

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GÜVENLİK SİSTEMLERİNDE KAYIP NESNE TESPİTİ

NAGEHAN ÇAPA
YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

GEBZE
2015

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

GÜVENLİK SİSTEMLERİNDE KAYIP
NESNE TESPİTİ

NAGEHAN ÇAPA
YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
YRD. DOÇ. DR. ALİ KÖKSAL HOCAOĞLU

GEBZE

2015

T.R.
GEBZE TECHNICAL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

**REMOVED OBJECT DETECTION IN
SECURITY SYSTEMS**

NAGEHAN ÇAPA
**A THESIS SUBMITTED FOR THE DEGREE OF
MASTER OF SCIENCE**
DEPARTMENT OF ELECTRONICS ENGINEERING

THESIS SUPERVISOR
ASSIST. PROF. DR. ALİ KÖKSAL HOCAOĞLU

GEBZE

2015

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 29/06/2015 tarih ve 2015/41 sayılı kararıyla oluşturulan jüri tarafından 24/08/2015 tarihinde tez savunma sınavı yapılan Nagehan ÇAPA'nın tez çalışması Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI) : Yrd. Doç. Dr. A. Köksal HOCAOĞLU

ÜYE

: Yrd. Doç. Dr. Ulaş VURAL

ÜYE

: Yrd. Doç. Dr. Serdar Süer ERDEM

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
..... tarih ve/..... sayılı kararı.

İMZA/MÜHÜR

ÖZET

Son yıllarda artan güvenlik talebi nedeniyle şüpheli durumların tespit edilmesi oldukça önemli ve gerekli bir hal almıştır. Güvenlik talebini arttıran önemli olaylardan biri de hırsızlık olaylarıdır. Çalınan eşyalar, arabalar, kamu mallarının ortadan kaybolması gibi olaylar başlıca hırsızlık olaylarıdır. Bu olaylar nedeniyle güvenlik sistemlerine duyulan ihtiyaç artmıştır. Bu nedenle güvenlik sistemlerinde şüpheli durumların otomatik olarak tespit edilmesi çok önemlidir, çünkü operatörün tespitine dayalı güvenlik sistemlerinde şüpheli durumlar gözden kaçabilmektedir. Bu çalışmada güvenlik kameralarından elde edilen görüntüler kullanılarak, önceden işaretlenmiş nesnelerin kaybolması durumunu otomatik olarak tespit edecek bir video gözetleme sistemi geliştirilmiştir. Bu problemin çözümü için önceden işaretlenmiş bir nesnenin kayıp olması durumunda alarm üreten bir yöntem önerilmiştir. Önerilen yöntem ışık değişimleri, gölgeler ve kısmi kapanma durumlarında dahi, sistemin yanlış alarm üretmesini en aza indirecek şekilde tasarlanmıştır. Tez çalışması kapsamında problemin çözümü için kaybolduğunda alarm üretilmesi istenilen nesnenin bir operatör tarafından seçilmesi sağlanmaktadır. İlgilenilen nesnenin tanınması için ise nesnenin yerel öznitelikleri çıkartılmakta ve sistemin eğitimi sağlanmaktadır.

Anahtar Kelimeler: Kayıp Nesne Tespiti, Öznitelik Çıkartımı, Öznitelik Seçimi, Sınıflandırma, Güvenlik Kamerası Sistemleri, Gerçek Zamanlı Sistemler.

SUMMARY

In recent years, the identification of suspect cases has taken a very important and necessary state with increasing security demands. One of the important events that increase the demand for security is theft. Stolen cars, properties, disappearance of public goods are major theft events. This kind of events causes increasing the need for security systems. Therefore, automatic detection of suspicious cases becomes very important in security systems. Because security systems based on the detection of suspected cases by operators could be overlooked. In this work, a video surveillance system will be developed which automatically detects the disappearance of pre-marked objects by using images obtained from security cameras. To solve that problem, a method which produces an alarm in case of the disappearance of pre-marked object is proposed. Proposed method is designed to minimize false alarms, which occurs in the condition of lighting changes, shadows and partial occlusions. In the thesis, to solve the problem, the object which causes system to produce an alarm is selected (pre-marked) by the security operator. To recognize the selected object, local features of the object are extracted and the system is trained.

Key Words: Removed Object Detection, Feature Extraction, Feature Selection, Classification, Video Surveillance Systems, Real Time Systems.

TEŐEKKÜR

BaŐta, yksek lisans eđitimimde ve akademik hayatımda desteđini ve yardımlarını hibir zaman esirgemeyip bilgisi ile bu alıŐmanın oluŐmasının yolunu aan danıŐmanım Yrd. Do. Dr. A. Kksal Hocaođlu'na,

Sevgi, anlayıŐ ve destekleri iin aileme,

GstermiŐ olduđu desteklerinden ve sabrından dolayı sevgili eŐim Birol apa'ya en iten teŐekkrlerimi sunarım.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	v
SUMMARY	vi
TEŞEKKÜR	vii
SİMGELER ve KISALTMALAR DİZİNİ	x
ŞEKİLLER DİZİNİ	xi
TABLolar DİZİNİ	xii
1. GİRİŞ	1
1.1. Tezin Amacı, Katkısı ve İçeriği	2
2. LİTERATÜR ÖZETİ	4
2.1. Arka Plan Çıkarımı Teknikleri	5
2.1.1. Gauss Karışımları Modeli	5
2.1.2. İkili Arka Plan Modeli	6
2.1.3. Çekirdek (Kernel) Yoğunluk Kestirimi	8
2.2. Performans Değerlendirme Kriterleri ve Veri Setleri	8
2.2.1. Veri Setleri	8
2.2.2. Performans Değerlendirme Kriterleri	10
3. İŞARETLENMİŞ NESNENİN KAYBOLMASININ TESPİTİ	11
3.1. İlgilenilen Bölge Seçimi ve Ön İşlemler	14
3.1.1. İlgilenilen Bölgenin Seçimi	14
3.1.2. Uygulanan Ön İşlemler	15
3.2. Öznitelik Çıkartımı	17
3.2.1. Öznitelik Seçimi	22
3.3. Sınıflandırma	24
3.4. Performans Ölçme	29
4. SAYISAL ÖRNEKVE DEĞERLENDİRMELER	30
4.1. Klasik Kayıp Nesne Tespit Algoritmaları	35
4.2. HOG Parametreleri	37
4.3. Öznitelik Sayısı	38

4.4. İlgilenilen Bölgenin Parçalara Ayrılması	39
4.5. SVM Uygulamanın Katkısı	40
4.6. Oylama Yönteminin Etkisi	42
5. SONUÇLAR	44
6. GELECEKTEKİ ÇALIŞMALAR	45
KAYNAKLAR	46
ÖZGEÇMİŞ	51

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler ve Açıklamalar Kisaltmalar

fps	:	Frame Per Second
sn	:	Saniye
AVSS	:	Advanced Video and Signal based Surveillance
CVSG	:	A chroma – based Video Segmentation Ground – truth
GTÜ	:	Gebze Teknik Üniversitesi
GYTE	:	Gebze Yüksek Teknoloji Enstitüsü
HOG	:	Histogram of Oriented Gradients
IEEE	:	Institute of Electrical and Electronics Engineers
i-LIDS	:	Imaginary Library for Intelligent Detection Systems
NN	:	Nearest Neighbor
PETS	:	Performance Evaluation of Tracking and Surveillance
PV	:	Parked Vehicle
SIFT	:	Scale Invariant Feature Transform
SVM	:	Support Vector Machine
VISOR	:	Video Surveillance Online Repository
	:	

ŞEKİLLER DİZİNİ

<u>Sekil No:</u>	<u>Sayfa</u>
3.1: Önerilen algoritmanın akış diyagramı.	13
3.2: a) Ofiste bulunan bir bisiklet, b) (avss2007_medium) metroda bulunan bir çanta, c) (pets2001_1) otoparktaki bir araç, d) bahçedeki bir bank.	15
3.3: Gri seviyeye dönüştürülmüş, ilgilenilen bölge.	16
3.4: İlgilenilen bölgenin küçük parçalara bölünmüş hali.	16
3.5: Öznitelik çıkarımı akış diyagramı.	18
3.6: Görüntünün gradyanlarının toplanması.	22
3.7: Öznitelik seçimi akış diyagramı.	23
3.8: İki sınıflı veri setini farklı şekilde ayıran düzlemler.	25
3.9: İki sınıflı svm örneği.	26
4.1: Örnek bir veri setinde kayıp nesne tespiti.	34
4.2: Önerilen yöntem (üstte), makaledeki yöntem (altta).	36

TABLolar DİZİNİ

<u>Tablo No:</u>	<u>Sayfa</u>
4.1: Kullanılan videoların özellikleri.	33
4.2: Önerilen yöntem ile klasik algoritmaların kayıp nesne tespit zamanları.	36
4.3: Gamma dönüşümünün etkisi.	37
4.4: Hog parametrelerinin sistemin performansına etkileri.	38
4.5: Farklı sayıda öznelik seçimi ile yapılan test sonuçları.	38
4.6: Küçük nesnelere için ilgilenilen bölgenin parçalara bölünmesinin performansa etkisi.	40
4.7: İlgilenilen bölgenin ızgaralara ayrılmasının etkisi (nn).	40
4.8: En yakın komşu kuralı ve svm ile sınıflandırmanın sonuçları.	41
4.9: Oylama mekanizmasının etkisi (nn).	43
4.10: Parçaların özneliklerini birleştirmenin etkisi.	43

1. GİRİŞ

Hırsızlık, şüpheli paket, terör olayları vb. güvenlik sorunları hızla artmaktadır. Bu sebeple, son yıllarda video güvenlik sistemlerinin kullanımı oldukça artmıştır. Ancak, mevcut güvenlik sistemlerinde, güvenlik kameraları kayıtlarının bir operatör tarafından izlenmesi yoluyla şüpheli durumların tespit edilmesi, gözden kaçabilecek durumların oluşmasına ve efor kaybına sebep olmaktadır. Bu nedenle bilgisayarla görme ve yapay zekâ teknikleri geliştirilmekte ve güvenlik sistemleri otomatikleştirmeye çalışılmaktadır [1]-[10].

Video güvenlik sistemlerinin geliştirilmesinde en çok ilgi çeken konulardan biri kayıp ve terkedilmiş nesne tespitleridir. Çünkü özellikle hava alanları, tren istasyonları, alışveriş merkezleri, parklar gibi kalabalık kamu alanlarında kayıp ve terkedilmiş nesnelere önemli ve sık karşılaşılan güvenlik problemleri arasındadır. Ancak kayıp ve terkedilmiş nesnelere takibi yoğun bir iş gücü gerektirir ve güvenlik maliyetlerinin artmasına sebep olur. Ayrıca monitöre uzun süre bakmaktan kaynaklı olarak dikkat dağınıklığı ve yorgunluk oluşması vb. sebeplerle operatör performansı düşebilmekte ve dolayısıyla güvenlik sorunu oluşabilmektedir [11]. Bu nedenlerle operatörün performansının artmasına yardımcı olmak amacıyla destek sistemlerine ihtiyaç duyulmaktadır.

Bu tez çalışması kapsamında kritik değere sahip bir nesnenin bulunduğu mekândan alınması veya kaybolması durumunda alarm üretilmesi problemi ele alınmaktadır. Sistem, nesne önceden belirlenmiş bir eşik değerinden daha uzun bir süre ortamdaki kaybolduğunda alarm üretecektir. Bu nesne sistemde kullanıcının seçimi ile belirlenmektedir. Nesne seçiminin amacı çalınabilecek bir kamu eşyası, müzedeki değerli bir tablo, herhangi bir çanta, park halindeki bir araç gibi önceden belirlenmiş, taşınabilecek bazı eşyaların güvenliğini sağlamaktır. Nesne seçilmesi ile çalışılan bu yöntemde, kullanılan videonun yakın alan veya uzak alan videosu olma gereksinimi ortadan kaldırılmaktadır. Yani sistem nesnenin boyutlarından bağımsız olarak çalışmaktadır.

Seçilen bölge, herhangi bir şekilde kapatılmaması, bir şey bırakılmaması istenilen bir bölge de olabilir. Örneğin kapatılmaması istenilen ilgili bölge park yasağı olan bir bölge olabilir.

Nesne tespiti çalışmalarında karşılaşılan temel problemlerden biri, kısmi kapanma durumlarıdır. Bu çalışmada kısmi kapanma problemini çözmek için

nesnenin farklı kısımlarının ayrı ayrı değerlendirildiği bir yaklaşım önerilmiştir. Bu yaklaşımda seçilen bölge ızgara şeklinde bölümlere ayrılır. Izgaranın her bir parçası için nesnenin o ızgaradaki kısmının varlığı ayrı ayrı kontrol edilir. Sonuçta videonun bir karesindeki ızgaralarda, nesnenin olduğu durum daha çok tespit edilirse, o karede nesne vardır denir. Nesnenin tamamının kapandığı durumda nesnenin kayıp olduğu varsayımı yapılmıştır. Nesnenin farklı kısımlarının ayrı ayrı değerlendirilmesine benzer bir yaklaşım insan tespiti için kullanılmıştır [12]. İnsan vücudu için 11 farklı bölge belirlenmiş ve her bir kısım için sistemin eğitimi ayrı ayrı yapılmıştır. Eğitim için farklı fotoğraflardan alınan veriler kullanılmıştır. Tez için yapılan bölümlenmede ise nesnenin ne olduğunun önceden bir ayrımı yapılmamışken, bölünen kısımlarından alınan veriler kullanılır. İlgilenilen bölgeyi gören kamera ise sabit bir kameradır ve nesneyi belli bir açıdan görmektedir. Bu sebeple aslında nesnenin tek bir görünümünün farklı ışıklardaki görüntülerinden veri alınır. Yani sistem nesnenin farklı bir açıdan görünüşünün tespitini yapamayabilir ancak böyle durumlarda sistemin eğitimi tekrarlanabilir.

Kayıp nesne tespitinin yapılabilmesi için SVM (Support Vector Machines) sınıflandırıcısı kullanılmıştır. Sınıflandırma için ızgaralara ayrılmış ilgilenilen bölgenin her bir parçasındaki kenar bilgilerinden faydalanılmıştır. Kenar bilgilerinin çıkarımında ise Histogram of Oriented Gradients (HOG) yöntemi kullanılmıştır.

1.1. Tezin Amacı, Katkısı ve İçeriği

Otomatik olarak çalışan video güvenlik sistemleri gerçekleştirirken şüpheli durumların tespit edilmesi için video işaretlerinden anlamlı bilgiler çıkarılması gerekmektedir. Sistemde ilgilenilen bölge kullanıcı tarafından seçilerek farklı nesnelere için sistemin, bu çıkarılan bilgilerle eğitimi sağlanmıştır. Sistemin eğitimi sırasında belli aralıklarla veri alınır. Gerektiği durumlarda eğitim sırasında nesnenin kameranın görüş alanına göre açıları değiştirilerek, farklı açılardan görünüşlerinin de verileri elde edilebilir.

İlgilenilen bölgenin kullanıcı tarafından seçilmesi önceden bilinmeyen bir nesnenin kendisinin ve yerinin korunmak istenmesindedir. Klasik kayıp nesne tespiti probleminde, sahnenin arka planında duran herhangi bir nesnenin yerinden alınma durumu alarm üretilmesine sebep olur. Sadece ilgilenilen bir bölgedeki nesnenin kaybolması durumunda alarm üretilmesini sağlamaya yönelik bir sistemde

klasik kayıp nesne algoritmaları da kullanılabilir. Bu durumda, sahnedeki bir nesnenin kaybolması durumunda sadece işaretli bölgedeki bir nesne kaybolursa alarm üretilip, seçili bölge seçili bölge dışında bir nesnenin kaybı halinde alarm üretilmez. Sadece seçili bölgeye yönelik bir kayıp nesne analizi yapıldığında, klasik kayıp nesne tespiti problemi için önerilen algoritmalar da dahil olmak üzere, seçili bölgedeki nesnenin çalınması durumunda alarm üretilmesi hem daha kolay hem de daha hızlı olmaktadır.

Sistem ayrıca kapanma durumlarında hemen alarm üretmeyecek şekilde tasarlanmıştır. Bu nesnenin varlığının parça parça kontrol edilmesi ile sağlanmıştır.

Algoritma performansı değerlendirilirken doğru tespit oranı değerlendirme kriteri olarak kullanılmıştır. Farklı parametrelerin performansa etkisi incelenmiştir ve farklı algoritmaların sonuçları ile önerilen algoritmanın sonuçları karşılaştırılmıştır.

Bu tez kapsamında, ikinci bölümde kayıp nesne problemi ile ilgili daha önce yapılmış çalışmalar anlatılmıştır. Üçüncü bölümde, önerilen yöntem anlatılmış ve kullanılan öznitelik çıkartımı ve sınıflandırıcı yöntemi ile ilgili genel bilgiler verilmiştir. Dördüncü bölümde ise deneysel sonuçlar verilmiş ve bu sonuçlar yorumlanmıştır.

2. LİTERATÜR ÖZETİ

Video gözetleme sistemleri günümüzde oldukça yaygınlaşmıştır ve hemen her yerde güvenlik kameraları bulunmaktadır. Bu sistemler bir operatörün video görüntülerini canlı olarak izleyip şüpheli durumları tespit etmesine dayalı olarak kullanılmaktadır. Ancak, hava alanları, metro istasyonları, yollar, şehir merkezleri gibi kamuya açık ve değerli kamu mallarının da bulunabileceği yerlerde bu malların çalınması, yer değiştirmesi, kırılması gibi durumlarda tamamıyla operatörün dikkatine dayalı bir tespit sisteminin kullanımı durumun erken fark edilip önlem alınmasına olanak vermeyebilmektedir.

Bu durumların gerçek zamanlı ve otomatik olarak tespit edilmesi çeşitli çalışmalarla incelenmiş ve farklı ortam koşullarında iyi çalışabilecek çeşitli yöntemler geliştirilmiştir [2]-[4]. Kayıp nesne tespiti literatürde çoğunlukla terkedilmiş nesne tespiti ile birlikte ele alınmaktadır. Arka plana sonradan eklenmiş veya arka planda olmasına karşın daha sonra kaldırılmış nesnelerin konumu tespit edilip segmentasyonu yapıldıktan sonra bulunan bölge çeşitli sınıflandırma yöntemleriyle, terkedilmiş nesne veya kayıp nesne olarak sınıflandırılmaktadır. Bu sınıflandırma yöntemlerinin çoğu, bulunan nesnenin alanının piksellerinin çevresindeki pikseller ile benzerlik gösterip göstermemesine dayanmaktadır [13]. Bu tez çalışması kapsamında ise kayıp nesnelerin tespiti problemi ele alınmakla birlikte yalnızca önceden işaretlenmiş nesnelerin kaybolması durumunda alarm üretilmesi konusu ile ilgilenilmiştir.

Nesne segmentasyonu ile statik bölge tespiti için önerilen yöntemler çoğunlukla arka plan çıkarımı tekniğine dayanmaktadır. Literatürde genellikle arka plan çıkarımı yapılarak bulunan ön plan nesnesi arka plana ait olmayan kısım olarak tanımlanmaktadır.

Kayıp nesnelerin tespiti için nesne takibide kullanılan yöntemler arasındadır. Takip bilgisi [14], Gauss karışımı [2], ikili arka plan yöntemi [4], kenar enerjileri yöntemi [9], resimlerin zamanla ortalamasını alma [15], Kalman filtresi gibi yöntemler bulunmaktadır. Bu bölümde kayıp nesne tespiti için yaygın olarak kullanılan Gauss karışımları, kenar enerjileri, ikili arka plan yöntemlerinin algoritmaları incelenecektir. Bölüm 2.2’de ise performans değerlendirme kriterleri ile literatürde nesne tespiti için yaygın olarak kullanılan veri setleri verilmektedir.

2.1. Arka Plan Çıkarımı Teknikleri

Terk edilmiş veya kayıp nesne tespiti gibi birçok güvenlik sistemleri uygulamasında hareketli nesnelerin tespiti çok önemlidir. Literatürde birçok algoritma hareketli nesnelerin tespitini arka plan çıkarımı tekniklerine dayalı olarak yapmaktadır. Bu yöntem için kullanılan arka plan modelinin ve modelin güncelleme prosedürüne göre farklılıklar gösteren çeşitli yaklaşımlar bulunmaktadır. Gauss karışımları [14], [16]-[19], resimlerin zamanla ortalamasının alınması [16], ikili arka plan modeli [4], hareketli ortalama [15], [21], [22], hareketli Gauss ortalaması [17], yaklaşık medyan filtresi [18] bu yaklaşımlardan bazılarıdır. Bu parametrik tekniklerle ön plan tahmin edilir ve yoğunluk değerlerinin dağılımlarına göre takip eden arka planlar güncellenir. Bu tekniklerin yanında, ön planı ve arka planı istatistiksel özelliklerin yoğunluğuna göre tespit eden yöntemler bulunmaktadır. Çekirdek (Kernel) yoğunluk kestirimi [25]-[27] ve ortalama değer kayması tahmini [28]-[29] bu parametrik olmayan modellerdendir. Bu bölümde bu yaklaşımlardan bir kaçını incelenecektir.

2.1.1. Gauss Karışımları Modeli

Gauss karışımları modeli en çok kullanılan yöntemlerden biridir. İlk olarak [14] tarafından önerilmiştir. Bu yöntemde her piksel bir Gauss karışımı ile modellenir. Daha sonra her piksel kendisini temsil eden Gauss dağılımının arka plan modelinin parçası olup olmamasına göre sınıflandırılır.

Genellikle arka planın 3 Gauss karışımı ile modellendiği sistemlerde birinci Gauss karışımı sabit pikselleri ve arka plan resmini gösterir. Tekrarlayan hareketler ve bu hareketlerle ilişkili bölgeler ikinci Gauss karışımına göre güncellenir. Hızlı değişimler olan pikseller ise üçüncü Gauss karışımı ile gösterilir [19]. Arka plan dağılımlarına uymayan pikseller ön plan olarak değerlendirilir. Bu arka plan modelleme metodu α , öğrenme sabiti ve T , arka plan olarak sayılması gereken veri oranı olmak üzere iki önemli parametre içerir.

Ayrıca hesaplama kolaylığı için kovaryans matrisinin 2.3'teki denklemdeki gibi olduğu varsayılır.

Bu yöntemin en önemli avantajlarından biri, bir nesne arka planın parçası olduğunda arka planın var olan modelini bozmamasıdır [14].

Arka planı belirten Gauss dağılımı seçilirken sezgisel olarak varyansı en düşük olan dağılım seçilir. T , arka plan olarak sayılması gereken minimum eşik değeridir. Eğer T değeri küçük seçilirse arka plan modeli genellikle tektir, büyük seçilirse, arka plan ağaçlardaki yapraklar, rüzgârdaki bayrak gibi tekrar eden hareketlerden dolayı çoklu modellenir.

Gauss karışımları ile arka plan çıkarımı yöntemi yavaş ışık değişimlerinde, karmaşık arka planlardaki periyodik hareketlerde, yavaş hareket eden nesnelere, uzun süreli ekran değişimlerinde ve kamera gürültülerinde oldukça başarılı olmasına rağmen hızlı ışık değişimleriyle ve gölgelerle başa çıkamamaktadır [2].

2.1.2. İkili Arka Plan Modeli

İkili arka plan modelinde farklı kare hızlarında güncellenen iki arka plan çıkarımı yönteminin kullanılması önerilmiştir [4]. Giriş videosu farklı kare hızlarında işlenerek biri kısa süreli biri uzun süreli olmak üzere arka planlar düzenlenir. Arka planların her biri Gauss karışımları ile modellenir ve gerçek zamanlı olarak Bayes güncellemesi ile uyarlanır [4]. Şimdiki kare arka planlarla karşılaştırılarak iki ikili ön plan haritası tahmin edilir ve ön plan maskelerine sezgisel bir dizi uygulayarak hareket istatistikleri olası bir resim üzerinde toplanır. Bu olası resim daha sonra hareketli nesnelere, geçici sabit bölgelere ve ekran arka planına ait pikselleri ayırt etmek için kullanılır. Son olarak bir karar verme mekanizması ile terk edilmiş veya kayıp nesne bulunur.

Arka planın öğrenme oranı değiştirilerek, sabit bir nesnenin ne kadar kısa zamanda arka plana eklenmesi gerektiği anlaşılır. Bu sayede, geçici sabit resim bölgeleri, çoklu ön planları farklı öğrenme hızlarında analiz ederek, uzun süreli arka plan ve hareketli bölgelerden ayrılabilir. Uygulamaya göre bu geçici sabit bölgeler terk edilmiş nesnelere, hatalı park edilmiş araçları, ortamdaki kaldırılmış nesnelere belirtebilir. Bu yöntem nesne başlatma, takip veya çevrimdışı eğitim gerektirmez.

Terk edilmiş veya kayıp nesnelere tespit edebilmek için öncelikle bu durumların zamansal ve uzamsal olarak video istatistiklerini nasıl değiştirdiğini bilmemiz gerekir. Bu, uzun süreli sabit ortamı ve hareketli ön plan bölgelerini öğrenerek, ilgilendiğimiz pikselin terk edilmiş veya kayıp bir nesneye ait olup olmadığı hipotezini kurabileceğimiz anlamına gelir [20].

Uzun süreli arka plan B_L ve kısa süreli arka plan B_S olmak üzere iki arka plan kullanılır. Bu iki arka plan da Gauss karışımları ile tanımlanır. İşlemler RGB renk uzayında gerçekleştirilir. Bu iki arka plan da özdeş başlangıç parametrelerine sahiptir.

Her karede şimdiki kare, I , B_L ve B_S arka planları ile karşılaştırılarak kısa ve uzun süreli ön planlar tahmin edilir. F_L ve F_S ikili ön plan maskeleri elde edilir. $F(x, y) = 1$, (x, y) pikselinin değiştiğini belirtir. Ön plan maskelerinin değerlerine bağlı olarak aşağıdaki önermeler geliştirilmiştir.

- $F_L(x, y) = 1$ ve $F_S(x, y) = 1$, (x, y) pikseli hareketli bir nesneye ait olabilir, çünkü $I(x, y)$ arka planlara uymaz.
- $F_L(x, y) = 1$ ve $F_S(x, y) = 0$, (x, y) pikseli geçici sabit bir nesneye ait olabilir. (Terkedilmiş nesne)
- $F_L(x, y) = 0$ ve $F_S(x, y) = 1$, (x, y) pikseli daha önce kapanmış bir arka plan bölgesine ait olabilir. (Kayıp nesne)
- $F_L(x, y) = 0$ ve $F_S(x, y) = 0$, (x, y) pikseli arka plana aittir, çünkü $I(x, y)$ iki arka plana da uyar.

Kısa süreli arka planın öğrenme oranı uzun süreli arka plandan daha yüksektir. Bu nedenle kısa süreli arka plan değişimlere daha çabuk adapte olur ve değişimler ortama daha çabuk karışır. Uzun süreli arka plan ise değişimlere karşı daha dirençlidir.

Sistemi hızlandırmak için her renk kanalına karşılık 3 tek değişkenli Gauss kullanılır. Daha sonra her renk kanalı bağımsız olarak güncellenir.

Sistem her piksel için genellikle 3 ile 5 arasında seçilen k adet katmanla başlar.

Bu yöntem uzun süre durgun olan bir insan ile terk edilmiş nesne gibi farklı nesnelere ayırt edememektedir. Bunun için sistemde insan ve araç tespiti yöntemleri kullanılabilir [20].

2.1.3. Çekirdek (Kernel) Yoğunluk Kestirimi

Dinamik sahnelerin çoğu sürekli hareket özellikleri sergilerler. Bu nedenle dinamik ortamların davranışlarını optik akış yoluyla modellemek doğal bir yaklaşımdır [23].

Dinamik karakteristiklerin modellenmesi sürecinde optik akış hesaplanır ve daha yüksek boyutlu uzayda bir özellik olarak kullanılır.

Çekirdek tabanlı yoğunluk kestirimi, çok değişkenli olasılık dağılımlarını modellemek için literatürde en sık kullanılan yöntemdir. Örüntü tanımada buna Parzen pencereleri de denir.

Yoğunluk tahmin edilecek olan her bir nokta için, her veri noktasında aynı boyutta ve yönde olan çekirdekler ortalanır. Yoğunluk kestirimi kestirim noktasında çekirdeklerin yüksekliklerinin ortalaması alınarak hesaplanır.

Yoğunluk yaklaşımı için uygun mekanizma kurulduktan sonra gözlemlenen data için bir sınıflandırma mekanizması geliştirilir. Sınıflandırma yeni bir gözlemin arka plana ait olma olasılığı üzerinde eşik değeri ile gerçekleştirilebilir. Eşik değeri, belli bir pikselde belirsizliğe veya arka plan dağılımının yayılmasına dayalı, adaptif ve kararlı olmalıdır. Ön plan dağılımı hakkındaki kullanılabilir her ön bilgiden yararlanılmalıdır.

2.2. Performans Değerlendirme Kriterleri ve Veri Setleri

Bu bölümde literatürdeki çeşitli algoritmalar için verilen performans değerlendirmeleri incelenecektir. Değerlendirmeler için PETS 2006, PETS 2007, i-LIDS AVSS2007 gibi bilinen veri setleri kullanılmaktadır. Değerlendirme kriterleri olarak ise ROC analizi, düzensizlik algılama oranı analizi ve benzerlik ölçüleri gibi yöntemler bulunmaktadır.

2.2.1. Veri Setleri

- PETS 2006

Dokuzuncu IEEE uluslararası seminerinde takip ve gözetim performans değerlendirmelerinde kullanılmıştır. Veri setleri İngiliz ulaştırma polisi ve raylı sistemlerin destek ve işbirliği ile elde edilmiştir.

Veri setleri artan sahne karmaşıklığı ile kayıp eşya senaryoları içermektedir. Bu bağlamda kayıp eşya sahibi tarafından terk edilmiş bagaj parçası olarak tanımlanır. Her eşyanın bir sahibi ve her kişinin en çok bir eşyası vardır.

- PETS 2007

Onuncu IEEE uluslararası seminerinde takip ve gözetim performans değerlendirmelerinde kullanılmıştır. Veri setleri İngiliz havalimanları işletmesi destek ve işbirliği ile elde edilmiştir.

Veri setleri artan sahne karmaşıklığı ile boş gezenler, bagaj hırsızlığı ve kayıp eşya senaryoları içermektedir. Boş gezenler sahneye giren ve t saniyeden fazla kalanlar olarak tanımlanır. Hırsızlık ise bir eşyanın sahibinden 3 metreden uzağa taşınması olarak tanımlanır.

- i-LIDS AVSS 2007

i-LIDS video analiz sistemleri için İngiltere hükümetinin değerlendirme verileridir. Uygulamalı bilim ve teknoloji merkezi tarafından ulusal altyapı koruma merkezi ortaklığı ile geliştirilmiştir.

AVSS ise video ve sinyal tabanlı gözetim sistemleri üzerine bir IEEE konferans serisidir. Bu konferansta i-LIDS veri setinin bir alt kümesi kullanılmıştır. Veri seti içindeki ilgili olaylar terk edilmiş bagaj ve park edilmiş araç senaryolarıdır.

- CVSG

2008 yılında Avrupa Komisyonu ve İspanyol hükümeti desteği ile video işleme ve anlama laboratuvarı tarafından geliştirilmiştir. Görüntü ayırma algoritmalarının nicel ve eşit olarak değerlendirilebilmesi için kullanılan video veri setidir. Özellikle hareket tabanlı ayırma maskeleri üzerine odaklanmıştır.

- VISOR

Farklı araştırma projelerine destek vermek amacıyla Modena Üniversitesi görüntü laboratuvarında geliştirilmiş, geniş bir multimedya veri setidir.

- CANDELA

Kapalı alanlarda bulunan terk edilmiş nesne ve kavşak videoları içeren bir veri setidir.

- CANTATA

Açık alanlarda terk edilmiş nesne senaryoları içeren bir veri setidir.

- WCAM

Thales Güvenlik Sistemleri'nin WCAM projesi için sağladığı güvenlik kameraları video veri setidir.

2.2.2. Performans Değerlendirme Kriterleri

Kayıp veya terk edilmiş nesne tespiti gibi güvenlik sistemleri uygulamalarında performans değerlendirmeleri için standart bir metot bulunmamaktadır ve değerlendirme kriterleri değişkenlik gösterebilmektedir. Bu nedenle literatürde performans değerlendirmeleri için farklı yöntemler bulunmaktadır.

Bazı çalışmalarda uygulanan performans değerlendirmesinde öncelikle uygulamaların hangi program ile gerçekleştirildiği ve kullanılan kütüphaneler belirtilmektedir [9]. Ayrıca uygulamaların çalıştırıldığı bilgisayarların özellikleri paylaşılmış ve sonuçlar farklı algoritmalarla kıyaslanarak değerlendirilmiştir.

Veri setleri değerlendirmeler için arka plan karmaşıklığına göre sınıflandırılır. Bu karmaşıklık kenarların, çok sayıda desen ve dokunun, arka plana ait nesnelerin varlığına göre belirlenir. Arka planın karmaşıklığı düşük, orta ve yüksek olarak ayrılır.

3. İŞARETLENMİŞ NESNENİN KAYBOLMASININ TESPİTİ

Tez çalışması kapsamında kayıp nesne tespiti yapılırken, bir operatör tarafından ilgilenilen bölgenin seçilmesi temel alınmıştır. Operatör, kaybolması veya yerinin değiştirilmesi halinde alarm üretilmesi istediği nesneyi seçmektedir. Sistemin belirlenen nesne için eğitimi yapılırken kamera görüntüleri canlı olarak izlenirken kullanılabilmesi gibi kameranın geçmiş kayıtları da kullanılabilir. Bu çalışma kapsamında önerilen sisteme ilişkin seçilen nesne ile ilgili herhangi bir boyut, renk, şekil vb. kısıt olmayıp, sistemin herhangi bir nesne için gerekli şartlar oluştuğunda alarm üretmesi mümkündür.

Kayıp nesne tespiti için önerilen algoritmalar çoğunlukla, arka plan çıkarımı veya ön plan tespitine dayanmaktadır. Bu yaklaşımda görüntüdeki arka plana ait herhangi bir nesnenin bir süre sonra arka plana dahil olmadığı tespit edilmesi, kayıp nesne alarmı üretilmesi için kullanılmıştır. Bu yaklaşımı operatör tarafından önceden işaretlenmiş nesnelerin tespitinde de kullanmak mümkündür. Kayıp bir nesne tespiti yapıldığında eğer tespit yapıldığı bölge operatörün önceden işaretlediği bölgeye tekabül ediyorsa kayıp nesne alarmı üretilebilir. Ancak bu yaklaşımda nesnenin kısmi kapanma durumlarında yanlış alarmlar üretilmesi sorunu ortaya çıkmaktadır.

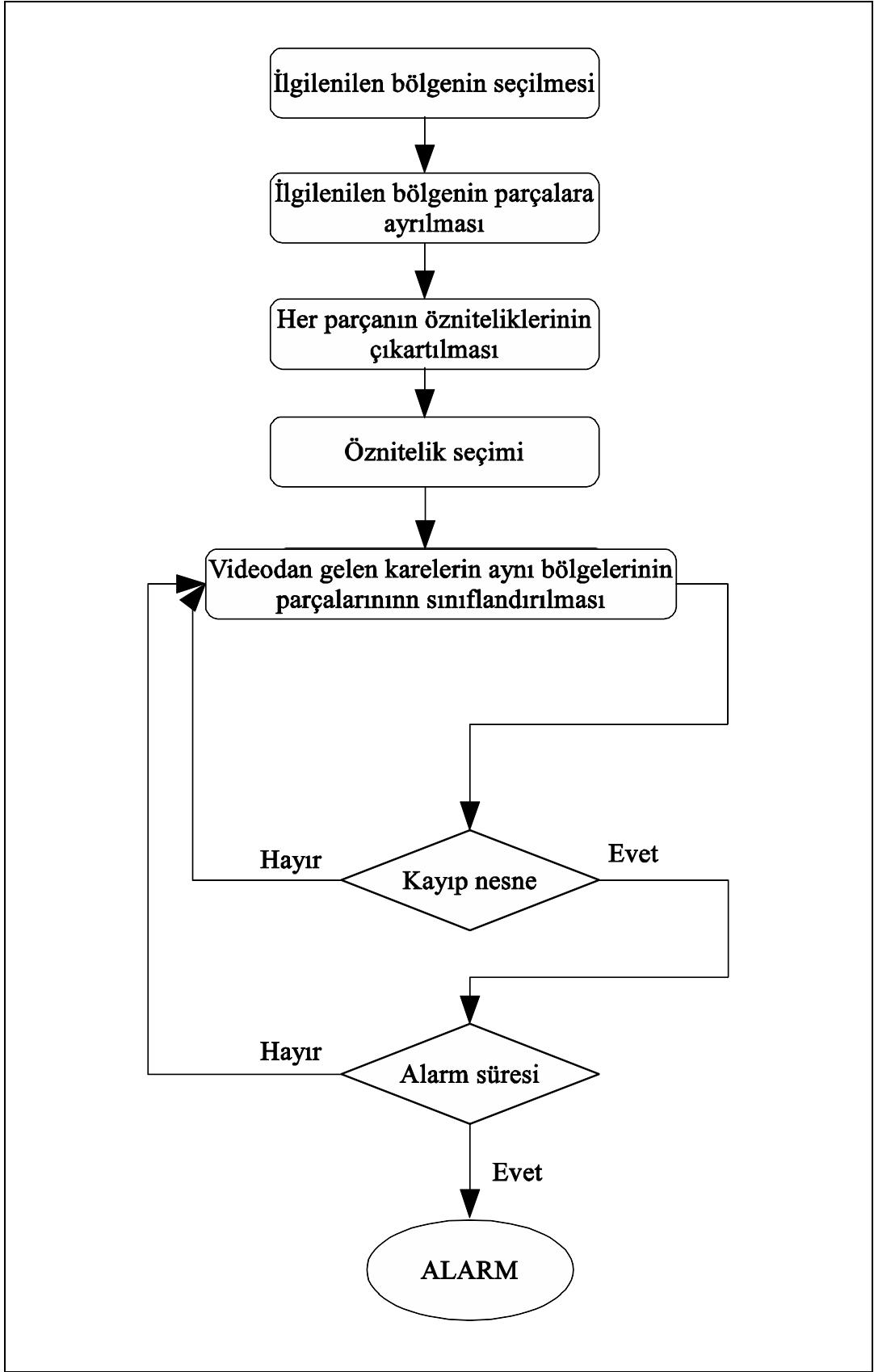
Kısmi kapanma durumlarında sistemin yanlış alarmlar üretmemesi için bu çalışmada ilgilenen bölgenin farklı kısımları ayrı ayrı değerlendirilmektedir. Bu farklı kısımlar ilgilenilen bölgenin parçalara ayrılması ile elde edilir.

İlgilenilen bölge seçildikten ve parçalara ayrıldıktan sonra, Şekil 3.1’de verilen akış diyagramında gösterildiği gibi her bir parçanın özneliği çıkartılır ve her parça için sistem ayrı ayrı eğitilir. İlgilenilen bölgede nesne olması ve olmaması (kapanması, kısmi kapanması) durumları eğitim veri setine dahil edilir. Nesnenin tam veya kısmi kapanması durumlarının ve nesnenin yerinden kaldırılması durumlarının eğitim veri setine dahil olmasının gerekliliği sebebiyle nesnenin taşınabilir bir nesne olması gerekmektedir. Eğitim veri setine dahil edilen kareler ardışık kareler olarak ele alınmayıp belli aralıklara sahip kareler öznelik çıkartılmak için kullanılmaktadır. Çünkü 25 fps hızındaki bir videonun her karesinde belirgin bir değişim olmayacaktır. Eğitim veri seti oluşturulurken ortamdaki ışık değişimleri,

yansımalar ve gölgelerden oluşan deęişimleri içerecek durumların dahil edilmesine dikkat edilir.

Deęerlendirilen karelerdeki ilgilenilen bölgelerin her bir parçası ayrı ayrı sınıflandırıldıktan sonra her bir kare için farklı parçaların etiketlerinden oluşan son durum oluşur. Bu son durumdaki etiketler bir karar sınırına göre deęerlendirilir. Yani ilgilenilen bölgenin her bir parçasının sınıflandırma sonucunda oluşan etiketlerinin sayısı karşılaştırılır. Hangi sınıfa ait etiket daha çok bulunursa bölge o sınıfa aittir denir. Bu sınır ilgilenilen bölgenin ne kadarının farklı sınıflandırılabileceęi ile ilgilidir.

Algoritmada son olarak bir karar sınırı daha bulunmaktadır. Bu sınır ise sistemin kısa süreli deęişikliklerde alarm üretmesini engellemek amacıyla koyulmuştur. Şekil 3.1’de gösterildięi gibi sistem, nesnenin önünün kısa süreli olarak kapanması durumlarında alarm üretmeyecektir.



Şekil 3.1: Önerilen algoritmanın akış diyagramı.

3.1. İlgilenilen Bölge Seçimi ve Ön İşlemler

Bu bölümde ilgilenilen bölgenin seçilmesi ve öznelik çıkartımı için yapılmış ön işlemler anlatılmıştır. Bu işlemler videonun bütün karelerine uygulanmıştır. Yapılan işlemler iki bölümde anlatılmıştır. İlk bölümde ilgilenilen bölgenin seçilmesi, ikinci bölümde ise seçilen bölgeye uygulanan ön işlemler anlatılmıştır. İkinci bölümde ayrıca seçilen bölgenin öznelik çıkarımını nasıl etkilediğinden bahsedilmiştir.

3.1.1. İlgilenilen Bölgenin Seçimi

Tez çalışması içinde müze, park alanı, hastane gibi yerlerde kontrol edilmek istenilen belirli bir nesnenin seçilmesi amaçlanmıştır. Bölge seçmenin amacı ise sadece ilgilenilen bir nesneyi baştan belirleyerek kaybolup kaybolmadığını hızlı bir şekilde tespit etmektir.

Sistem çalıştırıldığında ilk olarak kullanıcı tarafından videonun istenilen bir anından ilgilenilen bölge seçilir. İlgilenilen bölge bu çalışmanın konusu için ortamda olduğu kontrol edilmek istenen nesnedir. Ancak istenildiği takdirde boş bırakılması, ya da kapatılmaması istenilen bir bölge de seçilebilir.

İlgilenilen bölge uygulanacak ön işlemler için dikdörtgen şeklinde seçilmiştir. Şekil 3.2’de çeşitli videolardan, bir süre sonra kaldırılacak seçilmiş çeşitli nesnelere görülmektedir.

İlgilenilen bölge seçildikten sonra bölgenin sınırları öznelik çıkartma işlemine uygun hale getirilmiştir.



Şekil 3.2: a) Ofiste bulunan bir bisiklet, b) (AVSS2007_medium) metroda bulunan bir çanta, c) (PETS2001_1) otoparktaki bir araç, d) bahçedeki bir bank.

3.1.2. Uygulanan Ön İşlemler

İlgilenilen bölge seçildikten sonra videonun her karesine iki ön işlem uygulanmıştır. Bu işlemlerden ilki seçilmiş bölgelerden oluşan karelerin gri seviyeye dönüştürülmesi ve normalize edilmesidir. İncelenecek öznitelikler renkli resimlerde de çıkarılabilmektedir. Ancak işlemlerin hızlı ve kolay yapılabilmesi için gri seviyede çalışmak daha uygundur.

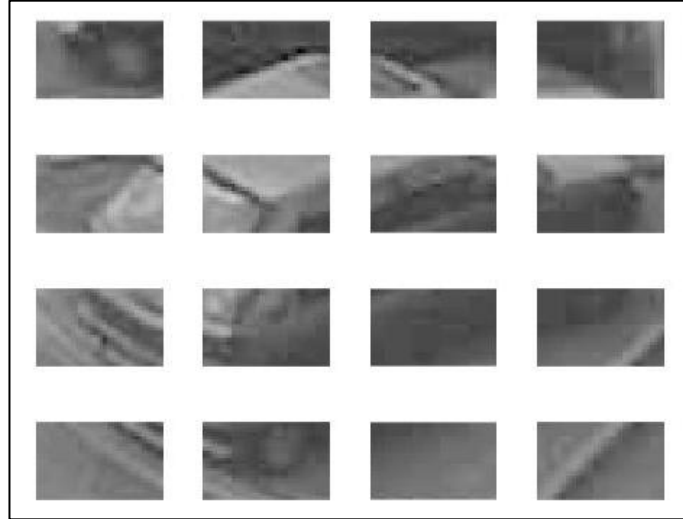
Uygulanan ikinci ön işlem ise kapanmadan kaynaklı hatalı sonuçları engellemek amacıyla yapılan bir ön işlemdir. Kayıp nesne tespiti için kullanılan yöntemlerde kısmi kapanma problemi en sık karşılaşılan sorundur. Bu problemin çözümü amacıyla videonun her bir karesi için seçilen bölge küçük dikdörtgen parçalara bölünmüştür. Bu bölme işlemi yapılmadan önce görüntünün boyutları eşit büyüklüklere bölünebilecek şekilde güncellenir. Bu parçalar algoritmanın sonraki aşamalarında ayrı ayrı sınıflandırılacak ve nesnenin kayıp olup olmadığına karar verilecektir. Kısmi kapanma probleminin çözümü için önerilen yöntemlerden biri, kapanmaya sebep olan nesne ve kapanmaya uğrayan nesne için ayrı ayrı yapılan bir

sınıflandırma tekniğidir [25]. Bu yaklaşımda kapanmaya sebep olacak nesnelerin önceden bilinmesi gerekmektedir. Sistem kapanmaya sebep olacak nesneler için farklı veri setleri ile eğitilir. Bu yöntem farklı nesneler için kapanmaya sebep olacak, farklı kapatıcı nesnelerin bilinmesini gerektirir. Dolayısı ile gerçek zamanlı olarak çalışacak bir sistemde veri toplanması için uygun değildir.

Şekil 3.3 ve 3.4'te sırasıyla ön işlemlerin uygulandığı seçilmiş bir bölgenin örneği görülmektedir.



Şekil 3.3: Gri seviyeye dönüştürülmüş, ilgilenilen bölge.



Şekil 3.4: İlgilenilen bölgenin küçük parçalara bölünmüş hali.

3.2. Öznitelik Çıkartımı

Nesne tespit ve tanıma algoritmaları nesnelerin görünüşleri arasında bir ilişki kurma temeline dayanmaktadır. Bu ilişki sayesinde nesnelerin sınıflandırılması yapılabilir ve durumları belirlenebilir. Bir nesnenin çeşitli resimlerden belirgin karakteristiklerinin çıkarılabilmesi için, görüntü özniteliklerinin çıkartılmasını sağlayan çeşitli yöntemler geliştirilmiştir.

Özniteliklerin bazıları global olmasına rağmen son zamanlarda önerilen yöntemlerin çoğu yerel öznitelikleri kullanmaktadır [12]. Çünkü yerel öznitelikler kısmi kapanmalara ve nesnelerin görünümdeki farklı kayıp parçalarına karşı daha az duyarlıdır. Yerel özniteliklerin çoğu şekil odaklıdır. Çünkü şekil, birçok nesne sınıfı için en tutarlı ve en belirgin özniteliktir.

Yerel özniteliklerden bazıları dalgacık tanımlamaları [26], Haar benzeri öznitelikler [27], dağınık dikdörtgen öznitelikleri [28], SIFT öznitelikleri [29], HOG öznitelikleri [30], kovaryans tanımlamalarıdır [31].

Bu çalışmada kullanılan ve yerel özniteliklerden olan HOG nesne tespit algoritmalarında çokça kullanılır. HOG, Dalal ve Triggs [30] tarafından yaya bulma problemi için diğer özniteliklerle karşılaştırılmış ve üstünlüğü kanıtlanmıştır.

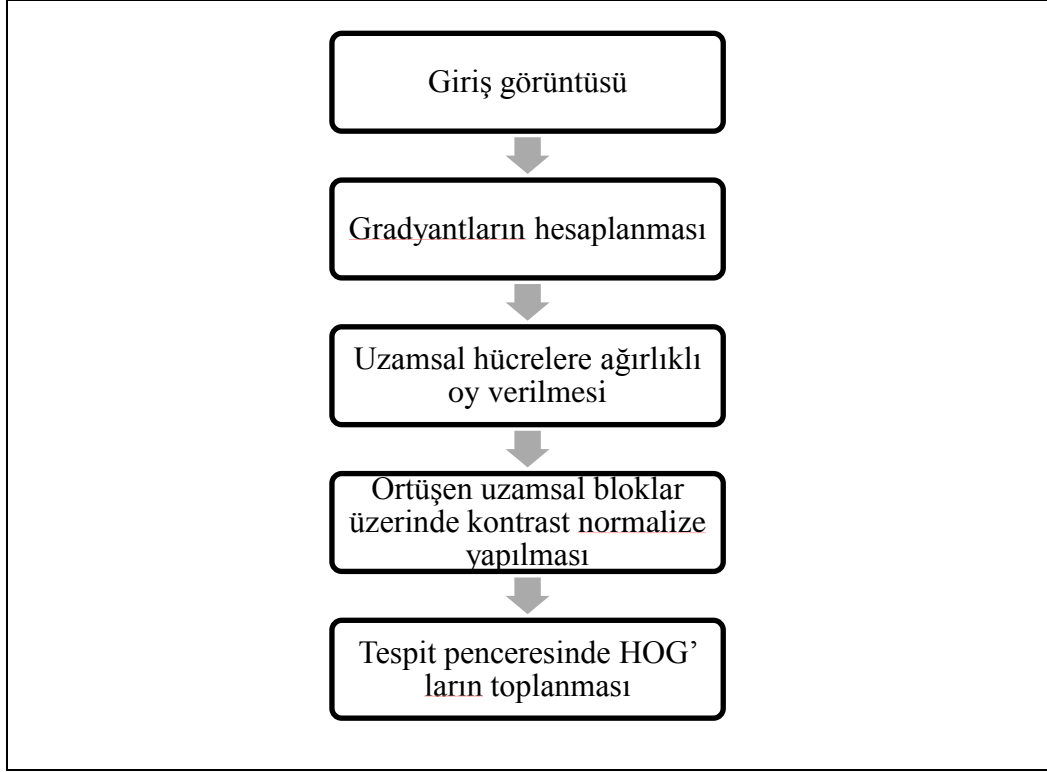
HOG ilk olarak insan tespiti için önerilmiş fakat daha sonra birçok probleme uygulanmıştır [30]. HOG öznitelik çıkarımının amacı bir nesneye farklı koşullar altında bakıldığında mümkün olduğunca yakın öznitelikler üretmektir. Böylece sınıflandırma problemi kolaylaşır. Nesnenin tamamının özniteliklerini bir vektöre aktarır. Öznitelik seçimi, genelleştirmeler ve işlem kolaylığı, HOG ile vektör şeklinde bulunan özniteliklerin diğer avantajlarıdır.

HOG öznitelik çıkarımının arkasındaki temel düşünce, bir görüntünün içindeki yerel nesnenin görünüm ve şeklinin, gradyant yoğunluklarının dağılımı ve kenar yönleri ile tanımlanabilmesidir. HOG bir kayan pencere algoritmasıdır [32], yani herhangi bir görüntü için bir pencere görüntünün her yerinde ve her ölçekte taşınarak öznitelikler hesaplanır.

HOG öznitelik çıkarımının uygulamasında görüntü hücre adı verilen bağlı küçük bölgelere ayrılır ve her hücre için hücre içinde piksellerin gradyant yönleri veya kenar yönelimleri bir histogramda toplanır. Daha sonra bu histogramların birleştirilmesi HOG'u oluşturur. Performansı geliştirmek için, yerel histogramlar kontrast görüntünün, blok adı verilen, daha büyük bir bölgesinde yoğunluk

hesaplanarak normalize edilir. Bu deęer kullanılarak bütün hücreler bu blok içinde normalize edilir. Normalize edilmiş öznitelikler ışık deęişimlerine ve gölgelere karşı daha sabit kalırlar. Yani HOG'un dayandığı esas resmin hücreleri içindeki gradyant yönelimlerinin, iyi bir şekilde normalize edilmiş yerel histogramlarının deęerlendirilmesidir.

Şekil 3.5'te HOG algoritmasının akış diyagramı görülmektedir.



Şekil 3.5: Öznitelik çıkarımı akış diyagramı.

Algoritmanın ilk aşamasında görüntü boyutları normalize edilir ve gamma normalizasyonu uygulanır. Görüntü boyutlarının normalize edilmesi, seçilen bölgenin alanının algoritmanın hücre boyutlarına uygun hale getirilmesidir. Bu işlem küçük boyutlu görüntüler için görüntünün boyutlarının yeniden ölçeklenmesidir. HOG öznitelik çıkarımının kullanıldığı farklı uygulamalarda da böyle bir ölçekleme yapılmıştır [32], [33]. Bu ölçekleme görüntüye ekstra satır ve sütun pikselleri eklenmesi demektir. Her yeni piksel, rengini orijinal görüntüdeki kendisine en yakın pikselden alır. Gamma dönüşümü, performansı iyileştirme etkisi küçük olduğu için (yaklaşık %1) ve işlem yükü getirdiği için her uygulamada kullanılmamaktadır. Bu çalışmada da gamma dönüşümünün performansa etkisinin çok küçük olduğu görülmüş ve görüntülere gamma dönüşümü uygulanmamıştır.

Algoritmanın uygulanmasındaki ikinci adım gradyant yönlerinin ve değerlerinin hesaplanmasıdır. Bu amaçla Dalal ve Triggs [30] tarafından görüntüye Gauss yumuşatma maskesi ve birkaç ayırık türev maskesi uygulayarak gradyantları hesaplamış ve sonuçlarını test edilmiştir. Hiç yumuşatma yapmamakla beraber farklı yumuşatma değerleri kullanılmıştır [30]. Test edilen maskelerden bazıları 1 boyutlu merkezsiz [-1, 1], merkezli [-1, 0, 1] ve kübik düzeltilmiş [1, -8, 0, 8, -1] maskeleridir. Ayrıca 3x3 Sobel maskeleri ve 2x2 diyagonal maskeler $\left(\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}\right)$ kullanılmıştır. Basit 1 boyutlu merkezli [-1, 0, 1] maskesi Gauss yumuşatma yapılmadığı durumda en iyi sonucu vermiştir, büyük maskelerin kullanılmasının ve yumuşatma işlemlerinin performansı her zaman azalttığı görülmüştür [30]. Ayrıca merkezsiz [-1, 1] türev maskesinin de performansı düşürdüğü söylenmektedir [30]. Çünkü yön merkezsiz bir maske ile yön bulunması sırasında x ve y maskeleri farklı merkezleri temel alacağından hatalı sonuçlar oluşmaktadır.

Elde edilen bu sonuçlardan sonra en çok kullanılan yöntem görüntüye yatay ve dikey yönde 1 boyutlu, merkezli ayırık türev maskesi uygulanmasıdır. Bu yöntem görüntüye aşağıdaki filtreleri uygulamayı gerektirir.

$$D_X = [-1 \ 0 \ 1], \quad D_Y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \quad (3.1)$$

Bir I görüntüsü için x ve y türevleri konvolüsyon ile bulunur.

$$I_X = I * D_X, \quad I_Y = I * D_Y \quad (3.2)$$

Gradyantların büyüklüğü ise aşağıdaki denklemlerle hesaplanır.

$$|G| = \sqrt{I_X^2 + I_Y^2} \quad (3.3)$$

Gradyanların yönü ise

$$\theta = \arctan \frac{I_Y}{I_X} \quad (3.4)$$

denklemiyle hesaplanır.

Algoritmanın üçüncü adımı ise hücre histogramlarının oluşturulmasıdır. Hücre içindeki her piksel tabanı yön temelli ve kanalları gradyant hesaplamada bulunan değerlere dayalı bir histograma ağırlıklı bir oy atar. Histogram binlerinin ağırlıkları gradyantların büyüklüğüne bağlıdır. Hücreler dikdörtgen veya dairesel biçimde olabilir. Histogram kanalları, gradyantların işaretli olup olmamasına göre, 0 ile 180 veya 0 ile 360 derece arasındadır. Örtüşmeyi azaltmak için, oylar yön ve pozisyon olarak komşu kanalların merkezleri arasında ikili değerlendirilir. Oyların ağırlığı ise her piksel için gradyant büyüklüğünün kendisi, karesi, karekökü veya kırılmış bir formu olabilir. Bu form pikselde kenar olmadığını veya yumuşak bir kenar olduğunu belirtir. Pratikte büyüklüğün kendisini kullanmak en iyi sonucu verir [30]. Gradyant büyüklüğünün karekökünü almak ise performansta azalmaya neden olur. Dalal ve Triggs [30] deneylerinde işaretli gradyantları farklı sayıda birleşik histogram kanalları ile göstermişlerdir. Histogram kanallarının sayısının artması performansı arttırmış ancak 9 histogram kanalından sonra çok küçük bir değişim gözlemlenmiştir [30]. SIFT öznelik çıkarımında olduğu gibi işaretli gradyantların kullanılması, iki katı sayıda kanal kullanılsa da performansı azaltıcı etki göstermiştir [30].

Dalal ve Triggs insan tespiti çalışmalarında yaya veri setleri kullanmışlardır. Bu sayede mümkün olduğunca birbirlerinden ayrı ve pozisyonları belli insanları incelemişlerdir. Bu çalışmada ise seçilen belirli bir nesnenin öznelikleri çıkartılacağı için kalabalık ve grup halindeki insanların tespiti problemdeki gibi sorunlar bulunmamaktadır. Ayrıca, insan tespitinde kişilerin farklı kıyafetleri ve arka plandaki karışıklıklar bilgi vermeyen kontrastlıklar oluşturduğu için Dalal ve Triggs işaretli gradyanlar kullanmışlardır. Bu sebeplerle nesne seçiminde böyle karışıklıklar oluşmadığı için HOG öznelikleri çıkarımında [30] ile aynı parametreler kullanılmıştır.

Öznelik çıkarımının son adımı ise ışık değişimleri ve kontrastlıklara karşı gradyantların normalize edilmesidir. Gradyantlar arka plan – ön plan kontrastlıkları ve ışık değişimlerine karşı farklılık gösterirler. Bu nedenle iyi bir performans için yerel kontrastlıkların normalize edilmesi çok önemlidir. Bu da hücrelerin bağlı bloklar içinde gruplandırılmasını gerektirir. Her hücre birden çok blok içinde bulunur. Her blok çoklu hücre histogramları içerir. Her bloğun histogramları ayrı ayrı normalize edilir. HOG öznelik vektörü, bileşenlerin blok bölgelerinin tümünden bulunan normalize hücre histogramları olur.

Görüntü üzerinde bloklar üst üste getirilirler, böylece her bir skaler hücre cevabı son durumda oluşan öznitelik vektörüne, her biri farklı bir bloğa göre normalize edilmiş birkaç parçadan katkı sağlamış olur. Bu işlem iyi bir normalizasyon için çok önemlidir ve bu üst üste getirme işlemi performansı önemli ölçüde iyileştirir [30].

Dikdörtgen veya dairesel olmak üzere iki farklı blok geometrisi tanımlanmıştır. Dikdörtgen bloklar dikdörtgen şeklindeki uzamsal hücrelerin birleşmesiyle oluşur. Dairesel bloklar ise logaritmik polar şeklindeki hücrelerden oluşur. Dikdörtgensel bloklar için R-HOG dairesel bloklar için C-HOG isimleri kullanılır.

R-HOG blokları SIFT [29] tanımlayıcısı ile çok benzerdir ancak farklı şekilde kullanılırlar. R-HOG blokları yön hizalaması olmadan, farklı tek ölçeklerde, yoğun dikdörtgen parçalar şeklindeki bölümlerde hesaplanır. SIFT tanımlayıcısı ise seyrek, değişmeyen ölçeklerde, görüntünün anahtar noktalarında hesaplanırlar ve yön hizalaması için döndürülürler ve ayrı ayrı hesaplanırlar. SIFT seyrek, geniş tabanlı eşleşme için optimize edilmiştir. R-HOG ise uzamsal formun sık ve güçlü kodlanabilmesi içindir. Kenar yönleri histogramları yönteminin diğer öncülleri de Freeman ve Roth'dur [34].

Bu çalışmada da R-HOG kullanılmıştır. HOG, her bloğun kaç hücre içereceği, her hücrenin kaç piksel içereceği, hücre histogramlarının kaç kanal içereceği olmak üzere üç parametre ile tanımlanır.

Farklı hücre boyutları ve farklı blok boyutları ile yaya veri setinde yapılan insan tespiti çalışmalarında çok büyük ve çok küçük bloklar kullanılmasının sonuçları kötü etkilediği görülmüştür [30]. 8x8, 6x6 ve 6x8 piksel boyutlarında hücre ve 2x2 ve 3x3 hücre boyutlarında blok kullanılmasının iyi sonuçlar verdikleri görülmüştür. Bu çalışmada ise 8x8 piksel boyutunda hücre ve 2x2 hücre boyutunda blok kullanılmıştır.

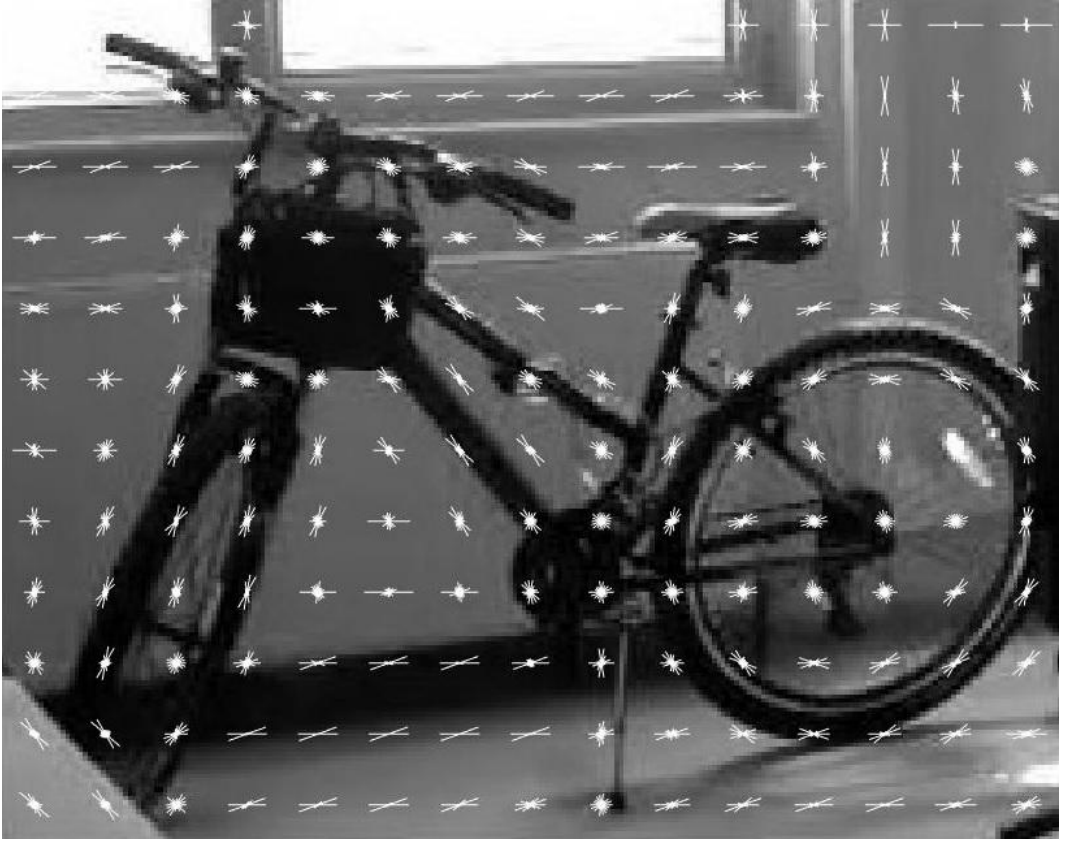
Her HOG geometrisi için dört farklı blok normalizasyon şeması bulunmaktadır. Bir bloktaki bütün histogramları içeren normalize edilmemiş vektör, $\|v\|$ k -norm ($k = 1,2$) ve e sonucu etkilemeyecek küçük bir sabit sayı olmak üzere, normalizasyon aşağıdaki denklemlerden biri ile yapılabilir.

$$L2 - norm = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}, \quad (3.5)$$

L2-Hys: L2-norma maksimum değeri 0.2 olacak şekilde bir kırpma uygulanır, (3.6)

$$L1 - norm = \frac{v}{\|v\|_1 + e}, \quad (3.7)$$

$$L1 - sqrt = \sqrt{\frac{v}{\|v\|_1 + e}}. \quad (3.8)$$



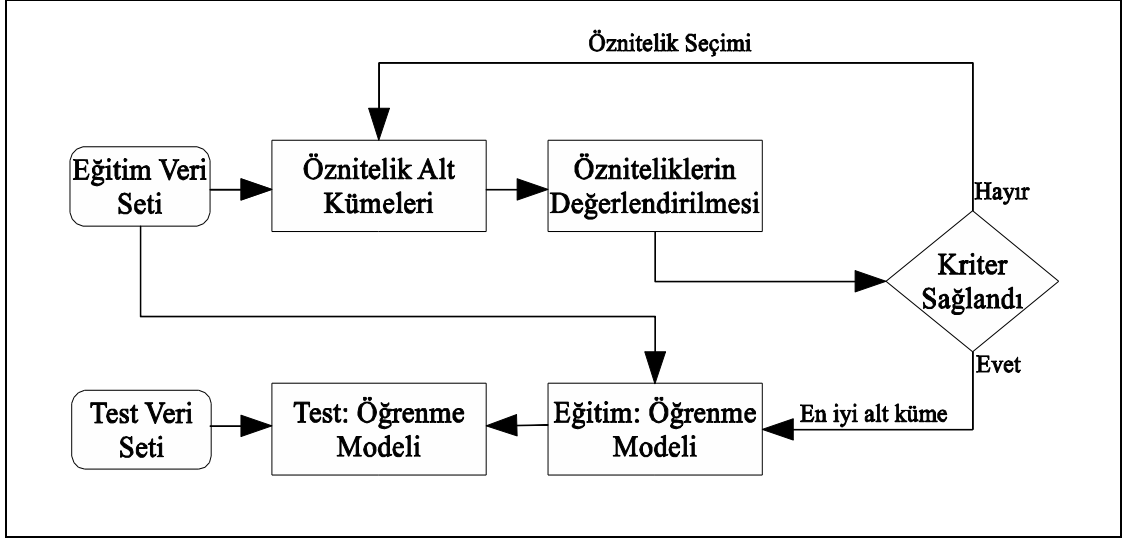
Şekil 3.6: Görüntünün gradyanlarının toplanması.

3.2.1. Öznitelik Seçimi

Öznitelik seçme belli kriterlere göre orijinal özniteliklerin altkümelerinin seçilmesidir. Sınıflandırma algoritmalarında yüksek boyutlu öznitelikler kullanılması algoritmaların iyi ve doğru çalışabilmesini zorlaştıran durumlar oluştururlar. Bu nedenle öznitelik seçimi ve boyut azaltma, başarılı bir sınıflandırma uygulaması için etkili ve önemli bir tekniktir [35].

Öznitelik seçimi öznitelik sayısının azaltılmasını, ilgisiz, gereksiz ve gürültülü özniteliklerin kaldırılmasını sağlar. Böylece algoritmanın hızını ve öğrenme doğruluğunu artırır [35].

Şekil 3.7’de öznitelik seçimi işleminin akış diyagramı görülmektedir [35].



Şekil 3.7: Öznitelik seçimi akış diyagramı.

Öznitelik seçimi işlemi üç adımdan oluşur [35]. Birinci adım belirli araştırma yöntemleriyle orijinal özniteliklerin bir alt kümesini içeren aday kümenin oluşturulmasıdır [35]. İkinci adımda bu aday küme değerlendirilir ve kümedeki özniteliklerin kullanılabilirliği tahmin edilir [35]. Değerlendirmeye göre aday öznitelik alt kümesindeki bazı öznitelikler atılabilir veya ilgilerine göre seçilen öznitelik kümesine eklenebilir. Son adımda ise seçilen özniteliklerin belirlenen bir durma kriterine göre iyi olup olmadıkları belirlenir. Eğer durma koşulu sağlanırsa öznitelik seçimi algoritması seçilen öznitelikleri verir, aksi takdirde durma koşulu sağlanana kadar iterasyon devam eder. Öznitelik kümesi seçildikten sonra sınıflandırıcı modeli ve sınıflandırma tahminleri için eğitim ve test veri setlerinin filtelenmesinde kullanılırlar.

Değerlendirme kriteri seçilen öznitelik alt kümesinin öğrenme algoritmasına bağlı olarak değerlendirilmesine göre iki gruba ayrılabilir [36]. Bağımsız kriterler genellikle bir öznitelik veya öznitelik alt kümesinin öğrenme algoritması uygulanmadan değerlendirilmesidir. Bu kriterlerden bazıları uzaklık ölçüsü, bilgi ölçüsü, bağlılık ölçüsü ve tutarlılık ölçüsüdür [36]. Bağımlı kriterlerde ise bir öznitelik veya öznitelik alt kümesinin iyiliği, bu özniteliklerin bir öğrenme

algoritmasına uygulanarak algoritmanın performansına göre değerlendirilir. Diğer bir deyişle bu kriter öğrenme algoritmasının performansını değerlendirmede kullanılan ölçüdür [36].

Bu çalışmada öznitelik seçimi için öznitelik sıralama algoritması kullanılmıştır. Bu algoritma ile öznitelikler bir değerlendirme ölçüsüne göre sıralanır. Öznitelik sıralama algoritması özniteliklerin herhangi bir öğrenme algoritmasına bağlı olmaksızın değerlendirilmesini sağladığı için çok basit bir yapıya sahiptir, kolay anlaşılır ve oldukça hızlı çalışır. Değerlendirme ölçüsü doğruluk, tutarlılık, bilgi, mesafe ve bağımlılıktan herhangi biri olabilir. Bu algoritma ile minimum öznitelik sayısının ne olması gerektiği bulunamaz ancak özniteliklerin önemi ve birbirleri ile ilgileri bulunur [37].

Özniteliklerin değerlendirilmesi için t-testi kullanılmıştır. t-testi bir öznitelik ile tüm öznitelikler arasındaki dağılımların farklılığının bir ölçüsüdür. Bir özneliğin değeri ile diğer özniteliklerin dağılımı arasındaki farkı bulmak için kullanılır. Eğer farkın belirli bir güven aralığı içinde olduğu tespit edilirse o öznitelik sınıflandırıcının eğitimi için yeni öznitelik vektörüne katılır. Böylece farklılığı fazla olan öznitelikler seçilmiş olur.

Sınıflandırıcının test aşamasında çıkarılan özniteliklerin seçimi ise seçilen özniteliklerin indisleri kullanılarak yapılır. Bu çalışmada ilgilenilen bölge seçildikten ve küçük parçalara ayrıldıktan sonra her bir parça için öznitelik çıkartımı yapılmaktadır. Seçilen bölgenin boyutlarına göre her bir parça için HOG öznitelik çıkarma algoritması ile yaklaşık 100 ile 300 arası öznitelik bulunmaktadır. Bu öznitelik sayıları sınıflandırma algoritmalarının düzgün ve doğru çalışabilmeleri için oldukça fazladır. Öznitelik seçimi işlemleri ile bu sayı her bir parça için istenilen sayıya düşürülmüştür.

3.3. Sınıflandırma

Parçalara ayrılan ilgilenilen bölgenin her bir parçası ayrı ayrı sınıflandırılır. Bu bölünmüş sınıflandırma sistemin kısmi kapanma durumlarına karşı daha duyarsız olmasını sağlar. Kapanma görüntü işleme sistemlerinde nesne tespiti için ortak bir problemdir.

Sınıflandırma için SVM kullanılmıştır. SVM lineer bir sınıflandırıcıdır ve danışmanlı öğrenme modelidir. SVM yönteminin önerilmesindeki temel amaç

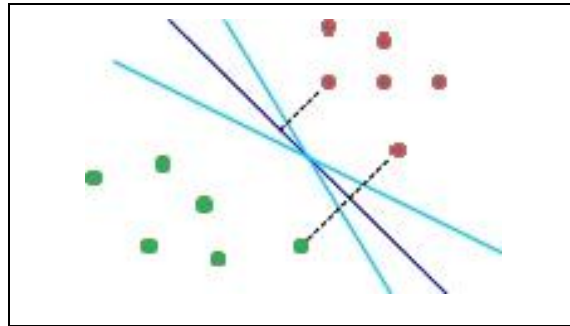
yüksek boyutlu verilerin doğru bir şekilde sınıflandırılabilmesini sağlamaktır. SVM veriyi sınıflandırırken sınıfların birbirlerine en yakın örneklerini kullanarak bu örneklerin ayırıcı yüzeye dik uzaklıklarını maksimize etmeyi amaçlar. Yani sınıfları lineer olarak ayıran belirlenebilecek farklı sınırlar arasından sınıflara uzaklıkları en fazla olan sınır karar sınırı olarak seçilir. SVM sayesinde ayırıcı yüzey her iki sınıfa da aynı mesafede ve maksimum uzaklıktadır. Bu seçim sistemin gürültülerden mümkün olduğunca etkilenmemesini sağlar.

SVM parametrik olmayan sınıflandırıcılardır. Veri setlerinin dağılımları hakkında herhangi bir ön bilgi varsayımı yoktur.

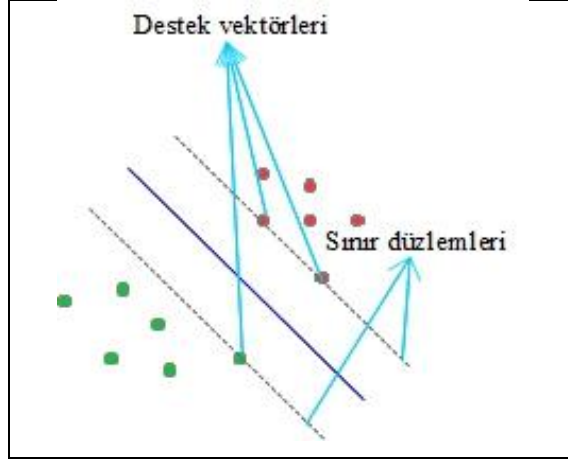
Lineer olarak ayıramayan öznitelikler için SVM çekirdek yöntemi kullanılır. Bu yöntem ile doğrusal olmayan karar sınırları bulunabilir.

Lineer SVM'lerde, p adet öznitelige sahip her x_i örneği için $y_i \in \{1, -1\}$ örneklerin ait oldukları sınıflar olmak üzere farklı sınıflara ait örnekleri birbirlerinden en iyi şekilde ayıracak, doğrusal hiper düzlemi bulunmaya çalışılır [38].

Burada w düzlemin normali ve ağırlık vektörü, b sabit bir sayı olarak tanımlanmıştır. Farklı sınıfları birbirinden ayıran birçok doğrusal düzlem bulunabilir. Ancak SVM destek vektörleri arasındaki uzaklığı maksimize eden hiper düzlemin bulunmasını amaçlar. Aşağıdaki şekillerde farklı ayırma düzlemleri ile iki sınıflı ve iki boyutlu bir sınıflandırma probleminin SVM ile ayrılması gösterilmiştir. Destek vektörleri her iki sınıfın da ayırma düzlemine en yakın olan noktalarıdır.



Şekil 3.8: İki sınıflı veri setini farklı şekilde ayıran düzlemler.



Şekil 3.9: İki sınıflı SVM örneği.

Destek vektörleri ait oldukları sınıfın sınırında ve ayırma düzlemine paralel düzlemler üzerinde bulunurlar [39]. Destek vektörlerinin üzerinde bulunduğu düzlemlere sınır düzlemleri denilir ve bu düzlemlerin ortalarından geçen her iki düzleme de eşit olan düzleme ise hiper düzlem denir.

En büyük sınıra sahip ayırma düzleminin bulunabilmesi için veri setindeki tüm örneklerin aşağıdaki eşitsizlikleri sağlaması gerekir.

$$x_i w + b \geq +1, \quad y_i = +1 \text{ için} \quad (3.9)$$

$$x_i w + b \leq -1, \quad y_i = -1 \text{ için} \quad (3.10)$$

Bu iki eşitsizlik birleştirilirse aşağıdaki denklem elde edilir.

$$y_i(w x_i + b) - 1 \geq 0 \quad (3.11)$$

Bir hiper düzlem, w normali ve orijinden dik uzaklığı $\frac{|b|}{\|w\|}$ olmak üzere, aşağıdaki denklem ile ifade edilir.

$$w x + b = 0 \quad (3.12)$$

Sınır düzlemleri arasında herhangi bir eğitim örneği yer almamaktadır [39].

Bu çalışmada destek vektör makinelerinin tercih edilmesinin sebebi, uygulamasının son derece kolay olması ve performansının oldukça yüksek olmasıdır.

Bu çalışmada, nesne olan ve olmayan durumlar için tanımlanan iki sınıflı problem ele alınmıştır. SVM yapısının oluşturulması için her sınıfa ait video karelerinden çıkartılan öznelikler kullanılır. Daha sonra, oluşturulan SVM yapısı, kontrol edilmek istenilen nesneyi içermesi beklenen görüntülerde kullanılır. Bu görüntülerde belli kare aralıkları ile nesnenin bulunup bulunmadığı incelenir. Nesnenin olmadığı durumlar tespit edildiğinde, sistem nesnenin olmadığı süreye bağlı olarak alarm üretebilir. Bu sürenin belirlenmesi için sınıflandırılan karelerin belli bir miktarı kullanılır. İstenildiği kadar uzun ya da kısa bir süre seçilebilir. Nesnenin tamamen kapandığı durumlar da nesnenin bulunmadığı durum olarak değerlendirilmiştir.

Kullanılan videoların her birinde videonun belirli zamanlarından, eğitim veri seti için öznelik çıkartımı yapılmıştır. Eğitim verileri birinci sınıf için operatör ekrandan ilgilenilen bölgeyi seçtikten sonra nesnenin bulunduğu belli bir aralıktan, ikinci sınıf için ise nesnenin bulunduğu yerden kaldırılması ile aynı ilgilenilen bölgeden belli bir aralıktan çıkarılır. Sistemin eğitim süresi, istenildiği kadar uzun tutulabilir. Öznelik çıkartımı yapılırken, farklı örneklerin birbirleri ile çok benzer olmamalarını sağlamak için belli aralıklarla kareler seçilmiştir. Bu aralık, tez kapsamındaki çalışılan videolar için 50 kare olarak belirlenmiştir. Belirlenen aralıkların her 50 karesinden çıkartılan öznelikler, öznelik seçimi işlemi uygulandıktan sonra SVM yapısı oluşturmak için kullanılmıştır. 50 kare aralık daha uzun seçilebilir. İstenilirse 1 gün boyunca çekilmiş bir videodan yarım saat aralıklarla da öznelik çıkartılabilir. Aralık seçmenin amacı ortamdaki ışık değişimleri, gölgeler, yansımalar gibi durumlardan eğitim verisi çıkartmaktır. Özellikle kapalı alanlarda yansıma ve gölgeler çok sık görülmektedir.

SVM yapısı oluşturulurken, eğitim seti %20 jackknife yöntemi ile test edilmiştir. Eğitim seti içerisindeki verilerin, en iyi sınıflandırma performansını veren %80'lik kısmı seçilmeye çalışılmıştır. Bu amaçla etiketleri bilinen eğitim veri seti öncelikle her sınıf için ayrı ayrı karıştırılmıştır. Daha sonra karıştırılan veri setlerinin %80'lik kısımları ile SVM yapıları oluşturulmuştur. Oluşturulan yapılar ile kalan %20'lik doğrulama veri seti, SVM sınıflandırıcısı kullanılarak sınıflandırılmıştır. Doğrulama veri seti kullanılarak, yapılan sınıflandırmanın sonuçlarına göre sınıflandırıcının performansı değerlendirilmiştir. %20 jackknife yöntemi kullanılarak, yapılan bu değerlendirmede toplam 5 farklı yapı ve 5 farklı sınıflandırıcı performansı elde edilmiştir. Sınıflandırıcıların performansları, doğru tespit oranlarına (correct rate) göre kıyaslanmıştır. Farklı eğitim verileriyle

oluşturulan sınıflandırıcıların performansları sıralandıktan sonra, en yüksek doğru tespit oranını veren eğitim veri seti ele alınmıştır. Bu eğitim veri seti ile SVM yapısı oluşturulmuştur. SVM yapısı kullanılarak, seçilen bölgenin bir parçası sınıflandırılmıştır. SVM yapısı seçme işlemi ilgilenilen bölgenin her bir parçası için ayrı ayrı yapılmıştır. Yani 4x4 toplam 16 parçaya bölünmüş bir bölge için farklı 16 adet SVM yapısı oluşturulur. Yapılan sınıflandırma işlemi ise videonun her bir karesi için yapılmaz, belli aralıklarla yapılır. Çünkü her karede nesnenin durumunu etkileyecek önemli bir değişiklik olmayacaktır. Sistemin işlem yükünü azaltmak ve önemsiz alarmlar oluşturmamak için belli aralıklarla sınıflandırma yapılmıştır.

Son durumda parçalara bölünmüş halde bulunan ilgilenilen bölgenin, her bir parçası ayrı ayrı sınıflandırılmış halde bulunmaktadır. Daha sonra ilgilenilen bölge için bölünmüş parçaların sınıflandırma sonuçları tek tek incelenir. En fazla hangi sınıf sonucu oluştuysa o kare için nesnenin durumu atanır. Nesnenin yarısı veya daha büyük bir kısmı kapanmaya uğramış olabilir. Kapanmanın büyüklüğüne göre nesne var veya yok diye sınıflandırılacaktır. Bu büyüklük ise nesnenin önemine göre değişebilir. Bu önem derecesine göre nesnenin yarısının veya farklı oranlarının kapanmaya uğraması durumunda nesne yok demek kullanıcının isteğine göre belirlenebilir. Ya da kapatılmaması istenilen bölgenin ne kadarını kapatan bir nesneye izin verilebileceği de tercih konusudur.

Ayrıca sistem nesnenin kısa süreli kaybolmalarına izin verecek şekilde tasarlanmıştır. Bu da nesnenin kısa süreli olarak tamamen kapanma gibi durumlarında sistemin alarm vermesini engeller. Bu eşik değeri de kullanıcı tarafından seçilebilir. Tez kapsamında çalışılan videolar için bu eşik değeri 25 kare olarak seçilmiştir. Yani sistem ancak 25 kare boyunca nesne yok olarak sınıflandırıldığında alarm üretmektedir. Bu 25 kare videonun işlendiği kare aralığıyla çarpıldığında gerçek aralığı vermektedir. Diğer bir deyişle 5 kare aralıklarla sınıflandırma yapılan bir videoda, 25 kare alarm verilmesi için seçilen süre olduğunda sistem gerçekte 125 karede alarm üretmektedir. Bu 125 kare, 25 fps hızla çalışan bir video için 5sn demektir. Kontrol edilmek istenilen nesne kıymetli olduğunda alarm üretilmesi için seçilen süre bu şekilde seçilebilir. Gerçek zamanlı olarak çalışacak bir sistemde alarm üretilmesi için gereken süre istenildiği kadar uzatılabilir.

3.4. Performans Ölçme

Uygulamanın sonuçları değerlendirilirken 3.3. bölümde anlatılan eşik değeri uygulanmış sınıflandırma sonuçlarının doğru tespit oranları kullanılmıştır. Bunun için öncelikle her videonun kesin bilgileri çıkartılmıştır. Daha sonra sınıflandırma sonuçları her kare için bu kesin bilgiler ile karşılaştırılarak sınıflandırıcının doğru tespit oranları, doğru pozitif oranları ve yanlış pozitif oranları hesaplanmıştır.

Doğru pozitif oranları sistemin nesnenin ortamda bulunduğu kareleri doğru olarak tespit etme oranıdır. Doğru negatif oranları ise sistemin nesnenin ortamda bulunmadığı kareleri doğru olarak tespit etme oranıdır. Bu oranlar ise aşağıdaki denklemlerle hesaplanmıştır.

- DT: Doğru tespit oranı
- DP: Doğru pozitif tespit sayısı
- DN: Doğru negatif tespit sayısı
- YP: Yanlış pozitif tespit sayısı
- YN: Yanlış negatif tespit sayısı

$$DT = \frac{DP+DN}{DP+DN+YP+YN} \quad (3.13)$$

$$DP \text{ Oranı} = \frac{DP}{DP+YN} \quad (3.14)$$

$$DN \text{ Oranı} = \frac{DN}{DN+YP} \quad (3.15)$$

4. SAYISAL ÖRNEK VE DEĞERLENDİRMELER

Bu çalışma kapsamında önerilen kayıp nesne tespit algoritmasının farklı veri setleri üzerinde sağladığı performansı ve başka algoritmalarla karşılaştırıldığında üstünlük sağladığı veya başarısız kaldığı durumların analizi bu bölümde ele alınmaktadır. Öncelikle, önerilen yöntemin ışık değişimleri, gölgeler ve kısmi kapanma durumlarında dahi klasik kayıp nesne tespit algoritmalarına göre yanlış alarm kriteri açısından değerlendirmesi yapılmıştır. Önerilen yöntemin nesnenin boyutlarından bağımsız olarak çalıştığını göstermek üzere aynı video farklı boyutlarda ölçeklendirilmiş ve algoritmanın performansı farklı ölçekler için karşılaştırılmıştır. Işık değişimlerine karşı önerilen yöntemin duyarlı olup olmadığını analiz etmek üzere aynı nesne için günün farklı saatlerinde alınan video kayıtları kullanılmıştır. Farklı HOG parametrelerinin ve farklı sayıda bölümlenmelerin sistem performansı üzerine etkileri de ayrıca incelenmiştir.

Çalışmanın değerlendirilmesi için önerilen yöntem AVSS ve PETS veri setlerinden alınmış 12 farklı video üzerinde denenmiştir. Ayrıca çekmiş olduğumuz 4 farklı videoda toplam 5 ayrı nesne için de sonuçlar alınmıştır.

AVSS Easy videosunda videonun başında sahne boşken bir süre sonra bir bavul getirilmekte bir süre bekletildikten sonra götürülmektedir. Bu video operatörün dikkatini çeken sahipsiz nesnenin belli bir süre içinde sahibi tarafından alınmadığında alarm üretilmesi istediği duruma örnek olarak verilebilir. Operatör bu sayede bavulu sürekli takip etmek zorunda kalmayacak veya dikkati dağıldığında bavulu takip etmeyi unutmayacak, operatörün belirlediği süre içinde bavul o bölgeden kaldırılmadığında alarm üreterek operatörün o ekrana veya bölgeye tekrar odaklanmasını sağlayacaktır. Video metroda çekilmiştir ve kapalı alandaki aydınlatmalardan ve sürekli dolaşan insanlardan kaynaklanan yansımalar ve kısmi kapanmalar mevcuttur. AVSS Medium1 videosunda ise aynı metroda videonun başında sahnede bulunan bir çantanın bir süre sonra götürüldüğü görülmektedir. Bu videoda operatörün ortamdaki alınmasını istemeyeceği bir nesnenin işaretlenmesi söz konusudur. AVSS Easy videosu ile aynı koşullarda, daha küçük bir boyutta ve daha fazla kısmi kapanma içeren bir nesnenin tespit edilebilmesinin gösterilmesi amacı ile kullanılmıştır. AVSS Medium2 videosunda ise AVSS Easy videosundakine benzer bir senaryo bulunmaktadır. Ancak nesnenin büyüklüğü AVSS Easy videosundaki nesneden yaklaşık 10 kat daha küçüktür.

AVSS veri setinden ayrıca park yasağı olan yerlere park etmiş araçların görüntüsü bulunan 4 farklı video alınmıştır. Bunlardan PV Easy videosunda yol kenarında duraklama yapan bir minibüs bulunmaktadır. Video, park yasağı olan bir yere araç park edilmesi durumunda, operatörün aracı işaretleyerek bir süre sonra bulunduğu yerden alınmaması durumunda sistemin alarm üretmesi durumuna örnek olarak verilebilir. Bu videodaki aracın rengi arka plandan farklı ve beyazdır. PV Medium videosunda ise aynı yerde duraklama yapan bir araba görüntüsü bulunmaktadır. Bu videodaki aracın rengi ise siyah olup, arka plan rengine yakındır, boyutu PV Easy videosundaki aracın büyüklüğünün yaklaşık 2 katıdır ve PV Medium videosunda, PV Easy videosundan daha fazla ışık değişimi görülmektedir. PV Hard videosunda ise uzak bir alanda duraklama yapan bir araba bulunmaktadır. Bu araç PV Easy videosundaki araçtan yaklaşık 5 kat daha küçük görünmektedir ve ışık değişimi daha fazladır. PV Night videosu ise yakın alanda yol kenarında duraklama yapan bir araba görüntüsü içeren gece çekilmiş bir videodur. Araçların farlarından kaynaklı olarak çok fazla ışık değişimi olmaktadır. PV videolarında ayrıca rüzgârdan kaynaklı olarak kamerada titreme olmaktadır ve bu videolar görüntü kalitesi düşük videolardır.

PETS 2001 veri setinden 3 video, PETS 2006 veri setinden 3 video olmak üzere toplam 6 video kullanılmıştır. PETS 2001 veri setinden alınan videolardan PETS2001 D1 C1 videosunda sahnede bulunan bir araç bir süre sonra gitmekte ve daha sonra geri gelip aynı yere farklı bir açı ile park etmektedir. Bu video belirli bir aracın bulunduğu yerden alınmak istemediği durumlarda, bu aracın bir operatör tarafından izlenerek takip edilmesi yerine, bu uygulama kullanılarak alarm üretmesi durumuna örnek verilebilir. Ayrıca, aracın bir süre sonra aynı yere farklı bir açı ile park edilmesi, uygulamanın nesnenin küçük hareketlerinde alarm üretip üretmeyeceğinin incelenmesini sağlamaktadır. PETS2001 D1 C2 videosu aynı sahnenin başka bir kamera ile farklı bir açıdan ve uzaklıktan çekilmiş halidir. PETS2001 D1 C2 videosundaki nesnenin büyüklüğü PETS2001 D1 C1 videosundaki nesnenin büyüklüğünün yaklaşık olarak 3 katıdır. PETS2001 D4 C3 videosunda ise aynı otoparkta bulunan başka bir araç bir süre sonra gitmektedir. Bu videoda bulunduğu yerden alınmaması gereken araç senaryosunun başka bir örneğidir. Bu videolarda araçların önünden gelip geçen insanlar ve yoldan geçen arabalar bulunmaktadır. PETS 2006 veri setinde ise sahnede bulunan ve bir süre sonra alınacak bir sırt çantasının farklı açılardan ve uzaklıklardan çekilmiş görüntüleri

bulunmaktadır. Bu videolar da kapalı bir alanda çekilmiştir ve yansımalar ve dolaşan insanlar bulunmaktadır. Bu videolar aynı nesnenin farklı açılarda ve boyutlardaki görüntülerinde uygulamanın başarısını test etmek için kullanılmıştır.

Kendi çektiğimiz videoların birinde 2 bank ve 1 araç bulunmaktadır. Nesnelere bir süre sonra sahneden kaldırılmıştır. Yöntem 3 nesne için de ayrı ayrı değerlendirilmiştir. Sonuçlar GTU Bench1, GTU Bench2 ve GTU Car video isimleri ile verilmiştir. Video bulutlu bir havada çekilmiştir ve nesnelere zaman zaman güneş ışığında zaman zaman da gölgede kalmaktadır. Video süresince banklara oturan insanlar bulunmaktadır. Ayrıca rüzgârdan kaynaklı kamera titremesi olan aralıklar bulunmaktadır. Bu videolardan GTU Bench1 ve GTU Bench2 uygulamanın kısmi kapanmalar ve ışık değişimlerindeki sonuçlarının incelenmesi için kullanılmıştır.

Çekilen diğer videoda ise kapalı bir ofiste bir bisiklet ve caraskal bulunmaktadır. Sonuçlar GTU Office Bike ve GTU Office Caraskal video isimleri ile verilmiştir. Video süresince ışık değişimleri sağlanmıştır ve sahnede dolaşan insanlar bulunmaktadır. Ayrıca zaman zaman nesnelere önüne sandalye, çanta gibi kısmi kapanmaya sebep olan cisimler bırakılmıştır. Videoda bir süre sonra bisiklet ve caraskal da sahneden çıkarılmış ve bir süre sonra da geri getirilmişlerdir. Bu videolar da ışık değişimleri ve kısmi kapanma durumlarının incelenmesi için kullanılmıştır.

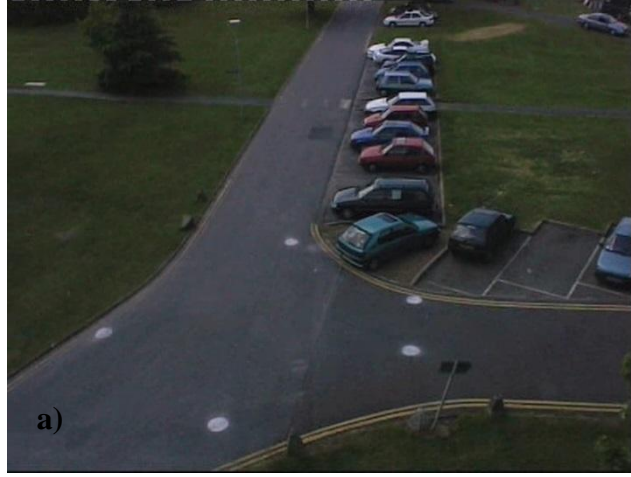
GTU Corridor Bike ve GTU Office Caraskal2 ise bisiklet ve caraskal nesnelere farklı ortamlarda incelenmeleri için çekilmiş videolardır. GTU Office Caraskal2 ışık değişimleri ve kısmi kapanmalar olan bir video, GTU Corridor Bike ise ışık değişimleri ve kısmi kapanmaların daha az olduğu bir videodur. Bu videoların kullanım amacı ise nesnelere şekillerinin uygulamanın sonuçlarına etkisini incelemektir. GTU Office Caraskal2 doğrusal bir şekle sahip olan caraskal için kısmi kapanmalar ve ışık değişimleri içeren bir video, GTU Corridor Bike ise doğrusal olmayan şekle sahip olan bisiklet için kısmi kapanmalar ve ışık değişimlerinin az olduğu bir videodur.

Tablo 4.1’de videoların özellikleri ile ilgili bilgiler verilmektedir. Işık değişimi ilgililenen bölgedeki bir pikselin, gri seviyeli görüntüde, video boyunca yaklaşık olarak aldığı en düşük ve en yüksek değer arasındaki farktır. Nesnenin boyutu ise ilgililenen bölgenin yaklaşık piksel büyüklüğüdür. Kayıp nesne olmaması ise, o video için şüpheli durumların incelenmesi anlamına gelmektedir.

Tablo 4.1: Kullanılan videoların özellikleri.

Video	Işık Değişimi	Kısmi Kapanma	Nesnenin Boyutu	Kayıp Nesne
AVSS Easy	150	Var	11400	Hayır
AVSS Medium1	140	Var	4000	Evet
AVSS Medium2	130	Var	1150	Hayır
PETS2001 D1 C1	40	Yok	3350	Evet
PETS2001 D1 C2	30	Yok	10250	Evet
PETS2001 D4 C1	30	Yok	3340	Evet
PETS2006 D1 C1	120	Yok	750	Evet
PETS2006 D1 C3	80	Yok	1250	Evet
PETS2006 D1 C4	45	Yok	950	Evet
PV Easy	130	Yok	1590	Hayır
PV Medium	150	Yok	780	Hayır
PV Hard	180	Yok	350	Hayır
PV Night	250	Yok	10400	Hayır
GTUOfficeBike	180	Var	40725	Evet
GTUCorrBike	100	Var	20600	Evet
GTUOffCaraskal	150	Var	83300	Evet
GTUOffCaraskal2	110	Var	32500	Evet
GTU Bench1	200	Var	4200	Evet
GTU Bench2	220	Var	7400	Evet
GTU Car	100	Yok	13200	Evet
GTU Way	50	Yok	3500	Evet
GTU Park	100	Var	600	Hayır

Şekil 4.1’de PETS 2001 veri setinden bir videoda algoritmanın uygulanışı görülmektedir. İlk resimde nesnenin seçildiği kare, ikincisinde ise alarm durumunda çıkan uyarı görülmektedir. Köşede duran turkuaz renkli arabanın park alanını terk ettiği zaman operatörün uyarılması beklenmektedir.



Şekil 4.1: Örnek bir veri setinde kayıp nesne tespiti. a) nesne bulunan görüntü, b) nesne bulunmayan görüntü.

Sistem, araç park edildiği konumun dışına alındığında alarm üretecektir. Yer değiştiren aracın orijinal park pozisyonu kırmızı bir kutu ile işaretlenecektir. Bu kırmızı kutular potansiyel olarak nesnenin kayıp olduğu durumları göstermektedir.

Sistemin değerlendirilmesi için doğru tespit oranları kullanılmıştır. Bu oran videolarda nesnenin bulunduğu ve bulunmadığı durumlarda uygulamanın ne kadar doğru tespit yaptığının bir göstergesidir.

Sistem çalıştırıldığı videoların hepsinde iyi sonuçlar vermiştir. Kayıp nesnelere tespit edebilmekte ve iyi tahminlerde bulunmaktadır.

Önerilen sistemin diğer sistemlere göre avantajları şunlardır:

- Nesne şeklinden ve büyüklüğünden bağımsız olarak tespit edilebilmektedir.
- Işık değişimleri ve görüntü kalitesinden etkilenmemektedir.

- Sistem gerçek zamanlı olarak çalışabilmektedir
- Kısmi kapanma durumlarında da iyi sonuçlar vermektedir.

Önerilen sistemin dezavantajları ise şunlardır:

- Sistem klasik yöntemlerden farklı olarak bir öğrenme aşaması içermektedir.
- Sistem ilgilenilen bölgede bir değişiklik olduğunda, kullanıcı kapatmadığı sürece alarm üretmektedir.

Sistem, çalışırken nesnenin kısmen kapanması veya geçici olarak önünden birinin geçmesi gibi durumları dikkate alarak bir olay tespiti yapıldığında anında alarm üretmek yerine olay tespitinin süreklilik arz etmesi durumunda alarm üretir. Bu çalışmada, tespit edilen olay için alarm üretme süresi operatöre bırakılmıştır. Operatörün istediği süreden daha kısa süren olay tespitlerinde alarm üretilmemektedir.

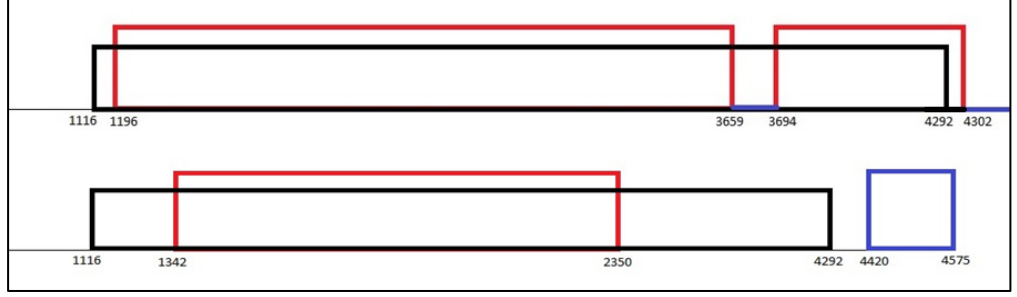
4.1. Klasik Kayıp Nesne Tespit Algoritmaları

Bu bölümde önerilen yöntemin klasik arka plan çıkarımı yöntemleri ile kıyaslaması yapılmıştır. Kıyaslama, her yöntem için, videoda gerçekleşen olayın kaç kare sonra tespit edildiğine göre yapılmıştır.

Önerilen yöntem [4] ve [1]'de uygulanan klasik arka plan çıkarımı yöntemlerine dayalı kayıp nesne tespit algoritmaları ile karşılaştırılmıştır. Ayrıca [7]'de uygulanan yöntem de denenmiştir. Karşılaştırmalar tespit zamanları açısından yapılmıştır. Bu yöntemler ikili arka plan çıkarımını temel almakla birlikte arka planın modellenmesi ve güncellenmesi açısından farklılık gösterirler. Statik nesne tespiti yapıldıktan sonra tespit edilen bölgenin renk çeşitliğine dayanarak terkedilmiş veya kayıp nesne olduğuna karar verilir. Şekil 4.2'de [7]'de uygulanan yöntemin AVSS Medium videosundaki tespit zamanlarının önerilen yöntem ile kıyaslanması gösterilmektedir.

Şekilde siyah çizgi videoda nesnenin bırakıldığı ve ortamda olduğu kare aralığını göstermektedir. Kırmızı çizgi nesnenin ortamda bulunduğu tespit aralığını göstermektedir. Mavi çizgi ise nesnenin ortamdaki tespit aralığını

göstermektedir. Önerilen yöntemde 3659 ile 3694 kare aralıklarında kısmi kapanmalardan dolayı yanlış alarm üretilmektedir. Nesnenin tespit zamanı önerilen yöntemde 55 kare gecikmekteyken makaledeki uygulamada 101 kare gecikmektedir. Önerilen yöntem her karede sınıflandırma yaptığı için nesnenin bulunmadığı süre boyunca alarm üretilmekte iken, makaledeki yöntemde statik bölge bir süre sonra ara plana karışmakta ve alarm kapanmaktadır.



Şekil 4.2: Önerilen yöntem (üstte), makaledeki yöntem (altta).

Tablo 4.2’de ise önerilen yöntem ile [4] ve [1] makalelerindeki yöntemlerin AVSS ve PETS2006 veri setlerinin bazı videolarında tespitlerinin gecikme süreleri görülmektedir.

Tablo 4.2: Önerilen yöntem ile klasik algoritmaların kayıp nesne tespit zamanları.

Video	[4]	[1]	Önerilen Yöntem
AVSS Easy	540	230	10
AVSS Medium	1620	200	55
PETS2006 D1 C1	201	301	3
PETS2006 D1 C3	231	321	5

Tablo 4.2’den de görüldüğü gibi klasik arka plan çıkarımı yöntemleri ile kayıp veya terkedilmiş nesne tespit süresi sınıflandırma işlemi ile tespit süresinden, kare hızı olarak daha yavaş olmaktadır.

4.2. HOG Parametreleri

Sistemin farklı HOG hücre boyutu ve blok boyutu parametreleri kullanıldığında performans değerleri ölçülmüş ve sonuçları tablolar halinde verilmiştir. Bu parametrelerin sistem performansına etkilerini incelemek için öncelikle kısmi kapanma durumlarının ve ışık değişimlerinin az olduğu videolar kullanılmıştır. Kullanılan videolarda ilgilenilen bölge seçildikten sonra ızgaralara bölünmemiştir. Bu durumlarda bulunan parametre değerleri sistemin ilgilenilen bölgenin ızgaralara ayrıldığı durumlarda kullanılacaktır.

İlk olarak Gamma dönüşümünün sistemin performansına etkisi incelenmiştir. 3. bölümde belirtildiği gibi Gamma dönüşümü HOG öznelikleri çıkarımı için görüntüye uygulanan bir ön işlemdir. Ancak pek çok çalışmada Gamma dönüşümü, sistemin performansına katkısı olmadığı ya da çok küçük bir katkısı olduğu gerekçesi ile uygulanmamıştır. Bu nedenle önerilen yöntemde Gamma dönüşümünün sistemin performansına etkisini görmek amacı ile Gamma dönüşümünün yapıldığı ve yapılmadığı durumlarda sistemin performansı ölçülmüştür.

Tablo 4.3: Gamma Dönüşümünün etkisi.

Kullanılan video	Gamma Dönüşümü Yapıldığında	Gamma Dönüşümü Yapılmadığında
PETS2001 D1C2	%63,5	%63,5
PETS2001 D4 C1	%98,1	%98,2
PETS2006 D1 C1	%99,8	%99,8
PETS2006 D1 C3	%99,2	%99,2

Tablo 4.3'te sistemin 4 farklı video için 2 durumdaki performans sonuçları verilmiştir. Sonuçta gamma dönüşümünün performansa bazı durumlarda çok küçük bir etkisi olduğu görülmüş ve daha sonraki testlerde gamma dönüşümü uygulanmamıştır.

Daha sonra sistemin aynı videolarda farklı hücre ve blok boyutları için doğru tespit oranları. Tablo 4.4'te hücre boyutunun 6 ve 8, blok boyutunun ise 2 ve 4 olduğu durumlarda sistemin performansı görülmektedir.

Tablonun sonuçlarından ise blok sayısını arttırmanın sistemin performansına olumlu bir etkisi olduğu görülmüştür. Hücre boyutu 16 ve blok boyutu da 8 veya 16

seçilebilir ancak bu sayılar resmin belli bir parçasından öznitelik yapılacağı durumlar için uygun değildir. İşlem sayısını artırmakla birlikte küçük parçalardan uygun öznitelik çıkarılmasına engel olurlar.

Tablo 4.4: HOG parametrelerinin sistemin performansına etkileri.

Kullanılan video	Hücre boyutu: 6 Blok boyutu: 2	Hücre boyutu: 6 Blok boyutu: 4	Hücre boyutu: 8 Blok boyutu: 2	Hücre boyutu: 8 Blok boyutu: 4
PETS2001 D1 C2	%63,5	%84,8	%63,5	%98,4
PETS2001 D4 C1	%99,3	%98	%98,3	%96,7
PETS2006 D1 C1	%99,8	%99,8	%99,8	%99,8
PETS2006 D1 C3	%99	%99,8	%99	%99

4.3. Öznitelik Sayısı

Bu bölümde sistemde uygun sayıda öznitelik kullanılması için farklı sayıda özniteliklerin sistemin performansına etkisi incelenmiştir. Bu işlemler için 3. bölümde anlatılan öznitelik sıralama yöntemi kullanılmıştır. HOG parametreleri ise 4.2. bölümde bahsedildiği gibi hücre boyutu 8 ve blok boyutu 4 seçilmiştir.

Sistemde öznitelik seçimi yapılırken seçilen öznitelik sayısı 10, 15 ve 20 olarak denenmiş ve sonuçları incelenmiştir. Tablo 4.5'te uygun parametrelerinin seçimi için kullanılan videolar kullanılarak yapılan testlerin sonuçları görülmektedir.

Tablo 4.5: Farklı sayıda öznitelik seçimi ile yapılan test sonuçları.

Kullanılan video	10 Öznitelik	15 Öznitelik	20 Öznitelik
PETS2001 D1 C2	%98,3	%70,3	%63,9
PETS2001 D4 C1	%96,7	%96,7	%97
PETS2006 D1 C1	%99,8	%99,8	%99,8
PETS2006 D1 C3	%99,2	%99,3	%99,6

Tablodaki sonuçlara bakılarak 10 öznitelik seçiminin daha iyi sonuçlar verdikleri görülmektedir. Bu nedenle daha sonraki çalışmalarda öznitelik sayısı 10 olarak belirlenmiştir.

4.4. İlgilenilen Bölgenin Parçalara Ayrılması

Kısmi kapanmalar ve ışık değişimleri gibi durumlarda sistemin doğru tespit oranlarının artması için ilgilenilen bölge parçalara ayrılmıştır. İlgilenilen bölgenin farklı sayıda parçalara ayrılması ile sistemin doğru tespit oranları ölçülmüştür. Bölgenin kaç parçaya ayrılacağı ilgilenilen bölgenin büyüklüğü ile ilgilidir. Örneğin 31x44 piksel değerlerine yakın boyutlardaki bir ilgilenilen bölge 2x3 parçaya ayrılabilirken, 204x252 piksel değerlerine yakın bir bölgenin 4x4 veya 3x4 bölgeye ayrılması gerekir. 438x246 piksel boyutlarına yakın ve daha büyük piksel boyutlarına sahip ilgilenilen bölgeler için 5x4 parçaya bölmek uygun olabilir. Ancak bu büyüklüklere sahip ilgilenilen bölgelerin daha fazla parçaya ayrılması avantaj sağlamayacağı gibi işlem yükünü de arttırır. Yaklaşık 30x40 piksel boyutlarındaki ilgilenilen bölgeler için ise bölgelere ayırma işlemi yapılırsa da yapılmazsa da sonuçlar çok farklı olmamaktadır. Çünkü küçük nesnelerin tamamen kapanmaya uğrama durumları kısmi kapanmaya uğrama durumlarına göre daha çok görülür ve bu durumda nesnenin parçalara ayrılma durumu anlamlı olmayacaktır. Bu bölümdeki testler için kısmi kapanmalar ve ışık değişimleri olan videolar kullanılmıştır. İlgilenilen bölgelerin parçalara ayrıldığı ve ayrılmadığı, kısmi kapanma içeren durumlarda sınıflandırıcının performans değerleri incelenmiştir.

Tablo 4.6’da küçük boyutlu 3 nesne için parçalara ayırmanın performansa etkisi görülmektedir.

Tablo 4.7’de ise bazı büyük boyutlu ve kısmi kapanma içeren videoların en yakın komşu ile sınıflandırma yapıldığında ızgara olan ve olmayan durumlar için bulunan performans değerleri görülmektedir. Bu değerler 8 hücre boyutu, 4 blok boyutu ve 10 öznitelik kullanıldığında bulunan değerlerdir.

Tablo 4.6’daki sonuçların yakın olmaları ilgilenilen bölgenin boyutlarının küçük (yaklaşık 30x40piksel) olmasındandır.

Yapılan testlere dayanarak ilgilenilen bölgeyi parçalara ayırmanın sonuçları iyileştirdiği ancak bir yerden sonra fazla bir değişim olmadığı söylenebilir. Ayrıca ilgilenilen bölge parçalara ayrılmadığında öznitelik çıkarımı için geçen süre,

parçalara ayrılmış durumdaki ilgilenilen bölgeden öznitelik çıkarımı için gereken süreden daha uzun olmaktadır.

Tablo 4.6: Küçük nesnelere için ilgilenilen bölgenin parçalara bölünmesinin performansa etkisi.

Video	İlgilenilen bölgenin parçalara ayrıldığı durum	İlgilenilen bölgenin parçalara ayrıldığı durum
AVSS AB Medium2	%80	%90
PETS2001 D1 C1	%95	%98
PETS2006 D1 C4	%94	%95

Tablo 4.7: İlgilenilen bölgenin ızgaralara ayrılmasının etkisi (NN).

Video	Doğru Pozitif Oranı		Doğru Negatif Oranı		Doğru Tespit Oranı	
	Izgaralı	Izgarasız	Izgaralı	Izgarasız	Izgaralı	Izgarasız
AVSS Medium2	%97.79	%98.33	%86.90	%46.49	%94.60	%76.22
PV Night	%94.43	%80.01	%17.75	%33.31	%41.06	%35.62
GTU OfficeBike	%89.23	%63.74	%52.22	%27.87	%74.76	%48.49
GTU CorrBike	%100	%51.86	%77.93	%80.08	%96.21	%71.50
GTU OffCaraskal	%78.37	%38.72	%98.64	%70.73	%89.04	%53.01

4.5. SVM Uygulamanın Katkısı

Tez çalışması kapsamında makine öğrenmesinin gerekliliğini görmek amacıyla video karelerini sınıflandırma işlemi en yakın komşu kuralı ile de yapılmıştır. Sonuçlar SVM ile yapılmış sınıflandırmaların sonuçları ile karşılaştırılmıştır. En yakın komşu kuralı uygulamasında nesnenin bulunduğu, kısmi kapanmalar içermeyen bir karenin öznitelikleri ile nesnenin bulunmadığı bir karenin öznitelikleri çıkartılmıştır. Daha sonra videonun karelerindeki ilgilenilen bölgenin öznitelikleri, bilinen karelerin özniteliklerine yakınlığına göre sınıflandırılmıştır. Sınıflandırılan parçaların çokluğuna dayalı oylama mekanizması aynı şekilde kullanılmıştır. Tablo 4.8'de iki yöntemin doğru tespit oranları görülmektedir.

Tablo 4.8'in sonuçlarında PETS2001 Dataset1 Camera1, PETS2006 Dataset1 Camera1, PV Medium, PV Hard, PV Night, GTU Bike Office ve GTU Bench1 videolarında %10'dan fazla iyileşme görülmektedir.

Tablo 4.8: En yakın komşu kuralı ve SVM ile sınıflandırmanın sonuçları.

Video	Doğru Pozitif Oranı		Doğru Negatif Oranı		Doğru Tespit Oranı	
	NN	SVM	NN	SVM	NN	SVM
AVSS Easy	%99.37	%99.55	%96.20	%96.07	%98.56	%98.79
AVSS Medium2	%97.79	%91.97	%86.90	%99.64	%94.60	%97.79
AVSS Medium1	%93.33	%99.84	%70.28	%95.07	%99.06	%99.04
PETS2001 D1 C1	%50.19	%96.54	%99.87	%100	%58.53	%97.98
PETS2001 D1 C2	%87.09	%96.12	%100	%100	%93.33	%97.91
PETS2001 D4 C1	%97.85	%95.60	%96.85	%100	%99.97	%98.02
PETS2006 D1 C1	%97.17	%95.08	%71.77	%100	%85.53	%98.87
PETS2006 D1 C3	%98.42	%95.25	%83.52	%100	%89.01	%98.94
PETS2006 D1 C4	%84.92	%95	%98.90	%100	%99.30	%98.94
PV Easy	%99.91	%98.17	%85.72	%100	%98.15	%98.63
PV Medium	%99.62	%99.25	%11.24	%90.58	%64.82	%94.36
PV Hard	%83.23	%96.55	%17.02	%98.45	%53.76	%98.08
PV Night	%94.43	%93.34	%17.75	%99.95	%41.06	%97.51
GTU OfficeBike	%89.23	%98.71	%52.22	%99.55	%74.76	%98.71
GTU CorrBike	%100	%96.02	%77.93	%100	%96.21	%99.93
GTU OffCaraskal	%78.37	%80.32	%98.64	%98.45	%89.04	%89.10
GTU OffCarask12	%99.28	%99.22	%94.89	%99.45	%98.58	%99.89
GTU Bench1	%88.24	%99.59	%61.46	%100	%89.52	%99.84
GTU Bench2	%97.90	%99.62	%96.42	%100	%99.87	%99.80
GTU Car	%98.37	%99.62	%96.21	%100	%99.85	%99.76
GTU Way	%99.50	%99.82	%100	%100	%99.86	%99.82
GTU Park	%87.27	%97.01	%97.19	%100	%98.56	%99.51

Bu videolardan PETS2001 Dataset1 Camera1 videosu park halindeki bir aracın bir süre sonra gitmesi ve daha sonra aynı yere tekrar park etmesinin görüntüsüdür. Araç tekrar park ettiğinde aracın duruş açısı değişmektedir. Bu durumda SVM algoritması aracı tespit edebilirken diğer algoritma tespit edememektedir. PETS2006

Dataset1 Camera1 videosunda ise en yakın komşu yönteminde, nesnenin boyutlarının küçük olması ve arka planın şekliinden dolayı, nesne kaldırıldığında alarm üretilmediği durumlar bulunmaktadır. PV Medium videosunda ise yol kenarında duraklama yapan araç yolun rengine yakın bir renktedir, kamera titremektedir ve ışık değişimleri bulunmaktadır. Bu durumlarda da en yakın komşu yönteminde aracın gittiği durum tespit edilememektedir. PV Hard videosunda ise araç kameradan uzakta bulunmaktadır ve görüntü net değildir. Bu durumda da aracın olmadığı durumlar tespit edilememektedir. PV Night videosunda ise görüntünün net olmaması ve araçların farlarından kaynaklı ışık değişimlerinden dolayı nesnenin olmadığı durumlar en yakın komşu yöntemi ile tespit edilememektedir. GTU Bike Office videosunda ise bisikletin şekli, ışık değişimleri ve kısmi kapanmalardan dolayı hatalar oluşmaktadır. GTU Bench1 videosunda ise yine kamera titremesi, ışık değişimleri ve kısmi kapanmalar bulunmaktadır. Bu nedenlerle bahsedilen videolardaki gibi senaryolarda SVM yöntemi sadece öznelik farkı uygulanmasından daha iyi sonuçlar vermektedir.

4.6. Oylama Yönteminin Etkisi

Bu bölümde 3.3. bölümünde anlatılan bir karenin ızgaralarının oylama yöntemi ile sınıflandırılması işlemine alternatif bir yöntem denenmiştir. Bu alternatif yöntemde videonun kareleri yine ızgaralara bölünmüştür. Daha sonra bu ızgaralardan çıkarılan öznelikler bir vektör şeklinde birleştirilmiş ve her bir kare tek parça halinde sınıflandırılmıştır. Bu test için en yakın komşu sınıflandırması kullanılmıştır. Tablo 4.9’da bu yöntemin sonuçları görülmektedir.

Sonuçların oylama mekanizmasından çok farklı olmadığı görülmüştür. Ancak bir video için bu yöntemin çalışması oylama mekanizmasının kullanılmasından çok daha uzun sürmektedir. Çünkü her bir parçanın özneliklerinin çıkarılarak vektör şekline getirilmesi, sınıflandırılıp, sınıflandırmanın sonucunu tutmaktan daha çok zaman almaktadır. Ayrıca normalde 10 öznelik ile yapılan sınıflandırma işlemi, bu durumda belirlenen ızgara sayısı ile öznelik sayısının çarpımı kadar bir öznelik vektörü ile yapılmaktadır. Bu da çalışma süresini uzatan nedenlerdendir.

Tablo 4.9: Oylama mekanizmasının etkisi (NN).

Video	Doğru Pozitif Oranı	Doğru Negatif Oranı	Doğru Tespit Oranı
AVSS Medium1	%93.45	%85.28	%99.06
AVSS Medium2	%97.51	%88.77	%98.05
PETS2006D1C1	%95.83	%99.01	%99.30
PETS2006 D1C3	%97.50	%99.06	%99.77
GTU BikeOffice	%86.81	%91.47	%90.21

Bu yöntemin doğruluğunun test edilmesi için çıkarılan öznitelik vektörüne tekrar bir öznitelik seçimi uygulanmıştır. İlgilenilen bölgenin öznitelik sayısı bir kare için 10'a düşürülmüştür. Daha sonra bu sınıflandırma işleminin sonuçları, ızgaralara ayrılmadan ilgilenilen bölgeden 10 öznitelik çıkarılması ile sınıflandırma yapıldığında, bulunan sonuçlar ile karşılaştırılmıştır. Kısaca her iki durumda da 1 kareden 10 öznitelik çıkartılmaktadır. Bu durumda sonuçların ızgaralara ayrılmadan yapılan sınıflandırma işleminden daha iyi olduğu görülmüştür. Çünkü alternatif yönteme tekrar bir öznitelik seçimi işlemi uygulandığında bilgi kaybı oluşmaktadır. Tablo 4.10'da bu iki yöntemin denendiği videolar için doğru tespit oranları görülmektedir.

Tablo 4.10: Parçaların özniteliklerini birleştirmenin etkisi.

Video	Izgaralardan çıkarılan öznitelikler	Izgaralama yapılmadan çıkarılan öznitelikler
AVSS Medium1	%40.81	%41.11
AVSS Medium2	%29.75	%87.58
PETS2006D1C1	%46.31	%58.29
PETS2006 D1C3	%38.46	%53.88
GTU BikeOffice	%26.37	%65.79

5. SONUÇLAR

Tez çalışması kapsamında klasik sınıflandırma algoritmaları ile kayıp nesne tespiti yapılmıştır. Sistemde klasik kayıp nesne tespit algoritmalarından farklı olarak nesnenin özniteliklerinden faydalanılmıştır.

Kullanılan öznitelikler için, özellikle insan tespiti gibi zor bir problemde başarılı olduğu bilinen ve farklı nesne tespitleri için de kullanılmış ve başarılı olduğu görülmüş HOG öznitelikleri kullanılmıştır.

Nesnenin tamamı için bir eğitim ve sınıflandırma işleminden farklı olarak parçalanmış kısımlarının sınıflandırılması fikri öne sürülmüştür. Bu sayede sistemin ışık değişimleri, kısmi kapanmalar, gölgeler gibi durumlara karşı duyarsız olması sağlanmıştır.

Sistemin testleri sırasında yaklaşık 450x250 ile yaklaşık 18x20 piksel büyüklüğündeki nesnelere değerlendirilmiştir. Bu piksellerden daha küçük boyutlardaki nesnelere için nesnenin parçalanarak sınıflandırılması işleminin gereksiz olduğu görülmüştür. Belirtilen piksellerden daha büyük boyutlardaki nesnelere için ise yaklaşık 5x5 parçadan sonra daha fazla parçaya ayırmanın gereksiz olduğu görülmüştür.

Sistemin makine öğrenmesine gerek duymadığı aynı öznitelikler ve ızgara sistemi kullanılarak en yakın komşu yöntemi ile test edilmiştir. Bu durumda nesnenin şeklinin belirgin olduğu ve ışık değişimlerinin az olduğu durumlarda sistemin makine öğrenmesine gerek duymadığı görülmüştür. Ancak doğrusal bir şekle sahip olmayan nesnelere, görüntü kalitesinden dolayı şekli belirgin olmayan nesnelere ve nesnenin renginin arka plana benzediği durumlarda makine öğrenmesi içeren sistemin doğru tespit oranlarının daha iyi olduğu görülmüştür. Ancak iki yöntemin çalışma zamanı açısından belirgin bir farkları yoktur.

6. GELECEKTEKİ ÇALIŞMALAR

Tez çalışması kapsamında ortamda bulunan belirli bir nesnenin korunması ile ilgilenilmiştir. İlerleyen çalışmalarda ortamdaki farklı ve birden fazla nesnenin seçilerek korunması düşünülmektedir. Ayrıca nesnenin bulunmadığı durum için nesnenin bulunduğu yerden alınması ile çıkarılan öznelilikler kullanılmaktadır. Nesnenin bulunmadığı durumlardaki özneliliklerin, ortamdaki nesnenin bulunmadığı herhangi bir yerden çıkarılarak kullanılması düşünülmektedir.

KAYNAKLAR

- [1] Wang W., Liu Z., (2010), "A new approach for real time detection of abandoned and stolen objects", International Conference on Electrical and Control Engineering, 128-131, Wuhan, China, 25-27 June.
- [2] Tian Y., Feris R. S., Liu H., Hampapur A., Sun M. T., (2011), "Robust detection of abandoned and removed objects in complex surveillance videos", IEEE Transactions on Systems, Man, and Cybernetics, 41 (5), 565-576.
- [3] Spagnolo P., Caroppo A., Leo M., Martiriggiano T., (2006), "An abandoned / removed objects detection algorithm and its evaluation on pets dataset", IEEE International Conference on Video and Signal Based Surveillance, 17, Sydney, Australia, November.
- [4] Porikli F., (2007), "Detection of temporarily static regions by processing video at different frame rates", IEEE Conference on Advanced Video and Signal Based Surveillance, 236-241, London, England, 5-7 September.
- [5] SanMiguel J. C., Martinez J. M., (2008), "Robust unattended and stolen object detection by fusing simple algorithms", IEEE 5th International Conference on Advanced Video and Signal Based Surveillance, 18-25, Santa Fe, NM, USA, 1-3 September.
- [6] Mathew R., Yu Z., Zhang J., (2005), "Detecting new stable objects in surveillance video", IEEE 7th Workshop on Multimedia Signal Processing, 1-4, Shanghai, China, 30 October-2 November.
- [7] Li Q., Mao Y., Wang Z., Xiang W., (2009), "Robust real time detection of abandoned and removed objects", 5th International Conference on Image and Graphics, 156-161, Xi'an, Shanxi, China, 20-23 September.
- [8] Liao H., Chang J., Chen L., (2008), "A localized approach to abandoned luggage detection with foreground mask sampling", IEEE 5th International Conference on Advanced Video and Signal Based Surveillance, 132-139, Santa Fe, NM, USA, 1-3 September.
- [9] Campos L. C., SanMiguel J. C., Martinez J. M., (2011), "Discrimination of abandoned and stolen object based on active contours", IEEE 8th International Conference on Advanced Video and Signal-Based Surveillance (AVSS), 101-106, Klagenfurt, Austria, 30 August-2 September.
- [10] Di Caterina G., Soraghan J. J., (2011), "An abandoned and removed object detection algorithm in a reactive smart surveillance system", 17th International Conference on Digital Signal Processing, 1-6, Corfu, Greece, 6-8 July.

- [11] Vural U., Akgül Y. S., (2009), “Eye-Gaze Based Real-Time Surveillance Video Synopsis”, *Journal of Pattern Recognition Letters*, 12 (30), 1151-1159, 12 September.
- [12] Wu B., Nevatia R., (2009), “Detection and segmentation of multiple partially occluded objects by grouping, merging, assigning part detection responses”, *International Journal of Computer Vision*, 82 (2), 185-204.
- [13] Pereira A., Saotome O., Sampaio D., (2015), “Patch based local histograms and contour estimation for static foreground classification”, *EUROSIP Journal on Image and Video Processing*, 2015 (1), 6.
- [14] Stauffer C., Grimson W. E. L., (1999), “Adaptive background mixture models for real-time tracking”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO, USA, 23-25 June.
- [15] Sigari M. H., Mozayani N., Pourreza H. R., (2008), “Fuzzy running average and fuzzy background subtraction: concepts and application”, *International Journal of Computer Science and Network Security*, 8 (2), 138-143.
- [16] Seki M., Wada T., Fujiwara H., Sumi K., (2003), “Background subtraction based on cooccurrence of image variations”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 65-72, Madison, WI, USA, 18-20 June.
- [17] Tang Z., Miano Z., Wan Y., (2007), “Background subtraction using running Gaussian average and frame difference”, *6th International Conference on Entertainment Computing*, 411-414, Shanghai, China, 15-17 September.
- [18] Singh A., Sawan S., Hanmandlu M., Madasu V. K., Lovell B. C., (2009), “An abandoned object detection system based on dual background segmentation”, *6th IEEE International Conference on Advanced Video and Signal Based Surveillance*, 352-357, Genova, Italy, 2-4 September.
- [19] Tian Y.-L., Lu M., Hampapur A., (2005), “Robust and efficient foreground analysis for real-time video surveillance”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1182-1187, San Diego, CA, USA, 20-25 June.
- [20] Porikli F., Ivanov Y., Haga T., (2008), “Robust abandoned object detection using dual foregrounds”, *EURASIP Journal on Advances in Signal Processing*, 2008 (30), 11.
- [21] Porikli F., Tuzel O., (2005), “Bayesian background modeling for foreground detection”, *3rd ACM International Workshop on Visual Surveillance and Sensor Networks*, 55-58, Singapore, 11 November.

- [22] Porikli F., Tuzel O., Meer P., (2005), "A bayesian approach to background modeling", IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 58, San Diego, CA, USA, 25 June.
- [23] Mittal A., Paragios N., (2004), "Motion-based background subtraction using adaptive kernel density estimation", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 302-309, Washington, DC, USA, 27 June-2 July.
- [24] Simoncelli E. P., (1999), "Bayesian multi-scale differential optical flow", 2, Handbook of Computer Vision and Applications, Academic Press.
- [25] Pepik B., Stark M., Gehler P., Schiele B., (2013), "Occlusion patterns for object class detection", IEEE Conference on Computer Vision and Pattern Recognition, 3286-3293, Portland, OR, USA 23-28 June.
- [26] Schneiderman H., Kanade T., (2000), "A statistical method for 3D object detection applied to faces and cars", IEEE Conference on Computer Vision and Pattern Recognition, 746-751, Hilton Head Island, SC, USA, 13-15 June.
- [27] Viola P., Jones M., (2001), "Rapid object detection using a boosted cascade of simple features", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 511-518, 1, Kauai, HI, USA, 8-14 December.
- [28] Huang C., Ai H., Li Y., Lao S., (2006), "Learning sparse features in granular space for multi view face detection", 7th International Conference on Automatic Face and Gesture Recognition, 401-406, Southampton, UK, 2-6 April.
- [29] Lowe D. G., (2004), "Distinctive image features from scale invariant keypoints", International Journal of Computer Vision, 60 (2), 91-110.
- [30] Dalal N., Triggs B., (2005), "Histograms of oriented gradients for human detection", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 886-893, San Diego, CA, USA, 25 June.
- [31] Tuzel O., Porikli F., Meer P., (2007), "Human detection via classification on riemannian manifolds", IEEE Conference on Computer Vision and Pattern Recognition, 1-8, Minneapolis, MN, USA, 17-22 June.
- [32] Prisacariu V. A., Reid I., (2012), "fastHOG a real time GPU implementation of HOG", Technical Report No: 2310/09, Department of Engineering Science, University of Oxford, UK.
- [33] Llorca D. F., Arroyo R., Sotelo M. A., (2013), "Vehicle logo recognition in traffic images using HOG features and SVM", 16th International IEEE Conference on Intelligent Transportation Systems, 2229-2234, The Hague, Netherlands, 6-9 October.

- [34] Freeman W. T., Roth M., (1995), "Orientation histograms for hand gesture recognition", IEEE Computer Society International Workshop on Automatic Face and Gesture Recognition, 296-301, Zurich, Switzerland, June.
- [35] Liu H., Motoda H., Setiono R., Zhao Z., (2010), "Feature selection: An ever evolving frontier in data mining", 4th International Workshop on Feature Selection in Data Mining, 4-13, Hyderabad, India, 21 June.
- [36] Motoda H., Liu H., (2002), "Feature selection, extraction and construction", Workshop on Toward the Foundation on Data Mining, 67-72, Taipei, Taiwan, April.
- [37] Liu H., Motoda H., (1998), "Feature Selection for Knowledge Discovery and Data Mining", 1st Edition, Kluwer Academic Publishers.
- [38] Soman K. P., Loganathan R., Ajay V., (2009), "Machine Learning With SVM and Other Kernel Methods", 1st Edition, PHI Learning Pvt..
- [39] Christopher J. C. B., (1998), "A tutorial on support vector machines for pattern recognition", Journal of Data Mining and Knowledge Discovery, 2 (2), 121-167.
- [40] Zheng Y., Fan L., (2010), "Moving object detection based on running average background and temporal difference", International Conference on Intelligent Systems and Knowledge Engineering, 270-272, Hangzhou, China, 15-16 November.
- [41] Park J., Tabb A., Kak A. C., (2006), "Hierarchical data structure for real-time background subtraction", IEEE International Conference on Image Processing, 1849-1852, Atlanta, GA, USA, 8-11 October.
- [42] Su S.-T., Chen Y.-Y., (2008), "Moving object segmentation using improved running gaussian average background model", Conference on Digital Image Computing: Techniques and Applications, 24-31, Canberra, ACT, Australia, 1-3 December.
- [43] Kim H., Sakamoto R., Kitahara I., Toriyama T., Kogure K., (2008), "Background subtraction using generalised Gaussian family model", Journal of Electronics Letters, 44 (3), 189-190.
- [44] Huang Z.-K., Chau K.-W., (2008), "A new image thresholding method based on Gaussian mixture model", Journal of Applied Mathematics and Computation, 205 (2), 899-907.
- [45] Zivkovic Z., (2004), "Improved adaptive Gaussian mixture model for background subtraction", 17th International Conference on Pattern Recognition, 28-31, Cambridge, Massachusetts, USA, 23-26 August.

- [46] Lee D.-S., (2005), "Effective Gaussian mixture learning for video background subtraction", IEEE Transactions on Pattern Analysis and Machine Intelligence, 27 (5), 827-832.
- [47] Harville M., Gordon G., Woodfill J., (2001), "Foreground segmentation using adaptive mixture models in color and depth", IEEE Workshop on Detection and Recognition of Events in Video, 3-11, Vancouver, BC, Canada, 8 July.
- [48] Porikli F., Tuzel O., (2003), "Human body tracking by adaptive background models and mean-shift analysis", IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, 1-9, Graz, Austria, March.
- [49] Elgammal A., Duraiswami R., Harwood D., Davis L. S., (2002), "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance", Journal of Proceedings of the IEEE, 90 (7), 1151-1163.
- [50] Comaniciu D., Ramesh V., Meer P., (2003), "Kernel-based object tracking", Journal of IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 (5), 564-577.
- [51] Piccardi M., Jan T., (2004), "Mean-shift background image modelling", International Conference on Image Processing, 3399-3402, Singapore, 24-27 October.
- [52] Belongie S., Malik J., Puzicha J., (2001), "Matching shapes", 8th IEEE International Conference on Computer Vision, 454-461, Vancouver, BC, Canada, 7-14 July.
- [53] Cortes C., Vapnik V., (1995), "Support vector networks", Journal of Machine Learning, 20 (3), 273-297.

ÖZGEÇMİŞ

Nagehan Mecidiye Çapa 1988 yılında Sinop'da doğdu. 2007 yılında başladığı İstanbul Üniversitesi (İÜ), Mühendislik ve Fen Bilimleri Enstitüsü, Elektrik Elektronik Mühendisliği Bölümü'nü 2012 yılında başarıyla tamamlayarak,2013 yılında yüksek lisans eğitimine Gebze Yüksek Teknoloji Enstitüsü (GYTE) Mühendislik ve Fen Bilimleri Enstitüsü Elektronik Mühendisliği Anabilim Dalında başladı. 2014 yılından buyana Gebze Teknik Üniversitesi'nde (GTÜ) görüntü işleme ve örüntü işleme konularında araştırma görevlisi olarak çalışmaktadır.