \bar{x} + Pb \quad (1.21)$$

Burada $P = (p_1 | p_2 | \dots | p_t)$ t adet özvektörü içermekte ve b de bir t boyutlu bir vektör olup formül (1.22) deki gibi gösterilmektedir.

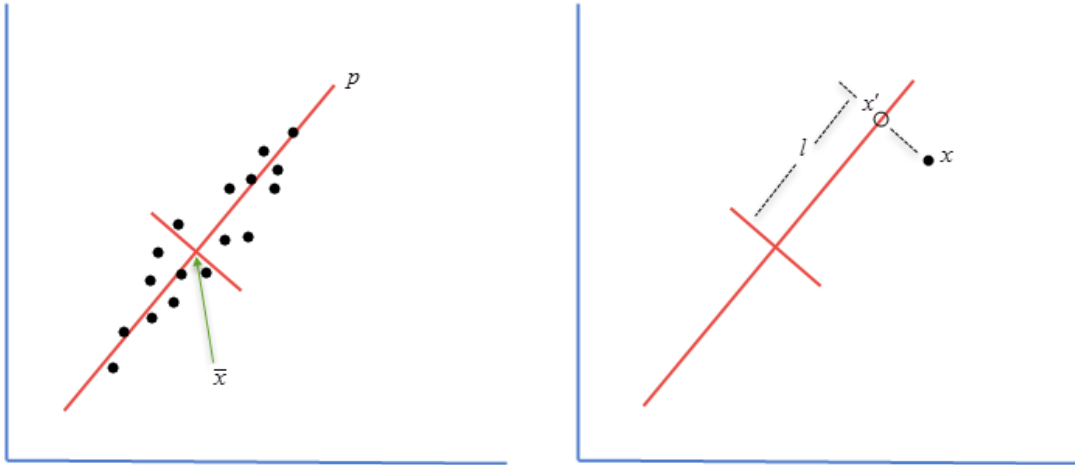
$$b = P^T (x - \bar{x}) \quad (1.22)$$

b vektörü deforme olabilen modelin parametreleri seti olarak tanımlanabilir. b nin elementlerini değiştirerek x modelini (1.22) eşitliği ile değiştirebiliriz.

Genellikle i^{th} parametreye uyan model varyasyonunu, modelin i^{th} modu olan b_i olarak adlandırırız. Özvektörler P orijinal şekil vektörleri kümesinin hizalanmasıyla döndürülmüş koordinat eksenleri olarak adlandırılır. b vektörü bu döndürülmüş uzayda bir nokta olarak tanımlanır.

p_i özvektörleri aslında verinin $2n - D$ uzayına dönüştürülmüş halini, b parametresi ise dönüştürülmüş veri kümesinin en anlamlı koordinatlarını temsil etmektedir [38].

Örnek olarak, Şekil 1.2 iki boyutlu $x - y$ uzayında dağılmış veri kümesinin temel eksenlerini göstermektedir. Bu durumda herhangi bir noktanın ortalama değer yardımı ile temel eksenlere en yakın noktaya izdüşümü bulunabilmektedir. l eksen boyunca uzaklığı temsil ederken x in ortalama değer yardımı ile eksene izdüşüm noktası $x \approx x' = \bar{x} + lp$ dir.



Şekil 1.2. TBA'nın iki boyutlu vektör setine uygulanması

Ayrıca eğer örnek sayısı boyut sayısından az ise işlemi hızlandırmak için farklı bir yaklaşım izlenebilmektedir. TBA yı $s < n$ iken s adet vektör x_i kümesini n -D boyutuna indirgediğini varsayalım. Bu durumda $n \times n$ lik kovaryans matrisinin hesaplanması çok zaman almaktadır. Bunun yerine x_i vektör kümesinin özvektörlerini ve özdeğerlerini veri kümesinden $s \times s$ matrisinden daha az hesaplama ile bulabilmektedir. Çünkü veri kümesinin özdeğer hesabı matrisin ölçülerinin küpü kadar zaman almakta ve bu dönüşüm önemli miktarda zaman kazancı sağlamaktadır.

Herbir veri vektöründen ortalama değer çıkarılmakta ve sonuç D matrisinde tutulmaktadır.

$$D = ((x_1 - \bar{x}) | \dots | (x_s - \bar{x})) \quad (1.23)$$

$n \times n$ kovaryans matrisi S formül (1.24) ile hesaplanmaktadır.

$$S = \frac{1}{s} DD^T \quad (1.24)$$

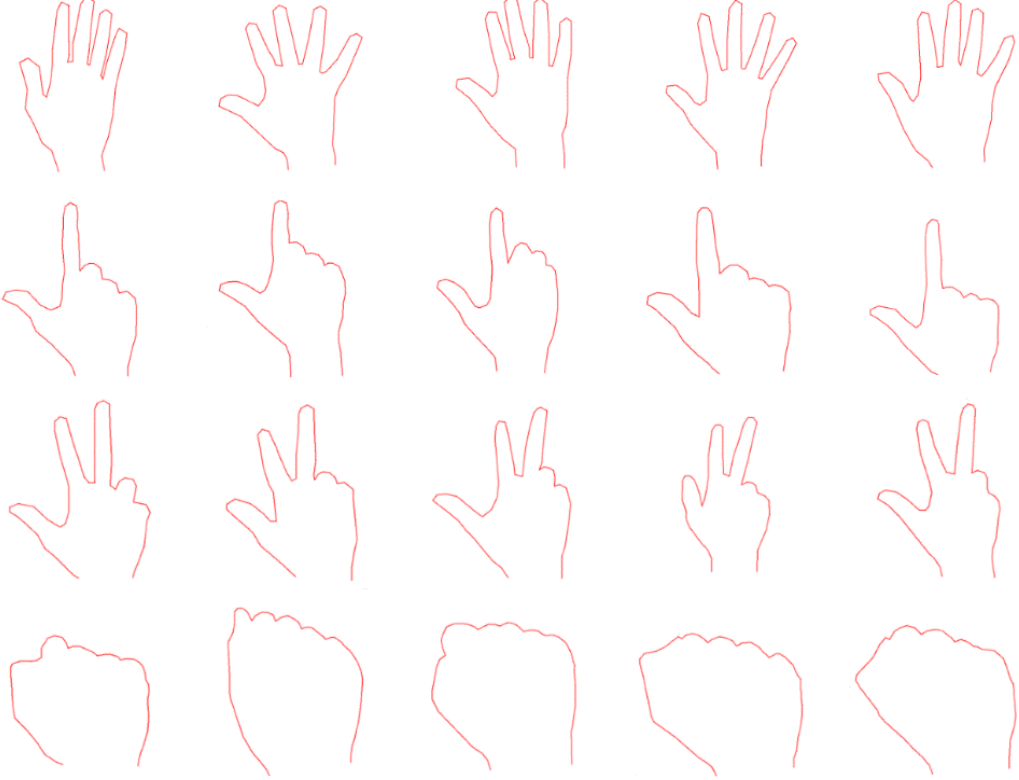
Yerine kullanılacak $s \times s$ matrisi T olsun. T matrisi formül (1.25) kullanılarak hesaplanmaktadır.

$$T = \frac{1}{s} D^T D \quad (1.25)$$

e_i , T nin s tane özvektörleri ve λ_i bu özvektörlerin büyükten küçüğe sıralı ilişkili özdeğerleri olsun. Buradan hepsi S nin özvektörü ve ilişkili özdeğerleri λ_i olan s tane De_i vektörüdür ve S nin kalan bütün özvektörlerinin ilişkili özdeğerleri sıfırdır. Ayrıca De_i birim uzunlukta olmak zorunda değildir ve normalize edilmesi gerekmektedir [39].

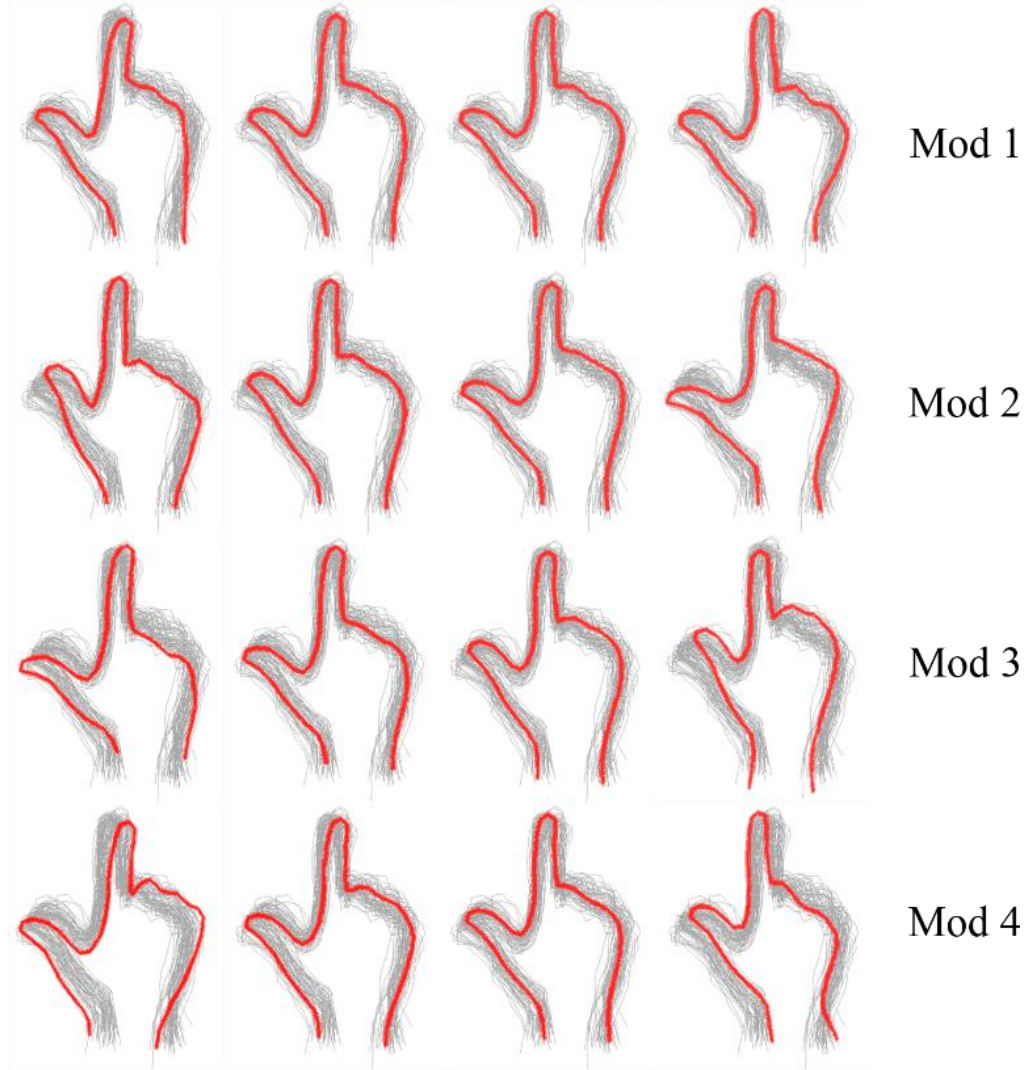
1.3.5.5. Eğitimde Kullanılan Şekil Modelleri

Çalışmada açık elin Aktif Şekil Modelini üretebilmek için 4 sınıf için 60 şar farklı el görüntüsünden her birinde 50 özellik noktası belirlenerek eğitim seti oluşturulmuştur. Her sınıfın modeli kendi sınıfı AŞM eğitim modelleri ile eğitilerek oluşturulmuştur.



Şekil 1.3. Eğitim görüntülerinden üretilmiş Şekli Modelleri

Hizalama gerçekleştirilip model hizalama tamamlandığında ve TBA işlemi gerçekleştirildiğinde modelin farklı modlardaki görünümünü TBA parametrelerini değiştirerek elde edebiliriz. Şekil 1.4 de ± 3 standart sapma ile TBA parametresi değiştirildiğinde elde edilen modlardan ilk dördünü görmekteyiz. Bu modlar ölçeğin değişmesi parmakların farklı pozisyonlara bakması elin z ekseninde hareketinden kaynaklanan derinlik farklılıkları gibi değişimleri temsilen kullanılabilir. Modların değişimiyle birlikte eğitim setinde bulunmayan bir görüntü için de model yerleştirme işlemi başarılı sonuç vermektedir.



Şekil 1.4. ± 3 standart sapma ile parametrelerin değışmesi sonucu elde edilebilen varyasyonlar.

1.3.5.6. Modeli Yeni Pozisyona Yerleřtirme

Şekil vektörünün özel bir değeri olan b , P tarafından tarif edilen döndürölmüş uzayda bir noktaya denk gelmektedir. Bu nedenle örnek bir model olarak kabul edilir. Bu vektör dönüşüm kullanılarak model koordinat ekseninden görüntü koordinat eksenine örnek bir model olarak dönüřtürölebilir.

Tipik olarak bu dönüşüm modelin görüntüdeki ötelenmesini (X_i, Y_i) , dönmesini θ , ölçeklenmesini s olarak tanımlayan öklit dönüşümü olarak kabul edilebilir.

Bu durumda görüntüdeki model noktaları X formül (1.26) deki gibidir.

$$X = T_{X_t, Y_t, s, \theta}(\bar{x} + Pb) \quad (1.26)$$

Burada $T_{X_t, Y_t, s, \theta}$ fonksiyonunda dönme θ ile ölçekleme s ve öteleme (X_t, Y_t) ile ifade edilmektedir. Örneğin tek bir (x, y) noktasına formül (1.27) deki gibi uygulanır.

$$T_{X_t, Y_t, s, \theta} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} X_t \\ Y_t \end{pmatrix} + \begin{pmatrix} s \cos \theta & -s \sin \theta \\ s \sin \theta & s \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (1.27)$$

X model örneğini yeni bir görüntü noktaları seti Y ye en iyi yerleşen model olarak konumlandırabilmek için formül (1.28) kullanabiliriz. İlişkili model ve görüntü noktaları arasındaki karesel uzaklıklar toplamını minimize etmemiz ifadeyi sadeleştirmemizle aynı anlamı taşır.

$$\|Y - T_{X_t, Y_t, s, \theta}(\bar{x} + p)\|^2 \quad (1.28)$$

Model yerleştirme işleminin adımları aşağıdaki gibidir;

Adım 1: Şekil parametrelerini setle, b yi sıfırla (ortalama şekil).

Adım 2: Model nokta pozisyonlarını $x = \bar{x} + P$ kullanarak üret.

Adım 3: Model model noktaları x i mevcut bulunan Y noktalarına en iyi hizalayan pozisyon parametreleri (X_t, Y_t, s, θ) bul.

Adım 4: T dönüşüm matrisi terslenerek Y nin model koordinat eksenine izdüşüm yapması sağla:

$$y = T_{X_t, Y_t, s, \theta}^{-1}(Y) \quad (1.29)$$

Adım 5: y yi ölçekleyerek \bar{x} in teğet düzlemine yansıt; $y' = y / (y \cdot \bar{x})$.

Adım 6: y' nü eşlemek için model parametrelerini güncelle:

$$b = P^T (y' - \bar{x}) \quad (1.30)$$

Adım 7: Yakınsama olmazsa Adım 2 ye dön.

Pozisyonda ya da şekil parametrelerinde önemli bir değişim gözlenmezse yakınsama sağlanarak işlem sonlandırılır. Yakınsama genellikle birkaç iterasyonda sağlanır.

1.3.6. TBA Öz El ile Özellik Çıkarma

Öz el yöntemi, el bölgelerini daha düşük boyutlu bir uzayda temsil etmek amacıyla geliştirilmiştir. Elde edilen yeni alt uzayda el olmayan görüntülerin dağılımı dikkate alınmamaktadır. Bu görüntü uzayı görüntünün sahip olduğu bütün piksellerin sayısı kadar boyuta sahip olup değerleri görüntü piksel değerleri aralığında değişmektedir. Örneğin $N \times M$ boyutlu gri ölçekli bir görüntünün alt uzay boyutu $N \times M$ olmakta ve her bir boyut görüntü gri seviyede olduğundan 0 ile 255 arasında değer almaktadır.

Bir görüntünün alt uzayı görüntü satırları birbiri ardına eklenerek elde edilmekte ve sonuç olarak bir sütun vektörü elde edilmektedir. Her el bölgesindeki yapılar aynı olduğu için elde edilen alt uzayda belli bölgelerde gruplaşmalar olmaktadır. Öz el yönetimiyle amaç görüntünün tamamını kullanmak yerine el bölgesi daha iyi temsil eden alt uzay kullanmaktır.

Öz el adımları şu şekilde sıralanmaktadır:

Adım 1: Genişliği M , yüksekliği N olan bir gri seviye görüntüsünün satırları art arda getirilerek el bölgesinde el vektörü elde edilmektedir.

$I \rightarrow M \times N$ boyutlarında gri seviye el görüntüsü

$P \rightarrow P \times 1$ boyutlarında el görüntüsü ($P = M \times N$)

Adım 2: Her bir el bölgesi görüntüsü el vektörüne dönüştürülerek eğitim matrisi oluşturulmaktadır. K tane eğitim görüntüsü olduğu düşünülürse $P \times K$ boyutlu eğitim seti matrisi elde edilmektedir.

$$X = [x_1 x_2 \dots x_K] \quad (1.31)$$

$$X = \begin{pmatrix} x_1^1 x_1^2 x_1^3 \dots x_1^K \\ x_2^1 x_2^2 x_2^3 \dots x_2^K \\ \dots \dots \dots \\ x_p^1 x_p^2 x_p^3 \dots x_p^K \end{pmatrix} \quad (1.32)$$

Adım 3: Eğitim setinin ortalaması hesaplanmaktadır. Hesaplanan ortalama vektörü, her bir eğitim vektöründen çıkarılarak sıfır ortalamalı yeni eğitim veri matrisi elde edilmektedir.

$$\mu = \frac{1}{M} \sum_{i=1}^M x_i = \begin{bmatrix} m_1 \\ m_2 \\ \dots \\ m_p \end{bmatrix} \quad (1.33)$$

$$r_i = x_i - \mu \quad (1.34)$$

$$A = [r_1 r_2 \dots r_K] \quad (1.35)$$

$$A = \begin{pmatrix} x_1^1 - m_1 x_1^2 - m_1 x_1^3 - m_1 \dots x_1^K - m_1 \\ x_2^1 - m_2 x_2^2 - m_2 x_2^3 - m_2 \dots x_2^K - m_2 \\ \dots \dots \dots \\ x_p^1 - m_p x_p^2 - m_p x_p^3 - m_p \dots x_p^K - m_p \end{pmatrix} \quad (1.36)$$

Adım 4: formül (1.37) kullanılarak A matrisinin kovaryans matrisi hesaplanarak özdeğer ve özvektörler elde edilmektedir.

$$C = A * A^T = \frac{1}{M-1} \sum_1^M (x_i - \mu) * (x_i - \mu)^T \quad (1.37)$$

$$A = \begin{pmatrix} (x_1^1 - m_1)^2 (x_1^2 - m_1) (x_2^1 - m_2) \dots (x_1^K - m_1) \\ (x_2^1 - m_2) (x_1^2 - m_1) (x_2^2 - m_2)^2 \dots (x_1^K - m_2) \\ \dots \\ (x_p^1 - m_p) (x_1^K - m_1) (x_p^2 - m_p) (x_2^K - m_2) \dots (x_p^K - m_p)^2 \end{pmatrix} \quad (1.38)$$

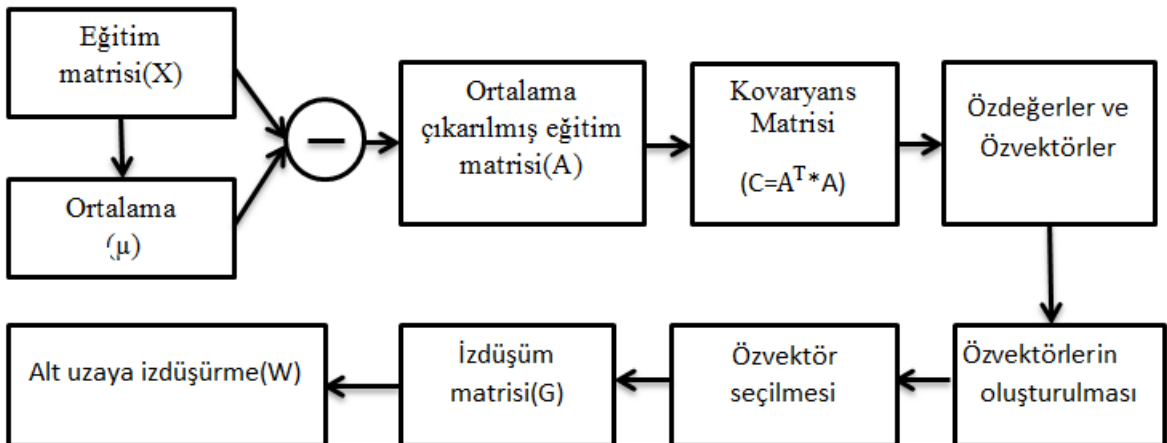
Adım 5: Elde edilen özdeğerler büyükten küçüğe sıralanmakta ve ilk n tanesinin özvektörü alınarak izdüşüm matrisi hesaplanmaktadır.

$$G^T = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_k \end{bmatrix} \quad (1.39)$$

Adım 6: Elde edilen matrise göre eğitim seti izdüşürülmektedir.

$$W = G^T A = G^T (X - \mu) \quad (1.40)$$

TBA Öz El özellik çıkarma algoritmasının takip edilen el görüntülerine uygulanışı eğitim ve test olarak iki aşamadan oluşmaktadır.



Şekil 1.5. TBA Öz El akış diyagramı

Akış diyagramından da görüldüğü üzere öncelikle eğitim görüntüleri eğitim matrisine dönüştürülmüştür. Bunun için her bir eğitim görüntüsünün satırları birbiri ardına eklenerek bir sütun vektörüne dönüştürülmüş ve sütun vektörü haline dönüştürülmüş görüntüler birleştirilerek eğitim matrisi G elde edilmiştir.

1.3.7. Görsel Kelime Çantaları ile Özellik Çıkarma

Görsel kelime çantaları görüntüdeki bulunmuş anahtar noktaları etrafındaki bölgeleri temsil eden yerel tanımlayıcıları bulmak için geliştirilmiştir. Görüntülerin anahtar noktaları etrafında belirgin bölgeler var olabilmektedir [40]. 128 boyutlu SIFT gibi yerel tanımlayıcılar bu bölgelerin karakteristiklerini belirlemek için kullanılabilirler. Fakat her görüntü için yerel tanımlayıcıların çıkarılması oldukça fazla yük getirmekte ve her tanımlayıcının en yakın komşusuna bakılması oldukça zaman gerektirmektedir. Görsel kelime çantaları ise tanımlayıcıları görsel kelimelere sayılaştırarak bu probleme çözüm getirmekte ve tanımlayıcıların boyutunu azaltmaktadır. Böylece her görüntü kolaylıkla ve hızla incelebilmektedir [41-43].

Görsel kelime çantası modelinde kullanılan genel adımlar şu şekildedir:

Adım 1: Anahtar noktası belirlemek için metod seçme ve anahtar noktaların etrafında yerel bölgeleri tanımlama.

Adım 2: Görsel sözlük oluşturulması

Adım 3: Görsel sözlük kullanılarak anahtar noktaların nicelenmesi

Adım 4: Görsel kelimelerin frekansları ile görüntünün temsil edilmesi

1.3.7.1.1. Anahtar Noktası Belirleyici ve Yerel Tanımlayıcılar

Anahtar nokta belirleme ve yerel tanımlayıcılar görsel kelime çantaları modelinin ön aşaması olarak düşünülmekte ve bu modelin temelini oluşturmaktadırlar.

1.3.7.1.1.1. Anahtar Nokta Belirleyici

Anahtar nokta belirleyiciler yerel olarak sabit noktalar bulmaktadır. Bu noktalar geometrik ve fotometrik dönüşümlere karşı oldukça dayanıklıdır. Literatürde yapılan çalışmalarda farklı belirleyiciler kullanılmıştır. Belirleyiciler kullanılan özellik ve boyuta

göre göre farklı sonuçlar göstermektedirler. Çalışmalarda kullanılan anahtar noktası belirleyicilerinin bazıları:

- Laplacian of Gaussian (LoG)
- Difference of Gaussian (DoG)
- Harris Laplace
- Hessian Laplace
- Harris Affine
- Hessian Affine
- Maximally Stable Extremal Regions (MSER)

1.3.7.1.1.2. Yerel Tanımlayıcılar

Yerel tanımlayıcılar belirleyiciler tarafından belirlenmiş olan anahtar noktaların komşularını tanımlamaktadırlar. Tanımlayıcılar ölçek ve aydınlatma değişimlerine, sıkıştırılmaya karşı dayanıklı olmalıdır. Tanımlayıcı performansı anahtar nokta belirleyicilerin gücüne bağlıdır. Literatürde kullanılan popüler yerel tanımlayıcılar:

- SIFT (Scale Invariant Feature Transform)
- SURF (Speeded Up Robust Features)
- LBP (Locally Binary Patterns)

SIFT yerel tanımlayıcısında anahtar nokta etrafındaki dairesel bölge 4x4 lük bölgelere ayrılmakta ve bu bölgelerin gradyan değerlerinin histogramı hesaplanmaktadır. Histogram aralığı tanımlayıcı boyunu düşük tutmak için 8 olarak belirlenmiştir. Bunun sonucunda her bir anahtar noktasının 128 boyutlu özellik vektörü elde edilmektedir. Yapılan çalışmalara göre SIFT özellik tanımlama yöntemi özellik noktası tanımlayıcıları arasında en iyilerdendir.

Yapılan çalışmada yerel tanımlayıcısı olarak SURF metodu kullanılmaktadır.

1.3.7.1.1.3. SURF Yerel Tanımlayıcısı

SURF algoritması bir görüntüde yerel özellik noktalarının belirlenmesi için kullanılmakta olup ilk olarak 2006 yılında Herbert Bay tarafından geliştirilmiştir [44-45].

SURF algoritması nesne tanıma, 3 boyutlu uygulamalar gibi birçok bilgisayarla görme alanında sıkça kullanılmaktadır. SIFT algoritması ile benzer adım ve prensiplere sahiptir, fakat SURF algoritması SIFT algoritmasına göre daha hızlı olduğu literatürde yapılan çalışmalarda ispatlanmaktadır.

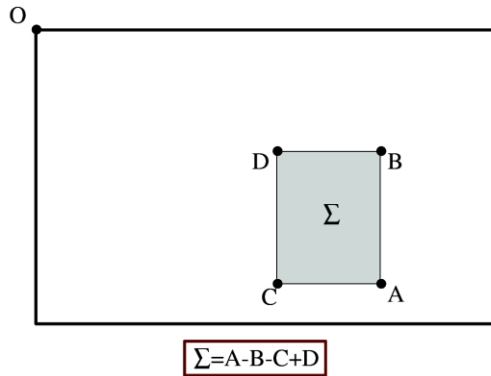
SURF özellik çıkarma algoritması görüntülerin özellik noktalarını bulmak için Hessian matris yaklaşımını kullanmaktadır. Bu yaklaşım Viola ve Jones tarafından geliştirilmiş olan tümlev görüntülerin kullanılmasını sağlamaktadır. Tümlev görüntü yaklaşımı hesaplama süresini oldukça düşürmektedir. Hessian matris determinanı ölçüt olarak kullanılmakta ve bölgeler arasındaki değişimler hakkında bilgi vermektedir.

$$H = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (1.41)$$

$$L_{xx}(x, \sigma) = I(x) * \frac{d^2}{dx^2} g(\sigma) \quad (1.42)$$

$$L_{xy}(x, \sigma) = I(x) * \frac{d^2}{dxy} g(\sigma) \quad (1.43)$$

Formüldeki $L_{xx}(x, \sigma)$ görüntünün gauss fonksiyonunun ikinci dereceden türevi ile konvelüsyonundan elde edilmektedir. Konvelüsyon işlemleri hem hesaplama karmaşıklığını düşürmekte hem de tümlev görüntülerin hesaplanmasını kolaylaştırmaktadır.

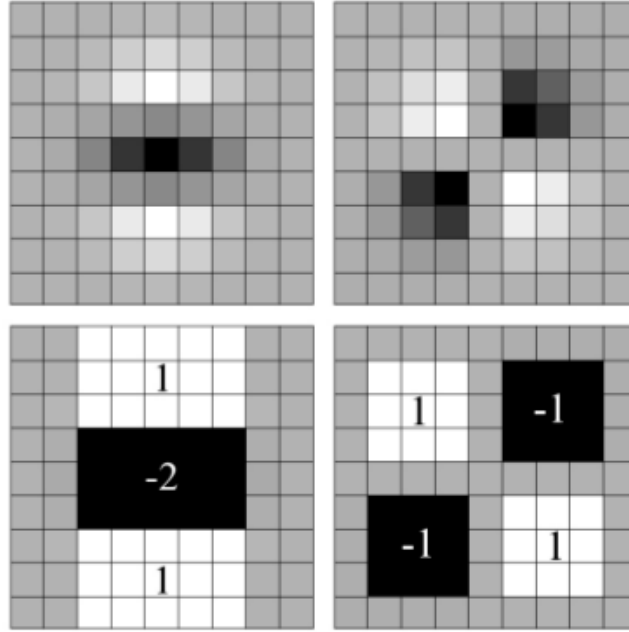


Şekil 1.6. Tümlev görüntü elde edilmesi

Tümlev görüntü ($I(x)$) formül kullanılarak görüntünün her pikseli için hesaplanmakta ve yeni bir görüntü elde edilmektedir. Bu görüntü x ve etrafındaki pikseller arasında kalan dikdörtgen bölgesinin piksel değerlerinin toplamından oluşmaktadır. Tümlev görüntü elde edildikten sonra dikdörtgen bir alanda piksel büyüklüklerinin toplamlarının sonucu sadece üç toplama işlemiyle hesaplanabilmektedir.

$$I(x) = \sum_{i=0} \sum_{j=0} I(x, y) \quad (1.44)$$

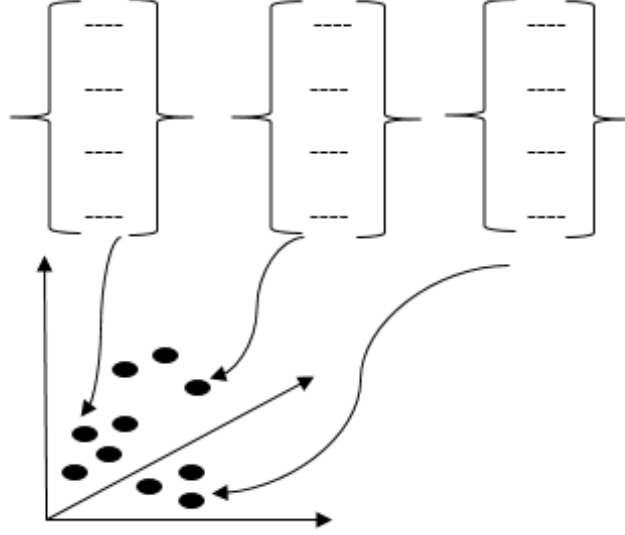
Hessian matris için kullanılan gauss süzgeçleri ($g(\sigma)$) uygulanmadan önce ayrıştırılıp kırılması gerekmektedir. SURF algoritması bu süzgeçleri kutu süzgeçlerle birlikte kullanmaktadır. Şekil 1.7 de 9x9 boyutunda $\sigma = 1.2$ değerli Gauss filtresi kullanılarak elde edilmiş ölçek uzayın en alt seviyesi bulunmaktadır. Şekilde gri alanlar 0'ı, beyazlar pozitif ve siyahlar ise negatif temsil etmektedirler.



Şekil 1.7. 9x9 boyutunda $\sigma = 1.2$ değerli Gauss filtresi kullanılarak elde edilmiş ölçek uzayın en alt seviyesi

Bütün ölçeklerde ilgi noktası tespit etmek için değişik boyutlarda ve ölçeklerde kutu süzgeçleri aynı anda tümlev görüntülere uygulayarak ölçek-uzay yapısı çıkarılmaktadır. Bu

ölçek-uzay yapısında Hessian determinantlarının sonuç tespiti yapılarak özellik noktaları çıkarılmaktadır.



Şekil 1.8. Hessian determinantları kullanılarak özellik noktaları çıkarılıyor

1.3.7.1.2. Görsel Sözlük Oluşturma ve Yerel Tanımlayıcıların Niceleştirilmesi

SIFT ve SURF gibi yerel tanımlayıcıları çok sayıda özellik noktası bulmayı sağlamaktadır. Bu yoğunluk dayanıklılık için önemli olmakla birlikte özellik sayısı boyunun fazla olması işlem yükünü artırmaktadır. Görsel kelime çantaları ile konumları değiştirilmeden özellik noktalarının toplanıp sayısallaştırılması sağlanarak tanımlayıcılar kısaltılmaktadır. Görsel kelimelerde yerel tanımlayıcıların sayısallaştırılması için ilk olarak görsel sözlükler oluşturulmalıdır. Her 128 boyut görsel kelimeler ile etkilenmektedir.

1.3.7.1.2.1. Görsel Sözlük Oluşturma

Görsel kelime sözlüklerini oluşturmak için özellik noktalarını giriş olarak alan kümeleme metotları kullanılmaktadır. Kümeleme metotlar kullanılarak her küme için sözlükten tek bir görsel kelime atanmaktadır. Görsel sözlükler en yakın kümelerin merkezlerinin etiketlenmesi ile özellik noktalarının sayısallaştırılmasında kullanılmaktadır.

Yapılan tez çalışmasında kümeleme metodu olarak k-ortalama algoritması kullanılmaktadır. K ortalama optimizasyon yöntemi adımları şu şekildedir:

Adım 1: K adet başlangıç küme merkezi, Z_1, \dots, Z_K , $S = \{X_1, X_2, X_3, \dots, X_n\}$ veri kümesinden rastgele seçilmektedir.

Adım 2: Veri kümesindeki her bir veri, eğer $\|X_i - Z_j\| < \|X_i - Z_p\|$, $p = 1, \dots, K$ ve $j \neq p$ ise $C_i j = 1, \dots, K$, kümesine atılmaktadır.

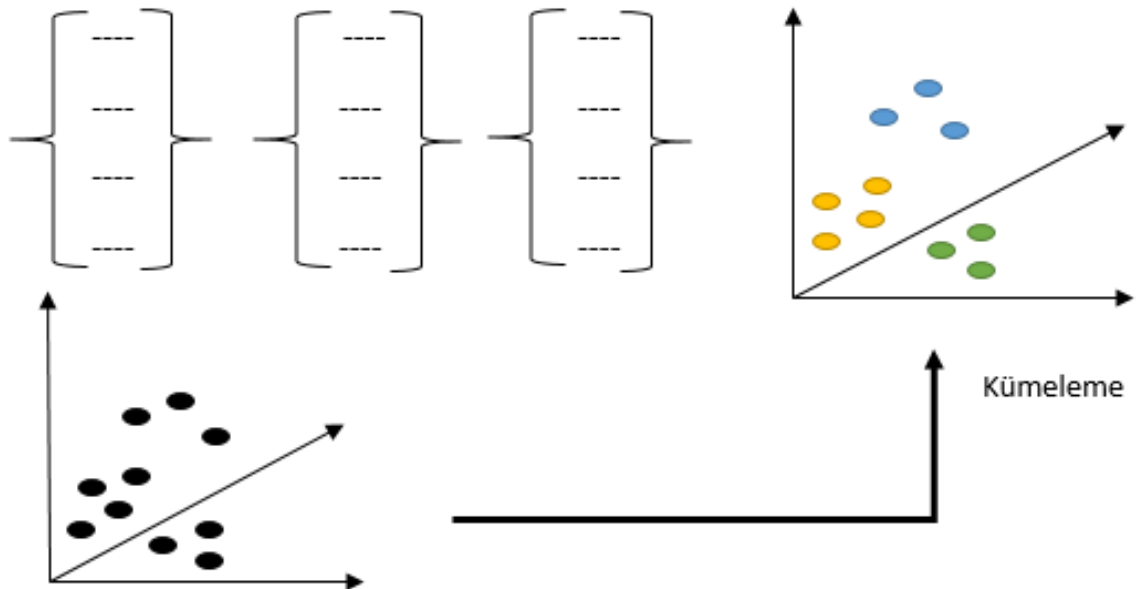
Adım 3: Her bir kümenin yeni küme merkezi yeniden hesaplanmaktadır.

$$Z_i^y = \frac{1}{n_i} \sum_{X_j \in C_i} X_j \quad (1.45)$$

Burada n_i , C_i kümesinde bulunan eleman sayısını göstermektedir.

Adım 4: Eğer $z_i^y = Z_i$, $i = 1, \dots, K$ ise işlem durdurulmaktadır. Değilse Adım 2 ye dönülmektedir.

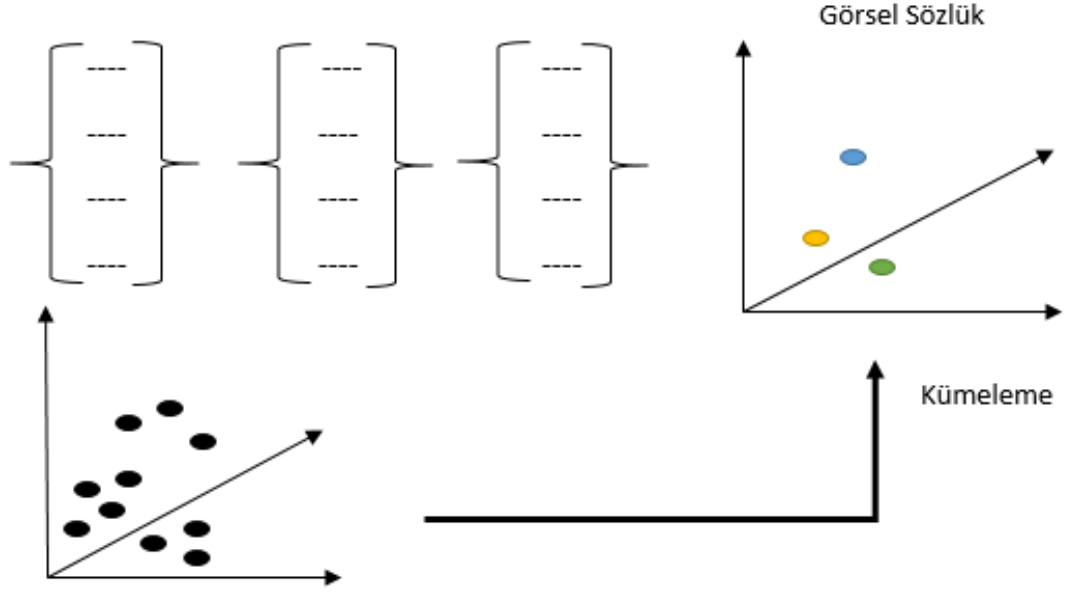
Eğer algoritma dördüncü adımdaki şarta bağlı olarak sonlandırılmazsa, belirlenen iterasyon sayısına ulaşıldığında işlem durdurulmaktadır.



Şekil 1.9. Kümelenmiş özellik noktaları

1.3.7.1.2.2. Yerel Tanımlayıcıların Niceleştirilmesi

Yerel tanımlayıcıların niceleştirilmesi bulunan özellik noktalarının herhangi bir görsel kelimeye atanması sürecidir. En yakın komşuluk algoritması özellik noktalarının görsel kelimeye atanmasında kullanılan en yaygın süreçlerdendir.



Şekil 1.10. Özellik noktasının herhangi bir görsel kelimeye atanması

Son adım olarak oluşturulan görsel kelimelerin frekanslarına göre görüntü temsil edilmektedir.

1.4. El Bölgesi Takibi

El bölgelerinin geometrik yapılarının ve modellerinin farklı olması insan-bilgisayar etkileşimi, sanal gerçeklik gibi alanlarda el bölgesi modeli ve takibini önemli kılmaktadır. Literatürde el bölgesi tanıma ve takibi için birçok çalışma geliştirilmiştir. Fakat hala güçlü ve tam doğrulukta çalışabilen bir sistem geliştirilememiştir. El bölgesi takip yaklaşımları sensör tabanlı ve bilgisayarla görme tabanlı olmak üzere iki alt gruna ayrılabilir [34].

1.4.1. Sensör Tabanlı El Bölgesi Takibi

Bazı el tanıma sistemleri el hareketlerini 3 yönlü akselerometreden gelen giriş işaretleri ile el hareketlerini tanımakta ve analiz etmektedir. Sistemler sensör veri birleşimi, segmentasyon ve tanıma kısımlarından oluşmaktadır. Sensör tabanlı el bölgesi tanıma sistemleri donanımsal araçlara ihtiyaç duymaktadırlar.

1.4.2. Bilgisayarla Görme Tabanlı El Bölgesi Takibi

Literatürde geliştirilmiş birçok el takibi metodları bilgisayarla görme tabanlıdır. Bu yaklaşımlarda bir veya daha fazla kamera yardımıyla el bölgeleri bilgisayara aktarılıp kaydedilmektedir. Bilgisayarla görme tabanlı el bölgesi takibi yaklaşımları ten rengi segmentasyonu, şekil tabanlı ve arka plan çıkarma olmak üzere üç alt başlığa ayrılmaktadırlar [46].

El bölgesi takibi için geliştirilmiş ilk çalışmalarda kullanılan yaklaşımların çoğu ten algılamaya dayalı algoritmalar [47]. Elin ten rengi kameradan bilgisayara aktarılan görüntülerden el takibi yapmak için uzun bir süre kullanılmış bir özelliktir. RGB (Jones ve Rehg [48]), normalize edilmiş RGB (Yang [49]), HSV (Herpers [50], Sandeep ve Rajagopalan [51], Sigal [52]), YUV (Kampmann [53]) gibi farklı renk uzayları ten algılamaya dayalı takip için kullanılmıştır.

Ek olarak el takibi için kullanılan algoritmalar;

Snake takip için kullanılan algoritmalar dandır. Takip görüntü enerjisinin minimize edilmesiyle sağlanır. Objeler arası mesafeler çok fazla ise klasik Snake algoritması takibi gerçekleştirememektedir. Frameler arası geçiş yaparken, Snake yerel minimuma takılacağından başarılı olamamaktadır [53].

Mean-shift algoritmasının parametrik olmayan ve karmaşık hesaplamalar gerektirmeyen bir yöntem olması nedeniyle nesne takibi, uygulamalarında kolaylıkla kullanılmaktadır. Bilgisayarlı görmede nesnenin modelini çıkarabilmek için bu nesnenin özelliklerinden faydalanılmaktadır. Dolayısıyla hareketli cismin modelini oluşturabilmek için çalışma da bir model oluşturma çerçevesi içerisine bu cismin girmesi beklenilmektedir. Daha sonra cisim bölgeye tam oturduğunda cismin renk dağılımı özelliğinden yararlanmak üzere normalize edilmiş hedef histogramı çıkarılmaktadır. Kameradan gelen bir sonraki kamera görüntüsünde aynı bölgenin tekrar normalize edilmiş

histogramı çıkarılmakta ve bu normalize edilmiş histogram aday histogramı oluşturulmaktadır [54].

Cam-Shift ilk olarak bir kullanıcı arabiriminde etkin bir şekilde yüz izleme gerçekleştirmek için tasarlanmıştır, Bradski (1998). Mean-Shift temeline dayalıdır. CAM-Shift ve Mean-Shift arasındaki temel fark, Cam-shift adaptif olasılık dağılımı kullanırken Mean-Shift in statik dağılım kullanmasıdır. Yani ilk belirlenen pencere boyutu görüntünün boyu değişse de ya da ölçeklense de değişmez. Cam-shift te ise görüntü özelliklerine göre hedef bölgenin boyutu değişmektedir.

SIFT algoritması ile görüntünün özellik noktaları çıkarılmaktadır. Hedef takibinde ise kameradan gelen görüntüler arasında aynı özellik noktaları bulunup takip edilerek takip etme işlemi gerçekleştirilmiş olunur. İlk olarak ilk görüntüdeki takip edilecek hedef belirlenir ve özellik noktaları çıkarılır. Diğer görüntüde ise özellik noktaları çıkarılarak önceki görüntü ile karşılaştırılır. Ve yeni görüntüde bulunan ortak noktalar çerçeveye alınır [55].

SURF algoritması ile görüntünün özellik noktaları çıkarılır. SIFT algoritması mantığına dayalı ve daha hızlı ve daha iyi sonuç vermektedir [56].

Hedef takip sistemlerinin durum-uzay modeli elde edilirken, hareketin yapısına göre hedefin konumu, hızı, ivmesi, dönüş hızı vd. sistem durum vektörünün bileşenleri olarak alınabilir. Ancak, hareket esnasında, lineer sistem modeli ile gerçek hedef dinamiği arasında bir uyumsuzluk meydana gelebilir. Sistem dinamiğinin lineer bir modelle tanımlanması ve ortamda başka hata kaynaklarının bulunma ihtimalinin olması nedeniyle, durum-uzay modelin yetersizliğini önleyebilmek ve daha iyi tahmin sonuçlarına ulaşabilmek amacıyla bazı uyarlı teknikler kullanılabilir. Bu tekniklerden biri çoklu model algoritmalarıdır.

Filtreleme problemi üzerinde, bugüne kadar birçok çalışma yapılmıştır ve yapılmaya da devam etmektedir. Fagin (1964), Fitzgerald (1971), Kalman filtresinin ıraksaması durumunu incelemiştir. Mehra (1972), modelde yer alan hata terimlerinin kovaryanslarının bilinmemesi durumunda, bu matrislerin tahmin edilmesi durumunu incelemiş ve kendini uyarlayan Kalman filtresi için çalışmalar yapmıştır. Hashemipour et al. (1988), Rao and Durant (1991), çok algılayıcı sistemler için paralel Kalman filtresini incelemişlerdir. Özbek (1993), kesikli-zaman durum-uzay modellerinde indirgemeli tahmin yöntemini incelemiş, Özbek ve Öztürk (1993), gayri safi millî hâsıla değerinin Kalman filtresi ile tahmini ile ilgili bir çalışma yapmışlardır. Lin and Atherton (1993), manevra yapan hedef

takibinde kullanılan Etkileşimli Çoklu Model (EÇM) algoritmasının bazı özellikleri ve farklı sayıdaki modeller ile farklı parametrelerin kullanılmasının etkilerinden bahsetmiştir. Li and Bar-Shalom (1993), Monte Carlo simülasyonları kullanılmadan EÇM algoritması için tahmin metodunun performansını örnekler ile tartışmışlardır. Xia et al. (1994), durum-uzay modelinin hatalı kurulması durumunda, Kalman filtresinde ortaya çıkan ıraksama durumunu ele almış ve filtrede bazı güçlendirmelerin yapılmasını sağlayacak unutm faktörünün hesaplanması için çeşitli algoritmalar önermişlerdir. Özbek vd. (1996), Kalman filtresinde modelin hatalı kurulmasından kaynaklanan ıraksama problemi ile ilgili bir çalışma yapmışlardır. Yeddanapudi et al. (1997), çoklu hedef, çoklu algılayıcılı hava trafik kontrolüne, EÇM algoritmasının uygulanması üzerinde durmuştur. Özbek (1997), durum-uzay modelinin hatalı kurulması durumunda, Kalman filtresinde meydana gelen ıraksama durumunu ele almış ve filtrede bazı güçlendirmeler yapacak bir algoritma önermiştir. Mazor et al. (1993), EÇM algoritmalarını hedef takibi açısından incelenmiş ve algoritmaların varsayımlarını ve çeşitli durumlardaki uygulamalarını göstermiştir. Özbek ve Aliev (1998), Xia et al. (1994)'ın önerdikleri unutm faktörü ile ilgili bir çalışma yapmışlardır [57].

Kameradan alınan görüntülerden el takibi yaptığımızı düşünelim. Her görüntüde elin konum değerlendirmesi yapılır. Genel olarak algoritmalarda bu tahmini şekilde bulunur. Bu tahmin gerçekte tamamiyle aynı değildir. Condensation algoritması bu tahmini gerçeğe en yakın şekilde bulan ve takip için en çok kullanılan algoritmalardan biridir. Bir kameradan alınan görüntü dizisinde hangi piksel değerlerinin hareket halinde olduğunu tanımlayabilmek önemli bir sorundur. Condensation bu problemi çözmek için geliştirilmiş istatistiksel bir algoritmadır. Condensation Algoritması Kalman Filtresine göre daha kolaylık sağlar ve gerçek zamana daha yakın çalışır. Algoritma ilk olarak Isard ve Blake (1998) tarafından geliştirilmiş ve uluslararası makalede bahsedilmiştir. Görüntünün bütün pikseller üzerinde işlem yapmaması bu algoritmanın en önemli özelliklerinden biridir.

Yapılan tez çalışmasında el bölgesi modeli oluşturulduktan sonra tekrar aktif şekil modeli kullanılarak el bölgesi takibi başlatılmaktadır.

1.5. El Bölgesi Hareketlerinin Yorumlanması ile Bilgisayar Kontrolü

El bölgesi takibi esnasında ilk olarak el bölgesi özellikleri çıkarılmakta, daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin

yorumlanması ile bilgisayar kontrolü gerçekleştirilmektedir. Yapılan çalışmada el bölgesi özellikleri çıkarılması için temel bileşen analizi (TBA) kullanılmaktadır. Temel bileşen analizi ile takip edilen el bölgesi çerçevesi buyut azaltımı amaçlanmaktadır. El bölgesi özelliklerinin sınıflandırılması için ise Öklid uzaklığı, yapay sinir ağları (YSA), destek vektör makinaları (DVM), çekirdek destek vektör makinaları (ÇDVM) gibi farklı algoritmalar kullanılmakta ve sonuçları karşılaştırılmaktadır.

1.5.1. El Bölgesi Hareketlerinin Sınıflandırılması

1.5.1.1. Naive Bayes Sınıflandırıcı

Naive Bayes sınıflandırıcı, sınıflar arası ilişkiyi analiz ederek verinin hangi sınıfa ait olduğu tahmin eden bir sınıflandırıcıdır. Naive Bayes, Bayes teoremi temeline dayalıdır. Verinin öğrenilmesi esasına dayalı olup olasılıkçı bir yaklaşımdır [58].

Naive Bayes sınıflandırıcı, etiketsiz kayıtlar içerisinde, sınıf etiketini tahmin etmekte kullanılabilecek kısa bir model oluşturmayı hedeflemektedir. Naive Bayes sınıflandırıcısı, kategorik verileri sınıflandırma konusunda basit ve hızlı bir teknik olup klasik değişkene bağlı sınıflandırıcılardır. Eğitim verileri için belirli bir olasılık dağılımı uygun kabul edilmektedir. Naive Bayes sınıflandırıcısı, ilgili eğitim setinde bulunan her bir bağımsız öznitelik arasındaki ilişkiyi ve her bir ilişki içerisindeki şartlı olasılığı çıkarmak için analiz etmektedir. Tahmin işlemi, bağımsız değişkenlerin, bağımlı değişkenler üzerindeki etkilerini bir araya getirerek yeni bir durumu sınıflandırmak için yapılmaktadır [59].

Naive Bayes sınıflandırıcı Bayes teoreminin basitleştirilmiş halidir. Bayes teoremi aşağıdaki denklemle ifade edilmektedir.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (1.46)$$

$P(A|B) \rightarrow B$ olayı gerçekleştiği durumda A olayının meydana gelme olasılığı.

$P(B|A) \rightarrow A$ olayı gerçekleştiği durumda B olayının meydana gelme olasılığı.

$P(A) \rightarrow A$ olayının önsel olasılığı

$P(B) \rightarrow B$ olayının önsel olasılığı

$x(i)$ veri kümesi ve S ayrıştırılacak sınıflar kümesi olarak düşünüldüğünde;

$$P(S_i | x) \times p(x) = p(x | S_i) \times P(S_i) \quad (1.47)$$

$$p(x) = \sum p(x | S_i) \times P(S_i) \quad (1.48)$$

$P(S_i) \rightarrow S_i$ nin öncel olasılığı

$P(S_i | x) \rightarrow S_i$ nin ardıl olasılığı

$P(x) \rightarrow x$ in olasılık yoğunluk fonksiyonu

$P(x | S_i) \rightarrow x$ in koşullu olasılık yoğunluk fonksiyonu

Sınıfı belli olmayan bir veri kümesinin (x) atanacağı sınıf (S_i) aşağıdaki formül kullanılarak belirlenmektedir.

$$P(S_i | x) > P(S_j | x) \quad (1.49)$$

1.5.1.2. Yapay Sinir Ağları (YSA)

Yapay sinir ağları, biyolojik sinir sistemlerinin matematiksel olarak modelleme fikrinden esinlenilerek geliştirilmiştir [60]. Gelişmiş birçok YSA sistemleri, öğrenme yeteneği gibi beynin önemli özellikleri gerçekleştirebilmekte ve çok büyük ölçekte paralel hesaplama yapabilmektedir. Literatürde yapılan çalışmaların katkılarıyla gittikçe gelişmekte olan YSA modelleri, görüntü işleme ve örüntü tanıma alanlarında sınıflandırma aracı olarak kullanılmaktadır [61].

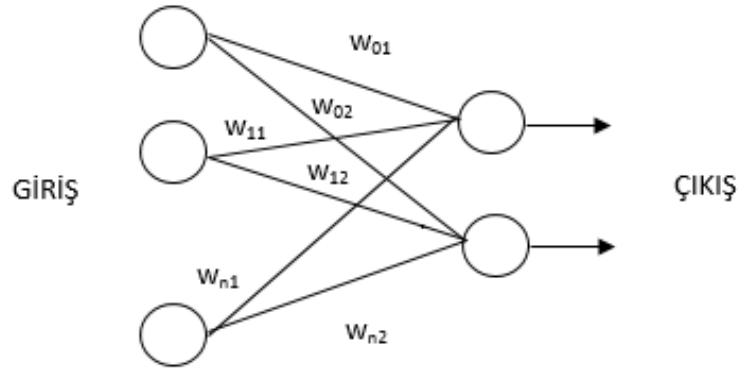
YSA, sinir bilimleri, matematik, istatistik, fizik, bilgisayar bilimleri ve mühendislik gibi birçok alanın kullanılmasıyla modellenmektedir. YSA modelleri ise sinyal işleme, örüntü tanıma, tahmin zaman serileri analizi gibi birçok alanla kullanılmaktadır (Haykin, 1999). McCulloch ve Pitts 1943 yılında temel hesaplama nöronunun ilk modelini ortaya çıkararak YSA alanında ilk çalışmayı yapmışlardır. 1949 yılında Donald Hebb nöronlar

arası bağlantıları güncelleme için yeni bir yaklaşım önermiştir. Rosenbatt 1958 yılında perseptron teorisini, Bernard Widrow ve Marcian Hoof ise 1960 yılında Widrow-Hoff öğrenme kuralını geliştirmişlerdir. Marvin Minsky ve Seymour Papert 1969 yılında perseptron modelinin sınırlandırılması üzerine çalışmalar yapmış, Werbos ise 1975 yılında geri yayılım ağını geliştirmiştir. Hopfield (1982), Kohenen (1982) ve Rumelhalt (1986) ise YSA alanında yeni modeller geliştirmişlerdir.

1.5.1.2.1. YSA'nın Genel Yapısı

YSA, hiyerarşik olarak birbirine bağlı ve birlikte çalışan yapay nöronlardan oluşmaktadır. YSA modeli perseptron, birbirine bağlı nöronlar, öğrenme kuralları ve stratejisinden oluşan bir model olarak düşünülebilmektedir. YSA'nın temel amacı ağa verilen girişlere göre çıkış üretmektir. YSA genel olarak iki fazdan oluşmaktadır: İlk olarak YSA'nın çıkışında probleme yönelik başarılı bir sonuç elde etmek için ağın girişinden eğitim seti verilerek ağın istenen yeteneği kazanması sağlanmaktadır (öğrenme fazı). Öğrenme sürecinden sonra ağa test verileri giriş olarak verilerek sonuçlar elde edilmektedir (test fazı).

YSA'nın ana işlem birimleri perseptron ve yapay nöronlardır. Yapay nöronların yapısı biyolojik nöronlardan daha basittir. Şekil 1.11'de basit bir perseptron yapısı gösterilmektedir.



Şekil 1.11. Perseptron yapısı

Bir perseptron giriş sinyalleri, ağırlıklar (w), aktivasyon fonksiyonu ve çıkışlardan oluşmaktadır. Toplama fonksiyonu ağırlıklar ve giriş fonksiyonun çarpılması ile

uygulanmaktadır. Aktivasyon fonksiyonları genelde doğrusal olmayan fonksiyonlardır ve net çıkış bulmak için net girişlere uygulanmaktadırlar. Eşik, Parçalı Doğrusal, Sigmoid ve Gauss fonksiyonları (Karlık 2007) YSA da aktivasyon fonksiyonları olarak kullanılmaktadır.

YSA modelleri genel olarak birbirine paralel üç katmandan oluşmaktadır. Bunlar giriş katmanı, ara katman ve çıkış katmanlarıdır. Giriş katmanı dışarıdan gelen giriş sinyalleri ile beslemekte ve bilgileri ara katmana iletmektedir. Ara katman ana operasyon merkezidir ve her giriş için ağırlıkları hesaplamaktadır. Son katman ise çıkış katmanıdır, çıkış katmanı ara katmandan beslenmekte ve dışarıdan gelen giriş sinyallerinin sonuçlarını göstermektedir.

YSA bağlantı mimarilerine göre ileri beslemeli YSA ve geri beslemeli YSA olarak iki gruba ayrılmaktadır. İleri beslemeli YSA döngü olmayan graflara, geri beslemeli YSA ise geri besleme bağlantılı graflara sahiptir.

1.5.1.2.2. YSA Öğrenme Algoritmaları

YSA da öğrenme, öğrenilmesi istenen çıkışlara göre giriş ağırlıklarının hesaplanması olarak tanımlanmaktadır. Ağırlıkları güncellemek için art arda matematiksel işlem serisi uygulanmaktadır. Öğrenme metodolojisine göre YSA danışmanlı öğrenme ve danışmansız öğrenme olarak iki kategoriye ayrılmaktadır.

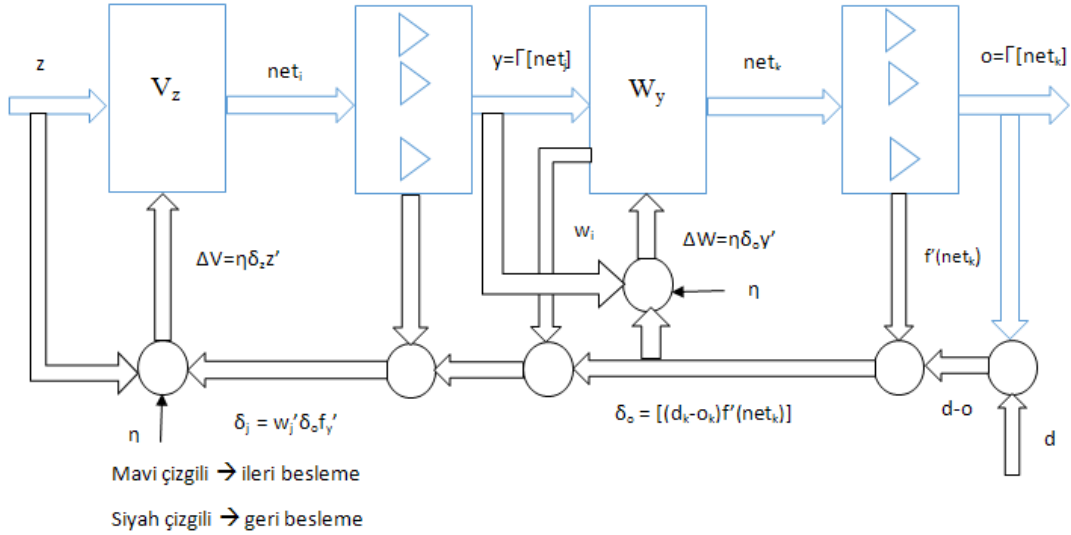
Danışmanlı öğrenmede, girişler ve istenen sonuçlar problemle ilişkilidir ve sistemden elde edilmektedir. Asıl amaç istenen ve gerçek sonuçlar arasındaki hatayı minimize ederek ağırlıkları bulmaktır. Widrow ve Hoff un 1960 yılında geliştirdiği geri yayılım algoritması danışmanlı öğrenme algoritmalarına örnek olarak verilebilmektedir.

Danışmansız öğrenmede, sadece yerel bilgiler kullanılmaktadır, girişler sistemden elde edilmekte fakat istenen çıkışlar sistemden elde edilmemektedir. Danışmansız öğrenmeler genelde sınıflandırma algoritmaları için kullanılmaktadır. 1982 yılında geliştirilen Kohonen, Hebbian ve Winner-take-all danışmansız öğrenme algoritmalarına örnek olarak gösterilebilmektedirler.

Literatürde birçok öğrenme algoritması bulunmaktadır. Tez çalışmasında ileri beslemeli hata geri yayımlı öğrenme algoritması kullanıldığı için bu kısımda sadece bu algoritmadan bahsedilecektir.

1.5.1.2.3. İleri Beslemeli Hata Geri Yayılımlı Öğrenme Algoritması

Blok diyagramı şekil 1.12 de gösterilen ağ, danışmanlı öğrenme modunda eğitilmektedir. z eğitim vektörü, danışman tarafından sağlanmış d istenen sonuç vektörü ile düzenlenmekte ve sonuçta o çıkış vektörü elde edilmektedir.



Şekil 1.12. İleri beslemeli hata geri yayılımlı öğrenme algoritması

Algoritma adımları:

Adım 1: V ve W ağırlıklarının ilk değer ataması yapılır.

Adım 2: Z girişleri ağa gönderilerek ilk ileri besleme evresi başlatılır. İlk olarak ara katman ve çıkış katmanının cevabı olan y ve o değerleri hesaplanır.

$$y = \Gamma[Vz] \quad (1.50)$$

$$o = \Gamma[Wy] \quad (1.51)$$

Adım 3: Döngü hatası hesaplanır.

$$E \leftarrow E + \frac{1}{2} \|d - o\| \quad (1.52)$$

Adım 4: Her katman için hata işaret vektörleri hesaplanır.

Çıkış katmanları için hata işareti:

$$\delta_{ok} = \frac{1}{2} (d_k - o_k) (1 - o_k^2) \quad k = 1, 2, 3, \dots, K \quad (1.53)$$

Ara katmanlar için hata işareti:

$$\delta_{yj} = \frac{1}{2} (1 - y_j^2) \sum_{k=1}^K \delta_{ok} w_{kj} \quad k = 1, 2, 3, \dots, K \quad (1.54)$$

$$j = 1, 2, 3, \dots, J$$

Adım 5: Çıkış katmanının ağırlıkları güncellenir.

$$w_{kj} \leftarrow w_{kj} + \eta \delta_{ok} y_j \quad (1.55)$$

Adım 6: Ara katmanın ağırlıkları güncellenir.

$$v_{ji} \leftarrow v_{ji} + \eta \delta_{yj} z_i \quad (1.56)$$

Adım 7: Eğitim kümesinde örnek kalıp kalmadığı kontrol edilir ve eğer örnekler tamamlanmadıysa ikinci adıma geri dönülüp adımlar tekrar edilir.

Adım 8: Eğitim kümesi örnekleri tamamlandıktan sonra bütün eğitim döngüsü için sonuç hata değeri hesaplanır. Öğrenme işlemi, sonuç hata değeri E_{max} değerinin altında olduğu zaman durdurulur.

1.5.1.3. Destek Vektör Makineleri (DVM)

DVM 1995 yılında Vapnik tarafında geliştirilmiş temeli istatistiksel öğrenmeye dayalı eğitilmiş sınıflandırma tekniklerinden biridir. Danışmanlı öğrenme temelli bu sınıflandırma tekniğinde verilerin aşırı öğrenmesinden kaçınılarak tahmin doğruluğunu

maksimum yapmak amaçlanmaktadır. Eğitim seti üzerinde öğrenme yaparak test verisi üzerinde doğru tahmin yapmayı genelleştirmeyi amaçlayan bir makine öğrenmesidir [62].

DVM sınıflandırıcılarında dağılım hakkında herhangi bir ön bilgi varsayımı olmadığından non parametrik sınıflandırıcılardır. Eğitim setinde girdi ve çıktılar eşlenmekte ve bu eşleşme aracılığıyla test setini sınıflandıracak karar fonksiyonları elde edilmektedir.

DVM sınıflandırıcılarında kullanılan girdi verileri doğrusal olarak ayrılabilirliğinde verileri ayırabilen birçok sayıdaki doğru arasından marjini en yüksek olan doğruyu seçmek hedeflenmektedir. Kullanılan verilen doğrusal olarak ayrılmıyor ise, veriler doğrusal olmayan haritalama kullanılarak yüksek boyuta çevrilmekte ve yeni boyutta marjini yüksek olan doğruyu bulmak amaçlanmaktadır [63-64].

DVM sınıflandırıcısının avantajları;

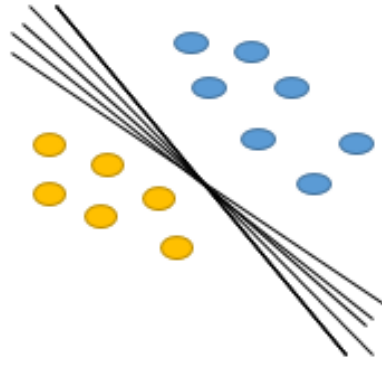
- Karmaşık karar sınırlarını modelleyebilmektedir.
- Çok sayıda bağımsız değişkenle çalışabilmektedir.
- Yüksek doğruluğa sahiptir.
- Hem doğrusal ayrılabilen verilere hem de doğrusal ayrılamayan verilere uygulanabilmektedir.

Literatürde DVM doğrusal ve doğrusal olmayan olmak üzere ikiye gruba ayrılmaktadır. Doğrusal DVM de veriler tamamen doğrusal olarak ayrılabiliriyorsa bu modele hard margin, az sayıda noktadan dolayı veriler tamamen ayrılmıyorsa bu modele de soft margin denilmektedir. Doğrusal olmayan DVM de ise verilerin tamamen ayrılması imkânsızdır.

1.5.1.3.1. Doğrusal Destek Vektör Makineleri

1.5.1.3.1.1. Hard Marjin

En temel destek vektör makinesi uygulaması olan hard marjinde bir veri seti ayırabilecek sonsuz sayıda doğru bulunabilmektedir. Amaç maksimum marjinli doğruyu bulabilmektir. Bilinmeyen bir veri seti ile karşılaşıldığında sınıflama hatasını minimum yapan doğru bulunmaktadır.



Şekil 1.13. Doğrusal Ayrılabilen Veriler

Eğitim seti: $\{x_i, y_i\} \quad i = 1, 2, 3, \dots, L$

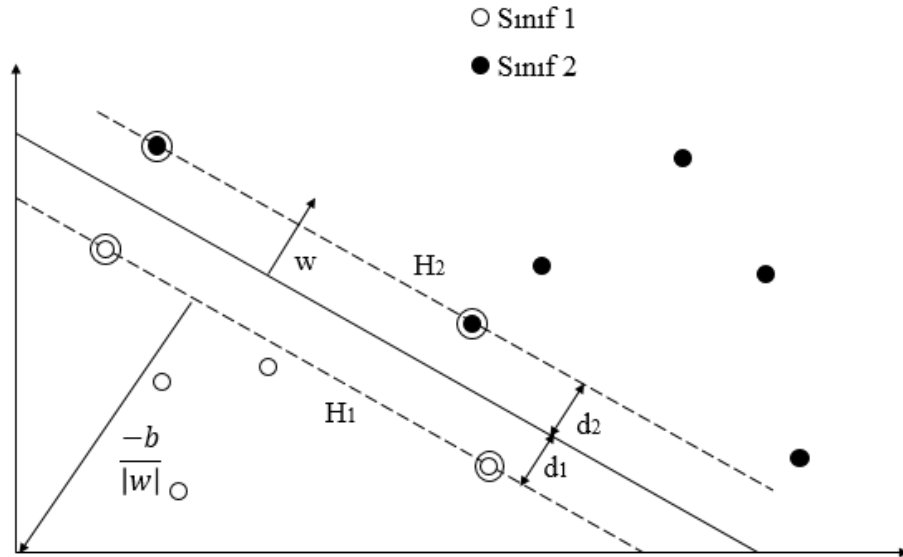
Sınıf etiketleri: $y_i \in \{-1, +1\}$

Ayrıcı düzlem: $\langle w, x \rangle + b = 0$

w : çoklu düzlemin normali (ağırlık vektörü)

b : bias

x : $\langle w, x \rangle + b = 0$ çoklu düzlem üzerindeki herhangi bir noktayı temsil etmektedir.



Şekil 1.14. İki doğrusal ayrılabilen sınıf aracılığıyla hiper düzlemin gösterimi

Kesikli çizgilerle gösterilen ve ayırıcı çoklu düzleme paralel olarak çizilmiş eşit uzaklıktaki iki çoklu düzlem (doğru) arasındaki uzaklığa marjin denilmektedir. Bu kesikli çizgiler şu şekilde ifade edilmektedir:

$$y_i = +1 \text{ için } \langle w, x_i \rangle + b = +1 \quad (1.57)$$

$$y_i = -1 \text{ için } \langle w, x_i \rangle + b = -1 \quad (1.58)$$

Destek vektörlerinin ayırıcı çoklu düzleme olan uzaklığı $1/\|w\|$ olarak hesaplanmakta olup dolayısıyla marjin $\rightarrow 2/\|w\|$ olarak bulunmaktadır.

Marjin maksimum seçildiğinde yeni gelen verileri sınıflandırırken ortaya çıkacak olan hata minimize edilmektedir. En iyi çoklu düzlemi bulmak için kullanılan nesne fonksiyonu ve kısıt formülleri şu şekildedir:

$$\min_{w,b} \tau(w) = \frac{1}{2} \|w\|^2 \quad (1.59)$$

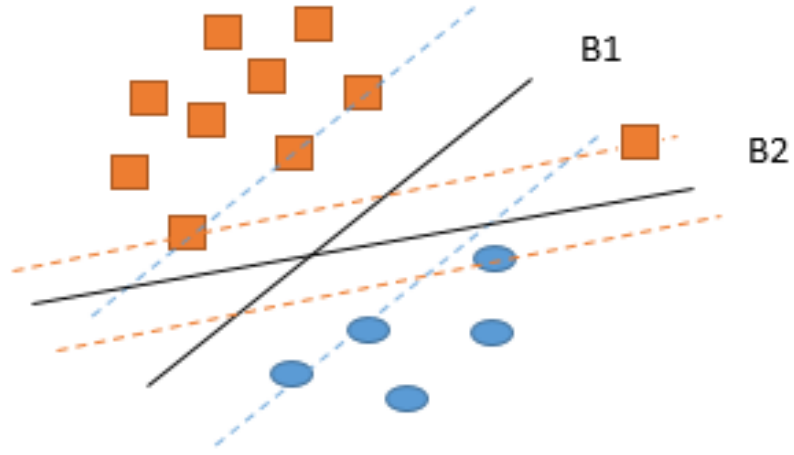
$$y_i (\langle x_i, w \rangle + b) - 1 > 0 \quad (1.60)$$

Karar fonksiyonu ise;

$$f(x) = \left(\sum \alpha_i y_i x_i \cdot x \right) + b \quad (1.61)$$

1.5.1.3.1.2. Soft Marjin

Gerçek hayatta veriler tamamiyle doğrusal ayıramamaktadırlar. Şekilde de görüldüğü gibi B1 düzlemi noktaları tamamiyle ayıramazken, B2 düzlemi noktaları tam olarak ayırmaktadır. Ancak B2 doğrusunun sahip olduğu marjin daha düşüktür. Soft marjin bu tip uygulamalarda hataları tolere etmek amacıyla kullanılmaktadır.



Şekil 1.15. Doğrusal ayrılamayan verilerde hiper düzlemin gösterimi

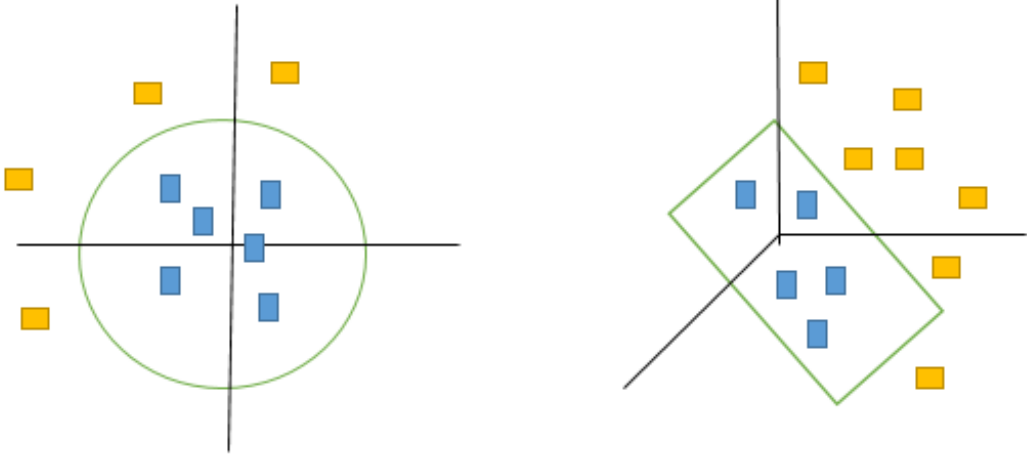
Soft marjinde minimizasyonu sağlamak için elde edilen kısıtlara slack değişkeni eklenmektedir. Slack değişkeni negatif olmayan bir değişken olup kullanıcı tarafından belirlenmektedir.

$$y_i = +1 \text{ için } \langle w, x_i \rangle + b \geq +1 - C \quad (1.62)$$

$$y_i = -1 \text{ için } \langle w, x_i \rangle + b \geq -1 + C \quad (1.63)$$

1.5.1.3.2. Doğrusal Olmayan Destek Vektör Makineleri

Doğrusal olmayan destek vektör makinelerinde veriler doğrusal olarak ayrıştırılmadığında, verilere haritalama yapılmakta ve orijinal veri uzayından daha yüksek boyutlu uzaya geçiş yapılmaktadır. Daha yüksek boyuttaki uzayda verileri en iyi şekilde ayrıştırabilecek doğru araştırması yapılmaktadır.



Şekil 1.16. Doğrusal ayrılamayan veriler

Burada $\Phi(x)$ kullanılarak yeni daha yüksek boyutlu uzaya geçilmektedir. Doğrusal destek vektörlerinden farklı olarak $\Phi(x)$ kullanılmıştır. Dönüştürülmüş uzaydaki karar fonksiyonu:

$$\langle w, \Phi(x) \rangle + b = 0 \quad (1.64)$$

Kullanılan nesne fonksiyonu ve kısıt formülleri şu şekildedir:

$$\min_{w,b} \tau(w) = \frac{1}{2} \|w\|^2 \quad (1.65)$$

$$y_i (\langle x, w \rangle + b) - 1 > 0 \quad (1.66)$$

Doğrusal olmayan destek vektör makinelerinde dönüştürülmüş uzayda oluşturulacak doğrusal karar sınırı ile ilgili nasıl bir haritalama yapılacağı açık değildir. Doğrusal olmayan destek vektör makinelerinde w ve b parametreleri şu şekilde hesaplanmaktadır:

$$W = \sum \alpha_i y_i \Phi(x_i) \quad (1.67)$$

$$f(x) = (\sum \alpha_i y_i \theta(x_i) \Phi(x_i)) + b = 0 \quad (1.68)$$

Oluşturulan formüllerde boyut sorunundan dolayı hesaplama zorlaşmaktadır. Bu sorunu önlemek için literatürde ‘çekirdek düzenlemesi’ önerilmektedir.

1.5.1.3.2.1. Çekirdek Düzenlemesi

Çekirdek düzenleme ile dönüştürülmüş uzaydaki $\theta(x)$ yerine veri uzayından elde edilmiş bir çekirdek fonksiyonu ile işlemler yapılmaktadır. Bu fonksiyon yardımıyla $\theta(x)$ fonksiyonunun ne olduğu bilmeye gerek kalmadan işlemler yapılabilir. Yapılan işlemlerin karmaşıklığı ve maliyeti daha düşük olmaktadır. Çekirdek fonksiyonları:

1. Doğrusal Fonksiyon

$$K(x_i, x_j) = x_i^T x_j \quad (1.69)$$

2. Polimiyal Fonksiyon

$$K(x_i, x_j) = (x_i x_j)^d \quad (1.70)$$

3. Sigmoid Fonksiyonu

$$K(x_i, x_j) = \tanh(kx_i x_j - \delta) \quad (1.71)$$

4. Radyal Tabanlı Fonksiyon

$$K(x_i, x_j) = \exp(-\gamma \|x_{(i)} - x_j\|^2) \quad (1.72)$$

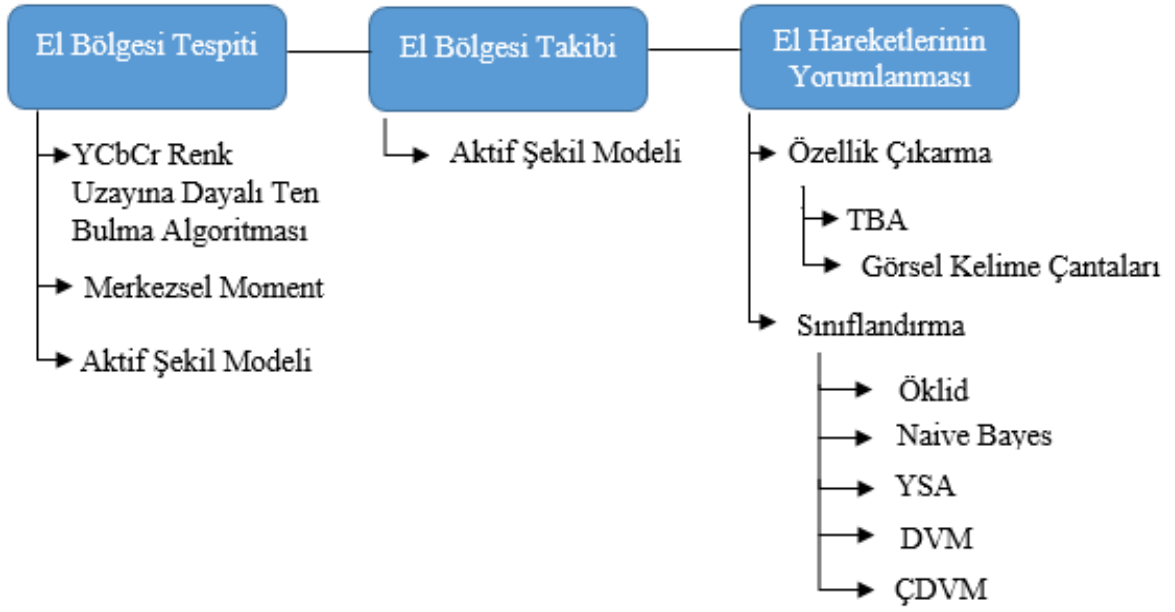
2. YAPILAN ÇALIŞMALAR, BULGULAR VE İRDELEME

2.1. Giriş

Bu tez çalışmasında kullanıcının hiçbir cihaza fiziksel temasta bulunmadan el hareketleriyle bilgisayarı yönetmesi ve kontrol etmesi sağlanmaktadır. Yapılan çalışmada görüş alanına giren bilgisayar kullanıcısının görüntülerinin kamera ile bilgisayara aktarımı sağlanarak bilgisayarı el hareketleri ile yönetmesi ve kontrolü amaçlanmaktadır.

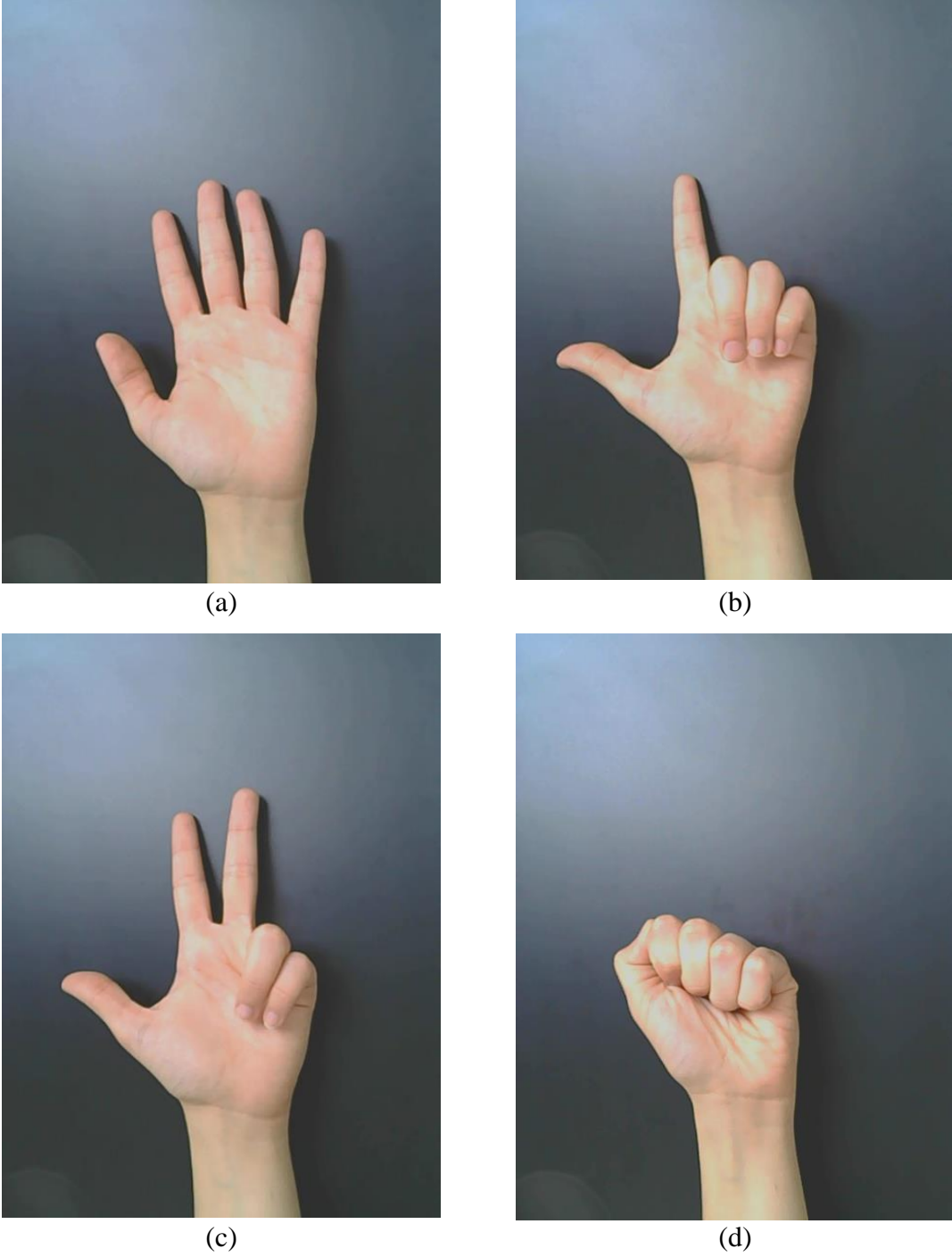
Yapılan tez çalışması el bölgesi tespiti, el bölgesi takibi ve el hareketlerinin yorumlanması ile bilgisayar kontrolü olmak üzere üç işlem adımından oluşmaktadır. Tez çalışmasında geliştirilen yaklaşımda ilk olarak bilgisayar kontrol edecek kişinin kamera görüş alanında belirlenen hedef çerçeveye girmesi beklenmektedir. Kamera görüş alanına giren kişinin YCbCr renk uzayına dayalı ten bulma algoritması kullanılarak kullanıcının hedef çerçevedeki ten bölgeleri ayırt edilmektedir. Ten rengi bulma algoritmasında ilk olarak RGB renk uzayından YCbCr renk uzayına dönüşüm yapılmakta daha sonra ise Cb ve Cr bileşenlerinin maksimum ve minimum değerleri hesaplanarak bu değerler arasında kalan pikseller ten rengi olarak işaretlenmektedir. Hedef çerçevede bulunan ten bölgesinin merkezci momentler kullanılarak orta noktası bulunmaktadır. Orta noktası bulunan ten bölgesine aktif şekil modeli kullanılarak belli bir iterasyonda modelin oluşup oluşmadığı denetlenmektedir. El bölgesi modeli konumlandırıldıktan sonra tekrar aktif şekil modeli kullanılarak el bölgesi takibi başlatılmaktadır. El bölgesi takibi esnasında ilk olarak el bölgesi özellikleri çıkarılmakta, daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile bilgisayar kontrolü gerçekleştirilmektedir.

Yapılan çalışmada el bölgesi özellikleri çıkarılması için temel bileşen analizi Temel Bileşen Analizi Öz El ve Görsel Kelime Çantaları yöntemleri kullanılmaktadır. El bölgesi özelliklerinin sınıflandırılması için ise Öklid uzaklığı, YSA, DVM ve ÇDVM gibi farklı algoritmalar kullanılmakta ve sonuçları karşılaştırılmaktadır.



Şekil 2.1. Sistemin temel işlem adımları

El hareketleri ile bilgisayar kontrolünü ve yönetimini amaçlayan bu çalışmada kullanılan algoritmaların performans değerlendirmesi 60 farklı kişiden alınmış görüntüler üzerinde gerçekleştirilmiştir. Oluşturulan veri tabanında 4 farklı sınıfa ait görüntüler bulunmaktadır. Veritabanındaki her kişinin 4 farklı sınıfa ait örnekleri gerçekleştirilen çalışmada gerçek zamanlı uygulamalarda kullanılmıştır.

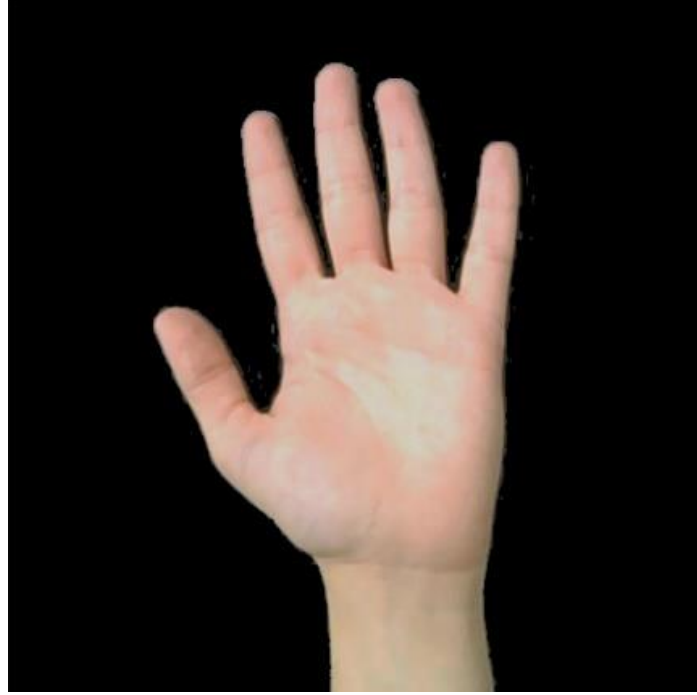


Şekil 2.2. Veritabanındaki 4 farklı sınıfa ait görüntüler. (a) Mouse serbest hareketi, (b) Mouse sol tıklama hareketi, (c) Mouse sağ tıklama hareketi, (d) Mouse taşıma hareketi

Şekil 2.2' de oluşturulan veri tabanında bir kişiye ait 4 farklı görüntü bulunmaktadır. Yapılan çalışmada ilk olarak bilgisayar kontrolünü yapacak kişinin kişinin kamera görüş alanında belirlenen hedef çerçeveye girmesi beklenmektedir. İlk eşleme ve mouse sabit konumdayken elin serbest dolaşmasında takibin devamlılığı için şekil 2.2 (a) görüntüsü

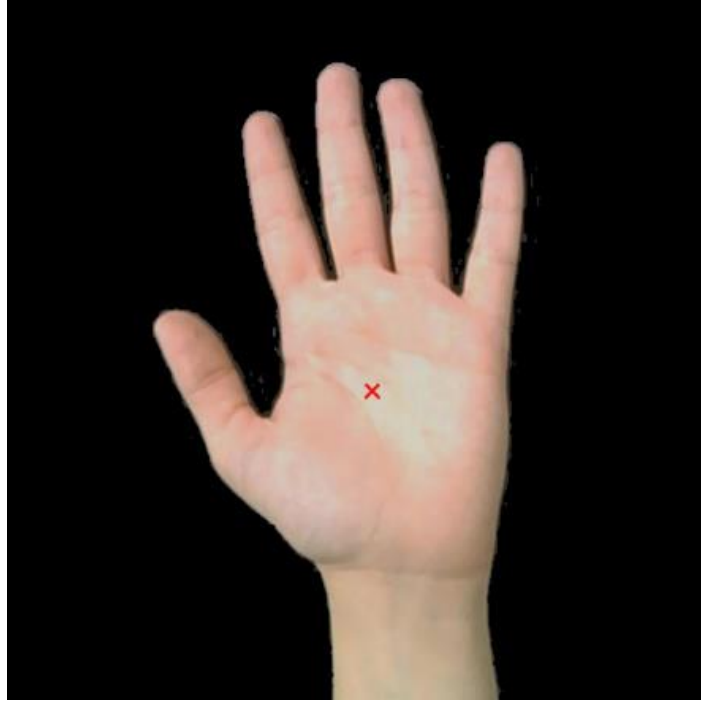
kullanılmaktadır. Oluşturulan sistemde mouse tek tıklama için şekil 2.2 (b), mouse sağa tıklama için şekil 2.2 (c), mouse konumu değiştirmek için şekil 2.2 (d) görüntüleri kullanılmaktadır.

Kamera görüş alanına giren kişinin sistemle ilk senkronizasyonu için şekil 2.2 (a) görüntüsü kullanılmaktadır. Hedef perncereye giren kullanıcının YCbCr renk uzayına dayalı ten bulma algoritması kullanılarak ten bölgeleri ayırt edilmiştir. Ten rengi bulma algoritmasında ilk olarak RGB renk uzayından YCbCr renk uzayına dönüşüm yapılmakta daha sonra ise Cb ve Cr bileşenlerinin maksimum ve minimum değerleri hesaplanarak bu değerler arasında kalan pikseller ten rengi olarak işaretlenmiştir. Yapılan çalışmalarda YCbCr renk uzayında $Y[0,255],Cb[0,255],Cr[0,255]$ olmak üzere ten rengi için Y değer aralığının $[0,255]$, Cb değer aralığının $[80,135]$ ve Cr değer aralığının ise $[131,185]$ arasında seçildiğinde başarılı ten rengi filtrelemesi yapılabildiği tespit edilmiştir.



Şekil 2.3. Ten bölgesi ayırt edilen görüntü

Hedef çerçevede bulunan ten bölgesinin orta noktası bulunmaktadır. Ten rengine ait renk küme dağılımının merkez noktasını bulmak için Merzcil Momentler kullanılmaktadır.

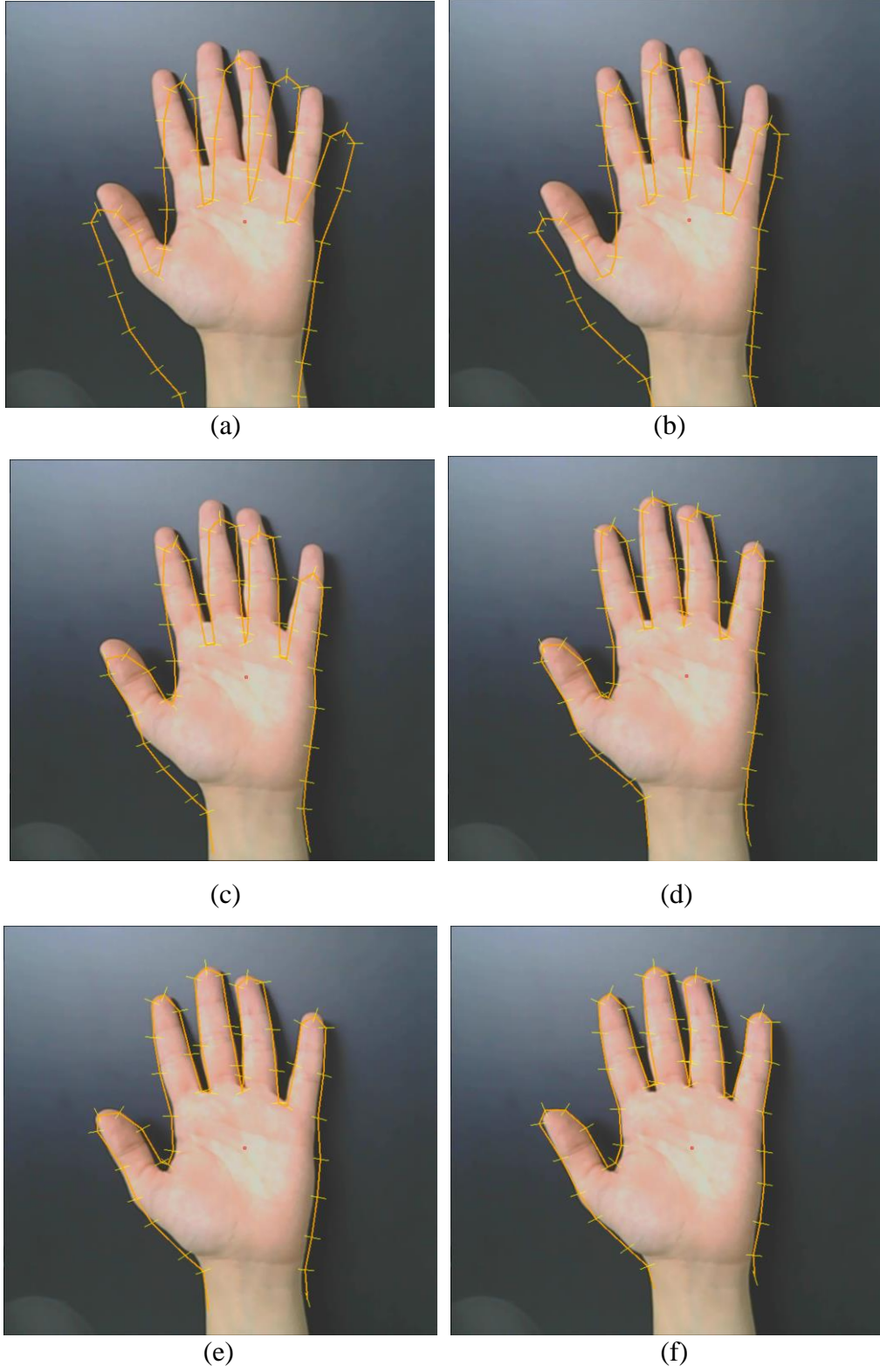


Şekil 2.4. Merkezsel moment kullanılarak orta noktası bulunmuş görüntü

Orta noktası bulunan ten bölgesine Aktif Şekil Model belli bir iterasyon sayısı aşılmadığı sürece yerleştirilmeye çalışılmaktadır. Eğer yakınsama sağlanırsa AŞM takibi başlatılmaktadır. Yakınsama olmaması durumunda hedef çerçeveye giren nesnenin el eşleme görüntüsü olmadığı varsayılarak yeni nesneler için beklemeye geçilmektedir.

Yapılan çalışmada aktif şekil modelinin şekil 2-2 (a) eşleme sınıfı görüntülerinde başlangıç eşlemesi için 50 iterasyondan daha az iterasyonda model oturabildiği tespit edilmiştir. Ayrıca gerçek zamanlı takip esnasında kişinin kameradan bir görüntü gelme süresinde elinin konumunu çok fazla değiştiremeyeceğinden bir sonraki model eşleme süresi daha da kısa sürmektedir.

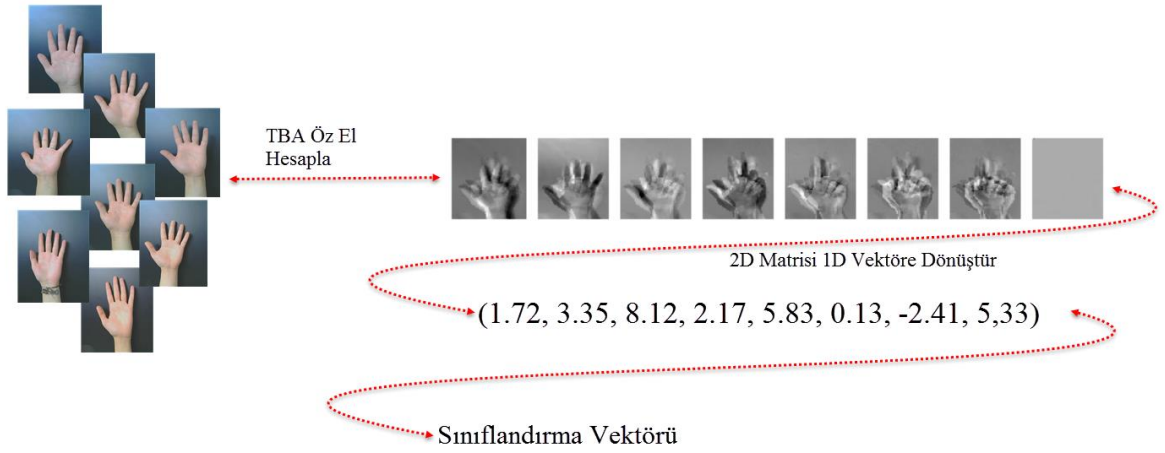
Takip esnasında 4 farklı sınıf için takip çerçevesinin sınıflandırma sonucuna göre de hedef model eğitilen modeller arasından tekrar güncellenmektedir ve yeni takip bu model ile devam ettirilmektedir.



Şekil 2.5. AŞM başlangıç eşleme işlemi iterasyonları. (a) Model oturtma, (b) 5. iterasyon, (c) 10. iterasyon, (d) 15. iterasyon, (e) 20. iterasyon, (f) 22. iterasyon model hedeflenen bölgeye yakınsadı.

Çalışmada takip sırasında kameradan gelen görüntülerde takip çerçevesinin sınıflandırılması işleminde öncelikle özellik çıkarma yöntemi olarak TBA Öz El özellik çıkarma yöntemine yoğunlaşmıştır. Daha sonra Görsel Kelime Çantaları yönteminin gerçek zamanlı deneysel sonuçlarının daha tatminkâr olduğu görülerek bu yönetime yoğunlaşmıştır.

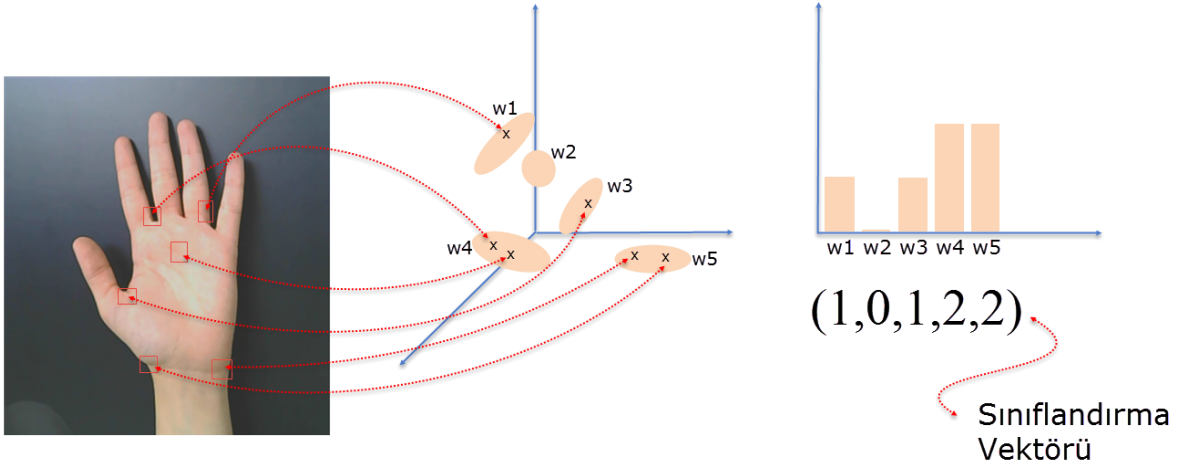
240 el görüntüsü içeren veritabanımızda 4 sınıf için toplam 168(%70) eğitim örneği kullanılmıştır. Dolayısıyla TBA 168 görüntünün özvektörlerinden oluşturulan eğitim matrisinin boyutu da 168 olmaktadır. TBA özellik çıkarma adımları Şekil 2-6 daki gibidir.



Şekil 2.6. TBA Öz El Özellik Vektörü Oluşturulması

Yapılan deneysel çalışmalarda TBA Öz El yönteminde görüntünün özvektörleri seçildiği için görüntü ebatları değişse de özvektörlerin değişmeyeceği mantığı ile orijinal görüntü daha hızlı sonuç elde edebilmek için küçültülmüştür.

Görsel Kelime Çantaları yönteminde ise SURF özellik çıkarma yöntemi ile eğitim görüntülerinden elde edilen özelliklerin istenen sayıda kümelere ayrılarak kümelenen bu özelliklerin sınıflandırılmasına dayalı bir yöntemdir. Bu çalışmada özellikleri kümeleme işlemi KMeans kümeleme yöntemi ile gerçekleştirilmiştir. Daha sonra kümelenen bu özelliklerin oluşturduğu kümelerdeki özellik sayılarından bir vektör oluşturulmaktadır. Bu vektör bir anlamda benzer özellikleri kümeleme yaptığı için özellik sınıflandırmada oldukça başarılı sonuçlar üretmektedir. Bu yöntemde dikkat edilmesi gereken temel nokta belirli bir kümeleme sayısından sonra sıfır elemanlı küme sayısı artmaktadır. Bunun nedeni küme sayısının SURF ile elde edilen özellik sayısını geçebilmesidir.



Şekil 2.7. Görsel Kelime Çantaları yöntemi ile özellik vektörü oluşturulması.

Daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile bilgisayar kontrolü gerçekleştirilmiştir. Yapılan tez çalışmasında sınıflandırma algoritması olarak Öklid, Bayes, YSA, DVM ve ÇDVM kullanılarak bu yöntemlerin sınıflandırma başarı istatistikleri elde edilmiştir.

Sınıflandırma yöntemlerinden Öklid ve Bayes eğitim setinden gelen özellik vektörüne göre sınıfları kümelendirmektedir ve yeni bir görüntü geldiğinde hangi kümeye yakın ise görüntüyü o kümenin etiketiyle sınıflandırmaktadır. Her iki yöntem de algoritma karmaşıklığı az olmasından dolayı gerçek zamanlı uygulamalarda etkili sonuçlar vermektedir.

Yapay Sinir ağları, Destek Vektör Makineleri ve Çekirdek Destek Vektör Makineleri ise bu çalışmada birer danışmanlı öğrenme yöntemi olarak kullanılmıştır ve görüntü sınıflandırma anlamında başarılı sonuçlar vermiştir.

Çalışmada kullanılan YSA modeli ileri beslemeli hata geri yayılım algoritması ile öğrenme işlemini gerçekleştirmektedir. YSA girişi olarak özellik çıkarma yöntemleri ile elde edilen özellik vektörleri kullanılmıştır ve giriş sayısı özellik vektörlerinin boyu kadardır. Ara katmanda giriş sayısının yarısı kadar sinir hücresi oluşturulmuştur ve Çıkış katmanında 4 sınıfı temsilen 4 çıkış verilmiştir. YSA çıkışları 0-1 aralığında değerlerdir ve her bir sınıfı temsilen ilgili çıkış 1'e yakınsamaktadır.

DVM ler sınıflandırma performansları en iyi sınıflandırıcılardandır. Destek Vektör Makineleri'nin en önemli avantajı, sınıflandırma problemini kareli optimizasyon problemine dönüştürüp çözmesidir. Böylece problemin çözümüne ilişkin öğrenme

aşamasında işlem sayısı azalmakta ve diğer teknik/algoritmalara göre daha hızlı çözüme ulaşılmaktadır. Teknik bu özelliğinden dolayı, özellikle büyük hacimli veri setlerinde büyük avantaj sağlamaktadır. Ayrıca optimizasyon temelli olduğundan sınıflandırma performansı, hesaplama karmaşıklığı ve kullanılabilirlik açısından diğer tekniklere göre daha başarılıdır. Çalışmada öncelikle sınıf sayısının 2 den fazla olması dolayısı ile Çok Sınıflı Destek Vektör Makineleri üzerine durulmuştur ve daha sonra özellik vektörünün uzayda lineer ayrışmamış olabileceği düşünülerek çekirdek fonksiyonu ile desteklenen Çekirdek Destek Vektör Makineleri üzerine yoğunlaşmıştır. ÇDVM kullanılırken farklı çekirdek fonksiyonları kullanılarak sınıflandırma başarıları takip edilmiştir. ÇDVM nin başarı istatistiklerinin oldukça yüksek olduğu yapılan deneysel çalışmalarda tespit edilmiştir.

2.2. Deneysel Sonuçlar ve Karşılaştırma

Çalışmada el bölgesi özelliklerinin çıkarılması için TBA Öz El, Görsel Kelime Çantaları ve çıkarılan bu özelliklerin sınıflandırılması için Öklid, Bayes, YSA, DVM ve ÇDVM yöntemleri karşılaştırılmıştır. Her bir yönteme özgü çeşitli parametrelerin değişkenliğine göre de sınıflandırmadaki doğruluk değerleri ve algoritma karmaşıklığından kaynaklanan hesaplama süresi farklılıkları hesaplanmıştır.

Çalışmada deneysel sonuçlar Intel(R) Core(TM) i7-4550 CPU @ 1.80GHz 2.40GHz işlemci, 4GB DDR3 ram bulunduran bilgisayar ile elde edilmiştir.

2.2.1. Kullanılan Özellik Çıkarma Yöntemlerinin Deneysel Hız Sonuçları

Çalışmada eğitimde kullanılan veri setinden geleneksel sınıflandırma yöntemleri ile sınıflandırma yaparken başarıya etki eden etkenleri(ölçek, konum, rotasyon v.b.) ortadan kaldırmak için özellik çıkarılarak bu özelliklerin sınıflandırılması amaçlanmıştır. Bu bağlamda 2 çeşit özellik çıkarma yöntemi üzerinde durulmuştur. Bu yöntemlerden biri eğitim setinde kullanılacak görüntülerin TBA ile boyut indirgenerek elde edilen 2 boyutlu Öz El görüntü matrislerinin TBA uzayına indirgenmiş 1 boyutlu vektörünün sınıflandırılması diğeri ise Görsel Kelime Çantaları yöntemi ile eğitim seti görüntülerinden elde edilen ortak özellik bölgelerinin sınıflandırılmasıdır.

Özellik çıkarma yöntemini seçerken dikkat edilmesi gereken birinci husus zaman diğeri sınıflandırma için yeterli özellik sayısının tespitidir. İnsan gözünün saniyede 24 adet

görüntüye kadar görüntü akışının sürekliliğindeki kopmaları algılayamadığını düşündüğümüzde seçilecek yöntemin saniyede en az 24 görüntüde özellik çıkarabilecek yetenekte olmalıdır. Yani gerçek zamanlı bir görüntü için özellik çıkarma süresi 0,04 saniyeyi geçmemelidir. Bu süre aşımı problemi görüntü atlatma yöntemi ile bir saniyedeki algoritmanın işleyebileceği görüntü sayısı belirlenip sadece bu sayının katları şeklinde gelen görüntüler üzerinde sınıflandırma işlemi gerçekleştirilerek aşılabılır. Ayrıca elde edilen özellik vektörünün eğitim setindeki görüntü sınıflarının doğru sınıflandırılabilmesi için yeterli özellik sayısına sahip olması gerekmektedir.

2.2.1.1. TBA Öz El Özellik Çıkarma Sonuçları

TBA görüntü setinden özellik çıkarma amacıyla kullanıldığında zamandan kazanç elde etmek için her bir görüntü için özellik çıkarma yapmadan önce görüntüden kabul edilebilir zamanda özellik çıkarabilmek için görüntünün yeniden boyutlandırılması gerekmektedir. Yapılan çalışmada gerçek zamanlı özellik çıkarma sürelerinin görüntü ebatlarına göre TBA hesaplama süreleri Tablo 2.1 deki gibidir.

Tablo 2.1. TBA Öz El Hesaplama Süreleri

Görüntü Boyutu (pixel)	Eğitim Görüntülerinden TBA Öz El Hesaplama Ortalama Süresi (Toplam Süre / Eğitim Görüntü Sayısı) (saniye)	Test Görüntülerinden TBA Öz El Özellik Çıkarma Ortalama Süresi (Toplam Süre / Test Görüntü Sayısı) (saniye)
100x100	0,085	0,039
200x200	0,340	0,164
300x300	1,570	0,903

TBA Öz El özellik çıkarma yöntemi kullanılması durumunda gerçek zamanlı donmaların önüne geçebilmek için 100x100 boyutlu görüntüler haricinde görüntü atlatma yöntemi kullanılması gerektiği yapılan deneysel sonuçlarda gözlemlenmiştir. Bu yöntem ile elde edilen özellik vektörünün sınıflandırılmasında özellik sayısını etkileyen faktör TBA Öz El eğitim setindeki görüntü sayısıdır. Eğitim setindeki görüntü sayısı kadar özellik içeren bir vektör oluşturulmaktadır ve her 2 boyutlu test görüntüsü bu sayıda özellik içeren

vektör ile temsil edilmektedir. Bununla birlikte TBA Öz El yönteminde görüntü ebatlarının 2 katına çıkması özellik vektörünün oluşturulma süresini 4 kat etkilemektedir.

2.2.1.2. Görsel Kelime Çantaları Özellik Çıkarma Sonuçları

GKÇ ile özellik çıkarma uygulanırken resim ebatları ve özellik küme sayısı gerçek zamanlı çalışma süresinde ve güçlü özellik vektörü oluşturmada etkili olmaktadır. Çalışmada GKÇ kümeleme yöntemi olarak KMeans kümeleme algoritması kullanılmıştır ve her bir özellik SURF yöntemi ile hedef çerçeveden çıkarılmıştır. Çalışmada öncelikle görüntü ebatlarının GKÇ özellik çıkarma süresine etkisi üzerinde durulmuştur ve Tablo 2.2 deki sonuçlar elde edilmiştir.

Tablo 2.2. GKÇ Özellik Çıkarma Süreleri

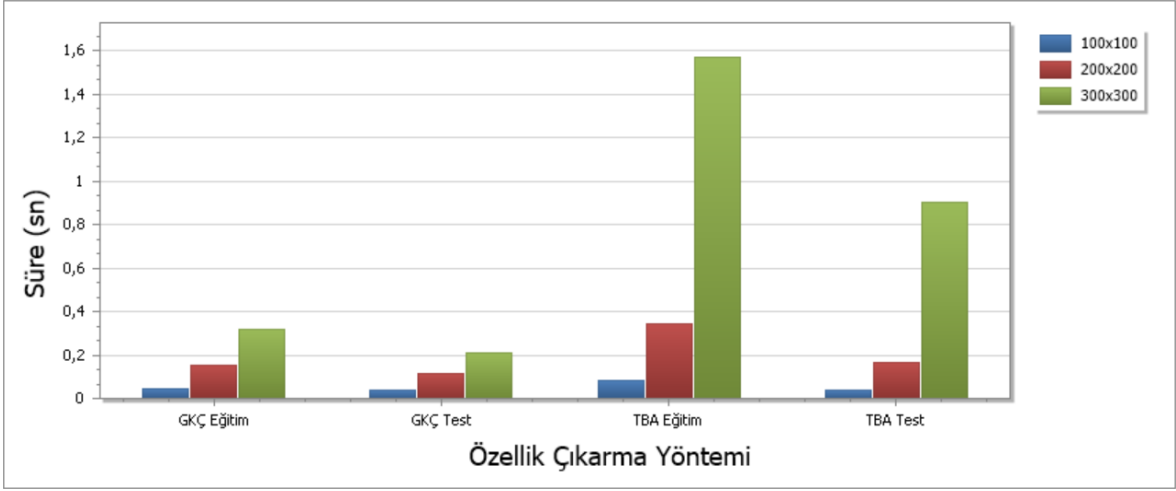
Görüntü Boyutu (pixel)	Eğitim Görüntülerinden GKÇ Özellik Çıkarma Ortalama Süresi (Toplam Süre / Eğitim Görüntü Sayısı) (saniye)	Test Görüntülerinden GKÇ Özellik Çıkarma Ortalama Süresi (Toplam Süre / Test Görüntü Sayısı) (saniye)
100x100	0,043	0,036
200x200	0,154	0,115
300x300	0,317	0,207

GKÇ yönteminde geçek zamanlı tepki alabilmek için 100x100 lük görüntü boyutlarından büyük boyutlu görüntülerde bir saniyede kameradan gelen görüntü lere görüntü atlatma yöntemi kullanılması gerekmektedir. Bununla birlikte görüntü ebatlarının 2 katına çıkması GKÇ özellik vektörü oluşturma süresinin de 2 katına çıkmasına sebep olmaktadır.

2.2.2. TBA ve GKÇ Özellik Çıkarma Sonuçlarının Karşılaştırılması

TBA da görüntü ebatlarında yapılan değişikliklerin hesaplama süresine oldukça fazla yük getirdiği GKÇ nin ise TBA ya oranla daha kabul edilebilir sürelerde hesaplama yapabildiği tespit edilmiştir. Her iki yöntemde de görüntü ebatlarının küçük seçilmesinin

gerçek zamanlı tepki açısından olumlu olacağı görülmüştür. Bununla birlikte TBA da özvektörlerin ölçekten bağımsız olacağı düşünüldüğünde teorik olarak büyük boyut seçmenin sınıflandırma başarısında artmaya sebep olmayabileceği düşünülmektedir. GKÇ de ise SURF ile bulunacak özellik sayısı kümeleme sayısının çok üzerinde seçilmediği durumlarda küçük boyutlar başarılı sonuç verebilecektir. Karşılaştırma toplamda 240 görüntü içeren veri tabanında 168(%70) eğitim ve 72(%30) test görüntüsü ile yapılarak toplam sürelerin ortalaması alınarak 1 görüntü için geçen süreler hesaplanmıştır.



Şekil 2.8. GKÇ-TBA Özellik Çıkarma Yöntemlerinin Farklı Boyutlardaki Görüntüler ile Eğitim ve Test Adınlarında Görüntü Boyutu(pixel) - Süre(saniye) karşılaştırmaları

TBA da eğitim seti görüntüsü kadar öz vektör oluşturulduğu için eğitim sayısında yapılacak oynamanın özellik çıkarma başarısını etkileyebileceği düşünülmektedir. GKÇ yönteminde ise görüntüden elde edilen SURF özellik sayısını aşmamak şartı ile kümeleme sayısında yapılacak değişim özellik çıkarma başarısını doğrudan etkileyeceği düşünülmektedir.

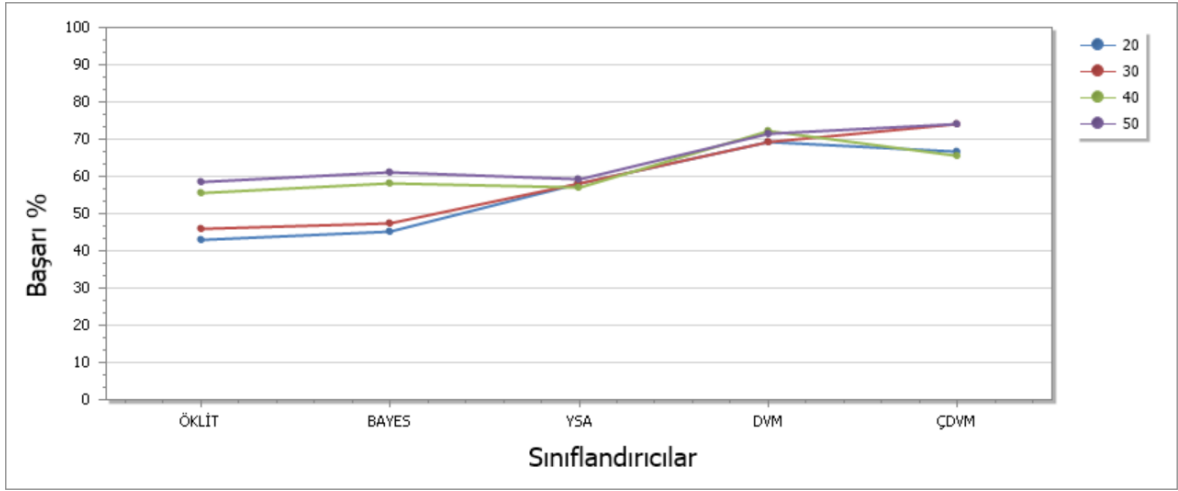
2.2.3. Sınıflandırıcıların Sınıflandırma Başarılarının Karşılaştırılması

Yapılan çalışmada Öklit, Bayes, YSA, DVM ve ÇDVM sınıflandırma yöntemleri ile denemeler yapılmıştır. Yöntemlerin gerçek zamanlı hesaplama süreleri ve sınıflandırma başarıları analiz edilmiştir. Elde edilen sonuçlarda bir görüntünün tüm yöntemlerde 0,0002

saniyenin altında sınıflandırılabilirdiği bu durumda gerçek zamanlı kullanımında zamanlama maliyetlerinin çok düşük olduğu görülmüştür ve sürelerle göre karşılaştırma yapmaya gerek duyulmamıştır.

Sınıflandırma başarıları kıyaslanırken 240 görüntüden oluşan veritabanındaki görüntülerin rasgele seçilen 168 i (%70) eğitim yine rasgele seçilen 72 si (%30) test sınıfı olarak gruplandırılmıştır.

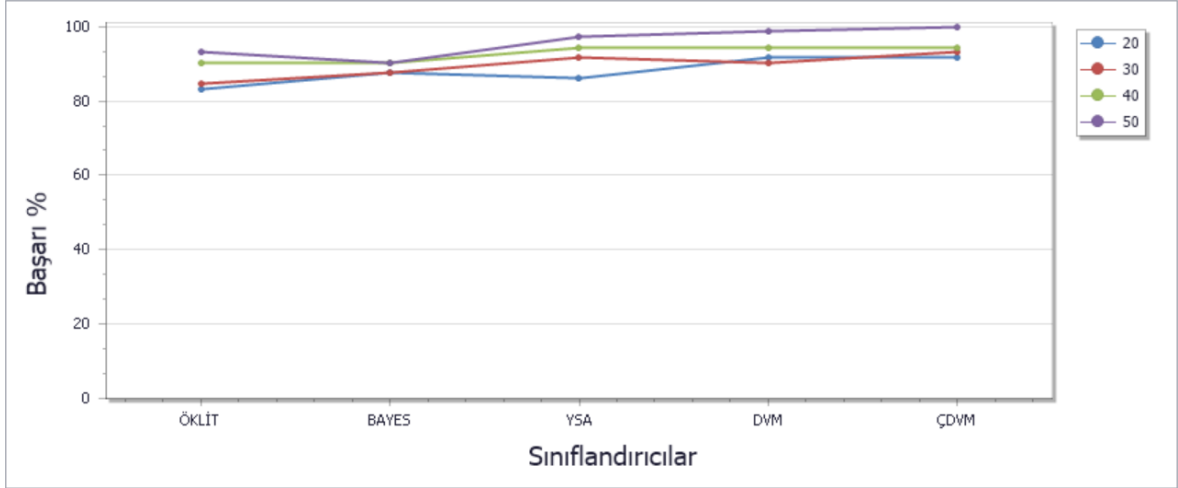
TBA da eğitim seti büyüklüğünde özellik vektörü oluşmaktadır ve bu vektör sınıflandırıcıların girişi olarak kabul edilmektedir. TBA öz vektöründen seçilen 20, 30, 40 ve 50 özelliğin Deneysel sonuçlarda başarı sınıflandırma yöntemlerinin başarı istatistikleri şekil 2.9 daki gibi elde edilmiştir.



Şekil 2.9. TBA Öz-El kullanılarak oluşturulan 20, 30, 40 ve 50 boyutlu özellik vektörlerinin sınıflandırılmasıyla elde edilen başarı grafiği

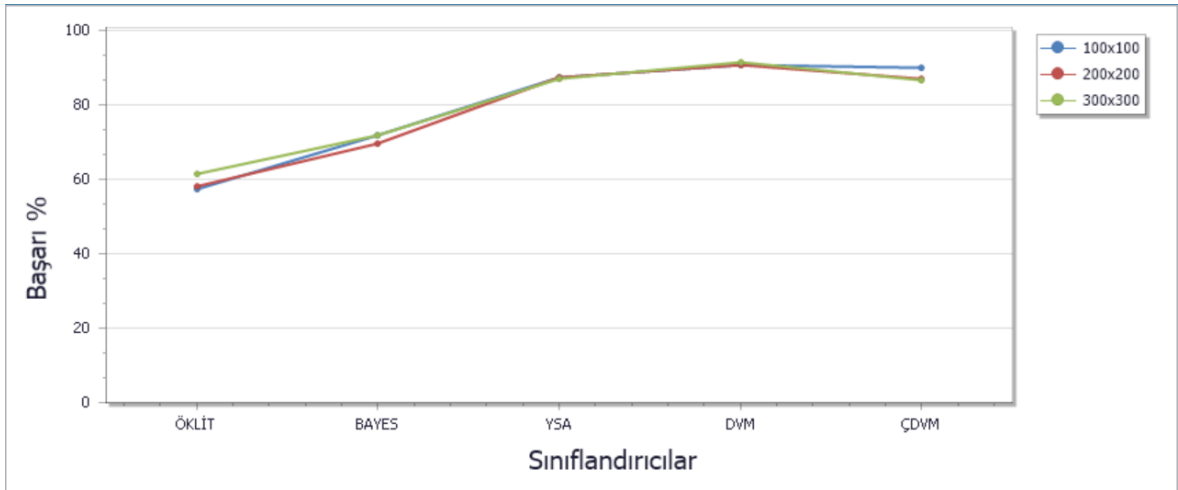
Deneysel sonuçlarında TBA ile yapılan sınıflandırma sonucunda en başarılı yöntemin %75 lerde kaldığı görülmüştür.

GKÇ de özellik vektörünün boyutu kümeleme sayısı kadardır. Bu yöntemde elde edilen özellik vektörü sınıflandırıcılara giriş olarak verilmektedir ve rasgele görüntülerden oluşturulmuş test verilerinin sınıflandırılması amaçlanmaktadır.



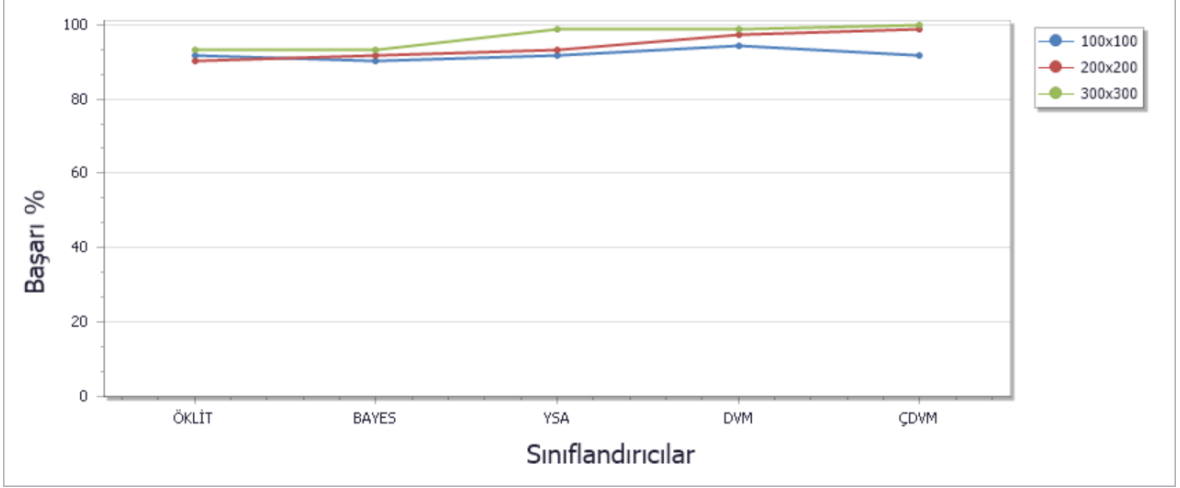
Şekil 2.10. GKÇ kullanılarak oluşturulan 20, 30, 40 ve 50 boyutlu özellik vektörlerinin sınıflandırılmasıyla elde edilen başarı grafiği

Ayrıca yapılan çalışmada 100x100, 200x200 ve 300x300 gibi farklı ebatlardaki görüntüler ile TBA ve GKÇ özellik çıkarma yöntemleri test edilmiştir. TBA Öz-El özellik çıkarma yöntemi ile elde edilen istatistiklerin grafiği şekil 2.11 deki gibidir.



Şekil 2.11. TBA Öz-El özellik çıkarma yöntemi kullanılarak 100x100, 200x200 ve 300x300 boyutlu görüntülerin sınıflandırılmasıyla elde edilen başarı grafiği

GKÇ özellik çıkarma yöntemi ile elde edilen sınıflandırılma başarıları istatistikleri şekil 2.12 deki gibi elde edilmiştir.



Şekil 2.12. GKÇ özellik çıkarma yöntemi kullanılarak 100x100, 200x200 ve 300x300 boyutlu görüntülerin sınıflandırılmasıyla elde edilen başarı grafiği

GKÇ ile çıkarılan özelliklerin sınıflandırma başarısının oldukça yüksek olduğu görülmektedir ve görüntü ebatları artmasına karşın özellik çıkarma süresinin TBA ya göre daha az olduğu görülmüştür.

Kullanılan sınıflandırıcılardan ÇDVM nin en yüksek sınıflandırma başarısı gösterdiği ve 300x300 lük görüntülerde SURF özelliklerinin 50 kümeye ayrılması ile elde edilen 50 elemanlı özellik vektörü kullanıldığında Test sınıfı sınıflandırma başarısının %100ü görüldüğü tespit edilmiştir.

3. SONUÇLAR VE TARTIŞMA

Çalışmamızda kullanıcının hiçbir cihaza fiziksel temasta bulunmadan el hareketleriyle bilgisayarı yönetmesi ve kontrol etmesi sağlanmaktadır. Yapılan çalışmada görüş alanına giren bilgisayar kullanıcısının görüntülerinin kamera ile bilgisayara aktarımı sağlanarak bilgisayarı el hareketleri ile yönetmesi ve kontrolü amaçlanmaktadır.

Yapılan çalışmada kullanıcının özellikle sunum anında bilgisayar ekranından uzakta rahatça hareket etmesi ve herhangi bir cihaza temas etmeden bilgisayarı kontrol etmesi sağlanmıştır. Çalışmada bilgisayarla görme ve görüntü işleme teknikleri kullanılarak kamera görüş alanına giren kullanıcının el bölgesinin yorumlanması ile hem bilgisayar kameraları daha fonksiyonel kullanılabilmesi hem de kullanıcının monitörden uzakta hiçbir fiziksel temasta bulunmadan bilgisayarı yönetmesi sağlanmıştır.

Yapılan tez çalışması el bölgesi tespiti, el bölgesi takibi ve el hareketlerinin yorumlanması ile bilgisayar kontrolü olmak üzere üç işlem adımından oluşmaktadır. El hareketleri ile bilgisayar kontrolünü ve yönetimini amaçlayan bu çalışmada kullanılan algoritmaların performans değerlendirmesi 60 farklı kişiden alınmış görüntüler üzerinde gerçekleştirilmiştir. Oluşturulan veri tabanında 4 farklı sınıfa ait görüntüler bulunmaktadır. Veri tabanındaki her kişinin 4 farklı sınıfa ait örnekleri gerçekleştirilen çalışmada gerçek zamanlı uygulamalarda kullanılmıştır.

Tez çalışmasında ilk olarak bilgisayar kontrol edecek kişinin kamera görüş alanında belirlenen hedef çerçeveye girmesi beklenmektedir. Kamera görüş alanına giren kişinin YCbCr renk uzayına dayalı ten bulma algoritması kullanılarak kullanıcının hedef çerçevedeki ten bölgeleri ayırt edilmektedir. Ten rengi bulma algoritmasında ilk olarak RGB renk uzayından YCbCr renk uzayına dönüşüm yapılmakta daha sonra ise Cb ve Cr bileşenlerinin maksimum ve minimum değerleri hesaplanarak bu değerler arasında kalan pikseller ten rengi olarak işaretlenmektedir. Hedef çerçevede bulunan ten bölgesinin merkezci momentler kullanılarak orta noktası bulunmaktadır. Orta noktası bulunan ten bölgesine aktif şekil modeli kullanılarak belli bir iterasyonda modelin oluşup oluşmadığı denetlenmektedir. El bölgesi modeli konumlandırıldıktan sonra tekrar aktif şekil modeli kullanılarak el bölgesi takibi başlatılmaktadır.

Çalışmamızda el bölgesi modeli konumlandırıldıktan sonra tekrar aktif şekil modeli kullanılarak el bölgesi takibi başlatılmaktadır. El bölgesi takibi esnasında ilk olarak el bölgesi özellikleri çıkarılmıştır. El bölgesi özellikleri çıkarılması için temel bileşen analizi

(TBA) ve görsel kelime çantası kullanılmıştır. Yapılan test işlemi sonucunda görsel kelime çantalarının temel bileşen analizine göre daha iyi sonuç verdiği görülmüştür. Görsel kelime çantası modelinde kümeleme ile özellik seçildiğinden süre olarak daha kısa olduğu gözlemlenmiştir. Ek olarak görsel kelime çantasında özellik sayısı artırıldığında başarının arttığı da görülmüştür.

Daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile bilgisayar kontrolü gerçekleştirilmiştir. El bölgesi özelliklerinin sınıflandırılması için ise Öklid uzaklığı, YSA, DVM ve ÇDVM gibi farklı algoritmalar kullanılmış ve sonuçları karşılaştırılmıştır. Kullanılan algoritmalar karşılaştırıldığında çekirdek vektör makinesinin daha iyi sonuç verdiği gözlemlenmiştir.

Tüm çalışma incelendiğinde kullanıcılardan alınan el bölgeleri yapılarının farklı olmaları, farklı renkte olmaları sistemin başarısını etkilemektedir. Bunun yanında bilgisayara görüntü aktarımını sağlayan kameranın çözünürlüğü, ışığın şiddeti de başarıyı etkileyen problemlerdendir.

Sonuç olarak önerilen yöntemler güncel yaklaşım olup el bölgesi tanıma ve sınıflandırma sistemlerinde kullanılmayan yaklaşımlar olduğundan ve diğer yöntemlerle kıyaslandığında performans açısından başarılı sonuçlar verdiği için değişik alanlardaki birçok probleme de genelleştirilebilir.

Çalışma sırasında geliştirilen yaklaşımın bir bölümünün yer aldığı 1 adet bildiri yayınlanmıştır. Bu bildiri 2014 yıllarında yapılan Sinyal İşleme ve İletişim Uygulamaları Kurultayında sunulmuş ve bildiri kitapçığında basılmıştır. Ayrıca çalışma konusu ile Bilim, Sanayi ve Teknoloji Bakanlığı Teknogirişim Sermayesi Desteği Programına başvurulmuştur ve çalışmamız 1094.TGSD.2014 numarası ile desteklenmeye değer bulunan projeler arasına girmiştir.

4. ÖNERİLER

Çalışmada 60 kişiden alınan 4 farklı sınıf görüntülerinden oluşan bir veritabanı kullanılmıştır. Kişi sayısı artırılarak daha başarılı sınıflandırma sonuçları elde edilebilir. Ayrıca sınıf sayısı artırılarak daha geniş bir veritabanı ile daha çok sınıf sınıflandırılarak bilgisayar kontrol komutları zenginleştirilebilir.

Bünyesinde derinlik sensörü bulunduran Kinect Kameralar kullanılarak da z ekseninde derinlik bilgisinin de yardımı ile daha hassas kontroller gerçekleştirilebilir. Böylece x-y eksen hareketlerine z derinlik eksen hareketi de eklenebilir ve kontrol bir anlamda 2D uzaydan 3D uzaya taşınabilir.

Özellik çıkarma yöntemi olarak farklı özellik çıkarma yöntemleri denenebilir ve gerçek zamanlı tepki süresi kısaltılabilir. Özellikle görüntü ebatlarının arttığı durumlarda daha hızlı çalışan bir özellik çıkarma yöntemi kullanılabilir.

Sınıflandırma yöntemi olarak diğer geleneksel sınıflandırma yöntemlerinden faydalanılabilir ve daha güçlü sınıflandırıcılar ile performans artışları gözlemlenebilir.

5. KAYNAKLAR

1. Murthy, G., R., S. ve Jadon R., S., A., Review Of Vision Based Hand Gestures Recognition. International Journal of Information Technology, 2, 2 (2009) 405-410.
2. Khan, R., Z. ve Ibraheem, N., A., Hand Gesture Recognition: A Literature Review. International Journal of Artificial Intelligence & Applications, 3, 4 (2012) 161-174.
3. Mehra, S., Verma, P., Bung, H. ve Bairagee, D., Survey Paper Based on Hand Gesture Hex Color Matrix Vector, International Journal of Computational Engineering Research, 3, 4 (2013) 39-42.
4. Khan, R., Z., ve Ibraheem, N., A., Survey on Gesture Recognition for Hand Image Postures, Computer and Information Science, 5, 3 (2012) 110-121.
5. Dogan, R., O. ve Kose, C., Computer Monitoring and Control with Hand Movements, 22nd Signal Processing and Communications Applications Conference (SIU) IEEE, Nisan 2014, Trabzon, Bildiriler Kitabı, 2110-2113.
6. Bolt, R., A., Put-that-there: Voice and Gesture at the Graphics Interface, ACM SIGGRAPH Computer Graphics, 14, 3 (1980) 262-270.
7. Starner T. ve Pentland, A., Real-Time American Sign Language Recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence, 20, 12 (1998) 1371–1375.
8. Ng C., W. ve Ranganath, S., Real-Time Gesture Recognition System and Application, Image and Vision computing, 20, 13 (2002) 993–1007.
9. Abe, K., Saito, H. ve Ozawa, S., Virtual 3-D Interface System via Hand Motion Recognition from Two Cameras, Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 32, 4 (2002) 536–540.
10. Hasanuzzaman, M., Ampornaramveth, V., Zhang, T., Bhuiyan, M. A., Shirai, Y. ve Ueno, H., Real-Time Vision-Based Gesture Recognition for Human Robot Interaction, In Robotics and Biomimetics, 2004. (ROBIO). IEEE International Conference on, Ağustos 2004, China, Bildiriler Kitabı, 413-418.
11. Segen, J., Controlling Computers with Gloveless Gestures, in Proc. Virtual Reality Systems Conf., 1993, New York, Bildiriler Kitabı, 66–78.
12. Kortenkamp, D., Huber, E. ve Bonasso, R. P., Recognizing and Interpreting Gestures on a Mobile Robot, Proceedings of the National Conference on Artificial Intelligence, 1996, Portland, Bildiriler Kitabı, 915–921.
13. Sánchez-Nielsen, E., Antón-Canalís, L. ve Hernández-Tejera, M., Hand Gesture Recognition for Human-Machine Interaction, Journal of WSCG, 12, 1-3 (2004) 395-402.

14. Zhou, H., Xie, L. ve Fang, X., Visual Mouse: SIFT Detection and PCA Recognition, Computational Intelligence and Security Workshops (CISW). International Conference on IEEE, Aralık 2007, Harbin, Bildiriler Kitabı, 263-266.
15. Bao, J., Song, A., Guo, Y. ve Tang, H., Dynamic Hand Gesture Recognition Based on SURF Tracking, Electric Information and Control Engineering (ICEICE). International Conference on IEEE, Nisan 2011, Wuhan, Bildiriler Kitabı, 338-341.
16. Alsheakhali, M., Skaik, A., Aldahdouh, M. ve Alhelou, M., Hand Gesture Recognition System, The International Conference on Information and Communication Systems (ICICS), Mayıs 2011, Irbid, Bildiriler Kitabı, 132-136.
17. Pansare, J., R., Gawande, S., H. ve Ingle, M., “Real-Time Static Hand Gesture Recognition for American Sign Language (ASL) in Complex Background”, Journal of Signal and Information Processing, 3, 3 (2012) 364-367.
18. Asaari, M., S., M. ve Suandi, S., A., Hand Gesture Tracking System Using Adaptive Kalman Filter, Intelligent systems design and applications (ISDA). International Conference on IEEE, Aralık 2010, Cairo, Bildiriler Kitabı, 166 -171.
19. Padam P., S. ve Bora, P., K., A robust static hand gesture recognition system using geometry based normalizations and Krawtchouk moments, Pattern Recognition, 46, 8 (2013) 2202-2219.
20. Cootes, T. ve Baldock, E., R. ve Graham, J., An introduction to active shape models, Image Processing and Analysis, (2000) 223-248.
21. Cootes, T., F., Taylor, C., J., Cooper, D., H. ve Graham, J., Active Shape Models – Their Training and Application, Computer Vision and Image Understanding, 61, 1 (1995) 38-59.
22. Cootes, T., F. ve Taylor, C., J., Statistical Models of Appearance For Computer Vision, Imaging Science and Biomedical Engineering, University of Manchester, Manchester M13 9PT, U.K., Manchester, 2001.
23. Liu, N. ve Lovell, B., C., Hand Gesture Extraction by Active Shape Models, Digital Image Computing: Techniques and Applications (DICTA). International Conference on IEEE, Aralık 2005, Queensland, Bildiriler Kitabı, 10-10.
24. Abdalla, S. ve Hemayed, E., E., Dynamic Hand Recognition of Arabic Sign Language using Hand Motion Trajectory Features, Global Journal of Computer Science and Technology Graphics and Vision, 13, 5 (2013) 26-33.
25. Liao, S., X., Image Analysis by Moments, A Thesis Submitted to the Faculty of Graduate Studies in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy, The Department of Electrical and Computer Engineering The University of Manitoba Winnipeg, Manitoba, Canada, 1993.

26. Burger, W. ve Burge, M., J., Principles of Digital Image Processing, Undergraduate Topics in Computer Science, Springer, London, 2009.
27. Baumberg, A., M. ve Hogg, D., C., An Efficient Method for Contour Tracking Using Active Shape Models, Proceedings of the 1994 IEEE Workshop on, Kasım 1994, Austin, Bildiriler Kitabı, 194-199.
28. Hsieh, C., C., Liou, D., H., Cheng, Y., M. ve Cheng, F., C., Robust Visual Mouse By Motion History Image, System Science and Engineering (ICSSE), Temmuz 2010, Taipei, Bildiriler Kitabı, 161-166.
29. Zhao, L., Active Shape Model for Facial Feature Points Location, Machine Learning and Cybernetics (ICMLC), Temmuz 2012, Xian, Bildiriler Kitabı, 421-426.
30. Froba, B., Kastner, T., Zink, W. ve Kublbeck, C., Real-time Active Shape Models for Face Segmentation, Image Processing, International Conference on IEEE, Ekim 2001, Thessaloniki, Bildiriler Kitabı, 205-208.
31. Dinges, L., Al-Hamadi, A. ve Elzobi, M., An Approach for Arabic Handwriting Synthesis Based on Active Shape Models, Document Analysis and Recognition (ICDAR), International Conference on IEEE, Ağustos 2013, Bildiriler Kitabı, 1260-1264.
32. Duda, R., O., Hart, P., E. ve Stork, D., G., Pattern Classification, Second Edition, Wiley Publishing, New York, 2001
33. Sirovich, L. ve Kirby, M., Low-Dimensional Procedure for The Characterization of Human Faces, Journal of the Optical Society of America, 4, 3 (1987) 519-524.
34. Hyvarinen, A., Survey on Independent Component Analysis, Neural Computing Surveys, 2, 4 (1999) 94-128.
35. Abdi, H. ve Williams, L., J., Principal Component Analysis, Wiley Interdisciplinary Reviews: Computational Statistics, 2, 4 (2010) 433-459.
36. Dardas, N., H. ve Petriu, E., M., Hand Gesture Detection and Recognition using Principal Component Analysis, Computational Intelligence for Measurement Systems and Applications (CIMSA), International Conference on IEEE, Eylül 2011, Ottawa, Bildiriler Kitabı, 1-6.
37. Yang, Y. ve Newsam, S., Bag-of-Visual-Words and Spatial Extensions for Land-Use Classification, GIS '10 Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, Kasım 2010, California, Bildiriler Kitabı, 270-279.
38. Csurka, G., Dance, C., Fan, L., Willamowski, J. ve Bray, C., Visual Categorization with Bags of Keypoints, Workshop on statistical learning in computer vision, ECCV, 1, 1 (2004) 1-22.

39. Mukherjee, J., Mukhopadhyay, J. ve Mitra, P., A Survey on Image Retrieval Performance of Different Bag Of Visual Words Indexing Techniques, Students' Technology Symposium (TechSym), Şubat 2014, Kharagpur, Bildiriler Kitabı, 99-104.
40. Kato, H. ve Harada, T., Image Reconstruction from Bag of Visual Words, Computer Vision and Pattern Recognition (CVPR), Haziran 2014, Columbus, Bildiriler Kitabı, 955-962.
41. Dai, J., Song, W., Pei, L. ve Zhang, J., Remote Sensing Image Matching via Harris Detector and SIFT Descriptor, Image and Signal Processing (CISP), International Conference on IEEE, Ekim 2010, Yantai, Bildiriler Kitabı, 2221-2224.
42. Bay, H., Ess, A., Tuytelaars, T. ve Van Gool, L., Speed-Up Robust Feature(SURF), Computer Vision and Image Understanding, 110,3 (2008) 346-359.
43. Singh, P. ve Lodhi, R., S., Hand Tracking and Head Movement Detection: A State-of – Art Survey, Journal of Computer and Advanced Engineering Research (IJCAER), 1, 2 (2014) 1-6.
44. Jones, M., J. ve Rehg, J., M., Statistical Color Models with Application to Skin Detection, International Journal of Computer Vision, 46, 1 (2002) 81-96.
45. Yang, J., Lu, W. ve Waibel, A., Skin-color modeling and adaptation, Computer Vision ACCV'98, 1352, (1998) 687-694.
46. Stauffer, C. ve Grimson, W., E., L., Adaptive Background Mixture Models for Real-Time Tracking, Computer Vision and Pattern Recognition, Haziran 1999, Fort Collins, Bildiriler Kitabı, 246-252.
47. Sandeep, K. ve Rajagopalan, A., N., Human Face Detection in Cluttered Color Images Using Skin Color and Edge Information, Indian Conference on Computer Vision, Graphics and Image Processing, Aralık 2002, Ahmedabad, Bildiriler Kitabı, 122-128.
48. Sigal, L., Sclaroff, S. ve Athitsos, V., Estimation and Prediction of Evolving Color Distributions for Skin Segmentation Under Varying Illumination, Computer Vision and Pattern Recognition (CVPR), Haziran 2000, Hilton Head Island, Bildiriler Kitabı, 152-159.
49. Kampmann, M., Segmentation of a Head into Face, Ears, Neck and Hair For Knowledge-Based Analysis-Synthesis Coding of Videophone Sequences, International Conference Image Processing (ICIP), Ekim 1998, Chicago, Bildiriler Kitabı, 876-880.
50. Kim, W. ve Lee, J., J., Visual Tracking Using Snake for Object's Discrete Motion. International Conference on Robotics and Automation (ICRA), Mayıs 2001, Seoul, Bildiriler Kitabı, 2608-2613.

51. Comaniciu, D., Ramesh, V. ve Meer, P., Real-Time Tracking of Non-Rigid Objects Using Mean Shift, *Computer Vision and Pattern Recognition (CVPR)*, Haziran 2000, Hilton Head Island, Bildiriler Kitabı, 142-149.
52. Gurjal, P. ve Kunnur, K., Real Time Hand Gesture Recognition Using SIFT, *International Journal of Electronics and Electrical Engineering*, 2, 3 (2012) 19-33.
53. Yao, Y. ve Li, C., T., Hand Posture Recognition Using SURF with Adaptive Boosting, *British Machine Vision Conference (BMVC)*, Eylül 2012, London, Bildiriler Kitabı, 50-60.
54. Denzler, J. ve Paulus, D., W., Active Motion Detection and Object Tracking, *International Conference on Image Processing (ICIP)*, Kasım 1994, Austin, Bildiriler Kitabı, 635-639.
55. Isard, M., A., Visual Motion Analyses by Probabilistic Propagation of Conditional Density, Ph.D. thesis, Robotics Research Group Department of Engineering Science, University of Oxford, Oxford, 1998.
56. Jiang, L., Wang, D., Cai, Z. ve Yan, X., Survey of Improving Naive Bayes for Classification, *Advanced Data Mining and Applications Lecture Notes in Computer Science*, 4632, (2007) 134-145.
57. Vidhya, K., A. ve Aghila, G., A Survey Naive Bayes Machine Learning Approach in Text Document Classification, *International Journal of Computer Science and Information Security*, 7, 2 (2010) 206-211.
58. Khatri, M., A Survey of Naive Bayes Algorithms for Similarity in Recommendation Systems, *International Journal of Advanced Research in Computer Science and Software Engineering*, 2, 5 (2012) 217-219.
59. Kröse, B. ve Smagt, P., *Introduction to Neural Networks*, Eighth Edition, The University of Amsterdam, Amsterdam, 1996.
60. Kul, S., *Diagnosis of Lumbar Disc Hernia From Images Using Artificial Neural Network*, Yüksek Lisans Tezi, Fatih University, Institute of Science and Engineering, Computer Engineering, İstanbul, 2008.
61. Zurada, J., M., *Introduction to Artificial Neural Systems*, PWS Publishing Company, West St. Paul, 1992.
62. Burges, C., J., C., A Tutorial on Support Vector Machines for Pattern Recognition, *Data Mining and Knowledge Discovery*, 2, 2 (1998) 121-167.
63. Abe, S., *Support Vector Machines for Pattern Classification*, Second Edition, Springer, Kobe, 2010.

64. Byun, H. ve Lee, S., W., Applications of Support Vector Machines for Pattern Recognition: A Survey, Pattern Recognition with Support Vector Machines, 2388, (2002) 213-236.

ÖZGEÇMİŞ

Ramazan Özgür DOĞAN, 1987 Arsin/Trabzon doğumludur. İlkokulu ve ortaokulu Kaşüstü Cumhuriyet İlköğretim Okulu'nda ve liseyi Kaşüstü Çok Programlı Lisesi'nde tamamlamıştır. Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü'nden 2011 yılında mezun olmuştur. 2011-2012 eğitim-öğretim yılının bahar döneminde Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda yüksek lisans programına başlamıştır. Temmuz 2011 yılından itibaren Gümüşhane Üniversitesi Meslek Yüksek Okulu'nda öğretim görevlisi olarak çalışmaktadır. İyi derecede İngilizce bilmektedir.