

**İNSANSIZ KARA ARAÇLARI İÇİN DİNAMİK NESNELERİN TANINMASI  
AMACIYLA GÖRÜNTÜ İŞLEME TABANLI BİR SİSTEM GELİŞTİRİLMESİ**

DOKTORA TEZİ

Güray SONUGÜR

DANIŞMAN

Yrd. Doç. Dr. Barış GÖKÇE

MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI

AĞUSTOS, 2016

Bu tez çalışması 14.FENBİL.37 numaralı proje ile BAPK tarafından desteklenmiştir.

**AFYON KOCATEPE ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**

**DOKTORA TEZİ**

**İNSANSIZ KARA ARAÇLARI İÇİN DİNAMİK NESNELERİN**  
**TANINMASI AMACIYLA GÖRÜNTÜ İŞLEME TABANLI BİR**  
**SİSTEM GELİŞTİRİLMESİ**

**Güray SONUGÜR**

**DANIŞMAN**

**Yrd. Doç. Dr. Barış GÖKÇE**

**MAKİNE MÜHENDİSLİĞİ ANABİLİM DALI**

**AĞUSTOS, 2016**

## TEZ ONAY SAYFASI

**Güray SONUGÜR** tarafından hazırlanan “**İnsansız Kara Araçları İçin Dinamik Nesnelerin Tanınması Amacıyla Görüntü İşleme Tabanlı Bir Sistem Geliştirilmesi**” adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 25/08/2016 tarihinde aşağıdaki jüri tarafından oy birliği ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **Makine Mühendisliği Anabilim Dalı’nda DOKTORA TEZİ** olarak kabul edilmiştir.

**Danışman** : Yrd.Doç.Dr. Barış GÖKÇE

**Başkan** : Prof. Dr. İbrahim MUTLU  
Afyon Kocatepe Ü. Teknoloji Fakültesi

**Üye** : Doç. Dr. Arif GÖK  
Amasya Ü. Teknoloji Fakültesi

**Üye** : Yrd. Doç. Dr. Ayhan GÜN  
Dumlupınar Ü. Mühendislik Fakültesi

**Üye** : Yrd. Doç. Dr. Said Mahmut ÇINAR  
Afyon Kocatepe Ü. Mühendislik Fakültesi

**Üye** : Yrd. Doç. Dr. Barış GÖKÇE  
Afyon Kocatepe Ü. Teknoloji Fakültesi

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü Yönetim Kurulu’nun

...../...../..... tarih ve

..... sayılı kararıyla onaylanmıştır.

.....

Prof. Dr. Hüseyin ENGİNAR

Enstitü Müdürü

**BİLİMSEL ETİK BİLDİRİM SAYFASI**  
**Afyon Kocatepe Üniversitesi**

**Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;**

- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

**beyan ederim.**

**25/06/2016**

**Güray SONUGÜR**

**ÖZET**  
Doktora Tezi

**İNSANSIZ KARA ARAÇLARI İÇİN DİNAMİK NESNELERİN TANINMASI  
AMACIYLA GÖRÜNTÜ İŞLEME TABANLI BİR SİSTEM GELİŞTİRİLMESİ**

Güray SONUGÜR

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü

Makine Mühendisliği Anabilim Dalı

**Danışman:** Yrd. Doç. Dr. Barış GÖKÇE

Bu çalışmada bir insansız kara aracı veya benzeri hareketli platformların kameraları aracılığıyla aldığı görüntüden rotası boyunca önünde bulunan hareketli engelleri tespit edip tanıyabilmesi için görüntü işleme tabanlı ve GPS destekli yardımcı bir sistem geliştirilmiştir. Gerçek zamanlı uygulama öncesinde rota üzerinde belirlenen referans noktalarda varsayılan arka planı temsil eden görüntü modelleri oluşturulur. Bu modellerin elde edilebilmesinde beş farklı yaklaşım geliştirilmiştir. Bu modeller görüntü matrislerinden çok daha küçük boyutlarda olan ve elde edildiği referans noktadaki varsayılan arka planı temsil eden öznitelik matrisleridir. İnsansız bir mobil araç gerçek zamanlı harekete başladığında elde edilen anlık görüntülerin modelleri ile hafızadaki ilişkili referansa ait model karşılaştırılarak sahnedeki hareketli nesnelere tespit edilebilmektedir. Nesnelere tanınması ve sınıflandırılması için ise yapay sinir ağı teknikleri kullanılmıştır. Geliştirilen yaklaşımlar ile hareketli kameraların oluşturduğu görüntünün dengelenmesi için karmaşık ve yüksek hesaplama maliyetli işlemlere gerek duyulmamaktadır. Ayrıca düşük maliyetli bilgisayar sistemleri ve kameralar ile yüksek başarı oranlarında nesne tespit ve tanımanın yapılabildiği görülmüştür.

**2016, xiii + 151 sayfa**

**Anahtar Kelimeler:** Görüntü İşleme, Hareketli Nesne Tanıma, İnsansız Kara Aracı, Hareketli Kamera

**ABSTRACT**  
PhD Thesis

DEVELOPMENT OF A COMPUTER VISION BASED SYSTEM TO DETECT  
DYNAMIC OBJECTS FOR UNMANNED GROUND VEHICLES

Güray SONUGÜR

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Mechanical Engineering

**Supervisor:** Assist.Prof. Barış GÖKÇE

In this study, a computer vision based and Global Positioning System (GPS) assistance supported system was developed for unmanned ground vehicle (UGV) or similar moving platforms to detect dynamic objects along its route. Before the real time application of the UGV, image models were created that represents a default background in all specified coordinates on the route. Five different approach were developed to obtain the image models. These models are sorts of feature matrixes which are much smaller than complete image matrices. The model matrices were recorded in the system database and a database relation was created between the model and its coordinate. The model belongs to related coordinates and feature matrices of instant images captured when the UGV moved are matched. When unmanned ground vehicle begins real-time motion, moving objects are determined in the scene by comparing the model of reference associated in memory with snapshots models. If the feature matrices are not compatible with each other, there is a possibility of dynamic objects in the scene. The artificial neural network techniques are used for recognition and classification of objects. Developed approaches provide alternatives to compensate for the noise generated by motion instead of High computational cost such complex operation. Object detection and recognition are observed to be done with low-cost computer systems and cameras in high success rate.

**2016, xiii+ 151 pages**

**Keywords:** Image Processing, Dynamic Object Detection, Unmanned Ground Vehicles, Moving Camera

## TEŐEKKÜR

Doktora sürecim boyunca tüm bilgi birikimini benimle paylaşarak bana yol gösteren, maddi ve manevi desteklerini esirgemeyen, yönlendirme ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren, bir danışman hocadan daha çok bir dost, bir arkadaş gibi yardımcı olan danışman hocam Sayın Yrd. Doç. Dr. Barış GÖKÇE'ye sonsuz teşekkürlerimi sunarım.

Tez aşamasında değerli katkılarından dolayı değerli hocalarım Sayın Prof. Dr. Süleyman TAŐGETİREN'e ve Sayın Prof. Dr. Abdurrahman KARABULUT'a teşekkür ederim.

Doktora sürecim boyunca çalışma hayatımı sürdürdüğüm iş yerimde benden desteklerini esirgemeyen Afyon Kocatepe Üniversitesi Bilgi İşlem Daire Başkanı Sayın Öğr. Grv. Ahmet YURDADUR'a teşekkür ederim.

Deneysel çalışmalarım sırasında kar yağışı altında, kuru ayazda ve sıcak yaz güneşi altında benimle birlikte ter döken, iş yerinde boşluğumu doldurmaya çalışan değerli iş arkadaşlarım Sayın Öğr. Grv. Emrah KUZU'ya, Sayın Memur Rahmi DÖNERKAYA'ya ve Sayın Teknisyen Hüseyin AKÇEKEN'e teşekkür ederim.

Projemizde bize destek olan Yiğit Akü A.Ő. 'ne teşekkürlerimi sunarım.

Son olarak; uzun doktora süreci boyunca benden manevi desteğini esirgemeyen, ev hayatında bana katlanan, hiçbir zaman şikayet etmeyen saygıdeğer eşim Sebila SONUGÜR'e ve benden dualarımı esirgemeyen saygıdeğer Anne ve Babama sonsuz teşekkürlerimi sunarım.

Güray SONUGÜR

AFYONKARAHİSAR, 2016

# İÇİNDEKİLER DİZİNİ

Sayfa

ÖZET .....	i
ABSTRACT .....	ii
TEŞEKKÜR .....	iii
İÇİNDEKİLER DİZİNİ.....	iv
KISALTMALAR DİZİNİ .....	vii
ŞEKİLLER DİZİNİ .....	ix
ÇİZELGELER DİZİNİ.....	xii
1. GİRİŞ.....	1
2. LİTERATÜR BİLGİLERİ .....	8
2.1 Mobil Otonom Robotlar (İKA).....	8
2.2 Mobil Robot Sistemlerinin Genel Yapısı .....	9
2.3 Veri Toplama.....	10
2.4 Yapay Görme.....	12
2.5 Görüntü Çerçevelerinde Nesne Tespiti ve Tanınması.....	13
2.5.1 Öznitelik Çıkarma .....	14
2.5.2 Nesne Tespiti .....	18
2.5.2.1 Optik Akış Yöntemi .....	18
2.5.2.2 Arka Plan Kestirimi Yöntemi.....	19
2.5.2.3 Çerçeve Farkı Yöntemi.....	21
2.5.3 Nesne Tanıma ve Sınıflandırma .....	21
2.5.4 Nesne Tanıma İşleminde Yaşanabilecek Zorluklar.....	39
3. MATERYAL ve METOT .....	41
3.1 Matlab Yazılımı.....	41
3.2 Ardupilot Mega Kontrol Kartı.....	43
3.3 GPS Alıcı Sistem.....	44
3.4 Mission Planner Yazılımı .....	47
3.5 Arduino Uno Kontrol Kartı .....	48
3.6 DC Motor Sürücü Kartı .....	49
3.7 Mobil Araçlar .....	50
3.7.1 Kamera.....	51
3.7.2 Bilgisayar Sistemi.....	51
3.8 Yapay Sinir Ağları (YSA) .....	52
3.8.1 Yapay Sinir Hücresinin Modellenmesi.....	53
3.8.2 Yapay Sinir Ağlarının Mimarisi .....	55



3.8.3	İleri Beslemeli Geri Yayınımlı Yapay Sinir Ağları.....	56
3.8.4	Yapay Sinir Ağlarının Eğitilmesi .....	57
3.9	Dinamik Nesne Tespit İşlemleri .....	58
3.9.1	Yöntem 1: GPS Destekli Hareket Bölgesi Belirleme ve Geometrik Dönüşüm.....	61
3.9.1.1	Referans Öznitelik Matris Modelinin Oluşturulması .....	63
3.9.1.2	Gerçek Zamanlı Hareketli Nesne Tespiti .....	68
3.9.2	Yöntem2: GPS Destekli Çerçeve Hizalama ve Çerçeve Çıkartma .....	70
3.9.2.1	Referans Öznitelik Matris Modelinin Oluşturulması .....	71
3.9.2.2	Gerçek Zamanlı Hareketli Nesne Tespiti .....	72
3.9.3	Yöntem 3: GPS Destekli Gradyan Tabanlı Gri Seviye Değişimi Modellemesi .....	75
3.9.3.1	Varsayılan Arka Plan Modelinin Oluşturulması .....	77
3.9.3.2	Gerçek Zamanlı Hareketli Nesne Tespiti .....	83
3.9.4	Yöntem 4: GPS Destekli Özelleştirilmiş Arka Plan Modellemesi.....	85
3.9.4.1	Varsayılan Arka Plan Modelinin Oluşturulması .....	85
3.9.4.2	Gerçek Zamanlı Hareketli Nesne Tespiti .....	86
3.9.5	Yöntem 5: Özelleştirilmiş İmge Çakıştırma.....	88
3.9.5.1	Varsayılan arka plan modelinin oluşturulması .....	89
3.9.5.2	Gerçek Zamanlı Hareketli Nesne Tespiti .....	92
3.10	Nesne Tanıma İşlemleri.....	92
3.10.1	Tanıma İşlem Adımları.....	93
3.10.2	Yapay Sinir Ağları ile Sınıflandırma.....	94
3.11	Ara Yüz Tasarımları .....	101
3.11.1	Video Akış Analizi Ara Yüzleri .....	101
3.11.1.1	Model Oluşturma.....	101
3.11.1.2	Nesne Tespit ve Tanıma Modülü .....	102
3.11.2	Gerçek Zamanlı Çalışma Ara Yüzleri .....	103
3.11.2.1	Gerçek Zamanlı Model Oluşturma.....	104
3.11.2.2	Gerçek Zamanlı Nesne Tespit ve Tanıma .....	105
4.	BULGULAR .....	106
4.1	Yöntem 1: GPS Destekli Hareket Bölgesi Belirleme ve Geometrik Dönüşüm	106
4.1.1	Farklı Öznitelik Çıkartma Algoritmalarının Performansları .....	106
4.1.2	Hareketli Otomobil Kullanılarak Yapılan Çalışma .....	108
4.1.3	Hareketli Yaya Kullanılarak Yapılan Çalışma .....	110
4.1.4	CPU Yükleme ve İşlem Hızları .....	112
4.2	Yöntem 2: GPS Destekli Çerçeve Hizalama ve Çerçeve Çıkartma .....	113
4.2.1	Farklı Öznitelik Çıkartma Algoritmalarının Performansları .....	113

4.2.2	Hareket Halinde Otomobil Kullanılarak Yapılan Çalışma.....	113
4.2.3	Hareket Halinde Yaya Kullanılarak Yapılan Çalışma.....	115
4.2.4	CPU Yükleme ve İşlem Hızları .....	116
4.3	Yöntem 3: Gradyan Tabanlı Gri Seviye Değişimi Modellemesi .....	117
4.4	Yöntem 4: GPS destekli özelleştirilmiş arka plan modellemesi .....	123
4.5	Yöntem 5: Özelleştirilmiş İmge Çakıştırma Yöntemi.....	128
4.6	Nesne Tanıma İşlemleri.....	131
5.	TARTIŞMA ve SONUÇ .....	134
6.	KAYNAKLAR.....	139
	ÖZGEÇMİŞ.....	151

## KISALTMALAR DİZİNİ

### Kısaltmalar

---

APM	Ardupilot Mega Kontrol Kartı
ASCII	American Standart Code for Information Interchange
ATDNN	Adaptive Time Delay Neural Network
ATM	Automatic Teller Machine
BM	Background Modelling
BBME	Block Based Moving Estimation
BPNN	Back Propagation Neural Network- Geri Yayınımlı Sinir Ağları
CCD	Charge-Coupled Device
CCL	Connected Component Labeling
CSS-KNN	Curvature Scale Space Representation K-Nearest Neighbor
CWT	Continuous Wavelet Transform-Sürekli Dalgacık Dönüşümü
DC	Direct Current – Doğru Akım
DLL	Dinamic Link Library-Dinamik Link Kütüphaneleri
DoG	Difference of Gaussian
DSTEI	Dynamic Spatio-Temporal Entropy Image
DWT	Discrete Wavelet Transform – Ayrık Dalgacık Dönüşümü
ED	Euclidean Distance – Öklid Uzaklığı
EKF	Extended Kalman Filter – Genişletilmiş Kalman Filtresi
EXE	Executable File – Çalıştırılabilir Dosya
FD-SVM	Fourier Descriptors Support Vector Machine
FFNN	Feed Forward Neural Network- İleri Beslemeli Yapay Sinir Ağı
GMM	Gaussian Mixture Model
GPS	Küresel Konumlandırma Sistemi (Global Positioning System)
IFD	Inter Frame Difference
I2C	Inter-Integrated Circuit
IMU	Inertial Measurement Unit
KDE	Kernel Density Estimation- Kernel Yoğunluk Kestirimi
KNN	K-Nearest-Neighbor
LKT	Lucas Kanade Tracking
LM	Levenberg-Marquardt Optimizasyonu
LVQ	Learning vector quantization
MLP	Multi Layer Perceptron – Çok Katmanlı Algılayıcı
MRF	Markov Random Fields – Markov Rastgele Alanları
NMEA	National Marine Electronics Association -ABD Ulusal Denizcilik Elektronikleri Birliği
NNDR	Nearest Neighbor Distance Ratio
OCR	Optical Character Recognition
OF	Optik Akış (Optical Flow)
OFM	Optical Flow Map – Optik Akış Haritası
PCA	Principal Component Analysis -Temel Bileşen Analizi
PODBFNN	Priorityorder Direction Basis Function Neural Network For Pattern Recognition
PTZ	Pan Tilt Zoom – Yatay hareket, dikey hareket, yakınlaştırma
RANSAC	Random Sample Consensus
RBF	Radial Basis Function

## **Kısaltmalar (Devam)**

---

RGB	Red – Blue - Green
SAD	Sum of Absolute Differences
SC-KNN	Shape Context K-Nearest Neighbor
SDG	Spatial Distribution of Gaussian
SIFT	Scale-Invariant Feature Transform-Ölçekten Bağımsuz Öznitelik Dönüşümü
SLAM	Simultaneous Localization and Mapping
SLAMMOT	Simultaneous Localization and Mapping and Moving Object Tracking
SONFIN	Self Constructing Neural Fuzzy Inference Network
SRI	Stanford Research Institute
SSD	Katı Hal Diskleri
SURF	Speeded Up Robust Features, Hızlandırılmış Gürbüz Öznitlikler
SVM	Support Vector Machine
TOF	Time of Flight
TUTGA	Türkiye Ulusal Temel GPS Ağı
UART	Universal Asynchronous Receiver and Transmitter-Genel Asenkron Alıcı ve Verici
UAV	Unmanned Aerial Vehicle – İnsansız Hava Aracı
UHF	Ultra High Frequency-Ultra Yüksek Frekanslar
WMM-ANN	Wavelet Modulus Maxima Artificial Neural Network
WNN	Wavelet Neural Network
WT-ANN	Wavelet Transform Artificial Neural Network
YCbCr	Parlaklık ve Renk

---

## ŞEKİLLER DİZİNİ

### Sayfa

Şekil 2.1 Elsie isimli mobil robot (İnt.Kyn.1).....	8
Şekil 2.2 Shakey mobil robot, “Computer History Museum.”.....	9
Şekil 2.3 İki sensörlü mobil robot (Wu <i>et al.</i> 2006).....	12
Şekil 2.4. Üç önemli imge sınıfı.....	15
Şekil 2.5 Yerel öznitelik yapılarıyla nesne tanıma prosedürü (Grauman and Leibe 2011) .....	17
Şekil 2.6 Tipik bir nesne tanıma sistemi (Andreopoulos and Tsotsos 2013).....	23
Şekil 2.7 Nesne tanıma sistemi bileşenleri (Nagabhushana 2005).....	23
Şekil 2.8 Hareketli nesnelere tanıma algoritması (Moriwaki, 2011).....	32
Şekil 3.1 Ardupilot Mega kontrol kartı (İnt.Kyn.6).....	44
Şekil 3.2 APM kontrol kartı güç regüle modülü.....	44
Şekil 3.3 GPS Temel İşlevi (Ünal 2012).....	45
Şekil 3.4 Çalışmada kullanılan GPS alıcısı (İnt.Kyn.4).....	47
Şekil 3.5 Mission planner ile oluşturulan yer istasyonu uygulaması resimleri(İnt.Kyn.7) .....	48
Şekil 3.6 Arduino Uno model kontrol kartı (İnt.Kyn.8).....	49
Şekil 3.7 Dc motor sürücü kartı.....	49
Şekil 3.8 Çalışmada kullanılan (a) ilk mobil araç, (b) ikinci mobil araç.....	50
Şekil 3.9 Sürüş sistemi blok diyagramı.....	50
Şekil 3.10 Bir yapay sinir hücresinin yapısı (Gurney 1996).....	53
Şekil 3.11 Çalışmada kullanılan 3 katmanlı ileri beslemeli yapay sinir ağı mimarisi ...	55
Şekil 3.12 Çalışmada kullanılan ileri beslemeli geri yayımlı yapay sinir ağı.....	56
Şekil 3.13 “1” numaralı prensibin uygulama adımları.....	59
Şekil 3.14 “2” numaralı prensibin uygulama adımları.....	60
Şekil 3.15 Yatay eksen geçiş aralıkları.....	62
Şekil 3.16 Mobil araç ilerleyişinde kullanılan genel akış şeması.....	62
Şekil 3.17 Belirlenmiş koordinatlardan elde edilen referans imgeler.....	64
Şekil 3.18 Örnek referans imge bölümlenmesi.....	66
Şekil 3.19 Öznitelikler arasındaki en kısa mesafenin bulunması amacıyla geliştirilen algoritma.....	67

<b>Şekil 3.20</b> <i>Ki</i> 'nci koordinat noktası için hafızadaki model ile gerçek zamanlı bilgilerin karşılaştırılması .....	69
<b>Şekil 3.21</b> Yöntem-2 dinamik nesne tespitinde kullanılan akış şeması .....	70
<b>Şekil 3.22</b> Yöntem-2 bölümlenmiş imge.....	71
<b>Şekil 3.23</b> Kayma vektörünün görsel ifadesi.....	73
<b>Şekil 3.24</b> <i>Ki</i> 'nci koordinat noktası için hafızadaki model ile gerçek zamanlı bilgilerin karşılaştırılması .....	75
<b>Şekil 3.25</b> Yöntem-3'de kullanılan mobil araç yatay geçiş aralıkları .....	76
<b>Şekil 3.26</b> Yöntem-3'de kullanılan mobil araç ilerleyişi genel akış şeması .....	77
<b>Şekil 3.27</b> Referans imgelerin orijinal ve mozaik durumları ve ortalama imge örneği .	79
<b>Şekil 3.28</b> Bir kısım referans görüntülerin ve ortalama görüntünün parlaklık geçişlerinin grafiksel ifadesi .....	80
<b>Şekil 3.29</b> Tüm referans ve ortalama görüntülere ait grafik.....	81
<b>Şekil 3.30</b> Yalın arka plan modeli ile nesne içeren görüntünün grafiksel karşılaştırması .....	81
<b>Şekil 3.31</b> <i>Ki</i> koordinat noktası için hafızadaki bilgiler ile gerçek zamanlı bilgilerin karşılaştırılması.....	84
<b>Şekil 3.32</b> <i>Ki</i> koordinat noktası için hafızadaki referans model ile gerçek zamanlı modelin karşılaştırılması.....	88
<b>Şekil 3.33</b> Yöntem-5 mobil araç ilerleyişinde kullanılan algoritmanın genel akış şeması .....	89
<b>Şekil 3.34</b> Yatay geçiş aralıkları.....	90
<b>Şekil 3.35</b> (a) Referans imge (b) bloklanmış mozaik görüntü (c) mozaik görüntüye ait matris .....	90
<b>Şekil 3.36</b> Nesne tanıma işlem adımları.....	93
<b>Şekil 3.37</b> Tespit edilen nesnelere dönüşüm işlemleri.....	97
<b>Şekil 3.38</b> Tasarlanan YSA mimarisi .....	99
<b>Şekil 3.39</b> Girdi olarak kullanılan bir kısım imgeler.....	100
<b>Şekil 3.40</b> Video akış analizi model oluşturma modülü ekran görüntüsü.....	102
<b>Şekil 3.41</b> Video akış analizi nesne tespit ve tanıma modülü ekran görüntüsü .....	103
<b>Şekil 3.42</b> Yöntem-2 için gerçek zamanlı modelleme modülünün ekran görüntüsü ...	104
<b>Şekil 3.43</b> Yöntem-2'ye ait nesne tespit ve tanıma modülü ekran görüntüsü.....	105
<b>Şekil 4.1</b> Çeşitli referans noktalarından üç farklı yatay geçiş aralığı kullanılarak elde edilen eşleştirme sonuçları .....	107

<b>Şekil 4.2</b> K1 koordinatından 22m uzaklıkta (1) varsayılan, (2) nesne içeren arka plan çerçevesinin bölümlenmiş görüntüleri .....	109
<b>Şekil 4.3</b> Geometrik dönüşüm ve ardışık çerçeve farkı ve eşikleme işlemleri sonrası elde edilen hareketli nesne silueti .....	110
<b>Şekil 4.4</b> K3 koordinatından 12 m. uzaklıkta (1) varsayılan, (2) nesne içeren arka plan çerçevesinin bölümlenmiş görüntüleri .....	110
<b>Şekil 4.5</b> Tespit edilen nesne silueti .....	111
<b>Şekil 4.6</b> K1 koordinatından 24 m uzaklıkta nesne içeren anlık çerçeve ile model arasındaki kayma (1), elde edilen nesne silueti (2), 10m uzaklıkta nesne içeren anlık çerçeve ile model arasındaki kayma (3), elde edilen nesne silueti (4) .....	114
<b>Şekil 4.7</b> K5 koordinatında 0,5m geçiş aralığı ile elde edilen model ile nesne içeren anlık çerçeve arasındaki kayma (1), elde edilen nesne silueti (2), 2m geçiş aralığı ile elde edilen model ile nesne içeren anlık çerçeve arasındaki kayma (3), elde edilen nesne silueti (4) .....	115
<b>Şekil 4.8</b> Yakın mesafeden hareketli nesne siluetinin elde edilmesi sürecinde oluşturulan imge örnekleri .....	118
<b>Şekil 4.9</b> Uzak mesafeden hareketli nesne siluetinin elde edilmesi sürecinde oluşturulan imge örnekleri .....	119
<b>Şekil 4.10</b> Hareketli nesne olarak otomobil kullanıldığında elde edilen imge örnekleri .....	120
<b>Şekil 4.11</b> Hareketli nesne olarak 20 m uzaklıkta bir otomobil kullanıldığında elde edilen imge örnekleri .....	121
<b>Şekil 4.12</b> Yakın bir mesafedeki hareketli nesnenin siluetinin elde edilmesi sürecinde oluşturulan imge örnekleri .....	124
<b>Şekil 4.13</b> Uzak bir mesafeden hareketli nesne siluetinin elde edilmesi sürecinde oluşturulan imge örnekleri .....	125
<b>Şekil 4.14</b> Yakın mesafede hareketli nesne olarak otomobil kullanıldığında elde edilen imge örnekleri .....	126
<b>Şekil 4.15</b> Uzak mesafede bir otomobil kullanıldığında elde edilen imge örnekleri ...	126
<b>Şekil 4.16</b> K1 koordinatından yakın mesafe yaya ve otomobil için elde edilen sonuçlar .....	129
<b>Şekil 4.17</b> K3 koordinatından uzak mesafe yaya ve otomobil için elde edilen sonuçlar .....	130
<b>Şekil 4.18</b> Eğitim, doğrulama ve test performans grafiği .....	133

## ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 3.1 \$GP GGA konum mesajı içeriği (İşeri 2006, İnt.Kyn.3) .....	46
Çizelge 3.2 \$GP GLL enlem/boylam mesajı içeriği (İşeri 2006, İnt.Kyn.3) .....	47
Çizelge 3.3 Mobil araç ve dc motor teknik özellikleri .....	51
Çizelge 3.4 Kenarları çıkartılmış silüet ile içi dolu silüet arasındaki giriş değişkeni farklılıkları.....	99
Çizelge 4.1 Farklı öznitelik çıkartma algoritmaları ve farklı karşılaştırma yöntemleri kullanılarak elde edilen ortalama eşlenik öznitelik sayıları .....	107
Çizelge 4.2 Şekil 4.1’de ifade edilen eşleştirmelere ait uzaklık ve açı değerleri .....	108
Çizelge 4.3 K1 koordinatından 22m uzaklıkta nesne içeren ve boş arka plan çerçevesinin bölüm başına öznitelik sayıları.....	109
Çizelge 4.4 K3 koordinatından 12 m uzaklıkta nesne içeren ve boş arka plan çerçevesinin bölüm başına öznitelik sayıları.....	111
Çizelge 4.5 Otomobil ve yayaları %85 doğrulukta tespit için gerekli eşik uzaklıklar.	111
Çizelge 4.6 Yöntemlerin performans karşılaştırması .....	112
Çizelge 4.7 Nesne uzaklığı değiştirilerek yapılan deneysel çalışmada elde edilen kayma vektörü değerleri. ....	114
Çizelge 4.8 Geçiş aralığı değiştirilerek yapılan deneysel çalışmada elde edilen kayma vektörü değerleri. ....	116
Çizelge 4.9 Otomobil ve yayaları %85 doğrulukta tespit için gerekli eşik uzaklıklar.	116
Çizelge 4.10 Yöntemlerin performans karşılaştırması .....	117
Çizelge 4.11 Yayalar ile 2 m geçiş aralığında yapılan deneme çalışması başarı oranları .....	122
Çizelge 4.12 Yayalar ile 1 m geçiş aralığında yapılan deneme çalışması başarı oranları .....	122
Çizelge 4.13 Otomobiller ile 2 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları .....	123
Çizelge 4.14 Otomobiller ile 1 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları .....	123
Çizelge 4.15 Yöntemlerin bilgisayar sistemi performans karşılaştırması.....	123
Çizelge 4.16 Yayalar ile 1 m geçiş aralığında yapılan deneme çalışması başarı oranları .....	127



<b>Çizelge 4.17</b> Yayalar ile 2 m geçiş aralığında yapılan deneme çalışması başarı oranları .....	127
<b>Çizelge 4.18</b> Otomobiller ile 1 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları.....	128
<b>Çizelge 4.19</b> Otomobiller ile 2 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları.....	128
<b>Çizelge 4.20</b> Yöntemlerin bilgisayar sistemi performans karşılaştırması.....	128
<b>Çizelge 4.21</b> 1 m geçiş aralığından elde edilen başarılı tespit oranları.....	130
<b>Çizelge 4.22</b> 2 m geçiş aralığından elde edilen başarılı tespit oranları.....	131
<b>Çizelge 4.23</b> Farklı metotların bilgisayar üzerindeki performansları.....	131
<b>Çizelge 4.24</b> Bir kısım deneme imgelerinin siluet haline dönüştürülmüş durumları...	132
<b>Çizelge 4.25</b> YSA'nın kategorilere göre sınıflandırma başarı oranları.....	133
<b>Çizelge 5.1</b> 3264x4928 çözünürlükte Yöntem-1 ve Yöntem-2 performans karşılaştırması .....	134
<b>Çizelge 5.2</b> 3264x4928 çözünürlükte Yöntem-3, Yöntem-4 ve Yöntem-5 performans karşılaştırmaları.....	135
<b>Çizelge 5.3</b> 480x640 çözünürlükte Yöntem-1 ve Yöntem-2 performans karşılaştırması .....	135
<b>Çizelge 5.4</b> 480x640 çözünürlükte Yöntem-3, Yöntem-4 ve Yöntem-5 performansları .....	136

## 1. GİRİŞ

Günümüz teknolojisinde insansız araçlar üzerinde yoğunlaşan çalışmalar kameralardan alınan gerçek zamanlı görüntülerden hareketli ya da hareketsiz nesnelere tespit edilmesi konusunda yapılan araştırmaların artmasına neden olmuştur (Shimizu *et al.* 2006, Gamez and Devy 2013). Özellikle insansız kara araçları, hava gözetleme sistemleri (Yu and Medioni 2007), araç sürüş yardımcı sistemleri (Kong *et al.* 2010), insansız deniz altı araçları (Foresti and Gentili 2000) konularında çalışmalar yoğunlaşmıştır.

Geleneksel video akış sistemleri özellikle güvenlik sektöründe kullanıldıklarında yoğun insan çabası gerektirmektedir. Dolayısıyla uzun süreli izlemelerde anlık tespit için uygun değildir. Yani geleneksel video izleme sistemleriyle sürekli gözlem yapılamamaktadır. Daha çok güvenlik zafiyetleri gerçekleştiikten sonra tespit yapılabilmektedir. Bu sistemler ancak otomatik nesne tespit ve tanıma sistemleri ile bütünleşik olarak sunuldukları takdirde sürekli gözlem yapılmasından bahsedilebilir. Bu tür bütünleşik donanımlar ile güvenlik sistemlerinin işletme güvenliklerine sağlayacağı faydalar artacaktır.

Büyük bir açık alana sahip bölgelerde çevrenin sağlıklı olarak izlenebilmesi ve gerekli tedbirlerin anlık olarak alınabilmesi için çok sayıda güvenlik personeline ve çok sayıda yüksek çözünürlüklü kameraya ihtiyaç duyulmaktadır. Bu durum işletmeler için yüksek maliyet oluşturacaktır. Geliştirilen sistem ile güvenlik sistemlerinde kamera bileşeni sabit olmaktan çıkarılarak belirli bir rota üzerinde gezici hale getirilmiştir. Kameralardan uzak olduğu için güvenlik sistemleri tarafından net olarak izlenemeyen konumlara kameraların insansız kara araçları (İKA) aracılığıyla yaklaştırılabilmesi sağlanmıştır. Bu şekilde güvenlik tedbirleri sağlanabilir olduğu gibi uzak konumlarda yeni kameralara olan gereksinim ortadan kaldırılmıştır.

Geleneksel sistemlerde kameralar genel olarak hareketsiz yani yerlerine sabit şekilde montajlanmıştır. PTZ (Pan – tilt - zoom) kameralarda yatay ve dikey hareket söz konusuysa da hareket joystickler ile verildiğinden yine insan eliyle yapılması şarttır. Üstte bahsedilen bütünleşik donanımların sabit kameralarla değil belirli bir rota üzerinde hareket edebilen platformlarda bulunan kameralar ile kullanıldığı düşünüldüğünde sağlayacağı faydalar daha da artacaktır. Hareketli platform olarak İKA kullanıldığı

takdirde geliştirilen sistem ile otonom bir devriye aracı için anlık nesne tespiti yeteneğine sahip bir bilgisayar görme sistemi elde edilmiştir.

Bir İKA için üzerindeki kameralar vasıtasıyla alınan çevresel görüntülerin yorumlanması büyük önem taşır. Bu tip araçlar üzerlerinde bulunan kameralar aracılığıyla elde ettikleri verileri kullanarak kendi uzamsal hareketlerine, haberleşme uygulamalarına, etkileşim sistemlerine yön vermektedirler. Bu nedenle nesne tespit ve tanıma uygulamaları bilgisayar görme sistemleri için vazgeçilmez unsurlardır.

İKA'nın dinamik çevre koşullarında engellerden kaçınarak ilerleyebilmesi, yol planlama yapabilmesi, yörünge takibi vs. yapabilmesi; sürekli değişen koşullara sahip dış dünyadan toplayacağı verilere bağlıdır. Bu noktada, dış dünyadan en net veri toplama aracı olarak son dönemlerde kameralar kullanılmaktadır. Günümüzde kamera gibi görüntü araçlarının sağladığı görüntüyü ve görüntü işleme algoritmalarını kullanarak karar verebilen nesnelere tanıma, taşıma ve yer değiştirme yapabilen robotik sistemler geliştirilmektedir. Görüntü işleme ve nesne tanıma çalışmaları arttıkça uygulama alanları da hızla gelişmektedir (Wahi *et al.* 2008, Wahi *et al.* 2010, Declercq and De Keyser 1999, Johan and Prabuwno 2011).

Günümüzde bilgisayar sistemlerinin boyutları küçülmüş ancak veri işleme hızları, depolama kapasiteleri ve SSD (Solid State Disk) diskler sayesinde disklerde depolanmış verilere hızlı erişim teknolojileri oldukça gelişmiş durumdadır. Yani hareketli nesne tespit ve tanıma çalışmalarında eskiden bir kısıt olarak kabul edilen veri tabanı erişiminin yoğun bir şekilde kullanımı, hafıza işlemleri, büyük veri depolama gibi işlemler artık kısıt olmaktan çıkmıştır. Bir kısım çalışmalarda (Andreopoulos and Tsotsos 2013, Guo *et al.* 2014, Koller *et al.* 1994, Bugeau and Pérez 2009) hesaplama maliyeti kısıtlarını göz önünde bulundurulduğu ifade edilmiştir. Yüksek hesaplama maliyetlerinden kaçınmak adına daha az maliyet gerektiren yaklaşımların geliştirildiği ve bu tür sistemlerin başarılarının da kısıtlı kaldığı anlaşılmaktadır. Bu çalışmada nesne tespit işlemlerinde yüksek hızda ve kapasitede erişim ve depolama özelliklerine sahip sistemler kullanılmıştır. Gerçekleştirilen sistemde arka plan görüntüsünün yabancı nesne içerip içermediğinin anlaşılabilmesi için önceden veri tabanına kaydedilmiş varsayılan arka plan modelleri kullanılmıştır.

Bu çalışmada sıkça kullanılacak olan terimlerden “varsayılan arka plan” terimi insansız kara aracının (İKA) rotası doğrultusunda önceden seçilmiş referans noktalarından alınmış ve içerisinde yabancı nesne içermeyen arka plan görüntülerini ifade etmektedir. “Arka plan modeli” ya da “referans model” terimleri ise bu görüntüleri en doğru şekilde temsil edebilecek nicelik ve niteliklerden oluşturulan ve referans olarak kullanılan vektör ya da matrislerdir. Bu vektör ya da matrisler görüntülerin alındığı referans noktasına ait konum bilgisi ile ilişkilendirilerek veri tabanına kaydedilmektedir. “Ön Plan” terimi imgenin arka planına ait olmayan sahneye sonradan dahil olan nesnelerin oluşturduğu bölümü ifade etmektedir. “Anlık görüntü” terimi İKA’nın gerçek zamanlı hareketi sırasında referans noktalardan geçerken aldığı görüntünün tamamını ifade etmektedir. “Anlık model” terimi ise bu görüntüleri temsil eden vektör ya da matrisleri ifade etmektedir. Burada önemli olan referans model ile anlık modelin aynı yöntemler ile oluşturulması ve yaklaşık olarak aynı noktalardan elde edilmiş olmalarıdır. Her bir görüntü çerçevesi (imge) kendi çözünürlüğü boyutlarında ve 0-255 arası tamsayılardan oluşan bir matris olarak kabul edilebilir. Bu şekilde varsayılan arka plan görüntüsünün tamamı model olarak kabul edilebilir, ancak bu durumda model matrisin boyutları büyük olacaktır. Bu şekilde karşılaştırma, eşleştirme vb. matematiksel işlemlerde hafıza kullanımı, işlem süreleri, CPU yükü ile birlikte hesaplama maliyetleri artacaktır. Bu sebeplerden dolayı sürdürülebilir bir sistem için modellerin daha küçük boyutlara indirgenmesi gereklidir. Geliştirilen yaklaşımlarda farklı tekniklerle modellemeler yapılarak arka plan modellerinin boyutları küçültülmüştür. Yöntem-1 ve Yöntem-2’de farklı teknikler ile elde edilen en güçlü özneliklerden oluşan matrisler model olarak seçilirken, Yöntem-3, Yöntem-4 ve Yöntem-5’de gri seviye farkları modellenmiştir.

Bu tez çalışmasında geliştirilen yaklaşımlar ile kamera hareketi nedeniyle hareket ediyormuş gibi görünen arka plan nesnelere değil tamamen bağımsız hareket eden nesnelerin tespit edilmesi amaçlanmıştır. Bunun için hareketli bir kamera ile belirli referans noktalardan alınan imgeler ile aynı noktalardan alınan varsayılan arka plan imgeleri çeşitli yöntemler ile karşılaştırılmıştır. Ancak yukarıda belirtildiği gibi karşılaştırma işlemlerinde görüntü matrislerinin tamamı değil model matrisleri kullanılmıştır. Karşılaştırma işlemlerinin sağlıklı sonuçlar verebilmesi için referans ve anlık görüntünün yaklaşık olarak aynı noktalardan alınması tespit başarısını arttırmaktadır. Aksi takdirde arka planda sabit olarak bulunan nesnelerin kamera lensine olan uzaklıkları değişeceğinden görüntüdeki büyüklükleri de değişecektir. Bu durumda

iki görüntüde de bulunan aynı nesnelere farklı nesnelermiş gibi algılanabilecek ve çalışmanın başarılı tespit oranı düşecektir. Bu nedenle; rota üzerinde yalnız arka plan ve anlık görüntülerin elde edileceği değişmez referans noktalarının hangi tekniklerle belirleneceğine karar verilmesi gerekmektedir. İKA üzerine yerleştirilebilecek çeşitli algılayıcılar ile rota üzerinde çeşitli değişmez noktaların tespit etmek mümkündür. Örnek olarak;

- Kızılötesi verici-alıcı sistem: Rota üzerinde belirli aralıklarla yerleştirilebilecek kızılötesi vericiler İKA üzerinde bulunan kızılötesi alıcılardan algılandıkları anda anlık görüntülerin alınması tekniğidir. Ancak rota boyunca döşenecek kızılötesi vericilerin enerji ihtiyaçlarının karşılanmasındaki zorluklar ve hassas cihazlar olan vericilerin dış ortamda sıklıkla arızalanabilmeleri nedeniyle pratikte kullanılmaları zordur.
- Lazer yansıtıcı-alıcı sistem: Rota üzerinde belirli aralıklar ile yerleştirilen lazer yansıtıcılar İKA üzerindeki lazer kaynağından gönderilen ışınları yansıttıkları anda anlık görüntülerin alınması tekniğidir. Yansıtıcıların pasif malzemeler olması nedeniyle enerjiye ihtiyaç duymamaktadır. Bu nedenle uygulanmaları kolaydır.
- Enkoder destekli çerçeve sayma yöntemi: Sabit hız ve belirli bir başlangıç noktasından başlamak şartıyla belirli zaman aralıklarında referans görüntü alınıp kaydedilmesi prensibine dayanmaktadır. Kat edilen mesafe enkoderlerden alınan sinyaller ile kontrol edilmektedir.
- GPS (Global Positioning System) koordinatları: Rota üzerinde önceden belirlenen coğrafi konumların GPS koordinatları tespit edilir ve İKA bu koordinatlardan geçerken anlık görüntüleri alarak referans modeller ile karşılaştırma yapar.

Bu çalışmada rota üzerindeki referans noktalarının belirlenmesinde GPS koordinatları tekniği ile enkoder destekli çerçeve sayma tekniği kullanılmıştır. Referans alınan tüm noktalarda anlık görüntüler alınmadan önce yalnız arka plan görüntüleri alınarak referans modelleri oluşturulmuştur. İKA üzerinde iki adet akıllı telefon destekli GPS alıcısı kullanılarak alınan koordinatların hassasiyetleri artırılmıştır. Geliştirilen yaklaşımlarda İKA'nın rotası üzerindeki referans ve anlık modellerin elde edildiği her bir referans nokta  $K_i$  olarak ifade edilmiştir. Referans alınan noktalar arası mesafe ne kadar azaltılırsa

çalışmanın başarısı aynı ölçüde artacaktır. Ancak iki referans nokta arası mesafe belirlenirken bilgisayar görme sisteminin verileri işleme süresi dikkate alınmalıdır. Ayrıca kullanılan GPS alıcı sistemlerin ve enkoder cihazlarının hassasiyetleri de önemlidir. Hassasiyeti 2 m olan bir alıcı sistem ile çalışırken bu mesafeden daha az bir aralık seçilmemelidir.

Görüntü çerçevelerindeki nesnelerin tanınması işlemlerinde ilk aşama nesnenin doğru bir şekilde tespit edilmesidir. Tespit edilecek nesne, ardışık çerçeveler içinde hareketsiz kalarak engel niteliği alabildiği gibi yine ardışık çerçeveler içerisinde hareketli de olabilir. Hareketsiz nesneler için pek çok yaklaşım geliştirilmiştir ancak hareketli nesnelerin tespiti için ardışık imge çerçeveleri içerisinde gizlenmiş bulunan bilgi parçacıklarını açığa çıkartmak gereklidir. Hareketli nesne tespiti amacıyla en sık yapılan işlem video akış analizidir. Farklı zamanlarda alınan iki veya daha fazla görüntü çerçevesi, görüntüleme sistemi ile sahne arasındaki bağıl hareket hakkında bilgiler içerir. Bu bilgi; çerçeveler arasındaki gri seviye ve renk değişimleri şeklinde veya nokta, sınır çizgileri ve bölütlenmiş alanların uzamsal ve öznitelik değişimleri şeklindedir (Chen 2008).

Bir İKA'a kamera yerleştirildiğinde alınan görüntülerden hareketli nesnelere tespit etmek için sabit kameralarda kullanılan yöntemler kullanılamaz. Çünkü buradan alınan görüntülerdeki arka planın tamamı aracın hareketi nedeniyle hareketli olarak algılanır. Kullanılacak olan algoritmaların kamera hareketi nedeniyle hareketliymiş gibi görünen arka plan ile gerçekten hareketli nesnelere ayırt edebilmesi gerekir.

Yapılan çalışmada hareketli nesne tespiti için 5 farklı yaklaşım geliştirilmiştir. Bu yaklaşımlar;

1. GPS Destekli Hareket Bölgesi Belirleme ve Geometrik Dönüşüm,
2. GPS Destekli Çerçeve Hizalama ve Çerçeve Çıkartma,
3. GPS Destekli Gradyan Tabanlı Gri Seviye Değişimi Modellemesi ile Hareketli Nesne Tespiti,
4. GPS Destekli Özelleştirilmiş Arka Plan Modellemesi ile Dinamik Nesnelerin Tespiti,
5. Özelleştirilmiş İmge Çakıştırma Yöntemi ile Hareketli Nesnelerin Tespiti Yöntemleridir.

Bu yaklaşımların hepsinde de genel bir prensip kullanılmıştır. Bu prensibe göre gerçek zamanlı hareket öncesinde İKA'nın rotası boyunca referans alınan noktalardan varsayılan arka planı temsil eden referans görüntüler alınarak modellenir ve referans model olarak hafızada depolanır. Gerçek zamanlı hareket başladığında İKA sürekli olarak GPS koordinatlarını ya da imge çerçeve sayılarını kontrol eder ve referans alınan noktalara geldiğinde anlık görüntüyü alır, modeller ve hafızadaki referans model ile karşılaştırarak aradaki farkları bulur. Eğer eşik değerden daha az fark var ise yoluna devam eder aksi takdirde rotasında yabancı bir nesne bulunduğuna karar verir ve nesne tanıma işlemlerini başlatır. Yukarıda listelenen yaklaşımlar arasındaki fark görüntülerin modelleme tekniklerindeki farklılıklardır. Geliştirilen yöntemler MATERYAL METOT bölümünde ayrı başlıklar altında incelenmiş, elde edilen deneysel sonuçlar ise BULGULAR bölümünde ayrı başlıklar altında verilmiştir.

Sabit kameralar ile yapılan çalışmalarda farklı yöntemler uygulanabilmektedir. Bunlardan başlıcaları; arka plan kestirimi yöntemleri (Jarraya *et al.* 2010, Spagnolo *et al.* 2006), optik akış tabanlı yöntemler (Yu *et al.* 2012, Kim *et al.* 2010, Zhang *et al.* 2006) ve ardışık çerçevelerin çıkarılmasına dayalı metotlardır (Weng *et al.* 2010). Kameralar sabit olduğu için genellikle kararlı sonuçlar alınmaktadır ancak sahnede oluşabilecek ışık değişimleri, gölge etkisi, kamera titremesi, ağaç sallanma hareketi ve deniz dalgası vs. şeklindeki gürültülerin sönümlenmesi gerekir.

Hareketli kameralar ile nesne tespiti uygulamalarında da optik akış yöntemleri (Roth and Black 2007, Kim *et al.* 2010) kararlı sonuçlar vermektedir. Bu yöntem öznitelik tabanlı veya alan tabanlı olarak uygulanabilmektedir (Liu *et al.* 1996). Ancak kameranın hareketi de dâhil olmak üzere ardışık imge çerçeveleri arasındaki nesne hareketini tespit etme zaman aralıkları çok düşük olduğundan bilgisayar sistemleri için işlem yükü çok ağırdır. Bu nedenle özel donanımlar haricinde kullanılması uygun değildir (Ren *et al.* 2003). Arka plan kestirimi yöntemlerinde ise sabit kameralar için kullanılan algoritmalar hareketli kameralar için kullanılamamaktadır. Hareketli kameralarda ardışık imge çerçeveleri arasında açısallık ve uzamsal farklar bulunur. Bu nedenle hareketli nesnelere tespit edilebilmesi için öncelikle iki çerçeve arasında kamera hareketi nedeniyle oluşan farkın dengelenmesi gereklidir. Bu yöntemde imge çakıştırma (image registration) yöntemi

denilmektedir (Sappa *et al.* 2008). Bu çalışmada kullanılan yöntemlerde de sıklıkla kullanılmıştır. Dört adımda gerçekleştirilir (Cheraghi and Sheikh 2012):

- Referans ve ardışık gelen imge çerçevelerinin özniteliklerinin çıkartılması,
- Özniteliklerin eşleştirilmesi,
- Dönüşüm modelinin belirlenmesi, (geometrik, lineer olmayan, izdüşel, vs.),
- Dönüşüm gerçekleştirilerek imgenin yeniden elde edilmesi.

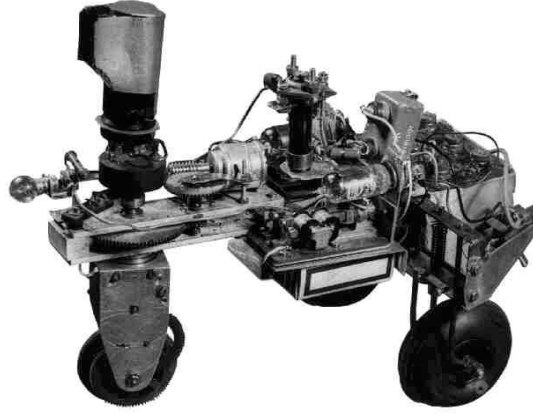
Her ardışık imge çerçevesi için yukarıda belirtilen adımların gerçekleştirilerek kamera hareketinin dengelenmesi bilgisayar sistemleri için ağır bir işlem yükü oluşturur. Bu nedenle gerçek zamanlı görüntü akışlarında uygulanması zordur. Bu çalışmada bilgisayar görme sistemleri için ağır işlem yükü ve yüksek bilgi depolama maliyetleri gerektirmeyen yürütülebilir yaklaşımlar geliştirilmeye çalışılmıştır.



## 2. LİTERATÜR BİLGİLERİ

### 2.1 Mobil Otonom Robotlar

Gerçek anlamda mobil otonom robotların başlangıçları 1950’li yılların başlarına dayanmaktadır. Bristol Üniversitesinden Greywalter (Walter 1950) tarafından üretilen “Elsie” isimli mobil robot ışık detektörüne, dokunma detektörüne, itme ve yönlendirme motorlarına ve iki vakum tüplü analog bir bilgisayara sahipti(İnt.Kyn.1). Kaplumbağa görünümlü olan bu robotlar, tepelerine monteli lambalar ve ışığa yönelimi sağlayan algılayıcıları ile görünüşte etkileşim içerisindeymiş gibi de olsa aralarında herhangi bir iletişim ya da karşılıklı tanıma işlemi bulunmamaktaydı (Fong *et al.* 2003). Elsie isimli robotun görüntüsü Şekil 2.1’de verilmiştir.



Şekil 2.1 Elsie isimli mobil robot (İnt.Kyn.1).

Sonrasında; Stanford Research Institute (SRI) tarafından bellek ve mantıksal akıl yürütme kullanarak sorunları çözen ve kendi ortamında hareket edebilen bir mobil robot olarak 1960'larda yani yapay zeka uygulamalarının revaçta olduğu bir dönemde “Shakey” (Nilsson 1969) geliştirilmiştir. Shakey bazı bakımlardan başarılı olduğu kadar, otonomi hususunda başarısız olmuştur. Çünkü Shakey harici bir bilgisayar tarafından kontrol ediliyordu ve herhangi bir kısıt bulunmayan çevresini algılamakta sorunlar yaşıyordu. Ayrıca oldukça da yavaştı (Ferrandez *et al.* 2010).

Shakey projesi sürekli geliştirilmeye devam edilmiş, 1984 yılında ana dilde verilen basit komutları anlayabilecek duruma gelmiştir (Nilsson 1984). Shakey mobil robotun bir örneği Şekil 2.2’de gösterilmiştir.



**Şekil 2.2** Shakey mobil robot, “Computer History Museum”.

Shakey mobil robotu ve 1972 yılı sonrası; pek çok robot kontrol mimarisi literatüre geçmiştir. Bunlar birbirlerinden çok farklı olmalarına rağmen temelde algılama, karar verme ve harekete geçme klasik prensibini benimsemişlerdir (Fraichard and Garnier 1999). Bağımsız olarak düşünebilen, çevresi ile etkileşim kurabilen, akıllı görevleri çözme yeteneğine sahip mobil robotları üretme girişimleri sonrası, otonom robot çalışmaları teknolojik bir darboğazla karşılaşmış ve 1970 ‘ler boyunca pek çok araştırma laboratuvarı bir durgunluk içerisine girmiştir. 1980 ‘lerin ortalarına gelindiğinde iki önemli olay mobil robot çalışmalarını dramatik olarak değiştirmiştir. Bunlardan birincisi; kişisel bilgisayarlar ve donanım küçültme teknolojilerindeki gelişmelerin performans artışı ile aynı eski sorunlar ile başa çıkmayı sağlaması, diğer olay ise “problem çözme” kavramının değişmesidir. Brooks ‘un “Modüler Hesaplama” (Brooks 1991) kavramını sunması sonrasında robotik teknolojisi devasa sorunlarla, sonsuz sayıda hesaplama çevrimleriyle ve devasa büyüklükte veri tabanlarıyla uğraşmak zorunda kalmamıştır. Hızlı operasyonlar yapabilen küçük modüler programlar ve bunlarla iletişim kurabilme teknolojisi ile daha operasyonel bir duruma gelmiştir (Ferrandez *et al.* 2010).

## **2.2 Mobil Robot Sistemlerinin Genel Yapısı**

Bir mobil robot sisteminin tasarımı, disiplinler arası doğası ve gerçek-zamanlı operasyon ile ilgili gereksinimleri nedeniyle zor ve karmaşık bir iştir (Travelyan 1997, Yavuz 1999). Mobil robotların tüm dünyada tasarlandığı, yapıldığı, denemelerinin yapıldığı yaygın

olarak bilinen bir gerçektir. Yine de; bu popülerliğe rağmen, sistemin alt yapısında bulunan yapısal ve işlevsel karmaşıklık tartışmaları çoğu zaman belirsizlikler ya da daha küçük alanlarda yoğunlaşmaktadır (Deng *et al.* 2000, Medland and Mullineux 2000). Literatürün gözden geçirilmesi neticesinde araştırmacıların odaklandığı başlıca konu başlıkları aşağıda listelenmiştir.

- Karar mekanizmaları (Guldner and Bauer 1995, Gopalakrishnan *et al.* 2004),
- Veri toplama alt sistemleri (Buhler *et al.* 1995, Gopalakrishnan *et al.* 2004, Gua and Silva 2004),
- Veri ve sinyal işleme alt sistemleri (Shimakawa *et al.* 2001, Winters and Santos 2002, Mihaylova *et al.* 2003),
- Adaptif kontrol sistemleri ve ilgili yapay zeka alt sistemleri (Dong 2002, Gopalakrishnan 2004, Das *et al.* 2006).

### 2.3 Veri Toplama

Robot hareket ediyorken etrafındaki çevresel bilgileri ve kendi içyapısında oluşan verileri sürekli olarak alması gereklidir. Dış kaynaklardan sadece bir sensor ile veri toplamak dış çevrenin karmaşıklığı ve kararsızlığı yüzünden oldukça zordur. Karmaşık, kararsız ve dinamik dış çevre şartlarında robotun otonomluğunu sağlayacak her çeşit verinin toplanabilmesi için farklı çeşitlerde detektörler gerekmektedir. Bunlardan bazıları: Küresel Konumlandırma Hizmeti (GPS), CCD kameralar, lazer mesafe ölçer, elektronik pusula, fotoelektrik kodlu diskler vs. Bu çeşitlilikte çoklu sensorler kullanıldığında veri toplamada problemler oluşması kaçınılmazdır (Zhou and Gao 2011).

Güvenilir bir robotik sistem, robotların zayıf yinelenebilirlikleri, çalışma alanlarındaki değişiklikler, temas edecekleri materyallerin pozisyonu ve boyutlarındaki değişiklikler nedeniyle çevresindeki belirsizliklere uyum sağlamak zorundadırlar. Bu nedenle çalışma alanlarından geri besleme alınması dolayısıyla da detektör kullanılması zorunludur. Detektörlerden alınan veriler, robotlara istenen görevlerin yapılabilmesi için yardımcı olur. Görme, dokunma ve yakınlık detektörleri kullanılacak algılayıcılardandır (Sood *et al.* 2003).

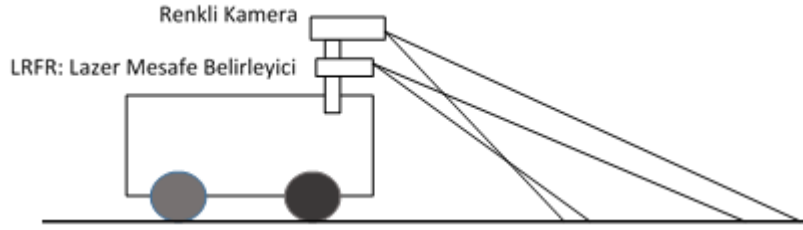
Tek detektörlü sistemler ancak kısmi bilgiler sağlayabilirler, bilinmeyen ya da kısmen bilinmeyen çevre şartlarını yorumlamada eksik kalırlar. Tek bilgi kaynağı baz alınarak

yapılan modelleme işlemlerinde bazı temel kısıtlar vardır. Bu nedenle insanların çok sensörlü sistemlere olan ilgisinin arttığı görülmektedir. Teoride ve pratikte çok sensörlü bütünleşik sistemlerin tek sensörlü sistemlerde bulunan belirsizlik problemini çözebildiği ve çevrelerini daha hassas olarak keşfedip yorumladığı ispatlanmıştır. Çevre koşullarından etkilenecek, çok sensörlü bir sistemin içindeki tek bir sensörden alınan bilgiler belirsiz, kısmi ve kimi zaman hatalı olabildiği gibi sıklıkla coğrafi veya geometrik olarak diğer sensörden gelen veriler ile uyumsuz özellikler içerebilirler. Sensörlerden alınan tüm verilerin işlenebilmeleri için doğru şekilde birbirlerine entegre edilmeleri gerekir (Wu *et al.* 2006).

Engellerden kaçınma hareketi bir mobil robot için vazgeçilmez bir hareketler bütünüdür. Bu konuda literatürde geniş araştırmalar yapılmıştır. Mobil robot etrafına bir dizi halinde ultrasonik detektör yerleştirmek suretiyle engellerden kaçınma işlemi ilgi duyulan bir yöntemdir (Kim and Kim 2010). Ultrasonik mesafe algılayıcıların maliyeti düşüktür ancak düşük açısız çözünürlüğe sahip olmaları nedeniyle hatalara neden olabilmektedirler. Kızıl ötesi detektörler bir diğer ucuz ve basit engel algılayıcılarıdır (Alwan *et al.* 2008). Optik algılayıcılar olduğu için mesafe ölçme kapasitesi ultrasonik detektörlere göre daha uzundur ancak aydınlatma koşullarına karşı hassastır. Robotlardan oldukça uzak mesafelerde bulunan nesnelere doğru olarak algılanması için lazer tarayıcılar kullanılır (Ye and Borenstein 2002). Ancak bu sistemler yüksek maliyetli ve hantal yapıdadır. Son dönemlerde geliştirilen TOF (time of flight) kameraların mobil robotların otomatik geziniminde büyük yararları bulunmaktadır. Lazer tarayıcılar ile karşılaştırıldıklarında; TOF kameralar gerçek zamanlı üç boyutlu bilgi üretirler ve daha hızlıdır (Werner *et al.* 2006, Kim and Do 2012).

Hemen hemen tüm tasarımlarda, çeşitli sensörler aynı anda çalışır ve farklı algılama prensipleri, kapasiteleri ve kapsama alanları paralel olarak kullanılır. Sonuç olarak sensör birleştirme işlemi, farklı detektör verilerini diğer alt sistemler tarafından kullanılacak güvenilir yaklaşımlara ve çevre modellerine çevirmek için gereklidir. Bu bağlamda sensör birleştirme; belirgin şekilde farklı sensörlerden obje tespiti ve çevresel parametrelerin tahmini, harita oluşturma, güzergah hesaplama, hareket planlama ve harekete geçme işlemleri için gelen verilerin entegre edilmesi anlamına gelmektedir (Kam *et al.* 1997).

Şekil 2.3’de Wu *et al.* (2006) tarafından yapılan bir çalışmada entegre edilen 2 adet sensörün şekilsel gösterimi verilmiştir. Burada; CCD kameradan alınan 2 boyutlu görüntü lazer mesafe ölçerden alınan derinlik bilgisi ile 3 boyutlu algılama yapılmaya çalışılmıştır.



Şekil 2.3 İki sensörlü mobil robot (Wu *et al.* 2006).

## 2.4 Yapay Görme

Modern yapay görme araştırmalarının başlangıcı 1960 ‘ların başlarına dayanmaktadır. İlk araştırmalar ofis otomasyonu ya da benzer işler için yazı karakteri tanıma amacıyla yapılan örüntü tanıma sistemleriydi (Roberts 1960, Triplet *et al.* 1965). 1960 ‘lı yıllarda Roberts (1963) tarafından yapılan çalışmada ilk kez bir nesnenin 3 boyutlu temsiliyle 2 boyutlu imgeden çıkartılan özniteliklerin eşleştirilme ihtiyacı belirlenmiştir. Sonraki araştırmalar; özellikle arka plan karmaşıklığının artması, aydınlatma değişkenliğinin artması, zaman, maliyet ve detektör sapması kısıtlarının oluşturduğu problemlerin güvenli ve tutarlı olarak aşılabilmesindeki zorluklar üzerine tesis edilmiştir (Andreopoulos and Tsotsos 2013).

Görme sistemleri konusundaki ilk çalışmalar, “makine görmesi” teriminin çıkış noktasını olan Japonya’ daki Hitachi Laboratuvarlarında yapılmıştır (Ejiri 2007). 1964 yılının başlarında elektronik devrelerde bulunan transistörlerin tel yapıştırma işlemlerinin insan eliyle değil otomasyon yoluyla yapılması hususunda araştırma baskısı bulunmaktaydı. Laboratuvar testlerinde bu sistemlerin %95 hassasiyette çalıştıkları belirlenmiş ve bu insan işçilerin değiştirilmesi için düşük bulunmuştu. Ancak 1973 yılına gelindiğinde tam otomatik montaj makineleri inşa edilmişti (Kashioka *et al.* 1976). Bu çalışma elektronik aygıtların montajında yapay görme sistemlerinin kullanıldığı ilk çalışma olmuştur.

Son yıllar içerisinde; endüstriyel çevrelerde, akıllı fabrika otomasyonuna doğru kararlı bir eğilim olduğu görülmektedir. Artan oranda mekatronik birimlere; kendi kendini optimize edebilen üretim hatları ve kalite kontrol sistemleri uygulanmaktadır. Bu eğilimde görüntü sistemleri özellikle süreç kontrol ve çeşitli alanlarda çevrim içi kalite değerlendirme işlemlerinde son derece önemli bir rol oynamaktadır. Gerçekten de görüntü sistemleri bulunduğu çevreye ait içinde pek çok bilgi barındıran tümenden bir görünüm sunarlar. Yıllar içerisinde durağan ve hareketli görüntüler içindeki nesnelere tanıyabilen ve konumlandırabilen pek çok algoritmalar geliştirilmiştir (Montironi *et al.* 2014).

Kalite takibi için de yapay görme sistemleri kullanılmaktadır. Bu nedenle yapay görme sistemleri; cam, kimya, gıda, ambalaj, elektrik-elektronik, inşaat, sağlık, otomotiv, ilaç, çevre, savunma ve tekstil endüstrisi gibi pek çok sektörde kullanılmaktadır. Endüstriyel kameralar ve yapay görme uygulamaları endüstrinin her alanında üretimin kalitesini arttırmak, yüksek hassasiyet ile hatasız, çok hızlı ve tekrarlanabilir ölçüm ve kontrol yapmak için oldukça yaygın bir şekilde uygulanmaktadır.

Ayrt edilecek nesnelere öznitelikleri birbirinden çok zıt özellikler taşıyor ise akıllı sistem (yapay sinir ağı, bulanık mantık vs.) ihtiyacı oluşmadan basit seçme algoritması ile de nesne tanıma yapılabilir. Nesne özniteliklerinin birbirine yakın olduğu uygulamalarda nesneye ait özel ağırlıklı öznitelik ve karakterlerini değerlendirecek akıllı sistemlere ihtiyaç oluşmaktadır. Eğer nesnelere birbirine çok benzer ise ayırt etme çok daha zor olur (Declercq and De Keyser 1999). Yapay sinir ağı tekniği nesne tanıma ve robot uygulamalarında geniş bir şekilde kullanılmaktadır (Moreno *et al.* 2009, Johan and Prabuwno 2011).

## **2.5 Görüntü Çerçevelerinde Nesne Tespiti ve Tanınması**

Tek bir görüntüde ya da bir video akışında bulunan nesnelere çeşitli algoritma ve yazılımlar desteğiyle bilgisayar sistemleri tarafından tespit edilip anlamlandırılması olarak tanımlanabilir. Bu tespit işlemi; görüntü üzerindeki tek bir nesnenin tespit edilip işlenmesi ve anlamlandırılması şeklinde (plaka tanıma, yüz tanıma vb.) olduğu gibi görüntü akışında hareket eden bir nesnenin uzamsal yer değiştirmesi, davranış şekli, geometrik ve şekilsel değişiklikleri de anlamlandırılabilir.

Bir mobil robot tarafından üzerindeki kameralar vasıtasıyla alınan çevresel görüntülerin yorumlanması büyük önem taşır. Mobil robotlar, çevresel algılama ile elde ettikleri verileri kullanarak uzamsal hareketlerine, haberleşme uygulamalarına, etkileşim sistemlerine yön vermektedirler. Bu nedenle nesne tanıma uygulamaları mobil robotlar için vazgeçilmez unsurlardır.

Görüntü çerçevelerindeki nesnelerin tanınması işlemlerinde ilk aşama nesnenin doğru bir şekilde tespit edilmesidir. Bu işlem için konum-zaman ilişkisi çerçevesinde bazı ayrımların yapılması gerekmektedir. Tespit edilecek nesne, ardışık çerçeveler içinde hareketsiz kalarak engel niteliği alabildiği gibi yine ardışık çerçeveler içerisinde hareketli olabilir. Ayrıca; görüntü çerçevelerinin elde edildiği kameralar sabit olabileceği gibi serbest hareketli ya da tekdüze hareketli olabilir. Bu nedenle; tespit algoritmaları oluşturulurken bu durumun dikkate alınması gerekmektedir.

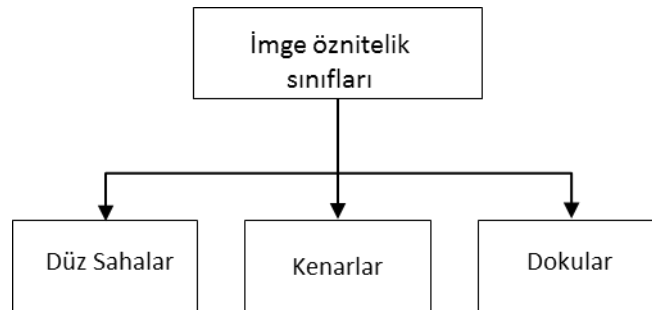
Juang ve Chen (2008) nesne tespit ve tanıma işlemini başlıca 3 adım olarak tarif etmiştir. Birinci adım ön plandaki nesnenin arka plandan ayrılması, ikinci adım nesnenin özniteliklerinin çıkarılması ve üçüncü adım ise nesnenin tanınması olarak ifade edilmiştir.

### **2.5.1 Öznitelik Çıkarma**

Chen (2009)'a göre nesne tanıma için nesneyi temsil eden kesin değerlere ihtiyaç vardır. Bir yapay görme sistemi tarafından bir imge çerçevesi içinde bulunan nesnelerin tanınması ve anlamlandırılması için o nesnelere ait bir takım özellikleri (kenar, köşe, doku, renk vs.) sayısal verilere dönüştürmesi gerekir. Bu işleme öznitelik çıkarma denilmektedir. Bu öznitelikler kimi zaman bir piksel olabildiği gibi pek çok pikselden ve aralarındaki ilişkiler yumağından oluşan değerler kümesi de olabilmektedir. Çıkarılan özniteliklerin bir nesneyi diğerlerinden ayırt edebilecek değerler taşıması gerekir. Ayrıca çıkarılan özniteliklerin sayı olarak yeterli miktarda olması gerekebilir. Bu durum bir miktar hesaplama yükünü arttıracaktır. Ancak bu şekilde sistemin hafızasında kayıtlı model ile eşleştirme işlemi doğru olarak yapılabilir. Doğru eşleştirmeler için çıkartılan özniteliklerin konumdan, dönmeden ve ölçekten bağımsız yani değişmez olmaları büyük önem taşır. Eşleştirilecek imge çiftleri farklı görüş açılarından görüntülenmiş, farklı çözünürlüklerde ya da farklı ölçeklerde alınmış olabilir. Böyle durumlarda öznitelikler değişmez yapıda olmadıkları takdirde iki imge arasındaki eşleştirme başarısız olacaktır.

Bu nedenle öznitelik çıkarma işleminde değişmez imge bilgileri kullanılarak dönme, konum ve ölçekten bağımsız değerler elde etmek gerekir. Ölçekten bağımsız yöntemler arasında SIFT (Lowe 1999) ve SURF (Bay *et al.* 2006) yöntemleri sayılabilir.

Yap vd. (2009) 'a göre imgelerdeki türdeşlik bağımsız formlarda bilgi taşıyan birden fazla imge özneliği tipi olduğuna işaret etmektedir. Her ne kadar farklı imgeler arasındaki varyasyonlar büyük olsa da pek çok sayıda imge çok az sayıda öznitelik tipi ile ifade edilebilmektedir. Bu öznitelik tipleri; düz sahalar, dokular ve kenarlardır. Düz sahalar imge içeriğinde en çok yer kaplayan alanlardır. İmgeyi daha uzaktan görüntülendikçe detayları kaybolarak düz sahaya dönüşür. Düz sahalar genelde tek bir gri seviye değeri ile temsil edilirler ve öznitelik çıkarmak için kısıtlı alanlardır. Düz sahaların aksine kenarlar imge içeriğinde en az yeri kaplarlar ancak imge hakkında pek çok bilgi içerirler. Kenar çizgileri sayesinde orijinal imge hakkında rahatlıkla çıkarsamalar yapılabilmektedir. İmgelerin içeriğinde bulunan nesnelere imgenin dokusunu oluşturur. Bu dokular her ne kadar gürültü içerse de aslında tanınabilecek örüntüler içerir. Yap vd. (2009) tarafından belirtilen öznitelik sınıflarının şekilsel ifadesi Şekil 2.4'de gösterilmiştir.



**Şekil 2.4.** Üç önemli imge sınıfı.

Çetin (2011) tarafından belirtilen tanımlamada öznitelikler global ve bölgesel öznitelikler olarak ikiye ayrılmıştır. Global öznitelikler imge içindeki nesnelere renk yapıları, renk dağılımları, dokuları ve nesnelere merkezleri gibi imgelerin genelini kapsayan özellikler olarak belirtilmişken bölgesel öznitelikler kenarlar, köşeler, kenarların eğrilikleri ve merkeze uzaklıklar gibi özellikler olarak belirtilmiştir. Çıkarılacak özniteliklerin global veya bölgesel seçilmesinde mevcut koşullar önem taşır. Örneğin, imge üzerindeki bir piksel öbeğinin insan yüzü olup olmadığına karar verilmesi için global özniteliklerden



faydalanmak gerekirken yüz tanıma yapılmak isteniyorsa bölgesel özniteliklerden faydalanmak doğrudur.

Türkoğlu (1996) 'nın yaklaşımına göre öznitelikler genel, yapısal ve ilişkisel öznitelikler olarak üçe ayrılmıştır. Genel öznitelikler nesnelerin çevre uzunlukları, alanları, eylemsizlik momentleri ve fourier dönüşüm özellikleri gibi bilgileridir. Bu öznitelikler kolay ve hızlı bir şekilde çıkartılıp vektörel olarak düzenlenebilmektedir. Ancak bu öznitelikler çıkarılırken bir takım kısıtlar bulunmaktadır. Bunlar:

- Görüş açısına göre bağımsız değildir,
- Nesnelere bir birini örtmemelidir ve şekilsel bozukluk içermemelidir,
- Bir imgedeki tüm nesnelerin ayrıştırılması için tek bir eşik değeri kullanılmalıdır.

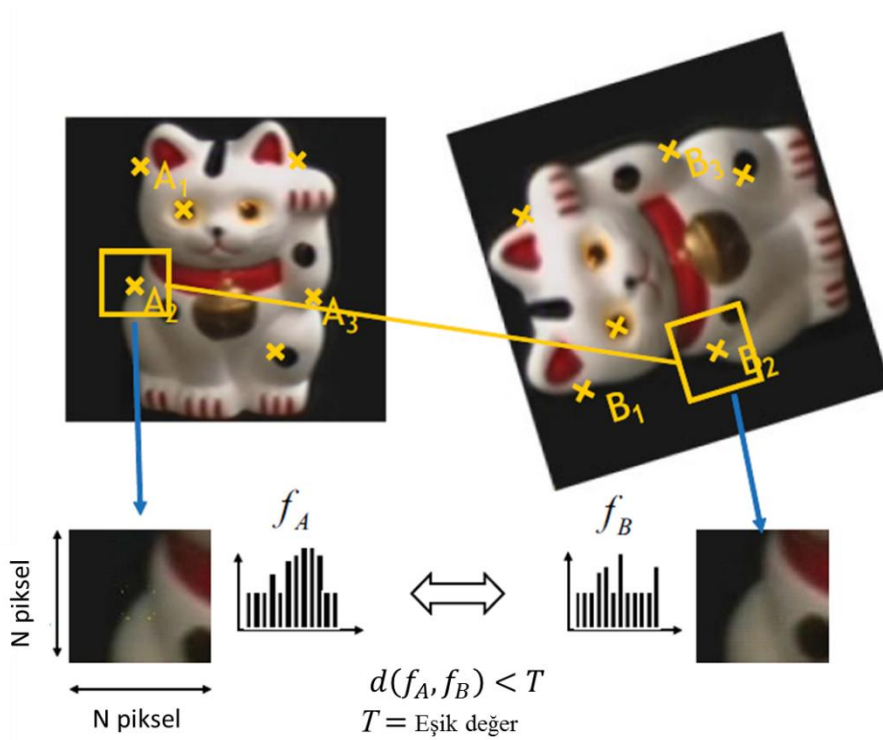
Yapısal öznitelikler, doğru parçası, eğri parçası, köşe ve kenar gibi nesne sınırlarını oluşturacak piksel bilgilerinden oluşur. İlişkisel öznitelikler ise kenar, köşe gibi özniteliklerin geometrik ilişkilerini dikkate alır.

Nixon ve Aquado (2012) yaptıkları tanımda öznitelikleri düşük seviyeli ve yüksek seviyeli olmak üzere iki kategoride değerlendirmişlerdir. Düşük seviyeli öznitelikler, herhangi bir uzamsal ilişki içermeyen şekilsel bilgiler dışında imge üzerinden kolayca elde edilebilen basit yapılardır. Bunlara örnek olarak çeşitli seviyelerde kenar belirleme işlemi verilebilir. Bu tür özniteliklerin çıkarılabilmesi için pek çok basit ve gelişmiş yaklaşımlar mevcuttur. Yüksek seviye öznitelik çıkarma işlemi imge içinde bulunan şekiller ve nesnelere ile ilgilidir. Örneğin yüz tanıma işleminde gözlerin elipsoidal olması, ağzın çift çizgi olarak ifade edilebilmesi şekilsel olarak tanımayı sağlayan unsurlardır. Bu tür nesnelere şekilsel özelliklerinden ayırt edilirler. Bu tür öznitelik çıkarma işlemleri insanların dünyayı algılama şekli ile aynı özellikleri taşımaktadır. Çok karmaşık sahneler daha basit şekiller halinde ayrıştırılabilir. Bu nedenle görüntü analizinde imge üzerindeki şekiller belirli bir düzen içinde sıralanarak işlem yapılmaktadır. Her iki öznitelik çıkarma işleminde de sonuçların farklı koşullar altında değişmemesi için değişmez öznitelikler taranır. Bu şekilde bulunan nesnelere konum, boyut ve yönelimden bağımsız olarak elde edilmiş olur. Basit anlamda bazı taranması gereken değişmezlikler aydınlık seviyesi, konum, dönme, nesnenin bir kısmının bir başka nesne tarafından kapanması veya gizlenmesi olarak sayılabilmektedir.

Grauman ve Leibe (2011) değişmez öznitelik yapılarının amaçlarının imgelerin üzerindeki yerel yapıların eşleştirilebilmesi için bir temsil sağlama olduğunu belirtmiştir. Yazarlar, güçlü öznitelik yapılarının ortaya çıkarılabilmesi için öznitelik çıkarma operatörlerinde bulunması gerekli iki önemli ölçüt belirtmişlerdir.

- Öznitelik çıkarma işlemi kesin ve tekrarlanabilir olmalıdır. Ancak bu şekilde iki ayrı imgeden çıkarılan aynı öznitelikler iki ayrı imgede de aynı nesneyi işaret edebilir.
- Özniteliklerin kendine özgü olması gerekir. Ancak bu şekilde farklı imge yapıları birbirinden ayırt edilebilir.

Yazarlar tarafından sıralanan yeterli sayıda ve doğru öznitelikler çıkarılabilmesi için gerekli işlem adımları aşağıda belirtilmiş ve Şekil 2.5’de ifade edilmiştir.



Şekil 2.5 Yerel öznitelik yapılarıyla nesne tanıma prosedürü (Grauman and Leibe 2011).

İşlem adımları:

- a. Belirgin anahtar noktalar bulunur,

- b. Her ana hat nokta etrafında ölçekten ya da uzamsal dönüşümlerden bağımsız bir alan tanımlanır,
- c. Bu alanın içeriği normalize edilir,
- d. Normalize edilen alandan bir tanımlayıcı hesaplanır,
- e. Yerel tanımlayıcılar eşleştirilir.

## 2.5.2 Nesne Tespiti

Görüntü işleme temelinde yeni teknolojilerin geliştirilmesiyle birlikte, nesne tanıma konusu bilgisayar görme ve video bilgi çıkarımı uygulama alanlarında en önemli araştırma konularından birisi olmuştur (Foresti *et al.* 2005, Huang *et al.* 2008).

Bir video akışı içerisinde hareketli nesnenin tespit edilebilmesi için belirli bir zaman döngüsü içerisinde görüntü çerçevelerinin çıkarılarak işlenmesi gerekir. Bu nedenle hareketli nesne tespiti ile hareketsiz nesne tespiti arasında benzerlikler bulunur. Bunların arasındaki fark; video akışındaki nesne tespit işleminin nesnenin zamanla akan görüntü çerçevelerindeki hareket karakteristiğine daha fazla bağlı olmasıdır.

Belirli bir görüntü akışı içerisinde hareketli bir nesnenin tespiti için pek çok yöntem geliştirilmiştir. Bunlardan başlıcaları optik akış tabanlı metotlar (Jung and Sukhatme 2004, Kim *et al.* 2010), arka plan kestirime dayalı metotlar (Zhou and Aggarwal 2001, Razali and Adznan 2006) ve ardışık çerçevelerin çıkarılmasına dayalı metotlardır (Weng *et al.* 2010).

### 2.5.2.1 Optik Akış Yöntemi

Optik akış yöntemi, görüntü çerçeveleri üzerindeki nokta, kenar veya alan gibi piksel gruplarının ardışık çerçeveler arasında yoğunluk (gri seviye veya renk) değerlerinin değişimini, uzamsal yer değişimlerini ve bu yer değiştirmenin yönünü inceler. Bu şekilde yukarıda bahsedilen piksel gruplarına ait yön ve şiddet değerlerini içeren hız vektörleri tahmin edilmeye çalışılır. Yani hareketli nesnelere tespit edilmiş olur. Matematiksel hesap yoğunluğu nedeniyle arka plan kestirimi yöntemi kadar sık kullanılmamaktadır.

Bu yöntemde; görüntü çerçeveleri arasındaki  $t_1$  den  $t_2$  'ye geçen çok küçük bir zaman aralığında bir piksel grubunun yer değiştirdiği durumda piksellerdeki yansıma ve

aydınlanma değerlerinde ihmal edilebilecek kadar küçük bir değişiklik olduğu varsayımı yapılır. Bu durum Denklem 2.1’de gösterildiği şekilde ifade edilebilir:

$$f(x + \Delta x, y + \Delta y, t + \Delta t) \approx f(x, y, t) \quad (2.1)$$

Bu ifadede  $f(x, y, t)$  işlevi  $(x, y)$  noktasındaki ve  $t$  anındaki yoğunluk değerini,  $\Delta x$ ,  $\Delta y$  uzamsal değişikliği,  $\Delta t$  ise zamandaki değişimi ifade etmektedir. Bu eşitliği sol tarafındaki ifade Taylor serisi açılımına tabi tutulduğunda Denklem 2.2 elde edilir.

$$f(x + \Delta x, y + \Delta y, t + \Delta t) = f(x, y, t) + \frac{\partial f}{\partial x} \Delta x + \frac{\partial f}{\partial y} \Delta y + \frac{\partial f}{\partial t} \Delta t + y.m.t \quad (2.2)$$

Burada;  $\frac{\partial f}{\partial x}$ ,  $\frac{\partial f}{\partial y}$ ,  $\frac{\partial f}{\partial t}$  ifadeleri  $x$ ,  $y$ ,  $t$  boyutlarında  $f(x, y, t)$  görüntü işlevinin kısmi türevleridir. Bu ifadede çok küçük piksel komşuluk değerleri üzerinde çalışma yapıldığı için yüksek mertebeli terimler ihmal edilmiştir. Sonuç olarak Denklem (2.2) üzerinde Denklem (2.1) yerine konulduğunda Denklem 2.3 elde edilir.

$$\nabla I \cdot v + I_t = 0 \quad (2.3)$$

Bu ifade optik akış kısıt eşitliği olarak adlandırılır. Burada;  $\nabla I = (I_x, I_y)$  uzamsal gradyan,  $v=(u,v) = (\Delta x, \Delta y)$  optik akış vektörü,  $I_t$  ise zamansal gradyandır (O’Donovan 2005).

Optik akış yönteminde sıklıkla kullanılan algoritmalarından bir kısmı aşağıda belirtilmiştir.

- Diferansiyel bazlı algoritmalar (Lucas and Kanade 1981, Horn ve Schunk 1981),
- Benzetim tabanlı algoritmalar (Barnard and Thompson 1980),
- Öznitelik tabanlı algoritmalar (Morovec 1977).

### 2.5.2.2 Arka Plan Kestirimi Yöntemi

Arka plan kestirim yöntemi basitçe anlık görüntü çerçevesi ile arka plan modeli olarak da adlandırılan sahnenin statik arka planı arasındaki fark olarak tanımlanabilir. Çıkarma

sonucu  $Th$  eşik değerinden büyük olan her bir  $i,j$  pikseli hareketli nesneye ait olarak nitelendirilebilir. Çıkarma işlemi Denklem 2.4’de gösterildiği şekilde ifade edilebilir.

$$|Hesaplanmış Arka Plan_{ij} - Anlık Çerçeve_{ij}| > Th \quad (2.4)$$

Arka plan kestirimi yönteminde en önemli adım karmaşık arka plan çeşitliliğine uyum sağlayabilecek bir model geliştirmektir. Bir başka önemli adım ise sistemin tüm performansını etkilemesi dolayısıyla dikkatli seçilmesi gereken eşik değeridir (Al-Khateeb and Petrou 2010).

Arka plan modeli oluşturmak için bir referans arka plan görüntü çerçevesi seçilerek model oluşturulmalı ve zamana bağlı olarak üretilen tüm çerçeveler boyunca bu modelin güncellenmesi gerekir. En basit haliyle arka plan modelleme Denklem 2.5’de gösterildiği şekilde ifade edilebilir.

$$B_{s,t+1} = (1-\alpha)B_{s,t} + \alpha \cdot I_{s,t} \quad (2.5)$$

Bu denklemde  $B_{s,t}$  t anındaki arka plan modeline ait s pikselinin şiddet değerini,  $I_{s,t}$  t anındaki s pikselinin görüntü çerçevesini,  $\alpha$  ise değeri 0 ila 1 arasında olan güncelleme sabitini ifade etmektedir (Benezeth *et al.* 2008).

Literatürde oldukça fazla kestirim modeli bulunmaktadır. Sıklıkla kullanılan arka plan kestirim algoritmaları aşağıda belirtilmiştir.

- Gauss Karışım modeli, GMM (Zha *et al.* 2009, Hu *et al.* 2009, Jarraya *et al.* 2010),
- Kernel Yoğunluk Kestirimi modeli, KDE (Jarraya *et al.* 2010),
- Hibrit yöntemler, GMM-KDE (Ding *et al.* 2010),
- Bulanık mantık tabanlı yöntemler (Patil *et al.* 2004, Juang and Chen 2008),
- Dönüşüm tabanlı yöntemler, Ayrık Dalgacık Dönüşümü – DWT (Wahi *et al.* 2010) ,
- Yapay sinir ağları tabanlı yöntemler (Luque *et al.* 2008) ,
- Tahmine dayalı yöntemler (Ridder *et al.* 1995).

### 2.5.2.3 Çerçeve Farkı Yöntemi

Kullanılabilecek en basit yöntemdir. Sadece kendinden önceki görüntü çerçevesini kullanabildiği için arka plandan ayrılacak hareketli nesneye ait tüm pikselleri tespit etme yeteneği yoktur. Eşik değeri doğru bir şekilde belirlenemediği takdirde çerçevelere binmiş görüntülerin başarıyı etkileyebilmesi nedeniyle çerçevelerin öncelikle çeşitli süzgeçlerden geçirilmesi gerekir. Söz konusu yöntem Denklem 2.6'da gösterildiği şekilde matematiksel olarak ifade edilir:

$$M(x, y, t) = \begin{cases} 1, & I(x, y, t) - I(x, y, t - 1) > \tau \\ 0, & I(x, y, t) - I(x, y, t - 1) < \tau \end{cases} \quad (2.6)$$

Bu denklemde;  $I(x, y, t)$  t anında  $(x, y)$  konumundaki pikselin şiddetini,  $I(x, y, t - 1)$  t-1 anında  $(x, y)$  konumundaki pikselin şiddetini,  $\tau$  eşik olarak belirlenen piksel şiddetini,  $M(x, y, t)$  ise çıkarma ve eşikleme neticesinde elde edilen görüntü işlevini ifade etmektedir. Her ne kadar hızlı ve arka plan değişimlerine çabuk adapte olan bir yöntem ise de eşik değerine karşı aşırı hassastır. Hatalı ya da en iyi olmayan eşik değerlerinde kesin sonuçlar üretememektedir. Bir kısım algoritmalarda bir önceki görüntü çerçevesi yerine kendinden önceki N adet çerçevenin ortalaması da alınabilmektedir (Ergezer 2007).

### 2.5.3 Nesne Tanıma ve Sınıflandırma

İlk tanıma sistemleri kromozom tanıma işlemleri için biyomedikal araştırmalarda kullanılmıştır. Bu çalışma başlangıçta sınırlı bir etkiye sahip olsa da önemi daha sonra açık bir şekilde anlaşılmıştır. Nesne tanıma teknolojileri gıda endüstrisinde (örnek olarak zirai ürünlerin otomasyon kullanılarak sınıflandırılması verilebilir), elektronik ve makine üretim endüstrilerinde (montaj otomasyonu ve endüstriyel denetim maksadıyla), ilaç endüstrisinde (tablet ve kapsüllerin sınıflandırılmasında) kullanılmaktadır. Nesnelere temsil etmekte kullanılan modeller aynı zamanda medikal görüntüleme aygıtları tarafından beyin, kalp karıncıkları vb. anatomik yapıların tutarlı olarak bölütlenmesinde etkin olarak kullanılmaktadır. El yazısı karakter tanıma sistemleri dokümanların sayısal bilgiye dönüştürülmesi ve otomatik olarak indekslenmesinde kullanıldığı gibi posta sıralama cihazlarında da kullanılmaktadır. Bununla beraber trafik görüntüleme ve plaka

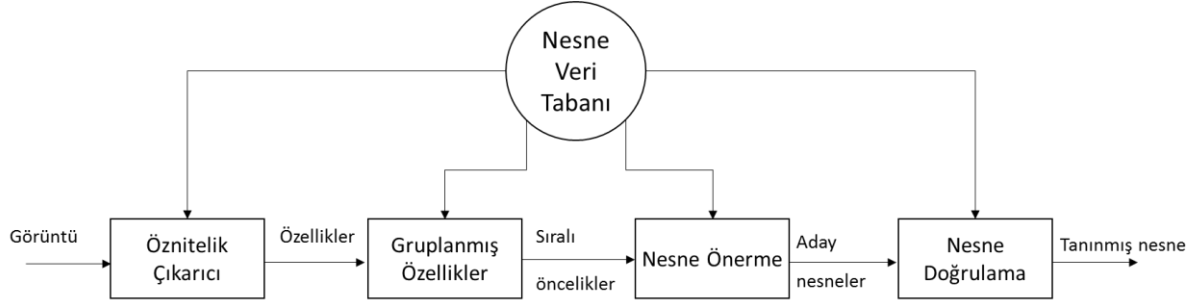
tanıma sistemlerinde, ATM cihazlarında para tanıma sistemlerinde de başarılı olarak kullanılmaktadırlar. Parmak izi tanıma, iris tanıma, parmak damar ve avuç içi damar tanıma gibi biyometrik görüntü sistemleri de kanun uygulayıcılar tarafından kabul görmüş ve genişçe kullanılan sistemlerdir (Andreopoulos and Tsotsos 2013). Ayrıca; video gözetleme uygulamalarında, otonom araçların yönetilmesinde, video sıkıştırma ve indeksleme işlemlerinde, hareketli nesnelerin akıllı izlenmesinde, otomatik hedef tanıma, yüz tanıma, robotik uygulamalar ve uzak görüntüleme teknolojilerinde kullanılmaktadır (Jarraya *et al.* 2010, Yu *et al.* 2012).

Bir imge akışı içinde nesne tanıma işlemi insan için basit bir işlem iken bilgisayar sistemlerinde algoritmalar yoluyla tanıma işlemi oldukça zordur. Ayrıca; resim ve video imgelerinde gürültü olarak tabir edilen ve tanımayı güçleştiren bir kısım durumlar da söz konusudur. Bir imge akışında nesnenin şekli, yönü, hızı veya boyutları imge çerçeveleri arasında değişebilmektedir. Hareketli nesneler, görüntü kaynağının açısı değişimi, aydınlatmadaki değişiklikler, komşu nesnelere yansımalar nedeniyle geniş bir renk değişimi gösterebilmektedirler (Murshed *et al.* 2011). Düşük kalitedeki görüntü kaynağı sebebiyle oluşan gürültüler ile birlikte kamera titremesi, kameranın otomatik ayarlama yapması, günün saatine göre ışık değişimleri, nesnenin kamufle olması, görüntü ön planında açıklıklar bulunması, hareketli arka plan nesnelere, arka plana sonradan dahil olan nesnelere, hareketsiz ön plan nesnelere ve gölgeler de gürültü olarak sayılabilmektedir (Bouwman 2014). Görüntü arka planı olarak hareketsiz olmaları öngörülen ağaçların sallanma hareketi, fiskiye ve deniz dalgaları da gürültü oluşturan unsurlardır (Zha *et al.* 2009). Tüm bu gürültü kaynakları nesneyi tanıma üzerine gerçekleştirilen algoritmaların tutarlılığını etkilemektedir.

Nesne tanıma konu başlığı 1960 'lı yılların başından bu yana aktif bir araştırma konusu olmuştur. Sağlam ve tutarlı bir çözüm için pek çok model ve yöntem geliştirilmiştir. İnsanların nesnelere tanıması, insanların nesnelere ait özellikleri algulamaları ve bu özellikleri daha önce öğrendikleri bilgiler ile karşılaştırmaları ve benzetme ile olur. Nesne tanıma sistemlerinde de benzer yöntem söz konusudur. Önce tanınacak nesnelere için veri tabanı oluşturulur. Bu veri tabanı sisteme öğretilir ve karşılaştırma algoritması ile nesne, daha önce öğrendiği ile kıyas yaptırılıp nesnenin tanınması sağlanır. Nesne tanımadaki başarı oranları; kullanılan modellere, görüntü işleme ve öğrenme algoritmasına, çevre şartlarına ve donanım özelliklerine bağlı değiştiği için değişkenlik göstermekte ve benzer

çalışmalar ile detaylı karşılaştırma yapmayı zorlaştırmaktadır (Johan and Prabuwno 2011).

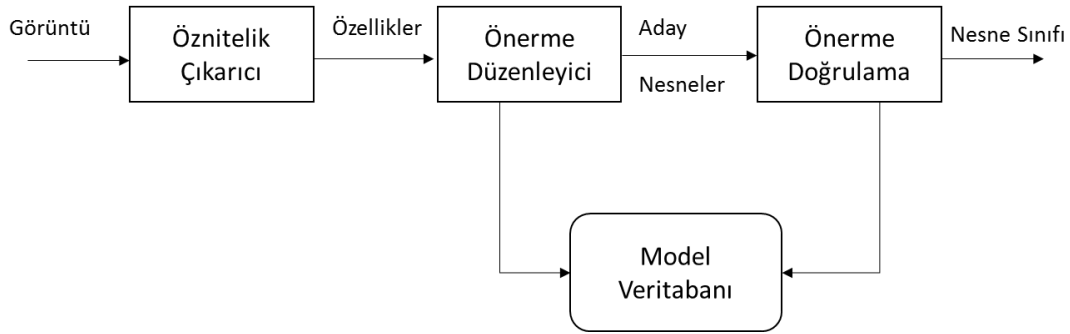
Andreopoulos ve Tsotsos (2013) tarafından yapılan çalışmada tipik bir nesne tanıma sistemi Şekil 2.6’da gösterildiği gibi tanımlanmıştır.



Şekil 2.6 Tipik bir nesne tanıma sistemi (Andreopoulos and Tsotsos 2013).

Bu sistem; görüntü özniteliklerinin çıkarılması, bulunan özniteliklerin gruplanması, nesne önerimlerinde bulunması ve aday nesnelere doğrulanması neticesinde nesnenin tanınması şeklinde özetlenebilmektedir.

Nagabhushana (2005), nesne tanıma sistemini Şekil 2.7’de gösterildiği şekilde bileşenlerine ayırmış ve beş adımda ele almıştır.



Şekil 2.7 Nesne tanıma sistemi bileşenleri (Nagabhushana 2005).

**Model Veri tabanı:** Sistem tarafından bilinen tüm modelleri içerir. Veri tabanındaki bilgiler tanıma sistemi tarafından kullanılacak yaklaşımlara bağlıdır. Niteliksel veya işlevsel tanımlardan, hassas geometrik yüzey bilgilerine kadar değişim gösterebilir. Çoğu zaman modeller öznitelik vektörlerinden oluşurlar.



**Öznitelik Çıkarıcı:** Görüntülere çeşitli işlemler uygulayarak önerme oluşturmada yardımcı olacak özniteliklerin yerlerini belirler. Öznitelikler tanınacak nesnenin tipine ve uygulanacak modelin organizasyonuna bağlı olarak değişmektedir.

**Önerme Düzenleyici:** Görüntüde saptanan öznitelikler ile nesnelere arasında eşleşme olabilecek adayları belirler.

**Önerme Doğrulama:** Önermeleri doğrulamak amacıyla nesne modellerini kullanır ve nesne olasılıklarını kullanılabilir hale getirir. Sonrasında en yüksek olasılığa sahip olan nesneyi seçerek tanıma işlemini sona erdirir.

Bu işlemlere ait adımlar Nagabhushana (2005) tarafından aşağıdaki şekilde ifade edilmiştir.

1. **Nesne veya model sunumu:** Nesnelere model veri tabanında hangi şekilde temsil edileceği, bu modellerde nesnelere hangi önemli özniteliklerinin bulunacağı ele alınmaktadır. Bazı nesnelere geometrik öznitelikler belirgin ve etkin olabilirken, bazı nesnelere ise kendine özgü veya işlevsel öznitelikler önem taşıyabilmektedir. Bir nesnenin temsil edilişi bilgi kirliliğine yol açmadan nesneyle ilgili tüm bilgileri içermeli ve bu bilgileri nesne tanıma sisteminin tüm bileşenleri tarafından kolayca erişilebilecek bir formda organize edilmelidir.
2. **Öznitelik çıkarma:** Görüntüye ait hangi öznitelikler çıkarılacağını ve bu özniteliklerin nasıl güvenilir şekilde çıkarılabileceği analiz edilir. Pek çok öznitelik 2 boyutlu görüntülerden elde edilir ancak her biri 3 boyutlu nesnelere temsil etmektedirler. Bu nedenle bazı özniteliklerin güvenilir olarak elde edilmesi kolay iken bazılarının zordur.
3. **Öznitelik-model eşleştirme:** Görüntüdeki öznitelikler ile veri tabanında bulunan modelin eşleştirilmesi gerekir. Pek çok tanıma işleminde kapsamlı bir eşleştirme yaklaşımı nesne tanıma sorununu çözecektir ancak çok yavaş kalıp fayda getirmeyebilir. Bir eşleştirme yaklaşımı geliştirirken etkili öznitelikler ve verimli bir eşleştirme tekniği bir arada ele alınmalıdır.
4. **Önerme düzenleme:** Bu aşamada seçilen öznitelik eşleştirme yöntemi baz alınarak olası nesnelere belirlenecektir ve hesaplanan olasılıklar mümkün olan nesnelere atanacaktır.

5. **Nesne doğrulama:** Analiz edilen görüntüde bulunan aday nesnelere grubundan en yüksek olasılığa sahip nesnenin model kullanılarak seçimi gerçekleştirilir. Olası her nesnenin varlığı kendi nesne modelleri üzerinden doğrulanmaktadır. Bir nesnenin varlığını doğrulamak ya da görmezden gelmek için her uygun önermenin incelenmesi gerekmektedir.

Chen (2009)'a göre nesne tanıma için nesneyi temsil eden kesin değerlere ihtiyaç vardır. Verilen bir imgeden öznitelik vektörünü oluşturmak örüntü tanımanın en önemli konularından biridir. Öncelikle en ile boy oranı basit bir nesne özneliğidir ve kolayca hesaplanabilir. Sonrasında nesnenin temel özniteliklerini bulmak için konturu yani kenarları çıkartılabilir. Nesnenin konturunun konumlarından elde edilecek veriler ile Ayırık Fourier Dönüşümü yapılabilir ve dönüşüm sonucu elde edilen veriler öznitelik olarak kullanılabilir. Bu değerler tekil olduğu gibi aynı zamanda dönme, öteleme ve ölçeklemeden bağımsızdır.

Nesne tanıma işleminde şekilsel temsilin (shape representation) önemi nedeniyle oldukça fazla sayıda şekilsel tanımlayıcı teknikleri geliştirilmiştir. Bunlara örnek olarak Zhang ve Lu (2004) ve Khabou vd. (2007) verilebilir. Şekilsel temsil yöntemleri iki ayrı kategoride değerlendirilebilir. Bunlar alan tabanlı (Ghoshtasby 1985) ve sınır tabanlı (Belongie 2002) yöntemlerdir. Alan tabanlı yaklaşım nesnenin tüm alanıyla ilgilenirken sınır tabanlı yaklaşım sadece nesnenin sınır çizgileriyle ilgilenir. Sınır tabanlı yöntemler alan tabanlı yöntemlere göre çok daha yaygındır çünkü iki boyutlu imgelerde nesnelere sınır çizgilerinin piksel sayıları alanlarının tamamındaki piksel sayısına göre önemli derecede azdır. Pek çok dalgacık tanımlayıcı ve şekilsel tanımlayıcı ki bu tanımlayıcılar değişmez momentler (Hu 1962, Gurta and Srinath 1987), Fourier tanımlayıcılar (Zahn and Roskies 1987) ve otoregresif modeller (Dubois and Glanz 1972) gibi global öznitelikler içerdiği gibi zincir kodları, şekilsel (Belongie *et al* 2002) ve eğrisel ölçekleme gösterimi (Mokhtarian and Abbasi 2002) gibi bölgesel öznitelikler içerir. Bu tanımlayıcılar farklı örüntülerin sınır çizgilerini tanımlamak için geliştirilmişlerdir. Global öznitelikleri kullanan yöntemlerin hesaplama maliyetlerinin düşüklüğü ve boyut değişikliği, öteleme hareketi ve yönelime karşı değişmezlik avantajları vardır. Ancak buna karşılık doğru tanıma oranları düşüktür ve sabit olmayan örüntülerin sınıflandırılmasında başarısız olma ihtimalleri yüksektir (Pan and Xia 2008).

Çetin (2011) 'e göre nesne tanıma görünüş tabanlı ve öznitelik tabanlı olarak ikiye ayrılabilir. Görünüş tabanlı tanımda bir şablon kullanılır ve bu şablon diğer imgenin üzerinde sıralı olarak kaydırılarak eşleşme araştırılır. Öznitelik tabanlı tanıma işleminde ise tanınacak nesneye ait öznitelikler imge içinden çıkarılan öznitelikler ile karşılaştırılır. Aralarındaki farkın en az olduğu öznitelik grupları taranarak eşleşmeler araştırılır.

Yu vd. (2012) hareketli nesnelerin tanınmasında arka plan çıkarımı ve optik akış olmak üzere başlıca iki yöntem sunmuşlardır. Jarraya vd. (2010) yaptıkları çalışmada hareketli nesne tanıma işlemini; görüntü çerçeveleri farkı (inter-frame difference - IFD), arka plan kestirimi (background modelling - BM), optik akış bazlı yöntemler (optical flow -OF) ve hibrit yöntemler olmak üzere dört ayrı kategoride değerlendirmişlerdir.

Nesne tanıma sistemlerinde en önemli parça sistem yazılımıdır. Sistemin nesne tanıma işlemini daha kesin gerçekleştirebilmesi için yazılımın çevreden aldığı girdi verilerinin de hatasız olması gerekir. Çevre girdisinde ışıklandırma ve kamera özellikleri yazılıma giden görüntü kalitesini etkileyen en önemli unsurlardır (Johan and Prabuwno 2011).

Fu ve Liu(2001) tarafından yapılan çalışmada; bir stereo kamera üzerinden aracın rotası boyunca alınan görüntülerden elde edilen nesnelere yayalar, araçlar ve diğerleri olarak 3 sınıfa ayrılmıştır. Çalışmada; engelin sınıfı belirlendikten sonra konumu ve araca olan uzaklığı hesaplanmış ve son olarak sürücünün uyarılıp uyarılmamasına karar veren bir sistem tasarlanmıştır. Yayaların belirlenmesinde; çeşitli kıyafette, cinsiyette ve pozisyonda bulunan yayaların yüzleri ve dış şekillerinden oluşan geniş bir veri tabanından faydalanılmıştır. Veri tabanında bulunan her yaya görüntüsünün dış şekli, göz, ağız ve burun parçaları Sobel kenar bulma yöntemiyle elde edilmiştir. Yapısal veri tabanı oluşturulurken en uygun kümeleme sonucunu elde etmek için Genetik K-Ortalamlar Algoritması kullanılmış ve benzerlik ölçümleri için ise Housdorff Uzaklığı yöntemi kullanılmıştır. Yapılan bu çalışma neticesinde 330ms~660ms aralığında %17 kayıp oranıyla araç tanıma döngüsü gerçekleştirilebilmiştir. Arka plan görüntüsündeki karmaşıklık arttıkça kayıp oranının arttığı ifade edilmiştir.

Zhou ve Aggarwal (2001) video kayıtlarından alınan görüntülerde hareket, mekânsal konum, renk ve şekil bilgilerini bütünleştirerek imge gruplarının tespiti ve izlenmesi

üzerine bir çalışma yapmıştır. Bu çalışmada; medyan filtresi tarafından otomatik güncellenen arka plan ile imge gruplarının piksel bazlı farkları kullanılarak söz konusu imge grupları bölütlenmiştir. Bu metotta temel olarak yapılan işlem, nesnelere görüntüye girerken şablonlar ile nesnelere arasında bir eşleşme kurulmaya çalışılmasıdır. Görüntülerden öznitelik çıkarma işlemlerinde; ışık değişimi etkisini azaltmak amacıyla Temel Bileşen Analizi (PCA) uygulanmıştır. Sınıflandırma işlemlerinde ölçüt olarak hareket yönü değişimleri ve yoğunluk değişimleri kullanılmıştır. Yapılan çalışmada; hareketli nesne tespit edilirken öncelikle hareket eden nesnenin arka plan görüntüsünden ayrılmış, ayrılma işlemi için bölütleme (segmentation) ve arka plan çıkarma (background subtraction) yöntemleri uygulanmıştır. Bu şekilde arka plandan ayrılan hareketli nesnenin öznitelikleri çıkartılarak üç farklı kategoriye ayrılmıştır. Bu kategoriler; tek kişi, insan grubu ve araç kategorileridir. Bu çalışma sonucunda insan ve araçların sınıflandırmaları yapılarak hareket yönleri kararlı olarak elde edilebilmiştir.

Gamez ve Devy (2013) bir mobil robot tarafından hareketli nesnelere tanımlanması ve takibi için aktif görüntü tabanlı bir yöntem önerilmiştir. Bu yöntemde bir görüntü üzerinde tanımlanan hareket gridi, her bir görüntü pozisyonu üzerinde hareketli nesneyi bulmayı sağlamaktadır. Bu metot farklı yaklaşımların bir araya getirilmesiyle oluşturulmuştur. Bu yaklaşımlar; geometrik çıkarsama yaklaşımı ve model bazlı nesne tanıma ve kavramsal çıkarsama yaklaşımlarıdır. Bu şekilde robot çevresindeki hareketli nesnelere varlığı saptanmaktadır. Stereo olarak bağlanmamış iki ayrı mono kameradan elde edilen görüntüler Sola vd. (2007) tarafından sunulan BiCam örneğine göre işlenmiş; ilk olarak robotun pozisyonu görüntü tabanlı EKF-SLAM yöntemiyle tahmini olarak bulunmuş, sonrasında tanımlanacak ve takip edilecek olan hareketli nesnelere pozisyonları ve hızları SLAMMOT yöntemiyle tahmini olarak bulunmuştur.

Hu vd. (2009) tarafından yapılan çalışmada karmaşık bir çevre ortamında hareket eden nesnelere tanımlanması için arka plan modelleme yöntemi kullanılmıştır. Görüntü algılama açısından arka plan görüntülerinin ön plan görüntülerine göre daha kararlı bir yapıda olması nedeniyle, bir video akışında arka planın belirlenerek elenmesi ve hareketli nesnelere kalan görüntüden çıkartılması prensibi benimsenmiştir. Yazarlar tarafından yapılan incelemede arka plan modelleme işlemlerinde;

- Ridder vd. (1995) her bir pikseli Kalman filtresi kullanarak ortamdaki ışık değişikliklerinden etkilenmeyecek bir şekilde modellendiği ancak görüntü geri yüklemesi oldukça yavaş olduğu,
- Richard vd. (1997) tarafından belirlenip takip edilecek nesnelere için çok sınıflı istatistiksel model kullanıldığı ancak arka plan modelleri olan piksel bazlı tekli Gaussian yönteminin dış ortamlardaki başarısı ile ilgili herhangi bir rapor bulunmadığı,
- Staufer ve Grimson (1999) tarafından Karışık Gaussian yöntemiyle arka planı görüntüden ayıklamayı başarıldığı, bu yöntemin ışık değişikliklerinin etkisi altında kararlı çalışan bir yöntem olduğu,
- Kahl vd. (2004) tarafından öznelik vektörü bazlı Temel Bileşen Analizi (PCA) kullanarak bir arka plan modeli oluşturulduğu,

sonuçlarına ulaşılmıştır. Yazarlar tarafından yapılan çalışmada ise; arka plan resmine ait piksellerin değerlerini uzun vadede değerini koruyacağı prensibine göre her piksel için arka arkaya gelen 100 resim çerçevesi boyunca bir zamansal bir eğitim dizisi oluşturulmuştur. Bu dizideki üst üste olmayan kararlı aralıklar belirlenmiş ve bu aralıkların medyanları alınarak arka plan görüntüsü keskinleştirilmeye çalışılmıştır. Sonuç olarak oluşturulan sistemin gerçek ortamdaki deneme çalışmalarında algoritmanın kararlı ve adaptif olduğu tespit edilmiştir.

Rui vd. (2008) adaptif renk eşleştirme ve Kalman filtrelerini kullanarak mobil robotlar ile bir imge tanıma ve takip etme algoritması ortaya koymuşlardır. Yaptıkları çalışmada; adaptif renk eşleştirme metodunu hareketli imgenin etrafındaki alanı daraltmakta ve imge ve çevresi üzerindeki ışık ve parlaklıkların azaltılması için kullanmışlardır. Bu nedenle ışık değişimlerine karşı daha kararlı olan YcrCb renk uzayında çalışma yapmışlardır. Kalman filtresini ise robotun hareket yörüngesi üzerinde hareket eden imgenin hareket vektörünü tahmin etmek üzere kullanmışlardır. İki ayrı modülün bulunduğu algorithmada tahmin modülü hareket eden imgenin hareket vektörünü tahmin ederken, güncelleme modülü de ışık değişimlerinin etkilerini azaltmak üzere renk eşleşmelerinin eşik değerlerini adaptif olarak düzenlemektedir. Yapılan deneysel çalışmalarda kurulan sistemin ışık değişimlerinin etkilerine karşı kararlı ve imge tanıma ve takibinde ise iyi performans gösterdiği sonucuna varılmıştır.

Wang vd. (2011) birden fazla parçadan oluşmuşçasına tanımlanabilecek imgelerin tanımlanabilmesi için bir yöntem ortaya koymuşlardır. Bu yöntem imge parçalarının hareket gruplamalarının sonuçlarını birleştirecek şekilde Örtülü Şekil Modelinin (Implicit Shape Model) genişletilmesi ile oluşturulmuştur. İmge parçaları başlangıçta hareket yönlerindeki benzerlik ve eşleşmelere göre gruplanmış sonrasında gruplanan sonuçlar parça deformasyonlarına karşı kararlı sonuçlar verebilen Hough Transform yapısında birleştirilmiştir. Bu yapıda imge parçalarının imge merkezi ve imge etiketine göre ağırlıkları hesaplanmaktadır. Ağırlık atama işlemlerinde daha önceden oluşturulmuş olan ve her bir kodun görünümü, sınıf etiketini ve imge parçasının imge merkezine uzaklığını belirttiği bir dizi imge setinden oluşan bir kod çizelgesi kullanılmaktadır. Yapılan deneysel çalışmada içinde yayalar ve bisikletlilerin bulunduğu 401 çerçeveden oluşan 720x576 çözünürlükteki görüntüler kullanılmış ve deneyler sonucunda yüksek oranda yayalar ve hareket yörüngeleri doğru olarak tespit edilmiş az miktarda bisikletli ise hatalı tespit edilmiştir.

Mejia ve Kang (2011) hareket ve renk bilgilerini bir arada kullanarak hibrit bir hareketli imge tanıma sistemi ortaya koymuşlardır. Bu sistemde; yüksek çözünürlüklü video akışı içinde hareketli imgenin hareket-renk öznitelikleri belirlenmiş ve imge sınırlarının kesinliği istatistiksel optimizasyon yöntemleri ile arttırılmıştır. İmge sınırlarının kesinleştirilmesi için yapılan işlem, H.264 kodlayıcısından elde edilen hareket vektörünü ve hareket eden imge gruplarının hareket-kenar haritalarını içermektedir. Söz konusu sistem vasıtasıyla elde edilen renk maskesi ve hareket maskesi birleştirilerek RGB değerlerine göre Gauss dağılımları alınarak hareket tahminleri yapılmaktadır. Bu çalışma sonucunda arka plan görüntüsüne bağlı kalınmadan ve nispeten az sayıda görüntü çerçevesi kullanılarak ve belirli objelere bağlı kalınmadan kararlı sonuçlar alınmıştır.

Murshed vd. (2011) tarafından yapılan bir çalışmada; Canny yöntemi ile kenarları çıkartılmış bir görüntü üzerinde kenar bölütleme yapılarak hareketli bir görünürünün tespit edilmesi sağlanmıştır. Bu çalışmada; öncelikle video görüntüleri çerçevelerine ayrılarak tekil görüntüler elde edilmiş, sonrasında Canny yöntemi ile kenarları çıkartılan görüntüler bölütlenmiş ve sonrasında bu bölütler İteratif K-Ortalamalar Kümesi (K-means Clustering) yöntemi yardımıyla gruplanmıştır. Sonrasında bu grup, kısmi bilgi toplama işlemlerinde kararlı olabilmesi amacıyla genelleştirilmiş Hough Transform yöntemi bazı şekil eşleştirme algoritmasında kullanılmıştır. Son olarak bir sonraki çerçevelerde

grupların hangi konumda olabileceklerini tahmin etmek maksadıyla Kalman Filtresi kullanılmıştır. Yapılan deneysel çalışmalarda, önerilen yöntemin bilinen bir yöntem olan ortalama kayma yöntemine göre iç sahalarda %27.27 dış sahada ise %21.33 daha az piksel hatası ile çalıştığı görülmüştür.

Wöhler ve Anlauf (2001) Adaptif zaman gecikmeli yapay sinir ağı (ATDNN) algoritması ile bir imge dizisi içinde zamana ve mekana bağlı olarak nesnelere şekilsel ve hareket özelliklerini baz alarak tanınmasını sağlayan bir uygulama ortaya koymuşlardır. Bu çalışmada ilk olarak geçiş yapan araçların bölütleme olmaksızın tanınması gerçekleştirilmiş ve %97 başarı oranı yakalanmıştır. Hareket halindeki yayaların daha yüksek kararlılıkta tanınabilmesi için ise polinom destek vektör makinesi ile ATDNN yaklaşımını birlikte kullanarak yüksek kararlılıkta sonuçlar elde etmişlerdir. Çalışmada başarı oranını arttırmak için çok sayıda imge ile yapay adaptif zaman gecikmeli yapay sinir ağı eğitilmiştir. Yayaları tanıyabilmesi için sistem 3926 adet yaya olan ve 4426 adet yaya olmayan imge ile eğitilmiştir. Geçiş yapan araçların tanınabilmesi için ise 886 adet araç olan ve 3073 adet araç olmayan imge ile sistem eğitilmiştir. Eğitim sırasında sisli ve yağmurlu havalarda elde edilmiş imgeler de dikkate alınmıştır. ATDNN algoritmasının geri besleme döngüleri dikkate alınmadan yapılan denemelerde %79.0 oranında yaya tanınabiliyorken geri besleme döngüleri dikkate alınarak yapılan çalışmalarda %88.3 oranında başarı elde edilmiştir.

Wahi vd.(2010) tarafından yapılan çalışmada Ayrık Dalgacık Dönüşümü (Discrete Wavelet Transform- DWT) metodu ile 5, 10, 15, 20, 25 derecelerde döndürülmüş görüntüler ile geri yayımlı Widrow-Hoff öğretim kurallı çok katmanlı ağ ile yapılan tanıma çalışmasında 5 derecede %95.9, 25 derecede %56.67 başarı elde edilmiştir.

Cao vd. (2006) tarafından yapılan çalışmada desen tanıma için öncelik sıralı yön esas fonksiyonlu sinir ağı (the priorityorder direction basis function neural network for pattern recognition PODBNN) ile yapılan çalışmada %98.5 doğruluk oranı elde edilmiştir

Patil vd.(2004) tarafından yapılan çalışmada 3 katmanlı ileri beslemeli fourier fuzzy sinir ağı(feed forward Fourier fuzzy neural network, FFNN) ile yapılan tanıma çalışmasında %83 başarılı tanıma gerçekleştirilmiştir.

Juang ve Chen (2008) tarafından yapılan çalışmada; hareketli nesnelere yaya, otomobil, motosiklet, köpekler olarak sınıflandırılmıştır. Her nesne bir dış şekilsel kontur olarak

oluşturulmuş ve nesne üzerindeki her bir noktanın kontur merkezine uzaklığı hesaplanmış ve yumuşatılmıştır. Bu şekilde elde edilen öznitelik vektörüne nesnenin dışsal şekline ait konturun uzunluk-genişlik değeri de ilave edilir. Öznitelik vektörünün son hali baz alınarak nesne tanıma işlemi için Kendi kendine Oluşturan Sinirsel Bulanık Çıkarım Ağı (Self Constructing Neural Fuzzy Inference Network, SONFIN) ve Geri Yayınımlı Yapay Sinir Ağı (Back Propagation Neural Network - BPNN) yöntemleri kullanılmıştır. Yazarların çalışmasında; performansı ölçmek amacıyla iki çeşit deneme yapılmış denemelerden birinde nesnenin dışsal şekli manuel olarak ayrılmış diğerinde ise görüntü işleme süreçlerinden otomatik olarak ayrılmıştır. Manuel ayırma şeklinde yapılan çalışmada her bir sınıftan 100'er örnek ile eğitilen sistem SONFIN metodu kullanılarak %99.75 doğrulukta bir başarı elde edilmiş aynı örnekler ile BPNN metodu ile yapılan denemelerde %98.25 doğrulukta sonuç elde edilmiştir. Nesnelerin otomatik olarak arkaplan çıkarımı yöntemiyle ayrıldığı denemelerde ise; SONFIN metodu ile %93 doğrulukta, BPNN metodu ile %87.5 doğrulukta sonuçlar elde edilmiştir.

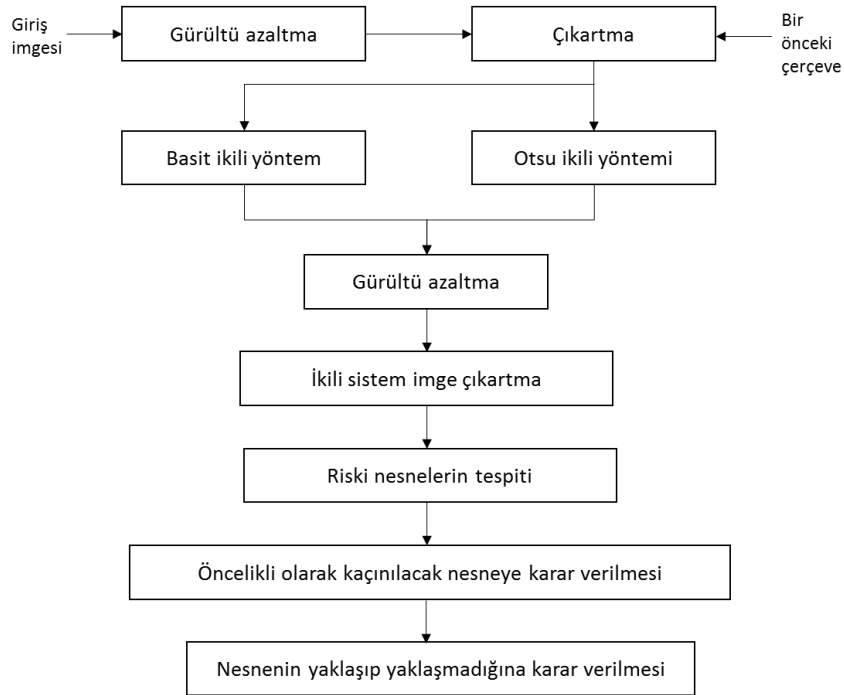
Razali ve Adznan (2006) tarafından sunulan hareketli insan ve taşıt algılama çalışmasında; üç modülden oluşan bir hareketli nesne tespit ve sınıflandırma sistemi sunulmuştur. Bu modüller;

- Adaptif arka plan çıkarma yöntemi ile hareketli aday bölgelerin tespit modülü,
- Değişmez moment (invariant moment) yöntemiyle nesne özniteliklerinin çıkartılması,
- İleri beslemeli yapay sinir ağları ile sınıflandırma öncesi Bağlantılı Bileşen Etiketleme (Connected Component Labeling -CCL) yöntemi ile nesne etiketleme adımlarıdır.

Arka plan çıkarımı işlemlerinde arka plan tahmini, ön plan tespiti işlemlerinde ayırım işlemi için her imge çerçevesinde arka plan ve ön plan için bir ağırlıklandırma yöntemi kullanılmıştır. Bu işlemden sonra elde edilen görüntüler gölge, parlama ve gürültülere karşı medyan filtreleme ile düzenlenmiştir. Bu şekilde elde edilen insan ve taşıt görüntülerinin momentlerinden çıkartılan öznitelik vektörleri ile ileri beslemeli geri yayınımlı yapay sinir ağları eğitilerek sonuçlar elde edilmiştir. Bu yöntem ile sadece hareketli insan ve taşıt tespiti değil farklı nesnelerin de tespit edilebileceği, elde edilen sonuçların kararlı ve hassas sonuçlar alındığı yazarlar tarafından ifade edilmiştir.



Moriwaki (2011) tarafından geliştirilen engelli insanların taşınabileceği mobil otonom robot sisteminde; ikili (binary) sistem ve Otsu (1979) yöntemleriyle eşiklenmiş görüntülerin arka plan çıkarma işlemleri neticesinde hareketli (yaya, bisikletli) ya da sabit engellerden kaçınma hareketleri modellenmiştir. Sabit ve hareketli engel tespiti ve tehlikeye yol açıp açmayacakları Şekil 2.6'daki akış şemasında açıklanan algoritma ile gerçekleştirilmiştir. Bu yöntem ile yaklaşan ve engel teşkil edebilecek nesnelere uzaklaşan nesne ayrımı %90 başarı ile yapılmışken, tüm yönlerden farklı rotalarda hareket eden nesnelere tespiti %70 başarı ile sağlanabilmiştir.



Şekil 2.8 Hareketli nesnelerin tanınma algoritması (Moriwaki 2011).

Do ve Huang (2014) yaptıkları çalışmada; statik arka planlar yerine dinamik arka planlarda gerçekleşen nesne hareketlerini tespiti yönelik yapay sinir ağları tabanlı bir yöntem geliştirmişlerdir. Bu yöntem ile çevresel değişimlerden bağımsız olarak hassas bir şekilde nesne hareketlerinin tespit edilebildiği ifade edilmektedir. Bu yöntem iki adımdan oluşmuş olup birinci adım; çoklu arka plan oluşturma, ikinci adım ise hareketli nesne tespitidir. Birinci adımda imge dizisinin her çerçevesinden gelen piksellerin seçilen referans arka plan aday piksellerine olan Euclidean uzaklıkları hesaplanarak elde edilen değerler Radyal Temelli Fonksiyon (Radial Basis Function - RBF) yapay sinir ağının gizli katmanına gönderilir. Ağın giriş katmanı ise her imge çerçevesine ait her bir pikselin ton,

saturasyon ve gri seviye değerlerinden oluşmaktadır. Bu şekilde öğreticisiz öğrenme ağı olan radyal temelli fonksiyon yapay sinir ağını kullanarak her imge çerçevesinin arka planındaki değişim algılanmaya çalışılır. Eğer değişim oluşursa yeni çerçevenin piksel değerleri yeni arka plan referans değerleri olarak seçilir. Yöntemin son adımı olarak oluşturulan RBF ağının çıkış katmanı algılama sonucu olarak binary görüntüde hareket algılama maskesinin elde edilmesini sağlar. Yapılan deneysel çalışmalar ile söz konusu yöntem diğer gelişmiş yöntemler ile niteliksel ve niceliksel olarak karşılaştırılmış ve Benzerlik ve F1 doğruluk oranları sırasıyla %82.08 ve %86.75 kadar diğer metodlardan daha üstün olduğu görülmüştür.

Zha vd. (2009) tarafından çok unsurlu ve sabit olmayan bir arka plan üzerinde hareketli nesnelerin kesin olarak tespit edilmesine yönelik bir çalışma sunulmuştur. Bu çalışmadaki odak nokta imgenin arka planı üzerinde rüzgârda salınan ağaç dalları ve sürekli oluşagelen deniz dalgaları gibi aslında arka planın bir parçası olan ancak hareketli nesne olarak algılanabilecek unsurların absorbe edilmesidir. Bu çalışmada imge ön planının tespit edilebilmesi için imge arka planını öğrenen çok ölçekli ayırt edici bir yöntem geliştirilmiştir. Bu yöntemde öncelikle Gauss Karışım Modeli (Gaussian Mixture Model - GMM) kullanılarak statik arka plan elde edilmeye çalışılmış sonrasında ise Markov Rastgele Alan (Markov Random Fields - MRF) Modeli kullanılarak imge arka plan/ön plan bölütlemesi yapılmıştır. 400 imge çerçevesi ile yapılan deneysel çalışmada önerilen yöntemde hata oranı 0.05-0.17 arasında değişirken, diğer yöntemlerde ise 0.2-0.46 ve 0.17-0.42 arasında değiştiği ifade edilmiştir.

Yu vd. (2012) tarafından değişken arka planda imge hareketlerinin tespiti için yapılan çalışmada; ardışık ikili gruplar halinde sisteme dâhil edilen imge çerçeveleri üzerinde Ölçekten Bağımsız Öznitelik Dönüşümü (Scale-Invariant Feature Transform - SIFT) algoritması kullanılarak tespit edilen kilit noktalar ele alınmıştır. Ardışık iki imge çerçevesi üzerindeki kilit noktalar arasındaki fark En yakın Komşu Uzaklık Oranı (Nearest Neighbor Distance Ratio - NNDR) kuralına göre hesaplanarak her iki imge çerçevesi arasında bir yer değiştirme vektörü oluşturulmuştur. Art arda pek çok imge çerçevesinden oluşan bir video görüntüsünde arka plan ve hareketli nesne bölütlerinin farklı hareket karakteristikleri olacağı düşüncesinden hareketle bu bölütlerin farklı karakteristikte yer değiştirme vektörleri oluşacağı ifade edilmiştir. Farklı karakterdeki vektörlerin sınıflandırılmasında ise Fuzzy C-Means Kümeleme algoritması

kullanılmıştır. Bu algoritmaya göre yapılan sınıflandırmada hareketli nesnelere ile arka plana ait kilit noktalarının kararlı bir biçimde ayrıştırılabildiği ifade edilmiştir.

Jarraya vd. (2010) yaptıkları çalışmada imge ön planının imge çerçevelerinden ayıklayabilmek için yeni bir arka plan modeli yaklaşımı geliştirmişlerdir. Bu yaklaşımda; ani ışık değişiklikleri, kamera titremesi, arka plan değişimi gibi gürültülerin modelin başarısına etkisinin azaltılması dikkate alınmıştır. Bu yaklaşımda öncelikle 2. ve 3. görüntü çerçevesinin 1. çerçeveden çıkartılması suretiyle RGB özelliklerinde model başlangıç koşullarına ayarlama işlemi yapılmaktadır. Bu işlemin ardından ardışık tüm çerçeveler için model güncelleme adımı icra edilmektedir. Bunun amacı otomatik eşikleme fonksiyonu kullanılarak piksellerin arka plan veya ön plan olarak etiketlenebilmesidir. Arka planda önemli miktarda değişikliğe uğrayan pikseller zamansal ayrıştırma ile ön plan olarak ayrılırlar. Bu işlemden sonra kullanılan bir imge ön planı maskesiyle hareketli pikseller bölgeler halinde bölütlenir ve bu şekilde hareketli nesnelere tespit edilmiş olur. Ani ışık değişimleri, kademeli ışık değişimleri, kamera titreşimleri ve arka plan bileşen değişimleri gibi gürültüler bulunan video akışlarında yapılan deneysel çalışmalarda, farklı 4 yöntem (1-IFD tabanlı DSTEI, 2- Gauss Karışım Modeli ile öz yinelemeli arka plan modeli, 3-KDE ile öz yinelemesiz arka plan modeli ve 4-hibrit yöntemi) ile önerilen yöntem karşılaştırılmış ve 0.7711 ile 1 arasında elde edilen kesinlik değerleriyle en doğru sonuçların elde edildiği yöntem olmuştur.

Ren vd. (2003) tarafından yapılan sabit olmayan arka plan üzerinde istatistiksel arka plan çıkarma işleminde her bir piksel zamansal ve uzamsal olarak modellenmiş ve her biri ön plan ya da arka plan olarak işaretlenmiştir. Çalışmada; istatistiksel arka plan modellemesinde ışık değişimleri, gölgeler ve diğer etkilere karşı daha kararlı sonuçlar alınacağı ve bu yaklaşımın diğer yaklaşımlara göre hareketli nesnelere tanımlanabilmesi için daha kararlı özellik setleri ürettiği bildirilmiştir. Önerilen yaklaşımda Konumsal Dağılımlı Gauss (Spatial Distribution of Gaussian - SDG) modeli kullanılarak imge arka planı modellenmiştir. Yazarlar tarafından yapılan deneysel çalışmalarda yanlış belirlenen imge alanının tüm imge alanına oranından oluşan hata oranında geleneksel yöntemlerle aynı sonuçlar alınsa da tespit edilemeyen ön plan alanının hedef alanına oranından oluşan hata türünde geleneksel yöntemlere göre çok daha başarılı sonuçlar elde edildiği ifade edilmiştir.

Spagnolo vd. (2006) tarafından yapılan çalışmada bir video akışında hareket eden nesnelerin tespit edilmesinde zamansal analiz kullanılmıştır. Çalışmadaki amaç hareketli nesnenin tespiti sırasında; ani ışık değişimlerinden en az şekilde etkilenen bir sistem oluşturmaktır. İmgenin bölütleme işlemi arka plan çıkarma işlemi ve zamansal analiz yöntemi kullanılarak yapılmaktadır. Çalışmada hareketli noktaların tespiti için ardışık imge çerçevelerinin uyuşan bölgeleri arasındaki radyometrik benzerlik kullanılmış, ayrıca ön plan nesnelerinin bölütlenmesi için de mevcut imge ile referans arka plan arasındaki radyometrik benzerlik fonksiyonları kullanılmıştır. Ancak arka plan güncelleme sürecinde sadece arka plana ait pikseller değil ön plandaki bölgelere ait pikseller de güncellenir. Yazarlar tarafından bu çalışmadaki başlıca yenilik olarak; arka plan modelindeki her bir pikselin yoğunluk değişimlerinin tespit edilmesinde imgedeki tüm pikseller tarafından sergilenen aynı miktardaki yoğunluk değişimlerinin dikkate alınması gösterilmektedir. Bu şekilde imge üzerindeki yavaş hareketler dikkate alınmamaktadır. Farklı mekanlarda yapılan deneysel çalışmalarda önerilen yöntemin doğru tespit oranı %93.78 ile %97.11 arasında değişirken, hatalı tespit oranı %2.11 ile %11.69 arasında değişiklik göstermiştir.

Pan ve Xia (2008) tarafından yapılan çalışmada sınırlarla temsil (boundary representation) ve dalgacık yapay sinir ağı (wavelet neural network, WNN) kullanılarak etkili bir nesne tanıma metodu geliştirilmiştir. Nesnenin şeklini ifade etmek için normalize edilmiş eğrilik fonksiyonları imgeye uygulanmıştır. Nesne sınırlarının eğrisel ifadesine CWT analizi uygulayarak sınırların eşsiz noktalarının yerel zaman frekans özellikleri elde edilmiştir. Bu şekilde; dalgacık yapay sinir ağı, dalgacık dönüşümlü yapay sinir ağı (wavelet transform artificial neural network, WT-ANN), en yakın K komşu (K-nearest-neighbor, k-NN), şekil bağlamı (shape context, SC-KNN) ve eğrilik derecesi uzaysal temsili (curvature scale space representation, CSS-KNN), destek vektör makinesi (the support vector machine, SVM), geleneksel fourier tanımlamaları (the traditional fourier descriptors, FD-SVM) ve en büyük dalgacık modüllü yapay sinir ağı (the wavelet modulus maxima artificial neural network, WMM-ANN) ile yapılan tanıma çalışmalarında %82.57 - %99.28 oranında doğru tanıma sonuçlarına ulaşılmıştır.

Kim vd. (2013) tarafından hareketli bir kameradan elde edilen görüntüler üzerinde nesne tanıma ve takibi çalışması yapılmıştır. Bu çalışmada karmaşık ve geniş arka plan modellemeleri yerine daha küçük ölçekli bir arka plan modellemesi yapılmıştır.

Modelleme işleminde görüntü eşleme hatalarından ve arka plan adaptasyon sorunlarından kaynaklanan hataları çözebilen uzamsal-zamansal dağılımlı Gauss modeli kullanılmıştır. Kamera hareketini kestirebilmek için Lucas Kanade İzleme (Lucas Kanade Tracking - LKT) yöntemi kullanılmış ve bu yöntemle kamera hareketi belirlendikten sonra hareketten önceki ve sonraki görüntü çerçeveleri Gaussian Farkı (Difference of Gaussian, DoG) filtresi ile keskinleştirilerek nesnelere daha belirgin hale getirilmiştir. Hesaplanan kamera hareketinin, filtrelenen arka plan modelini filtrelenen hareket sonrası çerçeveye doğru geometrik olarak eğmesiyle iki alanlı yeni bir arka plan modeli oluşturulmuştur. Bu yeni modelin öz niteliklerinin elde edilmesi ve bu işlemin zincirleme bir şekilde icra edilmesiyle hareketli nesnelere tespit edilmektedir. Yapılan çalışma insansız hava araçları (Unmanned Aerial Vehicle - UAV), Yatay ve dikey hareketli yakınlaştırmalı kamera (Pan Tilt Zoom - PTZ) ve el kameraları ile test edilmiş ve 0.916 dan 0.705 'e kadar değişen hassasiyetlerde çalıştığı belirlenmiştir.

Sappa vd. (2008) hareketli bir stereo kameradan alınan görüntülerde öz nitelik nokta kümelerinin çıkarılarak ardışık görüntü çerçevelerinde çakıştırılması prensibine dayanan bir çalışma gerçekleştirmişlerdir. Bu çalışmada; kullanılan stereo kameralar sayesinde 2 boyutlu görüntüler 3 boyutlu görüntülere çevrilmiş ve görüntülerin Haris Köşe Bulma yöntemiyle öz nitelik noktaları oluşturulmuştur. Sonrasında Quaternion yöntemi sayesinde bilinen eşleştirmeler kullanılarak iki adet 3 boyutlu nokta kümesinin çakıştırılması RANSAC (Random Sample Consensus) tabanlı bir yaklaşım ile gerçekleştirilmiştir. Sonuç olarak hesaplanan 3 boyutlu uzaklıklar iki ardışık görüntü çerçevesinin aynı koordinat sisteminde eşleştirilirmesi için kullanılmış ve en yüksek hataya sahip olan alanlar hareketli nesnelere tespit edilmiştir. Yapılan deneysel çalışmalarda ardışık iki görüntü üzerinde yaklaşık 300 adet öz nitelik noktası kullanılarak yapılan eşleştirme işlemi ortalama 31 ms olarak gerçekleştiği yazarlar tarafından tespit edilmiştir.

Lin ve Wolf (2010) kamera hareketini modelleyerek bir sonraki görüntü çerçevesinin durumunu tahmin etmeye ve tahmin edilen görüntü çerçevesiyle gerçek çerçevenin farkını alarak bir arka plan ön plan bölütlemesi gerçekleştirmeye dayalı bir çalışma yapmışlardır. Kamera hareketinin tahmin modelini oluşturmak amacıyla incelenecek iki görüntü çerçevesinde SURF algoritması kullanılarak kesin öz nitelik noktaları çıkartılmış ve bir hareket vektörü oluşturulmuştur. Kamera hareketinin modellenmesinde 3 boyutlu

dönme ve doğrusal hareketlerin modellenmesindeki zorluklar nedeniyle Genetik Algoritma yöntemi kullanılmıştır. Tahminsel kamera hareketi kullanılarak olması gereken görüntü çerçevesi ile mevcut görüntü çerçevesi karşılaştırılarak en yüksek hatanın olduğu bölümler hareketli nesnelere etiketlenmiştir. 2 ayrı video ile yapılan deneysel çalışmalar neticesinde hareketli nesnelere kesin bir şekilde belirlendiği yazarlar tarafından ifade edilmiştir.

Jung ve Sukhatme (2004) hareketli bir platform üzerinde bulunan bir kameradan elde edilen görüntülerdeki hareketli nesnelere tespit etmeye yönelik bir çalışma yapmışlardır. Bu çalışmada öncelikle kameranın öteleme, dönme ve ölçekleme şeklinde kendini gösteren hareketlerinin negatif etkisini telafi etme yoluna gitmişlerdir. Bunun için ardışık görüntü çerçevelerinden çıkartılan öz nitelik noktaları Lucas-Kanade optik akış tahmin yöntemiyle eşleştirilmiş ve buradan hareketle bilinear dönüşüm modeli uygulanarak kamera hareketleri tahmin edilmiştir. İki görüntü çerçevesi arasındaki uzaysal fark yüksek olduğu için doğrusal affine dönüşümü tercih edilmemiştir. Kamera hareketinin istenmeyen etkileri görüntü üzerinden kaldırıldıktan sonra elde edilen yeni çerçevenin bir önceki çerçeve ile farkı alınarak hareketli nesnenin uzaysal pozisyonu ve büyüklüğü tahmin edilmiştir. Tahmin işlemi için Bayes yöntemi ile nesnenin olasılık dağılımları bulunmuş ve adaptif partikül filtre algoritması ile sonuç elde edilmiştir. Yapılan çalışma robotik helikopter, iki tekerlekli ve dört tekerlekli kara araçlarında test edilmiş ve sırasıyla %82.86, %93.63 ve %66.28 nesne tespit oranları elde edilmiştir.

Kim vd. (2010); serbest hareketli kameralardan elde edilen görüntülerde hareket eden nesnelere tespit etmek amacıyla hareket kümeleme ve hareket dengeleme adımlarını içeren bir çalışma yapmışlardır. İlk aşamada; görüntü çerçeveleri uzaysal koordinat sisteminden optik akış haritasına (Optical Flow Map -OFM) aktarılmıştır. Bu koordinat sisteminde görüntü  $M \times N$  adet eşit karesel bloklara bölünmüş ve çerçevedeki öz nitelik noktaları blok numaralarına göre ayrıştırılmıştır. Bu zeminde Öklid uzaklığına dayalı basit kümeleme teknikleri kullanılarak optik akış ve çakışmayan öz nitelik noktalarının elenmesi bazı hareket kümeleme gerçekleştirilmiştir. Her bir kümede öz nitelik noktalarından çakışmayanların elenmesi ve çakışan noktaların seçilmesi işlemlerinde RANSAC yöntemi kullanılmış ve neticesinde elde edilen yeni kümeler ön plan ve arka plan sınıflandırması için kullanılmıştır. Sınıflandırma işlemleri için yeni kümeler tekrar uzaysal koordinat sistemine taşınmış ve yöntem gereği elde edilen kovaryans matrislerin

özdeğer vektörleri bulunarak dağınıklık hesabı yapılmıştır. Bu şekilde yüksek dağınıklık gösteren alanlar arka plan, daha az dağınıklık gösteren alanlar ise ön plan olarak sınıflandırılmıştır. Yapılan deneysel çalışmalarda özdeğer analizi ve RANSAC yöntemi kullanıldığında doğruluk oranı 0.86 çıkarken sadece kümelerdeki öz niteliklerin sayılması şeklinde yapılan denemelerde doğruluk oranı 0.72 olarak gerçekleşmiştir.

Kim ve Do (2012) görüntü çerçevelerini eşit büyüklükteki bloklara bölerek ardışık çerçevelerde blokların eşleştirilmesine dayalı Blok Tabanlı Hareket Tahmini (Block Based Moving Estimation - BBME) yöntemini kullanarak hareketli bir kameradan elde edilen görüntülerde hareketli nesnelerin tespiti çalışması gerçekleştirmiştir. Bu çalışmada 21x21 piksel büyüklüğündeki bloklara ayrılan ardışık çerçevelerde Mutlak Farkların Toplamı (Sum of Absolute Differences - SAD) ölçütü ile iki blok arasındaki eşleştirme yapılarak yatay hareket vektörü elde edilmiştir. Bu vektörler kullanılarak oluşturulan görüntü çerçeveleri arası mesafe haritasında en yüksek değerli bloklarda hareket eden nesnelerin bulunduğu varsayılarak hareketli nesneler tespit edilmiştir.

Zhang vd. (2006) yaptıkları çalışmada hareketli bir araç üzerinde monteli bir kameradan alınan görüntüdeki hareketli nesneleri uzak mesafelerden tespit edebilen kararlı bir algoritma sunmuşlardır. Sunulan algoritma ile bağımsız hareket eden nesneler tespit edilmeye çalışılmıştır. Bunun için öncelikle Haris Detektörü ile görüntü çerçevesindeki yüksek şiddetteki köşe ve kesişme noktaları belirlenmiş ve Lucas-Kanade yöntemiyle belirlenen noktaların optik akış vektörleri hesaplanmıştır. 3 boyutlu ortamda hareketsiz nesnelerin kamera hareketinden kaynaklanan optik akış hız vektörlerinin Yayılma Odağı (Focus of Expansion) olarak adlandırılan tek bir noktaya yakınsayacağı kuralına göre:

- a. Yayılma odağına yakınsayan tüm nesneler bulunmuş,
- b. Bulunan nesnelere oluşan piksel grubu anlık görüntü çerçevesinden çıkartılarak arda kalan nesneler hareketli nesneler olarak tanımlanmıştır.

Bu nesne grubunu içeren çerçeve üzerinde bölütleme yapılarak nesneler ayrıştırılmış ve etiketlenmiştir. Yapılan deneysel çalışmalarda 3000 görüntü çerçevesinden oluşan 30 video akışında en fazla 38 m. Uzaktan 12x29 piksel boyutlarındaki hareketli nesneler hatasız olarak belirlenebilmiştir.

Cheraghi ve Sheikh (2012) nesnelere ile arasında oldukça uzak mesafelerde bulunan ve doğası gereği hareketli bir platformdan görüntüleme yapan insansız hava araçları ile nesne tespiti yapabilen bir çalışma gerçekleştirmişlerdir. Bu çalışmada; öncelikle kamera hareketinin kestirilerek dengelenmesi amacıyla ardışık görüntü çerçeveleri üzerinde Shi & Tomasi yöntemiyle köşeler belirlenerek öznelikler çıkartılmış ve Farkların Kareleri Toplamı (Sum of Squared Differences - SSD) yöntemi ile eşleştirilmiştir. Eşleştirme neticesinde kamera hareketi kaynaklı yer değiştirme ilgin dönüşüm algoritması kullanılarak modellenmiştir. Dönüşüm modeline uyan arka plan ve bu modele uymayan hareketli nesnelere adaptif arka plan çıkarımı yöntemiyle bölütlenerek birbirinden ayrılmıştır. Yazarlar tarafından yapılan deneysel çalışmalarda %75 - %78 arası oranlarda doğruluk ifade eden sonuçlar elde edilmiştir.

#### **2.5.4 Nesne Tanıma İşleminde Yaşanabilecek Zorluklar**

Nesne tanımda başarıyı etkileyen pek çok etken bulunmaktadır. Başarı için öncelikle kaynak ve hedef imgelere ait özneliklerin doğru olarak çıkarılması gerekir. Ancak öznelikler çevresel koşullar sebebiyle doğru olarak tespit edilemeyebilir. Hedef imgelerde bulunan ve tanınmak istenen nesnelere

- Dönme, öteleme, sapma veya başka bir nesnenin arkasında kalma gibi sebeplerden dolayı ayırıcı özneliklerini kaybetmesi,
- Gürültü, aydınlatma sorunları ve yansıma gibi sebeplerden dolayı özneliklerin sayısal değerlerinin değişmesi,
- İnsan ve hayvan gibi esnek yapıda olan bir kısım nesnelere hedef imgelerde farklı formlarda bulunmaları nedeniyle bölgesel ve global özneliklerin kaynak imgedeki nesnelere göre farklılaşması

Nesne tanıma işlemlerinde yanlış tanıma ya da belirsizliğe yol açabilmektedir. Literatürde pek çok nesne tanıma teknikleri geliştirilmiş olmasına rağmen hiçbirinde kesin tanıma sağlanamamaktadır. Bir yöntem ile tamamen tanınabilen bir nesne, yukarıda sayılan olumsuzlukların herhangi birine maruz kaldığında tanınmasında sorun yaşanabilmektedir.



Tanım sorununun aşılabilmesi için farklı sınıflardaki nesnelere farklarını ortaya koyabilecek ayır edici özniteliklerin elde edilmesi gerekir. Nesne ile arka plan arasındaki renk zıtlığı yeterli olduğu sürece cisme ait alanlar kolaylıkla oluşturularak kullanılabilir (Ioannis 1993).

### 3. MATERYAL ve METOT

Bu tez çalışmasında dinamik nesnelerin tespiti için yazılım ve donanım tabanlı materyallerden faydalanılmıştır. Kullanılan kaynakların ayrıntılı açıklamaları bu bölümde verilmiştir. Geliştirilen yöntemde ise; öznitelik çıkartmak için SURF (Bay *et al.* 2006), FAST (Rosten and Drummond 2006), Harris (Harris and Stephens 1988) algoritmaları; öznitelik eşleşmelerini bulmak için Öklid Uzaklığı (ED), Toplam Mutlak Fark (SAD) ve Normalize Edilmiş Çapraz Korelasyon (NCC) yöntemleri; İmge çerçevelerinin dönüşümü için ise Geometrik Dönüşüm yöntemi kullanılmıştır. Nesne tanıma çalışmalarında yapay sinir ağı (YSA) tekniklerinden faydalanılmıştır. Kullanılan diğer yöntem ve teknikler ilgili bölümler altında açıklanmıştır. Kullanılan yazılımsal ve donanımsal materyaller;

- Matlab Yazılımı,
- Ardupilot Mega Kontrol Kartı,
- GPS Alıcı Sistemler,
- Mission Planner Yazılımı,
- Arduino Kontrol Kartı,
- DC Motor Sürücü Kartı,
- Mobil Araçlar ,
  - Görüntüleme Araçları,
  - Bilgisayar Donanımdır.

#### 3.1 Matlab Yazılımı

Matlab, çeşitli hesaplamalar ve matematiksel problemlerin çözümü için tasarlanmış bir yazılım geliştirme aracıdır. “MATrix LABoratory” kelimesinin kısaltması olan MATLAB, matrisler ve diziler ile çalışır. Matlab, Cleve Moler tarafından 1970’li yıllarda geliştirilmiştir.

Matlab yazılımı bünyesinde çeşitli mühendislik sorunlarının çözümü amacıyla geliştirilmiş olan görüntü işleme (image processing), yapay sinir ağları (artificial neural networks), sistem kimliklendirme (system identification), sayısal işaret işleme (signal

processing), optimizasyon (optimization), veri elde etme (data acquisition), veritabanı (database), süzgeç tasarımı (filter design), bulanık mantık (fuzzy logic) gibi araç kutusu olarak adlandırılan alt programları da barındırmaktadır.

Matlab yazılımında tüm imgeler birer matris olarak ifade edilebilmektedir. Bir imge Matlab yazılımına transfer edildiğinde imge üzerindeki her bir piksel matrisin bir elemanı olarak işlem görür. Bu elemanlar piksellerin taşıdıkları ışık şiddetini göstermektedirler. Bu piksellerin ifade ettikleri değerler ile yapılan filtreleme, iyileştirme, morfolojik işlemler vs. aslında matematiksel matris işlemleridir. Bir imgenin herhangi bir bölümü ifade edilmek istendiğinde imge matrisinin bir alt matrisi kullanılmaktadır. Denklem 3.1 'de  $i$  satır ve  $j$  sütundan oluşan bir  $f$  imgesinin Matlab yazılımındaki ifadesi verilmiştir.

$$I = \begin{bmatrix} f_{11} & \cdots & f_{1j} \\ \vdots & \ddots & \vdots \\ f_{i1} & \cdots & f_{ij} \end{bmatrix} \quad (3.1)$$

Matlab yazılımının temel özellikleri;

- Bir yazılım dilinin sağladığı söz dizimi (syntax), operatörler, veri tipleri, dizi indeksleme vb. özellikleri kullanıcılara sağlayabilmesi,
- Lineer cebir, basit istatistik hesapları, türev ve integral, Fourier ve Laplace dönüşümleri, filtreleme ve optimizasyon gibi matematiksel hesaplamaları gerçekleştirebilmesi,
- 2 veya 3 boyutlu çizimler, animasyonlar ve görselleştirme sağlayabilmesi,
- Programlama kod parçacıkları (script) ve fonksiyonların kullanılabilmesi, şartlı ve döngüsel ifadelerin kullanılabilmesi ve tüm kod yapısının düzenlenip debug edilebilmesi,
- Metin dosyaları, excel çalışma sayfaları, görüntü ve ses dosyaları, XML formatı ve bilimsel veri formatlarının içeri alma ve dışarı verme işlemlerinin TCP/IP iletişimi, FTP vb. web erişimleri ya da seri portlar üzerinden yapılabilmesi, Büyük verilere (big data) erişim ve işleme ve analiz adımlarının gerçekleştirilebilmesi,
- Uygulamalar için kullanıcı ara yüzlerinin oluşturulabilmesi,
- İleri yazılım geliştirme araçlarının bulunması, nesne yönelimli yazılımların geliştirilebilmesi, Java, .NET, C/C++ gibi yazılımlara dışsal ara yüzlerin kurulabilmesi,

- Geliştirilen yazılımların masa üstü uygulamaları olarak derlenebilmesi,
- Arduino, Raspberry Pi, webcam gibi 3. parti donanımların desteklenmesi, bu donanımlar ve seri port iletişimi sayesinde gerçek zamanlı olarak dış donanım (dc motor, servo motor vs.) kontrolü yapabilmesi,
- Modelleme ve simülasyon (benzetim) uygulamaları yapılabilmesi, örneğin bir robot kolunun modellenmesi yapılarak simule edilebilir
- Hazır kütüphaneleri ile kullanıcılara kolaylık sağlaması,
- Kısa ve kolay anlaşılır kod yapısı,
- Çeşitli işletim sistemi platformları ile uyum sağlayabilmesi,
- Zengin ve kolay ulaşılabilir bir dokümantasyonunun bulunması sayılabilir (İnt.Kyn.2).

MATLAB’de yazılan programlar, MATLAB’in kendine özgü dili kullanılarak yazılır ve MATLAB içinden çalıştırılır. Ayrıca yazılan programlar DLL ve EXE olarak derlenebildiği gibi C/C++ kodlarına da çevrilebilmektedir. İşlemler MATLAB’de komut satırından kod yazarak yapılabildiği gibi MATLAB GUI geliştirme aracını kullanarak, görsel öğeler içeren ara yüz tasarımları da yapılabilmektedir.

Bu çalışmada Matlab, 1920x4928 boyutundaki 6 adet görüntü matrisinin farklı matematiksel hesaplama, dönüştürme ve karşılaştırma işlemlerine tabi tutulup yaklaşık olarak 32 adet alt matris oluşturulmasıyla sonuçlanan bir dizi işlem için kullanılmıştır.

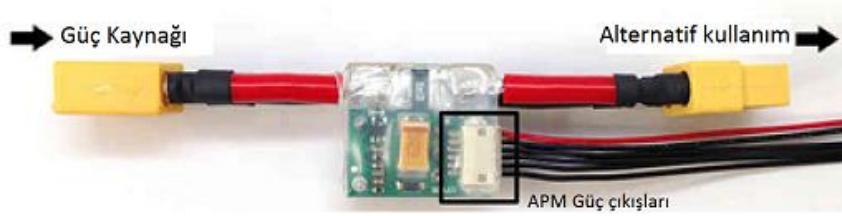
### **3.2 Ardupilot Mega Kontrol Kartı**

Ardupilot Mega (APM) profesyonel kalitede IMU (Inertial Measurement Unit) yani ivme ve dönüş miktarlarını ölçebilen teçhizat ile donatılmış Arduino Mega kontrol kartı tabanlı bir otopilot mekanizmasıdır. Bu otopilot sistemi, sabit kanatlı hava taşıtları (Arduplane), çok rotorlu ve geleneksel helikopterler(Ardicopter) ve İKA(Ardurover)’larını kontrol edebilecek yetenektedir. APM kontrol kartı otonom denge sağlayabilen, rota tanımlı navigasyon yapabilen ve kablosuz modülleri sayesinde çift yönlü uzaktan ölçme yapabilen yetenekte otopilot mekanizmaları ile donatılmıştır. APM cihazının görüntüsü Şekil 3.1’de verilmiştir.



Şekil 3.1 ArduPilot Mega kontrol kartı (İnt.Kyn.6).

Bu çalışmada APM 2.6 sürüm kontrol kartı kullanılmıştır. Kontrol kartı Arduover modda kullanılmış ve iki tekerlek üzerinden tank sürüşü yapılacak şekilde ayarlanmıştır. Kartın üzerinde GPS alıcı sistem, uzaktan ölçüm ve izleme sistemi (Telemetry) ve elektrik besleme girişleri için özelleştirilmiş konnektörler bulunmaktadır. Cihaz 5.37 V ile çalışmaktadır. Hassas bir şekilde beslemek için özelleştirilmiş güç modülleri regüle devreler kullanılmıştır. Bu devreler ile giriş en fazla 18 V gerilim uygulansa bile çıkışı 5.37 V 2.25 A olarak ayarlanabilmektedir. Kullanılan güç modülünün bir örneği Şekil 3.2’de sunulmuştur.



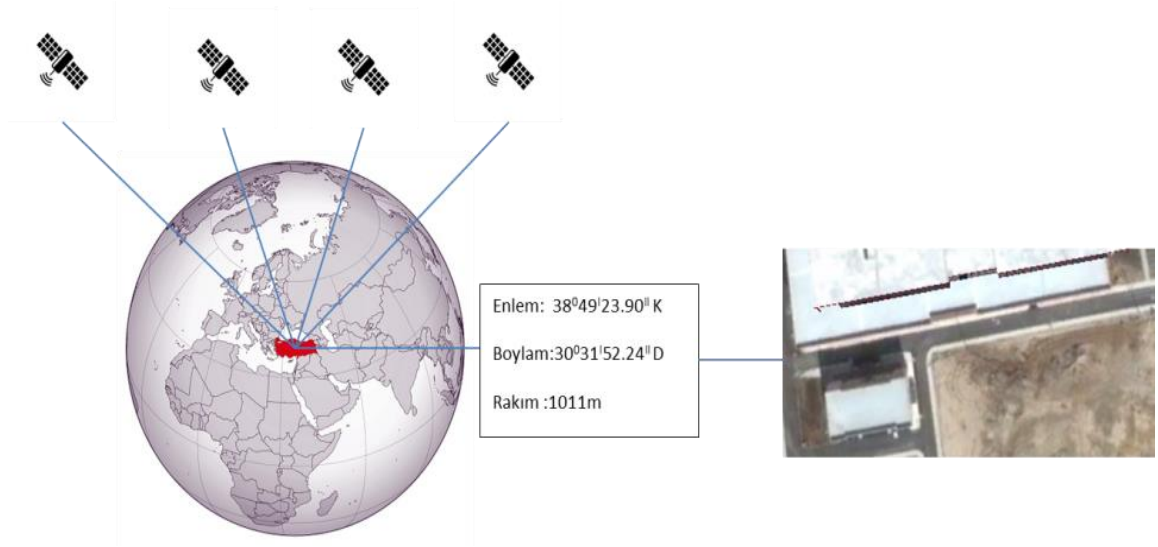
Şekil 3.2 APM kontrol kartı güç regüle modülü.

APM kontrol kartının ayrıca 8 adet giriş ve 8 adet çıkış konnektörü bulunur. Giriş uçlarına radyo kontrol cihazından gelen 5 adet kanal bağlanmıştır. Çıkış uçlarından ise İKA'nın DC motorlarının devir kontrolünde kullanılacak olan PWM sinyalleri alınmıştır.

### 3.3 GPS Alıcı Sistem

GPS, dünya üzerindeki herhangi bir coğrafi konumun enlem, boylam, rakım olmak üzere belirlenebilmesini sağlayan uydu tabanlı bir sistemdir. Dünya yörüngesinde konum bilgilerinin belirlenmesini sağlayan 24 adet uydu bulunmaktadır. İlk uydu 1978 son uydu ise 1994 yılında uzaya gönderilmiştir. Bu uydular yerden 20.180 km yüksekte, 6 farklı yörüngede ve her bir yörüngede 4 uydu bulunacak şekilde dönmektedirler. Her bir uydu

yörüngesini 12 saatte tamamlamakta olup bu şekilde yörüngesi üzerindeki her bir noktadan günde 2 defa geçmektedir. GPS dünya üzerinde sinyal alınabilen (yer altı, kapalı mekânlar vs. hariç) her yerde çalışabilmektedir. GPS temel işlevleri ile bir konumdan koordinat elde edilmesi işlemi Şekil 3.3’de ifade edilmiştir.



Şekil 3.3 GPS Temel İşlevi (Ünal 2012).

GPS sistemi ABD Savunma Bakanlığınca 1973 yılında geliştirilmiştir. Başlangıçta sistemin amacı; her türlü hava koşullarında 24 saat ABD ve müttefiklerinin askeri konum belirleme ihtiyaçlarının karşılanması olduğu için sivil kullanıcılar için sınırlı fırsatlar sunmaktaydı. Ayrıca atmosferik kırılma, yörünge doğruluğu, anten faz merkezi hataları, seçilen global referans sistemi, istasyonun konumuna ait özel hatalar nedeniyle günümüzde olduğu gibi sağlıklı olarak kullanılamamaktaydı. Ancak teknolojiye son gelişmeler nedeniyle GPS sisteminin performansını düşüren sınırlamaların azalması sivil uygulamalarda büyük bir artışa neden olmuştur.

Şu anda sistem dünyanın herhangi bir yerinde, asker ve sivil herkes için, sürekli ve ortak bir referans sisteminde; uyarlı, üç boyutlu konum, hız ve zaman bilgisi sağlamaktadır. Son yıllarda GPS kullanım oranı dünya ile orantılı olarak ülkemizde de artmıştır. Kişisel kullanımın yanı sıra ülkemizdeki haritalama çalışmaları için Tapu ve Kadastro Genel Müdürlüğü'nün yardımıyla Harita Genel Komutanlığı tarafından TUTGA (Türkiye Ulusal Temel GPS Ağı) kurulmuştur (Şanlıoğlu 2004).

Bu çalışmada GPS verilerini okumak için NMEA standart formatı kullanılmıştır. NMEA (National Marine Electronic Association) tüm GPS üreticileri tarafından desteklenen standart veri formatıdır. Donanım ve yazılım arasında en iyi uyumu yakalayabilmek adına geliştirilmiştir. Bu çalışmada kullanılan \$GPGGA ve \$GPGLL NMEA mesajlarının ayrıntılı açıklamaları Çizelge 3.1 ve Çizelge 3.2 'de verilmiştir.

- \$GPGGA: GPS Konum Mesajı

Anlık enlem ve boylam bilgisi ile beraber GPS cihazları için farklı bilgiler de içermektedir. Mesaj yapısı: \$GPGGA,m1,m2,c1,m3,c2,d1,d2,f1,f2,M,f3,M,f4,d3\*cc şeklindedir.

- \$GPGLL: GPS Enlem/Boylam Mesajı

Coğrafi enlem ve boylam bilgisini verir. İlave olarak zaman bilgisi de içerir. Sadece bu bilgiler gerektiğinde hızlı çözümler için kullanılabilir.

Mesaj yapısı: \$GPGLL,m1,c1,m2,c2,m3,c3\*cc

**Çizelge 3.1** \$GPGGA konum mesajı içeriği (İşeri 2006, İnt.Kyn.3).

Parametreler	Tanım	Aralık
M1	Anlık konumlamanın yapıldığı, saat dakika ve saniye cinsinden GPS zamanı (ssdddss,ss)	00-235959,50
M2	Derece ve ondalık dakika cinsinden enlem (dddd,mmmmmm)	0-90
C1	Enlem Yönü ( N : Kuzey, S: Güney)	N yada S
M3	Derece ve ondalık dakika cinsinden boylam (dddd,mmmmmm)	0-180
C2	Boylam Yönü ( W : Batı, E: Doğu)	W yada E
Di	Konumlama Türü;otomatik Konum RTCM diferansiyel düzeltilmiş konum	1,2
D2	Konum hesaplamada kullanılan GPS uydu sayısı	0-8
Fi	HDOP	0-99,9
F2	Referans elipsoidinden olan yükseklik	300000,00 +30000,00
M	Yükseklik Birimi ( M = metre)	M
F3	Geoit Yüksekliği ( metre)	999,99 - +999,99
M	Geoit Yükseklik Birimi M= metre )	M
D3	Diferansiyel düzeltme zamanı	0-999
D4	Baz istasyonu ID	0-1023
*cc	Kontrol Toplamı (checksum)	

**Çizelge 3.2** \$GPGLL enlem/boylam mesajı içeriği (İşeri 2006, İnt.Kyn.3).

Parametreler	Tanım	Aralık
m1	Derece ve ondalık dakika cinsinden enlem (dddd,mmmmmm)	0-90
c1	Enlem Yönü ( N : Kuzey, S: Güney)	N ya da S
m2	Derece ve ondalık dakika cinsinden boylam (dddd,mmmmmm)	0-180
c2	Boylam Yönü ( W : Batı, E: Doğu)	W ya da E
m3	Anlık konumlanmanın yapıldığı, saat dakika ve saniye cinsinden GPS zamanı (ssddss,ss)	00-235959,50
c3	Durum, A: Geçerli, V: Geçersiz	•ATV
*cc	Kontrol Toplamı (checksum)	

Bu çalışmada resmi Şekil 3.4 'de gösterilen MicroElectronika GPS Click model alıcı sistem kullanılmıştır. Cihaza üstündeki UART ve I2C arabirimleri ile bağlanılabilmektedir. Ayrıca USB arabirimiyle de PC bağlantısı yapılabilmektedir. Cihazda u-blox LEA-6S GPS /GALILEO modül bulunmaktadır. Cihaz 3.3V DC ile çalışmaktadır. Pasif ve aktif antenler ile kullanılabilmektedir.



**Şekil 3.4** Çalışmada kullanılan GPS alıcı (İnt.Kyn.4).

### 3.4 Mission Planner Yazılımı

Mission Planner yazılımı açık kaynak kodlu bir otopilot projesi olan Ardupilot projesi için hazırlanmış yetenekli bir yer istasyonu uygulamasıdır. Bilgisayarlara ücretsiz olarak yüklenerek arduplane, arducopter ve ardurover uygulamaları için yer istasyonu olarak kullanılmaktadır. Sadece Windows işletim sistemi ile uyumludur. Otonom araçlar için bir konfigürasyon aracı ya da dinamik kontrol aygıtı olarak da kullanılmaktadır.



Yazılım bilgisayar ortamına kurulduktan sonra APM teçhizatı ile USB ara yüzü üzerinden bağlantı kurmaktadır. Parametre değişiklikleri yazılım üzerinden yapılarak APM cihazına yüklenmektedir. Otonom araçlar ile ilgili bölümünde oluşturulan yer istasyonunun bir kısım resimleri Şekil 3.5’de verilmiştir.



Şekil 3.5 Mission planner ile oluşturulan yer istasyonu uygulaması resimleri(İnt.Kyn.7).

### 3.5 Arduino Uno Kontrol Kartı

Arduino elektronik projeler üretmek için kullanılan açık kaynak kodlu bir platformdur. Arduino donanımları programlanabilir bir işlemci içeren fiziksel bir donanım ile bilgisayarlara kurulabilen bir program parçacığından (IDE) oluşur. Bu program ile işlemciyi programlayacak kodlar yazılarak fiziksel donanıma aktarılabilir. Programlama dili basitleştirilmiş C++ yazılım dilidir. Arduino donanımları içerisinde bulunan pinler üzerinden aktifleştirilmiş bir sensor, basılan bir buton veya başka bir sinyal kaynağından gelen analog ya da dijital verileri okuyabilir ve bu verileri programlanan işlemcisindeki kodlara göre işleyerek bir motoru kumanda edebilir veya bir ledi yakabilir. Bu şekilde fiziksel aygıtlar ile sinyal alış-verişi yaparak kolayca etkileşime girebilmektedir. İşlemci olarak Atmel 8, 16, 32 bitlik işlemciler kullanılmıştır. Uno, Due, Mega ve Leonardo vs. modelleri bulunmaktadır. Her modelin işlemci hızları ve IO kapasiteleri ve bir kısım farklı özellikleri bulunmaktadır. Tüm donanımlar “shield” adı verilen donanım parçaları ile geliştirilebilecek modülerliktedir. Örneğin Arduino Uno modelinde normal olarak TCP/IP ünitesi bulunmamakta ancak TCP/IP shield’ı takıldığında Uno modeli bu özelliğe sahip olmaktadır.

Bu çalışmada Arduino Uno modeli kullanılmış olup APM kontrol cihazından gelen sinyallerin alınarak motorlara göndermeden önce filtrelenmesini ve daha esnek hale getirilmesi için kullanılmıştır. Arduino Uno kartının bir görünümü Şekil 3.6'da verilmiştir.



Şekil 3.6 Arduino Uno model kontrol kartı (İnt.Kyn.8).

### 3.6 DC Motor Sürücü Kartı

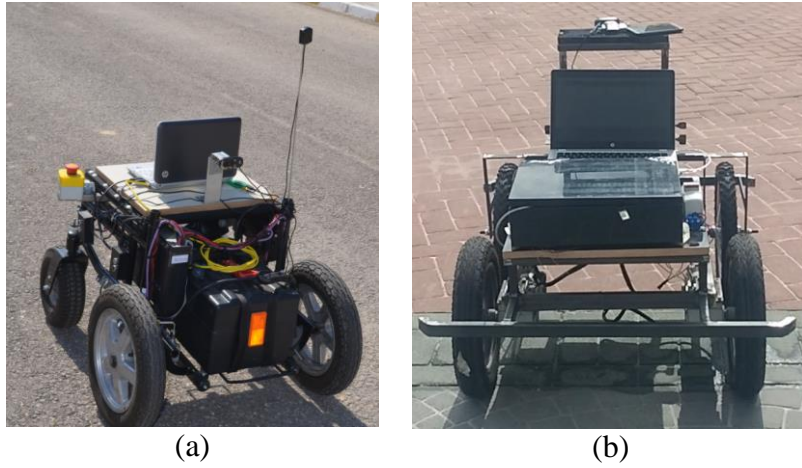
İKA'nın herbiri 420 W gücünde olan iki DC motorunu sürmek üzere BTS 7960 PN H köprüsü mosfetlerin kullanıldığı 2 adet sürücü kartı kullanılmıştır. Belirli pinlerine uygulanan PWM sinyali sayesinde dc motorlar üzerinde hız kontrolü rahatlıkla yapılabilmektedir. Motor kapasitesinin yüksek olması nedeniyle korumalı yüksek akımlı bir sürücü sistem seçilmiştir. BTS 7960 H köprüsü 43 ampere kadar akımlara dayanabilmektedir. Besleme gerilimi olarak 5,5 V ile 27,5 V aralığı kullanılabilir. Küçük yapısı nedeniyle devreler üzerinde fazla yer kaplamamaktadır. Sürücü kartının bir örneği Şekil 3.7'de verilmiştir.



Şekil 3.7 Dc motor sürücü kartı .

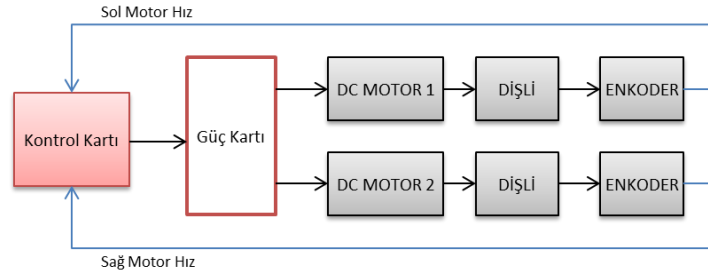
### 3.7 Mobil Araçlar

Bu çalışmada hareketli platform olarak kullanılan iki farklı tipte mobil araç kullanılmıştır. Bu araçlar 4 tekerlekli olup arka iki tekerlek 2 adet DC motor ve motorlara bağlı redüktörler tarafından hareketlendirilmektedir. Şekil 3.8 (a)'da gösterilen aracın ön tekerlekleri sarhoş teker yapısında olması nedeniyle engebeli arazilerde rotadan saparak deneysel çalışmalarda doğruluk tespit oranının düşmesine yol açmıştır. Bu nedenle Şekil 3.8 (b)'de gösterilen ön tekerlekleri sabit olan ikinci bir mobil araç imalatı yaptırılmış ve deneysel çalışmalara bu araçla devam edilmiştir.



Şekil 3.8 Çalışmada kullanılan (a) ilk mobil araç, (b) ikinci mobil araç.

Motor dişlilerinde bulunan enkoderler aracılığıyla alınan hız bilgileri kontrol kartı tarafından değerlendirilerek araç motorunun devir sayısı önceden tanımlanmış parametrelere göre yeniden belirlenebilmektedir. Kullanılan mobil aracın sürüş sisteminin blok diyagramı Şekil 3.9'da verilmiştir. Güç kartı kontrol kartından aldığı verileri motorları kumanda edebilecek formata çevirmektedir.



Şekil 3.9 Sürüş sistemi blok diyagramı.

Araçta kullanılan motorların teknik bilgileri Çizelge 3.3’de belirtilmiştir.

**Çizelge 3.3** Mobil araç ve dc motor teknik özellikleri.

<b>Parametre</b>	<b>Değerler</b>
Motorun elektrik rezistansı	100-150 mΩ
Motor Tipi	4200rpm Fırçalı DC Motor
Güç	420 W
Voltaj	24 V
İndüktans	240 μH ( 1kHz )
Boyutlar	Uzunluk:20cm Çap:8cm

Kullanılan araç bir engelli aracının yeniden düzenlenmesiyle oluşturulmuştur. Ancak aracın şasesinin ve omurgasının hassas bir şekilde ayarlı olması büyük önem taşımaktadır. Aracın belirlenen rotasından sapması, belirlenen referans noktalardan geçmemesine dolayısıyla da sistemin karşılaştırma yapamayarak nesne tespit edememesine yol açacaktır.

Araç üzerinde kurulu bulunan ve sistemin çalışması için gerekli bir kısım cihaz ve malzemeler bulunmaktadır. Bu cihazlar:

### **3.7.1 Kamera**

Geliştirilen sistem için görüntüleme araçlarının çok büyük bir önemi bulunmaktadır. Yapılan tüm tespit ve tanıma hesaplamaları görüntüleme araçlarından elde edilen imge çerçeveleri üzerinden gerçekleştirilmektedir. Deneysel çalışmalarda hesaplama maliyetlerini karşılaştırmak amacıyla iki farklı tipte kamera kullanılmıştır. Bunlardan birincisi Nikon D5100 model kameradır. Bu kamera ile yüksek çözünürlüklü görüntüler elde edilmiş ve deneysel çalışmalarda kullanılmıştır. İkinci kamera ise Logitech HD Webcam C270’dir. Bu kamera ile düşük çözünürlüklü görüntüler alınmıştır.

### **3.7.2 Bilgisayar Sistemi**

Bu tez çalışmasında görüntüleme sistemlerinden elde edilen tüm imge çerçeveleri üzerinde gerçekleştirilen analiz ve hesaplama işlemleri dört çekirdekli 2.8 GHz işlemciye sahip 4 GB RAM kapasiteli bir bilgisayar kullanılmıştır.

### 3.8 Yapay Sinir Ağları (YSA)

Yapay sinir ağlarının literatürde çok çeşitli tanımları yapılmıştır. Bu tanımlardan bir kısmı aşağıda verilmiştir.

İnsan beyni üstün özellikleri nedeniyle bilim adamları tarafından matematiksel olarak modellenmeye çalışılmıştır. Doğru modelleme için beynin nörofiziksel yapısı göz önüne alınmış ve çeşitli yapay hücre ve ağ modelleri geliştirilmiştir (Elmas 2003). YSA'lar deneyimlenen bilgiyi kaydeden ve kullanan fiziksel hücreli sistemlerdir (Sağiroğlu vd. 2003).

YSA, insan beyninin özelliklerinden olan öğrenme yolu ile yeni bilgi türetebilme, yeni bilgi oluşturabilme ve keşfedebilme gibi yetenekleri yardım almadan doğrudan gerçekleştirmek amacı ile geliştirilen sistemlerdir. Yapay sinir ağlarının, programlanması çok zor ve mümkün olmayan olaylar için geliştirilmiş adaptif bilgi işleme ile ilgilenen bir bilgisayar bilim dalı olduğu söylenebilmektedir (Öztemel, 2012).

YSA, beynin bir görevi yerine getirmek için kullandığı yöntemi modellemek için tasarlanmış bir sistemdir. YSA, yapay sinir hücreleri de denilen işlem elemanlarının birbirleri ile çeşitli şekillerde bağlanmasından oluşur ve genellikle katmanlar şeklinde düzenlenir (Haykin 1994).

YSA, paralel olarak işlem yapan basit işlem elemanlarından oluşan adaptif bir ağıdır. Bu işlem elemanları geliştirilirken biyolojik sinir sisteminden ilham alınmıştır. Bu ağın işlevi işlem elemanları arasındaki geniş çaplı bağlantılar tarafından oluşturulur (Atmaca *et al.* 2001 ).

YSA sistemleri geniş bilgi parçaları üzerinden sağladıkları adaptif öğrenme kabiliyetleri, gerçek zamanlı işlemlerdeki kapasiteleri, kendi kendine organize olabilmesi ve hata toleransları nedeniyle popülerdir (Wahi *et al.* 2010).

### 3.8.1 Yapay Sinir Hücresinin Modellenmesi

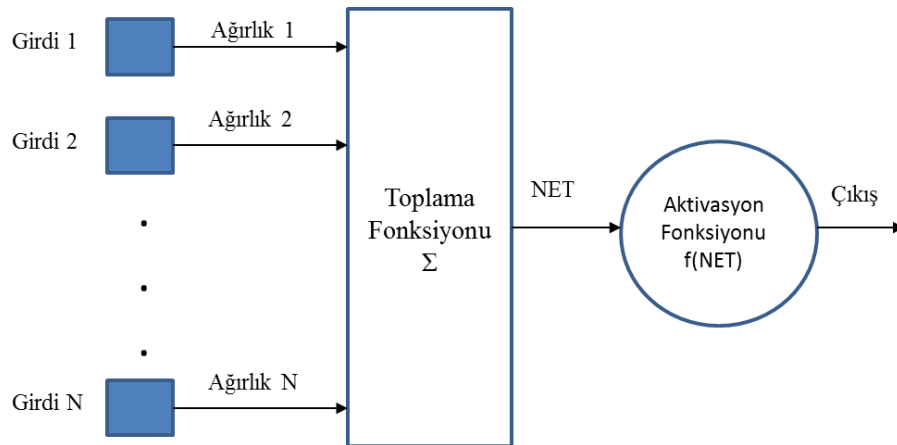
Biyolojik sinir ağlarının sinir hücrelerinden oluşması gibi, yapay sinir ağları da yapay sinir hücrelerinden meydana gelmektedir. Yapay sinir hücreleri ayrıca düğüm (node), birim (unit) veya işlem elemanı (processing unit) olarak da adlandırılmaktadır. Bir yapay sinir ağı, birbiriyle bağlantılı çok sayıda yapay sinir hücresinden meydana gelmektedir. Yapay sinir hücreleri biyolojik sinir hücrelerinin basit bir modelidir. Yapay sinir ağlarının içinde bulunan tüm sinir hücreleri bir veya birden fazla girdi alırlar ve tek bir çıktı verirler. Bu çıktı yapay sinir ağının dışına verilen bir çıktı olabileceği gibi başka bir yapay sinir hücresine girdi olarak da verilebilir (Baş 2006).

Bir yapay sinir hücresi genel olarak beş temel bileşenden oluşmaktadır.

- Girdiler
- Ağırlıklar
- Toplama fonksiyonu
- Aktivasyon fonksiyonu
- Çıktı

Bileşenler Şekil 3.10’da verilmiştir.

Girdiler, ağı öğrenmesi istenen örnekleri içermektedir. “Input” olarak da adlandırılırlar. Bu bilgiler dış ortamlardan ya da diğer işlem elemanlarından gelebilir. Vektörel olarak ifade edilmeleri gerekmektedir.



Şekil 3.10 Bir yapay sinir hücresinin yapısı (Gurney 1996).

Ağırlıklar, işlem elemanlarına giren bilgilerin hücre çıkışına yansıma derecesini belirleyen sayısal değerlerdir. Değeri ne kadar büyükse çıkış değerine o kadar fazla etkisi olmaktadır. Bir ağ yapısı içerisinde bir işlem elemanına gelen tüm bağlantıların bir ağırlık değeri bulunmaktadır. Bu şekilde her işlem elemanının sonuca ayrı ayrı etki edebilmektedir.

Toplama fonksiyonu bölümü işlem elemanlarının girişine uygulanan girdilerden net bir değer elde etmektedir. Bu çalışmada kullanılan toplama fonksiyonunda gerçekleştirilen işlemler Denklem 3.2’de ifade edilmiştir.

$$NET = \sum_{i=1}^N Ağırlık_i \cdot Girdi_i \quad (3.2)$$

Bu denklemde;  $i$  YSA işlem elemanının indis numarasını,  $N$  işlem elemanı sayısını ve  $NET$  işlem elemanının net çıktısını ifade etmektedir.

Şekil 3.10’da NET olarak ifade edilen toplam giriş bilgisi hesaplandıktan sonra işlem elemanı tarafından bu bilginin işlenerek bir çıktı üretilmesi gerekmektedir. Bu çıktının üretilmesi için kullanılan fonksiyonlara aktivasyon fonksiyonu, transfer fonksiyonu ya da harekete geçme fonksiyonu da denilmektedir. Yapılan literatür taramasında sıklıkla kullanıldığı gözlemlenen bir kısım aktivasyon fonksiyonlarından Doğrusal fonksiyon Denklem 3.3’de, Sigmoid fonksiyonu Denklem 3.4’de ve Hiperbolik tanjant fonksiyonu Denklem 3.5’de verilmiştir.

Doğrusal fonksiyon

$$Çıktı=F(NET) = NET \quad (3.3)$$

Sigmoid fonksiyonu

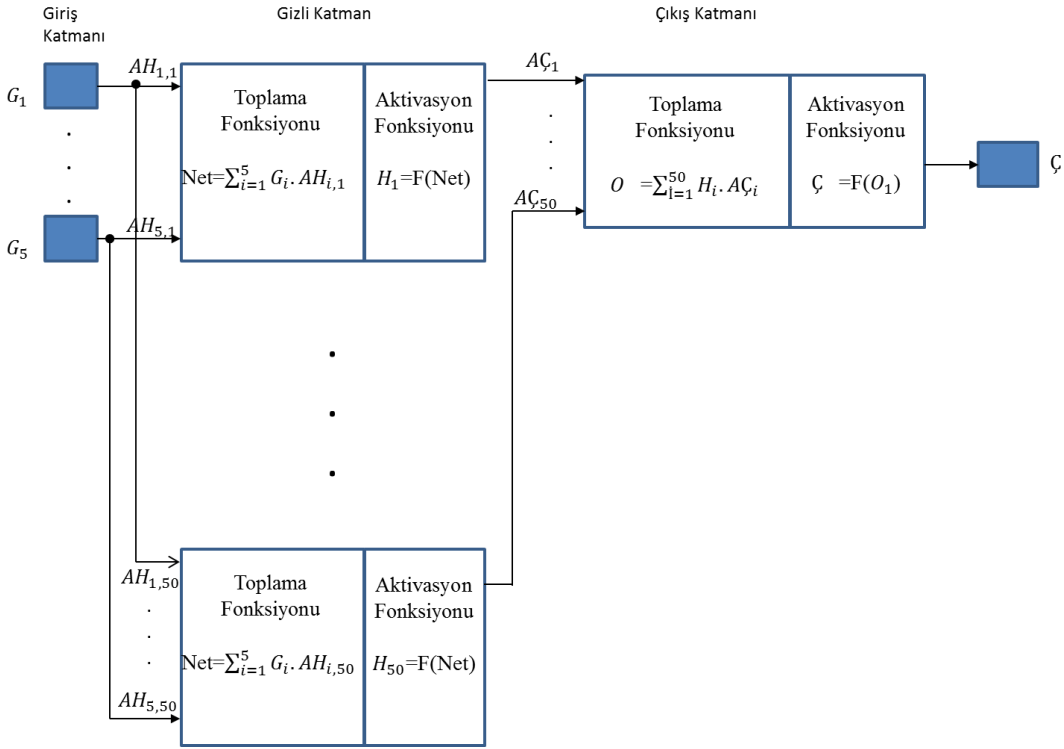
$$Çıktı=F(NET) = 1/1 + e^{-NET} \quad (3.4)$$

Hiperbolik tanjant fonksiyonu

$$Çıktı=F(NET) = (e^{NET} + e^{-NET})/(e^{NET} - e^{-NET}) \quad (3.5)$$

### 3.8.2 Yapay Sinir Ağlarının Mimarisi

Yukarıda açıklanan işlem elemanları bir araya gelerek yapay sinir ağını oluşturmaktadır. Bu elemanlar genellikle birbiriyle bağlantılı 3 katman halinde bir araya getirilmiştir. Bu 3 katman, girdi katmanı (input layer), ara katman (hidden layer) ve çıktı katmanı (output layer) olarak adlandırılır. Girdi katmanı, dış ortamdan alınan verileri ara katmanlara iletir. Bazı yapay sinir ağlarında bu katmanda herhangi bir bilgi işleme olmamaktadır. Ara katman, girdi katmanından gelen bilgileri işleyerek çıktı katmanına iletir. Gizli katman olarak da adlandırılan ara katman birden fazla olabilmektedir. Ara katmanlar çok sayıda işlem elemanı içermektedirler ve bu hücreler yapay sinir ağı içindeki diğer hücrelerle bağlantılıdır. Çıktı katmanı, ara katmandan gelen işlenmiş bilgileri dış ortama aktarmakla görevlidir (Baş 2006). Bu çalışmada kullanılan ileri beslemeli YSA mimarisinin Şekil 3.11’de verilmiştir.



Şekil 3.11 Çalışmada kullanılan 3 katmanlı ileri beslemeli yapay sinir ağı mimarisi.

Bu mimaride  $G$  girdi verilerini,  $AH$  gizli katman ağırlıklarını,  $A\zeta$  çıkış katmanı ağırlıklarını,  $H$  gizli katman çıkışlarını ve  $\zeta$  yapay sinir ağının çıktısını temsil etmektedir. Bu mimari 3 katmandan oluşmuştur. Girdi katmanında 5 adet girdi verisi bulunmaktadır. Gizli katman 50 adet işlem elemanından oluşmuştur. Çıktı katmanı ise bir adet çıktı





grubunu işleyerek çıktı bilgilerini hesaplar. Hata fonksiyonunun değerini en aza indirgeyen ağırlık vektörleri öğrenme sorununun çözümü olan ağırlık vektörleridir.

Geri yayılım algoritmasındaki en önemli süreçlerden biri de beklenen çıktılar ile gerçek çıktılar arasındaki hatanın en aza indirgenmesidir. Bu işlem hata fonksiyonunun yerel minimum noktalarını hesaplayarak gerçekleştirilir. Hata fonksiyonunun yerel minimum noktasına ulaşmak için iterasyonlar boyunca hata eğrisinin türevi alınarak eğim hesaplaması yapılır. Her seferinde eğrinin üst kısmından minimumun bulunduğu alt kısma doğru bir iniş gerçekleşir. Bu nedenle uygulanan bu yöntem Gradyan İniş Yöntemi denilmektedir.

Tasarlanan ağda öğrenme fonksiyonu olarak Matlab R2013 yazılımında '*learn\_gdm*' fonksiyonu olarak ifade edilen gradyan iniş (Gradient Descent) yöntemi kullanılmış, ağır hata performansı için ise OKH (ortalama karesel hatalar - MSE) yöntemi kullanılmıştır.

Bu çalışmada OKH fonksiyonunun ayrıntılı ifadesi Denklem 3.6'da verilmiştir. Bu eşitlikte  $N$  ağır çıktı sayısını,  $i$  ise çıktı indisini ifade etmektedir. Burada beklenen değer ( $B$ ) ile gerçek çıktı ( $C$ ) sayısal değerleri birbirine yaklaştırılmaya çalışılır.

$$Hata = \frac{1}{N} \sum_{i=1}^N (B_i - C_i)^2 \quad (3.6)$$

Eğitim için ağırlıkları Levenberg-Marquardt (LM) optimizasyonuna göre güncelleyen Matlab yazılımının '*train\_lm*' fonksiyonu kullanılmıştır. Bu fonksiyon en hızlı sonuç veren eğitim fonksiyonu olup diğer fonksiyonlara göre daha fazla bilgisayar hafızasına ihtiyaç duymaktadır (İnt.Kyn.5).

### 3.8.4 Yapay Sinir Ağlarının Eğitilmesi

Denetimli, denetimsiz ve destekleyici eğitim olmak üzere üç çeşit eğitim modeli bulunmaktadır. Denetimli öğrenmede yapay sinir ağına hem girdiler hem de girdiler karşılığında oluşması gereken çıktılar verilir. Sistemin görevi girdi değerlerinin çıktı değerlerine haritalanması dolayısıyla girdi değerleri ile çıktı değerleri arasındaki ilişkiyi çözmektir (Adıyaman 2007).

Denetimsiz öğrenmede yapay sinir ağına girdiler karşılığında oluşması gereken çıktılar verilmez. Ağın girdi ve çıktı ilişkisini kendi kendine öğrenmesi beklenir. Daha çok sınıflandırma problemlerinde kullanılır. Kohonen ve ART yapay sinir ağı modellerinde kullanılmaktadır.

Destekleyici öğrenmede yapay sinir ağına girdiler karşılığında çıktılar verilmez ancak bulunan çıktının doğru olup olmadığına dair bir sinyal verilir. LVQ yapay sinir ağı modelinde kullanılır.

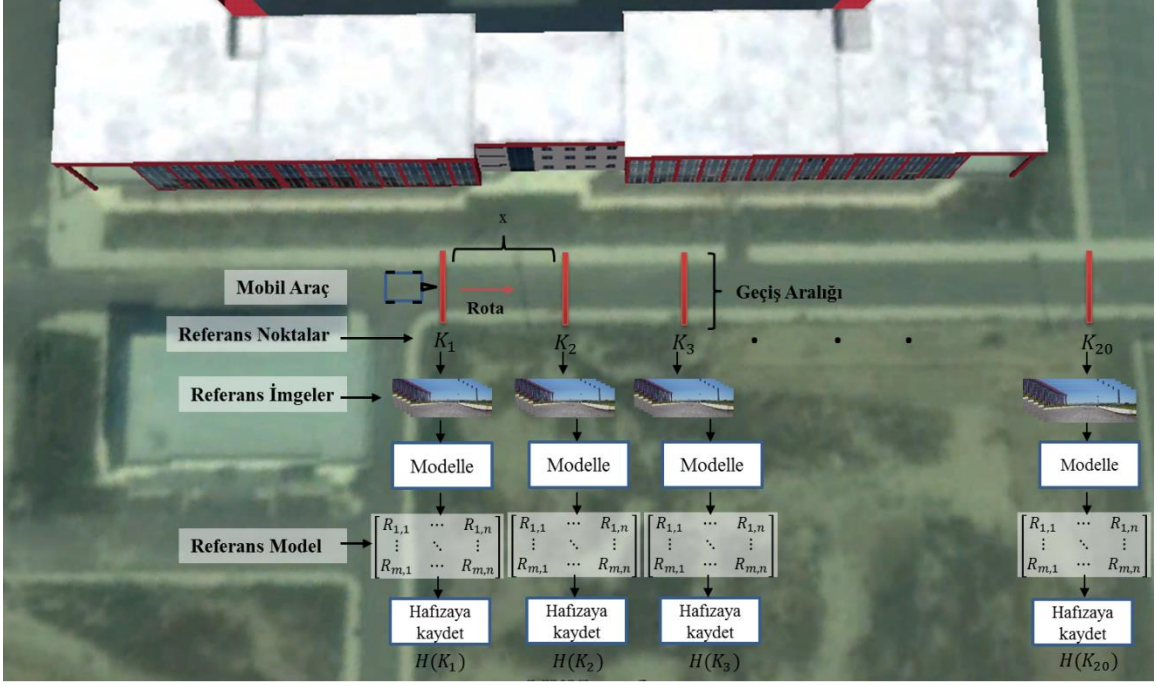
Yapılan çalışmada denetimli eğitim modeli kullanılmıştır. Tasarlanan ileri beslemeli geri yayınlı YSA'na girdi verileri ile birlikte beklenen çıktılar da verilerek tüm katmanlardaki ağırlık vektörlerinin güncellenmesi sağlanmıştır. Yayalar için 200 adet, otomobil ve bisikletliler için ise 100 adet veriden oluşan girdi setleri kullanılarak YSA eğitilmiştir.

### **3.9 Dinamik Nesne Tespit İşlemleri**

Dinamik nesne tespit işlemlerinde beş farklı yöntem uygulanmıştır. Bu yöntemlerin hepsinde de uygulanan iki adet genel prensip mevcuttur. Bu prensipler;

1. Gerçek zamanlı çalışma öncesi rota üzerinde varsayılan arka plan modelleme ve hafızaya kaydetme,
2. Gerçek zamanlı çalışma sırasında rota üzerinde anlık modelleme ve hafızadaki referans model ile karşılaştırmadır.

Gerçek zamanlı hareket öncesi yapılan işlemleri belirten “1” numaralı prensibin rota üzerinde  $K_1$  ile  $K_{20}$  referans noktaları arasında uygulanışını gösteren senaryo Şekil 3.13'de resim olarak verilmiştir.



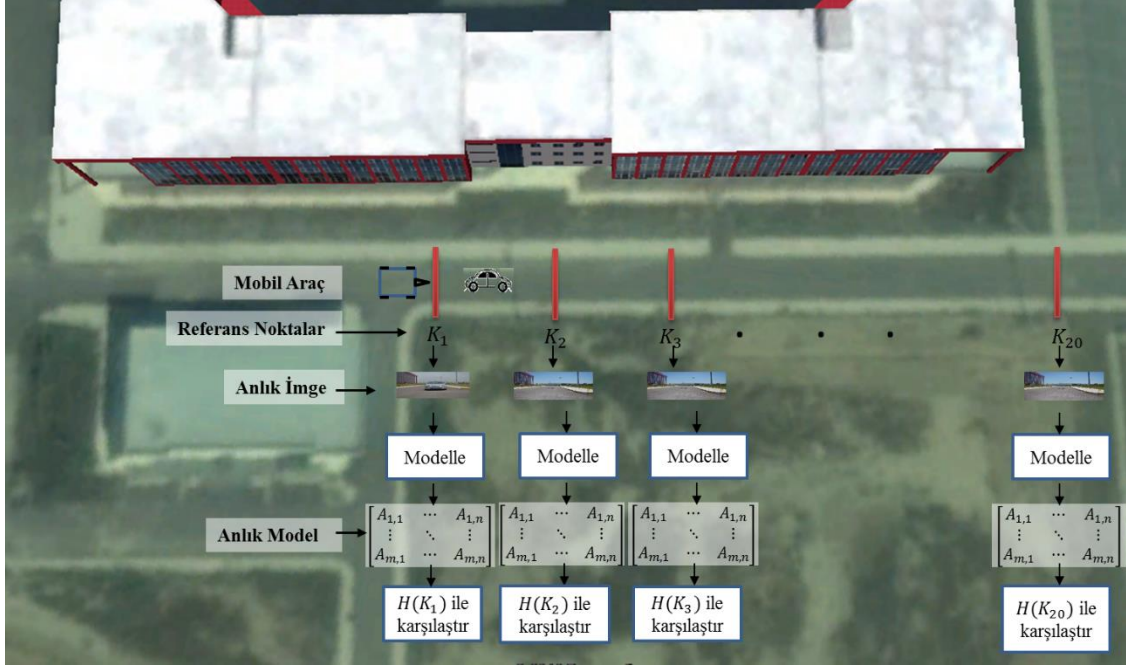
Şekil 3.13 “1” numaralı prensibin uygulama adımları.

Bu senaryoya göre;

- Mobil araç kullanılan yöntemle göre değişmekle beraber 1, 5 ya da 10 defa rotayı takip ederek referans noktalardan geçiş yapar.
- Her geçişte  $K_1$  ile  $K_{20}$  arasındaki referans noktaların tam üzerindeyken varsayılan arka plana ait bir adet görüntüyü fotoğraflar.
- Son geçişle birlikte  $K_1$  ile  $K_{20}$  arasındaki her bir referans noktası için seçilen yöntemle göre modelleme işlemi yapılarak referans modeller elde edilir.
- Her bir referans noktası için elde edilen referans modeller sistem hafızasına kaydedilir.

Şekilde iki referans noktası arasındaki mesafeyi gösteren  $x$  değişkeni ve *Geçiş Aralığı* parametrik olarak değiştirilebilmektedir.

Gerçek zamanlı hareket ile ilgili uygulamaların yapıldığı “2” numaralı prensibin rota üzerinde  $K_1$  ile  $K_{20}$  referans noktaları arasında uygulanışını gösteren senaryo Şekil 3.14’de resim olarak verilmiştir.



Şekil 3.14 “2” numaralı prensibin uygulama adımları.

Gerçek zamanlı hareket senaryosunda;

- Mobil araç gerçek zamanlı hareketine başlar.  $K_1$  ile  $K_{20}$  arasındaki referans noktalarının her birinin tam üzerinden geçerken arka plana ait görüntüyü fotoğraflar.
- Geliştirilen Yöntemlerden hangisi ile çalışıyorsa o yöntemle göre aldığı imge çerçevesini modeller.
- Oluşturduğu anlık model ile hafızada kayıtlı ilgili Yöntemin referans modelini karşılaştırır.
- Karşılaştırma sonucu nesne var ise tespit ve tanıma işlemine geçilir yok ise yola devam edilir.

Nesne tespit işlemlerinde beş farklı yöntem kullanılmıştır. Yöntemlerin tamamında yukarıda belirtilen genel prensipler kullanılmaktadır. Ancak varsayılan arka plan modelleme işlemleri ve nesnelerin tespit işlemlerinde farklı yöntemler kullanılmıştır. Kullanılan yöntemler:

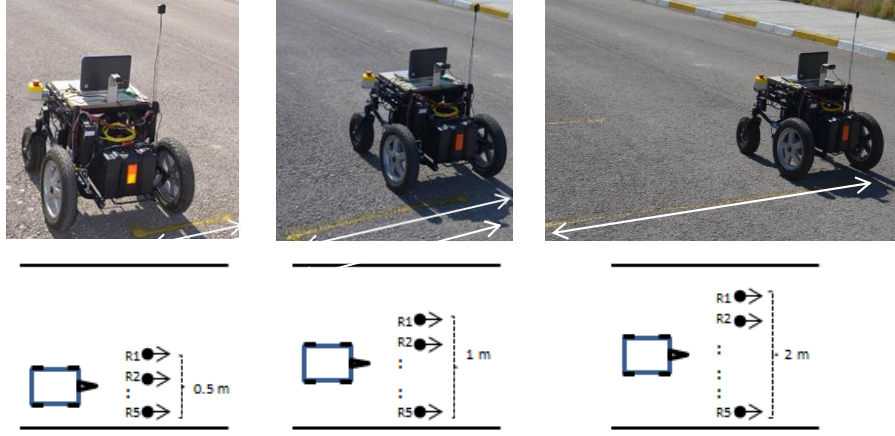
1. GPS Destekli Hareket Bölgesi Belirleme ve Geometrik Dönüşüm,
2. GPS Destekli Çerçeve Hizalama ve Çerçeve Çıkartma,

3. Gradyan Tabanlı Gri Seviye Değişimi Modellemesi ile Hareketli Nesne Tespiti,
4. Özelleştirilmiş Arka Plan Modellemesi ile Dinamik Nesnelerin Tespiti,
5. Özelleştirilmiş İmge Çakıştırma Yöntemi ile Hareketli Nesnelerin Tespiti Yöntemleridir.

### 3.9.1 Yöntem 1: GPS Destekli Hareket Bölgesi Belirleme ve Geometrik Dönüşüm

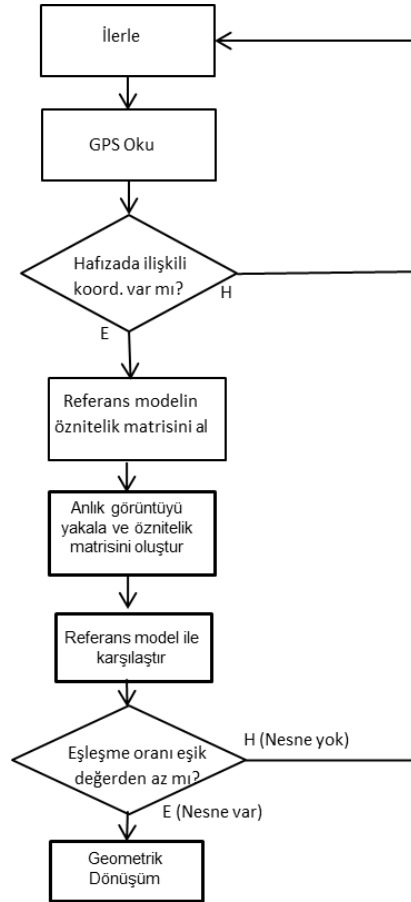
Kullanılan mobil araç programlandığı rotası boyunca yatay ekseninde yaklaşık olarak aynı noktalardan geçmektedir. Bu durumda herhangi bir  $K_i$  koordinatında yatay eksen boyunca aracın geçebileceği noktalardan alınacak varsayılan arka plan görüntüleri hemen hemen aynı karakteristikte olacaktır. Belirlenen  $K_i$  referans noktalarında arka planı temsil edebilecek bir görüntü nesnesi elde edilebildiği durumda anlık görüntüler ile kıyas edilerek görüntüdeki hareketli nesneler elde edilebilecektir. Bu çalışmada arka planı temsil etmesi için bir referans öznitelik matris modeli oluşturulmuştur. Bu modelin oluşturulması için; öncelikle araç üzerindeki kamera ile önceden belirlenmiş rota boyunca belirlenen  $K_i$  noktalarından varsayılan arka plan görüntü çerçeveleri alınır. Bu görüntü çerçeveleri referans görüntüler  $R_i$  olarak adlandırılmıştır. Bu çerçeveler Şekil 3.15'de gösterildiği gibi belirli yatay eksen aralıklarında ve belirli bir sayıda alınır. Bu çalışmada her bir koordinat için beşer adet görüntü çerçevesi referans alınmıştır. Her bir  $R_i$  çerçevesinin öznitelikleri çıkarıldıktan sonra her birinde ortak olan en güçlü öznitelikler belirlenir. Bu şekilde en güçlü özniteliklerden oluşan referans öznitelik matris modeli elde edilmiş olur. Bu model matrisi kendi GPS koordinatı ile birlikte sistem hafızasına kaydedilir. Bu model matrisleri mobil aracın gerçek zamanlı hareketi sırasında alınacak görüntülerle aynı arka plan özelliklerine sahip olacaktır. Bu şekilde  $K_i$  konumundaki varsayılan arka plan sisteme öğretilmiş ve aracın engel belirlemesi ve çevresindeki nesneleri tespit edebilmesi için değerli bilgiler çıkarabilmesi sağlanmıştır. Bu çalışmada; mobil aracın yatay eksen üzerinde geçiş yapabileceği aralıklar 0,5 m ve 2 m arasında değiştirilmiştir.

Çalışma iki adımdan oluşmuştur. 1. adımda gerçek zamanlı işlemler öncesi bir defa yapılan hazırlık çalışması yapılmış ve 2. adımda ise gerçek zamanlı işlemler gerçekleştirilmiştir.



Şekil 3.15 Yatay eksen geçiş aralıkları.

Hareketli nesnelerin tespiti için oluşturulan sistemin genel akış şeması Şekil 3.16’da verilmiştir. İnsansız kara aracı hareketli nesne tespiti için ilerlemeye başlamadan önce belirlenen güzergâha ait her bir  $K_i$  referans noktası için fotoğraflanan referans imgelerin öznelik matris modelleri oluşturulur.



Şekil 3.16 Mobil araç ilerleyişinde kullanılan genel akış şeması.

Bu modeller temel referans modeli olarak özniteliklerin karşılaştırılmasında kullanılacaktır. Oluşturulan algoritmaya göre araç belirlenen güzergâhta görüntü alarak ilerlemeye başlar ve eş zamanlı olarak GPS konumunu okur. Okuduğu konum hafızada ilişkili bir konum ise ilişkili konuma ait referans öznitelik matris modeli ile anlık görüntünün öznitelik matrisi nicel olarak eşleştirilir. Eşleşme oranı yüksek ise anlık görüntüde herhangi bir nesnenin bulunmadığına karar verilirken, eşleşme oranı düşük ise anlık görüntüde hareketli nesne ya da nesnelerin bulunabileceğine karar verilerek nesne silueti çıkarma adımlarına geçilir

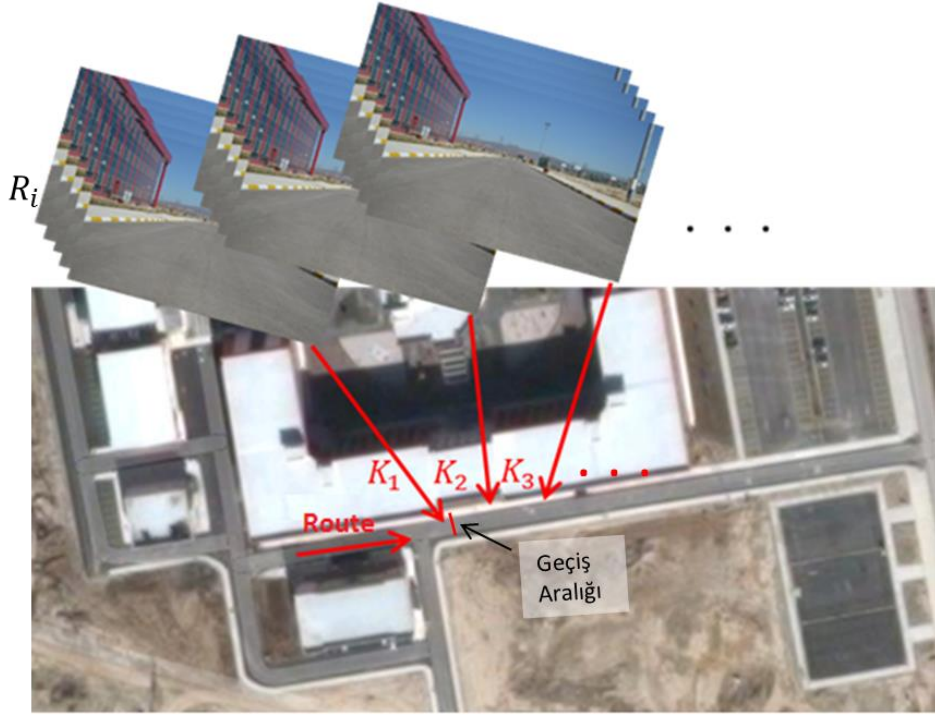
### **3.9.1.1 Referans Öznitelik Matris Modelinin Oluşturulması**

Birinci Yöntemde İKA'nın üzerinde bulunan kameranın ürettiği gerçek zamanlı (anlık) imge çerçeveleri ile GPS sisteminden elde edilen koordinat bilgilerinin senkron olarak çalışmasını sağlayan bir sistem geliştirilmiştir. Öncelikli hedef gerçek zamanlı görüntüler ile çalışırken bilgisayar sistemlerinin CPU ve hafıza kapasitelerini yormadan yüksek doğrulukta karşılaştırmalar yaparak sonuç almaktır. Aksi takdirde kararlı sistemler geliştirilse bile pratikte uygulama şansı bulunmayabilmektedir. Sistemin doğru olarak çalışabilmesi için uygulama öncesi gerçekleştirilmesi gereken bir takım hazırlıklar bulunmaktadır. Bu hazırlıklar aşağıdaki şekilde özetlenebilir;

#### **1. Adım**

Bu adımda referans imgeler elde edilmektedir. Üniversite kampüsü içerisinde belirlenen bir rota boyunca GPS koordinatlarının ( $K_i$ ) her belirgin değişimine karşılık gelen konumlardan varsayılan yani hareketli nesne içermeyen arka plan imgeleri fotoğraflanmıştır ve bunlar referans imgeler olarak adlandırılmışlardır. Bu yöntemde her bir referans GPS koordinatı için beş farklı referans imge kullanılmıştır. İmgeler her bir koordinat için Şekil 3.15'de verilen geçiş aralıkları dâhilinde aracın geçebileceği farklı noktalardan ve farklı bakış açılarıyla fotoğraflanarak elde edilmiştir. Yapılan işlem kısaca Şekil 3.17'de ifade edilmiştir.





Şekil 3.17 Belirlenmiş koordinatlardan elde edilen referans imgeler.

## 2. Adım

Bu adımda rota boyunca her bir  $K_i$  GPS koordinatında fotoğraflanan referans imgelerin öznitelikleri çıkartılmaktadır.

İmge üzerinde herhangi bir  $(x,y)$  konumunda bulunan her bir öznitelik  $1 \times 64$  boyutunda bir vektör olarak ifade edilmektedir. Bu çalışmada SURF, Harris ve FAST olmak üzere başlıca 3 adet öznitelik çıkartma algoritması kullanılmıştır. Söz konusu algoritmaların performansları karşılaştırılarak mevcut arka planın karakteristiğine en uygun öznitelik çıkartma algoritmasının belirlenmesi amaçlanmıştır.

SURF algoritması bir imgedeki öznitelikleri bulurken farklı standart sapmaya sahip Gauss filtreler uygulanmış ölçek uzaylarında işlem yapar. Farklı ölçeklerde bulunan imgelerden DoG (Difference of Gaussians) fonksiyonu uygulanarak, yerel en küçük ve en büyük noktaları öznitelik noktası olarak belirlenmektedir. Bu şekilde ölçekten bağımsız olan en kararlı noktalar öznitelik noktası olarak belirlenmektedir. FAST ve Harris algoritmaları daha çok imge üzerinde bulunan köşeleri öznitelik noktası olarak tespit etmektedir. FAST algoritması, piksellerin etrafında kesikli bir daire belirleyerek

ışık şiddetinin değişimine göre bu pikselin köşe ya da kenar olup olmadığını tespit etmektedir. Harris ise basitçe farklı yönlerdeki gradyan değişimlerinden bir matris oluşturmakta ve bu matrisin öz değerlerine göre köşeleri belirlemektedir (Erhan *et al.* 2013).

### 3. Adım

Belirlenmiş her bir GPS koordinatı için fotoğraflanmış 5 referans imge seti için farklı öznitelik matrisleri hesaplanmıştır. Bu adımdaki amaç her bir koordinatı tek bir öznitelik matrisi ile temsil etmektir. Bu nedenle referans imgelerin hepsinde de ortak olan en kararlı özniteliklerin bulunması gereklidir. Bu matris model öznitelik matrisi olarak adlandırılmıştır. Bu şekilde yazılımın çalışacağı bilgisayar sistemlerinde 3264x4928 ve 480x640 boyutlarındaki iki farklı imge çerçeve matrislerinin işlenmesi yerine arka planı temsil edecek çok daha küçük boyutlarda öznitelik matrisleri kullanılmıştır. Böylece bilgisayar CPU ve hafıza kapasitelerinde büyük ölçüde tasarruf sağlandığı gibi işlem hızında da büyük artışlar sağlanmıştır.

Mobil aracın doğrudan rotası üzerinde yakın çevresinde bulunabilecek ya da uzak çevresinde bulunabilecek hareketli nesnelerin tespitinin daha kesin olarak yapılabilmesi için referans imgeler 13 ayrı parçaya bölünülerek değerlendirilmiştir. Bölümlerden  $B_1$  daha az detay bulunduran yol kısmını içerirken  $B_2$  ile  $B_{13}$  arası bölümler detay içeren kısımlardan oluşmuştur. Parçalara ayrılmış örnek bir referans imge Şekil 3.18'de gösterilmiştir. Burada imgelerin üst kısmındaki hareketler uygulama alanı dışında olduğundan dikkate alınmamıştır. İmgenin orta kısmı ( $B_2.. B_{13}$ ) yoğun işlem alanı olarak belirlenmiştir. İmgenin alt kısmı ( $B_1$ ) aracın hareketi sırasında orta kısımda belirlenen nesne alt kısma geçeceğinden ve nesne büyüyeceğinden bilinen algoritmalar uygulanmış ve bir performans karşılaştırması yapılmamıştır.

Referans imge setinin tamamında ortak olarak bulunan özniteliklerin hesaplanabilmesi için 1x64 boyutunda ifade edilen öznitelik vektörleri arasındaki doğrusal uzaklıkların hesaplanarak en küçük olanın seçilmesi gereklidir. Vektörler arasındaki doğrusal uzaklıkların tespiti için ED, SAD ve NCC yöntemleri uygulanmıştır.



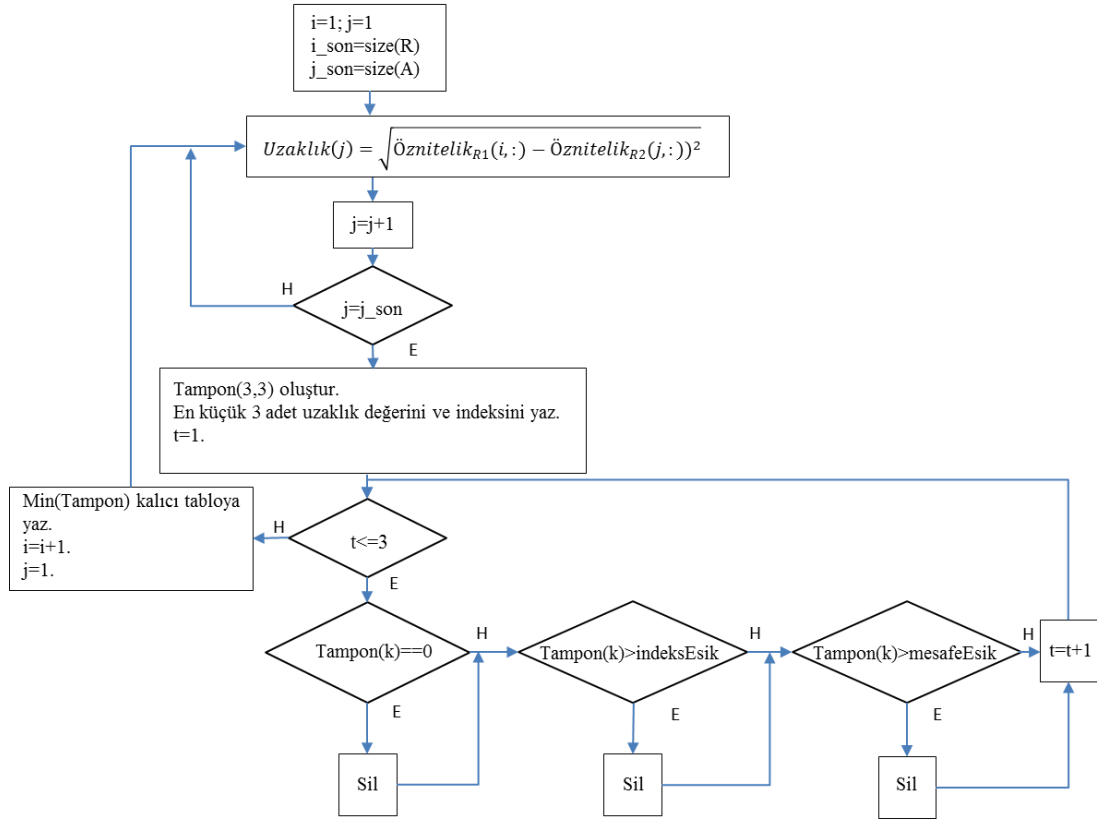
Şekil 3.18 Örnek referans imge bölümlemesi.

NCC yönteminde ED ve SAD yöntemlerinden farklı olarak vektörler arasındaki doğrusal uzaklık değil vektörlerin benzerliği hesaplanır. Bu nedenle oluşan değerler arasında en küçük değer değil en yüksek değer bulunmalıdır. Hesaplanan değer 1'e ne kadar yakın ise iki vektör birbirine o kadar benzer demektir(Martin and Crowley 1995).

Herhangi iki adet referans imgenin öznitelik matrisleri Denklem 3.7'de gösterildiği şekilde ifade edilebilir. Bu matrislerin her bir satırı bir adet öznitelige ait sayısal değerleri ifade etmektedir. Bu nedenle R1 ve R2 referans imgelerinde aynı noktayı işaret eden ortak özniteliklerin bulunması için;  $\text{Öznitelik}_{R1}$  matrisinin her bir satır vektörüyle  $\text{Öznitelik}_{R2}$  matrisinin tüm satır vektörleri döngü içerisinde karşılaştırılarak en küçük doğrusal uzaklığa sahip olanların ortak öznitelik olarak belirlenmesi gerekir.

$$\begin{aligned} \text{Öznitelik}_{R1} &= \begin{bmatrix} x_{1,1} & \cdots & x_{1,64} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,64} \end{bmatrix} \\ \text{Öznitelik}_{R2} &= \begin{bmatrix} y_{1,1} & \cdots & y_{1,64} \\ \vdots & \ddots & \vdots \\ y_{m,1} & \cdots & y_{m,64} \end{bmatrix} \end{aligned} \quad (3.7)$$

Ortak öznitelikleri belirlemede tesadüfi eşleşmeler nedeniyle bir takım zorluklar oluşabilmektedir. Bu sorunu aşmak amacıyla referans imgeler arasındaki aksel kaymaların belirli bir eşik değerden daha büyük olamayacağı varsayılarak *indeksEşik* ve *mesafeEşik* isiminde iki adet eşik değeri belirlenmiş ve eşleştirme işlemi için özgün bir algoritma geliştirilmiştir. Yukarıda verilen kod parçacığını akış çizelgesi olarak ifadesi Şekil 3.19'da sunulmuştur.



Şekil 3.19 Öznitelikler arasındaki en kısa mesafenin bulunması amacıyla geliştirilen algoritma.

Bu algoritmaya göre;  $\text{Öznitelik}_{R_1}$  matrisinin bir satırı ile  $\text{Öznitelik}_{R_2}$  matrisinin tüm satırları arasındaki doğrusal uzaklık hesaplanmakta ve en küçük üç adet değer indisleri ile birlikte *Tampon* isimli matrise yazdırılmaktadır. *Tampon* matriste tesadüfi eşleşmeler nedeniyle *mesafeEsik* ve *indeksEsik* değerlerini aşan satırlar silinerek matris filtrelenmekte ve kalan satırlar içinde en küçük değer bulunduğusu satır *Uzaklık* isimli diziye yazdırılmaktadır. Böylece  $\text{Öznitelik}_{R_1}$  matrisinin *i*'nci satırına en yakın olan  $\text{Öznitelik}_{R_2}$  satırı tespit edilmektedir. Bu işlem döngü içerisinde  $\text{Öznitelik}_{R_1}$  matrisinin tüm satırları için gerçekleştirilerek  $R_1$  ve  $R_2$  referans imgeleri arasındaki ortak öznitelikler tespit edilmektedir.

Bu işlemin diğer referans imgeler arasında da yapılması sonucunda; her bir GPS koordinatını temsil eden referans imge setinin tamamında ortak olarak bulunan en güçlü öznitelikler tek bir matris halinde tespit edilmiştir. Bu matris gerçek zamanlı uygulama yapılırken rota üzerindeki GPS koordinatlarından alınacak anlık arka plan imgelerinin karşılaştırılacağı model öznitelik matrisidir.

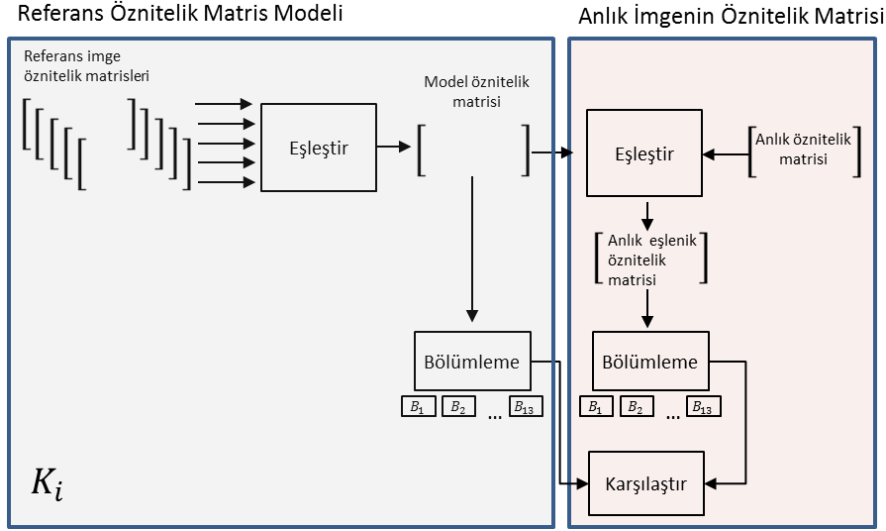
#### 4. Adım

Bu adımda o ana kadar elde edilen veriler hafızada depolanır. Hafızaya kayıt yapılırken sadece model öznitelik matrisinin depolanması yeterli olmaz. GPS koordinat bilgisi ve kullanılan öznitelik çıkartma algoritması da model öznitelik matrisiyle ilişkilendirilerek hafızada saklanır.

##### 3.9.1.2 Gerçek Zamanlı Hareketli Nesne Tespiti

Hareketli nesnelerin gerçek zamanlı tespiti için mobil araç kontrol sisteminin hafızasında rotası boyunca geçeceği her bir GPS koordinatı için bir model öznitelik matrisi oluşturulmuştur. Mobil araç belirlenen rota boyunca ilerlerken üzerinde bulunan GPS alıcı sistem sayesinde 500 ms’de bir koordinat bilgilerini almaktadır. Zamansal bir döngü içerisinde bu bilgi hafızadaki koordinat bilgileri ile karşılaştırılır. Uyuşma sağlandığı anda aracın üzerinde bulunan kamera ile anlık bir fotoğraf çekilir. Bu fotoğraf anlık görüntü olarak adlandırılır. Anlık görüntünün hafızada ilişkili öznitelik çıkartma algoritması ile anlık öznitelik matrisi oluşturulur. Bu adımdan sonra model öznitelik matrisi ile anlık öznitelik matrisinin eşleşen öznitelikleri ED, SAD ve NCC yöntemlerinden birisiyle hesaplanır. Hesaplama sonrası anlık eşlenik öznitelik matrisi elde edilir.

Rota üzerindeki  $K_i$  'nci GPS koordinat noktasındaki hareketli nesne tespiti için gerçekleştirilen gerçek zamanlı işlemler Şekil 3.20’de ifade edilmiştir. Şekilde sistem hafızasında oluşturulan öznitelik matris modeli (solda) ile gerçek zamanlı hareketli nesne tespit modeli (sağda) görülmektedir. Anlık eşlenik öznitelik matrisi üzerinde Şekil 3.18’de  $B_1$  ile  $B_{13}$  arasında ifade edilen bölümlerin eşleşen öznitelik nokta sayıları ayrı ayrı hesaplanır. Ölçüm neticesinde her bölüm için eşik değerin üstünde bir sayıda öznitelik uyuşması sağlanmış ise sahne boş yani herhangi bir yabancı nesne bulunmadığına kanaat getirilir. Bu durumda araç kontrol sistemine ilerle bilgisi iletilir. Herhangi bir bölümde eşik değerden daha az miktarda bir uyuşma var ise arka planda yabancı bir nesne bulunduğu kararı verilir.



**Şekil 3.20**  $K_i$  'nci koordinat noktası için hafızadaki model ile gerçek zamanlı bilgilerin karşılaştırılması.

Geliştirilen bu modelin CPU ve RAM bazındaki hesaplama yükü Geometrik Dönüşüm (Gonzalez *et al.* 2009) + Çerçeve Çıkarma (Weng *et al.* 2010) ve Gauss (Stauffer and Grimson 1999) yöntemlerine göre daha az olarak gerçekleşmektedir. CPU ve RAM bazında yapılan testler Çizelge 4.6'da verilmiştir.

Arka planda yabancı bir nesnenin bulunduğu anda nesne silüetinin tespiti için gerekli olan işlem adımlarına geçilir. Kameranın hareketinden doğan gürültünün dengelenmesi yani imge çerçevelerinin referans imgeye geometrik olarak hizalanması gerekir. Bu işlem birinci bölümde de bahsedildiği üzere bir imge çakıştırma yöntemi olarak geometrik dönüşüm ile gerçekleştirilir. Bu çalışmada geometrik dönüşümün öteleme hareketi hesaplanarak imge çerçeveleri arasında hizalama gerçekleştirilmiştir.

Girdi uzayındaki imgeyi  $f(w, z)$  olarak kabul ettiğimizde çıktı uzayında dönüştürülmüş  $g(x, y)$  imgesini  $f(w, z)$  cinsinden Denklem 3.8 'de görüldüğü şekilde ifade edilebilmektedir. Geometrik dönüşümün öteleme matrisini ifade eden T matrisi Denklem 3.9'da gösterildiği şekilde kullanılmaktadır (Gonzalez *et al.* 2009).

$$g(x, y) = f(T^{-1}\{(x, y)\}) \quad (3.8)$$

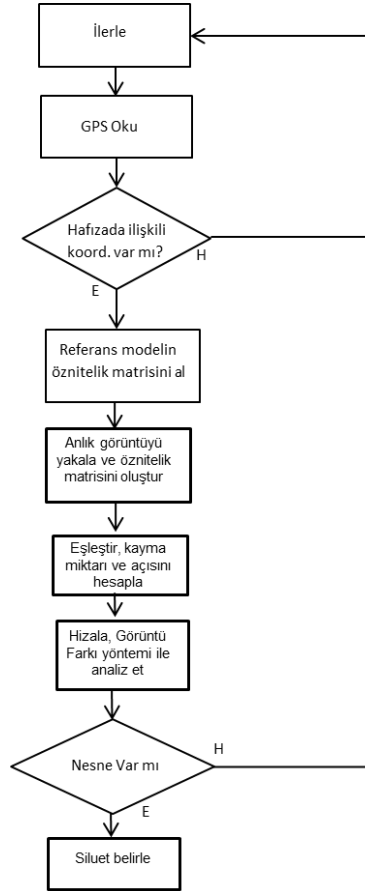
Dönüşüm matrisleri: Öteleme hareketi için,

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \delta_x & \delta_y & 1 \end{bmatrix} \quad (3.9)$$

Burada  $\delta_x$ ,  $x$  eksenindeki ötelenme miktarını,  $\delta_y$  ise  $y$  eksenindeki ötelenme miktarını piksel cinsinden ifade etmektedir. Gerçekleştirilen dönüşüm uygulamasıyla uzamsal değişimleri dengelenen gerçek zamanlı arka plan imge çerçevesi bir önceki imge çerçevesinden çıkartılarak nesne silueti elde edilir.

### 3.9.2 Yöntem2: GPS Destekli Çerçeve Hizalama ve Çerçeve Çıkartma

İki yöntem arka plan modellenmesi olarak aynı ancak nesne tespiti noktasında farklı özellikler taşımaktadır. Bu yöntemde oluşturulan algoritmanın genel akış şeması Şekil 3.21’de verilmiştir. Bu algoritmaya göre araç belirlenen rotada görüntü alarak ilerlemeye başlar ve eş zamanlı olarak GPS konumunu okur.



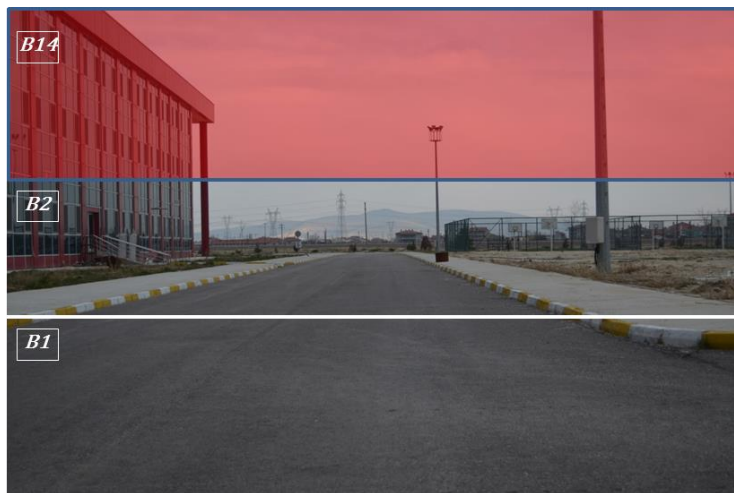
Şekil 3.21 Yöntem-2 dinamik nesne tespitinde kullanılan akış şeması.

Okuduğu konum hafızada ilişkili bir konum ise ilişkili konuma ait referans öznitelik matris modeli ile anlık görüntünün öznitelik matrisi eşleştirilir. Bu şekilde ardışık imge çerçeveleri arasında iki boyutlu çerçeve hizalama yöntemi (Gonzalez *et al.* 2009) ile iki matris arasındaki kayma vektörü hesaplanır. Bu aşama sonrasında çerçeve çıkarma işlemleri gerçekleştirilip anlık görüntüde hareketli ya da hareketsiz bir nesne bulunup bulunmadığı analiz edilir.

### 3.9.2.1 Referans Öznitelik Matris Modelinin Oluşturulması

Bu yöntemde Yöntem-1 ile aynı şekilde hafızada kaydedilmiş olan referans öznitelik matris modeli kullanılmaktadır. Bu model temel referans modeli olarak özniteliklerin karşılaştırılmasında kullanılmıştır. Model matrisin hazırlanışı Yöntem-1’de açıklanan “3.9.1.1 Referans Öznitelik Matris Modelinin Oluşturulması” bölümünde verilmiştir.

Yöntem-2’de yapılan imge bölümlenme işleminde Yöntem-1’den farklı olarak referans ve anlık imgeler 3 parçaya bölünerek değerlendirilmiştir. Bölümlere ayrılmış örnek bir imge Şekil 3.22’de gösterilmiştir.  $B_1$  daha az detay bulunduran yol kısmını içerirken  $B_2$  detay içeren kısımlardan oluşmuştur.  $B_1$  ve  $B_{14}$  bölümleri ilgilenilen alan dışında olduğu için dikkate alınmamıştır. İmgenin orta kısmı ( $B_2$ ) odaklanılacak bölüm olarak belirlenmiştir. Sadece bu bölümdeki hareketli nesnelere taranmış ve tespit edilmiştir.



Şekil 3.22 Yöntem-2 bölümlenmiş imge.

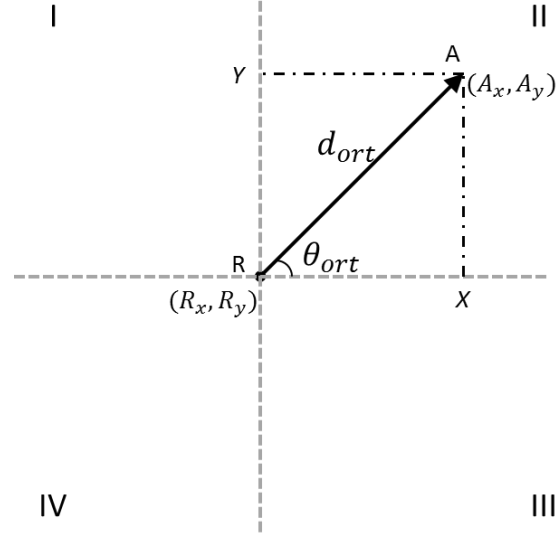


Referans imge setinin tamamında ortak olarak bulunan özniteliklerin hesaplanabilmesi için  $1 \times 64$  boyutunda ifade edilen öznitelik vektörleri arasındaki uzaklıkların hesaplanarak en küçük olanın seçilmesi gereklidir. Bu işlem Yöntem-1’de açıklandığı gibi ED, SAD ve NCC yöntemleri ile gerçekleştirilmiştir.

### 3.9.2.2 Gerçek Zamanlı Hareketli Nesne Tespiti

Gerçek zamanlı işlemlerin başlangıcında mobil araç kontrol sisteminin hafızasında rotası boyunca belirlenmiş her bir GPS koordinatı için varsayılan arka planı temsil eden bir model matris oluşturulmuş durumdadır. Mobil araç belirlenen rota boyunca ilerlerken üzerinde bulunan GPS alıcı sistem sayesinde 500 ms. ‘de bir koordinat bilgilerini almaktadır. Zamansal bir döngü içerisinde bu bilgi hafızadaki koordinat bilgileri ile karşılaştırılır. Uyuşma sağlandığı anda aracın üzerinde bulunan kamera ile arka planın görüntüsü yakalanır. Bu görüntü anlık görüntü olarak adlandırılır. Anlık görüntünün SURF öznitelik çıkartma algoritması ile kendine has anlık öznitelik matrisi oluşturulur. SURF yönteminde her bir öznitelik  $1 \times 64$  birimden oluşan satır vektörler halinde belirlenmektedir. Böylece hem model matris hem de anlık matris satır vektörler haline getirilmiştir. Bu adımdan sonra referans model matris ile anlık öznitelik matrisinin öznitelikleri eşleştirilerek aralarındaki iki boyutlu kayma vektörü bulunur. İki matrisin eşleştirilmesi amacıyla model matrisin her bir elemanı için anlık matrisin en yakın mesafede bulunan elemanı belirlenir. Vektörler arası doğrusal uzaklık belirleme için ED yöntemi kullanılmış ve model matrisin her bir elemanına en yakın anlık matris elemanlarından oluşan bir  $Uzaklık_{i,j}$  matrisi oluşturulmuştur. Bu matrisin oluşturulması işlemlerinde Şekil 3.19’da ayrıntıları açıklanan algoritma kullanılmıştır.

Hesaplama sonrası referans görüntü ile anlık görüntü arasındaki iki boyutlu kayma vektörü yaklaşık olarak belirlenmiştir. Kayma vektörü uzamsal bir kayma mesafesi ve bir kayma açısından oluşmuştur. Kayma vektörünün doğru olarak belirlenmesi önemlidir. Aksi takdirde referans görüntü ve anlık görüntü tam olarak hizalanmaz ve görüntü çıkarma yöntemi sonrası elde edilen sonuç yabancı nesnenin apaçık olarak belirlenmesi için yeterli olmaz. Kayma vektörünü hesaplamak için Şekil 3.23’de ifade edilen yöntem uygulanır.



**Şekil 3.23** Kayma vektörünün görsel ifadesi.

Şekilde belirtilen, R: Referans görüntü çerçevesi pikselini, A: Anlık görüntü çerçevesi pikselini,  $d_{ort}$ : Ortalama kayma miktarını,  $\theta_{ort}$ : Ortalama kayma açısını,  $R_x, R_y$ : Referans görüntü pikseline ait x ve y konumlarını,  $A_x, A_y$ : Anlık görüntü pikseline ait x ve y konumlarını ifade etmektedir.

Oluşturulan koordinat sistemi I, II, III, IV olmak üzere 4 bölgeye ayrılmıştır. Örnek olarak verilen  $\overrightarrow{RA}$  vektörü II. bölgede oluşturulmuştur. Anlık görüntüde eşlenik olarak bulunan her bir pikselin referans görüntüye hizalanabilmesi için kayma vektörünün x ve y eksenlerine olan izdüşümlerinin hesaplanmıştır. Bu izdüşüm değerleri Denklem 3.10 – 3.19’da ifade edildiği şekilde hesaplanmıştır.

$$X = d_{ort} \cdot \cos(\theta_{ort}) \quad (3.10)$$

$$Y = d_{ort} \cdot \sin(\theta_{ort}) \quad (3.11)$$

Kayma vektörünü bulmak için yön bilgisini de bulmak gerekmektedir. Referans görüntü piksellerinin koordinat sisteminin merkezi olarak varsayıldığında anlık görüntü piksellerinin yeni koordinatlarının hesabı bulunduğu bölgeye göre değişecektir. Hizalanmış anlık görüntünün her bir pikseli için yeni x ve y koordinatları farklı bölgeler için aşağıdaki şekilde hesaplanmıştır.

I. bölge:

$$A_x(\text{yeni}): A_x + X \quad (3.12)$$

$$A_y(\text{yeni}): A_y + Y \quad (3.13)$$

II. bölge:

$$A_x(\text{yeni}): A_x - X \quad (3.14)$$

$$A_y(\text{yeni}): A_y + Y \quad (3.15)$$

III. bölge:

$$A_x(\text{yeni}): A_x - X \quad (3.16)$$

$$A_y(\text{yeni}): A_y - Y \quad (3.17)$$

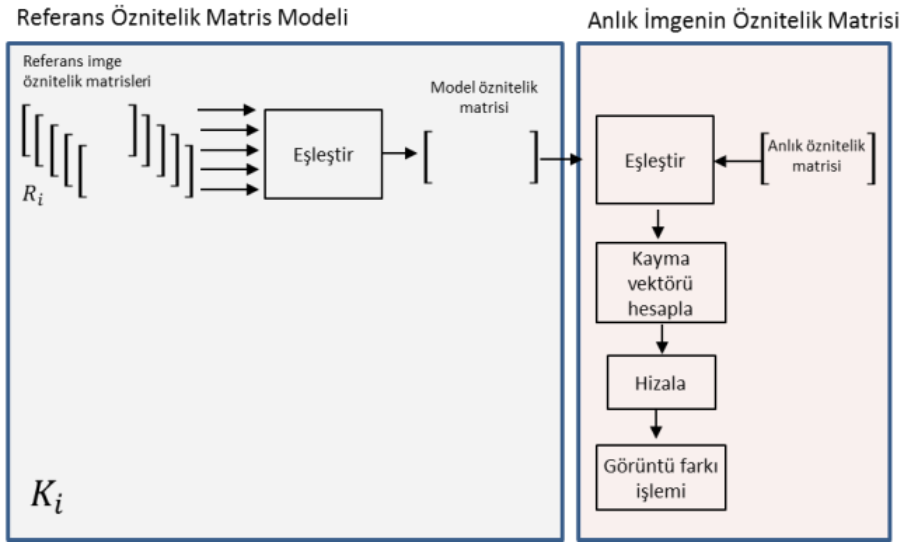
IV. bölge:

$$A_x(\text{yeni}): A_x + X \quad (3.18)$$

$$A_y(\text{yeni}): A_y - Y \quad (3.19)$$

Anlık görüntüde bulunan tüm pikseller referans görüntüdeki eşlenikleri ile hizalandıktan sonra çerçeve farkı yöntemi uygulanmıştır. Bu yöntem ile birbirine hizalanan anlık görüntü ve referans görüntü çerçevelerdeki tüm piksel değerleri aritmetik olarak birbirlerinden çıkartılır. Değişime uğramayan bölümlerde pikseller kendileriyle eş değer piksellerden çıkartıldıkları için değerleri 0'a yaklaşacaktır. Aksine, değişime uğrayan yani yalnız arka plan görüntüsüne yabancı nesnelere dahil olduğu bölgelerde piksel değerleri değişeceğinden çıkartma işlemi sonuçları 0'dan uzaklaşacaktır. Böylece farkı oluşturan nesnelere silüeti yaklaşık olarak ortaya çıkacaktır. Silüetlerin belirginleştirilmesi için eşikleme ve çeşitli morfolojik işlemlerin uygulanması gerekir. Bu çalışmada eşikleme için Otsu yöntemi ile morfolojik olarak da açma ve kapama işlemleri uygulanmıştır. Elde edilen sonuç görüntüde belirgin olarak görünen silüetlerin görüntüye sonradan dahil olan hareketli ya da hareketsiz nesne oldukları varsayılmıştır.

Rota üzerindeki  $K_i$ 'nci koordinat noktasındaki hareketli nesne tespiti için gerçekleştirilen gerçek zamanlı işlemler Şekil 3.24'de ifade edilmiştir.



**Şekil 3.24**  $K_i$  'nci koordinat noktası için hafızadaki model ile gerçek zamanlı bilgilerin karşılaştırılması.

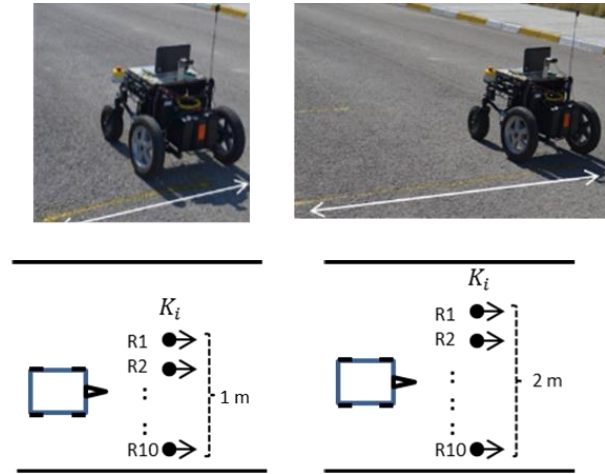
Şekilde sistem hafızasında oluşturulan öznitelik matris modeli (solda) ile gerçek zamanlı hareketli nesne tespit modeli (sağda) görülmektedir. Öncelikle kameranın hareketinden doğan gürültünün dengelenmesi yani imge çerçevelerinin referans imgeye geometrik olarak hizalanması gerekir. Bu işlem geometrik dönüşüm yapılarak gerçekleştirilir.

### 3.9.3 Yöntem 3: GPS Destekli Gradyan Tabanlı Gri Seviye Değişimi Modellemesi

Bu yöntemin uygulama adımları diğer yöntemlerle benzerlik göstermektedir. Yöntemin farkı; modelin oluşturulma yöntemidir. Bu yöntemde de İKA'nın rotaları üzerinde bulunan hareketli nesnelere bilgisayar görme sistemleri kullanarak tespit edebilmesi için GPS kullanılmıştır. Gerçek zamanlı çalışma öncesinde rota üzerinde belirlenen tüm koordinatlardaki yatay konumlarda varsayılan arka planı temsil eden bir model oluşturulmuştur. Bu model görüntü matrislerinden çok daha küçük boyutlarda olan ve görüntünün iki boyutlu kısmi türevinin alınmasıyla elde edilen gradyan bileşenlerinden oluşan bir matristir. Diğer yöntemlerde olduğu gibi bu model matris de elde edildiği koordinat bilgisi ile ilişkilendirilerek veri tabanına kaydedilir. Bu modeller referans model olarak adlandırılmıştır. Mobil aracın gerçek zamanlı hareketi sırasında belirlenen koordinatlara (referans modellerin elde edildiği koordinatlar) geldiği anda yakalanan anlık görüntüler aynı yöntem ile gradyanları alınarak modellenir ve koordinatla ilişkilendirilen referans model ile karşılaştırılır. İşlem sonucunda referans alınan model ile anlık model

arasında belirlenen eşik değerden daha uzak olan pikseller sahneye sonradan dahil olan dinamik nesnelere ait kabul edilir.

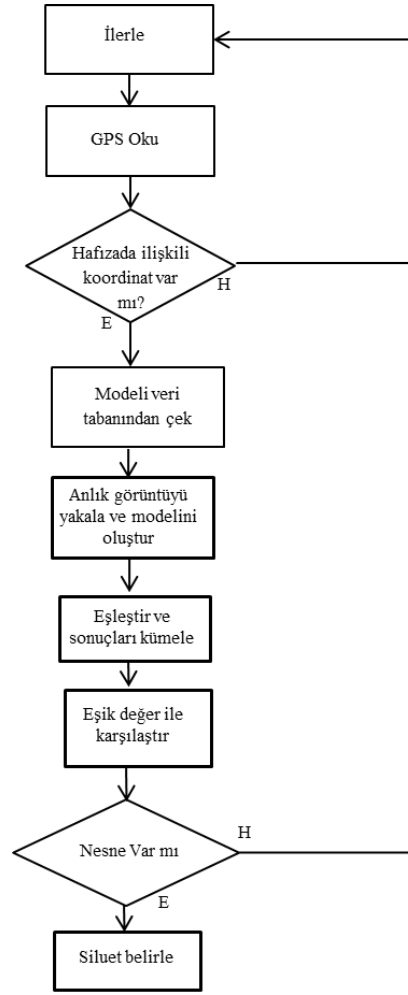
Bu yöntemde diğer yöntemlerden farklı olarak 10 adet referans imge kullanılmıştır. Her bir  $K_i$  koordinatı için referans modelin oluşturulmasında kullanılacak olan R1, R2, . . . R10 imgeleri Şekil 3.25’de gösterildiği gibi 1 m ve 2 m yatay eksen aralıklarında ve farklı görüş açılarından alınmıştır



Şekil 3.25 Yöntem-3’de kullanılan mobil araç yatay geçiş aralıkları.

Referans alınan bu imgelerin ortalamaları alınarak bulunulan koordinattaki varsayılan arka planı temsil eden tek bir referans görüntü elde edilir. Bu görüntü gri seviye görüntüye dönüştürülür ve karesel bloklara ayrılarak yatay ve dikey eksenler boyunca gri seviye değişim modelleri çıkartılır. Yöntem-3’de hareketli nesnelere tespiti için oluşturulan sistemin genel akış şeması Şekil 3.26’da verilmiştir.

Algoritmaya göre araç belirlenen rotada görüntü olarak ilerlemeye başlar ve eş zamanlı olarak GPS konumunu okur. Okuduğu konum referans nokta olarak belirlenmiş bir konum ise bu konumdan anlık bir görüntü çerçevesini alır, Yöntem-3’e göre modeller ve hafızadaki bu konuma ait referans modeli ile karşılaştırır. Alınan sonuçlara göre anlık görüntüde hareketli ya da hareketsiz bir nesne bulunup bulunmadığı tespit edilir. “Nesne yok” kararı verilirse ilerleme devam eder. “Nesne var” kararı verilirse nesne tanıma işlemi için silüet belirleme adımlarına geçilir.



Şekil 3.26 Yöntem-3’de kullanılan mobil araç ilerleyişi genel akış şeması.

### 3.9.3.1 Varsayılan Arka Plan Modelinin Oluşturulması

Gerçek zamanlı uygulama öncesi gerçekleştirilmesi gereken ön hazırlıklar 3 adımdan oluşmuştur. Bunlar;

#### 1. Adım

Rota boyunca GPS koordinatlarının her belirgin değişimine karşılık gelen konumlardan yani  $K_i$  koordinatlarından yatay geçiş aralıkları boyumca farklı görüş açılarıyla onar adet varsayılan arka plan imgesi fotoğraflanmıştır.

## 2. Adım

Bu adımda  $K_i$  koordinatlarından elde edilen referans imgelerin  $R_i$  olarak adlandırılan ortalamaları oluşturulmaktadır. Diğer yöntemlerde olduğu gibi bu yöntemde de rotanın sadece odaklanılan kısmında işlem yapabilmek için referans ve anlık imgelerin tamamı kullanılmamış ve 3 parçaya bölünerek değerlendirilmiştir.




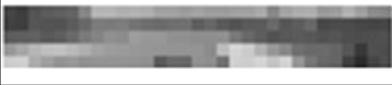

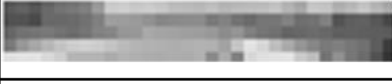

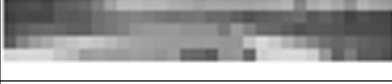

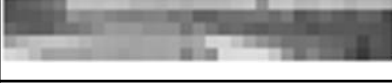



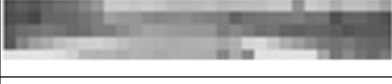

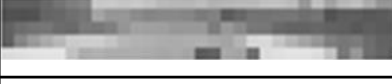

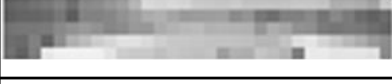

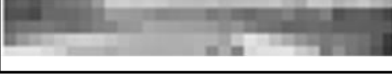

Odaklanılan kısım Şekil 3.22'de  $B_2$  olarak gösterilen kısımdır. Bu kısmın boyutu 4928x636 pikseldir. Yöntem-3'de de referans imgelerin tamamının kullanılmayarak sadece belirli bir bölümünün kullanılması bilgisayarlardaki hesaplama maliyetini azaltmaktadır.

Yöntem-3'de öncelikle tüm referans imgeler 160x160 boyutlarında kare bloklara bölünerek mozaik imge elde edilmiş ve her bir bloğun gri seviye ortalaması o bloğun parlaklık değeri olarak kabul edilmiştir. Bu şekilde her bir bloğun çevresindeki pikselleri en doğru şekilde temsil edebilmesi, küçük gürültülerin yok edilmesi ve az miktardaki yerel keskin ton değişimlerinin etkisinin azalması sağlanmıştır.

Tüm referans imgeler için bu işlemin tekrar edilmesinin ardından mozaik imgelerin gri seviye değerlerinin Denklem 3.20'de belirtildiği şekilde aritmetik ortalaması alınarak  $K_i$  koordinatında varsayılan arka planı temsil eden tek bir referans imge  $R_i$  elde edilmiştir.

$$R_i(x, y) = \sum_{x=1}^n \sum_{y=1}^k \frac{1}{N} (R1(x, y) + R2(x, y) + \dots + RN(x, y)) \quad (3.20)$$

Burada  $x, y$  değişkenleri bloklara ayrılmış referans imge matrislerinin satır ve sütunlarını göstermekte;  $n, k$  değişkenleri ise referans imge boyutlarının 160x160 boyutundaki karesel bloklara bölünmesiyle elde edilen sınır değerlerini ifade etmekte ve  $N$  referans imge sayısını göstermektedir. Herhangi bir  $K_i$  koordinatından alınmış referans imgelerin ve hesaplanan ortalama referans imgenin bir örneği Şekil 3.27'de verilmiştir.

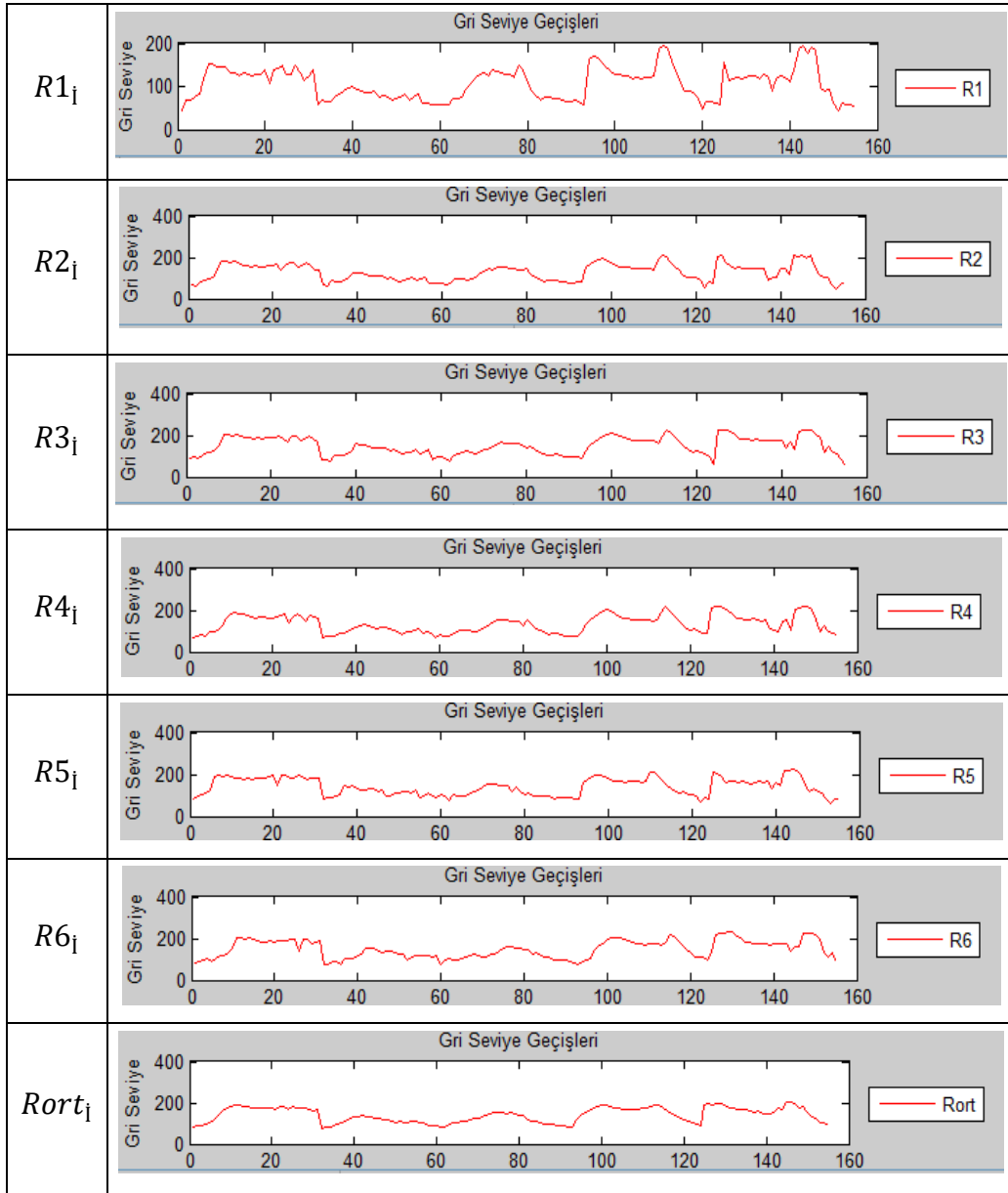
$R1_i$		
$R2_i$		
$R3_i$		
$R4_i$		
$R5_i$		
$R6_i$		
$R7_i$		
$R8_i$		
$R9_i$		
$R10_i$		
$Rort_i$		

Şekil 3.27 Referans imgelerin orijinal ve mozaik durumları ve ortalama imge örneği.

Karesel bloklara ayrılmış olan referans imgelerin parlaklık değerlerindeki değişim grafik ortamda izlendiğinde birbirine çok yakın değişimlerin olduğu gözlenebilmektedir. Yani referans olarak alınan varsayılan arka plan imgelerinin parlaklık değişimlerinin karakterleri birbirine benzer yapıdadır.

Bu değerlerin ortalamaları alındığında ise keskin değer geçişlerinin yumuşadığı ve karşılaştırma işlemi için daha net değerlerin elde edildiği görülmektedir. Herhangi bir  $K_i$  koordinatından elde edilen referans imgelerin bir kısmı ve söz konusu koordinattan elde edilen ortalama referans görüntünün grafiksel ifadeleri Şekil 3.28’de verilmiştir.

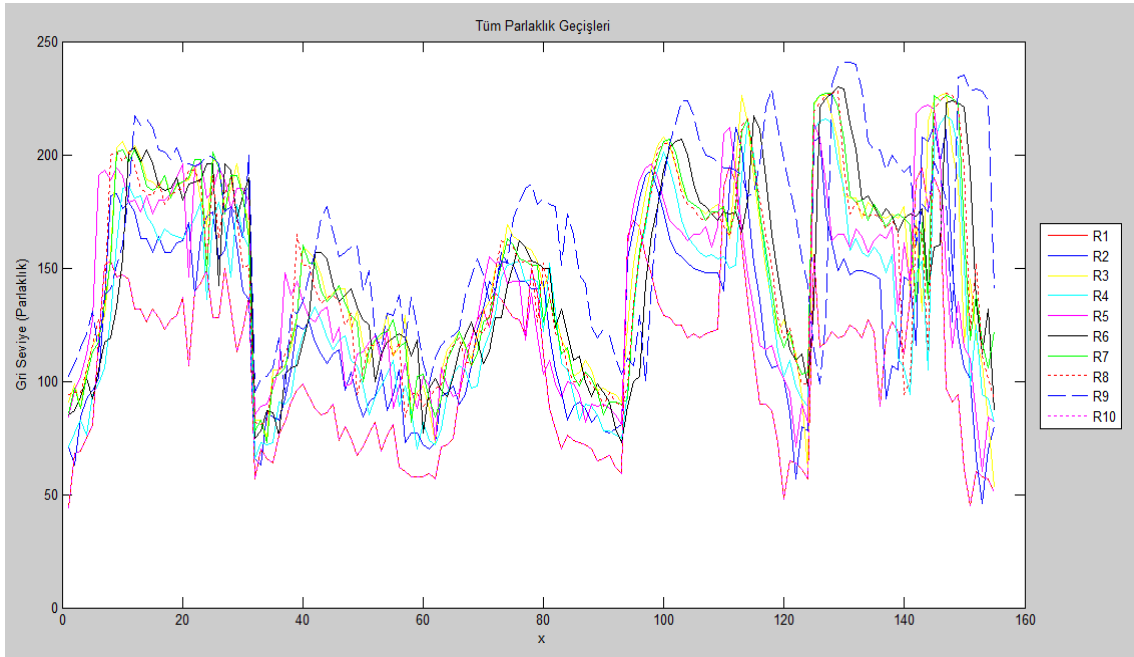




**Şekil 3.28** Bir kısım referans görüntülerin ve ortalama görüntünün parlaklık geçişlerinin grafiksel ifadesi.

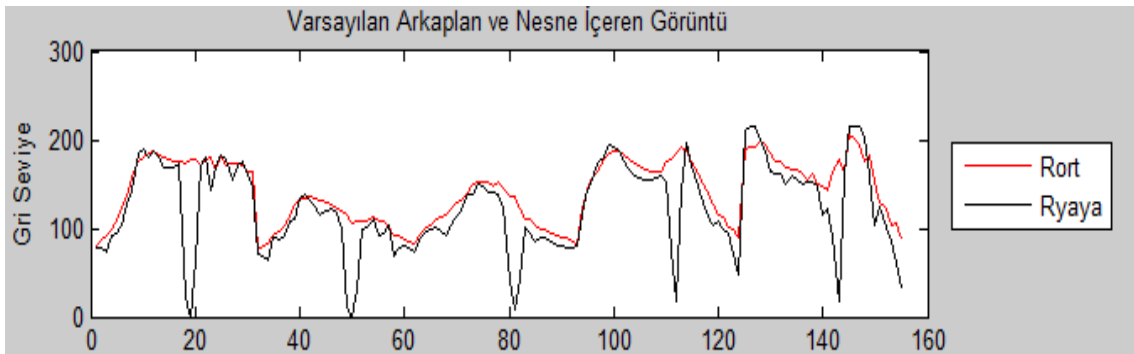
Referans imgelerin boyutları 4928x636'dır. Bu imgeler 160x160 boyutlarında karesel bloklara bölündüklerinde 5x31 boyutlarında mozaik görüntüyü ifade eden matrisler oluşmaktadır. Bu matrisler sütun matrisler haline dönüştürülüp her elemanının gri seviye parlaklık değerine göre grafikleri çizildiklerinde Şekil 3.28'de verilen grafikler oluşmaktadır. Söz konusu grafiklerin yatay eksenleri sütun matrisin eleman numarasını gösterirken, dikey eksen ise matris elemanının gri seviye parlaklık değerini göstermektedir. Tüm referans görüntüler ile ortalama görüntünün parlaklık geçişlerini bir

arada gösteren grafik Şekil 3.29’da verilmiştir. Bu grafikten referans imgelerin ve ortalama referans imgenin birbirine yakın karakterde olduğu anlaşılmaktadır.



Şekil 3.29 Tüm referans ve ortalama görüntülere ait grafik.

Herhangi bir  $K_i$  koordinatından elde edilen  $R_i$  ’nin grafiksel görünümü ile aynı koordinatta nesne içeren bir görüntünün karşılaştırması Şekil 3.30’da yapılmıştır.



Şekil 3.30 Yalın arka plan modeli ile nesne içeren görüntünün grafiksel karşılaştırması.

Model matristeki her bir satır çizgisi üzerinde yabancı nesnenin oluşturmuş olduğu fark grafik üzerinde açıkça gözlemlenebilmektedir. Bu fark değer iki görüntünün birbirinden ayırt edilebileceğini ifade etmektedir.

### 3. Adım

Bu adımdaki amaç  $R_i$ 'nin kare blokları arasında yatay ve dikey eksenler boyunca meydana gelen gri seviye değişimlerinin modellenmesidir.  $R_i$  ortalama referans imgesi  $x$ : satır ve  $y$ : sütun olmak üzere iki boyutlu bir  $f(x,y)$  fonksiyonu olarak kabul edilebilir. İki boyutlu fonksiyonların gradyanları  $x, y$  noktalarındaki en yüksek değişimin gerçekleştiği yönü işaret ederler ve bir vektör olarak ifade edilirler. Bu vektör Denklem 3.21'de ifade edilmiştir.

$$\text{gradyan}(f) = \nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (3.21)$$

Bu denklemde  $g_x$  ve  $(\partial f / \partial x)$  ifadeleri  $f(x,y)$  görüntü fonksiyonunun  $x$  değişkenine göre kısmi türevini,  $g_y$  ve  $(\partial f / \partial y)$  ifadeleri ise  $f(x,y)$  görüntü fonksiyonunu  $y$  değişkenine göre kısmi türevini göstermektedir. Gradyan vektörü 2 adet değerden oluşmuştur. Bu değerler; değişimin büyüklüğünü ifade eden genlik ve değişimin yönünü ifade eden açı değerleridir. Genlik bilgisi Denklem 3.22'de belirtilmiş olup,

$$\text{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (3.22)$$

açı bilgisi Denklem 3.23'de ifade edilmiştir.

$$\theta(x, y) = \tan^{-1} \begin{bmatrix} g_y \\ g_x \end{bmatrix} \quad (3.23)$$

Böylece her bir  $R_i$  için bir adet genlik ve bir adet açı bilgisi içeren iki adet model matris elde edilmiş olur. Bu model matrisler referans model  $RM_i$  olarak adlandırılır ve  $K_i$  koordinatındaki varsayılan arka planı temsil eden modelin bileşenleri olarak hafızaya kaydedilirler.

Bu şekilde yazılımın çalışacağı bilgisayar sistemlerinde oldukça büyük boyutlardaki iki farklı imge çerçeve matrisinin işlenmesi yerine arka planı temsil edecek çok daha küçük boyutlarda genlik ve açı matrisleri kullanılmıştır. Çalışmada kullanılan genlik ve açı matrisleri 4x30 boyutlu matrislerdir. Böylece bilgisayar CPU ve hafıza kapasitelerinde büyük ölçüde tasarruf sağlandığı gibi işlem hızında da büyük artışlar sağlanmıştır. Her koordinat için genlik ve açı matrislerinin oluşturulmasıyla ön hazırlık çalışması tamamlanmıştır. Bir sonraki adım olan gerçek zamanlı hareket işlemlerine geçilmiştir.

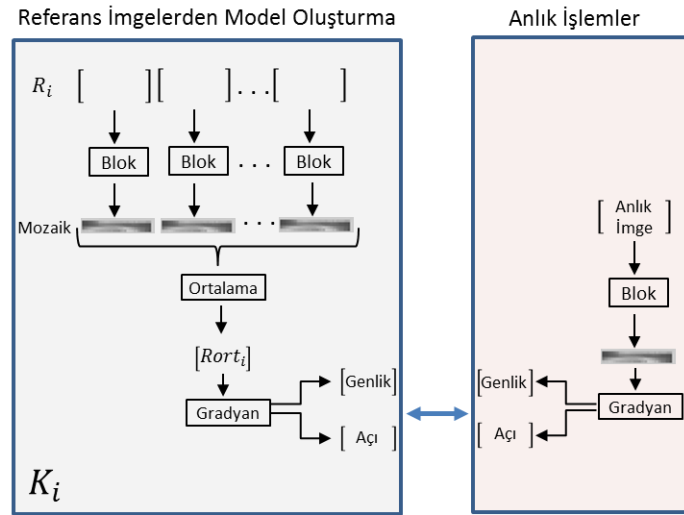
### 3.9.3.2 Gerçek Zamanlı Hareketli Nesne Tespiti

Mobil araç daha önce modelin oluşturulduğu rota boyunca ilerlerken üzerinde bulunan GPS alıcı sistem sayesinde 500 ms. 'de bir koordinat bilgilerini almaktadır. Zamansal bir döngü içerisinde bu bilgi hafızadaki koordinat bilgileri ile karşılaştırılır. Anlık alınan gps koordinatı ile hafızadaki bir koordinat uyuştuğu anda aracın üzerinde bulunan kamera ile arka planın anlık görüntüsü yakalanır. Anlık imgenin hafızadaki model matrisler ile karşılaştırılabilmesi için aynı yöntemlerle modellenmesi gerekir. Bu nedenle anlık imge, referans imgelerde olduğu gibi 160x160 boyutlarında bloklara ayrılır ve mozaik imge elde edilir. Mozaik imge üzerinde Denklem 3.21, 3.22 ve 3.23'de belirtilen işlemler ile gradyan alma işlemi gerçekleştirilerek genlik ve açı matrisleri oluşturulur. Böylece anlık imgeye ait model oluşturulmuş olur.

Modellerin oluşturulmasının ardından arka planda yabancı bir nesne olup olmadığının belirlenebilmesi için iki model arasında karşılaştırma yapılması gerekir. Bu karşılaştırma işlemi hafızadaki genlik ve açı bilgilerini içeren model matrisler ile anlık imgeye ait genlik ve açı bilgilerini içeren matris elemanları arasındaki Öklid uzaklığının hesaplanmasıyla gerçekleştirilir. Bir döngü içerisinde gerçekleştirilen hesaplama sonucunda *Uzaklık* matrisi elde edilir. Bu matrisin oluşturulması işlemlerinde Şekil 3.19'da ayrıntıları açıklanan algoritma kullanılmıştır.

*Uzaklık* matrisi elemanlarından belirlenmiş bir eşik değerinin üstünde bir değere sahip olanlar, bloklar arası gri seviye değişiminin fazlalığı nedeniyle arka plandaki yabancı bir nesneye ait olarak kabul edilir. Blokların yabancı bir nesneyi işaret edebilmeleri için bir arada seçilebilir şekilde kümelenmiş olmaları ve *Uzaklık* matrisi içinde ayırt

edilebilmesi gerekir. Bu ayırt etme işlemi bağlı bileşen işaretleme yöntemi (connected component labelling-CCL) (Heijden 1996) ile yapılmıştır. Bu nedenle değişimin meydana geldiği blokların anlık imge üzerindeki konumları hesaplanır ve yabancı nesnenin imge üzerindeki konumu belirlenmiş olur. Bu aşamadan sonra yabancı nesnenin silüetinin belirlenmesi işlemine geçilmektedir. Mobil araç üzerindeki kamera yabancı nesneyi tespit ettikten sonra hareketine son vererek silüet belirleme maksatlı video akışını alır ve ardışık imge çerçevelerine çerçeve farkı yöntemini uygular. Bu yöntem ile ardışık iki çerçevedeki tüm piksel değerleri aritmetik olarak birbirlerinden çıkartılır. Değişime uğramayan bölümlerde pikseller kendileriyle eş değer piksellerden çıkartıldıkları için değerleri 0'a yaklaşacaktır. Değişime uğrayan bölgelerde piksel değerleri değişeceğinden çıkartma işlemi sonuçları 0'dan uzaklaşacaktır. Böylece farkı oluşturan nesnelere silüeti yaklaşık olarak ortaya çıkacaktır. Silüetlerin belirginleştirilmesi için eşikleme ve çeşitli morfolojik işlemler uygulanmıştır. Bu çalışmada eşikleme için Otsu yöntemi ile morfolojik olarak da açma ve kapama işlemleri uygulanmıştır. Elde edilen sonuç görüntüde belirgin olarak gözlenebilen silüetlerin görüntüye sonradan dâhil olan hareketli ya da hareketsiz nesne oldukları belirlenmiştir. Rota üzerindeki  $K_i$ 'nci koordinat noktasındaki hareketli nesne tespiti için gerçekleştirilen gerçek zamanlı işlemler Şekil 3.30'da ifade edilmiştir. Şekilde sistem hafızasında her bir  $K_i$  koordinatı için hazırlanan model (solda) ile gerçek zamanlı işlemler (sağda) görülmektedir.



Şekil 3.31  $K_i$  koordinat noktası için hafızadaki bilgiler ile gerçek zamanlı bilgilerin karşılaştırılması.

Bu işlemler dizisindeki amaç; öncelikle kameranın hareketinden kaynaklanan gürültünün kaydedilmiş arka plan modelleri ile dengelenerek hareketli nesnelere tespit edilmesidir. Nesne silüetlerinin tanınması için gerekli işlemler tespit işleminden sonra gerçekleştirilmektedir.

### **3.9.4 Yöntem 4: GPS Destekli Özelleştirilmiş Arka Plan Modellemesi**

Bu yöntem Bölüm 3.9'da belirtildiği gibi iki adımdan oluşmuş olup bunlar; gerçek zamanlı işlemler öncesi bir defa yapılan hazırlık çalışması ve gerçek zamanlı işlemlerdir.

Yöntem-4'de oluşturulan sistemin genel akış şeması Şekil 3.26'da verilen Yöntem-3 akış şeması ile aynıdır. Her iki yöntemde de benzer işlem basamakları gerçekleştirilmektedir ancak modelleme teknikleri farklıdır.

#### **3.9.4.1 Varsayılan Arka Plan Modelinin Oluşturulması**

Gerçek zamanlı çalışma öncesinde gerçekleştirilen ön hazırlık çalışmasında; belirlenen rota üzerinde referans imgelerin fotoğraflanacağı referans noktaların coğrafi konum koordinatları belirlenmiştir. Bu koordinatlar  $K_i$  olarak ifade edilmiştir. Bu yöntemde Yöntem-3 ile aynı olacak şekilde 10 adet referans imge kullanılmıştır. Her bir  $K_i$  koordinatından varsayılan arka planı içeren  $R_1, R_2, \dots, R_{10}$  referans imgeleri Şekil 3.25'de gösterildiği gibi farklı geçiş aralıklarında ve farklı görüş açılarından fotoğraflanır. Asıl odaklanılan hedef her bir  $K_i$  koordinatı için varsayılan arka planı temsil eden minimum boyutlarda bir model oluşturmaktır. Modeli oluşturmak için öncelikle tüm referans imgeler 160x160 boyutlarında kare bloklara bölünerek mozaik imge elde edilmiş ve her bir bloğun gri seviye ortalaması o bloğun parlaklık değeri olarak kabul edilmiştir. Bu şekilde her bir bloğun çevresindeki pikselleri en doğru şekilde temsil edebilmesi, küçük gürültülerin yok edilmesi ve az miktardaki yerel keskin ton değişimlerinin etkisinin azalması sağlanmıştır. Tüm referans imgeler için bu işlemin tekrar edilmesinin ardından Denklem 3.20'da belirtildiği şekilde aritmetik ortalama alınarak  $K_i$  koordinatında varsayılan arka planı temsil eden tek bir referans imge  $R_i$  elde edilmiştir. Yöntem-4'de referans model olarak  $R_i$  kullanılmıştır. Böylece hesaplamalarda kullanılacak olan model matrisin boyutu gerçek imgeye göre 20.000 kat düşürülmüştür.  $R_i$  her bir  $K_i$  koordinatı

için tekildir ve kendi koordinatıyla aynı indisi alacak şekilde adlandırılmıştır. Bu yöntemde; gradyan bileşenleri kullanılmayıp sadece blokların seviye ortalamaları kullanılmıştır.  $K_i$  koordinatından alınmış referans imgelerin ve ortalama referans imgenin bir örneği Şekil 3.27’de verilmiştir.

Diğer yöntemlerde olduğu gibi bu yöntemde de rotanın sadece odaklanılan kısmında işlem yapabilmek için referans ve anlık imgelerin tamamı kullanılmamış ve 3 parçaya bölünerek değerlendirilmiştir. Odaklanılan kısım Şekil 3.22’de  $B_2$  olarak gösterilen kısımdır. Bu kısmın boyutu 4928x636 pikseldir. İmgenin tamamının kullanılmayarak sadece belirli bir bölümünün kullanılması bilgisayarlardaki işlem yükünü de azaltmaktadır.

#### **3.9.4.2 Gerçek Zamanlı Hareketli Nesne Tespiti**

Gerçek zamanlı işlemlerin başlangıcında mobil araç kontrol sisteminin hafızasında rotası boyunca belirlenmiş her bir  $K_i$  koordinatı için varsayılan arka planı temsil eden bir model matris oluşturulmuş durumdadır. Mobil araç belirlenen rota boyunca ilerlerken üzerinde bulunan GPS alıcı sistem sayesinde 500 ms. ‘de bir koordinat bilgilerini almaktadır. Zamansal bir döngü içerisinde bu bilgi hafızadaki koordinat bilgileri ile karşılaştırılır. Uyuşma sağlandığı anda aracın üzerinde bulunan kamera ile arka planın anlık görüntüleri yakalanır. Anlık imgenin model matrisler ile karşılaştırılabilmesi için aynı yöntemlerle modellenmesi gerekir. Bu nedenle anlık imge, referans imgelerde olduğu gibi 160x160 boyutlarında bloklara ayrılır ve her bir bloğa kendi ortalama gri seviye değeri verilerek mozaik imge elde edilir. Böylece anlık imgeye ait model oluşturulmuş olur.

Modellerin oluşturulmasının ardından arka planda yabancı bir nesne olup olmadığının belirlenebilmesi için iki model arasında karşılaştırma yapılması gerekir. Bu karşılaştırma işlemi model matris ile anlık imgeye ait matris elemanları arasındaki Öklid uzaklığının hesaplanmasıyla gerçekleştirilir. Bir döngü içerisinde gerçekleştirilen hesaplama sonucunda *Uzaklık* matrisi elde edilir. Bu matrisin oluşturulması işlemlerinde Şekil 3.19’da ayrıntıları açıklanan algoritma kullanılmıştır.

*Uzaklık* matrisi elemanlarından belirlenmiş bir eşik değerinin üstünde bir değere sahip olanlar bloklar arası gri seviye değişiminin fazlalığı nedeniyle arka plandaki yabancı bir

nesneye ait olarak kabul edilir. Söz konusu eşikleme işlemi Denklem 3.24’de ifade edilmiştir.

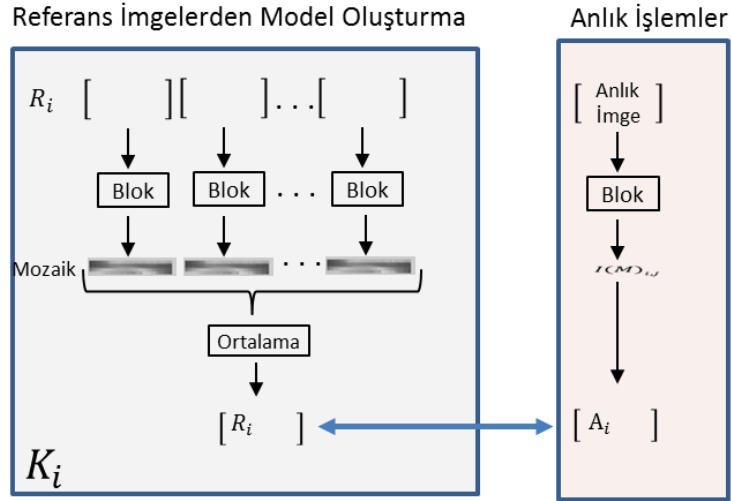
$$E_{i,j} = \begin{cases} 1, & \text{Uzaklık}_{i,j} > \tau \\ 0, & \text{diğer} \end{cases} \quad (3.24)$$

Burada  $E_{i,j}$  eşiklenmiş uzaklık matrisini,  $\tau$  ise eşik değeri ifade etmektedir. Blokların yabancı bir nesneyi işaret edebilmeleri için bir arada anlamlı bir şekilde kümelenmiş olmaları gereklidir. Sağlıklı bir tespit için bu kümelerin  $E_{i,j}$  içinde ayırt edilebilmesi gerekir. Bu ayırt etme işlemi CCL yöntemi ile yapılmıştır. Bu şekilde değişimin meydana geldiği bloklar anlık imge üzerindeki kümelenip işaretlenir böylece yabancı nesnenin imge üzerindeki konumu belirlenmiş olur.

Bu aşamadan sonra yabancı nesnenin silüetinin belirlenmesi işlemine geçilmektedir. Mobil araç yabancı nesneyi tespit ettikten sonra hareketine son vererek silüet belirleme maksatlı video akışını alır ve ardışık imge çerçevelerine çerçeve farkı yöntemini uygulanır. Bu yöntem ile ardışık iki çerçevedeki tüm piksel değerleri aritmetik olarak birbirlerinden çıkartılır. Değişime uğramayan bölümlerde pikseller kendileriyle eş değer piksellerden çıkartıldıkları için değerleri 0’a yaklaşacaktır. Aksine, değişime uğrayan bölgelerde piksel değerleri değişeceğinden çıkartma işlemi sonuçları 0’dan uzaklaşacaktır. Böylece farkı oluşturan nesnelere silüeti yaklaşık olarak ortaya çıkacaktır. Silüetlerin belirginleştirilmesi için eşikleme ve çeşitli morfolojik işlemlerin uygulanması gerekir. Bu çalışmada eşikleme için Otsu yöntemi ile morfolojik olarak da açma ve kapama işlemleri uygulanmıştır. Elde edilen sonuç görüntüde belirgin olarak görünen silüetlerin görüntüye sonradan dahil olan hareketli ya da hareketsiz nesne oldukları belirlenmiştir.

Rota üzerindeki  $K_i$ ’nci koordinat noktasındaki hareketli nesne tespiti için gerçekleştirilen gerçek zamanlı işlemler Şekil 3.33’de ifade edilmiştir. Şekilde sistem hafızasında her bir  $K_i$  koordinatı için hazırlanan model (solda) ile gerçek zamanlı işlemler (sağda) görülmektedir.  $R_i$  referans model matrisini,  $A_i$  anlık imgenin modellenmiş matrisini ifade etmektedir.

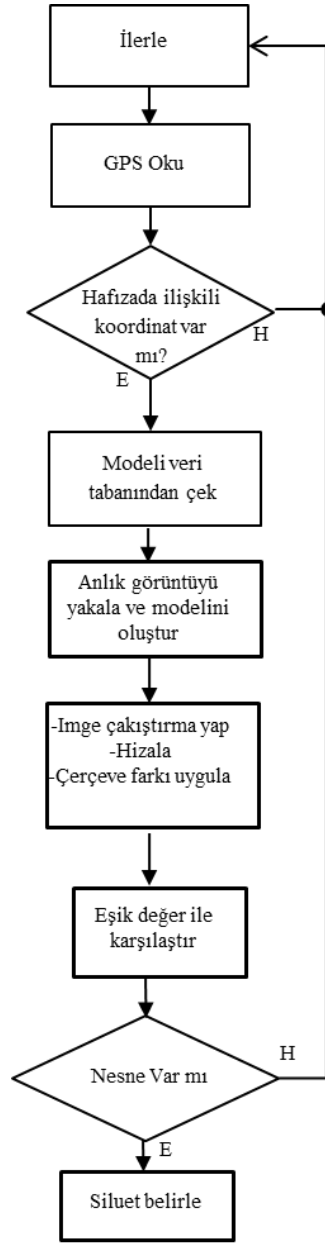




Şekil 3.32  $K_i$  koordinat noktası için hafızadaki referans model ile gerçek zamanlı modelin karşılaştırılması.

### 3.9.5 Yöntem 5: Özelleştirilmiş İmge Çakıştırma

Bu yöntemde Yöntem-2’de yapılan çalışmaya benzer olarak imge çakıştırma metodu kullanılmıştır. İmge çakıştırma işlemi referans imgeler ile anlık imgeler arasında yapılmaktadır. Ancak imge çakıştırma için kullanılan teknik farklıdır. Diğer yöntemlerde olduğu gibi Yöntem-5 de iki aşamadan oluşmuştur ve GPS yardımı kullanılmıştır. Yöntemde kullanılan algoritmanın akış şeması Şekil 3.34’de verilmiştir.



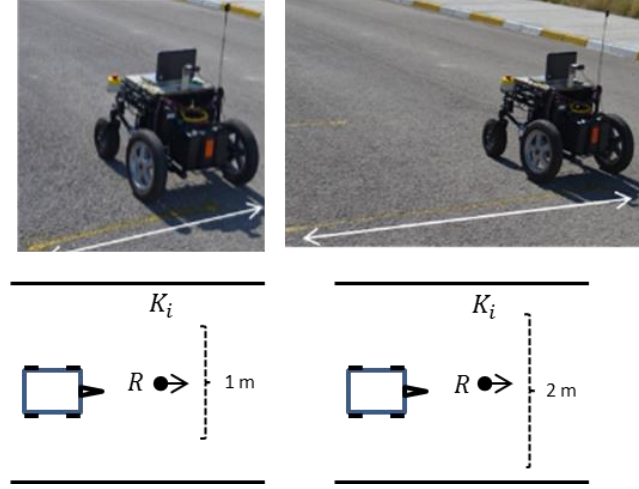
Şekil 3.33 Yöntem-5 mobil araç ilerleyişinde kullanılan algoritmanın genel akış şeması.

### 3.9.5.1 Varsayılan arka plan modelinin oluşturulması

#### 1. Adım

Üniversite kampüsü içerisinde belirlenen ve Şekil 3.17’de verilen rota boyunca her  $K_i$  koordinatında yalnız arka plan imgeleri fotoğraflanmıştır. Yöntem-5’de bu imgeler referans imgeler olarak adlandırılmışlar ve R ile gösterilmişlerdir. Bu yöntemde diğer yöntemlerden farklı olarak her bir  $K_i$  için sadece bir adet referans imge kullanılmıştır.

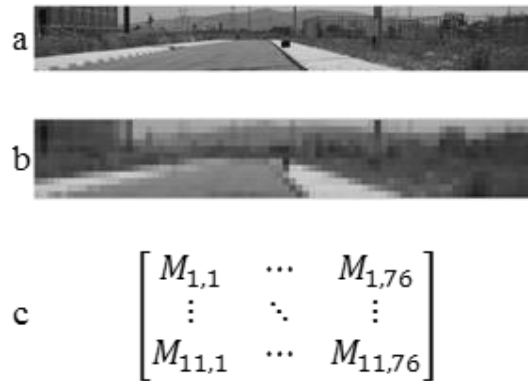
İmgeler her bir koordinat için Şekil 3.35’de verilen yatay geçiş aralıklarında aracın geçebileceği bir noktadan fotoğraflanarak elde edilmiştir.



Şekil 3.34 Yatay geçiş aralıkları.

## 2. Adım

Referans ve anlık imgeler üzerinde odaklanılan kısım Şekil 3.22’de  $B_2$  olarak gösterilen kısımdır. İlk işlem olarak tüm  $R_i$  imgeleri 64x52 boyutlarında dikdörtgen bloklara bölünerek mozaik imge elde edilmiş ve her bir bloğun gri seviye ortalaması o bloğun parlaklık değeri olarak kabul edilmiştir. Herhangi bir  $K_i$  koordinatından alınmış referans imge, bu imgeye ait bloklara ayrılmış mozaik görüntü ve mozaik görüntü matrisinin bir örneği Şekil 3.36’da verilmiştir.



Şekil 3.35 (a) Referans imge (b) bloklanmış mozaik görüntü (c) mozaik görüntüye ait matris.

636x4928 boyutlarında olan orijinal imge bloklara ayrıldığında 11x76 boyutlarında bir mozaik imge elde edilmektedir. Bu yöntemde Yöntem-4'den farklı olarak referans imgeler daha küçük boyutlu bloklara ayrılmıştır. Bu şekilde imge daha fazla detaylandırılmış ve imge çakıştırma için daha fazla öznelik elde edilmiştir.

### 3. Adım

Bu adımda referans model  $R_i$  elde edilmektedir. İmge çakıştırma yönteminde referans ve anlık modeller arasında karşılaştırma yapılarak kayma vektörünün hesaplanması gerekmektedir. Bu işlem için her iki modelde de ortak özneliklerin çıkarılması gerekmektedir.

Literatürde bulunan SURF, Harris ve FAST gibi öznelik çıkartma yöntemleri ayrıntılı ve karmaşık algoritmaları nedeniyle hızlı sonuçlar üretememektedir. Bu teknikler yerine Yöntem-5'de mozaik görüntüdeki bloklar arasındaki gri seviye değişimlerini modelleyecek dört adet öznelik oluşturulmuştur. Bu öznelikler;

İki boyutlu bir görüntü  $x$ : satır ve  $y$ : sütun olmak üzere  $f(x,y)$  fonksiyonu olarak ifade edildiğinde;

- Her bir bloğun yatay ve dikey eksen boyunca gri seviye değişimini gösteren ve Denklem 3.21'da verilen gradyan değerinin genlik bileşeni (Genlik bileşeni Denklem 3.22'de verilmiştir),
- Gradyan değerinin açı bileşeni (Denklem 3.23),
- Her bir bloğun birinci türev değeri,
- Her bir bloğun ikinci türev değeridir.

Mozaik görüntüdeki her bir blok için yukarıda listelenen öznelik değerlerinin hesaplanmasıyla  $R_i$  referans modeli oluşturulmuştur. Diğer yöntemlerde olduğu gibi bu model matris de elde edildiği coğrafi konumun GPS koordinat bilgisi ile ilişkilendirilerek veri tabanına kaydedilir. Öznelik matrisi mozaik imgedeki her blok için 1x4 boyutundaki vektörlerden oluşmaktadır. Her bir mozaik görüntüde  $11 \times 76 = 836$  adet dikdörtgen blok olduğu için referans model matrisi  $836 \times 4$  boyutlarında oluşmuştur.

### 3.9.5.2 Gerçek Zamanlı Hareketli Nesne Tespiti

Gerçek zamanlı işlemlerin başlangıcında mobil araç kontrol sisteminin hafızasında rotası boyunca belirlenmiş her bir  $K_i$  koordinatı için varsayılan arka planı temsil eden bir model matris oluşturulmuş durumdadır. Mobil araç belirlenen rota boyunca ilerlerken üzerinde bulunan GPS alıcı sistem sayesinde 500 ms. 'de bir koordinat bilgilerini almaktadır. Zamansal bir döngü içerisinde bu bilgi hafızadaki koordinat bilgileri ile karşılaştırılır. Uyuşma sağlandığı anda aracın üzerinde bulunan kamera ile arka planın anlık görüntüleri yakalanır.

Anlık imgenin model matrisler ile karşılaştırılabilmesi için aynı yöntemlerle modellenmesi gerekir. Bu nedenle anlık imge, referans imgelerde olduğu gibi 64x52 boyutlarında dikdörtgen bloklara ayrılır ve her bir bloğa kendi ortalama gri seviye değeri verilerek mozaik imge elde edilir. Böylece anlık imgeye ait model  $A_i$  oluşturulmuş olur. Modellerin oluşturulmasının ardından referans ve anlık imge çerçeveleri arasındaki kayma vektörünün belirlenebilmesi için iki model arasında karşılaştırma yapılması gerekir. Bu karşılaştırma işlemi  $R_i$  ile  $A_i$  matrisleri arasındaki Öklid uzaklığının hesaplanmasıyla gerçekleştirilir. Hesaplama sonucunda referans model ile anlık model arasında blok başına vektörel uzaklıklar çıkartılır. En kısa mesafeli bloklar arasındaki uzaklık değerleri ile *Uzaklık* matrisi oluşturulur. Bu matrisin oluşturulması işlemlerinde Şekil 3.19'da ayrıntıları açıklanan algoritma kullanılmıştır.

Bu çalışmada *Uzaklık* matrisinde en çok tekrar eden değerler iki model arasındaki kayma miktarı olarak kabul edilmiştir. Kayma miktarı yatay ve dikey eksen kayma değerlerini içermekte olup anlık imgenin referans imgeye hizalanması için kullanılmıştır.

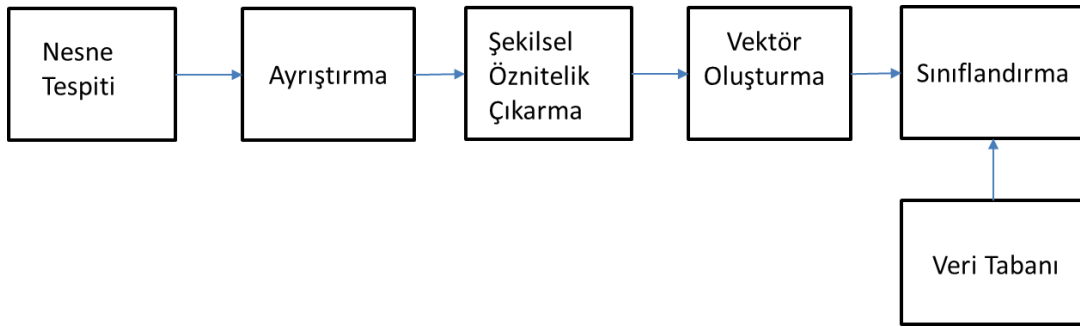
### 3.10 Nesne Tanıma İşlemleri

Bu çalışmada tanınacak olan nesnelere yayalar, otomobiller ve bisikletliler olarak kategorilere ayrılmıştır. Yani tüm nesnelere sınıfları önceden belirlenmiştir. Nesne tanıma işlemi veri tabanında kayıtlı olan birtakım özelliklerine göre sınıflandırılmış nesne modelleriyle anlık görüntüde nesne olarak belirlenen görüntü parçasının eşleştirilmesi sonucunda yapılmaktadır.

### 3.10.1 Tanıma İşlem Adımları

Nesne tanıma işlemi, sistemin girişine uygulanan nesne görüntüsünün özniteliklerini baz alarak kendisine en yakın özniteliklere sahip olan sınıfa atanması ile yapılmaktadır. Tüm bu işlemlerde sınıfları temsil eden modeller tanıma işleminden önce hazırlanmalıdır. Veri tabanında kayıtlı olan modeller tanınacak nesne ile ilgili tüm öznitelikleri kapsamalıdır.

Yukarıda belirtilen nesnelerin tanınması için öncelikle akan görüntü çerçeveleri içinde hedef nesnelere tespit edilmiştir. Ardından nesne görüntüden ayrıştırılarak tek başına ele alınmış ve bir takım işlemlerden geçirilerek sınıflandırılmıştır. Sınıflandırma adımları Şekil 3.37'de verilmiştir.



Şekil 3.36 Nesne tanıma işlem adımları.

Nesne tespiti işlemleri Bölüm 3.9'da beş ayrı yöntem kullanılarak yapılmıştır. Tanıma işlemlerinin ilk aşaması bu şekilde gerçekleştirilmiştir. Ayrıştırma işleminde ise tespit işlemi sonucunda oluşan ikilik görüntü üzerindeki gürültü kaynaklı piksel öbekleri ile gerçekten nesne olabilecek piksel öbekleri ayırt edilmiştir. Bunun için ikilik görüntü üzerinde bir kısım morfolojik işlemler gerçekleştirilerek elde edilen piksel öbeklerinin boyutlarına, piksel yoğunluklarına, tek parça olup olmadıklarına ve şekilsel özelliklerine göre hesaplamalar gerçekleştirilmiştir.

Genel olarak en büyük ve tek parçadan oluşan öbekler hareketli nesne olarak kabul edilmiştir. Ayrıştırma işleminin sonunda elde edilen ikilik imgedeki nesne silüetinin etrafı kutulanmıştır. Bu şekilde silüetleri kutunun ortasına hizalayarak elde edilen görüntünün ötelemeden bağımsız değişmez öznitelikleri elde edilebilmesinin yolu açılmıştır. Kutulanmış silüet kenarları çıkartılarak gereksiz hesaplama maliyeti oluşturan gereksiz piksel verileri de atılmıştır.

Üçüncü adım olan şekilsel özniteliklerin çıkarılması adımımda kutulanmış siluetin iki boyutlu düzlemdeki pozisyonundan çıkarılabilecek öznitelikler elde edilmiştir. Çetin (2011) çalışmasında Graves ve Batchelor (2003) tarafından ifade edilen şekilsel öznitelikleri bu çalışmada referans olarak kullanılmıştır. Bu öznitelikler;

- Nesne siluetinin kenarı üzerindeki en uzak noktanın, nesnenin merkezine olan uzaklığı,
- Nesne siluetinin kenarı üzerindeki en yakın noktanın, nesnenin merkezine olan uzaklığı,
- Yarıçapı, ilk iki maddede hesaplanan uzaklıkların ortalamasına eşit olan çemberin dışına çıkan yayların sayısı,
- Nesne siluetinin kenarında bulunan noktaların merkeze uzaklığının açısız pozisyon cinsinden ifadesi, (böylece, siluet polar koordinat terimleri ile ifade edilir. Bu tek değerli bir fonksiyon değildir.)
- Dairesellik = Alan / Çevre<sup>2</sup>, (Bu değer dış hatları parçalı olan düzensiz şekiller için sıfıra yaklaşırken, bir daire için maksimum değer olan  $\Pi/4$ 'e yaklaşır)
- Nesne siluetinin üzerinde bulunan hücrelerin sayısı,
- Nesne siluetinin kenarlarında, dıştan içeriye giren köşelerin sayısı,
- Nesne siluetinin üzerindeki, devam eden eğri sayısı ile, bu eğriler arasında kalan boşluk sayısının birbirinden çıkarılması ile elde edilen, Euler numarası,
- Gerçek resim alanı ile, resme ait dışbükey zarf (convex hull) alanının oranı,
- Gerçek resim alanı ile, resme ait çevrel daire (circumcircle) alanının oranı,
- Resim alanının, resme ait iskeletteki toplam dal uzunluklarının karesine oranı,
- İskeletteki eklem yerleri ile dalların bitişi arasındaki mesafeler,

Fu ve Liu (2001) çalışmalarında tespit edilecek nesnelerin  $h$  (height) ve  $w$  (width) olarak isimlendirdikleri nesne yüksekliği ve nesne genişliği özniteliklerini kullanmışlardır. Bu çalışmada da bu değerler kullanılmış ancak ölçek değişmezliği için bu iki bilginin  $h/w$  şeklinde oranı dikkate alınmıştır.

### 3.10.2 Yapay Sinir Ağları ile Sınıflandırma

Tanıma işlem adımlarında sınıflandırma işlemi matematiksel bir model ile yapıldığı takdirde veri tabanında kullanılan model ile ilişkili nesne bildirim modellerini tutmak

gerekir. Ancak sınıflandırma işleminde YSA türevi yöntemler kullanılacak ise nesne tanıma verileri sinir ağı ağırlıklarında tutulacağı için bir veri tabanına ihtiyaç bulunmamaktadır.

Genel olarak nesnelerin doğru olarak tanınması pek çok koşulun bir arada gerçekleşmesiyle yapılabilmektedir. Ancak çoğu zaman ideal şartlar oluşmaz ve bazı eksiklikler ve belirsizlikler meydana gelebilir. Değişken aydınlanma koşulları, nesnenin bir kısmının bir başka cismin arkasında kalması, aşırı hareketli arka plan, esnek yapıdaki nesnelere gibi belirsizlikler yapılacak işlemin başarısını olumsuz yönde etkilemektedirler. Bu nedenlerle matematiksel ve yapısal hesaplama yöntemleri çoğu zaman hız ve doğruluk anlamında istenilen sonucu verememektedir.

Buna karşılık paralel çalışan ve birbirleriyle bağlantılı işlem elemanlarından oluşan yapay sinir ağı sınıflandırıcıları insan düşünme ve sınıflandırma sistemini taklit edebilen sistemler olup nesne tanıma işlemlerinde geniş olarak kullanılmaktadırlar. Bu teknikte basit işlem elemanlarının kullanılmasına rağmen bu işlem elemanlarının paralel çalışması sonucunda hız ve doğruluk açısından tatmin edici sonuçlar alınabilmektedir.

Örnek olarak; bir üçgenin tanımı “üç kenar ve üç açığıya sahiptir” önermesiyle yapıldığında üçgenin kenarlarından birisi imge üzerinde görüntülenemeyecek durumda ise matematiksel modeller bu nesnenin üçgen olduğunu anlayamayacaktır. Bu sorunu çözmek için sınırlı bilgiler yardımıyla görüntülenen nesnenin üçgen olduğunu anlayabilecek bir model gerekir. Bu çıkarsamanın yapılması için insan düşünme ve algılama sistemine benzer bir model kullanılması gerekir. Bu tür sorunlarda en iyi çözüm YSA yaklaşımları kullanmaktır. Diğer bir güçlük ise nesnelere buldukları sınıflara göre tanımdır. Özel bir nesneyi tanımak, belirli bir sınıftaki nesnelere tanımdan daha zordur (Ikeuchi ve Hong 1991, Türkoğlu 1996)

Tasarlanan YSA mimarilerinde her bir çıkış farklı bir nesne sınıfını temsil eder. Sınıflandırma işlemi için pek çok algoritma geliştirilmiştir. Bu çalışmada Çok Katmanlı Ağ (Multi Layer Perceptron-MLP) mimarisinde Geri Yayınımlı Sinir Ağı (Back Propagation Neural Network-BPNN) algoritması uygulanmıştır.



Bu çalışmada 5 girişli ve 1 çıkışlı bir YSA mimarisi uygulanmıştır. Mimaride 50 işlem elemanından oluşan bir gizli katman bulunmaktadır. Girdi değişkenleri aşağıda sıralanmıştır.

- I. *En/Boy* oranı (EBO): Tespit edilen nesnenin piksel sayısı cinsinden eni ile boyu arasındaki orandır. Ölçek değişmezliği için oran olarak alınmıştır. Aksi takdirde nesnenin kamera lensine uzaklığına göre farklı değerler elde edilecektir.
- II. *Alan/Çevre* oranı (AÇO): Tespit edilen nesnenin piksel sayısı olarak toplam alanı ile çevresi arasındaki orandır. Yine ölçek değişmezliği için oran tercih edilmiştir.
- III. *Dairesellik* (D): 0 ile 1 arası skalar bir değer olup “1” değerine yakın olması daireselliğe yaklaştığını “0” değerine yakın olması ise çizgiselliğe yaklaştığını göstermektedir.
- IV. *Nesne/Kutu* oranı (NKO): Bu değer nesnenin piksel cinsinden alanının kutunun toplam alanına oranıdır. Oran olarak alındığı için ölçek değişmezliğine sahiptir.
- V. *Dip Nesne/Kutu oranı* (DNK): Bu değer nesnenin en alt 1/20’lik kısmının nesne/kutu oranıdır.

Çıkış ise aşağıdaki listede belirtildiği şekilde tasarlanmıştır. Gizli katman ve çıkış katmanında aktivasyon fonksiyonu olarak hiperbolik tanjant fonksiyonu kullanılmıştır.

0: Yaya

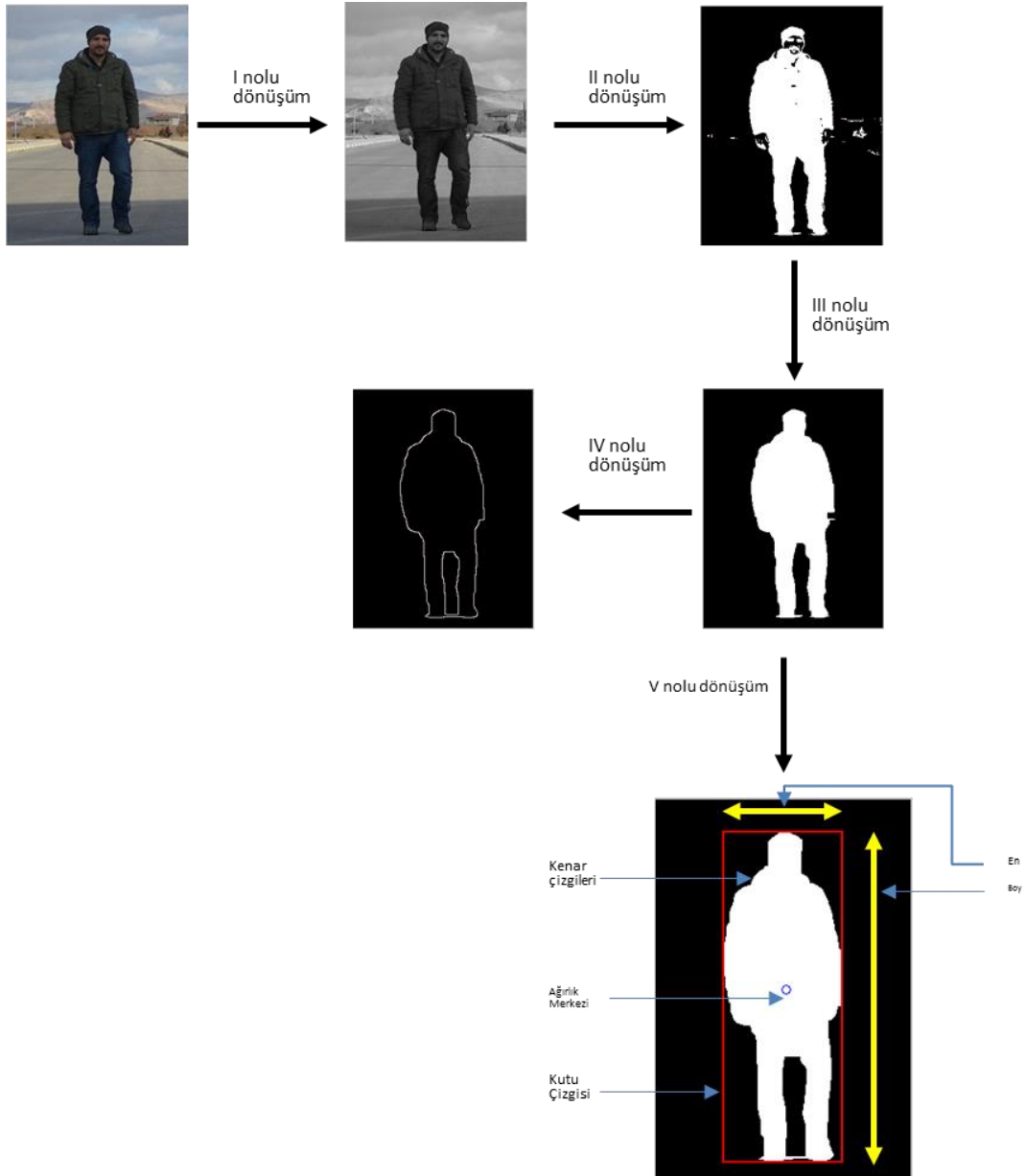
1: Otomobil

-1: Bisikletli

Tasarlanan YSA’nın girişlerinde kullanılacak olan nesne silüetlerinin şekilsel özniteliklerinin çıkartılarak girişlere uygulanabilmesi için bir takım dönüşümlerden geçirilmesi gerekmektedir. Dönüşüm işlemlerinin sıralaması ve nesne içeren imgeler üzerindeki etkisi Şekil 3.37’de örnek bir imge üzerinde gösterilmiş ve aşağıda sırasıyla açıklanmıştır.

- I. **RGB (Red-Green-Blue) görüntüden Gri seviye görüntüye dönüşüm işlemi.**  
Gri seviye resimler belirli bir aralık içerisinde sıralanmış aydınlık seviye değerlerinden oluşan matrisler tarafından temsil edilirler. Matris elemanları (0..1) arası gerçel sayı değerleri alabildiği gibi 8 bitlik ikili değerlerden oluşan (0..255)

arası sayılardan da oluşabilir. Bu sistemde 0 değeri siyah rengi temsil ederken 255 değeri siyah rengi gösterir.



Şekil 3.37 Tespit edilen nesnelerin dönüşüm işlemleri.



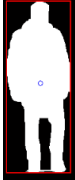
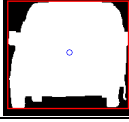
**II. Gri seviye görüntüden ikilik görüntüye dönüşüm işlemi.** İkilik resimler 0 ve 1 değerlerinden oluşan matrisler tarafından temsil edilirler. Siyah ve beyaz renklerden oluşurlar. Sadece gri seviye görüntülerden dönüştürülebilirler.

Dönüşüm işlemi için bir eşikleme işlemi yapılması gerekir. Yani belirli bir eşik değerden küçük olan pikseller 0 değerine atanırken büyük olanlar 1 değerine atanırlar. Genel olarak görüntü bölütleme işlemlerinde kullanılmaktadır. Bu çalışmada odaklanılan nesnenin görüntü arka planından ayrılması için kullanılmıştır. Eşik değerler manuel olarak belirlenebildiği gibi gri seviye görüntü histogramından faydalanılarak otomatik olarak da belirlenebilir. En yaygın kullanılan yöntem Otsu yöntemidir. Yapılan bu dönüşüm işleminde ortamdaki aydınlık seviyesi, arka planın aşırı detay içermesi, nesnenin arka plana çok yakın renk seviyelerinde olması vs. sebeplerden dolayı bir takım gürültüler oluşur. Görüntülerin YSA girişlerine uygulanarak doğru sonuçlar alınabilmesi için bu gürültülerin yok edilmesi gerekmektedir.

- III. Gürültülü ikilik görüntüden temiz ikilik görüntüye dönüşüm işlemi.** Bu dönüşüm işlemi için morfolojik işlemler olarak adlandırılan aşındırma (erosion) ve yayma (dilation) işlemleri ile bunların ardışık olarak gerçekleştirilmesiyle yapılan açma ve kapama işlemleri uygulanmıştır. Morfolojik işlemler çoğunlukla ikili görüntüler üzerinde uygulanmalarına rağmen gri seviye görüntülere de uygulanabilmektedirler. Bu dönüşümde temiz bir nesne silueti elde etmek amacıyla morfolojik işlemler ile beraber bir kısım filtreleme teknikleri de kullanılmıştır.
- IV. İkilik nesne siluetinden kenarları belirlenmiş nesne silüetine dönüşüm işlemi.** Bu dönüşüm, bilgisayar sistemlerini gereksiz hesaplama maliyetinden kurtarmak amacıyla fazla piksellerin temizlenmesi amacıyla yapılmıştır. Kenar belirleme algoritması olarak Canny(1986) algoritması kullanılmıştır.
- V. Nesne silüetinden öznitelik vektörüne dönüşüm işlemi.** Bu işlem adımında dönüşüm işlemleri sonucunda elde edilmiş olan içi dolu imgeden şekilsel özniteliklerin çıkarılması işlemi ile sayısal değerlerden oluşan YSA girdi vektörleri elde edilmektedir. Özellikle AÇO ve NKO değişkenlerinde kenarları çıkartılmış nesnelere ayırt edicilik sağlanamamaktadır. Bu durumda YSA hatalı sınıflandırmalar yapabilmektedir. Bu nedenle girdi değişkenlerinde ayırt ediciliği arttırmak için içi dolu ikilik görüntü tercih edilmiştir. İki tip görüntü arasındaki

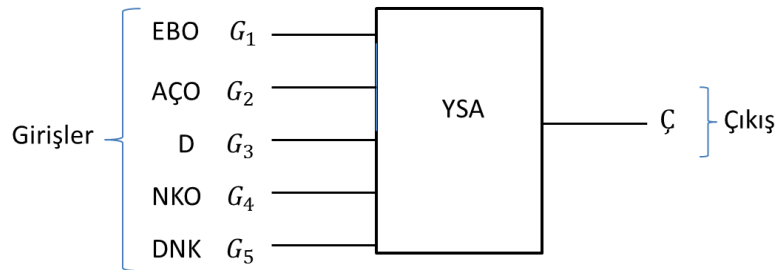
giriş değerleri farklılıkları Çizelge 3.4’de sayısal giriş değerleri verilerek ifade edilmiştir.

**Çizelge 3.4** Kenarları çıkartılmış silüet ile içi dolu silüet arasındaki giriş değişkeni farklılıkları.

Alan/çevre oranı	Nesne piksel/kutu piksel	Nesne
1,08	0,02	
1,05	0,023	
25,95	0,62	
31,01	0,81	

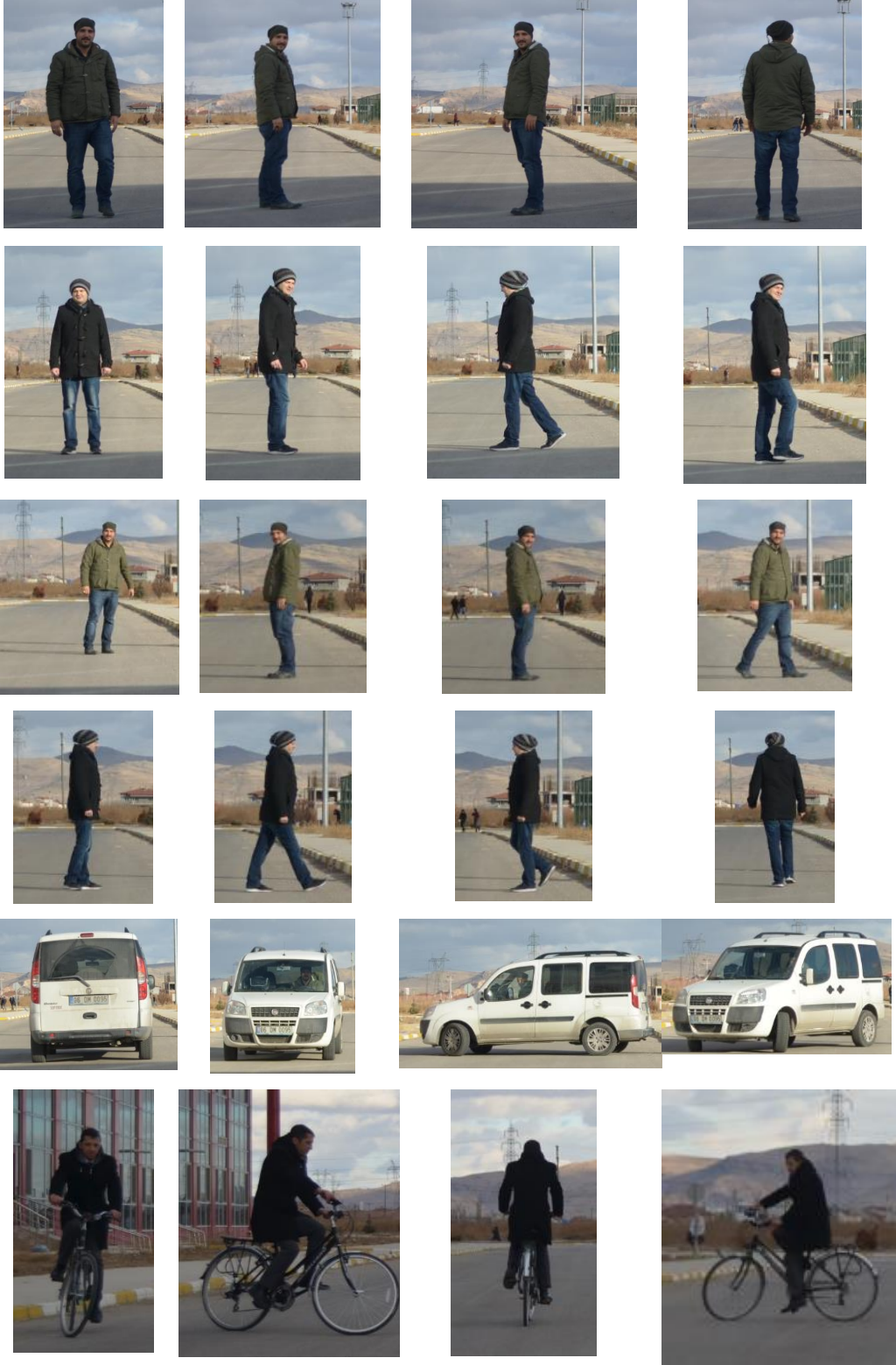
Yukarıdaki çizelgede kenarları çıkartılmış silüetteki AÇO ve NKO değerleri yaklaşık olarak aynı ya da az bir fark bulunuyor iken içi dolu silüette ise arada fark edilebilir bir değer oluşmaktadır. Tasarlanan YSA’nın doğru olarak sınıflandırma yapabilmesi için iki ayrı sınıfa ait girdi değerleri arasında ağırlık genelleştirebileceği sayısal fark değerlerinin bulunması gerekmektedir.

Bu bilgiler doğrultusunda tasarlanan YSA mimarisi kabaca Şekil 3.38’de gösterilmiştir.



**Şekil 3.38** Tasarlanan YSA mimarisi.

Tasarlanan YSA'nda girdi olarak kullanılan imgelerden bir kısmı Şekil 3.39'da verilmiştir.



Şekil 3.39 Girdi olarak kullanılan bir kısım imgeler.

Burada tasarlanan YSA'nın eğitilmesinde kullanılan farklı beden ölçülerinde, farklı pozisyonlarda (ön, yan, arka, yürüyüş pozisyonu, durma pozisyonu vs.) ve farklı kıyafetlerde yayalar, farklı pozisyonlarda araç ve bisikletliler verilmiştir.

### **3.11 Ara Yüz Tasarımları**

Ara yüz tasarım ve kodlamaları Matlab R2013a GUI araç kutusu kullanılarak gerçekleştirilmiştir. Yapılan çalışmada 2 tipte ara yüz geliştirilmiştir.

1. Video akış analizi yapmak için oluşturulan ara yüzler.
2. Gerçek zamanlı çalışmada kullanılan ara yüzler.

Tespit ve tanıma işlemini gerçekleştirmek için öncelikle varsayılan arka planların modellenmesi gerektiği için her iki tip ara yüzde de;

1. Model Oluşturma
2. Nesne Tespit Tanıma

modülü olmak üzere 2 adet modül mevcuttur. Bu ara yüz tasarımlarında modelleme işlemi Enkoder Destekli Çerçeve Sayma yöntemi ile oluşturulan referans noktalar üzerinden gerçekleştirilmiştir.

#### **3.11.1 Video Akış Analizi Ara Yüzleri**

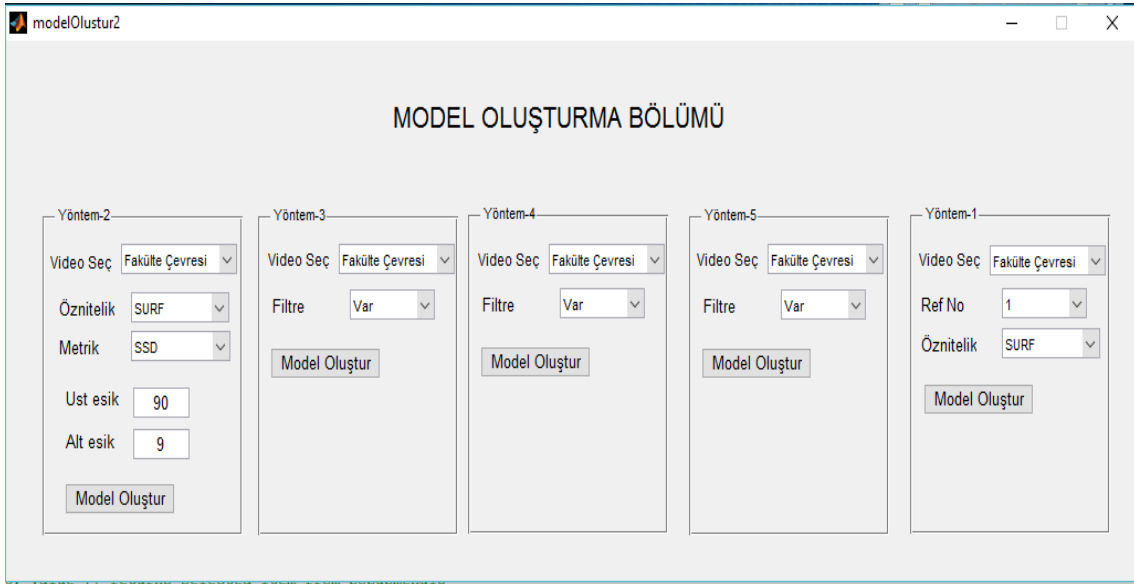
##### **3.11.1.1 Model Oluşturma**

Bu modülde nesne tespiti yapılan beş ayrı yonteme ait model oluşturma işlemleri yapılmaktadır. Her yöntemin kendine özel bir kısım parametreleri ara yüz üzerinden seçilebilir ya da klavye ile girilebilir şekilde tasarlanmıştır. Bunlardan bir kısmı aşağıda verilmiştir.

- Video Seç: Kontrolun yapılacağı rotanın seçimi yapılmaktadır.
- Öznitelik: Yöntem-1 ve Yöntem-2'de kullanılacak öznitelik çıkarma algoritmasının seçimi yapılmaktadır.

- Metrik: Referans imgeler arasında ortak olanlar belirlenirken kullanılacak olan vektörler arası doğrusal uzaklık hesaplama yöntemi seçilmektedir.
- Filtre: İmge çerçevelerinin modellenmesi yapılırken çeşitli filtreleme tekniklerinin kullanılıp kullanılmayacağı seçilir.
- Alt eşik / Üst eşik: Tesadüfi öznelik eşleşmelerini önlemek için geliştirilen algorithmada kullanılmaktadır.

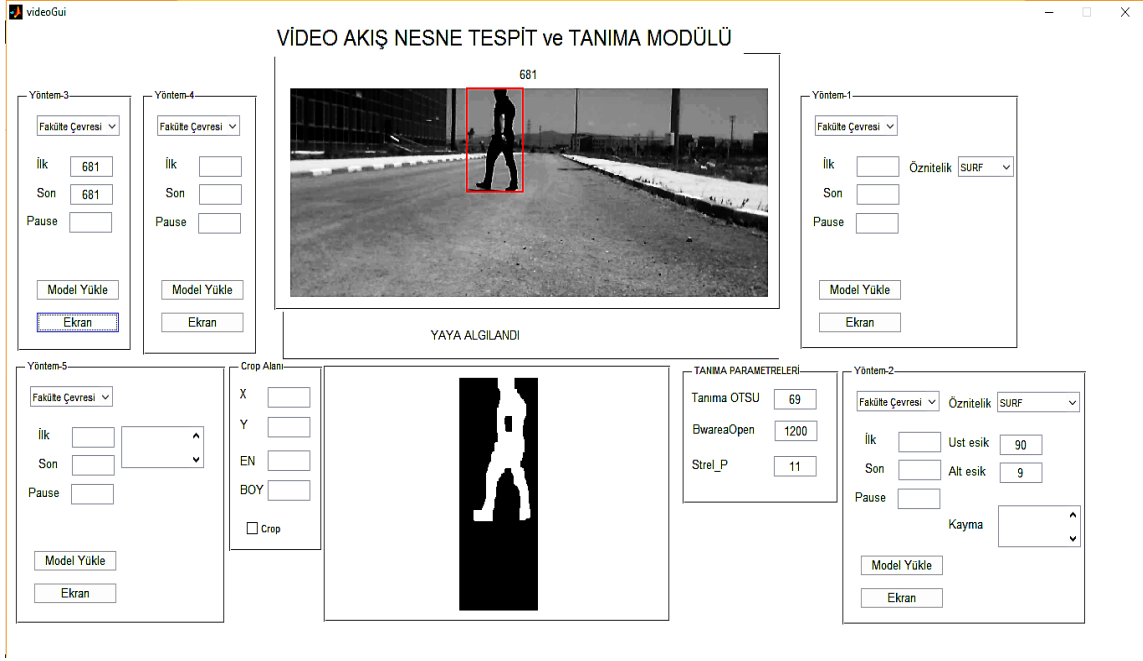
Ara yüz tasarımının ekran resmi Şekil 3.40’da verilmiştir.



Şekil 3.40 Video akışı analizi model oluşturma modülü ekran görüntüsü.

### 3.11.1.2 Nesne Tespit ve Tanıma Modülü

Bu modülde her yöntem için ayrı ayrı tespit ve tanıma yapılabilmektedir. Ara yüzün üzerinde bulunan iki ekrandan birisinde gerçek görüntü izlenebilirken diğer ekranda tespit edilen nesnenin silueti görüntülenmektedir. Nesne tespit ve tanıma modülünün ekran görüntüsü Şekil 3.41’de verilmiştir.



Şekil 3.41 Video akış analizi nesne tespit ve tanıma modülü ekran görüntüsü.

Tespit edilen nesnenin tanıma işlem gerçekleştirildikten sonra nesnenin sınıfı ekran üzerinde görüntülenmektedir. Her yöntemin kendine özel bir kısım parametreleri ara yüz üzerinden seçilebilmektedir. Bunlardan bazıları;

- Rota seçimi yapılabilmektedir.
- Belirlenen iki imge çerçevesinin arası görüntülenip analiz edilebilmektedir.
- İmge çerçevelerinin arasına gecikme süreleri konulabilmektedir.
- Tespit edilen nesne silueti üzerinde yapılacak olan morfolojik işlemlerin parametreleri belirlenebilmektedir.
- Odaklanılan alan kırpma (crop) işlemi yapılarak daraltılabilmektedir.
- İmge çakıştırma yapılan yöntemlerde tespit edilen kayma miktarları her çerçeve için görüntülenebilmektedir.

### 3.11.2 Gerçek Zamanlı Çalışma Ara Yüzleri

Bu bölümde gerçek zamanlı uygulama sırasında kullanılan ara yüzlerin tasarımları ve kullanımlarına ilişkin ayrıntılar verilmiştir.



### 3.11.2.1 Gerçek Zamanlı Model Oluşturma

Gerçek zamanlı çalışmaya geçmeden önce tüm Yöntemlerde ilk aşama olarak kontrolü yapılacak olan rotanın modelini oluşturmak ve kaydetmek gereklidir. Bu nedenle rota boyunca varsayılan arka plan görüntüleri İKA tarafından bu ara yüz kullanılarak modellenir. Varsayılan arka planın modellenmesi için yöntemine göre değişmekle beraber rota üzerinde 1, 5 ya da 10 tur görüntü alınması gerekir. Modelleme sırasında arka planın nesne içermemesi yani boş olması gerekmektedir.

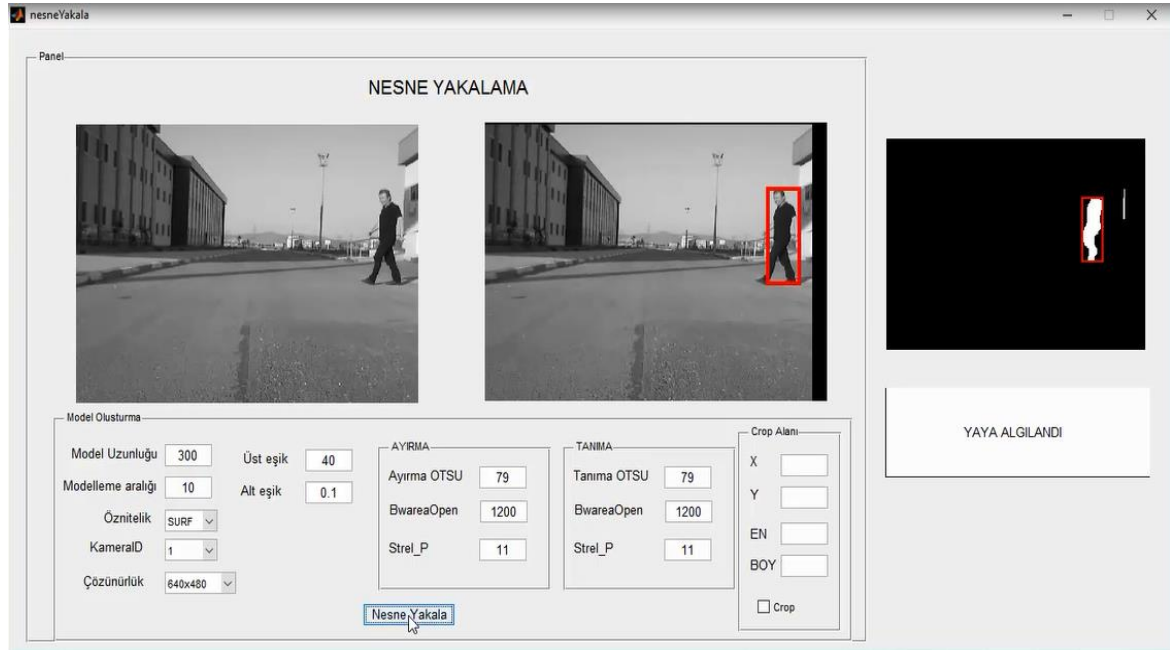
Modelleme için gerekli olan referans noktalar ara yüz üzerinden belirlenir. ‘Model Uzunluğu’ ve ‘Modelleme Aralığı’ parametreleri bu işlem için kullanılmaktadır. Bu değerler girilerek referans noktalar belirlenmiş olur. Her yöntemin kendine özel modelleme ara yüzü bulunmaktadır. Bu deneysel çalışmada Yöntem-2’ye ait olan modelleme ara yüzü kullanılmıştır. Ayrıca; Yöntem-1 ve Yöntem-2 için gerekli olan öznetelik çıkartma algoritması, kamera seçimi ve kamera çözünürlük seçimleri bu ara yüz üzerinden gerçekleştirilebilir. Gerçek zamanlı modelleme modülünün ekran görüntüsü Şekil 3.42’de verilmiştir.



Şekil 3.42 Yöntem-2 için gerçek zamanlı modelleme modülünün ekran görüntüsü.

### 3.11.2.2 Gerçek Zamanlı Nesne Tespit ve Tanıma

Bu modül kullanılarak bir önceki aşamada modellenen rota üzerinde gerçek zamanlı tespit ve tanıma işlemi yapılmaktadır. Her Yöntemin kendine özel bir tespit ve tanıma modülü bulunmaktadır. Bu deneysel çalışmada Yöntem-2'ye ait tespit ve tanıma modülü kullanılmıştır. Bu modülün ekran görüntüsü Şekil 3.43'de verilmiştir.



Şekil 3.43 Yöntem-2'ye ait nesne tespit ve tanıma modülü ekran görüntüsü.

Ara yüz üzerinde üç adet ekran mevcuttur. İlk ekranda canlı görüntü izlenebilirken ikinci ekran modele göre hizalanan görüntünün izlendiği ekrandır. Üçüncü ekranda ise tespit edilen hareketli nesnenin silueti izlenebilmektedir. Ara yüz üzerinde nesne tespit ve tanıma için iki ayrı morfolojik işlem parametre grubu bulunmaktadır. Bu şekilde tespit ve tanıma işlemleri için farklı parametre girilebilmektedir.

Ayrıca odaklanılan ekran parçasını daraltabilmek amaçlı ekran kırpma (crop) parametreleri girilebilmektedir. Daralan ekranda daha net ve daha hızlı tespit ve tanıma yapılabilmektedir. Ayrıca; siluet üzerinden tanıma işlemi gerçekleştirildiğinde nesne sınıfını ekrana yazan bir ekran daha bulunmaktadır.

## 4. BULGULAR

Bu çalışmada nesne tespiti için geliştirilen beş farklı yöntemden elde edilen bulgular Yöntem-1, Yöntem-2, Yöntem-3, Yöntem-4 ve Yöntem-5 olarak bölüm başlıklarında ifade edilecektir.

### 4.1 Yöntem 1: GPS Destekli Hareket Bölgesi Belirleme ve Geometrik Dönüşüm

Bu Yöntem kullanılarak yapılan deneysel çalışmalarda öncelikle hangi öznitelik çıkartma algoritması ve hangi doğrusal uzaklık çıkartma yönteminin kullanılacağı belirlenmiştir. Bu işlemin ardından farklı tipte nesnelerin tespit edilmesi ile ilgili deneysel çalışmalar yapılmış ve son olarak da Yöntem-1'in bilgisayarlar sistemlerindeki performansları test edilmiştir.

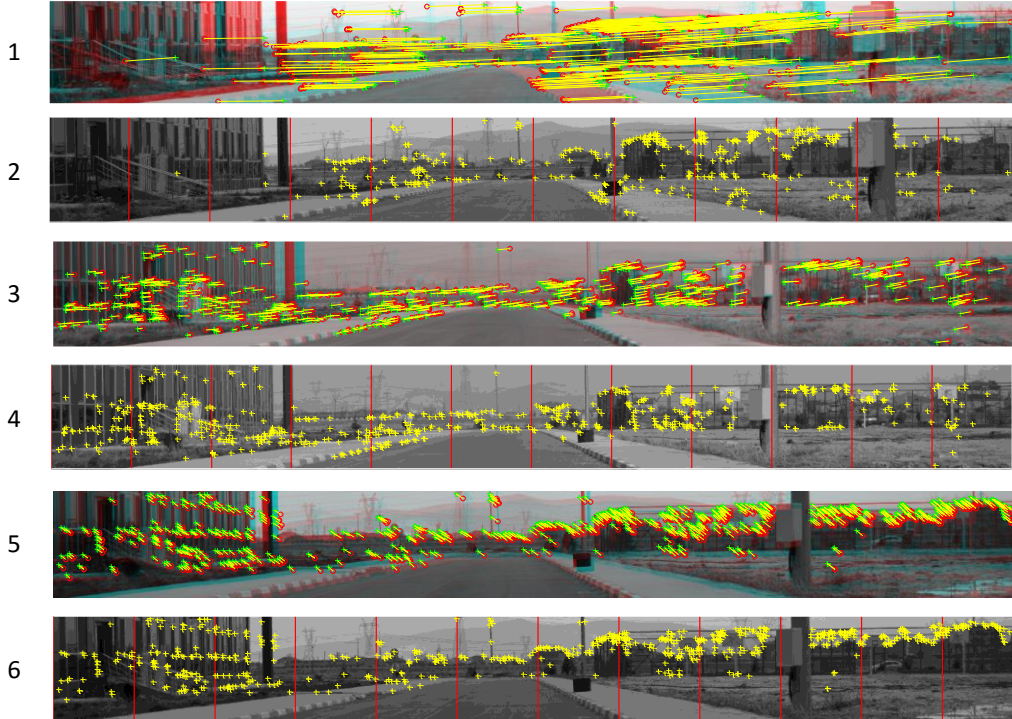
#### 4.1.1 Farklı Öznitelik Çıkartma Algoritmalarının Performansları

Çalışmada kullanılacak öznitelik çıkartma yöntemlerin belirlenebilmesi amacıyla Yöntem-1'de kullanılacak olan referans model SURF, Harris ve FAST yöntemleri kullanılarak belirlenmiştir. İmgelerin arka planlarındaki sahnenin karakterine göre kullanılacak yöntemler değişebilmektedir. Detaylı alanlarda SURF algoritması daha iyi sonuçlar verirken daha az detay barındıran yapıların bulunduğu sahnelerde Harris algoritması daha iyi sonuçlar vermektedir. Deneysel çalışmalarda Şekil 3.13'de belirtilen  $K_1$  ile  $K_{20}$  arasında alınan referans imge setlerinin SURF, Harris ve FAST öznitelik çıkartma algoritmaları ile öznitelikleri çıkartılmıştır. Bu öznitelikler ED, SAD ve NCC yöntemleriyle eşleştirilerek her bir koordinatta varsayılan arka planı temsil eden model matrisler elde edilmiştir. Bu model matrisler ile anlık öznitelik matrislerinin eşleştirilmesi sonucu bulunan ortalama öznitelik sayıları Çizelge 4.1'de verilmiştir. Bu çizelgede görüldüğü gibi  $B_2..B_{13}$  gibi ayrıntılı arka planlarda en iyi sonucun Surf algoritması ve ED Yönteminin,  $B_1$  gibi homojen arka planlarda ise Harris algoritması ile ED Yönteminin daha başarılı olduğu tespit edilmiştir. Fast algoritmasının ise her iki arka plan tipinde de başarısız olduğu tespit edilmiştir. Dar bir aralıktan elde edilen referans görüntülerde arka plan benzerliklerinin daha fazla olması nedeniyle geçiş aralığı mesafesi azaldıkça tüm yöntemlerde elde edilen eşlenik öznitelik sayısının arttığı da gözlemlenmiştir.

**Çizelge 4.1** Farklı öznitelik çıkartma algoritmaları ve farklı karşılaştırma yöntemleri kullanılarak elde edilen ortalama eşlenik öznitelik sayıları.

Yatay resimleme aralığı (m)	Koordinat	Yöntem	Uzaklık (piksel)					
			SURF		HARRIS		FAST	
			B1	B2	B1	B2	B1	B2
2	$K_1$	ED	25	332	110,6	173	0	11
		SAD	19,6	219	85,66	149,3	0	10,3
		NCC	15,6	215,3	38,6	75,2	0	7,6
1	$K_1$	ED	31,3	419,6	95	165,6	0	12,3
		SAD	26	290,3	80,6	156,3	0	11
		NCC	25,3	282	36	68,6	0	7,3
0,5	$K_1$	ED	24,5	520,2	137	247,7	0	13,2
		SAD	20,6	388	115,8	237,3	0	10,5
		NCC	20,7	378,7	52,7	108	0	9,2

Çeşitli koordinatlara ait referans modeller ile nesne içermeyen anlık modellerin üç farklı yatay geçiş aralığı kullanılarak gerçekleştirilen eşleştirme işlemlerine ait görüntüler Şekil 4.1’de verilmiştir. Daha fazla ayrıntı içerdiği için bu işlemde B<sub>2</sub>..B<sub>13</sub> arası bölümler kullanılmıştır.



**Şekil 4.1** Çeşitli referans noktalarından üç farklı yatay geçiş aralığı kullanılarak elde edilen eşleştirme sonuçları

Şekil 4.1’de; 1 ve 2 numaralı imgeler, 2 m geçiş aralığı için  $K_5$  referans noktasına ait model ile boş anlık modelin eşleştirme ve bölümlenme sonuçlarını; 3 ve 4 numaralı imgeler

1 m geçiş aralığı için  $K_1$  referans noktasına ait model ile boş anlık modelin eşleştirme ve bölümlenme sonuçlarını, 5 ve 6 numaralı imgeler 0,5 metre geçiş aralığı için  $K_3$  referans noktasına ait model ile boş anlık modelin eşleştirme ve bölümlenme sonuçlarını ifade etmektedir.

Çizelge 4.2’de ise her bir eşleştirme işlemi için eşlenik noktalar arasındaki uzaklık ve açı değerlerinin ortalamaları ile bu ortalama değerlere ait standart sapmalar verilmiştir. Model ve anlık imgelerin eşlenik öznitelik noktalarının konumları arasındaki uzaklık ve açı değerlerinin birbirlerine olan yakınlığı yani standart sapmalarının küçük olması yapılan işlemin doğruluğunu göstermektedir.

**Çizelge 4.2** Şekil 4.1’de ifade edilen eşleştirmelere ait uzaklık ve açı değerleri.

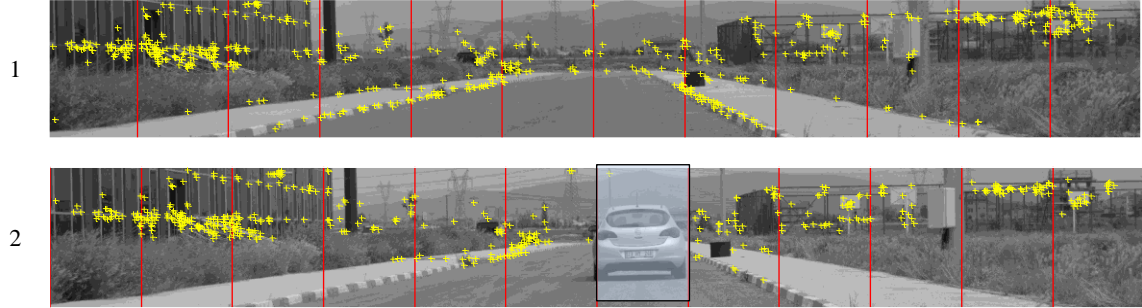
Koordinat	Yatay resimleme aralığı (m)	Ortalama Uzaklık (piksel)	Ortalama Açı (derece)	Uzaklık Standart Sapma	Açı Standart Sapma
K5	2	294,61	-2,72	41,54	1,77
K1	1	83,66	-6,35	22,11	1,98
K3	0,5	41,09	37,39	5,73	3,04

#### 4.1.2 Hareketli Otomobil Kullanılarak Yapılan Çalışma

Hareketli nesne tespiti için her bir referans nokta için oluşturulan modellerin bölüm başına ( $B_2 .. B_{13}$ ) öznitelik sayıları ile anlık eşlenik modellerin bölüm başına öznitelik sayıları bire bir karşılaştırma işlemi yapılmıştır. Karşılaştırma denemeleri için rota üzerindeki  $K_1$  ile  $K_5$  koordinatları arası seçilmiştir. En iyi sonuçların alınması için Çizelge 4.1’de belirlenen en iyi yöntem ikilisi olan SURF algoritması ve ED yöntemi kullanılmıştır.

Yapılan deneysel çalışmada hareketli nesne olarak otomobiller ve insanlar kullanılmıştır. Model öznitelik matrisini oluşturan referans imge seti 2 m geçiş aralığı üzerinden fotoğraflanmıştır. Şekil 4.2’de  $K_1$  koordinatından alınmış ve içinde hareketli bir otomobil yer alan bir anlık imge çerçevesi ile aynı noktadan elde edilmiş ve hafızaya kaydedilmiş model imge verilmiştir. Bu imge çerçevesinde bulunan hareketli nesne  $K_1$  koordinatından 22 m uzaklıkta bulunmaktadır. Bu görüntüye ait anlık öznitelik matrisi ile sistem hafızasında bulunan bu koordinata ait model öznitelik matrisinin eşleştirilmesi sonucunda

anlık eşlenik öznitelik matrisi oluşturulmuştur. Söz konusu matrisin elemanları anlık imge üzerinde işaretlenmiş ve bölümlenmiştir.



**Şekil 4.2**  $K_1$  koordinatından 22m uzaklıkta (1) varsayılan, (2) nesne içeren arka plan çerçevesinin bölümlenmiş görüntüleri.

Bir sonraki adımda anlık imge üzerinde bölüm başına bulunan öznitelik sayıları ile model öznitelik matrisinin bölüm başına öznitelik sayıları karşılaştırılmıştır. Model öznitelik matrisi ile anlık imge öznitelik matrisi eşleştirilirken hareketli nesnenin bulunduğu bölümlerde eşleşme gerçekleşmediği için bu bölümlerin öznitelik sayılarında hızlı bir düşme olacaktır. Bu nedenle karşılaştırma işleminde en yüksek azalma oranına sahip olan bölümler tespit edilmiştir. Çizelge 4.3’de model öznitelik matrisi ile anlık eşlenik öznitelik matrisinin bölüm başına öznitelik sayılarının karşılaştırması verilmiştir. Hesaplanan sonuçlara göre en yüksek azalma oranına sahip olan bölüm numarası çizelge ve imge üzerinde işaretlenerek hareketli nesne varlığı tespit edilmiştir.

**Çizelge 4.3**  $K_1$  koordinatından 22m uzaklıkta nesne içeren ve boş arka plan çerçevesinin bölüm başına öznitelik sayıları.

	$B_2$	$B_3$	$B_4$	$B_5$	$B_6$	$B_7$	$B_8$	$B_9$	$B_{10}$	$B_{11}$	$B_{12}$	$B_{13}$
Model												
Öznitelik Matrisi (Boş)	50	118	52	49	57	30	31	67	41	21	54	43
Anlık Eşlenik Öznitelik Matrisi (Dolu)	35	119	68	35	30	45	1	39	38	18	46	23
Azalma (%)	30	0,85	30,8	28,6	47,4	50	96,8	41,8	7,32	14,3	14,8	46,5

Hareketli nesne varlığı tespit edildikten sonra nesne silüetini çıkartma işlemlerine geçilmektedir. Bu işlem için ardışık video çerçeveleri arasında geometrik dönüşüm uygulayarak çerçevelerin aritmetik olarak farklarının alınması gerekmektedir. Bu çalışmada nesne tespiti için sürekli geometrik dönüşüm ve çerçeve farkı işlemi

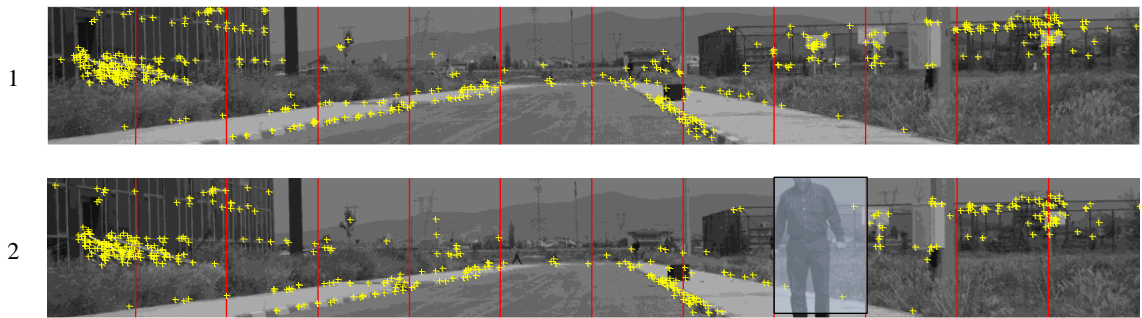
yapılmayıp CPU kapasitelerini çok zorlayan bu işlem sadece nesne tespit işleminden sonra başlamaktadır. Nesne tespitini takiben gerçekleştirilen geometrik dönüşüm ve çerçeve farkı işlemi sonucunda elde edilen hareketli nesne silueti Şekil 4.3’de sunulmuştur. Nesne tespiti için  $B_2..B_{13}$  arası bölümler kullanılmasına rağmen dönüşüm ve siluet oluşturma işlemlerinde tüm çerçeve kullanılmıştır. Çerçeve farkı işlemi sonrası eşikleme yapılmış ve küçük boyutlu gürültüler imge üzerinden kaldırılmıştır. Burada otomobilin siluetinin net olarak ortaya çıktığı görülmektedir.



Şekil 4.3 Geometrik dönüşüm ve ardışık çerçeve farkı ve eşikleme işlemleri sonrası elde edilen hareketli nesne silueti.

#### 4.1.3 Hareketli Yaya Kullanılarak Yapılan Çalışma

Otomobil gibi geniş hacimli nesnelere uzak mesafelerden tespit edilebilmiştir ancak hareketli nesnenin boyutları küçüldükçe tespit mesafesinin de kısaldığı gözlemlenmiştir. Şekil 4.4’de hareketli nesne olarak  $K_3$  koordinatından 12 m uzaklıkta bulunan bir yaya kullanılmıştır. Bu deneme çalışmasında sistem hafızasında bulunan  $K_3$ ’e ait model öznitelik matrisi 0,5 m geçiş aralığına sahip referans imgelerden oluşturulmuştur.



Şekil 4.4  $K_3$  koordinatından 12 m. uzaklıkta (1) varsayılan, (2) nesne içeren arka plan çerçevesinin bölümlenmiş görüntüleri.

Varsayılan arka planın  $B_{10}$  bölümünde bir miktar öznitelik bulunmaktayken anlık görüntünün  $B_{10}$  bölümünde bulunan yayadan dolayı öznitelik sayısının çok azaldığı ve dolayısıyla sayıların uyuşmadığı görülmektedir. Şekil 4.4’de verilen imge çerçevelerine

ait karşılaştırma tablosu Çizelge 4.4’de sunulmuştur. Çizelgeden de anlaşıldığı gibi en hızlı düşüş %90 oranıyla  $B_{10}$  bölümünde yaşanmıştır. Diğer bölümlerdeki azalma oranlarının yaklaşık değerlerde olduğu gözlenmiştir. Bu nedenle sistem  $B_{10}$  bölümünde hareketli bir nesnenin bulunduğunu belirlemiştir.

**Çizelge 4.4**  $K_3$  koordinatından 12 m uzaklıkta nesne içeren ve boş arka plan çerçevesinin bölüm başına öznelilik sayıları.

	$B_2$	$B_3$	$B_4$	$B_5$	$B_6$	$B_7$	$B_8$	$B_9$	$B_{10}$	$B_{11}$	$B_{12}$	$B_{13}$
Model												
Öznelilik Matrisi (Boş)	126	59	39	34	34	7	48	39	30	24	49	28
Anlık Eşlenik												
Öznelilik Matrisi (Dolu)	112	60	32	36	39	5	43	40	3	26	39	29
Azalma (%)	11,1	1,69	17,9	5,88	14,7	28,6	10,4	2,56	90	8,33	20,4	3,57

Geometrik dönüşüm ve çerçeve çıkarma yöntemlerinin ardından Şekil 4.5’de gösterilen nesne silueti elde edilmektedir. Burada yaya siluetinin net olarak ortaya çıktığı görülmektedir.



**Şekil 4.5** Tespit edilen nesne silueti.

Hareketli ya da hareketsiz nesnelerin kamera lensinden uzaklaştıkça hacimleri azaldığı için nesne tespit ölçütü olan bölüm bazında öznelilik sayısındaki azalma oranı düşmektedir. Bu nedenle belirli bir mesafeden sonra tespit için yeterli oranda azalma görülememektedir. 3264x4928 çözünürlükte yapılan deneysel çalışmalarda %85 doğruluk oranı için belirlenen eşik uzaklıklar Çizelge 4.5’de verilmiştir.

**Çizelge 4.5** Otomobil ve yayaları %85 doğrulukta tespit için gerekli eşik uzaklıklar.

Geçiş Aralığı Mesafesi	Nesne Uzaklığı (Metre)		Doğruluk Oranı
	Otomobil	Yaya	
2 m	17	11	% 85
1 m	20	13	% 85
0,5 m	22	14	% 85



Referans imge setlerinin geçiş aralığı mesafeleri arttıkça eşlenik öznitelik sayısının azalması nedeniyle otomobil ve yayaların tespit mesafelerinin kısaldığı anlaşılmaktadır. Ancak doğru tespitler için yeterli uzaklıklar elde edilmiştir.

#### 4.1.4 CPU Yükleme ve İşlem Hızları

Sistemin çalışma hızı ile ilgili deneysel çalışmalarda bilgisayarın CPU kullanım oranları ve imge çerçevesi başına işlem süreleri gözlemlenmiştir. Elde edilen sonuçlar Çizelge 4.6'da verilmiştir.

**Çizelge 4.6** Yöntemlerin performans karşılaştırması.

Yöntem	İmge çözünürlüğü	CPU yükü (%)	Çerçeve başına işlem süresi (sn)
Geometrik Dönüşüm +	3264x4928	%99	5,475
Çerçeve Çıkarma	480x640	%56	0,177
Geometrik Dönüşüm +	3264x4928	%99	5,91
Gauss	480x640	%61	0,24
Geliştirilen Yöntem	3264x4928	%4	0,02
	480x640	%3	0,012

Çizelgede belirtilen GD kısaltması Geometrik Dönüşüm Yöntemini (Gonzalez *et al.* 2009). ÇF kısaltması Çerçeve Farkı Yöntemini, GY kısaltması ise geliştirilen yaklaşımı ifade etmektedir.

Çizelgedeki ilk iki yöntemde nesne tespiti için sürekli olarak ardışık çerçeveler için çerçeve matrislerinin tamamının karşılaştırılması suretiyle işlem yapma zorunluluğu bulunmaktadır. Bir başka deyişle 3264x4928 boyutunda görüntü matrisleri üzerinde ardışık olarak dönüştürme ve çerçeve çıkarma işlemleri yapılmaktadır. Bu türden yüksek boyutlu matris işlemlerinin hızlı bir şekilde gerçekleştirilebilmesi için yüksek maliyetli bilgisayar sistemleri gerekmektedir. Bu çalışmada kullanılan bilgisayar sistemiyle yapılan deneme çalışmalarında %99 CPU yüklenmesi ve 5,475 sn gibi pratikte kullanılması mümkün olmayan bir çerçeve başına işlem zamanı elde edilmiştir. Bu nedenle 3264x4928 çözünürlüğe sahip kaliteli görüntülerde küçük ölçekli bilgisayar sistemleriyle bu yöntemlerin kullanılması mümkün değildir. Ancak 480x640 çözünürlükteki daha az kaliteli görüntülerde çerçeveler arası kısa aralıkta yer değiştirme ile yani kameranın bulunduğu mobil aracın hızını oldukça düşürerek sonuç alınabilmektedir. Bu şekilde bile CPU %56 oranında yüklenmektedir. Mevcut yöntemde ise saniye başına yakalanan imge çerçevesinin anlık öznitelik matrisiyle hafızada kayıtlı

mevcut GPS koordinatına ait model öznitelik matrisinin bölüm bazında karşılaştırması yapılmaktadır. Model öznitelik matrisleri ve  $B_2$  ile  $B_{13}$  arası değerler hafızada mevcut olduğundan karşılaştırma işlemi en yüksek çözünürlükteki çerçeveler için bile milisaniyeler mertebesinde sürmektedir.

## 4.2 Yöntem 2: GPS Destekli Çerçeve Hizalama ve Çerçeve Çıkartma

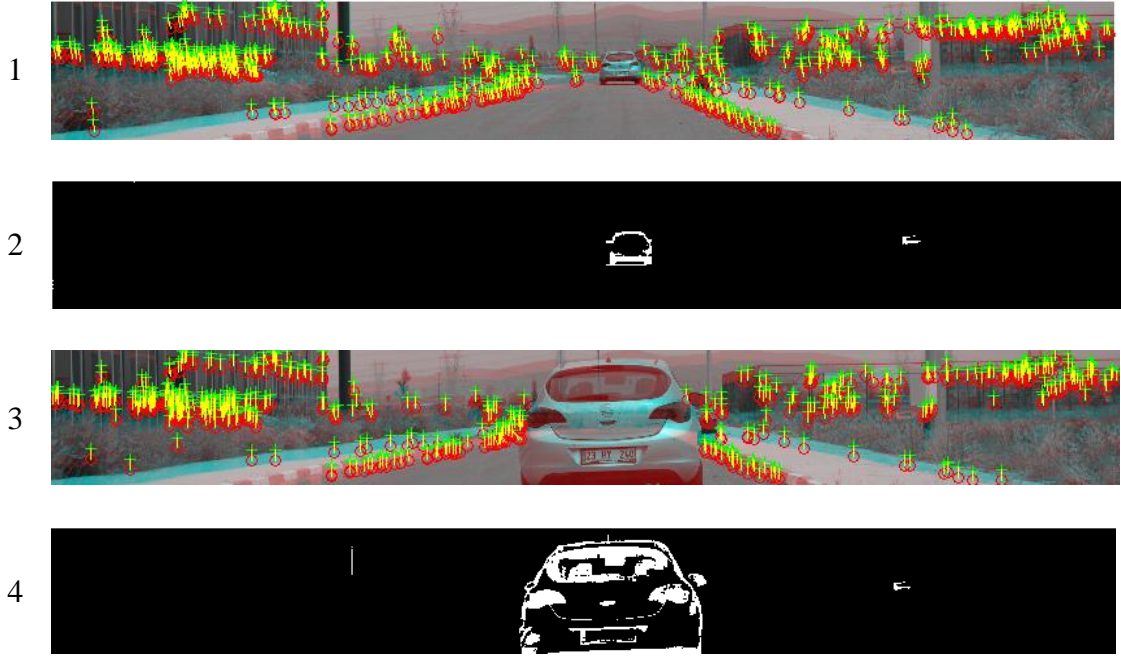
### 4.2.1 Farklı Öznitelik Çıkartma Algoritmalarının Performansları

Yöntem-2’de de ilk yöntem ile aynı öznitelik çıkartma metotları ve aynı referans imge setleri kullanılmıştır. Bu nedenle elde edilen sonuçlarda herhangi bir fark oluşmamıştır. Bu yöntemde imge bölümlenmeleri farklı olduğu için  $B_2$  gibi ayrıntılı arka planlarda en iyi sonucun Surf algoritması ve ED Yönteminin başarılı olduğu görülmüştür. Surf algoritması, daha çok komşu pikseller arasındaki gri seviye değişimlerini kullanması nedeniyle ayrıntılı arka planlarda daha fazla öznitelik çıkartmaktadır. Ancak  $B_1$  gibi homojen arka planlarda gri seviye değişimlerinin fazla olmaması nedeniyle bu tür arka planlarda başarısız olmaktadır. Bu tür arka planlarda ise Harris algoritması ile E Yönteminin daha başarılı olduğu tespit edilmiştir. Fast algoritması imge çerçevesinde bulunan nesnelere köşelerini bulmaya çalıştığı için homojen olan  $B_1$  bölümünde başarısız olmuştur.  $B_2$  bölümünde ise diğer yöntemler kadar öznitelik tespit edememiştir.

### 4.2.2 Hareket Halinde Otomobil Kullanılarak Yapılan Çalışma

Şekil 4.6 (1)’de  $K_1$  koordinatından alınmış ve içinde hareketli bir otomobil yer alan bir anlık imge çerçevesi ile sistem hafızasında ilişkili model arasındaki piksel kaymaları verilmiştir. Bu çerçevede bulunan hareketli nesne  $K_1$  koordinatından 24 m uzaklıkta bulunmaktadır. Bu görüntü çerçevesine ait anlık öznitelik matrisi ile sistem hafızasında bulunan model matrisinin eşleştirilmesi sonucunda iki çerçeve arasındaki 2 boyutlu kayma miktarları piksel cinsinden hesaplanmıştır. Kayma vektörü ortalamaları kullanılarak yapılan imge karşılaştırma işlemi neticesinde kamera hareketinden kaynaklanan anlık çerçevedeki kayma dengelenmiş ve oluşan gürültü yok edilmiştir. Geri kazanılan çerçeve ile anlık çerçevenin aritmetik farkı neticesinde elde edilen nesne silüeti Şekil 4.6 (2)’de verilmiştir.  $K_1$  koordinatındaki kamera kaynağına 10 m uzaklıkta bulunan nesnelere tespiti için yapılan bir diğer çalışmada elde edilen görüntüler Şekil 4.6 (3-4) ‘de verilmiştir. Yakın mesafede sahne gürültülerinden temizlenerek otomobil

detayları daha net olarak elde edilmiştir. Görüntülerden anlaşılacağı üzere geliştirilen yaklaşım ile uzak ya da yakında bulunan nesnelere net olarak tespit edilebilmektedir.



**Şekil 4.6**  $K_1$  koordinatından 24 m uzaklıkta nesne içeren anlık çerçeve ile model arasındaki kayma (1), elde edilen nesne silueti (2), 10m uzaklıkta nesne içeren anlık çerçeve ile model arasındaki kayma (3), elde edilen nesne silueti (4).

Yukarıdaki görüntüler için hesaplanan kayma vektörlerinin ortalama değerleri Çizelge 4.7’de verilmiştir. Çizelgede belirtilen kayma mesafesi ve kayma açısı eşleştirilen tüm öznitelikler için hesaplanmaktadır. Ancak bu değerler hatalı eşleşmelere karşı bir takım filtrelerden geçirilmektedir. Öncelikle aşırı büyük ve aşırı küçük değerler elenmektedir. Sonrasında hesaplanan ortalama değerlere standart sapma değerleri ilave edilerek ve çıkartılarak hesaplanan sınır değerlerin dışında kalan eşleşmeler elenmektedir. Bu eleme işlemi artık elenecek eşleşme kalmayınca kadar bir döngü içinde devam etmekte ve son ortalama değerler hesaplanmaktadır.

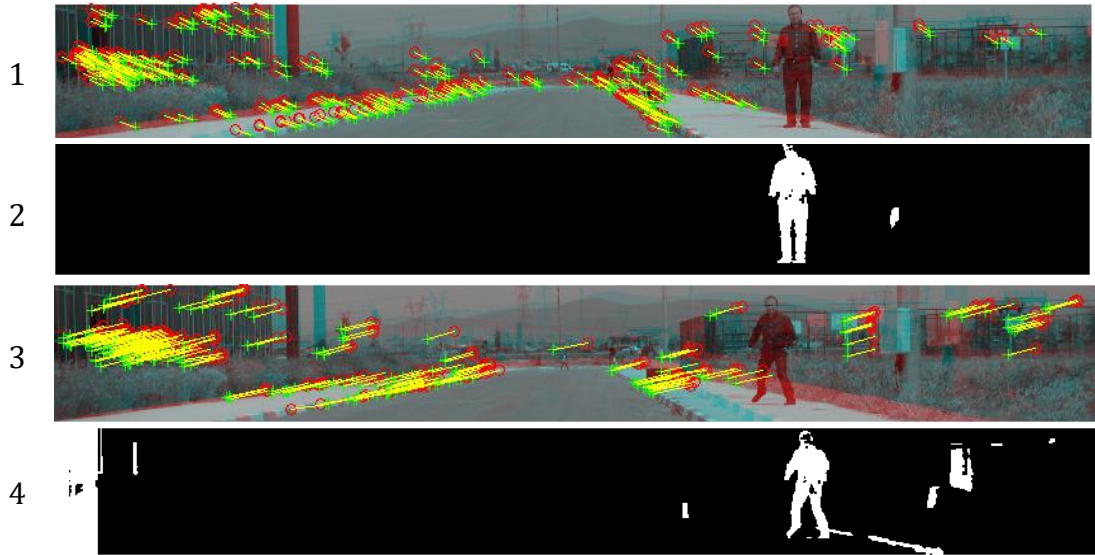
**Çizelge 4.7** Nesne uzaklığı değiştirilerek yapılan deneysel çalışmada elde edilen kayma vektörü değerleri.

Koordinat	Nesne Tipi	Nesne Uzaklığı (metre)	Yatay geçiş aralığı	Ortalama Kayma Mesafesi (piksel)	Ortalama Kayma Açısı (derece)	Mesafe Standart Sapma	Açı Standart Sapma
$K_1$	Otomobil	24	0,5m	46,72	83,01	2,60	1,32
$K_1$	Otomobil	10	0,5m	48,14	89,06	2,08	2,14

### 4.2.3 Hareket Halinde Yaya Kullanılarak Yapılan Çalışma

Model matrislerin elde edildiği geçiş aralığı mesafelerinin değiştirilmesinin nesne tespitine etkilerini belirlemek için yapılan deneysel çalışmada  $K_5$  koordinatında bulunan kamera kaynağına 30m mesafede bulunan yaya görüntüleri analiz edilmiştir. Şekil 4.7 (1) 'de 0,5m geçiş aralığı mesafesinden yakalanan referans çerçeveler ile elde edilen model matris ile anlık çerçeve arasındaki optik akış verilmiştir. Morfolojik işlemler ile gürültülerden arındırılan yaya silüetinin bulunduğu geri kazanılan çerçeve ise Şekil 4.7 (2) 'de verilmiştir. Görüldüğü gibi net bir silüet elde edilmiştir. Şekil 4.7 (3-4)'de ise 2m geçiş aralığı mesafesinden elde edilen model ve anlık çerçeve arasındaki eşleştirmeye ait görüntü ikilisi verilmiştir. Şekil 4.7 (3)'de iki çerçeve arasındaki kayma mesafesinin uzun olduğu görülmektedir. Şekil 4.7 (4)'de verilen geri kazanılmış görüntüde silüetin belirgin olarak ortaya çıkmasına rağmen gürültülerin tam olarak yok edilemediği tespit edilmiştir.

Geçiş aralığı değiştirilerek yapılan deneysel çalışmada ortalama kayma vektörü uzunluğunun 2m geçiş aralığında 0,5 m geçiş aralığına göre yüksek gerçekleştiği, eşleşen öznitelik sayısında ise belirgin bir azalma olduğu Çizelge 4.8'den anlaşılmaktadır.



**Şekil 4.7**  $K_5$  koordinatında 0,5m geçiş aralığı ile elde edilen model ile nesne içeren anlık çerçeve arasındaki kayma (1), elde edilen nesne silüeti (2), 2m geçiş aralığı ile elde edilen model ile nesne içeren anlık çerçeve arasındaki kayma (3), elde edilen nesne silüeti (4).

Bu durum açılı standart sapmasında yüksek bir değeri oluşmasını sağlamıştır. Standart sapmanın yüksek olması çerçeve çakıştırma için kullanılan ortalama kayma vektörü

değerlerinin hata payını yükseltmektedir. Bu nedenle Şekil 4.7 (4)'de görüldüğü gibi geri kazanılan çerçevede yaya silueti yanında gürültüler de oluşmuştur. Ancak 0,5 m geçiş aralığı bulunan çalışmada eşleşen öznitelik sayısının yüksek olması ile birlikte standart sapma değerleri düşük gerçekleşmiştir. Böylece daha kesin olan kayma vektörü değerleri ile geri kazanılan sonuçlar çerçevesinde yaya silueti daha net bir şekilde elde edilmiştir.

**Çizelge 4.8** Geçiş aralığı değiştirilerek yapılan deneysel çalışmada elde edilen kayma vektörü değerleri.

Koordinat	Nesne Tipi	Yatay geçiş aralığı	Eşleşen Öznitelik Sayısı	Ortalama Kayma mesafesi (piksel)	Ortalama Kayma Açısı (derece)	Mesafe Standart Sapma	Açı Standart Sapma
$K_5$	Yaya	2 m	240	153,96	-13,58	16,13	3,77
$K_5$	Yaya	0,5 m	378	66,65	25,71	5,3	1,47

Hareketli ya da hareketsiz nesnelerin kamera lensinden uzaklaştıkça görüntüdeki hacimleri azalmaktadır ve belirleyici niteliklerinin gürültülerden ayırt edilmesi zorlaşmaktadır. Ayrıca geçiş aralığı arttıkça gürültü oranı yükseldiği için bir görsel belirleme eşiği belirlenmesi gerekmektedir. Yapılan deneysel çalışmalarda 3264x4928 çözünürlükte belirlenen eşik uzaklıklar Çizelge 4.9'de verilmiştir.

**Çizelge 4.9** Otomobil ve yayaları %85 doğrulukta tespit için gerekli eşik uzaklıklar.

Geçiş Aralığı Mesafesi	Nesne Uzaklığı (Metre)		Doğruluk Oranı
	Otomobil	Yaya	
2 m	18	10	% 85
1 m	22	14	% 85
0,5 m	24	15	% 85

Geçiş aralığı mesafeleri arttıkça eşleşen öznitelik sayısının azalması nedeniyle ortalama kayma vektörünün hata payının arttığı bu nedenle de otomobil ve yayaların tespit mesafelerinin kısaldığı Çizelge 4.9'da verilen değerlerden anlaşılmaktadır. Ancak doğru tespitler için yeterli uzaklıklar elde edilmiştir.

#### 4.2.4 CPU Yükleme ve İşlem Hızları

Sistemin çalışma hızı ile ilgili denemelerde öncelikle bilgisayarın CPU kullanım oranı gözlemlenmiştir. Deneme çalışmalarında 4 çekirdekli 2,8 GHz işlemciye sahip 4 GB RAM kapasiteli bir bilgisayar kullanılmıştır. Elde edilen sonuçlar Çizelge 4.10'da verilmiştir.

**Çizelge 4.10** Yöntemlerin performans karşılaştırması.

Yöntem	İmge çözünürlüğü	CPU yükü (%)	Çerçeve başına işlem süresi (sn)
Geometrik Dönüşüm + Çerçeve Çıkarma	3264x4928	%99	5,475
Geometrik Dönüşüm + Gauss	480x640	%56	0,177
Geliştirilen Yöntem	3264x4928	%99	5,91
	480x640	%61	0,24
	3264x4928	%7	0,043
	480x640	%5	0,027

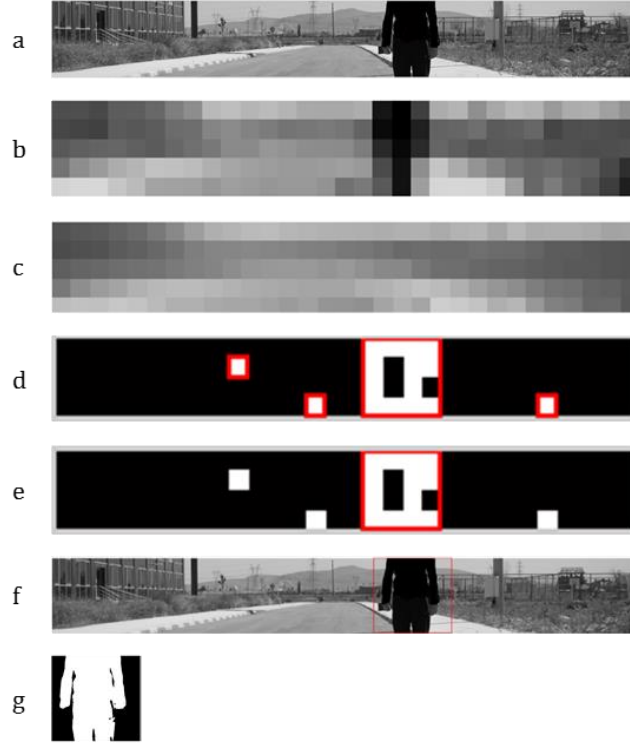
Yöntem-2’de de model matrisler hafızada mevcut bulunduğundan karşılaştırma işlemi en yüksek çözünürlükteki çerçeveler için bile milisaniyeler mertebesinde sürmektedir ve CPU yüklemesi diğer metotlara göre oldukça düşük gerçekleşmektedir.

### 4.3 Yöntem 3: GPS Destekli Gradyan Tabanlı Gri Seviye Değişimi Modellemesi

Geliştirilen yöntemi test etmek amacıyla Şekil 3.13’de belirtilen  $K_1$  ile  $K_{20}$  arası referans noktalara farklı mesafelerde bulunan nesnelere ile deneysel çalışmalar yapılmıştır. Bu yöntemde gerçek zamanlı çalışma öncesi rota üzerinde belirlenen referans noktalarında varsayılan arka planı temsil eden model matrisler hafızada hazır durumdadır. Yapılan deneysel çalışmalarda yabancı nesne olarak yaya ve otomobiller kullanılmıştır. Referans modeller oluşturulurken farklı geçiş aralıkları kullanılmış, anlık imgelerde ise kameraya farklı uzaklıklarda bulunan nesnelere kullanılmıştır. Yöntemde sonuca ulaşılırken çeşitli işlem basamakları icra edilmektedir. Her işlem basamağında farklı görüntü çerçeveleri ve ara değerler elde edilmektedir.  $K_1$  koordinatına 10 m uzaklıkta bir yayanın arka plana dahil olduğu bir imge çerçevesi ile deneysel çalışma yapılmıştır. Çalışma sonucunda işlem basamaklarında elde edilen imge çerçeveleri ve ara değerlerin görüntüleri Şekil 4.8’de verilmiştir. Nesne içeren görüntü mobil araca yakın olduğu için yaya silüetinin tamamı ortaya çıkmamıştır ancak nesne tanıma sistemlerinin bu tür eksik parçalı nesnelere de tanıyabiliyor yetenekte olmaları gereklidir.

Şekil 4.8 (a) bölümünde nesne içeren anlık imgenin orijinal hali, (b) bölümünde anlık imgenin bloklara ayrılmış mozaik hali, (c) bölümünde  $K_1$  kodinatından alınan referans imgelerin ortalaması ( $R_1$ ), (d) bölümünde model ile anlık imgenin karşılaştırılması sonucu elde edilen piksel öbeklerinin kümelenmiş görüntüsü, (e) bölümünde küçük alanlı öbeklerin elenmesi sonucu kalan en yüksek alanlı kümenin görüntüsü ve piksel öbeğinin sınır çizgileri, (f) bölümünde sınır çizgilerinin dönüştürülmesi sonucu orijinal görüntü

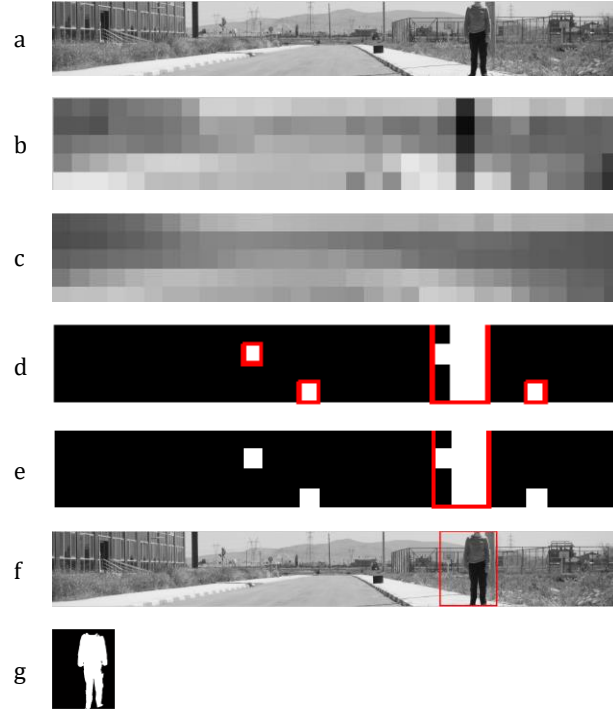
üzerindeki izdüşümleri, (g) bölümünde ise sınır çizgilerinin arasında kalan alan içinde bulunan nesnenin ikilik görüntüye çevrilmiş silueti verilmiştir.



**Şekil 4.8** Yakın mesafeden hareketli nesne silüetinin elde edilmesi sürecinde oluşturulan imge örnekleri.

Özellikle (d) bölümünde görülmekte olan bir blok boyutlu küçük öbekler ihmal edilmiştir. Yapılan deneme çalışmalarında genellikle bu tür durumların oldukça uzak mesafede bulunan hareketli nesnelere ya da arka planda oluşan gürültülerden kaynaklandığı belirlenmiştir.

Bu deneysel çalışmada referans imgeler 1 m geçiş aralığında oluşturulmuştur. Geçiş aralığı mesafeleri daraldıkça referans imgeler arasındaki fark azaldığı için ortalama imgenin doğru arka planı yansıtma oranı artmaktadır. Yapılan çalışmalarda elde edilen sonuçlar da bunu doğrular niteliktedir. Şekil 4.9'da yine  $K_1$  koordinatından 20 m uzaklıkta bulunan bir yaya içeren anlık imge çerçevesi ile yapılan çalışma sonucunda elde edilen görüntüler verilmiştir.

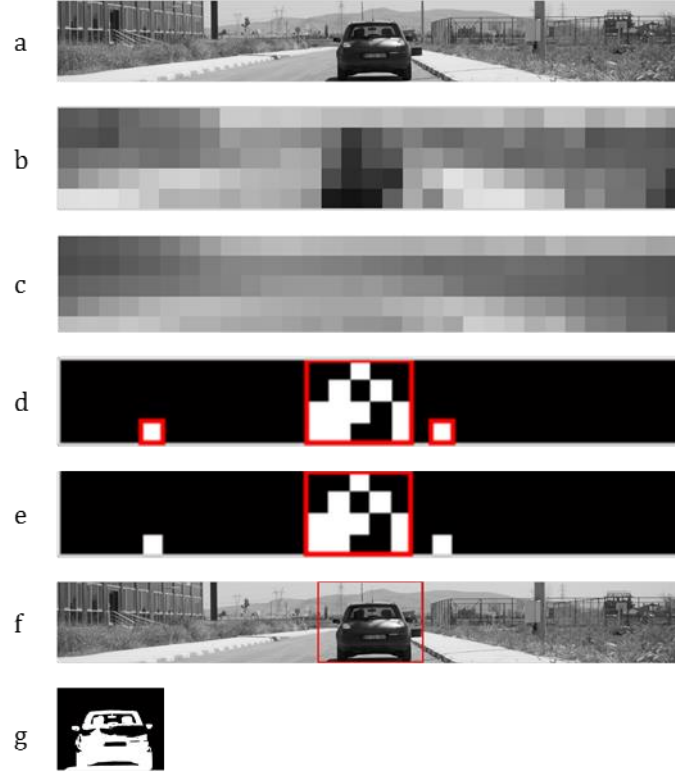


**Şekil 4.9** Uzak mesafeden hareketli nesne silüetinin elde edilmesi sürecinde oluşturulan imge örnekleri.

Yakın mesafede bulunan nesnelere geliştirilen sistem tarafından daha kolay ve yüksek başarıyla tespit edilebilmesine rağmen bir sonraki aşama olan nesne tanıma aşamasında başarıyı azaltmaktadır. Çünkü nesne yaklaştıkça tanımda kullanılacak bir takım ayrıntılar yok olmaktadır. Nesne uzaklaştıkça Şekil 4.9 (d) ve (e) bölümlerinden de anlaşılacağı gibi ikilik görüntülerde gürültü miktarı artmaktadır. (f) bölümünde verilen orijinal görüntü üzerindeki çerçeve içine alınmış nesne resminde ise çerçevenin nesnenin tam sınırlarından değil biraz daha geniş olarak çizildiği görülmektedir. Bu durum uzaklık nedeniyle artan gürültü sebebiyle oluşmuştur. Ancak (g) bölümünde elde edilen nesne silüeti morfolojik işlemler aracılığıyla net olarak elde edilmiştir. Bu durumda tanıma aşamasında ise silüet tam sınırlarından çerçeve içine alınabilmiştir (kutulanabilmiştir.).

Bu deneme çalışmasında geçiş aralığı mesafesi 1 m olarak alınmıştır. Geçiş aralığı mesafesinin artması ve nesnenin kameradan uzaklaşmasına rağmen nesne silüeti başarılı olarak çıkarılabilmektedir. Şekil 4.10'da sonuçları verilen deneme çalışmasında nesne olarak bir otomobil kullanılmıştır.

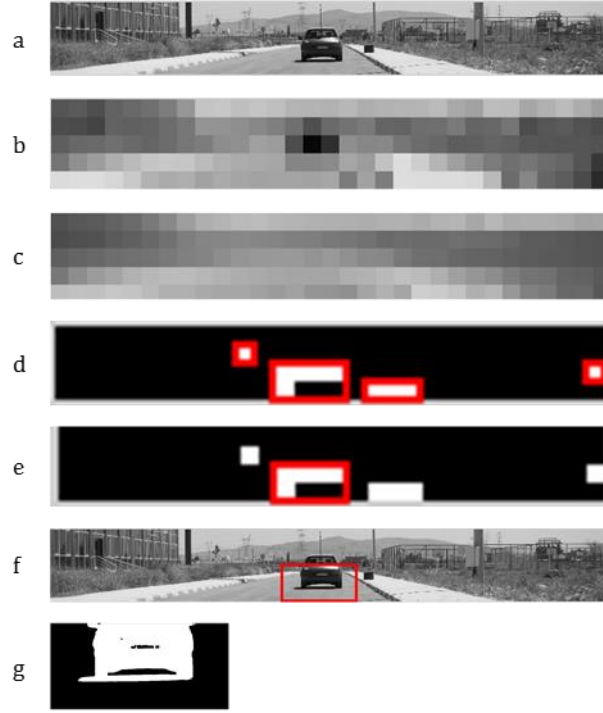




**Şekil 4.10** Hareketli nesne olarak otomobil kullanıldığında elde edilen imge örnekleri.

Burada otomobilin  $K_1$  koordinatından 10 m uzaklıkta bulunmaktadır. Otomobiller yayalara göre daha büyük boyutlu ve rijit yapıda oldukları için yayalara göre daha uzaktan doğru olarak tespit edilebilmektedir. Yayalar ise esnek yapıda olduklarından video akışı içerisinde bazı imge çerçevelerinde normal karakteristik görünümünden daha farklı şekillerde yakalanabilirler. Özellikle kamera kaynağına uzak bir noktada bulunduğu yine tespit edilebilmekte ancak sonraki aşamalarda tanıma işlemlerinde başarı oranı düşmektedir. Şekil 4.11’de kamera kaynağına 20 m uzaklıkta bulunan bir otomobil içeren imge çerçevesi ile yapılan deneysel çalışmanın görüntüleri verilmiştir.

Bu deneysel çalışmada; (d) bölümünde elde edilen piksel öbeklerinin boyutlarının ve sayılarının arttığı dikkat çekmektedir. Bu durumu engellemek için eşik değeri azaltıldığında ise hareketli nesne ile gürültü nedeniyle oluşan bloklar ayırt edilememektedir. Otomobil ve yayaların kamera kaynağına uzaklıkları arttıkça tespit ve tanıma işlemlerindeki başarı oranı düşmektedir.



**Şekil 4.11** Hareketli nesne olarak 20 m uzaklıkta bir otomobil kullanıldığında elde edilen imge örnekleri.

Şekil 4.11 (f) bölümünde otomobilin alt kısmının tespit edilebildiği tamamının tespit edilemediği görülmüştür. Otomobilin alt kısmındaki renk zıtlığının daha fazla olması, üst kısmında ise camların şeffaflığı nedeniyle arka plan ile yeterince zıtlık oluşturamaması bu şekilde yarım tespitin başlıca nedenidir. Ancak; yine de bu çalışmanın amacı olan yabancı nesne tespiti gerçekleştirilebilmiştir. (g) bölümünde tespit edilen kısım net bir şekilde ikilik görüntü olarak verilmiştir.

Bu ölçülerde tespit için Şekil 4.8, Şekil 4.9, Şekil 4.10 ve Şekil 4.11'in (d) bölümlerinde verilen kümelenmiş piksel öbekleri en az 2 blok kalıncaya kadar nesnelere kamera lensinden aşamalı olarak uzaklaştırılmıştır. 2 bloktan daha az olan kümeler tespit için yeterli görülmeyip ihmal edilmiştir. Tespit için yeterli görülen en uzak mesafede en fazla bloğa sahip sadece bir öbeğin kalmasına dikkat edilmiştir.

Ayrıca nesnelere doku renk tonları da tespit işleminde önemli bir rol oynadığı görülmüştür. Renkler arka plan renkleriyle ne kadar çok kontrast oluşturursa tespit başarısı o kadar artmaktadır. Bu tip durumlarda ikilik görüntü olarak tespit edilen silüetler daha net olarak elde edilebilmesi için eşik değerin daha hassas olarak belirlenmesi

gerekmektedir. Aynı şekilde siluetin elde edilmesi için yapılan morfolojik işlemlerde seçilen parametreler hassas bir şekilde belirlenmelidir.

Şekil 4.8, Şekil 4.9, Şekil 4.10 ve Şekil 4.11'in (g) bölümünde elde edilen ikilik görüntü için bir eşik değeri kullanılmalıdır. Bu eşik değeri ne kadar doğru seçilirse elde edilen siluet o kadar net olmaktadır. Ayrıca ortamdaki ışık seviyesine göre değiştirilmesi gerekir. Bu çalışmada eşik değeri doğru olarak belirlenebilmesi için mobil araç üzerinde ışık şiddeti ölçüm sensörü kullanılmıştır.

Yapılan deneme çalışmalarında 10 m, 20 m ve 25 m uzaklıktaki otomobil ve yayalar Yöntem-3 ile tespit edilmeye çalışılmıştır. Her bir uzaklık için 1 m ve 2 m geçiş aralıklarında 27 adet deneysel çalışma yapılmış ve elde edilen sonuçlar Çizelge 4.11 ve Çizelge 4.12'de verilmiştir.

**Çizelge 4.11** Yayalar ile 2 m geçiş aralığında yapılan deneme çalışması başarı oranları.

Nesne	Geçiş aralığı (m)	Uzaklık (m)	Başarı Oranı (%)
Yaya	2	10	96,3
Yaya	2	20	70,3
Yaya	2	25	62,9

**Çizelge 4.12** Yayalar ile 1 m geçiş aralığında yapılan deneme çalışması başarı oranları.

Nesne	Geçiş aralığı (m)	Uzaklık (m)	Başarı Oranı (%)
Yaya	1	10	96,3
Yaya	1	20	77,7
Yaya	1	25	66,6

Otomobiller ise her bir uzaklık değeri yine 1 m ve 2 m geçiş aralıklarında için 32 adet deneme çalışması yapılmış ve elde edilen sonuçlar Çizelge 4.13 ve Çizelge 4.14'de verilmiştir.

Hem yayalarla hem de otomobillerle yapılan çalışmalarda 20 m ile 25 m arasında başarı oranlarında hızlı bir düşüş görülmektedir. Bu gözlem neticesinde özellikle yayalar için 20 m sonrasında sağlıklı tespit yapılamayacağı otomobiller için ise ortalama bir tespit başarısının olacağı belirlenmiştir. Geçiş aralığı mesafelerinin ise azaldıkça tespit başarısını olumlu yönde etkilediği görülmüştür.

**Çizelge 4.13** Otomobiller ile 2 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları.

Nesne	Geçiş aralığı (m)	Uzaklık (m)	Başarı Oranı (%)
Otomobil	2	10	100
Otomobil	2	20	87,5
Otomobil	2	25	75

**Çizelge 4.14** Otomobiller ile 1 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları.

Nesne	Geçiş aralığı (m)	Uzaklık (m)	Başarı Oranı (%)
Otomobil	1	10	100
Otomobil	1	20	90,6
Otomobil	1	25	78,1

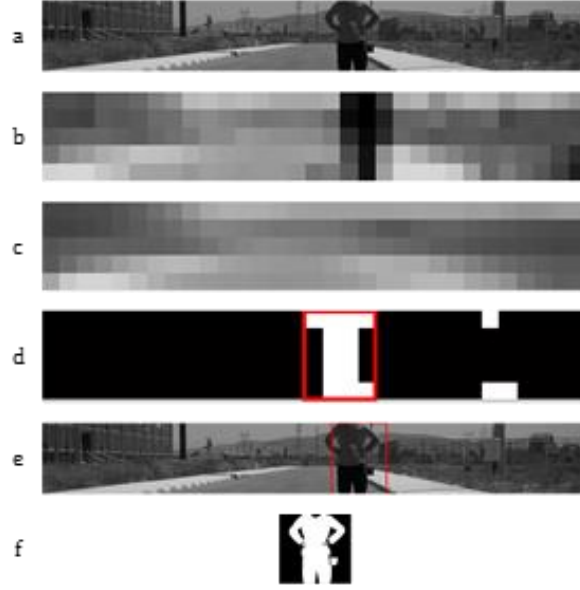
Sistemin çalışma hızı ile ilgili denemelerde öncelikle bilgisayarın CPU kullanım oranı gözlemlenmiştir. Deneme çalışmalarında 4 çekirdekli 2.8 GHz işlemciye sahip 4 GB RAM kapasiteli bir bilgisayar kullanılmıştır. Elde edilen sonuçlar Çizelge 4.15 'de verilmiştir.

**Çizelge 4.15** Yöntemlerin bilgisayar sistemi performans karşılaştırması.

Yöntem	İmge çözünürlüğü	CPU yükü (%)	Çerçeve başına işlem süresi (sn)
GD+ ÇF	3264x4928	%99	5,475
GD + Gauss	3264x4928	%99	5,91
GY	3264x4928	%4	0,022
	480x640	%3	0,019

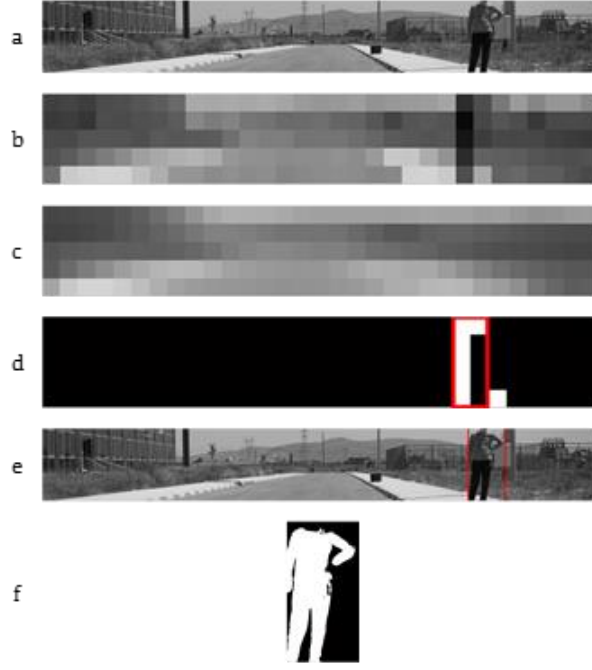
#### 4.4 Yöntem 4: GPS Destekli Özelleştirilmiş Arka Plan Modellemesi

Yöntem-4'ü test etmek amacıyla Şekil 3.13'de belirtilen  $K_1$  ile  $K_{20}$  arası referans noktalara farklı mesafelerde bulunan nesnelere ile deneysel çalışmalar yapılmıştır. Bu deneysel çalışmada  $K_1$  koordinatına 10 m uzaklıkta bulunan bir yayanın arka planda bulunduğu bir imge çerçevesi kullanılmıştır. Çalışma sonucunda işlem basamaklarında elde edilen imge çerçeveleri ve ara değerlerin görünüşleri Şekil 4.12'de verilmiştir. Ara değerler görüntünün dörtgensel bloklara ayrılması ve yabancı nesne tespit işlemleri için gerçekleştirilen görüntü işleme adımları sırasında elde edilen görüntü çerçeveleridir. Nesne içeren görüntü araca yakın olduğu için yaya silüetinin baş kısmı tam olarak görüntülenememiştir.



**Şekil 4.12** Yakın bir mesafedeki hareketli nesnenin silüetinin elde edilmesi sürecinde oluşturulan imge örnekleri

Şekil 4.12 (a) bölümünde nesne içeren orijinal görüntü, (b) bölümünde bu görüntüye ait bloklara ayrılmış mozaik görüntü, (c) bölümünde  $K_1$  koordinatından alınan referans imgelerin ortalaması ( $R_1$ ), (d) bölümünde model ile anlık imgenin karşılaştırılması sonucu elde edilen piksel öbeklerinin kümelenmiş görüntüsü, (e) bölümünde sınır çizgilerinin dönüştürülmesi sonucu orijinal görüntü üzerindeki izdüşümleri, (f) bölümünde ise sınır çizgilerinin arasında kalan alan içinde bulunan nesnenin ikilik görüntüye çevrilmiş silüeti verilmiştir. Özellikle (d) bölümünde görülmekte olan bir blok boyutlu küçük öbekler ihmal edilmiştir. Yapılan deneme çalışmalarında genellikle bu tür durumların oldukça uzak mesafede bulunan hareketli nesnelere ya da arka planda oluşan gürültülerden kaynaklandığı belirlenmiştir. Bu deneme çalışmasında referans imgeler 1 m geçiş aralığında oluşturulmuştur. Geçiş aralığı mesafeleri daraldıkça referans imgeler arasındaki fark azaldığı için ortalama imgenin doğru arka planı yansıtırma oranı artmaktadır. Yapılan çalışmalarda elde edilen sonuçlar da bunu doğrular niteliktedir. Yakın mesafede bulunan nesnelere geliştirilen sistem tarafından daha kolay ve yüksek başarıyla tespit edilebilmesine rağmen bir sonraki aşama olan nesne tanıma aşamasında başarıyı azaltmaktadır. Çünkü nesne yaklaştıkça tanımada kullanılacak bir takım ayrıntılar yok olmaktadır. Şekil 4.13’de yine  $K_1$  koordinatından 20 m uzaklıkta bulunan bir yaya içeren anlık imge çerçevesi ile yapılan çalışma sonucunda elde edilen görüntüler verilmiştir.

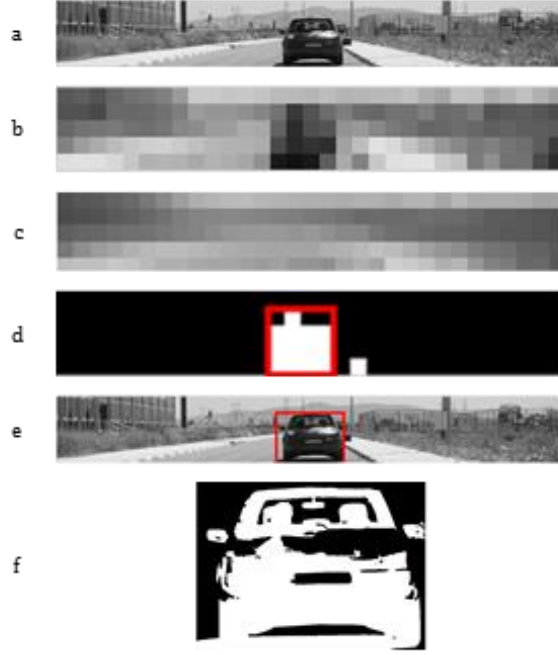


**Şekil 4.13** Uzak bir mesafeden hareketli nesne silüetinin elde edilmesi sürecinde oluşturulan imge örnekleri.

Otomobil ile yapılan bu deneme çalışmasında geçiş aralığı mesafesi 2 m olarak alınmıştır. Geçiş aralığı mesafesinin artması ve nesnenin kameradan uzaklaşmasına rağmen nesne silüeti başarılı olarak çıkarılabilmektedir. Şekil 4.14’de verilen deneme çalışmasında  $K_1$  koordinatından 10 m uzaklıkta bir otomobilin bulunduğu imge çerçevesi anlık imge olarak alınmıştır.

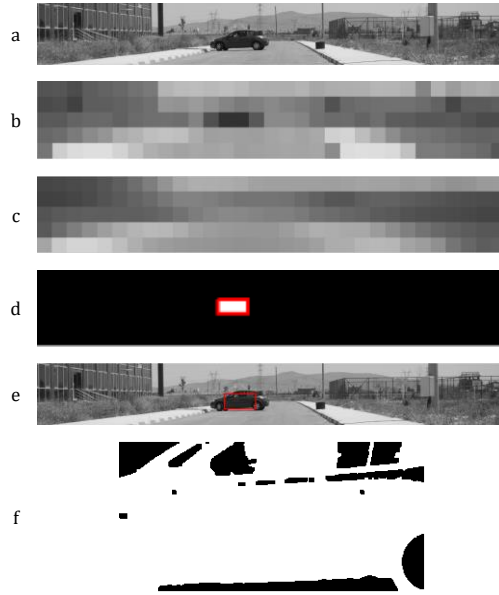
Otomobiller yayalara göre daha büyük boyutlu ve rijit yapıda oldukları için yayalara göre daha uzaktan doğru olarak tespit edilebilmektedir. Yayalar ise esnek yapıda olduklarından video akışı içerisinde bazı imge çerçevelerinde normal karakteristik görünümünden daha farklı şekillerde fotoğraflanabilmektedirler. Bu durum esnek yapıda olan yayaların varsayılan arka plandan ayrıştırılmak suretiyle tespitini zorlaştırmaktadır.

Ayrıca tespit işlemlerinden sonra yapılacak olan tanıma işlemleri de zorlaşmaktadır. Dolayısıyla nesne tanıma işlemleri için oluşturulacak olan yapay sinir ağlarında esnek yapıda olan nesnelerin yüksek doğrulukta tanınabilmesi için daha fazla figür ile eğitime tabi tutulmaları gerekmektedir.



**Şekil 4.14** Yakın mesafede hareketli nesne olarak otomobil kullanıldığında elde edilen imge örnekleri.

Özellikle kamera kaynağına uzak bir noktada bulduklarında yine tespit edilebilmekte ancak sonraki aşamalarda tanıma işlemlerinde başarı oranı düşmektedir. Şekil 4.15’de kamera kaynağına 30 m uzaklıkta bulunan bir otomobil içeren imge çerçevesi ile yapılan deneysel çalışmanın görüntüleri verilmiştir.



**Şekil 4.15** Uzak mesafede bir otomobil kullanıldığında elde edilen imge örnekleri.

Bu deneyle çalışmada; (d) bölümünde elde edilen piksel öbeklerinin boyutlarının ve sayılarının azaldığı dikkat çekmektedir. Bu durumu engellemek için eşik değeri azaltıldığında ise hareketli nesne ile gürültü nedeniyle oluşan bloklar ayırt edilememektedir. Otomobil ve yayaların kamera kaynağına uzaklıkları arttıkça tespit ve tanıma işlemlerindeki başarı oranı düşmektedir. Burada (e) bölümünde otomobilin alt ve orta kısmının tespit edilebildiği tamamının tespit edilemediği görülmüştür. Otomobilin alt kısmındaki renk zıtlığının daha fazla olması, üst kısmında ise camların şeffaflığı nedeniyle arka plan ile yeterince zıtlık oluşturamaması bu şekilde yarım tespitin başlıca nedenidir. Ancak; yine de bu çalışmanın amacı olan yabancı nesne tespiti gerçekleştirilebilmiştir. (f) bölümünde tespit edilen kısım net bir şekilde ikilik görüntü olarak verilmiştir.

Geliştirilen yöntemde; iki bloktan daha az olan kümeler tespit için yeterli görülmeyip ihmal edilmiştir. Sahnedeki en fazla alana sahip olan piksel öbeği yabancı nesne olarak kabul edilmiştir.

Yapılan deneme çalışmalarında 10 m, 20 m ve 30 m uzaklıktaki otomobil ve yayalar geliştirilen yaklaşım ile tespit edilmeye çalışılmıştır. Her bir uzaklık için 1 m ve 2 m geçiş aralıklarında 33 adet deneme yapılmış ve elde edilen sonuçlar Çizelge 4.16 ve Çizelge 4.17’de verilmiştir.

**Çizelge 4.16** Yayalar ile 1 m geçiş aralığında yapılan deneme çalışması başarı oranları.

Nesne	Uzaklık (m)	Başarı Oranı (%)
Yaya	10	93,9
Yaya	20	75,7
Yaya	30	60,6

**Çizelge 4.17** Yayalar ile 2 m geçiş aralığında yapılan deneme çalışması başarı oranları.

Nesne	Uzaklık (m)	Başarı Oranı (%)
Yaya	10	90,9
Yaya	20	72,7
Yaya	30	60,6

Otomobiller ise her bir uzaklık değeri yine 1 m ve 2 m geçiş aralıklarında için 33 adet deneme çalışması yapılmış ve elde edilen sonuçlar Çizelge 4.18 ve Çizelge 4.19’da verilmiştir



**Çizelge 4.18** Otomobiller ile 1 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları.

Nesne	Uzaklık (m)	Başarı Oranı (%)
Otomobil	10	100
Otomobil	20	90,9
Otomobil	30	69,7

**Çizelge 4.19** Otomobiller ile 2 m geçiş aralığında yapılan deneme çalışmaları başarı sonuçları.

Nesne	Uzaklık (m)	Başarı Oranı (%)
Otomobil	10	100
Otomobil	20	84,8
Otomobil	30	66,6

Hem yayalarla hem de otomobillerle yapılan çalışmalarda 20 m ile 30 m arasında başarı oranlarında hızlı bir düşüş gözlemlenmektedir. Bu gözlem neticesinde yayalar ve otomobiller için 30 m sonrasında sağlıklı tespit yapılamayacağı belirlenmiştir. Geçiş aralığı mesafelerinin azalmasının tespit başarısını olumlu yönde etkilediği görülmüştür. Sistemin çalışma hızı ile ilgili denemelerde öncelikle bilgisayarın CPU kullanım oranı gözlemlenmiştir.

**Çizelge 4.20** Yöntemlerin bilgisayar sistemi performans karşılaştırması.

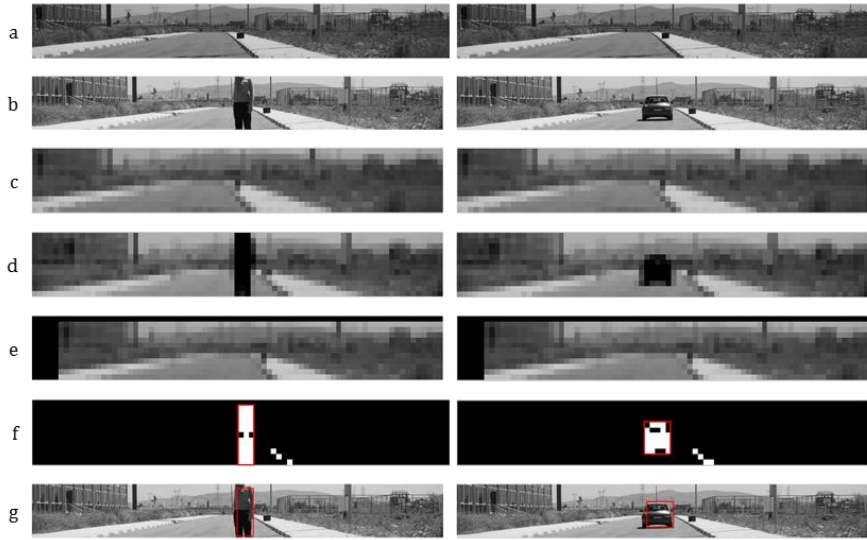
Yöntem	İmge çözünürlüğü	CPU yükü (%)	Çerçeve başına işlem süresi (sn)
GD+ ÇF	3264x4928	%99	5,475
GD + Gauss	3264x4928	%99	5,91
GY	3264x4928	%4	0,021
	480x640	%3	0,019

Geliştirilen yöntemin hız ve gerektirdiği işlem maliyeti açısından avantajlı olduğu çizelgede belirtilen değerlerden anlaşılmaktadır.

#### 4.5 Yöntem 5: Özelleştirilmiş İmge Çakıştırma Yöntemi

Geliştirilen yöntemi test etmek amacıyla Şekil 3.13’de belirtilen  $K_1$  ile  $K_{20}$  arası koordinatlara farklı mesafelerde bulunan nesnelere ile deneysel çalışmalar yapılmıştır. Rota üzerinde yapılan deneme çalışmalarında yabancı nesne olarak kameraya yakın ve uzak noktalarda konumlandırılan yaya ve otomobiller kullanılmıştır. Referans imgeler ise 1m ve 2m geçiş aralıklarından fotoğraflanmıştır. Mobil araç ilerleyişi sırasında söz konusu koordinatların her birine ulaştığında bir anlık görüntü çerçevesi alınmıştır. Sistem her bir  $K_i$ ’de yakaladığı anlık görüntüden elde ettiği anlık model ( $A_i$ ) ile hafızasında

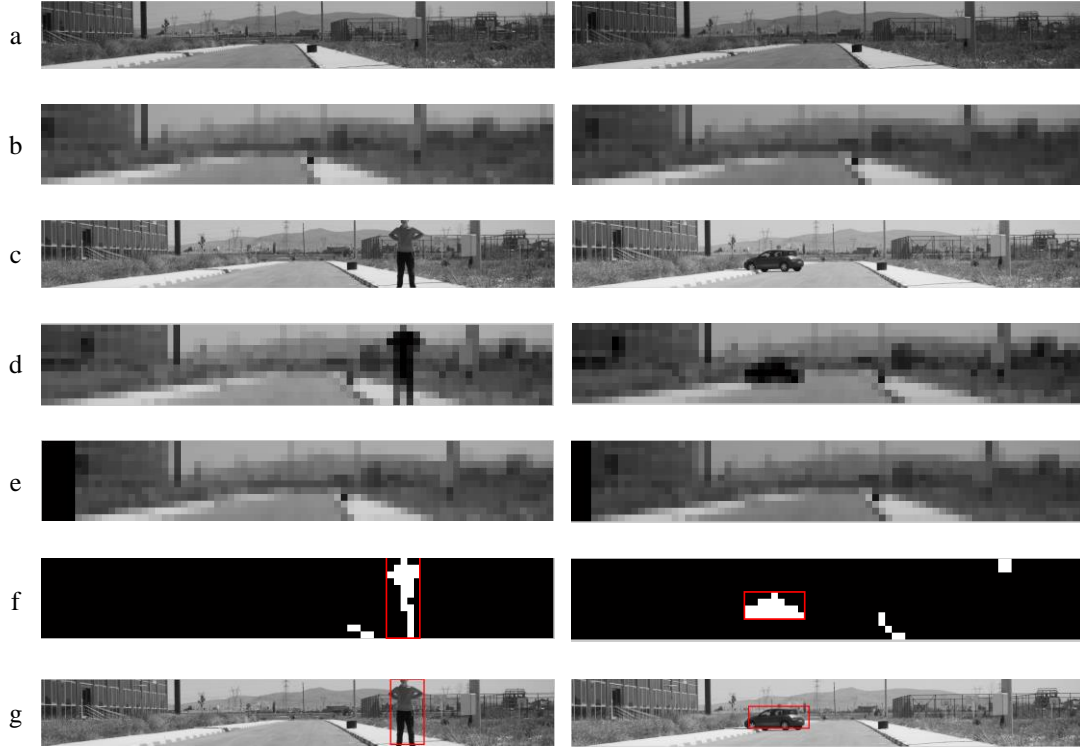
kayıtlı referans model ( $R_i$ ) arasında karşılaştırma yaparak kayma vektörünü hesaplamıştır. Kayma vektörü kullanılarak referans ve anlık imge çerçeveleri arasında hizalama yapılmış ve çerçeve farkı yöntemiyle nesnenin silueti bulunmuştur. Siluetler mozaik görüntü olarak elde edilmiştir.  $K_1$ 'e 10 m uzaklıkta bir yaya ve  $K_1$ 'e 20 m uzaklıkta bir otomobilin arka plana dahil olduğu imge çerçeveleri ile yapılan çalışma sonucunda işlem basamaklarında elde edilen görünüm Şekil 4.16'da verilmiştir. Deneme çalışmasında kullanılan referans imgeler 2 m geçiş aralığı mesafesinden fotoğraflanmıştır.



Şekil 4.16  $K_1$  koordinatından yakın mesafe yaya ve otomobil için elde edilen sonuçlar.

Burada, (a) bölümünde yalın arka planı içeren referans imgenin orijinal görüntüleri, (b) bölümünde nesne içeren orijinal anlık görüntü, (c) bölümünde bloklara ayrılmış mozaik görüntü formunda referans imge, (d) bölümünde mozaik görüntü formunda anlık imge, (e) bölümünde hesaplanan kayma vektörü kullanılarak ötelenen referans imge, (f) bölümünde çerçeve farkı yöntemi ile elde edilen mozaik siluet, (g) bölümünde ise siluetin sınır çizgilerinin dönüştürülmesi sonucu orijinal görüntü üzerindeki izdüşümleri verilmiştir. (f) bölümünde gri seviye farklarının en yüksek olduğu blokların yabancı bir nesneyi işaret edebilmeleri için bir arada anlamlı bir şekilde kümelenmiş olmaları gereklidir. Bu kümelerin ayırt etme işlemi CCL yöntemi ile yapılmıştır.  $K_3$ 'e 20 m uzaklıkta bir yaya ve  $K_3$ 'e 30 m uzaklıkta bir otomobilin arka plana dahil olduğu imge çerçeveleri ile yapılan çalışma sonucunda işlem basamaklarında elde edilen görünüm

Şekil 4.17’de verilmiştir. Deneme çalışmasında kullanılan referans imgeler 1 m geçiş aralığı mesafesinden fotoğraflanmıştır.



Şekil 4.17  $K_3$  koordinatından uzak mesafe yaya ve otomobil için elde edilen sonuçlar.

Yapılan deneme çalışmalarında 10 m, 20 m ve 30 m uzaklıktaki yayalar ve otomobiller geliştirilen yaklaşım ile tespit edilmeye çalışılmıştır. Her bir uzaklık için 1 m ve 2 m geçiş aralıklarında 40 adet deneme yapılmış ve elde edilen sonuçlar Çizelge 4.21 ve Çizelge 4.22’de verilmiştir.

Çizelge 4.21 1 m geçiş aralığından elde edilen başarılı tespit oranları.

Nesne	Uzaklık (m)	Başarı Oranı (%)
Yaya	10	95
Yaya	20	72,5
Yaya	30	55
Otomobil	10	100
Otomobil	20	82,5
Otomobil	30	67,5

Yayalarla ve otomobillerle yapılan çalışmalarda 20 m ile 30 m arasında başarı oranlarında hızlı bir düşüş gözlemlenmektedir. Bu gözlem neticesinde yayalar ve otomobiller için

30m sonrasında sağlıklı tespit yapılamayacağı belirlenmiştir. Geçiş aralığı mesafelerinin azaldıkça tespit başarısını olumlu yönde etkilediği görülmüştür.

**Çizelge 4.22** 2 m geçiş aralığından elde edilen başarılı tespit oranları.

Nesne	Uzaklık (m)	Başarı Oranı (%)
Yaya	10	90
Yaya	20	70
Yaya	30	52,5
Otomobil	10	95
Otomobil	20	80
Otomobil	30	67,5

Sistemin çalışma hızı ile ilgili denemelerde öncelikle bilgisayarın CPU kullanım oranı gözlemlenmiştir. Elde edilen sonuçlar Çizelge 4.23’de verilmiştir.

**Çizelge 4.23** Farklı metotların bilgisayar üzerindeki performansları.

Yöntem	İmge çözünürlüğü	CPU yükü (%)	Çerçeve başına işlem süresi (sn)
GD+ ÇF	3264x4928	%99	5,475
GD + Gauss	3264x4928	%99	5,91
GY	3264x4928	%11	0,21
	480x640	%10	0,19

Yöntem-5’de CPU yükü ve çerçeve başına işlem süresinin diğer yöntemlere göre daha yüksek olduğu görülmüştür. Bu yöntemde öznitelik elde etmek için bloklanmış görüntülerin birinci ve ikinci türevlerinin kullanılması hesaplama maliyetini arttırmıştır. Ayrıca daha net öznitelik elde edilmesi için karesel blokların boyutlarının küçültülerek referans model ve anlık model matrislerinin eleman sayılarını artması da bu sonuçların alınmasında etkili olmuştur.

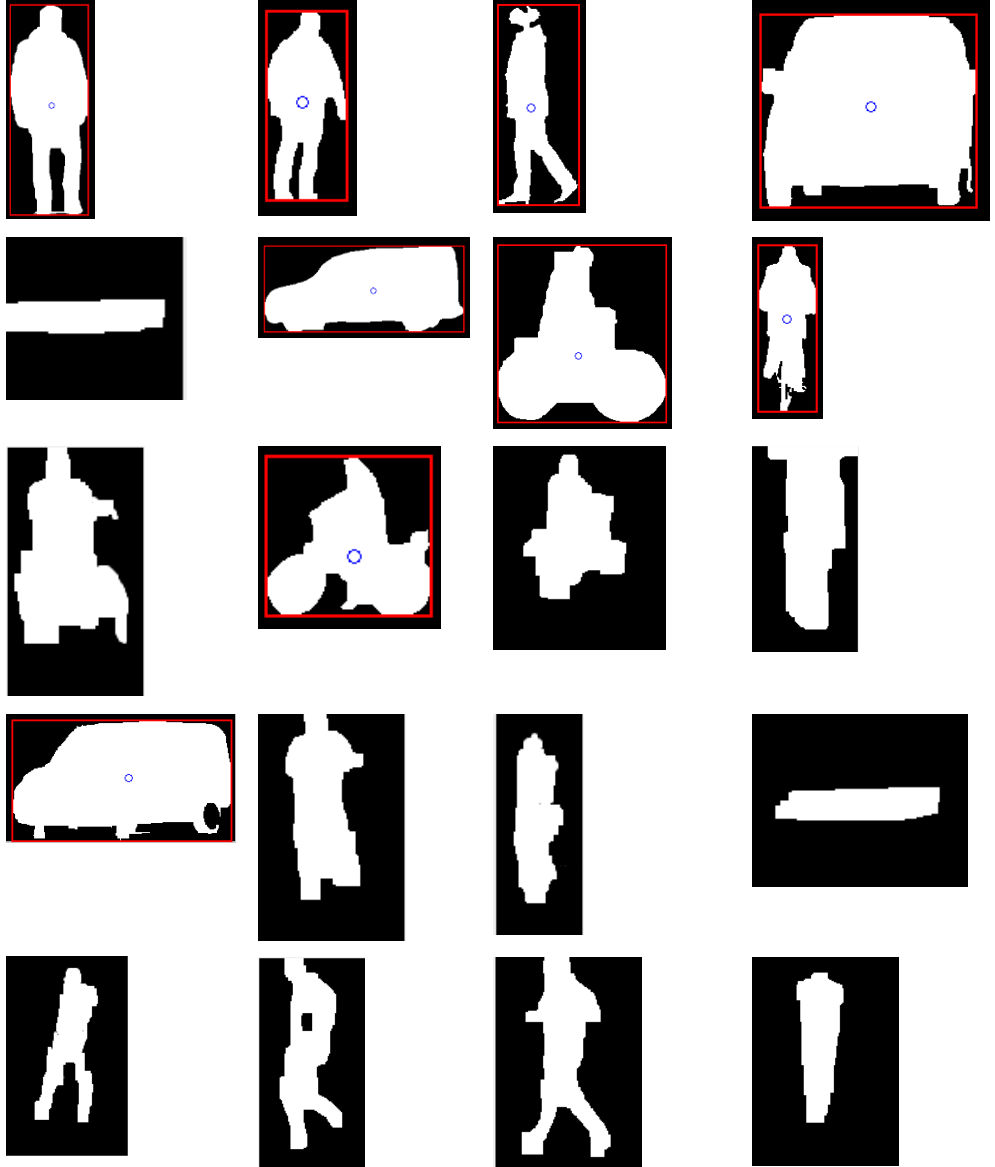
#### 4.6 Nesne Tanıma İşlemleri

YSA’nın eğitim çalışmasında yaya kategorisi için 100 adet, otomobil ve bisikletli kategorisi için 80 adet imge kullanılmıştır. Yayaların esnek yapılarından dolayı YSA daha fazla örnek veri ile eğitilmiştir. Otomobiller ve bisikletliler daha rijit yapıda olmaları nedeniyle daha az sayıda örnek veri kullanılmıştır.

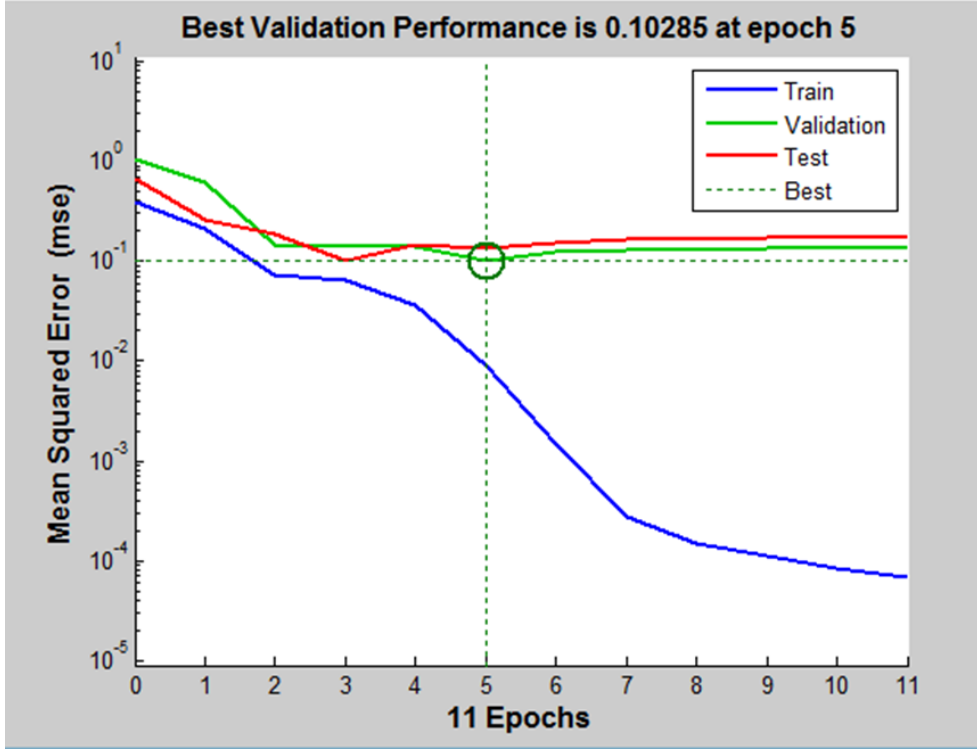
Yapılan deneysel çalışmalarda alınan sonuçlarda tasarlanan ađın ara katmanında 50 adet iřlem elemanı kullanılmasına ve aktivasyon fonksiyonu olarak sigmoid fonksiyonu kullanılmasına karar verilmiřtir.

Ađın giriřlerinde kullanılmak üzere 16 adet 32x32 pikselli g1zetelik vekt1rleri oluřturulan bir kısıml deneme imgeleri 1izelge 4.24'de verilmiřtir.

**1izelge 4.24** Bir kısıml deneme imgelerinin siluet haline d1nuřt1r1lm1ř durumlari.



Ađın eđitilmesi sonucunda Matlab yazılımından elde edilen hata performansı grafiđi Őekil 4.18'de verilmiřtir.



Şekil 4.18 Eğitim, doğrulama ve test performans grafiği.

YSA'nın denenmesi için her kategoriden olarak tespit edilmiş ve net silüetler haline dönüştürülmüş 30 adet imge ağın girişine uygulanmış ve Çizelge 4.25'de belirtilen sonuçlara ulaşılmıştır.

Çizelge 4.25 YSA'nın kategorilere göre sınıflandırma başarı oranları.

Kategori	Başarı (%)
Yaya	90
Otomobil	96,66
Bisikletli	90

Alınan sonuçlara göre en yüksek başarı rijit yapıda olan otomobillerde elde edilmiştir. Yaya ve bisikletli nesnelere yaşanan daha düşük tanıma oranı bu nesnelere esnek yapılarından kaynaklanmaktadır. Bu zorluk YSA'nın eğitiminde daha fazla sayıda ve farklı pozisyonlarda yaya ve bisikletli silüetinin kullanılması ile bir miktar aşılabilsede esnek yapıda olan bu nesnelere alabileceği tüm pozisyonlar YSA üzerinde eğitilemediği için başarı oranı düşük kalmıştır.

## 5. TARTIŞMA ve SONUÇ

Nesne tespit ve tanıma çalışmalarında tespit edilecek nesnelerin yapıları büyük önem taşımaktadır. Rijit yapıda olan nesnelere daha kolay tespit edilebilirken esnek yapıda olan nesnelerin tespitinde zorluklar ile karşılaşmaktadır. Bu nedenle deneysel çalışmalarda; rijit yapıları test etmek amacıyla otomobiller, esnek yapıları tespit etmek amacıyla da insanlar kullanılmıştır.

Bu çalışmada nesne tespiti için beş farklı yöntem kullanılmıştır. Her yöntemin başarısının tespiti için 3 farklı ölçüt belirlenmiştir. Bunlar;

- Nesne varlığının doğru olarak tespiti,
- Nesne tanımaya elverişli net silüet belirleme,
- Bilgisayar sistemlerinde düşük hesaplama maliyetidir.

Yöntem-1 ve Yöntem-2’de %85 başarı oranı temel alınarak yaya ve otomobiller için en uzak tespit mesafeleri bulunmuştur. Yöntem-3, Yöntem-4 ve Yöntem-5 için ise nesnelere belirli uzaklıklara konumlandırılarak belirli sayıda denemeler yapılmış ve bu denemelerde başarılı olan sonuçların oranları ölçülmüştür. Aynı ölçme tiplerinin kullanıldığı Yöntem-1 ve Yöntem-2’de 3264x4928 çözünürlükte 1 m geçiş aralığı mesafesinden elde edilen sonuçların belirlenen ölçütlere göre karşılaştırmalı listesi Çizelge 5.1’de verilmiştir.

**Çizelge 5.1** 3264x4928 çözünürlükte Yöntem-1 ve Yöntem-2 performans karşılaştırması.

	<b>Yaya En Uzak Mesafe</b>	<b>Otomobil En Uzak Mesafe</b>	<b>CPU Yüğü(%)</b>	<b>Çerçeve başına işlem süresi (sn)</b>
Yöntem-1	13 m	20 m	4	0,02
Yöntem-2	14 m	22 m	7	0,043

Yukarıdaki çizelgeden anlaşılacağı gibi Yöntem-2’nin Yöntem-1’e göre daha başarılı sonuçlar ürettiği ancak bilgisayar sistemleri açısından hesaplama maliyetinin daha yüksek olduğu görülmektedir. İKA’nın düşük hızlarda devriye yapacağı düşünüldüğünde Yöntem-2’nin çerçeve başına işlem süresi herhangi bir sorun teşkil etmeyecektir. Bu sonuçlara göre Yöntem-1’in bilgisayardaki işlem hızı olarak daha hızlı sonuca ulaştığı ancak Yöntem-2’nin ise daha uzaktan doğru nesne tespiti yapabildiği tespit edilmiştir.

Yöntem-3, Yöntem-4 ve Yöntem-5’de de aynı ölçme teknikleri kullanılarak nesne tespit performansları belirlenmiştir. Alınan sonuçlar 1 m geçiş aralığından ölçülen ve 10 m uzakta konumlandırılan yaya ve otomobiller için alınmıştır. Söz konusu sonuçların karşılaştırmalı listesi Çizelge 5.2’de verilmiştir.

**Çizelge 5.2** 3264x4928 çözünürlükte Yöntem-3, Yöntem-4 ve Yöntem-5 performans karşılaştırmaları

	<b>Yaya Başarı Oranı(%)</b>	<b>Otomobil Başarı Oranı(%)</b>	<b>CPU Yüğü(%)</b>	<b>Çerçeve başına işlem süresi (sn)</b>
Yöntem-3	96,3	100	4	0,022
Yöntem-4	93,9	100	4	0,021
Yöntem-5	95	100	11	0,21

Bu listeden anlaşıldığı üzere Yöntem-3’ün 10 m mesafedeki yaya ve otomobilleri diğer yöntemlerde göre daha başarılı tespit etmektedir. Ayrıca Yöntem-3 ve Yöntem-4’ün hesaplama maliyetlerinin yaklaşık olarak aynı olmasına rağmen Yöntem-5’de CPU yükü ve çerçeve başına işlem süresinin diğer yöntemlere göre oldukça yüksek olduğu ancak performansının kabul edilebilir seviyelerde olduğu hatta Yöntem-4’den daha iyi olduğu gözlemlenmiştir.

Bu yöntemlerde otomobiller ve yayalar için 20 m ve 30 m uzaklıktaki nesnelere daha az başarı oranları ile tespit edilebilmektedir. Bu durumun sebepleri arasında; mesafe arttıkça nesnelere görüntü hacminin düşmesi, gölge, ağaç sallanma hareketleri ve ışık şiddeti farklılıkları gibi gürültü oluşturan sebeplerin etkisinin mesafe arttıkça yükselmesi gösterilebilir.

Görüntü çözünürlüklerinin düşürülmesi kullanılan kameranın maliyetini düşürdüğü için önemlidir ancak yüksek çözünürlükte elde edilen sonuçların yanında kabul edilebilir seviyelerde sonuç elde edilmesi gerekir. Ayrıca görüntü piksel sayılarındaki azalma nedeniyle hesaplama maliyetleri de düşecektir. 1 m geçiş aralığından ölçülen ve 10 m uzakta konumlandırılan yayalar için daha düşük çözünürlükte (480x640) yapılan çalışmalarda gözlemlenen sonuçlar ise Çizelge 5.3 ve Çizelge 5.4’de verilmiştir.

**Çizelge 5.3** 480x640 çözünürlükte Yöntem-1 ve Yöntem-2 performans karşılaştırması.

	<b>Yaya En Uzak Mesafe</b>	<b>Otomobil En Uzak Mesafe</b>	<b>CPU Yüğü(%)</b>	<b>Çerçeve başına işlem süresi (sn)</b>
Yöntem-1	11 m	15 m	3	0,012
Yöntem-2	12 m	18 m	5	0,027



**Çizelge 5.4** 480x640 çözünürlükte Yöntem-3, Yöntem-4 ve Yöntem-5 performansları.

	<b>Yaya Başarı Oranı(%)</b>	<b>Otomobil Başarı Oranı(%)</b>	<b>CPU Yüklü(%)</b>	<b>Çerçeve başına işlem süresi (sn)</b>
Yöntem-3	96,3	100	4	0,022
Yöntem-4	93,9	100	4	0,021
Yöntem-5	95	100	11	0,21

Güvenlik sistemlerinde kameralardan alınan görüntülerin anlık olarak izlenmesi insan faktörüne dayalı dolayısıyla da insan hatasına açık olan bir işlemdir. Bu çalışma ile kameralardan alınan görüntülerin izlenmesine ve kaydedilmesine dayalı güvenlik sistemlerine farklı bir platform kazandırılmıştır. Bu platform kameraları sabit ve pasif olmaktan çıkartıp hareketli hale getirmiştir. Kameralardan uzak olduğu için güvenlik sistemleri tarafından net olarak izlenemeyen konumlara kameraların İKA'ları aracılığıyla yaklaştırılabilmesi sağlanmıştır. Bu şekilde güvenlik tedbirleri sağlamlaştırılabildiği gibi uzak konumlara yeni kameralara olan gereksinim ortadan kaldırılmıştır. Ayrıca sistem üzerinde bulunan yapay zeka unsurları ile güvenlik tedbirleri anlık olarak tespit edilebilmekte, istendiği takdirde geri bildirimde bulunulabilmekte bu şekilde de insan faktörüne bağımlılık azaltılmaktadır. Tüm bu özellikleri ile birlikte geliştirilen sistem İKA'ları üzerine yerleştirildiğinde insansız bir güvenlik devriyesi olarak kullanılabilir. Devriye görevi yapan güvenlik personelinin bir maliyeti bulunmaktadır. Bu görev 7 gün 24 saat çalışma esnasında 4 personel ile verilebileceğinden işletmeler için yıllık bazda ağır bir maliyet teşkil etmektedir. Düşük maliyetli donanımlar ve yazılımlar ile otonom hareket edebilen İKA'ları tarafından gerçekleştirilecek devriye görevleri işletmelere kazanç sağlayacaktır.

Geliştirilen sistem İKA'ları ile birlikte kullanıldığı takdirde; geniş alana yayılmış fabrika alanlarının ya da geniş alana yayılmış devlet kurumlarına ait yerleşkelerin 7 gün 24 saat esasına göre güvenliğinin sağlanması için devriye görevlerinde kullanılabilir. Böylece devriye görevi yapması gereken güvenlik personelinin sayısı azalacaktır. Ayrıca; geliştirilen sistem rota takibi yapabilen insansız hava araçlarında da kullanılabilir.

Sistem ucuz maliyetlidir. Kullanılan ekipman içinde özel bir cihaz kullanılmayıp tüm ekipman piyasada rahatça erişilebilecek malzemelerden seçilmiştir. Örnek olarak kullanılan kamera HD Webcam'dir. Kontrol kartları piyasada yaygın olarak kullanılan

modellerde seçilmiştir. Kullanılan araç bir engelli aracının modifiye edilmesiyle oluşturulmuştur.

Gerçekleştirilen literatür çalışmasında geliştirilen yaklaşımlardan Yöntem-1'in daha önce hiç denenmemiş olduğu anlaşılmıştır. Bu yöntemin başarı oranı her ne kadar diğer yöntemlere göre düşük olsa da hesaplama maliyeti diğerlerine göre daha düşüktür. Yöntem-3, Yöntem-4 ve Yöntem-5'de uygulanan görüntünün dörtgensel bloklara ayırma yöntemi başka çalışmalarda da yapılmış ancak bu tipte bir çalışmada kullanıldığı görülmemiştir. Bloklara ayırma işlemi ile nesne tespiti için oluşturulan modellerin boyutları küçültülmüş dolayısıyla da hesaplama maliyetleri düşmüştür. Piksel piksel işlem yapıldığında oluşacak olan işlem yoğunluğu dörtgensel blokların piksel cinsinden alanları ile ters orantılı olarak azaltılmıştır. Dörtgensel blokların büyüklükleri belirlenirken sahnenin özelliklerini yitirmemesine dikkat edilmiştir. Ayrıca referans ve anlık modelleri oluşturan özniteliklerin belirlenmesinde yoğun hesaplama maliyeti gerektiren yöntemler (SURF, SIFT, Harris vb.) değil birkaç yöntemin bir arada kullanıldığı basit ancak sonuç veren hibrit yöntemler kullanılmıştır. Böylece 2 yönlü bir kazanç elde edilmiş olup bunlar;

- İmge çerçevesi optimal seviyede dörtgensel bloklara ayrılarak resim boyutu ve dolayısıyla model boyutu küçültülerek hesaplama maliyetleri azaltılmıştır,
- Basit öznitelikler hibrit olarak yapılandırılarak hesaplama maliyeti azaltılmıştır.

Bu şekilde yüksek çözünürlükteki görüntülerin işlenip hareketli ya da hareketsiz nesnelerin tespiti için gerekli olan bilgisayar sistemi kapasitesi ve hacmi en aza indirgenmiştir. İhtiyaç duyulan bilgisayar sistemi kapasitesinin düşmesi maliyetleri de düşürdüğü gibi sistem hacmi ve ağırlığının azalması sonucu enerji tüketimi de azalacaktır. Bu çalışma için ileride yapılmak üzere bir kısım geliştirilmeler düşünülmektedir. Bunlar;

- İKA gece görüşü için IR spotlar ve termal kameralar ile desteklenerek gece devriyesine de çıkabilecek duruma getirilebilecektir,
- Nesne tanımda sınıf sayısı artırılarak daha ayrıntılı tanıma yapılabilecektir,
- Çözünürlüğü yüksek kameralara kullanılarak yaya sınıfında yapılan tanımlara yüz tanıma eklenebilecektir,

- Nesne tespitinde sms, e-posta veya siren gibi çeşitli geri bildirim unsurları sisteme eklenebilecektir,
- Sisteme RF telemetry cihazları eklenerek güvenlik birimleriyle entegre olarak çalışması sağlanabilecektir,
- Tespit edilen nesnelere saat tarih bazında kayıt altına alınabilecektir,
- İKA üzerindeki sarsıntı azaltılarak referans modeller tek bir görüntüden elde edilebilecektir. Bu şekilde ön çalışma süresi azaltılabilecektir.
- NI CompactRIO vb. gömülü sistem platformları kullanılarak gerçek zamanlı harekette daha hızlı ve işlevsel sonuçlar alınabilecektir.

## 6. KAYNAKLAR

- Adıyaman, F. (2007). Talep Tahmininde Yapay Sinir Ağlarının Kullanılması, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Al-Khateeb, H., and Petrou, M. (2010). Automatic change detection in an indoor environment. In Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE Computer Society Conference on, San Fransisco, 13-18 June, 53-58
- Alwan, M., Wagner, M. B., Wasson, G., and Sheth, P. (2005). Characterization of infrared range-finder PBS-03JN for 2-D mapping. In Robotics and Automation, ICRA Proceedings of the 2005 IEEE International Conference on, 3936-3941
- Andreopoulos, A., Tsotsos, J.K. (2013). 50 Years of object recognition: Directions forward. *Computer Vision and Image Understanding*, **117**: 827–891
- Atmaca, H., Cetisli, B., and Yavuz, H. S. (2001). The Comparison Of Fuzzy İnference Systems And Neural Network Approaches With ANFIS Method For Fuel Consumption Data. In Second International Conference on Electrical and Electronics Engineering Papers ELECO, Bursa, 7-11 Eylül.
- Baş, N. (2006). Yapay Sinir Ağları Yaklaşımı ve Bir Uygulama.Yüksek Lisans Tezi, Mimar Sinan Güzel Sanatlar Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- Belongie, S., Malik, J and Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**: 509-522
- Benezeth, Y., Jodoin, P. M., Emile, B., Laurent, H. and Rosenberger, C. (2008). Review and evaluation of commonly-implemented background subtraction algorithms. In Pattern Recognition, 2008. ICPR 2008. 19th International Conference on, 1-4
- Bernard, S.T. and Thompson, W.B. (1980). Disparity analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**: 333-340
- Bouwman, T. (2014). Traditional and recent approaches in background modeling for foreground detection: An overview. *Computer Science Review* **11-12**: 31-66
- Brooks, R.A. (1991) Intelligence without reason, MIT A.I. Memo No. 1293.
- Bugeau, A. and Pérez, P. (2009). Detection and segmentation of moving objects in complex scenes. *Computer Vision and Image Understanding*, **113(4)**: 459-476.

- Canny, J. (1986). A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **6**: 679-698.
- Cao, W., Lu, F. and Wang, S. (2006). Priority ordered direction basis function neural networks and the application for object recognition. *Granular computing, 2006 IEEE International Conference*, 229-231
- Chen, X. (2008). Application of Matlab in Moving Object Detection Algorithm. *International Seminar on Future BioMedical Information Engineering, Wuhan*, 18-18 December, 114-117
- Chen, T. (2009). Object Tracking Based on Active Contour Model by Neural Fuzzy Network. In *Control, Automation and Systems Engineering. IITA International Conference on, Zhangjiajie*, 11-12 July, 570-574.
- Cheraghi, S. A. and Sheikh, U. U. (2012). Moving object detection using image registration for a moving camera platform. In *Control System, Computing and Engineering (ICCSCE), 2012 IEEE International Conference on*, 355-359
- Çetin, F.H. (2011). Bir Görüntüdeki Nesnenin Bir Başka Görüntüde Bulunması. Yüksek Lisans Tezi, Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Ankara
- Das, T., Kar, I.N. and Chaudhury, S. (2006). Simple neuron-based adaptive Controller for a nonholonomic mobile robot including actuator dynamics, *Neurocomputing* **69**: 2140–2151.
- Declercq, F. And De Keyser, R.D. (1999). Object recognition and selective handling by a robot. *Journal of Intelligent and Robotic Systems*, **25**:121–132.
- Deng, Y.M., Tor, S.B. and Britton, G.A. (2000). A dual-stage functional modelling Framework with multi-level design knowledge for conceptual mechanical design, *Journal of Engineering Design* **11**: 347–375.
- Ding, J., Li, M., Huang, K. and Tan, T (2010). Modeling complex scenes for accurate moving objects segmentation. *Asian Conference on Computer Vision, ACCV, New Zealand*, 8-12 November, 82–94.
- Dong, W. (2002). On trajectory and force tracking control of constrained Mobile manipulators with parameter uncertainty, *Automatica*, **38**: 1475–1484.

- Dubois, S.R. and Glanz, F.H. (1986). An autoregressive model approach to two-dimensional shape classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **1**: 55-65.
- Ejiri, M. (2007). Machine vision in early days: Japan's pioneering contributions. 8th Asian Conference on Computer Vision (ACCV)
- Elmas, Ç. (2003). Yapay Sinir Ağları (Kuram, Mimari, Eğitim, Uygulama). Seçkin Yayıncılık, Ankara.
- Ergezer, E. (2007). Visual Detection And Tracking Of Moving Objects. Yüksek Lisans Tezi, Ortadoğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ankara
- Ferrández, J.M., de la Paz, and de Lope, F. J. (2010). Intelligent robotics and neuroscience, *Robotics and Autonomous Systems*, **58**:1221–1222
- Fraichard, T., Garnier, P. (2001). Fuzzy control to drive car-like vehicles. *Robotics and Autonomous Systems*, **34**: 1–22
- Fong, T., Nourbakhsha, I. and Dautenhahn, K. (2003). A survey of socially interactive robots. *Robotics and Autonomous Systems*, **42**: 143–166.
- Foresti, G. L., Gentili, S. (2000). A vision based system for object detection in underwater images, *Int. J. Pattern Recogn*, **14(02)**: 167-188.
- Foresti, G.L., Micheloni, C. and Picarelli, C. (2005). Detecting moving people in video streams. *Pattern Recognition Letters*, **26**:2232-2243
- Fu, L.C. and Liu, C.Y. (2001). Computer vision based object detection and recognition for vehicle driving. Robotics and Automation, ICRA. In Proceedings of IEEE International Conference on, **3**:2634-2641.
- Gamez, D.A.M. and Devy, M. (2013). Active vision-based moving objects detection from a Motion Grid. Mobile Robots (ECMR), 2013 European Conference on, 373-378.
- Gonzalez, R. C., Woods, R. E., and Eddins, S. L. (2009). Digital image processing using MATLAB, Gatesmark, USA.
- Gopalakrishnan, B., Tirunellayi, S. and Todkar, R. (2004). Design and development of an autonomous mobile smart vehicle: A mechatronics application, *Mechatronics* **15**: 491–514

- Goshtasby, A. (1985). Description And Discrimination Of Planar Shapes Using Shape Matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **7**: 738-743.
- Grauman, K., Leibe, B. (2011). Visual Object Recognition. Morgan & Claypool Publisher, USA
- Graves, M. and Batchelor, B. (2003). Machine vision for the inspection of natural products. Springer Science & Business Media.
- Greywalter, W. (1950) An Imitation of Life, *Scientific American*, **May 1950**: 42-45
- Gua, J.S. and de Silva, C.W. (2004). Development and implementation of a real-time open-architecture control system for industrial robot systems, *Engineering Applications of Artificial Intelligence*. **17**: 469–483.
- Guldner, J. and Bauer, R. (1995). A three-layered hierarchical path control system for Mobile robots: Algorithms and experiments. *Robotics and Autonomous Systems*, **14**: 133–147.
- Guo, Y., Chen, Y., Tang, F., Li, A., Luo, W., and Liu, M. (2014). Object tracking using learned feature manifolds. *Computer Vision and Image Understanding*, **118**: 128-139.
- Gurney, K. (1996). Computers and Symbols versus Nets and Neurons. UCL Draft Papers, UCL Press Limited, U.K.
- Gurta, L. and Srinath, M.D. (1987). Contour sequence moments for the classification of closed planar shapes. *Pattern Recognition*, **20**: 267-271
- Haykin, S. (1994). Neural Networks. Macmillan College Publishing Company, USA
- Heijden, F. (1996) Image based measurement systems: object recognition and parameter estimation. Wiley, USA.
- Horn, B. K., and Schunck, B. G. (1981). Determining optical flow. In 1981 Technical Symposium East, International Society for Optics and Photonics, 319-331.
- Hu, M. K. (1962). Visual pattern recognition by moment invariants. *IEEE Transactions on Information Theory*, **8**: 179–187.

- Hu, H., Li, Z., Qu, Z. and Wang, D. (2009). Vision-Based Moving Objects Detection with Background Modeling. *Measuring Technology and Mechatronics Automation, ICMTMA '09. International Conference on*, **2**: 436-439
- Huang, K., Wang, L., Tan, T. and Maybank, S. (2008). A real-time object detecting and tracking system for outdoor night surveillance. *Pattern Recognition*, **41**: 432 - 444
- Huang, S.C. and Do, B.H. (2014). Radial Basis Function Based Neural Network for Motion Detection in Dynamic Scenes. *Cybernetics, IEEE Transactions on*, **44**: 114-125
- Ikeuchi, K. and Hong, K. S. (1991). Determining linear shape change: Toward automatic generation of object recognition programs. *CVGIP: Image Understanding*, **53(2)**: 154-170.
- İşeri B. (2006). Personel Yer Belirleme Sisteminde Gps Desteğinin Kullanılması. Yüksek Lisans Tezi, Dicle Üniversitesi, Fen Bilimleri Enstitüsü, Diyarbakır.
- Ioannis, P. (1993). *Digital Image Processing Algorithms*. Prentice Hall International Series in Signal Processing. New York.
- Jarraya, S.K., Hammami, M. and Ben-Abdallah, H. (2010). Accurate Background Modeling For Moving Object Detection in a Dynamic Scene. *Digital Image Computing: Techniques and Applications (DICTA), International Conference*, 52-57
- Johan, T. M., Prabuwno, A. S. (2011). Recognition of bolt and nut using artificial neural network. *Pattern analysis and intelligent robotics (ICPAIR) 2011 international conference*, 165-170.
- Juang, C.F. and Chen, L.T. (2008). Moving object recognition by a shape-based neural fuzzy network. *Neurocomputing*, **71**: 2937-2949
- Jung, B. and Sukhatme, G.S. (2004). Detecting Moving Objects using a Single Camera on a Mobile Robot in an Outdoor Environment. *8th Conference on Intelligent Autonomous Systems, Amsterdam, 10-13 March*, 980-987
- Kahl F., Hartley R. and Hilsenstein V. (2004). Novelty detection in image sequences with dynamic background. In *2nd Workshop on Statistical Methods in Video Processing (SMVP), European Conference on Computer Vision, Prague*, 117-128.



- Kam, M., Zhu, X., and Kalata, P. (1997). Sensor fusion for mobile robot navigation. *Proceedings of the IEEE*, **85(1)**: 108-119.
- Kashioka, S., Ejiri, M., Sakamoto, Y.(1976). A transistor wire-bonding system utilizing multiple local pattern matching techniques. *IEEE Transactions on Systems, Man and Cybernetics* 6, **8**: 562–570
- Khabou, M.A., Hermi, L. and Rhouma, M.B.H. (2007). Shape recognition using eigenvalue of the Dirichlet Laplacian. *Pattern Recognition*, **40**:141-153.
- Kim, J. and Do, Y. (2012). Moving obstacle avoidance of a mobile robot using a single camera. *Procedia Engineering*, **41**: 911 – 916
- Kim, J., Ye, G. and Kim, D. (2010). Moving Object Detection Under Free-Moving Camera. Proceedings of IEEE 17th International Conference on Image Processing, Hong Kong, 26-29 September, 4669-4672
- Kim, S. and Kim, H.(2010). Optimally overlapped ultrasonic sensor ring design for minimal positional uncertainty in obstacle detection. *Int. J. Control Autom. Systems*, **8(6)**: 1280-1287.
- Kim, S.W., Yun,K., Yi, K.M., Kim, S.J. and Choi, J.Y. (2013). Detection of moving objects with a moving camera using non-panoramic background model. *Machine Vision and Applications*, **24**:1015–1028.
- Koller, D., Luong, Q. T., and Malik, J. (1994). Using binocular stereopsis for vision-based vehicle control. Proceedings of the Intelligent Vehicles' 94 Symposium, 24-26 October, 237-242.
- Kong, H., Audibert, J., Ponce, J. (2010). Detecting abandoned objects with a moving camera. *IEEE Transactions on Image Processing* **19(8)**:2201–2210.
- Lin, C.C. and Wolf, M. (2010). Detecting Moving Objects Using a Camera on a Moving Platform. Pattern Recognition (ICPR), 2010 20th International Conference on, Istanbul, 23-26 August, 460-463.
- Lippmann, R. (1987). An introduction to computing with neural nets. *IEEE Assp Magazine*, **4(2)**: 4-22.

- Liu, H., Hong, T.H., Herman, M., Camus, T and Chellappa, R. (1998). Accuracy vs. efficiency trade-offs in optical flow algorithms. *In Computer Vision and Image Understanding*. **72(3)**: 271–286.
- Lowe, D.G. (1999). Object Recognition from Local Scale-Invariant Features. In International Conference on Computer Vision (ICCV), Corfu, Greece, 20-27 Eylül, 1150-1157.
- Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In IJCAI, **81**: 674-679
- Luque, R., Dominguez, E., Palomo, E. and Munoz, J. (2008). A neural network approach for video object segmentation in traffic surveillance surveillance. International Conference on Image Analysis and Recognition, ICIAR 2008, 151–158.
- Medland, A.J., Mullineux, G.A. (2000). Decomposition strategy for conceptual design, *Journal of Engineering Design*, **11**: 3–16.
- Mejia, V. and Kang, E.Y. (2011). Automatic moving object detection using motion and color features and bi-modal Gaussian approximation. *Systems, Man, and Cybernetics (SMC), IEEE International Conference on*, Anchorage, 9-12 October, 2922 - 2927
- Mihaylova, L., De Schutter, J. and Bruyninck, H. (2003) A multisine approach For trajectory optimization based on information gain, *Robotics and Autonomous Systems*, **43**: 231–243.
- Mokhtarian, F. and Abbasi, S. (2002). Shape similarity retrieval under affine transforms. *Pattern Recognition*, **35**: 31-41.
- Montironi, M.A., Castellini, P., Stroppa, L. and Paone, N. (2014). Adaptive autonomous positioning of a robot vision system: Application to quality control on production lines. *Robotics and Computer-Integrated Manufacturing*, **30**: 489–498
- Moreno, F., Alarcón, J., Salvador, R. and Riesgo, T. (2009). Reconfigurable hardware architecture of a shape recognition system based on specialized tiny neural networks with online training. *Industrial electronics, iee transactions*, **56**: 3253-3263.

- Moriwaki, K. (2011). Recognition of moving objects by image processing and its applications to a guide robot. System Integration (SII), IEEE/SICE International Symposium on, Kyoto, 20-22 December, 537- 542
- Morovec, H.P. (1977). Towards automatic visual obstacle avoidance. Proc. 5th Int. Joint Conf. Artificial Intell, 584
- Murshed, M., Morshed, M. and Chae, O. (2011). Moving Edge Matching for Moving Object Tracking. Proceedings of 14th International Conference on Computer and Information Technology, Dhaka, Bangladesh, 22-24 December, 355 – 359
- Nagabhushana, S. (2005). Computer Vision and Image Processing. New Age International, 1st edition, Delhi, INDIA
- Nilsson, N.J. (1969). A mobile automaton: an application of artificial intelligence techniques, in: Proc. of the First Int. Joint Conf. on AI, IJCAI, **69** : 509–520.
- Nixon, M.S., Aguado, A.S. (2012). Feature Extraction & Image Processing for Computer Vision. Elsevier AP, 3. Edition, London, UK.
- O'Donovan, P. (2005). Optical Flow: Techniques and Applications. The University of Saskatchewan, TR, 502425.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Trans. Systems, Man Cybernet*, **9(1)**: 62-66
- Pan, H. and Xia, L.Z. (2008). Efficient Object Recognition Using Boundary Representation and Wavelet Neural Network. *IEEE Transactions On Neural Networks*, **19** : 2132 - 2149
- Patil P.M, Deshmukh, M., Bonde, P.V., Dhabe, P.S ve Sontakke, T.R. (2004). Fourier fuzzy neural network for clustering of visual objects based on their gross shape and its application to handwritten character recognition. Neural Networks, Proceedings. IEEE International Joint Conference on, 25-29 July, **3**: 2391-2396.
- Razali, M.T. and Adnan, B.J. (2006). Detection and Classification of Moving Object for Smart Vision Sensor. Information and Communication Technologies, ICTTA '06, Damascus, **1**:733 – 737
- Ren, Y., Chua, C.S. and Ho, Y.K. (2003). Statistical background modeling for non-stationary camera. *Pattern Recognition Letters*, **24** :183–196

- Richard, W.C., Ali, A. and Trevor, D. (1997). Pfunder: real-time tracking of the human body. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, **19(7)**: 780-785.
- Ridder, C., Munkelt, O. and Kirchner, H. (1995). Adaptive background estimation and foreground detection using kalman filtering. International Conference on recent Advances in Mechatronics, 193-199.
- Roberts, L.G.(1960) Pattern recognition with an adaptive network. IRE International Convention Record. 66–70.
- Roth, S., Black, M. J. (2007). On the spatial statistics of optical flow. *International Journal of Computer Vision*, **74(1)**: 33-50.
- Rui, L., Zhijiang, D., Fujun, H., Minxiu, K. and Lining, S. (2008). Tracking a moving object with mobile robot based on vision. *Neural Networks, IJCNN 2008. (IEEE World Congress on Computational Intelligence)*. IEEE International Joint Conference on, Hong Kong, 1-8 June 2008, 716 – 720
- Sappa, A.D., Dornaika, F., Ger'onimo, D. and L'opez, A. (2008). Registration-based moving object detection from a moving camera. *IROS 2nd Workshop: Planning, Perception and Navigation for Intelligent Vehicles*.
- Shimakawa, H., Ido, G., Takada, H. and Takegaki, M.(2001). Real-time reactions in Supervisory control according to data freshness. *Real-Time Systems*, **20**:197–210.
- Shimizu, S., Yamamoto, K., Wang, C., Satoh, Y., Tanahashi, H., and Niwa, Y. (2006). Moving object detection by mobile Stereo Omni-directional System (SOS) using spherical depth image. *Pattern analysis and applications*, 9(2-3), 113-126.
- Sood, D., Repko, M. C. And Kelley, R.B.(1989) Design and Implementation of a Multi-Sensor Robot System for Printed Circuit Board Insertion. *Robotics and Automation, Proceedings, IEEE International Conference*, **1**:377 – 382
- Spagnolo, P., Orazio, T.D., Leo, M. and Distanto, A. (2006). Moving object segmentation by background subtraction and temporal analysis. *Image and Vision Computing*, **24**:411–423
- Stauffer, C. and Grimson, W. E. L. (1999). Adaptive background mixture models for real-time tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2: 246-252.

- Şanlıoğlu İ. (2004). Global Konum Belirleme Sistemi(Gps) Yazılımlarının Veri İşleme Modüllerinin Uluslararası Gps Servisi (Igs) Ürünleri Kullanarak Test Edilmesi. Doktora Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya
- Trevelyan, J.(1997) Simplifying robots A challenge for research. *Robotics and Autonomous Systems*, **21**: 207–220.
- Tippett, J.T. , Borkowitz, D.A., Clapp, L.C., Koester, C.J. and Vanderburgh, A.J.(1965). Optical and Electro-Optical Information Processing, MIT Press, Massachusetts, USA
- Türkoğlu, İ. (1996). Yapay Sinir Ağları İle Nesne Tanıma. Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.
- Ünal Ü. (2012). Gps Yönlendirmeli Tarımsal Bir Robotun Geliştirilmesi ve Anız Yoğunluğunun Belirlenmesi Örneğinde Kullanılması Üzerine Bir Araştırma. Doktora Tezi, Akdeniz Üniversitesi, Fen Bilimleri Enstitüsü, Antalya
- Wahi, A., Ravi P. and Saranya, M. (2010). A neural network approach to rotated object recognition based on edge features recognition rate and cpu time improvement for rotated object recognition using dwt. Computing communication and networking technologies (ICCCNT), international conference, 29-31 July, 1-6.
- Wahi, A., Athiq, F. M. and Palanisamy, C. (2008). A hybrid feature extraction method-based object recognition by neural network. Computing, communication and networking, International Conference, 18-20 Dec, 1-5.
- Wang, Z., Cui, J., Zha, H., Kegesawa, M. and Ikeuchi, K. (2011). Object detection by common fate Hough transform. Pattern Recognition (ACPR), First Asian Conference on, Beijing, 28-28 November, 613-617
- Weng, M., Huang, G. and Da,X. (2010). A New Interframe Difference Algorithm for Moving Target Detection. Image and Signal Processing (CISP), 3rd International Congress, Yantai, 16-18 October, 285-289
- Werner B., Surmann H. and Pervolz K. (2006). 3D time-of-flight cameras for mobile robotics. In: Proc. IROS, Beijing, China, 790-795.
- Winters, N. and J. Santos-Victor.(2002). Information sampling for vision-based robot navigation. *Robotics and Autonomous Systems*, **41** :145–159.

- Wohler, C. and Anlaut, J.K. (2001). Real-time object recognition on image sequences with the adaptable time delay neural network algorithm: applications for autonomous vehicles. *Image and Vision Computing*, **19**: 593-618
- Wu, Y.G., Yang, J.Y. and Liu, K. (2006). Obstacle detection and environment modeling based on multisensor fusion for robot navigation. *Artificial Intelligence in Engineering*, **10**: 323-333
- Yap, K.H., Guan, L., Perry, S. W., and San Wong, H. (2009). Adaptive image processing: a computational intelligence perspective second edition. CRC Press, Boca Raton, USA
- Yavuz, H.(1999) Conceptual design and development of an autonomous mobile robot, Ph.D. Thesis, Engineering Department, Lancaster University, Lancaster, UK,
- Ye, C., and Borenstein, J. (2002). Characterization of a 2-D laser scanner for mobile robot obstacle negotiation. International Conference on Robotics and Automation., Washington DC, USA, 10-17 May 2002, 2512-2518
- Yu, Q., Medioni, G. (2007) Map-enhanced detection and tracking from a moving platform with local and global data association. In: Proc. IEEE Workshops on Motion and Video Computing, Austin, Texas.
- Yu, X., Chen, X and Gao, M. (2012) . Motion Detection in Dynamic Scenes Based on Fuzzy C-Means Clustering. International Conference on Communication Systems and Network Technologies, 11-13 May, 306-310.
- Zahn, C.T. and Roskies, R.Z. (1972). Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, **C-21**: 269-281.
- Zha, Y., Bi, D. and Yuan Yang. (2009). Learning complex background by multi-scale discriminative model. *Pattern Recognition Letters*, **30**: 1003–1014
- Zhang, Y., Kiselewich, S. J., Bauson, W. A., & Hammoud, R. (2006). Robust moving object detection at distance in the visible spectrum and beyond using a moving camera. In Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on, 17-22 June, 131-131
- Zhang, D. and Lu, G. (2004). Review of shape representation and description techniques. *Pattern Recognition*, **37**:1-19

Zhou,M., and Gao,W.(2011). Multi-sensor Data Acquisitionfor an autonomous mobile outdoor robot. Fourth International Symposium on ComputationalIntelligenceand Design.

Zhou, Q. and Aggarwal, J. K. (2001). Tracking and classifying moving objects from video. In Proceedings of IEEE Workshop on Performance Evaluation of Tracking and Surveillance. Hawaii, USA.

### **İnternet Kaynakları**

1-<https://www.frc.ri.cmu.edu/~hpm/project.archive/robot.papers/2000/robot.evolution.-html>, 12.12.2014

2-[http://www.mathworks.com/products/matlab/features.html?s\\_tid=srchtitle](http://www.mathworks.com/products/matlab/features.html?s_tid=srchtitle), 07.07.2015

3- <http://www.gpsinformation.org/dale/nmea.htm>, 08.07.2015

4- <http://www.mikroe.com/click/gps>,09.07.2015

5- <http://www.mathworks.com/help/nnet/ref/trainlm.html> (24.12.2015)

6- <http://www.ardupilot.co.uk> (15/05/2016)

7- <http://ardupilot.org/planner/docs/mission-planner-overview.html> (15/05/2016)

8-<https://en.wikipedia.org/wiki/Arduino> (15/05/2016)

## ÖZGEÇMİŞ

Adı Soyadı : Güray SONUGÜR  
Doğum Yeri ve Tarihi : Altınova/05.03.1973  
Yabancı Dili : İngilizce  
İletişim (Telefon/e-posta) :0.505.4364142 / gsonugur@aku.edu.tr

### Eğitim Durumu (Kurum ve Yıl)

Lise : Afyon Lisesi (1987-1990)  
Lisans :Anadolu Üniv. Müh. Mim. Fak. Elektrik-  
Elektronik Müh. (1990 - 1994)  
Yüksek Lisans :Afyon Kocatepe Üniv. Fen Bilimleri Ens.  
Bilgisayar A.B.D. (2010 - 2012)

Çalıştığı Kurum/Kurumlar ve Yıl :Gürmas Makine Sanayi (1995-1996)  
Türk Telekom Afyonkarahisar İl Müdürlüğü  
(1997-2011)  
Afyon Kocatepe Üniversitesi Bilgi İşlem Daire  
Başkanlığı (2011- )

Yayımları (SCI ve diğer) :

### Uluslararası İndekslerde Taranan Dergiler

Gökçe, B. ve Sonugür, Güray. (2016). GPS Destekli İmge Çakıştırma Yöntemi ile Hareketli Nesnelerin Tespiti. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*. (Baskıda)

### Uluslararası Bilimsel Toplantılarda Sunulan ve Bildiri Kitabında Basılan Bildiriler

Gökçe, B. ve Sonugür, Güray. (2016). Detecting Moving Objects From Unmanned Ground Vehicles Using Hybrid Image Registration Method. In: International Conference on Engineering and Natural Science. Sarajevo, 24-28 May 2016



