T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KAVŞAK SENARYOLARINDA NESNE TAKİBİNİN ETKİLEŞEN ÇOKLU MODELLER KULLANILARAK İYİLEŞTİRİLMESİ

Çağrı HAKKOYMAZ

YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Programı

Danışman
Dr. Öğr. Üyesi Erkan USLU

Şubat, 2026

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KAVŞAK SENARYOLARINDA NESNE TAKİBİNİN ETKİLEŞEN ÇOKLU MODELLER KULLANILARAK İYİLEŞTİRİLMESİ

Çağrı HAKKOYMAZ tarafından hazırlanan tez çalışması 21.02.2026 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Dr. Öğr. Üyesi Erkan USLU
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Dr. Öğr. Üyesi Erkan USLU, Danışman
Yıldız Teknik Üniversitesi _____

Doç. Dr. Hamza Osman İLHAN, Üye
Yıldız Teknik Üniversitesi _____

Dr. Öğr. Üyesi Kamil ERDAYANDI, Üye
Marmara Üniversitesi _____

Danışmanım Dr. Öğr. Üyesi Erkan USLU sorumluluğunda tarafımca hazırlanan KAVŞAK SENARYOLARINDA NESNE TAKİBİNİN ETKİLEŞEN ÇOKLU MODELLER KULLANILARAK İYİLEŞTİRİLMESİ başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Çağrı HAKKOYMAZ

İmza

Her zaman yanımda olan aileme
ve arkadaşlarıma

TEŞEKKÜR

Kavşak Senaryolarında Nesne Takibinin Etkileşen Çoklu Modeller Kullanılarak İyileştirilmesi çalışmamın gerçekleştirilmesi konusunda sağladığı katkılar için Dr. Öğr. Üyesi Erkan USLU'ya ve robotik alanında kendimi geliştirmemi sağlayan takım arkadaşlarıma ve tez yazım sürecinde yanımda olan aileme, nişanlıma ve arkadaşlarıma teşekkür ederim.

Çağrı HAKKOYMAZ

İÇİNDEKİLER

SİMGE LİSTESİ viii

KISALTMA LİSTESİ ix

ŞEKİL LİSTESİ xi

TABLO LİSTESİ xiii

ÖZET xiv

ABSTRACT xvi

1 GİRİŞ 1

1.1 Tezin Amacı . . . 1

1.1.1 Katkılar . . . 3

1.2 Literatür Özeti . . . 3

1.2.1 Algılama (Perception) . . . 6

1.2.2 Lokalizasyon (Localization) . . . 13

1.2.3 Planlama . . . 15

1.2.4 Kontrol . . . 17

1.2.5 Yörünge Tahmini (Trajectory Prediction) . . . 18

1.2.6 Tezin Genel Organizasyonu . . . 19

2 NESNE TAKİBİ ALGORİTMALARI 21

2.1 Takip Yönetimi . . . 23

2.1.1 M/N Yöntemi . . . 23

2.1.2 İz Skoru (Track Score) . . . 24

2.2 Veri Eşleştirme . . . 24

2.2.1 MCTrack'te Veri Eşleştirme: Ro GDIoU ve İki Aşamalı Eşleştirme . . . 25

2.2.2 Kapılama (Gating) . . . 26

2.2.3 Hungarian Algoritması . . . 28

2.2.4 GNN - Global Nearest Neighbor . . . 29

2.2.5 JPDA - Joint Probabilistic Data Association . . . 29

2.2.6 MHT - Multiple Hypothesis Tracking 30
2.2.7 Öğrenme Tabanlı (Makine Öğrenmesi) Veri İlişkilendirme Yaklaşımları 30
2.3 Kalman Filtresi 31
2.3.1 Kalman Filtresi (KF) 31
2.3.2 Extended Kalman Filtresi (EKF) 32
2.3.3 Unscented Kalman Filtresi (UKF) 32
2.3.4 Sabit Hız Hareket Modeli (CV) 36
2.3.5 Sabit İvme Hareket Modeli (CA) 36
2.3.6 Sabit Dönüş Hızı ve Sabit Hız Hareket Modeli (CTRV) . . . 36
2.3.7 Sabit Dönüş Hızı ve Sabit İvme Hareket Modeli (CTRA) . . 37
2.4 Etkileşen Çoklu Modeller (IMM) 37
2.4.1 Model Varsayımı ve Notasyon 38
2.4.2 IMM Algoritmasının Adımları (Denklemler ve Terimler) . . 39
2.4.3 IMM Algoritmasının Pseudocode Gösterimi 42
2.4.4 Sağlamlık İyileştirmeleri 43
2.5 Önerilen Nesne Takip Sistemi 43
2.5.1 MCTrack Çatısı ve Bu Tezdeki Kullanımı 44
2.5.2 Kullanılan Nesne Takip Yapısı 44
2.5.3 Nesne Tespiti (VirConv ve CasA) 45
2.5.4 Durum Vektörleri ve Ölçüm Modeli 48
2.5.5 Veri İlişkilendirme ve Kapılama 48
2.5.6 İz Yönetimi 48
2.5.7 IMM Konfigürasyonu ve Mod Olasılıkları 48

3 TEST ORTAMI, DEĞERLENDİRME METRİKLERİ VE SONUÇLAR 50
3.1 KITTI Test Veri Seti 50
3.1.1 Etiket Formatı ve Alanlar 52
3.2 Karşılaştırma Metrikleri: HOTA ve Alt Metrikler 53
3.2.1 Lokalizasyon (Localization) 54
3.2.2 Algılama (Detection) 55
3.2.3 Eşleştirme (Association) 56
3.2.4 HOTA Skorunun Hesaplanması 58
3.3 Sensör Bozulmasının Modellenmesi 58
3.3.1 Ölçüm Belirsizliği ve Ölçüm Düşürme Senaryoları 59
3.3.2 Başlatma (Initial Conditions) ve Parametre Ayarları 59
3.4 Test Kurgusu, Senaryo Seçimi ve Temsil Yeteneği 61
3.5 Sonuçların Tartışılması 69

3.5.1 Farklı Algılayıcılar İçin Referans Durum Sonuçları 69
3.5.2 Farklı Algılayıcılar İçin Ölçüm Düşürme Senaryosunun Etkisinin İncelenmesi 69
3.5.3 Farklı Algılayıcılar İçin Yüksek Gürültü Senaryosunun Etkisinin İncelenmesi 70
3.5.4 Kavşak Senaryolarında Etkinin Detaylı İncelenmesi 71
3.5.5 Ölçüm gürültüsü (R) arttıkça modelin baskın hale gelmesi . 75
3.5.6 Ölçüm düşürme senaryolarında model sonuçlarının incelenmesi 76
3.5.7 CA ve IMM'nin dönüş segmentlerindeki davranışı 77
3.5.8 Bir senaryonun daha küçük alt segmenlere ayrılıp incelenmesi 78
3.5.9 Çalışma Zamanı ve Hesaplama Karmaşıklığı 80

4 SONUÇ **81**

KAYNAKÇA **83**

TEZDEN ÜRETİLMİŞ YAYINLAR **91**

SİMGE LİSTESİ

S	Artış Kovaryansı
F	Durum Geçiş Matrisi
P	Hata Kovaryans Matrisi
ν	Kalman Artışı (Innovation)
K	Kalman Kazancı Matrisi
π_{ij}	Model Geçiş Olasılıkları
μ	Model Olasılıkları Vektörü
R	Ölçüm Gürültüsü Kovaryans Matrisi
H	Ölçüm Matrisi
Q	Süreç Gürültüsü Kovaryans Matrisi
$\hat{x}$	Tahmin Edilen Durum Vektörü
z	Ölçüm Vektörü

KISALTMA LİSTESİ

ADAS	Gelişmiş Sürücü Destek Sistemleri (Advanced Driver Assistance Systems)
ACC	Adaptif Hız Sabitleyici (Adaptive Cruise Control)
AssA	İlişkilendirme Doğruluğu (Association Accuracy)
AssPr	İlişkilendirme Kesinliği (Association Precision)
AssRe	İlişkilendirme Duyarlılığı (Association Recall)
CA	Sabit İvme (Constant Acceleration)
CTRA	Sabit Dönüş Hızı ve İvme (Constant Turn Rate and Acceleration)
CTRV	Sabit Dönüş Hızı ve Hız (Constant Turn Rate and Velocity)
CV	Sabit Hız (Constant Velocity)
DARPA	Savunma İleri Araştırma Projeleri Ajansı (Defense Advanced Research Projects Agency)
DIoU	Mesafe Tabanlı Kesişim / Birleşim Oranı (Distance Intersection over Union)
DWA	Dinamik Pencere Yaklaşımı (Dynamic Window Approach)
DetA	Algılama Doğruluğu (Detection Accuracy)
DetPr	Algılama Kesinliği (Detection Precision)
DetRe	Algılama Duyarlılığı (Detection Recall)
EKF	Genişletilmiş Kalman Filtresi (Extended Kalman Filter)
GIoU	Genelleştirilmiş Kesişim / Birleşim Oranı (Generalized Intersection over Union)
GNN	Global En Yakın Komşuluk (Global Nearest Neighbor)
GNSS	Küresel Navigasyon Uydu Sistemi (Global Navigation Satellite System)
GPS	Küresel Konumlama Sistemi (Global Positioning System)

GT	Gerçek Değer (Ground Truth)
HOTA	Yüksek Dereceli Takip Doğruluğu (Higher Order Tracking Accuracy)
IMM	Etkileşimli Çoklu Model (Interacting Multiple Model)
IMU	Ataletsel Ölçüm Birimi (Inertial Measurement Unit)
IoU	Kesişim / Birleşim Oranı (Intersection over Union)
JPDA	Ortak Olasılıksal Veri İlişkilendirme (Joint Probabilistic Data Association)
KF	Kalman Filtresi (Kalman Filter)
KITTI	Karlsruhe Teknoloji Enstitüsü ve Toyota Teknoloji Enstitüsü Veri Seti (Karlsruhe Institute of Technology and Toyota Technological Institute Dataset)
LiDAR	Işık Tespiti ve Uzaklık Tayini (Light Detection and Ranging)
LRR	Uzun Menzilli Radar (Long Range Radar)
LocA	Lokalizasyon Doğruluğu (Localization Accuracy)
M/N	M/N Mantığı (M-of-N Logic)
MPC	Model Öngörülü Kontrol (Model Predictive Control)
ODD	Operasyonel Tasarım Alanı (Operational Design Domain)
RRT	Hızlı Keşif Yapan Rastgele Ağaç (Rapidly-exploring Random Tree)
SRR	Kısa Menzilli Radar (Short Range Radar)
UKF	Örneklemeli Kalman Filtresi (Unscented Kalman Filter)

ŞEKİL LİSTESİ

Şekil 1.1 Örnek otonom sürüş platformları. 4
Şekil 1.2 Monoküler ve stereo kurulumlarda derinlik çıktısına örnek. [19]. . 8
Şekil 1.3 Otomotiv radarlarında kısa menzil geniş açılı SRR (yeşil) ve uzun menzil dar açılı LRR (turuncu) kapsama bölgelerine örnek. [20]. . 8
Şekil 1.4 LiDAR sensör mimarilerine örnek: spinning ve solid-state. 9
Şekil 1.5 KITTI otonom sürüş veri setinden örnek: sahne çıktıları üzerinde 3B sınır kutuları ile 3B algılama/tespit örnek görselleştirmesi [7]. 11
Şekil 1.6 Kamera tabanlı algılamada kuş-bakışı görünüm (BEV) temsiline örnek [27]. 11
Şekil 1.7 Sabit hız (constant velocity) hareket modeli ve Gauss gürültüsü simülasyonu ile yörünge tahminine örnek. Elipsler, tahmin edilen konumların belirsizliğini (kovaryansını) göstermektedir [66] . . . 20
Şekil 2.1 MCTrack'in ilgili değerlendirme liderlik tablosundaki sıralamasının güncel durumu. 22
Şekil 2.2 Kutular arası benzerlik ölçütlerinde (IoU/GIoU/DIoU) örtüşme, çevreleyen alan ve merkez mesafesi kavramlarına örnek. 26
Şekil 2.3 Dikdörtgen ve Eliptik Kapıların (Gating) Gösterimi [81] 28
Şekil 2.4 GNN algoritmasında iki farklı nesnenin aynı ölçüme talip olma senaryosu [82] 29
Şekil 2.5 Uygulanan IMM algoritması diagramı 41
Şekil 2.6 Önerilen Nesne Takip Sisteminin Gösterimi 44
Şekil 3.1 KITTI veri toplama aracı. [7] 51
Şekil 3.2 Sensörlerin araç üzerindeki yerleşimi. [7] 51
Şekil 3.3 IoU (kesişim/birleşim) kavramının şematik gösterimi [90]. 55
Şekil 3.4 Eşleştirme (Association) için iz-kesişimi (track IoU) örneği: **Gri** (tahmin) ve **mavi** (gerçek) tespit/yörüngeler üzerinden, kırmızı kutu ile seçilen **TP of interest (c)** için; aynı iki iz boyunca eşleşen TP'ler **TPA(c)** (**yeşil** kesik çizgili bölge), tahmin izinde kalıp bu gerçekle eşleşmeyenler **FPA(c)** (**sarı** kesik çizgili bölge) ve gerçek izde kalıp bu tahminle eşleşmeyenler **FNA(c)** (**kahverengi** kesik çizgili bölge) olarak gösterilmiştir [90]. 57

Şekil 3.5 Genel senaryolardan örnek dedektör girdileri (2B yer düzleminde araç tespitleri). 62
Şekil 3.6 Kavşak Senaryosuna ait 1.kamera görüntüsü 64
Şekil 3.7 Kavşak senaryosuna ait 1.LiDAR nokta bulutu görüntüsü 64
Şekil 3.8 Kavşak Senaryosuna ait 2.kamera görüntüsü 65
Şekil 3.9 Kavşak senaryosuna ait 2.LiDAR nokta bulutu görüntüsü 65
Şekil 3.10 Kitti 12. Senaryo Girdi Ölçümlerin Gösterimi 66
Şekil 3.11 Kitti 12. Senaryo R=5 IMM - Takip Çıktı Sonuçları Gösterimi . . 67
Şekil 3.12 Kitti 15. Senaryo Girdi Ölçümlerin Gösterimi 67
Şekil 3.13 Kitti 15. Senaryo R=5 IMM - Takip Çıktı Sonuçları Gösterimi . . 68
Şekil 3.14 Kitti 15. Senaryo Nesne 5 model olasılıkları 73
Şekil 3.15 Kitti 15. Senaryo Nesne 5 Pozisyon değişimi 73
Şekil 3.16 Kitti 15. Senaryo Nesne 9 model olasılıkları 74
Şekil 3.17 Kitti 15. Senaryo Nesne 9 Pozisyon değişimi 74
Şekil 3.18 Senaryo 0012 v için CA/CTRA/IMM yöntemlerinde HOTA'nın ölçüm gürültüsüne (R) göre değişimi. 76
Şekil 3.19 Senaryo 0015 v için CA/CTRA/IMM yöntemlerinde HOTA'nın ölçüm gürültüsüne (R) göre değişimi. 77
Şekil 3.20 Senaryo 15'in dört alt segmente ayrılmasıyla elde edilen dedektör tespitleri (girdi) ve takip izleri (çıktı) görselleştirmeleri. 79

TABLO LİSTESİ

Tablo 3.1 Çoklu nesne takibi değerlendirme metrikleri 53
Tablo 3.2 Sensör bozunumu koşulları: ölçüm gürültüsü ($\mathbf{R}$) ve ölçüm düşürme modeli. 59
Tablo 3.3 VirConv Dedektörü (Nominal: %0 Kayıp, R=0.5) – Sadece Combined Sonuçlar 69
Tablo 3.4 CasA Dedektörü (Nominal: %0 Kayıp, R=0.5) – Sadece Combined Sonuçlar 69
Tablo 3.5 VirConv Dedektörü (%20 Kayıp) – Sadece Combined Sonuçlar . 69
Tablo 3.6 CasA Dedektörü (%20 Kayıp) – Sadece Combined Sonuçlar . . . 70
Tablo 3.7 VirConv Dedektörü (No Drop, R=50) – Sadece Combined Sonuçlar 70
Tablo 3.8 CasA Dedektörü (No Drop, R=50) – Sadece Combined Sonuçlar 70
Tablo 3.9 *val-car* 0012 sekansı için CA, CTRA ve IMM yöntemlerinin HOTA metrikleri (R değerine göre gruplandırılmıştır). Her R bloğu içinde en iyi sonuçlar kalın olarak gösterilmiştir. 71
Tablo 3.10 *val-car* 0012 sekansı için $R = 5$ koşulunda ölçüm düşürmeli senaryo sonuçları. Her düşürme oranında en iyi sonuçlar kalın olarak gösterilmiştir. 72
Tablo 3.11 *val-car* 0015 sekansı için ölçüm düşürme olmayan senaryoda farklı ölçüm gürültüsü seviyelerinde elde edilen sonuçlar. Her gürültü seviyesinde en iyi sonuçlar kalın olarak gösterilmiştir. . . 72
Tablo 3.12 *val-car* 0015 sekansı için $R = 5$ koşulunda ölçüm düşürmeli senaryo sonuçları. Her düşürme oranında en iyi sonuçlar kalın olarak gösterilmiştir. 72
Tablo 3.13 Ortalama konum hatası bileşenleri (dx, dy) ve normu 78
Tablo 3.14 Senaryo 15'in Alt Segmentlerinin Performans Metrikleri 78
Tablo 3.15 Çalışma zamanı (senaryo/saniye). Üç tekrar üzerinden ortalama $\pm$ std. 80

ÖZET

Kavşak Senaryolarında Nesne Takibinin Etkileşen Çoklu Modeller Kullanılarak İyileştirilmesi

Çağrı HAKKOYMAZ

Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Dr. Öğr. Üyesi Erkan USLU

Kavşaklar, çok sayıda nesne barındıran ve bu nesnelerin aynı senaryoda farklı hareket örüntüleri sergileyebildiği dinamik ortamlardır (örneğin sağa dönüş senaryosunda nesne doğrusal hareketle başlayıp dönüş manevrası yaparak tekrar doğrusal harekete dönebilir). Bu geçişlerin doğru modellenmesi başarılı takip performansı için kritik öneme sahiptir. Özellikle sensörlerin güvenilirliğinin azaldığı ve ölçüm doğruluklarının düştüğü ortamlarda, kullanılan hareket modelinin nesneyi iyi temsil edebilmesi takip hassasiyeti açısından avantaj sağlamaktadır.

Bu tez çalışmasında, kavşak senaryolarında nesne takibinin doğruluğunu artırmak amacıyla Etkileşen Çoklu Model (Interacting Multiple Model - IMM) yaklaşımı kullanılmıştır. Çalışma kapsamında, KITTI veri setindeki kavşak senaryoları seçilerek analizler gerçekleştirilmiştir. IMM çerçevesinde Sabit İvmeli (CA) ve Sabit Dönüş Oranı ve İvme (CTRA) hareket modelleri arasında dinamik geçişler uyarlanmıştır.

Azalan sensör güvenini simüle etmek için ölçüm kovaryans matrisi kontrollü biçimde artırılarak ölçüm belirsizliği yükseltilmiştir. Ayrıca, geçici algılama kayıplarını temsil etmek amacıyla belirli oranlarda ölçüm düşürme (eksik gözlem) senaryoları uygulanmıştır.

Performans farklı algılayıcılarla (VirConv ,CasA), HOTA ve alt bileşenleri (DetA, AssA, DetRe, DetPr, AssRe, AssPr) üzerinden değerlendirilmiştir. %20 ölçüm kaybında IMM en iyi sonucu vermiştir (VirConv: +0.245 HOTA CTRA'ya göre;

CasA: +0.913 HOTA CA'ya göre). Yüksek gürültüde (R=50) CA en iyi olsa da IMM, CTRA'ya göre daha dayanıklıdır (VirConv: +1.458 HOTA; CasA: +0.783 HOTA). Örneğin, senaryo 0012'de R=50 koşulunda HOTA skoru CA için 82.002'den IMM ile 82.557'ye yükselmiş; dönüş segmentlerinde ortalama konum hatası normu CA'da 0.6961 m iken IMM'de 0.6546 m olarak ölçülmüştür. Elde edilen sonuçlar, IMM tabanlı takip sisteminin kavşak senaryolarında daha yüksek doğruluk ve kararlılık sunduğunu göstermektedir. Ayrıca, sensör güvenilirliğinin azaldığı koşullarda adaptif model kullanımının performansı anlamlı şekilde iyileştirdiği gözlemlenmiştir.

Anahtar Kelimeler: Nesne takibi, Etkileşen çoklu model (IMM), kavşak senaryoları, sensör belirsizliği, ölçüm düşürme

ABSTRACT

Improving Object Tracking in Junction Scenarios Using Interacting Multiple Models

Çağrı HAKKOYMAZ

Department of Computer Engineering
Master of Science Thesis

Supervisor: Asst. Prof. Erkan USLU

Junctions are dynamic environments that contain a high number of objects, each potentially following different motion patterns within the same scenario. For example, in a right-turn scenario, an object may initially move linearly, then execute a turning motion, and finally continue moving linearly again. Accurately modeling these transitions is critically important for achieving successful tracking performance. This is especially true in conditions where sensor reliability decreases and measurement accuracy is reduced. In such cases, the ability of the selected motion model to effectively represent object dynamics offers a clear advantage in terms of tracking precision.

In this thesis, the Interacting Multiple Model (IMM) approach is utilized to improve object tracking accuracy in intersection scenarios. Intersection-specific scenes from the KITTI dataset were selected for analysis. Within the IMM framework, two motion models—Constant Acceleration (CA) and Constant Turn-Rate and Acceleration (CTRA)—were dynamically combined and adapted based on their suitability for the scenario.

To simulate reduced sensor confidence, the measurement covariance matrix was deliberately increased to emulate higher observation uncertainty. In addition, measurement dropping (intermittent missing observations) was applied to evaluate robustness under temporary sensing dropouts.

Performance was evaluated using two different detectors (VirConv and CasA) with

HOTA and its components (DetA, AssA, DetRe, DetPr, AssRe, AssPr). Under 20% measurement drop, IMM achieved the best performance (VirConv: +0.245 HOTA vs. CTRA; CasA: +0.913 HOTA vs. CA). Under high measurement noise (R=50), CA yielded the best HOTA, while IMM remained more robust than CTRA (VirConv: +1.458 HOTA; CasA: +0.783 HOTA). For instance, on sequence 0012 under R=50, HOTA increases from 82.002 with CA to 82.557 with IMM, while the mean position error norm over turning segments decreases from 0.6961 m (CA) to 0.6546 m (IMM). The results demonstrate that the IMM-based tracking system provides higher accuracy and stability in intersection scenarios. Furthermore, it was observed that adaptive model switching improves performance when sensor reliability is compromised.

Keywords: Object tracking, Interacting multiple model (IMM), intersection scenarios, sensor uncertainty, measurement dropout

1
GİRİŞ

Otonom sürüş sistemleri, dinamik bir çevrede algılama planlama kontrol döngüsünü gerçek zamanlı çalıştırmayı gerektirir. Bu döngünün güvenli biçimde işletilebilmesi için çevredeki dinamik aktörlerin (araç, yaya vb.) konum ve hız gibi durumlarının zaman içinde tutarlı biçimde izlenmesi kritik önemdedir.

Prototip ürün gösterimlerinden seri üretim odaklı sistemlere geçişte ise yalnızca doğruluk değil; maliyet, güç tüketimi, entegrasyon karmaşıklığı ve bakım gibi kısıtlar da belirleyici hale gelmektedir. Bu durum ölçümlerin belirsizliğinin artması veya zaman zaman ölçüm kayıpları yaşanması gibi pratik sorunları daha görünür hale getirebilir ve özellikle manevraların sık olduğu senaryolarda takip kararlılığını zorlayabilir.

1.1 Tezin Amacı

Bu tez çalışması, otonom sürüşte *çoklu nesne takibi* (Multi-Object-Tracking-MOT) problemine odaklanmaktadır. MOT kapsamında amaç, sahnedeki birden fazla dinamik nesnenin zaman içinde tutarlı biçimde izlenmesi; konum kestiriminin korunması ve iz kimliği sürekliliğinin sağlanmasıdır. Bu tezde özellikle sensör ölçümlerinin belirsizleştiği ve/veya kısmen kaybolduğu koşullarda, farklı hareket modelleri ve Etkileşen Çoklu Model (IMM) yaklaşımının takip performansına etkisi incelenmektedir.

Otonom sürüş teknolojilerinin geliştirilme süreci, ilk aşamalarda konsept doğrulama ve teknolojik yeterliliğin gösterilmesine odaklanmıştır. Bu dönemde sensörlerin maliyeti, hacmi veya enerji tüketimi gibi operasyonel kısıtlar göz ardı edilmiş; sistemlerin güvenilir çalışması için LiDAR, radar ve kamera gibi sensörler bir arada ve genellikle yedekli şekilde kullanılmıştır. Algılama, takip ve planlama görevlerinde yüksek doğruluk elde etmek adına, sensörler arası füzyon yoğun biçimde uygulanmış ve filtreleme algoritmalarında ölçüm belirsizlikleri göz ardı edilmiştir.

Ancak konseptten ürüne geçiş sürecinde; özellikle son kullanıcıya hitap eden ticari araçların geliştirilmesinde, sensör doğruluğu kadar maliyet, güç tüketimi, entegrasyon zorluğu ve bakım süresi gibi kriterler de öne çıkmaya başlamıştır. Bu durum, sistem tasarımında daha düşük hassasiyetli sensörlerin tercih edilmesine neden olabilmektedir. Diğer bir deyişle, sistemdeki sensör güvensizliği artık yapısal bir tasarım parametresi haline gelmiştir.

Bu tez, otonom sürüşte *çoklu nesne takibi* problemine odaklanmakta; sensör ölçümlerinin belirsizleştiği ve/veya kısmen kaybolduğu durumlarda iz (track) sürekliliği ve konum kestiriminin nasıl etkilendiğini ve filtre düzeyinde nasıl iyileştirilebileceğini incelemektedir. Bu tez çalışmasının temel amacı, otonom sistemlerde sensör güvensizliğinin neden olabileceği takip hatalarını anlamak ve bu hataların filtre düzeyinde nasıl telafi edilebileceğini araştırmaktır. Bu tez kapsamında takip problemi yer düzlemi üzerinde iki boyutlu (2D) olarak ele alınmış; değerlendirmeler konumun x–y düzlemindeki takibi ve kimlik sürekliliği üzerinden gerçekleştirilmiştir. Özellikle, düşük güvenilirliğe sahip ölçümlerin yer aldığı ya da bazı ölçümlerin tamamen kaybolduğu senaryolarda, hareket modelinin sisteme katkısı ve takip başarımı üzerindeki etkisi incelenmiştir. Bu kapsamda, aşağıdaki özel hedefler tanımlanmıştır:

- Gerçek dünya veri seti (KITTI) kullanılarak CA ve CTRA hareket modellerinin, ayrıca bu iki modeli birleştiren IMM yaklaşımının takip doğruluğuna etkisinin karşılaştırılması
- Etkileşen Çoklu Model (IMM) filtresinin, sensör bozulması altındaki senaryolarda takip kararlılığına katkısının analiz edilmesi
- Ölçüm güvenilirliğinin yapay olarak bozulduğu senaryolar tasarlanarak (ölçüm kovaryansının artırılması ve belirli oranlarda ölçüm düşürme (dropout)), filtrelerin bu koşullarda verdiği tepkinin nicel olarak incelenmesi
- HOTA ve alt bileşenleri (DetA, AssA, DetRe, DetPr, AssRe, AssPr) yardımıyla izleme performansının bütüncül şekilde değerlendirilmesi

Bu çalışmanın uzun vadeli amacı, ticari otonom sistemlerde kullanılan takip yazılımlarının, yalnızca ideal şartlarda değil, aynı zamanda sensörün ölçüm yapamadığı ya da hatalı ölçüm yaptığı durumlarda da güvenilir çalışmasını sağlamaktır. Bu bağlamda, tez kapsamında geliştirilen ve test edilen yaklaşımlar, düşük maliyetli sensör kurulumlarıyla yüksek güvenilirlik hedefleyen mimariler için bir referans oluşturmaktadır.

1.1.1 Katkılar

Bu tez çalışmasının başlıca katkıları şunlardır:

- **(K1) IMM tabanlı kestirim:** CA ve CTRA hareket modellerinin IMM çatısı altında etkileşimli biçimde birleştirilmesi ve kavşak manevralarında daha kararlı takip elde edilmesi.
- **(K2) Pratik sağlamlık iyileştirmeleri:** CA–CTRA etkileşiminin sayısal kararlılığını artırmak amacıyla düşük hızda sapma açısı (yaw) sabitleme, kovaryans dönüşümünde kararlılık önlemleri ve sapma açısı (yaw) için açı-uyumlu karıştırma stratejilerinin uygulanması.
- **(K3) Sensör bozunumu altında değerlendirme:** İdeal ölçüm koşullarıyla sınırlı kalmadan, ölçüm belirsizliği artırımı ve ölçüm düşürme senaryoları altında takip performansının sistematik olarak incelenmesi.
- **(K4) Kavşak odaklı analiz ve görselleştirme:** HOTA gibi toplu metriklere ek olarak, nesne-bazlı grafikler ve görsel çıktılarla manevra anlarında filtre davranışının yorumlanmasını sağlayan analiz/görselleştirme bileşenlerinin geliştirilmesi.

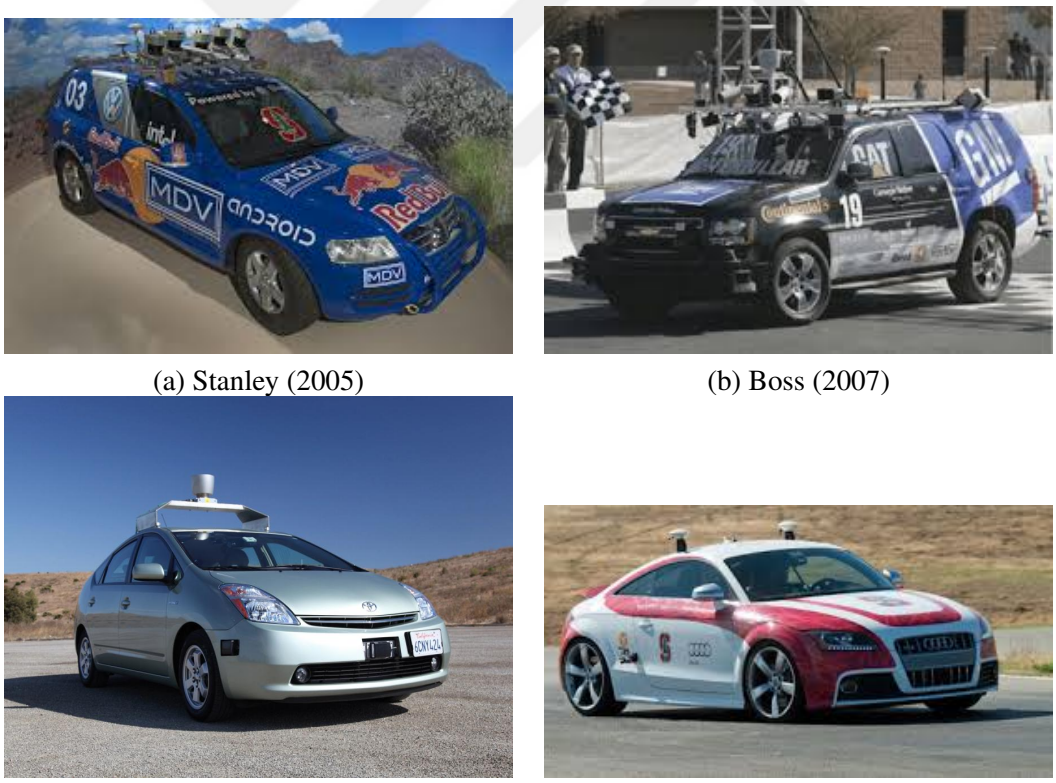
Sonuç olarak bu tez, gerçek hayattaki sınırlı sensör verisi koşullarında çalışan daha sağlam, güvenli ve maliyet etkin takip algoritmalarına yönelik mühendislik bakışını teorik ve uygulamalı olarak ortaya koymayı amaçlamaktadır.

1.2 Literatür Özeti

Otomotiv sektöründe uzun bir süredir otonom sürüşün mümkün kılınması adına çalışmalar yapılmaktadır. İlk olgun çalışmaların tarihi 1980'li yıllara kadar gitmektedir. Bu tarihlerde, Ernst D. Dickmanns liderliğinde yürütülen çalışmalar ile görüntü işleme tabanlı ilk otonom sürüş sistemleri geliştirilmiştir [1]. 1985 yılında tanıtılan bu sistemde, kamera görüntülerinden yol şeklinin anlaşılması ve araç yönlendirmesinin gerçekleştirilmesi sağlanmıştır. Bu yaklaşım, sensör verilerinin gerçek zamanlı işlenmesiyle, aracın aktif olarak çevresini algılayıp kontrol edebilmesine olanak tanımıştır. Böylece, otonom sürüş için gerekli olan algılama–planlama–kontrol döngüsünün temel adımları atılmıştır. Otonom sürüş sistemlerinde yapılan çalışmalar, 2004 yılında ilki düzenlenen Defence Advanced Research Projects Agency (DARPA) Grand Challenge ile büyük bir ivme kazanmıştır. Yarışmanın ilk yıllarından itibaren Stanford Üniversitesi, Carnegie

Mellon Üniversitesi, Massachusetts Institute of Technology gibi öncü akademik kurumlar bu alanda aktif rol alarak teknolojik ilerlemenin temelini atmıştır.

2004 yılında DARPA Grand Challenge yarışmasının başlatılmasıyla birlikte üniversitelerde otonom araç geliştirme problemine olan ilgi artmış, bu alanda kurulan takımlar prototip araçlar geliştirmeye başlamış ve bir yıl sonra Stanford'un geliştirdiği Stanley aracı bu yarışmayı kazanarak geniş yankı uyandırmıştır [2]. Takip eden yıllarda (2005) bu alandaki ilk girişimler ortaya çıkmış, 2007 yılında düzenlenen DARPA Urban Challenge ile çok daha karmaşık olan şehir içi otonom sürüş senaryoları akademik toplulukta tartışmaya açmıştır. Carnegie Mellon Üniversitesi'nin "Boss" aracı (Tartan Racing), DARPA Urban Challenge'ı kazanarak şehir içi otonom sürüş teknolojilerinin öncüllerinden biri olmuştur[3]. 2009 yılında Google, daha sonra Waymo'ya dönüşecek olan kendi otonom araç projesini başlatmıştır. 2012'de Stanford'un geliştirdiği otonom yarış aracı, Audi TTS ile Pikes Peak'e tırmanarak yüksek hızda tam otonom kontrolü ile çalışmaya sunmuştur [4]. Bu döneme ait öne çıkan platformlar Şekil 1.1 üzerinde örneklenmiştir.



(a) Stanley (2005)

(b) Boss (2007)

(c) Google prototipi (2009)

(d) Stanford aracı (2012)

Şekil 1.1 Örnek otonom sürüş platformları.

2013-2017 yılları arasında otonom sürüş araştırmalarında, özellikle algılama (perception) tarafında derin öğrenme tabanlı yaklaşımların yaygınlaşmasıyla belirgin bir ivmelenme yaşanmıştır. Bu dönemde nesne tespitinde Faster R-CNN

gibi bölge önerimli ağlar ve semantik bölümlemede FCN gibi tam evrişimli mimariler, kamera tabanlı algı modüllerinin doğruluk ve genellenebilirliğini artırmıştır [5, 6]. Ayrıca KITTI gibi veri kümeleri etrafında oluşan karşılaştırmalı değerlendirme kültürü, algılama ve yerelleştirme problemlerinde ortak metriklerle ilerlemeyi hızlandırmıştır [7]. Aynı zaman aralığında LiDAR tabanlı 3B algılama ve çoklu sensör füzyonu tarafında MV3D benzeri yaklaşımlar öne çıkmış; nokta bulutu temsillerinde PointNet gibi öğrenme tabanlı yöntemler, 3B algı ve takip/füzyon hattının önemli bileşenlerinden biri haline gelmiştir [8, 9]. Planlama ve kontrol tarafında ise otonom sürüşe yönelik yöntemlerin sistematik biçimde ele alındığı derlemeler, sahada kullanılan yaklaşım sınıflarının (ör. örnekleme tabanlı planlama ve MPC) daha net ayrışmasına katkı sağlamıştır [10].

2017 sonrası dönemde literatürde yalnızca tekil modüllerin doğruluğunu artırmanın ötesinde, belirsizliğin modellenmesi ve emniyet/güvenlik odaklı karar verme yaklaşımları daha görünür hale gelmiştir; bu çerçevede RSS gibi güvenlik modelleri tartışılmıştır [11]. Aynı dönemde açık ve büyük ölçekli veri kümeleri (örn. nuScenes, Waymo Open Dataset) genellenebilirliğin ölçülmesi ve farklı sensör kurulumlarının kıyaslanması açısından önemli bir rol oynamıştır [12, 13]. Bununla birlikte CARLA gibi simülasyon ortamları, senaryo tabanlı test ve geliştirme yaklaşımlarının yaygınlaşmasına katkı sağlamıştır [14]. Son yıllarda otonomi kabiliyetinin değerlendirilmesinde operasyonel tasarım alanı (ODD) ve otonomi seviye tanımları gibi çerçeveler, akademik ve endüstriyel çalışmalarda ortak bir terminoloji oluşturmuştur [15].

Bu gelişmeler, otonom sürüş yazılım yığınının her bir katmanında (algılama, konumlama/haritalama, tahmin, planlama ve kontrol) yoğun bir literatür oluşmasını teşvik etmiştir. Tezin bağlamını netleştirmek amacıyla, otonom araçlardaki temel alt görevler aşağıda ayrı başlıklar altında özetlenmiş; özellikle *çoklu nesne takibi* bileşeninin algılama ve sensör füzyonu çıktılarıyla nasıl beslendiği ve planlama/kontrol tarafını nasıl etkilediği ilişkilendirilerek açıklanmıştır. Tipik bir otonom sürüş hattında sensörlerden (kamera/LiDAR/radar vb.) gelen ham veriler kalibrasyon ve zaman senkronizasyonu adımlarından geçer; ardından algılama ve/veya sensör füzyonu modülleri ile şeritler, serbest alan ve dinamik nesneler (araç, yaya vb.) tespit edilerek anlık ölçümler üretilir. Bu ölçümler zaman içinde gürültü içerebilir veya bazı anlarda eksik kalabilir.

MOT, bu algılama/füzyon çıktıları üzerinden veri ilişkilendirme, iz yönetimi (track management) ve durum kestirimi gerçekleştirerek her nesne için zaman tutarlı bir iz kimliği ile birlikte konum, hız gibi hareket durumlarını üretir; böylece planlama modülüne aktarılabilecek dinamik bir çevre modeli sağlar. MOT

çıktıları üzerinden yürütülen tahmin (prediction) aşamasında nesnelerin kısa vadeli gelecekteki hareketleri öngörülür ve bu bilgi, çarpışma riski değerlendirmesi ile güvenli manevra seçimi açısından kritik girdi oluşturur. Planlama modülü, tahmin edilen dinamik çevre ve trafik kuralları altında davranış seçimi ile yörünge/hız profili üretirken, kontrol modülü de bu referansları direksiyon gaz fren komutlarına dönüştürerek araç üzerinde uygular. Bu tez kapsamında MOT problemine odaklanılarak, ölçümlerin belirsizleştiği ve/veya kaybolduğu durumlarda hareket modelinin (CA, CTRA) ve çoklu-model yaklaşımının (IMM) iz sürekliliği ve konum doğruluğu üzerindeki etkisi incelenmiştir.

1.2.1 Algılama (Perception)

Algılama modülü, otonom aracın çevresini ham sensör ölçümlerinden başlayarak planlama ve tahmin katmanlarının doğrudan kullanabileceği bir *çevre temsiline* dönüştürür. Bu temsil; (i) dinamik aktörlerin (araç, yaya, bisikletli vb.) tespiti ve sınıflandırılması, (ii) 3 boyutlu konum ve araç sapma açısının (heading/yaw) bulunması, (iii) şerit/yol sınırı, araç sürülebilir alan (free-space) ve semantik bölütleme (semantic segmentation) gibi statik sahne bilgileri ve (iv) nesne takibi ile birlikte belirsizlik (uncertainty/covariance) içeren tutarlı izlerin (tracks) üretilmesi gibi çıktıları kapsar [12, 16]. Algılama çıktıları, özellikle dinamik engellerin geleceğe taşınması ve risk-duyarlı karar verme için tahmin (prediction) ve planlama modüllerine temel girdi oluşturur.

Algılama hattının omurgasını çoğu pratik mimaride kamera LiDAR radar üçlüsü oluşturur: kamera yüksek çözünürlüklü semantik ayrıntı (sınıf, şerit, levha/ışık benzeri görsel ipuçları) sağlarken, LiDAR 3B geometri ve hassas mesafe bilgisinde güçlüdür; radar ise olumsuz hava/aydınlatma koşullarında dayanım ve doğrudan hız ölçümü avantajı sunar [12, 16]. Kısa menzil yakın çevre için ultrasonik sensörler (park/düşük hız manevraları) tamamlayıcı rol oynayabilir. Bu çoklu-sensör yapıda güvenilir algılama için zaman senkronizasyonu ve kalibrasyon (intrinsic/extrinsic) kritik önemdedir; tüm ölçümlerin ortak bir araç koordinat sisteminde (ego-frame) tutarlı biçimde ifade edilmesi gerekir [12]. Bu nedenle algılama hattı pratikte çoğunlukla (i) sensör ön-işleme, (ii) ego-hareket telafisi (motion compensation), (iii) füzyon ve (iv) görev-bağımlı çıktı üretimi (tespit/bölütleme/takip) aşamalarından oluşur.

Algılama ve takip için kullanılan sensörler, Ignatious ve ark. [17] tarafından genel olarak iki grupta ele alınmaktadır: (i) çevre algılama (exteroceptive) sensörleri ve (ii) aracın kendi durumunu/konumunu belirlemeye yönelik propriyoseptif sensörler. Çevre algılama tarafında kamera, LiDAR, radar ve ultrasonik sensörler yaygınken;

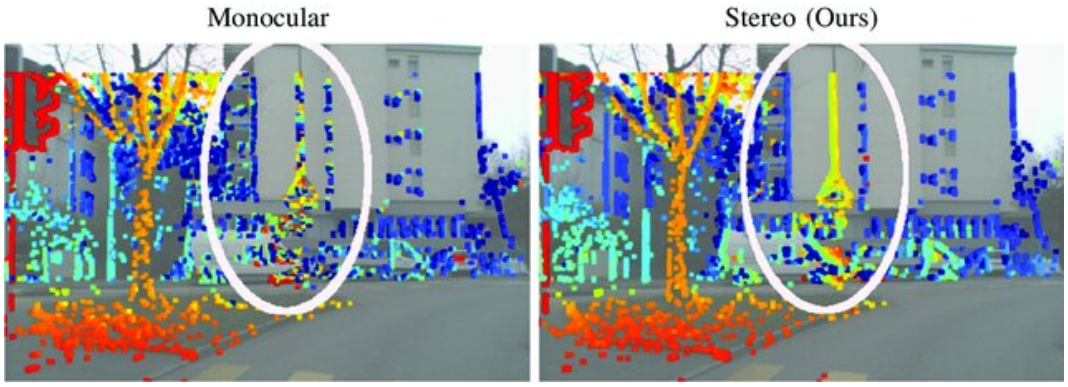
bazı sistemlerde kızılötesi/termal kameralar da kullanılmaktadır [17]. Araç durum kestirimi ve konumlama için ise GNSS, IMU (ivmeölçer/jiroskop) ve tekerlek odometrisi/enkoderler gibi sensörler yaygın biçimde kullanılmaktadır [17].

- **Çevre algılama (exteroceptive) sensörleri:**
  - Kamera (RGB/mono/stereo; bazı sistemlerde termal/kızılötesi)
  - Radar (SRR/LRR; bazı uygulamalarda yüksek çözünürlüklü/imaging radar)
  - LiDAR
  - Ultrasonik sensörler (özellikle kısa mesafe: park/kör nokta benzeri yakın çevre)
- **Araç durumu/konumlama (propriyoseptif) sensörleri:**
  - Global Navigation Satellite System (GNSS)
  - IMU (ivmeölçer/jiroskop)
  - Tekerlek odometrisi / enkoderler

Kamera, insan gözüne benzer şekilde ortamın görüntüsünü sağlayan sensörlerdir. Tekli (monoküler) veya stereo olarak kullanılabilir. Kameralar yüksek çözünürlükte detaylı görsel bilgi sunabilir; yol şeritleri, trafik işaretleri ve nesne sınıflandırması gibi görevlerde yaygın olarak tercih edilir. Düşük maliyetli ve enerji açısından verimli olmaları önemli bir avantajdır. Zayıf aydınlatma, güneş kamaşması, sis gibi hava koşullarında performansları ciddi şekilde düşebilir.

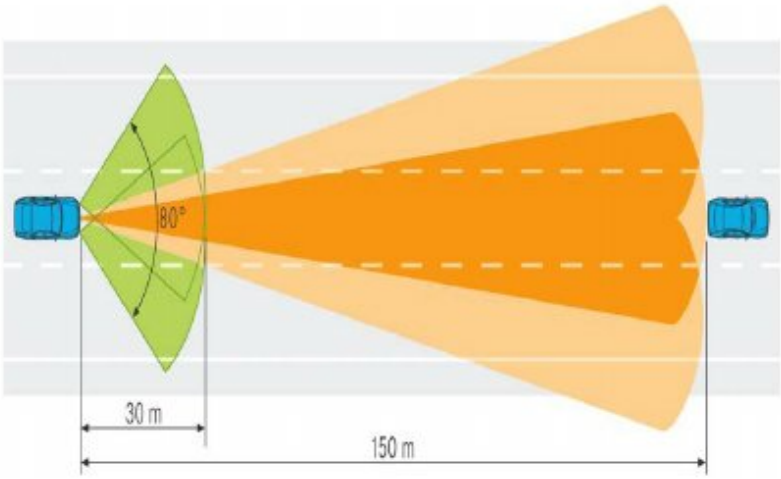
Monoküler kameralar doğrudan derinlik bilgisi sağlayamaz; ancak derin öğrenme yöntemleri kullanılarak derinlik tahmini son yıllarda otonom sürüş sistemlerinde giderek önem kazanmaktadır. [18] bu alandaki yöntemleri detaylı şekilde incelemiş ve özellikle derin öğrenme tabanlı, ölçek geri kazanımı sağlayan yaklaşımlar üzerine kapsamlı bir derleme sunmuştur. Monoküler ve stereo kurulumların derinlik bilgisi üretme kabiliyeti Şekil 1.2 üzerinde örneklenmiştir.

Radar, radyo dalgaları kullanarak çevresindeki nesnelerin mesafesini, hızını ve açı bilgisini tespit edebilen aktif bir sensördür. Özellikle kötü hava koşullarında (yağmur, sis, gece) kararlı çalışması ve hareketli nesnelerin hızını doğrudan ölçebilmesiyle öne çıkar. Uzun menzilli algılamalar için uygundur. Ancak açısal çözünürlüğü düşüktür; bu nedenle nesne boyutunu ve şeklini belirlemede yetersiz kalabilir. Ayrıca radarlar bazı düşük yansıtıcılığa sahip nesneleri (örneğin yayalar veya bisikletliler) algılamada sınırlıdır.



Şekil 1.2 Monoküler ve stereo kurulumlarda derinlik çıktısına örnek. [19].

Radar sistemleri genellikle Long Range Radar (LRR) ve Short Range Radar (SRR) olmak üzere ikiye ayrılır. **LRR**, 200 metreye kadar algılama yapabilir ve genellikle öndeki araçları takip etmek, çarpışma önleme ve adaptif hız sabitleyici (ACC) sistemlerinde kullanılır. Daha düşük açısal çözünürlüğe sahiptir fakat uzun menzilde kararlıdır. **SRR** ise 30-50 metre aralığında çalışır; kör nokta tespiti, çapraz trafik uyarısı ve park yardımı gibi kısa menzilli uygulamalar için uygundur. Daha geniş görüş açısına sahiptir ve yakın çevredeki nesneleri algılamada etkilidir. Şekil 1.3, SRR geniş açılı kısa menzil kapsamasını (yeşil) ve LRR'nin daha dar açılı uzun menzil kapsamasını (turuncu) görsel olarak özetlemektedir.



Şekil 1.3 Otomotiv radarlarında kısa menzil geniş açılı SRR (yeşil) ve uzun menzil dar açılı LRR (turuncu) kapsama bölgelerine örnek. [20].

LiDAR, lazer ışınları gönderir ve geri dönen yansımaları okuyarak çevredeki nesnelerin üç boyutlu nokta bulutunu oluşturur. Elde edilen yüksek çözünürlüklü çevre temsili, karmaşık sahnelerde nesnelerin yüksek doğrulukla belirlenebilmesini sağlar. LiDAR sensörleri gece-gündüz fark etmeksizin çalışabilmesi nedeniyle de tercih edilebilir. Buna karşın yüksek maliyet, bazı modellerde hareketli mekanik parçalar ve olumsuz hava koşullarında (yağmur, kar, toz) performans düşüşü gibi

dezavantajları bulunmaktadır. Otonom araçlarda kullanımı yaygın iki LiDAR mimarisi olan mekanik taramalı (spinning) ve katı-hâl (solid-state) LiDAR örnekleri Şekil 1.4 üzerinde gösterilmiştir.



(a) Spinning LiDAR (b) Solid-state LiDAR

Şekil 1.4 LiDAR sensör mimarilerine örnek: spinning ve solid-state.

Ultrasonik sensörler, yüksek frekanslı ses dalgaları ile kısa mesafede (yakın çevre) engel algılamaya yarayan aktif sensörlerdir. Özellikle park yardımı, düşük hız manevralarında çarpışma önleme ve çok yakın mesafedeki nesnelerin algılanması gibi görevlerde yaygın olarak kullanılır [17]. Düşük maliyetli olmaları önemli bir avantajdır; ancak menzillerinin kısa olması ve ölçümlerin yüzey özelliklerinden ve çevresel koşullardan etkilenebilmesi (ör. yağmur, kar, yumuşak/emme yapan yüzeyler) temel sınırlılıklarıdır.

Termal kızılötesi kamera, nesnelerin yaydığı ısı radyasyonuna dayalı görüntü üreterek özellikle düşük aydınlatma koşullarında (gece sürüşü) yaya ve hayvan gibi sıcaklık farkı belirgin hedeflerin algılanmasını destekleyebilir [17]. Buna karşın çözünürlük, maliyet ve bazı çevresel koşullarda kontrastın azalması gibi etkenler nedeniyle genellikle kamera/LiDAR/radarın yerine değil, tamamlayıcı bir sensör olarak değerlendirilmektedir.

Küresel Navigasyon Uydu Sistemi (Global Navigation Satellite System – GNSS), aracın dünya koordinatlarındaki mutlak konumunu (enlem, boylam, yükseklik) tahmin etmek için kullanılan konumlama sensörüdür [17]. GNSS küresel ölçekte mutlak konum avantajı sağlasa da şehir kanyonları, tüneller ve çok yollu yansıma (multipath) gibi durumlarda doğruluk ve süreklilik sorunları görülebilir; bu nedenle pratikte genellikle IMU ve odometri gibi sensörlerle birleştirilerek daha kararlı konum kestirimi elde edilir.

Ataletsel Ölçüm Birimi (Inertial Measurement Unit – IMU), ivmeölçer ve jiroskop ölçümlerinden aracın açısal hızını ve ivmesini yüksek frekansta sağlayan birimdir [17]. IMU kısa zaman aralıklarında dinamik hareketi iyi yakalasa da ölçümlerin entegrasyonundan kaynaklanan sürüklenme (drift) nedeniyle tek başına uzun süreli mutlak konum sağlamada yetersiz kalabilir; bu nedenle GNSS ve

tekerlek odometrisiyle füzyon içinde kullanımı yaygındır.

Tekerlek odometrisi / enkoderler (Wheel Odometry / Encoders), tekerlek dönüşlerinden aracın hızını ve kat ettiği mesafeyi tahmin etmek için kullanılır [17]. Yerel hareket kestiriminde faydalı olmakla birlikte lastik kayması, yol koşulları ve lastik aşınması gibi etkiler hataya neden olabilir; bu yüzden çoğunlukla IMU ve GNSS ile birlikte değerlendirilir.

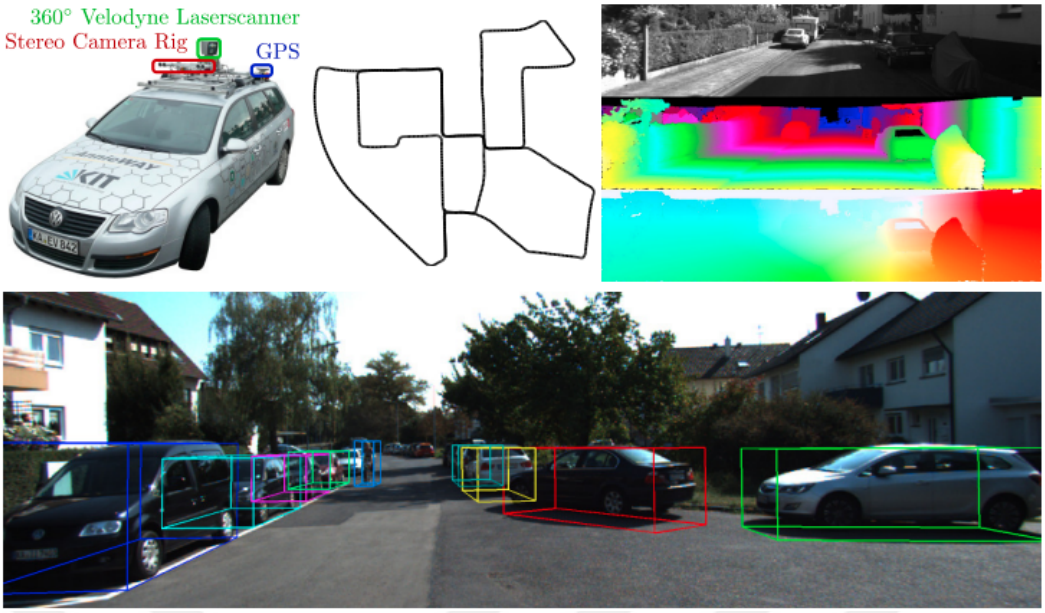
Bu sensörlerin birlikte ve tamamlayıcı şekilde kullanımı, algılama hattının dayanıklılığını artırır. Qian ve ark. [21] derleme çalışmasında çoklu sensör füzyonunu; füzyon seviyesi (erken/veri düzeyi, özellik düzeyi, nesne düzeyi ve karar düzeyi), kalibrasyon ve zaman senkronizasyonu, belirsizlik yönetimi ve gerçek zamanlılık gibi pratik kısıtlar bağlamında sınıflandırmakta ve otonom sürüşte kamera, LiDAR, radar kombinasyonlarının tamamlayıcı rolünü vurgulamaktadır. Örneğin kamera semantik ayrıntı sağlarken, LiDAR 3B geometri ve hassas mesafe bilgisinde güçlüdür; radar ise hız ölçümü ve olumsuz hava koşullarındaki dayanımıyla öne çıkar [21].

Algılama mimarileri, üretilen temsilin tipine göre görüntü uzayı (image-space), nokta bulutu/voxel temsili (point/voxel) veya kuş-bakışı görünüm (Bird's-Eye View, BEV) ekseninde ele alınabilir. Kamera tabanlı gerçek zamanlı nesne tespitte YOLO gibi tek-aşamalı mimariler yaygın bir başlangıç noktasıdır [22]. LiDAR tabanlı 3B tespit tarafında PointPillars gibi verimli kodlayıcılar ve CenterPoint gibi merkez tabanlı yaklaşımlar, otonom sürüş veri setlerinde güçlü ve pratik temel karşılaştırma yöntemleri olarak öne çıkmaktadır [23, 24]. KITTI veri setinden örnek bir sahnede, kamera görüntüsü üzerinde gösterilen 3B sınır kutusu (3B sınırlayıcı kutu) anotasyonları/çıktıları Şekil 1.5 üzerinde verilmiştir [7].

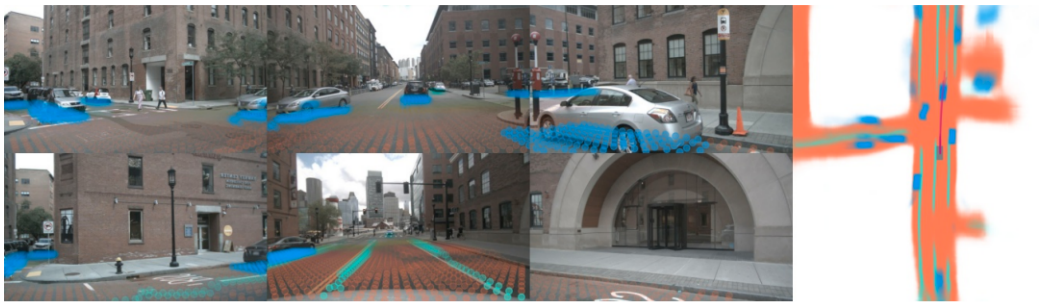
Statik sahne çıktıları (şerit/serbest alan/semantik) için ise DeepLabv3+ benzeri encoder–decoder segmentasyon mimarileri ve Mask R-CNN benzeri örnek-temelli (instance) ayrıştırma yaklaşımları yaygın araçlardır [25, 26]. Son yıllarda kamera tabanlı algılamada BEV temsiline "lift-and-project" yaklaşımıyla geçiş de popülerleşmiştir [27]. Bu dönüşümün sezgisel bir örneği, çoklu kameradan elde edilen görüntü ipuçlarının kuş-bakışı (BEV) düzleminde birleştirilerek sürülebilir alan/işgal (occupancy) benzeri bir temsil üretmesi açısından Şekil 1.6 üzerinde gösterilmiştir [27].

1.2.1.1 Çoklu-Nesne Takibi (MOT):

Çoklu-sensör mimarilerinde **füzyon**, bilginin hangi seviyede birleştirildiğine göre genellikle erken (veri düzeyi), orta (özellik düzeyi) ve geç (nesne/karar



Şekil 1.5 KITTI otonom sürüş veri setinden örnek: sahne çıktıları üzerinde 3B sınır kutuları ile 3B algılama/tespit örnek görselleştirmesi [7].



Şekil 1.6 Kamera tabanlı algılamada kuş-bakışı görünüm (BEV) temsiline örnek [27].

düzeyi) füzyon olarak sınıflandırılır. Kamera+LiDAR füzyonuna örnek olarak AVOD ve PointPainting; kamera+radar füzyonuna örnek olarak CenterFusion gibi yaklaşımlar, sensörlerin tamamlayıcı özelliklerinin algılama başarımını artırabildiğini göstermektedir [28–30]. Uygulamada füzyon tasarımı; gecikme (latency), hesaplama bütçesi ve kalibrasyon hassasiyeti gibi gerçek-zamanlılık kısıtları ile birlikte değerlendirilir.

Algılama çıktılarının planlama/tahmin için güvenilir hale gelmesi çoğu zaman **nesne takibi (multi-object tracking, MOT)** ile sağlanır. Otonom sürüşte en yaygın yaklaşım *tespit temelli takip* (tracking-by-detection) olup, her zaman adımında algılayıcı tarafından üretilen tespitler (2B/3B kutular veya merkez temsilleri) bir önceki zamandan taşınan iz (track) hipotezleri ile veri ilişkilendirme (data association) üzerinden eşleştirilir; eşleşen izler bir filtre ile güncellenir, eşleşmeyen tespitlerden yeni izler başlatılır ve belirli süre eşleşmeyen izler sonlandırılır [31–34]. Veri ilişkilendirme adımında pratikte çoğunlukla bir geçit (gating) mekanizması ile olası olmayan eşleşmeler elenir (ör. Mahalanobis uzaklığı eşiği) ve kalan adaylar üzerinde eniyileme tabanlı atama (çoğunlukla Hungarian algoritması) gerçekleştirilir [32, 33]. Bu süreç sonunda, planlama ve tahmin modüllerine aktarılmak üzere belirsizliği tanımlanmış (uncertainty/covariance) tutarlı izler (tracks) üretilir.

2B takip tarafında SORT, her nesne için Kalman filtre tabanlı hareket kestirimi yapar ve tespit–iz eşleştirmesinde IoU tabanlı maliyet kullanarak çevrim-içi ve hafif bir temel yöntem sunar [32]. DeepSORT ise bu yapıyı tespitlerden çıkarılan görünüş (appearance) gömütleriyle (re-identification) güçlendirerek özellikle örtüşme (occlusion) ve yoğun trafik gibi durumlarda kimlik sürekliliğini (ID switch azaltımı) iyileştirmeyi hedefler [33]. Bu tür 2B yaklaşımlar, kamera tabanlı algılamanın baskın olduğu sistemlerde yaygın olarak kullanılmaktadır.

MCTrack ve benzeri çok-ipuçlu takip yaklaşımları, veri ilişkilendirme kararını yalnızca tek bir ölçüte (ör. yalnızca IoU) dayandırmak yerine; hareket tutarlılığı, geometrik yakınlık, sınıf tutarlılığı ve (varsa) görünüş/özellik benzerliği gibi birden fazla ipucunu birlikte değerlendirerek daha kararlı eşleştirme yapmayı amaçlar. Bu tasarım, özellikle tespit güveninin düştüğü (ölçüm bozunumu) veya kavşak/yoğun trafik gibi sahnelerde örtüşmelerin arttığı durumlarda iz sürekliliğini artırmaya yardımcı olabilir.

3B takip tarafında temel fikir, LiDAR tabanlı 3B tespitlerin (3B sınırlayıcı kutu) veya BEV düzlemindeki merkez temsillerinin zaman içinde ilişkilendirilmesidir. AB3DMOT, 3B kutu parametreleri üzerinde Kalman filtre ile durum kestirimi

yapıp eşleştirmede 3B IoU/mesafe tabanlı maliyetler kullanan, çevrim-içi ve yaygın bir temel yaklaşım sunar [34]. Benzer şekilde CenterPoint gibi merkez-tabanlı 3B tespit yaklaşımları, takip aşamasında merkez yakınlığı ve hız tutarlılığı gibi ölçütlerle basit ancak etkili ilişkilendirme stratejileriyle birleştirilebilmektedir [24]. Kavşak/yoğun trafik gibi manevra çeşitliliğinin arttığı durumlarda tek hareket modeli yetersiz kalabildiğinden, IMM gibi çoklu-model filtreler ve JPDA gibi olasılıksal ilişkilendirme yöntemleri belirsizlik altında iz sürekliliğini artırmak için önem kazanır [35, 36]. Sonuçta algılama+takip hattı; planlama ve tahmin modüllerine, belirsizliği tanımlanmış dinamik bir çevre modeli aktararak güvenli karar verme sürecinin temelini oluşturur.

Son yıllarda diğer araçların yörüngelerini tahmin etme odaklı veri kümelerinin ölçek ve çeşitlilik olarak hızla büyümesi, etkileşimli sürüşün (interactive driving) ve özellikle **kavşak/ada (roundabout) gibi karmaşık senaryoların** ayrı bir araştırma ekseni olarak olgunlaştığının güçlü bir göstergesidir. Örneğin RWTH Aachen University, Institute for Automotive Engineering (ika) tarafından oluşturulan highD (otoyol), inD (kavşak) ve rounD (dönel kavşak/ada) veri kümeleri; doğal sürüş verisi üzerinde özellikle planlama ve tahmin problemlerinin değerlendirilmesini mümkün kılacak şekilde yayınlanmış ve akademik kullanım için ücretsiz paylaşılmıştır [37–40]. Benzer biçimde INTERACTION veri kümesi, semantik harita bilgisiyle birlikte farklı ülkelerden etkileşimli sürüş senaryoları sunarak, karşıt/işbirlikçi davranışların modellenmesine ve kavşak benzeri senaryolarda tahmin değerlendirmelerine zemin hazırlamıştır [41]. Bu tür veri kümeleri öncelikle yörünge tahminleme için tasarlansa da, kavşak senaryolarının ayrı ve zorlayıcı bir problem sınıfı olarak ele alınmaya başlanması; bu çalışmanın odaklandığı algılama ve nesne takibi hattının kavşak/yoğun etkileşim koşullarında incelenmesi ve sağlamlaştırılması motivasyonunu doğrudan desteklemektedir.

1.2.2 Lokalizasyon (Localization)

Lokalizasyon modülü, otonom aracın dünya/harita koordinat sistemindeki anlık pozunu (konum ve yönelim) ve çoğu zaman hız gibi türetilmiş durum değişkenlerini, bunlara eşlik eden belirsizlik (covariance/uncertainty) ile birlikte kestirerek planlama ve kontrol katmanlarına sağlar. Planlama; şerit geometrisi ve statik/dinamik kısıtları harita üzerinde anlamlandırabilmek için doğru ve sürekliliği yüksek bir poz tahminine ihtiyaç duyar. Kontrol katmanı ise yörünge takip hatasını kapatabilmek için, gecikmesi düşük ve kısa vadede kararlı bir ego-durum kestirimine ihtiyaç duyar [42, 43].

Pratik otonom sürüş mimarilerinde lokalizasyon, farklı hata karakteristiklerine

sahip sensörlerin birbirini tamamlayacak şekilde birleştirilmesiyle gerçekleştirilir. GNSS mutlak konum avantajı sağlarken şehir kanyonları, tüneller ve çok yollu yansıma (multipath) gibi durumlarda süreklilik/doğruluk sorunları yaşar; IMU yüksek frekansta dinamiği iyi yakalar ancak entegrasyon nedeniyle sürüklenir (drift); tekerlek odometrisi kısa vadede faydalı olsa da kayma ve lastik/yol koşullarından etkilenir [43]. Bu nedenle modern sistemlerde yaygın yaklaşım, GNSS+IMU+odometri temelli çıkarımsal konum hesaplama (dead-reckoning) hattını, kamera/LiDAR gibi çevre algılama sensörlerinden türetilen göreli ölçümler ve/veya önceden oluşturulmuş haritalarla (HD map/point cloud map) destekleyerek hem mutlak doğruluğu hem de sürekliliği artırmaktır.

Lokalizasyon algoritmaları genel olarak iki ana çerçevede ele alınır: (i) **filtre tabanlı kestirim** ve (ii) **optimizasyon/faktör-graf tabanlı kestirim**. Filtre tabanlı yaklaşımda (örn. EKF/UKF) sistem dinamiği bir durum-uzayı modeli ile temsil edilir ve sensör ölçümleri çevrim-içi olarak birleştirilir; UKF, doğrusal olmayan sistemlerde türev gerektirmeyen sigma-nokta yaklaşımı ile pratikte yaygın kullanılan bir alternatiftir [42, 44]. Optimizasyon tabanlı yaklaşımda ise geçmiş pencere (sliding window) veya tam yörünge üzerinde hata fonksiyonu minimize edilir; bu yaklaşımın modern robotikteki standart formülasyonlarından biri faktör-graf (factor graph) gösterimidir [45, 46]. Faktör-graf tabanlı yöntemler; çoklu sensör ölçümlerini, zaman senkronizasyonunu ve ek kısıtları (ör. döngü kapama/loop closure, harita eşleştirme) daha esnek biçimde modele katabilmesi nedeniyle özellikle görsel-ataletsel (VIO) ve LiDAR-ataletsel (LIO) odometri/SLAM hatlarında yaygınlaşmıştır.

Çevre algılama sensörleriyle lokalizasyon iki tip problem olarak karşımıza çıkar: **(i) SLAM/odometri** ve **(ii) harita-tabanlı lokalizasyon**. SLAM/odometri yaklaşımında araç, kendi oluşturduğu yerel harita ve ardışık ölçümler üzerinden göreli hareketini (ego-motion) tahmin eder; VIO tarafında VINS-Mono gibi sistemler kamera+IMU ölçümlerini birleştirerek gerçek zamanlı ve düşük sürüklenmeli odometri sağlayabilir [47]. LiDAR odometri tarafında LOAM, nokta bulutu hizalama üzerinden yüksek doğruluklu hareket kestirimi sunan temel çalışmalardan biridir [48]; bunun daha hafif varyantları ve gerçek zamanlı SLAM sistemleri (örn. Cartographer) döngü kapama ile uzun vadeli sürüklenmeyi azaltmayı hedefler [49]. Harita-tabanlı lokalizasyonda ise araç, önceden oluşturulmuş bir haritaya (örn. LiDAR nokta bulutu haritası, semantik landmark haritası, şerit/kenar geometrisi) anlık ölçümlerini eşleştirerek mutlak pozunu düzeltir; bu yaklaşım özellikle GNSS'in zayıfladığı şehir içi senaryolarda kritik rol oynar [50].

Uygulamada bu iki yaklaşım çoğunlukla birlikte kullanılır: kısa vadede IMU/odometri/VIO/LIO ile süreklilik ve düşük gecikme sağlanırken, harita eşleştirme ve GNSS gibi mutlak kaynaklar periyodik düzeltme ile birikimli hatayı sınırlar [43, 50]. Ayrıca güvenlik açısından lokalizasyon hattında çalışma sağlığının takibi (örn. GNSS kalite ölçütleri, tutarsızlık denetimi, sensör arızası veya kayma tespiti) yer alması; belirsizliğin büyüdüğü durumlarda planlama ve hız profilinin daha kontrollü hale getirilmesi için önemlidir [43].

1.2.3 Planlama

Planlama modülü, aracın hedefe ulaşması için çevre bilgisi, yol durumu, trafik kuralları ve dinamik objeleri göz önünde bulundurarak araç için rota oluşturma işlemidir. Otonom araçlarda planlama, klasik UGV literatüründeki "global/local" ayrımına ek olarak; rota seçimi, trafik içinde manevra/davranış seçimi ve araç kısıtları altında yörünge üretimi gibi adımları birlikte ele alan daha hiyerarşik bir yapıda incelenmektedir [10, 51]. Bu hiyerarşide planlamanın çıktısı çoğu zaman *izlenebilir bir referans* (yörünge ve/veya hız profili) iken, bu referansın araca uygulanabilir komutlara dönüştürülmesi ve hatanın kapatılması *kontrol* katmanının sorumluluğundadır [10]. Aşağıda bu bileşenler aynı içerik korunarak rota davranış yörünge ekseninde özetlenmiştir.

Rota (Route) planlama: Aracın başlangıç noktasından hedefe kadar olan tüm rotasını çevresel harita bilgisine dayanarak önceden belirler. Bu tür algoritmalar genellikle harita üzerinde bir yol grafiği oluşturur ve bu grafikte en kısa veya en güvenli yolu bulmaya çalışır. Graf tabanlı Dijkstra [52] ve A* [53] gibi yol bulma algoritmalar bu sınıfta yer alır. Fakat bu algoritmaların durum uzayı büyüdükçe hesaplama maliyetinin artması, örnekleme tabanlı algoritmalara popülerlik kazandırmıştır. Örnekleme tabanlı algoritmalara Rapidly-exploring Random Tree (RRT) [54] ve bu algoritmanın geliştirilmesi ile ortaya çıkan RRT* [55] algoritmaları örnek gösterilebilir. Bu algoritmalar, yüksek boyutlu durum uzayı ve karmaşık ortamlar için daha hızlı ve yeterince etkili çözümler sunmaktadır. Rota planlama, yolun tamamını göz önünde bulundurduğu için optimal sonuçlar sunar; ancak dinamik engelleri veya aracın anlık olarak dikkate alması gereken anlık isterleri hesaba katmaz.

Davranış (Behavior) planlama: Davranış seçimi, kısa vadede uygulanacak manevranın belirlenmesini kapsar (ör. şerit koru/şerit değiştir/acil dur). Otonom sürüş literatüründe bu adım çoğu zaman "motion planning" kapsamında; davranış seçimi, yörünge üretimi, hız profili oluşturma ve takip (tracking) adımlarının birlikte ele alındığı bir bileşen olarak tanımlanır [10, 56, 57]. Bununla birlikte pratik

mimarilerde "takip/izleme (tracking)" fonksiyonu çoğunlukla kontrol katmanında gerçeklenir; planlama tarafı ise izlenecek referansı (yörünge/hız profili) üretir [10]. Farag ve Ali [58] çalışmasında, otoyol sürüşü için yerel planlama problemi; (i) kural-tabanlı bir *Finite State Machine* (FSM) ile manevra seçimi, (ii) quintic polinom/spline tabanlı (jerk-minimized) yörünge üretimi ve (iii) üretilen referansın *Model Predictive Control* (MPC) ile izlenmesi şeklinde bütünleşik bir mimariyle ele alınmıştır.

Hareket/Yörünge (Motion/Trajectory) planlama: Aracın anlık çevresine ve dinamik koşullara göre kısa vadeli kararlar almasını sağlar. Aracın görüş alanındaki engeller, hareketli nesneler ve ani değişiklikler göz önüne alınarak izlenecek yol/yörünge sürekli güncellenir. Bu bağlamda yerel planlayıcı, algılama ve füzyon modülünün sağladığı çevre modeli üzerinden çarpışmasız ve dinamik olarak uygulanabilir bir yörünge üretmekle sorumludur. Çalışmada quintic spline seçimi; başlangıç ve bitişte konum, hız, ivme sınır koşullarını aynı anda sağlayabilmesi ve jerk sürekliliği ile konfor/kararlılık açısından avantaj sağlaması ile gerekçelendirilmiştir [58]. Benzer şekilde, yörünge üretiminde Frenet koordinatlarının kullanımı ve şerit merkez hattına göre (s, d) tanımlı planlama, şerit değiştirme ve engelden kaçınma problemlerini sadeleştiren yaygın bir yaklaşımdır [59, 60].

Bu amaçla pratikte MPC tabanlı yöntemler, örnekleme tabanlı yaklaşımlar ve eğri tabanlı (Bezier/spline/polinom) yörünge üretimi sıklıkla kullanılmaktadır [56, 57]. Örneğin Bezier eğrileri ile MPC'nin birlikte kullanıldığı şehir içi senaryolara yönelik çalışmalar, eğri tabanlı yörünge üretimi ile öngörülü kontrolün birlikte kullanılmasının yerel planlama açısından etkili bir hat oluşturduğunu göstermektedir [61]. Farag ve Ali [58] ise bu yaklaşımı otoyol sürüşüne uyarlayarak, A*, RRT ve Bezier tabanlı planlamaya kıyasla daha düşük jerk ve daha iyi şerit-izleme/çarpışma kaçınma metrikleri raporlamıştır.

DWA algoritması, robotun anlık hız ve yönelme aralığını dikkate alarak, çarpışma riski taşımayan ve hedefe yönelen hız kombinasyonlarını değerlendirir [62]. DWA gerçek zamanlı karar verme açısından güçlü bir temel sunmakla birlikte, otonom araçlarda yüksek hız, şerit geometrisi ve konfor (ivme/jerk kısıtları) gibi gereksinimler nedeniyle çoğu zaman eğri tabanlı yörünge üretimi ve model-tabanlı takip yaklaşımlarıyla birlikte değerlendirilir [56].

Planlama ve kontrol arayüzü ve MPC tabanlı bütünleşik yaklaşımlar: Öte yandan, MPC yaklaşımı sistemin (çoğu kez kinematik bisiklet modeli gibi) bir araç modeline dayanarak gelecekteki hareketleri tahmin eder ve belirli bir zaman

ufkunda (prediction horizon) kontrol komutlarını optimize eder [63, 64]. MPC; direksiyon, hız, ivme ve yol sınırları gibi kısıtları doğrudan optimizasyona dahil edebilmesi nedeniyle, planlama katmanında *kısıtlı yörünge üretimi/optimizasyonu* fikrini destekleyen bir çerçeve sunduğu gibi, kontrol katmanında da *üretilen referansın izlenmesi* için yaygın biçimde kullanılmaktadır [10, 58]. Bu nedenle literatürde MPC kimi mimarilerde planlama ile kontrol arasında bir köprü (veya bütünleşik çözüm) olarak konumlanabilmektedir [10].

Rota planlama, davranış seçimi ve yörünge üretimi birlikte çalışarak otonom aracın hem uzun vadeli hedefe ulaşmasını hem de çevresel değişimlere anlık olarak uyum sağlamasını mümkün kılar. Üretilen yerel referans, nihai olarak kontrol modülüne iletilerek aracın direksiyon, hız ve fren gibi alt sistemlerine uygun komutlar üretilmesini sağlar.

1.2.4 Kontrol

Kontrol birimi, planlayıcıdan gelen yol/yörünge bilgisini aracın aktüatörlerine uygulanabilir komutlara (direksiyon, gaz ve fren) dönüştürmekle sorumludur. Bu süreçte amaç; takip hatasını azaltırken güvenlik, konfor ve fiziksel kısıtları (ör. maksimum direksiyon açısı, ivme sınırları, tekerlek tabanına bağlı dönme yarıçapı) sağlamaktır. Bu açıdan kontrol katmanı, planlama tarafından üretilen referansın *izlenmesi (tracking)* ve bozucu/ belirsizliklere karşı kararlı davranışın korunması görevlerini üstlenir [10]. Pratikte kontrol problemi çoğunlukla *lateral kontrol* (şerit/yol takibi için direksiyon) ve *longitudinal kontrol* (hız/mesafe takibi için gaz-fren) olarak ele alınır.

Kontrol yöntemleri genel olarak iki grupta özetlenebilir: (i) geometri/hataya dayalı klasik takip yaklaşımları (PID, Pure Pursuit, Stanley vb.) ve (ii) model tabanlı/optimizasyon tabanlı yöntemler (LQR, MPC/NMPC vb.). Paden ve ark. [10] çalışması, otonom araçlarda planlama ve kontrolün birlikte ele alındığı klasik mimarileri derleyerek kontrol tarafında özellikle geri besleme kontrolörleri ile optimizasyon tabanlı yöntemlerin kullanım alanlarını özetler. Daha güncel bir derleme olan Yurtsever ve ark. [65] ise güncel otonom sürüş sistemlerinde kontrol katmanını; algılama–planlama–kontrol hattı içerisinde konumlandırarak endüstride yaygın kullanılan yöntemler ile araştırma odaklı gelişmiş yöntemleri karşılaştırır.

Bu derlemelerde öne çıkan ortak değerlendirmeler şu şekilde özetlenebilir:

- **PID ve benzeri klasik geri besleme kontrolörleri:** Basit, hızlı ve uygulaması kolaydır; ancak ince ayar (tuning) ihtiyacı yüksektir ve

hız arttıkça/viraj keskinleştikçe model belirsizlikleri ile performans kaybı görülebilir [10, 65].

- **Pure Pursuit/Stanley gibi geometrik takip yöntemleri:** Düşük hesaplama maliyetiyle pratik çözümler sunar; buna karşın eğrilik değişimlerinde salınım, ileri bakış mesafesi gibi parametrelere hassasiyet ve dinamik kısıtların doğrudan ele alınamaması gibi sınırlara sahiptir [10, 65].
- **Model tabanlı yöntemler (LQR, MPC/NMPC):** Araç modeli üzerinden takip hatasını sistematik biçimde minimize eder. Özellikle **MPC**, planlamanın ürettiği referansı izlerken direksiyon/hız limitleri ve yol sınırları gibi kısıtları doğrudan optimizasyona dahil edebilmesi nedeniyle güvenlik ve konfor hedeflerini aynı çerçevede ele almakta güçlüdür; temel dezavantajı ise hesaplama yükü ve doğru modele duyulan ihtiyaçtır [10, 65].

Özetle, güncel eğilim; gerçek zamanlı çözücülerin gelişmesiyle birlikte kısıtları açıkça ele alabilen MPC/NMPC tabanlı kontrolün öneminin artması, buna karşılık klasik yöntemlerin düşük maliyet ve kolay entegrasyon avantajı nedeniyle hâlâ yaygın kullanılması şeklindedir [65]. Ayrıca kontrol tasarımında kullanılacak kinematik ve dinamik araç modelleri, hem takip performansı hem de kısıtların gerçekçiliği açısından kritik rol oynar [64].

1.2.5 Yörünge Tahmini (Trajectory Prediction)

Tahmin (prediction) modülü, algılama füzyon ve çoklu nesne takibi (MOT) katmanlarından elde edilen dinamik nesne izlerini (konum, hız, yönelim gibi durum değişkenleri ve bunlara ait belirsizlikler) kullanarak, çevredeki aktörlerin kısa vadede nasıl hareket edeceğine dair öngörü üretir. Bu öngörü; (i) planlayıcının çarpışma riski ve güvenli boşluk (gap) değerlendirmesi yapabilmesi, (ii) davranış seçimi (şerit koru/şerit değiştir/yol ver vb.) için etkileşimli senaryoların analiz edilmesi ve (iii) yörünge üretiminde dinamik engel kısıtlarının geleceğe taşınması açısından kritik bir girdidir. Literatürde tahmin problemi; tek-modlu (tek bir geleceğin kestirilmesi) yaklaşımlardan ziyade, sürücü davranışındaki belirsizlik ve çoklu niyet (intent) olasılığı nedeniyle çoğunlukla çok-modlu (multimodal) olasılıksal bir çıktı üretme çerçevesinde ele alınır [66, 67].

Tahmin yöntemleri genel olarak üç ana sınıfta özetlenebilir. **(i) Fizik tabanlı / hareket modeli tabanlı tahmin** yaklaşımında, aktörün hareketi kinematik/dinamik varsayımlar altında (ör. sabit hız, sabit ivme, sabit dönüş oranı) ileriye taşınır. Sabit hız (CV) modeliyle yapılan örnek bir yörünge tahmini ve Gauss gürültüsü altında

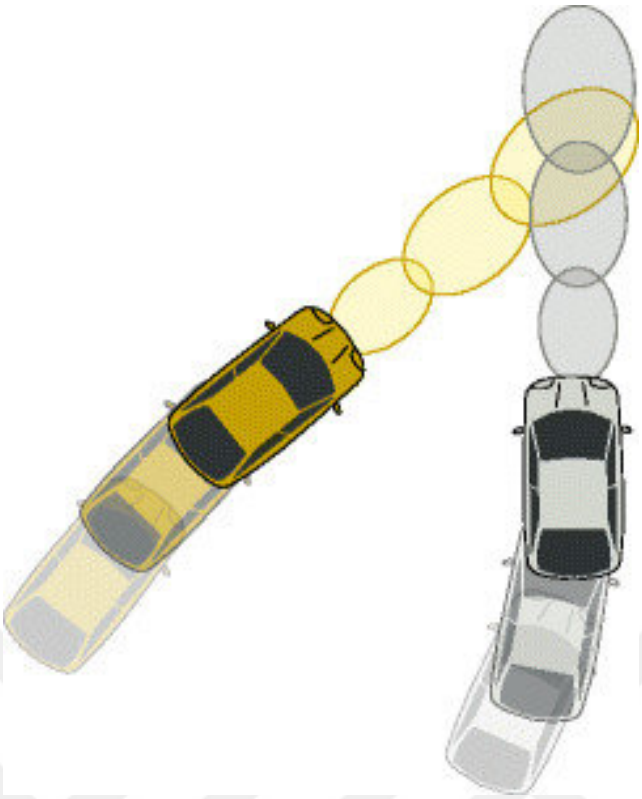
belirsizlik elipsleri Şekil 1.7 üzerinde gösterilmiştir. Bu yöntemler düşük hesaplama maliyeti ve yorumlanabilirlik açısından avantajlıdır ancak kavşak manevraları, agresif hız değişimleri ve etkileşimli sürüş gibi durumlarda model-uyumsuzluğu nedeniyle performans kaybı görülebilir [66]. **(ii) Manevra/niyet tabanlı tahmin** yaklaşımında, önce yüksek seviyeli bir manevra sınıfı (örn. şerit değiştir, düz git, sola dön) tahmin edilir; ardından her manevra hipotezi için ayrı bir hareket modeli veya öğrenilmiş yörünge dağılımı kullanılarak çok-modlu çıktı üretilir. Bu çerçevede manevra koşullu LSTM tabanlı yöntemlerin otoyol senaryolarında etkinliği raporlanmıştır [68]. **(iii) Veri güdümlü / öğrenme tabanlı tahmin** yöntemleri ise aktör geçmişini ve sahne bağlamını (komşu araçlar, harita/şerit geometrisi vb.) kullanarak geleceği öğrenir; özellikle etkileşimleri modellemek için RNN tabanlı yapılar, koşullu üretici modeller (örn. CVAE) ve grafik sinir ağları gibi yaklaşımlar yaygınlaşmıştır. Bu sınıfta, etkileşimli sahnelerde çok-modlu yörünge üretimine odaklanan DESIRE [69] ve otonom sürüş odaklı büyük ölçekli davranış tahmini hatlarında kullanılan çok-hipotezli yöntemler (örn. MultiPath, CoverNet) öne çıkan örneklerdir [70, 71].

Pratik otonom sürüş sistemlerinde tahmin modülü çoğunlukla **olasılıksal** bir çıktı üretir: gelecekteki konumların tek bir çizgi (deterministik) olarak verilmesi yerine, birden fazla aday yörünge ve bu yörüngelere ait olasılık/ağırlıkların raporlanması, planlama tarafında risk-duyarlı karar vermeyi kolaylaştırır [66, 70]. Ayrıca tahmin çıktısının güvenilirliği, doğrudan takip (MOT) katmanının sağladığı durum kestirimi ve belirsizlik bilgisine bağlıdır. Ölçümlerin belirsizleştiği veya ölçüm düşürme yaşanan durumlarda iz kovaryansının büyümesi, tahmin dağılımının da genişlemesine ve planlama kısıtlarının daha muhafazakâr hale gelmesine neden olabilir. Bu nedenle takip katmanında belirsizliğe dayanıklı kestirim (ör. model geçişlerini temsil edebilen IMM) yalnızca iz sürekliliği için değil, aynı zamanda tahmin ve planlama hattının tutarlılığı açısından da önem taşır.

1.2.6 Tezin Genel Organizasyonu

Bu tez, kavşak senaryolarında nesne takibinin hareket geçişlerinden ve sensör güvenilirliğindeki azalmadan nasıl etkilendiğini incelemekte ve bu etkileri azaltmak amacıyla Etkileşen Çoklu Model (Interacting Multiple Model – IMM) tabanlı bir takip yaklaşımı önermektedir. Bu bölümde çalışmanın motivasyonu verilmiş, otonom sürüşte planlama, kontrol, algılama ve nesne takibine ilişkin literatür özeti sunulmuş; ardından tezin amacı ve tez kapsamında hedeflenen katkılar tanımlanmıştır.

Bölüm 2'de nesne takibi probleminin temel bileşenleri detaylandırılmaktadır. Bu



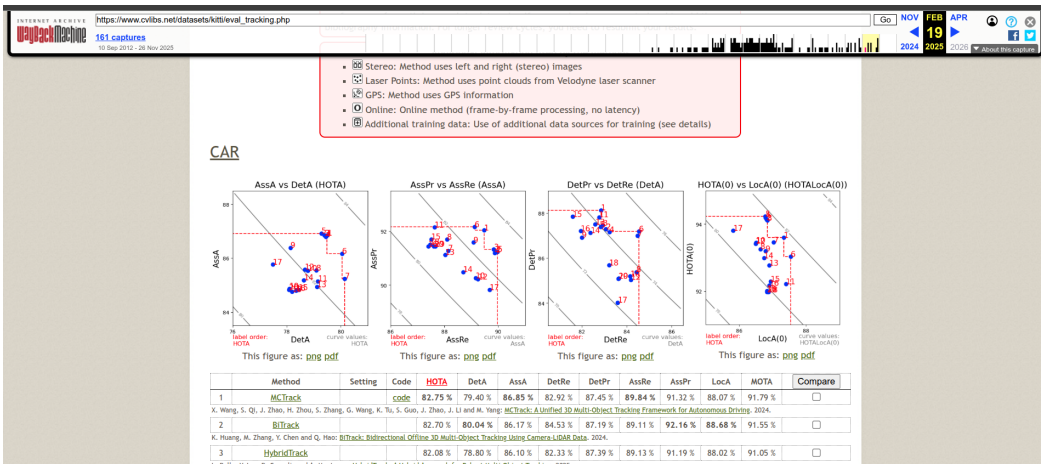
Şekil 1.7 Sabit hız (constant velocity) hareket modeli ve Gauss gürültüsü simülasyonu ile yörünge tahminine örnek. Elipsler, tahmin edilen konumların belirsizliğini (kovaryansını) göstermektedir [66] .

bölümde takip yönetimi, veri ilişkilendirme ve kapılama yöntemleri ile Kalman filtresi türevleri ve farklı hareket modelleri ele alınmakta; ayrıca manevralı senaryolarda model geçişlerini temsil edebilen IMM yaklaşımı açıklanmakta ve tez kapsamında kullanılan/önerilen takip sistemi bileşenleri sunulmaktadır. Bölüm 3'te deneysel değerlendirme kurulumu tanımlanmakta; kullanılan KITTI veri seti, değerlendirme metrikleri (HOTA ve alt metrikler), sensör bozunumu senaryoları (ölçüm belirsizliği artışı ve ölçüm düşürme) ve CA/CTRA/IMM yöntemlerinin karşılaştırmalı sonuçları raporlanmakta ve tartışılmaktadır. Son olarak Bölüm 4'te elde edilen bulgular özetlenmekte, çalışmanın sınırlılıkları belirtilmekte ve gelecekte yapılabilecek iyileştirmelere yönelik öneriler sunulmaktadır.

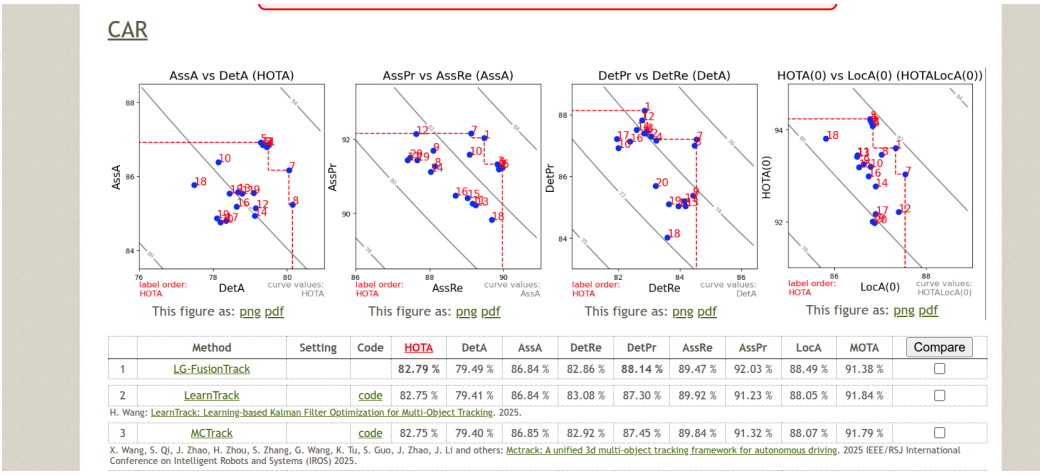
2
NESNE TAKİBİ ALGORİTMALARI

Bu bölümde, otonom sürüş senaryolarında çoklu nesne takibinde (MOT) kullanılan temel bileşenler ve yaygın algoritmalar özetlenmiştir. Bu kapsamda iz yönetimi (track management) mekanizmaları, veri ilişkilendirme (data association) yaklaşımları ve kapılama (gating) ile arama uzayının daraltılması ele alınmıştır. Eşleştirme tarafında atama-tabanlı yöntemler (Hungarian, GNN) ile belirsizlik altında kullanılabilen olasılıksal yaklaşımlar (JPDA, MHT) incelenmiştir. Ayrıca durum kestirimi için Kalman filtresi ve türevleri (KF/EKF/UKF) ile farklı hareket modelleri (CV/CA/CTRV/CTRA) takip bağlamında açıklanmıştır.

Modern takip sistemlerinde yaygın olarak kullanılan MCTrack çerçevesi, tespit temelli takip hattını modüler bir tasarımla ele alarak ölçüm işleme, kapılama/eşleştirme, filtreleme ve iz yönetimi bileşenlerini ayrıştırılmış biçimde sunmaktadır [72]. Bu tez kapsamında MCTrack'in referans uygulama çatısı olarak seçilmesinin üç temel nedeni vardır: (i) çalışmanın başlatıldığı dönemde raporlanan sonuçlarının Kitti Nesne Takip Liderlik tablosunda 1.sırada olması (liderlik tabloları zaman içinde yeni yöntemlerin eklenmesiyle güncellendiğinden sıralama bilgisi değişmiştir. Bu değişim Şekil 2.1'de görülmektedir.) (ii) modüler yapısı sayesinde CA/CTRA tabanlı IMM yaklaşımının takip modülüne doğrudan entegre edilebilmesi ve farklı algılayıcı (detector) çıktılarıyla aynı deney hattı üzerinde kolayca test yapılabilmesi, (iii) takip bileşeninin öğrenme-tabanlı bir yapı yerine model-tabanlı ve izlenebilir bir akış sunması; bu sayede algoritma davranışının analiz edilebilir, çalıştırılmasının ise nispeten hızlı çalışan ve tekrarlanabilir olmasıdır. Son olarak, tez kapsamında kullanılan takip yaklaşımının bileşenleri (ölçüm modeli, veri ilişkilendirme adımları, filtreleme ve iz yönetimi seçimleri) bu bölümün ilgili alt başlıklarında açık biçimde verilmiştir.



(a) Deneylerin yürütüldüğü dönemde liderlik tablosu ekran görüntüsü (MCTrack üst sırada).



(b) Güncel liderlik tablosu ekran görüntüsü

Şekil 2.1 MCTrack'in ilgili değerlendirme liderlik tablosundaki sıralamasının güncel durumu.

2.1 Takip Yönetimi

Çoklu nesne takibi sistemlerinde iz yönetimi (track management), takibin yapılması gereken ve sensörlerin görüş alanında olan alana yeni giren nesnelerin izlenmeye başlanması ve görünürlüğünü kaybeden ya da takip problemi için kritik olmayan nesnelerin takibinin sonlandırılması süreçlerini kapsar. Dinamik ortamlarda, takip edilecek nesnelerin sayısı önceden bilinmediğinden, takip algoritmalarının esnek ve kolay adapte edilebilecek şekilde olması gerekmektedir. Bu bağlamda, takip yönetimi yalnızca takip algoritmasının doğruluğu için değil, sistemin gerçek zamanlı çalışabilmesi ve sistemde kullanılan entegre bilgisayarların verimli kullanımı için de kritik önemdedir.

İz yönetimi, iki temel yapı taşından oluşur:

- **İz başlatma (track initiation)**: Yeni bir nesnenin takibe alınması
- **İz silme (track deletion)**: Artık sahnede olmadığı düşünülen ya da ilgi alanı dışında ki nesnenin takibinin sonlandırılması

Bu işlemlerin gerçekleştirilmesinde literatürde yaygın olarak iki yöntem kullanılmaktadır: **M/N yöntemi** [73] ve **iz skoru (track score)**[74].

2.1.1 M/N Yöntemi

M/N yöntemi (M–N logic ya da history-based logic), bir nesne takibinin başlatılması veya sonlandırılmasında son N çevrimdeki ölçüm eşleşme sıklığına göre karar veren, iz yaşam döngüsü yönetiminde yaygın bir yaklaşımdır [75, 76]. Son N çerçevede en az M adet ölçümün ilgili nesne ile eşleştiği durumda iz başlatılır; M'den az ölçümle eşleşme varsa iz sonlandırılır:

- **İz başlatma:** Son N çevrimin en az M adedi kadar eşleşme varsa
- **İz silme:** Son N çevrimin M adedinden daha az eşleşme varsa

Bu yöntem düşük hesaplama maliyeti ve kolay uygulanabilirliği nedeniyle gerçek zamanlı sistemlerde tercih edilir [76]. Ancak bazı dezavantajları da vardır:

- Ani manevralarda veya sensör görüş dışına çıkma durumlarında yanlış iz silme riski
- Algılama modülünün kalitesine (kaçırılan tespitler, yanlış pozitifler) yüksek bağımlılık ve M/N parametrelerinin dikkatli seçilmesi gerekliliği [75, 76]

2.1.2 İz Skoru (Track Score)

İz skoru, M/N mantığına benzer şekilde iz başlatma/silme kararları üretmekle birlikte; ek olarak sensörün istatistiksel performansını (ör. algılama olasılığı, yanlış alarm oranı) ve iz–ölçüm uyumunu (ör. artık/residual mesafe) hesaba katan bir skor fonksiyonu kullanır [75, 77]. Skor, başarılı güncellemelerde artarken kaçırılan tespitlerde azalır; skor belirli bir eşiği aşarsa aday iz doğrulanır, belirli bir eşik altına düşerse aday ya da doğrulanmış iz silinebilir [77].

İz skoru yöntemi, aşağıdaki avantajları sunar:

- Dinamik karar verme: Skor sürekli güncellendiği için kararlar daha esnek alınabilir
- Sahte izleri azaltma: Düşük skorlu, güvenilir olmayan takiplere son verilebilir
- İzlerin yaşam döngüsünün daha iyi kontrolü [77]

2.2 Veri Eşleştirme

Çoklu nesne takip sistemlerinin bir diğer kritik yönü ise veri ilişkilendirmesidir. Sensörler mükemmel olmadığından, bir ölçümün belirli bir iz (track) ile %100 doğrulukla eşleştirilmesi her zaman mümkün değildir. Ayrıca aynı sahneyi gözleyen farklı sensörlerden gelen ölçümlerin (kamera, LiDAR, radar vb.) bir arada değerlendirilmesi; ölçümlerin farklı gürültü karakteristiklerine, farklı gecikmelere ve farklı gözlenebilirlik bölgelerine sahip olması nedeniyle ilişkilendirme problemini daha da zorlaştırır [78].

Bu nedenle ilişkilendirme; (i) mevcut izlere ölçümlerin atanması (ölçümden ize ilişkilendirme) veya (ii) mevcut izlerin ölçümlere atanması (izden ölçüme ilişkilendirme) süreci olarak ele alınır. İlişkilendirme adımı gecikmeye duyarlıdır; zira bu adımda yaşanacak herhangi bir gecikme, takip/füzyon çıktısında gecikmeli tahminlere ve kimlik sürekliliğinin bozulmasına (identity switch) yol açabilir [78]. Bu nedenle pratik sistemlerde amaç, fiziksel/istatistiksel olarak olası olmayan eşleşmeleri erken aşamada eleyerek arama uzayını daraltmak (ör. kapılama/gating) ve ölçüm sayısı arttıkça hızla büyüyen hesaplama maliyetini yönetilebilir seviyede tutmaktır [78].

Literatürde veri ilişkilendirme yöntemleri genel olarak şu yaklaşımlar altında incelenmektedir: (i) *deterministik/atama-tabanlı* yöntemler (ör. en yakın komşu yaklaşımları ve Hungarian tabanlı tekil atama), (ii) *olasılıksal* yöntemler (ör. JPDA, MHT gibi belirsizlik altında çoklu hipotez veya olasılık dağıtımı yapan

yaklaşımlar), (iii) *global optimizasyon* temelli yöntemler (ör. ağ akışı / min-cost flow ile çoklu-zaman adımı boyunca global eniyileme) ve (iv) son yıllarda öne çıkan *öğrenme tabanlı* yaklaşımlar (ör. görünüş/özellik temelli metrik öğrenme, grafik/derin öğrenme ile ilişkilendirme) [79, 80]. Özellikle Emami ve ark. [79], çoklu nesne takibinde veri ilişkilendirme için makine öğrenmesi tabanlı yöntemleri kapsamlı biçimde derleyerek; el ile tasarlanmış maliyet fonksiyonlarının yerine veriden öğrenilen benzerlik/ilişkilendirme fonksiyonlarının kullanılabildiğini ve bu sayede yoğun sahnelerde kimlik sürekliliğinin iyileştirilebildiğini vurgulamaktadır.

2.2.1 MCTrack'te Veri Eşleştirme: Ro GDIoU ve İki Aşamalı Eşleştirme

MCTrack [72], veri ilişkilendirme (data association) tarafında yalnızca kutu örtüşmesi (IoU/GIoU) veya yalnızca merkez mesafesine dayalı maliyetlerin tüm veri setlerinde ve tüm sınıflarda kararlı performans vermediğini vurgulayarak, hem şekil benzerliğini hem de merkez mesafesini birlikte dikkate alan bir maliyet fonksiyonu önermektedir. Bu amaçla MCTrack, BEV (Bird's-Eye View) düzleminde tanımlanan Ro GDIoU maliyetini kullanır [72].

MCTrack'te Ro GDIoU, GIoU ve DIoU fikirlerini birleştirerek, kutular arası örtüşme/kuşatma ilişkisini ve merkez mesafesini aynı maliyete entegre eder. Makaledeki tanıma göre (BEV düzleminde) Ro GDIoU şu formda ifade edilir:

$$\text{Ro GDIoU} = \text{Ro IoU} - \omega_1 \cdot \frac{C - U}{C} - \omega_2 \cdot \frac{c^2}{d^2}, \tag{2.1}$$

burada $\omega_1 + \omega_2 = 2$ olup; U birleşim alanını, C minimum çevreleyen dikdörtgenin alanını, c iki kutunun merkezleri arasındaki Öklidyen mesafeyi ve d ise minimum çevreleyen dikdörtgenin köşegen uzunluğunu temsil eder [72]. $\omega_1 + \omega_2 = 2$ kısıtı, kuşatma farkı ve merkez uzaklığı terimleri arasındaki dengeyi ayarlarken bu iki ceza teriminin toplam etkisini sabit tutarak maliyetin ölçeğini kontrol etmeyi amaçlar. Bu yapı, yalnızca IoU türevlerinin ayırt edemediği bazı konfigürasyonlarda daha kararlı bir benzerlik/maliyet üretmeyi hedefler.

Ro GDIoU için hesaplamasında kullanılan terimler Şekil 2.2 üzerinde görülmektedir.

MCTrack, maliyet hesaplamasında sadece anlık benzerliği değil, önceki zaman adımındaki ilişkiyi de dikkate alan bir formülasyon kullanır. Makalede verilen biçime göre maliyet, mevcut zaman adımı ve bir önceki zaman adımı maliyetlerinin ağırlıklı toplamı şeklinde hesaplanır:

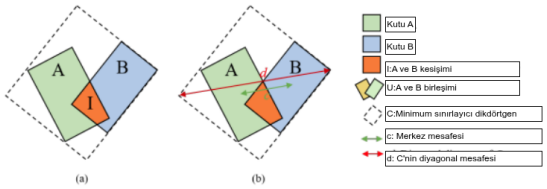
$$L_{\text{cost}} = \alpha \cdot C(D_\tau, T_\tau) + (1 - \alpha) \cdot C(D_{\tau-1}, T_{\tau-1}), \tag{2.2}$$

burada $C(\cdot)$ Ro GDIoU benzerliğidir ve $\alpha \in [0, 1]$ bir ağırlık parametresidir [72]. Bu yaklaşım, eşleştirmenin kısa vadeli zaman sürekliliğini desteklemeyi amaçlar.

MCTrack, algılanan derinlik bilgisindeki dalgalanmaların (özellikle kamera tabanlı algılamada) BEV üzerinde association hatalarına yol açabildiğini belirterek iki aşamalı bir eşleştirme stratejisi uygular:

- **1. aşama (BEV):** BEV düzleminde Ro GDIoU maliyeti ile maliyet matrisi kurulur ve Hungarian algoritması ile eşleştirme yapılır.
- **2. aşama (RV):** İlk aşamada eşleşmeyen iz ve tespitler Range View (RV) düzlemine projekte edilerek ikinci bir perspektiften eşleştirilir; makalede bu ikinci aşamada greedy benzeri bir eşleştirme stratejisinden bahsedilmektedir [72].

Bu iki-perspektifli eşleştirme, BEV düzleminde oluşan hataları azaltarak ID-switch ve fragmentation gibi sorunları düşürmeyi hedefler [72].



Şekil 2.2 Kutular arası benzerlik ölçütlerinde (IoU/GIoU/DIoU) örtüşme, çevreleyen alan ve merkez mesafesi kavramlarına örnek.

2.2.2 Kapılama (Gating)

Kalabalık bir trafik senaryosu çok sayıda nesne ve ölçüm içerebilir. Bu ölçüm ve nesnelerin hangilerinin ilintili olduğunu bulmak için tüm ölçüm ve nesnelerin aynı anda kullanılması çok büyük bir hesaplama gücü gerektirir.

Kapılama, hangi ölçümlerin hangi nesneler için anlamlı olabileceğini belirlemek için kullanılan bir ön eleme yöntemidir. Genellikle Mahalanobis mesafesi kullanılarak ölçüm ile tahmin arasındaki farkın anlamlılığı kontrol edilir. Euclidean yerine Mahalanobis mesafesi kullanımı, ölçüm varyanslarını da dikkate alarak daha doğru kapılmaların yapılmasını sağlar. Kapılama yaklaşımında kullanılan dikdörtgen ve eliptik kapı örnekleri Şekil 2.3 ile gösterilmiştir.

Kapılama Kullanılmadığı Durum

Kapılama yapılmadığında her ölçüm, her iz ile eşleştirilmeye çalışılır. Bu durumda:

- Nesne (track) sayısı: N
- Ölçüm (measurement) sayısı: M

Toplam ilişkilendirme hesaplama karmaşıklığı:

$$\mathcal{O}(N \cdot M) \tag{2.3}$$

Tüm olası eşleştirme kombinasyonlarının hesaba katılması durumunda, problem atama problemine dönüşür ve karmaşıklık:

$$\mathcal{O}(\max(N, M)!) \tag{2.4}$$

Örneğin $N = 10$, $M = 10$ için:

$$10! = 3{,}628{,}800 \text{ olası eşleştirme}$$

Kapılama ile

Kapılama uygulandığında, yalnızca belirli bir eşiği (örneğin Mahalanobis mesafesi) geçen ölçümler dikkate alınır. Her iz için ortalama k adet ölçüm kalır ($k \ll M$).

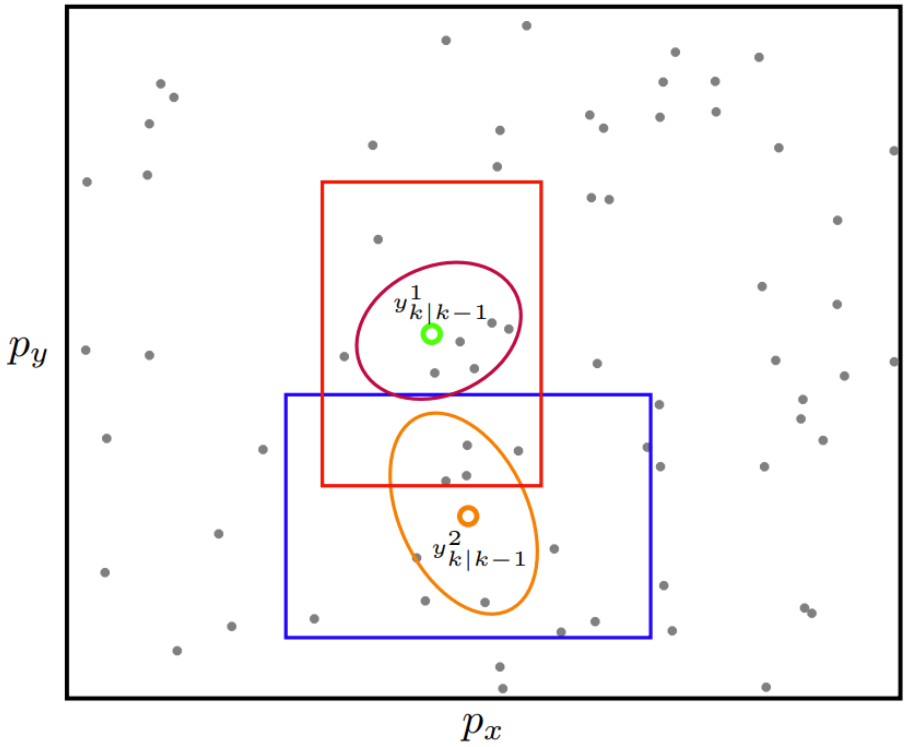
Bu durumda karmaşıklık:

$$\mathcal{O}(N \cdot k) \tag{2.5}$$

Bu yöntem, daha az sayıda ilişkilendirme ile daha verimli bir eşleme yapılmasını sağlar.

$$D^2 = (z - Hx)^T S^{-1} (z - Hx) \tag{2.6}$$

Bu mesafe, ölçüm gürültüsü de göz önüne alınarak tahminle ne kadar uyumlu olduğunu değerlendirir. Mesafe belli bir eşikten küçükse, o ölçüm "kapı içinde" sayılır ve daha ileri eşleştirme işlemlerine dahil edilir.



Şekil 2.3 Dikdörtgen ve Eliptik Kapıların (Gating) Gösterimi [81]

2.2.3 Hungarian Algoritması

Çoklu nesne takibinde ölçüm–iz eşleştirme adımı, genellikle bir **atama problemi** (assignment problem) olarak modellenir. Kapılama (gating) sonrasında elde edilen uygun iz–ölçüm çiftleri için bir maliyet matrisi $\mathbf{C} \in R^{N\times M}$ tanımlanır. Burada N iz sayısını, M ise ölçüm sayısını göstermektedir. Her bir C_{ij} terimi, i'inci iz ile j'inci ölçümün eşleştirilme maliyetini ifade eder (ör. Mahalanobis mesafesi, $1-\mathrm{IoU}$ veya benzeri).

Tekil (birebir) eşleştirme varsayımı altında amaç, toplam maliyeti minimize edecek eşleştirmeyi bulmaktır. İkili karar değişkeni $x_{ij} \in \{0,1\}$ ile,

$$\min_{\{x_{ij}\}} \sum_{i=1}^{N}\sum_{j=1}^{M} C_{ij}\, x_{ij} \tag{2.7}$$

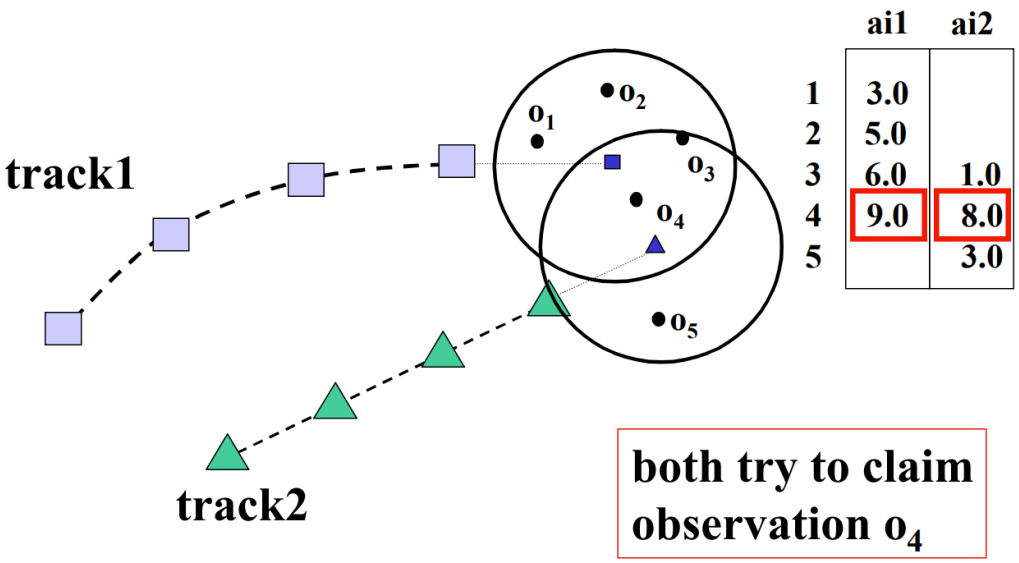
$$\text{koşuluyla: } \sum_{j=1}^{M} x_{ij} \le 1 \quad \forall i, \qquad \sum_{i=1}^{N} x_{ij} \le 1 \quad \forall j. \tag{2.8}$$

Bu problem, pratikte **Hungarian (Kuhn–Munkres) algoritması** ile eniyilenir. Hungarian algoritması, atama problemini polinom zamanda çözerek global optimum eşleştirme üretir ve tipik karmaşıklığı $\mathcal{O}(n^3)$ mertebesindedir (burada $n =$

$\max(N, M)$). $N \neq M$ durumunda maliyet matrisi, sahte (dummy) satır/sütunlar eklenerek kare matrise tamamlanabilir. Ayrıca, eşleşmelerin kabul edilmesi için pratikte bir eşikleme uygulanır; örneğin $C_{ij} > \tau$ olan eşleşmeler geçersiz sayılıp maliyet matrisinde elenebilir veya büyük bir ceza maliyeti atanabilir.

2.2.4 GNN - Global Nearest Neighbor

GNN algoritması, her iz için yalnızca bir ölçüm atar ve bu atama, tüm çözümlerin global olarak en uygun olanını seçmesiyle gerçekleştirilir. Bu problem, genellikle Hungarian algoritması kullanılarak çözülen bir atama problemi şeklindedir. GNN basit ve verimlidir fakat çoklu hedeflerin birbirine yakın olduğu senaryolarda yanlış atama yapma riski taşır. GNN yaklaşımında iki farklı iz için aynı ölçümün en yakın aday olarak seçilmesiyle oluşabilecek çakışma durumu Şekil 2.4'te örneklenmiştir.



Şekil 2.4 GNN algoritmasında iki farklı nesnenin aynı ölçüme talip olma senaryosu [82]

2.2.5 JPDA - Joint Probabilistic Data Association

JPDA algoritması, her hedefin birden fazla ölçümle ilişkili olabileceği varsayımına dayanır. Her bir ölçüm için tahmin edilen hedeflere ait olasılıklar hesaplanır. Bu olasılıklar, hedeflerin durum tahminine ağırlık katmak için kullanılır:

$$\hat{x} = \sum_i \beta_i \hat{x}_i \tag{2.9}$$

$$P = \sum_i \beta_i (P_i + (\hat{x}_i - \hat{x})(\hat{x}_i - \hat{x})^T) \tag{2.10}$$

GNN algoritmasının hesaplama maliyeti daha düşük olsa da JPDA algoritması özellikle çok sayıda hedefin yoğun olduğu ve gürültülü ortamlarda daha doğru eşleşme performansı sunar[83].

2.2.6 MHT - Multiple Hypothesis Tracking

Multiple Hypothesis Tracking (MHT), veri ilişkilendirme belirsizliğini tek bir en iyi eşleşmeye indirgemek yerine, zaman içinde birden fazla olası ilişkilendirme hipotezini birlikte taşıyan ve bu hipotezleri ölçümler geldikçe güncelleyen bir yaklaşımdır [80, 84]. Bu çerçevede her zaman adımında, ölçümlerin izlere farklı biçimlerde atanmasına karşılık gelen alternatif hipotezler üretilir ve hipotezler bir ağaç/orman yapısında temsil edilir. Hipotezlerin olasılık/puanları; ölçüm uyumu, yanlış alarm ve kaçırılmış tespit modelleri gibi istatistiksel varsayımlar üzerinden güncellenir.

MHT'nin temel avantajı, yoğun sahnelerde ve yüksek belirsizlik altında kimlik sürekliliğini daha sağlam biçimde koruyabilmesidir; çünkü erken aşamada hatalı tekil bir atamaya kilitlenmek yerine alternatif açıklamaları bir süre daha yaşatabilir [80]. Bununla birlikte hipotez sayısı kombinatoryal biçimde arttığından, pratik uygulamalarda hipotez budama (pruning), birleştirme (merging) ve N-scan gibi gecikmeli karar mekanizmaları ile hesaplama maliyeti kontrol altında tutulur [84]. Bu nedenle MHT, doğruluk–hesaplama maliyeti dengesi açısından güçlü olmakla birlikte gerçek zamanlı kısıtların sıkı olduğu sistemlerde dikkatli mühendislik gerektirir.

2.2.7 Öğrenme Tabanlı (Makine Öğrenmesi) Veri İlişkilendirme Yaklaşımları

Klasik veri ilişkilendirme yöntemlerinde maliyet fonksiyonu çoğunlukla el ile tasarlanır (ör. Mahalanobis mesafesi, IoU tabanlı benzerlik, hız/yön uyumu gibi). Son yıllarda ise bu maliyetin veya doğrudan ilişkilendirme kararının veriden öğrenildiği yaklaşımlar yaygınlaşmıştır [79, 80]. Emami ve ark. [79] veri ilişkilendirme için makine öğrenmesi tabanlı yöntemleri; (i) benzerlik/metrik öğrenme (appearance/özellik temelli), (ii) sınıflandırma/olasılık kestirimi ile eşleşme skoru tahmini, ve (iii) grafik tabanlı öğrenme ve ardışıl modeller gibi daha bütüncül çerçeveler altında incelemektedir.

Öğrenme tabanlı yaklaşımlar özellikle kamera verisinden elde edilen görünüş ipuçlarını (ör. re-identification benzeri özellikler) kullanarak, yalnızca geometrik yakınlığa dayalı atamalarda sık görülen kimlik karışmalarını azaltmayı hedefler

[79]. Buna karşın bu yöntemlerin performansı; eğitim verisinin temsil gücüne, alan kaymasına (domain shift), hesaplama bütçesine ve gerçek zamanlılık gereksinimlerine bağlı olarak değişebilir [80]. Bu tezde ise veri ilişkilendirme adımı, daha kontrol edilebilir ve tekrarlanabilir bir değerlendirme hattı sağlamak amacıyla kapılama ve atama-tabanlı klasik yöntemler üzerinden ele alınmıştır.

2.3 Kalman Filtresi

En yaygın tahmin filtrelerinden biri olan Kalman filtresi, aynı zamanda bir veri füzyon yöntemi olarak da kullanılır. Kalman filtresi, bir sistemin durumunu periyodik ölçümler temelinde sürekli olarak tahmin edebilen ideal, istatistiksel ve yinelemeli bir algoritmadır. Kalman filtresi, sistemin nasıl davranacağına ve yapılan gözlemlerin bu sistemle nasıl ilişkili olduğuna dair açık istatistiksel modeller kullanır [85].

Her bir filtre yinelemesi, tüm önceki ölçümleri yeniden işlemek zorunda kalmadan çalışır; çünkü tüm bu bilgiler mevcut durum ve korelasyon matrisi aracılığıyla zaten mevcuttur. Kalman filtresi kolayca diğer sensörleri de kullanacak şekilde genişletilebilir.

2.3.1 Kalman Filtresi (KF)

Doğrusal sistemler ve Gauss gürültüsü varsayımları altında en optimal tahmini verir.

- Sistem modeli: $x_k = Fx_{k-1} + Bu_{k-1} + w_{k-1}$
- Ölçüm modeli: $z_k = Hx_k + v_k$

Burada $w_k \sim \mathcal{N}(0, Q)$ ve $v_k \sim \mathcal{N}(0, R)$ gürültüleridir.

Tahmin ve güncelleme denklemleri:

$$\text{Tahmin:} \quad \hat{x}_{k|k-1} = F\hat{x}_{k-1|k-1} + Bu_{k-1} \tag{2.11}$$

$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q \tag{2.12}$$

$$\text{Güncelleme:} \quad K_k = P_{k|k-1}H^T(HP_{k|k-1}H^T + R)^{-1} \tag{2.13}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - H\hat{x}_{k|k-1}) \tag{2.14}$$

$$P_{k|k} = (I - K_kH)P_{k|k-1} \tag{2.15}$$

Kalman filtresine ilişkin sözde kod akışı Algoritma 2.1 ile verilmiştir.

Algorithm 2.1 Kalman Filtresi

Require: Önceki durum $\hat{x}_{k-1|k-1}$, kovaryans $P_{k-1|k-1}$, kontrol u_{k-1}, ölçüm z_k
Ensure: Güncellenmiş durum $\hat{x}_{k|k}$, kovaryans $P_{k|k}$

Tahmin (Prediction):
$\hat{x}_{k|k-1} \leftarrow F\,\hat{x}_{k-1|k-1} + B\,u_{k-1}$
$P_{k|k-1} \leftarrow F\,P_{k-1|k-1}\,F^T + Q$
Güncelleme (Update):
$\nu_k \leftarrow z_k - H\,\hat{x}_{k|k-1}$ ▷ inovasyon
$S_k \leftarrow H\,P_{k|k-1}\,H^T + R$
$K_k \leftarrow P_{k|k-1}\,H^T\,S_k^{-1}$
$\hat{x}_{k|k} \leftarrow \hat{x}_{k|k-1} + K_k\,\nu_k$
$P_{k|k} \leftarrow (I - K_k H)\,P_{k|k-1}$

2.3.2 Extended Kalman Filtresi (EKF)

EKF, doğrusal olmayan sistemlerde kullanılır. Sistem fonksiyonlarının doğrusal yaklaşımı (Taylor serisi ile) kullanılması temellidir.

- Sistem: $x_k = f(x_{k-1}, u_{k-1}) + w_{k-1}$
- ölçüm: $z_k = h(x_k) + v_k$

Jacobian matrisleri $F_k = \frac{\partial f}{\partial x}, H_k = \frac{\partial h}{\partial x}$ hesaplanarak Kalman adımları uygulanır:

$$\text{Tahmin:} \quad \hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}) \tag{2.16}$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q \tag{2.17}$$

$$\text{Güncelleme:} \quad K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R)^{-1} \tag{2.18}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - h(\hat{x}_{k|k-1})) \tag{2.19}$$

$$P_{k|k} = (I - K_k H_k) P_{k|k-1} \tag{2.20}$$

Genişletilmiş Kalman Filtresine ilişkin sözde kod akışı Algoritma 2.2 ile verilmiştir.

2.3.3 Unscented Kalman Filtresi (UKF)

UKF, EKF'nin lineerleştirme tabanlı yaklaşımının aksine doğrusal olmayan dönüşümü *Unscented Transform* ile yaklaşıklar. Temel fikir, durum dağılımını temsil eden belirli sayıda *sigma noktası* üretip bu noktaları doğrusal olmayan sistem/ölçüm fonksiyonlarından geçirerek ortalama ve kovaryansı daha doğru biçimde elde etmektir.

Algorithm 2.2 Genişletilmiş Kalman Filtresi (EKF)

Require: Önceki durum $\hat{x}_{k-1|k-1}$, kovaryans $P_{k-1|k-1}$, kontrol u_{k-1}, ölçüm z_k
Ensure: Güncellenmiş durum $\hat{x}_{k|k}$, kovaryans $P_{k|k}$

Tahmin (Prediction):
$\hat{x}_{k|k-1} \leftarrow f(\hat{x}_{k-1|k-1}, u_{k-1})$
$F_k \leftarrow \left.\dfrac{\partial f}{\partial x}\right|_{\hat{x}_{k-1|k-1},\, u_{k-1}}$ ▷ Jacobian
$P_{k|k-1} \leftarrow F_k\, P_{k-1|k-1}\, F_k^T + Q$
Güncelleme (Update):
$H_k \leftarrow \left.\dfrac{\partial h}{\partial x}\right|_{\hat{x}_{k|k-1}}$ ▷ Jacobian
$\nu_k \leftarrow z_k - h(\hat{x}_{k|k-1})$ ▷ inovasyon
$S_k \leftarrow H_k\, P_{k|k-1}\, H_k^T + R$
$K_k \leftarrow P_{k|k-1}\, H_k^T\, S_k^{-1}$
$\hat{x}_{k|k} \leftarrow \hat{x}_{k|k-1} + K_k\, \nu_k$
$P_{k|k} \leftarrow (I - K_k H_k)\, P_{k|k-1}$

Doğrusal olmayan ayrık zamanlı sistem için:

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1}, \tag{2.21}$$

$$z_k = h(x_k) + v_k, \tag{2.22}$$

burada $w_k \sim \mathcal{N}(0, Q)$ ve $v_k \sim \mathcal{N}(0, R)$.

UKF'de durum boyutu L olmak üzere ölçek parametreleri α, β, κ tanımlanır ve

$$\lambda = \alpha^2(L + \kappa) - L \tag{2.23}$$

şeklinde hesaplanır.

(1) Sigma noktalarının üretilmesi: $2L + 1$ adet sigma noktası

$$\chi_{0,k-1} = \hat{x}_{k-1|k-1}, \tag{2.24}$$

$$\chi_{i,k-1} = \hat{x}_{k-1|k-1} + \left[\sqrt{(L+\lambda)P_{k-1|k-1}}\right]_i, \quad i = 1, \ldots, L, \tag{2.25}$$

$$\chi_{i+L,k-1} = \hat{x}_{k-1|k-1} - \left[\sqrt{(L+\lambda)P_{k-1|k-1}}\right]_i, \quad i = 1, \ldots, L. \tag{2.26}$$

Burada $\sqrt{\cdot}$ ifadesi pratikte genellikle Cholesky ayrışımı ile elde edilir.

Ağırlıklar:

$$W_0^{(m)} = \frac{\lambda}{L+\lambda}, \tag{2.27}$$

$$W_0^{(c)} = \frac{\lambda}{L+\lambda} + (1-\alpha^2+\beta), \tag{2.28}$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2(L+\lambda)}, \quad i = 1, \ldots, 2L. \tag{2.29}$$

(2) Zaman güncellemesi (tahmin):

$$\chi_{i,k}^{-} = f(\chi_{i,k-1}, u_{k-1}), \quad i = 0, \ldots, 2L, \tag{2.30}$$

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \chi_{i,k}^{-}, \tag{2.31}$$

$$P_{k|k-1} = \sum_{i=0}^{2L} W_i^{(c)} \left(\chi_{i,k}^{-} - \hat{x}_{k|k-1}\right)\left(\chi_{i,k}^{-} - \hat{x}_{k|k-1}\right)^T + Q. \tag{2.32}$$

(3) Ölçüm güncellemesi: Önce ölçüm uzayına aktar:

$$\gamma_{i,k} = h(\chi_{i,k}^{-}), \quad i = 0, \ldots, 2L, \tag{2.33}$$

$$\hat{z}_{k|k-1} = \sum_{i=0}^{2L} W_i^{(m)} \gamma_{i,k}. \tag{2.34}$$

İnovasyon kovaryansı ve çapraz kovaryans:

$$S_k = \sum_{i=0}^{2L} W_i^{(c)} \left(\gamma_{i,k} - \hat{z}_{k|k-1}\right)\left(\gamma_{i,k} - \hat{z}_{k|k-1}\right)^T + R, \tag{2.35}$$

$$P_{xz,k} = \sum_{i=0}^{2L} W_i^{(c)} \left(\chi_{i,k}^{-} - \hat{x}_{k|k-1}\right)\left(\gamma_{i,k} - \hat{z}_{k|k-1}\right)^T. \tag{2.36}$$

Kalman kazancı ve güncelleme:

$$K_k = P_{xz,k} S_k^{-1}, \tag{2.37}$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\left(z_k - \hat{z}_{k|k-1}\right), \tag{2.38}$$

$$P_{k|k} = P_{k|k-1} - K_k S_k K_k^T. \tag{2.39}$$

Unscented Kalman Filtresine ilişkin sözde kod akışı Algoritma 2.3 ile verilmiştir.

Algorithm 2.3 Unscented Kalman Filtresi (UKF)

Require: Önceki durum $\hat{x}_{k-1|k-1}$, kovaryans $P_{k-1|k-1}$, kontrol u_{k-1}, ölçüm z_k, Q, R
Require: Parametreler α, β, κ
Ensure: Güncellenmiş durum $\hat{x}_{k|k}$, kovaryans $P_{k|k}$
$L \leftarrow$ durum boyutu
$\lambda \leftarrow \alpha^2(L + \kappa) - L$
Sigma noktaları:
$S \leftarrow \text{chol}((L + \lambda)\, P_{k-1|k-1})$
$\chi_0 \leftarrow \hat{x}_{k-1|k-1}$
for $i \leftarrow 1$ **to** L **do**
    $\chi_i \leftarrow \hat{x}_{k-1|k-1} + S_{(:,i)}$
    $\chi_{i+L} \leftarrow \hat{x}_{k-1|k-1} - S_{(:,i)}$
end for
Ağırlıklar:
$W_0^{(m)} \leftarrow \lambda/(L + \lambda)$
$W_0^{(c)} \leftarrow \lambda/(L + \lambda) + (1 - \alpha^2 + \beta)$
for $i \leftarrow 1$ **to** $2L$ **do**
    $W_i^{(m)} \leftarrow 1/(2(L + \lambda))$
    $W_i^{(c)} \leftarrow 1/(2(L + \lambda))$
end for
Tahmin (Time update):
for $i \leftarrow 0$ **to** $2L$ **do**
    $\chi_{i,k}^{-} \leftarrow f(\chi_i, u_{k-1})$
end for
$\hat{x}_{k|k-1} \leftarrow \sum_{i=0}^{2L} W_i^{(m)}\, \chi_{i,k}^{-}$
$P_{k|k-1} \leftarrow Q$
for $i \leftarrow 0$ **to** $2L$ **do**
    $P_{k|k-1} \leftarrow P_{k|k-1} + W_i^{(c)}(\chi_{i,k}^{-} - \hat{x}_{k|k-1})(\chi_{i,k}^{-} - \hat{x}_{k|k-1})^T$
end for
Ölçüm tahmini:
for $i \leftarrow 0$ **to** $2L$ **do**
    $\gamma_{i,k} \leftarrow h(\chi_{i,k}^{-})$
end for
$\hat{z}_{k|k-1} \leftarrow \sum_{i=0}^{2L} W_i^{(m)}\, \gamma_{i,k}$
$S_k \leftarrow R$
$P_{xz,k} \leftarrow 0$
for $i \leftarrow 0$ **to** $2L$ **do**
    $S_k \leftarrow S_k + W_i^{(c)}(\gamma_{i,k} - \hat{z}_{k|k-1})(\gamma_{i,k} - \hat{z}_{k|k-1})^T$
    $P_{xz,k} \leftarrow P_{xz,k} + W_i^{(c)}(\chi_{i,k}^{-} - \hat{x}_{k|k-1})(\gamma_{i,k} - \hat{z}_{k|k-1})^T$
end for
Güncelleme (Measurement update):
$K_k \leftarrow P_{xz,k}\, S_k^{-1}$
$\hat{x}_{k|k} \leftarrow \hat{x}_{k|k-1} + K_k(z_k - \hat{z}_{k|k-1})$
$P_{k|k} \leftarrow P_{k|k-1} - K_k S_k K_k^T$

2.3.4 Sabit Hız Hareket Modeli (CV)

Bu modelde nesnenin hızı sabittir ve yalnızca konumu değişir. Modelde ivme ve dönüş yoktur. Durum vektörü $\mathbf{x} = [\mathbf{x}, \mathbf{y}, \mathbf{v_x}, \mathbf{v_y}]^T$ şeklindedir. Sistem dinamiği şu şekilde tanımlanır:

$$\begin{aligned} x_{k+1} &= x_k + \Delta t \cdot v_{x,k} \\ y_{k+1} &= y_k + \Delta t \cdot v_{y,k} \\ v_{x,k+1} &= v_{x,k} \\ v_{y,k+1} &= v_{y,k} \end{aligned} \tag{2.40}$$

2.3.5 Sabit İvme Hareket Modeli (CA)

Bu modelde ivme sabittir. Durum vektörü $\mathbf{x} = [\mathbf{x}, \mathbf{y}, \mathbf{v_x}, \mathbf{v_y}, \mathbf{a_x}, \mathbf{a_y}]^T$ şeklindedir. Sistem denklemleri:

$$\begin{aligned} x_{k+1} &= x_k + v_{x,k}\Delta t + \frac{1}{2}a_{x,k}\Delta t^2 \\ y_{k+1} &= y_k + v_{y,k}\Delta t + \frac{1}{2}a_{y,k}\Delta t^2 \\ v_{x,k+1} &= v_{x,k} + a_{x,k}\Delta t \\ v_{y,k+1} &= v_{y,k} + a_{y,k}\Delta t \\ a_{x,k+1} &= a_{x,k} \\ a_{y,k+1} &= a_{y,k} \end{aligned} \tag{2.41}$$

2.3.6 Sabit Dönüş Hızı ve Sabit Hız Hareket Modeli (CTRV)

CTRV modelinde durum vektörü $\mathbf{x} = [\mathbf{x}, \mathbf{y}, \mathbf{v}, \boldsymbol{\psi}, \boldsymbol{\omega}]^T$ olup, nesnenin sabit hız ve dönüş hızı ile hareket ettiği varsayılır.:

$$\begin{aligned} x_{k+1} &= x_k + \frac{v_k}{\omega_k}\left(\sin(\psi_k + \omega_k\Delta t) - \sin(\psi_k)\right) \\ y_{k+1} &= y_k + \frac{v_k}{\omega_k}\left(-\cos(\psi_k + \omega_k\Delta t) + \cos(\psi_k)\right) \\ \psi_{k+1} &= \psi_k + \omega_k\Delta t \\ v_{k+1} &= v_k \\ \omega_{k+1} &= \omega_k \end{aligned} \tag{2.42}$$

Not:

$$\omega \approx 0$$

ise Taylor açılımı kullanılır:

$$\begin{aligned} x_{k+1} &= x_k + v_k \Delta t \cos(\psi_k) \\ y_{k+1} &= y_k + v_k \Delta t \sin(\psi_k) \end{aligned} \tag{2.43}$$

2.3.7 Sabit Dönüş Hızı ve Sabit İvme Hareket Modeli (CTRA)

CTRA modelinde hem sabit ivme hem de sabit dönüş oranı varsayılır. Durum vektörü şu şekildedir: $\mathbf{x} = [\mathbf{x}, \mathbf{y}, \boldsymbol{\psi}, \mathbf{v}, \boldsymbol{\omega}, \mathbf{a}]^T$:

$$\begin{aligned} x_{k+1} &= x_k + \frac{1}{\omega^2}\Big[(v\omega + a\omega\Delta t)\sin(\psi + \omega\Delta t) + a\cos(\psi + \omega\Delta t) \\ &\quad - v\omega\sin(\psi) - a\cos(\psi)\Big] \\ y_{k+1} &= y_k + \frac{1}{\omega^2}\Big[-(v\omega + a\omega\Delta t)\cos(\psi + \omega\Delta t) + a\sin(\psi + \omega\Delta t) \\ &\quad + v\omega\cos(\psi) - a\sin(\psi)\Big] \\ \psi_{k+1} &= \psi_k + \omega\Delta t \\ v_{k+1} &= v_k + a\Delta t \\ \omega_{k+1} &= \omega_k \\ a_{k+1} &= a_k \end{aligned} \tag{2.44}$$

Bu model, virajlardaki ivmeli hareketi en iyi şekilde yakalamak için tercih edilir.

2.4 Etkileşen Çoklu Modeller (IMM)

Gerçek dünyada bir nesne her zaman aynı hareket modeline uymayabilir. Bu nedenle birden fazla modelin bir arada çalıştığı sistemler kullanılır.

Interacting Multiple Model (IMM) algoritması, birden fazla hareket modeline dayanan filtrelerin paralel çalıştırıldığı ve aralarında olasılıklı geçişlerin tanımlandığı bir izleme (tracking) yöntemidir. Her model kendi filtrelemesini bağımsız yürütür ve daha sonra bu filtrelerin çıktıları birleştirilerek nihai tahmin elde edilir.

IMM algoritması ilk olarak Blom ve Bar-Shalom (1988) tarafından önerilmiştir [35]. Ani manevralar yapan hedeflerin takibinde hem etkin hem de hesaplama açısından verimli bir yaklaşımdır. Özellikle çoklu hedef takibi problemlerine diğer algoritmalarla kombine şekilde uygulanabilir.

IMM'nin temel felsefesi, her an birden fazla modelin aktif olabileceğini ve sistemin bu modeller arasında geçiş yapabileceğini kabul etmektir. Her bir filtreye bir olasılık (ağırlık) atanır ve bu olasılıklar zamanla ölçümlerle güncellenir.

2.4.1 Model Varsayımı ve Notasyon

Her bir model j için durum ve ölçüm denklemleri:

$$x_k^{(j)} = F^{(j)} x_{k-1}^{(j)} + w_{k-1}^{(j)} \tag{2.45}$$

$$z_k = H^{(j)} x_k^{(j)} + v_k^{(j)} \tag{2.46}$$

Burada kullanılan terimler aşağıdaki gibidir:

- k: zaman adımı.
- $j \in \{1, \ldots, M\}$: model indeksi, M: toplam model sayısı.
- $x_k^{(j)}$: j modeli altında k anındaki durum vektörü (örn. konum, hız vb.).
- z_k: k anındaki ölçüm vektörü, n: ölçüm vektörünün boyutu.
- $F^{(j)}$: j modeli için durum geçiş matrisi.
- $H^{(j)}$: j modeli için ölçüm matrisi.
- $w_{k-1}^{(j)}$: süreç (process) gürültüsü, $w^{(j)} \sim \mathcal{N}(0, Q^{(j)})$.
- $v_k^{(j)}$: ölçüm gürültüsü, $v^{(j)} \sim \mathcal{N}(0, R^{(j)})$.
- $Q^{(j)}$: süreç gürültüsü kovaryansı, $R^{(j)}$: ölçüm gürültüsü kovaryansı.

Geçiş Olasılıkları:

$$\Pi = \begin{bmatrix} \pi_{11} & \cdots & \pi_{1M} \\ \vdots & \ddots & \vdots \\ \pi_{M1} & \cdots & \pi_{MM} \end{bmatrix} \tag{2.47}$$

- π_{ij}: bir önceki adımda i modelinde iken bir sonraki adımda j modeline geçiş olasılığıdır.
- Model olasılıkları vektörü $\mu_k = [\mu_k^{(1)}, \ldots, \mu_k^{(M)}]^T$ olup, $\mu_k^{(j)}$ k anında j modelinin olasılığıdır.

2.4.2 IMM Algoritmasının Adımları (Denklemler ve Terimler)

1. **Modeller arası etkileşim (Data interaction):** Önce normalizasyon terimi hesaplanır:

$$c_j = \sum_{i=1}^{M} \pi_{ij} \mu_{k-1}^{(i)}. \tag{2.48}$$

Burada c_j, j modeline giden geçişlerin toplam olasılığını temsil eden normalizasyon terimidir.

Ardından karıştırma olasılıkları:

$$\mu_{k-1}^{(i|j)} = \frac{\pi_{ij} \mu_{k-1}^{(i)}}{c_j}. \tag{2.49}$$

Burada $\mu_{k-1}^{(i|j)}$, k anında j modeli koşulu altında önceki adımda i modelinden gelme olasılığıdır.

Her model j için karıştırılmış başlangıç durumu ve kovaryans:

$$\bar{x}_{0|k-1}^{(j)} = \sum_{i=1}^{M} \mu_{k-1}^{(i|j)} \hat{x}_{k-1|k-1}^{(i)} \tag{2.50}$$

$$\bar{P}_{0|k-1}^{(j)} = \sum_{i=1}^{M} \mu_{k-1}^{(i|j)} \left(P_{k-1|k-1}^{(i)} + (\hat{x}_{k-1|k-1}^{(i)} - \bar{x}_{0|k-1}^{(j)})(\hat{x}_{k-1|k-1}^{(i)} - \bar{x}_{0|k-1}^{(j)})^T \right). \tag{2.51}$$

Burada $\hat{x}_{k-1|k-1}^{(i)}$ ve $P_{k-1|k-1}^{(i)}$, i modeli için bir önceki adım sonundaki (posterior) durum tahmini ve kovaryanstır. $\bar{x}_{0|k-1}^{(j)}$ ve $\bar{P}_{0|k-1}^{(j)}$ ise j modeli filtresini başlatmak için farklı modellerden gelen bilgilerin olasılıklı olarak karıştırılmasıyla elde edilen başlangıç değerleridir.

2. **Model bazlı tahmin ve güncelleme:** Her model için KF/EKF/UKF adımları çalıştırılır ve $\hat{x}_{k|k-1}^{(j)}, P_{k|k-1}^{(j)}$ ile başlayıp $\hat{x}_{k|k}^{(j)}, P_{k|k}^{(j)}$ üretilir. Burada $\hat{x}_{k|k-1}^{(j)}$ ve $P_{k|k-1}^{(j)}$ öncül (prior) tahminleri; $\hat{x}_{k|k}^{(j)}$ ve $P_{k|k}^{(j)}$ ise ölçüm sonrası (posterior) tahminleri ifade eder.

3. **Olasılık (likelihood) ve model olasılığı güncelleme:**

$$\Lambda_k^{(j)} = \frac{1}{(2\pi)^{n/2} |S_k^{(j)}|^{1/2}} \exp\left(-\frac{1}{2} \nu_k^{(j)T} S_k^{(j)-1} \nu_k^{(j)} \right), \tag{2.52}$$

$$\mu_k^{(j)} = \frac{\Lambda_k^{(j)} \sum_{i=1}^{M} \pi_{ij} \mu_{k-1}^{(i)}}{\sum_{l=1}^{M} \Lambda_k^{(l)} \sum_{i=1}^{M} \pi_{il} \mu_{k-1}^{(i)}}. \tag{2.53}$$

Bu ifadelerde:

- $\nu_k^{(j)}$: j modeli için yenilik (innovation) vektörü (genellikle $\nu_k^{(j)} = z_k - H^{(j)}\hat{x}_{k|k-1}^{(j)}$).
- $S_k^{(j)}$: yenilik kovaryansı (genellikle $S_k^{(j)} = H^{(j)}P_{k|k-1}^{(j)}H^{(j)T} + R^{(j)}$).
- $\Lambda_k^{(j)}$: j modeli altında ölçümün olasılığı (likelihood), yani modelin ölçümü ne kadar iyi açıkladığını belirtir.
- $\mu_k^{(j)}$: güncellenmiş model olasılığı (posterior model probability).

4. **Çıktı birleştirme (Output Combination):**

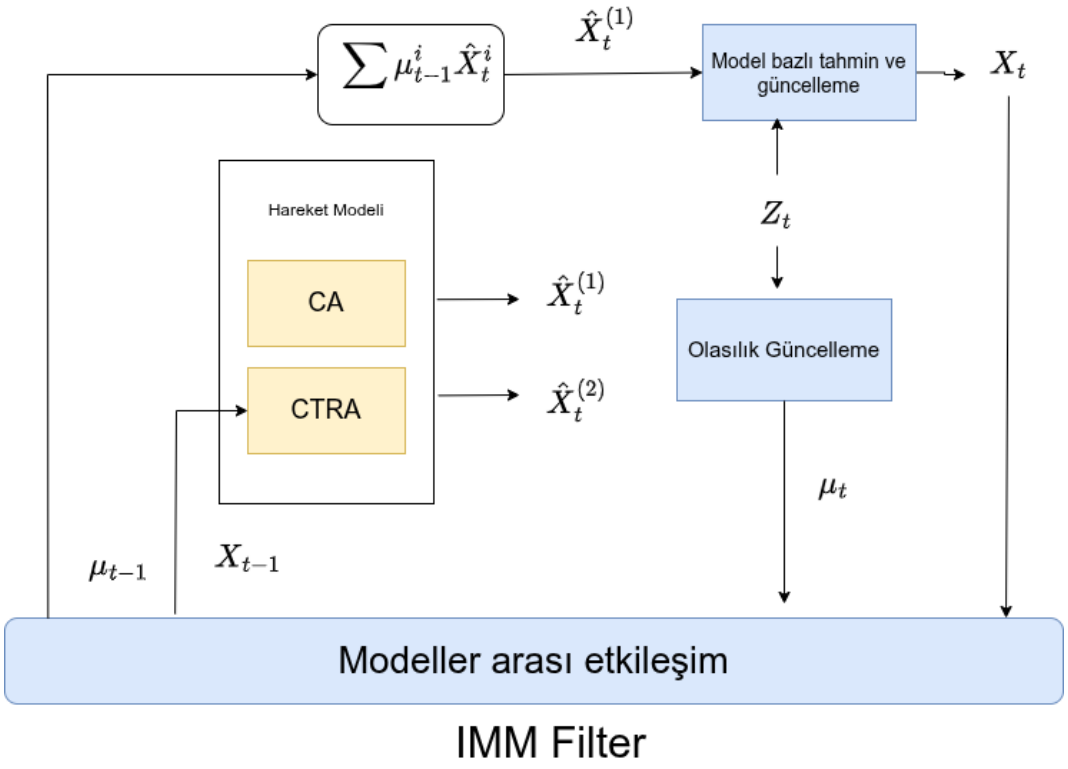
$$\hat{x}_k = \sum_{j=1}^{M} \mu_k^{(j)} \hat{x}_{k|k}^{(j)} \tag{2.54}$$

$$P_k = \sum_{j=1}^{M} \mu_k^{(j)} \left(P_{k|k}^{(j)} + (\hat{x}_{k|k}^{(j)} - \hat{x}_k)(\hat{x}_{k|k}^{(j)} - \hat{x}_k)^T \right). \tag{2.55}$$

Burada $\hat{x}_k$ ve P_k, tüm modellerin olasılıklı birleşimiyle elde edilen nihai durum tahmini ve kovaryanstır.

Bu adımlar her zaman diliminde tekrarlanarak hedefin en uygun durumu modellenir ve özellikle kavşak senaryoları gibi ani manevralarda takip performansı artar.

Bu tez kapsamında kullanılan IMM algoritmasının diagramı Şekil 2.5 üzerinde görülmektedir.



Şekil 2.5 Uygulanan IMM algoritması diagramı

2.4.3 IMM Algoritmasının Pseudocode Gösterimi

Etkileşen Çoklu Modeller (IMM) Filtresine ilişkin sözde kod akışı Algoritma 2.4 ile verilmiştir.

Algorithm 2.4 Interacting Multiple Model (IMM) Algoritması [35]

Require: Model sayısı M, geçiş olasılık matrisi $\Pi = [\pi_{ij}]$
Require: Başlangıç model olasılıkları $\mu_0^{(j)}$
Require: Her model için başlangıç durum/kovaryans $(\hat{x}_{0|0}^{(j)}, P_{0|0}^{(j)})$
Require: Her model için filtre modeli (KF/EKF/UKF): $(F^{(j)}, H^{(j)}, Q^{(j)}, R^{(j)})$

for $k = 1, 2, \ldots$ **do**
  (1) Etkileşim / Karıştırma
  for $j = 1$ to M **do**
    $c_j \leftarrow \sum_{i=1}^{M} \pi_{ij} \mu_{k-1}^{(i)}$
    for $i = 1$ to M **do**
      $\mu_{k-1}^{(i|j)} \leftarrow \dfrac{\pi_{ij} \mu_{k-1}^{(i)}}{c_j}$
    end for
    $\bar{x}_{0|k-1}^{(j)} \leftarrow \sum_{i=1}^{M} \mu_{k-1}^{(i|j)} \hat{x}_{k-1|k-1}^{(i)}$
    $\bar{P}_{0|k-1}^{(j)} \leftarrow \sum_{i=1}^{M} \mu_{k-1}^{(i|j)} \Big(P_{k-1|k-1}^{(i)} + (\hat{x}_{k-1|k-1}^{(i)} - \bar{x}_{0|k-1}^{(j)})(\hat{x}_{k-1|k-1}^{(i)} - \bar{x}_{0|k-1}^{(j)})^T \Big)$
  end for
  (2) Model-bazlı filtreleme (Prediction/Update)
  for $j = 1$ to M **do**
    $(\hat{x}_{k|k}^{(j)}, P_{k|k}^{(j)}, \nu_k^{(j)}, S_k^{(j)}) \leftarrow \text{FilterUpdate}\big(\bar{x}_{0|k-1}^{(j)}, \bar{P}_{0|k-1}^{(j)}, z_k\big)$
  end for
  (3) Model olasılık güncelleme
  for $j = 1$ to M **do**
    $\Lambda_k^{(j)} \leftarrow \dfrac{1}{(2\pi)^{n/2} |S_k^{(j)}|^{1/2}} \exp\Big(-\dfrac{1}{2} \nu_k^{(j)T} S_k^{(j)-1} \nu_k^{(j)} \Big)$
  end for
  for $j = 1$ to M **do**
    $\mu_k^{(j)} \leftarrow \dfrac{\Lambda_k^{(j)} \sum_{i=1}^{M} \pi_{ij} \mu_{k-1}^{(i)}}{\sum_{l=1}^{M} \Lambda_k^{(l)} \sum_{i=1}^{M} \pi_{il} \mu_{k-1}^{(i)}}$
  end for
  (4) Çıktı birleştirme (Combination)
  $\hat{x}_k \leftarrow \sum_{j=1}^{M} \mu_k^{(j)} \hat{x}_{k|k}^{(j)}$
  $P_k \leftarrow \sum_{j=1}^{M} \mu_k^{(j)} \Big(P_{k|k}^{(j)} + (\hat{x}_{k|k}^{(j)} - \hat{x}_k)(\hat{x}_{k|k}^{(j)} - \hat{x}_k)^T \Big)$
end for

2.4.4 Sağlamlık İyileştirmeleri

IMM algoritmasının standart formülasyonunda bu bölümdeki ek düzenlemeler doğrudan yer almamaktadır. Ancak uygulama sırasında gözlenen sayısal kararsızlıklar ve model dönüşümlerinden kaynaklı sapmaların takip performansını olumsuz etkileyebilmesi nedeniyle, pratikte kararlılığı artırmak amacıyla aşağıdaki üç sağlamlık iyileştirmesi sonradan sisteme entegre edilmiştir.

- **Düşük Hızda Sapma Açısı (yaw) Sabitleme:** Düşük hızlarda v_x ve v_y üzerinden hesaplanan sapma açısı (yaw) sayısal olarak kararsız hale gelebilmektedir. Bu nedenle hız belirli bir eşik değerin altına düştüğünde sapma açısı (yaw) değeri son güvenilir değerde sabitlenmiştir.
- **Kovaryans Dönüşümünde Sayısal Kararlılık:** CA $\rightarrow$ CTRA dönüşümünde hızın sıfıra yakın olduğu bölgelerde Jacobian tabanlı kovaryans dönüşümü kovaryansın aşırı büyümesine neden olabilmektedir. Bu etkiyi sınırlamak için düşük hız bölgesinde mevcut durumu koruma eğilimi yüksek (bounded) bir dönüşüm Jacobian'ı kullanılmıştır.
- **Sapma Açısı (yaw) için Açı-Uyumlu Etkileşim (Angle-Aware Interactıon):** sapma açısı (yaw) değişkeni periyodik olduğundan, doğrusal ağırlıklı ortalama kullanımı $\pm\pi$ sınırında süreksizlik üretebilmektedir. Bu sebeple, sapma açısı (yaw) karıştırma adımında dairesel ortalama yaklaşımı kullanılmış ve açı farkları uygun aralıkta sarılarak (wrap) hesaplanmıştır.

2.5 Önerilen Nesne Takip Sistemi

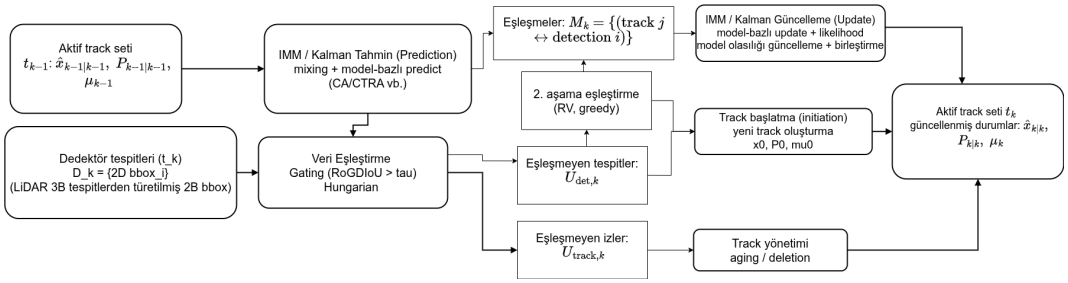
Bu tezde kullanılan çoklu nesne takip sistemi, tespit temelli takip paradigması ile ele alınmıştır. Bu kapsamda, her zaman adımında algılayıcı/detektör çıktıları ölçüm olarak alınmış, veri ilişkilendirme ile mevcut izlere atanmış ve ardından durum kestirimi, Kalman türevi filtreler ve IMM çatısı altında gerçekleştirilmiştir. Tezdeki nesne takip çalışması, hazır algılayıcıların araçlara daha çok yoğunlaşması nedeniyle yalnızca araçlar için gerçekleştirilmiştir.Bu tezde takip problemi **yer düzlemi üzerinde iki boyutlu (2D)** olarak ele alınmış; takip filtreleri nesnelerin x–y düzlemindeki hareketini ve kimlik sürekliliğini izlemektedir.

Uygulama altyapısı olarak MCTrack kod tabanı [72] temel alınmış; veri okuma/yazma, değerlendirme arayüzü bu çatı üzerinden kullanılmıştır.

2.5.1 MCTrack Çatısı ve Bu Tezdeki Kullanımı

MCTrack [72], otonom sürüş senaryoları için 3B çoklu nesne takibi (3D MOT) problemini tespit temelli takip paradigması altında ele alan, bileşenleri ayrıştırılmış ve karşılaştırılabilir hale getiren bir çerçevedir(framework). Temel amaç; farklı takip akışı tercihlerinin (hareket modeli, kapılama, eşleştirme maliyeti, atama yöntemi, iz yönetimi vb.) aynı değerlendirme arayüzünde sistematik biçimde değerlendirilebilmesini ve tekrarlanabilir deneyler yürütülebilmesini sağlamaktır.

Bu tezde MCTrack, bir *referans takip hattı* ve deney altyapısı olarak kullanılmış; özellikle veri okuma/yazma, ölçüm arayüzü, değerlendirme metriklerinin hesaplanması ve görselleştirme/analiz çıktılarının üretilmesi süreçleri MCTrack çatısı üzerinden yürütülmüştür. Öte yandan, bu çalışmanın özgün katkısı takip filtresi seviyesinde olup; tek bir hareket modeline bağımlılığı azaltmak amacıyla CA ve CTRA modelleri IMM çatısı altında etkileşimli biçimde birleştirilmiş ve kavşak manevraları ile sensör bozunumu koşulları altında davranışı karşılaştırmalı olarak analiz edilmiştir.



Şekil 2.6 Önerilen Nesne Takip Sisteminin Gösterimi

2.5.2 Kullanılan Nesne Takip Yapısı

Sistem, her zaman adımı k için aşağıdaki adımlarla çalışır Şekil 2.6 ile gösterilmiştir.

1. Girdi tespitlerinin alınması ve ön işlenmesi
2. Mevcut izlerin model bazlı tahmini (CA ve CTRA)
3. Kapılama ile olası eşleşmelerin azaltılması
4. Veri ilişkilendirme (maliyet matrisi + Hungarian/GNN)
5. Eşleşen izlerde filtre güncellemesi ve IMM olasılık güncellemesi

6. Eşleşmeyen izlerde yalnızca hareket modeli ile tahmin ve iz yönetimi
7. Yeni ölçümlerle iz başlatma ve düşük kaliteli izleri silme
8. İz çıktılarının KITTI formatında raporlanması ve HOTA ile değerlendirme

Bu çalışmada takip için, MCTrack [72] çerçevesinin modüler yapısı üzerinde kurgulanmıştır. Bu sayede algılama tespitleri sabit tutularak, takip performansındaki değişimlerin ağırlıklı olarak filtre/hareket modeli tercihlerinden kaynaklanması hedeflenmiştir. Özetle:

- **Sabit tutulan kısımlar:** ölçüm okuma/formatlama, temel değerlendirme hattı, kapılama ve atama tabanlı eşleştirme akışı (gating + atama).
- **Çalışmaya özgü odak:** durum kestirimi katmanında CA/CTRA/IMM karşılaştırması ve sensör bozunumu (ölçüm belirsizliği artırımı, ölçüm düşürme) altında dayanıklılık analizi.

2.5.3 Nesne Tespiti (VirConv ve CasA)

Bu çalışmada takip modülüne ölçüm girdisi olarak kullanılan nesne tespitleri, KITTI sahnelerinde yer alan LiDAR nokta bulutu (ve değerlendirme amacıyla kamera görüntüsü) üzerinden çalışan hazır 3B dedektör çıktılarından elde edilmiştir. Bu bağlamda, tezde temel olarak VirConv [86] dedektörünün ürettiği tespitler kullanılmıştır. VirConv, LiDAR nokta bulutu üzerinde 3B nesne tespiti için sanal seyrek evrişim (virtual sparse convolution) temsiline dayanan ve özellikle seyrek nokta bulutunda verimli çıkarım yapabilen bir yaklaşımdır [86]. Literatürde LiDAR tabanlı 3B tespit için dikkat (attention) mekanizmalarını kademeli (cascade) biçimde kullanan CasA [87] gibi yöntemler de bulunmaktadır. Bu tez kapsamında algılama tarafında yeni bir dedektör önerilmemekte; takip performansını etkileyen unsurların (hareket modeli seçimi, IMM kullanımı ve sensör bozunumu senaryoları) daha net incelenebilmesi için dedektör çıktıları tüm deneylerde sabit tutularak takip modülüne ölçüm olarak beslenmektedir.

Sensör kullanımı ve işleme hattı (VirConv–CasA karşılaştırması): VirConv [86] ve CasA [87] her ne kadar aynı veri setinde (KITTI) 3B kutu üreten dedektörler olsa da, LiDAR ve kamera bilgisini kullanma biçimleri belirgin şekilde farklıdır. CasA, *yalnızca LiDAR nokta bulutu* ile çalışan LiDAR-tabanlı bir iki-aşamalı (two-stage) algılama hattı üzerine kuruludur: ham LiDAR noktaları önce voxel temsiline dönüştürülür, ardından 3B seyrek evrişim tabanlı bir omurga ağ ile 3B özellik hacimleri çıkarılır ve bu özellikler yükseklik ekseni boyunca

sıkıştırılarak BEV (bird's-eye view) gösterimi elde edilir. Bu BEV gösterim üzerinde çalışan 2B algılama başlığı (RPN), aday 3B kutuları üretir. Üretilen adaylar daha sonra CasA'nın kademeli iyileştirme ağı (CRN) içerisinde birden fazla aşamada rafine edilir; burada önerilen CAM (cascade attention module) ile farklı aşamalardaki RoI özellikleri birleştirilerek özellikle uzak/seyrek örneklerde proposal kalitesinin daha kararlı biçimde iyileştirilmesi hedeflenir. Bu yapı gereği CasA'nın algılama tarafında kamera verisi veya görüntüden türetilmiş ek bir geometrik girdi kullanılmaz; kamera yalnızca tez kapsamında değerlendirme (ör. 2B IoU/HOTA hesabı) için dolaylı bir rol üstlenmektedir.

Buna karşılık VirConv [86], LiDAR geometrisini kamera görüntüsünden elde edilen tamamlayıcı ipuçları ile zenginleştiren *multimodal* bir yaklaşımdır. Temel fikir, RGB görüntü üzerinde derinlik tamamlama ile üretilen "sanal noktalar" (virtual points) sayesinde özellikle uzak mesafelerde seyrek kalan LiDAR örneklemesinin yoğunlaştırılması ve böylece nesne geometrisinin daha iyi temsil edilmesidir. VirConv hattında LiDAR noktaları ile görüntüden türetilen sanal noktalar aynı koordinat sisteminde ele alınarak voksel/seyrek-evrişim tabanlı bir omurgaya beslenir. Ancak sanal noktaların çok yoğun ve gürültüye açık olabilmesi nedeniyle VirConv, yakın mesafede gereksiz yoğunluğu azaltmaya yönelik StVD (stochastic voxel discard) bileşeniyle hesaplama yükünü düşürür; ayrıca NRConv (noise-resistant submanifold convolution) ile LiDAR uzayındaki voksel özelliklerini kamera düzlemine projeksiyon üzerinden de değerlendirerek (LiDAR–kamera kalibrasyonu ve projeksiyon ilişkisi kullanılarak) gürültünün baskılanmasını amaçlar. Özetle, CasA yalnızca LiDAR verisini kullanan ve aday 3B kutuları ürettikten sonra bunları çok aşamada kademeli olarak iyileştiren bir yaklaşımla, LiDAR'ın seyrek ve düzensiz örneklemesinin olumsuz etkilerini azaltmayı hedeflerken, VirConv LiDAR geometrisini kamera görüntüsünden türetilen sanal noktalar ile yoğunlaştırıp, bu multimodal girdinin getirdiği yoğunluk ve gürültü sorunlarını mimari bileşenlerle dengelemeye odaklanır. Tezde takip modülüne verilen ölçüm, her iki yöntemde de nihayetinde LiDAR referans çerçevesinde ifade edilen 3B kutulardır; kamera verisi CasA tarafında dedektör girdisi değildir, VirConv tarafında ise tespit çıktısı hazırlanırken multimodal füzyonun bir parçası olarak kullanılabilmektedir.

Dedektör çıktısı (3B tespit formatı): VirConv/CasA gibi 3B dedektörler tipik olarak her tespit için sınıf etiketi ve güven skoru ile birlikte 3B kutu parametreleri üretir. Bu parametreler genel olarak kutu merkez konumu, boyutları ve yönelimini içerir:

- **Sınıf bilgisi**: (tezde yalnızca *Car* sınıfı değerlendirmeye dahil edilmiştir),

- **Güven skoru (confidence)**: düşük güvenli tespitlerin elenmesi için eşikleme uygulanmıştır,
- **3B kutu merkezi**: (x, y, z) ,
- **3B kutu boyutları**: (l, w, h) ,
- **Yönelim (yaw)**: ψ .

Takip modülüne aktarılan ölçümler: Bu tezde takip filtresine ölçüm olarak (i) nesnenin yer düzlemi üzerindeki konumu ve (ii) değerlendirme amaçlı 2B sınırlayıcı kutu bilgileri aktarılmıştır. Buna göre VirConv çıktılarından aşağıdaki alanlar kullanılmaktadır:

- **Konum ölçümü**: takip filtresine ölçüm olarak nesnenin BEV düzlemindeki (x, y) konumu verilmiştir,
- **2B sınırlayıcı kutu**: (left, top, right, bottom) bilgileri; HOTA metriğinin IoU tabanlı hesaplanmasında kullanılmıştır.

Girdi/çıktı örneği (şematik): Aşağıda bir tespitin takip modülüne aktarılmadan önceki tipik alanları şematik olarak gösterilmektedir (değerler veri seti/çıkış formatına göre değişebilmektedir):

Nesne Tespiti Bir nesneye ait tespit bilgileri.

sınıf: Tespit edilen nesnenin sınıfı (örn: "Car", "Pedestrian").

skor: Tespitin güven skoru (0.0 ile 1.0 arası bir ondalıklı sayı).

bbox3d: Nesnenin 3 boyutlu sınırlayıcı kutu bilgileri.

x, y, z: Nesnenin merkez noktasının koordinatları.

l, w, h: Nesnenin uzunluğu, genişliği ve yüksekliği.

yaw: Nesnenin yönelim açısı.

bbox2d: Nesnenin 2 boyutlu sınırlayıcı kutu bilgileri.

left, top: 2D kutunun sol üst köşesinin koordinatları.

right, bottom: 2D kutunun sağ alt köşesinin koordinatları.

Bu yaklaşım ile 3B dedektör tabanlı dedektör kaynaklı algılama hataları (yanlış pozitif/negatifler ve gürültülü konum ölçümleri) takip sistemine gerçekçi biçimde yansıtılırken, her bir deney setinde dedektör (VirConv veya CasA) sabit tutularak

ve aynı eşikleme/ön-işleme adımları uygulanarak algılama katmanının etkisi kontrol altında tutulmuştur. Böylece CA/CTRA/IMM tabanlı takip yöntemleri, aynı algılama girdileri altında adil biçimde karşılaştırılabilmiş; ayrıca dedektör değişiminin takip performansına etkisi ayrı bir faktör olarak değerlendirilebilir hale gelmiştir.

2.5.4 Durum Vektörleri ve Ölçüm Modeli

Bu tezde iki farklı hareket modeli kullanılmaktadır:

- CA modeli durum vektörü: $x_k^{CA} = [x, y, v_x, v_y, a_x, a_y]^T$
- CTRA modeli durum vektörü: $x_k^{CTRA} = [x, y, \psi, v, \omega, a]^T$

Ölçüm modeli her iki mod için konum ölçümü kabul edilerek tanımlanmıştır: $z_k = Hx_k + v_k$. Bu kapsamda H matrisi, durum vektöründen (x, y) bileşenlerini seçmektedir. Ölçüm gürültüsü $v_k \sim \mathcal{N}(0, R)$ olarak modellenmiştir.

2.5.5 Veri İlişkilendirme ve Kapılama

İlişkilendirme adımında her iz için ölçüm-tahmin uyumu bir maliyet fonksiyonu ile hesaplanmıştır. Hesaplama maliyetini azaltmak amacıyla kapılama uygulanmış ve kapı dışı kalan eşleşmeler maliyet matrisinden elenmiştir. Nihai atama işlemi Hungarian algoritması ile çözülmüştür.

2.5.6 İz Yönetimi

Yeni bir iz başlatma ve iz silme işlemleri için M/N kuralı benzeri skor tabanlı bir yöntem kullanılmıştır. Bu tezde kullanılan parametreler MCTrack açık kaynak kod deposundaki varsayılan ayarlardan doğrudan alınmış ve deneyler boyunca sabit tutulmuştur.

2.5.7 IMM Konfigürasyonu ve Mod Olasılıkları

IMM yapısında iki mod kullanılmıştır: CA ve CTRA. Mod geçiş olasılıkları matrisi ve başlangıç mod olasılıkları, sistemin fiziksel dinamiklerini en iyi yansıtacak şekilde sabit seçilmiştir. Özellikle, mod geçiş olasılıkları matrisi (Π) ve başlangıç mod olasılıkları (μ_0) aşağıdaki gibi belirlenmiştir:

$$\Pi = \begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}, \quad \mu_0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}.$$

Bu seçimde, hedefin mevcut hareket modunu (CA veya CTRA) yüksek bir olasılıkla sürdüreceği (0.95) ve modlar arasında geçişin görece nadir olduğu (0.05) varsayılmıştır. Gerçek dünya senaryolarında, bir hedefin sürekli ivmeli hareketten dönüşlü harekete veya tam tersine ani ve sık geçişler yapması genellikle düşük bir ihtimaldir. Bu nedenle, modlar arası geçiş olasılığının düşük tutulması, filtrenin kararlılığını ve kestirim performansını artırmaktadır.

Daha yüksek geçiş olasılıkları (örneğin 0.15 veya 0.2 gibi) içeren alternatif matris değerleri ile yapılan denemelerde, filtrenin modlar arasında gereksiz yere sık geçişler yaptığı ve bu durumun kestirim sonuçlarını olumsuz etkilediği gözlemlenmiştir. Özellikle, CA modundan CTRA moduna geçişin gerçekçi olmayan bir şekilde yüksek olasılıkla gerçekleştiği senaryolarda, filtre performansı düşmüştür. Bu gözlemler, seçilen 0.95 ve 0.05 değerlerinin, hedefin dinamiklerini en doğru ve istikrarlı şekilde modellemek için uygun parametreler olduğunu doğrulamaktadır.

3
TEST ORTAMI, DEĞERLENDİRME METRİKLERİ VE SONUÇLAR

Bu bölüm, tez kapsamında geliştirilen takip yaklaşımının deneysel olarak nasıl değerlendirildiğini bütüncül biçimde sunmak amacıyla hazırlanmıştır. Amaç; CA, CTRA ve bu iki modelin etkileşimli biçimde birleştirildiği IMM yaklaşımının, kavşak gibi manevra geçişlerinin yoğun olduğu sahnelerde ve sensör güvenilirliğinin azaldığı koşullarda (ölçüm belirsizliği artışı ve ölçüm düşürme senaryoları) performansını karşılaştırmalı olarak ortaya koymaktır. Bu kapsamda, kullanılan veri seti ve etiket yapısı, değerlendirme metrikleri, bozunum senaryoları ve deney protokolü açık biçimde tanımlanmıştır.

Bölümün devamında sırasıyla: (i) KITTI veri seti ve takip benchmark yapısı, (ii) HOTA metriği ve alt bileşenleri üzerinden performansın nasıl ölçüldüğü, (iii) sensör bozunumu senaryolarının (ölçüm kovaryansı ölçekleme ve deterministik ölçüm düşürme) nasıl kurulduğu, (iv) seçilen kavşak senaryoları üzerinde CA/CTRA/IMM yöntemlerinin nicel sonuçları ve örnek görselleştirmeleri sunulmaktadır. Ayrıca, yöntemlerin pratik kullanılabilirliğini desteklemek amacıyla çalışma zamanı ölçümleri de raporlanmaktadır.

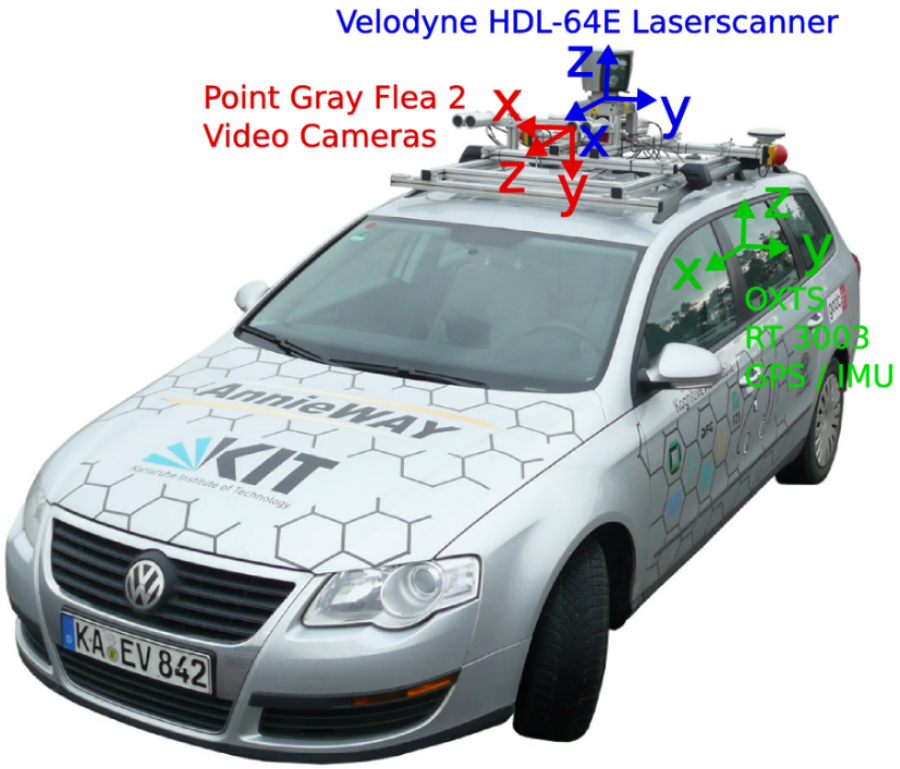
3.1 KITTI Test Veri Seti

KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) test veri seti, otonom sürüş sistemlerinin algılama, takip ve lokalizasyon gibi görevlerde performanslarını ölçmek amacıyla yaygın olarak kullanılan açık kaynaklı bir veri setidir. Veriler, aşağıdaki sensörlerden eşzamanlı olarak elde edilmiştir:

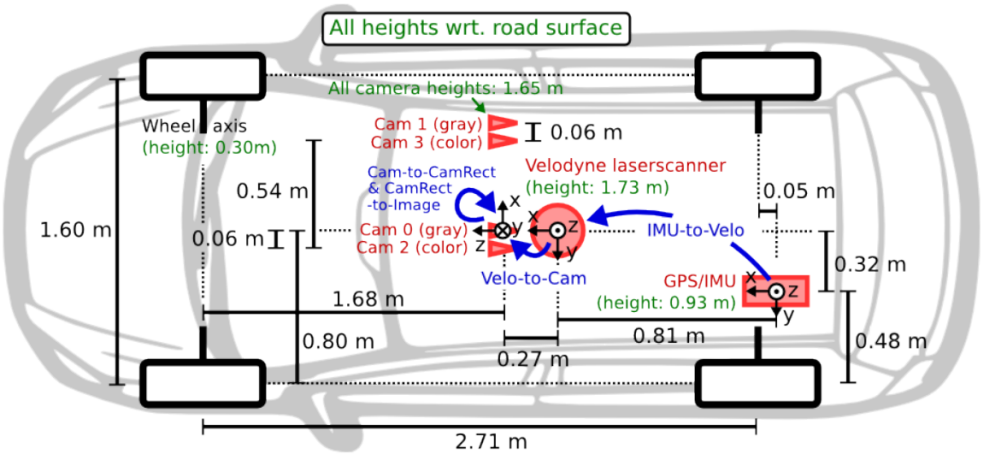
- 2 adet renkli kamera (stereo görünüm için)
- Lidar (velodyne 3D nokta bulutu verisi)
- Küresel Konumlama Sistemi (Global Positioning System - GPS)

(konumlandırma verisi)

- Ataletsel Ölçüm Birimi (Inertial Measuremet Unit - IMU) (yönelim verisi)
- Araç CAN verileri (araç hızı, direksiyon açısı vb.)



Şekil 3.1 KITTI veri toplama aracı. [7]



Şekil 3.2 Sensörlerin araç üzerindeki yerleşimi. [7]

Toplanan veriler, hem çevresel nesne algılama hem de çoklu nesne takibi (MOT) için görevleri için etiketlenmiş ve değerlendirmeye uygun biçimde sunulmuştur

[7]. KITTI veri setinin veri toplama sürecinde kullanılan sensör donanımı ve araç platformu Şekil 3.1'te ,sensörlerin araç üzerindeki fiziksel yerleşim diyagrami ve konfigürasyonu ise Şekil 3.2 ile gösterilmiştir.

3.1.1 Etiket Formatı ve Alanlar

Takip benchmark'ında her nesneye ilişkin etiketler aşağıdaki alanları içerir:

- **frame:** nesnenin göründüğü çevrim numarası.
- **track id:** Aynı nesneye ait benzersiz takip kimliği.
- **type:** nesne tipi ('Car', 'Pedestrian', 'Cyclist' vb.).
- **truncated:** Görüntü sınırında kesilmiş nesne oranı (0–1 arası).
- **occluded:** Görünürlük durumu: 0 (tam görünür), 1 (kısmen), 2 (büyük ölçüde), 3 (bilinmiyor).
- **alpha:** nesnenin kameraya göre görüş açısı (azimuth), $[-\pi, \pi]$ aralığında.
- **bbox:** 2D nesne çerçevisini gösteren dikdörtgen sınırlayıcı kutu (left, top, right, bottom).
- **dimensions:** nesnenin 3D boyutları (yükseklik, genişlik, uzunluk) [metre].
- **location:** Kamera koordinatlarında (x, y, z) konumu [metre].
- **rotation_y:** Y ekseni etrafında döndürülme açısı $[-\pi, \pi]$.
- **score:** (Yalnızca sonuçlarda): Tespit güven skoru.

Bu yapılandırma, hem 2D hem de 3D izleme algoritmalarının detaylı analizine imkân tanır.

KITTI takip benchmark'ı, 2D ve 3D nesne takibi performansını değerlendirmek için ayrılmış sahneler içerir. Bu sahneler genellikle şu şekilde sınıflandırılır:

- **Araç (Car)**
- **Yaya (Pedestrian)**
- **Bisikletli (Cyclist)**

Veri seti,bu tez için aşağıdaki amaçlar doğrultusunda kullanılmıştır:

- **Filtre Performansının Karşılaştırılması:** CA ve CTRA gibi farklı hareket modellerine dayanan Kalman filtresi tabanlı algoritmaların başarıları değerlendirilmiştir.
- **IMM Uygulaması:** Kavşak gibi hareket desenlerinin sık değiştiği durumlarda IMM yaklaşımı test edilmiştir.
- **Sensör Belirsizliği Simülasyonu:** Ölçüm düşürme (missing measurement) ve ölçüm belirsizliği artırımı senaryoları altında takip doğruluğu analiz edilmiştir.

3.2 Karşılaştırma Metrikleri: HOTA ve Alt Metrikler

Çoklu nesne takibi (MOT) alanında sistemlerin doğruluğunu değerlendirmek için çeşitli metrikler kullanılmaktadır. Bu metrikler bazı daha temel metriklerin bir araya getirilmesi ile elde edilmektedir.Kompleks metrikleri elde etmek için kullanılan bazı ham metrikler aşağıda gösterilmiştir.

Metrik	Açıklama
FP	False Positives: Yanlış pozitif tespit sayısı
FN	False Negatives: Gerçek nesnelerin tespit edilememesi
IDSW	Identity Switches: Takip kimliğinin yanlış atanması
GT	Ground Truth: Gerçek (etiketlenmiş) nesne sayısı
MT	Mostly Tracked: %80'den fazla süreyle doğru takip edilen nesne sayısı
PT	Partially Tracked: %20–80 arası takip edilen nesne sayısı
ML	Mostly Lost: %20'den az süreyle takip edilen nesne sayısı
Frag	Fragments: Takibin kesildiği ve yeniden başladığı durumların sayısı

Tablo 3.1 Çoklu nesne takibi değerlendirme metrikleri

Bu basit metrikler kullanılarak bazı metrik aileleri tanımlanmıştır.Bunlardan en yaygın olarak kullanılan ikisi HOTA (Higher Order Tracking Accuracy)[88] ve CLEAR MOT [89] metrik ailesidir.

Clear MOT metrik ailesi 2008 yılında ortaya atılmış ve o tarihten bu yana uzun yıllar kullanılmıştır.Tespitin doğruluğu ve bir nesne atanan takip kimlik numarasının sürekliliği gibi daha klasik metriklerden oluşur.Detay analizler için Clear MOT metrikleri kullanılarak nesne takip performansı ölçülebilir.

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FN}_t + \text{FP}_t + \text{IDSW}_t)}{\sum_t \text{GT}_t} \tag{3.1}$$

- FN_t: Zaman adımı t'deki yanlış negatif (False Negative) sayısı
- FP_t: Zaman adımı t'deki yanlış pozitif (False Positive) sayısı
- IDSW_t: Zaman adımı t'deki kimlik değiştirme (ID switch) sayısı
- GT_t: Zaman adımı t'deki gerçek (ground truth) nesne sayısı

$$\text{MOTP} = \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \tag{3.2}$$

- d_t^i: Zaman adımı t'de doğru eşleşen i'inci tahmin ile gerçek etiketlenmiş (ground truth) nesnesi arasındaki mesafe (örneğin, IoU farkı ya da Öklidyen mesafe)
- c_t: Zaman adımı t'deki doğru eşleşme (correct match) sayısı

Fakat günümüzde daha karmaşık takipçilerin performans karşılaştırılması için daha bütüncül bir yaklaşıma ihtiyaç duyulmuştur.Bu ihtiyaca cevap olarak HOTA metrik ailesi 2020 yılında akademide kullanılmaya başlanmıştır.Günümüzde de en geçerli ve bütüncül özelliği nedeni ile takip algoritmalarını ölçmek için HOTA metriği kullanılmaktadır.

HOTA metriği 3 ana alt metriğin IoU skorlarının birleişimi olarak düşünülebilir. HOTA metriğinin temelini oluşturan üç alt görev olan **Lokalizasyon (localization)**, **algılama (detection)** ve **eşleştirme (association)** için IoU skorunun nasıl hesaplandığı ve daha sonra bu skorlarının sonuçta elde edile metriği nasıl ortaya koyduğu açıklanmaktadır. Bu çalışmada HOTA metrikleri 2B sınırlayıcı kutu IoU üzerinden hesaplanmıştır.

3.2.1 Lokalizasyon (Localization)

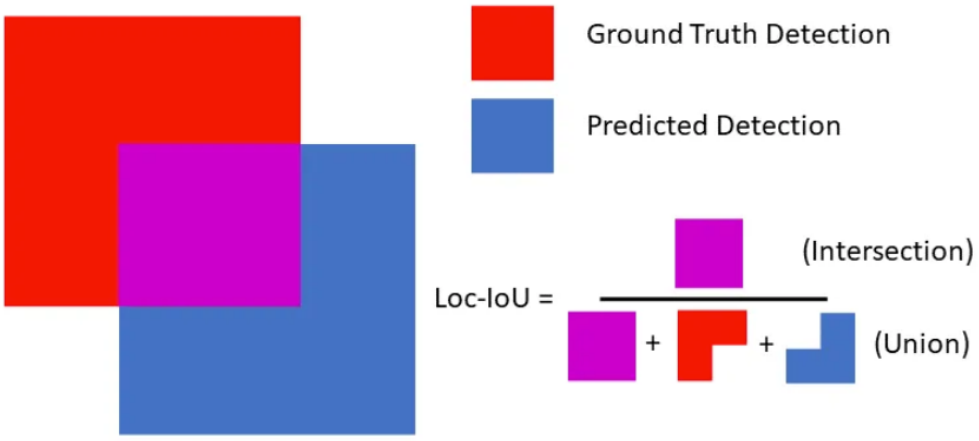
Lokalizasyon, bir tahmin edilen tespit ile bir gerçek etiketli (ground-truth) tespiti arasındaki uzamsal hizalanmayı ölçer. Bu amaçla sıklıkla kullanılan metrik **Localization IoU (Loc-IoU)**'dur. Loc-IoU, iki tespit arasındaki örtüşme (kesişim) alanının, toplam kapsadıkları alanın (birleşim) oranı olarak tanımlanır. Lokalizasyon doğruluğunu ölçmek için kullanılan IoU kavramı Şekil 3.3'da şematik olarak gösterilmiştir.

Şekil 3.3'da **kırmızı** kutu gerçek etiket (ground-truth) sınırlayıcı kutusunu, **mavi** kutu ise algoritmanın ürettiği tahmini (predicted) sınırlayıcı kutuyu temsil etmektedir. İki kutunun ortak örtüştüğü **mor** bölge kesişim alanını göstermektedir. Birleşim alanı ise kırmızı ve mavi alanların toplamının, kesişim bölgesinin bir kez sayılacak şekilde birleştirilmesiyle elde edilir. Buna göre, gerçek kutu B_{gt} ve tahmin kutusu B_{pred} olmak üzere Loc-IoU şu şekilde hesaplanır:

$$\text{LocIoU} = \frac{|B_{\text{gt}} \cap B_{\text{pred}}|}{|B_{\text{gt}} \cup B_{\text{pred}}|}. \tag{3.3}$$

Bu kavram, nesne çerçevesini gösteren sınırlayıcı kutular için olduğu kadar segmentasyon maskeleri için de geçerlidir. Loc-IoU değeri arttıkça, tahmin edilen ve gerçek tespitler daha iyi hizalanmış olur ve lokalizasyon doğruluğu artar. Örneğin, iki kutu hiç örtüşmüyorsa Loc-IoU değeri 0 olurken, tamamen çakışmaları durumunda 1 değerini alır.

Tüm veri seti üzerindeki eşleşmiş tahmin, gerçek tespit çiftlerinin Loc-IoU değerlerinin ortalaması alınarak, genel lokalizasyon doğruluğu (**Localization Accuracy – LocA**) elde edilir.



Şekil 3.3 IoU (kesişim/birleşim) kavramının şematik gösterimi [90].

3.2.2 Algılama (Detection)

Algılama, tahmin edilen tespitler kümesi ile gerçek etiketli (ground-truth) tespitler kümesi arasındaki genel hizalanmayı ölçer. Bu kapsamda **Detection IoU (Det-IoU)** metriği kullanılır. İlk olarak, hangi tahminlerin hangi gerçek tespitlerle örtüştüğünü belirlemek için bir Loc-IoU eşiği (α, örneğin 0.5) tanımlanır. Bu eşik değerin üzerinde kalan çiftler örtüşüyor kabul edilir.

Birden fazla tespitin birbirleriyle örtüşmesini önlemek için **Hungarian algoritması** kullanılarak birebir eşleşme yapılır. Eşleşen çiftler **True Positive (TP)**, eşleşmeyen tahminler **False Positive (FP)** ve eşleşmeyen gerçek etiketli (ground-truth) tespitler **False Negative (FN)** olarak tanımlanır.

Det-IoU şu şekilde hesaplanır:

$$\text{Det-IoU} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FP}| + |\text{FN}|} \tag{3.4}$$

Bu formül, Loc-IoU'nun tekil tespitler arasındaki yapıdan kümeler düzeyine genelleştirilmiş halidir. Det-IoU metriği, aynı zamanda Jaccard İndeksi olarak da bilinir.

Tüm veri seti için bu oran hesaplanarak genel algılama doğruluğu (**Detection Accuracy - DetA**) elde edilir.

3.2.3 Eşleştirme (Association)

Eşleştirme, bir takip algoritmasının tespitleri zamana bağlı şekilde doğru kimliklerle (ID) bağlayıp bağlayamadığını ölçer. Bunun için, önce belirli bir lokalizasyon eşiği altında (ör. Loc-IoU $> \alpha$) Hungarian gibi bir yöntemle tahmin tespitleri ile gerçek etiket (ground-truth) tespitleri arasında birebir eşleşmeler (TP) elde edilir; ardından bu eşleşmiş TP çiftlerinin ait olduğu izlerin (track) zaman içindeki tutarlılığı değerlendirilir [90].

Eşleştirme değerlendirmesinde kullanılan iz-kesişimi (track intersection) mantığı Şekil 3.4'de görselleştirilmiştir. Şekilde:

- **Tahmin edilen tespitler ve yörüngeler (Predicted detections and trajectories)** gri tonlu noktalar ve kesik çizgiler ile,
- **Gerçek etiketli tespitler ve yörüngeler (Ground-truth detections and trajectories)** mavi tonlu noktalar ve kesik çizgiler ile gösterilmektedir.
- **İlgilenilen doğru pozitif eşleşme, c (TP of interest (c))** kırmızı kutu ile işaretlenmiş, belirli bir zaman anındaki eşleşmiş tahmin–gerçek tespit çiftini ifade eder.

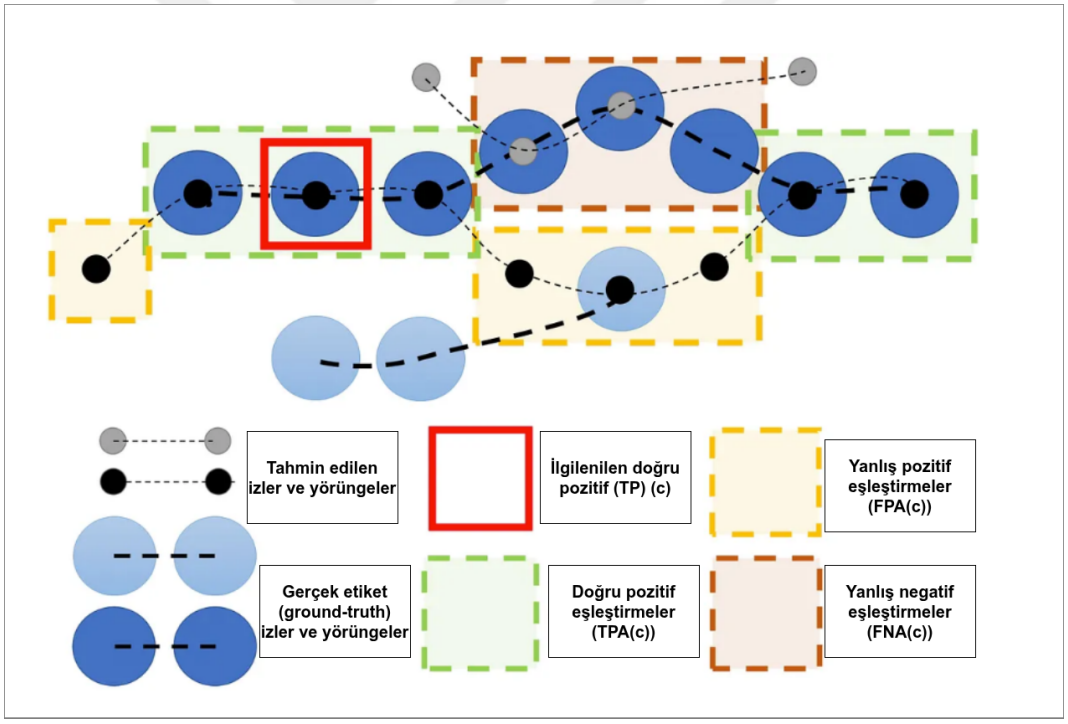
Bu kırmızı kutu ile seçilen TP çifti için, ilgili tahmin izi ile ilgili gerçek iz arasındaki "kesişim", iki iz boyunca birbiriyle eşleşen tüm TP'lerin sayısı ile ölçülür ve bunlar

True Positive Associations (TPA(c)) olarak adlandırılır (Şekil 3.4'de **yeşil** kesik çizgili bölge). Tahmin izinde yer alıp bu gerçek izle eşleşmeyen (başka bir gerçek izle eşleşen veya hiç eşleşmeyen) tespitler **False Positive Associations (FPA(c))** olarak tanımlanır (Şekil 3.4'de **sarı** kesik çizgili bölge). Benzer şekilde, gerçek izde yer alıp bu tahmin izle eşleşmeyen tespitler ise **False Negative Associations (FNA(c))**'dır (Şekil 3.4'de **kahverengi** kesik çizgili bölge).

Bu tanımlara göre Association IoU (**Ass-IoU**) şu şekilde hesaplanır:

$$\text{Ass-IoU} = \frac{|\text{TPA}|}{|\text{TPA}| + |\text{FPA}| + |\text{FNA}|}. \tag{3.5}$$

Bu metrik, her eşleşmiş tespit çifti (TP) için izler arası eşleştirmenin (ID sürekliliğinin) ne kadar başarılı olduğunu ölçer. Tüm veri setindeki tüm TP çiftleri için ortalama alınarak genel eşleştirme doğruluğu (**Association Accuracy – AssA**) hesaplanır [90].



Şekil 3.4 Eşleştirme (Association) için iz-kesişimi (track IoU) örneği: **Gri** (tahmin) ve **mavi** (gerçek) tespit/yörüngeler üzerinden, kırmızı kutu ile seçilen **TP of interest (c)** için; aynı iki iz boyunca eşleşen TP'ler **TPA(c)** (**yeşil** kesik çizgili bölge), tahmin izinde kalıp bu gerçekle eşleşmeyenler **FPA(c)** (**sarı** kesik çizgili bölge) ve gerçek izde kalıp bu tahminle eşleşmeyenler **FNA(c)** (**kahverengi** kesik çizgili bölge) olarak gösterilmiştir [90].

3.2.4 HOTA Skorunun Hesaplanması

Takip performansı, lokalizasyon, algılama ve eşleştirme gibi üç temel bileşene bağlıdır. Bu nedenle, bu üç alt metrik bir arada değerlendirilmelidir. Ancak pratikte tek bir skorla takipçilerin genel başarımını kıyaslamak istenir. Bu amaçla tanımlanan birleşik metrik **HOTA (Higher Order Tracking Accuracy)**'dır.

DetA ve AssA, belirli bir Loc-IoU eşiği (α) kullanılarak yapılan eşleşmelere dayanır. Bu nedenle, farklı α değerleri için DetA ve AssA hesaplanır. Her eşik için:

$$\text{HOTA}_\alpha = \sqrt{\text{DetA}_\alpha \cdot \text{AssA}_\alpha}$$

Bu işlem tüm α aralığı için entegre edilerek, lokalizasyon doğruluğu da HOTA skoruna dahil edilmiş olur.

Geometrik ortalama kullanılması, DetA ve AssA'ya eşit ağırlık verilmesini sağlar ve herhangi birinin sıfır olması durumunda HOTA'nın da sıfır olmasına neden olur. Ayrıca HOTA, Det-IoU gibi bir yapı olarak da yorumlanabilir: her bir TP tespiti, kendi Ass-IoU değeri ile ağırlıklandırılarak toplam başarıma katkı sağlar.

3.3 Sensör Bozulmasının Modellenmesi

Otonom sürüşte sensör ölçümleri; çevresel koşullar, algılayıcı hataları ve perdeleme (occlusion) gibi nedenlerle daha gürültülü hale gelebilir veya bazı çevrimlerde tamamen kaybolabilir. Bu nedenle takip algoritmalarının, sensör güvenilirliğinin azaldığı koşullarda dayanıklılığını değerlendirmek önemlidir.

Bu tezde sensör bozunumu, gerçek arızaları birebir modellemek yerine kontrollü ve tekrarlanabilir bir simülasyon olarak ele alınmıştır. Bozunum iki şekilde uygulanmıştır: (i) ölçüm belirsizliğinin artırılması (Kalman güncellemesinde kullanılan ölçüm kovaryans matrisi $\mathbf{R}$ ölçeklenerek), (ii) ölçüm düşürme (measurement dropout). Ölçüm düşürmenin uygulandığı çevrimlerde ölçüm güncellemesi atlanmış, yalnızca tahmin adımı çalıştırılmıştır.

Ölçüm belirsizliği senaryolarında kullanılan kovaryans matrisi, $\mathbf{R} = \sigma^2 \mathbf{I}_2$ biçiminde tanımlanmıştır. Burada $\mathbf{I}_2$, 2×2 boyutlu birim (kimlik) matrisini ifade eder ve ölçüm gürültüsünün x ve y eksenlerinde eşit ve bağımsız varsayıldığını gösterir. σ^2 katsayısı artırıldıkça ölçümlere verilen ağırlık azalmakta, takip çıktısı daha fazla hareket modeline dayanmaktadır.

Gerçek sürüşte ölçüm kaybı çoğu zaman tekil ve bağımsız değildir; özellikle perdeleme gibi durumlarda ardışık şekilde oluşabilir. Bu tezde ölçüm düşürme, yöntemler arasında adil karşılaştırma ve tekrarlanabilirlik sağlamak amacıyla kontrollü bir zamanlama ile uygulanmış; aynı düşürme zamanlaması tüm yöntemler (CA, CTRA, IMM) için sabit tutulmuştur.

3.3.1 Ölçüm Belirsizliği ve Ölçüm Düşürme Senaryoları

Ölçüm belirsizliği senaryolarında, Kalman güncellemesinde kullanılan $\mathbf{R}$ artırılarak ölçümlere verilen ağırlık azaltılmıştır. Bu sayede takip çıktısının, ölçüm yerine hareket modeline daha fazla dayandığı koşullar test edilmiştir.

Ölçüm düşürme senaryolarında ise geçici algılama kayıpları simüle edilmiştir. Ölçümün düşürüldüğü çevrimlerde ilgili iz için güncelleme adımı atlanmış, yalnızca tahmin (prediction) adımı işletilmiştir. Kullanılan ölçüm düşürme oranları ve zamanlama kuralları Tablo 3.2 içinde özetlenmiştir (bu bölümde ölçüm düşürme senaryoları $\mathbf{R} = 5\,\mathbf{I}_2$ koşulu altında uygulanmıştır; birimler: m^2). Bu tezde ölçüm düşürme oranı en fazla %20 ile sınırlandırılmıştır; bunun nedeni, %20 ve üzerindeki kayıp oranlarının algı sürekliliğini belirgin biçimde bozarak takip çıktılarının kararlılığını düşürmesi ve pratikte güvenli sürüş için gerekli algı güvenilirliği gereksinimlerine yaklaşan (sınır) koşulları temsil etmesidir.

Tablo 3.2 Sensör bozunumu koşulları: ölçüm gürültüsü ($\mathbf{R}$) ve ölçüm düşürme modeli.

Koşul	$\mathbf{R}$	**Ölçüm düşürme**
Normal	$\mathbf{R} = 0.5\,\mathbf{I}_2$	Yok
Orta gürültü	$\mathbf{R} = 5\,\mathbf{I}_2$	Yok
Yüksek gürültü	$\mathbf{R} = 50\,\mathbf{I}_2$	Yok
Düşürme-10%	$\mathbf{R} = 5\,\mathbf{I}_2$	$T_{\text{on}} \sim \text{Exp}(\lambda_{\text{drop}})$ $T_{\text{off}} \sim \text{Exp}(\lambda_{\text{rec}})$ $p_{\text{drop}} = \dfrac{\lambda_{\text{drop}}}{\lambda_{\text{drop}} + \lambda_{\text{rec}}} = 0.10$
Düşürme-20%	$\mathbf{R} = 5\,\mathbf{I}_2$	$T_{\text{on}} \sim \text{Exp}(\lambda_{\text{drop}})$ $T_{\text{off}} \sim \text{Exp}(\lambda_{\text{rec}})$ $p_{\text{drop}} = \dfrac{\lambda_{\text{drop}}}{\lambda_{\text{drop}} + \lambda_{\text{rec}}} = 0.20$

3.3.2 Başlatma (Initial Conditions) ve Parametre Ayarları

Tüm deneylerde takip algoritması 10 Hz frekansta çalıştırılmıştır ve örnekleme zamanı

$$\Delta t = 0.1 \text{ s}$$

olarak alınmıştır. Bu tezde takip problemi yer düzlemi üzerinde iki boyutlu (2D) olarak ele alınmaktadır; dolayısıyla takip filtresine beslenen temel ölçüm, nesnenin BEV düzlemindeki konumu olan $[x, y]$ bileşenleridir. Başka bir ifadeyle yükseklik (z) ekseni ve tam 3B kutu takibi bu çalışmanın kapsamı dışındadır. CTRA modelindeki ψ değişkeni de 3B bir yönelim değil, yer düzlemindeki sapma (yaw) açısını temsil etmektedir.

Yeni bir takip izi (track) başlatılırken ilk algılamadan yalnızca konum bilgisi kullanılmış; hız ve ivme bileşenleri başlangıçta sıfır olarak atanmıştır. Bu tercih, iz başlangıcında hız/ivme için güvenilir bir doğrudan ölçüm bulunmadığından, filtreyi hatalı bir başlangıç tahminiyle önyargılamamak amacıyla yapılmıştır. Bunun yerine, hız ve ivmeye karşılık gelen durum bileşenlerinin başlangıç belirsizlikleri (kovaryansları) yüksek seçilerek, filtrenin ardışık ölçümlerle hızlı biçimde yakınsaması hedeflenmiştir. CTRA modeli için başlangıç sapma açısı (yaw) açısı algılayıcı tarafından sağlanan sapma açısı (yaw) bilgisinden alınmış; sapma açısı değişim oranı (yaw-rate) ve boylamsal ivme sıfır başlatılmıştır. Bu sayede, ilk anda yönelim bilgisi mevcutken ω ve a gibi türev büyüklüklerin ölçüm gürültüsünden aşırı etkilenmeden zaman içinde öğrenilmesi amaçlanmıştır.

CA modeli için durum vektörü

$$[x, y, v_x, v_y, a_x, a_y]$$

olarak tanımlanmış ve başlangıç kovaryansı

$$\mathbf{P}_{\mathrm{CA}} = \mathrm{diag}(1.0, 1.0, 100.0, 100.0, 100.0, 100.0)$$

seçilmiştir. Burada kovaryans matrisinin köşegen elemanları, durum vektöründeki sıralamaya karşılık gelecek şekilde her bir bileşenin başlangıç belirsizliğini temsil eder. Konum bileşenleri için daha düşük bir belirsizlik (1.0) seçilirken,

$$v_x, v_y, a_x, a_y$$

bileşenleri için daha yüksek değerler (100.0) kullanılarak iz başlangıcındaki dinamik belirsizlik modellenmiştir. Süreç gürültüsü kovaryansı

$$\mathbf{Q}_{\mathrm{CA}} = \mathrm{diag}(1, 1, 5.0, 5.0, 5.0, 5.0)$$

olarak belirlenmiştir. Süreç gürültüsü, CA modelinin ideal varsayımlarından sapmalara (ani hızlanma/yavaşlama, sürüş manevraları, model uyumsuzluğu vb.) tolerans kazandırır ve izlerin aşırı "katı" hale gelmesini engeller.

CTRA modeli için durum vektörü

$$[x, y, \psi, v, \omega, a]$$

olarak tanımlanmış ve başlangıç kovaryansı

$$\mathbf{P}_{\mathrm{CTRA}} = \mathrm{diag}(1.0, 1.0, 1.0, 100.0, 1.0, 100.0)$$

seçilmiştir. Bu modelde x, y konumuna ek olarak ψ (yaw) ve ω (yaw-rate) terimleri ile 2B düzlemde viraj/ dönüş hareketi daha iyi temsil edilebilmektedir. Başlangıçta v ve a için yüksek belirsizlik (100.0) atanması, iz ilk oluştuğunda hız/ivmenin güvenilir şekilde bilinmediğini; buna karşın ψ ve ω için daha düşük belirsizlik seçimi ise yönelim bilgisinin algılayıcıdan geldiği ve kısa vadede daha kararlı varsayıldığı durumları yansıtır. Süreç gürültüsü kovaryansı

$$\mathbf{Q}_{\mathrm{CTRA}} = \mathrm{diag}(1.0, 1.0, 0.5, 5.0, 0.5, 5.0)$$

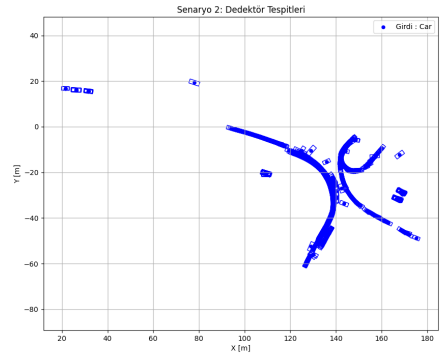
olarak belirlenmiştir. Burada özellikle ψ ve ω için süreç gürültüsünün daha kontrollü seçilmesi, dönüş dinamiği modellemesi yapılırken ölçüm gürültüsünün filtreyi aşırı dalgalandırmasını sınırlamayı hedefler.

IMM yapısında iki mod kullanılmıştır: CA ve CTRA. Mod geçiş olasılıkları sabit bir geçiş matrisi ile tanımlanmış ve her yeni iz için mod olasılıkları eşit başlatılmıştır. Bu başlatma, iz başlangıcında nesnenin hareket tipine (yaklaşık doğrusal/ivmeli mi yoksa dönüşlü mü) dair güçlü bir ön kabule gitmeden, ölçüm dizisi ilerledikçe mod olasılıklarının veriye dayalı olarak ayrışmasına olanak sağlar. Böylece IMM, 2D takip senaryosunda farklı hareket karakteristikleri arasında adaptif geçiş yaparak hem konum kestiriminin kararlılığını hem de kimlik sürekliliğini (association) desteklemeyi amaçlar.

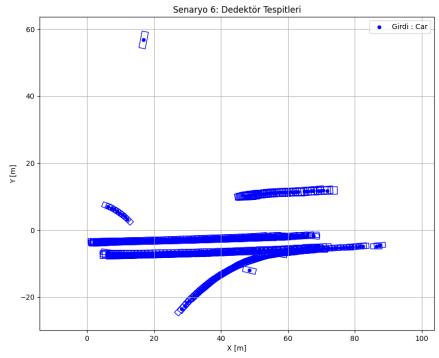
3.4 Test Kurgusu, Senaryo Seçimi ve Temsil Yeteneği

Bu çalışmada değerlendirme iki aşamalı bir yaklaşımla kurgulanmıştır. İlk aşamada, yöntemlerin genel geçer davranışını ve sensör bozunumu altında (ölçüm güveninin düşürülmesi) performans değişimini gözlemleyebilmek amacıyla KITTI sekanslarının tamamı üzerinde testler gerçekleştirilmiştir. İkinci aşamada ise, ilk aşamadaki bulguların ışığında, etkilerin en belirgin şekilde ortaya çıktığı ve kavşak içeriğinin sekans boyunca baskın olduğu sahneler seçilerek detaylı analiz yapılmıştır. Bu nedenle bölümün geri kalanında önce genel test kurgusu, ardından kavşak senaryolarının seçimi ve temsil yeteneği açıklanmaktadır. Tüm sekanslar

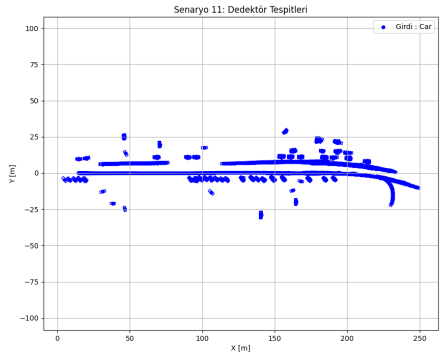
içinden rastgele seçilen bazı senaryolara ait örnek dedektör girdileri Şekil 3.5 üzerinde görülebilmektedir.



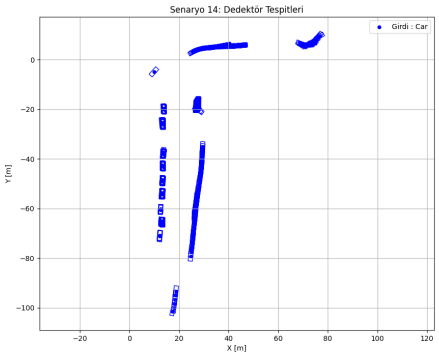
(a) Senaryo 2: Dedektör tespitleri (girdi)



(b) Senaryo 6: Dedektör tespitleri (girdi)



(c) Senaryo 11: Dedektör tespitleri (girdi)



(d) Senaryo 14: Dedektör tespitleri (girdi)

Şekil 3.5 Genel senaryolardan örnek dedektör girdileri (2B yer düzleminde araç tespitleri).

İlk aşamada tüm sekanslardan elde edilen genel sonuçlar, sensör bozunumu etkisinin hangi tür sahnelerde daha görünür hale geldiğini ortaya koymuştur. Bu tümdengelimli akış doğrultusunda, kavşak dinamiklerini (dur-kalk, dönüş, çapraz trafik etkileşimi) sekans boyunca yoğun biçimde barındıran aşağıdaki senaryolar, nitel ve görsel açıdan detaylı inceleme için seçilmiştir.

Bu tezde "kavşak senaryosu", ego aracın ilgili KITTI sekansı boyunca tüm çevrimlerde kavşak bölgesinde bulunduğu ve sahnede dur-kalk, dönüş manevrası, çapraz trafik etkileşimi gibi hareket örüntülerinin yoğun olarak gözlendiği sekanslar olarak tanımlanmıştır. Bu koşulu sağlayan sekanslar, kamera görüntüleri incelenerek belirlenmiştir. Değerlendirmeler, seçilen sekansların tamamı kullanılarak (sekans kırpma uygulanmadan) gerçekleştirilmiştir; böylece tüm yöntemler aynı veri üzerinde adil biçimde karşılaştırılmıştır.

Düz sürüşün baskın olduğu sahnelerde hedef dinamikleri çoğunlukla yaklaşık doğrusal olduğundan, CV/CA gibi basit kinematik modeller genellikle yeterli

takip performansı sunabilmektedir. Bu tür sahnelerde çoklu-model yaklaşımının (ör. IMM) sağlayacağı kazanım sınırlı kalabilir; ayrıca dönüş hipotezi içeren bir model ailesi, manevra bulunmayan zaman aralıklarında olasılık karışımında ek belirsizlik üretebilir. Bu tez çalışmasının amacı genel düz-sürüş senaryolarında en iyi performansı göstermekten ziyade, kavşaklarda sık gözlenen hareket geçişlerinin (düz gidiş, dönüş, yeniden hızlanma/dur-kalk) takip kararlılığı üzerindeki etkisini incelemektir. Bu nedenle değerlendirme, kavşak içeriğinin sekans boyunca baskın olduğu sekanslar üzerinde gerçekleştirilmiştir.

- Senaryo 0012
- Senaryo 0015

Yukarıdaki tanıma uyan kavşak senaryoları olmaları neden ile incelenmek için uygun görülmüştür. Görselde 3.12, bulunan araçların hareketleri görülebilmektedir. Bu senaryoda referans araç önce bir kavşağa gelip kırmızı ışıkta bekler. Bu sırada kavşakta bekleyen diğer araçlar için ışık yanar araçların bazıları düz giderken bazılarının dönüş yaptığı görülmektedir. 3.6 ve 3.7 senaryodan alınan senkron görüntülerdir. 19 Nolu aracın kavşakta yaptığı dönüş hareketi 3.8 ve 3.9 görüntü grubuna bakılarak anlaşılabilmektedir.

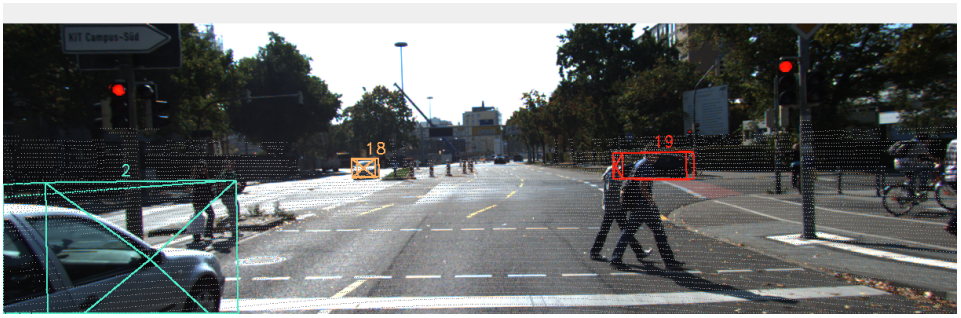
Düşük sensör güvenilirliğini simüle etmek amacıyla ölçüm kovaryans matrisi bir ölçek faktörü ile artırılmıştır (örneğin, $R = sR_0$, $s = 50$). Bu koşul altında testler gerçekleştirilmiştir.

IMM filtresi birden çok model ile çalışabilir.Model sayısı arttıkça hem bellek kullanımı hem de cpu kullanımı artmaktadır.Bu nedenle seçilecek modellerin kullanıma uygun ve akıllıca seçilmesi gerekmektedir.

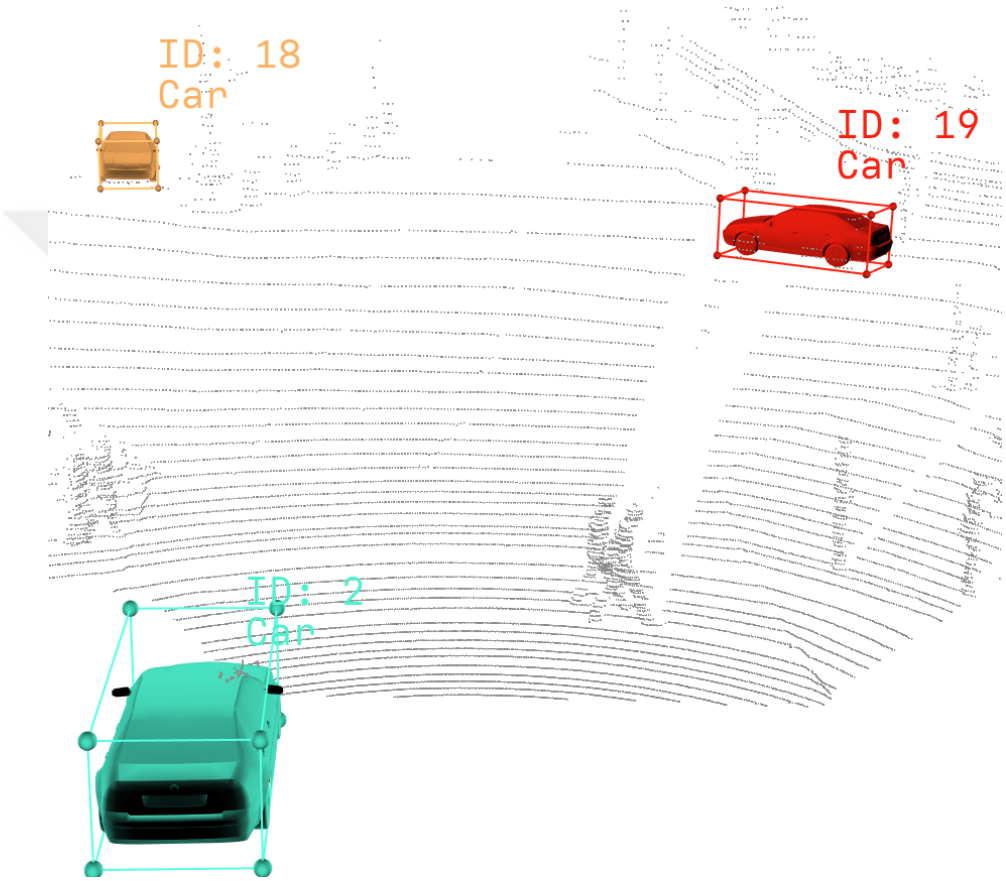
Kavşaklarda en çok karşılaşılan araç manevraları

- Hızlanma -Yavaşlama
- Dönme
- Düz gitme

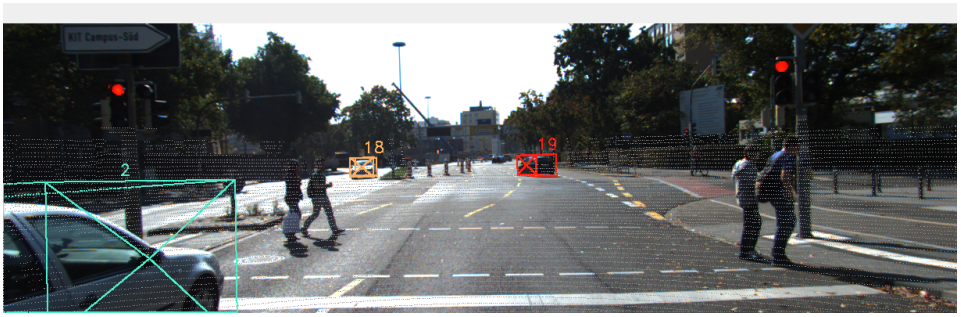
olduğu düşünülerek önceki bölümde incelenen hareket modellerinden ivme içerenlerin kullanılması uygun bulunmuştur. Düz giden araçları daha iyi modellemek için sabit ivme (CA) dönen araçları daha iyi modellemek için ise sabit dönüş hızı ve sabit ivme (CTRA) modellerinin kullanılması uygun bulunmuştur.



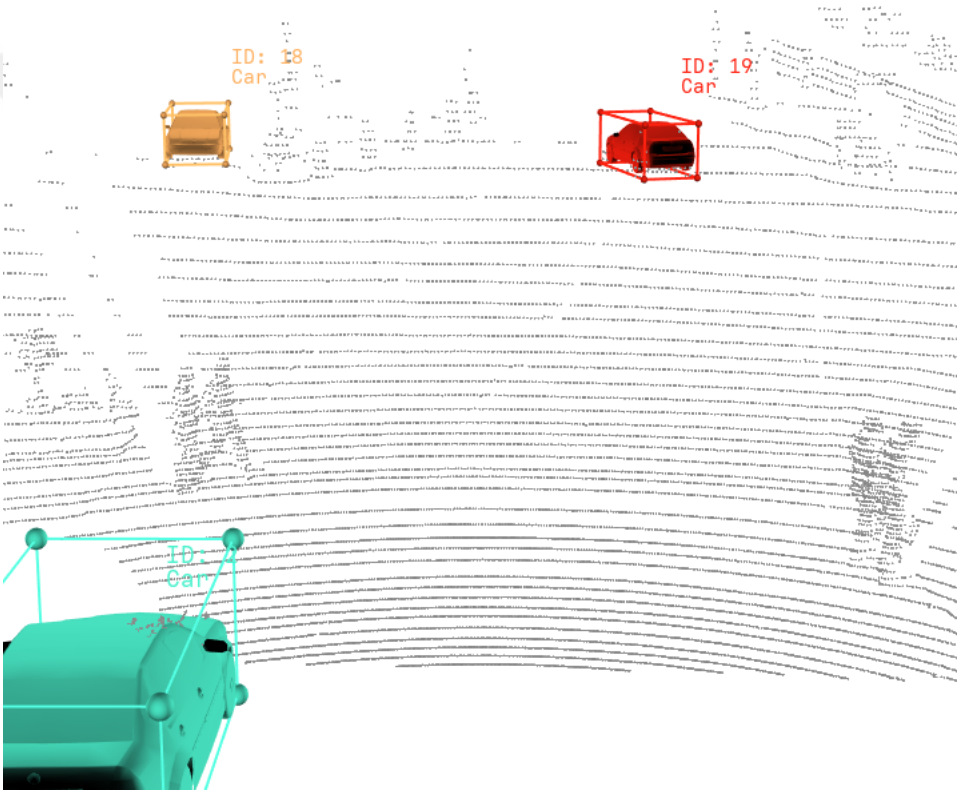
Şekil 3.6 Kavşak Senaryosuna ait 1.kamera görüntüsü



Şekil 3.7 Kavşak senaryosuna ait 1.LiDAR nokta bulutu görüntüsü

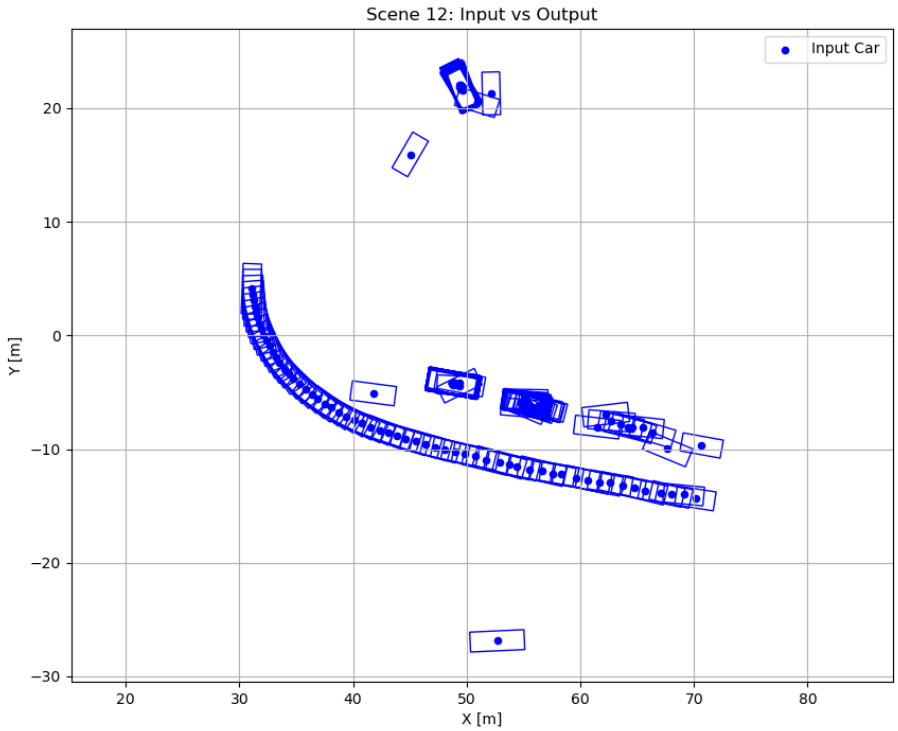


Şekil 3.8 Kavşak Senaryosuna ait 2.kamera görüntüsü

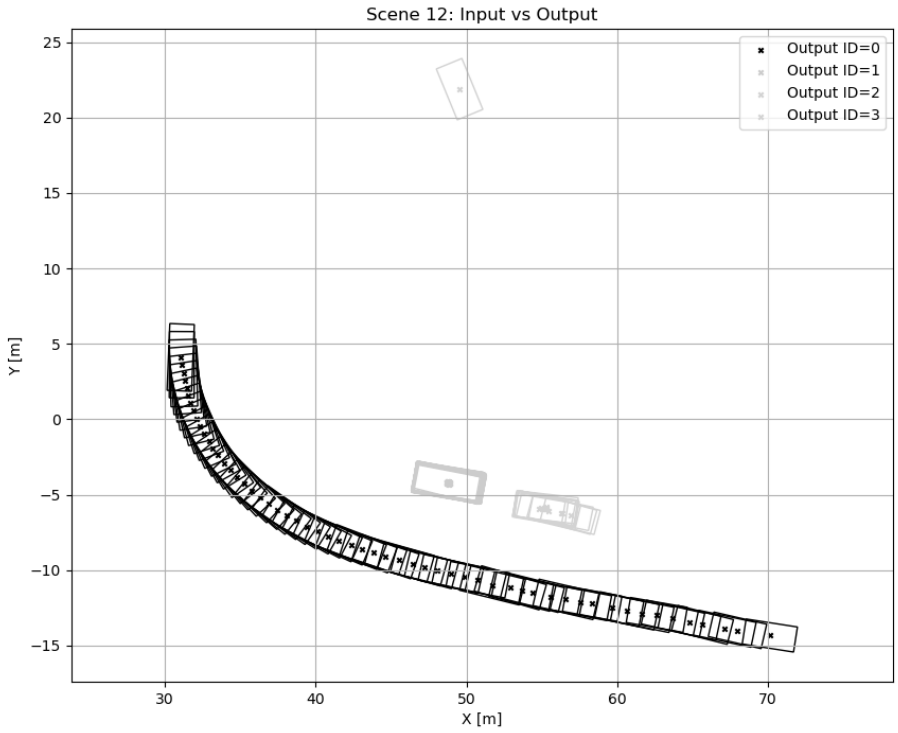


Şekil 3.9 Kavşak senaryosuna ait 2.LiDAR nokta bulutu görüntüsü

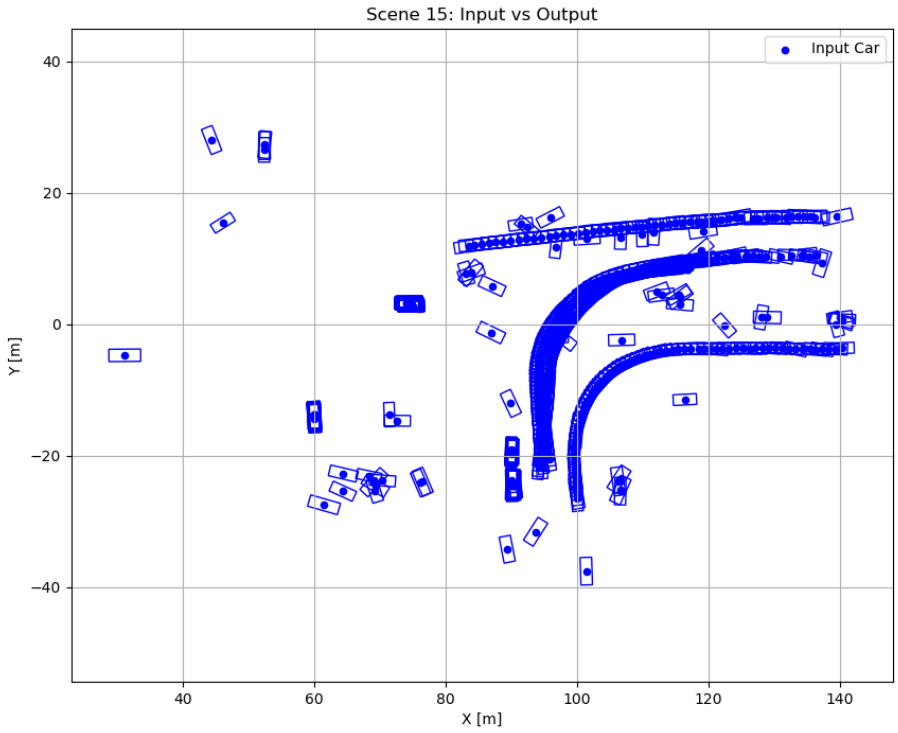
Nesne takip modelini test etmek için literatürde mevcut ve başarılı sonuçlar veren algılama (detection) yöntemleri ile üretilmiş tespitler hazır girdi olarak kullanılmaktadır. Bu tez kapsamında, KITTI veri seti üzerinde LiDAR nokta bulutu tabanlı 3B nesne tespiti yapan VirConv dedektörünün çıktıları ölçüm olarak kullanılmıştır [86].VirConv, her nesne için 3B sınırlayıcı kutu parametreleri ve güven skoru üretmektedir. Bu tezde takip problemi 2D yer düzleminde ele alındığından, takip filtresine ölçüm olarak yalnızca tespit kutusunun konum bileşenleri $z_k = [x_k, y_k]^\top$ aktarılmış; 2B sınırlayıcı kutu bilgileri ise yalnızca HOTA değerlendirmesinde IoU hesabı için kullanılmıştır.



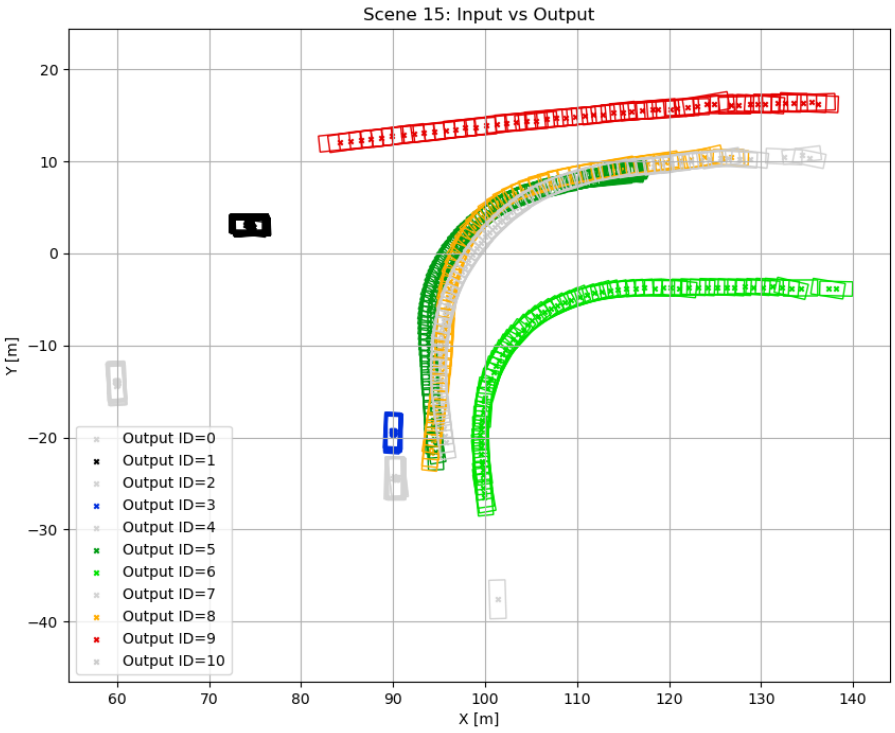
Şekil 3.10 Kitti 12. Senaryo Girdi Ölçümlerin Gösterimi



Şekil 3.11 Kitti 12. Senaryo R=5 IMM - Takip Çıktı Sonuçları Gösterimi



Şekil 3.12 Kitti 15. Senaryo Girdi Ölçümlerin Gösterimi



Şekil 3.13 Kitti 15. Senaryo R=5 IMM - Takip Çıktı Sonuçları Gösterimi

3.5 Sonuçların Tartışılması

3.5.1 Farklı Algılayıcılar İçin Referans Durum Sonuçları

Tablo 3.3 VirConv Dedektörü (Nominal: %0 Kayıp, R=0.5) – Sadece Combined Sonuçlar

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
CA	82.454	79.445	85.723	86.710	87.520	89.900	92.658
CTRA	**82.537**	79.439	**85.902**	86.703	87.503	**90.119**	92.455
IMM (CA+CTRA)	82.515	**79.444**	85.642	**86.712**	**87.516**	89.848	**92.565**

Tablo 3.4 CasA Dedektörü (Nominal: %0 Kayıp, R=0.5) – Sadece Combined Sonuçlar

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
CA	78.609	74.069	83.610	85.039	81.213	88.534	**90.242**
CTRA	**78.791**	**74.109**	**83.951**	**85.055**	**81.248**	**88.899**	90.241
IMM (CA+CTRA)	78.686	74.095	83.542	85.055	81.238	88.487	90.223

Nominal durumda (Tablo 3.3 ve Tablo 3.4) üç takip modelinin (CA, CTRA, IMM) HOTA değerleri birbirine çok yakındır; çünkü ölçümler her karede mevcut ve güven seviyesi yüksektir (R=0.5). Bu koşulda filtre her adımda ölçümle kendini düzelttiği için, hareket modeli kaynaklı küçük farklar birikmeden bastırılır; dolayısıyla performansı belirleyen ana unsur çoğunlukla dedektörün ölçüm kalitesi olur.

VirConv tarafında en yüksek HOTA CTRA ile 82.537'dir; IMM 82.515 ile yalnızca 0.022 puan geride, CA ise 82.454 ile CTRA'dan 0.083 puan düşüktür (Tablo 3.3). Benzer şekilde CasA tarafında en yüksek HOTA 78.791 ile yine CTRA'dadır; IMM 78.686 ile 0.105 puan, CA ise 78.609 ile 0.182 puan geridedir (Tablo 3.4). Özetle nominal senaryoda model seçimi ikincil etkide kalmakta; VirConv'ın CasA'ya göre daha yüksek HOTA üretmesi (yaklaşık 82.5 vs 78.7–78.8) dedektör kaynaklı ölçüm kalitesi farkının daha belirleyici olduğunu göstermektedir.

3.5.2 Farklı Algılayıcılar İçin Ölçüm Düşürme Senaryosunun Etkisinin İncelenmesi

Tablo 3.5 VirConv Dedektörü (%20 Kayıp) – Sadece Combined Sonuçlar

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
CA	80.287	**77.472**	83.322	**85.136**	**86.276**	88.112	**91.248**
CTRA	80.218	77.321	83.346	84.994	86.117	88.125	91.151
IMM (CA+CTRA)	**80.463**	77.459	**83.492**	85.119	86.273	**88.309**	91.182

%20 kayıp senaryosunda (Tablo 3.5 ve Tablo 3.6) ölçümler zaman zaman gelmediği için takipçi o anlarda kendini ölçümle düzeltemez ve daha fazla hareket modeli tabanlı öngörüye dayanmak zorunda kalır. Bu durum pratikte iki sonucu büyütür:

Tablo 3.6 CasA Dedektörü (%20 Kayıp) – Sadece Combined Sonuçlar

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
CA	76.179	72.709	79.958	83.580	80.522	85.229	88.941
CTRA	76.487	72.645	80.698	83.496	80.451	86.017	88.938
IMM (CA+CTRA)	**77.092**	**72.747**	**81.649**	**83.594**	**80.559**	**87.079**	**88.976**

(i) hedef beklenmedik manevra yaparsa tek bir modele (CA veya CTRA) bağlı kalmak hata birikimini artırabilir, (ii) hata büyüdükçe gating ve veri ilişkilendirme zorlaşır; bu da AssA gibi ilişkilendirme metriklerine ve dolayısıyla HOTA'ya yansır. IMM'nin çok-modlu yapısı, ölçümün yeniden geldiği anlarda daha uygun hareket hipotezine daha hızlı yakınsama sağlayabildiği için bu tip aralıklı gözlenebilirlik koşullarında avantaj üretebilir.

VirConv için %20 kayıpta en iyi HOTA, IMM ile 80.463'tür; CA'ya göre +0.176 ve CTRA'ya göre +0.245 puan daha yüksektir (Tablo 3.5). Kazanım sınırlı olsa da tutarlıdır ve özellikle AssA'nın 83.492 ile en yüksek değere çıkması, ilişkilendirme tarafındaki stabilizasyonu işaret eder. CasA için ise %20 kayıpta IMM farkı daha belirgindir: IMM 77.092 ile en iyi sonuçtur; CA'ya göre +0.913 ve CTRA'ya göre +0.605 puan üstündür (Tablo 3.6). CasA'da AssA (81.649) ve AssRe (87.079) gibi ilişkilendirme odaklı metriklerdeki artış, ölçüm kesintileri arttıkça IMM'nin iz sürekliliğini korumada daha görünür bir katkı sağlayabildiğini göstermektedir.

3.5.3 Farklı Algılayıcılar İçin Yüksek Gürültü Senaryosunun Etkisinin İncelenmesi

Tablo 3.7 VirConv Dedektörü (No Drop, R=50) – Sadece Combined Sonuçlar

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
CA	**81.748**	**78.609**	**85.177**	**86.106**	**86.794**	**89.575**	**92.091**
CTRA	78.376	74.684	82.507	83.292	83.980	87.367	90.378
IMM (CA+CTRA)	79.834	77.260	82.681	85.133	85.817	87.416	91.014

Tablo 3.8 CasA Dedektörü (No Drop, R=50) – Sadece Combined Sonuçlar

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
CA	**77.926**	**73.387**	**82.944**	**84.472**	**80.662**	**88.090**	**89.663**
CTRA	74.943	70.062	80.470	81.875	78.282	86.024	88.138
IMM (CA+CTRA)	75.726	72.298	79.569	83.732	80.021	85.398	88.246

Yüksek gürültü senaryosunda (R=50) ölçüm mevcut olsa bile ölçüme duyulan güven düştüğü için takipçi ölçüm güncellemesini daha "temkinli" yapar; buna karşılık modelin öngörüsü daha belirleyici hale gelir. Bu nedenle R=0.5'e kıyasla modellerin birbirinden ayrışması beklenir: özellikle manevra serbestliği daha yüksek olan modeller (ör. CTRA) uygun olmayan sekanslarda hata birikimine

daha yatkın hale gelebilir; hata birikimi büyüdükçe ilişkilendirme zorlaşır ve HOTA düşer.

VirConv tarafında R=0.5 $\rightarrow$ R=50 geçişinde CA sadece 0.706 puan düşerken (82.454 $\rightarrow$ 81.748), CTRA 4.161 puan düşmektedir (82.537 $\rightarrow$ 78.376); IMM ise 2.681 puan düşer (82.515 $\rightarrow$ 79.834) (Tablo 3.3, Tablo 3.7). R=50 altında sıralama CA (81.748) > IMM (79.834) > CTRA (78.376) şeklindedir; yani IMM, CTRA'ya göre 1.458 puan daha yüksek HOTA üretse de CA kadar kararlı değildir. Bu tablo, yüksek gürültüde "daha basit" bir modelin (CA) daha az hata biriktirip ilişkilendirmeyi daha az zorlayabildiğini; IMM'nin ise CTRA'nın olası yüksek hatasını kısmen dengeleyerek ara bir dayanıklılık sunduğunu düşündürür.

CasA tarafında da benzer eğilim görülür: CA 0.683 puan düşerken (78.609 $\rightarrow$ 77.926), CTRA 3.848 puan düşer (78.791 $\rightarrow$ 74.943) ve IMM 2.960 puan düşer (78.686 $\rightarrow$ 75.726) (Tablo 3.4, Tablo 3.8). R=50 altında CA en iyidir (77.926).

3.5.4 Kavşak Senaryolarında Etkinin Detaylı İncelenmesi

Tablo 3.9 *val-car* 0012 sekansı için CA, CTRA ve IMM yöntemlerinin HOTA metrikleri (R değerine göre gruplandırılmıştır). Her R bloğu içinde en iyi sonuçlar kalın olarak gösterilmiştir.

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
$R = 0.5$							
CA	83.166	82.576	83.775	84.505	93.676	85.027	94.778
CTRA	83.178	82.473	**83.930**	84.468	93.635	**85.267**	**95.023**
IMM	**83.21**	**82.624**	83.812	**84.579**	**93.758**	85.078	94.836
$R = 5$							
CA	83.113	82.449	83.819	84.431	93.594	85.170	94.923
CTRA	81.256	79.903	82.805	82.738	91.718	84.566	94.103
IMM	**83.220**	**82.547**	**83.929**	**84.542**	**93.717**	**85.257**	**95.018**
$R = 50$							
CA	82.002	80.993	83.118	83.364	92.411	84.620	94.237
CTRA	73.457	69.362	78.519	75.819	84.047	81.478	90.125
IMM	**82.557**	**81.570**	**83.659**	**83.842**	**92.942**	**85.128**	**94.811**

Tablo 3.10 *val-car* 0012 sekansı için $R = 5$ koşulunda ölçüm düşürmeli senaryo sonuçları. Her düşürme oranında en iyi sonuçlar kalın olarak gösterilmiştir.

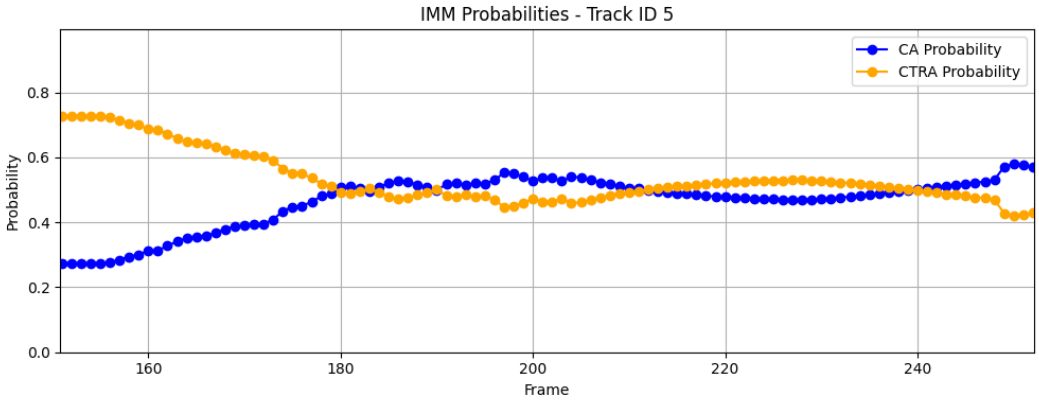
Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
Drop = 10%							
CA	82.818	82.131	83.539	84.211	93.350	84.920	94.636
CTRA	82.667	81.837	83.559	83.990	93.105	84.944	94.636
IMM	**82.923**	**82.251**	**83.622**	**84.321**	**93.472**	**84.969**	**94.694**
Drop = 20%							
CA	**82.145**	**81.264**	83.083	**83.658**	**92.738**	84.618	94.251
CTRA	81.561	80.415	82.810	83.070	92.085	84.350	93.907
IMM	82.110	81.194	**83.097**	83.585	92.656	**84.640**	**94.270**

Tablo 3.11 *val-car* 0015 sekansı için ölçüm düşürme olmayan senaryoda farklı ölçüm gürültüsü seviyelerinde elde edilen sonuçlar. Her gürültü seviyesinde en iyi sonuçlar kalın olarak gösterilmiştir.

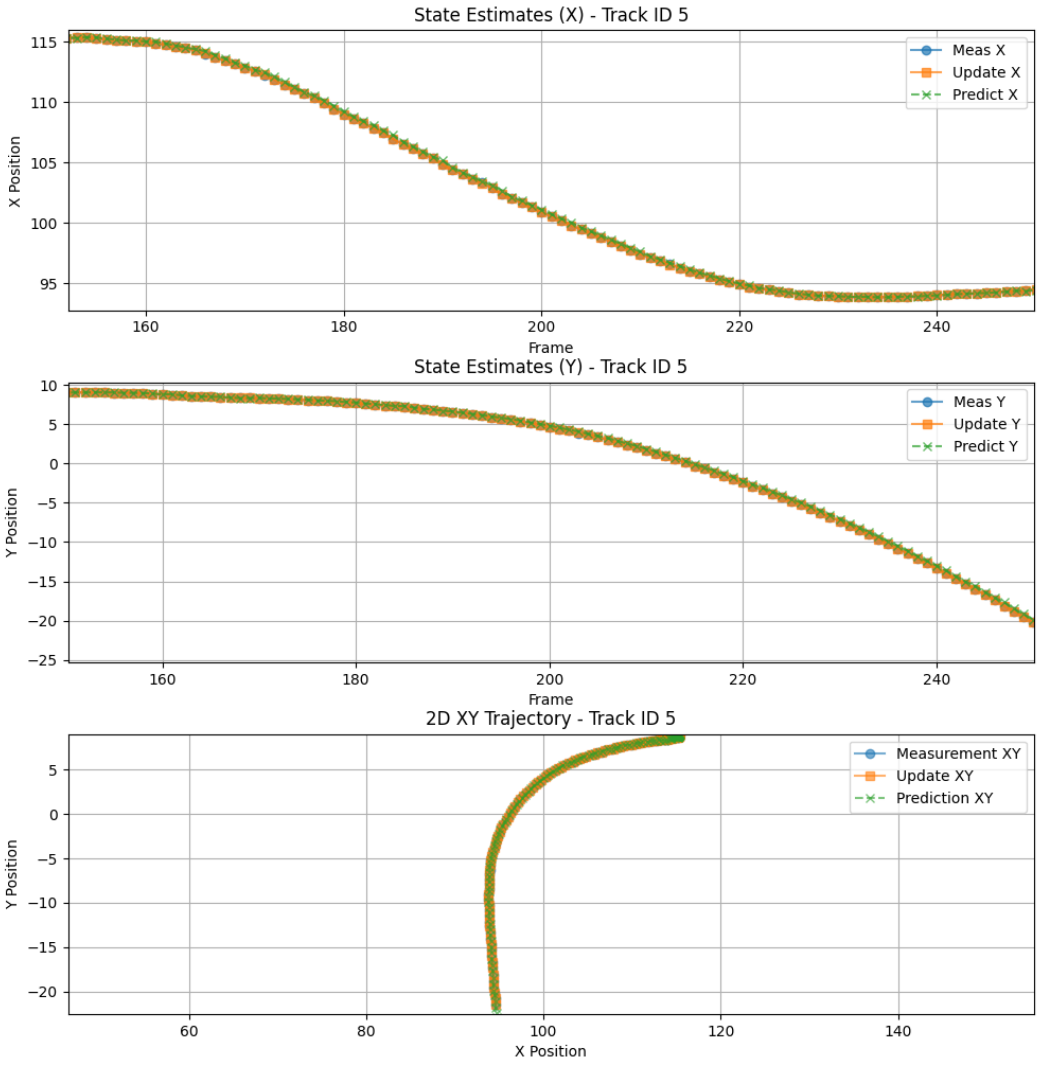
Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
R = 0.5							
CA	84.059	83.398	84.787	**86.211**	**90.217**	87.445	91.389
CTRA	84.063	83.395	84.797	86.192	90.198	87.435	91.360
IMM	**84.067**	**83.402**	**84.799**	**86.211**	**90.217**	**87.448**	**91.392**
R = 5							
CA	84.160	83.503	84.887	86.118	90.287	87.353	**91.441**
CTRA	83.666	82.901	84.500	85.669	89.817	87.001	90.978
IMM	**84.188**	**83.537**	**84.9**	**86.164**	**90.336**	**87.354**	91.416
R = 50							
CA	**83.610**	**82.878**	**84.443**	**85.501**	**89.640**	**86.905**	**90.971**
CTRA	78.511	76.156	81.122	81.191	85.122	84.189	88.455
IMM	82.256	81.213	83.416	84.5	88.757	86.261	90.42

Tablo 3.12 *val-car* 0015 sekansı için $R = 5$ koşulunda ölçüm düşürmeli senaryo sonuçları. Her düşürme oranında en iyi sonuçlar kalın olarak gösterilmiştir.

Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
Drop = 10%							
CA	**82.988**	**82.314**	**83.763**	**85.220**	**89.346**	**86.587**	**90.687**
CTRA	76.435	73.948	79.190	79.396	83.240	82.641	86.758
IMM	82.940	82.257	83.712	85.183	89.307	86.529	90.602
Drop = 20%							
CA	80.228	79.469	**81.129**	83.276	87.308	**84.870**	**88.892**
CTRA	71.838	70.980	72.994	77.106	80.839	76.785	84.996
IMM	**80.233**	**79.486**	81.105	**83.332**	**87.366**	**84.870**	88.876



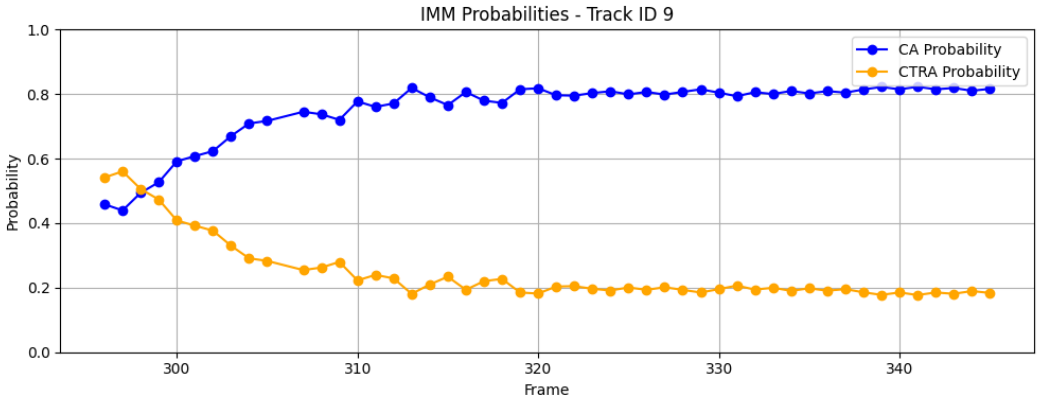
Şekil 3.14 Kitti 15. Senaryo Nesne 5 model olasılıkları



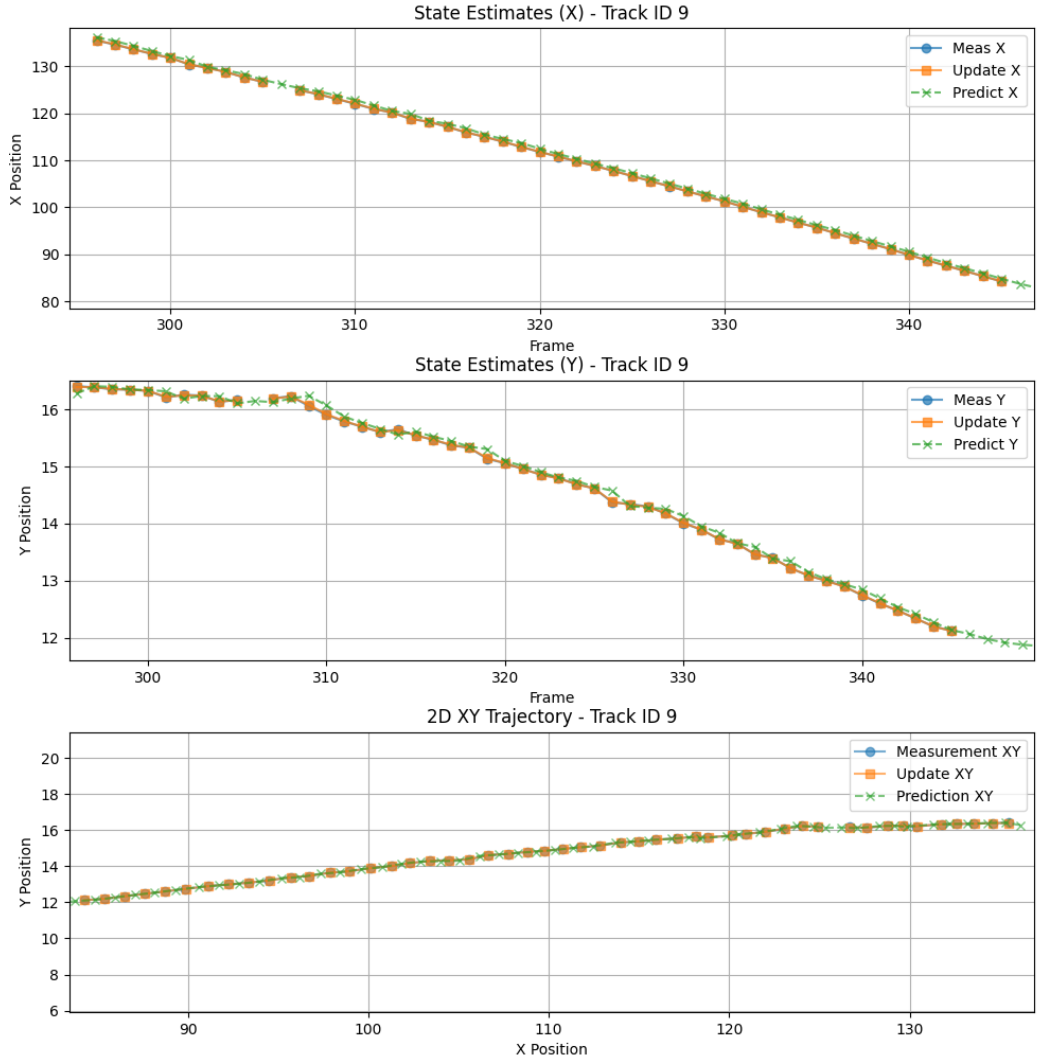
Şekil 3.15 Kitti 15. Senaryo Nesne 5 Pozisyon değişimi

İlk olarak IMM çalışmasını gösteren bir kaç farklı nesne poz ve olasılık grafikleri analiz edilmiştir.

Senaryo 0012 için girdi ölçümleri ve IMM takip çıktıları sırasıyla Şekil 3.10 ve



Şekil 3.16 Kitti 15. Senaryo Nesne 9 model olasılıkları



Şekil 3.17 Kitti 15. Senaryo Nesne 9 Pozisyon değişimi

Şekil 3.11 ile gösterilmiştir.

Düşük hız rejiminde CTRA modelinin (turn-rate) parametreleri ölçümlerden yeterince iyi ayrıştırılamadığından, bu bölgede yapılan model seçimi her zaman güvenilir olmayabilmektedir (bkz. Şekil 3.14). Burada araç düşük hızda iken CTRA modeli yüksek olasılığa çıkmıştır. Hareket düze yakın şekilde başladıktan sonra olasılıklar dengeli bir seviyeye gelmiştir. Yörüngede belirgin bir dönüş gözlense bile, hız/ivme profilinin CA modeli ile kısmen açıklanabildiği durumlarda modeller arasındaki olabilirlik (likelihood) farkı küçük kalır. Bu durumda mod olasılıklarının zaman içinde sıralı biçimde öne çıkması ve CA–CTRA arasında yumuşak geçişler göstermesi IMM yapısı için beklenen bir davranıştır; IMM, sert bir model değişimi yerine olasılıksal etkileşim yaklaşımı uygular (Şekil 3.14). Buna rağmen, birleşik durum kestiriminin ölçüm ve güncelleme sonuçlarıyla uyumlu ve genel olarak kararlı kaldığı gözlenmektedir (Şekil 3.15).

Buna ek olarak, düz ve kararlı hareketin baskın olduğu bir örnekte CA modelinin belirgin biçimde daha yüksek olasılık aldığı görülmektedir (Şekil 3.16). Araç yaklaşık sabit hızla ve düşük manevra içeren bir yörünge izlediğinden, CA modeli gözlemleri daha tutarlı açıklamakta ve IMM karışım ağırlıkları CA lehine yakınsamaktadır. Bu senaryoda mod olasılıklarının zamana bağlı dalgalanmasının sınırlı kaldığı, birleşik durum kestiriminin de ölçüm ile güncelleme sonuçlarını yakından takip ederek düzgün ve kararlı bir pozisyon değişimi ürettiği gözlenmiştir (Şekil 3.17).

Bu bölümde Tablo 3.9–3.12 ile raporlanan nicel sonuçlar, hangi koşullarda hangi modelin neden daha avantajlı olduğunu açıklamak amacıyla tartışılmaktadır. Bu tez kapsamında algılayıcı tespitleri sabit tutulduğundan, gözlenen farklar esas olarak takip filtresinin (hareket modeli + veri ilişkilendirme) davranışından kaynaklanmaktadır.

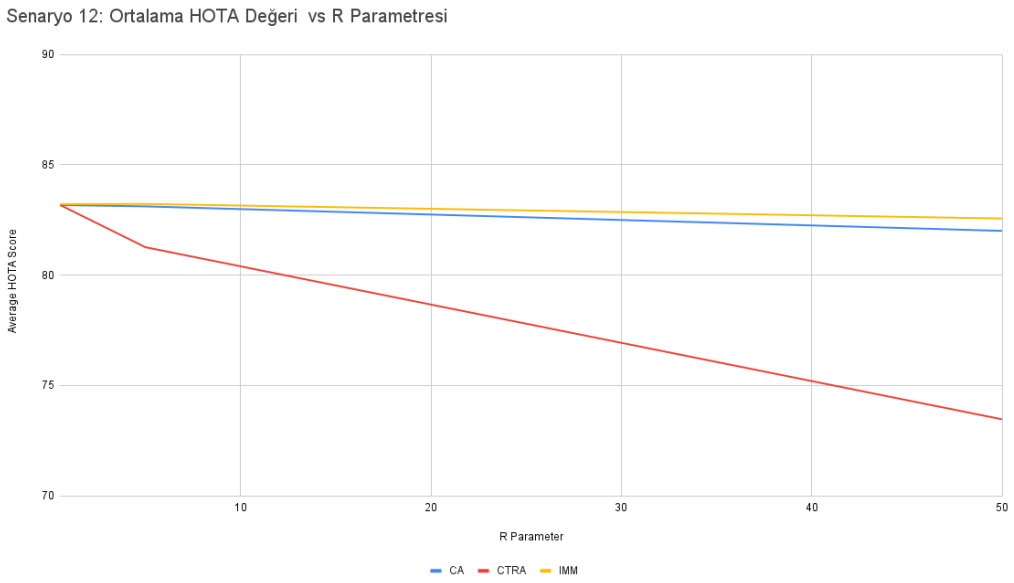
3.5.5 Ölçüm gürültüsü (R) arttıkça modelin baskın hale gelmesi

Ölçüm kovaryansı R büyüdükçe Kalman güncellemesi ölçüme daha az ağırlık verir ve takip çıktısı giderek hareket modeline dayalı hale gelir. Bu nedenle yüksek R koşulları, model seçiminin etkisinin en net görüldüğü senaryolardır.

Seq 0012 sonuçlarında (Tablo 3.9), $R = 50$ seviyesinde CTRA modelinin HOTA/DetA/AssA metriklerinde belirgin şekilde düştüğü görülürken, IMM yaklaşımı CA'ya yakın kalmakta ve hatta CA'ya göre küçük fakat tutarlı bir artış sağlayabilmektedir (ör. HOTA: CA 82.002 $\rightarrow$ IMM 82.557). Bu gözlem, kavşak gibi manevra geçişlerinin bulunduğu sahnelerde tek bir modele bağlı kalmanın

yüksek belirsizlik altında takip kararlılığını zayıflattığını göstermektedir.

Buna karşın seq 0015 için (Tablo 3.11) özellikle $R = 50$ durumunda CA modelinin IMM'den daha iyi sonuç verdiği görülmektedir. Bu nedenle IMM'in katkısı, her sahnede ortalama metrikleri artırmaktan ziyade, manevra yoğun ve belirsizliğin yüksek olduğu koşullarda sağlamlık (robustness) kazancı sağlıyor olmasıdır. Ölçüm gürültüsüne bağlı HOTA değişimi, sırasıyla senaryo 0012 ve 0015 için Şekil 3.18 ve Şekil 3.19'te özetlenmiştir.

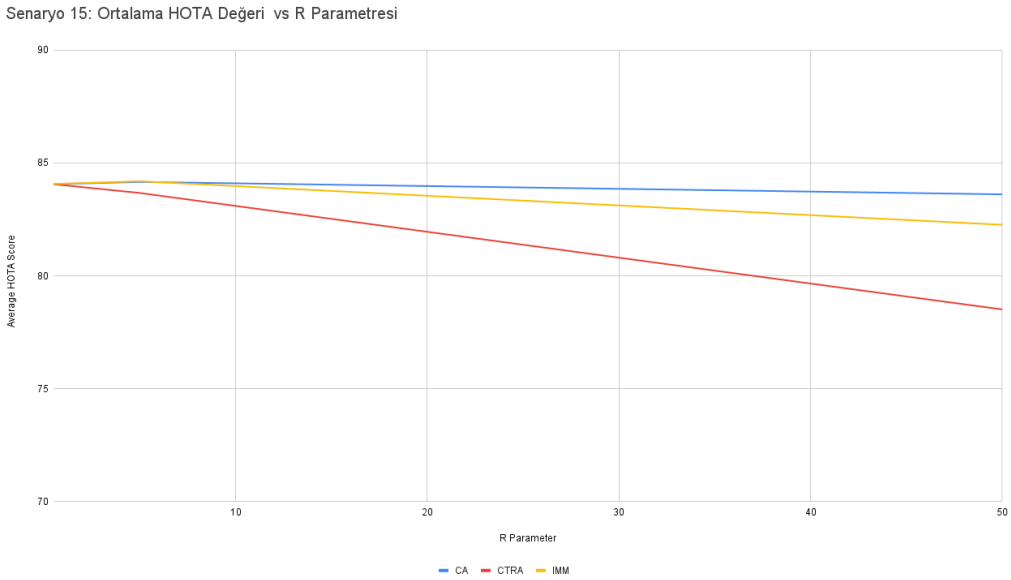


Şekil 3.18 Senaryo 0012 v için CA/CTRA/IMM yöntemlerinde HOTA'nın ölçüm gürültüsüne (R) göre değişimi.

3.5.6 Ölçüm düşürme senaryolarında model sonuçlarının incelenmesi

Ölçüm düşürme uygulanan çevrimlerde güncelleme yapılmadığı için izler sadece hareket modeli ile tahmin ile ilerler. Bu durumda hata birikimi artar; ayrıca veri ilişkilendirme (association) daha zor hale gelir. Tablo 3.10 ve Tablo 3.12 incelendiğinde, ölçüm düşürme oranı arttıkça HOTA'nın genel olarak düştüğü görülmektedir.

Özellikle seq 0015 için CTRA modelinin ölçüm düşürme durumunda ciddi şekilde zayıflaması (ör. Drop=10% için HOTA 76.435) CTRA'nın bu koşullarda daha fazla eşleşme problemi üretebildiğini göstermektedir. IMM ise bu çöküşü engelleyerek CA seviyesine yakın bir performansla takip edebilmekte; bu da IMM'in pratikte "en kötü durum" davranışını yumuşatma avantajını göstermektedir. Bununla birlikte, CA ile IMM arasındaki fark bazı koşullarda çok küçük olduğundan, IMM tercihi daha çok manevra yoğun sahnelerde güvenlik marjı sağlama hedefiyle



Şekil 3.19 Senaryo 0015 v için CA/CTRA/IMM yöntemlerinde HOTA'nın ölçüm gürültüsüne (R) göre değişimi.

gerekçelendirilmelidir.

3.5.7 CA ve IMM'nin dönüş segmentlerindeki davranışı

Tablolar incelendiğinde CA ve IMM sonuçlarının birçok koşulda birbirine çok yakın olduğu, bazı koşullarda ise CA'nın daha yüksek HOTA ürettiği görülmektedir (ör. seq 0015, $R = 50$). Bu durumun temel nedeni, kavşak sahnelerinde dahi hedeflerin zamanın önemli bir bölümünde düz/az manevralı hareket etmesi ve CA modelinin bu rejimde oldukça güçlü olmasıdır. Bu nedenle IMM'in temel katkısı, ortalama skor artışından ziyade manevra ve kısa süreli ölçüm kaybı anlarında kararlılığı artırma (robustness) yönündedir.

Bu hipotezi daha görünür kılmak için, senaryo 0015 şekil 3.13 üzerinde dönüş içeren araçlar seçilerek (nesne numaraları: 5, 6, 8, 10) yalnızca dönüş segmentlerinde ve bu segmentlerle çakışan ölçüm düşürme aralıkları sonrasında oluşan konum hatası analiz edilmiştir. Bir çevrimden bir sonraki çevrime gerçek sapma açısı (yaw) değişimi 0.01'den büyükse araç "dönüş halinde" kabul edilmiştir. Dönüş aralıkları belirlendikten sonra, ilgili aralıklarda CA ve IMM için konum hatası bileşenleri (dx, dy) ve hata normu hesaplanmış; sonuçlar Tablo 3.13 içinde raporlanmıştır.

Tablo 3.13 sonuçlarına göre IMM, hem x hem de y ekseninde daha düşük ortalama hata üretmiştir. Bu bulgu, perdeleme (occlusion) benzeri ardışık ölçüm kaybı

koşullarında IMM'in, en azından dönüş segmentlerinde, sınırlı fakat tutarlı bir doğruluk avantajı sağlayabildiğini göstermektedir.

Tablo 3.13 Ortalama konum hatası bileşenleri (dx, dy) ve normu

Model	Mean_dx	Mean_dy	Mean_norm
IMM	0.4640	0.4007	0.6546
CA	0.4952	0.4224	0.6961

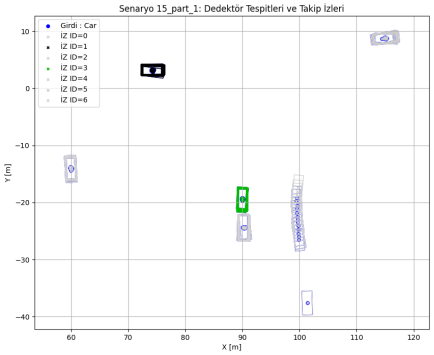
3.5.8 Bir senaryonun daha küçük alt segmenlere ayrılıp incelenmesi

Senaryo 15, sahne içerisindeki farklı hareket dinamiklerinin (düz gidiş ve dönüş manevraları gibi) takip performansına etkisini daha net incelemek amacıyla dört eşit alt segmente ayrılmıştır. Bu alt segmentlere ait nicel sonuçlar Tablo 3.14'te, dedektör tespitleri ve takip izlerinin görsel karşılaştırmaları ise Şekil 3.20'te sunulmaktadır.

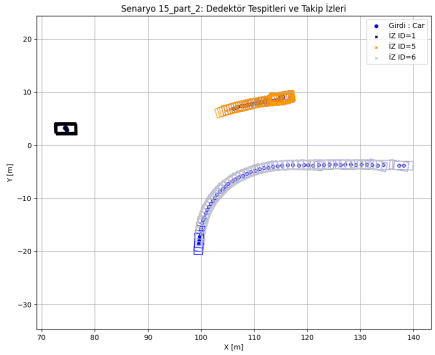
Tablo 3.14'e göre, düz giden araçların daha baskın olduğu Bölüm 1 (0–94) ve Bölüm 2 (95–188) aralıklarında CA modeli HOTA açısından öne çıkmaktadır. Buna karşılık, dönüşün daha belirgin olduğu Bölüm 3 (189–282) ve Bölüm 4 (283–376) aralıklarında ise IMM modelinin daha iyi sonuçlar verdiği görülmektedir. Bu durum, farklı hareket tiplerinde model varsayımlarının (sabit ivme ve koordineli dönüş gibi) performans üzerinde belirleyici olduğunu göstermektedir.

Tablo 3.14 Senaryo 15'in Alt Segmentlerinin Performans Metrikleri

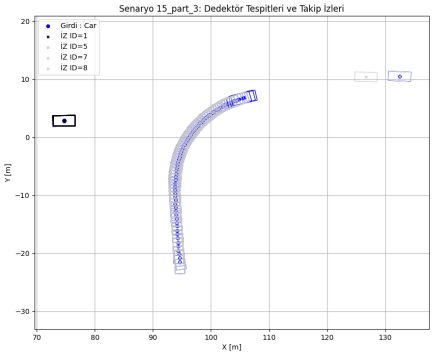
Kare Aralığı	Model	HOTA	DetA	AssA	DetRe	DetPr	AssRe	AssPr
0-94	CA	**41.791**	**25.867**	**67.644**	**26.269**	**88.56**	**70.619**	**90.879**
	CTRA	41.716	25.84	67.482	26.241	88.465	70.492	90.77
	IMM	41.767	**25.867**	67.572	**26.269**	**88.56**	70.562	90.878
95-188	CA	**36.036**	**23.705**	**54.795**	**23.904**	**90.932**	**55.735**	**91.307**
	CTRA	35.991	23.676	54.726	23.876	90.825	55.658	91.239
	IMM	36.026	23.698	54.782	23.895	90.896	55.717	91.267
189-282	CA	18.259	10.573	31.485	10.603	94.875	31.7	94.969
	CTRA	18.261	10.583	31.479	10.603	94.887	31.706	94.975
	IMM	**18.264**	**10.592**	**31.494**	**10.62**	**94.904**	**31.726**	**94.999**
283-376	CA	46.153	25.137	84.75	25.418	89.441	87.371	90.557
	CTRA	46.188	**25.158**	84.807	**25.437**	**89.507**	87.414	90.605
	IMM	**46.204**	**25.158**	**84.868**	**25.437**	**89.507**	**87.452**	**90.641**



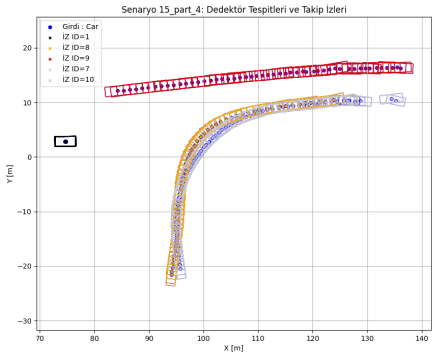
(a) Senaryo 15: Bölüm 1: Dedektör Tespitleri ve Takip İzleri



(b) Senaryo 15: Bölüm 2: Dedektör Tespitleri ve Takip İzleri



(c) Senaryo 15: Bölüm 3: Dedektör Tespitleri ve Takip İzleri



(d) Senaryo 15: Bölüm 4: Dedektör Tespitleri ve Takip İzleri

Şekil 3.20 Senaryo 15'in dört alt segmente ayrılmasıyla elde edilen dedektör tespitleri (girdi) ve takip izleri (çıktı) görselleştirmeleri.

3.5.9 Çalışma Zamanı ve Hesaplama Karmaşıklığı

Bu bölümde, yöntemlerin pratik kullanılabilirliğini değerlendirmek amacıyla çalışma zamanı ölçümleri raporlanmıştır. Çalışma zamanı, her bir senaryonun (sekansın) tamamını işlemek için takip modülünün harcadığı toplam süre (senaryo/saniye) olarak tanımlanmıştır. Her yöntem her sekans için üç kez çalıştırılmış ve sonuçlar ortalama ile örnek standart sapma (std) cinsinden özetlenmiştir.

Tablo 3.15 Çalışma zamanı (senaryo/saniye). Üç tekrar üzerinden ortalama $\pm$ std.

Sekans	**CA (s)**	**CTRA (s)**	**IMM (s)**
0012	3.228 ± 0.123	3.216 ± 0.056	3.797 ± 0.054
0015	5.938 ± 0.236	5.979 ± 0.055	8.148 ± 0.113

Tablo 3.15 sonuçları, iki modlu IMM (CA+CTRA) yaklaşımının tek-modlu filtrelere kıyasla daha yüksek toplam çalışma zamanı gerektirdiğini göstermektedir. Bu farkın temel nedeni, IMM'de her zaman adımında birden fazla model için hesap yapılması ve modlar arası etkileşim adımlarının ek maliyetidir. Ayrıca IMM'de bellek kullanımı mod sayısıyla yaklaşık doğrusal olarak artmaktadır; çünkü her mod için durum ve kovaryans ayrı ayrı tutulur.

4
SONUÇ

Bu tez çalışması, kavşak gibi yüksek karmaşıklığa sahip trafik bölgelerinde çoklu hareket modellerinin etkileşimli kullanımı ile nesne takibinin daha güvenilir ve doğru hale getirilebileceğini göstermektedir. Özellikle senaryo 0012'de yüksek ölçüm belirsizliği altında (R=50) IMM, CA'ya kıyasla HOTA skorunu 82.002'den 82.557'ye yükseltmiş ve dönüş segmentlerinde ortalama konum hatası normunu 0.6961 m'den 0.6546 m'ye düşürmüştür.

Farklı algılayıcılarla (VirConv, CASA) yapılan değerlendirmelerde nominal durumda model farkının sınırlı kaldığı (VirConv: 0.083 HOTA aralığı; CASA: 0.182 HOTA aralığı), %20 ölçüm kaybında ise IMM'in en iyi sonuçları verdiği görülmüştür (VirConv: +0.245 HOTA; CASA: +0.913 HOTA). Yüksek gürültüde (R=50) CA en yüksek HOTA'yı üretse de IMM, CTRA'ya kıyasla daha dayanıklı kalmıştır (VirConv: +1.458 HOTA; CASA: +0.783 HOTA). Bu bulgular birlikte değerlendirildiğinde, IMM'in katkısının her koşulda ortalama skoru artırmaktan ziyade belirsizlik ve eksik gözlem altında kararlılığı artırmak olduğu görülmektedir.

Bununla birlikte, sonuçların genellenebilirliği açısından iki temel kısıt bulunmaktadır. Birincisi, sensör bozunumunun gerçek sensör arıza kayıtları üzerinden değil, simülasyon yoluyla modellenmiş olmasıdır. Bu tezde bozunum; ölçüm kovaryans matrisinin artırılması ($\mathbf{R}$ büyütülmesi) ve deterministik ölçüm düşürme zamanlaması ile uygulanmıştır. Bu yaklaşım tekrarlanabilirlik sağlasa da, gerçek dünyadaki bozunum türleri (yansıma, ağır örtüşme/oklüzyon, yanlış pozitifler, senkronizasyon hataları vb.) daha karmaşık davranışlara yol açabilir. İkinci olarak, IMM yapısındaki bazı kritik parametreler (mod geçiş olasılık matrisi Π ve başlangıç mod olasılıkları μ_0) bu çalışmada sabit/heuristik olarak belirlenmiştir; bu parametrelerin veri güdümlü şekilde öğrenilmesi veya sahne bağlamına göre adaptif ayarlanması, farklı trafik koşullarında genellenebilirliği artırabilecek bir iyileştirme alanıdır.

Gelecek çalışmalarda, IMM parametreleri veri güdümlü ve çevrim içi biçimde

uyarlanabilir: ölçüm yeniliği (innovation) istatistiklerinden yararlanılarak sensör güvenilirliğindeki değişime bağlı şekilde ölçüm kovaryansı matrisi $\mathbf{R}$ (ve gerekirse süreç gürültüsü kovaryansı $\mathbf{Q}$) dinamik olarak güncellenebilir; böylece oklüzyon/ölçüm düşürme gibi durumlarda filtre tutarlılığı artırılabilir. Buna ek olarak, sabit bir mod geçiş matrisi Π kullanmak yerine, araç kinematiği, kavşak yakınlığı, şerit geometrisi/eğriliği ve çevresel etkileşim ipuçları gibi bağlam özelliklerine koşullu olarak her zaman adımında Π_k üreten hafif bir veri güdümlü model ile model geçişleri sahneye duyarlı hale getirilebilir.

KAYNAKÇA

[1] E. D. Dickmanns A. Zapp, "Guiding Land Vehicles Along Roadways by Computer Vision," *Proceedings of Congrès Automatique, AFCET*, Toulouse, France, 1985, ss. 233–244.

[2] S. Thrun ve diğ., "Stanley: The robot that won the DARPA Grand Challenge," *Journal of Field Robotics*, c. 23, no. 9, ss. 661–692, 2006. doı: 10.1002/rob.20147.

[3] A. Makarenko, B. Upcroft, M. Moser, R. Fitch, W. Uther ve diğ., "Tartan Racing: A Multi-Modal Approach to the DARPA Urban Challenge," *Proceedings of the DARPA Urban Challenge*, 2007.

[4] K. Kritayakirana, "Autonomous Vehicle Control at the Limits of Handling," Includes demonstration of autonomous Pikes Peak climb, dok. tezi, Stanford University, 2012.

[5] S. Ren, K. He, R. Girshick J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

[6] J. Long, E. Shelhamer T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[7] A. Geiger, P. Lenz R. Urtasun, "Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2012, ss. 3354–3361.

[8] X. Chen, H. Ma, J. Wan, B. Li T. Xia, "Multi-View 3D Object Detection Network for Autonomous Driving," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] C. R. Qi, H. Su, K. Mo L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[10] B. Paden, M. Cáp, S. Z. Yong, D. Yershov E. Frazzoli, "A Survey of Motion Planning and Control Techniques for Self-Driving Urban Vehicles," *IEEE Transactions on Intelligent Vehicles*, c. 1, no. 1, ss. 33–55, 2016. doı: 10.1109/TIV.2016.2578706.

[11] S. Shalev-Shwartz, S. Shammah A. Shashua, *On a Formal Model of Safe and Scalable Self-driving Cars*, arXiv preprint arXiv:1708.06374, 2017.

[12] H. Caesar, V. Bankiti, A. H. Lang, S. Vora ve diğ., "nuScenes: A Multimodal Dataset for Autonomous Driving," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[13] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard ve diğ., "Scalability in Perception for Autonomous Driving: The Waymo Open Dataset," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[14] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez V. Koltun, "CARLA: An Open Urban Driving Simulator," *Proceedings of the 1st Annual Conference on Robot Learning (CoRL)*, 2017.

[15] "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," SAE International, tek. rap. J3016_202104, 2021.

[16] J. Janai, F. Güney, A. Behl A. Geiger, "Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art," *Foundations and Trends® in Computer Graphics and Vision*, c. 12, no. 1–3, ss. 1–308, 2020. doı: 10.1561/0600000079.

[17] H. A. Ignatious, Hesham-El-Sayed M. Khan, "An overview of sensors in Autonomous Vehicles," *Procedia Computer Science*, c. 198, ss. 736–741, 2022, 12th International Conference on Emerging Ubiquitous Systems and Pervasive Networks / 11th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare. doı: 10.1016/j.procs.2021.12.315. erişim adresi: https://www.sciencedirect.com/science/article/pii/S1877050921025540.

[18] A. Bhoi, "Monocular Depth Estimation: A Survey," *CoRR*, c. abs/1901.09402, 2019. arXiv: 1901.09402. erişim adresi: http://arxiv.org/abs/1901.09402.

[19] S. Ghosh G. Gallego, "Multi-Event-Camera Depth Estimation and Outlier Rejection by Refocused Events Fusion," *Advanced Intelligent Systems*, 2022. doı: 10.1002/aisy.202200221. arXiv: 2207.10494 [cs.CV]. erişim adresi: https://arxiv.org/abs/2207.10494.

[20] Y. Salameh H. Khraisat, "Simulation of the 24GHz Short Range, Wide Band Automotive Radar," *2010 11th International Radar Symposium (IRS)*, Ocak 2010.

[21] H. Qian, M. Wang, M. Zhu H. Wang, "A Review of Multi-Sensor Fusion in Autonomous Driving," *Sensors*, c. 25, s. 6033, 2025. doı: 10.3390/s25196033.

[22] J. Redmon, S. Divvala, R. Girshick A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, ss. 779–788. doı: 10.1109/CVPR.2016.91.

[23] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang O. Beijbom, "PointPillars: Fast Encoders for Object Detection from Point Clouds," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, ss. 12 697–12 705.

[24] T. Yin, X. Zhou P. Krähenbühl, "CenterPoint: Center-based 3D Object Detection and Tracking," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, ss. 11 784–11 793.

[25] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, ss. 801–818.

[26] K. He, G. Gkioxari, P. Dollár R. Girshick, "Mask R-CNN," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, ss. 2980–2988. doı: 10.1109/ICCV.2017.322.

[27] J. Philion S. Fidler, "Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D," *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[28] J. Ku, M. Mozifian, J. Lee, A. Harakeh S. L. Waslander, "AVOD: Aggregate View Object Detection," *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[29] S. Vora, A. H. Lang, B. Helou O. Beijbom, "PointPainting: Sequential Fusion for 3D Object Detection," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[30] R. Nabati H. Qi, "CenterFusion: Center-based Radar and Camera Fusion for 3D Object Detection," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[31] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, c. 82, no. 1, ss. 35–45, 1960. doı: 10.1115/1.3662552.

[32] A. Bewley, Z. Ge, L. Ott, F. Ramos B. Upcroft, "Simple Online and Realtime Tracking," *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, ss. 3464–3468. doı: 10.1109/ICIP.2016.7533003.

[33] N. Wojke, A. Bewley D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, ss. 3645–3649. doı: 10.1109/ICIP.2017.8296962.

[34] X. Weng, E. Ohn-Bar K. M. Kitani, "AB3DMOT: A Baseline for 3D Multi-Object Tracking and New Benchmark," *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[35] H. A. P. Blom Y. Bar-Shalom, "The Interacting Multiple Model Algorithm for Systems with Markovian Switching Coefficients," *IEEE Transactions on Automatic Control*, c. 33, no. 8, ss. 780–783, Ağus. 1988. doı: 10.1109/9.1299.

[36] T. E. Fortmann, Y. Bar-Shalom M. Scheffe, "Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association," *IEEE Journal of Oceanic Engineering*, c. 8, no. 3, ss. 173–184, 1983. doı: 10.1109/JOE.1983.1145579.

[37] R. Krajewski, J. Bock, L. Kloeker L. Eckstein, "The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems," *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2018, ss. 2118–2125. doı: 10.1109/ITSC.2018.8569552.

[38] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater L. Eckstein, "The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections," *2019 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2019, ss. 1929–1934. doı: 10.1109/IVS.2019.8814098.

[39] R. Krajewski, T. Moers, J. Bock, L. Vater L. Eckstein, "The rounD Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany," *2020 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2020, ss. 224–231. doı: 10.1109/IV47402.2020.9304827.

[40] *highD Dataset (RWTH Aachen University, Institute for Automotive Engineering (ika))*, https://www.highd-dataset.com/, Accessed: 2026-02-13.

[41] W. Zhan ve diğ., "INTERACTION Dataset: An INTERnational, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps," *arXiv preprint arXiv:1910.03088*, 2019. erişim adresi: https://arxiv.org/abs/1910.03088.

[42] S. Thrun, W. Burgard D. Fox, *Probabilistic Robotics*. MIT Press, 2005, ısbn: 9780262201629.

[43] P. D. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2. bs. Artech House, 2013.

[44] S. J. Julier J. K. Uhlmann, "A New Extension of the Kalman Filter to Nonlinear Systems," *Proceedings of SPIE: Signal Processing, Sensor Fusion, and Target Recognition VI*, c. 3068, 1997, ss. 182–193.

[45] F. Dellaert, "Factor Graphs and GTSAM: A Hands-on Introduction," Georgia Institute of Technology, tek. rap., 2012.

[46] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard F. Dellaert, "iSAM2: Incremental Smoothing and Mapping Using the Bayes Tree," *The International Journal of Robotics Research*, c. 31, no. 2, ss. 216–235, 2012. doı: 10.1177/0278364911430419.

[47] T. Qin, P. Li S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, c. 34, no. 4, ss. 1004–1020, 2018. doı: 10.1109/TRO.2018.2853729.

[48] J. Zhang S. Singh, "LOAM: Lidar Odometry and Mapping in Real-time," *Robotics: Science and Systems (RSS)*, 2014.

[49] W. Hess, D. Kohler, H. Rapp D. Andor, "Real-Time Loop Closure in 2D LIDAR SLAM," *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, ss. 1271–1278. doı: 10.1109/ICRA.2016.7487258.

[50] R. W. Wolcott R. M. Eustice, "Robust LIDAR localization using multiresolution Gaussian mixture maps for autonomous driving," *The International Journal of Robotics Research*, c. 36, no. 3, ss. 292–319, 2017. doı: 10.1177/0278364917696568.

[51] M. Reda, A. Onsy, A. Y. Haikal A. Ghanbari, "Path planning algorithms in the autonomous driving system: A comprehensive review," *Robotics and Autonomous Systems*, c. 174, s. 104 630, 2024, ıssn: 0921-8890. doı: https://doi.org/10.1016/j.robot.2024.104630. erişim adresi: https://www.sciencedirect.com/science/article/pii/S0921889024000137.

[52] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, c. 1, no. 1, ss. 269–271, 1959.

[53] P. E. Hart, N. J. Nilsson B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems Science and Cybernetics*, c. 4, IEEE, 1968, ss. 100–107.

[54] S. M. LaValle, "Rapidly-Exploring Random Trees: A New Tool for Path Planning," *Technical Report TR 98-11, Computer Science Dept., Iowa State University*, 1998.

[55] S. Karaman E. Frazzoli, "Sampling-based Algorithms for Optimal Motion Planning," *International Journal of Robotics Research*, c. 30, SAGE Publications, 2011, ss. 846–894.

[56] L. Claussmann, M. Revilloud, D. Gruyer S. Glaser, "A Review of Motion Planning for Highway Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, c. 21, no. 5, ss. 1826–1848, 2020. doı: 10.1109/TITS.2019.2913998.

[57] J. González, V. Pérez, D. Milanés F. Nashashibi, "A Review of Motion Planning Techniques for Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, c. 17, no. 4, ss. 1135–1145, 2016. doı: 10.1109/TITS.2015.2498841.

[58] W. A. Farag O. Ali, "Intelligent Local Path Planning for Autonomous Driving: Integrating Quintic Splines with Predictive Control," *International Journal of Intelligent Transportation Systems Research*, c. 23, ss. 2168–2189, 2025. doı: 10.1007/s13177-025-00565-x.

[59] M. Werling, J. Ziegler, S. Kammel S. Thrun, "Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2010. doı: 10.1109/ROBOT.2010.5509799.

[60] D. Kim, G. Kim, H. Kim K. Huh, "A Hierarchical Motion Planning Framework for Autonomous Driving in Structured Highway Environments," *IEEE Access*, c. 10, ss. 20 102–20 117, 2022. doı: 10.1109/ACCESS.2022.3152187.

[61] X. Qian, I. Navarro, A. de La Fortelle F. Moutarde, "Motion Planning for Urban Autonomous Driving Using Bézier Curves and MPC," *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, ss. 826–833. doı: 10.1109/ITSC.2016.7795651.

[62] D. Fox, W. Burgard S. Thrun, "The Dynamic Window Approach to Collision Avoidance," CMU, tek. rap., 1997, IEEE Robotics Automation Magazine.

[63] Y. Gao, "Model Predictive Control for Autonomous and Semiautonomous Vehicles," dok. tezi, University of California, Berkeley, 2014.

[64] J. Kong, M. Pfeiffer, G. Schildbach F. Borrelli, "Kinematic and Dynamic Vehicle Models for Autonomous Driving," *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015.

[65] E. Yurtsever, J. Lambert, A. Carballo K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," *IEEE Access*, c. 8, ss. 58 443–58 469, 2020. doı: 10.1109/ACCESS.2020.2983149.

[66] S. Lefèvre, D. Vasquez C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *Robotics and Autonomous Systems*, c. 63, ss. 1–15, 2014.

[67] A. Rudenko, L. Palmieri, M. Herman, K. Kitani, D. M. Gavrila K. O. Arras, "Human motion trajectory prediction: A survey," *The International Journal of Robotics Research*, c. 39, no. 8, ss. 895–935, 2020.

[68] N. Deo M. M. Trivedi, "Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs," *2018 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2018, ss. 1179–1184.

[69] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr M. Chandraker, "DESIRE: Distant Future Prediction in Dynamic Scenes with Interacting Agents," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, ss. 336–345.

[70] Y. Chai, B. Sun, R. Zhao, Q. Chen D. Xu, "MultiPath: Multiple Probabilistic Anchor Trajectory Hypotheses for Behavior Prediction," *Conference on Robot Learning (CoRL)*, 2019.

[71] T. Phan-Minh, D. Bae, C. De Souza, H. Pandya ve diğ., "CoverNet: Multimodal Behavior Prediction using Trajectory Sets," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[72] X. Wang ve diğ., "MCTrack: A Unified 3D Multi-Object Tracking Framework for Autonomous Driving," *arXiv preprint arXiv:2409.16149*, 2024.

[73] S. S. Blackman R. Popoli, *Design and Analysis of Modern Tracking Systems* (Artech House radar library). Artech House, 1999, ısbn: 9781580530064. erişim adresi: https://books.google.com.tr/books?id=lTIfAQAAIAAJ.

[74] F. Pektas M. Sabuncu, "A Score-based Track Initiation Procedure to Eliminate False Tracks in the Presence of Sea Clutter," *IEEE National Radar Conference - Proceedings*, May. 2012. doı: 10.1109/RADAR.2012.6212252.

[75] S. J. Rabinowitz, *Strategies for Automatic Track Initiation: Copies of Papers and Discussions Presented at the Avionics Panel Specialists' Meeting Held in Monterey, California, U.S.A., 16–17 October 1978* (AGARD Conference Proceedings). AGARD, 1979, North Atlantic Treaty Organization (NATO), Advisory Group for Aerospace Research and Development (AGARD), Avionics Panel, ısbn: 9789283502364. erişim adresi: https://books.google.com.tr/books?id=uhMSugEACAAJ.

[76] S. S. Blackman R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, 1999, ısbn: 9781580530064. erişim adresi: https://books.google.com.tr/books?id=lTIfAQAAIAAJ.

[77] F. Pektas M. Sabuncu, "A score-based track initiation procedure to eliminate false tracks in the presence of sea clutter," *IEEE National Radar Conference - Proceedings*, May. 2012. doı: 10.1109/RADAR.2012.6212252.

[78] B.-N. Vo, B.-T. Vo D. Phung, "Multitarget Tracking," içinde *Wiley Encyclopedia of Electrical and Electronics Engineering*, cit. on pp. 27, 28, Wiley, 2015, ss. 1–25. doı: 10.1002/047134608X.W8275.

[79] P. Emami, P. M. Pardalos, L. Elefteriadou S. Ranka, "Machine learning methods for data association in multi-object tracking," *ACM Computing Surveys (CSUR)*, c. 53, no. 4, ss. 1–34, 2020.

[80] *Data Association in Multi-Object Tracking (survey), arXiv:2008.07725*, 2020. arXiv: 2008.07725 [cs.CV]. erişim adresi: https://arxiv.org/abs/2008.07725.

[81] R. Selvan, "Bayesian tracking of multiple point targets using Expectation Maximization," (cit. on p. 29), dok. tezi, University of Copenhagen, Tem. 2015.

[82] R. Collins, *Lecture Notes: Data Association*, https://www.cse.psu.edu/~rtc12/CSE598C/datassocPart1.pdf, Accessed: 2025-06-22, 2012.

[83] J. McMillan S. S. Lim, *Data Association Algorithms for Multiple Target Tracking*, Unpublished manuscript or internal report, p. 37 (cited on p. 32), Tem. 1990.

[84] D. B. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Transactions on Automatic Control*, c. 24, no. 6, ss. 843–854, 1979. doı: 10.1109/TAC.1979.1102177.

[85] J. Z. Sasiadek, "Sensor fusion," *Annual Reviews in Control*, c. 26, no. 2, ss. 203–228, 2002, cit. on p. 24, ıssn: 1367-5788. doı: 10.1016/S1367-5788(02)00045-7. erişim adresi: https://www.sciencedirect.com/science/article/pii/S1367578802000457.

[86] H. Wu, C. Wen, S. Shi, X. Li C. Wang, "Virtual Sparse Convolution for Multimodal 3D Object Detection," *arXiv preprint arXiv:2303.02314*, 2023. arXiv: 2303.02314. erişim adresi: https://arxiv.org/abs/2303.02314.

[87] H. Wu, J. Deng, C. Wen, X. Li C. Wang, "CasA: A Cascade Attention Network for 3D Object Detection from LiDAR point clouds," *IEEE Transactions on Geoscience and Remote Sensing*, 2022.

[88] J. Luiten ve diğ., "HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking," *International Journal of Computer Vision (IJCV)*, 2020. doı: 10.1007/s11263-020-01375-2.

[89] K. Bernardin R. Stiefelhagen, "Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics," *EURASIP Journal on Image and Video Processing*, c. 2008, Hindawi, 2008, ss. 1–10. doı: 10.1155/2008/246309.

[90] J. Luiten ve diğ., "HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking," *International Journal of Computer Vision*, c. 129, no. 2, ss. 548–578, 2021. doı: 10.1007/s11263-020-01375-2.

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirisi

1. Hakkoymaz, Ç., Uslu, E.; “Improve Performance on Junction Scenarios via IMM Filtering”, Grad Symposium 2025 – Mühendislikte Sürdürülebilir Çözümler, Yıldız Teknik Üniversitesi, Davutpaşa Kongre ve Kültür Merkezi, 15 Mayıs 2025.