

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ARAÇ-YAYA KAZALARINI ÖNLEMEK İÇİN
STEREO GÖRÜNTÜ TABANLI UZAKLIK TESPİT
SİSTEMİ GELİŞTİRİLMESİ**

DOKTORA TEZİ

Emre GÜNGÖR

**Enstitü Anabilim Dalı : BİLGİSAYAR VE BİLİŞİM
MÜHENDİSLİĞİ**

Tez Danışmanı : Doç. Dr. Ahmet ÖZMEN

Ekim 2018

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

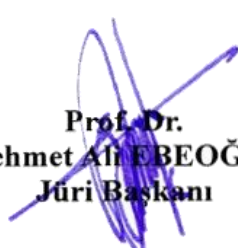
**ARAÇ-YAYA KAZALARINI ÖNLEMEK İÇİN
STEREO GÖRÜNTÜ TABANLI UZAKLIK TESPİT
SİSTEMİ GELİŞTİRİLMESİ**

DOKTORA TEZİ

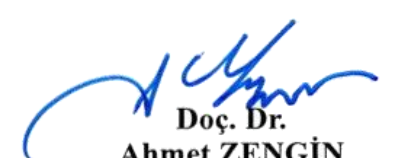
Emre GÜNGÖR

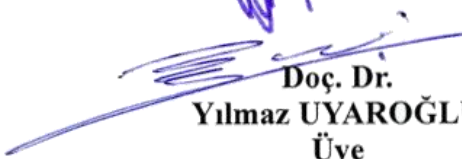
Enstitü Anabilim Dalı : **BİLGİSAYAR VE BİLİŞİM
MÜHENDİSLİĞİ**

Bu tez 19 / 10 /2018 tarihinde aşağıdaki jüri tarafından oybirliği/oyçokluğu ile kabul edilmiştir.


**Prof. Dr.
Mehmet Ali EBEOĞLU**
Jüri Başkanı


**Doç. Dr.
Ahmet ÖZMEN**
Üye


**Doç. Dr.
Ahmet ZENGİN**
Üye


**Doç. Dr.
Yılmaz UYAROĞLU**
Üye


**Doç. Dr.
İhsan PEHLİVAN**
Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Emre GÜNGÖR

19.10.2018

TEŞEKKÜR

Öncelikle beni yetiştiren, her daim sevgisiyle yanımda olan, elindeki imkanları benim okumam için seferber eden annem Gönül Güngör'e; tez süresince manevi desteğini sağlayan kardeşim Kevser Güngör'e; doktora eğitimim boyunca değerli bilgi ve deneyimlerinden yararlandığım, her konuda bilgi ve desteğini vermekten çekinmeyen, araştırmanın planlanmasından yazılmasına kadar tüm aşamalarında yardımlarını esirgemeyen, teşvik eden değerli danışman hocam Doç. Dr. Ahmet Özmen'e teşekkürlerimi sunarım. Hayatımda desteklerini esirgemeyen ve başarılarımda her daim yer tutacak olan aileme ve ilkokul öğretmenim Nebahat Yılmaz'dan başlayarak doktora danışman hocam Doç. Dr. Ahmet Özmen de dahil bütün eğitim hayatımda bana yol gösteren ve emekleri geçen bütün değerli hocalarıma müteşekkirim.

İÇİNDEKİLER

TEŞEKKÜR	i
İÇİNDEKİLER	ii
SİMGELER VE KISALTMALAR LİSTESİ	v
ŞEKİLLER LİSTESİ	vi
TABLolar LİSTESİ	xv
ÖZET	xvi
SUMMARY	xvii
BÖLÜM 1.	
GİRİŞ	1
BÖLÜM 2.	
KAYNAK ARAŞTIRMASI	6
2.1. Kümeleme	7
2.1.1. K-means kümeleme	7
2.1.2. DBSCAN kümeleme	8
2.1.3. OPTICS kümeleme	9
2.1.4. GMM-EM kümeleme	9
2.1.5. Spektral kümeleme	10
2.1.6. Parametresiz diğer kümeleme yaklaşımları	10
2.1.7. Gauss fonksiyonunu kullanan diğer kümeleme yaklaşımları	11
2.2. Segmentasyon	13
2.3. Stereo görüntü işleme ve engel tanıma	16
2.3.1. Epipolar görüntü doğrultma ve derinlik haritası çıkarımı	16
2.3.2. Stereo görüntüde yol, engel tanıma ve uzaklık ölçümü	18
2.4. Yaya tanıma sistemleri ve HOG algoritması	24

2.5. Kullanılan veri tabanları, veri yapıları ve yöntemler.....	25
---	----

BÖLÜM 3.

MATERYAL VE YÖNTEM	29
3.1. GDD kümeleme	30
3.1.1. Veri yapıları ve değişkenler.....	31
3.1.2. Algoritma gerçekleştirilmesi.....	35
3.1.3. Kümeleme sistemi deney sonuçları	41
3.2. GDD kümeleme yöntemiyle renk segmentasyonu.....	41
3.2.1. Ön işleme ve HSV renk uzayı dönüşümü.....	44
3.2.2. Çözünürlük işlemleri için blok oluşturma.....	46
3.2.3. Mean-HSV ve ADP renk çözünürlüğü düşürme yaklaşımları.....	48
3.2.4. Renk bölgelerinin elde edilmesi.....	52
3.2.5. GDD renk segmentasyonu deney, test ve bulgular.....	53
3.2.6. GDD renk segmentasyonu sonuçları.....	71
3.3. Stereo görüntüde engel tanıma ve yer düzleminin bulunması.....	71
3.3.1. V-disparity verisinin elde edilmesi.....	72
3.3.2. V-disparity verisini kullanarak yer düzleminin bulunması.....	73
3.3.3. Engel tanıma sistemi test ve analizi.....	87
3.4. Yaya tespit sistemi: HOG filtresi uygulaması	91
3.4.1. HOG özellik vektörü çıkarımı.....	91
3.4.2. HOG özellik vektörü ile SVM eğitimi.....	93
3.5. Temel sistem parçalarının ilişkilendirilmesi.....	99
3.5.1. AIR (Above immediate road) bölge tespiti.....	102
3.5.2. Alan genişletme ile yol üzerindeki nesnelerin bulunması.....	105
3.5.3. Yer düzlemi ve AIR bölgelerinin renk segmentasyonu ile ayrılması.....	107
3.5.4. Uzaklık ve renk tabanlı öncelik listesi oluşturulması.....	115
3.5.5. HOG filtresi ön-işlemleri.....	122
3.5.6. Uzaklık tespiti.....	127

BÖLÜM 4.

ARAŞTIRMA BULGULARI.....	132
4.1. Giriş verileri ve derinlik haritası çıkarımı	133
4.2. Engel bölgelerinin tespiti, test ve sonuçları.....	137
4.3. AIR bölgesi ve yol üzerinde bulunan nesnelerin tespit ve testleri.....	149
4.4. HOG filtresi test ve sonuçları	153
4.5. Tüm sistemin başarısı ve elde edilen veriler.....	160
4.6. Gelecekte yapılması planlanan çalışmalar.....	169

BÖLÜM 5.

TARTIŞMA VE SONUÇ	173
KAYNAKLAR	175
EKLER	187
ÖZGEÇMİŞ	225

SİMGELER VE KISALTMALAR LİSTESİ

ADP	: Area Dominance Peak
AG	: Alan Genişletme - Region Growing
AIR	: Above Immediate Road
BBA	: Bağlı Bileşen Analizi - Connected Component Analysis
ÇS	: Çerçeve Sınır – Bounding Box
FDT	: Fixed Distance Threshold
FGDT	: Fixed Gaussian Density Threshold
GDD	: Gaussian Density Distance
GDT	: Gradient Distance Threshold
GGDT	: Gradient Gaussian Density Threshold
GM	: Gaussian Matrix – Gauss Matrisi
GMM	: Gaussian Mixture Model – Gauss Karışım Modeli
GMM-EM	: Gaussian Mixture Model – Expectation Maximization
HOG	: Histogram of Oriented Gradients
HSV	: Hue Saturation Value - Renk Doyum Değer
LIDAR	: Laser Imagining Detection and Ranging
MTU	: Minimum Tanımlama Uzunluğu - Minimum Descriptive Length
NSL	: Neighbor Search List
N-cut	: Normalized Cut – Normalize kesim
NMI	: Normalized Mutual Information
ROI	: Region Of Interest
SPL	: Samples in Proximity List
SVM	: Support Vector Machine – Destek Vektör Makinesi
WLSS	: Weighted Least Square Sum – Ağırlıklı En Küçük Kareler Toplamı

ŞEKİLLER LİSTESİ

Şekil 2.1. JAIN veri seti için farklı kümeleme algoritmaları çıktıları (K-means, DBSCAN, OPTICS, GMM-EM ve Spektral). Her şekilde önerilen parametreler kullanılmış ve bulunan kümeler farklı renklerle gösterilmiştir. (a) Giriş verisi, (b) K-means kümeleme $k=2$, (c) DBSCAN kümeleme $\text{eps}= 1.8565$ ve $\text{minPts}= 4$, (d) OPTICS kümeleme DBSCAN parametreleriyle aynı, (e) GMM-EM kümeleme $k=2$, (f) Spektral yöntem ile kümeleme $k=2$, $\text{lev}=2$	13
Şekil 2.2. 64x64 piksel kare dama tahtası örneği 640x640 çözünürlüğünde	17
Şekil 2.3. KITTI veri tabanı kalibrasyon test görüntüsü [55].....	17
Şekil 2.4. HSV bazlı tek görüntü üzerinden yol tespiti [73].....	19
Şekil 2.5. Stereo görüntüsü derinlik haritasından elde edilen U-disparity ve V-disparity görüntüleri [78].....	20
Şekil 2.6. Ağırlıklı V-disparity verilerinin kullanılarak engellerin tespiti [69]	21
Şekil 2.7. Sayısal olarak derinlik haritası ve V-disparity karşılığı [81].....	21
Şekil 2.8. U-disparity ile bulunan engel bölgelerinin komşuluklarının incelenmesi [73].....	22
Şekil 2.9. Stereo görüntüde dikey değişime karşı uzaklık algısı [90].....	
Şekil 2.10. Uzaklığa karşı derinlik haritası değerleri [91].....	
Şekil 2.11. Test görüntüsü (sol) ve HOG özellik vektörü (sağ) [66].....	25
Şekil 3.1. Sistem genel blokları ve bölüm alt başlıkları.....	29
Şekil 3.2. İki farklı rastgele noktaya göre eliptik dağılımı ve Ayrık Toplanır Gauss verisi üzerindeki etkisi (Mavi çizgi j noktasının i noktası üzerindeki Gauss verisi etkisini gösterir. Üstteki mor çizgi ise ayrık toplanır Gauss değerlerinin doğrularla kesişimlerini belirtmektedir).....	33

Şekil 3.3. GDD kümeleme akış diyagramı.....	37
Şekil 3.4. GDD renk segmentasyonu akış diyagramı.....	44
Şekil 3.5. GDD kümeleme ile kümelenmiş renk örnekleri; HSV renk uzayı sınır belirleme görüntüsü (sol) ve pozitif kartezyen koordinatlarında temsil edilen HSV renk uzayı karşılığı (sağ).....	45
Şekil 3.6. Görüntüde piksel ve bloklar için 2-boyuttan 1-boyuta dönüşüm işlemi...	48
Şekil 3.7. ADP çözünürlük düşürme yöntemi.....	51
Şekil 3.8. GDD renk segmentasyonu örneği (513x383 piksel): (a) Orjinal görüntü, (b) iş parçası 1: ADP + GDD kümeleme, (c) İş parçası 2: mean-HSV + GDD kümeleme, (d) İş parçacıklarından gelen görüntü kesişimleri ile elde edilen sonuç görüntüsü.....	52
Şekil 3.9. Test görüntüleri üzerinden GDD renk segmentasyonu iş parçacıkları sonuçları ve elde edilen sonuç görüntüsü: (a, d) İş parçası 1: ADP + GDD kümeleme, (b, e) İş parçası 2: mean-HSV + GDD kümeleme, (c, f) İş parçacıklarından gelen görüntü kesişimleri ile elde edilen sonuç görüntüleri.....	53
Şekil 3.10. Taşlar görüntüsü (204x153 piksel) segmentasyon karşılaştırma sonuçları: (a) Orjinal görüntü, (b) K-means renk segmentasyonu, (c) K-means uzamsal segmentasyon, (d) Mean-Shift renk segmentasyonu (e) Mean-Shift uzamsal segmentasyon, (f) Normalized-Cut segmentasyonu, (g) ADP+GDD segmentasyonu, (h) MeanHSV+GDD segmentasyonu, (i) GDD segmentasyon sonucu.....	57
Şekil 3.11. Farklı tipteki bulut görüntülerinde göz algısı farklılıkları: (a) İki temel büyük ve bir küçük bulut parçası, (b) Homojen dağılımlı bulutlar (Altokümülüs bulut grubu) [111] [112].....	58
Şekil 3.12. Vazo görüntüsü segmentasyon sonuçları (321x481 piksel): (a) Orijinal görüntü, (b) Gerçeklik tablosu, (c) K-means renk segmentasyonu (d) K-means renk ve uzamsal segmentasyon, (e) Meanshift segmentasyonu, (f) Meanshift renk ve uzamsal segmentasyon Normalized-cut segmentasyonu, (h) ADP+GDD segmentasyonu, (i) GDD sonuç renk segmentasyonu.....	61
Şekil 3.13. Eski faithful gayzer püskürme veri seti: (a) Ölçeklenmemiş ifadesi (b)	

Matlab ortamında eşit eksenler arası uzaklığın eşit olarak ifadesi, (c-d) GDD kümeleme sonuçları.....	62
Şekil 3.14. Mantar görüntüsü segmentasyon karşılaştırma sonuçları (321x481 piksel) (321x481 piksel): (a) Orijinal görüntü, (b) Gerçeklik tablosu, (c) K-means renk segmentasyonu, (d) K-means renk ve uzamsal segmentasyon, (e) Meanshift renk segmentasyonu, (f) Meanshift renk ve uzamsal segmentasyon, (g) Normalized-cut segmentasyonu, (h) ADP+GDD segmentasyonu, (i) GDD sonuç renk segmentasyonu.....	64
Şekil 3.15. Farklı GDD renk segmentasyonu sonuçları: (a) Palmiye ağacı (46076), (b) Kabile (101087), (c) Duvar (374067), (d) Tören (145086), (e) Fil (296059), (f) Piramit (299091), (g) Uçak (3096), (h) Kaplan-1 (187039), (i) Kaplan-2 (160068).....	66
Şekil 3.16. 46076 görüntüsü palmye ağacı piksellerinde renk değişimi ve karışımı	67
Şekil 3.17. Düşük çözünürlüklü test görüntüsü segmentasyon sonuçları: (a) Düşük çözünürlüklü görüntü, (b) K-means renk (17 segment), (c) K-means uzam + renk (17 segment), (d) Meanshift renk (1 segment), (e) Meanshift uzam+renk (17 segment), (f) Normalized-Cut (11 segment), (g) ADP+GDD (32 Segment), (h) MeanHSV+GDD (10 segment), (i) GDD renk segmentasyon sonucu (57 segment).....	68
Şekil 3.18. Orijinal çözünürlüklü test görüntüsü segmentasyon sonuçları: (a) Orijinal görüntü, (b) K-means renk (12 segment), (c) K-means uzam + renk (12 segment), (d) Meanshift renk (1 segment), (e) Meanshift uzam+renk (12 segment), (f) Normalized-Cut (0 segment), (g) ADP+GDD (32 Segment), (h) MeanHSV+GDD (10 segment), (i) GDD renk segmentasyon sonucu (57 segment).....	69
Şekil 3.19. Sıkça kullanılan renk segmentasyonu görüntülerinin orijinali (sol) ve GDD renk segmentasyon sonucu çıktısı (sağ).....	70
Şekil 3.20. '355'nolu stereo test görüntüsünden V-disparity grafiğini edilmesi aşamaları: (a) Sol test görüntüsü, (b) Sağ test görüntü Derinlik haritası, (d) V-disparity grafiği (eşik değeri uygulanmış).....	73
Şekil 3.21. Engel tanıma sisteminin blok diyagramı.....	74
Şekil 3.22. V-disparity verilerinde en uzakta kalan gökyüzü ve hataları temsil	

eden derinlik verilerin filtrelenmesi: (a) Orijinal V-disparity verisi, (b) filtrelenme sonucu elde edilen V-disparity verisi (bkzn. en soldaki düşey çizginin filtrelenmesi).....	75
Şekil 3.23. Yer düzlemi tespiti için yapılan işlemler: (a) Tolerans bölge tespiti, (b) Minimum değerler üzerinde WLSS regresyon doğrusu, (c) V-disparity verileri üzerinde tolerans bölgesi ve regresyon doğrusunun gösterimi...	76
Şekil 3.24. '355'nolu test görüntüsü üzerinde; (a) Ham V-disparity verileri, (b) Eşik değeri üzerindeki V-disparity verileri.....	78
Şekil 3.25. '355'nolu test görüntüsü üzerinde: (a) Eşik değeri üzerindeki V-disparity verileri, (b) Her derinliğe karşılık gelen minimum değerler grafiği.....	78
Şekil 3.26. '355'nolu test görüntüsü üzerindeki V-disparity verisinden elde edilen minimum noktalar ile oluşturulmuş regresyon doğrusu.....	79
Şekil 3.27. '355'nolu test görüntüsü üzerindeki V-disparity minimum değerlerin türevi ile ağırlıklı aykırı (outlier) değer filtresi ile elde edilen tolerans bölgesi (kırmızı çizgiler arasında kalan alan) sonucu.....	81
Şekil 3.28. Minimum değerler ile elde edilen regresyon doğrusu ve tolerans bölgesi.....	83
Şekil 3.29. Engel tanımada kamera etkisi noktası 'Pc' noktası tahmini ile düzlem doğru tahminlerinin etkileri: (a) Sadece yer doğrusu düzlem tahmini, (b) 'Pc' noktasına göre yapılan doğru değişiminin etkisi.....	85
Şekil 3.30. Regresyon doğrusu ve tolerans bölgeleri ile hesaplanan ufuk noktası (Ph) ve kamera noktası (Pc) tahmini üzerinden yer düzlemi (turuncu doğru) tahmini.....	86
Şekil 3.31. Yer düzlemini belirleyen doğruyu kullanarak derinlik haritasından engel haritası çıkarım işlemi: (a) V-disparity üzerinde yer düzlem doğrusunun belirlenmesi, (b) Derinlik haritası üzerinde yer düzlem doğrusunun altında ve üstünde kalan bölgelerin tespiti, (c) Yer düzlem doğrusu üzerinde kalan derinlik bölgelerinin engel olarak belirlenip engel haritasının oluşturulması.....	87
Şekil 3.32. 2011_09_26_drive_0070 seti 355'nolu görüntüde; kamera etkisi hesaba katılmadığında elde edilen engeller (üst) ve önerilen yer	

düzlemi tahmini ile elde edilen engel (alt) görüntüleri.....	88
Şekil 3.33. 2011_09_26_drive_0070 seti 32'nolu görüntüde; kamera etkisi hesaba katılmadığında elde edilen engeller (üst) ve önerilen yer düzlemi tahmini ile elde edilen engel (alt) görüntüleri.....	88
Şekil 3.34. KITTI-2011_09_26_drive_0070 görüntü setindeki sırasıyla 3-72-156-234-414 nolu görüntülerde engel tanıma algoritması test sonuçları.....	90
Şekil 3.35. Zorlu negatif görüntü örnekleri.....	96
Şekil 3.36. Zorlu pozitif görüntü örnekleri.....	96
Şekil 3.37. Örnek test görüntüsü üzerinden HOG filtresi pozitif yaya görüntü sonuçları.....	99
Şekil 3.38. Ayrıntılı sistem blok diyagramı.....	100
Şekil 3.39. Orijinal test görüntüsü (üst) ile Yer düzlemi ile engelleri ayıran doğru kullanılarak elde edilmiş bölge sınır çizgileri (alt).....	102
Şekil 3.40. Test görüntüsünden elde edilen U-disparity çıktısı (üst) ve buradan eşik değerlerine göre U-disparity'de AIR sınır çizgisi (beyaz) ve komşuluklarının gösterimi (mavi) (alt).....	104
Şekil 3.41. AIR bölgesi ve uzaklığa bağlı renksel etiketleri.....	105
Şekil 3.42. Sütun tabanlı derinlik bilgisine göre yapılan maskeleme sonucu (renkler derinlik verisine göre değişim göstermektedir).....	105
Şekil 3.43. Alan Genişletme (AG) yöntemiyle elde edilen segmentler.....	106
Şekil 3.44. Filtrelenmiş AIR bölgesi ve uzaklığa bağlı renksel etiketleri.....	106
Şekil 3.45. AIR bölgesi segmentlerinin bağlı bileşen analizi çerçeve sonuçları.....	107
Şekil 3.46. Kırmızı-Camgöbeği (Yeşil + Mavi) kanallarında stereo giriş görüntüsü (üst), renkli sol giriş görüntüsü üzerinde GDD renk segmentasyon sonucu gösterimi (alt).....	108
Şekil 3.47. GDD renk segmentasyon ara sonuçları; ADP+GDD (üst), MeanHSV+GDD (orta) ve sonuç segmentasyonu (alt).....	109
Şekil 3.48. GDD renk segmentasyonu iş parçacıkları süreleri ve sonuç (paralel iletişim kayıpları yok sayılmıştır).....	
Şekil 3.49. Engel maskesi tersi (üst) ve AIR kullanılarak elde edilen yer düzlemi maskesi (alt).....	111
Şekil 3.50. GDD renk segmentasyonunun yer düzlemindeki görüntüsü (üst),	

Derinlik haritası (orta), Yer düzlemi renk bölgelerinin derinlik bilgisi eklendiğindeki görünümü (alt).....	112
Şekil 3.51. AIR bölgesi segment maskesi (üst), AIR bölgesindeki GDD renk segment verileri (orta), AIR bölgesinde GDD renk segmentlerinin uzaklık bilgisine göre değişimi (alt).....	113
Şekil 3.52. AIR ve yer düzlemi bölgelerinin birleşimi (üst), tersi alınarak elde edilen bölge (alt).....	114
Şekil 3.53. Yer düzlemi ile bağlantısı olmayan bölgelerin renk etiketleri.....	115
Şekil 3.54. AIR bölgesi kullanılarak elde edilen öncelik listesi.....	116
Şekil 3.55. Bağlı bileşen analizi (BBA) işlem süresi fark tespiti.....	117
Şekil 3.56. Uzaklığa bağlı (derinlik verileri) nesne listeleme.....	117
Şekil 3.57. AIR-BBA etiketleri içerisinde GDD renk segment etiketi, renk segmenti etiket piksel sayısı ve bütün görüntüye göre belirli segment içerisinde ortaya çıkma olasılığını barındıran matris değerleri.....	119
Şekil 3.58. Öncelikli işlem listesine renk bilgilerinin eklenmesi.....	120
Şekil 3.59. İşlem önceliği listesi sonuç çıktı örneği.....	121
Şekil 3.60. Öncelik sırasına göre tanımlanmış ve etiketlenmiş engel nesnelerini barındıran görüntü.....	122
Şekil 3.61. Segmentlerin olası insan bölgelerinin antropometrik ölçütlere göre belirlenmesi. Orijinal segment listesi (üst), işlenmiş segment listesi (alt).....	125
Şekil 3.62. Aynı segmentlerin farklı çerçeve dolgu miktarları kullanılarak HOG filtresine gönderildiğinde elde edilen sonuçlar.....	126
Şekil 3.63. INRIA yaya veri seti örneği, çevresel dolgu çözünürlük oranları.....	126
Şekil 3.64. Stereo kamera sisteminde epipolar doğru ve P noktasının izdüşümleri..	128
Şekil 3.65. Kamera üzerinde tanımlı odak uzaklığı ve diğer belirtilmiş bazı terimlerin görsel ifadesi.....	129
Şekil 3.66. KITTI-2011_09_26_drive_0070 serisi 355'nolu test görüntüsü...	
Şekil 3.67. Bulunan yaya üzerinden hesaplanan uzaklık tespiti.....	
Şekil 3.68. Uzaklık hesaplaması doğrulama testleri (375 piksel yüksekliğe orantılı eşitleme).....	131
Şekil 3.69. KITTI-2011_09_28_drive_0038 serisi 7'nolu görüntüde uzaklık	

tespiti.....	131
Şekil 4.1. Doğrultma için oluşturulan üç boyutlu dama tahtası görüntüleri.....	133
Şekil 4.2. Ön çalışma sürecinde 100cm aralıklı araç içi stereo kamera testleri.....	134
Şekil 4.3. KITTI veri tabanından örnek görüntü (2011_09_26_355 sol) [55].....	135
Şekil 4.4. Veri tabanı görüntüsü derinlik haritası.....	135
Şekil 4.5. Kenarlık kaldırma algoritmasından sonra elde edilen derinlik haritası.....	136
Şekil 4.6. Belirlenen Stereo görüntü kenarlıklarının sol orijinal görüntüden kaldırılmasıyla elde edilen görüntü.....	136
Şekil 4.7. Veri seti ‘2011_09_26_drive_0070’ ve ‘012’ nolu görüntü verisi kullanılarak oluşturulan önerilen metotla elde edilen engel çıktısı (üst), Hough dönüşümü yöntemiyle elde edilen engel verisi çıktısı (alt).....	138
Şekil 4.8. Veri seti ‘2011_09_26_drive_0070’ ve ‘355’ nolu görüntü verisi kullanılarak oluşturulan önerilen metotla elde edilen engel çıktısı (üst), Hough dönüşümü yöntemiyle elde edilen engel verisi çıktısı (alt).....	139
Şekil 4.9. Veri seti ‘2011_09_26_drive_0070’ ve ‘414’ nolu görüntü verisi kullanılarak oluşturulan önerilen metotla elde edilen engel çıktısı (üst), Hough dönüşümü yöntemiyle elde edilen engel verisi çıktısı (alt).....	139
Şekil 4.10. ‘2011_09_26_drive_0070-414’ nolu test görüntüsü Vdisparity grafiği (sol), 5 tepe değeri (kırmızı kareler) gösterir Hough dönüşümü çıktısı (sağ).....	140
Şekil 4.11. Hough dönüşümü tepe değerleri kullanılarak elde edilen yer düzlemi aday doğruları (sol) ve seçilen en iyi doğru (sağ).....	141
Şekil 4.12. ‘2011_09_26_drive_0070-003’ nolu test görüntüsü engel tespit sonuçları; tez aşamasında önerilen yöntem (üst), Hough dönüşümü elde edilen engeller (alt).....	
Şekil 4.13. ‘2011_09_26_drive_0070-003’ nolu test görüntüsü Hough dönüşümü (sol), adaylar (orta) ve seçilen en iyi sonuç doğrusu (sağ).....	142
Şekil 4.14. ‘2011_09_26_drive_0070-003’ nolu test görüntüsü V-disparity üzerinde belirtilen yer düzlemi doğruları; önerilen metot (sol), Hough	

dönüşümü (sağ).....	143
Şekil 4.15. ‘2011_09_26_drive_0070-012’ nolu test görüntüsü üzerinde uygulanan Hough dönüşümü ile elde edilen yer düzlemi doğrusu ve aşamaları.....	144
Şekil 4.16. ‘2011_09_26_drive_0070-012’ nolu test görüntüsü V-disparity üzerinde belirtilen yer düzlemi doğruları; önerilen metot (sol), Hough dönüşümü (sağ).....	145
Şekil 4.17. ‘2011_09_26_drive_0070-012’ nolu test görüntüsü engel tespit sonuçları; tez aşamasında önerilen yöntem (üst), Hough dönüşümü ile elde edilen engeller (alt).....	146
Şekil 4.18. ‘2011_09_28_drive_0038-063’ nolu test görüntüsü V-disparity üzerinde belirtilen yer düzlemi doğruları; önerilen metot (sol), Hough dönüşümü (sağ).....	147
Şekil 4.19. ‘2011_09_28_drive_0038-063’ nolu test görüntüsü engel tespit sonuçları; tez aşamasında önerilen yöntem (üst), Hough dönüşümü ile elde edilen engeller (alt).....	147
Şekil 4.20. ‘2011_09_28_drive_0038-063’ nolu test görüntüsü derinlik haritası.....	148
Şekil 4.21. Eğim sınırlaması yapıldıktan sonra engel tahmini önerilen metot (sol), Hough dönüşümü (sağ) sonuçları.....	149
Şekil 4.22. Eğim sınırlaması yapıldıktan sonra engel tahmini önerilen metot (üst), Hough dönüşümü (alt) sonuçları.....	149
Şekil 4.23. AIR bölge tespiti için yer düzlemi doğrusu üzerinden elde edilen uzaklığa bağlı renklerle gösterilen engeller ile yeri ayıran çizgi.....	150
Şekil 4.24. AG ile elde edilen segment sonucu.....	150
Şekil 4.25. Yer düzlemi doğrusunun altındaki alanların çıkarılmasıyla elde edilen AIR bölgesi.....	151
Şekil 4.26. Uzaklık bilgisine göre bulunan AG bölgesi (üst), yer düzlemi kesimiyle elde edilen AIR bölgesi (alt).....	151
Şekil 4.27. BBA ile AIR bölgesindeki engellerin birbirinden ayrılarak birbirinden bağımsız engel nesnelerinin ÇS (Çerçeve Sınırlar) yöntemiyle gösterimi.....	152
Şekil 4.28. Öncelik listesi sırasına göre AIR bölgesindeki nesnelerin (engellerin) sıralanması ve etiketlenmesi (yaya tespit filtresine gönderim sırası).....	152

Şekil 4.29. Uzaklık tabanlı renk öncelikli engeller üzerinden yapılan yaya tespit sonucu.....	153
Şekil 4.30. '355' numaralı veri tabanı görüntüsü engel haritası çıktısı.....	153
Şekil 4.31. HOG+SVM filtresi test sonuçları INRIA, MIT, Daimler veri setleriyle.	154
Şekil 4.32. HOG filtresini test etmek için kullanılan test görüntüsü.....	155
Şekil 4.33. Test görüntüsünden elde edilen pozitif HOG filtre sonuçları.....	156
Şekil 4.34. Sol sütun gerçek pozitif (GP), sağ sütun yanlış negatif (YN) yaya tanıma sonuçları.....	158
Şekil 4.35. Farklı dolgu değeri ile alınmış yaya görüntüleri (Üst) ve ölçeklenmiş sonuçları (alt).....	159
Şekil 4.36. Farklı dolgu değeri ile ele alınan bir diğer örnek (sol) ve ölçeklenmiş pozitif yaya sonucu (sağ).....	159
Şekil 4.37. Uzaklık tabanlı alansal eşikleme öncesi elde edilen pozitif HOG sonuçları ve uzaklıkları.....	161
Şekil 4.38. Sistem sonucu elde edilen pozitif yaya sonuç ve uzaklıkları.....	162
Şekil 4.39. Eşikleme sonrası alansal HOG öncelik listesi (üst), iki farklı dolgu kullanıldığında HOG eğitim sonucu (orta, alt).....	163
Şekil 4.40. '2011_09_26_drive_0070' veri seti görüntüleri sistem sonuç çıktı örnekleri.....	164
Şekil 4.41. '2011_09_26_drive_0070' ve '2011_09_28_drive_0038' üzerinden sistem sonuç çıktısı örnekleri.....	165
Şekil 4.42. '2011_09_28_drive_0038' veri seti sistem sonuç çıktı örnekleri.....	166
Şekil 4.43. '2011_09_28_drive_0045' veri seti üzerinden elde edilen sistem sonuç çıktısı örnekleri.....	167
Şekil 4.44. '2011_09_28_drive_0047' veri setleri giriş görüntüleri sistem sonuç çıktı örnekleri.....	168
Şekil 4.45. '2011_09_26_drive_0070' örnekleri ile dönüşlerdeki sistem sonuç çıktıları ve performansı.....	168
Şekil 4.46. '2011_09_26_drive_0070' örnekleri ile dönüşlerde sistem tarafından tespit edilen engeller.....	169

TABLolar LİSTESİ

Tablo 3.1. Saflık, F-ölçütü ve NMI kullanarak GDD renk segmentasyonu ile çok bilinen segmentasyon yöntemlerinin karşılaştırması.....	55
Tablo 3.2. Daimler, INRIA ve MIT yaya veri setleri kullanılarak yapılan SVM eğitim ve test sonuçları.....	95
Tablo 3.3. INRIA ve MIT yaya veri setleri kullanılarak yapılan SVM eğitim ve test sonuçları.....	95
Tablo 3.4. Daimler, INRIA ve MIT yaya veri setlerinden zorlu negatif görüntülerin çıkarılması ile elde edilen SVM sistemi eğitim ve test sonuçları.....	95
Tablo 3.5. INRIA ve MIT yaya veri setleri üzerinde öncül SVM eğitimleri ve test sonuçları.....	97
Tablo 3.6. Öncül filtreleme tasarımından sonra elde edilen sonuç SVM filtresi eğitim ve test değerleri.....	97
Tablo 3.7. Yaya veri setlerine, ek eğitim görüntülerinin eklenmesinden sonra oluşturulan öncü filtre sonucu.....	97
Tablo 3.8. Yaya veri setleri ve gerçek hayat görüntülerinin eklendiği SVM eğitim ve test sonuçları.....	98
Tablo 3.9. Derinlik tabanlı bölge hata payı eşik değerleri.....	103
Tablo 3.10. Antropometrik cm hassasiyetli ölçümler [115].....	123
Tablo 3.11. Antropometrik olarak ölçüm oranları.....	124

ÖZET

Anahtar Kelimeler: Stereo görüntü tabanlı araç-yaya kaza önleme sistemi, yaya tespit ve uzaklık tespiti, engel tanımlama, stereo görüntü tabanlı araç çevresel bölge tanımlama.

Otonom araçlar ile sürüş destek sistemleri, trafik güvenliği ve araç sürüş kolaylığı açısından önemli avantajlar sunmaktadır. Ayrıca yeni sensör ve donanımsal teknolojilerin gelişimi, daha hızlı ve yüksek başarımlı yazılım metotları ile bu akıllı araç sistemleri her geçen gün daha etkin ve daha güvenli hale gelmektedir. Tez kapsamında, görüntü işleme yöntemleri kullanılarak yayaların, diğer engellerin tespiti, araca olan uzaklıklarının hesaplanması ve sürüş-yer düzleminin bulunmasında kullanılabilecek stereo kamera tabanlı bir sistem geliştirilmiştir.

Kamera sistemlerinin yaygınlığı göz önüne alındığında, bu sistemlerin araçlara entegrasyonu diğer yaklaşımlara göre (LIDAR gibi) ekonomik bir çözüm ortaya koymaktadır. Tez çalışması sırasında, görüntü üzerindeki yol, nesne ve yayaların renk özelliklerine göre birbirinden ayırımı için yeni kümeleme ve renk segmentasyonu algoritmaları geliştirilmiş ve sisteme uygulanmıştır. Geliştirilen kümeleme algoritması, verilerin gözetimsiz ve parametre olmadan uzaklık ve yoğunluk özelliklerine göre ayırımı sağlamış ve bu metot görüntü üzerinde renk bölgelerinin bulunmasında kullanılmıştır. Bunun yanında, sisteme HOG filtresi ilave edilerek araç çevresindeki yayalar belirlenmiştir. Ayrıca, yeni bir engel tespit algoritması geliştirilmiş ve diğer yöntemlerle birlikte yol üzerindeki engellerin renk, uzaklık ve komşuluk gibi özelliklerine göre birbirinden ayırımı ve aracın gittiği ortamdaki bölgelerin tespiti sağlanmıştır. Olası yaya bölgeleri incelenirken antropometrik oranlar ve uzaklığa bağlı alansal büyüklükler de göz önüne alınmıştır. Bu tez çalışmasında üretilen sonuçlar ile sürüş destek sistemleri ve otonom araçlar içerisinde, stereo kamera tabanlı ekonomik sistemlerin kullanımının yaygınlaşacağı düşünülmektedir. Yaygınlaşan sürüş destek sistemleri ve otonom araçlar, araç-yaya kazalarını azaltacağından maddi ve manevi kayıplar da azalmış olacaktır.

DEVELOPMENT OF DISTANCE ESTIMATION SYSTEM BASED ON STEREO VISION TO PREVENT VEHICLE-PEDESTRIAN ACCIDENTS

SUMMARY

Keywords: Stereo image-based vehicle-pedestrian accident prevention system, pedestrian detection and distance estimation, obstacle detection, stereo camera-based vehicle environment segmentation.

Autonomous vehicles and driving support systems offer significant advantages in terms of traffic safety and vehicle driving convenience. With the development of new sensors and hardware technologies, faster and more powerful software methods, these intelligent vehicle systems are becoming more effective and safer every day. In this study, a stereo vision-based system is developed which identify ground-plane for driving, detect obstacles, calculate the distances of pedestrians and other objects in the driving region.

Given the prevalence of camera systems, integration of these systems into vehicles offers an economical solution to other approaches (such as LIDAR). During the thesis study, new clustering and color segmentation algorithms have been developed and applied to the system to distinguish road, objects and pedestrians using color features of the image. The developed clustering algorithm distinguishes the data according to the distance and density properties without parameters and supervision, and this method is used to find color spaces on the image. In addition, the HOG filter is added to the system to determine the pedestrians around the vehicle. Besides, a new obstacle detection algorithm has been developed, and it has been possible to distinguish the obstacles on the road according to the characteristics such as color, depth and neighborhood with identifying the regions in the environment where the vehicle is going. Anthropometric proportions and spatial ranges depending on the distance are also taken into consideration when examining possible pedestrian zones. It is considered that the use of stereo camera based economical systems in driving support systems and autonomous vehicles will be widespread with the results produced in this thesis study. Increased driving support systems and autonomous vehicles will reduce vehicle-pedestrian accidents and will also reduce financial and moral losses.

BÖLÜM 1. GİRİŞ

Trafik kazalarını azaltmak ve sürüş konforunu arttırmak için tasarlanmış otonom araçlar ve araç sürüş destek sistemleri zaman geçtikçe yaygınlaşmaktadır. Yakın gelecekte mevcut araçların yerini alacak otonom araçlar üzerinde yapılan çalışmalar yoğun bir şekilde ilerlemekte, bu sayede gerçekleştirilen bazı prototip araba veya tren setlerinin gerçek hayatta testlerini görebilmekteyiz. Sürüş destek sistemlerinin veya otonom araçların kontrol biriminin en temel görevi, araç civarındaki engellerin belirlenmesi, uzaklıklarının tespit edilmesi ve yol düzlemin ortaya çıkarılmasıdır. Bu amaçla engelleri algılayan çeşitli sensör sistemleri ve işaret/görüntü işleme algoritmaları üzerinde yoğun çalışmalar yapılmaktadır. Hareket halindeki araçların etrafındaki engellerin belirlenmesi için LİDAR, kamera, RADAR, ultrason gibi birçok sensör sistemlerinden faydalanılmaktadır. Bu çalışmada, ekonomik avantaj dikkate alınarak stereo kamera sisteminden elde edilen görüntülerden çıkarım yapılması öngörülmüş ve başarılı sonuçlar elde edilmiştir. Elektrik enerjisi ile çalışan araçlarla beraber otonom araçların yakın gelecekte yaygınlığının artacağı öngörüldüğünden, engele tanıma, yol düzlemi belirleme çalışmalarına son yıllarda yoğun ilgi gösterilmektedir.

Otonom araçlarda karar mekanizması sistem üzerinden otomatik gerçekleştirilirken, araç sürüş destek sistemlerinde ise sistemin elde ettiği bilgi ve uyarılar doğrultusunda sürücünün karar alması beklenmektedir. Araca kısmi etki gösteren bazı sistemler ise nihayetinde sürücüyü koruma amaçlı olup bütün inisiyatifi eline alarak kararda bulunmadığından, araç sürüş destek sistemi alanı içerisinde yer almaktadır. Bütün bu sistemlerde nesne tespiti; LIDAR, radar, kamera ve benzeri sensör ve alıcılar kullanılarak yapılmaktadır. Fakat buna karşın bu alanlar konusunda, son on yıl içerisinde yoğun çalışma ve araştırmaların ortaya çıktığı görülmektedir. LIDAR sensörler, ışık temelli tanıma ve kapsam tespitini sağlayan sensörler olarak

tanımlanabilir. Radar tipi sensörlerde ise ışık yerine radyo dalgaları kullanılarak kapsam tespitiyle veriler elde edilerek kullanılmaktadır. Çalışmada öne sürülen yeni teknik ve teknolojiler bu alanlar içerisinde, görüntü işleme alanında ve araç güvenliği teknolojilerinde önemli yenilik ve gelişmeler sağlamaktadır. Sistemde kamera sensörlerinin veri girişinde seçilmesinin sebebi ise yaygınlık ile ekonomik olarak temin edilmesinin ve kurulumunun kolaylığıdır. Ayrıca kamera kullanımında; kompakt kullanım kolaylığı, renk bilgisi sayesinde çevre algılamada kolaylık sağlaması, sınıflandırma ve doku özelliklerini tanımlanması gibi konular avantaj olarak öne çıkmaktadır. Ayrıca kamera, donanımları ile yakınlaştırma ve odaklanma özellikleri kullanılabildiğinden, özelleştirilmiş şekilde verilerin alınması ve bunların algoritmalar yardımıyla işlenebilmesi olanaklı hale gelmektedir.

Araç-yaya uzaklık tespiti ve tespit sistemi temel olarak; engel, nesne ve yer düzlemi tespiti; görüntü renk bilgisi tespiti ile buna göre alanlara ayrılması; yaya tespit sistemi eğitim ve filtreleme işlemleri şeklinde üç ana işlem akışı içerisinde tanımlanabilir. Engellerin tanımlanması ve devamında nesnelerin belirlenmesi için stereo kameralar yardımıyla alınan görüntüler kullanılarak derinlik haritası oluşturulur. Sonrasında ise derinlik haritasındaki bilgiler ve değişimler analiz edilerek verilerde görülen düzlemsel özellikler ve bu özelliklere uymayan noktalar belirlenir. Bu veriler kullanılarak engel teşkil eden nesneler tanımlanmış ve yer düzlemi tespit edilmiş olur. Nesne tanımlamada yayalar ve engellerin ayrımı algısal işlemler üzerinden yürütüldüğünden ve çevresel bilgiler derinlik üzerinden olduğu kadar renk bilgilerine göre de ayrım göstermesinden dolayı, çalışma kapsamında geliştirilen yeni parametre gereksinimi olmayan renk segmentasyonu algoritması kullanılarak çevresel renk bölgelerinin ayrım işlemi gerçekleştirilmiştir. Yaya olabilecek bölgeler belirlendikten sonra renksel önceliklere ve bölgelerin yakınlıklarına göre yaya tespiti işlemi gerçekleştirilmiştir. Yaya tespiti öncesinde veri tabanları kullanılarak eğitim yapılmış ve sonrasında bu filtre sistem içerisinde kullanılmıştır. Bahsedilen temel üç işlem akışı içerisinde sistem, çevresel bölge ve nesneleri özellikleriyle birlikte kullanıcıya sunulmaktadır. İstenildiği takdirde çıktı verileri, araç sürüş destek sistemi olarak uyarı şeklinde kullanıcıya iletebilir veya karar verme sistemi entegre edilerek otonom araçlar için uygun hale getirilebilir.

Sistem kullanımı konusunda son kullanıcı için ihtiyaç odaklarına göre çözümünün üretilmesi ve çıktılarının sunulması tavsiye edilmektedir. Oluşturulan yazılımlar kapsam dahilinde her iki uygulama için de bir temel vazifesi görececek şekilde oluşturulmuştur.

Yer düzleminin bulunması, engellerin ve yayaların uzaklıklarıyla tespit edilmesi sürücü karar destek sistemleri ve otonom araçlar için en önemli temel operasyonlardır. Diğer yapılan bütün manevra ve alınacak önlemler bu temel işlemler üzerinden yürütülmektedir. Çevresel nesne ve koşulların tespiti araç-yaya kazalarının önlenmesi için kritik öneme sahip olup atılması gereken en önemli adımı oluşturmaktadır. Bu nedenle tez çalışması kapsamında bu konu üzerinde çalışılmış ve yazılım tabanlı oluşturulan sistemle veriler üzerinden sistemin uygulanabilirliği ve sonuçları test edilmiştir. Oluşturulan sistem araç-yaya kazalarının önlenmesi için tasarlanmış olsa da aynı şekilde araçların karıştığı araba, kamyon, motosiklet, bisiklet dahil bütün hareketli ve hareketsiz engeller ile yapılan kazaları önleme konularında da aynı şekilde sistemin nesne tespit kısımları özelleştirilerek uygulanabilir. Sonrasında ise aracın hız ve içinde bulunduğu koşullar içerisinde tespit edilen engel ve değişimlerine uygun önlemleri alması sağlanmalıdır. Yapılan çalışma ile hayati ve mali kayıpların önlenmesi için çevresel etmenlerin tespitini yapan sistem başarılı bir şekilde test edilmiştir.

Kaza önleme sistemleri, günümüzde sürücüsüz veya sürücü destekli olmak üzere yoğun bir araştırma konusu olmakla birlikte, trafikte yaşanan ölümlü veya sakat bırakıcı kazaların azaltılması için çalışılmaktadır. Sosyal ve ekonomik açıdan bütün dünyada çok büyük zararları olan bu kazaların azaltılması ve önlenmesi önem arz etmektedir. Dünya sağlık örgütü WHO'nun hazırladığı istatistiksel rapora göre dünya çapında 1,2 milyon insan yol kazalarında ölmekte ve her yıl 50 milyon insan ise yaralanmaktadır [1]. Amerikan ulusal trafik güvenlik birimi NHTSA'nın verilerine göre de ölüm oranlarının %11 ila %14'lük kısmı araç-yaya kazalarını kapsamaktadır. Bu oran ise dünya genelinde 168 binlere kadar çıkmaktadır. Bu sayıya yaralanmalar da dahil edildiğinde araç-yaya kazalarının sebep olduğu etki 12 milyon insanı kapsamaktadır. Ayrıca Amerika'da sadece 14 yaş ve altı çocukların sağlığı için

harcanan masraflar yıllık 5,2 milyar doları bulmaktadır [2]. Bu noktada yaşanan iş gücü kaybı ve finansal etkilerini göz önüne alacak olursak araç kazalarının astronomik boyutlarda bir kayba neden olduğu açıkça görülmektedir. Tez kapsamında yapılan bu çalışmanın temel hedefi araç kazalarının sebep olduğu bütün bu kayıpları minimuma indirmektir.

Çalışmada öne çıkan yenilikler; engellerin daha yüksek doğrulukta tespiti, yeni kümeleme ve renk segmentasyon algoritmaları, yaya tespitinde antropometrik ölçüt ve oranların kullanımı, renk ve uzaklık tabanlı yaya tespiti için öncelikli işlem listesi oluşturulması sayılabilir. Ayrıca yaya tespitinde uzaklıkların ve etkilerinin incelenmesi araç-yaya kazalarının önlenmesi için büyük önem taşımaktadır. Yapılan bu çalışmalar sistem güvenliği ve başarımını artırmaktadır. Öne çıkan bu özelliklerin yanı sıra sistemin uyumlu ve başarılı biçimde çalışma, çevre bölgelerin, nesnelerin tanımlanması ve tespiti çalışmada öne çıkan önemli unsurlardır.

Yapılan deneyler ve sonuçları 4. bölüm içerisinde çıktılarıyla birlikte incelenmiştir. Sistem-karar veya uyarı mekanizmaları; optimizasyon ile kriter uyumluluğu ve donanımsal aktarım ile bunların uygulanması tez çalışması kapsamı dışında tutulmakla beraber bu alanlarla ilgili öneri ve görüşler uygun ölçülerde yeri geldiğinde konuları dahilinde aktarılmıştır. Bahsedilen bu konularla ilgili çalışmalar tez kapsamı dahilinde olmadığından gelecekte yapılması planlanan çalışmalar altında incelenmiştir. Tez kapsamında yapılan çalışmalar teorik ve pratik olarak yazılım modelleme kısımlarını kapsamakla birlikte, uygulamanın temelini oluşturmak amacıyla yoğun bir çalışma sonucunda test sonuçlarıyla ortaya konulmuştur.

Yazının devamında 2. Bölüm içerisinde sistem tasarlanırken yararlanılan kaynaklar, yöntemler ve ilişkili literatürde yapılmış çalışmaların örnek verildiği kaynak araştırması kısmı bulunmaktadır. Sonrasında, tez aşamasında kullanılan metotlar, literatüre yapılan katkı ve bu yöntemlerin nasıl uygulandığı 3. Bölüm materyal ve yöntem kısmında açıklanmaktadır. Araştırma bulguları genel sistem dahilinde 4. Bölümde incelenmiş ve test görüntüleri üzerindeki performanslarıyla birlikte irdelenerek gösterilmiştir. Sistemin doğruluğunu artırabilecek noktalar da yine bu

kısımda incelenmiştir. Alt sistemlerin çalışma ve bulguları, genel sistem testleri ve sonuçlarıyla karıştırılmaması için 3. Bölüm içerisinde konu başlıkları halinde incelenmiştir. En son bölüm olan 5. Bölüm içerisinde ise tartışma ve sonuçlar ortaya konulmuş, sistemin öne çıkan özellikleri, test görüntülerle elde edilen başarımlar göz önünde bulundurularak ileride atılması gereken adımlardan bahsedilmiştir. Ekler kısmında ise tez konusuyla ilgili olan fakat konu bütünlüğünü bozmamak amacıyla dahil edilmeyen bilgilere yer verilmiştir.



BÖLÜM 2. KAYNAK ARAŞTIRMASI

Tez projesi kapsamında birden fazla disiplin üzerinde çalışılmış ve farklı alanlarda yapılan çalışmalar araştırılmıştır. Çalışmada amaç, araç-yaya kazalarını minimuma indirmek olduğundan yaya tanıma, renk bölgelerinin ayrılması, engel tanıma ve yer düzleminin tespiti gibi alanlarda inceleme yapılmıştır. Renk bölgeleri ayrımı için, kümeleme ve renk segmentasyonu algoritmaları; nesnelerin ve engellerin ayrımı için, derinlik bilgisi ve stereo işleme metotları incelenerek karşılaştırılmıştır. Ayrıca bilinen insan tanıma algoritmaları içerisinde yüksek başarımlı gösteren metotlar uygulamada kullanılması için seçilmiştir.

Her biri ayrı ayrı ele alınan ve alt başlıklar halinde incelenen bu konuları aşağıdaki listedeki gibi tanımlayabiliriz.

1. Kümeleme
2. Segmentasyon
3. Stereo görüntü işleme ve engel tanıma
4. Yaya tanıma yöntemleri
5. Kullanılan veri tabanları, veri yapıları ve alt yöntemler

Kaynak araştırması bölümünde, temel olarak her çalışma için ayrı kaynak araştırması yapılarak ilgili metotlar incelenmiş, uygun görülenler gerek uygulama olarak gerekse karşılaştırma amaçlı kullanılmıştır. İhtiyaca cevap vermeyen metotlar yerine amaca yönelik yöntemler geliştirilmiştir. İyi sonuç veren metotlar ise yazarlar tarafından belirtildiği şekilde oluşturularak sistem içerisinde kullanılmıştır.

2.1. Kümeleme

Kümelemeyi temel olarak tanımlayacak olursak, herhangi bir grupta veya sınıflandırma bilgisi bulunmayan, tanımlanmayan verilerin gözetimsiz bir biçimde gruplara ayırma işlemidir. Hiyerarşik ve bölüm işlemi olarak kümeleme algoritmaları iki ana temel kategori altında incelenebilir. Tez kapsamında ortaya konulan GDD kümeleme algoritması, verilerin yoğunluk ve uzaklık gibi kriterlere göre birbirleriyle olan ilişkilerini tespit etmekte ve bu ilişkiler uyarınca verileri birbirinden ayırarak kümeleri oluşturmaktadır.

Literatürde farklı uygulamalar için öne sürülmüş birçok kümeleme algoritması bulunmaktadır. Standart sapma, varyans, yoğunluk, dağılım ve uzaklık gibi bilgiler kümeleme yapılacak verilerin ayrımı için önemli ipuçları/özellikler barındırmaktadır. Bu sebeple, genel olarak, her kümeleme algoritması farklı verileri ya küme üyeliği kriterlerine ve/veya üretilen kümelerin sayısına göre gruplamaktadır. Birçok kümeleme yönteminde aynı veri setinde farklı koşullarda farklı sonuçlar alınmaktadır. Ayrıca farklı parametre girdileri de aynı veride farklı sonuçlar vermektedir. Bu bölüm temel ve sıklıkla bilinen kümeleme algoritmaları ve bunların etkileri hakkında bilgiler vererek, bölüm 3.1.'de bahsedilen tez çalışması kapsamında oluşturulan parametre gerektirmeyen GDD kümeleme algoritmasının önemi, benzerliklerini ve farklılıklarının anlaşılmasını sağlayacaktır.

2.1.1.K-means kümeleme

K-means algoritması, k küme sayısı giriş parametresine göre veri örnekleri arasındaki uzaklıkları kullanarak kümeleme yapan, en yaygın bilinen ve kullanılan yöntemlerden biridir [3]. Rastgele olarak seçilen k adet veri örneğini küme merkezi olarak kabul ederek veriler bölümlenmeye başlanır. Sonrasında tekrarlayan şekilde kümeler, örnekler arası uzaklıklar ölçülerek oluşturulur. K-means algoritması farklı kümeleme algoritmalarına kıyasla çok fazla parametre gerektirmemekle birlikte, gerçekleşmesi hızlı bir şekilde yapılabilmektedir.

K-means'ın kısıtlamaları aşağıdaki gibi listelenebilir:

1. Başlangıç parametrelerine olan sıkı bağımlılığı. Sonuçlar başta seçilen parametrelere göre büyük ölçüde farklılık gösterebilmektedir.
2. Küresel şekiller dışındaki şekilleri algılamada başarılı olamaması. Küresel şekilde küme merkezine en yakın olacak şekilde kümeler yakınsar.
3. Kümelerin benzer boyutlarda olması beklenir. Bu sebeple en yakın küme merkezi doğru olarak kabul edilir.

Birçok farklı K-means türevi algoritma yukarıda sıralanan kısıtlamaları ve karmaşıklık sorununu ortadan kaldırmaya çalışmıştır. K-median bunlardan biri olmakla birlikte, ortalama yerine medyan değerlerini kullanmaktadır [4, 5]. Benzer şekilde, K-medoids algoritmasında ortalama yerine uzaklıklar toplamını minimize etmeye çalışmaktadır [6]. Bir başka algoritma ise Fuzzy C-means ismiyle, her veri örneğinin diğer küme merkezlerine aitlik derecesine göre küme oluşturmaktadır [7]. Daha iyi başlangıç küme merkezlerini seçme amacıyla K-means++ literatürde öne sürülmüştür [8]. K-means'in uygulama anındaki hızını artırmak için ise KD-ağaçları [9], çekirdek setleri [10] ve üç köşeli eşitsizlik [11] çalışmaları önerilmiştir. Lokal minimumdan kaçma ise bir diğer önemli olarak karşımıza çıkmaktadır [12].

2.1.2. DBSCAN kümeleme

DBSCAN noktalar arası uzaklığa göre yoğunluğu belirli çap içerisinde olan noktaları olarak komşuluklarla birlikte kümeleme yapan bir algoritmadır [13]. Eğer veri örneği ϵ (eps) yarıçapındaki alan içerisinde ise ve bu yarıçap içerisinde yer alan bölge belirlenmiş minimum sayıda (minPts) örnek içeriyorsa bölge küme olarak tanımlanır ve bu şekilde bölgeler tanımlanır. DBSCAN algoritması bu iki giriş parametresi gereksinimine ihtiyaç duymaktadır. DBSCAN için parametre tahmin çalışmaları bulunmaktadır. Bu parametreler performansı etkilemekle birlikte, çok küçük seçildiğinde birçok bölge tanımsız kalmaktadır. Diğer yandan çok büyük seçildiğinde kümeler birleşerek istenilen sonuca ulaşılamayabilir. DBSCAN algoritması kümeleri

geometrik olarak kümelemekle birlikte karmaşıklığı eğer veri yapıları kullanılarak optimize edilerek tekrardan yapılandırılırsa $O(N.logN)$ 'e düşebilmektedir [14].

2.1.3. OPTICS kümeleme

OPTICS, bir DBSCAN türevi olarak, kümeleri DBSCAN' dekinden farklı olarak sabit bir yarıçap değeri yerine, yarıçap erişim uzunluğu parametresi yardımıyla farklı yoğunlukları yakalamada ve küme olarak ifade etmede daha güçlüdür [15]. Bu sebeple OPTICS, farklı yoğunlukları olan iç içe geçmiş kümeleri bulmada daha başarılı bir sonuç vermektedir. Ayrıca, dinamik metot olan DM-DBSCAN, yarıçap uzunluğunu dinamik olarak global değerlerle tespit ederek bu alanda ilerleme sağlamıştır [16]. Özet olarak OPTICS, DBSCAN ile aynı temellerde işlem yapmasına rağmen farklı yoğunluktaki kümeleri tespit etmede kullanılabilir.

2.1.4. GMM-EM kümeleme

GMM-EM algoritması veri setindeki her bir örneği Gauss fonksiyonu olarak tanımlayarak kümelerin bulunmasını sağlar [17]. Sorumluluk faktörü, hangi Gauss değerinin kümeleme sistemine dahil edileceğini ve hangi noktalarla güçlü ilişkisi olduğunu tanımlar. GMM-EM algoritmasında, kümelerin bulunması için beklenti maksimizasyonu uygulanmaktadır [18]. Kümeleme için en uygun GMM değerlerini bulmak belirleyici olmayan polinom (NP) problemi olmakla birlikte iyileştirme işlemi karmaşık bir işlem olarak karşımıza çıkmaktadır. Beklenti maksimizasyonu (EM), veriler Gauss dağılımı şeklinde dağılım gösterdiğinde, GMM çözümü için kullanışlıdır. Fakat farklı şekil ve örnek gruplarını kümelemek bu metotla çözüm arandığı zaman bulunması zorlaşmaktadır. Kısacası, GMM-EM metodu giriş parametresinde verilen sayıda Gauss verisini uzayda yoğunluklara en uygun bir biçimde kapsaması sağlayarak kümeleme işlemini tamamlamaya çalışmaktadır. Gauss değerleri beklenti maksimizasyonunda bulunan yapıya uygun olarak istenilen değerlere ulaşırsa, işlem sonlandırılır. Fakat bu metot zaman alıcı ve bazı koşullarda şekiller Gauss şekline oturmada hata verebilir.

2.1.5. Spektral kümeleme

Spektral kümeleme, benzerlik matrisindeki öz değerleri kullanarak verileri ayırır. Özellikle görüntü analizinde kullanımı uygundur. Farklı türevleri bulunmakla birlikte gerçekleşmesi kolay ve etkili biçimde lineer cebir metotlarıyla çözümlenebilir. Örneğin graf-kesimi yöntemi, graf veri yapısındaki verileri yol yöntemi ile bölgelere ayırmaktadır [19, 20]. Kesim yolunun bulunması ve kesim maliyetlerinin en aza indirilmesi, graf-kesim yönteminin birincil odak noktasıdır. Bir diğer türev spektral kümeleme yöntemi ise K-means yaklaşımını kullanarak kümeleri bulmaktadır. Giriş parametreleri veya benzerlik graf yapısı spektral kümelemede sonuçları önemli derecede etkilemektedir [21].

2.1.6. Parametresiz diğer kümeleme yaklaşımları

Literatürdeki parametresiz kümeleme çalışmalarından biri, uzamsal veri tabanları ve uygulamaları ile ilişkilidir [22]. Bu çalışmada hiyerarşik graf bazlı kümeleme yöntemi oluşturularak kümeler istatistiksel medyan ve varyans hesaplamalara göre oluşturulmuştur. Kümeleme tanımı noktasında GDD kümeleme yöntemi ile benzerlik göstermesine rağmen, kümelerin oluşturulmasında temel olarak graf veri yapısının kullanımı açısından farklılıklar bulunmaktadır [23].

Bir diğer parametresiz graf tabanlı çalışma ise biyolojik olgu olan protein-protein etkileşimi (PPI) ağlarının ilişkisine dayanmaktadır [24]. Bu yaklaşımda benzerlik matrisi PPI ve minimum kesim yolu, kümeleri elde etmek için aranmaktadır. Bu çalışma graf-kesim algoritmasına benzemekte olup, benzerlik matrisi farklılık göstermektedir. Diğer farklı parametre gerektirmeyen graf tabanlı yöntem “PaCK: Scalable Parameter-Free Clustering on K-Partite Graphs” olarak adlandırılmaktadır [25]. PaCK, kümeleme yöntemini bir sıkıştırma metodu gibi ele alarak çözmeye çalışmaktadır. Amaç olarak, ikili veri bağıntı matrisinin parametresiz olarak desenlerini bulma amacı taşır. Ayrıca Minimum Tanımlama Uzunluğu (MTU) prensibi ile kriterler tasarlanıp, bu kriterler ağgözlülük yöntemiyle minimize edilmektedir. Temel olarak k-parçalı grafların kullanımı ve sıkıştırma yöntemlerinin

kümeleme alanında kullanılmasıyla GDD kümeleme tekniğinden farklı bir yol izlemektedir PaCK kümeleme algoritması.

Graf tabanlı olmayan parametresiz kümeleme yöntemlerinden birisi de MTU'yu kullanarak benzerlik ve sıkıştırma yöntemleriyle veri gruplarını ayırmaktadır [26]. Bu çalışmada Gauss yoğunluk fonksiyonu kullanılmasına rağmen, genel işlem Huffman-kodlamadakine benzer bir biçimde entropi işlemine verileri tabi tutmaktadır. GDD kümeleme metodu, Gauss bölgesi tahmini yerine, Gauss yoğunluk değerlerini her nokta için kullanılmaktadır. Bir diğer parametresiz kümeleme çalışması RIC: Parameter-Free Noise-Robust Clustering olarak adlandırılmakla birlikte veri setindeki doğal kümeleri bulmak için MTU yöntemini kullanmaktadır [27]. RIC kümelemeyi, kümenin korelasyon yapısını gürültü ortamında tahminiyle başarmaktadır. RIC yöntemi de Böhm ve arkadaşlarının yaptığı sonraki çalışma benzeri entropi ve sıkıştırma işlemlerini yapmaktadır. RIC temel olarak küme üzerindeki gürültü etkilerine odaklanmış, GDD kümeleme yöntemi ise gürültü koşullarını incelememektedir.

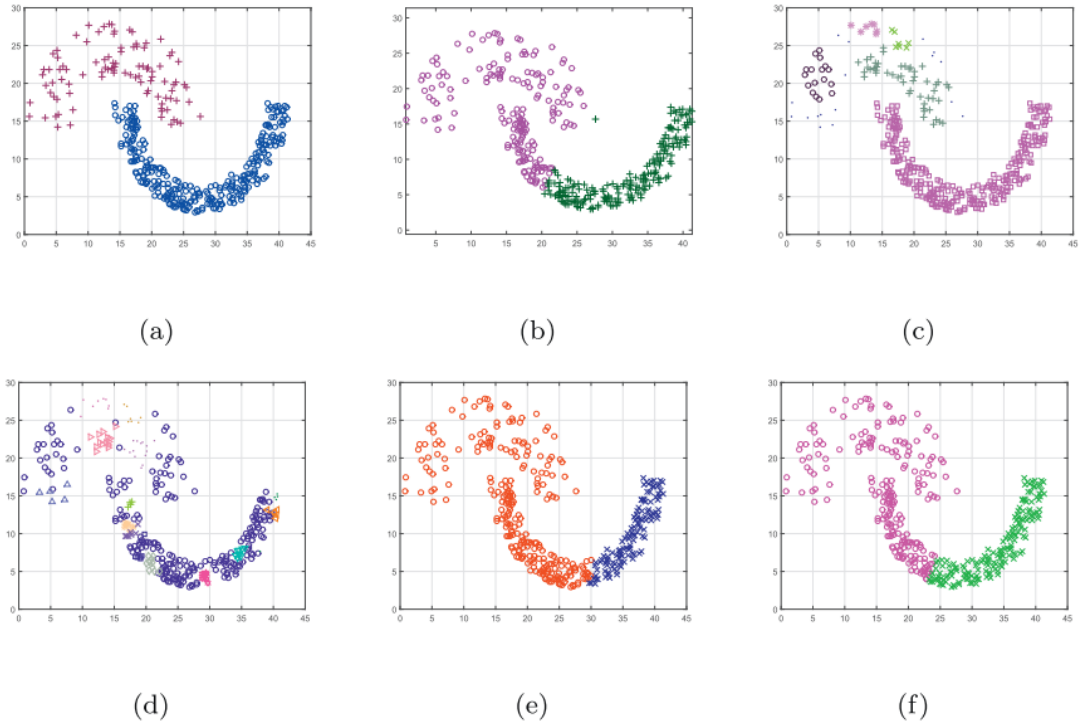
Bir başka parametresiz çalışma tek bağıntı metodudur [28]. Bu çalışmanın farkı ise GDD'nin veri yapılarıyla veri uzayı örnekleri üzerinde herhangi bir bağıntı oluşturulmadan ve veri yapısıyla tanımlanmadan GDD'nin kümeleme yapmasıdır. GDD kümeleme yönteminde komşuluklar aranırken, tek bağıntı metodunda örnekler arasındaki bağıntı kullanılmaktadır [29].

2.1.7. Gauss fonksiyonunu kullanan diğer kümeleme yaklaşımları

Verileri ayırmada Gauss fonksiyonu kullanan başka kümeleme metotları da bulunmaktadır. Bunlardan bazıları olan GMM-EM ve DENCLUE, Gauss fonksiyonlarının nasıl kümelemede kullanılabildiğinin birer örneğidir [30, 31]. DENCLUE yönteminde Gauss verileri sadece lokal minimum ve maksimum değerleri bulmak için kullanılmaktadır. Kümeleme işleminin tamamlanması için, tepe tırmanışı prosedürü sunulmuş ve uygulanmıştır. Fakat bu durum işlem karmaşıklığını algoritma için artırmaktadır.

Yoğunluk tabanlı algoritmalar, veri uzayında yoğun bölgeleri inceleyerek veriler yüksek yoğunluğa sahip olduğunda çevresine göre küme oluşturmaktadır. DBSCAN en çok bilinen yoğunluk tabanlı algoritma olarak çevre örneklerin uzunluğuna göre hesaplamalar yapmaktadır. Farklı tip şekillerde ve koşullarda iyi performans vermesiyle birlikte parametre tahmini için DBSCAN-GM gibi yarıçap uzunluğunu Gauss fonksiyonuyla tahmin edilmesi için çalışmalar yapılmıştır [32]. Fakat DBSCAN metodu kümeleri benzer oranda yoğunlukta kabul ettiğinden farklı yoğunlukları olan küme verilerinde istenmeyen kümeleme sonuçları elde edilebilmektedir.

Bir diğer çalışma ise, Baudry ve arkadaşları tarafından, sınırlı Gauss değerleri kullanılarak k parametresi ve karışımları kullanılarak kümeleme işlemi yapılmıştır [33]. GDD kümeleme yönteminde ise, bütün örnek veriler diğer örnek veriler üzerinde Gauss bileşenini barındırmaktadır. Ayrıca Baudry ve arkadaşları Gauss verileri karışımını hiyerarşik biçimde toplayarak entropi verisine göre bilgileri birleştirmektedir. GDD kümeleme metodunda ise, veri örnekleri Gauss yoğunluk verileri ve Öklid uzaklıklarının oranları ile kümeler oluşturulur.



Şekil 2.1. JAIN veri seti için farklı kümeleme algoritmaları çıktıları (K-means, DBSCAN, OPTICS, GMM-EM ve Spektral). Her şekilde önerilen parametreler kullanılmış ve bulunan kümeler farklı renklerle gösterilmiştir. (a) Giriş verisi, (b) K-means kümeleme $k=2$, (c) DBSCAN kümeleme $\text{eps}=1.8565$ ve $\text{minPts}=4$, (d) OPTICS kümeleme DBSCAN parametreleriyle aynı, (e) GMM-EM kümeleme $k=2$, (f) Spektral yöntem ile kümeleme $k=2$, $\text{lev}=2$.

Şekil 2.1.'de yukarıda bahsi geçen sık kullanılan kümeleme yöntemlerinin birbirleri arasındaki ilişkisi Matlab ve ELKI veri madenciliği sistemi aracılığıyla tespit edilerek karşılaştırılmıştır [34].

2.2. Segmentasyon

Segmentasyon ya da bölütleme olarak literatürde geçen yöntem, görüntü üzerindeki bölgelerin aydınlık, renk, kenar veya buna benzer özellikler yardımıyla ayrımı ve tanımlanması işlemidir. Bu tanımlama sonrasında bir veri tabanı ya da gözetim eşliğinde ilişkilendirme yapılıyorsa, buna görüntüde nesne tanıma işlemi denilir. Segmentasyon işlemi genellikle sadece görüntü üzerindeki bölgelerin özellikler yardımıyla ilişkilendirilmesi ve belirlenmesi işlemidir. Görüntü üzerindeki özellikler genel olarak köşe, kenar, doku, renk, aydınlık ve uzamsal uzaklıklar gibi bilgilerin okunması veya hesaplanmasıyla elde edilir.

Tez kapsamında kümeleme bazlı renk segmentasyon yöntemi geliştirilmiş. Ayrıca bazı segmentasyon algoritmalarından da alt sistemler içerisinde yararlanılmıştır. Çalışma kapsamında incelenen kümeleme bazlı segmentasyon algoritmalarından bazıları K-means, C-means ve Meanshift olarak tanımlayabiliriz. Kümeleme kaynak araştırması kapsamında incelenen yöntemlerle birlikte, farklı metotlar ve bunların testleri veri tabanlarıyla birlikte karşılaştırmaları sonraki bölümlerde incelenmiştir. Segmentasyon uygulamalarında da kümeleme metotlarında olduğu gibi uzaklık ve yoğunluk değerleri kullanılmaktadır. Fakat bu değerler, görüntü üzerindeki verilerden ve elde edilen özellikler kullanılarak ölçülüp küme aidiyetleri farklı renk, ışık, kenar gibi bölgelerin tanımlanmasında kullanılır. Parametre girdi ihtiyacı kümeleme metotlarına benzer olarak birçok segmentasyon algoritmasında da bulunmaktadır. Meanshift, GDD'ye benzer olarak parametresiz küme sayısını veri yoğunluklarına göre belirler. Meanshift çeşitli global küme sayısı veya eşik belirleme metotlarına oranla daha iyi başarımlar gösterebilmektedir. GDD'de ise hem global hem de lokal kümeleme farklılıkları her elemana göre ayrı ayrı ele alınıp lokal düzeydeki değişimler global ve lokal elemanlara bağlı olarak aitliklerini uzaklık ve yoğunluk ölçümlerine göre hesaplanmaktadır. Bu sebeple hem uygulama olarak hem de metodoloji olarak GDD kümeleme yöntemi, görüntü segmentasyon uygulamalarını geliştirmek amacıyla kullanılmıştır.

Meanshift yöntemi, yoğunluklar üzerinden uzay üzerindeki maksimumların tahmini noktalarını bularak kümeleme işlemini parametresiz olarak gerçekleştirmekte ve bu sayede kümeleme analizi ve görüntü işleme alanlarında sıkça kullanılmaktadır [35, 36, 37]. Segmentasyon uygulamalarında ve literatürde sıkça geçen Meanshift yöntemi iyi renk segmentasyonu sonuçları vermesinin yanında hareket takibi gibi uygulamalarda da kullanılmaktadır [38, 39, 40, 41]. En büyük avantajı olarak yoğun bölgelerdeki yerel maksimumları otomatik olarak tahmin ettiğinden küme sayısına ihtiyaç duymamasını gösterebiliriz.

K-means metodu kümeleme işlemlerinde kullanıldığı gibi renk segmentasyonunda da gerek kıyaslama için gerekse kolay gerçekleştirilmesinden dolayı sıkça

kullanılmaktadır [42, 43, 44]. K-means 'teki kesin aidiyet değerleri yerine, ağırlıklar kullanılarak kümelerin yakınsamasına bakan Fuzzy C-means kümeleme metodu da kullanılabilir [45, 46]. Fakat K-means metodunun sahip olduğu, küme elemanlarının rastsal olarak atanması ve kümelemenin bunun üzerinden yürümesi dezavantajı Fuzzy C-means algoritmasında da bulunmaktadır.

Tez kapsamında Normalized-Cut ve Region-Growing segmentasyon algoritmaları, gerek karşılaştırma veya alt sistemlerde yardımcı çözüm aracı olarak kullanılmıştır. Normalized-Cut (N-cut) metodu, ağırlıklı graflar üzerinde bir setteki düğümün diğer bütün düğümlerle olan toplam bağıntısına bakılıp normalize edilerek kesim gerçekleştirmektedir. Problem polinom olmayan zorluk derecesinde (NP-hard) olduğundan, yaklaşık sonuç öz değer problem çözümüyle elde edilmektedir [20]. N-cut temel olarak grafi iki parçaya ayırmaktadır. Fakat segmentasyon sırasında tekrarlayan bir biçimde uygulanarak ikiden fazla bölgenin elde edilmesinde kullanılabilmektedir.

Alan Genişletme (AG) metodu, segmentasyon uygulamalarında sıkça kullanılan ve gerçekleşmesi görece olarak daha kolay olan, işlem esnasında piksel verileri üzerine uzamsal bilgileri de kullanan bir algoritmadır. Metot başlangıç için kullanıcıdan etrafındaki bölgeleri kontrol edebilmek için bir nokta veya bölge girişi yapmasını istemektedir. Ayrıca, ayırım yapılacak bölgenin genişletilmesi esnasındaki parametre ve eşik değerlerinin de önceden tanımlanması gerekmektedir. Genişletilmesi istenen bölge verilen eşik değerleri ve benzerlik bilgisi yardımıyla komşu piksel veya bölgelere bakılarak hedeflenen bölgeler bulunmaya çalışılır. Başlangıç parametreleri, metodun genişlemesi ve referans alması için temel oluşturulduğundan bu parametrelerin seçimi algoritmanın başarısı açısından önem taşır. Parametre girdisi ihtiyacı dezavantaj olmasına rağmen, birçok başlangıç noktası seçim metotları ve karşılaştırmalı çalışmalar literatürde bulunmaktadır [47, 48, 49]. Bu metodun avantajı ise, başlangıç noktası ve parametreler iyi seçilmesi durumunda yüksek başarımlı göstermesi ve kod olarak gerçekleşmesinin göreceli kolaylığıdır. Fakat gürültü ve bölgelerdeki ani değişimlere yani farklı ışık koşullarına duyarlılığı da göz ardı edilmemelidir. Benzerlikler, bulunan bölgelerin istenilenden çok daha büyük

çıkmasına neden olmakla birlikte, homojenlik ve aşamalı geçiş bölgeleri sonuçların istenilen şekilde çıkmamasına neden olabilir [50].

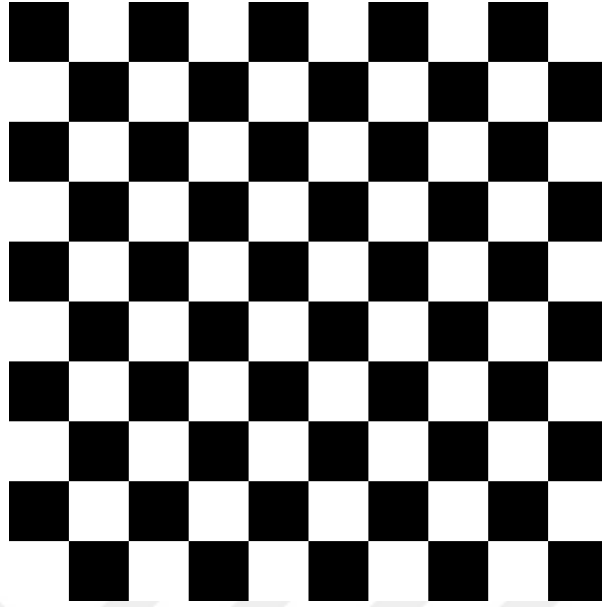
Segmentasyon algoritmalarının incelenmesinde Berkeley segmentasyon veri seti ve karşılaştırma için temel veri etiketleri kullanılmıştır [51]. Renk bölgelerinin tespitinin başarımlarını ölçmek için de Purity, F-measure, NMI gibi metrikler kullanılmıştır [52].

2.3. Stereo görüntü işleme ve engel tanıma

Araç sürüş destek sistemleri ve otonom araçlar için engellerin belirlenmesinde, farklı sensörler kullanılabilmekle birlikte stereo kamera verileri kullanılarak da aracın gittiği düzlem ve engeller tespit edilebilmektedir. Bölüm 2.3.1.'de girdi olarak kullanılan stereo verilerinin doğrultulmasına ve derinlik haritası çıkarım yöntemlerine değinilirken, Bölüm 2.3.2.'de ise çalışmada önemli yer tutan engel tespiti ile uzaklık tespiti için literatürde bulunan ve sıkça kullanılan yöntemler incelenmiştir.

2.3.1. Epipolar görüntü doğrultma ve derinlik haritası çıkarımı

Epipolar geometri, iki kameradan alından görüntüler arası ilişkileri incelemektedir. Kameraların aldığı sahne görüntüsünden bağımsız sadece kameraların içsel parametreleri de bu alanda incelenmektedir [53]. Kameralar nedeniyle görüntüde oluşan çarpıklığı giderme işlemi, derinlik haritasından uzaklık ölçüm işlemleri yine kameralar arası ve kamera içi özelliklerden yararlanılarak bulunabilmektedir. Kameradan kaynaklanan çarpık görüntüleri doğrultma (rectification) sürecinde, kamera görüntülerini çarpıklığını saptamak için dama tahtası kullanılmak suretiyle veriler alınır ve görüntü alımı üzerindeki bu bozukluklar giderilir. Dama tahtası, eşit büyüklüklü siyah ve beyaz karelerden oluşan görüntünün kâğıda çıktısı alınarak kullanılır. Bu araç kullanılarak sabit ölçüleri olan dama tahtasından alınan görüntüdeki kameranın etkisi hesaplanmış olan kamera matrisleri ve denklemlerle bulunabilmektedir [54].



Şekil 2.2. 64x64 piksel kare dama tahtası örneği 640x640 çözünürlüğünde

Şekil 2.2.'de görülen dama tahtası şekli kamera lens etkileri gibi veya kameradan kaynaklık hataları gidermede kullanılan en etkin yöntemlerden biridir.



Şekil 2.3. KITTI veri tabanı kalibrasyon test görüntüsü [55].

Yukarıdaki Şekil 2.3.'te görülen görüntüde olduğu gibi KITTI veri tabanındaki stereo görüntüler dama tahtası kullanılarak kalibre edilmiş, doğrultulmuş ve veri tabanına gözlem veri kayıtları eklenmiştir [55].

Kamera kaynaklı hatalar giderildikten sonra, stereo görüntü üzerinde derinlik haritası, kalibre edilmiş görüntünün aynı satırında yer alan aynı özellik veya pikseli bularak uzaklıktan kaynaklanan farklılıkların bulunması işlemi yapılır. Derinlik haritası çıkarımında birçok farklı derinlik haritası çıkarım yöntemi bulunmaktadır [56]. Görüntü gri seviye aydınlık farkları, özelliklerin kullanımı, pikseller veya

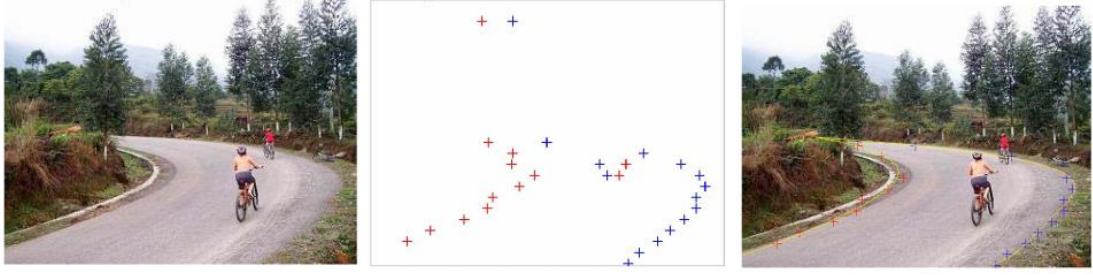
özellikler arasındaki korelasyon veya olasılıksal yöntemler ile derinlik tanımlaması yapılabilmektedir [57, 58, 59]. Tez çalışması kapsamında kullanılan derinlik haritası yöntemi, yarı global blok eşleştirme yöntemiyle bulunan blok görüntüler sonrasında her blok içerisindeki piksellerin SAD değeri hesaplanarak elde edilmiştir [57, 60, 61].

2.3.2. Stereo görüntüde yol, engel tanıma ve uzaklık ölçümü

Araç-yaya kazalarını önleyebilmek için, araçların üzerinde gittiği yolun bulunması, engel ve insan tespiti için bir gerekliliktir. Yaya tespiti için görüntü üzerinde sadece insan bulma yöntemlerin kullanılması, gökyüzünde insan aramaya benzemektedir. Hem yayaların bulma zamanını artıracak hem de çevre modellemesi ve engellerin tespiti olmadan yapılan çalışmanın araç güvenliği açısından da eksiklikleri bulunacaktır. Yayalar uzaklık mesafesine göre değiştiğinden veya engellerin üstünde olabileceğinden kaynaklanan hataların giderilmesi zor olacaktır. Bu sebeple yol tanımlanması, yol üzerinde bulunan yaya, bisikletli, engellerin bulunması, araç ve yaya güvenliği açısından önemlidir.

Yol tespiti için tekli görüntü ve çoklu görüntü sistemleri, farklı detektör ve araçların kullanımı literatürde yer almaktadır. Kamera sistemi, lazer veya benzeri detektörlerin maliyet ve temininden daha uygun olduğundan tez çalışması için seçilmiştir. Görüntü üzerinden yol, engel tespiti üzerine yapılan çalışmalar bu bölümde incelenmiştir.

HSV renk uzayı kullanılarak, yol kenarlarının tespiti ile aracın gidebileceği yolun tespiti ve yol bölgelerinin bulunması Huang ve arkadaşları tarafından incelenmiştir [62]. Çalışmada Şekil 2.4.'de görüldüğü gibi, tek görüntü üzerinden yol bölgesinin satırları incelenerek ve kenarlara bakılarak elde edilen noktalar doğru veya eğri şeklinde temsil ederek yol tanıma sistemi geliştirilmiştir.



Şekil 2.4. HSV bazlı tek görüntü üzerinden yol tespiti [73].

Tek görüntü üzerinden farklı mevsimlerde elde edilen farklı yapıdardaki yolların tespiti için ufuk noktası tahmini ağırlıklı Gabor filtresi yardımıyla piksellerin doku ve yönlerine göre yapılarak yol düzlemi tahmin edilmeye çalışılmıştır [63].

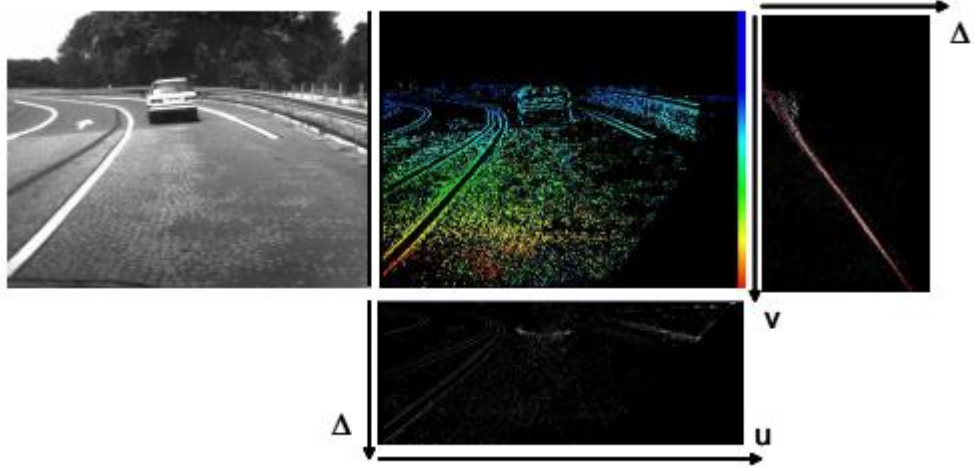
Işık değişiminin etkisini gidererek gölge veya farklı ışık koşullarından etkilenen kamera sistemlerinin daha iyi sonuç verilebilmesi için HSI renk uzayı kullanılarak benzerlik tabanlı sınıflandırmayla yol tanımlama çalışması gerçekleştirilmiştir [64]. Işıktan etkilenen bölge tanıma sistemlerinin yol tespitinde, bu çalışma kullanılarak ışığın bölge bulma üzerindeki etkisi giderilebilir.

Yol tanıma algoritmalarını değerlendirme ve testleri için farklı çalışmalar literatürde yer almaktadır. Bunlardan birisi Alvarez ve Lopez tarafından geliştirilen yol tanıma hedef değerlendirme çalışmasıdır [65]. Bir diğeri ise yol tanımlama algoritmaları için KITTI yol veri setinin kullanıldığı piksel bazlı test ve değerlendirme çalışması Fritsch ve arkadaşları tarafından yapılmıştır [66].

Tek görüntü yerine, çift kamera ile stereo görüntüde yer düzlemi ve engel tespiti hakkında başarılı çalışmalar bulunmaktadır. Stereo görüntü üzerinden zemin ve yol tespiti, tek görüntü üzerinden yapılan çalışmalardan farklı olarak, derinlik bilgisinin ölçümü ve geometrik yer düzleminin tanımlanabilmesiyle başarılı sonuçlar elde edilmiştir.

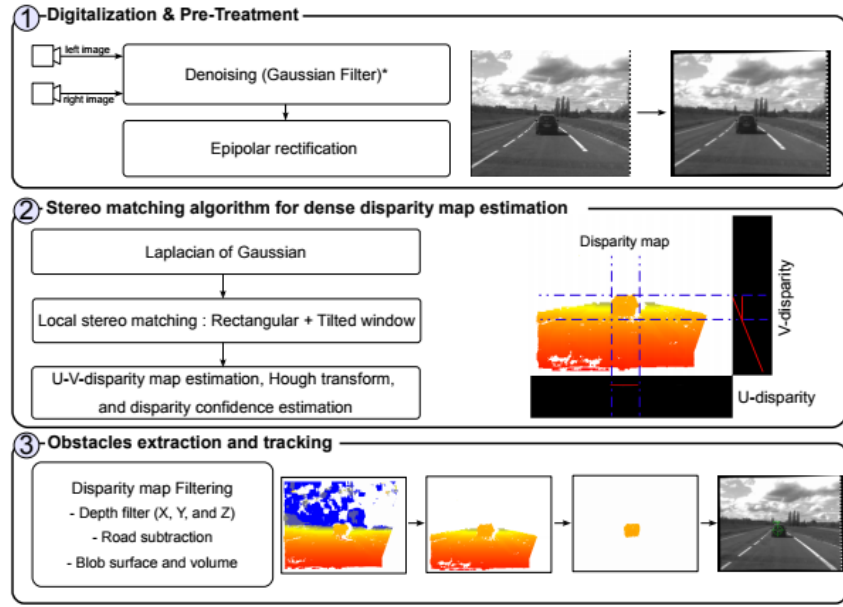
Squet ve arkadaşları derinlik haritası kullanarak geometrik dönüşüm ile V-disparity verisi elde etmiş, sonrasında V-disparity üzerinden Hough dönüşümü ile engeller ve yol düzlemini tespit etmişlerdir [67]. Ayrıca Şekil 2.5.'de görüldüğü üzere V-

disparity'nin yanında U-disparity ile düzlem üzerindeki engellerin tanımlanmasını kolaylaştıran dönüşümler incelenmiştir. Aynı alanda Labayrade ve arkadaşlarının V-disparity verisi ile düz olmayan zeminlerde engellerin tanımlanması için çalışmaları bulunmaktadır [68].



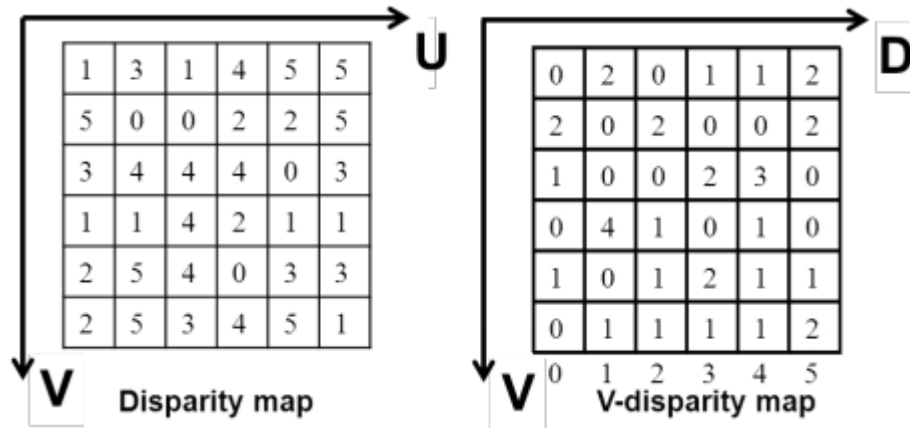
Şekil 2.5. Stereo görüntüsü derinlik haritasından elde edilen U-disparity ve V-disparity görüntüleri [78].

Normal V-disparity yöntemi üzerine ağırlık verisi ekleyerek engellerin otoyol üzerindeki yerlerinin tespiti, lokal derinlik hesaplamasıyla ve Hough dönüşümü kullanılarak Şekil 2.6.'deki gibi gerçekleştirilmiştir [69].



Şekil 2.6. Ağırlıklı V-disparity verilerinin kullanılarak engellerin tespiti [69].

Off-road ortamlarda ve kompleks yer düzlemlerinde v-disparity'nin kullanımı mümkün olmaktadır [70]. Yer düzlemi eğim durumu belirtilmiş ve disparity'den v-disparity dönüşümünde sayısal olarak dönüşümün nasıl gerçekleştirildiği Şekil 2.7.'deki gibi ifade edilmiştir.

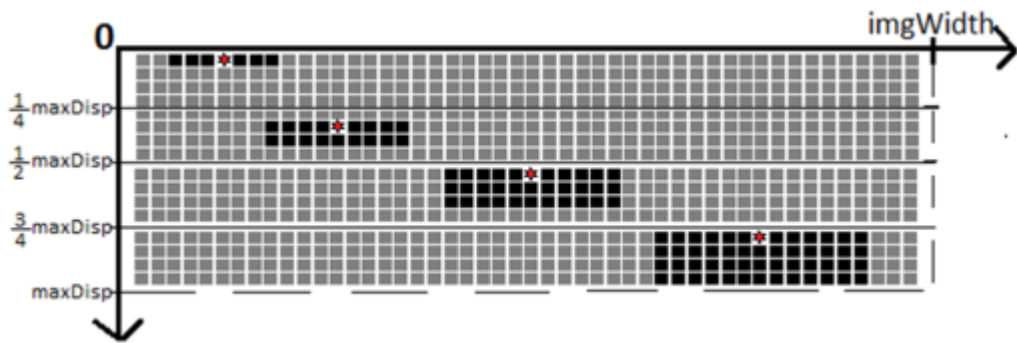


Şekil 2.7. Sayısal olarak derinlik haritası ve V-disparity karşılığı [81].

Genellikle kullanılan Hough dönüşümü yerine global korelasyon yöntemi yol düzleminin belirli olmadığı veya hataya düşebildiği koşullar için kullanılması Zhao ve arkadaşları tarafından yapılmış çalışmada önerilmiştir [71].

Global korelasyon yerine Kostavelis ve arkadaşlarının yaptığı çalışmada V-disparity içerisinde Hough dönüşümüyle elde edilen doğru ile tolerans bölgesi tanımlanmışlardır [72]. Hough dönüşümü kullanıldıktan sonra belirlenen tolerans bölgesi dışındaki alanlar özellik vektörü şeklinde toplanarak tanımlanmış ve işlenmiştir. Bu özellik vektörü derinlik görüntüsünden alınan herhangi bir yükseklikteki satır verilerinin V-disparity karşılıklarına denk gelmektedir. Çalışmada tolerans değeri sayesinde Hough dönüşümünden elde edilen doğrudaki hatalar azaltılmıştır.

Şehir trafiğinde UV-disparity tabanlı engel tanıma ve yaya tanıma çalışması Iloie ve arkadaşları tarafından geliştirilmiştir [73]. SORT*SGM stereo yeniden yapılandırma tekniği kullanılarak derinlik haritası noktaları oluşturulur. Sonrasında V-disparity ile elde edilen yol düzlemi ve U-disparity ile tespit edilen engeller boyut kısıtlamalarına göre HOG filtresinden geçirilerek özellikler elde edilir. Sonrasında ise PCA ve SGM ile sınıflandırma sisteminden geçirilerek yaya olup olmadığı sonucuna ulaşılmaktadır. Burada insan tanıma için seçili bölgelerden alınan uzaklığa bağlı komşuluk alanı seçimi önemlidir. İncelenen bölge sayısını belirli bir genişlik eşiği kullanarak düşürmektedirler. Çalışmalarında yer düzleminin tespiti için Hough dönüşümü kullanılmıştır. Sistemde bulunan bölgeler görüntü boyutunun HOG filtresine uygun boyutlandırılmasından sonra insan tanıma sonuçları incelenmiştir.

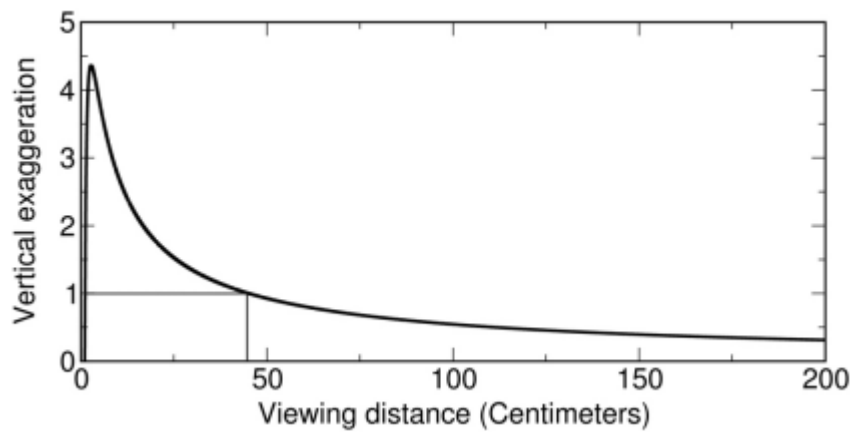


Şekil 2.8. U-disparity ile bulunan engel bölgelerinin komşuluklarının incelenmesi [73].

Şekil 2.8.'da görülen U-disparity kullanımı yaya ve engellerin uzaklıklara göre boyutlarının tahmini açısından önemlidir. Yaya tanıma sistemi ve çevrenin modellenmesi tez çalışması kapsamında yapılan çalışmalarla doğrudan ilişkili olduğundan bahsi geçen sistem incelenmiş ve farkları bölüm 3. ve bölüm 4. üzerinden görülebilir.

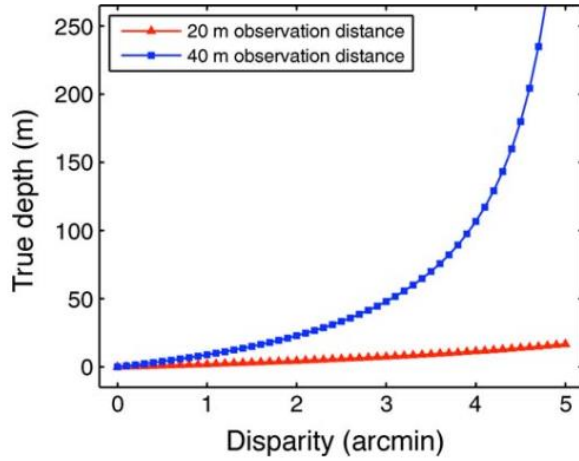
Sistem tasarımı sonrasında tespit edilen yayaların güvenliğini artırmak amacıyla nesne takibi ve analiz yöntemleri kullanılabilir. Özellikle bu alanda birden fazla nesne takibi ve araçlar içerisindeki kameralarla nesnelerin hareketlerinin bulunması incelenmiştir [74, 75].

Uzaklık ölçümünde, eğer ikili kamera sistemindeki kameralar arası uzaklık ve bütün kamera parametreleri biliniyorsa, derinlik haritasından elde edilen uzaklık bilgisi kamera çözünürlüğü de kullanılarak çarpıklığa (distortion) uğramamış görüntü verisinden nirengi (triangulation) yöntemiyle uzaklık bilgisi ölçülebilmektedir [53, 76, 77]. Ayrıca dikkat edilmelidir ki bu hesaplarda ikili stereo görüntü bilgisinden elde edilen derinlik bilgisi, uzaklıkla doğrusal olmayan bir biçimde tersten orantılıdır [78]. Bu oran, Rosas ve arkadaşlarının yaptığı çalışmada stereo görüntü üzerinden nesnelerin algılanma oranının değişimini gösterdiği Şekil 2.9.'te görülebilmektedir [79].



Şekil 2.9. Stereo görüntüde dikey değişime karşı uzaklık algısı [90].

Nesne algısı ve değişim yerine, test görüntüsündeki gerçek uzaklık bilgisi ile derinlik haritası görüntüsü arasındaki oran Şekil 2.10.'de görüldüğü gibi Palmisano ve arkadaşları tarafından incelenmiştir [80].



Şekil 2.10. Uzaklığa karşı derinlik haritası değerleri [91].

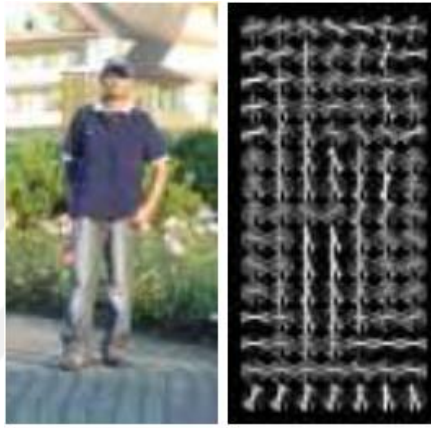
2.4. Yaya tanıma sistemleri ve HOG algoritması

Araç destek sistemlerindeki hataları ve kazaları minimuma indirmek için, araç ortamının modellenmesi ve nesnelerin tanımlanması önem arz etmektedir. Bu sebeple görüntü üzerinden insan ve araç tanımlama, renk veya ışık bazlı bölge tespitinden sonraki aşama olarak uygulanmalıdır. Tespit edilen bölgelerin hangi nesneye ait olduğunun bulunması nesne tanıma sisteminin bir gerekliliğidir. Aranılan nesnenin bölgesi ve özellikleri tanımlanmadığında tüm görüntünün incelenmesi ve nesnelerin büyüklüklerinin belirlenmemesi hem hata oranını artıracak hem de işlem sürecini uzatacaktır. Çalışma kapsamında yayaların tespiti hedeflendiğinden literatürdeki insan tanıma sistemleri incelenmiştir.

Yaya tanıma sistemlerinde ışık bilgisi üzerinden silüet çıkarımı yaparak insan tanıma işlemini yapan sistemler bulunmaktadır. Bunlardan bazıları matematiksel denklemlerle belirli bir eşik değeri belirlemekle birlikte bazı çalışmalar da Fourier tanımlayıcıları kullanmıştır [81, 82]. Farklı bir biçimde lokal olarak insan uzuvlarını tanıyarak hiyerarşik insan tanıma yapan sistemler bulunmaktadır [83]. Öğrenme

metotları kullanarak insan pozundan bağımsız olarak yaya tanıma sistemi Lin ve Davis tarafından incelenmiştir [84].

Günümüzde kabul gören ve yaygın olarak kullanılan sistemlerden birisi de eğitim tabanlı özellik vektörü kullanılarak eğitilen karar sistemleridir [85, 86] . Şekil 2.11.'de örnek görüntüsü verilen, Dalal ve Triggs tarafından geliştirilen HOG sistemi INRIA veri tabanı üzerinde yüksek başarı oranı gösterdiğinden dolayı tez çalışmasında alt sistem olarak tanımlanarak kullanılmıştır [87].



Şekil 2.11. Test görüntüsü (sol) ve HOG özellik vektörü (sağ) [66].

Stereo sistemlerde yaya algılama için HOG tabanlı çeşitli kıyaslama, değerlendirme ve gerçek zamanlı uygulama sistemleri geliştirilmiştir [88, 89]. Ayrıca ileriki aşamalarda bulunan insanların gelecek yön vektörlerinin tahmini için çalışmalar yapılmıştır [90]. Performansın artırılması konusunda, yaya güvenliği için önemli bölgelerin tespiti de literatürde incelenmiştir [91].

2.5. Kullanılan veri tabanları, veri yapıları ve yöntemler

Tez içerisinde kullanılan veya kullanılması önerilen metotları kısaca inceleyecek olursak;

a) QuadTree ağaç veri yapısı:

Özellikle kümeleme ve segmentasyon gibi alanlarda çalışma hızı artırımı için kullanılabilir. Bunun dışında görüntüdeki özellikler ve bu özelliklerin veri yapılarında tutulması önerilir. QuadTree ağaç veri yapısı Atallah ve Blanton tarafından yazılmış algoritmalar ve hesaplama teorisi kitabından incelenebilir [92].

b) Ölçek-Uzay Yöntemleri (Scale-Space Processes):

Görüntüde çözünürlük düşürmede kaybedilen ve korunan veriler, göz açısından önemli gördüğümüz özellik veya görmezden gelinebilecek noktalar üzerinden bize ipucu vermektedir. Ayrıca bu özellikler renk alanı değişimi ve o alanın baskın renk özellikleriyle de ilgili bilgiler içermektedir. Farklı ölçeklerde yapılarda görülen değişimler Lindeberg'in çalışmasından ve çıkardığı kitap üzerinden inceleyebilir [93, 94].

c) Gauss Karışım Modeli (GMM):

Görüntüde gözle gördüğümüz fakat sadece uzaklık olarak ölçtüğümüzde bazı bölgelerin yoğunluk bilgisinin kaybolduğunu ve ölçülemediği GDD kümeleme yöntemi araştırılırken gözlemlenmiştir. Bu sebeple noktaların dağılımlarının olasılıksal temsili, ancak uzaklıklarla birlikte incelendiği zaman ve bu aradaki ilişki ortaya çıkarıldığında daha doğru sonuçlar vermektedir. GDD kümeleme metodunda yoğunluğu belirlemek için ve olasılıksal dağılımını modellemek için GMM (Gaussian mixture model) yöntemi kullanılmıştır. GMM yöntemi McNicholas'ın kitabında ayrıntılı olarak anlatılmaktadır [95].

d) Öklid Uzaklığı ve diğer uzaklık ölçüm yöntemleri:

İki nokta arasındaki uzaklığın ölçümü, veri uzayındaki noktaların dağılımına ilişkin özellikler barındırmaktadır. Genellikle Öklid uzaklığı bu ilişkileri lineer olarak hesaplamak ve göstermek amacıyla kullanılmıştır. Farklı uzaklık ölçümlerinin kümelemeyi nasıl etkilediği ile ilgili testler K-means üzerinde Singh ve arkadaşları tarafından yapılmıştır [96] .

e) Kümeleme Ölçüm teknikleri (Purity- NMI- Fscore):

Kümelerin analizi ve kümeler arası doğruluk, hata oranları doğruluk tablolarına (ground truth) bakılarak hazırlanmıştır. Manning ve arkadaşlarının yayınladıkları kitapta yer alan kümelerin değerlendirilmesi başlığında Purity, NMI ve F-score gibi küme başarımları analiz yöntemleri tez çalışması kapsamında incelenmiş ve kullanılmıştır [52].

f) Destek vektör makineleri:

Destek vektör makinesinin İngilizce karşılığının kısaltması olan SVM (Support Vector Machine), gözetimciyle (supervised) eğitim sistemlerinde karar alma sistemi olarak tez kapsamında kullanılmıştır. İnsan tanıma filtresinde, eğitimden sonra karar alma işlemini gerçekleştiren sınıflandırma algoritmasıdır. İşlem sonunda girdi verisine etiket atanarak istenilen sonuca ulaşıp ulaşılamadığı tespit edilmektedir. Destek vektör makineleri ve uygulamalarıyla ilgili ayrıntılı bilgiler Ma ve arkadaşları tarafından yayınlanan kitap üzerinden incelenebilir [97].

g) Regresyon eğrileri:

Çalışma kapsamında V-disparity üzerinde doğru tespiti için ağırlıklı en küçük kareler (WLSS) regresyon çözümleme yöntemi olarak kullanılmıştır. Regresyon eğrileri ve WLSS konusunda daha fazla bilgiye Alfassi ve arkadaşlarının yayınladığı analitik verilerin istatistiksel işlenmesi hakkındaki kitabının regresyon analizi bölümünden ulaşılabilir [98] .

h) Hough Dönüşümü:

Şekil ve doğru segmentlerinin tespitinde Hough dönüşümü sıklıkla kullanılmaktadır. Hough dönüşümü ile ilgili ayrıntılı bilgilere Leavers'ın yayınladığı kitap yoluyla ulaşılabilir [99].

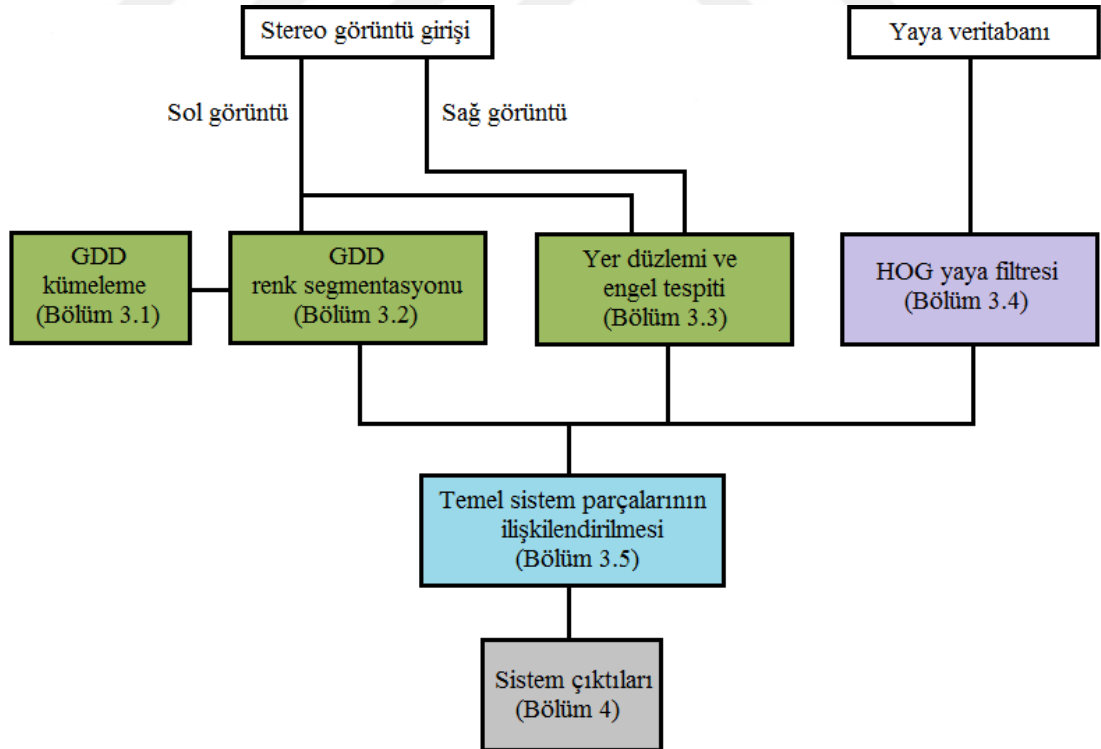
i) Veri tabanları:

Tez çalışması kapsamında birçok farklı veri tabanı kullanılmıştır. Kümeleme metodu geliştirilmesi ve testleri esnasında şekil setleri Pasi ve arkadaşlarının oluşturduğu websitesi üzerindeki belirtilen makalelerden referans alınarak incelenmiştir [100]. Renk segmentasyonu için Berkeley'in segmentasyon veri seti ve karşılaştırmalarından yararlanılmıştır [51]. İnsan tanıma sistemi için seçilen HOG algoritmasının test ve eğitimi için ise çalışma esnasında kamera ile çekilen görüntülerin yanı sıra INRIA, MIT ve Daimler Chrysler yaya veri setlerinden yararlanılmıştır [87, 101, 102].



BÖLÜM 3. MATERYAL VE YÖNTEM

Bu çalışmada; engel tanıma, yer düzlemi belirleme, renk özelliklerine göre araç sürüş bölgesinin ayrımı, yaya tespiti ve uzaklık tespiti işlemlerini yapan stereo görüntü tabanlı bir sistem geliştirilmiştir. Sistem, Şekil 3.1.'de yaya veri tabanını yaya filtresi eğitiminde, stereo görüntüleri de giriş olarak kullanmaktadır. Ayrıca sistem; kümeleme, renk segmentasyonu, engel ve yer düzlemi tespiti, HOG yaya tanıma algoritması gibi alt birimleri içermektedir. Her bir birim yazının devamında kendi alt başlığında detaylı olarak irdelenmiştir. Daha sonra, bu alt birimlerin birleştirilmesiyle elde edilen sonuçlar, birbirleriyle ilişkilendirilerek Bölüm 3.5. içerisinde Şekil 3.38.'de tanımlandığı gibi kullanılmıştır.



Şekil 3.1. Sistem genel blokları ve bölüm alt başlıkları.

Sistemde, farklı alanlarda işlev gören birçok farklı alt sistem geliştirilmiş ve yapılan testlerde kullanılan veri tabanları ile başarımlar ölçümleri, metotların incelendiği alt başlıklar içerisinde belirtilmiştir. Tez kapsamında; kümeleme, renk segmentasyonu ve görüntü engel tespiti alanlarında yenilik içeren metotlar geliştirilmiştir. Ayrıca stereo görüntü işlemlerinde kullanılan derinlik haritası tespiti ile yaya tanıma sistemlerinde kullanılan HOG algoritması sistem içerisinde kullanılmıştır. Bunlar haricinde; yayaların antropometrik oranlarının kullanımı, öncelikli işlem listesi oluşturulması, ilgi bölgesi (region-of-interest) tespiti gibi farklı yöntemler daha yüksek başarımlı sistem çıktılarının üretilmesi için kullanılmıştır.

3.1. GDD kümeleme

Genel olarak en çok bilinen kümeleme algoritmalarına baktığımızda, birçoğu rastsal olarak giriş parametrelerine göre kümeleme işlemlerini tamamlamaktadırlar. Genellikle kümelemeyi sağlayan değerler ilk başta belirlenerek bu değerlere göre bütün uzaydaki veriler gruplanmaktadır. Bazı parametre tahmin yöntemleri ile meanshift benzeri parametre istemeyen yöntemler de bulunmaktadır. Bu tez çalışması kapsamında oluşturulan GDD kümeleme yöntemi, herhangi bir parametre gerektirmeyen, rastsal olmayan, Gauss olasılıksal yoğunluk dağılımı ile Öklid uzaklık bilgisini kullanan ve bu ikisinin birbirleriyle olan doğal ilişkisiyle kümeleri tanımlayan bir çalışmadır [23].

Tez kapsamında yeni kümeleme metodu olan GDD kümeleme, tasarlanmış, gerçekleştirilmiş ve test edilmiştir. Bu kümeleme çalışmasında amaç olarak, kümeleme algoritmalarının gereksinim duyduğu parametre ihtiyacı ve sonuçlarda oluşan rastsallığın giderilmesi hedeflenmiştir. Farklı yoğunluğa sahip bölgelerin farklı kümelerde ifade edilmesi, birbirleriyle ilişkili olan bölgelerin ise bir küme içerisinde ifade edilmesi algoritmanın önemli bir adımını oluşturmaktadır. Giriş verilerine bağlı olarak doğal kümelerin bulunması Henning'in çalışmasında incelenmiştir [103]. Birçok kümeleme metodunun sahip olduğu kısıtlamaları aşmak ve doğal kümelerin bulunması amacıyla GDD kümeleme hedef olarak aşağıdaki özellikleri barındırmaktadır:

1. Birbirine benzer derecede yoğunluk ve uzaklıkla bağlı veri örneklerinin aynı kümede temsil edilmesi.
2. Farklı yoğunluktaki birbirine bağlı olmayan verilerin farklı kümeler içerisinde temsil edilmesi.
3. Kademeli olarak azalan veya artan uzaklık ile yoğunluklar birbirine bağlı olacağından aynı küme içerisinde tanımlanması.

Kümeyi tanımlayan sınırlar hakkında ve amaçlarıyla ilgili spesifik sabit bir tanımlama bulunmadığından, yukarıdaki amaçlar GDD kümelemede kriter olarak kullanılmış ve yazılım bu temeller üzerine oluşturulmuştur. Henning tarafından ortaya konulan bazı hedef ve tanımlar, görüntü segmentasyonu ve nesne tanımlama alanlarında GDD kümeleme algoritmasına fayda sağlamıştır [104]. GDD algoritması, yoğunluk bilgisini, kümelerin toplanma yerleri ve merkezlerini bulmak amacıyla yoğunlukla uzaklığın ilişkisini hesaplayarak kümeleri oluşturmaktadır. Kümeleme çalışmasının önemli bir kısmını, veri uzayındaki her bir örneğin diğerine Gauss değeri etkisini ifade etmek için kullanılan GMM'in (Gaussian Mixture Model) oluşturulması ve bu etkilerin uzaklıkla ilişkisinin tanımlanması oluşturmaktadır. GMM hesaplandıktan sonra küme bölgelerini tanımlamak için standart sapma gibi istatistiksel yöntem ve değerler kullanılmıştır. En yoğun noktalar küme merkezi olarak belirlenerek küme temel noktası olarak kabul edilip; çevre veri örnekleri, yoğunluk değişimleri, uzaklık değişimlerine göre bütün komşuluklar belirlenir. Komşuluk taramasında değişen bölgesel yoğunluk ve uzaklık parametrelerine göre veriler, incelenen kümeye eklenir. Burada dikkat edilmesi gereken nokta, küme merkezi dışında diğer veri örneklerinin de birbirleriyle olan komşuluk ilişkisinin, örneklerin kümelere aidiyetini etkilemesidir.

3.1.1. Veri yapıları ve değişkenler

Giriş veri matrisinde, n değerini veri uzayındaki veri örneği sayısı, d değerini boyut veya özellik sayısı ve veri matrisini $X_{n \times d}$ olarak tanımlarsak, giriş veri setimiz aşağıdaki Denklem 3.1'de görüldüğü şekilde tanımlanabilir.

$$X_{nxd} = [X_1 X_2 X_3 \dots X_n]^T \quad (3.1)$$

Giriş verisindeki her satır veri örneği vektörü olarak tanımlandığında, örnek vektör aşağıda belirtilen Denklem 3.2'deki gibi olacaktır.

$$X_i = [X_{i1} X_{i2} X_{i3} \dots X_{id}] \quad (3.2)$$

X_{nxd} 'deki sütun sayısı veya boyut numarası, yeni özellikler eklendiğinde artmaktadır. Veri içerisindeki özelliklerin ilişkisi kümeleme hedefine uygun olmalıdır. Kümelemede eğitim safhası bulunmadığından verilerin ilişkisi ve oranları (m, cm gibi) önem taşımaktadır.

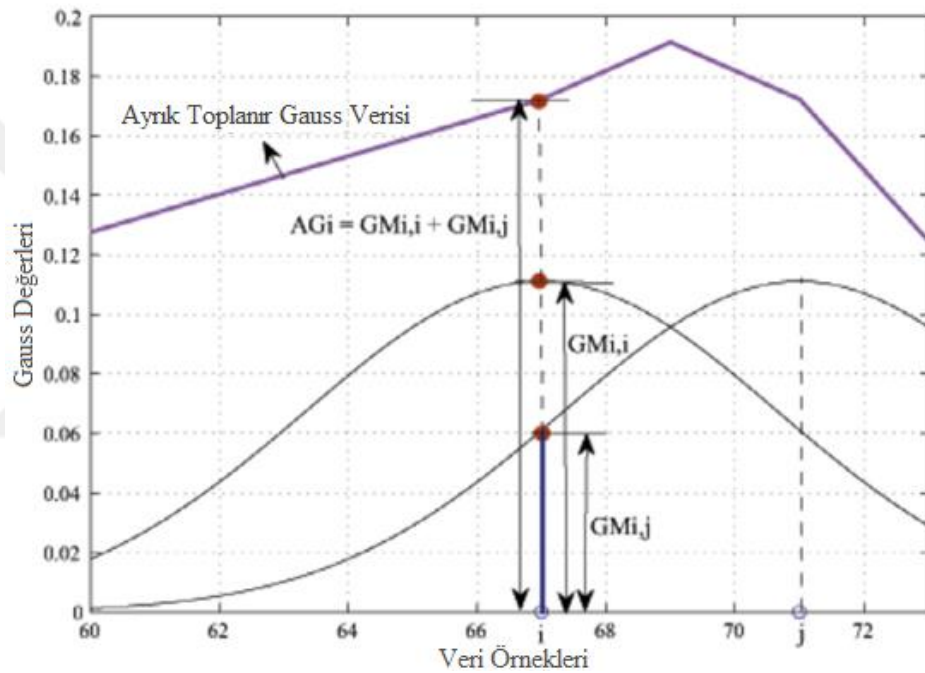
Algoritmada, matris, vektörler ve listeler gibi veri yapıları kullanılmaktadır. Bu veri yapıları aşağıdaki gibi tanımlanabilir:

1. Giriş verisi ortalama ve sapma vektörleri, μ_d ve σ_d : Sayısal olarak X_{nxd} matrisi verileriyle, μ_j ve σ_j verileri her sütun için hesaplanır. Bu değerler hem ön-işleme hem de Gauss dağılımı hesaplamalarında kullanılmaktadır.
2. Gauss matrisi $GM_{n \times n}$: $GM_{n \times n}$ matrisi veri örneklerinin karşılıklı Gauss dağılımı etkilerini tutmaktadır. Bu negatif olmayan, simetrik, köşegen (diyagonal) matriste, Denklem 3.3'te görüldüğü şekilde uzaydaki diğer veri örneklerinin incelenen veri setine olan toplam etkisi hesaplanarak saklanmaktadır.

$$GM_{i,j} = \exp\left(-\sum \frac{(x_{ik} - x_{jk})^2}{2c}\right), \quad c = \sqrt{\frac{\mu_k \sigma_k}{2\pi}} \quad (3.3)$$

Üstteki denklemde 'j' noktasının Gaussian yoğunluk dağılımının 'i' noktasındaki üzerindeki etkisi 'k' boyutlu eliptik Gauss fonksiyonu $f(\mathbf{X}_i, \mathbf{X}_j)$ ile hesaplanması için kullanılmıştır. Bu etki Şekil 3.2.'de mavi çizgi ile gösterilmiştir. Denklem 3.3'te "c" değeri, örnekler arası bağıntı katsayısı olarak adlandırılmıştır. Eğer örnekler bir

boyut üzerinde geniş bir biçimde dağılmış ise, bu dağılmış noktaların Gauss etkisinin yeterli bir biçimde etkileşime geçebilmeleri, doğru kümelerin bulunmasını sağlar. Bu nedenle Gauss dağılımını daha geniş bölgeye yaymak için bu boyuttaki istatistiksel değerler olan ortalama ve sapma kullanılarak örnekler arasında bağıntıyı sağlayacak olan 'c' değeri bu verilere göre hesaplanmıştır. Genel manada, bu katsayı; komşuluklara ve boyuta göre yoğunluk dağılımının nasıl modellenmesi gerektiğini belirlemektedir. Her boyutun dağılımı diğerinden farklı şekil ve oranda olacağından bu katsayı, o boyuta uygun olarak hesaplanmaktadır.



Şekil 3.2. İki farklı rastgele noktaya göre eliptik dağılımı ve Ayrık Toplanır Gauss verisi üzerindeki etkisi (Mavi çizgi j noktasının i noktası üzerindeki Gauss verisi etkisini gösterir. Üstteki mor çizgi ise ayrıklıkların Gauss değerlerinin doğrularla kesişimlerini belirtmektedir).

Karşılıklı bağıntı katsayısı, sabit bir değer olmamakla birlikte boyut özelliklerine göre değişmektedir. Bu katsayı, algoritmada önemli bir değer olmasına rağmen kullanıcının giriş verisine bağımlı bırakılmamasının nedeni bu değerın kullanıcı tarafından tahmininin ve girişinin zor olmasıdır. Ayrıca örneklerin farklı boyutlarının birbirleriyle ilişkisi ve yoğunluk modellemesi bu değerlere bağlıdır. Eğer bu değer çok düşük alınırsa, Gauss değerleri birbiri üzerine gelemeyecek ve bu sebeple bu değerlerin karışımı tek Gauss fonksiyon değeri olan $GM_{i,i}$ şeklinde olacaktır. Bu

durumda kümeleme işlemi, her nokta ayrı bir küme olarak görüneceğinden anlamını yitirecektir.

Her bir noktanın etkisinin bulunduğu veri uzayında yoğunlukların ve uzaklıkların doğru ifade edilebilmesi ve verilerin yukarıda bahsedilen şekilde birbirlerine bağımlılığı bulunduğu halde sadece noktasal olarak tek elemanlı kümeler şeklinde tanımlanmasının önüne geçmek için, örnekler arası ortalama ve sapma değerleri hesaplanarak verilerin göreceli komşuluklarında nasıl dağılım gösterdiği hesaplanmıştır.

3. Toplanır Gauss vektörü, $\mathbf{AG}_n$: Toplanır Gauss (Additive Gaussian) değerlerini barındıran $\mathbf{AG}_n$ vektörü, $\mathbf{GM}_{n \times n}$ satırındaki değerlerin Denklem 3.4'teki şekilde toplanmasıyla elde edilir. Ayrık toplanır Gauss değerlerinin doğrusal ifadesi Şekil 3.2.'deki mor çizgiden görülebilir.

$$AG_i = \sum GM_{i,j} \quad i, j = 1, \dots, n. \quad (3.4)$$

4. Gauss nokta ortalama vektörü, $\mathbf{GPM}_n$: $\mathbf{GPM}_n$ değeri $\mathbf{GM}_{n \times n}$ matris satırlarının diyagonal matris elemanları dışında toplamının eleman sayısına bölünmesiyle Denklem 3.5'te görüldüğü gibi elde edilmektedir.

$$GPM_i = \left(\frac{1}{n-1} \right) \sum GM_{i,j} \quad i, j = 1, \dots, n. \quad (3.5)$$

5. Gauss nokta sapma vektörü, $\mathbf{GPD}_n$: $\mathbf{GPD}_n$ değeri $\mathbf{GM}_{n \times n}$ matris satırlarındaki Denklem 3.5'ten elde edilen GPM_i değerlerinin standart sapmasının hesaplanmasıyla bulunur.
6. Uzaklık matrisi, $\mathbf{DM}_{n \times n}$: veri seti örnek noktaların birbirleri arasındaki Öklid uzaklıklarının değerlerini tutan matristir. Pozitif, sıfır diyagonal, simetrik bir matristir. Kullanılan yöntem 'k' boyutlu uzayda Denklem 3.6'dan görülebilir.

$$DM_{i,j} = \sqrt{\sum (x_{ik} - x_{jk})^2} \quad (3.6)$$

7. Uzaklık nokta ortalama vektörü, **DPM_n** : **DPM_n** değeri, diyagonal **$DM_{n \times n}$** değerlerinin dışındaki satır değerlerinin eleman sayısına bölünmesiyle Denklem 3.7'deki şekilde elde edilir.

$$DPM_i = \left(\frac{1}{n-1} \right) \sum DM_{i,j} \quad i, j = 1, \dots, n. \quad (3.7)$$

8. Uzaklık nokta sapma vektörü, **DPD_n** : **DPD_n** değeri **$DM_{n \times n}$** matris satırlarındaki Denklem 3.7'den elde edilen DPM_j değerlerinin standart sapmasının hesaplanmasıyla bulunur.

3.1.2. Algoritma gerçekleştirilmesi

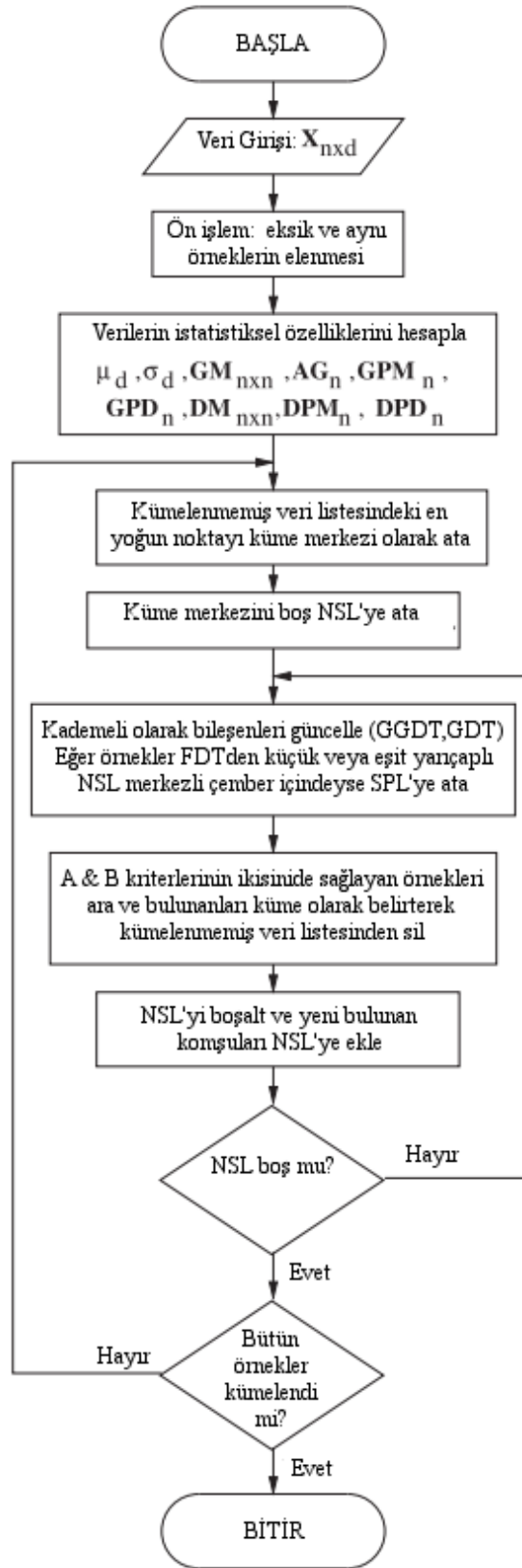
Giriş verisi elde edildikten sonra, daha iyi sonuçlar elde edilmesi için ön işlem den geçirilir. Özellikle değişkenlik göstermeyen sütun değerlerinin çıkarılması, işlem yükünü azaltacaktır. Sütun değerlerinin değişiklik göstermemesi durumunda incelenen boyuttaki verilerin, **$GM_{i,j}$** değerlerine etkisi sıfır olacağından buna bağlı **AG_i** toplam değerleri de sıfır olacaktır. Bu sebeple incelenen boyuttaki bu hesaplamalar, kümeleme işlemine etkisi olmamakla birlikte hesap yükünü artırmış olacaktır.

Aşağıdaki liste, yapılan ön-işlem basamaklarını göstermektedir:

1. Filtreleme: Her örnek, veri setinde taranarak eğer eksik özellik ve bilgi bulunduruyorsa veri girdi setinden çıkarılır. Bu aşamada, eşit değerlere sahip kopya örnek veri girdileri de aynı şekilde liste dışına atılır.
2. Boyut azaltma: Eğer bir boyut (özellik) vektörü sıfır varyansa sahipse, bu boyut girdi verisindeki sütunu çıkarılır. Yine aynı kısımda boyut indirgeme algoritmaları da kullanılabilir.

Ön-işleme işleminden sonra, veriler Şekil 3.3.'deki akış diyagramına göre hesaplanır. Başlangıçta hesaplanan değişkenler sadece bir kere hesaplandıktan sonra algoritmanın geri kalan kısımlarında kullanılır.





Şekil 3.3. GDD kümeleme akış diyagramı.

Sonraki adımda, kümelenmemiş veri örnek listesi, giriş verilerinin tutulması için oluşturulur. Her örnek, kendine ait numara ile saklanarak örneklerin izlenmesi ve kontrolü sağlanır. Başlangıçta kümelerin en yoğun noktasını ifade eden ağırlık tabanlı merkezler, kümelenmemiş veriler arasından en yüksek değere sahip ortalama Gauss nokta değerlerine göre belirlenir. Kümeye ait elemanlar belirlendikten sonra, listeden kümelenmiş örnekler çıkarılıp, kümelenmemiş örnekler üzerinden işlem bütün örnekler kümelenene kadar tekrar eder.

Kümeleme işlemi esnasında komşuların ve eşik değerlerinin belirlenmesi için iki önemli liste oluşturulur ki bunlar yakın örnek listesi SPL ve komşu arama listesi NSL olarak adlandırılmıştır. SPL uzaydaki her örnek kullanılarak oluşturulmakla birlikte NSL sadece kümelenmemiş örnekler arasından seçilmektedir. Başlangıçta NSL sadece küme merkezini listede tutar. SPL ise NSL'deki her örnek verisinin, sabit uzaklık eşiği FTD yarı çapını içeren bölgedeki verileri içerir. FTD değerleri, Denklem 3.10 ile Öklid uzaklıkları kullanılarak hesaplanır. SPL Denklem 3.11 ve Denklem 3.13'de kademeli yoğunluk ve uzaklık değişimlerini belirlemek için kullanılır. Bu sebeple bölgelerdeki her örneğin hesaplamalara dahil edilmesi gerekli olmaktadır.

FTD, küme merkezine bağlı ve oluşturulduktan sonra komşularından etkilenmeyen göreceli bir eşik değeridir. FTD her küme için farklı değerde olmasına rağmen, küme elemanlarının bulunma aşamasında sabit kalmaktadır. FTD kullanarak, kümelerin genel yoğunluklarına göre bir belirleme ve komşulukları tanımlanması sağlanır ve kümeler oluşturulur. Küme komşuluğunda bulunan örnekler ayrıca, kademeli uzaklık eşiği GDT, sabit Gauss yoğunluk eşiği FGDT, Gauss kademeli yoğunluk eşiği GGDT değerlerini de sağlaması gerekir. Bu eşik denklemleri, komşu örneklerin küme içi ve kademeli değişim farklarına uyarak kümeye olan aitliklerini belirler.

Kümeler farklı şekil ve bağlantılar gösterebildiğinden, kademeli değişim eşikleri bu tarz durumlarda rastsal şekillerin yoğunluk ve uzaklıklarını kontrol etmek amacıyla kümeleme işleminde önemli rol oynar. Her kümeleme tespiti taramasında örnekler

kümeye bu dinamik değerler kullanılarak eklenir. Kademeli eşik hesaplamaları iki adımdan oluşur:

1. İncelenen örnekteki komşulukların varyansı
2. Önceki hesaplamalardan elde edilen ortalama ve sapma varyans değerleri

GDD terimi, örneklerin kümeye dahil olmasını sağlayan kriterlerden gelir. Burada yoğunluk kriteri ve uzaklık kriteri (A ve B kriterleri) olarak Denklem 3.8 ve Denklem 3.9 ile tanımlanan iki kriter bulunmaktadır. Yoğunluk kriterinde örnek, Denklem 3.12 ve Denklem 3.13’de görüldüğü üzere FGDT ve GGDT farklarından büyük olup olmadığı kontrol edilir. Uzaklık kriterinde ise örnek, Denklem 3.10 ve Denklem 3.11’de görüldüğü gibi uzaklık verileri olan FDT ve GDT toplamlarından küçük veya eşit olup olmadığına bakılır. Eğer bu kriterler sağlanırsa, incelenen örnek kümeye dahil edilir.

Şekil 3.3.’de görülen A ve B koşulları için veri setindeki herhangi bir rastsal nokta ‘s’ için verilen hesaplamalar aşağıdaki Denklem 3.8 ve Denklem 3.9 gibi yapılmaktadır.

$$A: Gauss(merkez, s) \geq FGDT - GGDT \quad (3.8)$$

$$B: Uzaklık(merkez, s) \leq FDT + GDT \quad (3.9)$$

NSL üyeliği tekrarlanan şekilde çalışmaktadır. Kümeleme işlemi esnasında, ilk geçiş sağlandığında sadece küme merkezini temsil eden örnek kümeye dahil edilerek NSL’e eklenir. Sonraki döngüde, merkeze ait komşuluklar taranarak bulunan kriterleri sağlayan örnekler NSL’ eklenir ve küme güncellenir. Eklenen örnekler bir sonraki aşamada her biri sanki birer merkezmiş gibi davranarak kriterlere uyan örnek kalmayana kadar tekrarlanır. Bütün örnekler kümелendiğinde algoritma otomatik olarak sonlandırılır.

A ve B kriterlerinden kesin sonuç alınmaktadır. Bu şekilde minimum ve maksimum, uzaklık ve yoğunluk değerleriyle; küme için örnekler arasında benzerlik ilişkisi oluşturulur. A ve B kriterlerinde olan denklemler ise:

Sabit uzaklık eşiği FDT:

$$FDT = \left| DPM \frac{(merkez)}{\left(\ln(GPM(merkez) * N + DPD(merkez)) \cdot \ln(GPD(merkez)) \right)} \right| \quad (3.10)$$

Her kümeleme merkezi, çevre komşulukları belirleyerek örneklerin katılımını veya çıkarımını denetler. Bu küme merkezine bağlı dinamik olarak oluşturulan eşik değeri Denklem 3.10'da görülebilir. Bu denklem, veri setine bağlı olarak noktalar arası uzaklık ve yoğunluk oranlarıyla belirlenir. Bu değerler farklı veri setlerinde test edilerek denklemlerin performansları ölçülmüştür.

Kademeli uzaklık eşik değeri GDT:

$$GDT = \mu(SPLsapma) \cdot \frac{AG(NSL)}{AG(merkez)} + 2 \cdot \pi \cdot \sigma(SPLsapma) \cdot \frac{\mu(GPD)}{GPD(NSL)} \quad (3.11)$$

Denklem 3.11'te $\mu(GPD)$, bütün GPD değerlerinin veri uzayındaki ortalaması alınarak oluşturulur. Kademeli uzaklık eşiği, küme merkezine bağlı olarak incelenen, örnekler küme merkezinden uzaklaştığında önem kazanan, kümelemenin uzaklık kriterinin bir parçasıdır. Her komşu noktası küme merkezi gibi davranarak, çevresel değişkenlerin değişimine göre kümenin bağıllık kriterlerini de değiştirmesi gerekmektedir. Bu yakınsama ve açılma hesaplamaları göreceli olarak GDT değeri ile belirlenir. Eğer veri setindeki incelenen örnekler arası çok açılmış ve dağılmışsa, kümeye ait olmayan örnekler bu kriter sayesinde küme dışında bırakılır.

Sabit Gauss yoğunluk eşiği FGDT:

$$FGDT = \frac{GPM(merkez)}{\pi \cdot \sqrt{GPD(merkez)}} - \pi \cdot GPD(merkez) \quad (3.12)$$

Kademeli Gauss yoğunluk eşiği GGDТ:

$$GGDT = \sigma(SPLvaryans) \quad (3.13)$$

Denklem 3.12 ve Denklem 3.13'te uzaklık verilerine benzer olarak Gauss yoğunluk değerlerine göre eşikler belirlenmektedir. GDD kümeleme algoritması Matlab programlama dili kullanılarak geliştirilmiştir [105].

3.1.3. Kümeleme sistemi deney sonuçları

GDD kümeleme algoritma testleri, tez kapsamında yayınlanan makalenin deneysel kurulum, sonuç ve tartışma kısımlarından incelenebilir [23]. Ayrıca GDD kümeleme yöntemi, Matlab ortamında kod olarak istenildiği takdirde Mathworks'un web sitesinden indirilerek test edilebilir [106].

3.2. GDD kümeleme yöntemiyle renk segmentasyonu

Tez kapsamında, GDD kümeleme yöntemi kullanılarak renk bölgelerinin ayrımını sağlayan renk segmentasyonu algoritması geliştirilmiştir. Bu kapsamda renk segmentasyonunda tez için önem arz eden temel alanlar; derinlik haritası hata azaltma, v-disparity ile belirlenen yer düzlemindeki renksel farklılık ve bölgelerin belirlenmesi, derinlik haritasından kaynaklı engel ve yer düzlemi hatalarının tespiti, yaya tanımada kullanılacak engel alanlarının ayrıştırılması ve renk tabanlı işlem öncelik sırasının oluşturulmasını kapsar. Derinlik haritası çıkarımı yöntemlerinin oluşturulması, çalışma içerisinde yer almadığından dolayı şu aşamada GDD renk segmentasyonu bu alana uygulanmamıştır. Bununla birlikte renk segmentasyonu, yer düzlemi bölgelerinin belirlenmesi ve renksel özelliklere göre olası yaya

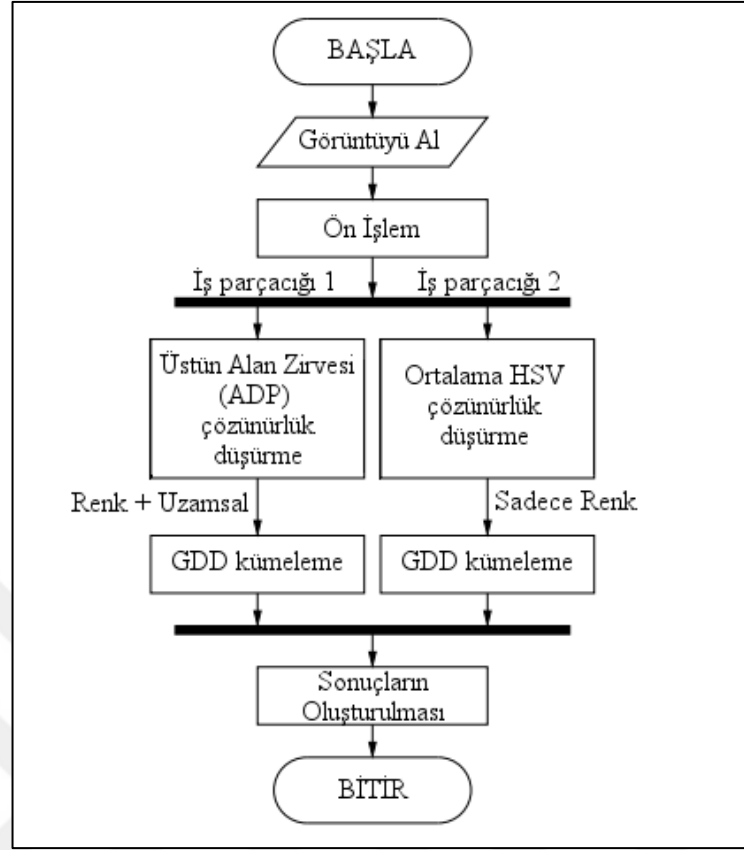
barındırabilecek bölgelerin öncelik listesine dahil edilmesinde kullanılmıştır. Renk segmentasyonu algoritmalarında birçok metot parametre girişi istediğinden çalışmada öne sürülen ve oluşturulan GDD renk segmentasyon algoritması, sürüş destek sistemleri ile otonom sistemler için avantaj sağlamaktadır. Ayrıca renk segmentasyon algoritması kullanımı sayesinde araçlarda yol ve çevre modellemesiyle birlikte yayaların tespiti konusunda da sistem başarımının artırılması hedeflenmiştir. İlave olarak Bölüm 3.3.'te anlatılan yer düzlemi ile engellerin tespiti sonrasında doğrulama işlemi de yine GDD renk segmentasyonu ile sağlanabilir.

GDD renk segmentasyonu ile görüntü bölgelerinin ayrımı işleminde, hem piksellerdeki renk bilgilerinin yakınlığından kaynaklanan hataları azaltmak hem de veri sayısını azaltarak işlem hızını artırmak için çözünürlük düşürme işlemleri uygulanmıştır. Tez kapsamında, yaygın olarak kullanılan kümeleme bazlı segmentasyon metotları ile öne sürülen GDD kümeleme tabanlı renk segmentasyonu karşılaştırılarak başarımları test edilmiştir.

Kaynak araştırması bölümünde bahsedilen segmentasyon yöntemlerinden K-means, Meanshift ve GDD renk segmentasyonlarının sonuçları Bölüm 3.2.5.'te karşılaştırılmıştır. K-means ile Meanshift yöntemlerinde sadece renk uzayı kullanımı ve renk uzayıyla birlikte uzamsal verilerin de kullanıldığı iki farklı segmentasyon girdisi sonuçları incelenmiştir. Meanshift yönteminde de yoğunluk bilgisi kullanıldığından GDD renk segmentasyonu ile farkını incelemek gerekirse, GDD renk segmentasyon işlemi Meanshift'ten farklı olarak yoğunlukların zirve noktasını değil, verilen girdilerin yoğunluk ve uzaklıklarının oranlarına göre kümeleme işlemini tamamlamaktadır. Meanshift'te zirve tahmini yapılırken algoritma her çalışmada farklı sonuç verebilmektedir. Zirve tahmini ile elde edilen küme merkezi, verilen veri uzayı elemanlarından oluşmadığından, oluşturulan kümelerde minimize edilen uzaklıklar dolayısıyla segmentasyonda her uygulama koşulunda farklı sonuçlar elde edebilmektedir. Bu sebeple her farklı koşulda Meanshift farklı sonuçlar verirken GDD kümeleme yöntemi istikrarlı bir biçimde aynı sonuçları çıktı olarak vermektedir.

Tez çalışması kapsamında, parametresiz olarak farklı koşullarda istikrarlı sonuç veren bir renk segmentasyon algoritması geliştirilmiştir. Genel akış diyagramı Şekil 3.4.'te görüldüğü gibidir. Renk–Doyum–Değer (HSV) renk uzayı modeli hesaplamalarda tercih edilmiştir. Bununla birlikte diğer renk uzaylarının da başarıma etkisinin incelenmesi kapsam dışında tutulmuştur. HSV renk uzayı, renklerin birbirinden ayrımını net olarak yapılabilmesi, kartezyen uzayında temsil edilebilmesi ve renk bilgisinin az olduğu gri seviyeye ait bölgelerin merkezde toplanmasından dolayı tercih sebebi olmuştur. Fakat renk özelliğinin az olduğu merkezde oluşan toplanma, hataların oluşmasının en önemli sebebidir. HSV renk uzayı kullanıldığından dolayı ilk görüntü elde edildikten sonra RGB renk bilgileri, HSV renk uzayına ön işlem safhasında dönüştürülmektedir.

Dahili bir özellik olarak GDD renk segmentasyonunda, görüntülerin çözünürlükleri düşürülmektedir. Renk bölgesi ayrımında çözünürlüklerin düşürülerek belirli bir alan dahilinde renklerin ayırt edilme işlemi, gözde renk algılayan sinirler ile ışığı algılayan sinirlerin oranı yani gözdeki çubuk ve koni oranlarından ilham alınarak uygulanmıştır. Renksel algı sağlayan sinirlerin sayısı, ışık bilgisini algılayarak beyne iletim sağlayan sinirlerden sayıca az olduğu bilinmektedir [107]. Bu sebeple renk özelliklerini gösteren bölgelerin ayrımının daha iyi yapılabilmesi için çözünürlük düşürme yöntemleri kullanılmıştır. En-boy oranının bir olduğu durumlarda segmentasyon başarımları ve hız açısından, 1200 örnekleme ile bölgelerin otomatik bulunmasının en iyi sonucu verdiği gözlemlenmiş ve bu şekilde testler uygulanmıştır. En-boy oranının farklılaştığı koşullarda adaptif olarak bu sayı ve bölgeler otomatik olarak hesaplanmaktadır. Sonuç olarak, farklı en-boy oranlarına bağlı olan çözünürlük düşürme yöntemi, parametre girişi istenmeden hesaplaması gerçekleştirilmektedir.



Şekil 3.4. GDD renk segmentasyonu akış diyagramı.

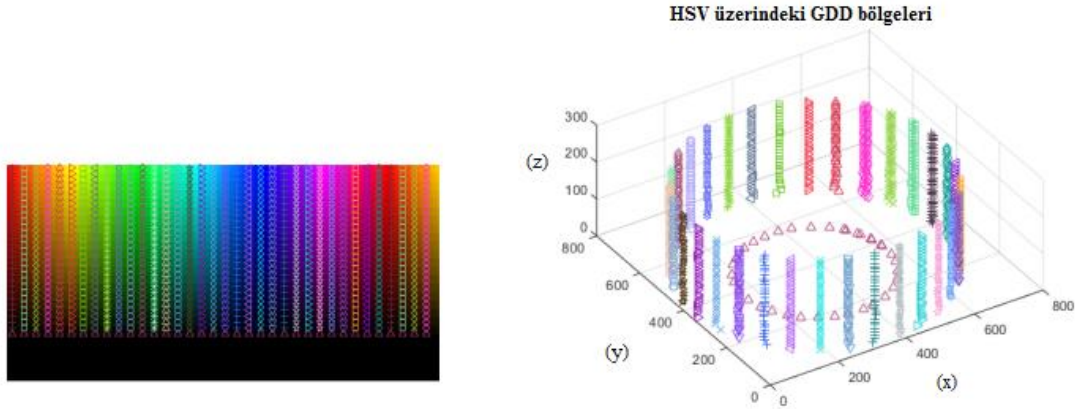
3.2.1. Ön işleme ve HSV renk uzayı dönüşümü

Ön işlem olarak; gürültünün giderilmesi, renk uzayı dönüşümleri, silindirik koordinat sisteminden pozitif değerli kartezyen koordinatlarına geçiş gibi işlemler yapıldığı gibi çözünürlük düşürme öncesinde yapılan görüntünün bloklara ayrımı işlemini de bu gruba dahil edebiliriz.

Elde edilen görüntüler, günümüzde dijital kamera tarafından dijital ortama genellikle RGB renk uzayında ‘png’ veya ‘jpg’ formatlarında kaydedilmektedir. Renk uzayının bölgelere ayırma işlemlerinde daha iyi ifadesi için RGB renk uzayından HSV renk uzayına dönüşüm gerçekleştirilmiştir [108]. Dönüşüm sonrası normalize edilmiş HSV renk, doyum ve aydınlık bilgileri elde edilmiştir. Bu değerlerin her birini $[H_n, S_n, V_n]$ şeklinde gösterirsek silindirik koordinat düzleminde $[0 - 1]$ aralığında değerler elde etmiş oluruz. Bu değerleri doğru bir biçimde uzaklık ölçümünde

kullanabilmek ve görsel olarak inceleyebilmek için silindirik koordinat sisteminden kartezyen koordinat sistemine geçiş yapılmıştır. Bu geçişte sırasıyla renk ve doyum değerleri için açısal olarak derece hassasiyetine mutabık gelecek şekilde ‘360’ ile değerler genişletilmiştir. Uzamsal $[x, y]$ koordinatlarında bütün değerlerin pozitif eksenlerde görülmesi için de ‘360’ birim kaydırma işlemi yapılmıştır. Aydınlık değerler için 8-bitlik değerlere denk gelecek şekilde veriler $[0 - 255]$ değerleri arasına Denklem 3.14’teki şekilde hesaplanarak atanmıştır. GDD kümelemenin negatif olmayan değerlerle çalışmasından dolayı, değerler kartezyen koordinat sisteminde pozitif alanda tutulmaktadır. Bu şekilde görüntü üzerinde herhangi bir pozisyonda bulunan pikselin HSV renk uzayı karşılığı Denklem 3.14’te görüldüğü üzere $P_x(H, S, V)$ olacaktır.

$$P_x(H, S, V) = \left[360 + \left(S_n \cdot \frac{720}{2} \right) \cdot \cos(H_n \cdot 360), \right. \\ \left. 360 + \left(S_n \cdot \frac{720}{2} \right) \cdot \sin(H_n \cdot 360), \right. \\ \left. V_n \cdot 255 \right] \quad (3.14)$$



Şekil 3.5. GDD kümeleme ile kümelenmiş renk örnekleri; HSV renk uzayı sınır belirleme görüntüsü (sol) ve pozitif kartezyen koordinatlarında temsil edilen HSV renk uzayı karşılığı (sağ).

Şekil 3.5.’te test görüntüsünden alınan verilerin, GDD ile kümelenmiş ve şekillerle belirtilen noktaların, HSV uzayındaki kartezyen dönüşümünden sonraki karşılıkları gösterilmiştir. RGB test görüntüsü HSV uzayının sınır noktalarını belirtecek şekilde en yüksek doyum değerlerinde seçilmiş ve renk değişimi bütün renk skalasını

kapsayacak şekilde oluşturulmuştur. Renk ile doyum değerlerine göre her nokta birer dereceyi kapsayacak şekilde $[0-360]$, aydınlık değerleri de $[0-255]$ aralığında maksimum doyum değerleri seçilerek test görüntüsü oluşturulmuş ve HSV renk uzayı karşılıkları Şekil 3.5. üzerinde belirtilmiştir.

3.2.2. Çözünürlük işlemleri için blok oluşturma

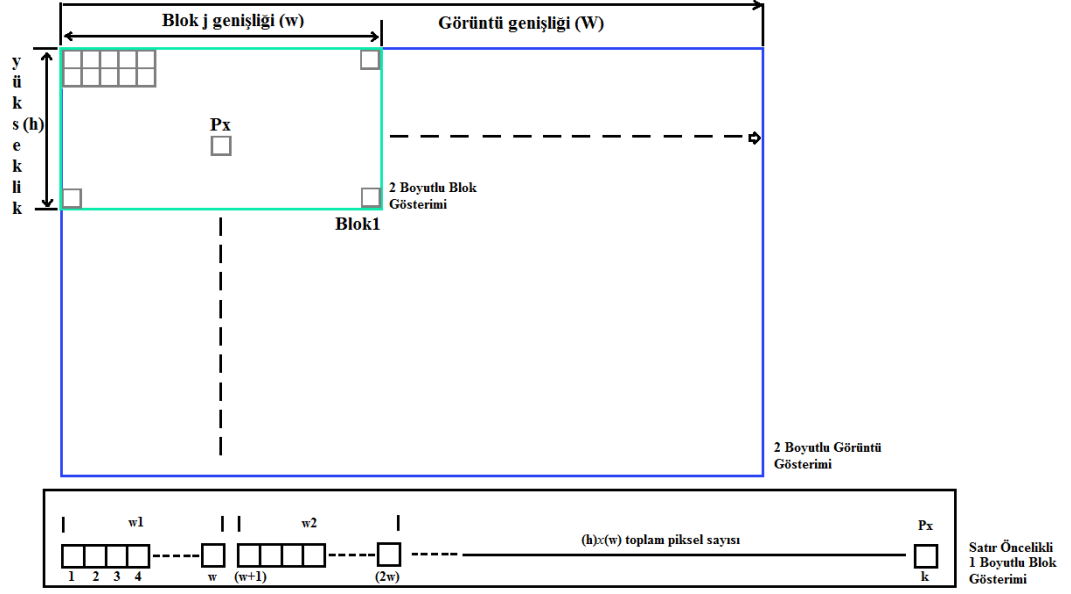
Görüntü üzerindeki bloklar, akış diyagramı Şekil 3.4.'te görülebileceği gibi iki paralel iş parçacığı üzerinde iki farklı çözünürlük düşürme metodunu gerçekleştirmek için ön işlem safhasında oluşturulmaktadır. Her blok, çözünürlük düşürme işlemi uygulandıktan sonra düşük çözünürlüklü görüntüde bir pikseli temsil edecek şekilde tanımlanır. Yüksek çözünürlüklü görüntülerde, piksel komşulukları arasındaki renk değişim değerleri birbirine çok yakın olduğundan ve nesnelerin renkleri ani değişimlerden ziyade pikseller içinde birbirine yakın değerlerle geçiş sağlandığından dolayı sayısal olarak ayırım işleminde zorluklar ve hatalar meydana gelmektedir. GDD kümeleme metodunda benzer özellik gösteren örnekler aynı kümede gösterildiğinden dolayı, bölge ayırma işlemlerinde renk farklılıklarının daha net ortaya çıkması için çözünürlük düşürme işlemleri uygulanmıştır. Bu sebeple görüntüde, doğal renk segmentlerinin düşük çözünürlükte renk özelliklerini koruyabilmesi için paralel olarak iki iş parçacığının ilkinde alansal üstünlük sağlayan tepe değerler ADP (Area Dominance Peak) ile tanımlanmıştır. Şekil 3.4.'te görülen ikinci iş parçacığında ise düşük çözünürlük oluşturmak için ortalama renk değerleri bulunarak, geçişler arasında daha genel bir renk ayırımı yapılması ve büyük renk değişimine maruz kalan alanların tespiti sağlanmıştır.

Çözünürlük düşürmede, pikseli temsil edecek blokların toplamı kare görüntü oluşturacak şekilde ele alınmıştır. Görüntü çözünürlüğünde yükseklik ve genişlik aynı olduğu takdirde, dahili blok sayısı yaklaşık 1200 olacak belirlenmiştir. Segmentasyon işleminde, dahili blok sayısı 1200 parça olarak alındığı zaman optimum sonuç verdiği gözlemlenmiş ve bu sebeple 1200 pikselden büyük görüntülerde çözünürlük düşürme işlemi uygulanmıştır. 1200 blok sayısı, yapılan testlerde uygun sonuçlar üreten dahili bir değer olmakla birlikte istenirse parametre

olarak da uygulama içinde harici olarak alınabilir. Fakat yüksek çözünürlükteki görüntüde, pikseller arası renk değişimleri az olduğundan dolayı çözünürlük düşürmek için belirlenen örnek sayısının artırılması aşırı segmentasyona sebep olacağı gibi azaltılması da bazı renk bölgelerinin doğru ifade edilememesine neden olacaktır.

Orijinal görüntünün kullanıldığı durumlarda dikkat edilirse, ilişkili yakın benzerlikteki bölgelerin tek küme içerisinde yer alma özelliğinden kaynaklı aşırı segmentasyon (over-segmentation) durumu ortaya çıkarak istenilenden daha büyük ve geniş bölgeler elde edilmektedir. Bu sebeplerden kaynaklı piksellerin ön işlem olmadan kümelemeye gönderilmesi hem zamansal çalışma açısından hem de hatalı sonuçların artmasından dolayı önerilmemektedir. Çözünürlük düşürme işlemi dışında, kenar tanımlamayla kenarlar belirlenerek bu bölgelerin paralel olarak işlenmesi önerilir.

Birinci iş parçasığında çalışan ADP çözünürlük düşürme metodu, görüntüde oluşan renk kaybını en aza indirmeye çalışmaktadır. MeanHSV metodunda ise blok içerisindeki HSV renk bilgisi ortalamasının alınmasıyla ikinci iş parçasığı sonucunda büyük ve yoğunluklu renk bölgelerinin ortaya çıkması sağlanmıştır. İkinci iş parçasığında, kümeleme giriş verisinde uzamsal bilgi bulunmadığından sadece renk uzayına bağımlı değişimler gözlenirken birinci iş parçasığında ise renk bilgileriyle birlikte uzamsal bilgiler GDD kümelemeye gönderilir. Bu iki farklı yöntem çıktılarının kesişimlerine bakılarak sentezlenmesi sonucu renk segmentasyonu bilgisi elde edilmektedir.



Şekil 3.6. Görüntüde piksel ve bloklar için 2-boyuttan 1-boyuta dönüşüm işlemi.

Şekil 3.6.'da görüldüğü gibi görüntü ve bloklardaki iki boyutlu piksel matrisi, blok işlemlerinden sonra kod içerisinde gerekli görüldüğü durumlarda bir boyutlu şekilde ifade edilmiştir. Bunun sebebi, blok işlemlerinde elde edilen blok özelliklerinin bir vektör olarak kullanılması ve dönüşüm işlemleri sırasında matris ile özellik vektörü indislerinin korunmasıdır. Örneğin veri kümeleme işlemine gönderildiğinde blok özellik vektörü tek boyutta ifade edildiği takdirde, renk uzayı verileri her bir nokta için piksel veya blok halinde işlenebilmektedir. ADP çözünürlük düşürme yönteminde; indeks değeri ' i ' ile $[x, y]$ uzamsal ilişkisi ve tek boyutlu vektör olarak kullanımı denklemler üzerinden incelenebilir.

3.2.3. Mean-HSV ve ADP renk çözünürlüğü düşürme yaklaşımları

Mean-HSV çözünürlük düşürme işlemi kısaca, blok içerisindeki piksellerin ortalama değerini alarak o blok içerisinde bu değer düşük çözünürlük renk değeri olarak temsil edildiği bir yöntemdir. Bu yöntem genel olarak renk bölgelerinin bütün görüntü ve blok görüntüleri üzerindeki tahmini için kullanılmıştır.

ADP değeri, çözünürlük düşürürken renk bilgisinin kaybını azaltmak ve çözünürlüğü düşürülecek alandaki en baskın rengin temsil edilebilmesi için ortaya konulan bir

özellik çıkarım ve çözünürlük düşürme yöntemidir. ADP algoritması çözünürlük düşürürken, şeklin değil pikseli temsil edecek baskın rengin korunmasını ön planda tutar. Seçilen temsili pikseldeki renk verisini bloktaki piksel verilerine göre belirlerken bölgedeki en baskın rengi bulabilmek için yapılan istatistiksel hesaplamalar denklemleriyle birlikte aşağıda belirtilmiştir.

Görüntü blokları içindeki her bir pikseli HSV renk uzayında $\vec{P}_x(H, S, V)$, uzamsal uzayda ise $P_x(x, y)$ şeklinde tanımlarsak. Çözünürlüğü $1/8$ oranında düşürülerek tekrar eski çözünürlüğüne getirilmiş yani yeniden ölçeklendirilmiş blok piksellerini ise $\vec{P}_{xre}(H, S, V)$ ile tanımlarsak değişim vektörü $\vec{C}_p$ Denklem 3.15'deki gibi olacaktır.

$$\vec{C}_p = \vec{P}_x(H, S, V) - \vec{P}_{xre}(H, S, V) \quad (3.15)$$

Denklem 3.15'de $\vec{P}_x(H, S, V)$ ve $\vec{P}_{xre}(H, S, V)$ verileri uzamsal ortamdaki piksellerin orijinal ve yeniden ölçeklenmiş hallerini temsil etmektedir. Burada $\vec{C}_p$ renk farklarının uzamsal düzlemdeki dağılımı, GMM (Gaussian Mixture Model) dağılımı ile Denklem 3.16'daki gibi hesaplanarak temsil edilmektedir. Denklem 3.16'da görülebileceği üzere piksellerdeki renk bilgileri, Gauss dağılımlarındaki ağırlıklarla yayılmakta birlikte sonrasında toplanarak bloktaki renk yoğunluğunun oluşturulması amacıyla yoğunluk dağılım modeli oluşturulmaktadır.

$$f_{sum}(\vec{P}_x(H, S, V) | \omega, \mu, s) = \omega_1 \cdot f_m(P_{x1}(x, y) | \mu, s) + \omega_2 \cdot f_m(P_{x2}(x, y) | \mu, s) + \dots + \omega_k \cdot f_m(P_{xk}(x, y) | \mu, s), \quad s = \sqrt{(1/k)\sigma^2} \quad (3.16)$$

Denklem 3.16'daki s değeri örnek veri uzayının ortalamasının standart hata (standard error of the mean) değeridir. Yani değerler arası sapma, genel standart sapmaya göre $\sqrt{(1/k)}$ derece tahmini oranda değişim göstermektedir. Burada k , blok içerisindeki piksel sayısını temsil ederken, σ^2 verilerdeki varyansı vermektedir. Ağırlık ω_i değerleri, $\vec{C}_p$ vektörü büyüklüğüne ters oranda etkileyen biçimde Denklem 3.17'deki

şekilde hesaplanmaktadır. Denklem 3.17’de orantı-uzayı işlemi yapılarak maksimum değişimler minimum ağırlıklarla, minimum değişimlerse maksimum ağırlıklarla temsil edilmektedir.

$$\omega_i = 1 - \frac{\|\vec{C}_p(i)\|}{\sum_{j=1}^k \|\vec{C}_p(i)\|}, \quad \sum_{i=1}^k \omega_i = 1 \quad (3.17)$$

Denklem 3.17’deki ‘ k ’, bloktaki toplam piksel sayısını temsil ederken, ‘ i ’ değeri ise Şekil 3.6.’da verilen örnekteki gibi indis değerlerini ifade etmektedir. Her bloktaki uzamsal bağlamda Gauss dağılımı ‘ f_m ’ Denklem 3.18’deki gibi tanımlanabilir.

$$f_m(\vec{P}_x(x, y)|\mu, s) = A. \exp\left(-\left(\frac{(x - x_0)^2}{2(\sqrt{\mu_x \cdot s_x}/2\pi)^2} - \frac{(y - y_0)^2}{2(\sqrt{\mu_x \cdot s_x}/2\pi)^2}\right)\right) \quad (3.18)$$

Denklem 3.18’de tanımlanan olasılıksal dağılımda ‘ A ’ değeri, ‘ f_m ’ fonksiyonunun integralinin bir değerine eşit olacağı şekilde hesaplanabilir. Denklem 3.20 maksimum argümanı kullandığından dolayı ‘ A ’ katsayısı, bütün ‘ f_m ’ fonksiyonlarında aynı olduğundan hesaplanmasına gerek duyulmamıştır. Uzamsal uzayda $[x_0, y_0]$ noktaları, incelenen pikseller için referans noktası olarak kullanılmaktadır. Denklem 3.16’yı toplam sembolüyle kısaca, Denklem 3.19’daki şekilde ifade edebiliriz.

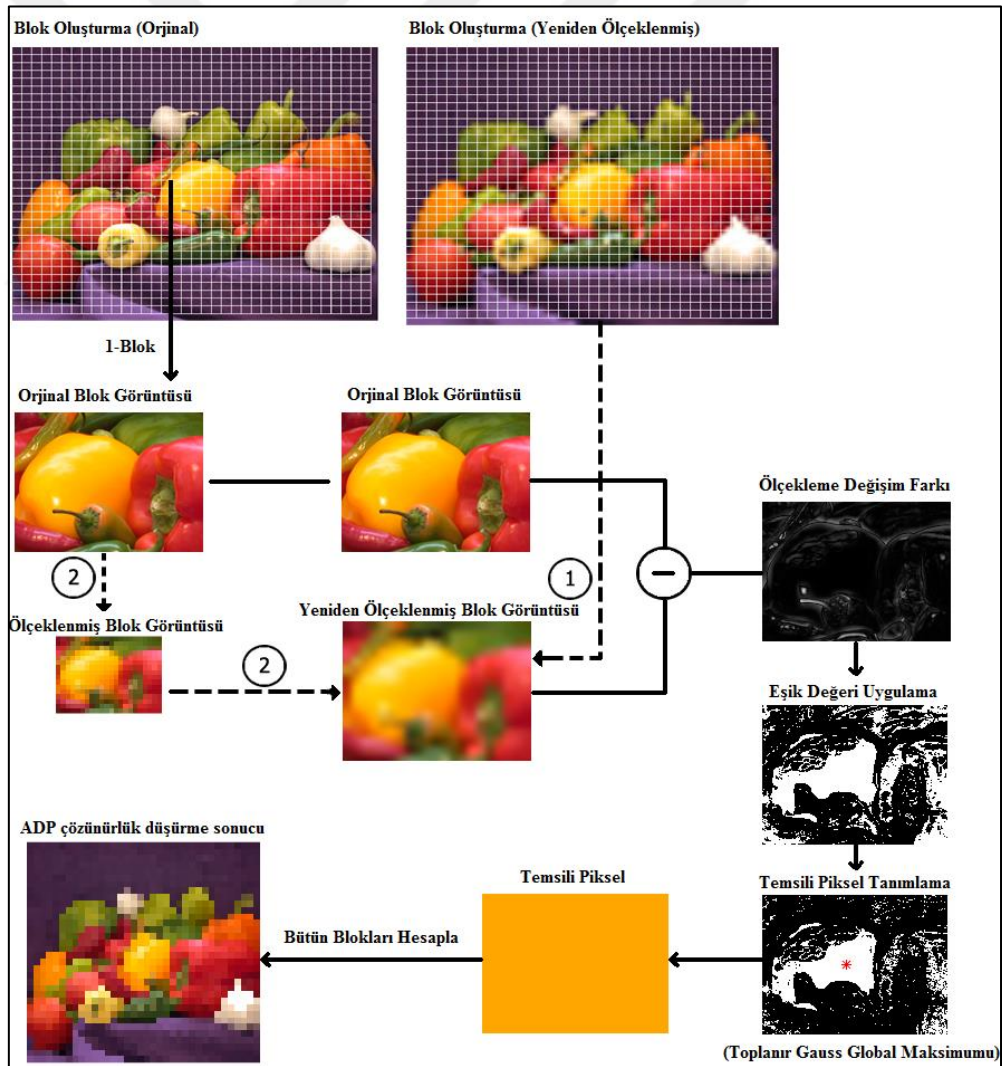
$$f_{sum}(\vec{P}_x(H, S, V)|\omega, \mu, s) = \sum_{i=1}^k \omega_i f_m(\vec{P}_x(x, y)|\mu, s) \quad (3.19)$$

Denklem 3.19’dan elde edilen dağılım verileri, Denklem 3.20’de kullanılarak o piksel bloğunu temsil eden düşük çözünürlükteki piksel rengi olarak tanımlanır. Bu temsili piksel Denklem 3.19’daki olasılıksal dağılımın tepe noktasını temsil etmektedir.

$$[H, S, V]_{blok} = [H, S, V]_{argmax(f_{sum})} \quad (3.20)$$

ADP ile MeanHSV algoritmalarından sonra, bloklarda işlenen ve bulunan piksel verileri; ilk iş parçacığında uzamsal ve renk bilgisiyle birlikte, ikinci iş parçacığında ise sadece renk bilgisi kullanılarak GDD kümelemeye iletilir.

ADP ile çözünürlük düşürme yönteminde, yukarıdaki denklemlerle elde edilen veriler kullanarak aşağıdaki Şekil 3.7.'de görüldüğü gibi çözünürlük düşürme işlemi gerçekleştirilmiş olunur. Burada Şekil 3.7.'de yuvarlak içinde numara '1' ile belirtilen şekilde çözünürlük düşürülüp sonrasında orijinal çözünürlüğe yükseltme işlemi gerçekleştirilerek uygulamada ve tez çalışmasında ADP yönteminde kullanılmıştır. İşlem esnasında belirtilen eşik değeri uygulamasının sebebi, az değişim gösteren bölgelerin elde edilmesinde hız sağlanması içindir.

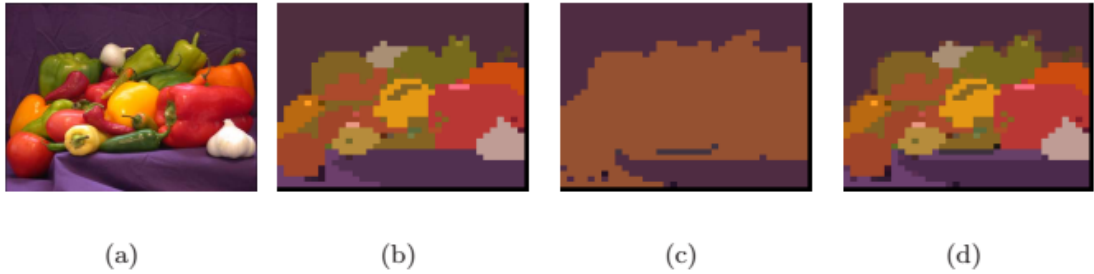


Şekil 3.7. ADP çözünürlük düşürme yöntemi.

3.2.4. Renk bölgelerinin elde edilmesi

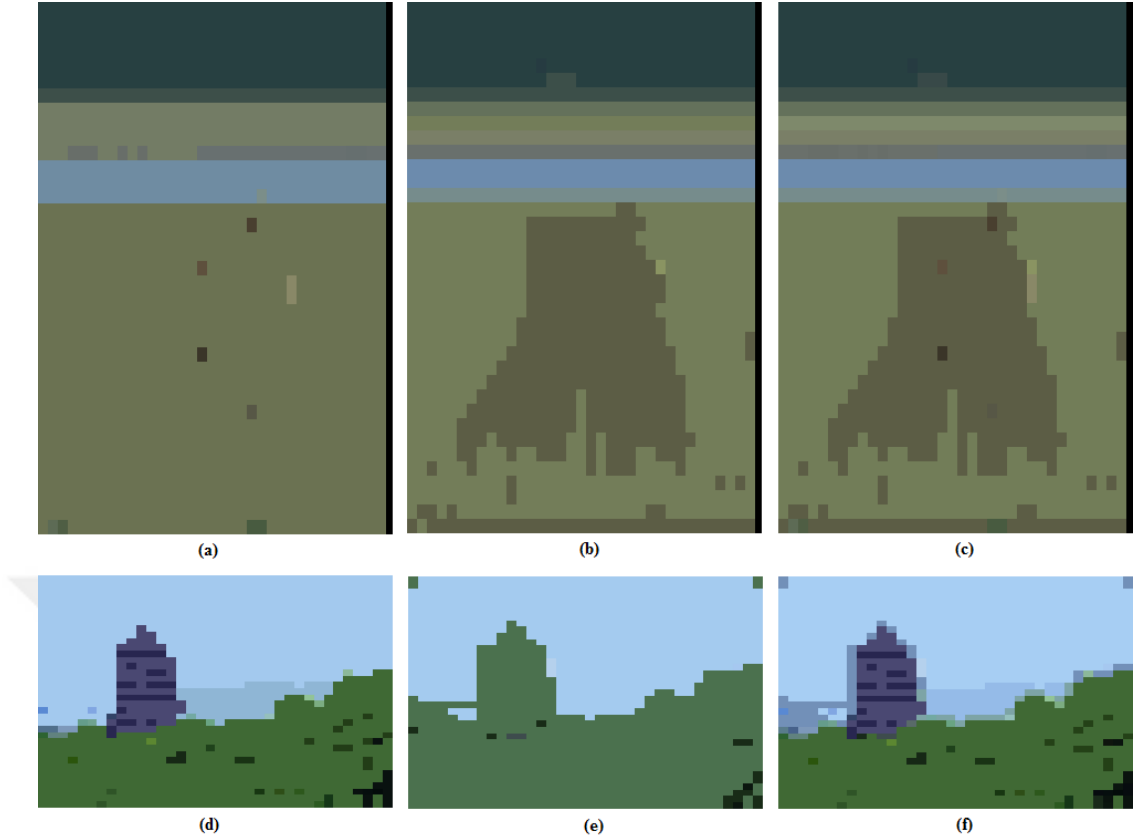
Her iki iş parçacığında, kümeleme işlemleri sonucunda elde edilen segmentler kesişim işleminden geçirilerek sonuç renk ayrımları belirlenir. Paralel farklı iki iş parçacığından elde edilen segment etiketleri aynı olması durumunda güçlü segment, farklı renkleri belirten etiketlerden oluşuyorsa zayıf segment olarak tanımlanır. Genellikle kenar veya spesifik bölgeleri tanımlayan bu segmentler, zayıf olarak adlandırılrsa bile renk uzayının daha iyi tanımlanması için önemlidir.

Renk segmentasyon karşılaştırmalarında sıklıkla kullanılan biberler (peppers) görüntüsü, orijinal olarak '513x383' piksel büyüklüğünde girdi olarak belirlenmiş ve iş parçaları ara çıktı sonuçlarıyla birlikte segmentasyon sonuçları Şekil 3.8.'de verilmiştir. Görüntüden de görülebileceği üzere farklı iş parçacıkları farklı sonuçlar doğurmakla birlikte aşırı segmentasyona uğrayan bölgeler, diğer iş parçacığından elde edilen sonuçlarla iyileştirilerek GDD renk segmentasyonu çıktısı elde edilmektedir.



Şekil 3.8. GDD renk segmentasyonu örneği (513x383 piksel): (a) Orijinal görüntü, (b) iş parçası 1: ADP + GDD kümeleme, (c) İş parçası 2: mean-HSV + GDD kümeleme, (d) İş parçacıklarından gelen görüntü kesişimleriyle elde edilen sonuç görüntüsü

Sonuç segmenti elde edilmesi esnasında Şekil 3.8.'de '(b)' sonucu, en son elde edilen '(d)' verisinden daha iyi görünse de iş parçacıkları farklı durumlarda farklı başarımlar vermektedir. Şekil 3.9.'da görüldüğü gibi farklı görüntülerde iş parçacıklarının performansı değişim göstermektedir. Bunun sebebi görüntülerdeki baskın renkli bölge ve alanların, genel görüntü üzerindeki etkisinin görüntüden görüntüye değişmesidir.



Şekil 3.9. Test görüntüleri üzerinden GDD renk segmentasyonu iş parçacıkları sonuçları ve elde edilen sonuç görüntüsü: (a, d) İş parçası 1: ADP + GDD kümeleme, (b, e) İş parçası 2: mean-HSV + GDD kümeleme, (c, f) İş parçacıklarından gelen görüntü kesişimleriyle elde edilen sonuç görüntüleri.

3.2.5. GDD renk segmentasyonu deney, test ve bulgular

GDD renk segmentasyonu algoritması Matlab programlama diliyle yazılmış ve gerçekleştirilmiştir [105]. 32GB Ram ve Intel i7-2600K 3.40GHz işlemcisi ile çalıştırılarak denenmiştir. Tez kapsamında oluşturulan yeni segmentasyon algoritması, birçok farklı görüntüyle test edilmiş ve “The Berkeley Segmentation Dataset and Benchmark” veri setinden alınan on bir farklı görüntüyle sonuç karşılaştırılmaları bu bölümde incelenmiştir [51]. Genellikle ‘321x481’ piksel çözünürlükte olan görüntülere, GDD renk segmentasyonu optimizasyon yapılmadan uygulandığında sonuçları yaklaşık 8-10 saniye civarında sürmektedir. İlk iş parçasığı genel olarak ikinci iş parçasığından yaklaşık 10 kat daha fazla zaman almaktadır.

Tablo 3.1.'de çok bilinen segmentasyon algoritmalarından K-Means, Meanshift, Normalized-Cut metotları ile GDD renk segmentasyon sonuçları karşılaştırılmıştır [109, 38, 20]. Alireza Asvadi tarafından Mathworks'un internet sitesine yüklenen K-Means, Meanshift, Normalized-Cut algoritmalarının kodları karşılaştırma amacıyla kullanılmıştır [110]. Karşılaştırma için kullanılan kod yazımları ve yüklenmesi sırasıyla Alireza Asvadi, Bryan Feldman and Naotoshi Seo yazarları tarafından yapılmıştır. Karşılaştırma verisi için gerçeklik tablosu ve karşılık gelen segmentleri görüntü veri setinden alınmıştır [51]. K-Means için gereken parametre girişi, gerçeklik tablosundaki segment sayısına göre atanmıştır.

Testlerde yapılan karşılaştırma; Saflık (Purity), F-ölçütü (F-score) ve Normalize Ortak Bilgi (NMI) metrikleri kullanılarak yapılmış ve Tablo 3.1.'de sunulmuştur. Saflık, kümeleme ve segmentasyon algoritmaları için basit ve şeffaf bir değerlendirme yöntemidir [52]. Her segment en yoğun frekansı bulunan sınıfa atanır, sonrasında ise hassasiyet doğru atanmış eleman sayısının toplam eleman sayısına bölünmesiyle Denklem 3.21'deki gibi bulunur.

$$Saflık(\Omega, \mathbb{C}) = \frac{1}{N} \sum_k maksimum_j |\omega_k \cap c_j| \times 100 \quad (3.21)$$

Denklem 3.21'de Saflık verisi, $\Omega = \{\omega_1, \omega_2, \dots, \omega_K\}$ dizisindeki segmentlere ve $\mathbb{C} = \{c_1, c_2, \dots, c_J\}$ dizisindeki sınıflara göre hesaplanmaktadır. Normalde Saflık sıfır ile bir arası değer almakla birlikte sonuçların daha iyi anlaşılması için yüzdelik ölçüm kullanılmış ve değerler bu nedenle yüz ile genişletilmiştir. Saflık, F-ölçütü ve NMI ölçüm yöntemleri hakkında daha detaylı bilgi için Manning ve arkadaşlarının yayınladıkları kitap incelenebilir [52].

Tablo 3.1. Saflık, F-ölçütü ve NMI kullanarak GDD renk segmentasyonu ile çok bilinen segmentasyon yöntemlerinin karşılaştırması.

Saflık (Purity)								
Görüntüler (Veri Tabanı No.)	K-Means (%)		Meanshift (%)		N-Cut (%)	ADP	GDD (%)	
	Renk	Uzam + Renk	Renk	Uzam + Renk			M-HSV	SON
Ayı (100080)	44,7	30,1	58,2	50,2	40,7	69	77,2	76,8
Kabile (101087)	41,8	39,3	56,3	42,3	29,2	56	54,6	54,7
Tören (145086)	27,6	33	50,1	58,5	31,8	75,6	69,6	73,9
Askerler (170057)	14,7	27,7	23,5	44,7	23,5	24	25,2	25,5
Mantar (208001)	21,8	34,2	27,6	50,1	24,5	31,9	23,9	31,7
Vazo (227092)	39,3	32,3	51,9	40,3	38,6	88,5	71,7	87,7
Kayalar (241004)	52,1	54,6	55,2	60,1	17,5	53	58,9	60,7
Zebra (253027)	20,2	18,4	73,2	31,9	70,2	68,3	55,9	55,8
Filler (296059)	21,1	35,4	37,4	54,3	20,5	44,3	43,1	47,8
Piramit (299091)	37,6	52,4	24,6	66,3	24,6	45,1	46,3	46,2
Duvar (74067)	48	58,2	68,2	71,9	31,7	76,3	72,6	71,9
F-ölçütü (measure) ($\beta=1$)								
Görüntüler (Veri Tabanı No.)	K-Means (%)		Meanshift (%)		N-Cut (%)	ADP	GDD (%)	
	Renk	Uzam + Renk	Renk	Uzam + Renk			M-HSV	SON
Ayı (100080)	28,7	14,2	50,1	32,2	28,4	54,8	70,5	73,6
Kabile (101087)	55,5	19,3	48,7	28,3	15,7	56,8	47,1	56,8
Tören (145086)	10,1	12,4	40,6	55,1	18,4	79,2	73	79,4
Askerler (170057)	2,9	9,7	10,4	27,9	10,4	10,1	11,5	11,3
Mantar (208001)	5,1	19,7	12,4	29,5	11,3	14	10	13,9
Vazo (227092)	20	13,9	39,9	24,6	25,9	85,8	62,5	86,2
Kayalar (241004)	45,1	36,1	39,1	64	5,9	36,2	40,9	44,8
Zebra (253027)	8,1	3,6	69,5	13,9	65,9	62,7	40,3	40,2
Filler (296059)	7,2	15,1	18,6	34,7	8,1	24,9	29,9	37,1
Piramit (299091)	19,2	40,4	11,4	60	11,4	28,9	30,8	30,8
Duvar (74067)	31,2	46,9	51,9	62,1	18,2	63,9	59,7	60,5
Normalize Ortak Bilgi (NMI)								
Görüntüler (Veri Tabanı No.)	K-Means (%)		Meanshift (%)		N-Cut (%)	ADP	GDD (%)	
	Renk	Uzam + Renk	Renk	Uzam + Renk			M-HSV	SON
Ayı (100080)	48,3	49,7	35,9	51,3	0	68	67,5	70,2
Kabile (101087)	45,7	57,9	44,2	60,9	54,1	58,2	51,3	58,5
Tören (145086)	50,8	63,2	54,1	67,1	0	72,4	66,5	72,5
Askerler (170057)	2,2	5,4	0	50,1	34,6	7,4	13,6	16,4
Mantar (208001)	28,4	55,8	6,7	61	0	23,5	2,8	23,8
Vazo (227092)	50,8	42,4	34,5	57,4	0	73,5	53,5	73,5
Kayalar (241004)	60,8	69,7	59,4	77,1	0	58,9	61,2	63,4
Zebra (253027)	15,1	26,3	9	24,1	0	1,4	16,5	16,6
Filler (296059)	35,7	61,9	33,4	61,5	0	44,3	45,5	52,5
Piramit (299091)	51,1	64	0	68,4	0	43,3	43,8	43,9
Duvar (74067)	49,7	67,8	58,7	69,9	0	70,2	66,2	67,2

Kayalar (241004) görüntüsünde segmentasyon algoritmaları karşılaştırma sonuçları Şekil 3.10.'da görüldüğü gibidir. Bu test görüntüsündeki en önemli özellik, nesnelerin farklı renk tonlarını barındırması ve farklı bölgelerde benzer renkli alanların bulunmasıdır. Şekil 3.10.'da görüldüğü gibi kayaların genel olarak renkleri, GDD renk segmentasyonu ile daha iyi bir başarımla birbirinden ayrılmaktadır. GDD renk segmentasyonu, benzer aydınlık ve renge sahip alanları birleştirerek benzer değişimleri görüntü boyunca uzamsal olarak da birleştirmektedir. Şekil 3.10. (i)'de ortadaki kayanın sol yanında aşırı segmentasyona uğrayan bölgeler bulunsun da genel görüntü üzerinde kaya nesneleri başarılı bir şekilde birbirinden ayrılmıştır.

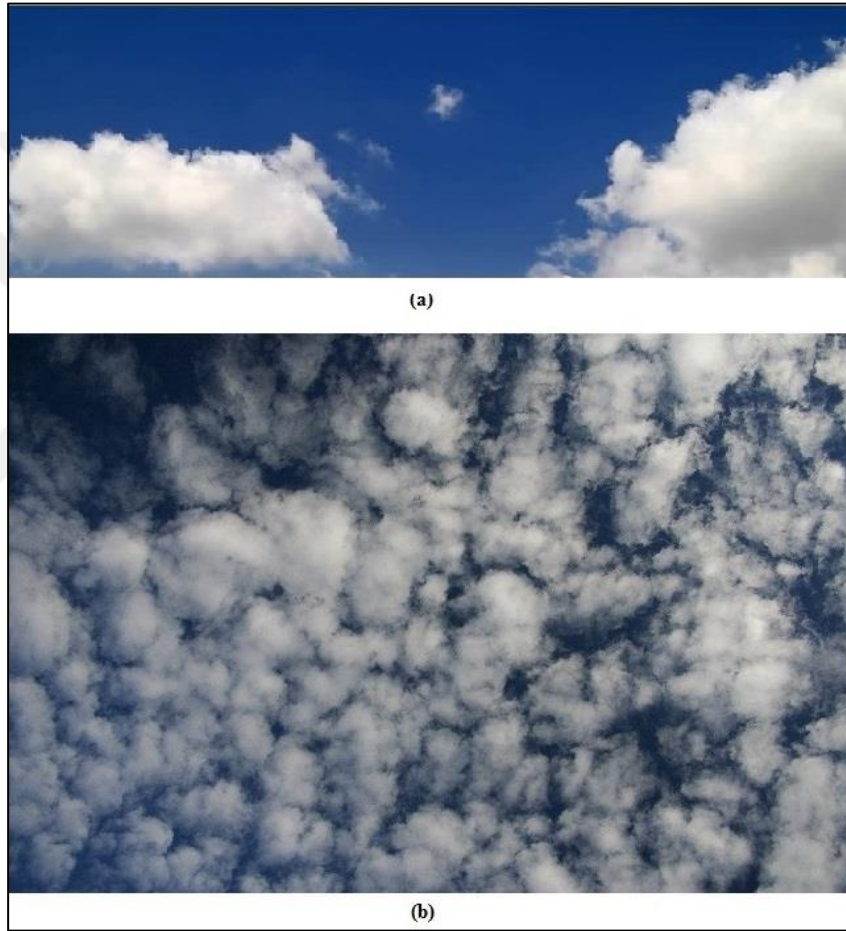
Uzamsal ve renk bilgilerinin kullanıldığı Meanshift segmentasyonunda parçalı segmentlerin elde edildiği Şekil 3.10. (e) üzerinden görülebilir. Burada Meanshift, aydınlık ve gölge değişimlerinde kayaları bölünmüş olarak göstermektedir. Çözünürlük düşürme yöntemiyle birlikte GDD renk segmentasyonu daha yüksek başarımda ve daha hızlı sonuçlar ortaya koymuştur. Kayalar görüntüsünde segmentasyon işlem zamanları; uzamsal ve renk bilgisiyle koşulan Meanshift'te 13,65 saniye, '*ADP+GDD*' (iş parçacığı 1)'de 4,06 saniye, '*MeanHSV+GDD*' (iş parçacığı 2)'de 0,96 saniye, birleşim işleminde 2,43 saniye harcayarak toplam GDD renk segmentasyonu işlem süresi olarak 7,04 saniyede görüntü üzerinde renk bölgelerinin tespit işlemi tamamlanmıştır. Testler sırasında algorithmada hız kazandıracak optimizasyon ve yöntemler belirlenerek Bölüm 4. içerisinde gelecekte yapılması planlanan gelişmeler kapsamında anlatılmıştır.



Şekil 3.10. Taşlar görüntüsü (204x153 piksel) segmentasyon karşılaştırma sonuçları: (a) Orjinal görüntü, (b) K-means renk segmentasyonu, (c) K-means uzamsal segmentasyon, (d) Mean-Shift renk segmentasyonu (e) Mean-Shift uzamsal segmentasyon, (f) Normalized-Cut segmentasyonu, (g) ADP+GDD segmentasyonu, (h) MeanHSV+GDD segmentasyonu, (i) GDD segmentasyon sonucu.

Görüntüdeki renk ve uzamsal piksel bilgisi, gözümüzle bölgeleri ayırt ettiğimiz zaman anlaşılacağı gibi birbirinden bağımsız değildir. Fakat ortamın göreceli ışık, renk yoğunluğu da algılarımızda önemli bir yer arz etmektedir. Bu sebeple sadece renk bilgisi kullanıldığında sonuçlar aşırı-segmentasyon veya eksik bölge bilgilerinin elde edildiği bir durumu ortaya çıkarabilmektedir. Genellikle gerçek hayat görüntülerinde renkler; ışık, doku ve derinlik etkilerinden dolayı eşit bir şekilde yayılım göstermemektedir. Doğal segmentler, görüntü üzerinde birbirlerine yakın bir

şekilde dağılım gösterdikleri zaman, genellikle algılarımızda bu dağılımın tamamını büyük bir segment şeklinde alır ve ifade ederiz. Şekil 3.11. (b)'de Altokümülüs bulutlarını bütün bir şekilde bulutlar grubu olarak adlandırırken, birbirinden ayrık iki farklı bulut parçasını Şekil 3.11. (a)'da farklı olarak inceleyerek sağ-sol gibi yönleri kullanarak 'sağdaki bulut' gibi kavramlarla ifade ederiz. Bu gerçekler göz önüne alındığında, uzamsal bilgileri ele almayan sadece renk bilgisini kapsayan segmentasyon uygulaması, doğal olarak hatalar barındıran bir sonuç verecektir.



Şekil 3.11. Farklı tipteki bulut görüntülerinde göz algısı farklılıkları: (a) İki temel büyük ve bir küçük bulut parçası, (b) Homojen dağılımlı bulutlar (Altokümülüs bulut grubu) [111] [112].

GDD renk segmentasyonunun birleştirici ve toplayıcı yapısından kaynaklı başarımlar sonucu uzamsal birliktelik yapısına göre artmaktadır. ADP ve MeanHSV çözünürlük düşürme yöntemleriyle elde edilen segmentasyon sonuçları, farklı koşullar altında avantaj ve dezavantajları bulunan sonuçlar doğurmaktadır. Renk segmentasyonu

çalışmasında belirtilen bu iki farklı çözüm (MeanHSV+GDD ile ADP+GDD) birlikte ele alınarak oluşturulan segmentasyon sonucunda, bir iş parçacığından elde edilen sonuç diğer iş parçacığından elde edilen hatalı bölgelerin açığını kapayarak daha başarılı doğru bir sonuç elde edilmesini sağlamaktadır. Karşılaştırmalı test sonuçlarını barındıran Şekil 3.12.'deki renk bölgelerinin tanımlanmasında öne sürülen yöntemin diğer segmentasyon algoritmalarına kıyasla daha iyi bir sonuç verdiği görülebilmektedir.

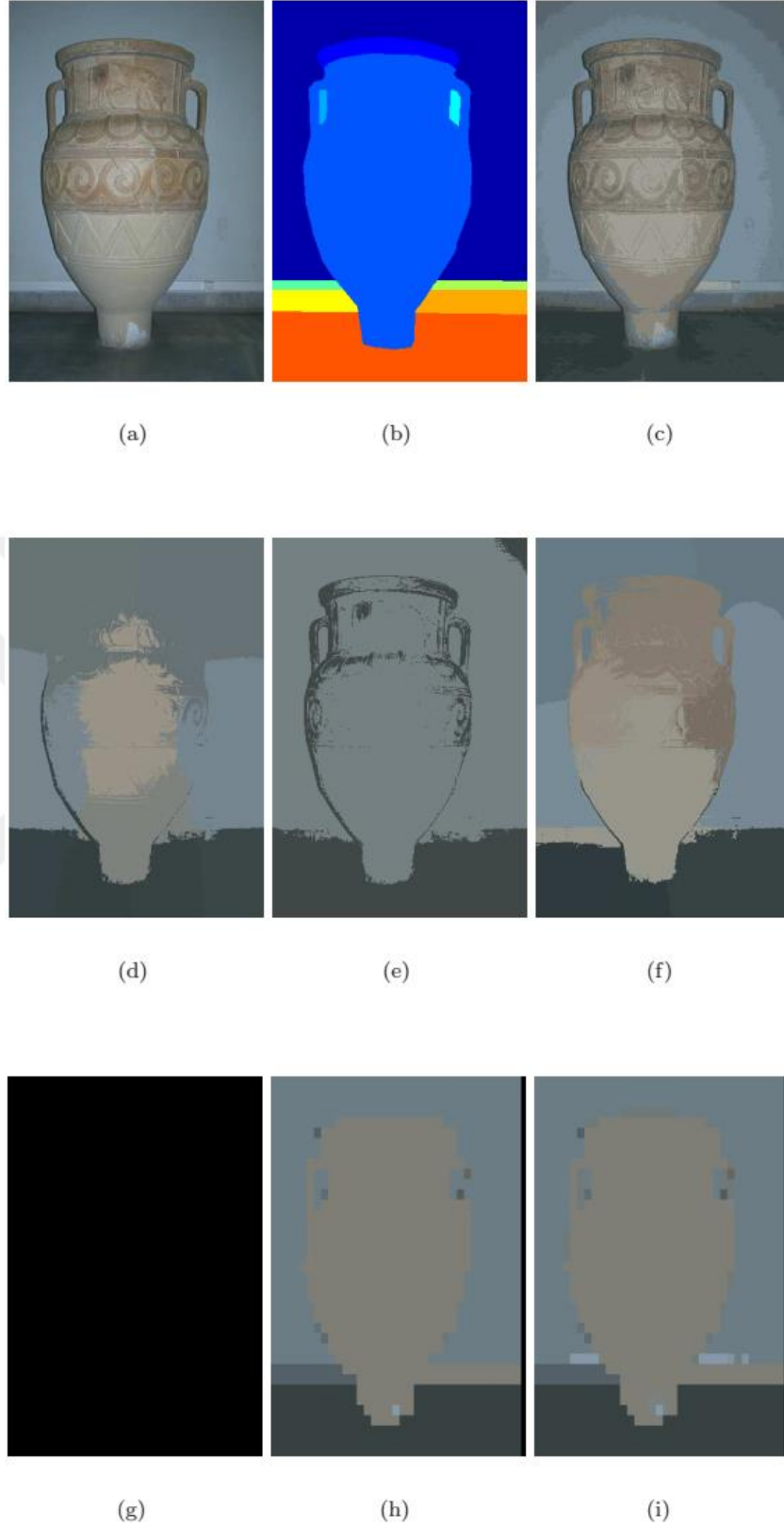
GDD renk segmentasyonunda bağıntısı olduğu sürece yakın renk farklılıkları uzakta olsa bile eğer aşamalı olarak bağlantısı bulunan komşuluklar barındırıyorsa birlikte ele alınarak bir segment oluşturulur. Fakat bu özellikten dolayı bazı piksel ve alanlar incelenen bölgeye dahil edilerek aşırı-segmentasyon sonucunu ortaya çıkarabilmektedir. Bu tarz durumlar için doku bilgisi veya belirgin bölge (salient-region) tespit çalışmaları, başarımlarını artırmak için kullanılabilir.

Dikkat edilmesi gereken bir diğer nokta ise, renk ve uzamsal bilgiler birbirleriyle görüntüde ilişkide olmalarına rağmen bu ilişki lineer bir biçimde değildir. Örneğin bir piksellik gürültü büyük bir alanı kapsayan bir segment içinde fark edilmeyebilir. Ya da birden fazla fakat periyodik olmayan ve uzak aralıklarla bulunan küçük alanlı noktalar da aynı şekilde, etrafını çevreleyen alanlar içerisinde insan gözü tarafından algılanmayabilir. Bu durum birçok segmentasyon algoritması açısından da bölgelerin ayrımı konusunda zorluk oluşturmaktadır.

Çok boyutlu verilerde, bağıntısı bilinmeyen farklı uzay bilgilerinin boyutlarla herhangi bir ilişkisel ağ kurulmadan birleştirilmesi, hataların oluşmasına sebep olmaktadır. Özellikle Öklid uzaklığını ele aldığımızda, iki boyut içerisinde bir özellik cm cinsinden ölçümler gösterirken diğer boyuttaki ölçüm özellikleri cm yerine km karşılığı yer alırsa, ölçüm değerleri doğru olsa bile uzaklıktaki yakınsama vektörel olarak büyük değerlerle ifade edilen cm cinsinden ifade edilen boyut tarafına kayacaktır. Ayrıca eşit ölçekteki bu değerlerin yoğunluk verileri de uzaklık değerlerinden dolayı hatalı bir şekilde dağılım gösterme durumunu ortaya çıkarabilmektedir. Özellikle vektör büyüklüğü ve açısı büyük değişim gösterecektir.

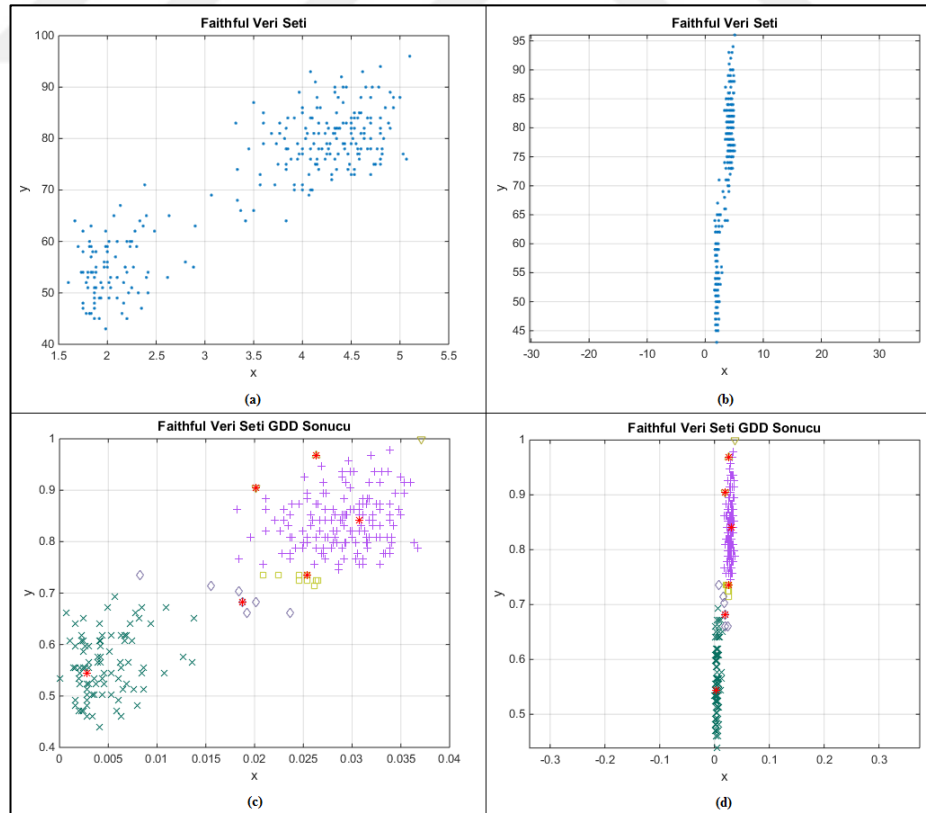
Bu sebeple bu tarz çok boyutlu verilerde, boyutlar ve uzaylar (koordinat ve renk uzayı gibi) arası ilişkiler incelenerek sonuçların daha doğru şekilde ifade edilmesi sağlanmalıdır. Özellikle HSV renk uzayında düşük doyum ve aydınlık değerlerini barındıran pikseller, lineer olmayan renk uzayı yapısından kaynaklı segmentasyon sonucunda hatalar oluşturabilmektedir.





Şekil 3.12. Vazo görüntüsü segmentasyon sonuçları (321x481 piksel): (a) Orijinal görüntü, (b) Gerçeklik tablosu, (c) K-means renk segmentasyonu (d) K-means renk ve uzamsal segmentasyon, (e) MeanShift renk segmentasyonu, (f) MeanShift renk ve uzamsal segmentasyon, (g) Normalized-cut segmentasyonu, (h) ADP+GDD segmentasyonu, (i) GDD sonuç renk segmentasyonu.

Bir verinin farklı boyutlarında farklı ölçeklerle gösterimi hakkında eski faithful veri setini inceleyebiliriz [113]. Şekil 3.13.'de görülen faithful veri seti Matlab ortamında '(a)' grafiğinde standart matlab komutuyla çizdirilmiş '(b)' grafiğinde ise eksenler eşit büyüklükte gösterilecek şekilde karşılığı çizdirilmiştir. Verilerde herhangi bir değişiklik yapılmamıştır. Bu sebeple '(c)' ve '(d)' çizimlerinde yer alan GDD kümeleme sonuçları değişmemiştir. Fakat verilerin 'x' eksenini farklı sınır bölgelerine ([1,5:5,5] ile [-30:30] sınır aralıklarına) göre gösterimi, insan gözü tarafından algılanan nokta ve kümelerinin sonucunu değiştirmektedir. '(a)' grafiğinde iki küme olduğu izlenimi oluştururken '(b)' görüntüsü tek küme izlenimi vermektedir. Kısacası boyutların ifade biçimi ve incelenen sınır değer bölgeleri, kümeleri nasıl algıladığımızı etkilemektedir. Bu sebeple verileri incelediğimiz her boyutun ifadesi ve birbirleriyle ilişkisinin doğru tanımlanması, sonucun başarısını büyük ölçüde etkilemektedir. GDD kümelemede Şekil 3.13.'de 'x' eksenini sınırları, '(b)'deki şekilde alındığında eliptik Gauss değerlerinin 'x' eksenini üzerinde yayılımı artacağından kümeleme sonucu tek küme olacak şekilde sonuçlanacaktır.



Şekil 3.13. Eski faithful gayzer püskürme veri seti: (a) Ölçeklenmemiş ifadesi (b) Matlab ortamında eşit eksenler arası uzaklığın eşit olarak ifadesi, (c-d) GDD kümeleme sonuçları.

Şekil 3.14.'deki mantar (208001) görüntüsünde HSV renk uzayı pikseller bazında incelendiğinde, renklerin birçok bölgede birbirine karıştığı ve ayrımının yakın ölçekte insan gözüyle dahi zor yapılabildiği görülebilir. Bu sebeple uzamsal bilgiler kullanılarak, renklerin büyük alanlı bölgelerde yoğunlaştığı noktalar tespit edilebilir. GDD renk segmentasyonu işte bu yüksek yoğunluklu renk bilgilerini barındıran alanları tespit edebilmektedir. Benzer yoğunlukta ve yakın renk değişimi gösteren alanlar ise birbirlerine dahil edilerek tek bir segment olarak ifade edilmektedir. GDD renk segmentasyonu yazılımının şu anki versiyonunda uzamsal bilgiler, normalize edilmeden ve ölçekleme yapılmadan serbest olarak renk bilgilerine dahil edilmiş ve kümeleme metoduna aktarılmıştır.

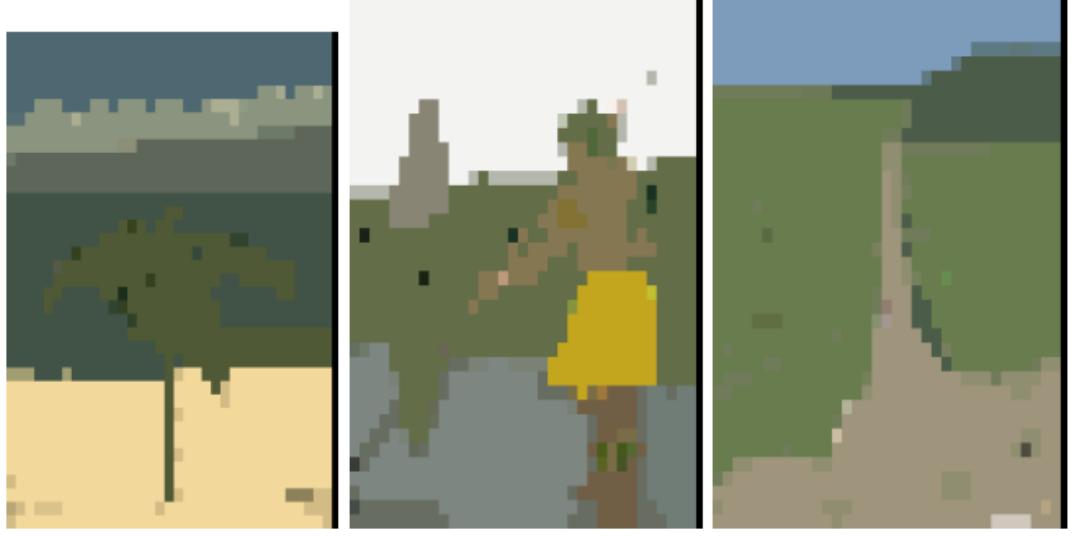
HSV renk uzayında düşük kontrast ve aydınlık değerlerin belirli bir noktada yoğunlaşmasından dolayı GDD renk segmentasyon algoritması, renk bilgisinin az olduğu gölge barındıran bölgelerde aşırı segmentasyon sonucu üretebildiği gözlemlenmiştir. Ayrıca bölgelerde renk farkı belirgin olsa bile, bölgeler arasında benzer renklere sahip bir alan bulunduğu zaman iki farklı renk bölgesi tek segment halinde aşırı-segmentasyon sonucu olarak GDD renk segmentasyonunda ortaya çıkabilmektedir. Bu durumların iyileştirilebilmesi için, kenar bulma algoritmaları gibi bazı yöntemler kullanılarak başarımlar artırılabilir.



Şekil 3.14. Mantar görüntüsü segmentasyon karşılaştırma sonuçları (321x481 piksel) (321x481 piksel): (a) Orijinal görüntü, (b) Gerçeklik tablosu, (c) K-means renk segmentasyonu, (d) K-means renk ve uzamsal segmentasyon, (e) Meanshift renk segmentasyonu, (f) Meanshift renk ve uzamsal segmentasyon, (g) Normalized-cut segmentasyonu, (h) ADP+GDD segmentasyonu, (i) GDD sonuç renk segmentasyonu.

GDD renk segmentasyonu, genel anlamıyla alansal büyüklüklerle orantılı renksel bölgelere bakıldığında başarımlar konusunda avantaja sahiptir. Örnek olarak incelemek gerekirse, bir yayayı uzaktan gördüğümüzde gelenin bir kişi olduğunu anlamamıza rağmen ancak yakına geldiği zaman detayları (üzerindeki giysileri, yürüyüş tarzı ve elbiselerinin desenlerini) fark edebilir veya belirli bir derece yakınlığa geldiğinde tanıdık olup olmadığını çıkarabiliriz. Bu sebeple segmentasyon bilgisi ve ayrımı, alansal olarak görülebilen verilerin çokluğuyla yakından ilişkilidir. GDD renk segmentasyonu algoritması, segmentlerin bahsi geçen şekilde kapladığı alanı, rengi ve uzaklık bilgisiyle birlikte parlaklık bilgisini kullanarak bölgelerin renksel çıkarımını ortaya koymaktadır.

Berkeley görüntü segmentasyonu veri setinden seçilen bazı görüntüler kullanılarak elde edilen GDD renk segmentasyonu çıktıları Şekil 3.15.'de görülebilir. Şekil 3.15.'de ilk üç görüntü '321x481' piksel boyutundayken geri kalan görüntüler yatay olarak '481x321' piksel boyutundadır. Bütün bu sonuçlardan da görülebileceği gibi GDD renk segmentasyonu, renk bölgelerini çok net ve başarılı bir şekilde ortaya çıkarmaktadır.



(a)

(b)

(c)



(d)

(e)

(f)



(g)

(h)

(i)

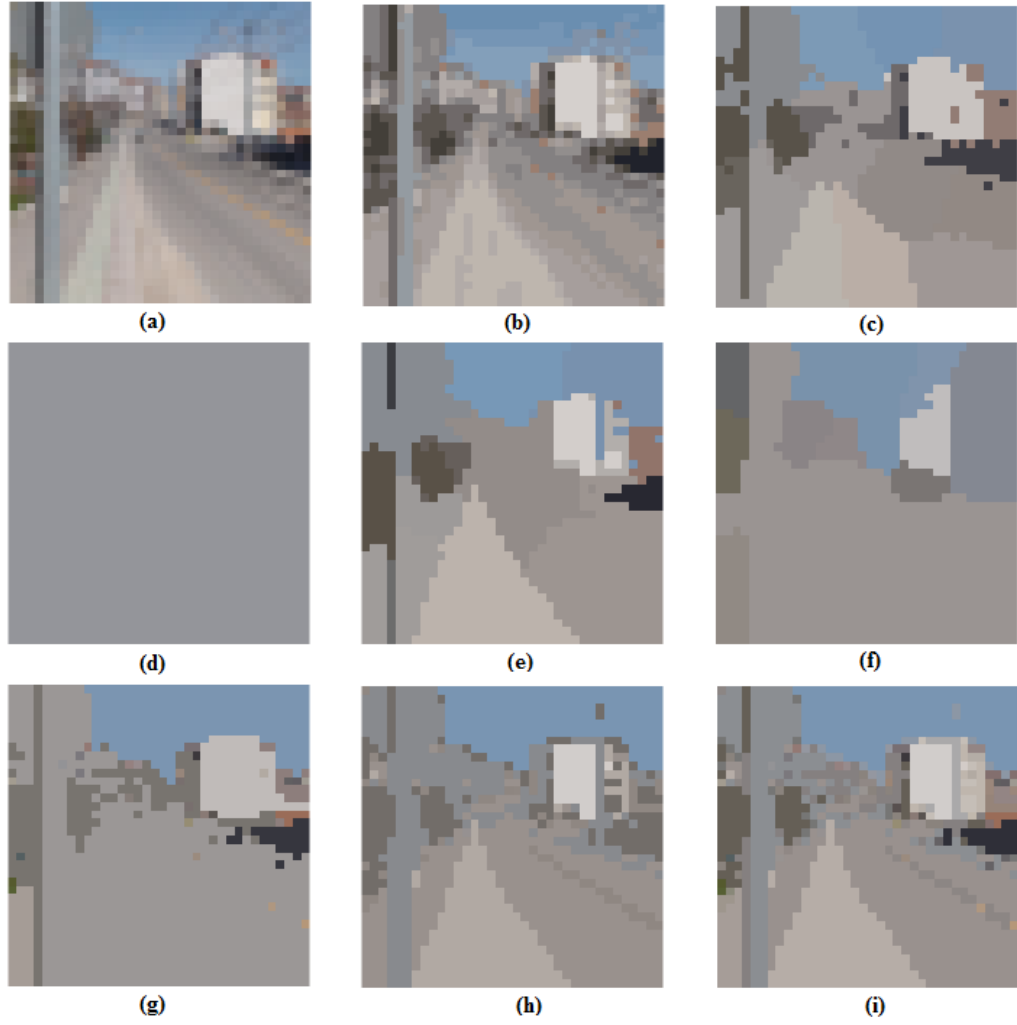
Şekil 3.15. Farklı GDD renk segmentasyonu sonuçları: (a) Palmiye ağacı (46076), (b) Kabile (101087), (c) Duvar (374067), (d) Tören (145086), (e) Fil (296059), (f) Piramit (299091), (g) Uçak (3096), (h) Kaplan-1 (187039), (i) Kaplan-2 (160068)



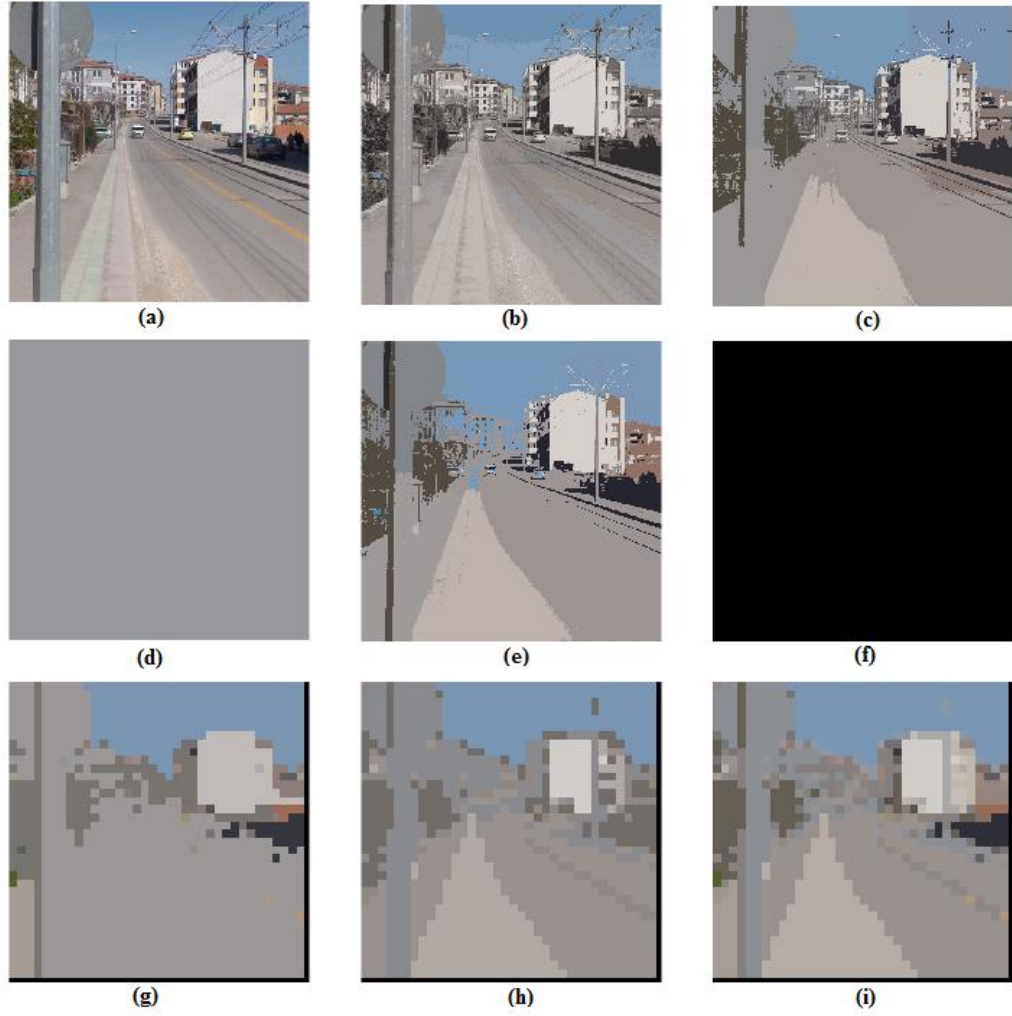
Şekil 3.16. 46076 görüntüsü palmye ağacı piksellerinde renk değişimi ve karışımı.

Şekil 3.16.'da GDD renk segmentasyonu sonucunda, Palmye ağacı (46076) görüntüsünde aşırı segmentasyona neden olan bölge yakından görülmektedir. Segmentasyon sonucu Şekil 3.15. (a) çıktısından incelenebilir. Burada daha önceden bahsedildiği gibi, piksellerdeki renk değerleri yakınlığı çok az olduğundan ve palmye ağacı segmenti ile kumsaldaki kum renkleri aynı veya çok az farkla temsil edildiğinden, hatalı segmentasyon işlemi gerçekleşmektedir.

GDD renk segmentasyonu çözünürlük düşürülerek yapıldığından dolayı, karşılaştırmaların daha iyi anlaşılması için farklı gerçek hayat görüntüleriyle hem düşük çözünürlük hem de orijinal çözünürlük üzerinden testler gerçekleştirilmiştir. Bu görüntüler içerisinde örnek olması amacıyla belirlenen test görüntüsü sonuçları, Şekil 3.17. ve Şekil 3.18. üzerinden görülebilir.



Şekil 3.17. Düşük çözünürlüklü test görüntüsü segmentasyon sonuçları: (a) Düşük çözünürlüklü görüntü, (b) K-means renk (17 segment), (c) K-means uzam + renk (17 segment), (d) Meanshift renk (1 segment), (e) Meanshift uzam+renk (17 segment), (f) Normalized-Cut (11 segment), (g) ADP+GDD (32 Segment), (h) MeanHSV+GDD (10 segment), (i) GDD renk segmentasyon sonucu (57 segment)



Şekil 3.18. Orijinal çözünürlüklü test görüntüsü segmentasyon sonuçları: (a) Orijinal görüntü, (b) K-means renk (12 segment), (c) K-means uzam + renk (12 segment), (d) Meanshift renk (1 segment), (e) Meanshift uzam+renk (12 segment), (f) Normalized-Cut (0 segment), (g) ADP+GDD (32 Segment), (h) MeanHSV+GDD (10 segment), (i) GDD renk segmentasyon sonucu (57 segment)

Görüldüğü üzere Şekil 3.17. ve Şekil 3.18.'deki sonuçlar, karşılaştırma için seçilen görüntünün çözünürlüğüne göre değişmektedir. Yüksek çözünürlükte Mean-Shift ve GDD segmentasyon yöntemleri başarılı sonuçlar üretmektedir. Düşük çözünürlükle yapılan işlemlerde ise GDD renk segmentasyonu yöntemi, çözünürlük düşürme işlemlerinde orijinal görüntüden daha fazla veriyle özellik elde ettiğinden GDD kümeleme daha ayrıntılı ve başarılı sonuçlar vermiştir. Karşılaştırmaların doğru ve sağlıklı olması açısından, veri seti görüntüleri orijinal çözünürlüğünde alınarak başarımlar elde edilmiştir. Fakat uygulama esnasında diğer segmentasyon yöntemlerinde de çözünürlük düşürme uygulanması durumunda, aradaki başarımlar

farkının gözlenmesi açısından Şekil 3.17. ve Şekil 3.18.'de yapılan karşılaştırmalar önemlidir.

Literatürdeki renk segmentasyonu çalışmalarında sıkça kullanılan görüntülerden bazılarının orijinal halleri ve GDD renk segmentasyonu sonuçları aşağıdaki Şekil 3.19.'dan incelenebilir.



Şekil 3.19. Sıkça kullanılan renk segmentasyonu görüntülerinin orijinali (sol) ve GDD renk segmentasyon sonucu çıktısı (sağ).

3.2.6. GDD renk segmentasyonu sonuçları

Renk segmentasyonu çalışmasında, GDD kümeleme metodunu kullanan yeni bir renk segmentasyonu yöntemi öne sürülmüş, geliştirilmiş ve test edilmiştir. Algoritma rastsal olmayan, stabil sonuçlar üreten (farklı koşumlarda aynı sonucu veren) ve harici parametre gerektirmeyen bir yapıdadır. Global renk yoğunlukları ve uzamsal bilgilerin birlikte kullanımıyla kümeleme işlemleri üzerinden bölgelerin ayrımı sağlanmış ve segmentasyon sonucunun doğruluğu artırılmıştır.

Bir diğer yenilikçi özellik ise, paralel olarak farklı segmentasyon sonuçlarını iş parçacıklarında hesaplayarak farklı tipteki düşük çözünürlüklü özellikleri (ADP ve MeanHSV) sentezlemesidir. Bu iki farklı çözünürlük düşürme yaklaşımı, iki farklı perspektifte renksel bölgelerin ayrımını sağlayarak sentezlenen sonuç verisinin başarımını artırmıştır. Segmentasyon işlemi, paralel yapıda gerçekleştirilen bir sistem olduğundan dolayı sentezleme ve parçacıkların veri transferleri dışında iş parçacıklarından en uzun süre alan taraf kadar zamansal işlem süresi kullanmaktadır.

Sonuç olarak GDD renk segmentasyon algoritması, çalışma öncesi herhangi bir parametre gerektirmemekle birlikte hem renk hem de uzamsal verileri kullanarak bu alanda başarılı net sonuçlar üretmektedir.

3.3. Stereo görüntüde engel tanıma ve yer düzleminin bulunması

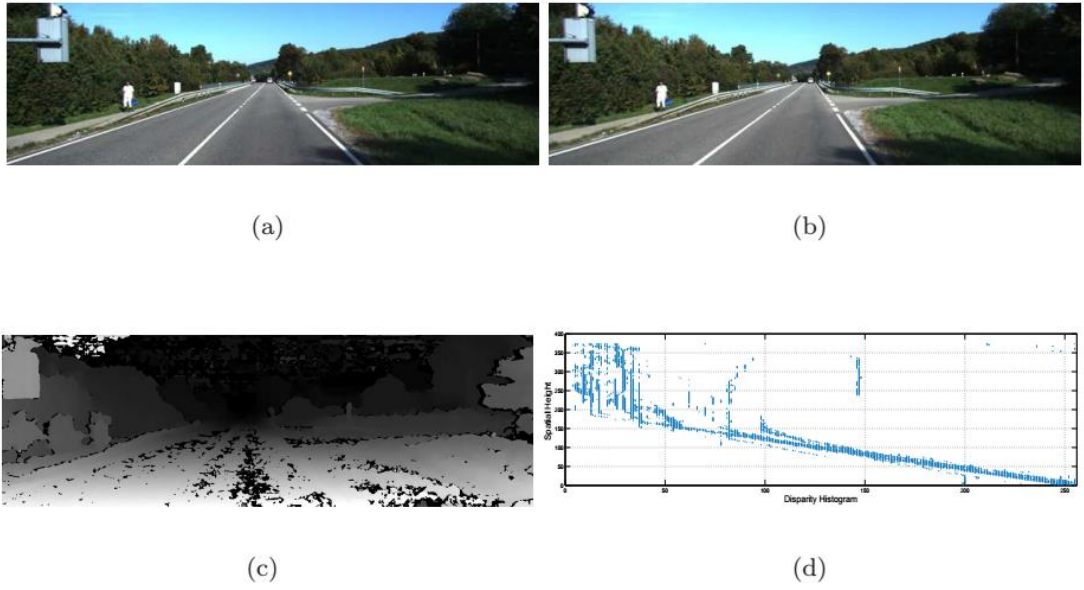
Yol üzerindeki yayaların tespitinde, yer düzlemi ve engellerin bulunması gerekli değil gibi görünse de yayaların karıştığı kazaların azaltılmasında ve sürüş destek sistemi geliştirilmesi aşamasında, aracın gittiği yer düzlemi ve engellerin doğru tespiti büyük önem taşımaktadır. Başlangıçta yayalar tespit edilmeden önce engel tanıma yöntemiyle, engeller yer düzleminden ayrılır. Sonrasında engeller üzerinden uzaklığa göre tarama işlemi yapılarak yaya tespit işlemi gerçekleştirilir. Bu açıdan farklı uzaklıktaki yayaların daha doğru tespiti için yer düzlemi ve engellerin tespiti gerekli olmaktadır. Uygulama açısından ayrıca görüntünün tamamında sabit çözünürlükte yaya aranması, gökyüzünde insan aramak ve insanların her uzaklıkta

aynı büyüklüğe sahip olduğu varsayımını yapmak anlamına gelmektedir. Bu sebeple yer düzleminin ve engellerin elde edilmesi hem yaya tespitinde hem de otonom araçlarla birlikte araç sürüş destek sistemlerinde çevre ve araç güvenliğinin sağlanması açısından önem arz etmektedir.

Tez kapsamında araçlar üzerinden alından stereo görüntüler kullanılarak, literatürde yer alan yer düzlem tespit sistemleri incelenerek ve eksiklikleri ile geliştirilebileceği alanlar çalışılarak bu alanda ilerleme sağlanması hedeflenmiştir. Yer düzlemi tespitinde derinlik haritası ve V-disparity dönüşümü kullanılmış ve öne sürülen farklı yaklaşımlar test sonuç görüntüleriyle birlikte sunulmuştur.

3.3.1. V-disparity verisinin elde edilmesi

V-disparity, Labayrade ve arkadaşları tarafından öne sürülmüş ve derinlik haritası kullanılarak derinlik bilgisinin yatay düzlemde bulunan frekanslarına göre oluşturulduğu bir dönüşüm veri uzayıdır [68]. Öncelikle stereo görüntü üzerinde blok eşleştirme yöntemi kullanılarak iki görüntü arasındaki özelliklerin uzamsal değişimi hesaplanır. Uzamsal değişim miktarı bize derinlik haritasını vermektedir. Tez kapsamında derinlik haritası tespiti için yarı global blok eşleştirme (Semi-Global Block Matching) algoritması kullanılmıştır [61]. Sonrasında Labayrade ve arkadaşları tarafından uygulanan dönüşümsel denklemler çerçevesinde, derinlik haritasında bulunan her yatay düzleme karşılık gelen bir nokta oluşturularak V-disparity verisi elde etmişlerdir [68]. V-disparity'deki her noktanın parlaklık değeri, aynı yatay ekseninde bulunan derinlik haritasındaki toplam frekans değerlerine göre atanmaktadır. Şekil 3.20.'de KITTI veri tabanından alınan '2011_09_26_drive_0070' setindeki '355'nolu sağ ve sol görüntü kullanılarak yarı global blok eşleştirme metoduyla derinlik haritası elde edilmiştir. Sonrasında derinlik haritasından elde edilen, eşik değeri uygulanmış V-disparity verileri Şekil 3.20. (d)'deki grafikte temsil edilmiştir. Derinlik haritası çıkarımında blok genişliği '9' piksel olan kare bir blok yapısı tercih edilmiştir.



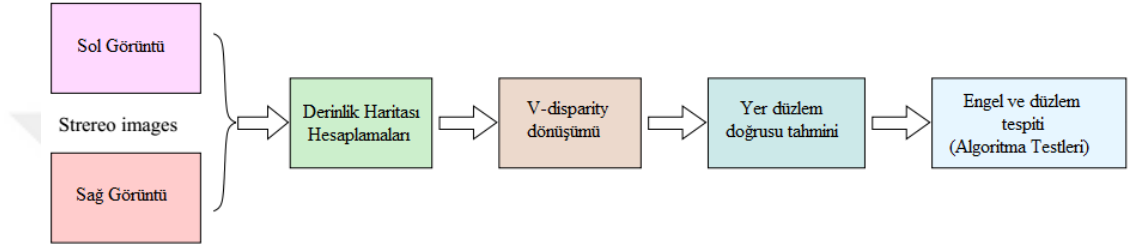
Şekil 3.20. '355'nolu stereo test görüntüsünden V-disparity grafiğinin elde edilmesi aşamaları: (a) Sol test görüntüsü, (b) Sağ test görüntüsü, (c) Derinlik haritası, (d) V-disparity grafiği (eşik değeri uygulanmış).

3.3.2. V-disparity verisini kullanarak yer düzleminin bulunması

Literatürdeki çalışmalar incelendiği zaman V-disparity ile yer düzleminin elde edilmesi noktasında birçok çalışma, yer düzlemini belirleyen doğruyu bulmak için Hough dönüşümü yöntemini kullanmıştır [68, 67, 72]. Fakat bu yöntemin sonucunda elde edilen yer düzlemini belirtir doğru, Zhao ve arkadaşları tarafından yapılan çalışmada incelendiği üzere, yer düzleminde belirgin yol özelliklerinin olmadığı koşullar altında istenilen sonucu verememektedir [71]. Aynı çalışmada incelendiği üzere eğimsel değişimlerde, V-disparity'deki yer düzlemini belirtir doğru paralel olarak salınım göstermektedir. Tez kapsamında yapılan testlerde, genel olarak yer düzlemini belirleyen noktalar uzaktan bir doğru olarak görünse de aslında kendi içerisinde düşey düzlemde değişimler gösteren rastsal bir alanı ifade etmektedir. Bu kapsamda Kostavelis ve arkadaşlarının yaptığı çalışmadaki gibi hataya sebep olabilecek bölge belirlenmesi başarımlarını artırmaktadır [72]. Kostavelis ve arkadaşları çalışmalarında, Hough dönüşümü ile lineer doğru tespiti yaptıktan sonra belirli bir bölge tanımlayarak dışında kalan alanları engel özellik vektörü olarak tanımlamıştır. Sonrasında bunları düzlem ve engellerin tespiti için SVM (Support Vector Machine) yöntemi eğitim için kullanmıştır. Çalışmalarındaki doğrulama ve

test sonuçları makaleden incelenebileceği gibi bölge tespiti hakkında ayrıntılı bilgi verilmemiştir.

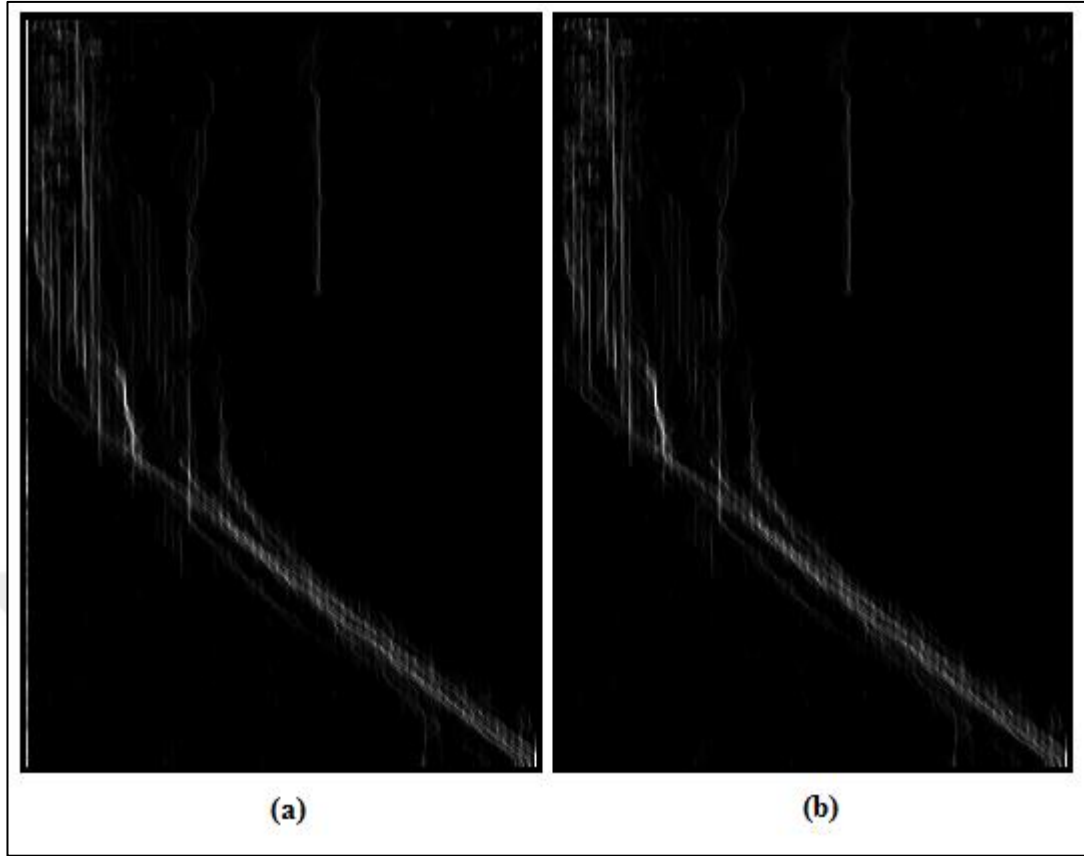
Tez çalışması kapsamında yer düzlemini belirleyen doğru ve bu bölgedeki değişimlerin tahmin edilmesiyle hesaplanan rastsal değişkenleri barındıran alan, Hough dönüşümü ve global korelasyon yöntemlerinden farklı olarak ağırlıklı regresyon ve aykırı değer (outlier) filtresi yardımıyla elde edilmiştir.



Şekil 3.21. Engel tanıma sisteminin blok diyagramı.

Şekil 3.21.'de genel manada yer düzlemi ve engellerin bulunması sırasında izlenecek yol, blok diyagram halinde gösterilmiştir. Şekil 3.21.'de görüleceği üzere başarımların sonuçları, yer düzlemi tahmini ile olduğu kadar derinlik haritasının başarısına ve V-disparity elde edilmesinde kullanılan yöntemlere göre de değişim göstermektedir. V-disparity verisi frekans kullanımından kaynaklı derinlik haritasındaki hatalara karşı tolerans göstermektedir. Fakat frekans baskınlığını etkileyen yüksek hata içeren derinlik haritası verileri bütün sistemin başarısını negatif yönde etkilemektedir.

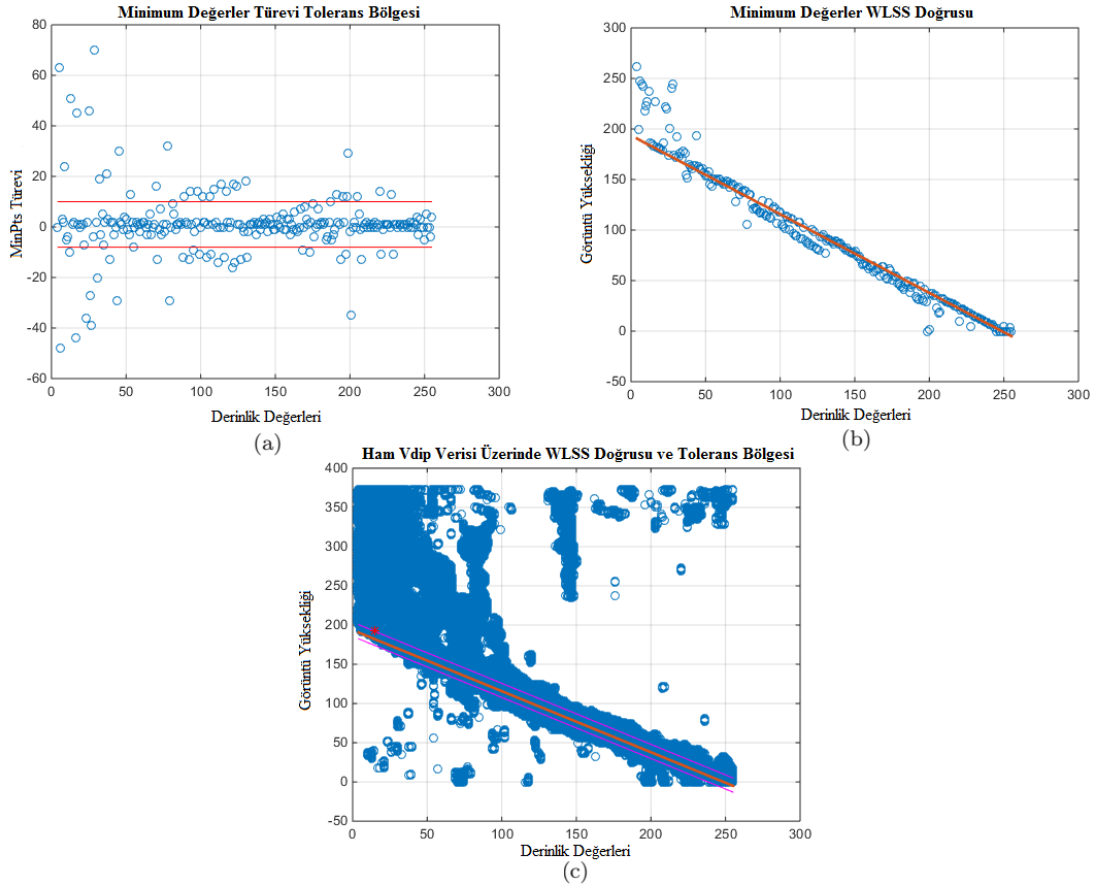
Derinlik haritasında gökyüzü bölgesi, hatalar dışında sıfıra yakın bir değer aldığından dolayı, [0-255] arası ölçeklenmiş derinlik verileri içerisinde en yüksek uzaklık değerini barındıran derinlik bilgilerinin V-disparity'den çıkarılmasıyla gökyüzünü tanımlayan veriler elenmiştir. Bu sayede sadece yer düzlemi ve engel bölgelerini barındıran V-disparity verisi, Şekil 3.22. (a)'dan filtreleme ile Şekil 3.22. (b)'de görüldüğü gibi elde edilir. Bu işlem ayrıca derinlik haritası içerisinde hatalı olarak hesaplanan sıfır değerindeki verilerin de V-disparity'den çıkarılması anlamına gelmektedir.



Şekil 3.22. V-disparity verilerinde en uzakta kalan gökyüzü ve hataları temsil eden derinlik verilerin filtrelmesi: (a) Orijinal V-disparity verisi, (b) filtrelme sonucu elde edilen V-disparity verisi (bknz. en soldaki düşey çizginin filtrelmesi).

Şekil 3.23.'te, Şekil 3.21.'deki blok diyagramının yer düzlemi tahmininden bir bölümü göstermektedir. Burada derinlik haritasının hata barındırmadığı farz edildiğinde, bir satırdaki derinlik bilgilerinde -eğer satır tamamen engelden oluşmuyorsa- gök yüzü dışında en uzağı temsil eden veriler, yer düzlemini temsil etmesi gerekir mantığı üzerinden yola çıkılarak algoritma geliştirilmiştir. Şekil 3.22.'den de görüleceği üzere bir satırdaki derinlik verilerine bakacak olursak bir engelin derinlik değeri, yer düzlemi derinlik bilgisine her zaman eşit veya yüksek olmaktadır. Bu bilgiler ışığında hatalar dışında düşey düzlemdeki en küçük V-disparity değerleri, yer düzlemini belirtir doğruyu oluşturması gerekir sonucuna ulaşırız. Elbetteki bu bilgiler, gözlem verilerine dayandığından ve bulunan noktadaki yol verileri ile yer düzleminin uzaklığa bağlı yükselti değişikliklerinden etkilendiğini düşünürsek, bu değerleri rastsal gözlem verisi olarak inceleyebiliriz. Derinlik haritasındaki herhangi bir satırdaki hata, belirli bir oranı aşmadığı sürece frekansı

baskın olmayacağından V-disparity oranlarına eşik değer uygulayarak; Şekil 3.23. (b)'de V-disparity düşey minimum değerlerden elde edilen grafik, yer düzlemi tespiti için kullanılmıştır.



Şekil 3.23. Yer düzlemi tespiti için yapılan işlemler: (a) Tolerans bölge tespiti, (b) Minimum değerler üzerinde WLSS regresyon doğrusu, (c) V-disparity verileri üzerinde tolerans bölgesi ve regresyon doğrusunun gösterimi.

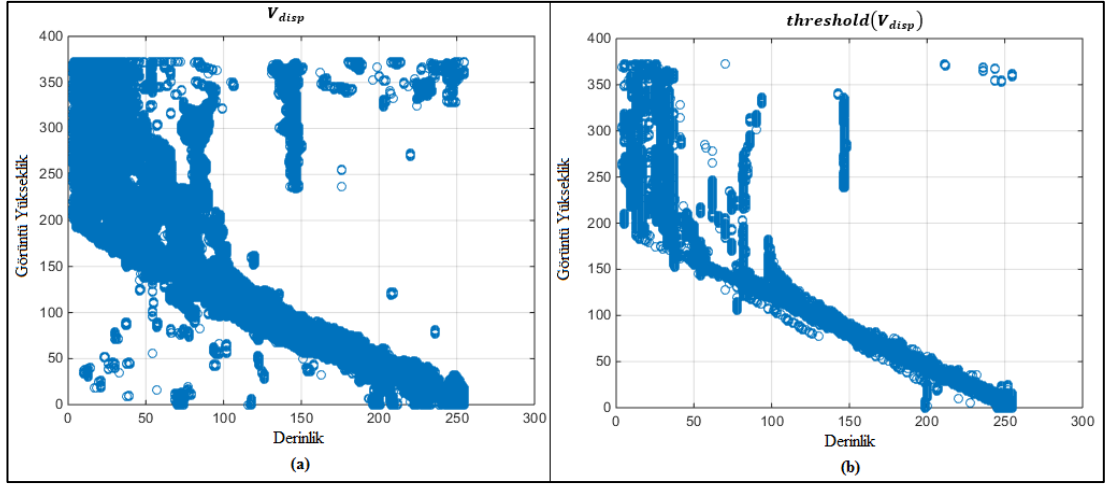
Şekil 3.23. (c)'deki grafiğe bakacak olursak yer düzlem doğrusu altında Şekil 3.22.'de görülmeyen düşük frekans değerini barındıran hataların olduğunu anlayabiliriz. Eşik değeri kullanarak düşük frekanslı bölgelerin giderilmesi sırasında, ufuk noktasına yakın düşük frekanslı hatalar da eğer hata oranı düşüğe giderilmiş olur. Fakat ufuk noktasına doğru yaklaştıkça yer düzlemini belirleyen doğrunun tespitinde kamera kadrājına daha fazla nesne girdiğinden ve yolu temsil eden alan azaldığından yer düzlemi tespiti zorlaşmaktadır. Bu sebeple frekans değerlerinin ağırlık olarak temsili ile yer düzlemini belirlemenin daha uygun olacağı görülmüştür.

Yer düzlemi doğrusunun tespiti için yapılan işlemleri ele alacak olursak; eşik değeri tespiti, düzlem doğrusu tahmini, hata tolerans bölgesinin hesaplanması ve kamera yüksekliğinin V-disparity üzerindeki ve yer düzlemindeki etkisinin bulunması olarak sıralayabiliriz.

V-disparity’de yer düzlemini belirleyen minimum değerlerin elde edilmesi için hataların giderilmesi ve eşik değerlerinin uygulanması Denklem 3.22 ve Denklem 3.23 ile sağlanmıştır. Bu noktada hataların giderilmesi için daha farklı teknikler de kullanılabilir. Kameraya yakın olan bölgelerde, V-disparity yer düzlemi derinlik bilgilerinin frekans değerleri, hatalara oranla çok daha yüksektir. Hata barındırmaması durumunda ise V-disparity içerisindeki derinlik bilgileri yalnızca yer düzlemini veya üzerinde kalan engel bölgelerini ifade edecektir. Bu durumda V-disparity içerisinde her düşey düzlemdeki minimum nokta yer düzlemini temsil eden bir nokta olacaktır. Hatalar da dikkate alındığında yer düzleminin altında yer alan pikseller her zaman üstünde yer alan piksellerden az olacaktır. Bu nedenlerden dolayı yer düzlemi ve engellerin genel olarak daha az hata ile tespit edilebilmesi için Denklem 3.22’de eşikleme (threshold) uygulanarak sonrasında Denklem 3.23 ile yer düzlemini temsil eden minimum noktalar (MinPts) elde edilmiştir.

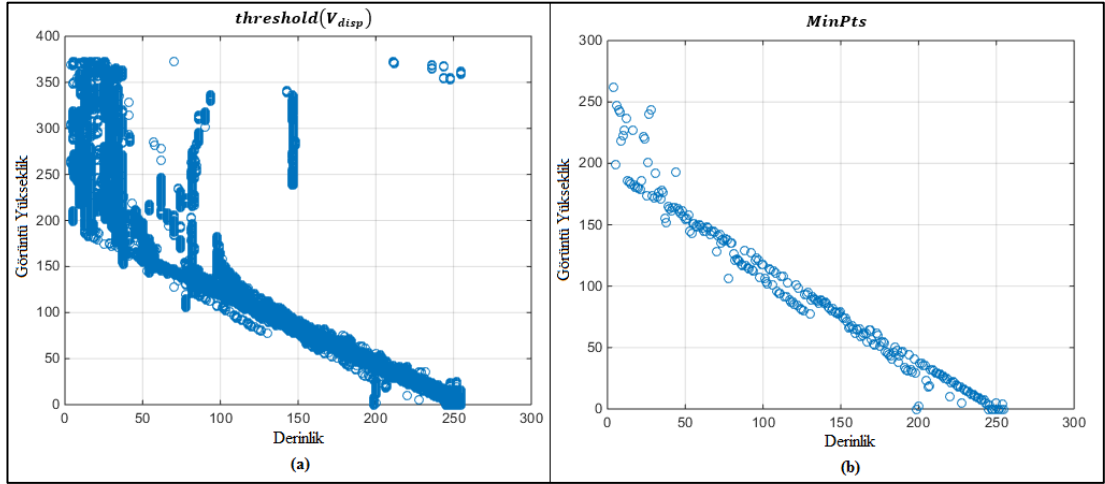
$$threshold(V_{disp}) = [V_{disp} \geq \mu(V_{disp}) + \sigma(V_{disp})] \quad (3.22)$$

$$MinPts = \min(threshold(V_{disp})) \quad (3.23)$$



Şekil 3.24. ‘355’ nolu test görüntüsü üzerinde; (a) Ham V-disparity verileri, (b) Eşik değeri üzerindeki V-disparity verileri.

Şekil 3.24. üzerinde görülebileceği üzere derinlik haritasından V-disparity elde edildikten sonra Denklem 3.22 uyarınca Şekil 3.24. (a)’dan Şekil 3.24. (b)’ verisi elde edilmiştir. Düşük frekanslı verilerin de net görülebilmesi için Şekil 3.24.’de frekans değerleri belirtilmemiştir. Frekans değerleri olan versiyonu için Şekil 3.22. incelenebilir.

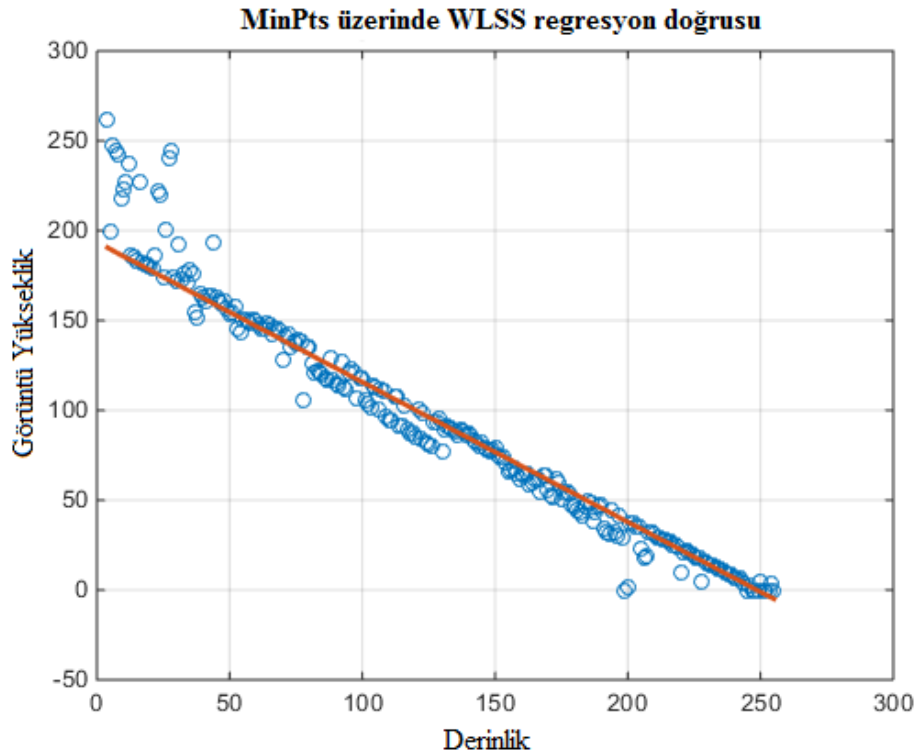


Şekil 3.25. ‘355’ nolu test görüntüsü üzerinde; (a) Eşik değeri üzerindeki V-disparity verileri, (b) Her derinliğe karşılık gelen minimum değerler grafiği.

Şekil 3.25.’de görülebileceği üzere Denklem 3.22’den sonra elde edilen veriler, her derinlik noktasına karşılık gelen düşey düzlemdeki noktalara uygulanarak Denklem 3.23 uyarınca Şekil 3.25. (b)’deki gibi elde edilmiştir.

Yer düzlem doğrusu tahmininde, tahmin değerleri ufuk çizgisine doğru gittikçe yatay düzlemdeki engellerin sayı ve boyutu arttığından frekans değerleri azalarak yüksek değişim göstermektedir. Kameraya yakın bölgelerde ise tam tersi yer düzlemini belirtir noktalar daha yüksek frekansla temsil edilmektedir. Bu sebeple olasılıksal olarak yol olma ihtimali daha yüksek alanların frekans değeri de daha yüksek olduğundan, doğrunun en iyi ifade edilebilmesi için; Denklem 3.24'teki ağırlıklı en küçük kareler toplamı (WLSS) regresyon doğrusu bulma yöntemi kullanılmıştır.

$$WLSS(\beta, \vec{\omega}) = \sum_{i=1}^n \omega_i (y_i - x_i \cdot \beta)^2 \quad (3.24)$$



Şekil 3.26. '355'nolu test görüntüsü üzerindeki V-disparity verisinden elde edilen minimum noktalar ile oluşturulmuş regresyon doğrusu.

Denklem 3.23'te hesaplanan minimum noktalar derinlik frekans bilgileri ile birlikte Denklem 3.24'te kullanılmış ve Şekil 3.26.'teki grafik elde edilmiştir. Denklem 3.24'teki ' x_i ' değerleri derinlik değerlerini temsil ederken ' y_i ' değerleri görüntü

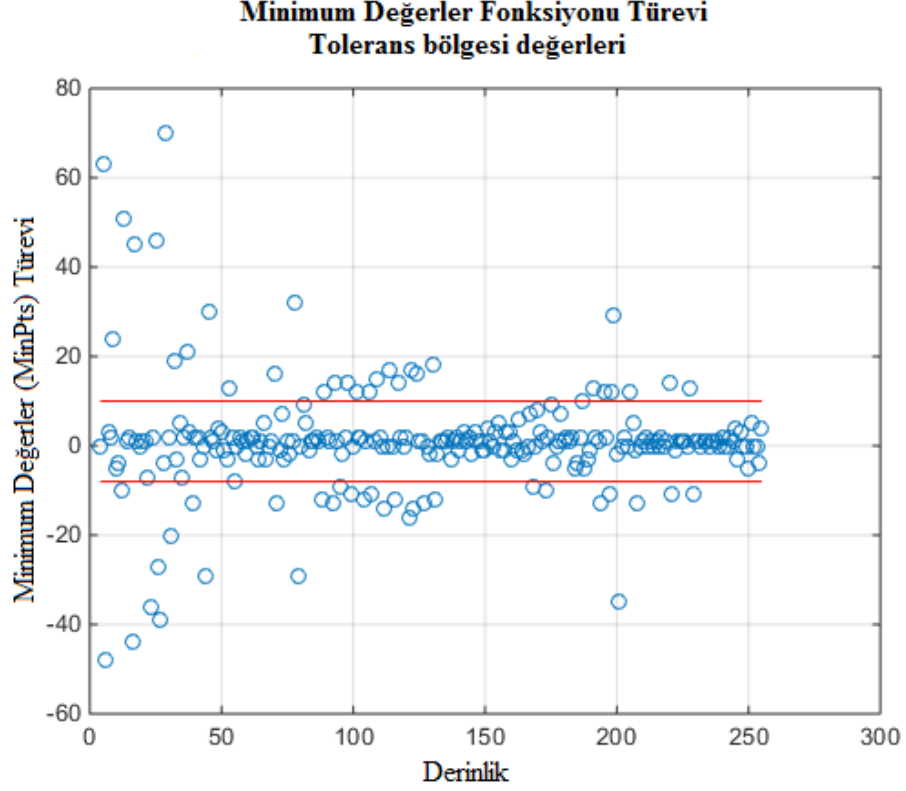
yüksekliğindeki satır sayısını belirtmektedir. Ayrıca denklem içerisindeki ' ω_i ' değeri frekans değerlerini barındıran ağırlığı temsil eden değişkendir. Bu nedenle kameraya yakın olan noktaların önemi daha fazla olup uzaklık arttıkça yer düzlemi derinlik frekans bilgisi de azaldığından ve doğrunun tespiti için gerekli noktaların tespiti zorlaştığından dolayı frekans değerlerinin kullanımıyla yüksek başarımda doğru tespiti sağlanmaktadır.

Denklem 3.24'te belirlenen doğru sadece minimum noktaları kapsamaktadır. Yer düzlemini belirleyen ve V-disparity grafiğini ikiye ayıran bölgenin, sadece tek bir doğru yerine doğrusal bir alan olduğunu göz önüne alırsak; yer düzlemini ve engelleri ayıran bölgeyi, elde edilen ağırlıklı regresyon doğrusu ve aykırı değer (outlier) filtresi yardımıyla daha doğru bir şekilde tespit edebiliriz. Aykırı değer filtresi ile düzlem eğiminde gürültünün neden olduğu hataların kapsamının belirlenmesi sağlanır. Burada eğer yol düzleminin doğru olduğunu varsayarsak eğim sabit bir ' m ' değerini alacaktır. Fakat gözlemler sonucunda yer düzlemi ile engelleri ayıran bölgenin rastsal bir değişim gösterdiği görülmüştür. Burada doğruyu belirleyen gözlem noktalarını rastsal değişken olan ' R ' olarak tanımlarsak, gözlem verilerinin türevini aldığımızda, her birim bölge ' Δx ' (derinlik bilgisi) için türev değerleri farklılık göstermektedir. V-disparity ile elde ettiğimiz fonksiyonun türevinin alınabilirliğini, sonsuz V-disparity verisini ele alırsak belirli bir noktadaki engeller yer düzlemine yakınsayacağından, Denklem 3.25'teki gibi temsil edilebiliriz. Rastsal değişkenler ve türevinin alınabilirliği hakkında daha fazla bilgi için Stensby'nin notları incelenebilir [114].

$$\lim_{\Delta x \rightarrow 0} E \left[\frac{R(x - \Delta x) - R(x)}{\Delta x} - R'(x) \right]^2 = 0 \quad (3.25)$$

Denklem 3.25 üzerinde her ' Δx ' değeri için türev değerini o bölgedeki parçalı türev ile temsil edersek, buradan her ' Δx ' bölgesi için eğim elde etmiş oluruz. Bütün görüntü üzerindeki ' Δx ' bölgelerinin toplamı görüntü sayısı sonsuza gittiğinde yer düzlemini temsil eden doğruya yakınsar. Buradan hareketle Şekil 3.27.'daki türev

verilerini ve ağırlıklı aykırı değer filtresi işlemleriyle elde edilen tolerans bölgesini gösteren grafik elde edilmiştir.



Şekil 3.27. '355'nolu test görüntüsü üzerindeki V-disparity minimum değerlerin türevi ile ağırlıklı aykırı değer (outlier) filtresi ile elde edilen tolerans bölgesi (kırmızı çizgiler arasında kalan alan) sonucu.

Şekil 3.27.'de görülebileceği gibi parçalı türev verileri, belirli bir bölgeye yakınsamakla birlikte bölge dışında gürültüden ve verilerin elde edilmesine kadar geçen aşamalardan kaynaklı hatalar oluşmaktadır. Ayrıca engeller de bu alanda aykırı noktalar oluşturabilmektedir. Burada genel eğimi en iyi temsil edebilecek noktaların bulunduğu en olası bölgeyi tanımlayacak olursak, bu bölge Şekil 3.27.'de görülen tolerans bölgesi dahilinde olacaktır. Tolerans bölgesi, verilerin Denklem 3.26 ile medyanı hesaplandıktan sonra Denklem 3.27 ve Denklem 3.28 kullanılarak birinci ve üçüncü kartiller (Q1 ve Q3) belirlenerek, Denklem 3.29'daki gibi bu değerlerden kartiller arası uzaklığın hesaplanmasıyla elde edilir. Hatalardan ve rastsal değişimlerden kaynaklanan bu tahmini bölgenin tespiti tolerans değişkeni olarak Denklem 3.30 ile hesaplanmaktadır. Kısacası çalışmada aykırı değer filtresi

içerisinde kartiller arası uzaklığın bir buçuk katı yerine kendisi kullanılarak oluşturulan alan, yer düzlemi rastsal alanı belirten tolerans değişkeni adı ile ifade edilmiştir.

$$Medyan = l + \frac{h}{f} \left(\frac{N}{2} - c \right) \quad (3.26)$$

$$Q_1 = l + \frac{h}{f} \left(\frac{N}{4} - c \right), \quad (3.27)$$

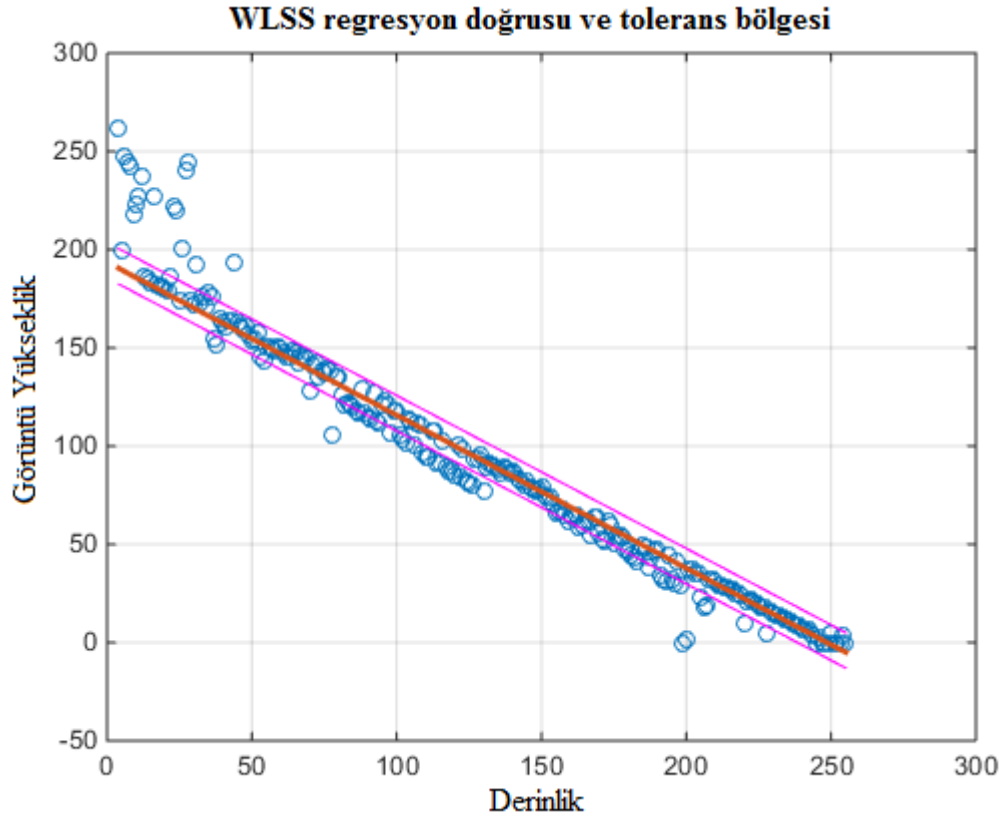
$$Q_3 = l + \frac{h}{f} \left(\frac{3N}{4} - c \right) \quad (3.28)$$

Sembollerden ' l ' kartil grubu alt sınırını, ' h ' kartil grubunun genişliğini, ' f ' kartil aralığı frekansını, ' N ' toplam gözlem sayısını (toplam frekans $\sum f_{R(x_i)}$ değerini), ' c ' ise takip eden kümülatif frekansları ifade etmektedir. Aykırı değer filtresi hakkında daha fazla bilgi için Jaisson'un ders notları incelenebilir [115].

$$IQ = |Q_3 - Q_1| \quad (3.29)$$

$$Tolerans = \max(|Q_1 - IQ|, |Q_3 + IQ|) \quad (3.30)$$

Aykırı değerler belirlenirken, verilerin derinlik haritasındaki her satırda belirme sıklıkları da frekans olarak V-disparity verisinden alınarak hesaplanır. Denklem 3.30'da görüldüğü üzere tolerans değeri kartiller arası fark (IQ) ile tanımlanarak ağırlıklı regresyon doğrusu üzerindeki ve altındaki bölgeler yer düzlem tahminindeki olası değişim bölgesi olarak belirlenir.



Şekil 3.28. Minimum değerler ile elde edilen regresyon doğrusu ve tolerans bölgesi.

Denklem 3.30’da belirlenen tolerans bölgesi, Şekil 3.28.’de görüldüğü üzere tespit edilen regresyon doğrusu etrafındaki olası değişimlerin tahmini olarak belirlenir. Eflatun rengi doğrular arasında kalan alan bu tahmini bölgeyi temsil etmektedir. Yer düzlemini içeren noktaları engel olarak göstermemek için regresyon eğrisinde bulduğumuz düzlem tolerans kadar yukarı çekilir. Her satırda hatalar dışında kalan minimum derinlik noktaları yer düzlemini belirtir tanımlaması gereğince bu düzlemde yukarı çekme işlemi sayesinde tolerans bölgesi dahilinde hataların azaltılması sağlanmıştır.

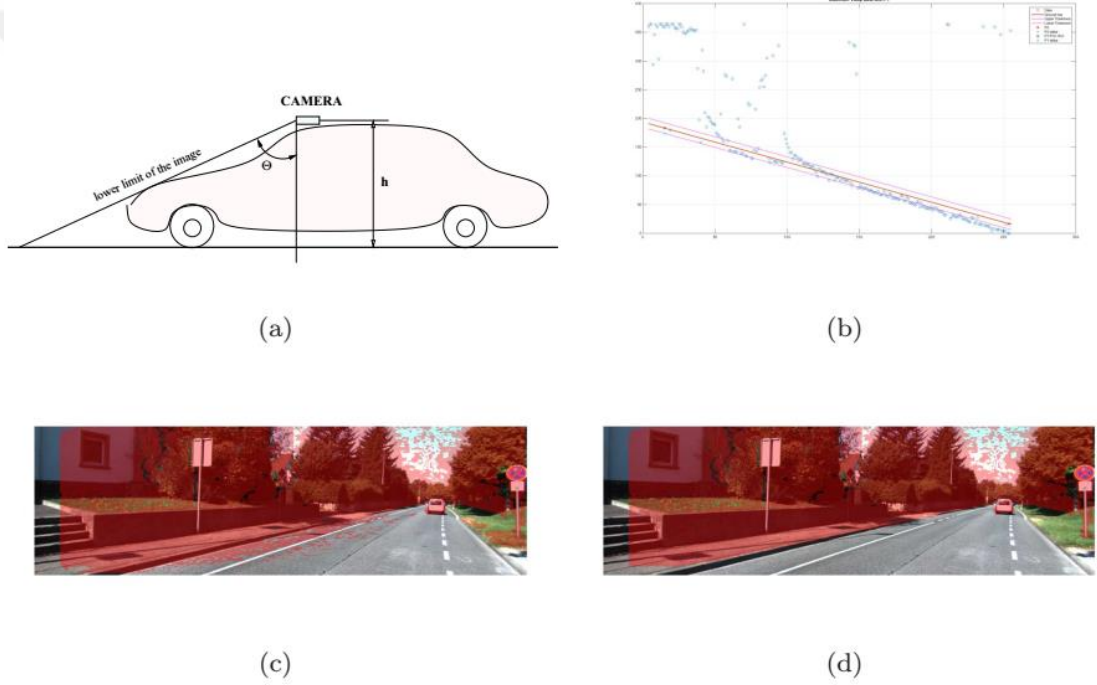
Ayrıca unutulmamalıdır ki burada bahsedilen tolerans, yer düzlemini belirleyen doğrunun değişim gösterebileceği aralığını temsil etmektedir. Gözlemlerden ortaya çıkan ve rastsal değişkenlerin dağılımına göre tanımlanan tolerans aralığı yer düzlemi verilerinin yoğunlukla gözlemlenme aralığını belirtmek için kullanılmıştır. Bu sebeple regresyonda kullanılan tolerans veya istatistikte kullanılan diğer tolerans aralıkları ile karıştırılmamalıdır.

Yapılan hesaplamalar ile yer düzlemi doğrusu, engel tanımada iyi sonuç vermesine karşın kameranın araç üzerindeki yüksekliği hesaba katılmadığından dolayı belirli bir hata payının oluştuğu sonucu yapılan testlerde görülmüştür. Bunun sebebi farklı yükseklikte kameraların kurulumu veya aracın çukur gibi eğimli yerlerden geçmesi durumunda V-disparity verilerinin değişmesidir. Kamera yüksekliği ile açısı, yer düzlemini ayıran doğruyu belirlemede ve başarımın artırılmasında büyük önem taşımaktadır. Bu nedenlerden dolayı kamera yüksekliği ve açısına göre, yer düzlem doğrusunun doğrultulması işlemi yapılmalıdır.

Araç kamera yüksekliği elde edilen V-disparity verisindeki doğrusal noktaların yatay düzlemde değişimini, kamera açısı ise engellerle yer düzlemini belirtir verilerin V-disparity temsilde açısını etkilemektedir. Bu sebeple yukarıda daha önceden tespit ettiğimiz doğru sadece yer düzlemine yükseklik verisi olmadan bakıldığında yapılan tahmini doğruyu göstermektedir. Kamera yüksekliği ve açısını ele aldığımız zaman ise bütün görüntüdeki derinlik değerleri bu verilere göre değişiklik göstereceğinden dolayı kamera parametrelerine ve kameranın araç üzerindeki yüksekliğine göre kamera gözlemci noktası P_1 , değişikliğe uğratarak doğrunun en sağ noktasından yukarı yönlü bir yükseltmeyle kamera ve ufuk çizgisi arasında engelleri tanımlayan doğru oluşturulur. Bu yükseltme işlemi, kameraya yakın yer düzlemi doğrultusunda düşük yükseklikteki engellerin tespitinde hataya sebep olmasına rağmen genel yer düzleminde engel olarak görülebilecek hatalı alanların giderilmesini sağlamaktadır.

Belirlenen yer düzlem doğrusunda derinlik ' 0 ' noktasını ufuk çizgisi olarak varsaydığımızda ufuk çizgisinde yer düzlemini belirleyen alan içerisinde kalan en uzaktaki nokta, yer düzleminde ufuk noktasına yakınsayan bir ' P_h ' noktası olacaktır. Kamerayı temsil edecek nokta ise belirlenen alanın dışında ' 255 ' değerinden büyük olmasına rağmen yükseklik ve açıyı göz önüne alarak bu noktayı da ' P_c ' olarak adlandıralım. Bu durumda bu iki nokta arasındaki yer düzlemini ayıran doğru, daha önceden tahmin ettiğimiz alansal doğrunun kameraya yakın noktasından biraz yukarisından başlayarak ufuk noktası ' P_h ' noktasına uzanan bir şekilde oluşur. ' P_c ' noktasını, kamera öz değerlerine göre gerçek kamera yüksekliğinin piksel bazında

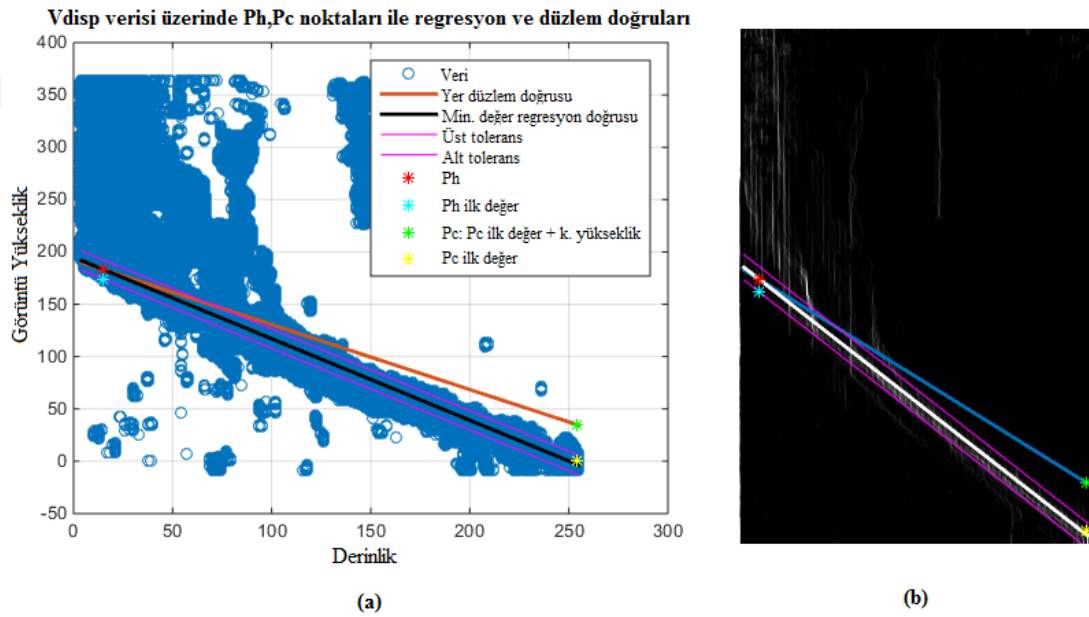
hesaplanmasıyla ve daha önceden belirlenen yer düzlem bölgesinin kullanımıyla belirleyebiliriz. Ayrıca görüntü üzerindeki bilgiler kullanılarak sadece o sisteme ait, tahmini sabit bir değer belirlenebilir. Kamera yüksekliği her araç için monte edilme işleminden sonra sabit olduğundan dolayı bu belirlenen yükseklik, bütün görüntüler için sabit bir değer taşıyacaktır. Çalışmada sadece veri tabanı görüntüleri kullanıldığından dolayı kamera yüksekliğinin ve açısının etkilerini Denklem 3.22'dekinin aksine, minimum değil verilerin maksimumlarını ve bu noktalardaki toleranslarını Denklem 3.30'daki gibi tahmin ederek ' P_c ' noktası tespit edilmiş ve testler gerçekleştirilmiştir.



Şekil 3.29. Engel tanımda kamera etkisi noktası ' P_c ' noktası tahmini ile düzlem doğru tahminlerinin etkileri: (a) Sadece yer doğrusu düzlem tahmini, (b) ' P_c ' noktasına göre yapılan doğru değişiminin etkisi.

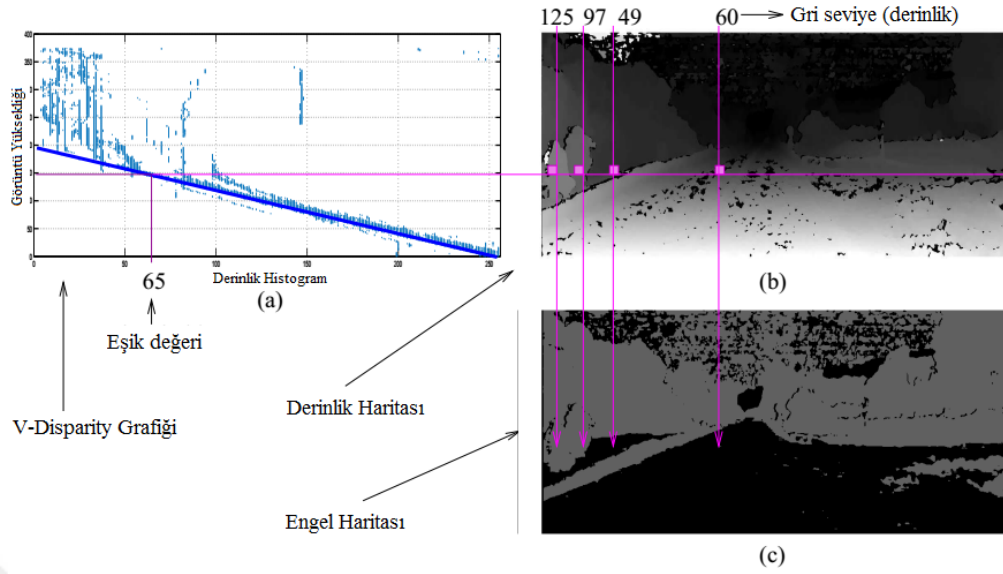
Zhao ve arkadaşlarının yaptığı çalışmada incelendiği üzere ' θ ' açısı ve yüksekliği değişiminde V-disparity '(b)'deki doğru açısı ve pozisyonu değişim göstermektedir [82]. Tez çalışmasında, ufuk noktasına yakın ' P_h ' noktası tahmin edilmesine karşın Şekil 3.29. (c)'de görülen hatalar ' P_c ' noktası tahmini ve atanmasıyla giderilerek Şekil 3.29. (d)'de daha doğru ve düzgün bir sonuç veren hale getirilmektedir. Engel tespitinde ' P_c ' tahmini değeri her görüntü için adaptif olarak hesaplanmaktadır. Sabit

bir değer kullanılması istendiği takdirde, ' P_c ' noktası tahmininde ağırlıklı aykırı değer filtresinden elde edilen değerler kullanılabilir. Bu takdirde yakında araç veya engel olmadan bir kere hesaplama yapılması durumunda, bu değer o araç ve parametreler için -yükselti değişimleri hariç- bütün görüntülerdeki engel tespitinde kullanılabilir. Fakat bu durum yükselti değişiminin olmaması varsayımını gerektirmektedir. Ayrıca derinlik verilerinden kaynaklı engel tespit hataları, kameranın bulunduğu noktada eğim değişiminin $[-2, +4]$ tolerans aralığında tutulmasıyla giderilmiştir.



Şekil 3.30. Regresyon doğrusu ve tolerans bölgeleri ile hesaplanan ufuk noktası (P_h) ve kamera noktası (P_c) tahmini üzerinden yer düzlemi (turuncu doğru) tahmini.

Şekil 3.30.'da görüldüğü gibi regresyon eğrisi ve yer düzlemi ufuk çizgisine gittikçe eşitlenmektedir. Kameraya yaklaştıkça da kamera yüksekliğine göre farklılık göstermektedir. Kamera yüksekliği, maksimum noktalar arasında kalan minimum nokta ve yükseklik tahmini değeri tolerans kadar alınarak tespit edilir. ' P_h ' ile ' P_c ' arasından geçen doğru altında kalan alanlar Şekil 3.31.'de görüldüğü gibi yer düzlemini belirterek üzerinde kalan alanlar engel olarak tanımlanır.



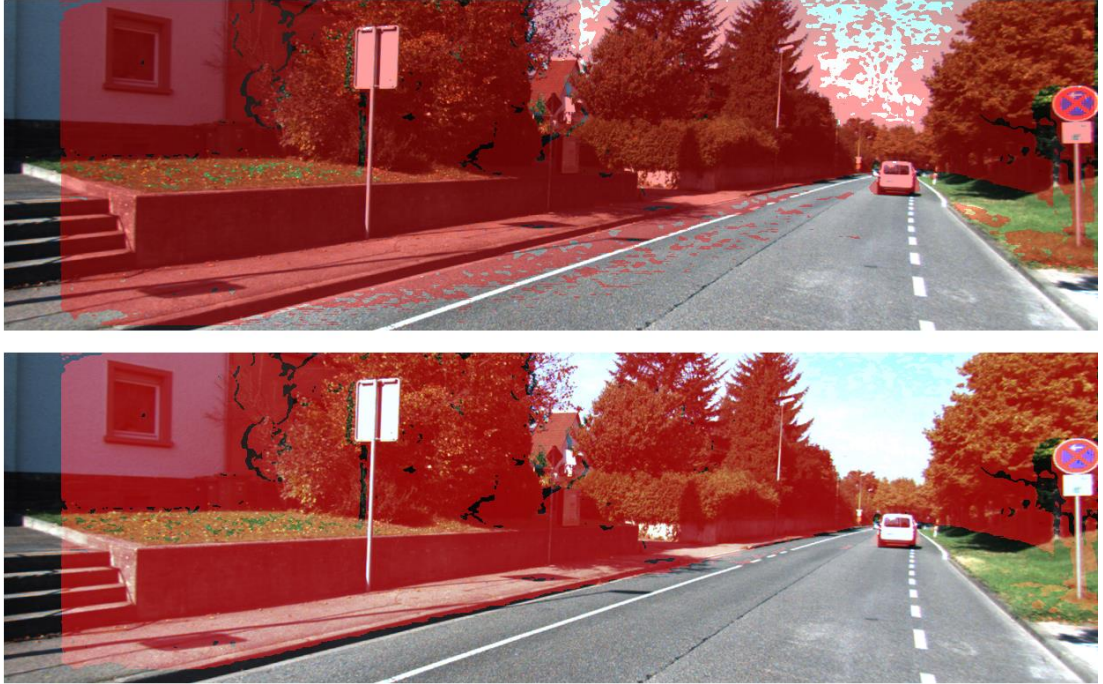
Şekil 3.31. Yer düzlemini belirleyen doğruyu kullanarak derinlik haritasından engel haritası çıkarım işlemi: (a) V-disparity üzerinde yer düzlem doğrusunun belirlenmesi, (b) Derinlik haritası üzerinde yer düzlem doğrusunun altında ve üstünde kalan bölgelerin tespiti, (c) Yer düzlem doğrusu üzerinde kalan derinlik bölgelerinin engel olarak belirlenip engel haritasının oluşturulması.

3.3.3. Engel tanıma sistemi test ve analizi

Engel tanıma sisteminde; tahmini yer düzlemi doğrusu, önceki aşmada elde edilen regresyon ve tolerans bölgesi kullanılarak elde edilen engel verileri birlikte karşılaştırmalı olarak Şekil 3.32. ve Şekil 3.33.'de ifade edilmiştir. Şekillerde belirtilen üst görüntü ile alt görüntü farkları, kamera yükseklik tahmininin etkisini göstermektedir. Sonuç engel haritası değişimlerine bakıldığında kamera yüksekliğinin hesaba katılması yer düzlemindeki hataları azaltmaktadır. Yer düzlemi tahmininde yer düzlemi doğrusu lineer olarak ele alınmıştır. Bu sebeple yükselti değişimleri çalışmada engel olarak ortaya çıkabilmektedir. Yükselti değişimlerinin de engel tanıma sistemine dahil edilmesi için, sunulan yöntem U-disparity verileri üzerinden geliştirilebilir.



Şekil 3.32. 2011_09_26_drive_0070 seti 355'nolu görüntüde; kamera etkisi hesaba katılmadığında elde edilen engeller (üst) ve önerilen yer düzlemi tahmini ile elde edilen engel (alt) görüntüleri.



Şekil 3.33. 2011_09_26_drive_0070 seti 32'nolu görüntüde; kamera etkisi hesaba katılmadığında elde edilen engeller (üst) ve önerilen yer düzlemi tahmini ile elde edilen engel (alt) görüntüleri.

Şekil 3.34.'te KITTI veri setindeki görüntüler üzerinde yapılan engel tespit sistemi sonuçları görülmektedir. Şekil 3.34.'te alttan ikinci görüntüde, ufuk noktasına yaklaştıkça yolda görülen hatalar yer düzleminin V-disparity üzerinde yer düzlemi doğrusunun lineer biçiminde ifade edilmesinden kaynaklanmaktadır. Engeller arası görülen boşluklar ise derinlik haritası elde edildiğinde derinlik haritasında bulunan hatalı değerlerden ve uzaklıkların engel haritasında gösterilmemesinden kaynaklanmaktadır. Engel haritasında uzaklıkların etkisinin hesaba katılması ve daha yüksek başarımda derinlik verisi elde edilmesi bu hataları azaltacaktır.





Şekil 3.34.KITTI-2011_09_26_drive_0070 görüntü setindeki sırasıyla 3-72-156-234-414 nolu görüntülerde engel tanıma algoritması test sonuçları.

Sonuç olarak tez kapsamında, içerisinde araca zarar verebilecek nesnelerin veya yayaların bulunduğu engeller, stereo görüntüler kullanılarak tespit edilmiştir. Tez içerisinde alt sistem olarak oluşturulan engel tanıma yöntemi; araç ve yaya güvenliği kapsamında kazaya sebebiyet verebilecek nesnelerin tespitinde, uzaklık tabanlı yaya tespitinde ve aracın gidebileceği serbest bölgenin belirlenebilmesi için kullanılmaktadır.

3.4. Yaya tespit sistemi: HOG filtresi uygulaması

Yaya tanıma sistemi olarak tez çalışmasında HOG filtresi kullanılmıştır. HOG insan tanıma filtresi, Dallal ve Triggs tarafından geliştirilmiştir [85]. Tez kapsamında HOG özellik çıkarımı kodlanarak test edilmiştir. HOG filtresi, temel olarak görüntü üzerindeki eğimlerin (gradient) hesaplanması ve yapay zekâ teknikleriyle sistemin eğitilmesi sonucu verilen girdi üzerinde insan ya da nesne olup olmadığını ölçmeye yönelik bir algoritmadır. Tez çalışmasında, yaya tanımlama için kullanılan bu yöntemin esasını oluşturan özellik çıkarımı kısaca Bölüm 3.4.1.'de ifade edilmiş, Bölüm 3.4.2.'de ise filtre eğitiminde izlenen yol ve elde edilen başarımlar açıklanmıştır. Temel olarak algoritmanın basamaklarını aşağıdaki listedeki gibi tanımlayabiliriz.

1. Normalizasyon
2. Eğimlerin hesaplanması
3. Vektörel eğimlerin belirlenmesi ve ağırlıklı olarak tanımlanması
4. Uzamsal blokların normalize edilmesi
5. Tanımlanan pencere üzerindeki HOG verilerinin toplanması
6. Lineer Destek Vektör Makinesi ile sistemin eğitilmesi

3.4.1. HOG özellik vektörü çıkarımı

Özellik çıkarımı kısmında algoritma için önemli olan eğim (gradient) değerleri, iki boyutlu görüntü için Denklem 3.31'de görüldüğü gibi parçalı türevlerin toplamı kullanılarak hesaplanmaktadır. Eğim değerinin hesaplanmasıyla farklı ışık koşulları

altında aynı kenar veya bölgenin ifade değerlerinin sabit kalması sağlanmaktadır. Kısacası eğim değerleri, özellik değerlerinin temsilinin ışık değişiminden etkilenmemesi için kullanılmaktadır. Ancak ışığın aynı bölgeye farklı oranlarda düşmesi durumunda bu değerler farklılık gösterecektir. Görüntüde eğim vektör değerleri, 'x' ve 'y' eksenlerine göre parçalı türevlerini alınarak Denklem 3.31'deki gibi hesaplanmaktadır. Eğim vektörünün büyüklüğü Denklem 3.32, açısı ise Denklem 3.33 ile hesaplanmaktadır.

$$\nabla a = \frac{\partial a}{\partial x} \vec{t}_x + \frac{\partial a}{\partial y} \vec{t}_y = (h_x \otimes a) \vec{t}_x + (h_y \otimes a) \vec{t}_y \quad (3.31)$$

$$|\nabla a| = \sqrt{(h_x \otimes a)^2 + (h_y \otimes a)^2} \quad (3.32)$$

$$\Psi(\nabla a) = \arctan \left\{ \frac{(h_y \otimes a)}{(h_x \otimes a)} \right\} \quad (3.33)$$

Görüntüde özellik temsili için hücreler belirlenerek bütün hücreler için eğim değerleri hesaplanır. Bu sayede tanımlanan hücreler topluluğu olan bloklar için eğim değerleri tanımlanmış olur. Bu eğim değerlerinin hesaplanması için bilgisayar ortamında temel türevsel filtreler Denklem 3.34 ve Denklem 3.35'teki gibi oluşturulur.

$$[h_x] = [h_y]^t = [1, (-1)] \quad (3.34)$$

$$[h_x] = [h_y]^t = [1, 0, (-1)] \quad (3.35)$$

Denklem 3.34 ve Denklem 3.35'te görülen filtre değerleri ile görüntü verisinin konvolüsyonu alındığında, eksenlere göre eğim değerleri elde edilmiş olunur. Sonrasında HOG özellik vektörü için elde edilen eğim değerleri, her blok için birleştirilerek açısal olarak histogramları elde edilmektedir. Açısal olarak histogram

alınırken hesaplanan eğim vektörlerinde açılar '180' dereceye göre modları alınarak bu açısal değerler, histogramda temsil edilmektedir. Bunun nedeni benzerlik ve farklılıkların histogramda daha net bir şekilde ifade edilmesi içindir.

Unutulmamalıdır ki bloklar hesaplandıktan sonrada hatanın azaltılması için histogram ambarları arası ağırlıklara göre üçlü interpolasyon değerleri hesaplanmaktadır. Ayrıca bloklar arasında da L1 veya L2 normalizasyonunun yapılması önerilmektedir.

Tez çalışması kapsamında oluşturulan kod gerçeklemede ve orijinal algoritmada, tanımlama çerçevesi olarak '64x128' bitlik bir görüntü penceresi kullanılmıştır. Bu tanımlama penceresi, toplam '105' bloktan oluşan '7' blok genişliğinde ve '15' blok yüksekliğinde bir yapıdır. Her blok toplam '9' kanallı '4' hücreden oluştuğundan her blok toplam '36' değer taşımaktadır. Bu yapı, vektör boyutu '7x15x4x9'luk toplam '3780' uzunluğunda bir özellik vektörünü temsil eder. Her blok, hücreler üzerinde '%50' oranında işlenen pencere boyutunun kaydırılmasıyla oluşur.

'64x128' çözünürlüğünde bir görüntü üzerindeki tanımlama blokları, hücrelerin '2x2'lik birleşiminden oluşmaktadır ve kelime çantalama (bag of words) denilen yöntem ile '3780' adet özellik barındıran bir özellik vektörü haline getirilmektedir. Sonrasında elde edilen bu özellik vektörü, SVM sistemine eğitim veya test için gönderilmektedir.

3.4.2. HOG özellik vektörü ile SVM eğitimi

HOG özellik vektörü çıkarımından sonra elde edilen veriler, bir yapay zekâ sınıflandırma yöntemi olan lineer SVM ile eğitilerek sistem testleri yapılmıştır.

Bu bölümde kullanılan eğitim ve test sonuç değer tabloları aşağıda belirtilen kısaltmalar kullanılarak eğitilen SVM sistemlerinin başarısı belirtilmektedir.

Bu kısaltmalar:

- TPG: Toplam Pozitif Görüntü
- TNG: Toplam Negatif Görüntü
- PEG: Pozitif eğitim görüntü sayısı
- NEG: Negatif eğitim görüntü sayısı
- PTG: Pozitif test görüntü sayısı
- NTG: Negatif test görüntü sayısı
- GP: Gerçek pozitif (True Positive)
- YP: Yanlış pozitif (False Positive)
- GN: Gerçek negatif (True Negative)
- YN: Yanlış negatif (False Negative)
- PTD: Pozitif Tahmin Değeri (%)
- NTD: Negatif Tahmin Değeri (%)

Yukarıda belirtilen bu kısaltmalarla tanımlanan veriler, sınıflandırma başarısı ölçümünde olduğu kadar kümeleme ve renk segmentasyon metotlarında da başarımlar ölçümünde kullanılmaktadır.

Yaya filtresi SVM eğitimi noktasında, INRIA yaya veri setinin internet sitesi üzerindeki başarımlar oranlarının incelenmesi için uygulanan yöntemler ile başarımlar oranı değişimleri test edilmiştir [87]. Başarımlar oranlarını etkileyen zorlu negatif görüntülerin ele alınış biçimi yine bu sitede belirtilmiştir. Yaya veri setleri incelendiğinde (INRIA vb.) pozitif veriler, sabit ve düşük çözünürlüklü (64.128,70.134,96.160 gibi) yaya barındıran görüntülerden oluşurken negatif görüntüler, yaya barındırmayan değişken ve yüksek çözünürlüklü (437x292, 320x240, 648x432, 640x480 gibi) görüntülerden oluşmaktadır. Bu aşamada, filtre için kullanılan boyuta göre negatif eğitim ve test görüntüleri, eşit çözünürlükler kullanılarak parçalanıp eğitime ve teste tabi tutulmuşlardır. Zorlu negatif görüntülerin INRIA yaya veri setinin internet sitesinde belirtilen ele alınış biçimi; öncül SVM eğitim setleri oluşturularak bu setlerin zorlu negatif görüntülerinin tespit edilip çıkarılmasıyla piramit şeklinde daha yüksek başarımlar sağlayan HOG filtresi

elde edildiği belirtilmiştir. Bu yöntemin tez çalışması süresince yapılan testlerde gerçek pozitif sonuçların tespit oranını artırdığı gözlemlenmiştir.

Daimler veri seti, INRIA ve MIT yaya veri setleri birleştirilerek eğitim yapıldığında elde edilen test sonucu pozitif tahmin değerinin (PTD) '%75' civarı olduğu Tablo 3.2.'de görülmektedir. Eğitim ve testler sırasında Daimler, MIT ve INRIA dışında ek eğitim görüntüleri katılarak yaya tanıma sistemlerinin başarımları ölçülmüştür. Eğitimlerde hangi veri setlerinin kullanıldığı Tablo 3.2. ve diğer tablo başlıklarında ifade edilmiştir.

Tablo 3.2. Daimler, INRIA ve MIT yaya veri setleri kullanılarak yapılan SVM eğitim ve test sonuçları.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
1	16786	27424	15660	18800	1126	8624	850	276	8542	82	75.48	99.04

Veri setleri içerisinde tanımlı test görüntülerini kullanarak sadece INRIA ve MIT veri setleri ile eğitim yaptığımızda, başarı oranının %90 üzerine olduğu Tablo 3.3.'de görülmektedir.

Tablo 3.3. INRIA ve MIT yaya veri setleri kullanılarak yapılan SVM eğitim ve test sonuçları.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
2	4466	27441	3340	13720	1126	8232	1040	86	8195	37	92.36	99.55

Daimler, INRIA ve MIT veri setleriyle eğitim için pozitif görüntüler ile zorlu negatif görüntülerden arındırılmış negatif görüntüleri kullandığımızda, oluşturulan HOG filtresi başarımlar değerleri Tablo 3.4.'te görülebilmektedir.

Tablo 3.4. Daimler, INRIA ve MIT yaya veri setlerinden zorlu negatif görüntülerin çıkarılması ile elde edilen SVM sistemi eğitim ve test sonuçları.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
3	20126	27424	19000	18800	1126	8624	1081	45	8479	145	96.00	98.31

INRIA veri setinin bulunduğu web sitesinde görüldüğü gibi, öncül HOG filtresi ile zorlu negatifler elimine edilmesi durumunda Şekil 3.35.'deki tarzda görüntüler

eđitime katılmadıđından, bu grntlerin HOG filtresine iletilmesi durumunda sonular yanlış pozitif olarak elde edilecektir.



Şekil 3.35. Zorlu negatif grnt rnekleri.

Şekil 3.35.'deki zorlu negatif grnt rnekleriyle Şekil 3.36.'da grlen zorlu pozitif rnekler ele alındıđında, eđitimde eđim benzerliklerinden kaynaklanan performans dşmne neden olan faktrler incelenebilir.



Şekil 3.36. Zorlu pozitif grnt rnekleri.

Tablo 3.5.'den grlebileceđi zere ncl SVM eđitimlerinde, eđitim grnt verileri toplanarak yarısı ncl test iin seilmiř, bu sayede eđitim seti grntleri ierisindeki zorlu grntlerin ayıklanmıřtır. ncl filtre oluřturulması ařamasında veri setlerinin test grntleri kullanılmamıřtır. Bylece zorlu negatif grntlerin ıkarılmasının sistem zerindeki etkisi grlebilmektedir. ncl filtrelerin kullanımında, Tablo 3.5.'den grldđ gibi piramit řeklinde ařamalı bir eleme yntemi kullanılmıřtır. nceki testlerde elenmemiř grntler, bir sonraki eđitimde kullanılmaktadır.

Tablo 3.5. INRIA ve MIT yaya veri setleri üzerinde öncül SVM eğitimleri ve test sonuçları.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
4	4466	18800	3340	9400	1126	9400	1045	81	9299	101	92.80	98.92
5	4466	18699	3340	9349	1126	9349	1063	63	9277	72	94.40	99.23

Zorlu örnekler çıkarıldıktan sonra uygulanan eğitimle elde edilen sonuç HOG filtresinde, veri tabanlarının test görüntüleri üzerinden elde edilen sonuçlar Tablo 3.6.'dan görülebilmektedir.

Tablo 3.6. Öncül filtreleme tasarımından sonra elde edilen sonuç SVM filtresi eğitim ve test değerleri.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
6	20126	27250	19000	18626	1126	8624	1092	34	8455	169	96.98	98.04

Daimler, INRIA ve MIT yaya veri setlerine ek olarak tez çalışması kapsamında elde edilen görüntüler de sistemin geliştirilmesi amacıyla eğitim sürecine dahil edilmiştir. Bu noktada veri tabanlarının test görüntüleri üzerinde değişiklik yapılmadığından, eğitim sürecinde kullanılan bu görüntülerin test sonuçlarına etkisi daha doğru bir şekilde görülebilmektedir. Eğitim sürecine eklenen gerçek hayat görüntülerindeki zorlu negatif örneklerinin elde edildiği öncül SVM eğitim test sonucu Tablo 3.7. üzerinden, elde edilen en son HOG filtresinin eğitim ve test sonuçları ise Tablo 3.8. üzerinden görülebilir.

Tablo 3.7. Yaya veri setlerine, ek eğitim görüntülerinin eklenmesinden sonra oluşturulan öncü filtre sonucu.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
7	4466	26946	3340	13473	1126	13473	1055	71	13376	97	93.69	99.28

Tablo 3.8.'de sonuçları verilen HOG filtresi, tez kapsamında yaya tanıma için kullanılmıştır. Yeni eğitim görüntülerinin eklenmesi, pozitif doğru insan bulma oranını düşük oranda azaltmasına karşın, negatif görüntülerin doğru bulunma oranını daha yüksek oranda artırmıştır.

Tablo 3.8. Yaya veri setleri ve gerçek hayat görüntülerinin eklendiği SVM eğitim ve test sonuçları.

Test No	TPG	TNG	PEG	NEG	PTG	NTG	GP	YP	GN	YN	PTD	NTD
8	20126	35474	19000	26850	1126	8624	1089	37	8483	141	96.71	98.36

HOG filtresi eğitiminde, veri sayısının çokluğu sistemin daha iyi sonuç vereceğini garantilemediği görülmüştür. Ayrıca veri setlerindeki eğitim görüntüleri dışında gerçek hayat görüntülerinin de başarıyı artırabileceği gözlemlenmiştir. Tez çalışmasında INRIA, MIT, Daimler veri setleri ve ek görüntülerle eğitilen filtre, sistem genelinde kullanılmaktadır.

HOG filtresiyle yapılan yaya tanıma çalışmalarında görüldüğü kadarıyla negatif görüntüler, gökyüzü dahil olarak pencerelere bölünerek eğitime dahil edilmektedir. Pozitif veriler ise sabit çözünürlükte ve yayanın merkez noktada olduğu görüntülerle oluşturulmuştur. Bu sebeple doğru bulunan pozitif ve negatif görüntülerin, bütün görüntü ele alındığında yüksek başarımlı sonucu gösterdiği eğitim ve test verilerinden görülebilmektedir. Eğim (gradient) verilerinin yüksek değişim gösterdiği bölgelerde (yol, ağaçlar, tabela ve işaretler) ise HOG filtresinin, eğitim ve test görüntülerine oranla daha fazla hatalı sonuç alındığı gözlemlenmiştir. Tez kapsamında sistemin genel olarak yaya tespiti noktasında başarımları, araştırma bulguları bölümünden incelenebilir.

SVM ile oluşturulan HOG filtresinde dikkat edilmesi gereken en önemli özellik; SVM eğitiminde negatif eğitim görüntülerinin, yüksek çözünürlükteki resmin pencerelere bölünerek oluşturulması ve sonrasında bu görüntülerin eğitime gönderilmesidir. Bu yöntemle eğitilen sistemde, gökyüzü gibi yayaların bulunamayacağı bölgeler de eğitime katılmış olmaktadır. Bu nedenle negatif tahmin başarımlı değerleri, pozitif tahmin başarımlı değerlerine oranla daha yüksek çıkmaktadır. Fakat burada önemli olan gökyüzü gibi çok fazla eğim (gradient) değeri taşımayan bölgelerin de eğitime yüksek oranda dahil olmasından dolayı, elde edilen sistemde değişim gösteren bölgeler filtrelendiğinde sonuç pozitifliğe doğru eğilim göstermektedir. Ayrıca öncül filtrelerle zorlu örneklerin çıkarım yöntemi de başarımlı

oranını artırırken sistemin yatkınlığını pozitif değer çıkarımına yaklaştırdığı tez çalışması dahilinde yapılan testlerde gözlemlenmiştir.

Yaya filtresinde eğitim, zorlu görüntüler çıkarılmadan yapıldığında başarı oranı Tablo 3.2.'deki gibi yüksek oranda düşmekte, çıkarıldığında ise yüksek değişim gösteren bölgelerin çoğunlukla pozitif sonuç verdiği gözlemlenmiştir. Bu şekilde eğitilen sistem, özellikle yayaların bulunabileceği alanlarda Şekil 3.37.'deki ilk dört görüntü gibi yanlış pozitif sonuçlar verebildiğinden, insan dışındaki -ağaç, çit, direk gibi- nesneler de insan olarak algılanmaktadır. Tez çalışmasında, uzak noktalarda da yayalar incelendiğinden, yapılan oransal büyütme ve dönüşüm işlemleri sonrasında elde edilen filtre sonuçları, hatalı olarak yanlış pozitif sonuçlar verebilmektedir.



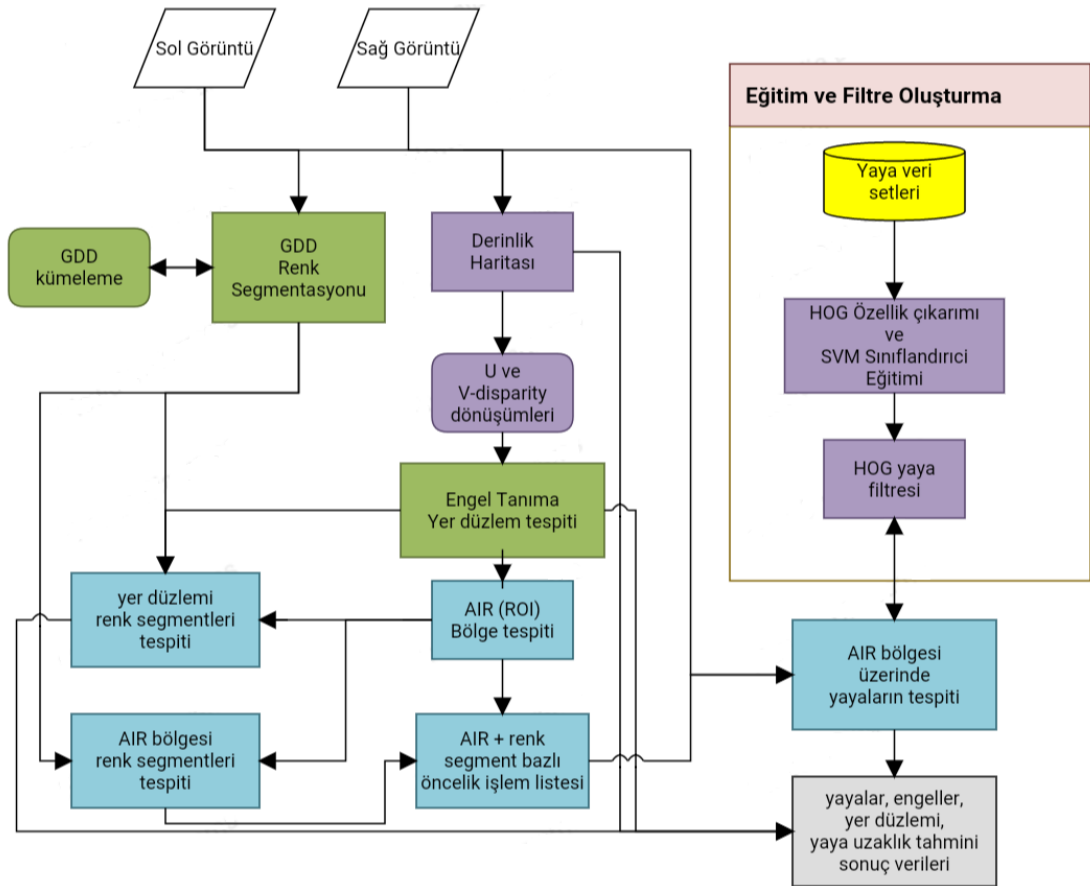
Şekil 3.37. Örnek test görüntüsü üzerinden HOG filtresi pozitif yaya görüntü sonuçları.

Tez kapsamında yayaların tespitinde, yayaların bulunamama durumu oluşmasındansa yanlış pozitif alanlarının bulunması tercih edilmektedir. Sonuç olarak, çalışmada hedef çerçevesinde HOG filtresinin eğitimiyle veri seti test örnekleri üzerinde %90 üzeri pozitif tahmin değeri elde edilmiş ve üst sistemde uygulanmıştır.

3.5. Temel sistem parçalarının ilişkilendirilmesi

Önceki başlıklarda tez kapsamında görüntüler üzerinde, renk bölgeleri tespit edilmiş; engeller ve yer düzlemi tanımlanmış; yaya tanıma algoritması gerçekleştirilmiştir. Bu bölümde ise bahsi geçen yöntemler alt sistem olarak ele alınarak üst sisteme nasıl entegre edildiği anlatılmıştır.

Çalışmanın tamamını kapsayan üst sistemin blok diyagramı, Şekil 3.38.'de görüldüğü gibidir. Sistemin daha yüksek doğrulukta çalışması ve verilerle aracın gittiği ortamın daha net tanımlanması için alt sistem blokları arasındaki ilişkileri kuran önemli metotlar bu bölümde ortaya konulmuştur. Bu metotlar, aşağıdaki Şekil 3.38.'de mavi renkle gösterilmiştir.



Şekil 3.38. Ayrıntılı sistem blok diyagramı.

Şekil 3.38.'de paralelkenar veri giriş blokları, sağ ve sol stereo kamera görüntü girdi verilerini gösterirken; sarı renkle gösterilen veri tabanı bloğu ise yaya tespit sistemi için sınıflandırıcının eğitiminde kullanılan pozitif ve negatif yaya görüntülerini kapsamaktadır. Mor renkli bloklar, literatürde bilinen metotlar kullanılarak oluşturulan ve üzerlerinde yapılan çalışmaları temsil etmektedir. Bu metotlar; derinlik haritası tespiti, UV-disparity dönüşümleri, HOG filtresi ve sınıflandırıcıları

kapsamaktadır. Bütün bunlar sistem için uygun hale getirilmiş ve uygulamalarla birlikte test edilmiştir.

Yeşil renkli bloklar, tez aşamasında temel oluşturan ve yenilik içeren algoritmaları temsil etmektedir. Bu bloklarda yeni bir kümeleme metodu ve renk segmentasyonu yöntemi ortaya konulmuştur. Ayrıca engel tanıma ve yer düzlemi tespiti noktasında daha iyi sonuç veren başarıyı yüksek bir metot geliştirilmiştir.

Bu bölümde incelenen mavi renk bloklar alt sistem elemanları kullanılarak sonuca ulaşılabilmesi için gerçekleştirilen algoritma ve adımları ifade etmektedir. Bu adımlarda uzaklık ve renk tabanlı öncelik listesiyle yaya tanımlama önceliği oluşturulmuş, çevresel bölgelerin belirlenmesi ve engellerin nesne olarak tespiti için gerekli aşamalar kodlanmıştır. Gri renkli arka planla belirtilen blok ise bütün sistem çıktılarını tanımlamaktadır.

Şekil 3.38.'de alt sistemler ilişkilendirilirken kullanılan metotlar (mavi bloklar), aşağıdaki listedeki şekilde tanımlanarak alt başlıklarda incelenmiştir.

- AIR (Above Immediate Road) bölgesinin bulunması
- Alan Genişletme ile yol üzerindeki nesnelerin bulunması
- Yer düzlemi ve AIR bölgelerinin renk segmentasyonu ile ayrılması
- Uzaklık ve renk tabanlı öncelik listesi oluşturulması
- HOG filtresi ön-işlemleri
- Nesnelerin uzaklık tespiti

Yukarıda belirtilen listedeki bu işlemler, önceki aşamalarda elde edilen bilgilerin kullanımıyla istenilen çıktıları sağlayabilmek için çevresel bölgelerin daha ayrıntılı tanımlanmasını ve sınıflandırılmasını sağlamaktadır.

3.5.1. AIR (Above immediate road) bölge tespiti

V-disparity kullanılarak yer düzlemi ile engelleri belirleyen bölge hesaplandıktan sonra yayaların tespiti ve yol üzerindeki engellerin bulunması için yolun tam olarak üzerinde bulunan bölgelerin tespiti gereklidir. Bunun sebebi engeller bulunduğunda, gökyüzünde ve yolun üzerinde olmayan engellerin de hesaba katılmasıdır. Bir trafik lambası veya uçan bir kuşun yaya olup olmadığına dair teste tabi tutmak hem işlem zamanını artıracak hem de bu esnada yol güvenliği bakımından süreç olarak öncelikli bulunması gerekli engellerin tespitini geciktirecektir. Bu sebeple yol üzeri segmentlerin yola temas eden bölgelerinin bulunmasıyla, ROI (Region of Interest) bölge tespit işlemi yapılarak AIR (Above immediate road) yani yol üstü bölge tanımlaması yapılmıştır.

Öncelikli bölge tespitinde, bölüm 3.3.'de hesaplanan yol ile engelleri ayıran doğru baz alınmıştır. Bu doğru V-disparity'den görüntü düzlemine dönüştürülerek Şekil 3.39.'da görüldüğü gibi sınır çizgileri elde edilmiştir.



Şekil 3.39. Orijinal test görüntüsü (üst) ile Yer düzlemi ile engelleri ayıran doğru kullanılarak elde edilmiş bölge sınır çizgileri (alt).

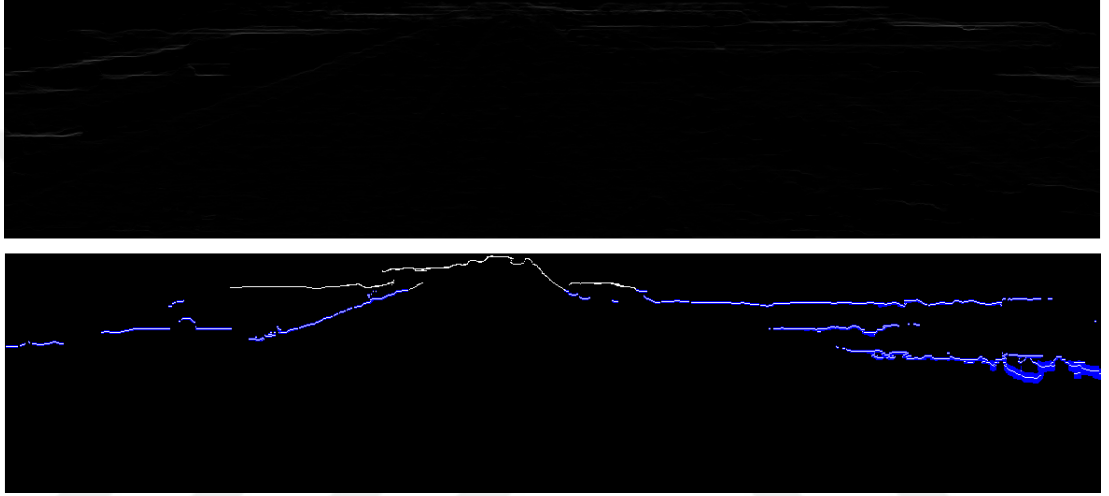
Iloie ve arkadaşlarının çalışmalarında belirttikleri gibi nesneler stereo kamera sistemine yaklaştıkça hacimsel olarak nesnelerin belirginliği de artmaktadır [84]. Yani uzaktayken düşük derinlik derecesine sahip nesne, yakına geldiğinde hacimsel özellikleri netleşerek uzaklık açısından da derinlik kazanacaktır. Örneğin uzaktayken insanın gövde, kolları ve elinde tuttuğu nesneler yaklaşık aynı derinlik bilgisinde tutulurken, yakına geldiğinde kollarının birbirleri arasındaki uzaklık farkı, ellerinde tuttuğu nesnelerin derinlik verisi farkı daha belirgin hale gelecektir. Bu sebeple Iloie ve arkadaşlarının U-disparity üzerinde segment belirlerken kullandıkları eşik değeri sistemine benzer bir eşik değeri sistemi uygulanmıştır. Ancak tez çalışmasında derinlik tabanlı hacimsel bölge tespitinde kullanılan eşik değerleri, bölgeleri dört parçaya ayırmak yerine derinlik bilgisine göre daha sık aralıklarla Tablo 3.9.'da gösterildiği gibi tanımlanmıştır. Bunun sebebi derinlik bilgisinin uzaklığa ters oranla daha net ve doğru ölçülebilmesidir.

Tablo 3.9. Derinlik tabanlı bölge hata payı eşik değerleri.

Derinlik	Hata Payı Eşik Değeri
0	0
5	0,25
10	0,5
30	1,5
40	2
80	2,5
120	10
240	18
256	18

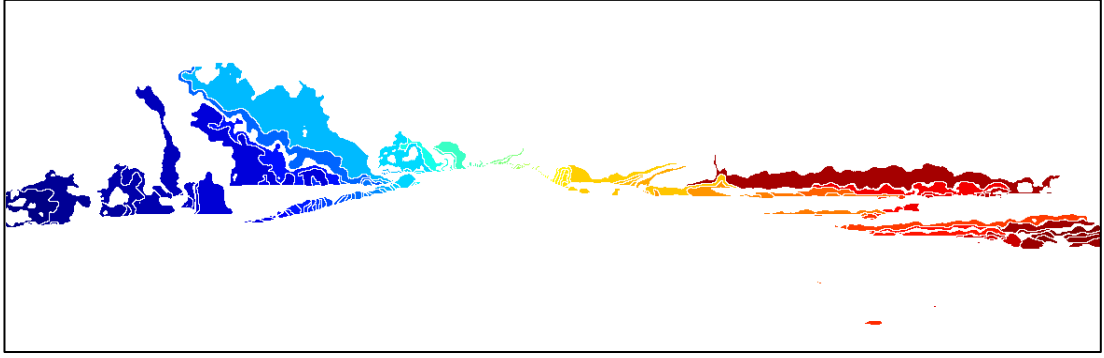
Tablo 3.9.'a bağlı değerler, derinlik haritası üzerinden nesnelerin yakınlıklarına göre kapladığı alanlara bakılarak ve yapılan testlerde ölçümlenerek belirlenmiştir. Bununla birlikte derinlik bilgisine bağlı hata oranları ve değişim grafikleriyle denklem haline getirilerek daha net ve ölçümlenebilir bir sistem geliştirilebilir.

AIR bölgelerinin U-disparity üzerinde ve derinlik haritasında kaplayacağı alanların tahmini büyüklükleri, Şekil 2.8.'den farklı olarak Şekil 3.40.'da Tablo 3.9. kullanılarak hem uzaklığa göre derinlik eşiği tanımlaması hem de sadece AIR bölgesinin incelenmesiyle, yol üzerindeki nesnelerin daha net bir şekilde tanımlanması sağlanmıştır. Bu veriler elde edilirken AIR sınır çizgisinin V-disparity'deki ifadesi kullanılarak U-disparity görüntüsündeki karşılığı hesaplanmıştır.



Şekil 3.40. Test görüntüsünden elde edilen U-disparity çıktısı (üst) ve buradan eşik değerlerine göre U-disparity'de AIR sınır çizgisi (beyaz) ve komşuluklarının gösterimi (mavi) (alt).

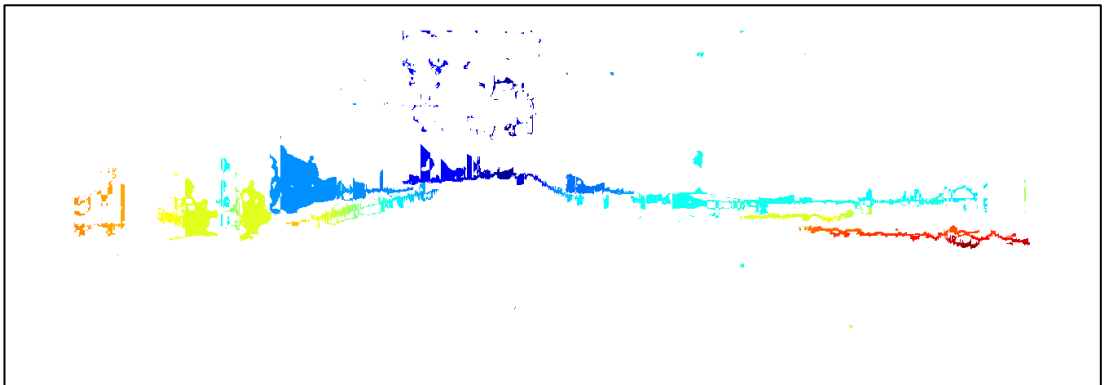
Şekil 3.40.'da alt görüntüde görülen beyaz renkli tanımlanan AIR sınır çizgisiyken Tablo 3.9.'daki tanımlı uzaklık değerlerine göre yol üstündeki segmentlerin bulunması için aranan hacimsel bölge ise beyaz ve mavi renkli bölgelerin toplamıdır. AIR bölgesi, beyaz ve mavi renkli bölgelerdeki komşulukların derinliklerine göre belirlenmesiyle elde edilen bölgeye denilmektedir. Alan genişletme metoduyla elde edilen AIR bölgesi Şekil 3.41.'den görülebilir.



Şekil 3.41. AIR bölgesi ve uzaklığa bağlı renksel etiketleri.

3.5.2. Alan genişletme ile yol üzerindeki nesnelerin bulunması

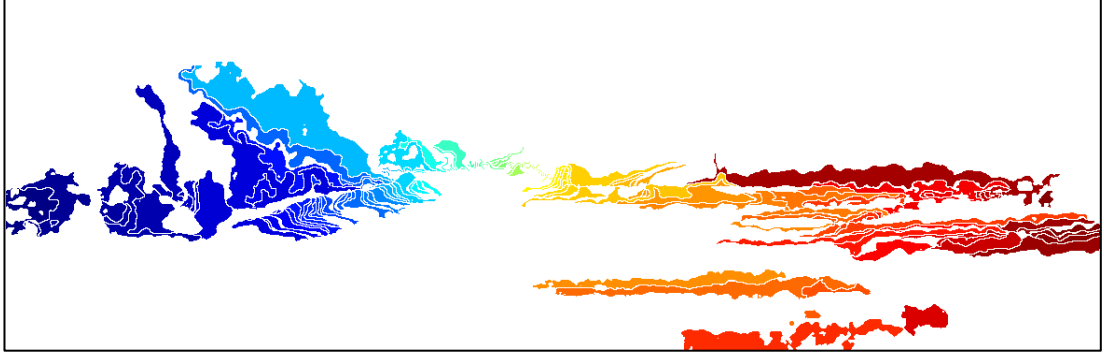
Başlangıçta AIR bölgesine göre derinliği belirlenen segmentlerin, uzunluk ve genişliklerinin tespiti için her satırda o sütun için belirlenen hata payı uzaklığındaki nesnelerin çıkarımı, Şekil 3.42.'deki gibi maskeleme yöntemiyle yapılmıştır. Fakat bu noktada, sadece yol üzerinde değil belirtilen uzaklığa sahip havadaki nesne ve hatalar da segment kapsamına dahil edildiğinden dolayı AG (Alan Genişletme) algoritması kullanılarak sadece AIR çizgisi üzerinde bulunan bölgeler tespit edilmiştir.



Şekil 3.42. Sütun tabanlı derinlik bilgisine göre yapılan maskeleme sonucu (renkler derinlik verisine göre değişim göstermektedir).

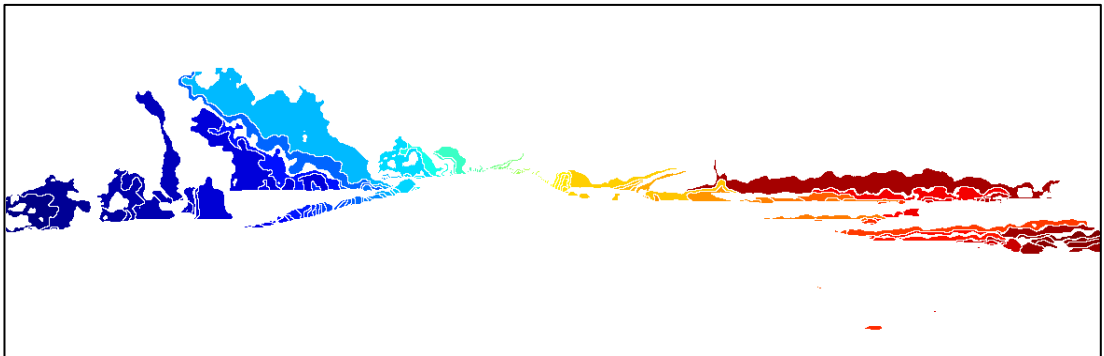
AG algoritması ile tespit edilen segmentler ise Şekil 3.43.'de görülebilir. Burada istenilen veriler Şekil 3.39.'deki AIR çizgisi üzerinde, önceden tanımlanmış Tablo

3.9.'daki deęerlere gre piksel komřuluklarına bakılarak, eřik deęerleri dahilinde bitişik blgelerin tanımlanması yapılmıştır.



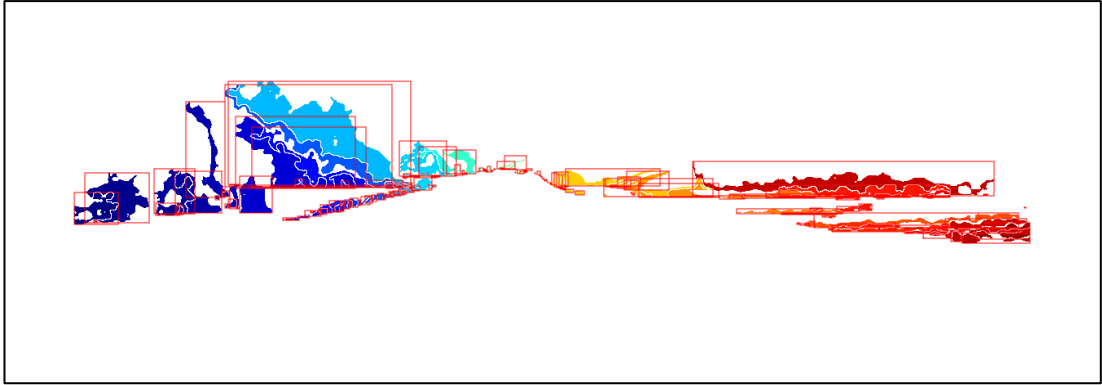
Şekil 3.43. Alan Geniřletme (AG) yöntemiyle elde edilen segmentler.

Fakat AG ile elde edilen görüntülerdeki her bir etikette Şekil 3.43.'den görüleceęi üzere yola bitişik alandaki nesneler, yer düzleminin AIR üzerindeki uzaklıęının nesneye uzaklıęıyla aynı olmasından trü aşırı-segmentasyon durumuna sebebiyet vermiştir. Bu durumu ortandan kaldırmak için AIR blgesi altında kalan yer düzlemi maskelenerek filtrelenmiştir. Elde edilen yol üzerindeki nesnelerin etiketleriyle tanımı Şekil 3.44.'den görlebilir.



Şekil 3.44. Filtrelenmiş AIR blgesi ve uzaklıęa baęlı renksel etiketleri.

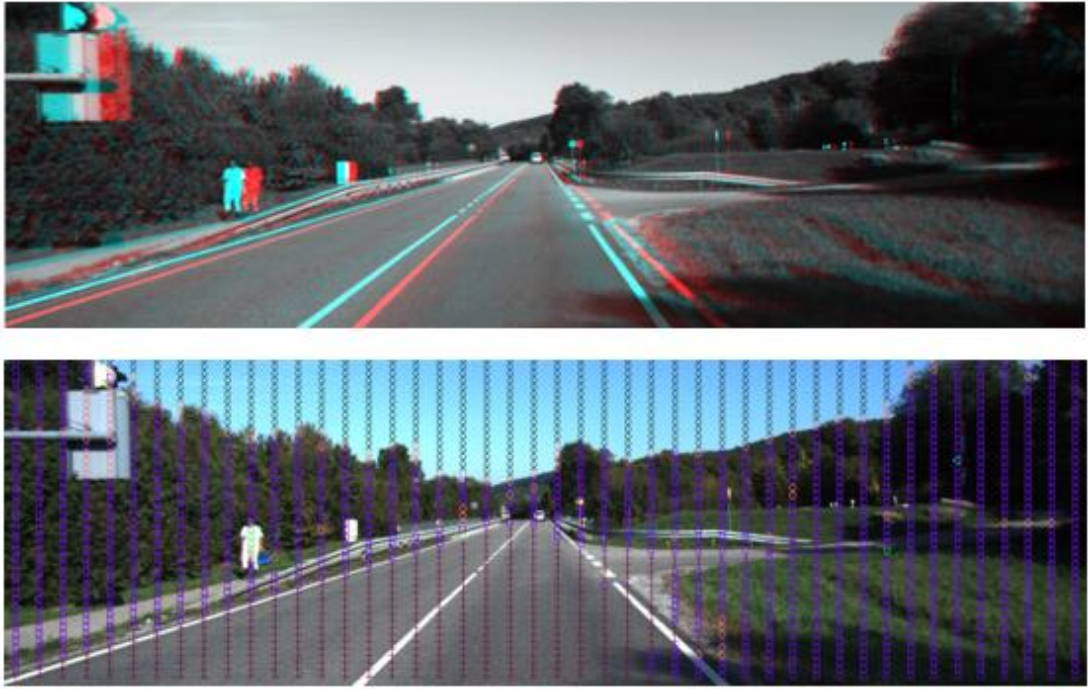
Şekil 3.44.'deki her bir nesne birbirine baęlı olduęu etiketi taşıması gerektięinden ve komřuluęu olmayan blgelerin ayrımının istenmesi nedeniyle BBA (Baęlı Bileřen Analizi) işleminin AIR segment görüntüsü üzerine uygulanarak yol üzerinde bulunan nesnelerin tanımlanması Şekil 3.45.'teki gibi saęlanmışır.



Şekil 3.45. AIR bölgesi segmentlerinin bağlı bileşen analizi çerçeve sonuçları.

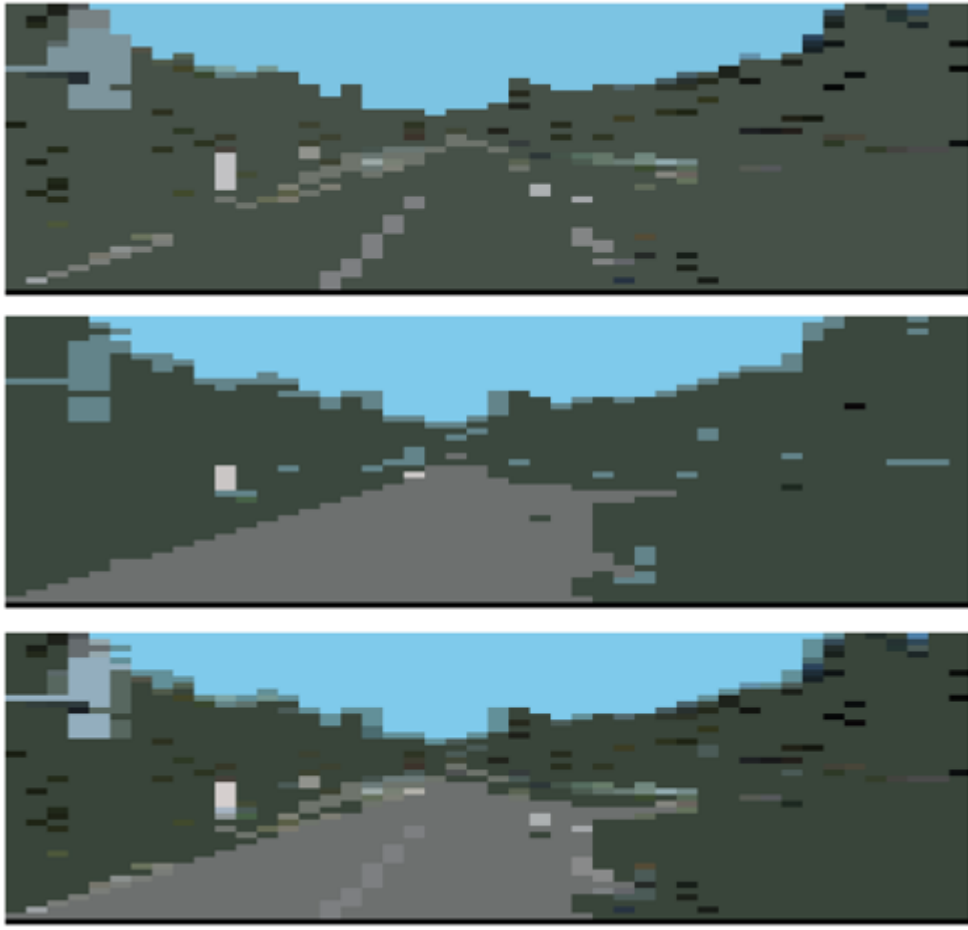
3.5.3. Yer düzlemi ve AIR bölgelerinin renk segmentasyonu ile ayrılması

Soquet ve arkadaşlarının yaptığı çalışmada, yer düzlemi bulunduğundan sonra renk ve doygunluk (Hue-Saturation) kullanılarak 2-boyutlu histogram üzerinden ISODATA algoritması ile küme oluşturularak, yer düzleminde renk bölgelerine göre sürüşün serbest olduğu alanları tanımlamışlardır [67]. GDD renk segmentasyonunda, HSV renk uzayının renk ve saturasyonla birlikte aydınlık değerleri de kullanıldığından, yer düzleminde gölge ve aydınlık alanlar birbirinden ayrılmasına karşın, araç sürüşüne uygun olan bölgelerin tespiti için kullanılabileceği görülmüştür. Ayrıca sadece renk ve doygunluk (Hue-Saturation) verileri kullanılarak da GDD renk segmentasyon bölgesi tespitiyle, ışığın sonuçlar üzerindeki etkisi azaltılabilir. Fakat sadece yol bölgesi değil bütün görüntünün segmentlere ayrılması gerektiğinden, bütün görüntü üzerinden HSV renk uzayının tamamı tez çalışması kapsamında renk segmentasyonu için kullanılmıştır.



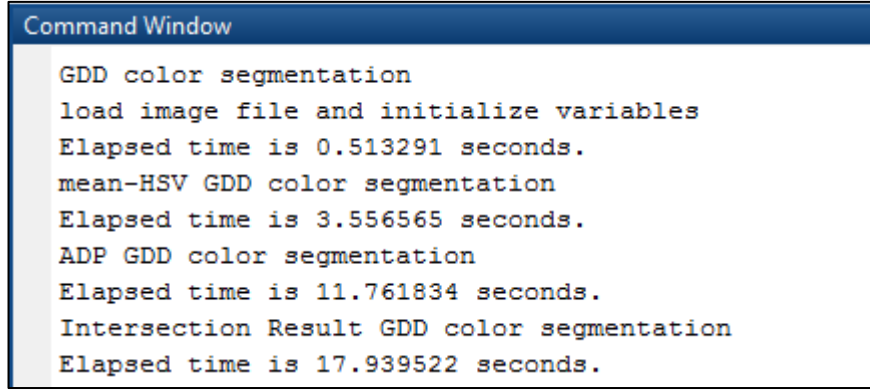
Şekil 3.46. Kırmızı-Camgöbeği (Yeşil + Mavi) kanallarında stereo giriş görüntüsü (üst), renkli sol giriş görüntüsü üzerinde GDD renk segmentasyon sonucu gösterimi (alt).

Şekil 3.46. üzerinde görüldüğü gibi bütün görüntüde elde edilen GDD renk segmentasyonu sonucunda; yol, ağaçlık alanlar ve gök yüzü yani alansal olarak çok yer kaplayan ve birbirine yakın renkleri barındıran bölgeler, doğru bir şekilde tanımlanmıştır. Renkli bölgeler tanımlanırken, çözünürlük düşürme yöntemi kullanıldığından bazı alanlarda özellikle renk bölgeleri değişiminin olduğu kenar bölgelerinde, aykırı segmentlerin oluştuğu görülebilir.



Şekil 3.47. GDD renk segmentasyon ara sonuçları; ADP+GDD (üst), MeanHSV+GDD (orta) ve sonuç segmentasyonu (alt).

GDD renk segmentasyonu içerisinde düşük çözünürlükteki renk bölgeleri görüntünün özelliklerine göre farklı iş parçacıklarında farklı sonuçlar vermektedir. Sürüş için uygun olan yol bölgesi ile engellerin, renk farklılıklarının daha iyi ortaya konması ve öne çıkarılması için Şekil 3.47.'de görüldüğü üzere farklı sonuç etiketleri kullanılabilir. Ayrıca daha hızlı sonuç alınmak istendiği takdirde, sadece belirli bir iş parçacığının koşulması ve diğer iş parçacığının da zaman kısıtlaması dışında belirli aralıklarla çalıştırılması uygun olacaktır.



```

Command Window

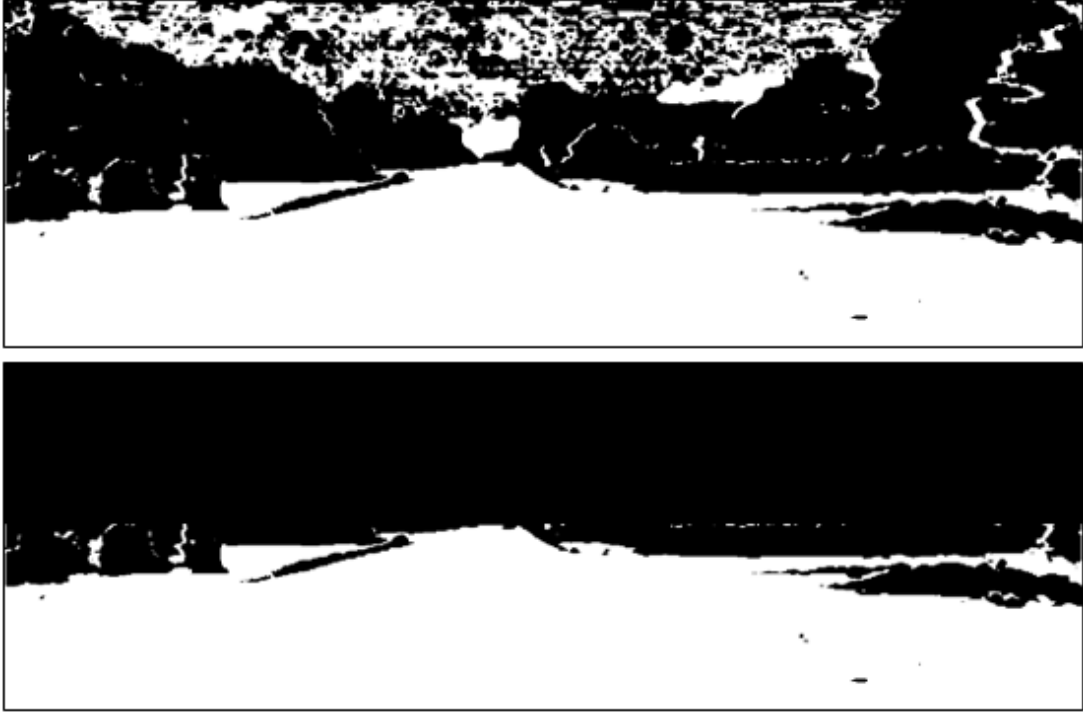
GDD color segmentation
load image file and initialize variables
Elapsed time is 0.513291 seconds.
mean-HSV GDD color segmentation
Elapsed time is 3.556565 seconds.
ADP GDD color segmentation
Elapsed time is 11.761834 seconds.
Intersection Result GDD color segmentation
Elapsed time is 17.939522 seconds.

```

Şekil 3.48. GDD renk segmentasyonu iş parçacıkları süreleri ve sonuç süresi (paralel iletişim kayıpları yok sayılmıştır).

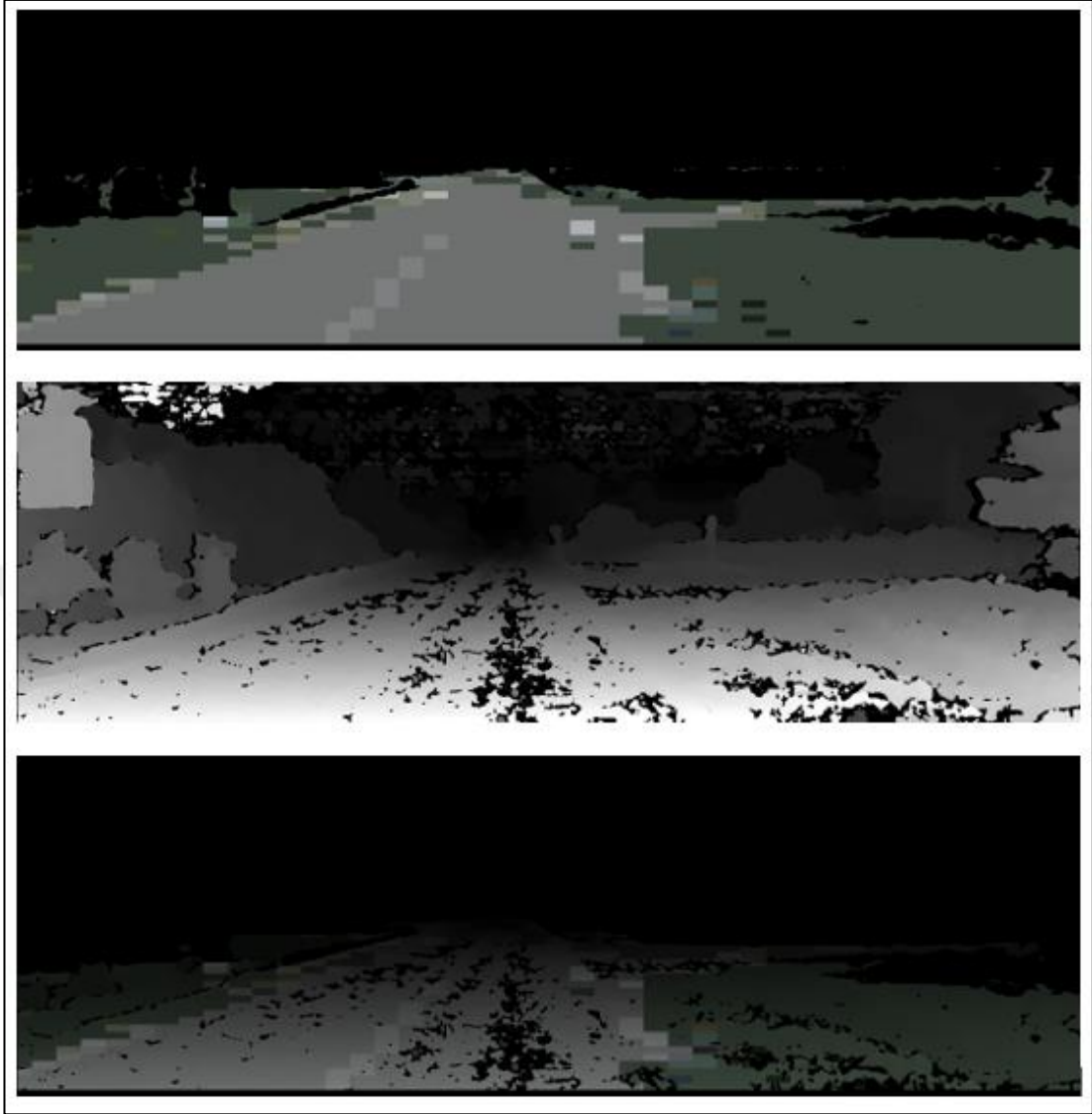
‘355’ nolu görüntüde Şekil 3.47.’de görülen sonuçlar elde edilirken, iletişim kayıpları yok sayılarak farklı alt renk segment iş parçacıklarının ve genel sürecin yaklaşık olarak ne kadar zaman gerektirdiği Şekil 3.48.’de ifade edilmiştir. Koşum süre gereksinimleri; kullanılan donanım, çözünürlük ve optimizasyona bağlı olarak değişecektir. Kullanılan gerçeklemede ADP+GDD renk segmentasyon iş parçacığı, yaklaşık 10 saniye farkla daha fazla süre gereksinimi duymaktadır. Bu süre azaltılsa da her zaman ortalama alınarak elde edilen görüntü renk bölgesi bilgisini kullanan iş parçacığından (HSV+GDD) daha fazla olacaktır.

GDD renk segmentasyonu ile elde ettiğimiz bölgeler ile yer düzlemi maskelenerek yer düzlemi içerisinde sürüşe müsait alanların belirlenmesi sağlanabilmektedir. Maskeleme işlemi öncesinde, engel barındırmayan bölgeler ve AIR bölgesi kullanılarak yer düzlemi bölgesi tespit edilmekte, sonrasında maskeleme işlemi uygulanmaktadır.



Şekil 3.49. Engel maskesi tersi (üst) ve AIR kullanılarak elde edilen yer düzlemi maskesi (alt).

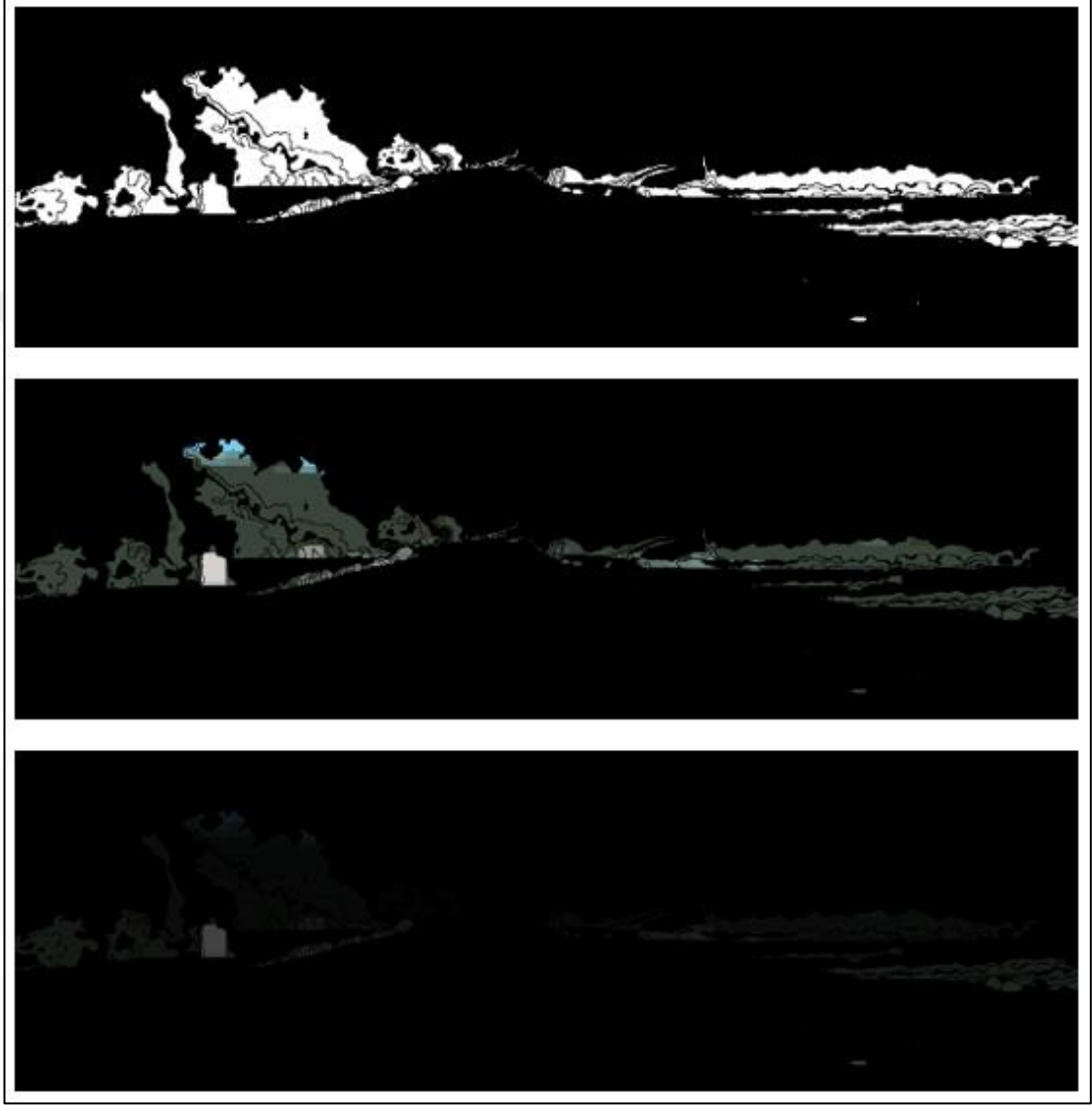
V-disparity ile tanımlanan engel dışındaki alanlar, Şekil 3.49.'da görüldüğü gibi ikili sayı sistemiyle temsil edilmiştir. Beyaz ile temsil edilen alanlar engel barındırmayan bölgeleri göstermektedir. Alttaki yer düzleminde engel barındırmayan bölge maskesi; üstteki maskede AIR bölgesi altında kalan noktalar kullanılarak tespit edilmiştir. Bu sayede araç sürüş bölgelerinin belirlenebilmesi için maske işlemi tamamlanmış olur.



Şekil 3.50. GDD renk segmentasyonunun yer düzlemindeki görüntüsü (üst), Derinlik haritası (orta), Yer düzlemi renk bölgelerinin derinlik bilgisi eklendiğindeki görünümü (alt).

Şekil 3.49.'de alt görüntüde görülen yer düzlemi bölge verisi kullanılarak renk segment bilgisi maskelendiğinde, Şekil 3.50.'de üst görüntüde görülen yer düzleminin renklere ayrılmış etiketlerini elde etmiş oluruz. Ortadaki görüntüde belirtilen hesaplanmış derinlik verilerini kullanarak yer düzlemi renk etiketleri işlendiğinde, alt görüntüde görülen yer düzlemindeki renk bölgelerinin uzaklıklara göre dağılımı elde edilmiş olunur. Bu sayede özellikle aracın bulunduğu gri renkli yol dışında kalan farklı renkteki bölgeler, sürücü için yol dışı alanları tanımlamaktadır. Derinlik verileri üzerindeki hataların giderilmesi için daha yüksek doğrulukta derinlik verisi bulan algoritmalar kullanılabilir. Fakat bu seçim zamansal

açından daha fazla işlem gereksinimi ortaya çıkaracaktır. Renk verileri kullanılarak derinlik hata verisi barındıran alanların aynı etiketlerdeki sıfırı gösteren uzaklık verileri, yatay düzlem komşulukları ve benzerliklerine göre eşitlendiği takdirde daha yüksek başarılı sürüş bölgesi tanımlanmış olur.



Şekil 3.51. AIR bölgesi segment maskesi (üst), AIR bölgesindeki GDD renk segment verileri (orta), AIR bölgesinde GDD renk segmentlerinin uzaklık bilgisine göre değişimi (alt).

Yol dışında AIR bölgesi tanımlamasında bulunan ve engel olarak tespit edilen nesnelerin renksel farklılıkları, yaya bulma önceliğinde önem arz etmektedir. Şekil 3.51.’deki üst görüntüden görüleceği üzere AG algoritmasıyla tespit ettiğimiz AIR bölge maskesi kullanılarak Şekil 3.51.’de ortadaki görüntüde görülen AIR

bölgesindeki renk segmentleri tespit edilmiştir. AIR bölgesi renk segmentlerine uzaklık verisinin eklenmesiyle elde edilen Şekil 3.51.'deki alttaki görüntü, farklı nesnelerin belirlenebilmesi için kullanılan temel verileri temsil etmektedir. Bu veriler, uzaklık ve renk tabanlı öncelik listesinin oluşturulmasında kullanılmaktadır.

Verilen örneklerdeki GDD renk segmentlerinin AIR üzerindeki durumlarına dikkat edilirse, yaya farklı renk kıyafeti nedeniyle arka plandaki yeşil renkli segmentlerden ayrılmıştır. Uzaklık bilgisine göre filtrelenmiş görüntüsünde ise renklerin derinlik bilgisi dahilinde dağılımı incelenebilir. Özellikle arka plandan farklı renkteki nesneler, yaya ve araç kazalarını önlemede önemli bir rol oynamaktadır. Ayrıca aracın seyir halinde kendi hızına oranla, engellerin uzaklık tahminlerine göre işleme alınması, sürücü tarafından öncelikli dikkat edilmesi gereken bölgelerin belirlenmesi için önemlidir.



Şekil 3.52. AIR ve yer düzlemi bölgelerinin birleşimi (üst), tersi alınarak elde edilen bölge (alt).

Yer düzlemi ve AIR bölgesi dışında kalan bölgeler; yere temas etmeyen bir yükseklikte kalan engeller ile hata barındıran bölgeleri kapsadığından dolayı önceki

iki maskenin birleşimi kullanılarak elde edilen Şekil 3.52.'deki üstteki görüntünün tersi alınarak Şekil 3.52.'de altta görülen gökyüzü bölgesi (yol ile direk ilişkisi olmayan alanlar) tanımlanmıştır. Eğer yer düzleminde ve AIR bölgesinde kalan alanlar yok edilmek isteniyorsa morfolojik olarak genişletme (dilation) işlemiyle bu alanlar giderilebilir.



Şekil 3.53. Yer düzlemi ile bağlantısı olmayan bölgelerin renk etiketleri.

Öncelik oluşturan yer ve yere bağlı engeller dışında kalan alanların, renksel etiketleri Şekil 3.53.'deki gibidir. Trafik lambası gibi nesnelerin ve öncelik önemi düşük olarak tanımlanmış havadaki nesnelerin takibi buradan yapılabilir.

GDD renk segmentasyonu; sistem içerisinde öncelikli alanların tespitinde, araç sürüş alanlarının belirlenmesinde, nesnelerin birbirleri arasında farklı ve benzer özelliklerinin tanımlanmasında, takibinde ve hataların giderilmesinde kullanılabilir.

3.5.4. Uzaklık ve renk tabanlı öncelik listesi oluşturulması

Bir önceki aşamada elde edilen ve engel teşkil eden nesnelerin etiket verilerine derinlik bilgisi ve renk bilgisi katılmadığında görüntüler sadece iki boyutlu düzlemde ele alınacağından dolayı yaya saptama işlemi, AIR bölgesi üzerinden sabit çerçeve kullanılarak gerçekleştirilecektir. Bu durumda farklı uzaklıktaki yayalar sabit çerçeve ile tespit edilemeyecektir. Ayrıca bu yöntem, bütün görüntüyü işleme almaktan daha mantıklı olsa bile işlem önceliğinin olmaması, yakında kazaya sebebiyet verebilecek

kritik bölgelerin veya dikkat edilmesi gereken noktaların zamanında işleme alınamamasına neden olacaktır. Bunların olmaması için hangi bölgelerin daha önemli ve işlem önceliğinin olduğu, hangi bölgelerin ise daha sonra inceleneceği tez aşamasında oluşturulan öncelik listesi yardımıyla tespit edilmektedir.

Tanımlanan bu öncelik listesi, kazaya sebebiyet verebilecek araca yakın ve tehlike teşkil edebilecek nesnelerin daha hızlı tanımlanması açısından önem arz etmektedir. Öncelik listesi oluşturulması amacıyla, GDD renk segmentasyonu ile elde edilen sonuç verileri içerisinde düşük alana sahip ve komşuluklarından farklı renk özelliği gösteren bölgeler, arka plan bölgelerinden ayrılmaktadır. Bu farklı renge sahip alanlar, kapladığı büyüklük ve nesnelerin yakınlıklarına göre öncelik listesinde tanımlanmıştır. İşlem öncelik listesi, AIR bölgesinde yol üzeri engelleri barındıran BBA segment listesine, uzaklık ve renk verilerinin eklenmesiyle Şekil 3.54.'de görüldüğü gibi oluşturulmuştur.

Variables - AirSegLabelsBB					
AirSegLabelsBB					
33x1 struct with 5 fields					
Fields	Area	Centroid	BoundingBox	Disparity	ColorSeg
1	168	[755.6190,203.2500]	[736.5000,197.5000,36,9]	53.2440	[65,122,4.9996e-04]
2	91	[332.5824,193.2418]	[326.5000,186.5000,13,11]	37.6990	[21,91,9.9993e-04]
3	897	[220.8986,207.8751]	[205.5000,183.5000,37,40]	81.0643	[19,767,0.0020]
4	436	[731.7362,200.0917]	[687.5000,187.5000,93,19]	51.9965	[33,141,4.9996e-04]
5	1486	[27.0707,219.7342]	[1.5000,201.5000,56,35]	99.5523	[135,128,3.1479e-04]
6	1312	[136.5015,200.2652]	[98.5000,175.5000,82,48]	81.0345	[86,186,4.9996e-04]
7	247	[701.2713,188.7935]	[678.5000,177.5000,53,21]	40.6575	[126,64,4.9996e-04]
8	133	[1.1172e+03,195.2707]	[1.1045e+03,187.5000,27,17]	49.2752	[29,46,9.9993e-04]
9	20	[472.8000,175.6000]	[468.5000,169.5000,7,13]	15.4315	[76,11,4.9996e-04]
10	157	[428.5032,192.8917]	[416.5000,185.5000,21,13]	40.4697	[24,86,0.0015]
11	474	[608.5949,180.9030]	[589.5000,157.5000,57,36]	32.5459	[77,88,4.9996e-04]
12	313	[450.8818,166.8722]	[424.5000,147.5000,46,36]	16.1905	[76,101,4.9996e-04]
13	14	[510.3571,176.2857]	[508.5000,171.5000,4,8]	9.3601	[25,7,4.9996e-04]
14	434	[479.9885,163.1820]	[454.5000,151.5000,40,29]	12.1429	[55,111,4.9996e-04]
15	40	[388.6500,194.0500]	[383.5000,191.5000,10,5]	36.4159	[26,34,0.0085]
16	55	[600.4727,187.1818]	[594.5000,180.5000,11,16]	36.5988	[8,31,0.0095]
17	19	[550.5789,178.6842]	[547.5000,176.5000,6,4]	12.1429	[26,19,0.0085]
18	830	[1.0568e+03,229.7060]	[979.5000,220.5000,141,16]	100.9427	[7,830,0.4962]
19	1491	[61.5902,207.8229]	[17.5000,180.5000,75,54]	97.1547	[7,1491,0.4962]
20	407	[123.5627,209.3882]	[105.5000,195.5000,41,29]	82.8556	[7,407,0.4962]

Şekil 3.54. AIR bölgesi kullanılarak elde edilen öncelik listesi.

Öncelikli işlenmesi gereken nesneler listesi için, AIR çizgisi bulunduktan sonra AG ile segmentler tespit edilerek BBA ile etiket atama işlemi uygulanmıştır. BBA uygulanması işlem süresi, Şekil 3.55.'de görüldüğü gibi '0,02' saniye civarındadır.

```
>> tic,
[ AirSegLabelsBB, AIRlabelSegMap ] = func_AIR_RG( imAir, ObsMapCut, OrgDispMapCut );
toc,

tic,
[ AirSegLabelsBB, AIRlabelSegMap, CCAsegments] = func_AIR_RG_CCA( imAir, ObsMapCut, OrgDispMapCut );
toc,
Elapsed time is 0.728410 seconds.
Elapsed time is 0.745697 seconds.
```

Şekil 3.55. Bağlı bileşen analizi (BBA) işlem süresi fark tespiti.

AIR bölgesindeki bağlı bileşen nesnelerinin ayrı ayrı incelenmesi gerekmektedir. Her etikette bulunabilecek arka plan ile incelenmesi gereken nesnelerin özellikleri, yaya tanıma açısından önemlidir. Uzaklık ve renk bilgisi, etiket içi nesneleri ayırt etmede önemli ipuçlarını barındırmaktadır. Bu sebeple AIR bölgesinde BBA ile elde edilen her nesne bilgisine, GDD segment renk bilgisi ve uzaklık verileri eklenir. Bu aşamada nesnelerin sınır bölgelerinin belirlenmesi ile ÇS (Çerçeve Sınır) verisi de eklenir. Her nesnenin barındırdığı derinlik verisinin ortalaması bu etikete atanarak uzaklığa göre sıralama, Şekil 3.56.'te görüldüğü şekilde yapılır.

Variables - AirSegLabelsBB				
AirSegLabelsBB				
189x1 struct with 4 fields				
Fields	Area	Centroid	BoundingBox	Disparity
1	57	[932.4737,344.4...	[922.5000,342.5000,18,4]	248.3294
2	2	[997.5000,327]	[996.5000,326.5000,2,1]	222.1131
3	2	[874.5000,302]	[873.5000,301.5000,2,1]	186.1905
4	1	[872,301]	[871.5000,300.5000,1,1]	184.9256
5	211	[1.1011e+03,26...	[1.0895e+03,253.5000,25,12]	132.5763
6	132	[1.1620e+03,26...	[1.1415e+03,254.5000,33,9]	127.8948
7	170	[1.0814e+03,25...	[1.0705e+03,249.5000,20,14]	126.8036
8	2	[1.0445e+03,259]	[1.0435e+03,258.5000,2,1]	124.7173
9	79	[1.1169e+03,25...	[1.1065e+03,251.5000,22,8]	121.6976
10	4	[1.0585e+03,256]	[1.0565e+03,255.5000,4,1]	121.1124
11	236	[1.1552e+03,25...	[1.1355e+03,247.5000,39,11]	121.0437
12	6	[1.0708e+03,25...	[1.0685e+03,253.5000,5,2]	118.5615

Şekil 3.56. Uzaklığa bağlı (derinlik verileri) nesne listeleme.

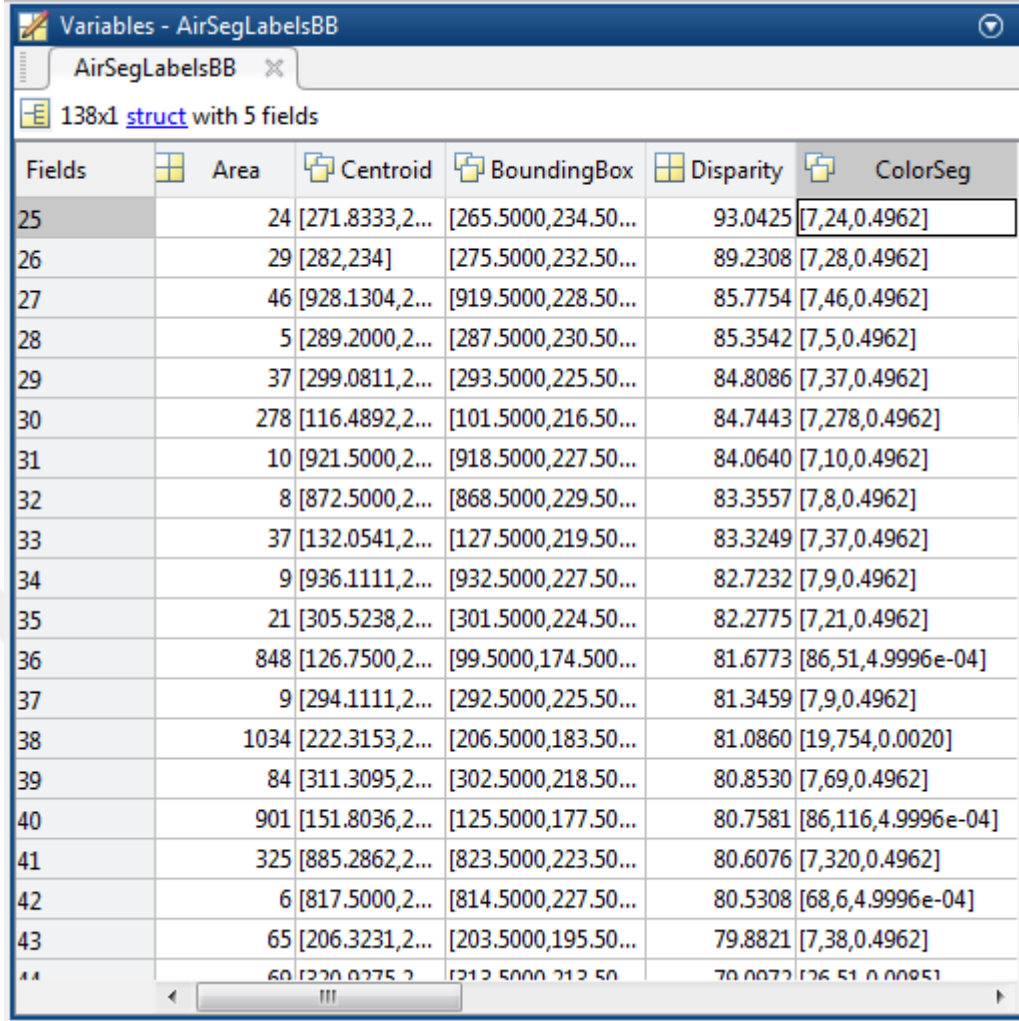
Şekil 3.56.'te dikkat edilecek olursa 2, 4, 6 piksel alana sahip nesneler, kameraya çok yakın olmasına rağmen listede yer almaktadır. Bu nesneler alansal olarak uzaklığa bağlı eşik değerlerinden geçirilerek elimine edilir. Uzaklığa bağlı tanımlanan eşik değerleri bölüm 3.5.5.'te açıklanmıştır. Bu işlemin uygulanmasının nedeni, derinlik haritasından kaynaklanan veya hatalı olarak tespit edilen ufak boyutlu hatalardan kurtulmaktır. Bu noktada belirlenen eşik değerleri, gerçekleştirilen uzaklık ve boyut ilişkisi testleri sonucunda en iyi başarıyı verecek şekilde belirlenmiştir.

Öncelik listesine renk bilgisinin aktarımı konusunda, GDD renk segmentasyonu sonucunda elde edilen segment sonuçları yapılan testlerde görüldüğü üzere aşırı segmentasyona yatkın olmasına karşın ufak alan kapsayan ve farklılık gösteren bölgeler, genellikle arka plandan ayrılan özel durumları ortaya koymaktadır. Bu noktada çevresel arka plan renklerinden ve ışık etkilerinden ayırım gösteren bölgelerde yaya olma ihtimalinin daha yüksek olduğundan bahsedebiliriz. Bu sebeple BBA segmentlerinde bulunan renk piksel sayısının tüm görüntüde aynı rengi barındıran piksel sayısına oranı, görüntüdeki etiket bölgesinde o rengin bulunma olasılığını vermektedir. Farklı engel bölgelerinin renk segmenti etiketleri, piksel sayıları ve olasılıkları Şekil 3.57.'de gösterildiği gibi matriste tutulmaktadır.

	1	2	3
28	7	5	0.4962
29	7	37	0.4962
30	7	278	0.4962
31	7	10	0.4962
32	7	8	0.4962
33	7	37	0.4962
34	7	9	0.4962
35	7	21	0.4962
36	86	51	4.9996e-04
37	7	9	0.4962
38	19	754	0.0020
39	7	69	0.4962
40	86	116	4.9996e-04
41	7	320	0.4962
42	68	6	4.9996e-04
43	7	38	0.4962
44	26	51	0.0085
45	26	22	0.0085
46	9	33	4.9996e-04
47	7	104	0.4962

Şekil 3.57. AIR-BBA etiketleri içerisinde GDD renk segment etiketi, renk segmenti etiket piksel sayısı ve bütün görüntüye göre belirli segment içerisinde ortaya çıkma olasılığını barındıran matris değerleri.

Renk segmenti etiket bilgisi ve olasılıkları tanımlandıktan sonra Şekil 3.57.'deki listeye, Şekil 3.58.'de görüldüğü gibi öncelik sıralama işlemi yapılması amacıyla eklenir.



Fields	Area	Centroid	BoundingBox	Disparity	ColorSeg
25	24	[271.8333,2...	[265.5000,234.50...	93.0425	[7,24,0.4962]
26	29	[282,234]	[275.5000,232.50...	89.2308	[7,28,0.4962]
27	46	[928.1304,2...	[919.5000,228.50...	85.7754	[7,46,0.4962]
28	5	[289.2000,2...	[287.5000,230.50...	85.3542	[7,5,0.4962]
29	37	[299.0811,2...	[293.5000,225.50...	84.8086	[7,37,0.4962]
30	278	[116.4892,2...	[101.5000,216.50...	84.7443	[7,278,0.4962]
31	10	[921.5000,2...	[918.5000,227.50...	84.0640	[7,10,0.4962]
32	8	[872.5000,2...	[868.5000,229.50...	83.3557	[7,8,0.4962]
33	37	[132.0541,2...	[127.5000,219.50...	83.3249	[7,37,0.4962]
34	9	[936.1111,2...	[932.5000,227.50...	82.7232	[7,9,0.4962]
35	21	[305.5238,2...	[301.5000,224.50...	82.2775	[7,21,0.4962]
36	848	[126.7500,2...	[99.5000,174.500...	81.6773	[86,51,4.9996e-04]
37	9	[294.1111,2...	[292.5000,225.50...	81.3459	[7,9,0.4962]
38	1034	[222.3153,2...	[206.5000,183.50...	81.0860	[19,754,0.0020]
39	84	[311.3095,2...	[302.5000,218.50...	80.8530	[7,69,0.4962]
40	901	[151.8036,2...	[125.5000,177.50...	80.7581	[86,116,4.9996e-04]
41	325	[885.2862,2...	[823.5000,223.50...	80.6076	[7,320,0.4962]
42	6	[817.5000,2...	[814.5000,227.50...	80.5308	[68,6,4.9996e-04]
43	65	[206.3231,2...	[203.5000,195.50...	79.8821	[7,38,0.4962]
44	60	[220.0275,2...	[212.5000,212.50...	79.0072	[26,51,0.0085]

Şekil 3.58. Öncelikli işlem listesine renk bilgilerinin eklenmesi.

Görüntü segment verisi Şekil 3.58.’deki gibi eklendikten sonra; uzaklık, renk segmentinin belirli bölgede ortaya çıkma olasılığı, farklı bölgelerdeki belirli bir rengin işgal ettiği alanların toplamının incelenen alanın tamamına oranı Denklem 3.36’daki tanımlamalara göre sıralama sırasında değerlendirilerek bu değerler büyükten küçüğe göre sıralanmıştır.

$$SıralamaDeğeri = d. \left(\frac{1}{P} \right). \left(\frac{\sum Piksel_c}{A} \right) \quad (3.36)$$

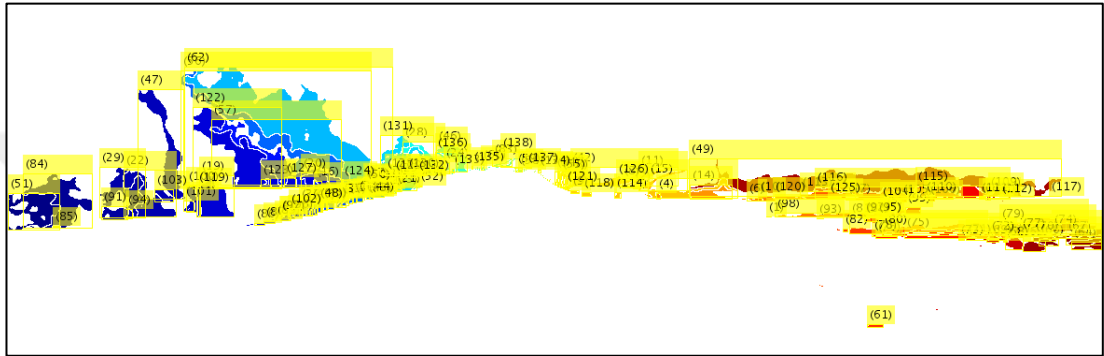
Denklem 3.36’daki tanımlamaya göre, ‘d’ derinlik verisini ifade etmektedir. AIR bölge segmentinde bulunan renk etiketine sahip toplam piksel sayısı ‘ $\sum Piksel_c$ ’ ile

temsil edilerek AIR bölge segmentinde bulunan toplam alana bölünmüştür. Olasılık verisi olan ‘ P ’ değeri ise işlenen nesnede bulunan renk etiketi piksel sayısının, görüntü üzerindeki aynı renk etiketine sahip piksel sayısı toplamına oranıdır. Bu işlemle belirli bir segment içerisinde renk etiket verisinin gözlenme olasılığı elde edilmektedir. Sonrasında ise hesaplanan sıralama değeri kullanılarak büyükten küçüğe öncelik listesi belirlenir. Sonucunda düşük olasılıkla ortaya çıkan az sayıda bulunan renk bölgelerine öncelik tanınarak daha iyi bir işlem önceliğinin sağlanması hedeflenmiştir. Sonuç Şekil 3.59.’da görüldüğü gibi, Şekil 3.58.’de 36.sırada olan 848 piksellik nesne işlem önceliğinde 29.sıraya, 38.sırada olan 1034 piksellik nesne ise 19.sıraya yükselmiştir. Bu sayede yüksek piksel barındıran ve kameraya yakın olan nesnelerin renk özellikleri ve büyüklüklerine göre daha erken işleme alınması sağlanmıştır.

Fields	Area	Centroid	BoundingBox	Disparity	ColorSeg
16	40	[332.8000,1...	[329.5000,189.50...	37.8516	[21,40,9.9993e-04]
17	78	[926.9872,2...	[899.5000,207.50...	53.6277	[78,27,4.9996e-04]
18	75	[599.1600,1...	[590.5000,178.50...	33.6931	[77,34,4.9996e-04]
19	1034	[222.3153,2...	[206.5000,183.50...	81.0860	[19,754,0.0020]
20	315	[332.2476,1...	[315.5000,180.50...	36.7434	[21,217,9.9993e-04]
21	19	[419.6842,1...	[416.5000,196.50...	41.2751	[24,17,0.0015]
22	901	[151.8036,2...	[125.5000,177.50...	80.7581	[86,116,4.9996e-04]
23	26	[404.1538,1...	[395.5000,191.50...	33.3929	[24,24,0.0015]
24	20	[472.8000,1...	[468.5000,169.50...	15.4315	[76,11,4.9996e-04]
25	198	[430.1465,1...	[418.5000,185.50...	40.3382	[24,96,0.0015]
26	5	[444,185]	[441.5000,184.50...	18.7202	[24,5,0.0015]
27	36	[393.8333,1...	[382.5000,190.50...	35.1707	[24,17,0.0015]
28	313	[450.8818,1...	[424.5000,147.50...	16.1905	[76,101,4.9996e-04]
29	848	[126.7500,2...	[99.5000,174.500...	81.6773	[86,51,4.9996e-04]
30	22	[323.7727,2...	[321.5000,217.50...	77.6522	[26,22,0.0085]
31	14	[332.2857,2...	[330.5000,218.50...	75.5315	[26,14,0.0085]
32	48	[347.5208,2...	[342.5000,210.50...	68.7410	[26,48,0.0085]
33	21	[358.0952,2...	[353.5000,208.50...	65.9184	[26,21,0.0085]
34	20	[369.1500,2...	[365.5000,207.50...	61.1570	[26,20,0.0085]
35	24	[362.2500,2...	[358.5000,207.50...	62.7184	[26,22,0.0085]

Şekil 3.59. İşlem önceliği listesi sonuç çıktı örneği.

Öncelikli işlem sırası; bilgisayar simülasyonları açısından göz ardı edilebilir olsa da, paralel işlem planlaması, zamansal kritik işlemlerin tanımlanması ve testleri, sistemin gerçekleşmesi noktalarında önemlidir. Bu şekilde algoritma çalıştırıldığında öncelik listesine göre elde edilen veriler, Şekil 3.60.'ta görülen sırada takip edilerek yaya tespiti yapılmaktadır. Bu işlem sırası ayrıca araçlar gibi diğer nesnelerin incelenmesi sırasında da kullanılabilir.



Şekil 3.60. Öncelik sırasına göre tanımlanmış ve etiketlenmiş engel nesnelerini barındıran görüntü.

Sonuçta sadece yakın engellerin değil, incelenen farklı renk özellikleriyle yayaların bulunabileceği bölgeleri de ön planda tutacak şekilde işlem sırası düzenlenmiş, Şekil 3.60.'ta görülen işlem öncelikli alanlar oluşturulmuştur. Bu sayede yaya olabilecek bölgelerin, diğer engellere göre daha hızlı incelenmesi sağlanarak yayaların tespit süresi azaltılmıştır.

3.5.5. HOG filtresi ön-işlemleri

HOG filtresine gönderilen nesnelerin büyüklükleri ve uzaklıkları, engel tanıma yöntemiyle belirlendikten sonra AG (Alan Genişletme), BBA (Bağlı Bileşen Analizi) yöntemleriyle hesaplanıp ÇS (Çerçeve Sınırı) metoduyla etiketlenerek ve öncelik listesi oluşturularak tutulmaktadır.

Belirlenen farklı büyüklükteki nesneler, uzaklıklarına göre ölçeklenmektedir. Ayrıca insanların en-boy oranlarına benzerlik gösteren alanlar HOG filtresiyle

işlenmektedir. Araca yakın ufak büyüklüğe sahip veya uzak olmasına rağmen çok büyük yer kaplayan alanlar yayaları temsil edemeyeceğinden dolayı bu alanlar işlem listesinden çıkarılarak HOG filtresine gönderilmemektedir. Bu sayede yaya olamayacak büyüklükteki alanlar hem yanlış pozitif sonuç vermemiş hem de işlenmediğinden iş yükü azaltılmış olur.

Belirlenen etiketlerdeki segment alanlarının yaya barındırıp barındıramayacağı, antropometrik değerler kullanılarak incelenmiştir [115]. Antropometrik verilerde, sadece yetişkin bireyler referans alındığından ve farklı uzaklıklarda uzunlukların görüntü üzerine izdüşümleri farklı olacağından dolayı uzaklıkların en-boy gibi direk ölçümleri yerine birbirlerine karşı oranları kullanılmıştır. Tablo 3.10.'da görüldüğü üzere insanların 'cm' cinsinden antropometrik değerleri; '2208' bayan ve '1774' erkek üzerinden alınan veriler ile elde edilmiştir.

Tablo 3.10. Antropometrik cm hassasiyetli ölçümler [115].

#		Cinsiyet	Ortalama	Sapma	Maks	Min	# Örnek
1	Dirsek-Bilek boyu	Bayan	26,25	1,54	33,40	17,00	2208
		Erkek	29,03	1,54	35,00	22,60	1774
2	Omuzlar arası genişlik	Bayan	46,85	3,47	60,90	37,30	2208
		Erkek	54,61	4,36	72,52	39,90	1774
3	Boy	Bayan	162,94	6,36	187,00	142,80	2208
		Erkek	175,58	6,68	204,20	149,70	1774
4	Kalça uzunluğu	Bayan	34,27	2,24	42,00	27,00	2208
		Erkek	34,18	2,03	41,60	28,20	1774
5	Açık kollar arası genişlik	Bayan	167,19	8,13	196,80	135,60	2208
		Erkek	182,31	8,19	215,90	147,40	1774

Antropometrik ölçüm verilerine göre HOG filtresine gönderilecek segmentler incelendiğinde, çerçeve genişlik ve yükseklik verisinde hata olmadığı sürece yayanın farklı uzaklıklarda en-boy oranlarının birbirlerine benzer bir şekilde değişim göstermesi beklenir.

Tablo 3.10 kullanılarak belirlenen yaya tahmini için önemli oranlar; açık kollar arası genişlik (#5) / boy (#3), kalça (#4) / boy (#3), omuz (#2) / boy (#3) olmakla birlikte bu oranlar yaya ön ve arka yüzü bakımından, dirsek-bilek (#1) / boy (#3) uzunluğu ise yaya yan olarak kameraya dönük ise önemli olacaktır. Buna göre maksimum ve minimum oranlar aşağıdaki Tablo 3.11.'de görülebilir.

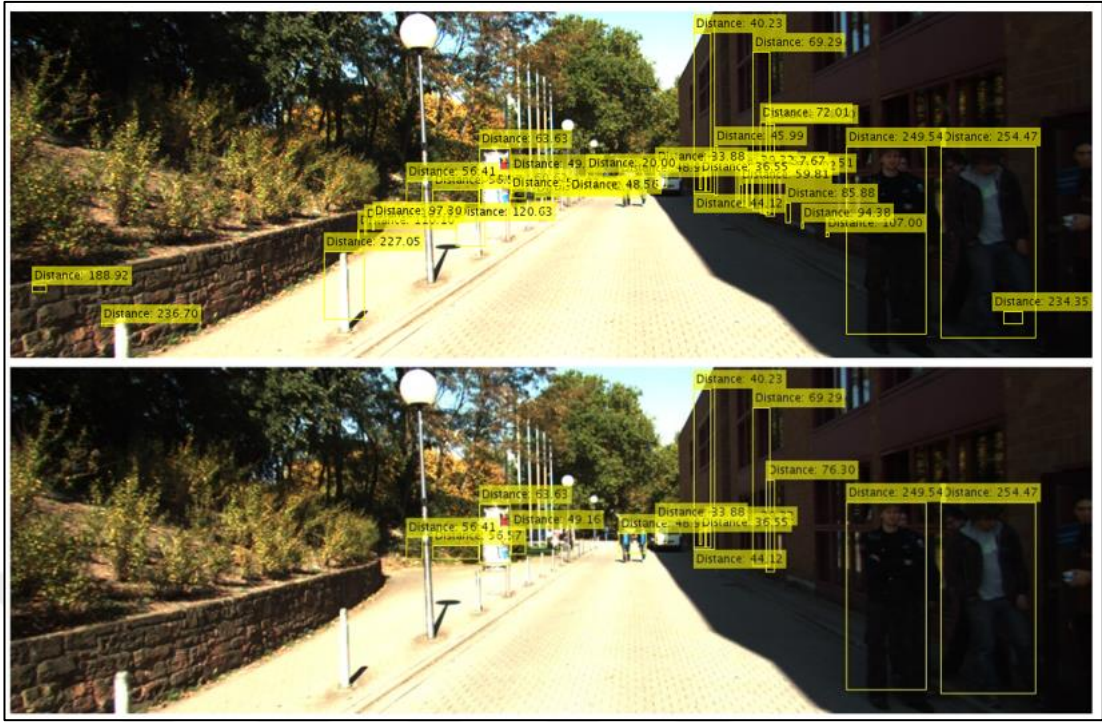
Tablo 3.11. Antropometrik olarak ölçüm oranları.

Oran #	Cinsiyet	Min.	Maks.
5/3	Hepsi	0,6641	1,4422
4/3	Hepsi	0,1322	0,2806
2/3	Hepsi	0,1827	0,4844
1/3	Hepsi	0,0832	0,2338

Yayaların antropometrik olarak birbirine oranlarından elde edeceğimiz minimum ve maksimum alanlara belirli bir hata payı eklenerek, segment çerçevelerinde yaya temsili için uygun olmayanların belirlenmesinde ve elenmesinde kullanılabilir. Örneğin Tablo 3.11.'de '5/3' numaralı oranda maksimum değer '1,4422'lik oranı '1,5' şeklinde aldığımızda, yaya genişliği yüksekliğin '1,5' katından fazla olan segment bölgeleri sadece bir tek yayayı barındırmıyor anlamına gelecektir. Bu sebeple bu bölgede kayan pencere yöntemiyle yayaların varlığı araştırılabilir. Eğer uzaklık-yükseklik ve uzaklık-genişlik değerleri beklentiden çok büyük veya küçük değilse bu yöntemin kullanımı uygun olacaktır.

Yayaların yüzü kameraya doğru dönük olmadığı durumda yayaların genişlik ve yükseklik oranı en küçük '0,0832' değerini almaktadır. Hatalardan kaçınmak için bu değeri yaklaşık '0,0800' olarak alabiliriz. Yani segmentin boyu enin '12,5' katından fazla ise bu segmentin, direk veya farklı bir tür engel olma olasılığı yüksek olacaktır. Fakat insan olma ihtimali bu durumda azalmaktadır. Bu değerlere göre, uygun olmayan segmentleri listeden çıkararak HOG filtresine gönderdiğimizde hem işlememiz gereken nesne miktarı azalmış hem de insan boyut ölçütlerine uymayan segmentlerin oluşturduğu hata oranı azalmış olacaktır.

Yayaların bulunabileceği alanları belirlemek için yapılan uzaklığa bağlı alan büyüklüğü eşiklemesi ve antropometrik oranlara göre filtreleme işlemleri tamamlandıktan sonra elde edilen sonuçlar üzerindeki değişimler Şekil 3.61.'dan görülebilmektedir.



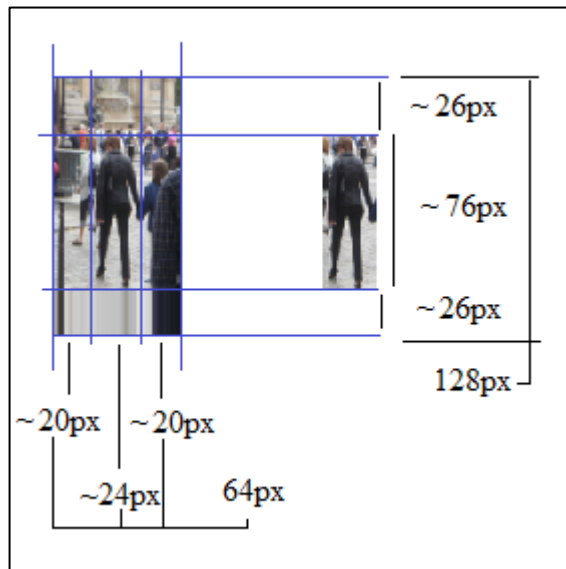
Şekil 3.61. Segmentlerin olası insan bölgelerinin antropometrik ölçütlere göre belirlenmesi. Orijinal segment listesi (üst), işlenmiş segment listesi (alt).

HOG filtresi için farklı boyutta dolgu değerlerinin belirlenmesi ve farklı boyutta yapılan ölçekleme işlemleri, filtre sonucunu çok büyük ölçüde etkilemektedir. Şekil 3.62.’deki üst ve alt görüntüde segmentlerin işlenmesi esnasında iki farklı çerçeve dolgu alanı kullanıldığında elde edilen HOG filtresi sonuçları görülmektedir.



Şekil 3.62. Aynı segmentlerin farklı çerçeve dolgu miktarları kullanılarak HOG filtresine gönderildiğinde elde edilen sonuçlar.

HOG filtresinde eğitim esnasında, insan ile görüntü çerçevesi arasında belirli bir orantı bulunmaktadır. Bu orantı, Şekil 3.62.'de görüldüğü gibi testlerin başarımlarını da etkilemektedir. INRIA veri setinden aldığımız bir pozitif görüntüyü inceleyecek olursak, yaklaşık oranları aşağıdaki Şekil 3.63.'deki gibi görebiliriz.



Şekil 3.63. INRIA yaya veri seti örneği, çevresel dolgu çözünürlük oranları.

Yukarıda Şekil 3.63.'de görüldüğü gibi yükseklikte dolgu/yükseklik oranı '52/128' iken, yatay olarak baktığımızda bu oran '40/64' olmaktadır. Boş alan ekleme (dolgu) ve çözünürlük dönüşümü (dolguların uzaklığa bağlı belirlenmesi) esnasında bu oranlar dikkate alınmıştır.

3.5.6. Uzaklık tespiti

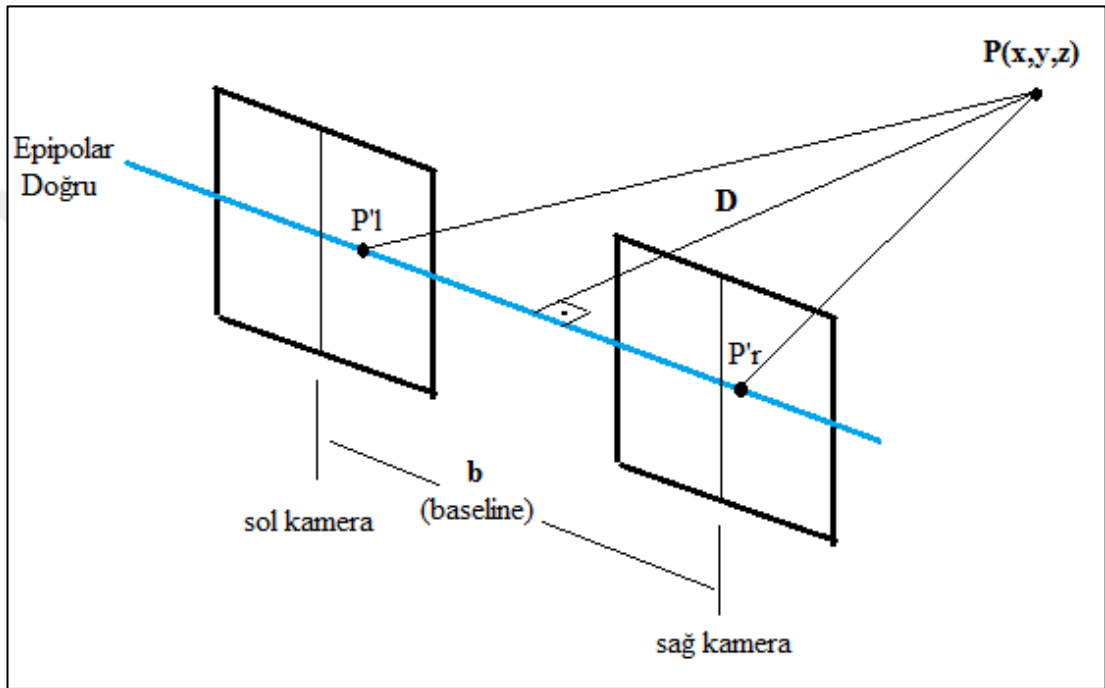
Kamera parametreleri ve referans değerleri bilindiği durumda, derinlik haritası verileri kullanılarak nesnelerin gerçek uzaklıklarının tespiti yapılabilmektedir. Bu verilerin ölçülebilmesi için piksel cinsinden odak uzaklığı, metre cinsinden kameralar arası uzaklık ve piksel cinsinden ölçülecek nesnenin sağ ve sol görüntüdeki piksel farklarının değerinin bilinmesi gerekmektedir [77, 76].

$$D = \frac{b \cdot f}{P'_l - P'_r} \quad (3.37)$$

Tez kapsamında, sağ ve sol görüntüden elde edilen eşleşen noktaların farklıyla oluşturulan ve normalize edilen derinlik görüntü bilgisi üzerinden gerçek uzaklıklar Denklem 3.37 yardımıyla tespit edilmiştir. Uzaklık tespitinde sol ve sağ görüntüde eşleşen noktaların bulunmasındaki hata, gerçek uzaklık tespitinde de hataya sebep olacaktır. Yukarıdaki Denklem 3.37'de; ' D ' değeri gerçek uzaklık tespit değerini belirtirken, ' b ' değeri kameralar arasındaki uzaklığı, ' f ' değeri ise kameranın odak uzaklığını ifade eder. Şekil 3.64.'deki ' P'_l ' ve ' P'_r ' noktaları ise, gerçek uzaydaki bir ' $P_{(x,y,z)}$ ' noktasının görüntüler üzerindeki karşılığı olan izdüşümlerini belirtmektedir. Burada kamera parametreleri kullanılarak elde edilen veriler ve derinlik haritası kullanılarak piksel cinsinden ölçülen noktalar arasındaki fark olan ' d ' değeri kullanılarak tahmini gerçek uzaklık değeri ' D ' Denklem 3.38'deki şekilde hesaplanmıştır.

$$D = \frac{b \cdot f}{d} \quad (3.38)$$

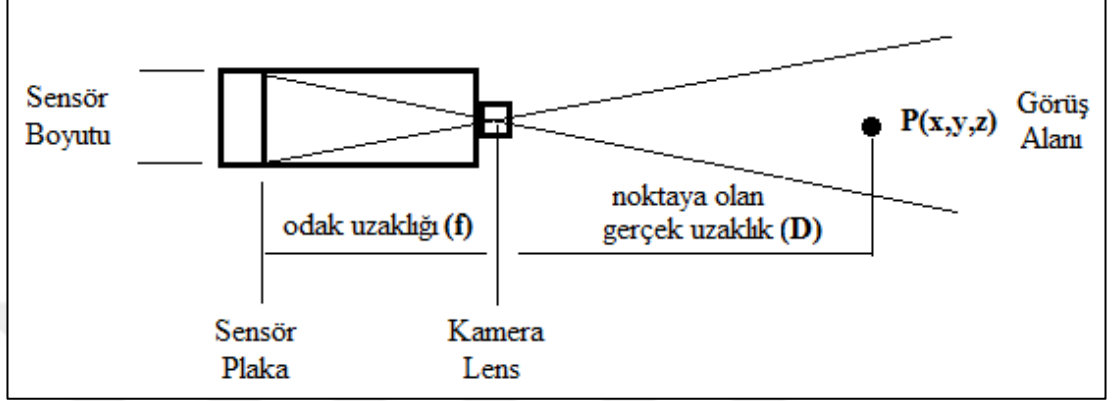
Denklem 3.38’de derinlik haritasından elde edilen ‘ d ’ değeri, kaynak araştırması bölümünde de belirtildiği üzere, kameradan uzaklaşarak ufuk noktasına doğru gittikçe hata değeri artan bir eğilim göstermektedir. Bu sebeple gerçek uzaklık değeri olan ‘ D ’ değerinin de hata oranı uzaklıkla lineer olmayan biçimde doğru orantılı artacaktır.



Şekil 3.64. Stereo kamera sisteminde epipolar doğru ve P noktasının izdüşümleri.

Sistem başarımlı testi için kullanılan KITTI veri setinde kullanılan kamera parametreleri ve sistemlerinin kurulumu, Geiger ve arkadaşları tarafından yayınlanan makalelerinde bahsedilmiştir [55]. Burada belirtilen kamera parametreleri ‘data_calib.zip’ klasöründe tutulmaktadır. ‘ f ’ odak uzaklığı değerini barındıran kamera esas (intrinsic) değerler matrisi de bu klasörün içinde ‘calib__cam_to_cam.txt’ dosyasına yazılmıştır. Bu dosya içindeki parametrelerin Matlab ortamında tanımlanabilmesi için Lisin’in yorumları ile Geiger ve arkadaşlarının makalesinden yararlanılmıştır [116, 55]. Veri setinde renkli görüntüler kullanıldığından, sol ve sağ renkli kameraları temsil eden üçüncü ve dördüncü kameraların matrisleri işlem için alınmıştır.

$$CM_i = \begin{bmatrix} fx & 0 & 0 \\ s & fy & 0 \\ cx & cy & 1 \end{bmatrix} \quad (3.39)$$



Şekil 3.65. Kamera üzerinde tanımlı odak uzaklığı ve diğer belirtilmiş bazı terimlerin görsel ifadesi.

Şekil 3.65. üzerinde görüldüğü gibi sensör plaka üzerine düşen görüntülerden elde edilen kalibrasyon verileri kullanılarak içsel kamera matrisi, Denklem 3.39'daki formatta kaydedilmiştir. Denklem 3.39'da tanımlanan kamera esas (intrinsic) matrisi ' CM_i ' içerisinde odak uzaklığı, piksel değerleri cinsinden $[fx, fy]$ şeklinde bulunmaktadır [117]. Matristeki 's' parametresi çarpıklık katsayısını (skew coefficient), $[cx, cy]$ ise piksel cinsinden optik merkez koordinatlarını belirtmektedir.

KITTI veri tabanından '2011_09_26_drive_0070' görüntü serisi çekimlerindeki 355'nolu görüntü verisi (Şekil 3.66.), test için seçilmiştir. Veri üzerindeki yaya görüntüsüne uzaklık tespiti işlemlerini uyguladığımızda Şekil 3.67.'deki sonuçlar elde edilmiştir.



Şekil 3.66. KITTI-2011_09_26_drive_0070 serisi 355'nolu test görüntüsü.

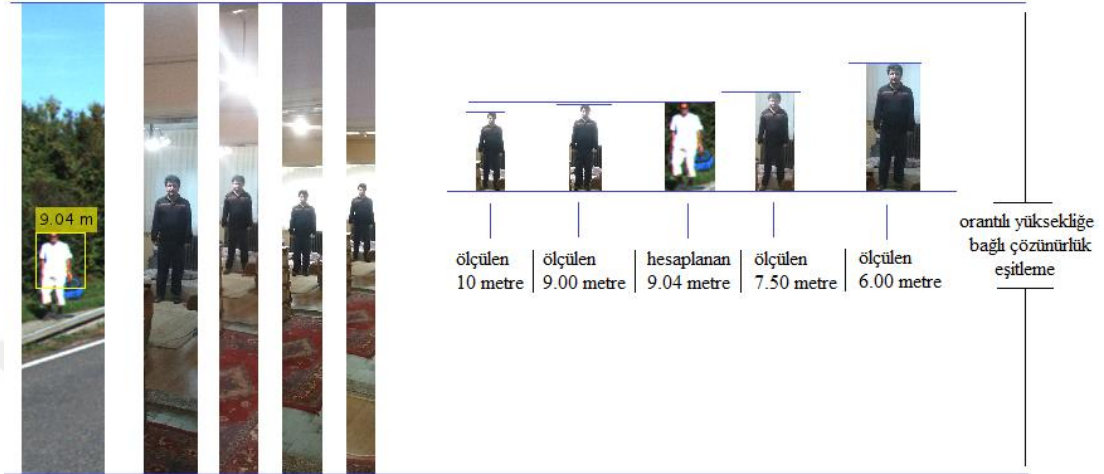


Şekil 3.67. Bulunan yaya üzerinden hesaplanan uzaklık tespiti.

İncelenen uzaklık tespit yöntemi dışında Matlab fonksiyonu olan '*triangulate*' komutu da kullanılabilmektedir [118]. Fakat bu yöntemle tespit yapıldığında KITTI veri tabanı kamera parametrelerinden kaynaklı olarak uzaklık tespit sonuçlarının yüksek hata verdiği gözlemlenmiştir.

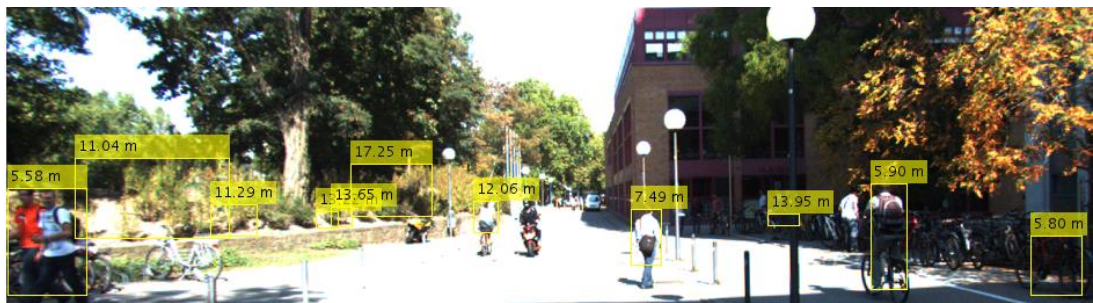
Veri tabanı görüntüleri kullanılarak yapılan uzaklık tespitinin doğrulaması için farklı kameralarla belirli aralıklarda uzaklık ölçümleri yapılmıştır. 6 ve 10 metre arasındaki uzaklıkların ölçüldüğü doğrulama görüntüleri yüksekliğe bağlı (geniş-açı, dar-açı gibi kamera çözünürlüğü durumundan etkilenmemesi için) oransal çözünürlük eşitleme uygulanarak hesaplanan uzaklıklar ile gerçek uzaklık ölçümleri karşılaştırılmıştır. Şekil 3.68.'deki karşılaştırmada referans yüksekliği '1,74' metre olarak alınmış, antropometrik erkek boy ortalaması '1,75' metre ve standart sapma '6,68' santimetre olduğundan bu değerler doğrultusunda yaklaşık hata değeri belirlenmiştir. Yükseklik değeri sapma aralığı '[5,68 – 7,68]' santimetre olarak farklı uzaklıklarda yükseklik piksel değişimleri oranları karşılaştırılmıştır. Standart boy

sapma değerleri göz önüne alındığında Şekil 3.68.'de ölçülen uzaklık tespitindeki hatanın, gerçek uzaklığa oranla 9 metrede en yüksek $[1 - 1,5]$ metre civarında olduğu hesaplanmıştır.



Şekil 3.68. Uzaklık hesaplaması doğrulama testleri (375 piksel yüksekliğe orantılı eşitleme).

KITTI veri seti '2011_09_28_drive_0038' görüntü serisi üzerinden '0000000007' numaralı görüntüde bazı bölgelerin uzaklık testi yapıldığında Şekil 3.69.'deki gibi bir sonuç elde edilmektedir. Görüntü daha iyi görülebilmesi için kesim ve büyütme işlemi uygulanmıştır.



Şekil 3.69. KITTI-2011_09_28_drive_0038 serisi 7'nolu görüntüde uzaklık tespiti.

BÖLÜM 4. ARAŞTIRMA BULGULARI

Bu bölümde, materyal ve yöntem bölümünde bahsedilen aşamalar uygulanarak, tez çalışması kapsamında yazılan kodların koşumu sonrasında elde edilen çıktılar incelenmiştir. Oluşturulan sistemin test görüntüler üzerinden elde edilen sonuçları, karşılaşılan hatalar ve başarımlar incelenerek aktarılmıştır.

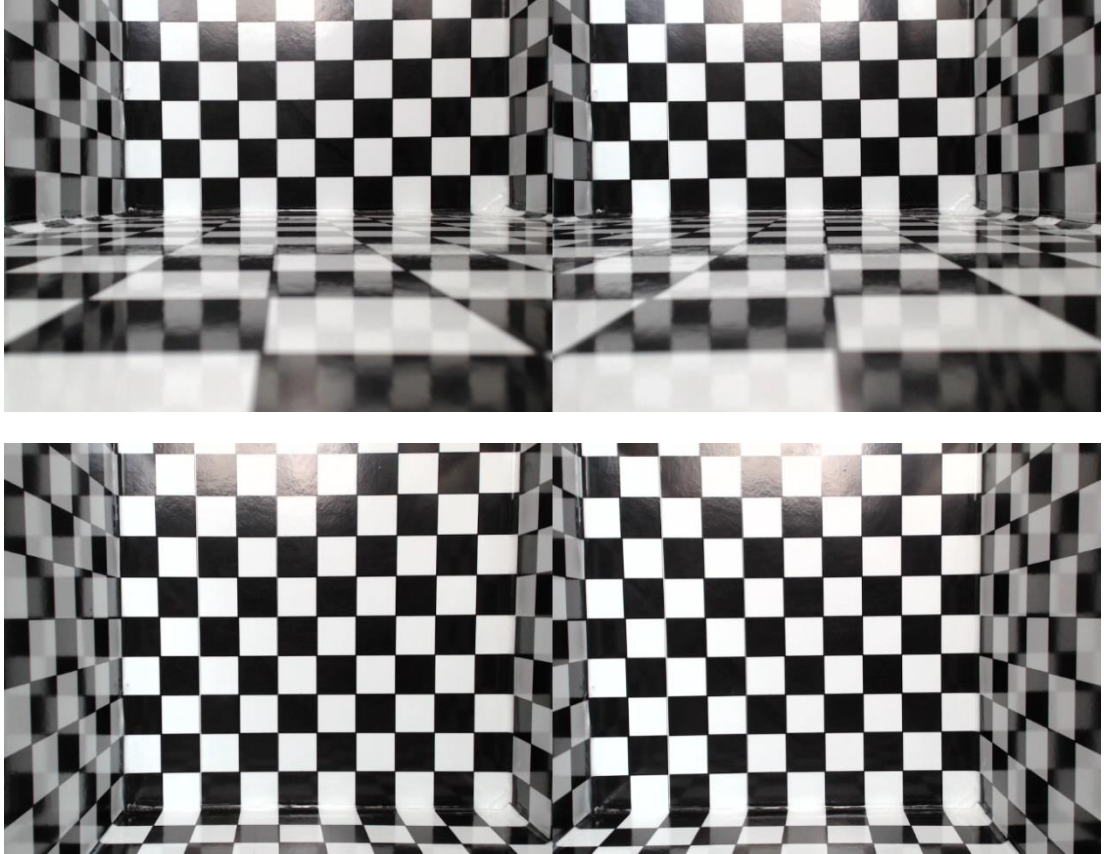
Gerçekleştirilen testler çoklu alt-sistem elemanlarını barındırdığından; kullanılan veri tabanlarının girdileriyle birlikte sonuçları ve tartışılması alt başlıklar içerisinde belirtilmiştir. Araç sürüş destek sistemi olarak veya otonom araçlarda kullanılabilecek bu tez çalışmasındaki her temel sistem ayrı ayrı veya bütünlük bir sistem içerisinde entegre edilerek uygulanabilir.

Yapılan işlemler sırasıyla alt başlıklarda da belirtildiği üzere, veri tabanlarından elde edilen stereo görüntülerinden derinlik haritası çıkarımıyla başlamakta ve sonrasında V-disparity dönüşümü ve engel tespiti yöntemleriyle yer düzlemi ve engeller tespit edilmektedir. Bu veriler kullanılarak engel barındıran bölgelerden, yaya olabilecek bölgeler tespit edilerek HOG yaya tanıma filtresi uygulanmıştır. Araştırma bulgularının incelendiği bu bölümde, gerçekleştirilen işlemlerle birlikte öne sürülen engel tanıma yöntemi, literatürdeki en çok kullanılan engel tanıma yöntemiyle karşılaştırılmıştır. Tespit edilen olası yaya bölgelerindeki nesnelerin incelenmesi, HOG yaya tanıma filtresinde görülen değişim ve sonuç çıktıları, en son olarak da tüm sistemin başarısı yine bu bölümde incelenmiştir. Bütün bu alanların geliştirilebilmesi için gelecekte yapılması planlanan çalışmalar da bölüm sonunda ifade edilmiştir.

4.1. Giriş verileri ve derinlik haritası çıkarımı

Tez kapsamında, başlangıçta araç ve kameralar kullanılarak veri girişi elde edilmesi için çalışmalar gerçekleştirilmiştir. Gerçekleme esnasında engel tanıma sisteminin yüksek başarımından dolayı, sistemin hata ve başarısının daha iyi oranda ölçümlenmesi ve çalışmanın devamlılığı için yüksek çözünürlüğe sahip KITTI veri tabanı kullanılmıştır [55].

Ön hazırlık kapsamında; ortaya konulan ve oluşturulan araçlar, test mekanizmaları ile kameralarda yapılan çalışma ve testler çalışmanın geçirdiği aşamaların anlaşılması bakımından önemlidir.



Şekil 4.1. Doğrultma için oluşturulan üç boyutlu dama tahtası görüntüleri.

Şekil 4.1.'de çalışmanın başlangıç safhasında 9.5cm aralıklı oluşturulan stereo kamera düzeneğiyle alınan üç boyutlu dama tahtaları görüntüleri, araç üzerinde

kullanılması planlanan kameraların kalibrasyonu için oluşturulmuştur. Sonrasında uzaklıkların daha iyi belirlenebilmesi için 100cm aralıklı olarak araç içi stereo kamera düzeneği araca yerleştirilerek, Şekil 4.2.'deki gibi ön çalışma test görüntüleri elde edilmiştir. Daha sonrasında ise araç olmadan da testlerin yapılabilmesi ve oluşturulan sistemin çalıştırılabilmesi için KITTI veri tabanı kullanılarak testler gerçekleştirilmiştir.



Şekil 4.2. Ön çalışma sürecinde 100cm aralıklı araç içi stereo kamera testleri.

KITTI veri tabanı, araç üzerine konulmuş stereo kameralardan elde edilen görüntülerden oluşmakta olup, derinlik ve üç boyutlu ortam verilerinin çıkarımı için gerekli verileri sağlamaktadır. Şekil 4.3.'de görüldüğü gibi aracın yol üzerindeki görüntü kayıtları tez aşamasında başarımın incelenmesi için girdi olarak kullanılmaktadır.



Şekil 4.3. KITTI veri tabanından örnek görüntü (2011_09_26_355 sol) [55].

Şekil 4.4.'de ise Şekil 4.3.'deki sol görüntüsü ve aynı set numaralı sağ görüntü ile elde edilen derinlik haritası görülebilmektedir. '15' piksel boyutundaki bloklar kullanılarak Semi-Global eşleştirme metodu ile derinlik tahmini yapılmaktadır.



Şekil 4.4. Veri tabanı görüntüsü derinlik haritası.

Görüntü üzerinde sol ve sağ görüntülerdeki ortak olmayan alanların oluşturduğu dikey alanlar (Şekil 4.4.'de görülen sol ve sağ taraflarda bulunan boşluklar) yazılan görüntü kesim kodu ile giderilmiş ve Şekil 4.5.'de elde edilen görüntü derinlik haritası olarak kullanılmıştır.



Şekil 4.5. Kenarlık kaldırma algoritmasından sonra elde edilen derinlik haritası.

Girdi görüntülerde de aynı şekilde derinlik haritası elde edilmesinde tespit edilen sol ve sağ taraflarda ortak olmayan kesim görüntüleri, gerekli görüldüğü durumlarda Şekil 4.6.'daki gibi kaldırılmaktadır.



Şekil 4.6. Belirlenen Stereo görüntü kenarlıklarının sol orijinal görüntüden kaldırılmasıyla elde edilen görüntü.

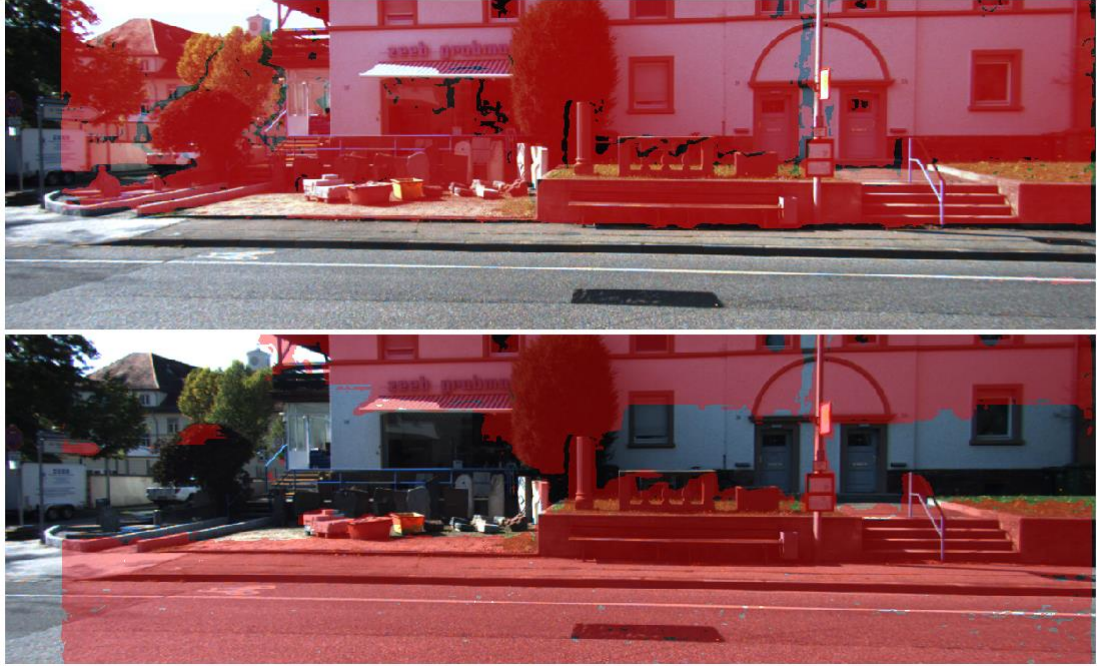
Derinlik haritası tespitinde, yarı global blok eşleştirme algoritması kullanılmıştır. Fakat farklı başarımları sergileyen derinlik tespit algoritmaları da kullanılabilir. Derinlik haritasında yatay düzlemde yüksek sayıda hata bulunması durumunda çıktıların başarımı negatif yönde etkilenmesine rağmen tezde önerilen sistem sonuçlarının genel olarak derinlik haritasında bulunan gürültü ve hata değerlerine karşı dayanıklı olduğu gözlemlenmektedir. Ayrıca daha yüksek doğrulukta derinlik haritası tespit yöntemleri kullanımında işlem gücü ve koşum zamanı önemli bir tasarım kriteri olarak öne çıkmaktadır. Daha az doğrulukla ve hızlı sonuç veren derinlik haritası yöntemlerinin kullanımında ise dikkat edilmesi gereken nokta V-disparity dönüşümü içerisinde elde edilen satır bilgilerindeki baskın frekansların

hatalardan oluşmaması gerekliliğidir. Bu durumda derinlik haritası tespitin doğruluğunun artırılması gerekliliği ortaya çıkmaktadır.

4.2. Engel bölgelerinin tespiti, test ve sonuçları

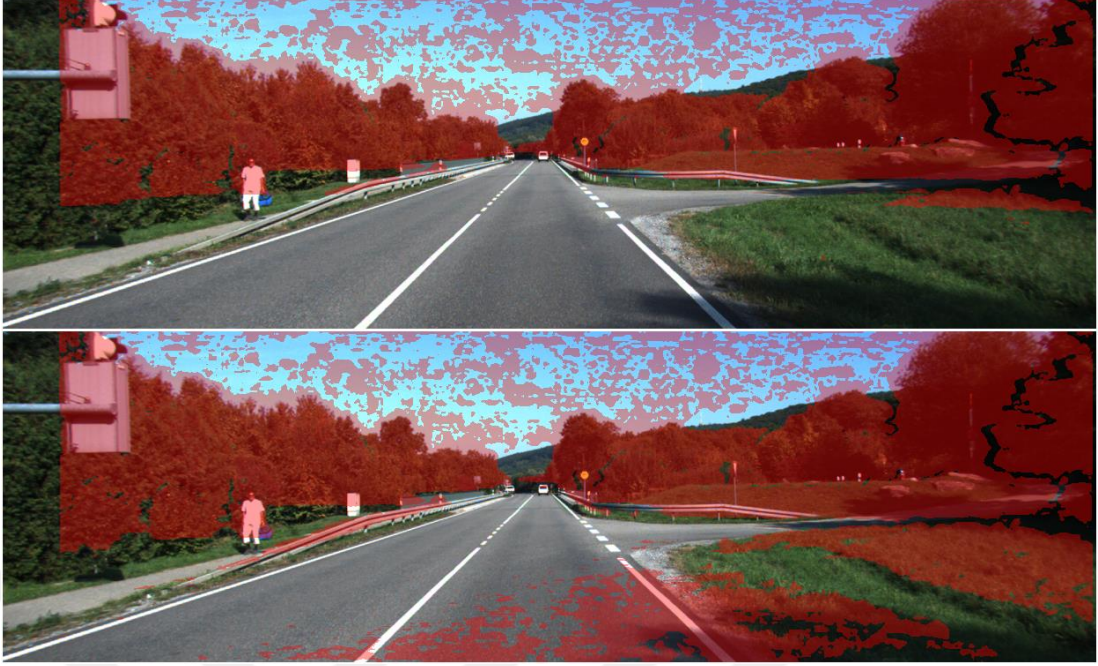
Araçların gidebileceği yol ve engellerin tespiti, yaya ve araç güvenliği konusunda önem arz ettiğinden ve aracın gittiği bölgelerin tanımlanması sistemin başarısını pozitif yönde etkilemesinden dolayı bu kapsamda yapılan çalışmalar önem kazanmaktadır. Bu aşamada tezde geliştirilen engel tanıma yöntemi, daha önceden engel tanımada kullanılan Hough dönüşümü yönteminden daha başarılı sonuçlar vermektedir. Önerilen yöntem ve Hough dönüşümü ile elde edilen sonuçlar bu başlıkta incelenmiş, performansları ve karşılaşılan hatalar veri tabanı görüntü test sonuçlarıyla ortaya konulmuştur.

Şekil 4.7.'de önerilen engel bulma yöntemi sonucu (üstteki görüntü) ve literatürde engel tespiti için kullanılan yöntem olan Hough dönüşümü ile elde edilmiş engel çıktısı (alttaki görüntü) karşılaştırılmıştır. Belirli yol ve şekillerde Hough dönüşümü kullanıldığında elde edilen yer düzlemi doğrusu ve ortaya çıkan engel sonuçları tez çalışmasında elde edilen sonuçlara benzerlik gösterse de farklı yol ve dönemeç gibi koşullarda önerilen engel bulma yönteminin daha yüksek başarımla elde ettiği görülmüştür.



Şekil 4.7. Veri seti ‘2011_09_26_drive_0070’ ve ‘012’ nolu görüntü verisi kullanılarak oluşturulan önerilen metotla elde edilen engel çıktısı (üst), Hough dönüşümü yöntemiyle elde edilen engel verisi çıktısı (alt).

Bir diğer örnek olan Şekil 4.8.’deki ‘355’ nolu görüntüde ise yol yapısı belirli olduğundan sonuçlar önerdiğimiz yöntemle benzerlik göstermesine karşın Hough dönüşümü ile elde edilen sonuçlarda ufuk noktasındaki ormanlık alanlardan dolayı kameraya yakın bölgede bir miktar hata ürettiği gözlemlenmiştir. Ormanlık alanın uzaklık açısından daha yakın ve yoğun olduğu bölgelerde ise Hough dönüşümü sonucu oluşan hatanın Şekil 4.9.’da ‘414’ nolu görüntü testinde görüldüğü gibi daha fazla arttığı tespit edilmiştir.



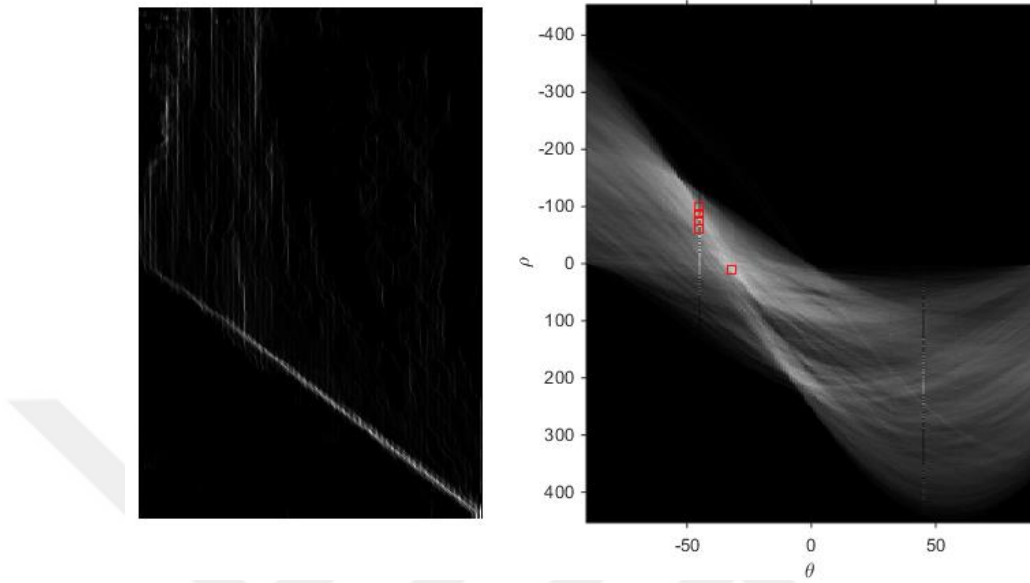
Şekil 4.8. Veri seti '2011_09_26_drive_0070' ve '355' nolu görüntü verisi kullanılarak oluşturulan önerilen metotla elde edilen engel çıktısı (üst), Hough dönüşümü yöntemiyle elde edilen engel verisi çıktısı (alt).



Şekil 4.9. Veri seti '2011_09_26_drive_0070' ve '414' nolu görüntü verisi kullanılarak oluşturulan önerilen metotla elde edilen engel çıktısı (üst), Hough dönüşümü yöntemiyle elde edilen engel verisi çıktısı (alt).

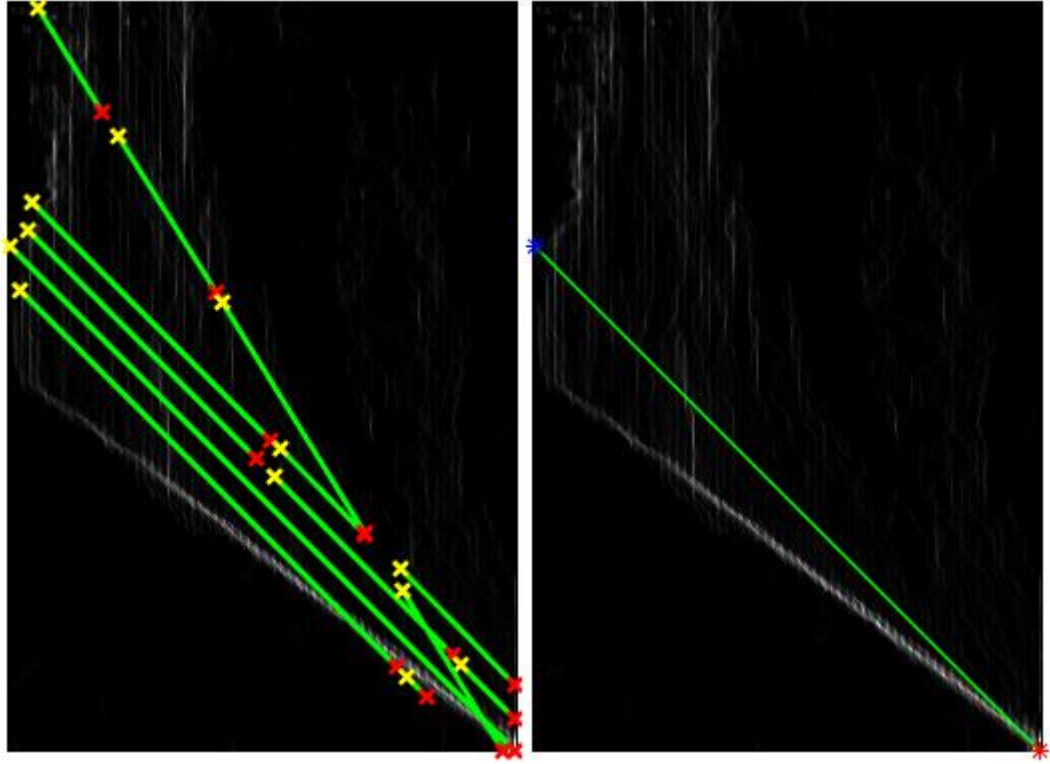
Karşılaştırmalı testler için oluşturulan Hough dönüşümü ile elde edilen engel haritasının nasıl elde edildiğini inceleyecek olursak, başlangıçta Şekil 4.10.'da solda

görülen 414'nolu görüntü test V-disparity verisi Hough dönüşümü uygulanarak sağdaki dönüşüm grafiği elde edilerek 5 adet tepe değeri seçilir.



Şekil 4.10. '2011_09_26_drive_0070-414' nolu test görüntüsü Vdisparity grafiği (sol), 5 tepe değeri (kırmızı kareler) gösterir Hough dönüşümü çıktısı (sağ).

Şekil 4.10.'da elde edilen tepe değerleri kullanılarak, Şekil 4.11. üzerinde sol görüntüde görülen V-disparity verisi üzerinde olası yer düzlemi doğruları tespit edilir. Bu olası aday doğrular arasından seçilecek doğru tüm V-disparity üzerinde yer düzlem doğrularını temsil etmesi gerektiğinden en uzun doğru, en iyi aday olarak alınır ve derinlik haritasındaki engellerin tespitinde kullanılır.



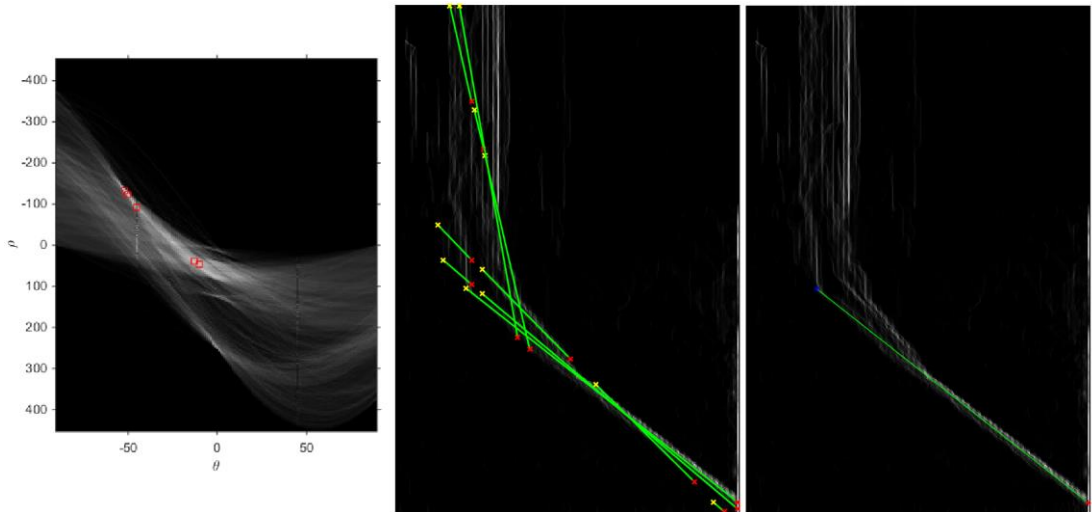
Şekil 4.11. Hough dönüşümü tepe değerleri kullanılarak elde edilen yer düzlemi aday doğruları (sol) ve seçilen en iyi doğru (sağ).

Tespit edilen en iyi aday doğru engel tespiti için kullanıldığında elde edilen sonuçlar ile tez aşamasında oluşturulan sistem sonuçlarının karşılaştırılması bu şekilde yapılmıştır. Hough dönüşümünde tepe değerler seçilirken, %30'luk bir eşik değeri filtrelemesinden sonra en yüksek değerdeki 5 farklı tepe değeri işleme alınmıştır. Yer düzlemi ile engel tespit yöntemi çalışmasında elde ettiğimiz sonuçlar ve diğer benzeri çalışmalarda kullanılan Hough dönüşümü kullanılarak elde edilen engellerin karşılaştırmalı olarak sonuçlarını aşağıda inceleyebilirsiniz. Gerekli görüldüğü alanlarda V-disparity verisi üzerinden yer düzlemi doğruları da karşılaştırmalı olarak sunulmuştur.



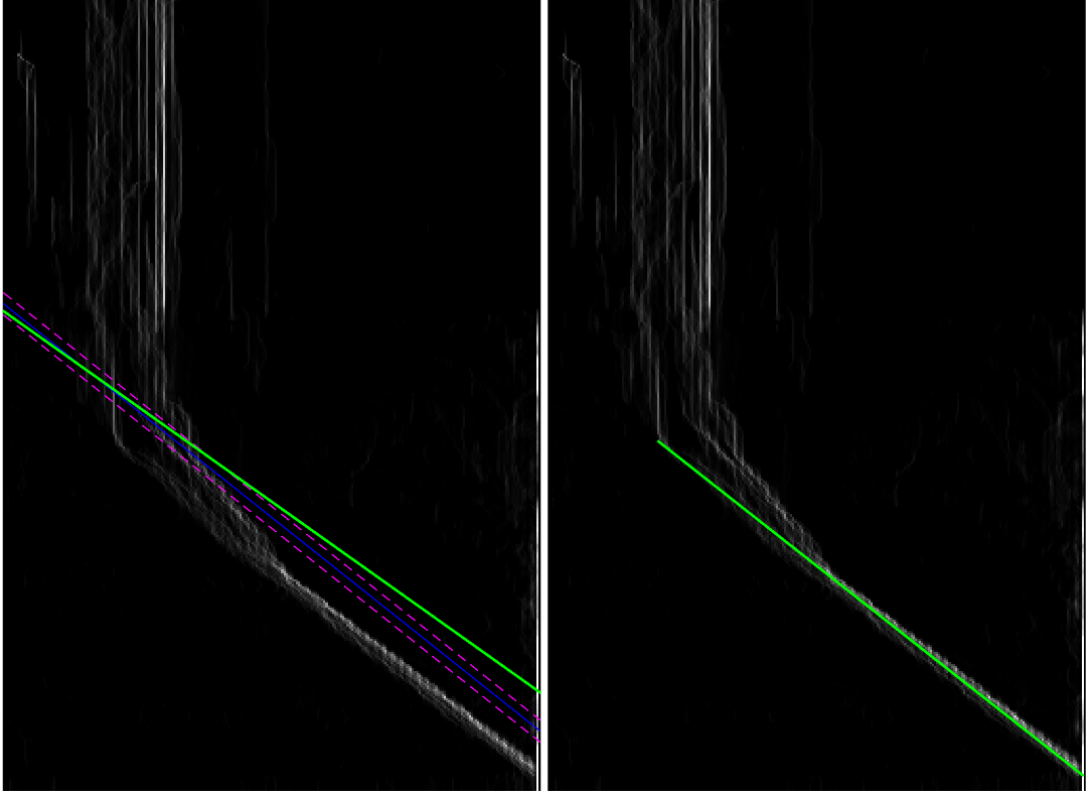
Şekil 4.12. ‘2011_09_26_drive_0070-003’ nolu test görüntüsü engel tespit sonuçları; tez aşamasında önerilen yöntem (üst), Hough dönüşümü ile elde edilen engeller (alt).

Şekil 4.12.’deki görüntüde Hough dönüşümü hatalı olarak görünse de burada hata yer düzlemi doğrusunun tespitinden sonra bir miktar yükseltme yapıldığı takdirde verilen sonucun çok da kötü olmadığı Şekil 4.13.’te görülen en sağdaki yer düzlemi doğrusundan anlaşılabılır.



Şekil 4.13. ‘2011_09_26_drive_0070-003’ nolu test görüntüsü Hough dönüşümü (sol), adaylar (orta) ve seçilen en iyi sonuç doğrusu (sağ).

Şekil 4.14. üzerinde de Şekil 4.12.’deki sonuçların sebebi olan doğruların nasıl konumlandırıldığı görülebilir. Bu karşılaştırmada önerdiğimiz yöntemdeki mavi çizgiler yer düzlemi doğrusunu, eflatun çizgiler yer düzlemini barındırabilecek hata bandını, yeşil doğru ise kullanılan engel tespit doğrusunu göstermektedir. Karşılaştırmadan da görülebileceği gibi hataların oluşturabileceği yükseklik, her iki doğruda da yer düzlem doğrusu uygun şekilde belirlenmiş olsa da olası hataların incelenmesi ve sisteme aktarımı daha yüksek doğrulukta sonuç elde edilmesini sağlamıştır.

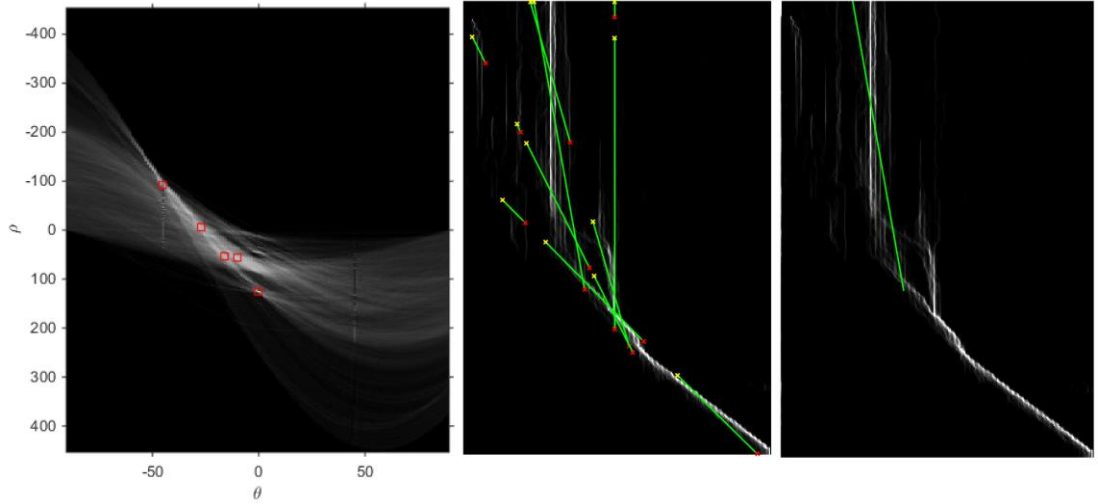


Şekil 4.14. ‘2011_09_26_drive_0070-003’ nolu test görüntüsü V-disparity üzerinde belirtilen yer düzlemi doğruları; önerilen metot (sol), Hough dönüşümü (sağ).

Adaylar arasından seçilen en uzun doğrunun Şekil 4.14.’deki görüldüğü şekilde V-disparity üzerinde yer düzlemini belirten doğruya uyması beklenir. Fakat bu varsayım Hough dönüşümü ile elde edilen engeller için, yapılandırılmamış yollarda veya ağaçlık bölgelerde hatalı sonuç verdiği gözlemlenmiştir. Ayrıca belirlenen doğrular ancak belirli bir eşik değeri ile yükseltilerek kullanıldığı takdirde bu yöntem için uygun sonuçlar elde edilmiş olunur. Aşağıdaki Şekil 4.14.’de de adaylar

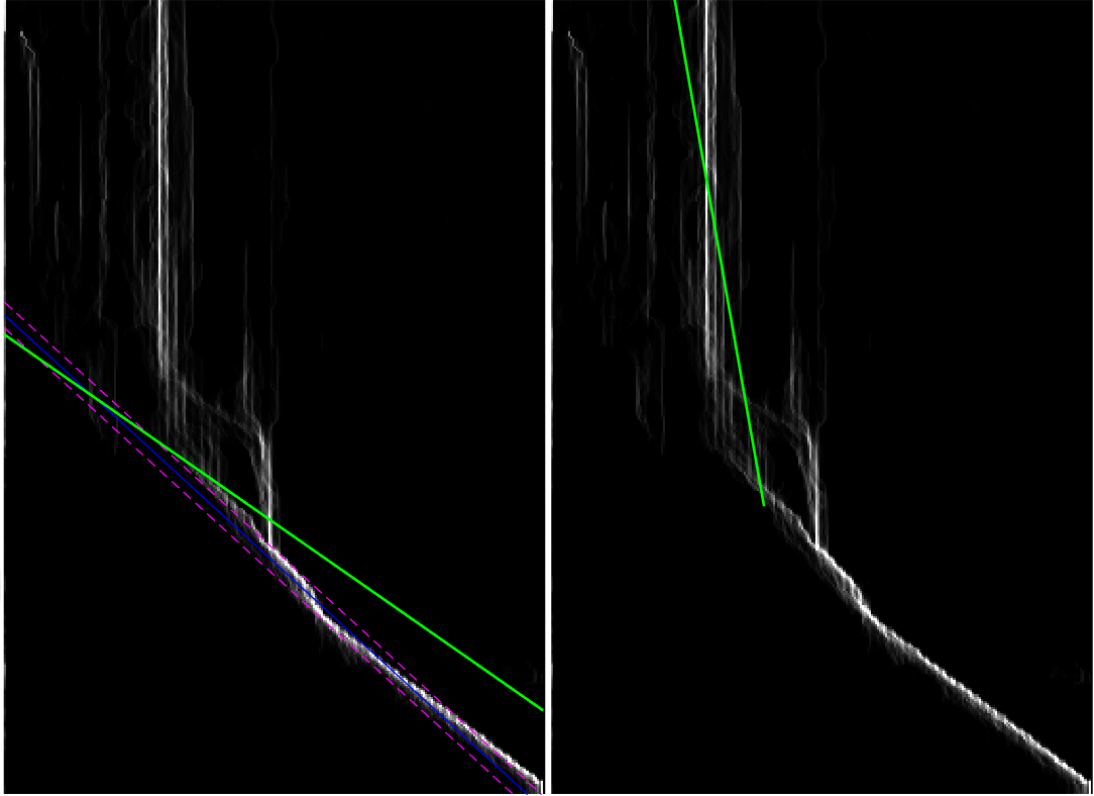
arasından bahsedilen şekilde önerilen yöntem (sol) ve Hough dönüşümü ile elde edilmiş yer düzlemi doğrusu (sağ) V-disparity üzerinde belirtilmiştir. Önerilen yöntem, Hough dönüşümü kullanılarak elde edilen engellerden daha iyi performans verdiği Şekil 4.12.'de gözlemlenmiştir.

Hough dönüşümü yöntemiyle bulunan engellerin tespitindeki hatalara gelecek olursak; yapılan testlerde Hough dönüşümü yöntemiyle elde edilen sonuçların bazılarında, özellikle engellerin özelliğine göre hatalı doğruların elde edildiği gözlemlenmiştir. KITTI veri setinde '2011_09_26_drive_0070' nolu setteki '0000000012' nolu görüntü bu tarz hataların nasıl oluştuğuna dair örnek olarak Şekil 4.15.'deki işlem aşamalarına bakılarak anlaşılabilir.



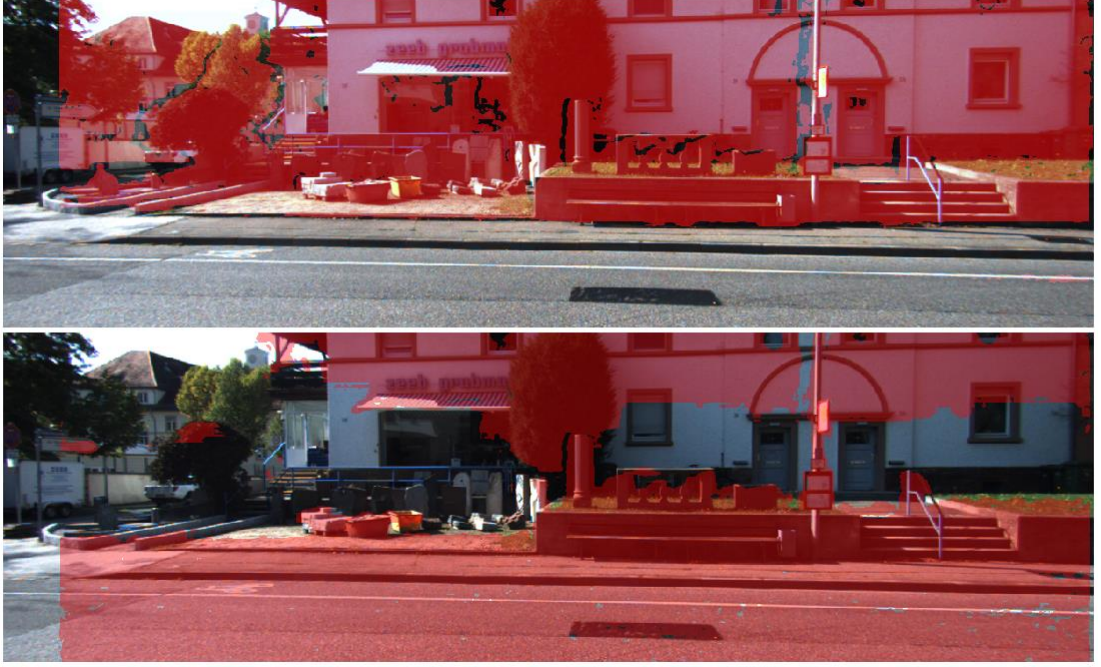
Şekil 4.15. '2011_09_26_drive_0070-012' nolu test görüntüsü üzerinde uygulanan Hough dönüşümü ile elde edilen yer düzlemi doğrusu ve aşamaları.

Şekil 4.15.'de görüldüğü gibi Hough dönüşümü ile aday doğrular oluşturulurken giriş görüntülerindeki yüksek uzunluğa sahip engellerin varlığı Şekil 4.15. ve Şekil 4.16.'da sağ taraftaki grafikte görüldüğü üzere hatalı sonuç doğurmaktadır.



Şekil 4.16. '2011_09_26_drive_0070-012'nolu test görüntüsü V-disparity üzerinde belirtilen yer düzlemi doğruları; önerilen metot (sol), Hough dönüşümü (sağ).

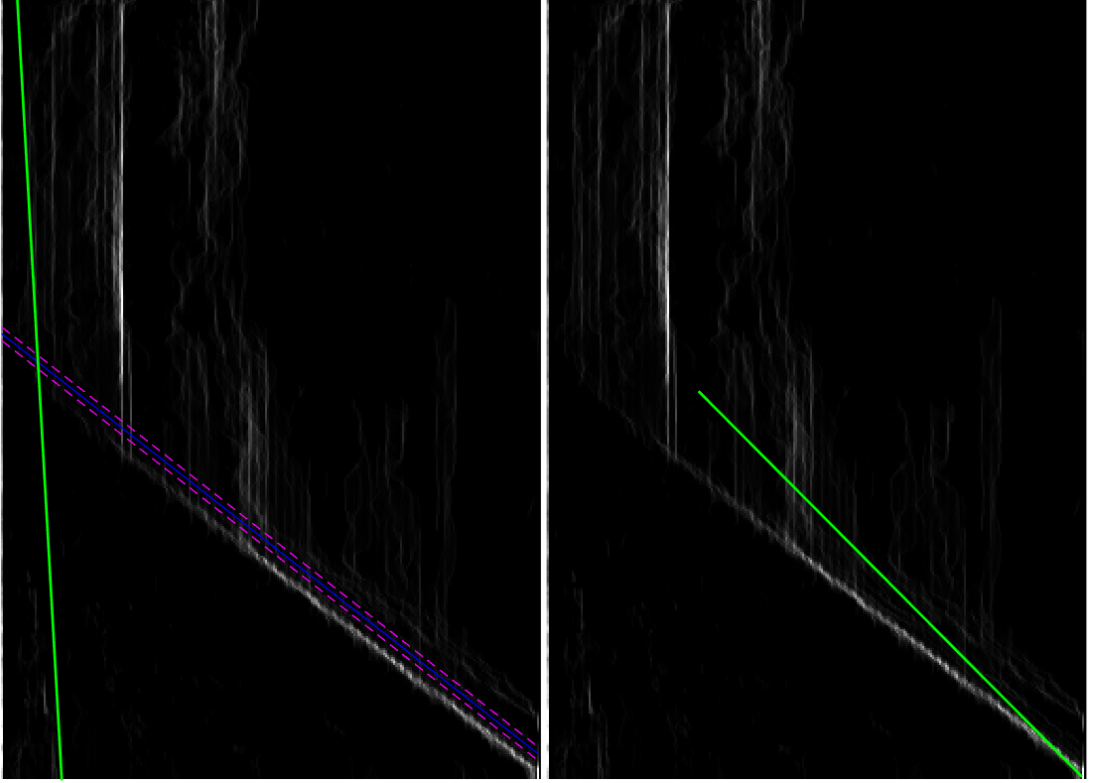
Yukarıdaki Şekil 4.16.'da görülen yer düzlemi doğruları sonucunda, aşağıdaki Şekil 4.17.'de alttaki görüntüde görüldüğü gibi Hough dönüşümü ile elde edilen sonuç hatalı olmakta, Şekil 4.16. sol grafikte görülen ve tezde önerilen yöntem ise Şekil 4.17. üstte görülen sonuç çıktısında görüldüğü gibi yüksek başarımda bir sonuç vermektedir. Dikkat edilirse Şekil 4.16.'da sol V-disparity grafiğinde görülebilecek yoğun engeller tez aşamasında önerilen yöntemdeki bulunan doğruyu da bir miktar yukarı doğru yükseltmiş ve sol tarafa yönelik dönmesine sebep olmuştur. Fakat frekans ağırlıklarına göre doğru belirlendiğinden bu hata kabul edilebilir sınırlar dahilinde sonuç gösterdiği elde edilen engel haritasından Şekil 4.17.'deki üst görüntüden görülebilir.



Şekil 4.17. ‘2011_09_26_drive_0070-012’ nolu test görüntüsü engel tespit sonuçları; tez aşamasında önerilen yöntem (üst), Hough dönüşümü ile elde edilen engeller (alt).

Tezde oluşturulan engel tanıma çalışmasında yer düzlemi doğrusu tespitinde herhangi bir hata görülmemiş olmasına rağmen sonrasında kamera yüksekliğini tahmin ederken geliştirilmiş olan algoritmanın bazı görüntülerde hataya sebep olduğu gözlemlenmiştir. Bu durum, kameraya yakın olan bölgelerde derinlik haritasında satır bazında yüksek sayıda hatalı derinlik verisi bulunmasından kaynaklanmakla birlikte yer düzlem doğrusunun düzeltilmesi aşamasında eğim kısıtlaması kullanılarak giderilmiştir. V-disparity üzerinde, kameraya yakın olan bölgedeki yüksek frekansa sahip hataların sola yaslı (yüksek uzaklıkta/düşük değişim) olmasını göz önüne alarak derinlik hatalarının çıkarımı sistem başarısını daha da arttırmaktadır. Sistemin gerçek hayat verilerini aldığı esnada, her görüntü çerçevesi için yükseklik tahmini işleminin gerekli olmadığı; kamera yükseklik parametresinin veya birkaç çerçeve görüntü kullanılarak elde edilen yükseklik tahmini değerinin doğrulama amaçlı kullanılmasının yeterli olacağı düşünülmektedir.

Şekil 4.18.’de kamera yüksekliği tahminindeki hatadan kaynaklanan doğrunun yapısı görülebilmektedir. Kamera yüksekliği ile doğru düzenlemesi yapılmadan önce tespit edilen doğru mavi, tolerans bölgesi ise eflatun renkle belirtilmiştir.



Şekil 4.18. ‘2011_09_28_drive_0038-063’ nolu test görüntüsü V-disparity üzerinde belirtilen yer düzlemi doğruları; önerilen metot (sol), Hough dönüşümü (sağ).



Şekil 4.19. ‘2011_09_28_drive_0038-063’ nolu test görüntüsü engel tespit sonuçları; tez aşamasında önerilen yöntem (üst), Hough dönüşümü ile elde edilen engeller (alt).

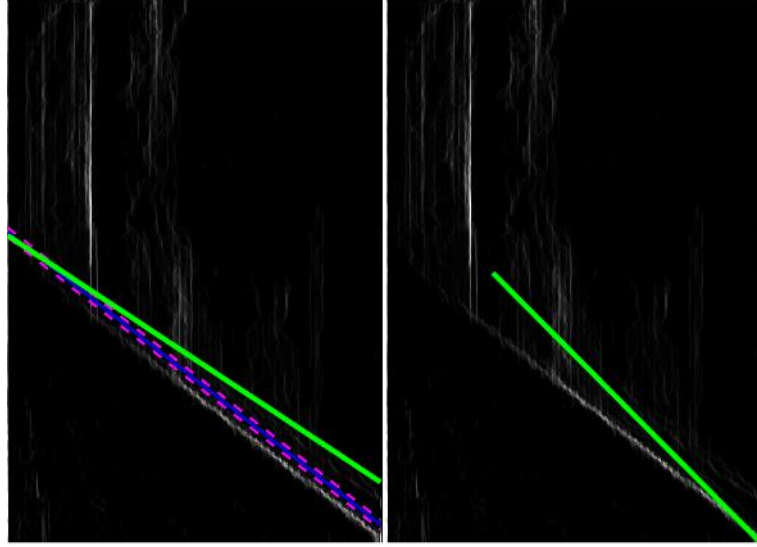
Şekil 4.18.'deki doğru verileri ile engel sonuçları belirlendiğinde Şekil 4.19.'daki görüntü elde edilmektedir. Şekil 4.19. görüntülerinde altta yer alan görüntü ise Hough dönüşümü ile elde edilen sonuç karşılaştırmasıdır.

Şekil 4.18.'de sol grafikteki hatalı sonucu doğuran Şekil 4.20.'deki derinlik haritası giriş verilerine baktığımızda kamera yakınında bulunan bölgelerdeki hataların yüksek miktarda olduğu görülmektedir. Bu sebeple de bölgedeki piksel verilerinin yatay ortamdaki baskın histogram değerlerinin de değiştiği görülebilmektedir. Bu durumda yatay satırlarda baskın histogram frekans değeri kullanıldığında, kamera yüksekliğini tahmin eden değerler hatalı olarak düşük derinlik değerini seçebilmektedir. Hatanın buradan kaynaklandığı, basit bir sınırlama değeri kullanım yöntemiyle ve kamera yükseklik bölgesini tanımlayan algoritmanın geliştirilmesiyle bu tip hataların oluşmasının önüne geçilmiştir.

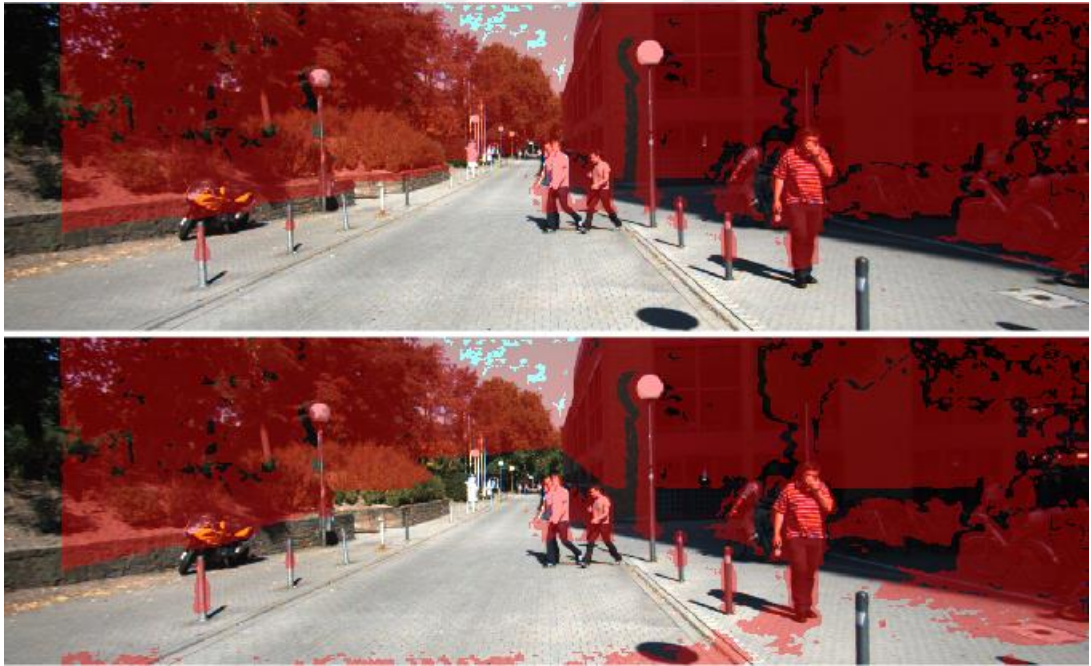


Şekil 4.20. '2011_09_28_drive_0038-063' nolu test görüntüsü derinlik haritası.

Yükseklik tahminindeki hatalar, metot kısmında da belirtildiği üzere verilerin işleme aralığıyla eğim sınırlaması yapılarak giderilmiştir. Kamera yüksekliği için alınan verilerin $[-2, +4]$ tolerans bölgesinde sınırlandırılması ile eğimin de bu bölge aralığında kalması sağlanmış ve engeller Şekil 4.21. ve Şekil 4.22. üzerinde görüldüğü gibi başarılı bir şekilde bulunmuştur.



Şekil 4.21. Eğim sınırlaması yapıldıktan sonra engel tahmini önerilen metot (sol), Hough dönüşümü (sağ) sonuçları.

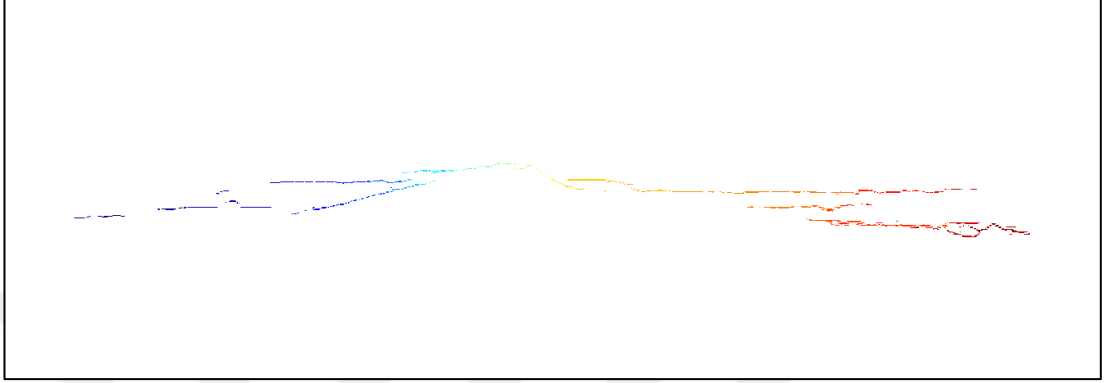


Şekil 4.22. Eğim sınırlaması yapıldıktan sonra engel tahmini önerilen metot (üst), Hough dönüşümü (alt) sonuçları.

4.3. AIR bölgesi ve yol üzerinde bulunan nesnelerin tespit ve testleri

AIR (Above Immediate Road); yer düzlemine bağlı engel bölgelerini ifade eden ROI (Region Of Interest) bölgesini yani incelenecek özel bölgeyi temsil etmektedir.

Bölüm 3.3.'de elde engeller kullanılarak Bölüm 3.5.1.'de tanımlandığı üzere AG (Alan Genişletme) ve BBA (Bağlı Bileşen Analizi) ile elde edilen bu bölgeler yaya tespitinde önem arz etmekte ve sonunda da sistem performansını büyük ölçüde etkilemektedir.



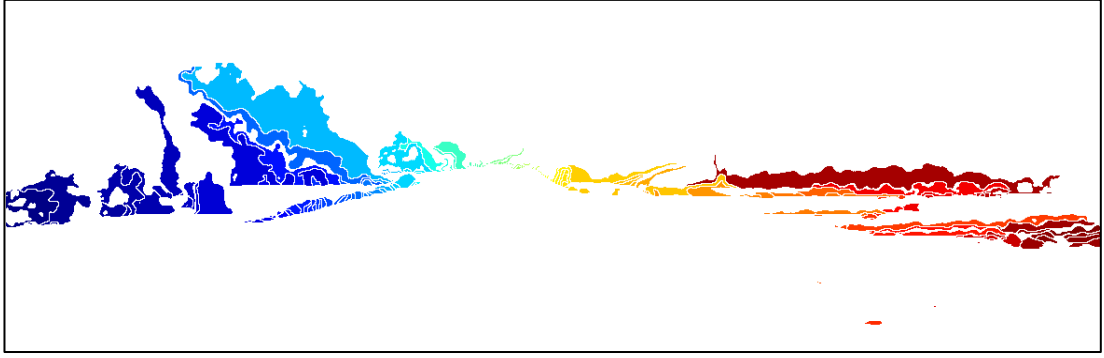
Şekil 4.23. AIR bölge tespiti için yer düzlemi doğrusu üzerinden elde edilen uzaklığa bağlı renklerle gösterilen engeller ile yeri ayıran çizgi.

Şekil 4.23.'de AIR isimli ROI bölgesini tanımlayabilmek için seçilmiş yer düzlemine bitişik piksel tabanlı bölgeler görünmektedir. Bu pikseller, yer düzlemi ile engellerin barındığı üst bölgeleri birbirinden ayıran bariyer görevi görmektedir. Şekil 4.24.'de ise Şekil 4.23.'deki çizginin AG metodu ile genişletilmiş halini vermektedir.



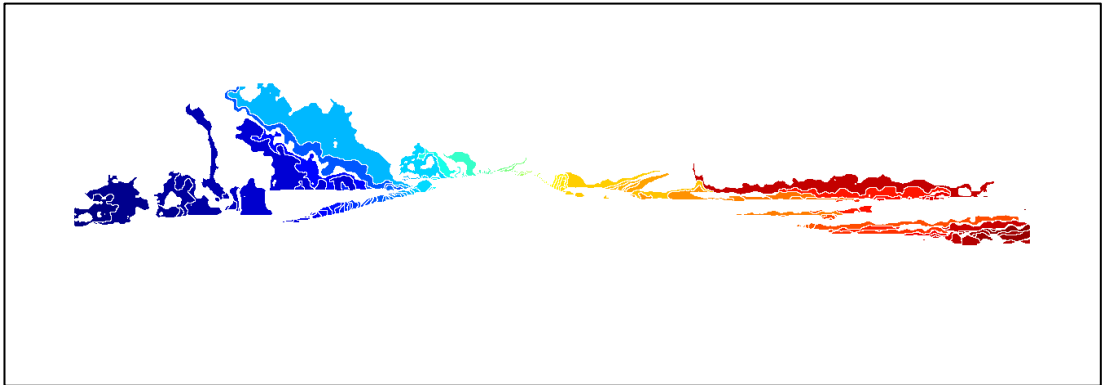
Şekil 4.24. AG ile elde edilen segment sonucu.

Şekil 4.24.'de görülebileceği gibi fazla olan alanlar, yer düzlemi doğrusu ile kesişen noktalarda kesilmiş ve çalışmada ufak alanlı bölgeler, eşik alan değerlerine göre çıkarılarak Şekil 4.25.'deki gibi kullanılmıştır.

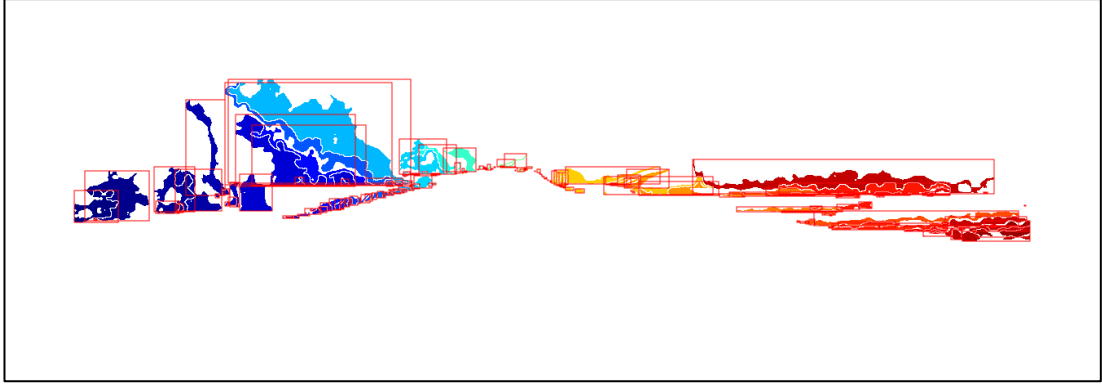


Şekil 4.25. Yer düzlemi doğrusunun altındaki alanların çıkarılmasıyla elde edilen AIR bölgesi.

Şekil 4.24.'de incelenen komşuluk benzerlik değerleri, derinlik haritasının hassasiyetine göre ayarlanmalıdır. Dikkat edilirse bazı bölgeler, aşırı yayılma gösterdiğinden sonuçların negatif yönde etkilenmemesi için Alan Genişletme (AG) benzerlik değerleri, derinlik haritasının hassasiyetine göre (Tablo 3.9.) derinlik tabanlı bölge hata payı eşik değerleri kullanılarak yeniden düzenlendiğinde Şekil 4.26.'de görülen sonuçlar elde edilmiştir.

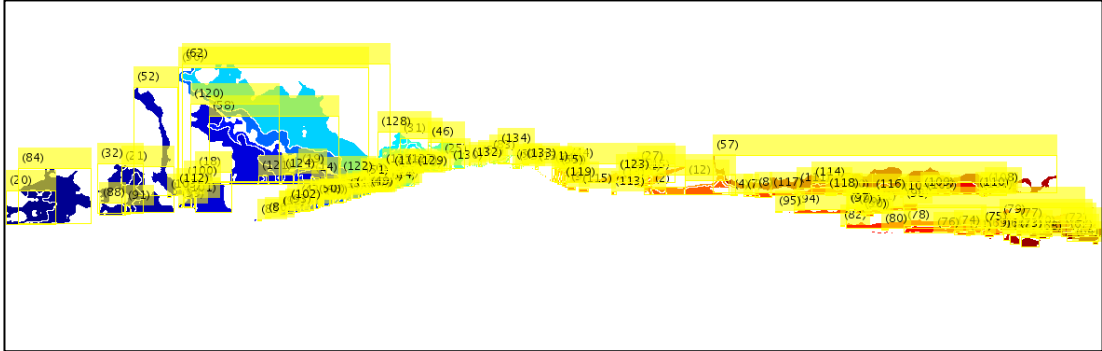


Şekil 4.26. Uzaklık bilgisine göre bulunan AG bölgesi (üst), yer düzlemi kesimiyle elde edilen AIR bölgesi (alt).



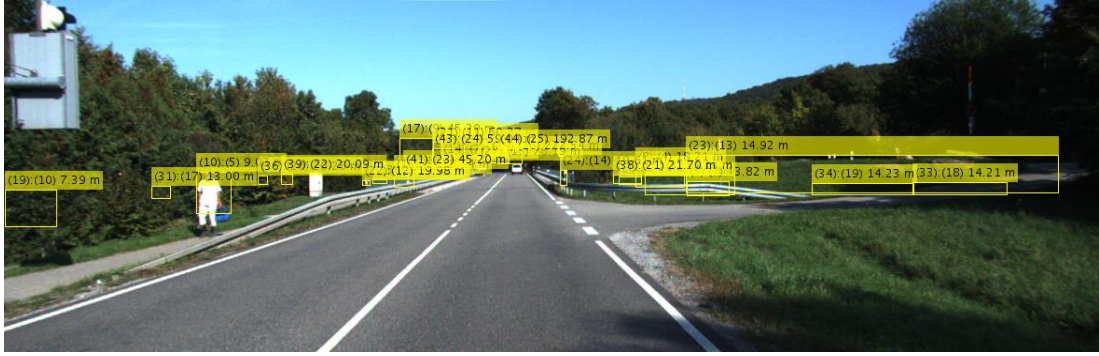
Şekil 4.27. BBA ile AIR bölgesindeki engellerin birbirinden ayrılarak birbirinden bağımsız engel nesnelerinin ÇS (Çerçeve Sınırlar) yöntemiyle gösterimi.

Şekil 4.27. ile elde edilen ve birbirinden uzaklıklarına ve komşuluk bağlantılarına göre ayrılan nesneler, yöntem bölümünde bahsedilen şekilde öncelik sırasına konulduğunda Şekil 4.28.'de gösterildiği sırada yaya tespit filtresine gönderilmiştir.



Şekil 4.28. Öncelik listesi sırasına göre AIR bölgesindeki nesnelerin (engellerin) sıralanması ve etiketlenmesi (yaya tespit filtresine gönderim sırası).

Şekil 4.28.'de elde edilen veriler girdi olarak kullanılarak, Şekil 4.29.'daki HOG yaya filtresi sonrasında pozitif sonuç veren çıktılar elde edilmiştir. Dikkat edilecek olursa hataların genellikle bölümlenmemiş yanlış oranlanan geniş bölgeler, ağaçlar veya fazla eğim bilgisi barındıran bölgeler olduğu görülebilir. Ayrıca çok uzaktaki küçük alan içeren bölgelerde, yaya tespiti için gerekli bilgilerin ölçekleme esnasında çok fazla kayba uğramasından kaynaklı olabilmektedir. Fiziksel ekipman sağlandığı takdirde, farklı uzaklıklarda yaya bölge genişlikleri ölçümlenerek hatalar azaltılabilir.



Şekil 4.29. Uzaklık tabanlı renk öncelikli engeller üzerinden yapılan yaya tespit sonucu.

Şekil 4.29.'da elde edilen sistem çıktısı görüntüsü Şekil 4.28.'den elde edilen veriler sayesinde tespit edilmiştir. Yanlış pozitif görüntü çerçeveleri, Şekil 4.30.'da görüldüğü gibi yol üzerinde olmayan ve engellerden oluşan bölgelerden oluşmaktadır. Yaya filtresi eğitim hedefi yüksek pozitif tanımlama oranı olduğundan dolayı yanlış pozitif görüntüler sistemde sıklıkla görülebilmektedir. Ayrıca yatay uzunluğu fazla olan alanların hata oluşturmasının bir diğer sebebi de ölçeklemeden dolayı eğim değerlerinin değiştiği ve bu sebeple yanlış pozitif değerler ürettiği görülmüştür.



Şekil 4.30. '355' numaralı veri tabanı görüntüsü engel haritası çıktısı.

4.4. HOG filtresi test ve sonuçları

HOG filtresi oluşturulurken, görüntülerden HOG özellikleri çıkarılarak SVM ile eğitim ve testleri gerçekleştirilir. Bu noktada eğitimin nasıl yapıldığı ve eğitimde kullanılan örnekler önem kazanmaktadır.

SVM ile Bölüm 3.4.'te belirtildiği gibi birçok farklı veri seti kullanılarak eğitim yapılmış ve başarımları test edilmiştir. Eğitim verisinin çokluğu başarı oranı konusunda daha iyi bir sonuç vereceği düşünülse de yapılan testlerde, özellikle zorlu görüntülerin sayısının artmasından dolayı başarı oranının daha da azaldığı tespit edilmiştir. Ayrıca INRIA yaya veri seti oluşturulurken kullanılan piramit eliminasyon yöntemi, tez çalışması için çekilen görüntüler, INRIA, MIT ve Daimler yaya veri setlerinin birleşimi için de kullanılarak Şekil 4.31.'deki sonuçlarla eğitim setinin verimliliği artırılmaya çalışılmıştır. Fakat burada dikkat edilmesi gereken eğitimde yayanın bulunamama durumu oluşmasındansa daha yüksek sayıda yanlış pozitif (YP) görüntülerin oluşması kriter olarak seçilmiştir. Bu sebeple zorlu pozitif (hard positive) örneklerin eğitimde kullanılması ve zorlu negatif (hard negative) örneklerin çıkarılması yanlış pozitif görülme oranını artırmıştır. Ayrıca bu başlık altında incelenen uzaklık tabanlı ölçekleme sonrası elde edilen sonuçlarda, yanlış pozitif oranının özellikle ufuk noktasına doğru yaklaştıkça arttırdığı görülmüştür. Bir diğer önemli nokta ise, yapılan testlerde HOG filtresinden elde edilen sonuçlarda; ağaçlar, araç tekerlekleri ve yol tabelalarının yanlış pozitif çıktı verdiği gözlemlenmiştir.

```

Command Window

Total Positive Number of Images: 20126
Positive Training Image Number : 19000
Total Negative Number of Images: 35474
Negative Training Image Number : 26850
Positive Test Image Number : 1126
Negative Test Image Number : 8624
trialNimage Count: 8624
coreNimage Count: 0
True Positive: 1092 - False Positive: 34
True Negative: 8455 - False Negative: 169
Positive Predictive Value: 0.9698
Negative Predictive Value: 0.9804
fx >>

```

Şekil 4.31. HOG+SVM filtresi test sonuçları INRIA, MIT, Daimler veri setleriyle.

Tez çalışmasında, test görüntülerinde yaya tespit işlemi yaptığımızda hataların daha çok önceki başlıklarda ele alınan AIR bölgesinde toplandığı görülmüştür. Yol

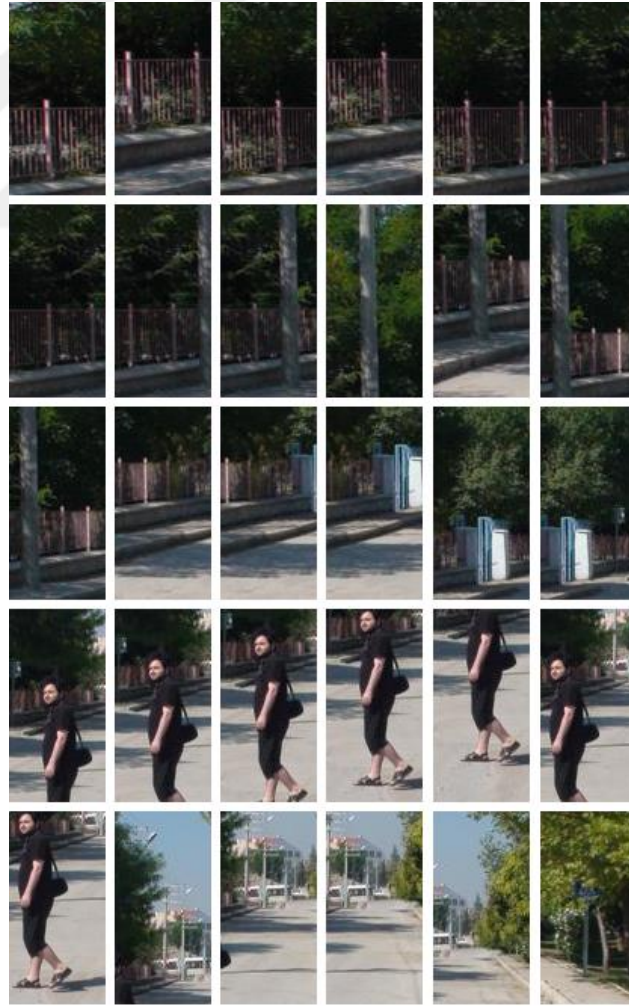
üzerindeki nesnelerin eğimlerinin, görüntü üzerindeki diğer alanlara oranla daha fazla değişim gösterdiği gözden kaçırılmamalıdır. Bu sebeple eğitim ve testlerde kullanılan görüntülerin ve testlerin başarımının yol üzerinde olmayan alanlardan dolayı daha yüksek çıktığı belirlenmiştir. Şekil 4.31.'de görüldüğü gibi sonuçların testlerde ve literatürde sayısal olarak yüksek başarımlarla ifade edilebilmelerine karşın (genellikle %90 ve üzeri) bu oranların, eğim değişimlerin yüksek olduğu bölgelerde azaldığı, ölçekleme koşullarında ise daha düşük başarımlı sonuçlar elde edildiği görülmüştür. Yol üzerindeki alanlardaki bu yüksek değişimler ve nesnelerin birbirinden ayrımı için daha yüksek başarımlı elde etmek amacıyla, sadece AIR bölgesindeki farklı uzaklıklarda yaya ve nesneleri içeren bir veri seti oluşturarak eğitim yapmak ve başarımların ölçmek daha sağlıklı sonuçları içeren bir sistem oluşturmak ve karşılaştırma yapabilmek için gereklidir.



Şekil 4.32. HOG filtresini test etmek için kullanılan test görüntüsü.

Bahsi geçen AIR bölgesinde hatalı yaya tahminin çokluğu ile ilgili durumu, herhangi bir görüntü üzerinden inceleyecek olursak; Şekil 4.32.'deki bütün görüntü ele

alındığında ve HOG filtresinden geçirildiğinde Şekil 4.33.'deki gibi çıktılar elde edilmektedir. Dikkat edilirse AIR bölgesi olarak tanımladığımız yer düzlemi üzerindeki bölgelerden, tüm görüntüye göre daha çok pozitif sonuç elde edilmektedir. Bu sebeple tez çalışmasında bulunan hatalı bölgelerde, HOG özellik çıkarımından ve eğitimdeki seçimlerden kaynaklanan hatalar sonuçlara yansımaktadır. Ayrıca bazı hataların da HOG filtresi dışında, uzaklığa bağlı alanlarda uzaklık bilgisi kullanılarak yapılan nesne bulma işlemlerinde, benzer uzaklıktaki yakın nesnelerin hepsinin bir alınmasından ve alansal olarak yaya olması muhtemel olmayan alanların çıkarılmasından dolayı istisnai durumların olduğu gözlemlenmiştir. Bu sebeple HOG filtresine gönderilen ve gönderimden önce eşik değerinden geçirilen alanların da ön incelemeye tabi tutulması veya paralel olarak işlenmesi, hataların azaltılması için önerilmektedir.



Şekil 4.33. Test görüntüsünden elde edilen pozitif HOG filtre sonuçları.

Şekil 4.33.'de HOG yaya filtreleme sonrası elde edilen sonuçlar içerisinde, yer düzlemini barındıran bölgelerin hatalı olarak yaya tespitinde pozitif sonuç verebildiği gözlemlenmiştir. Tez kapsamında stereo görüntü kullanımıyla yer düzlemi tespit edilip bu alanların çıkarılmasından kaynaklı bu tarz yanlış pozitif sonuçların oluşması engellenerek başarı oranı artırılmıştır. Diğer alanlar içerisinde ağaç, direk ve demir parmaklık gibi nesneler hatalara sebep olmaktadır. En boy oranlarının antropometrik oranlara uyduğu durumlarda direk veya çitler gibi engeller, hataların oluşmasına neden olabilmektedir. Fakat antropometrik oranların kullanımı sayesinde bu oranlar dışında kalan engellerin hataya düşmeleri engellenmiştir.

Yapılan testlerde HOG filtresinin yaya boyutuna yüksek duyarlılık gösterdiği görülmüştür. Bu sebeple, uzaktaki veya yakındaki engel görüntüleri gerçekte yaya barındırsa bile HOG filtresi kullanıldığında negatif sonuç elde edildiğinden, filtre işlemi öncesinde görüntüler uzaklıklarına göre ölçekleme işlemine tabi tutulmuştur.

Görüntüde ölçeklemenin başarımlı üzerindeki etkisi, uzaklık tabanlı dolgu tanımlamaları ile birlikte KITTI veri seti görüntüleri üzerinden incelenmiştir. Şekil 4.34. üzerinde verilen örneklerde görüldüğü gibi sistemce bulunan değişik çözünürlük ve çerçevedeki yaya görüntüleri, farklı sonuçlar ortaya koymaktadır. İlk satır ikinci sütunda yer alan orijinal test görüntüsünden elde edilen yaya görüntüsü, uzaklığa göre ölçeklenerek ilk satır ve ilk sütunda görülen şekilde HOG filtresinde pozitif sonucu vermektedir. İkinci satırdaki görüntülerde ise tam tersi bir şekilde aynı algoritma uygulandığında ikinci satır birinci sütundaki orijinal görüntü pozitif sonuç verirken ikinci satır ikinci sütundaki ölçeklenen görüntü hatalı sonuç vermektedir.



Şekil 4.34. Sol sütun gerçek pozitif (GP), sağ sütun yanlış negatif (YN) yaya tanıma sonuçları.

Başlangıçta farklı uzaklıklar için belirli aralıklarla 4 farklı çözünürlük için yaya filtresi oluşturulması planlanmıştır. Fakat bu farklı uzaklıklara göre eğitim seti bulunmamasından ötürü, veriler ölçeklenerek eğitim yapıldığında sonuçların hata oranının yüksek olduğu görülmüştür. Bu sebeple kullanılan yaya veri setlerinde verilen orijinal çözünürlükler kullanılarak tek bir yaya filtresi eğitilmiş, elde edilen görüntülerin yaya olma durumları uzaklıklarına göre yaya tespiti sırasında ölçeklenerek sonuçlar elde edilmiştir.

Şekil 4.34.'de görülen, orijinal veya uzaklığa bağlı ölçeklenen görüntüler üzerinde yapılan çerçeve sınır belirleme işlemlerinden kaynaklanan hatalar, tez içerisinde iki farklı çerçevenin ve ölçeklemenin paralel olarak HOG filtresine gönderilerek ardından 'veya' lojik işlemine tabi tutulmasıyla çözülmüştür.



Şekil 4.35. Farklı dolgu değeri ile alınmış yaya görüntüleri (Üst) ve ölçeklenmiş sonuçları (alt).

HOG yaya tespitinde gerçek pozitif sonuçların oranının artırılması için yapılan işlemler, daha fazla yanlış pozitif oranını da beraberinde arttırmış olsa da HOG filtresinin çözünürlük ve dolgu değişimlerine karşı hassasiyetini koruduğu gözlemlenmiştir. Şekil 4.35. üzerinde birinci satırda iki farklı dolgu (padding) değeri alınarak ikinci satıra karşılık gelen görüntülere ölçeklenmesine rağmen HOG sonuçları negatif olmaktadır. Alınan dolgu değerleri, Şekil 4.36.'daki gibi olduğu takdirde ölçeklenen sağ görüntü HOG filtresinde pozitif olarak sonuç vermektedir.



Şekil 4.36. Farklı dolgu değeri ile ele alınan bir diğer örnek (sol) ve ölçeklenmiş pozitif yaya sonucu (sağ).

Yaya tespit işlemine etki eden faktörlerden bir diğeri ise, filtreye gönderme işlemi öncesinde AIR bölgesindeki nesnelerin büyüklük ve uzaklıklarına göre engeller, yaya ve bisikletlilerin daha doğru tahmini için antropometrik oranlara göre eşik değerinden geçirilmektedir. Sonrasında ise olası yaya bölgeleri öncelik sıralarına

göre belirlenerek çerçeve değerlerinin oranının belirlenmesi ile uzaklık tabanlı ölçekleme yapılmaktadır.

Uzaklık tabanlı alansal olarak çerçevede dolgu ekleme işlemi ve çözünürlük açısından eşitleme işlemi yapılırken HOG filtresinin çok hassas bir şekilde sonuçlar ürettiği görülmüştür. Bu hassasiyetten dolayı sonuçlarda yaya olmayan bölgeler yanlış pozitif, yaya olan bölgeler ise yanlış negatif şeklinde çıktı üretebilmektedir. Daha iyi bir filtre tasarımı için her uzaklıkta stereo kameralarla çekilmiş insan görüntüleri ile eğitim ve testler gerçekleştirilerek başarımlar test edilmelidir. Bu sayede farklı koşullarda uzaklık ve antropometrik oranlar daha net bir biçimde belirlendiğinden ölçekleme oranları daha sağlıklı yapılabilecektir.

4.5. Tüm sistemin başarısı ve elde edilen veriler

Tez çalışması kapsamında geliştirilen yöntemlerin ve kullanılan sistemlerin ürettiği çıktılar üzerinden elde edilen pozitif yaya bölgeleri ve uzaklık sonuçları bu alt başlık içerisinde incelenmiştir. Çıktılarda, yaya tespit sonuçları ve uzaklıkları kullanılan test görüntüleri üzerinde belirtilmiştir. Test görüntülerinde, gerçek pozitif yaya bölgeleri başarılı bir şekilde bulunurken belirli koşullar altında yanlış negatif sonuçların da elde edildiği gözlemlenmiştir. Yanlış negatif çıktı veren yaya sonuçlarının nedenleri belirlenerek geliştirilmesi planlanan çalışmalar olarak Bölüm 4.6. içerisinde incelenmiştir. Unutulmamalıdır ki tez kapsamında geliştirilen engel tespitinde yüksek başarımlı sonuçlar elde edildiğinden, yayaların tespit edilememesi durumunda dahi bu bölgeler engel olarak tanımlanmakta ve sistem içerisinde dikkat edilmesi gereken nesneler olarak belirtilmektedir. Veri seti kullanılarak çoklu kareler yerine tekli stereo kare görüntü çifti üzerinde çalışılmıştır. Bu sebeple sunulan sonuç ve çıktılar, herhangi bir ön bilgi kullanımı olmadan elde edilen başarımları göstermektedir.



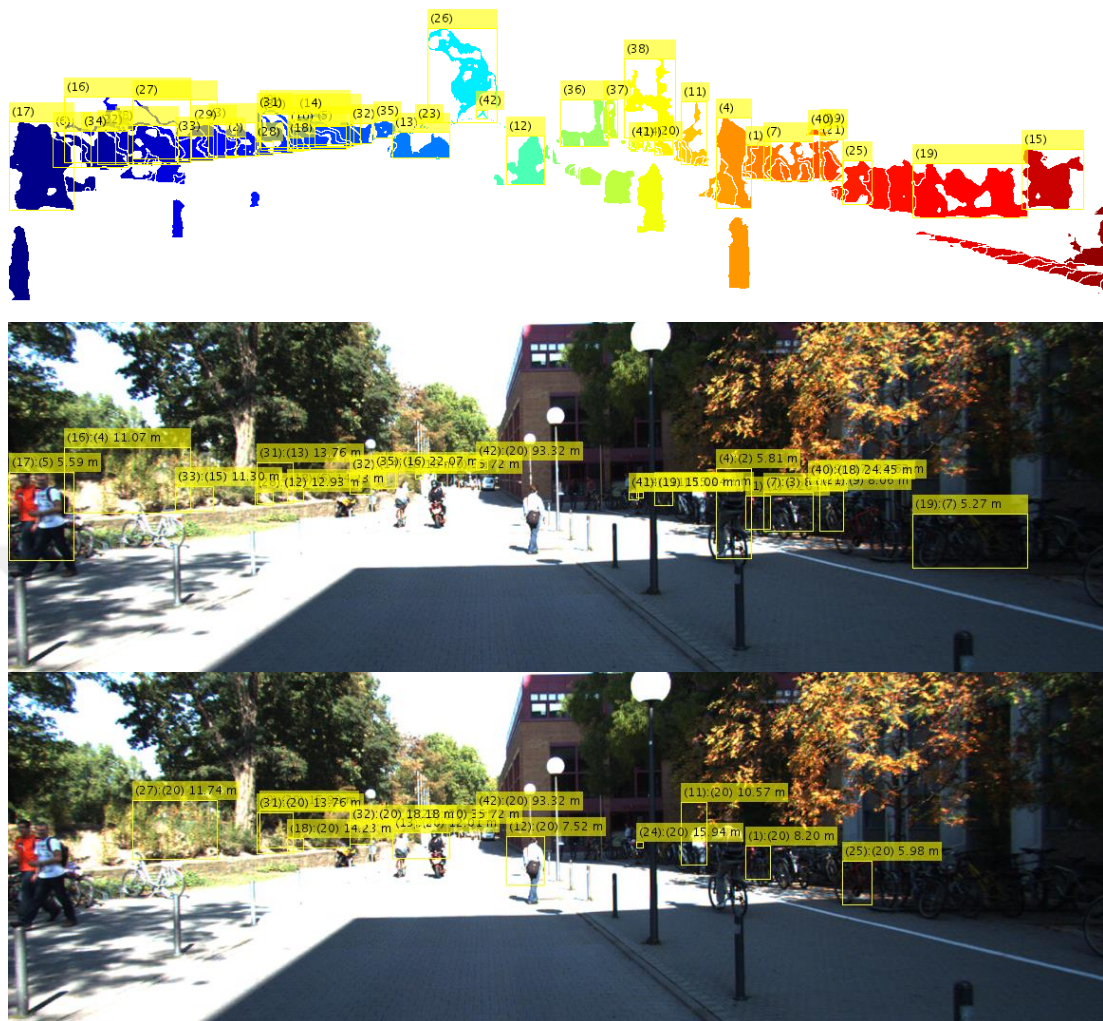
Şekil 4.37. Uzaklık tabanlı alansal eşikleme öncesi elde edilen pozitif HOG sonuçları ve uzaklıkları.

Şekil 4.37.'de alansal eşikleme yapmadan önce iki farklı test görüntü verisi sonucunda elde edilen çıktılarda görüldüğü gibi uzak noktalarda çok büyük veya yakın noktalarda yaya olamayacak kadar çok küçük alanların olduğu ve bunların işlenerek yanlış pozitif değerler verdikleri görülmüştür. Bu sebeple yayaların oran ve orantılarına yani antropometrik oranlara göre alan hesabı yapılmış; bisikletli, motosikletli insanları da kapsayacak şekilde değerler yaklaşık olarak belirlenmiştir. Yayaların büyüklükleri dolayısıyla alanları uzaklığa göre değiştiğinden materyal ve yöntem bölümündeki Tablo 3.11. oluşturularak yayaları barındırabilecek alanlar, çıkabilecek hata oranlarının tahmini etkisi göz önünde bulundurularak oluşturulmuştur. Yapılan bu işlemler sonucunda ise Şekil 4.38.'de görüldüğü gibi uzak mesafedeki yaya barındırması muhtemel olmayan alanlar çıkarılarak başarı oranı artırılmıştır.



Şekil 4.38. Sistem sonucu elde edilen pozitif yaya sonuç ve uzaklıkları.

Dikkat edilirse Şekil 4.37. ve Şekil 4.38.'deki görüntülerde mavi renkli aracın olduğu bölgelerde ve diğer sonuç görüntülerinde de gözlemlendiği üzere yatay olarak uzun engel barındıran alanlar ölçeklemeden dolayı, HOG özellikleri çıkarımından sonra yanlış pozitif değer verdikleri görülmüştür. Fakat bu uzun genişliğe sahip alanların çıkarılması da yayaların tespitinde hatalara sebep olacaktır. Örneğin birden fazla yayanın yürümesi, yayaların nesnelerin yanında veya önünde olmaları durumunda bu yayalar tespit edilmeden tanımlama işleminden çıkarılmış olacaktır. Bu hataların, yatay olmayan ve uzaklığa bağlı alansal kesimin yapıldığı durumlarda da görülebildiği tespit edilmiştir. Bu tarz hataların giderilmesi için, derinliğe bağlı yaya çerçeve değerlerinin gerçek hayatta ölçümlerinin yapılması ve sonrasında belirlenen uzaklık-yaya çerçevelerinin bu alanlar içerisinde gezdirilmesiyle yayaların tespit edilmesi gerekmektedir. Dikkat edilirse antropometrik oranlar, sadece engeller üzerinde incelendiğinden engeller yaya olarak kabul edilip ölçeklenmiştir. Bu sebeple incelenen bölgenin, uzaklığa bağlı yaya uzunluk ve genişlik değerlerine uyup uymadığı ölçüm değerleri olmadığından incelenememiştir. Veri seti kullanımı zorunluluğundan dolayı bu işlem gelecekte yapılması planlanan çalışmalar kısmına dahil edilmiştir.



Şekil 4.39. Eşikleme sonrası alansal HOG öncelik listesi (üst), iki farklı dolgu kullanıldığında HOG eğitim sonucu (orta, alt).

HOG filtresinin yaya tespiti sonuçlarında, girdi görüntü oluştururken belirlenen dolgu ve uzaklığa bağlı ölçekleme oranlarına göre çok farklı sonuçlar ortaya koyabildiği Bölüm 4.4.'de incelenen test görüntülerinde görülmektedir. Bazen bu hatalar bir grup yayanın tespitinde çok iyi sonuç verirken diğer yayaların tespitinde negatif sonuçlar verebilmektedir. Şekil 4.39.'daki görüntülerde, orta ve alt kısımda bulunan sistem çıktı verileri bu anlatılan durumu yansıtmaktadır. Bu hataların en aza indirilmesi için, HOG filtresi eğitiminde yüksek pozitif sonuç elde edilmesi amacıyla zorlu negatif görüntüler eğitim setinden çıkarılmıştır. Bunun sonucunda filtrenin yanlış pozitif bulma oranı artmış olsa da yanlış negatif tespit oranının azaltılması sistem önceliği olarak belirlenmiştir. Tespit edilen yanlış pozitif alanlar, yaya olmasa

bile engel teşkil eden bölgeler olduğundan aracın kaza yapmasını önlemek için bu alanlara dikkat edilmesi gerekmektedir. Şekil 4.40. ve bu başlık altındaki devam görüntülerdeki sistem çıktı sonuçları '15.11.2017' tarihinde elde edilmiştir.

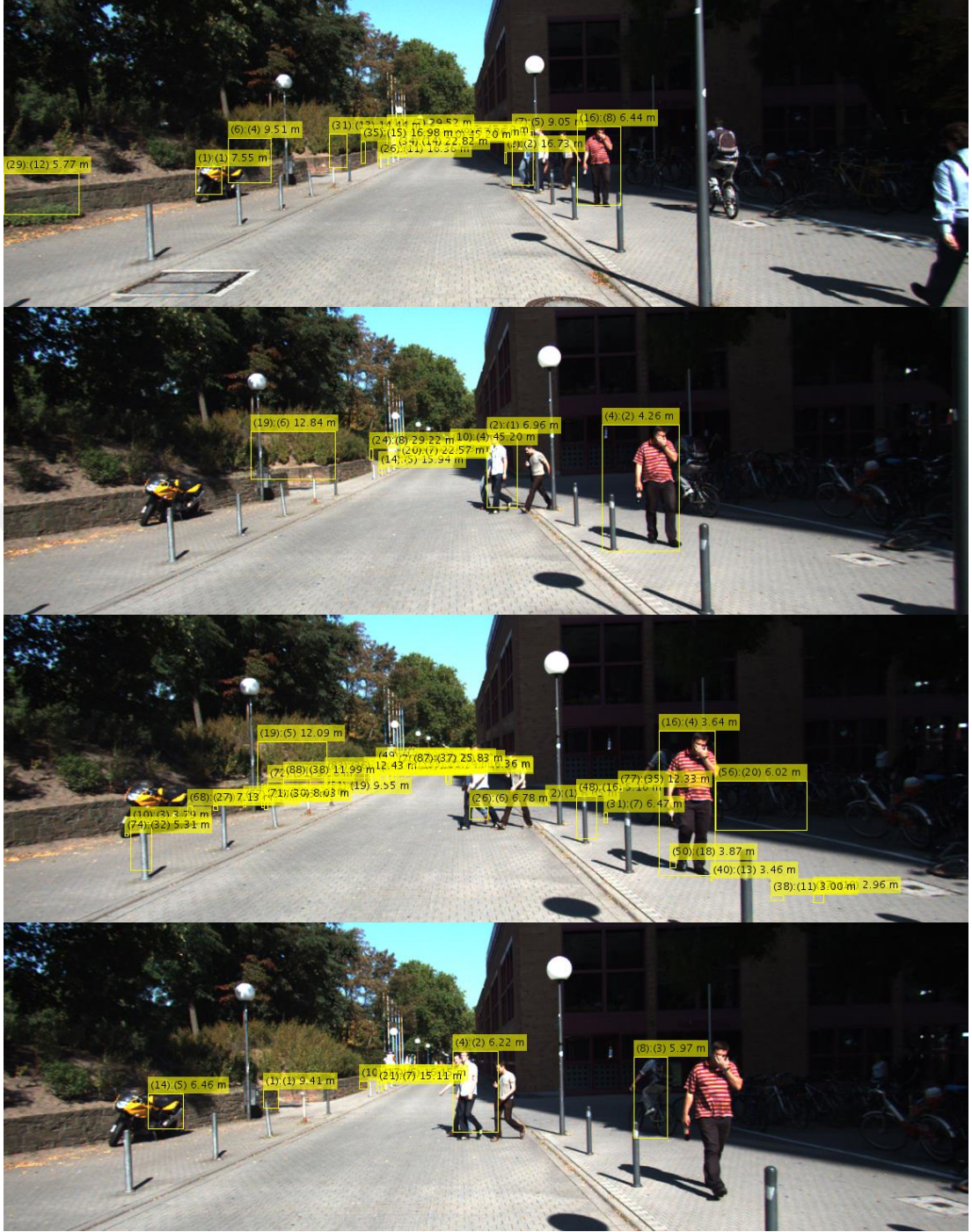


Şekil 4.40. '2011_09_26_drive_0070' veri seti görüntüleri sistem sonuç çıktı örnekleri.

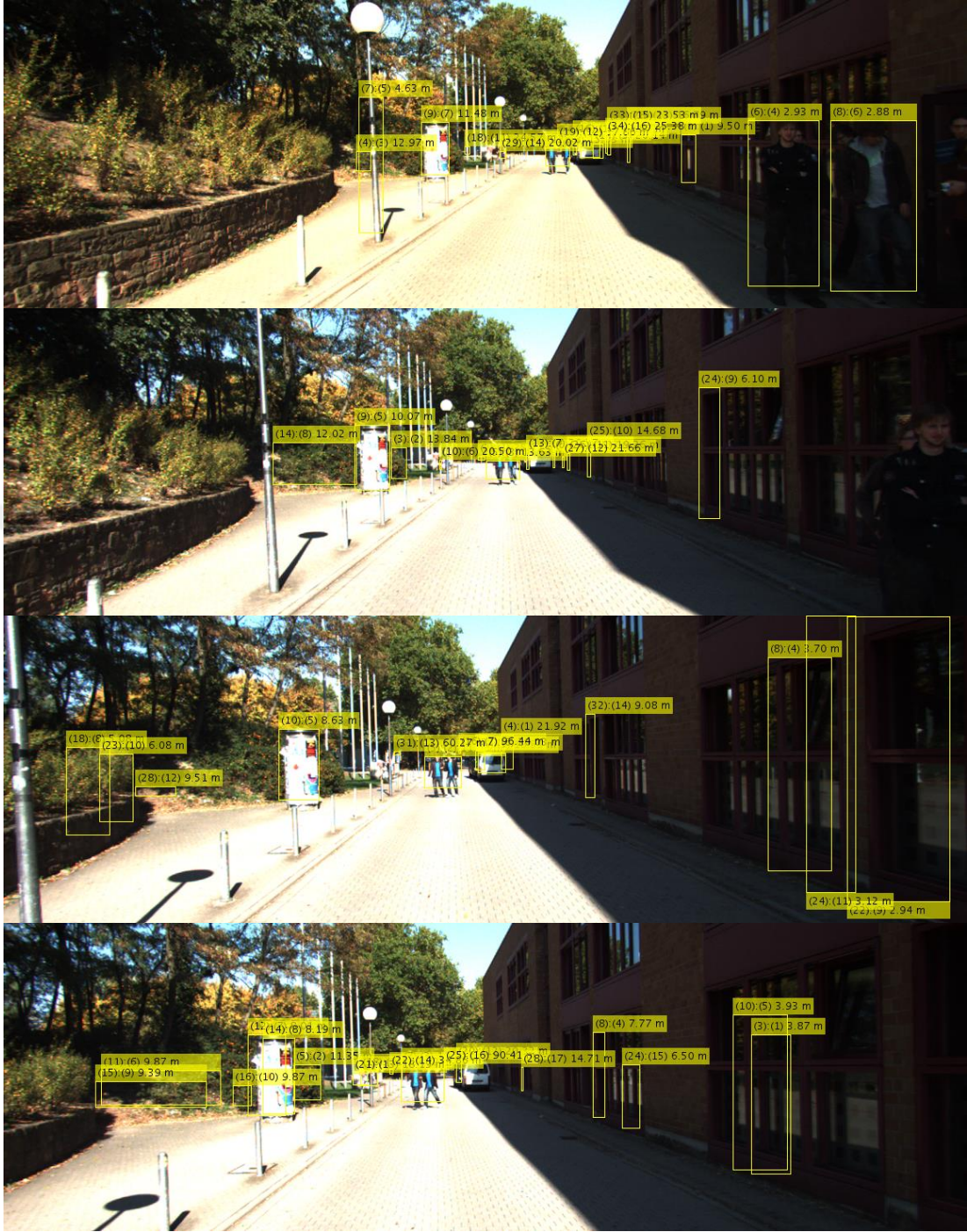


Şekil 4.41. '2011_09_26_drive_0070' ve '2011_09_28_drive_0038' üzerinden sistem sonuç çıktısı örnekleri.

Şekil 4.41. ve Şekil 4.42. üzerinde de görülebildiği gibi yayaların tespiti uygun bir biçimde gerçekleştirilebilmektedir. Bazı engellerin ise yaya tespitine yaya ile birlikte tek bir kare içerisinde ele alındığı da görülebilmektedir.

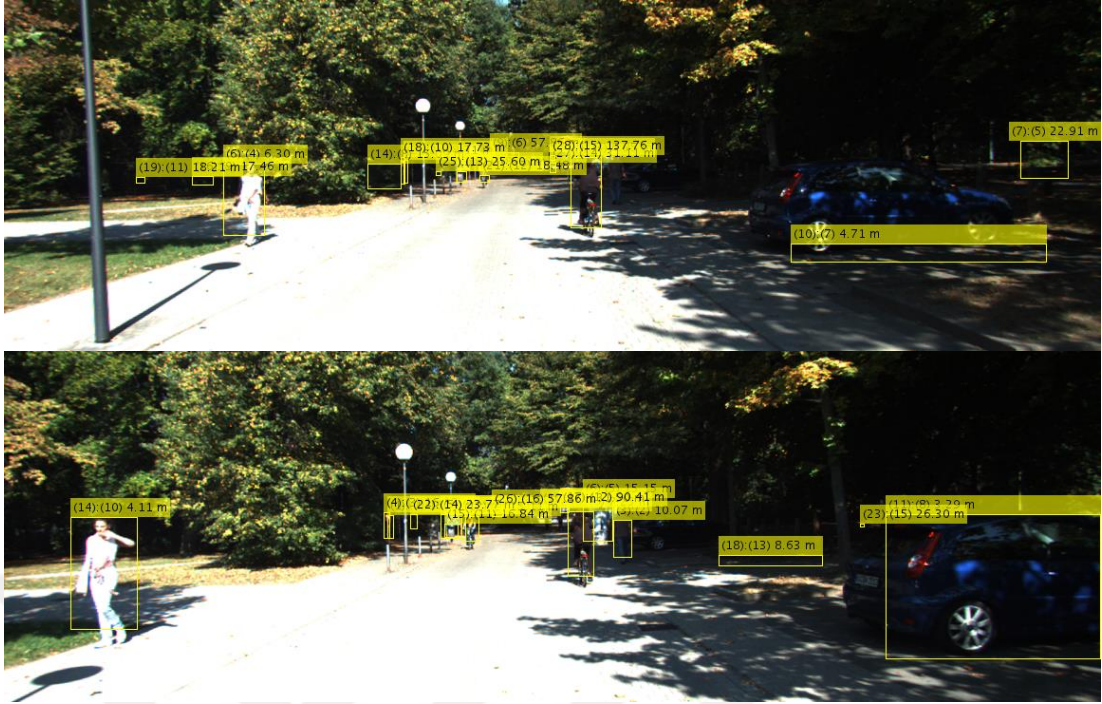


Şekil 4.42. '2011_09_28_drive_0038' veri seti sistem sonuç çıktı örnekleri.



Şekil 4.43. '2011_09_28_drive_0045' veri seti üzerinden elde edilen sistem sonuç çıktısı örnekleri.

Şekil 4.43. ve Şekil 4.44. üzerinde görülen test sonuçları, farklı yol ortamlarında algoritmanın performansının anlaşılması açısından önemlidir.



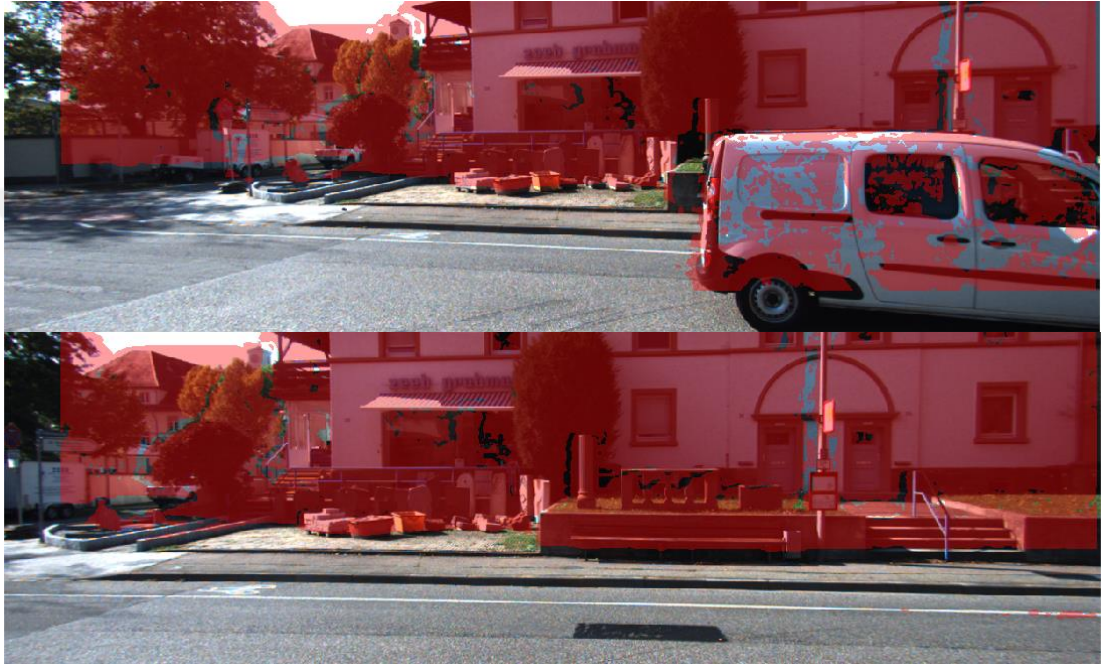
Şekil 4.44. '2011_09_28_drive_0047' veri setleri giriş görüntüleri sistem sonuç çıktı örnekleri.



Şekil 4.45. '2011_09_26_drive_0070' örnekleri ile dönüşlerdeki sistem sonuç çıktıları ve performansı.

Şekil 4.43. ve Şekil 4.44. görülen farklı yol koşullarındaki sonuç çıktıları önemli olmakla birlikte, araçların dönüş esnasında sistemin elde ettiği sonuçlar da en az

bunlar kadar önemlidir. Kazaların birçoğunun aracın dönüşü sırasında, yaya ve engellerin sürücü tarafından görülmemesinden kaynaklandığını dikkate alacak olursak, sistemin bu koşullarda gösterdiği performansın önemi daha iyi anlaşılabilir. Bu sebeple test görüntülerinde dönüşlerde yaya olmamasına karşın sistem sonuçları ve tespit edilen engellerin başarımı, Şekil 4.45. sonuç çıktıları ve Şekil 4.46. engel tespit karşılıkları incelenerek görülebilir.



Şekil 4.46. '2011_09_26_drive_0070' örnekleri ile dönüşlerde sistem tarafından tespit edilen engeller.

Sistem sonuçlarında görüldüğü üzere; yaya tespiti, uzaklık tespiti ve engel tespit işlemleri gerçekleştirilmiştir. Tezde oluşturulan sistem, araçtaki kameralarla elde edilen çevresel verilerin işlenmesiyle hedeflenen kriterlere göre başarılı çıktılar elde ederek araç sürüş destek sistemlerinde veya otonom araçlarda kullanılabilecek bir mimari ortaya koymuştur.

4.6. Gelecekte yapılması planlanan çalışmalar

GDD kümeleme ve renk segmentasyonunda; bütün verilerde noktalar arası hesaplamaların yapılması yerine 'Quadtree' veri yapısını kullanımı, verilerin yoğun bulunduğu bölgelerde değerlerin veri sayısınca tahmini, iki boyutlu matris üzerinden

sabit filtre kullanımıyla toplam Gauss değerlerin atanması, yapılan işlemlerin hızını artıracaktır. Veri barındırmayan bölgelerin veya birincil dereceden ilişki barındırmayan verilerin incelenmesi yerine, yoğunlukların tahmini olarak belirlenmesi ve bu değerlerin görüntü üzerinde filtre oluşturularak atanması, kümelemeyi kullanan alt sistem elemanlarının hızının artırılması için önerilmektedir. Çok boyutlu veri uzayında ise Quadtree içerisinde ilişki derecesine göre yaklaşık değerlerin kullanımı, işlem hızını aynı şekilde artıracaktır. Tez içerisinde kümelemede, görüntü piksel komşuluklarına bakılmadan uzaklıkların ve yoğunlukların hesaplanması işlem zamanını artırmaktadır. Bu sebeple kümelemede iki boyutlu görüntü verileri için iyileştirme yapılması tez sisteminde performans artırımı için gereklidir.

Engelleri ve yer düzlemini tanımlayan doğrunun belirlenmesi sırasında yükseklik tahmin işlemleri yerine kameranın araç üzerindeki yükseklik, açı gibi konum bilgileri kullanıldığında daha hızlı bir doğru tespiti yapılabilecektir. Belirlenen yer düzlem doğrusu yükselti parametresinin aracın seyir halindeyken tekrar tekrar hesaplanması yerine hareket sensörü yükselti değişim verilerinin kullanımı, aracın yükseklik değişimini gerçek zamanlı ölçerek doğrunun buna göre güncellenmesi sonuç bulma hızını ve doğruluğunu artıracaktır.

Yer düzlemi doğrusu, tez çalışması içerisinde lineer olarak tanımlanmış ve bu hesaplamalara göre testlerde kullanılmıştır. Eğim içeren yollarda sonuçların geliştirilmesi için U-disparity ve GDD renk segmentasyon yöntemleri kullanılabilir. Aracın yol üzerinde olması gerektiği varsayımından yola çıkarak yol segmentini öncelikli ve ağırlıklı olarak hesaplayarak, ağaç, eğim ve diğer engellerin oluşturabileceği hata paylarının giderilmesi veya azaltılması sağlanarak daha doğru sonuçlar elde edilebilir.

Uzaklığa bağlı hacim tahminlerinde, hacimlerin gerçek değerleri ile hata payları arasındaki ilişki test edilerek, daha ayrıntılı bölge hata payı eşik değerleri belirlenebilir. Bu sayede nesnelerin daha doğru tanımlaması gerçekleştirilebilecektir.

Derinlik haritası çıkarımı, Renk segmentasyonu işlemleri gibi bazı alt sistem elemanları, FPGA veya mikro işlemciler içerisine gömülü olarak entegre edilerek paralel bir biçimde hızlı olarak hesaplanabilir. Fakat bunun için öncelikle yukarıda belirlenen optimizasyonların gerçekleştirilmesi ve zamansal olarak testlerinin gerçekleştirilmesi gerekmektedir. Hızlı derinlik haritası tespiti, sistemin koşum hızını büyük ölçüde artıracaktır.

Kümeleme ve renk segmentasyonu alt sistemler olduğundan sonuçları materyal ve yöntem bölümünde incelenmekle birlikte, bu bölümde direk çıkış sonuçlarını ilgilendiren alanlar incelenmiştir. Renk segmentasyonu konusunda yer düzlemi içerisinde aracın gidebileceği yollar şu anki haliyle belirlenebilecek durumda olsa da mevsimsel ve yol değişimlerinin özelliklerinin kullanımı ile yol profili oluşturularak daha doğru gidilebilir araç yolu ve dikkat edilmesi gereken yaya yolu tahmin sistemi oluşturulabilir.

Ardışık görüntü karelerinin kullanımıyla kameranın uzamsal hareketinin tahmini Ego-motion olarak adlandırılmakla birlikte literatürde uzun yollarda stereo görüntü tabanlı konumlandırmada kullanılmaktadır [119]. Gelecek çalışmalarda Ego-motion yöntemi kullanarak aracın ortamdaki konum değişimi belirlenebilir. Bunun dışında araç hız sensörleri yardımıyla yaya ve engellerin değişimi tahmin edilerek göreceli hızları ve hareket vektörlerinin tahmini yapılabilir. Özellikle statik konumda bulunan ağaç veya levhalar ile hareket halindeki yayaların tespiti ve yayaların sonraki hareketlerinin tahmini sistemin daha güvenilir ve yüksek doğrulukta çalışmasını sağlayacaktır. Özellikle önceki kare görüntü verilerinin işleme alınarak gelecek olası engel bölgesi tahmini ve hız öneri sistemi gelecekte çalışılması planlanan konular arasındadır.

Derinlik haritasının tamamı yerine yer düzlemini kestikten sonra kalan uzaklık verileri kullanılarak alan genişletme işleminin yapılması nesne tespit başarımını artıracaktır. Çünkü yolun uzaklığı ile engellerin uzaklıkları aynı olduğunda yere bağlı nesneler birbirinden ayıramamaktadır. Uzaklığa bağlı alansal ayırım algoritması da gerçek hayat ölçümleri yapılarak geliştirilmelidir. Direk ve benzeri nesnelerin olduğu

alanlarda yayaların direk ile birlikte ele alınması hataya sebep olduğundan farklı uzaklıklardaki yayaların yükseklik ve genişlik değerleri ölçülerek sistemde aranılan yayanın yaklaşık alansal büyüklükleri belirlenebilir. Bu sayede incelenen nesnelerin alan büyüklüklerine göre parçalanarak veya bütün olarak ele alınması sağlanabilir.

Sürüş destek sistemlerinde sürücünün dikkat edemediği veya görülmesi zor noktalarda kameraların doğru pozisyonda konumlandırılmalarıyla sürücünün kaza yapma oranını azaltacak uyarılar sistem tarafından üretilebilir. Ek olarak, farklı nesnelerin tespitinin sisteme eklenmesi ile hızlı bir şekilde araç, trafik işaretleri gibi nesne tanımlamaları yapılabilir. Bütün bahsi geçen geliştirme ve optimizasyon alanındaki iyileştirmeler, gerekli destek ve kaynak bulunduğu takdirde hazırlanarak gerçek zaman kriterleri için daha uygun hale getirilebilir. Bu sayede stereo kamera sistemleri kullanarak araç sürüş destek sistemlerinde ve otonom araçlarda daha güvenli bir sürüş sağlanmış olacaktır.

BÖLÜM 5. TARTIŞMA VE SONUÇ

Bu tez çalışması, çalışma esnasında kullanıcı girişine ihtiyaç duymaksızın stereo kamera verilerini kullanarak yer düzlemini, engelleri ve yaya olabilecek bölgeleri uzaklık tahminleriyle birlikte otomatik olarak tespit eden bir sistemi tanımlamaktadır. Yaya tespit bileşeni HOG tabanlı filtreleme yardımıyla yaya olabilecek bölgeler tespit edilmiş, kamera parametreleri farklılıkları, veri tabanları ve farklı kameralarla elde edilen görüntü verileri kullanılarak uzaklık ölçüm tahminleri gerçekleştirilmiştir.

Stereo kamera tabanlı sistem; araç-yaya uzaklıkları, araç-engel uzaklıkları, renksel bölge ve nesne tanımlamalarıyla otonom araçlar ve sürüş destek sistemleri için tatmin edici sonuçlar sağlamaktadır. Sistemde giriş verileri tek kare stereo görüntüler kullanılarak gerçekleştirilmiştir. Bu sayede gelecekte uygulama aşamasında çok kare görüntüler üzerinde çalışıldığında tekrar hesaplama gerektirmeyen alanlar çıkarılarak hız artırımı yapılabilir şekilde sistem geliştirilmiştir. Bu noktada donanım kullanımı ve özelliklerine göre sisteme girilmesi gereken sabit değerler noktasında da adaptif değer tahmin algoritmaları oluşturulmuştur. Uzaklığa bağlı olarak piksel boyutlarına göre gerçekte nesne boyutlarının değişimi göz önüne alınarak uzaklığa göre alan ve oran bazlı eşik değerleri uygulanmıştır. Bu sayede engel ve nesneler ile yaya olabilecek bölgeler birbirinden ayrılmıştır. Fakat bu yaklaşımdan kaynaklı direk ve ağaç gibi yayadan daha büyük nesnelerin yanında bulunan yayalar engeller içerisinde tanımlı kalabilmektedir. Bu konuda alansal eşik değeri uygulamak yerine yüksek alan barındıran bölgelerin parçalar halinde analizi daha fazla işlem gücü gereksinimi karşılığında yapılabilir.

Öne sürülen sistem görsel verilerin kullanımıyla renk ve uzaklık tabanlı yaya, engel ve bölge tespitini sağlamaktadır. Yaya olabilecek bölgeler için değişen uzaklıklara göre seçilecek alanların olası insan boyutlarına uygun olabilmesi için antropometrik ölçümler kullanılarak uzaklık tabanlı orantısal bölge seçimi işlemi uygulanmıştır. Ayrıca renk bölgeleri kullanılarak ve renksel bölgelerin çevresel değişimlerine göre uzaklıklar göz önünde bulundurularak yaya tespitinin daha başarılı ve daha hızlı gerçekleştirilebilmesi için öncelikli işlem listesi oluşturulmuştur. Sistem çıktıları olarak alınabilecek sonuç verileri; olası yaya içerir bölgeler ve tahmini uzaklıkları, engel teşkil eden bölge ve uzaklık tahminleriyle nesnelerin renklere göre ayrımı ve renksel özelliklere göre belirlenmiş yer düzlemini barındırır. HOG yaya filtresi dışında sistem herhangi bir eğitim, gözetim veya parametre girişine ihtiyaç duymamaktadır. Daha yüksek doğrulukta uzaklık değerleri elde edilebilmesi için ise içsel kamera parametrelerinin bilinmesi ve deneysel araç-kamera konumlandırılışına göre testlerin yapılması önerilmektedir.

Tez kapsamında yeni olarak geliştirilen ve önerilen yöntemler kullanılarak olası yaya ve engel bölgeleri tespit edilmiştir. Araçların karıştığı yaya kazalarının önlenmesinde kullanılması amacıyla oluşturulan yöntem ve uygulamalar veri tabanlarındaki görüntü örnekleriyle test edilmiş ve sistemin kullanılabilirliği ortaya konulmuştur. Gerçek hayat verilerinden elde edilmiş veri tabanı görüntüleri kullanılarak yapılan bu çalışma ile kamera tabanlı yaya tespitinde, çevresel nesne ve bölgelerin tespitinin önemli olduğu gösterilmektedir. Elde edilen sonuçlar ile bu alanda ilerleme sağlanabilecek birçok uygulamanın önünü açılmıştır. Elbetteki sistemin daha fazla geliştirilmesi için gerçek zaman kriterlerine uygun optimizasyon ile farklı yol ve koşullar altında yapılacak testlere ihtiyaç duyulmaktadır. Fakat oluşturulan sistemin şu anki haliyle başarılı sonuçlar elde ettiği görülebilmektedir. Bu sebeple bu alandaki ilerlemelerin kazalar nedeniyle oluşan maddi ve manevi kayıpların önüne geçilmesi için geliştirilmesi ve uygulanması önem arz etmektedir. Umarız ülkeler bazında milyar dolarlarla ölçülen maddi kayıpların, geri alınamayacak hayati kayıplar ile sakatlanmaların ve dolayısı ile getirdiği iş gücü kaybı ve manevi kayıpların önüne geçilmek için atılan adımlar hedefine ulaşır ve yapılan bu alandaki çalışmalar başarılı bir şekilde gerçek hayata geçirilerek uygulanır.

KAYNAKLAR

- [1] WHO, «World Health Organization, World report on road traffic injury prevention,» cilt ISBN: 9241591315, 2004.
- [2] DOT, «Department Of Transportation (US) National Highway Traffic Safety Administration (NHTSA). Traffic Safety Facts 2013: Pedestrians.,» 2015.
- [3] J. MacQueen, «"Some methods for classification and analysis of multivariate observations." Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1. No. 14,» 1967.
- [4] Bradley, P. S., O. L. Mangasarian ve v. W. N. Street, «"Clustering via concave minimization." Advances in neural information processing systems.,» 1997.
- [5] Jain, A. K. ve v. R. C. Dubes, «Algorithms for clustering data. Prentice-Hall, Inc.,» 1988.
- [6] Kaufman, Leonard ve v. P. Rousseeuw, «Clustering by means of medoids. North-Holland,» 1987.
- [7] J. C. Bezdek, «Cluster Validity. In Pattern Recognition with Fuzzy Objective Function Algorithms (pp. 95-154). Springer US.,» 1981.
- [8] D. Arthur ve S. Vassilvitskii, «k-means++: The advantages of careful seeding. In Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms (pp. 1027-1035). Society for Industrial and Applied Mathematics.,» 2007.
- [9] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman ve A. Y. Wu, «An efficient k-means clustering algorithm: Analysis and implementation. IEEE transactions on pattern analysis and machine intelligence, 24(7), 881-892.,» 2002.
- [10] G. Frahling ve C. ve Sohler, «A fast k-means implementation using coresets. International Journal of Computational Geometry & Applications, 18(06), 605-625.,» 2008.

- [11] C. Elkan, «Using the triangle inequality to accelerate k-means. In Proceedings of the 20th International Conference on Machine Learning (ICML-03) (pp. 147-153).», 2003.
- [12] J. A. Hartigan ve M. A. Wong, «Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics), 28(1), 100-108.,» 1979.
- [13] M. Ester, H. P. Kriegel, J. Sander ve X. Xu, «A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd (Vol. 96, No. 34, pp. 226-231).», 1996.
- [14] D. Arlia ve M. Coppola, «Experiments in parallel clustering with DBSCAN. In European Conference on Parallel Processing (pp. 326-331). Springer, Berlin, Heidelberg.,» 2001.
- [15] M. Ankerst, M. M. Breunig, H. P. Kriegel ve J. Sander, «OPTICS: ordering points to identify the clustering structure. In ACM Sigmod record (Vol. 28, No. 2, pp. 49-60). ACM.,» 1999.
- [16] M. T. Elbatta ve W. M. Ashour, «A dynamic method for discovering density varied clusters. International Journal of Signal Processing, Image Processing and Pattern Recognition, 6(1), 123-134.,» 2013.
- [17] N. Shental, A. Bar-Hillel, T. Hertz ve D. Weinshall, «Computing Gaussian mixture models with EM using equivalence constraints. In Advances in neural information processing systems (pp. 465-472).», 2004.
- [18] A. P. Dempster, N. M. Laird ve D. B. Rubin, «Maximum likelihood from incomplete data via the EM algorithm. Journal of the royal statistical society. Series B (methodological), 1-38.,» 1977.
- [19] L. Hagen ve A. B. Kahng, «New spectral methods for ratio cut partitioning and clustering. IEEE transactions on computer-aided design of integrated circuits and systems, 11(9), 1074-1085.,» 1992.
- [20] J. Shi ve J. Malik, «Normalized cuts and image segmentation.,» *IEEE Transactions on pattern analysis and machine intelligence*, no. 22(8), pp. 888-905., 2000.
- [21] A. Y. Ng., M. I. Jordan ve Y. Weiss., «On spectral clustering: Analysis and an algorithm. In Advances in neural information processing systems (pp. 849-856).», 2002.

- [22] K. H. Anders ve M. Sester, «Parameter-free cluster detection in spatial databases and its application to typification.,» *International Archives of Photogrammetry and Remote Sensing*, no. 33(B4/1; PART 4), pp. 75-83, 2000.
- [23] E. Güngör ve A. Özmen, «Distance and density based clustering algorithm using Gaussian kernel.,» *Expert Systems with Applications*, no. 69 , pp. 10-20, 2017.
- [24] N. Ngonga ve Axel-Cyrille, «Parameter-free clustering of protein-protein interaction graphs.,» *Proceedings of the Fourth International Workshop on Machine Learning in Systems Biology (MLSB).*, 2010.
- [25] J. He, H. Tong, S. Papadimitriou, T. Eliassi-Rad, C. Faloutsos ve J. Carbonell, «Pack: Scalable parameter-free clustering on k-partite graphs.,» *In SDM Workshop on Link Analysis, Counterterrorism and Security (Vol. 110)*, 2009.
- [26] C. Böhm, S. Goebel, A. Oswald, C. Plant, M. Plavinski ve B. Wackersreuther, «Integrative parameter-free clustering of data with mixed type attributes.,» *In Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer*, pp. 38-47, 2010.
- [27] C. Böhm, C. Faloutsos, J. Y. Pan ve C. Plant, «RIC: Parameter-free noise-robust clustering.,» *ACM Transactions on Knowledge Discovery from Data (TKDD)*, pp. 1(3), 10, 2007.
- [28] W. Stuetzle ve R. Nugent, «A generalized single linkage method for estimating the cluster tree of a density.,» *Journal of Computational and Graphical Statistics*, no. 19(2), pp. 397-418, 2010.
- [29] W. Stuetzle, «Estimating the cluster tree of a density by analyzing the minimal spanning tree of a sample.,» *Journal of classification*, no. 20 (1), pp. 025-047., 2003.
- [30] A. Hinneburg ve D. A. Keim, «An efficient approach to clustering in large multimedia databases with noise.,» *KDD*, no. 98, pp. 58-65, 1998.
- [31] N. Shental, A. Bar-Hillel, T. Hertz ve D. Weinshall, «Computing Gaussian mixture models with EM using equivalence constraints.,» *In Advances in neural information processing systems*, pp. 465-472, 2004.

- [32] A. Smiti ve Z. Elouedi, «DBSCAN-GM: An improved clustering method based on Gaussian Means and DBSCAN techniques.,» *2012 IEEE 16th International Conference on Intelligent Engineering Systems (INES)*, pp. 573-578, 2012.
- [33] J. P. Baudry, A. E. Raftery, G. Celeux, K. Lo ve R. Gottardo, «Combining mixture components for clustering.,» *Journal of computational and graphical statistics*, no. 19(2), pp. 332-353 , 2010.
- [34] E. Achtert, H. P. Kriegel, E. Schubert ve A. Zimek, «Interactive data mining with 3D-parallel-coordinate-trees.,» *In Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data ACM*, pp. 1009-1012), 2013.
- [35] Y. Cheng, «Mean shift, mode seeking, and clustering.,» *IEEE transactions on pattern analysis and machine intelligence* , no. 17.8, pp. 790-799, 1995.
- [36] D. Comaniciu ve P. Meer, «Mean shift: A robust approach toward feature space analysis.,» *IEEE Transactions on pattern analysis and machine intelligence*, no. 24(5), pp. 603-619, 2002.
- [37] K. Fukunaga ve L. Hostetler, «The estimation of the gradient of a density function, with applications in pattern recognition.,» *IEEE Transactions on information theory*, no. 21(1), pp. 32-40, 1975.
- [38] W. Tao, H. Jin ve Y. Zhang., «Color image segmentation based on mean shift and normalized cuts.,» *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* , no. 37.5 , pp. 1382-1389, 2007.
- [39] C. Yang, R. Duraiswami, N. A. Gumerov ve L. Davis, «Improved fast gauss transform and efficient kernel density estimation.,» *ICCV* , cilt 1, p. 464, 2003.
- [40] D. Comaniciu, V. Ramesh ve P. Meer, «Real-time tracking of non-rigid objects using mean shift.,» *IEEE Conference on In Computer Vision and Pattern Recognition 2000. Proceedings.* , cilt 2, pp. 142-149, 2000.
- [41] H. Zhou, Y. Yuan ve C. Shi, «Object tracking using SIFT features and mean shift.,» *Computer vision and image understanding.,* no. 113(3), pp. 345-352, 2009.
- [42] A. R. Weeks ve G. E. Hague, «Color segmentation in the HSI color space using the K-means algorithm.,» *In Proceedings of SPIE* , no. 3026, pp. 143-154, 1997.

- [43] T. W. Chen, Y. L. Chen ve S. Y. Chien, «Fast image segmentation based on K-Means clustering with histograms in HSV color space.,» *In Multimedia Signal Processing, IEEE 10th Workshop*, pp. 322-325, 2008 .
- [44] L. Lucchese ve S. K. Mitra, «Unsupervised segmentation of color images based on k-means clustering in the chromaticity plane.,» *In Content-Based Access of Image and Video Libraries (CBAIVL'99) Proceedings. IEEE Workshop on. IEEE.*, pp. 74-78, 1999.
- [45] J. C. Dunn, «A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters.,» *Journal of Cybernetics*, no. 3(3), pp. 32-57, 1973.
- [46] C. Bezdek James, «Pattern Recognition with Fuzzy Function Algorithms.,» ISBN 0-306-40671-3, 1981.
- [47] S. L. X. Fan, Z. Man ve R. Samur, «Edge based region growing: a new image segmentation method.,» *In Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, pp. 302-305, 2004.
- [48] A. Tremeau ve N. Borel, «A region growing and merging algorithm to color segmentation.,» *Pattern recognition*, no. 30(7), pp. 1191-1203, 1997.
- [49] A. Melouah, «Comparison of automatic seed generation methods for breast tumor detection using region growing technique.,» *In IFIP International Conference on Computer Science and its Applications. Springer, Cham.*, pp. 119-128, 2015.
- [50] F. Y. Shih ve S. Cheng, «Automatic seeded region growing for color image segmentation.,» *Image and vision computing*, no. 23(10), pp. 877-886, 2005.
- [51] D. Martin, C. Fowlkes, D. Tal ve J. Malik, «A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics.,» *In Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference*, 2001.
- [52] C. D. Manning, P. Raghavan ve H. Schütze, «Evaluation of clustering.,» %1 içinde *Introduction to Information Retrieval*, ISBN: 0521865719, Cambridge University Press, 2008, 2009, p. 482.
- [53] R. Hartley ve A. Zisserman, *Multiple view geometry in computer vision.*, Cambridge university press., 2003, pp. 239-261.

- [54] G. Bradski ve A. Kaehler, Learning OpenCV: Computer vision with the OpenCV library., O'Reilly Media, Inc, 2008.
- [55] A. Geiger, P. Lenz, C. Stiller ve R. Urtasun, «Vision meets robotics: The KITTI dataset.,» *The International Journal of Robotics Research*, no. 32(11), pp. 1231-1237, 2013.
- [56] R. A. Hamzah ve H. Ibrahim, «Literature survey on stereo vision disparity map algorithms.,» *Journal of Sensors*, 2016., 2015.
- [57] B. J. Tippetts, D. J. Lee, J. K. Archibald ve K. D. Lillywhite, «Dense disparity real-time stereo vision algorithm for resource-limited systems.,» *IEEE Transactions on Circuits and Systems for Video Technology*, no. 21(10), pp. 1547-1555., 2011.
- [58] A. Fusiello, U. Castellani ve V. Murino, «Relaxing symmetric multiple windows stereo using markov random fields.,» *EMMCVPR*, pp. 91-104, 2001.
- [59] S. I. Satoh, «Simple low-dimensional features approximating NCC-based image matching.,» *Pattern Recognition Letters*, no. 32(14), pp. 1902-1911, 2011.
- [60] K. Konolige, «Small vision systems: Hardware and implementation.,» *Robotics research, Springer, London.*, pp. 203-212, 1998.
- [61] H. Hirschmuller, «Accurate and efficient stereo processing by semi-global matching and mutual information.,» *Computer Vision and Pattern Recognition CVPR 2005. IEEE Computer Society Conference*, pp. 807-814, 2005.
- [62] J. Huang, B. Kong, B. Li ve F. Zheng, «A new method of unstructured road detection based on HSV color space and road features.,» *In Information Acquisition, 2007. ICIA'07. IEEE International Conference on.*, pp. 596-601, 2007.
- [63] H. Kong, J. Y. Audibert ve J. Ponce, «General road detection from a single image.,» *IEEE Transactions on Image Processing*, no. 19(8), pp. 2211-2220, 2010.
- [64] J. M. Á. Alvarez ve A. M. Lopez, «Road detectionbased on illuminantinvariance.,» *IEEE Transactions on IntelligentTransportationSystems*, no. 12(1), pp. 184-193, 2011.

- [65] J. M. Alvarez ve A. Lopez, «Novel index for objective evaluation of road detection algorithms.,» *In Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference.*, pp. 815-820, 2008.
- [66] J. Fritsch, T. Kuhn ve A. Geiger, «A new performance measure and evaluation benchmark for road detection algorithms.,» *In Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference.*, pp. 1693-1700, 2013.
- [67] N. Soquet, D. Aubert ve N. Hautiere, «Road segmentation supervised by an extended v-disparity algorithm for autonomous navigation.,» *In Intelligent Vehicles Symposium, 2007 IEEE*, pp. 160-165, 2007.
- [68] R. Labayrade, D. Aubert ve J. P. Tarel, «Real time obstacle detection in stereovision on non flat road geometry through" v-disparity" representation.,» *In Intelligent Vehicle Symposium, IEEE 2002.*, cilt 2, pp. 646-651, 2002.
- [69] N. Fakhfakh, D. Gruyer ve D. Aubert, «Weighted V-disparity approach for obstacles localization in highway environments.,» *In Intelligent Vehicles Symposium (IV), 2013 IEEE*, pp. 1271-1278, 2013.
- [70] D. Yiruo, W. Wenjia ve K. Yukihiro, «Complex ground plane detection based on v-disparity map in off-road environment.,» *In Intelligent Vehicles Symposium (IV), 2013 IEEE* , pp. 1137-1142, 2013.
- [71] J. Zhao, J. Katupitiya ve J. Ward, «Global correlation based ground plane estimation using v-disparity image.,» *In Robotics and Automation, 2007 IEEE International Conference*, pp. 529-534, 2007.
- [72] I. Kostavelis, L. Nalpantidis ve A. Gasteratos, «Supervised traversability learning for robot navigation.,» *Towards Autonomous Robotic Systems*, pp. 289-298, 2011.
- [73] A. Iloie, I. Giosan ve S. Nedevschi, «UV disparity based obstacle detection and pedestrian classification in urban traffic scenarios.,» *In Intelligent Computer Communication and Processing (ICCP), 2014 IEEE International Conference*, pp. 119-125, 2014.
- [74] J. Wang, Z. Hu, H. Lu ve K. Uchimura, «Motion detection in driving environment using uv-disparity.,» *Computer Vision-ACCV 2006*, pp. 307-316., 2006.

- [75] B. Wang, S. A. R. Florez ve V. Frémont, «Multiple obstacle detection and tracking using stereo vision: application and analysis.,» *In Control Automation Robotics & Vision (ICARCV), 2014 13th International Conference*, pp. 1074-1079, 2014.
- [76] M. A. Mahammed, A. I. Melhum ve F. A. Kochery, «Object distance measurement by stereo vision.,» *International Journal of Science and Applied Information Technology (IJSAIT)*, , no. 2(2), pp. 05-08, 2013.
- [77] FLIR, «Machine Vision - How is depth determined from a disparity image.,» FLIR, 14 5 2015. [Çevrimiçi]. Available: <https://www.ptgrey.com/KB/10102>. [Erişildi: 08 11 2017].
- [78] J.-H. Chen, «Computer Vision, CSE 455, Lecture 16 slides.,» 2009. [Çevrimiçi]. Available: <https://courses.cs.washington.edu/courses/cse455/09wi/Lects/lect16.pdf>. [Erişildi: 06 11 2017].
- [79] H. Rosas, W. Vargas, A. Cerón, D. Domínguez ve A. Cárdenas, «A Mathematical Expression for Stereoscopic Depth Perception.,» *Photogrammetric Engineering & Remote Sensing*, no. 76(3), pp. 301-306, 2010.
- [80] S. Palmisano, B. Gillam, D. G. Govan, R. S. Allison ve J. M. Harris, «Stereoscopic perception of real depths at large distances.,» *Journal of Vision*, no. 10(6), pp. 19-19., 2010.
- [81] Y. Kuno, T. Watanabe, Y. Shimosakoda ve S. Nakagawa, «Automated detection of human for visual surveillance system.,» *Pattern Recognition, 1996., Proceedings of the 13th International Conference on IEEE* , no. 3, pp. 865-869, 1996.
- [82] R. D. De Leon ve L. E. Sucar, «Human silhouette recognition with Fourier descriptors.,» *Pattern Recognition, 2000. Proceedings. 15th International Conference on IEEE.*, no. 3, pp. 709-712, 2000.
- [83] Z. Lin, L. S. Davis, D. Doermann ve D. DeMenthon, «Hierarchical part-template matching for human detection and segmentation.,» *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on IEEE.*, pp. 1-8, 2007.

- [84] Z. Lin ve L. S. Davis, «A pose-invariant descriptor for human detection and segmentation.,» *European Conference on Computer Vision. Springer, Berlin, Heidelberg.*, pp. 423-436, 2008.
- [85] N. Dalal ve B. Triggs, «Histograms of oriented gradients for human detection.,» *Computer Vision and Pattern Recognition, 2005. CVPR 2005.*, pp. 886-893, 2005.
- [86] Y. T. Chen ve C. S. Chen, «Fast human detection using a novel boosted cascading structure with meta stages.,» *IEEE Transactions on image processing*, , no. 17(8), pp. 1452-1464, 2008.
- [87] N. Dalal, «INRIA Person Dataset.,» [Çevrimiçi]. Available: <http://pascal.inrialpes.fr/data/human/>.
- [88] C. G. Keller, M. Enzweiler ve D. M. Gavrilă, «A new benchmark for stereo-based pedestrian detection.,» *Intelligent Vehicles Symposium (IV), 2011 IEEE* , pp. 691-696, 2011.
- [89] A. Toya, Z. Hu, T. Yoshida, K. Uchimura, H. Kubota ve M. Ono, « Pedestrian recognition using stereo vision and histogram of oriented gradients.,» *In Vehicular Electronics and Safety, 2008. ICVES 2008. IEEE International Conference* , pp. 57-62, 2008.
- [90] C. G. Keller ve D. M. Gavrilă, «Will the pedestrian cross? A study on pedestrian path prediction.,» *IEEE Transactions on Intelligent Transportation Systems*, no. 15(2), pp. 494-506, 2014.
- [91] D. F. Llorca, M. A. Sotelo, A. M. Hellín, A. Orellana, M. Gavilán, I. G. Daza ve A. G. Lorente, «Stereo regions-of-interest selection for pedestrian protection: A survey.,» *Transportation research part C: emerging technologies*, no. 25, pp. 226-23, 2012.
- [92] M. J. Atallah ve M. (. Blanton, *Algorithms and theory of computation handbook, volume 2: special topics and techniques.*, CRC press., 2009.
- [93] T. Lindeberg, «Scale-space theory: A basic tool for analyzing structures at different scales.,» *Journal of applied statistics*, no. 21(1-2), pp. 225-270, 1994.
- [94] T. Lindeberg, *Scale-Space Theory in Computer Vision*, Springer US, 1994.
- [95] P. D. (. McNicholas, *Mixture model-based classification.*, CRC Press., 2016.

- [96] A. Singh, A. Yadav ve A. Rana, «K-means with Three different Distance Metrics.,» *International Journal of Computer Applications*, no. 67(10), 2013.
- [97] Y. Ma ve G. (. Guo, *Support vector machines applications.*, Springer Science & Business Media., 2014.
- [98] Z. B. Alfassi, Z. Boger ve Y. Ronen, *Statistical treatment of analytical data.*, CRC Press., 2005.
- [99] V. F. Leavers, *Shape detection in computer vision using the Hough transform*, London: Springer, 1992.
- [100] «Pasi Fr. ve arkadaşları, Clustering datasets,» 2015. [Çevrimiçi]. Available: <http://cs.uef.fi/sipu/datasets/>. [Erişildi: 06 11 2017].
- [101] C. Papageorgiou ve T. Poggio, «A trainable system for object detection.,» *IJCV*, no. 38(1), pp. 15-33, 2000.
- [102] M. S. ve D. M. Gavrilu, «An experimental study on pedestrian classification,» *IEEE T.PAMI*, no. 28(11), pp. 1863-1868, 2006.
- [103] C. Hennig, M. Meila, F. Murtagh ve R. (. Rocci, «Clustering strategy and method selection,» %1 içinde *Handbook of cluster analysis.*, CRC Press, 2015, p. 703.
- [104] C. Hennig, «What are the true clusters?.,» *Pattern Recognition Letters* , no. 64, pp. 53-62, 2015.
- [105] «MATLAB 8.0, Natick, Massachusetts, United States.,» The MathWorks, Inc., [Çevrimiçi]. Available: <https://www.mathworks.com>.
- [106] E. Güngör ve A. Özmen, «GDD Clustering - Distance and density based clustering,» 13 10 2017. [Çevrimiçi]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/64687-gdd-clustering-distance-and-density-based-clustering>. [Erişildi: 22 11 2017].
- [107] J. C. Russ, *Image Processing Handbook*, Fourth Edition, Boca Raton, FL, USA,: CRC Press Inc., 2002.
- [108] A. R. Smith, «Color gamut transform pairs.,» *ACM Siggraph Computer Graphics.*, no. 12(3), pp. 12-19, 1978.

- [109] A. Asvadi, M. Karami-Mollaie, Y. Baleghi ve H. Seyyedi-Andi, «Improved object tracking using radial basis function neural networks,» *Machine Vision and Image Processing (MVIP), 2011 7th Iranian, IEEE*, pp. 1-5, 2011.
- [110] A. Asvadi, «Mathworks,» Mathworks, 27 08 2015. [Çevrimiçi]. Available: <https://www.mathworks.com/matlabcentral/fileexchange/52698-k-means--mean-shift-and-normalized-cut-segmentation>. [Erişildi: 22 11 2017].
- [111] «google görselleri,» [Çevrimiçi]. Available: <https://ak2.picdn.net/shutterstock/videos/31172992/thumb/1.jpg>. [Erişildi: 23 11 2017].
- [112] Bidgee, «Altocumulus photographed in Wagga Wagga, New South Wales, Australia.,» 11 03 2009. [Çevrimiçi]. Available: https://en.wikipedia.org/wiki/Altocumulus_cloud. [Erişildi: 23 11 2017].
- [113] S. Weisberg, Applied linear regression (Vol. 528), John Wiley & Sons., 2005.
- [114] J. L. Stensby, «Random Signals and Noise Course Notes,» University of Alabama in Huntsville, 2010. [Çevrimiçi]. Available: <http://www.ece.uah.edu/courses/ee385/603ch12.pdf>. [Erişildi: 06 08 2018].
- [115] A. Jaison, «<http://www.cut.ac.zw>,» Chinhoyi University of Technology | CUT · School of Natural Sciences and Mathematics, [Çevrimiçi]. Available: <http://www.cut.ac.zw/espace/ajaison/1395303523.pdf>. [Erişildi: 06 08 2018].
- [116] H. S. Lim ve v. H. Park, «"A dense disparity estimation method using color segmentation and energy minimization.",» *Image Processing, 2006 IEEE International Conference.*, pp. 1033-1036, 2006.
- [117] C. C. Gordon, T. Churchill, C. E. Clauser, B. Bradtmiller, J. T. McConville, I. Tebbetts ve R. A. Walker, «Anthropometric survey of US Army personnel: Summary statistics, interim report for 1988.,» *ANTHROPOLOGY RESEARCH PROJECT INC YELLOW SPRINGS OH.*, 1989.
- [118] D. Lisin, «How to use calibration parameters from KITTI,» Mathworks, 11 03 2016. [Çevrimiçi]. Available: <https://www.mathworks.com/matlabcentral/answers/270513-how-to-use-calibration-parameters-from-kitti>. [Erişildi: 08 11 2017].
- [119] Mathworks, «What Is Camera Calibration,» Mathworks, [Çevrimiçi]. Available: <https://www.mathworks.com/help/vision/ug/camera-calibration.html>. [Erişildi: 08 11 2017].

- [120] Mathworks, «triangulate,» Mathworks, [Çevrimiçi]. Available: <https://www.mathworks.com/help/vision/ref/triangulate.html>. [Erişildi: 08 12 2017].
- [121] C. F. Olson, L. H. Matthies, M. Schoppers ve M. W. Maimone, «Robust stereo ego-motion for long distance navigation,» *CVPR IEEE*, p. 2453, 2000.
- [122] J. Shi ve J. Malik, «Normalized cuts and image segmentation.,» *IEEE Transactions on pattern analysis and machine intelligence*, no. 22(8), pp. 888-905, 2000.

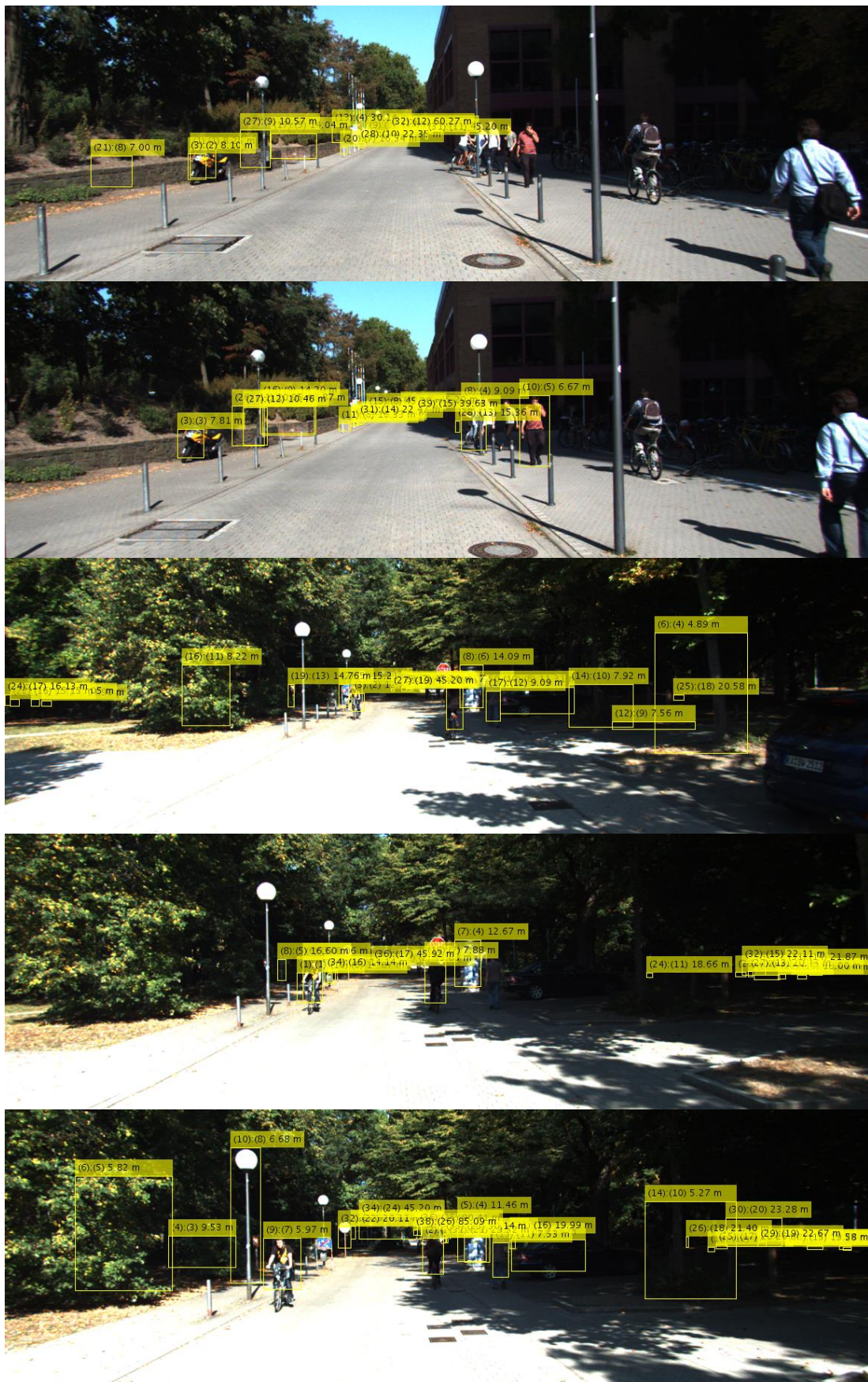


EKLER

EK 1: Sistem test görüntü sonuçları

Tez içerisinde belirtilen örnekler dışında sistemin performansı için yapılan test sonuçlarının yaya tespit ve uzaklık çıktıları aşağıdaki figürlerde gösterilmiştir.

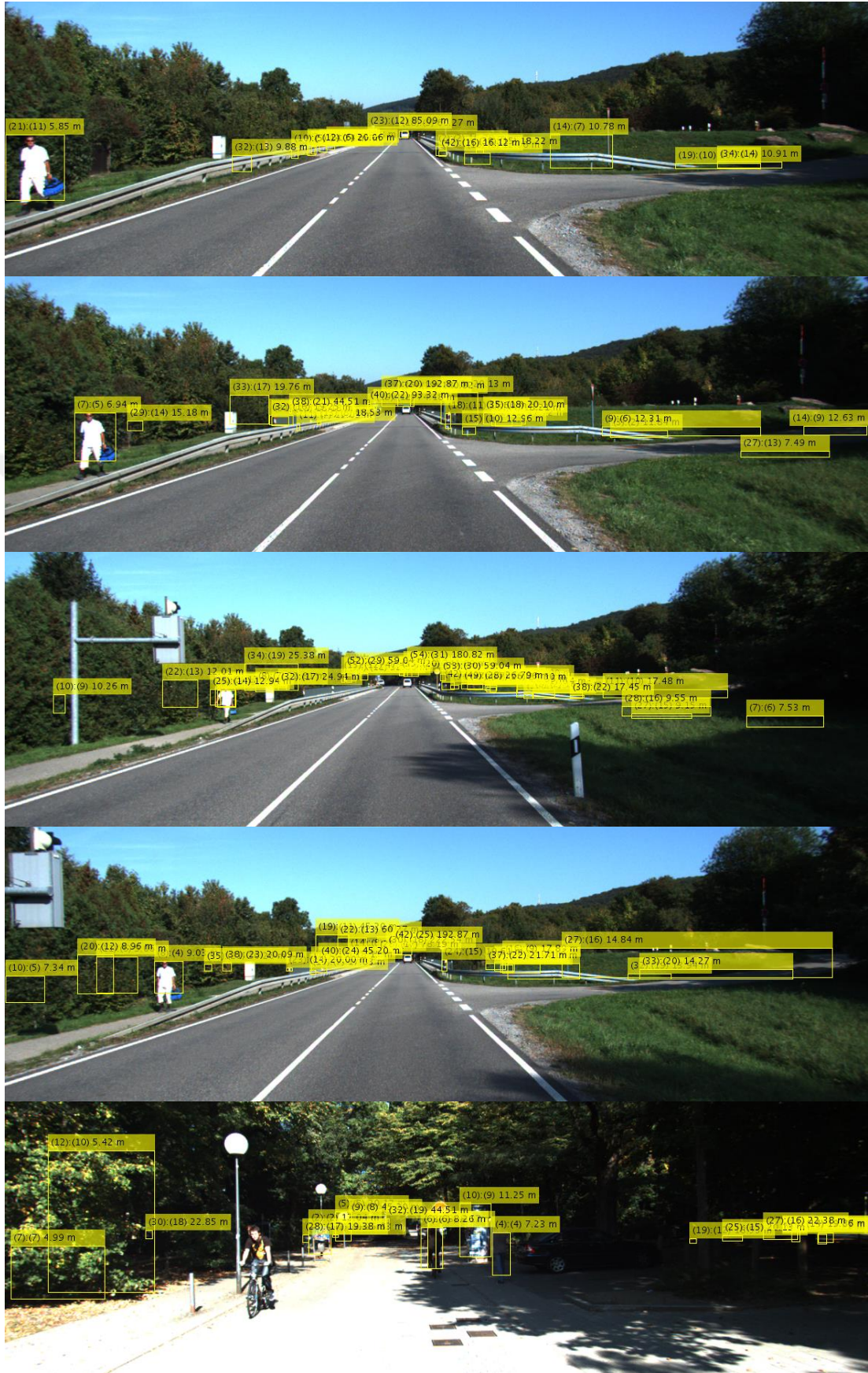


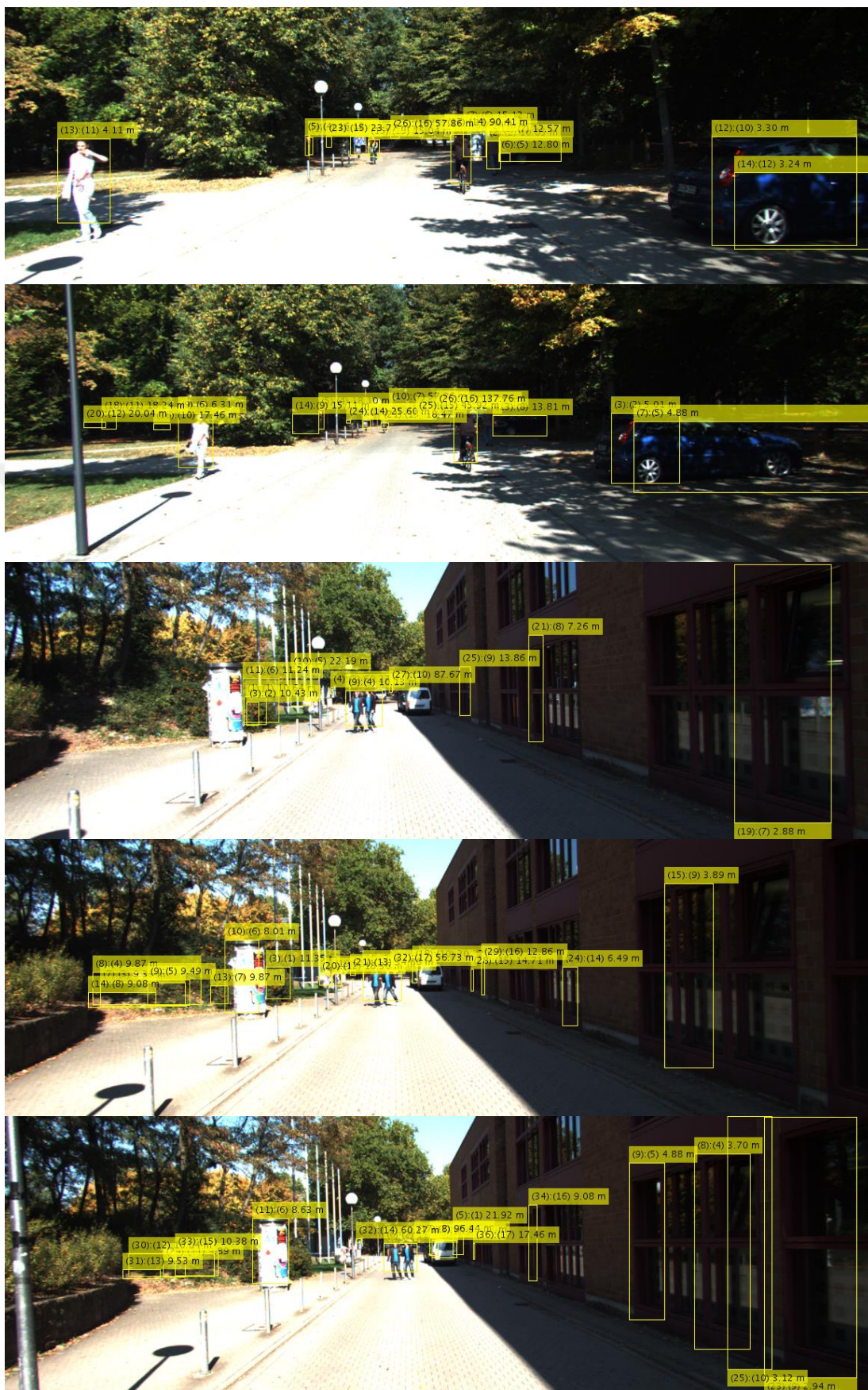


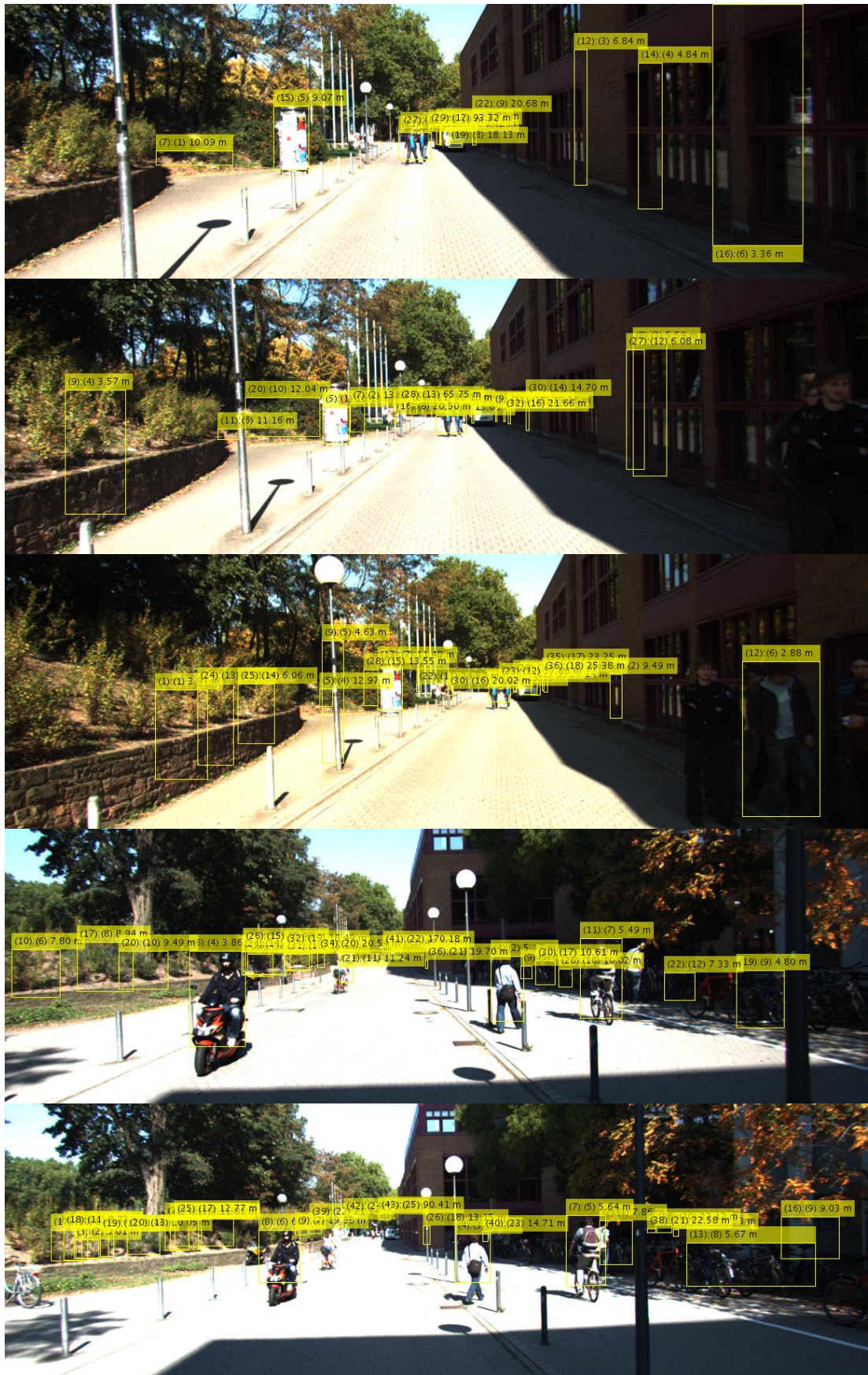


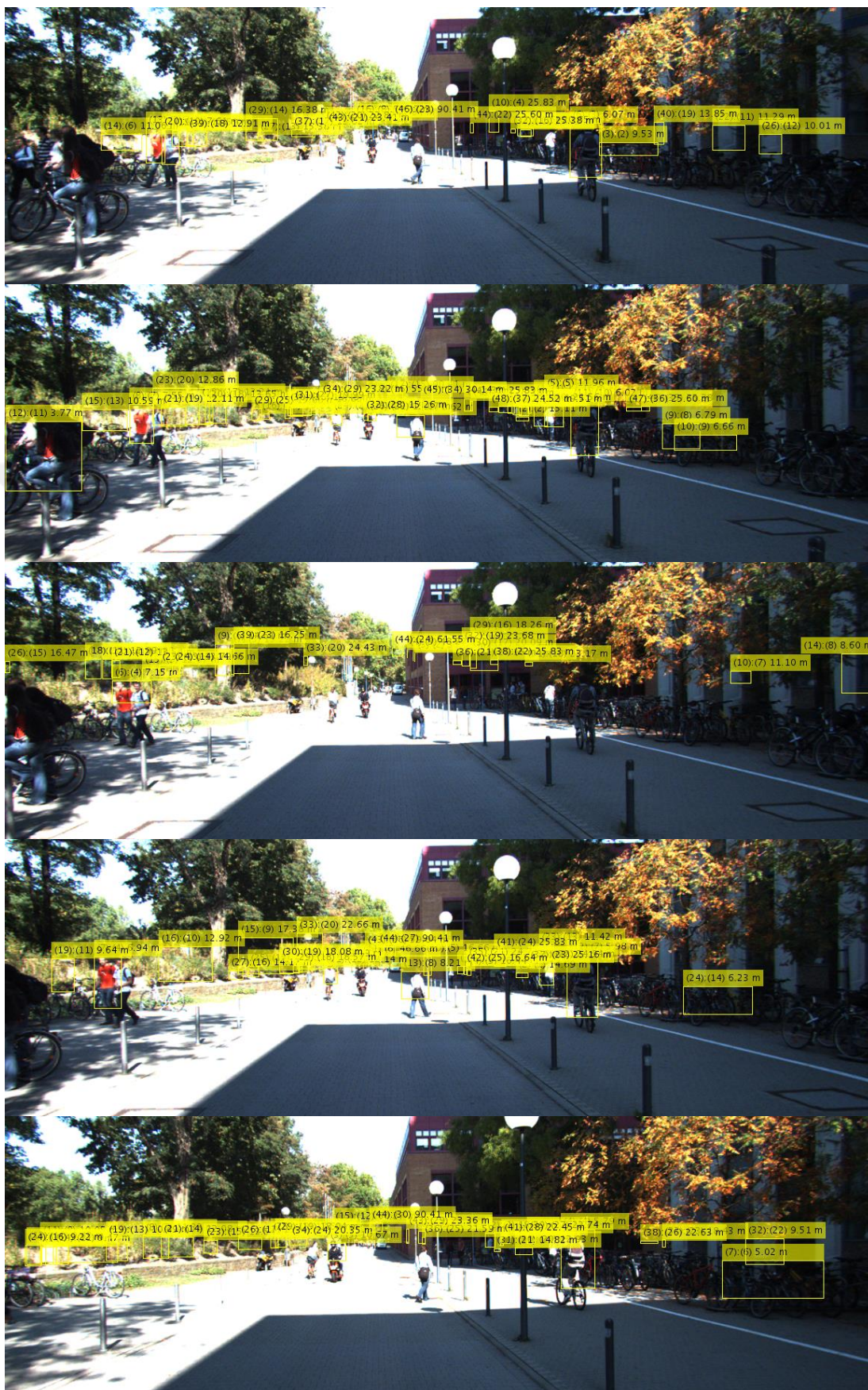




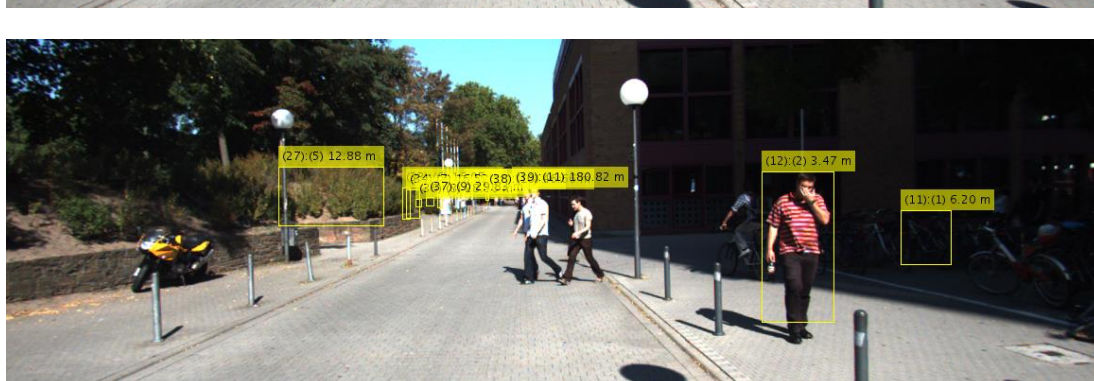










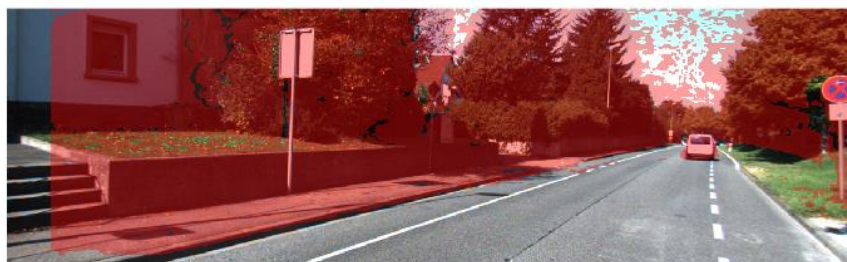


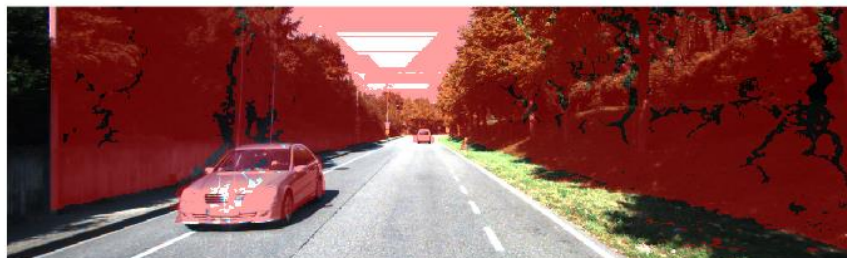
EK 2: Engel Tespit ek test ve sonuçlar

Belirtilen örnekler dışında da geliştirilen algoritmanın başarısı, her görüntü örneğinde üstteki görüntü tez kapsamında geliştirilen sistemin engel sonucu, alttaki görüntü ise Hough dönüşümü ile elde edilen engel sonuçları olmak üzere aşağıdaki görüntü çıktılarında karşılaştırmalı olarak verilmiştir.











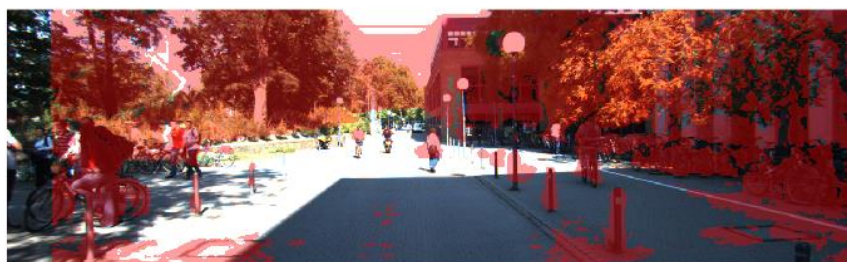
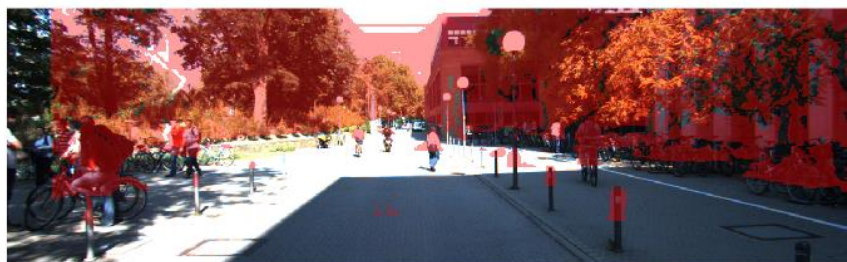








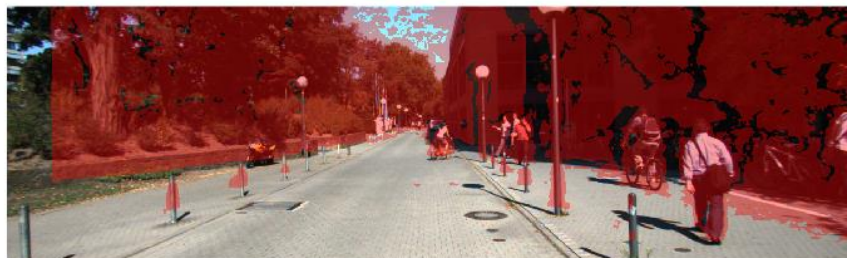














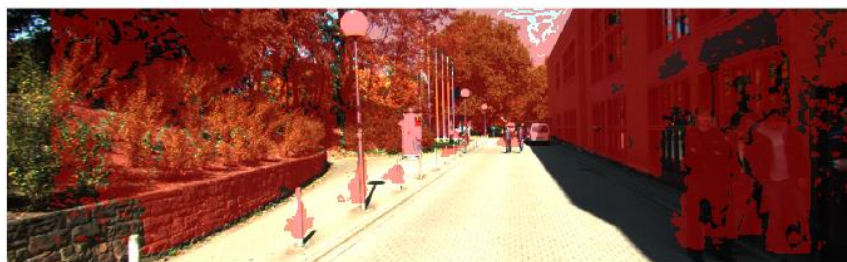






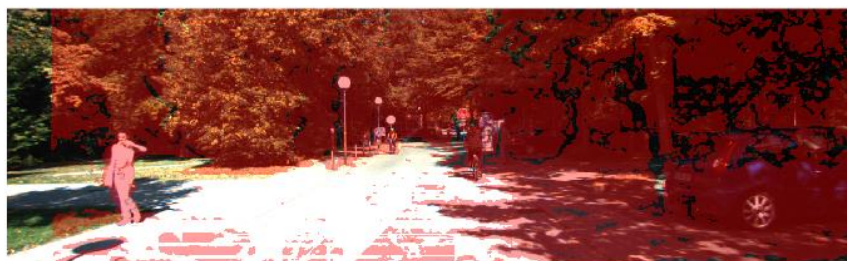


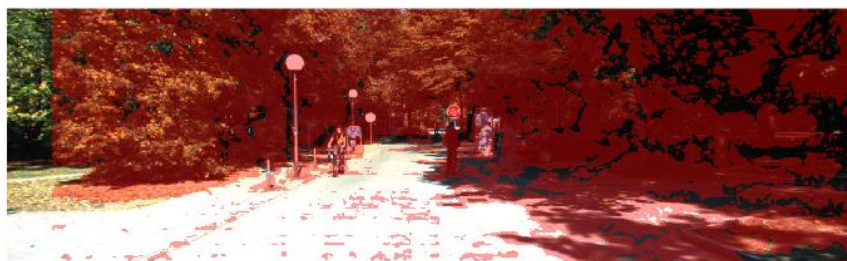














ÖZGEÇMİŞ

Emre Güngör, 02.10.1984 yılında Eskişehir’de doğdu. Kıbrıs İlkokulu, Osmangazi Orta Okulu, Eskişehir Kılıçoğlu Anadolu Lisesinden sonra Eskişehir Osmangazi Üniversitesinde Elektrik Elektronik Mühendisliği alanında lisans eğitimini tamamladı. 2004 yılında başladığı lisans eğitimi sonrasında 2012 yılında Bilecik Şeyh Edebali Üniversitesinde Bilgisayar Mühendisliği alanında yüksek lisansını birincilikle bitirdi. Lisans eğitimi sonrası yüksek öğrenim hayatında 2010-2014 yılları arası Bilecik Şeyh Edebali Üniversitesinde Bilgisayar Mühendisliği alanında Araştırma Görevliliği yapmıştır.