



**KAMERA GÖRÜNTÜLERİNDEN OTONOM BİR  
ARACIN KONUMUNUN VE YÖNÜNÜN  
BELİRLENMESİ**

**Nurgüneş ÖNAL**

**Yüksek Lisans Tezi**

**Elektrik-Elektronik Mühendisliği Ana Bilim Dalı**

**2020**

(Her hakkı saklıdır.)

T.C.  
ATATÜRK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**KAMERA GÖRÜNTÜLERİNDEN OTONOM BİR ARACIN KONUMUNUN VE  
YÖNÜNÜN BELİRLENMESİ**

(Determining The Position and Orientation of an Autonomous Vehicle From The Camera  
Images)

YÜKSEK LİSANS TEZİ

Nurgüneş ÖNAL

Danışman: Prof. Dr. Bülent ÇAVUŞOĞLU

Erzurum  
Ağustos, 2020

## KABUL VE ONAY TUTANAĞI

Nurgüneş Önal tarafından hazırlanan “Kamera Görüntülerinden Otonom Bir Aracın Konumunun ve Yönünün Belirlenmesi” başlıklı çalışması / .. / 20.. tarihinde yapılan tez savunma sınavı sonucunda başarılı bulunarak jürimiz tarafından Elektrik Elektronik Mühendisliği Ana Bilim Dalı, Haberleşme Bilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Jüri Başkanı: Doç. Dr. İ. Yücel Özbek  
*Atatürk Üniversitesi* .....

Danışman: Prof. Dr. Bülent ÇAVUŞOĞLU  
*Atatürk Üniversitesi* .....

Jüri Üyesi: Dr. Öğr. Üyesi Nurbanu Güzey  
*Erzurum Teknik Üniversitesi* .....

Enstitü Yönetim  
Kurulunun .../.../....  
tarih ve ..... sayılı  
kararı.

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili maddelerinde belirtilen şartları yerine getirdiğini onaylarım.

**Prof. Dr. Mehmet KARAKAN**  
**Enstitü Müdürü**

**Not:** Bu tezde kullanılan özgün ve başka kaynaklardan yapılan bildiriş, çizelge, şekil ve fotoğrafların kaynak olarak kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunundaki hükümlere tabidir.

## ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU

Yüksek Lisans Tezi olarak Prof. Dr. Bülent ÇAVUŞOĞLU danışmanlığında sunulan “Kamera Görüntülerinden Otonom Bir Aracın Konumunun ve Yönünün Belirlenmesi” başlıklı çalışmanın tarafımızdan bilimsel etik ilkelere uyularak yazıldığını, yararlanılan eserlerin kaynakçada gösterildiğini, Fen Bilimleri Enstitüsü tarafından belirlenmiş olan Turnitin Programı benzerlik oranlarının aşılmadığını ve aşağıdaki oranlarda olduğunu beyan ederiz.

Tez Bölümleri	Tezin Benzerlik Oranı (%)	Maksimum Oran (%)
Giriş	%0	30
Kuramsal Temeller	%1	30
Materyal ve Yöntem	%2	35
Bulgular ve Tartışma	%13	20
Sonuç ve Öneriler	%0	20
Tezin Geneli	%5	25

*Not: Yedi kelimeye kadar benzerlikler ile Başlık, Kaynakça, İçindekiler, Teşekkür, Dizin ve Ekler kısımları tarama dışı bırakılabilir. Yukarıdaki azami benzerlik oranları yanında tek bir kaynaktan olan benzerlik oranlarının %5'den büyük olmaması gerekir.*

Beyan edilen bilgilerin doğru olduğunu, aksi halde doğacak hukuki sorumlulukları kabul ve beyan ederiz.

Tez Yazarı (Öğrenci)	Tez Danışmanı
Nurgüneş ÖNAL	Prof. Dr. Bülent ÇAVUŞOĞLU
28.7.2020	28.7.2020
İmza:	İmza:

\* Tez ile ilgili YÖKTEZ’de yayınlamasına ilişkin bir engelleme var ise aşağıdaki alanı doldurunuz.

Tezle ilgili patent başvurusu yapılması / patent alma sürecinin devam etmesi sebebiyle Enstitü Yönetim Kurulunun ....../....../.... tarih ve ..... sayılı kararı ile teze erişim 2 (iki) yıl süreyle engellenmiştir.

Enstitü Yönetim Kurulunun ....../....../.... tarih ve ..... sayılı kararı ile teze erişim 6 (altı) ay süreyle engellenmiştir.

## TEŐEKKÜR

Yüksek lisans tez çalışmamın yürütülmesinde yol gösteren, bilgi ve tecrübelerini benden esirgemeyen danışman hocam Sayın Prof. Dr. Bülent ÇAVUŐOĐLU'na emeklerinden dolayı teşekkür ederim.

Tez çalışmam boyunca vermiş oldukları desteklerden dolayı arkadaşlarıma ve hayatımın her aşamasında yanımda olan aileme şükranlarımı sunarım.

Nurgüneő ÖNAL



## ÖZET

### YÜKSEK LİSANS TEZİ

## KAMERA GÖRÜNTÜLERİNDEN OTONOM BİR ARACIN KONUMUNUN VE YÖNÜNÜN BELİRLENMESİ

Nurgüneş ÖNAL

Danışman: Prof. Dr. Bülent ÇAVUŞOĞLU

**Amaç:** Stereo kamera ve özellik tabanlı görsel odometri yöntemi kullanılarak, otonom araçlar için görsel girdiden hareket yörüngesini tahmin etme problemi ele alınarak etkili bir algoritmanın kurulması ve algoritma aşamalarının yörünge tahminindeki başarıya etkilerinin değerlendirilmesi amaçlanmıştır.

**Yöntem:** Temel yapısı iki görüntü arasında özellik noktası takibi, 3B model ve 2B görüntüler arasında örtüşmeyi kurarak poz tahmini ve yeni özellikleri üçgenleme yapmaya dayanan özellik tabanlı bir görsel odometri algoritması sunulmuştur. Algoritma ayrıca aykırı değerlere karşı sağlamlığı artırmak için M-Kestiricisi Örnek Konsensüsü (M-SAC) kullanır. Algoritmanın sonuçlarını değerlendirmek amacıyla KITTI veri seti kullanılmıştır. KITTI veri seti otonom sürüş algoritmalarının değerlendirilmesinde en sık kullanılan veri kümelerinden birini temsil etmektedir.

**Bulgular:** Literatürde yapılan Mittal (2018)'in çalışması, tezde kullanılan algoritma ile karşılaştırılmış ve tezde kullanılan algoritmanın kesin yer referansını daha doğru bir şekilde kestirdiği gösterilmiştir.

**Sonuç:** Bu tez kapsamında elde edilen sonuçlar değerlendirildiğinde dikdörtgen (bucket) büyüklüğü, özellik sayısı gibi unsurların yön takibi başarısında önemli bir rol oynadığı tespit edilmiştir. Veri setinin özelliklerine göre önerilen algoritmaların belirli bir uyarlama sürecinden sonra kullanılmasının veya uyarlanabilir algoritmalarla birlikte entegre edilmesinin daha faydalı olabileceği değerlendirilmektedir.

**Anahtar Kelimeler:** görsel odometri, poz kestirimi, çoklu görüş geometrisi, fotogrametri, bilgisayarlı görme, görüntü işleme

Ağustos 2020, 64 Sayfa

## ABSTRACT

### MASTER'S THESIS

#### DETERMINING THE POSITION AND ORIENTATION OF AN AUTONOMOUS VEHICLE FROM THE CAMERA IMAGES

Nurgüneş ÖNAL

**Supervisor: Prof. Dr. Bülent ÇAVUŞOĞLU**

**Purpose:** It is aimed to establish an effective algorithm and evaluate the effects of algorithm steps on success in trajectory prediction by addressing the problem of predicting motion trajectory from visual input for autonomous vehicles by using stereo camera and feature-based visual odometry method.

**Method:** A feature-based visual odometry algorithm, whose basic structure is based on feature point tracking between two images, pose estimation and triangulation of new features by establishing the overlap between 3D model and 2D images, is presented. The algorithm also uses the M-Estimator Sample Consensus (M-SAC) to increase robustness against outliers. KITTI data set is used to evaluate the results of the algorithm. The KITTI dataset represents one of the most frequently used datasets in the evaluation of autonomous driving algorithms.

**Findings:** The study of Mittal (2018) in the literature has been compared with the algorithm used in the thesis, and it has been shown that the algorithm used in the thesis predicted the exact location reference more accurately.

**Results:** As the results obtained within the scope of this thesis are evaluated, it has been determined that factors such as the size of the rectangle (bucket) and the number of features play an important role in the success of following the direction. It is considered that it may be more beneficial to use the proposed algorithms after a certain adaptation process or to integrate them with adaptive algorithms based on the characteristics of the data set.

**Keywords:** visual odometry, ego-motion estimation, multi view geometry, photogrammetry, computer vision, image processing

**August 2020, 64 pages**

## İÇİNDEKİLER

KABUL VE ONAY TUTANAĞI.....	i
ETİK BİLDİRİM VE İNTİHAL BEYAN FORMU .....	ii
TEŞEKKÜR .....	iii
ÖZET .....	iv
ABSTRACT .....	v
İÇİNDEKİLER.....	vi
TABLolar DİZİNİ.....	viii
ŞEKİLLER DİZİNİ .....	ix
KISALTMALAR DİZİNİ .....	xi
GİRİŞ.....	1
KURAMSAL TEMELLER.....	5
Görüntü Oluşumu.....	5
Homojen Koordinat Sistemi.....	6
Pinhole Kamera.....	7
Pinhole Kamera Genelleştirilmiş Model.....	9
Stereo Vizyon.....	12
Epipolar Geometri.....	13
Üçgenleme .....	17
Görsel Odometri Yöntemleri .....	18
Özellik Tabanlı Yöntemler.....	19
MATERYAL ve YÖNEM .....	25
Özellik Seçimi ve Takibi.....	25
Özellik İşleme .....	25
Özellik Eşleştirme .....	28
Özellik Seçimi.....	30
Kamera Hareketinin Tahmini.....	30
Üçgenleme .....	31
P3P Algoritması .....	31
RANSAC ve MSAC .....	32
ARAŞTIRMA BULGULARI ve TARTIŞMA .....	35
Doğruluk Belirleme.....	35



KITTI Veri Seti .....	37
KITTI Veri Seti Deęerlendirmesi .....	38
SONUÇ ve ÖNERİLER .....	45
KAYNAKLAR .....	47
ÖZGEÇMİŞ .....	51



## TABLolar DİZİNİ

<b>Tablo 1.</b> KITTI veri setinde görsel odometri sonuçları ve hata karşılaştırması.....	43
<b>Tablo 2.</b> KITTI dizi-04 hata karşılaştırması .....	44



## ŞEKİLLER DİZİNİ

Şekil 1. Pinhole kamera modeli.....	6
Şekil 2. Paralel çizgilerin sonsuzda bir noktada kesişmesi .....	6
Şekil 3. Pinhole kamera yaygın gösterimi .....	8
Şekil 4. Kamera ve dünya referans çerçeveleri arasında koordinat dönüşümü.....	10
Şekil 5. Dış ve iç parametreler ile referans çerçeve dönüşümleri .....	12
Şekil 6. Stereo kamera örneği.....	12
Şekil 7. Stereo vizyon.....	13
Şekil 8. Epipolar geometri.....	14
Şekil 9. Epipolar doğru demetleri .....	14
Şekil 10. İdeal durumda üçgenleme .....	17
Şekil 11. Üçgenlemenin gerçek geometrik yorumu .....	17
Şekil 12. Görsel odometri temel adımlar .....	19
Şekil 13. Algoritmanın genel akış şeması .....	25
Şekil 14. Blob ve köşe detektör kernelleri .....	26
Şekil 15. Köşe ve blob özellikleri .....	28
Şekil 16. SAD örneği .....	29
Şekil 17. Dairesel Eşleme.....	29
Şekil 18. P3P Geometrisi .....	31
Şekil 19. RANSAC hata tolerans eşiğine göre uyumlu noktaların belirlenmesi.....	34
Şekil 20. Göreceli poz hatası.....	36
Şekil 21. KITTI- dizi 07’de bulunan sağ kameradan alınan 121.görüntü.....	38
Şekil 22. KITTI- dizi 07’de bulunan sol kameradan alınan 121.görüntü.....	39
Şekil 23. 121. Görüntüde dikdörtgenler kullanarak özellik seçimi; Kırmızı işaretler zayıf özellikleri, yeşil işaretler güçlü özellikleri temsil etmektedir. ....	39
Şekil 24. Sol kamerada 120. ve 121. görüntüler arasında özellik eşleşmesi; Yeşil işaretler t anını ve kırmızı işaretler t-1 anını temsil etmektedir.....	39
Şekil 25. KITTI dizi-05 .....	40
Şekil 26. KITTI dizi-07 .....	40
Şekil 27. KITTI dizi-09 .....	41
Şekil 28. KITTI dizi-10 .....	41

<b>Şekil 29.</b> Kesin yer referansının, tezde sunulan algoritmanın sonucunun ve Mittal (2018)'e ait algoritmanın sonucunun birlikte çizdirilmesi .....	42
<b>Şekil 30.</b> KITTI-04 dizisinde 270. görüntü .....	43
<b>Şekil 31.</b> KITTI dizi-04 .....	44



## KISALTMALAR DİZİNİ

ATE	: Absolute Translation Error (Mutlak Translasyon Hatası)
CNN	: Convolutional Neural Network (Evrışimsel Sinir Ağları)
DoG	: Difference of Gaussian (Gauss'ların Farkı)
EKF	: Extended Kalman Filter (Genişletilmiş Kalman Filtresi)
GPS	: Global Positioning System (Küresel Konum Belirleme Sistemi)
IMU	: Inertial Measurement Unit (Ataletsel Ölçüm Birimi)
INS	: Inertial Navigation System (Atalet Seyrüsefer Sistemi)
KITTI	: Karlsruhe Institute of Technology and Toyota Institute of Technology (Karlsruhe Teknoloji Enstitüsü ve Toyota Teknoloji Enstitüsü)
LIDAR	: Light Detection and Ranging (Işık ile Tespit Etme ve Menzil Tayini)
M-SAC	: M Estimator Sample and Consensus (M- kestiricisi Örnek ve Konsensüs)
P3P	: Perspective 3 Point (Perspektif 3 Nokta)
PnP	: Perspective n Point (Perspektif n Nokta)
RADAR	: Radio Detection and Ranging (Radyo ile Tespit Etme ve Menzil Tayini)
RANSAC	: Random Sample Consensus (Rastgele Örnek Konsensüs)
RGB	: Red Green Blue (Kırmızı Yeşil Mavi)
RMSR	: Root Mean Square Error (Kök Ortalama Kare Hatası)
RPE	: Relative Pose Error (Göreceli Poz Hatası)
SAD	: Sum of Absolute Differences (Mutlak Farkların Toplamı)
V-SLAM	: Visual Simultaneous Localization and Mapping (Görsel Eş Zamanlı Lokalizasyon ve Haritalama)

## GİRİŞ

Otonom araçların durmakta olan bir noktaya göre yaptığı yer değiştirmeyi tahmin etmek, modern otonom sürüş uygulamaları için en önemli görevlerden biridir. Navigasyon, hareket takibi, engel tespiti, algılama, planlama ve kontrol için, otonom araç zaman içindeki konumu ve yönü hakkında bilgi sahibi olmalıdır. Sensörler, otonom araçların konumunu doğru bir şekilde tahmin etmesi için gereken temel bileşenlerdir. Günümüzde, küresel konumlandırma sistemi (GPS), Ataletsel ölçüm birimi Sistemi (IMU), LIDAR, RADAR, sonar, tekerlek odometrisi ve kameralar gibi çeşitli sensörler bu amaç için çeşitli uygulamalarda öne çıkmaktadır.

GNSS (*Global Navigation Satellite Systems*), mevcut ve gelecekteki tüm küresel navigasyon sistemleri için kullanılan ortak bir terimdir. Bu sistemler, üçgenleme kullanarak bilinen konumlara sahip birkaç görünür uyduyu seçerek alıcının konumunu enlem, boylam ve yükseklik olarak hesaplar (Noureldin *et al.* 2013). Bu sistemlerin içinde GPS (*Global Positioning System*) en çok kullanılanıdır. GPS, bilinen bir hata oranıyla ve mutlak bir doğrulukla konumu belirler (Aqel *et al.* 2016). Ancak, söz konusu konum bilgilerini, yeryüzündeki alıcı en az dört GPS uydusunu engelsiz görebildiğinde elde edebilir (Zekavat *et al.* 2019). Ancak, GPS alıcıları, kentsel alanlarda, tünellerde ve kapalı alanlarda uydular ile iletişimini kaybedebilir (Mohamed *et al.* 2019). Ayrıca, uydulardan gelen sinyallerin bir kısmı bina, ağaç veya herhangi bir yansıtıcı yüzeyden yansıdıktan sonra alıcıya gelebilir (Kuutti *et al.* 2018). Buna çok yollu iletim etkisi denir ve alıcıya yansiyarak gelen sinyal doğrudan gelen sinyalden daha uzun bir yol aldığı için konum doğruluğunu etkiler (Kos *et al.* 2010).

INS (Inertial Navigation System), bir nesnenin bilinen bir başlangıç noktasına, yönüne ve hızına göre konumunu, yönünü ve hızını sürekli olarak hesaplamak için bilgisayar, hareket sensörleri (ivme ölçerler) ve dönüş sensörleri (jiroskoplar) kullanan navigasyon yardımcısıdır (Wang *et al.* 2014). Dezavantajı ölçümlerindeki hataların zaman geçtikçe artarak birikmesidir. Bu nedenle, konumun GPS gibi başka bir navigasyon sisteminin girişi ile periyodik olarak düzeltilmesi gerekir (Maklouf *et al.* 2014). Kalman filtresi, INS'ten alınan verileri kullanarak GPS'ten gelen gürültülü verileri filtrelemek için kullanılabilir. Zhang *et al.* (2012), GPS'deki düşük doğruluğu ve IMU sensöründeki kümülatif hataları iyileştirmek için yaptıkları sensör füzyonu çalışmasında, GPS ve IMU'dan gelen bilgileri entegre etmişlerdir. Yöntem, tahmin edilen konum doğruluğunu artırmada sadece GPS veya sadece IMU sensörünün sağladığı

konum hesaplarına göre başarılı olmuştur ancak sistemde biriken hatalar varlığını sürdürmeye devam etmiştir.

Ultrasonik sensörler, ses dalgaları kullanarak nesnelere algılamak ve bir nesneye olan mesafeyi ölçmek için kullanılan düşük maliyetli sensörlerdir (Rhee and Seo 2019). Bu tür sensörler bir ultrasonik dalga vericisi ve alıcısından oluşur. Verici ultrasonik ses dalgası gönderir ve bu dalga yakındaki nesnelere yansdıktan sonra alıcı tarafından geri alınır (Aqel *et al.* 2016). Sensör, dalga iletiminden alımına kadar geçen süre olan uçuş süresini ölçer ve ses hızıyla dalga yayılma süresini çarparak nesnelere olan mesafeyi hesaplar (Rhee and Seo 2019). Ultrasonik sensörler, iki farklı pozda elde edilen her iki sensör girişi arasındaki poz değişikliğini hesaplayarak üçgenleme yoluyla otonom araçları lokalize etmek için kullanılabilir (Aqel *et al.* 2016). Fakat daha çok, otonom araç lokalizasyonunda engelleri, aracın kaldırım kenarlarına olan mesafesini belirlemek gibi uygulamalarda yardımcı olarak kullanılır. Bu sensörlerin algılama aralığı 3 m ile sınırlıdır ve bu da otonom bir aracın engel algılama sistemi için yeterli değildir (Kuutti *et al.* 2019). Dolayısıyla, ultrasonik sensörler otonom araç lokalizasyonu için tek başlarına yeterince güçlü değildirler fakat diğer sistemlerle veya sensörlerle birlikte kullanıldıklarında etkili bir yöntem elde edilebilir.

LIDAR (Işık algılama ile menzil sensörü) ve RADAR (Radyo Algılama ile Menzil Sensörü), cisimlerinin yerini, büyüklüğünü ve hızını tespit etmek için otonom araçların lokalizasyonunda engel algılama ve engelden kaçma amaçlı yardımcı eleman olarak kullanılabilir (Woo *et al.* 2019). LIDAR, ultrasonik sensörlere benzer şekilde çalışır ancak ses dalgaları yerine ışık, yani lazer darbeleri kullanır (Kocic *et al.* 2018). IMU, GPS ve kamera ile entegre edildiğinde otonom aracın bulunduğu çevrenin 3B haritasını çıkarmak ve otonom aracı haritada bulmak için de kullanılır. RADAR ise LIDAR'a benzer bir ilke ile çalışır, ancak lazer darbeleri yerine, radyo dalgaları yayar. RADAR' lar daha ucuzdur ve LIDAR'lardan daha uzun mesafelerde nesnelere algılayabilir, ancak doğruluğu LIDAR'a göre daha az veri topladığı için daha düşüktür (Kuutti *et al.* 2019). Bununla birlikte, LIDAR teknolojisi diğer algılama teknolojilerine kıyasla çok daha pahalıdır ve performansı yağmur veya kar gibi çevresel koşullara duyarlıdır (Yurtsever vd 2020). Her iki teknoloji de otonom araç lokalizasyonunda kameralar ve ultrasonik sensörlerle birlikte kullanıldığında daha iyi bir sonuç verir.

Odometri, otonom aracın başlangıç noktasına göre zaman içindeki konum değişikliğini tahmin etmek için hareket sensörlerinden gelen verilerin kullanılmasıdır (Taketomi *et al.* 2017). Tekerlek odometrisi (kilometre sayacı), otonom araçların konumunu ve hızını tahmin etmek için kullanılan temel yöntemlerden biridir. Bu yöntem, zeminle temas halinde olan tekerleklerin devir sayısını izlemek için, araç üzerine monte edilen tekerlek kodlayıcılarına dayanır (Aqel *et*

al. 2016). Devir sayısı, robotun başlangıç noktasına göre mevcut konumunu belirlemek için kullanılır. Her yeni pozisyon bir önceki pozisyona dayandığı için hesaplanan her yeni konumda hata gittikçe birikir ve bu da uzun mesafeler için tahminin gerçek konumdan sapmasına neden olur (Yousif *et al.* 2015). Dolayısıyla, tekerlek odometrisi basit ve ucuz bir lokalizasyon tekniği olsa da, tek başına uzun vadeli güvenilir bir lokalizasyon sistemini gerektiren uygulamalar için uygun değildir (Mohamed *et al.* 2019).

Görsel odometri (*Visual Odometry*), otonom araç üzerine yerleştirilmiş bir veya daha fazla kameradan gelen görüntülerden faydalanarak, aracın konumunu ve yönünü tahmin etmek için kullanılan bir yöntemidir (Scaramuzza and Fraundorfer 2011). Genel olarak yöntem, görüntüler arasındaki belirli özelliklerin fiziksel konumlarının takibine dayanır (Delmas and Gee 2019). Bilinen ilk gerçek zamanlı görsel odometri yöntemini Moravec (1980) yaptığı tezde ele almıştır. 1980-2000 yılları arasında, NASA 2004 Mars Misyonuna hazırlanırken görsel odometri araştırmalarına ağırlık vermiştir ve NASA, Mars keşif programında görsel odometriyi başarıyla kullanmıştır (Cheng *et al.* 2006). Nister *et al.* (2004) tarafından yapılan çalışmada görsel odometri terimi ilk defa kullanılmıştır. Görsel odometri giderek daha fazla ilgi görmüştür ve farklı çalışmalarda uygulanabilmesi için araştırma konusu olmuştur.

Görsel odometri hareketten yapı (Structure from Motion-SfM) olarak adlandırılan bir tekniğin özel bir durumudur ve temelini bu yöntemden alır. SfM, sıralı ya da sırasız görüntü dizilerinden, kameranin hareketini ve çevrenin 3B yapısını çıkarmak için kullanılan bir tekniktir ve gerçek zamanlı olarak çalışmaz (Woo *et al.* 2019). Görsel odometri ise, kamera pozlarını tahmin etmenin yanı sıra, kamera pozlarını sırasıyla ve gerçek zamanlı olarak tahmin etmeye odaklanır (Janai *et al.* 2019). Yöntem, çevrenin yapısı, engel tespiti, araç lokalizasyonu ve 3B yeniden yapılandırma (rekonstrüksiyon) gibi uygulamalar için kullanışlıdır (Poddar *et al.* 2018). Bir diğer teknik olan Görsel Eş Zamanlı Haritalama ve Lokalizasyon (*Visual Simultaneous Localization and Mapping-VSLAM*) ile görsel odometri son derece birbiriyle ilgili tekniklerdir. V-SLAM yönteminin amacı, çevrenin haritasını çıkarırken aynı zamanda kendisini bu harita içinde lokalize etmektir (Li *et al.* 2019). Görsel odometri ve V-SLAM arasındaki temel fark, ilki bölgesel tutarlılığa odaklanıp kameranin hareketini pozdan poza kademeli olarak tahmin etmeye çalışırken, ikincisi kameranin yörüngesini ve haritayı genel tutarlılıkla tahmin etmeyi amaçlar (He *et al.* 2020). Kameranin gittiği yolun genel tutarlılığı bazı uygulamalar için tercih edilebilir olsa da görsel odometri kameranin önceki tüm geçmişini takip etmeye gerek kalmadan, gerçek zamanlı performans için uygulamadaki sadelikle beraber bölgesel tutarlılığı, genel tutarlılığa tercih eder (Scaramuzza ve Fraundorfer 2011). Ayrıca, V-SLAM tekniği, görsel odometriye mevcut kamera görüntülerinden daha önce ziyaret edilen bir



yeri tanınması için ortam algılama (*loop closure detection*) algoritması eklenerek oluşturulabilir (Taketomi *et al.* 2017).

Görsel odometri, tekerlek odometrisi ile karşılaştırıldığında %0,1 ile %2 arasında değişen bir hata ile konumu daha doğru tahmin eden bir tekniktir (Scaramuzza and Fraundorfer 2011). Ayrıca bu teknik, iç mekânlar, kentsel alanlar ve gezegenler gibi küresel konumlandırma sisteminin (GPS) mevcut ya da yeterli olmadığı durumlarda çalışabilir (Mohamed *et al.* 2019). Görsel odometri, benzer görevleri yerine getirebilen LIDAR tabanlı lokalizasyona kıyasla çok daha ucuz bir çözümdür (Woo *et al.* 2019). Diğer lokalizasyon yöntemlerinden farklı olarak, görsel odometri, konumu ve yönü doğru bir şekilde kestirebilmek için, çevrede yeterli aydınlatmanın varlığına ihtiyaç duyar. Yöntem, yüksek doğruluk, düşük maliyet ve bol miktarda veri bilgisi avantajlarına sahiptir. Bu nedenlerden dolayı, otonom bir aracın konumunu ve yönünü belirlemek için görsel odometri kullanmak aktif bir araştırma alanıdır ve kentsel otonom sürüş araçlarında önemli bir rol oynar.

Bu tezde özellik tabanlı bir görsel odometri algoritması sunulacaktır. Algoritma, gerçek ölçekte kestirimi ve sistemin kolay başlatılmasını sağlayan bir stereo kamera kullanır. Bu tezin amacı, otonom bir aracın sadece 2B bilgi taşıyan görüntüleri, kendisini ve çevresini 3B alanda bulmak için nasıl kullanabileceğini göstermektir. Algoritmanın performansı, otonom sürüş algoritmalarının değerlendirilmesinde yaygın olarak kullanılan KITTI (Geiger *et al.* 2012) veri seti üzerinde değerlendirilecektir.

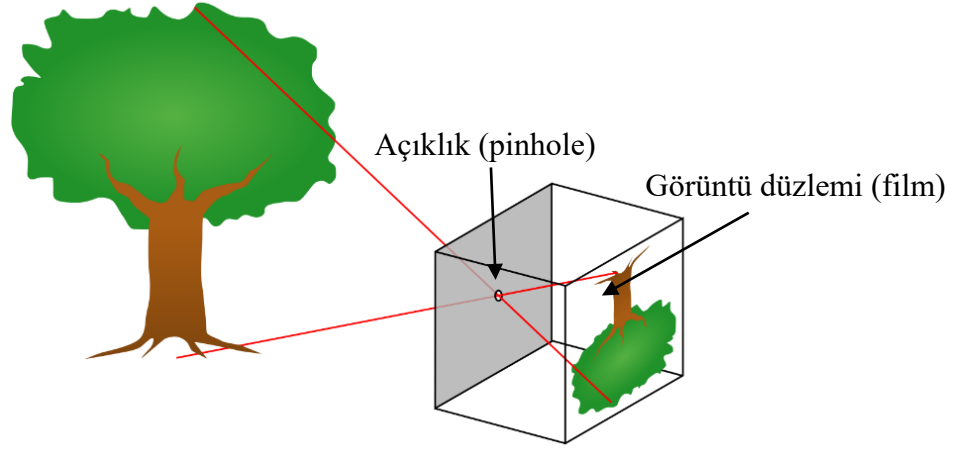
## KURAMSAL TEMELLER

Görsel odometri bir aracın yalnızca kendisine bağlı tek veya birden fazla kameranın girişini kullanarak konumunu ve yönünü tahmin etme işlemidir (Scaramuzza and Fraundorfer 2011). Görsel odometri yaklaşımları, bir kamera kullanıldığında monoküler görsel odometri ve birden fazla kamera kullanıldığında stereo görsel odometri olarak sınıflandırılır. Monoküler kamera, görsel odometri, robotik ve bilgisayarlı görme uygulamalarında zorlu bir problem olarak bilinir. Monoküler bir görsel odometri sisteminin temel zorluğu, özellik derinliklerinin (feature depths) ve ölçeğin (*scale*) ölçülemez olması ve tahmin edilmesi gerektiğidir (Woo *et al.* 2018). Stereo kamera tabanlı görsel odometri yöntemlerinde, ölçeği tahmin etmek için stereo kamera donanımının kameraları arasındaki uzaklık olarak bilinen taban çizgisinden (*base-line*) yararlanır. Bu nedenle stereo kamera tabanlı yöntemler, monoküler kamera tabanlı yöntemlerden daha iyi bir performans gösterirler (Hartley and Zisserman 2003).

Görsel odometrinin nihai hedefi, bilinmeyen bir ortamı keşfederken aracın hareket yörüngesini tahmin etmektir.

### Görüntü Oluşumu

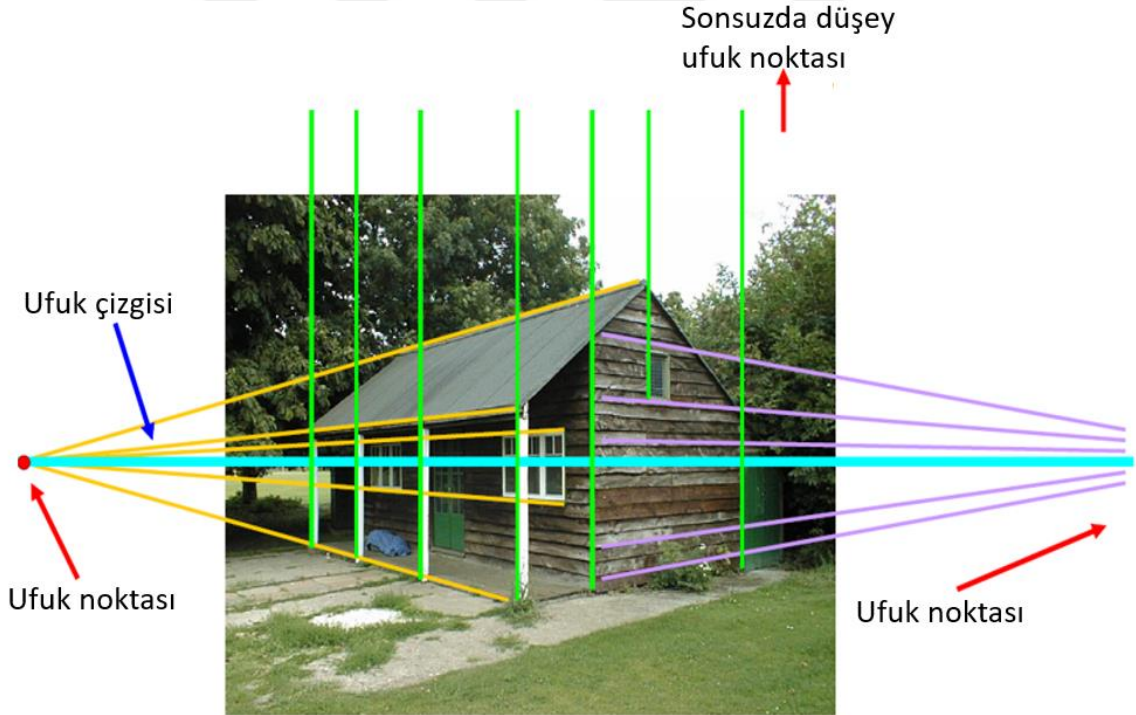
Görsel odometride sunulan yöntemleri anlamak için ilk olarak kameranın geometrisini ve görüntülerin nasıl oluştuğunu anlamak gerekir. Pinhole kamera (iğne deliği kamera) fotoğrafın icadına yol açan tarihteki ilk kamera örneğidir ve görüntü oluşumunun anlaşılması için en temel modeldir. Şekil 1’de gösterildiği gibi pinhole kamerada, ışık ışınları karanlık bir odanın bir tarafındaki küçük bir açıklıktan geçer (*pinhole, aperture*) ve görüntünün oluştuğu görüntüleme düzlemine çarpar. Açıklığın, görüntü düzleminde elde edilen görüntünün keskin olması için son derece küçük olduğu varsayılır. Kamera açıklığından geçen ışık ışınlarının bir sonucu olarak oluşan görüntü, gerçek sahneye göre ters çevrilmiş olarak elde edilir (Hartley and Zisserman 2003).



Şekil 1. Pinhole kamera modeli (Mellish 2005)

### Homojen Koordinat Sistemi

Öklid uzayında, aynı düzlemdeki iki paralel çizgi sonsuzda dahi kesişmezler. Fakat Şekil 2'deki görüntüde birbirine paralel olan çizgiler sonsuzda bir nokta olan ufuk çizgisinde kesişiyormuş gibi görünür.



Şekil 2. Paralel çizgilerin sonsuzda bir noktada kesişmesi (Scaramuzza 2019)

Öklid uzay (kartezyen uzay) 2B/3B geometriyi açıklar fakat iz düşümsel geometriyi (*projective geometry*) ve perspektifin görsel etkisini açıklamakta yetersiz kalır. Homojen koordinat sistemi iz düşümsel geometriyi açıklamayı mümkün kılar ve N-boyutlu koordinatları N+1 tane bileşenden oluşturarak boyutunun artmasını sağlar. Denklem (1)' de, kartezyen

koordinat sisteminde  $(X, Y)$  ile gösterilen bir nokta, homojen koordinat sisteminde, üçüncü bir koordinat eklenerek  $(x, y, w)$  halini alır.

$$(X, Y) \Rightarrow (x, y, w) \quad (1)$$

Denklem (2) ve denklem (3)'te Homojen koordinattaki tüm bileşenler en son eklenen bileşene bölünürse, Kartezyen koordinat sistemine geri dönmüş olur.

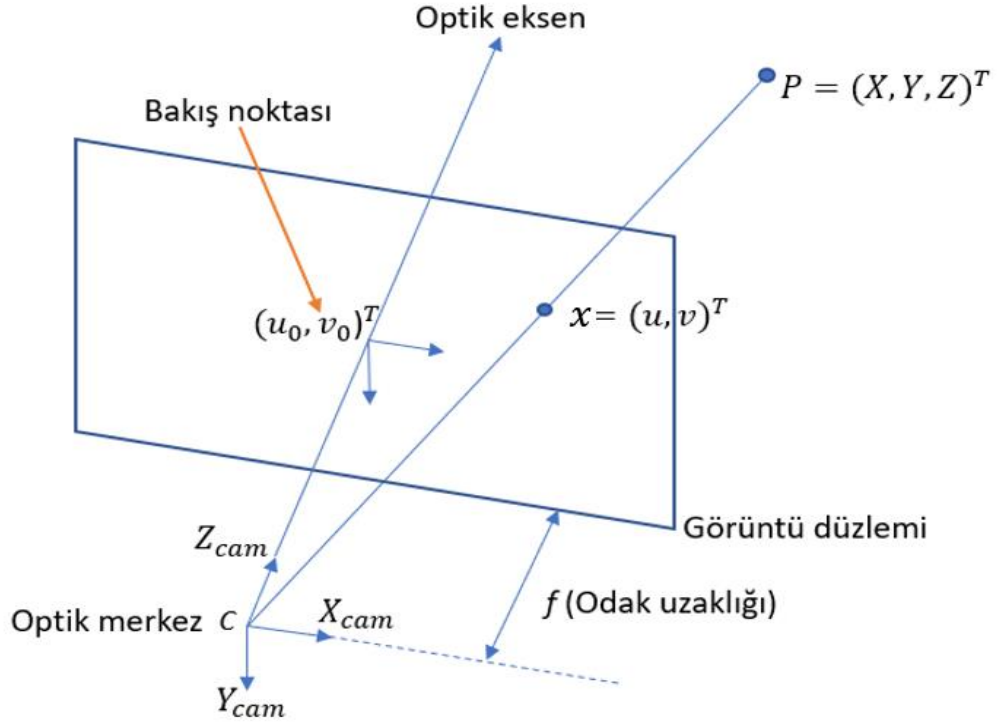
$$X = \frac{x}{w} \quad \text{ve} \quad Y = \frac{y}{w} \quad (2)$$

$$(x, y, w) \Rightarrow \left( \frac{x}{w}, \frac{y}{w} \right) \quad (3)$$

Örneğin; homojen koordinat sisteminde  $(1, 2, 3)$ ,  $(2, 4, 6)$ ,  $(4, 8, 12)$ ,  $(1a, 2a, 3a)$  noktaları, Kartezyen uzayında (veya Öklid uzayında)  $\left(\frac{1}{3}, \frac{2}{3}\right)$  olarak aynı noktayı gösterirler. Bu nedenle, bu noktalar "homojendir". Başka bir deyişle, homojen koordinatlar ölçekle değişmezdir (Siegwart *et al.* 2011).

### **Pinhole Kamera**

Görüntü düzlemindeki görüntünün düz bir şekilde elde edilmesi için, pinhole kamera modelinin yaygın bir gösterimi ise, açıklığın ve görüntü düzleminin yer değiştirilerek elde edildiği Şekil 3'te gösterilen modeldir.



**Şekil 3.** Pinhole kamera yaygın gösterimi (Morvan 2018)

Pinhole aynı zamanda optik merkez, projeksiyon merkezi veya kamera merkezi olarak da adlandırılır. Optik merkez ve görüntü düzlemi arasındaki mesafeye odak uzaklığı (*focal length*) denir ve genellikle  $f$  ile gösterilir. Görüntü düzlemine dik olan ve projeksiyon merkezinden geçen düzleme optik eksen denir. Optik eksen ve görüntü düzleminin kesiştiği noktaya ise bakış noktası (*principal point*) denir. Basitleştirilmiş modelde,  $(X, Y, Z)$ 'nin optik merkezin referans koordinat eksenleri olduğu,  $Z$ -ekseninin optik eksenle çakıştığı kabul edilir. Bu modelde, kamera referans koordinat eksenlerinin dünya referans koordinat eksenleriyle aynı olduğu varsayılır. Bu nedenle, dünya çerçevesinde herhangi bir  $P$  noktasının, kamera koordinat sistemine dönüştürülmesine gerek kalmaz (Hartley and Zisserman 2003). Denklem (4)'teki gibi  $P = (X, Y, Z)^T$  ile temsil edilen bir nokta,  $P$  noktasını kamera merkezine birleştiren doğrunun görüntü düzlemiyle kesiştiği noktaya  $p = (u, v)^T$  olarak yansır.

$$P(X, Y, Z)^T \rightarrow p(u, v)^T \quad (4)$$

Üçgen benzerliğinden denklem (5) bulunur.

$$u = \frac{Xf}{Z} \text{ ve } v = \frac{Yf}{Z} \quad (5)$$

Denklem (5), denklem (4)'te yerine konulduğunda denklem (6) elde edilir.

$$P(X, Y, Z)^T \rightarrow p \left( \frac{Xf}{Z}, \frac{Yf}{Z} \right)^T \quad (6)$$

İdeal pinhole kamera modeli, Öklid uzayında bir 3B nokta ve ona karşılık gelen Öklid uzayında 2B projeksiyonu ile görüntü düzlemi arasındaki ilişkiyi açıklar. Buna perspektif iz düşüm denir. Denklem (6)'da görüldüğü gibi, 3B koordinatlardan 2B koordinatlara eşleme doğrusal değildir. Bunun yerine homojen koordinatlar kullanmak doğrusal denklemler elde etmemizi sağlar (Siegwart *et al.* 2011).

$\tilde{p}$  ve  $\tilde{P}$  sırasıyla  $p$  ve  $P$ 'nin homojen uzaydaki koordinatlarını temsil ederse, perspektif iz düşüm denklem (7)'deki gibi yazılabilir.

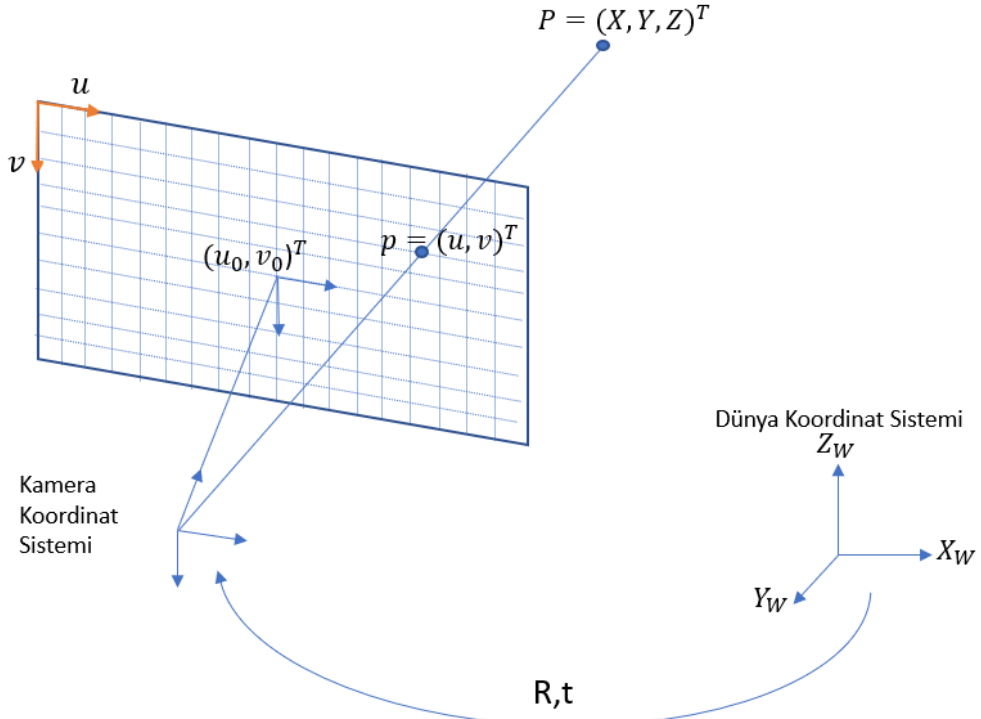
$$\tilde{P} = [X, Y, Z, 1]^T \rightarrow \tilde{p} = [fX, fY, Z]^T \quad (7)$$

Denklem (7)'deki perspektif iz düşüm,  $\lambda$  ölçekleme faktörü olmak üzere denklem (8)'deki gibi yazılabilir.

$$\lambda \tilde{p} = \begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (8)$$

### Pinhole Kamera Genelleştirilmiş Model

Basitleştirilmiş model, görüntü düzlemi koordinatlarının orijininin bakış noktaları olduğunu varsayar. Fakat gerçekte orijin, Şekil 4'te gösterildiği gibi görüntü düzleminin sol üst köşesi kabul edilir.



**Şekil 4.** Kamera ve dünya referans çerçeveleri arasında koordinat dönüşümü (Kornia Developers 2019)

Görüntü düzleminde bir noktanın koordinatları piksel olarak ifade edilir. Bu nedenle, kameranın çözünürlüğünü piksel / metre olarak bilmemiz gerekir. Eğer pikseller tam olarak kare ise, çözünürlük hem u (yatay) hem de v (dişey) yönlerinde birbiriyle aynıdır. Ancak, daha genel bir durum için, sırasıyla u ve v yönlerinde  $k_u$  piksel/metre ve  $k_v$  piksel/metre çözünürlüğe sahip dikdörtgen pikseller olduğu kabul edilir. Bu nedenle, denklem (9) ve denklem (10)'da  $p'$  yi piksel cinsinden ölçmek için, u ve v koordinatları sırasıyla  $k_u$  ve  $k_v$  ile çarpılmalıdır (Siegwart *et al.* 2011).

$$u = k_u \frac{Xf}{Z} + u_0 \quad (9)$$

$$v = k_v \frac{Yf}{Z} + v_0 \quad (10)$$

$u_0$  ve  $v_0$  bakış noktalarının koordinatları,  $k_u$  ve  $k_v$  sırasıyla u ve v yönleri boyunca piksel/metre cinsinden ifade edilen ölçeklendirme sabitleridir. Böylece projeksiyon denklem (11)'de gösterilen hali alır (Corke 2017);

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} fX \\ fY \\ Z \end{bmatrix} = \begin{bmatrix} fk_u & 0 & u_0 & 0 \\ 0 & fk_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (11)$$

$fk_u$  ve  $fk_v$  piksel cinsinden sırasıyla u ve v yönlerindeki odak uzaklıklarıdır.

3B koordinatlardan 2B koordinatlara dönüşümü tanımlayan gerçekçi bir kamera modeli, dünya ve kamera arasındaki yer değiştirme (*translation*) ve dönme (*rotation*) hareketlerini (*rigid body transformation*) de dikkate alır. Genel olarak dünya referans sistemi ( $X_w, Y_w, Z_w$ ), kamera referans sistemi ( $X_C, Y_C, Z_C$ ) ile çakışmaz.

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + t \quad (12)$$

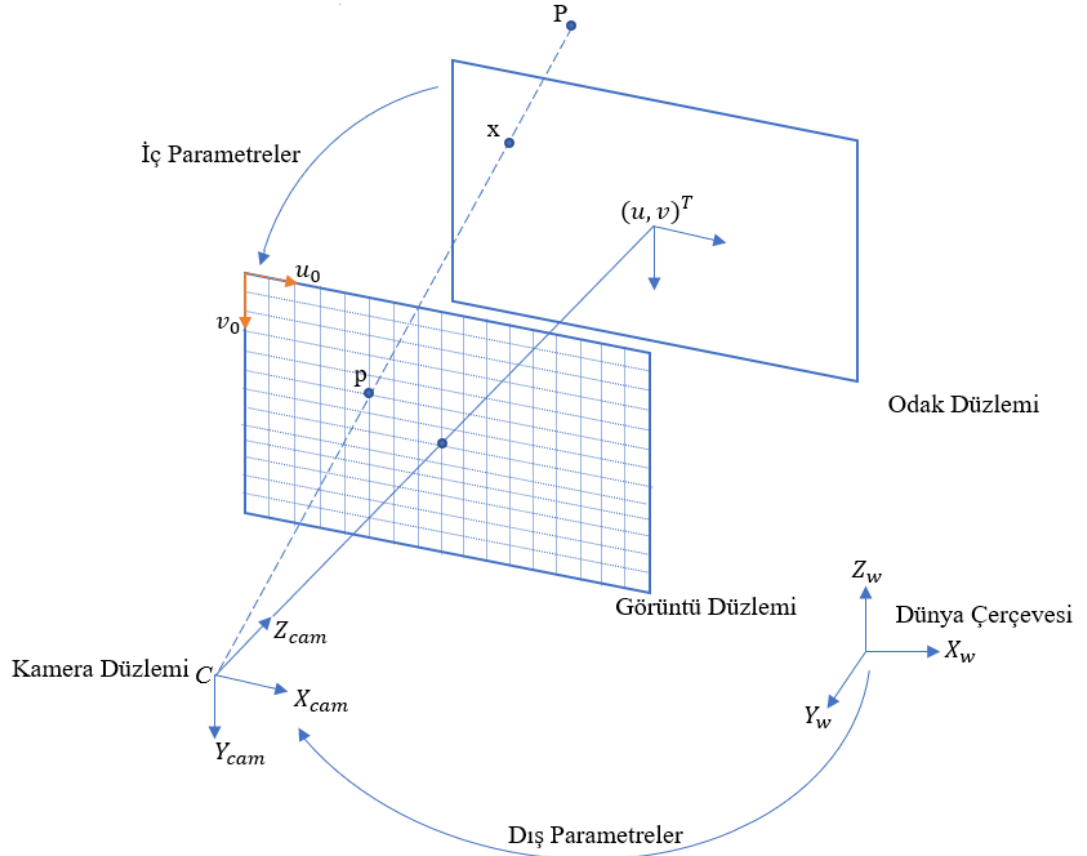
Şekil 4 'te gösterilen ve denklem (12)'de ifade edilen dönüşüm kullanılarak, denklem (11) tekrar yazılırsa projeksiyon denklemi, denklem (13)'teki gibi elde edilir. Denklem (14) ise denklem (13)'ün kapalı formatta yazılmış halidir.

$$\lambda \tilde{p} = \begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} fk_u & 0 & u_0 \\ 0 & fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (13)$$

$$\lambda \tilde{p} = K[R|t]\tilde{P}_w \quad (14)$$

Denklem (14)'te  $K = \begin{bmatrix} fk_u & 0 & u_0 \\ 0 & fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$  kamera iç parametre matrisi (*intrinsic parameter matrix*),  $fk_u$ ,  $fk_v$ ,  $u_0$ ,  $v_0$  kamera iç parametreleri (*camera intrinsic parameters*),  $R$  (rotasyon) ve  $t$  (translasyon, yer değiştirme) kamera dış parametreleri (*camera extrinsic parameters*),  $[R|t]$  kamera dış parametre matrisi (*extrinsic parameter matrix*) olarak adlandırılır.  $\tilde{P}_w$  homojen koordinatlarda dünya referans noktasıdır. Şekil 5'te gösterildiği gibi, dünya noktaları kamera dış parametreleri kullanılarak kamera koordinatlarına dönüştürülür. Kamera koordinatları, kamera iç parametreleri aracılığı ile görüntü düzlemine eşlenir. Bu iç ve dış kamera parametreleri, kamera kalibrasyonu olarak bilinen bir işlemde eşzamanlı olarak tahmin edilebilir (Siegwart *et al.* 2011).

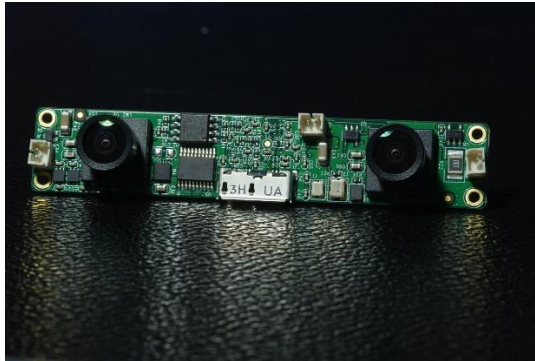




Şekil 5. Dış ve iç parametreler ile referans çerçeve dönüşümleri

### Stereo Vizyon

3B uzayda nesnelere doğru bir şekilde belirlemek, yani nesne derinliğini tahmin etmek için stereoskopik bir görüş yaklaşımı gereklidir. Stereo kamera, aralarında taban çizgisi olarak adlandırılan sabit bir mesafeye sahip iki özdeş pinhole kameradan oluşur (Siegwart *et al.* 2011). Şekil 6, bir stereo kamera örneği göstermektedir.



Şekil 6. Stereo kamera örneği (Cognard 2014)

Stereo kamera çiftinin önceden kalibre edildiği varsayılırsa, sağ ve sol kameraların iç parametre matrisleri  $K_L$ ,  $K_R$  ve dış parametre matrisleri  $[R_R|t_R]$ ,  $[R_L|t_L]$  elde edilmiş olur. Stereo vizyonda, dünya koordinat sisteminin orijininin sol kameranın optik merkezi olduğu kabul edilir. Bu nedenle I birim matrisi temsil etmek üzere,  $R_L = I$  ve  $R_R = R$  yazılabilir

(Scaramuzza 2019). Denklem (14)'te elde edilen projeksiyon denklemi sol kamera için denklem (15) ve sağ kamera için denklem (16)'deki gibi olur.

$$\lambda_I \tilde{p}_I = K_I [I|0] \tilde{P}_w \quad (15)$$

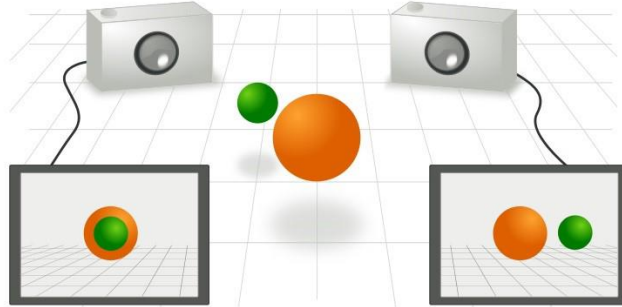
$$\lambda_R \tilde{p}_R = K_R [R|t] \tilde{P}_w \quad (16)$$

Dış parametre matrisi ile dünya koordinat sistemindeki bir nokta çarpıldığında kamera koordinat sistemine geçileceği için denklem (16)'da  $[R|t] \tilde{P}_w$ , denklem (17)'de  $\tilde{P}_R^C$  olarak yazılabilir.

$$\lambda_R \tilde{p}_R = K_R \tilde{P}_R^C \quad (17)$$

### Epipolar Geometri

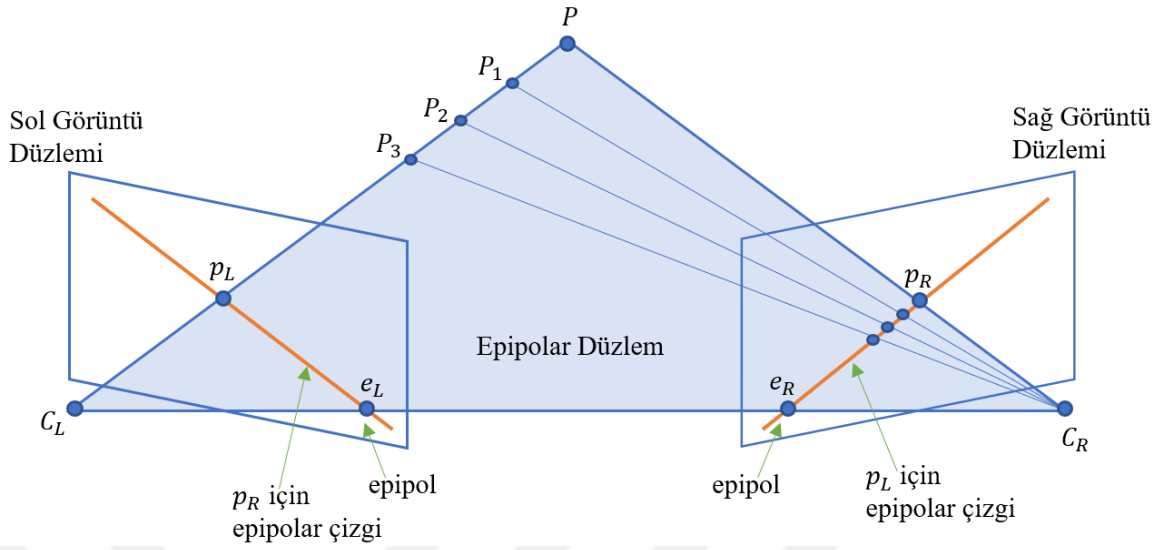
Epipolar geometri, stereo vizyonun geometrisidir. Bir sahne Şekil 7'deki gibi farklı perspektiflerden iki kamera tarafından görüntülendiğinde, görüntüler arasındaki ilişkiyi tanımlamak ve bir görüntüde bir piksel verildiğinde, diğer görüntüdeki hangi piksele karşılık geldiğini belirlemek için epipolar geometri kullanılır.



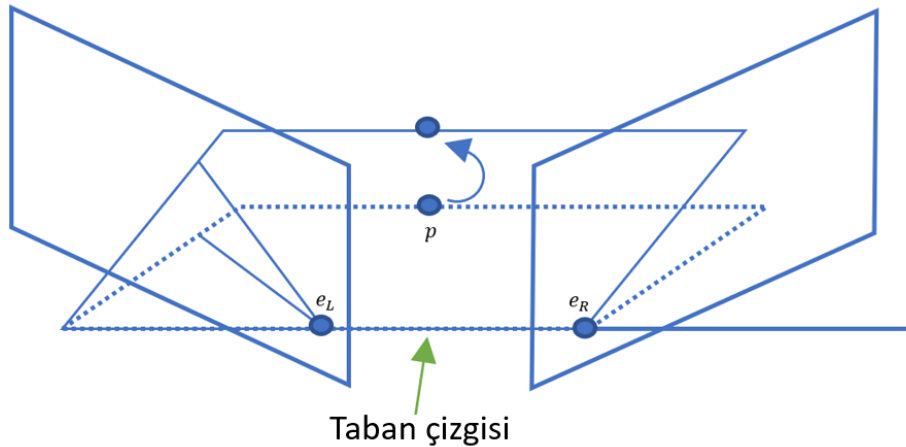
Şekil 7. Stereo vizyon (Nordmann 2008)

Şekil 8'de epipolar geometri gösterilmektedir. Burada,  $C_R$  ve  $C_L$  stereo kameranın sırasıyla sağ ve sol kameralarının optik merkezleridir. İki kamera dünya koordinatlarında bulunan aynı  $P$  noktasını görmektedir.  $P$  noktası sağ ve sol görüntü düzlemlerine sırasıyla piksel koordinatlarında  $P_R$  ve  $P_L$  olarak yansımaktadır. Her kameranın optik merkezi diğer kameranın görüntü düzlemlerinde farklı bir noktaya yansır. Bu iki görüntü noktası,  $e_R$  ve  $e_L$ , epipolar noktalar olarak adlandırılır. Epipolar noktalar, taban çizgisinin görüntü düzlemleriyle çakıştığı noktalar olarak tanımlanabilir.  $P$ ,  $C_R$  ve  $C_L$ 'nin oluşturduğu alana epipolar düzlemi denir. Epipolar düzleminin görüntü düzlemlerini kestiği doğrular  $P_R - e_R$  ve  $P_L - e_L$  ise epipolar doğrular olarak adlandırılır ve sırasıyla  $l_R$  ve  $l_L$  olarak gösterilmiştir. Aynı zamanda,  $l_R$  epipolar doğrusu,  $P - C_L$ 'nin oluşturduğu doğrunun görüntü düzlemine yansımasıdır ve sol kamera için benzer simetrik argüman geçerlidir. Bir görüntü düzleminde bulunan bütün epipolar doğrular,

kendi görüntü düzlemlerinde bulunan epipoller ile Şekil 9’da gösterildiği gibi çakışır (Hartley and Zisserman 2003).



Şekil 8. Epipolar geometri (Chotrov 2018)



Şekil 9. Epipolar doğru demetleri (Hartley and Zisserman 2003)

Epipolar geometride eğer iki kameranın birbirine göre pozisyonu ve iz düşüm noktası  $P_L$  biliniyorsa,  $l_R$  epipolar doğrusu da bilinir ve  $P$  noktası'nın sağdaki görüntü düzlemine iz düşümü  $P_R$ ,  $l_R$  epipolar doğrusunun üzerinde olmalıdır. Bu sınırlamaya epipolar sınırlama denir. Şekil 8’de  $P - C_L$ ’nin oluşturduğu doğrunun üzerinde bulunan farklı 3B noktaların ( $P_1, P_2, P_3, \dots$ ) izdüşümlerinin  $l_R$  epipolar doğrusu üzerinde olduğu görülmektedir.

Asıl matris (E matrisi-*Essential Matrix*) epipolar geometrinin cebirsel bir temsidir. Sahne üzerindeki  $P$  noktasının birinci ve ikinci görüntü düzleminde karşılık gelen noktaları homojen görüntü koordinatları  $\tilde{x}_r$  ve  $\tilde{x}_l$  ile temsil edilirse, E matrisi kullanılarak, bu iki eş nokta arasında bir denklem kurulabilir. Epipolar kısıtlama, ilk kameranın optik merkezinden ilk görüntü düzlemine, ikinci kameranın optik merkezinden ikinci görüntü düzlemine ve bir optik merkezden diğerine olan vektörlerin hepsinin eş düzlemli olduğunu söyler. E matrisinde  $\tilde{x}_r$  ve

$\tilde{x}_l$  noktaları normalize edilmiş görüntü koordinatlarıdır. Normalize edilmiş görüntü koordinatlarında odak uzaklığı 1'dir ve orijin optik merkezdedir. Bu nedenle normalize edilmiş görüntü koordinatları kamera koordinatları ile aynı anlama denk gelir. Denklem 18'de gösterildiği gibi R ve t iki kameranın koordinat düzlemleri arasındaki rotasyon ve translasyonu gösterir ve  $\tilde{x}_r$ 'ı sol görüntü düzleminin koordinat çerçevesine dönüştürür (Hata and Savarese 2018).

$$R\tilde{x}_l + t = \tilde{x}_r \quad (18)$$

Denklem (19)-(23) E matrisini elde etmek için uygulanan temel adımları gösterir

$$t \times R\tilde{x}_l + t \times t = t \times \tilde{x}_r \quad (19)$$

$$t \times R\tilde{x}_l = t \times \tilde{x}_r \quad (20)$$

$$\tilde{x}_r(t \times R\tilde{x}_l) = \tilde{x}_r(t \times \tilde{x}_r) \quad (21)$$

$$\tilde{x}_r(t \times R\tilde{x}_l) = 0 \quad (22)$$

$$\tilde{x}_r^T [t]_{\times} R\tilde{x}_l = 0 \quad (23)$$

$$\tilde{x}_r^T E \tilde{x}_l = 0 \quad (24)$$

Denklem (24) epipolar geometri sınırlamasını temsil eden denklemdir. Burada, denklem (23)'te  $[t]_{\times} R$ , denklem (24)'te E matrisini ifade etmektedir. Denklem (23)'te  $[t]_{\times}$  ters simetrik matristir ve herhangi iki vektör arasındaki vektörel çarpımı, matris-vektör çarpımı olarak temsil etmek için kullanılır. E matrisi sadece kamera dış parametrelerini içerir.

E matrisi hesaplanırken kameraların kalibre edilmiş olduğu kabul edilir. Kalibre edilmemiş kameralar için ise temel matris (F matrisi- *Fundamental matrix*) tanımlanır. F matrisi, E matrisinin daha genelleştirilmiş bir halidir ve piksel koordinatlarında tanımlıdır. Kalibre edilmemiş kameralar söz konusu olduğu için, F matrisi bulunurken kamera iç parametrelerini de içerir. Sırasıyla  $\tilde{p}_l$  ve  $\tilde{p}_r$ , P noktasının birinci ve ikinci görüntü düzleminde karşılık gelen homojen piksel koordinatlarını ve  $\tilde{P}_l^C$  ve  $\tilde{P}_r^C$  homojen kamera koordinatlarını göstermek üzere denklem (17), denklem (25)'te sol ve sağ kameralar için yeniden ifade edilmiştir. Denklem (26)'da  $\tilde{P}_l^C$  ve  $\tilde{P}_r^C$  yalnız bırakılıp, denklem (24)'te  $\tilde{x}_l$  ve  $\tilde{x}_r$ 'nin yerine yazılırsa denklem (27) elde edilir (Hata and Savarese 2018).

$$\tilde{p}_l = K_l \tilde{P}_l^C \text{ ve } \tilde{p}_r = K_r \tilde{P}_r^C \quad (25)$$

$$\tilde{P}_l^C = K_l^{-1} \tilde{p}_l \text{ ve } \tilde{P}_r^C = K_r^{-1} \tilde{p}_r \quad (26)$$

$$(K_r^{-1} \tilde{p}_r)^T E (K_l^{-1} \tilde{p}_l) = 0 \quad (27)$$

$$\tilde{p}_r^T (K_r^{-1})^T E K_1^{-1} \tilde{p}_1 = 0 \quad (28)$$

$$\tilde{p}_r^T F \tilde{p}_1 = 0 \quad (29)$$

Denklem (28)'de  $(K_r^{-1})^T E K_1^{-1}$  ifadesi F matrisini tanımlar. Denklem (29) epipolar sınırlamanın F matrisi cinsinden ifade edilmiş halidir. F matrisi kameranın hem iç parametrelerini hem de dış parametrelerini içerir.

Epipolar doğrular ve epipoller de E matrisi cinsinden ifade edilirler. Şekil 8'de gösterildiği gibi ikinci görüntü düzleminde iz düşüm noktası  $\tilde{x}_r$ 'nin karşılığı olan  $\tilde{x}_1$ , birinci görüntü düzleminde epipolar çizgi üzerinde bulunur. 2B'de iz düşümsel geometride, bir noktanın bir doğru üzerinde bulunması ancak ve ancak doğru ve noktanın iç çarpımları sıfır ise mümkün olabilir. Denklem (24)'te verilen epipolar kısıtlama bir noktanın ve doğrunun nokta çarpımı olarak denklem (30)'da yeniden ifade edilebilir.

$$\tilde{x}_r^T \tilde{l}_r = 0 \quad (30)$$

Denklem (30)'da  $\tilde{x}_r$  iz düşüm noktasını ve  $\tilde{l}_r$ ,  $\tilde{x}_r$ 'nin ait olduğu epipolar doğruyu göstermektedir. Denklem (31)'de  $\tilde{l}_r$ 'nin, E matrisi cinsinden gösterimi ifade edilmiştir.

$$\tilde{l}_r = E \tilde{x}_1 \quad (31)$$

$\tilde{x}_r$ , iz düşüm noktasının ait olduğu epipolar doğru  $\tilde{l}_1$  ise denklem (32)'de gösterilen gibidir.

$$\tilde{l}_1 = E^T \tilde{x}_r \quad (32)$$

Denklem (31) ve denklem (32)'de epipolar doğru, içinde epipol noktasını içerir. Dolayısıyla,  $e_1$  ve  $e_r$  sırasıyla birinci ve ikinci görüntü düzlemlerinin epipol noktaları olmak üzere, denklem (33) ve denklem (34) her  $\tilde{x}_r$  ve  $\tilde{x}_1$  noktası için geçerlidir.

$$e_r^T E \tilde{x}_1 = 0 \quad (33)$$

$$\tilde{x}_r^T E e_1 = 0 \quad (34)$$

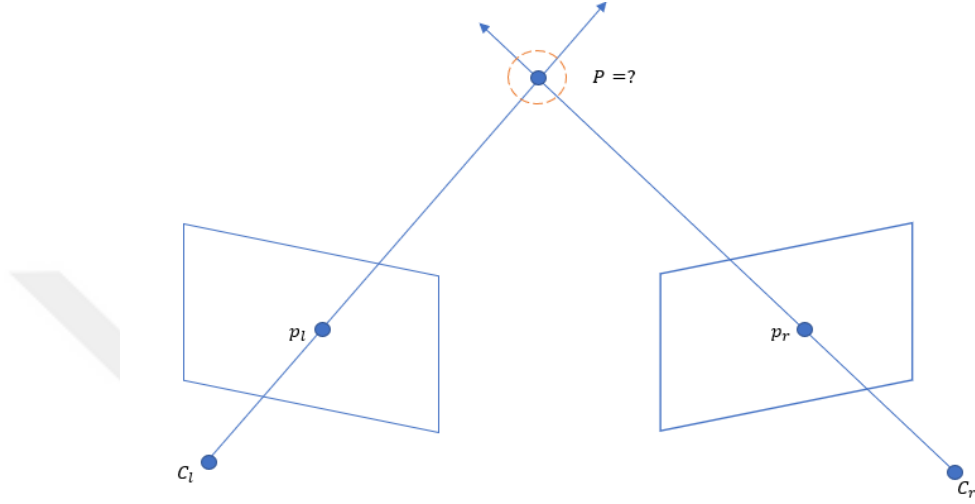
Epipol noktaları kendi düzlemlerindeki tüm epipolar doğrulara ait oldukları için, denklem (33) ve denklem (34), sırasıyla denklem (35) ve denklem (36)'ya dönüşür (Collins 2007).

$$e_r^T E = 0 \quad (35)$$

$$E e_1 = 0 \quad (36)$$

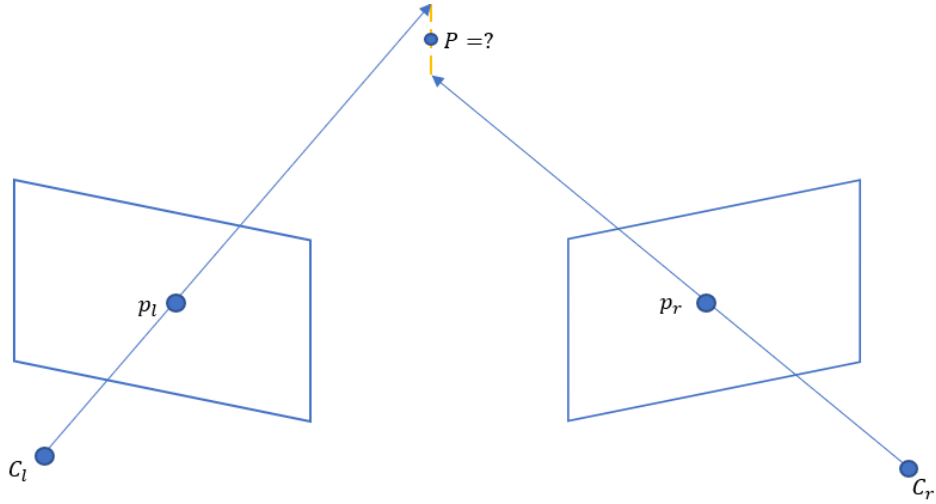
## Üçgenleme

Bir görüntüde bir nokta verildiğinde, temel / asıl matrisle çarpılması bize ikinci görünümde hangi epipolar çizginin aranacağını söyleyecektir. Dolayısıyla, birbirinin karşılığı olan iki iz düşüm noktası  $p_l$  ve  $p_r$  biliniyorsa  $p_l$  ve  $p_r$ 'den çıkan doğrultular Şekil 10'daki gibi P'de kesişmelidir. İki görüntü noktasından, 3B'li P noktasını hesaplama işlemine ise üçgenleme (*triangulation*) denir.



Şekil 10. İdeal durumda üçgenleme (Kitani 2017)

Fakat gürültüden dolayı bu iki doğru asla 3B'de kesişmez bu yüzden sadece bir yaklaşım hesaplanabilir. Şekil 11'deki gibi P, iki doğruya dik olan ortak orta nokta olarak bulunur.



Şekil 11. Üçgenlemenin gerçek geometrik yorumu (Kitani 2017)

Denklem (15) ve denklem (16)'da verilen projeksiyon denklemleri denklem (37) ve denklem (38)'de yeniden ifade edilmiştir ve burada  $M_l = K[I|0]$  ve  $M_r = K[R|t]$  kamera matrislerini göstermektedir.

$$\lambda \tilde{p}_l = M_l \tilde{P}_W \quad (37)$$

$$\lambda \tilde{p}_r = M_r \tilde{P}_W \quad (38)$$

$$\tilde{p}_l \times M_l \tilde{P}_W = 0 \quad (39)$$

$$\tilde{p}_r \times M_r \tilde{P}_W = 0 \quad (40)$$

Vektörel çarpımı, matris-vektör çarpımı olarak temsil etmek için denklem (41)'deki ters simetrik matris  $[p_j]_x$  kullanıldığında denklem (39) ve denklem (40), denklem (42) ve denklem (43)'deki gibi olur.

$$[p_j]_x = \begin{bmatrix} 0 & -p_{j,z} & p_{j,y} \\ p_{j,z} & 0 & -p_{j,x} \\ -p_{j,y} & p_{j,x} & 0 \end{bmatrix} \quad (41)$$

$$[p_l]_x M_l \tilde{P}_W = 0 \quad (42)$$

$$[p_r]_x M_r \tilde{P}_W = 0 \quad (43)$$

$$\begin{bmatrix} [p_l]_x M_l \\ [p_r]_x M_r \end{bmatrix} P_W = 0 \quad (44)$$

Denklem (45)'te  $A = \begin{bmatrix} [p_l]_x M_l \\ [p_r]_x M_r \end{bmatrix}$  olmak üzere, denklem (44), denklem (45)'te matris formunda yazılabilir (Kitani 2017).

$$AP_W = 0 \quad (45)$$

Denklem (45)'te  $P_W$ , homojen lineer sistem çözümü için kullanılan tekil değer ayrışımı (*singular value decomposition*) yöntemi ile bulunur.

## Görsel Odometri Yöntemleri

Görsel odometri yöntemleri görüntülerden çıkarılan özellik bilgisine göre, doğrudan (görünüşe dayalı) yöntem ve özellik tabanlı (geometrik) yöntemler olmak üzere ikiye ayrılır. Her iki yaklaşım da bir noktanın (piksel veya özellik) bir görüntü çerçevesinden diğerine hareketini tahmin ederek, kamera hareketini elde etmeye çalışır. Doğrudan yöntemler, tüm görüntü üzerinde çalışır ve her bir piksel yoğunluğunun, gözlemlendiği görüntüler üzerinde sabit olması gerekir. Bununla birlikte, aydınlatma, vinyet etkisi ve otomatik pozlama gibi fotometrik kamera parametrelerindeki değişiklikler, piksel değerlerinde değişiklik oluşturabilir. Bu değişiklikler, fotometrik hata olarak adlandırılır ve doğrudan yöntemler fotometrik hatayı en aza indirmeye çalışır. Doğrudan yöntemler, ayırt edici özelliklerin az olduğu ortamlarda

başarılıdırlar, ancak genellikle çok zaman alıcı bir algoritma gerektirirler. Özellik tabanlı yöntemler, görsel odometri için daha çok tercih edilen bir çözümdür (Woo *et al.* 2018).

## Özellik Tabanlı Yöntemler

Kamera ile elde edilen görüntülerde toplam piksel sayısı genellikle yüzbinlere ulaşır. Bu kadar büyük miktarda veriyle başa çıkmak için, görüntülerden özellikler çıkarılır. Böylece görüntü dizisi, daha küçük bir görüntü özellikleri kümesine indirgenebilir. Çıkarılan özellikler; köşeler, çizgiler veya kabarcıklar (*blobs*) olabilir. Özellikler tespit edildikten sonra, iki özelliğin ne kadar benzer olduğunu belirlemek için özellik tanımlayıcı (*feature descriptor*) kullanılır. Özellik tanımlayıcı; ışık, yer değiştirme (özelliklerin pozisyonlarındaki değişim), ölçek (özelliklerin boyutlarının değişimine) ve döndürme (görüntünün düzlem içi dönüşüne) işlemlerinden etkilenmemesi için özelliğin çevresindeki görüntü yamasına uygulanan bir fonksiyondur. Görüntü değişikliklerine karşı, özelliklerin değişmez olması gereklidir. Köşe özelliklerini tespit etmek ve tanımlamak için ORB (*Oriented FAST and rotated BRIEF*) (Rublee *et al.* 2011), çizgi özellikleri tespiti için LSD (*Line Segment Detection*) (Von *et al.* 2012), blob özelliklerini tespit ve tanımlamak için SIFT (Scale-Invariant Feature Transform) (Lowe 2004) ve SURF (Speeded Up Robust Features) (Bay *et al.* 2006) algoritmaları en çok tercih edilen yöntemler arasındadır. Özelliklerin yer değiştirmesi analiz edilerek kameranın hareketi hesaplanır (Woo *et al.* 2018).

Görsel odometri algoritmalarında özellik tabanlı yöntemler için temel adımlar Şekil 12’de gösterilmiştir. Her yeni stereo görüntü çifti (ya da monoküler kamerada her yeni görüntü) için ilk iki adım, 2B özelliklerin tespit edilmesi ve önceki karedekiler ile eşleştirilmesinden oluşur. Üçüncü adım ise, gidilen yörüngeyi elde etmek için kameranın hareketini hesaplamaktır. Kameranın hareketi aşamalı olarak, pozdan poza hesaplanır (Scaramuzza and Fraundorfer 2011).



**Şekil 12.** Görsel odometri temel adımlar

Görsel odometri yöntemleri için literatürde çeşitli çalışmalar yapılmıştır.

Mirabdollah and Mertsching (2015) gerçek zamanlı ve özellik tabanlı bir görsel odometri algoritması sunmuşlardır. Özellik noktası takibi için özellik çıkarımında, aykırı değerleri etkin bir şekilde eleme yöntemi olarak RANSAC (*Random Sample Consensus*)



(Fischler and Bolles 1981) algoritmasını uygulamışlar ve kareler arası hareketi tahmin etmek için ise yinelemeli beş noktalı yöntemi (Nister 2003) kullanmışlardır. Hareket ölçeğini tahmin etme amacıyla zemin düzlemindeki yüksek kaliteli özellikleri takip etmek için bir teknik önermişlerdir. Bu tekniğe göre, olasılıksal bir üçgenleme yöntemi kullanarak yer işaretlerinin konumunu almışlar ve yer düzlemindeki seyrek, yüksek kaliteli özelliklerden hareketin ölçeğini tahmin etmişlerdir. Çalışmada hem uzak hem de yakın yer işaretleri gürbüz (*robust*) bir şekilde ele alınmıştır. Kullandıkları yaklaşımın, algoritmayı büyük ölçüde hızlandırdığı ve algoritmanın aykırı değerlere karşı gücünü artırdığı sonucuna varmışlardır.

Escalera *et al.* (2016) sadece zemin yüzeyine ait özellikleri kullanan bir özellik tabanlı stereo görsel odometri tekniği önermişlerdir. Böylece poz tahmini için kullanılan toplam özellik sayısını azaltmışlar ve görüntüler arasındaki bağıl hareketi elde etmek için sadece statik noktaların dikkate alınmasını garanti etmişlerdir. Özellik çıkarımı için SIFT (*Scale Invariant Feature Transform*) algoritmasını kullanmışlardır. Kalman Filtresi ile tahminleri iyileştirmiş ve RANSAC algoritması ile aykırı değerleri elemişlerdir. Kamera yer değiştirmesini, Perspektif-n-Noktası (PnP) algoritması kullanarak elde etmişlerdir. Çalışmada, yöntemin konumu ve yönü düşük bir hatayla elde edebildiğini belirtmişlerdir.

Fanani *et al.* (2016) çalışmalarında doğrudan bir yaklaşım izlemişlerdir ve görüntülerden özellik noktalarının üç boyuttaki yörüngelerini elde etmek için yayılım tabanlı takip yöntemini kullanmışlardır. Doğru özellik noktası eşlemesini RANSAC yerine, özellik noktalarının üç boyuttaki pozisyonlarını bir sonraki kareye yayarak elde etmişlerdir. Sistemi, bir yükselme (pitch)-sapma (yaw)- yatış (roll) tahmincisi, bir poz tahmincisi ve bir ortak epipolar izleyici gibi bir dizi bileşenden oluşturmuşlardır ve yöntemlerini KITTI veri setinin ilk 11 görüntü dizisi üzerinde denemişlerdir. Sonuç olarak yaklaşımlarını gerçek zamanlı olarak uygulamadıklarını fakat özellik noktalarının konum ve yön tahminlerinde veri setinin çoğunda iyi bir doğruluk elde ettiklerini belirtmişlerdir. Yörüngelerin doğruluğunun daha da artması ve aynı zamanda aykırı değerlerin miktarının azaltılması için bundle adjustment algoritmasının yöntemine eklenmesini önermişlerdir.

Fanani *et al.* (2017) yapmış oldukları çalışmada, bir önceki çalışmalarında (Fanani *et al.* (2016)) sundukları yöntemi geliştirmişlerdir. Özellik noktası takibi için hem köşelerden hem de kenarlardan çıkarılan özelliklerin takibini yapmışlardır. Çalışmada vurguladıkları nokta, özellik noktası tespit etmede aykırı değerleri elemek için RANSAC algoritmasını kullanmamış olmalarıdır. Epipolar geometri kısıtlamasını uygulayarak aykırı değerleri elemişlerdir. Monoküler kamera ile elde edilemeyen ölçeği belirlemek için ise, evrişimli sinir ağı (*Convolutional Neural Network-CNN*) kullanarak yol sınıflandırması ile yer düzlemini tespit

etikten sonra, yer düzlemini takip etmişlerdir. Çalışmanın sonucunda elde edilen doğruluğun, KITTI veri setiyle yapılmış görsel odometri sistemlerinin oluşturulmasındaki ana yöntemlerin doğruluğundan daha fazla olduğunu belirtmişlerdir.

Pereira *et al.* (2017) diğer yaklaşımların aksine ileriye dönük bir kameradan alınan görüntüleri araç geriye doğru gidiyormuş gibi tersine kullanmışlardır. Bunun için özellik tabanlı seyrek görsel odometri (Sparse Visual Odometry) yaklaşımını izlemişlerdir. Seyrek yöntemler yalnızca seçilen bağımsız noktalar kümesini (genel olarak köşeler) izlerken, yoğun yöntemler 2B görüntü alanındaki tüm pikselleri izlemeye çalışır. Bu izlenen görüntü özellikleri, 3B konumların tahmin edildiği tek noktalardır. Öne bakan kamera için, görüntü kareleri ileriye doğru gittikçe yeni özelliklerin uzak mesafeden ilk olarak gözlemlendiğini fakat ters sıra dikkate alındığında, yeni özelliklerin en yakın karede algılandığını belirtmişlerdir. Bu nedenden dolayı çalışmanın sonucunda, görüntü karelerinde ters sıra dikkate alındığında, derinlik tahmininin dolayısıyla özellik noktalarının konumlarının çok daha doğru tespit edildiği sonucuna varmışlardır.

Fheng *et al.* (2017) hem doğrudan hem de özellik tabanlı yöntemeye dayanan bir görsel odometri yaklaşımı sunmuşlardır. Algoritmaları iki modülden oluşmuştur: özellik tabanlı modül; görüntü karesinde yeterli özellik varsa pozunu tahmin etmek için ve doğrudan modül; düşük dokulu özellik barındırmayan ortamlar için kullanılmıştır. İlk önce, özellik çıkarma yöntemiyle görüntülerdeki özellik noktalarını belirlemişlerdir. Belirlenen bir eşik değerine göre, çıkarılan özellik noktalarının sayısı eşik değerinden büyükse, özellik tabanlı yöntemi kullanmaya devam etmişlerdir. Özellik noktalarının sayısı eşik değerinden küçükse doğrudan yöntemi kullanmışlardır. Bu durumda, kameranın hareketini tahmin etmek yalnızca sahnedeki piksel parlaklık bilgisine dayanmıştır. Çalışmada sonuç olarak, sadece yöntemlerin her birinin kullanıldığı yaklaşımlara göre daha güvenilir bir sonuç elde edildiğini belirtmişlerdir.

Engel *et al.* (2018) monoküler görsel odometri için doğrudan ve seyrek (*direct-sparse*) bir yaklaşım izlemişlerdir. Yöntemin, özellik çıkarımı için yeterli özellik olmayan yerler dahil olmak üzere görüntü bölgelerinden pikselleri örnekleyebildiğini belirtmişlerdir. Özellik noktası seçim stratejisi, görüntüde iyi dağıtılmış ve yakın çevresine göre yeterince yüksek yoğunluk gradyan büyüklüğüne sahip noktaları seçmeyi amaçlamıştır. Özellik nokta adayları, epipolar çizgi boyunca ayrı bir arama kullanılarak takip eden karelerde izlenmiştir. Böylece fotometrik hatayı en aza indirmeyi başarmışlardır. Fotometrik hatanın belirli bir eşiği aştığı özellik noktalarını aykırı değerler olarak kabul etmişler ve sonraki karelerde takip edilmesinin önlenmesi için aykırı değerleri elemişlerdir. Ayrıca, hassas bir tahmin elde etmek için yöntem hem geometrik hem de fotometrik kamera kalibrasyon sonuçlarını kullanmıştır. Yöntem,

gerçek zamanlı çekilen bir video üzerinde denenmiştir. Çalışmalarının sonucunda, doğrudan-seyrek görsel odometrinin, özellik tabanlı yöntemlere göre doğruluğunun daha yüksek olduğunu belirtmişlerdir.

Zheng *et al.* (2018) hem nokta hem de çizgi özelliklerine dayanan Genişletilmiş Kalman Filtresi (EKF) tabanlı bir stereo görsel odometri sistemi önermişlerdir. Çizgi özelliklerini, nokta özelliklerinin algılanmasının zor olduğu düşük dokulu ortamlarda kullanmışlardır. EKF, IMU ölçümlerinden gelen durum ve görüntülerden çıkarılan özellik noktaları arasındaki korelasyonu belirleyerek aracın konumunu ve yönünü belirlemiştir. Sapmayı azaltmak için bundle adjustment yöntemini kullanmak yerine filtre tabanlı bir çerçeve önermişlerdir. Bir nokta ve/veya çizgi özelliği artık izlenmediğinde EKF güncellemesi tetiklenmiştir. Görüntülerden özellik çıkarımı için FAST (*Features from Accelerated Segment Test*) (Rosten and Drummond 2006) algoritması ve aykırı değerleri elemek için RANSAC algoritmasını kullanmışlardır. Çizgi özellikleri çıkarımı için ise çizgi segment detektörü (Von Gioi *et al.* 2008) algoritmasını uygulamışlardır. Yöntemi, mikro hava aracı ile oluşturulmuş EuRoC veri seti (Burri *et al.* 2016) ve kendi oluşturdukları Trifo Ironsides veri seti ile denemişlerdir. Çalışmanın sonunda, EuRoC veri seti kullanılan diğer görsel odometri yöntemlerine göre doğruluğu daha yüksek sonuçlar ve Trifo Ironsides veri setinde yer referansına (*ground truth*) yakın sonuçlar elde ettiklerini belirtmişlerdir.

Li *et al.* (2018) monoküler görsel odometride en büyük zorluk olan ölçek iyileştirmesi için bir yöntem sunmuşlardır. UnDeepVO ismini verdikleri yöntemde, monoküler bir kameranın 6-DoF pozunu ve görüş derinliğini derin sinir ağları kullanarak tahmin edebilmişlerdir. Ölçeği iyileştirmek için eğitim aşamasında etiketsiz KITTI veri setinin stereo görüntü çiftlerini kullanmışlardır. Eğitim sırasında stereo görüntü çiftlerinin kullanılması sayesinde sadece tahmini poz değil, aynı zamanda derinlik tahmini de yapmışlardır. Eğitimden sonra, monoküler görüntülerle konum ve yön bulunarak görsel odometri gerçekleştirilmiştir. KITTI veri seti üzerinde yapılan deneyler sonucunda, UnDeepVO yönteminin poz doğruluğu açısından iyi bir performans elde ettiğini belirtmişlerdir.

Cai *et al.* (2019) dış mekân mobil robotu için gerçek zamanlı bir konumlandırma ve navigasyon sistemi sunmuşlardır. Çalışma, görsel odometri, IMU ve GPS'ten gelen verilerin genişletilmiş Kalman Filtresi ile füzyonuna dayanmıştır. IMU manyetometre ve açısız hızdan yön bilgilerini, GPS ise mobil robotun mutlak konum bilgilerini sağlamıştır. Özellik tabanlı görsel odometriyi, hareket mesafesini ve ek lokalizasyon bilgilerini elde etmek için kullanmışlardır. Görsel odometri için monoküler RGB-D kamera tercih edilmiştir. Görüntülerden ORB (Rubblee *et al.* 2011) özellikleri çıkarıldıktan sonra kamera poz tahminini

yaparak, GPS ve IMU'dan gelen verileri genişletilmiş Kalman Filtresi kullanarak füzyon etmişlerdir. Sonuçları, GPS ham verileri ile karşılaştırmışlar ve sunulan yöntemin performansını değerlendirmişlerdir. Sonuç olarak, görsel odometrinin başlangıçta güvenilir bir bilgi verdiği fakat mesafe arttıkça, hata birikiminin de arttığını belirtilmişlerdir. Bu sorunun çözümü olarak, mesafe hatasını sınırlaması için GPS ve yön bilgisini iyileştirmesi için IMU, görsel odometriye Kalman Filtresi kullanılarak füzyon edilmiştir. Yöntemi iki farklı dış mekân ortamında test etmişler ve füzyonun pozisyon doğruluğunu artırdığını belirtmişlerdir.

Yoon and Kim (2019) stereo görsel odometriye uygulanabilen, fotogrametrik özellik optimizasyon yöntemi geliştirmişlerdir. Görüntülerden özellikler çıkarılıp eşleştirme yapıldıktan sonra, karşılık gelen noktaları optimize etmek için, önceki karelerde bulunan özellik noktalarıyla, mevcut kareden çıkarılan ve önceki kareye yansıyan özellik noktaları arasındaki fark belirli bir eşik değerine göre büyük kabul edilip, bu özellik eşleştirmeleri elenmiştir. Mevcut ve sonraki kareler arasında ise, aykırı eşlemeleri elemek için RANSAC algoritmasını kullanmışlardır. Fotogrametrik analiz yoluyla gerçek zamanlı aykırı değer eleme yöntemini geliştirilen yöntemin avantajı olarak vurgulamışlardır. Çalışmada, özellik çıkarımından aykırı değerlerin doğru bir şekilde elenmesinin poz tahminlerini iyileştirdiğini ifade etmişlerdir.

Dias and Laureano (2019), dikkatli bir özellik seçimine dayanan stereo görsel odometri için bir yaklaşım sunmuşlardır. Görüntülerden özellik çıkarımı yapmak için FAST algoritmasıyla köşe noktalarının tespitini yapmışlardır. Ardından, iki stereo kare arasında özellik eşleştirmesi için, normalize edilmiş mutlak farklılıkların toplamı yöntemini kullanmışlardır. Farklı zaman anlarındaki birbirini takip eden kareler arasında özellik takibi için ise KLT (Tomasi and Kanade) algoritmasını uygulamışlardır. Her iki işlemde de epipolar geometri kısıtlamasını uygulayarak aykırı değerleri elemişler ve kalan noktalar ile poz tahminini yapmışlardır. Sonuç olarak, yöntemin KITTI veri seti üzerinde denendiğini ve düşük bir sapma hatası elde edildiğini belirtmişlerdir.

Chiodini *et al.* (2020) RGB kamera ve menzil sensörü ölçümleri arasındaki füzyona dayalı bir görsel odometri yöntemi sunmuşlardır. RGB kameradan alınan görüntülerden SURF algoritmasıyla özellik çıkarımı ve eşleştirmesi yapıldıktan sonra P3P metodu ile poz kestirimini ve RANSAC algoritması ile aykırı değerlerin elemesini yapmışlardır. Yöntemde ölçek bilgisini elde edebilmek için uçuş süresi kamerasını kullanmışlardır (bu kamera türü, ışığın geri dönüşü üzerinden objelere olan mesafeyi ölçebilmektedir). İlk olarak RGB kamerayı, uçuş süresi kamerasıyla kalibre etmişlerdir. Kalibrasyon işlemi, düşük çözünürlüklü uçuş süresi kamerasından gelen aralıklar ile RGB görüntüsünde Harris köşeleri ile ilişkili olan köşelerin algılanmasına ve bunların görüntü düzleminde yeniden yansıtılmasına (*reprojection*)

dayanmıştır. Yöntem ilk olarak TUM-RGB-D veri seti (Sturm *et al.* 2012) üzerinde test edilmiştir ve verilen yer referansı bilgisiyle doğruluğu karşılaştırılmıştır. Daha sonra, gerçek zamanlı olarak dış mekanda test edilmiştir ve GPS ile elde edilen yer referansı verisiyle doğruluğu test edilmiştir. Sonuç olarak, ölçeği elde etmek için menzül sensörü kullanımının kararlı bir ölçek bilgisi verdiği ve görsel odometri için başarılı bir yaklaşım elde edildiğini belirtmişlerdir.

Zhang *et al.* (2020) trafik işaretlerinin tespitine dayalı monoküler görsel odometri için bir bundle adjustment planı önermişlerdir. Algoritmayı, Mur-Artal and Tardos (2017) tarafından sunulan ORB-SLAM2 yönteminin üzerine kurmuşlardır. ORB-SLAM2 görüntülerden ORB özelliklerinin çıkarılmasına dayanan kameranın poz tahmininin yanı sıra, çevreyi seyrek olarak yeniden inşa eden bir yöntemdir. Trafik işaretlerinden çıkarılan özellikleri, tahmini pozları optimize etmek ve bundle adjustment sürecinde biriken sapma hatalarını kısıtlamak amacıyla ORB özellikleri ile birlikte kullanmışlardır. Bu özellikler, iyi eğitilmiş bir derin öğrenme trafik işareti tespiti ve gürbüz bir hat segmenti detektörüne (Von Gioi *et al.* 2012 ) dayanarak trafik işaretlerinden çıkarılan noktalardır. Görüntülerdeki trafik işaretlerini tespit etmek için CNN tabanlı derin öğrenme yaklaşımı olan Faster R-CNN (Ren *et al.* 2015) yöntemini kullanmışlardır. Çalışmanın sonunda, uygulanan yöntemin görsel odometri performansını artırdığını ifade etmişlerdir.

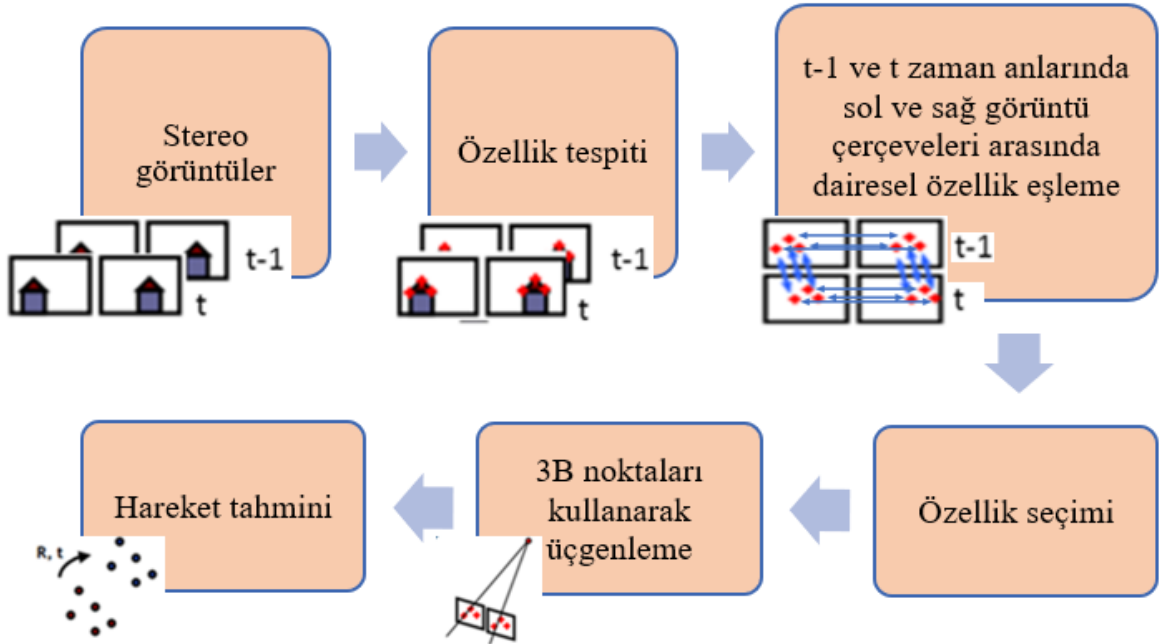
Wen *et al.* (2020), Engel *et al.* (2018) tarafından yapılan DSO (Direct Sparse Odometry) çalışmasını, stereo kamera ve IMU füzyonu ile birleştirmişlerdir. Çalışmada, IMU ve stereo görüş füzyonunun, yöntemlerin her birisinin bireysel olarak yaptıkları hataları telafi edebildiği belirtilmiştir. Stereo kamera, ölçek kaymasını ve IMU'nun uzun vadeli sapma hatasını telafi edebilmiştir. IMU ise, doğrudan ve seyrek (direct-sparse) yöntemin kullandığı fotometrik takip özelliğini iyileştirebilmiş ve görsel bilgi olmadan büyük kareler arası hareketi izlemesine izin vermiştir. Çalışmada EuRoC veri kümesi ile iç mekân lokalizasyonuna odaklanmışlardır. Sunulan yöntemin doğruluğunu, DSO tabanlı yöntemlerle ve veri setinin yer referansı ile karşılaştırmışlardır. EuRoC veri setinde, önceden ziyaret edilen lokasyonların tekrar ziyaret edildiği (loop-closure) görüntü dizilerinde başarının diğer DSO tabanlı yöntemlere göre daha düşük çıktığını belirtmişlerdir. Ancak, bu durumun olmadığı görüntü dizilerinde başarının diğer DSO tabanlı yöntemlere göre daha yüksek olduğunu ifade etmişlerdir.

## MATERYAL ve YÖNEM

Bu tez çalışmasında kullanılan görsel odometri algoritması, Cvišic and Petrovic (2015) tarafından yayımlanan çalışmanın, Mittal (2018) tarafından yapılan ve açık kaynak kod barındırma platformu GitHub'ta bulunan uyarlamasının bir versiyonudur.

### Özellik Seçimi ve Takibi

Özellik seçimi ve takibi, algoritmanın özellik işleminde takip ettiği yolu açıklamaktadır. Bu aşamada, kamera hareketi (*egomotion*) tahmininde kullanılacak olan görüntü koordinatları elde edilir. Algoritmanın akış şeması Şekil 13'te gösterilmiştir. Algoritmanın girdisi, şimdiki ( $t$ ) ve bir önceki ( $t-1$ ) ana ait olan ardışık stereo görüntü çiftleridir.



Şekil 13. Algoritmanın genel akış şeması

### Özellik İşleme

Görsel odometri algoritmasının ilk aşaması, girişe verilen ardışık stereo görüntü karelerinden özelliklerin çıkarılmasıdır. Mittal (2018) görüntülerden çizgi ve blob (kabarcık) özelliklerini çıkarmıştır. Giriş kareleri Şekil 14'te verilen  $5 \times 5'$  lik blob ve köşe maskeleri ile filtrelenir. Görüntü işlemede maske, konvolüsyon matrisi veya kernel olarak adlandırılan bir

matristir. Filtreleme işlemi ise, giriş görüntüleri üzerinde ayrı ayrı köşe ve blob kernellerinin konvolüsyon edilmesiyle gerçekleştirilir (Geiger *et al.* 2011).

-1	-1	-1	-1	-1
-1	+1	+1	+1	-1
-1	+1	8	+1	-1
-1	+1	+1	+1	-1
-1	-1	-1	-1	-1

Blob dedektör kerneli

-1	-1	0	+1	+1
-1	-1	0	+1	+1
0	0	0	0	0
+1	+1	0	-1	-1
+1	+1	0	-1	-1

Köşe dedektör kerneli

#### Şekil 14. Blob ve köşe detektör kernelleri

Bu tezde Mittal (2018)'den farklı olarak, görüntü karelerinde bulunan özelliklerin çıkarılması için, Harris Köşe Dedektörü kullanılmıştır. Köşe, bir veya daha fazla kenarın kesişimi olarak tanımlanır. Köşeler tüm yönlerde, yoğunlukta büyük değişiklik gösteren bölgelerdir. Harris Köşe Dedektörü temel olarak her yönde (u,v) yoğunluk farkını hesaplar. Özellik, Köşe Tepki Fonksiyonu (Corner Response Function) ile tespit edilir. Köşe Tepki Fonksiyonu, Harris skoru denilen  $R_{Harris}$  değerine göre köşe olarak kabul edilir veya elenir. Harris skoru  $R_{Harris}$ 'nin hesaplanması denklem (46)'da verilmiştir.

$$R_{Harris}(u, v) = \det(M(u, v)) - k \text{trace}^2(M(u, v)) \quad (46)$$

Denklem (46)'daki M denklem (47)'de tanımlanmıştır.

$$M(u, v) = \begin{bmatrix} \sum I_x^2(u, v) & \sum I_x(u, v)I_y(u, v) \\ \sum I_x(u, v)I_y(u, v) & \sum I_y^2(u, v) \end{bmatrix} \quad (47)$$

Denklem (47)'de  $I_x$  ve  $I_y$ , x ve y yönlerinde görüntü türevleridir. Görüntü türevleri denklem (48) 'de verilen Sobel filtreleri ile görüntünün konvolüsyon işlemine tabi tutulmasıyla yapılır.

$$I_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I \text{ ve } I_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I \quad (48)$$

Bu tezde,  $R_{Harris}$  skoru bulunduktan sonra en yüksek skora sahip olan 200 tane nokta, Harris köşesi olarak seçilmiştir.

Köşe özellikleri yüksek lokalizasyon doğruluğuna sahiptirler ve görsel odometri için kullanışlıdır fakat ayırt edicilikleri çok fazla (*distinctive feature*) değildir. Bu nedenle köşelerden daha ayırt edici olan kabarcık (blob) özellikleri de görüntülerde tespit edilmiştir. Kabarcık özellikleri, kendisini çevreleyen komşularına kıyasla parlaklık veya renk gibi özelliklerde farklılık gösterirler. Görüntülerden kabarcık özelliklerinin çıkarılması için bu tezde Gauss'ların Farkı (*Difference of Gaussians-DoG*) algoritması kullanılmıştır. DoG orijinal görüntünün bulanık bir versiyonunun, daha az bulanık bir versiyonundan çıkarılması ile gerçekleştirilir. Denklem (49)'da, gauss fonksiyonunun nasıl bulunacağı verilmiştir ve  $5 \times 5$ 'lik gauss filtresi için, denklem (50) ve denklem (51)'de boyut = 5 olarak alınmıştır. Denklem (49)'daki sigmanın Gauss filtresindeki rolü, dağılımın ortalama değeri etrafındaki değişimi kontrol etmektir. Sigma büyüdükçe ortalama etrafında daha fazla varyansa izin verilir ve sigma küçüldükçe ortalama etrafında izin verilen varyans azalır. Bu nedenle, sigma değeri azaldıkça görüntüdeki detay artar. Denklem (50)-(53)'te DoG hesaplamak için kullanılan fonksiyonlar gösterilmiştir.

$$\text{gauss} = \text{fspecial}('gaussian', \text{boyut}, \text{sigma}) \quad (49)$$

$$\text{gauss1} = \text{fspecial}('gaussian', 5, 15) \quad (50)$$

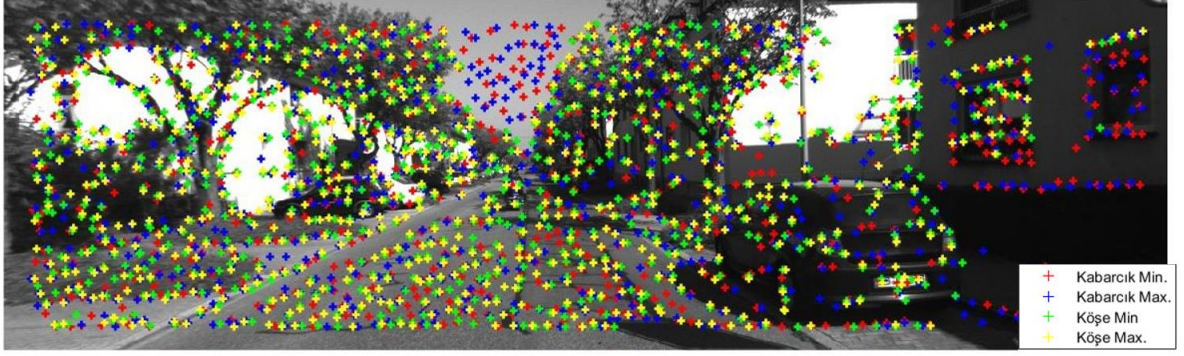
$$\text{gauss2} = \text{fspecial}('gaussian', 5, 20) \quad (51)$$

$$\text{DoG} = \text{gauss1} - \text{gauss2} \quad (52)$$

$$\text{DoGFiltreliGörüntü} = \text{conv2}(\text{double}(\text{görüntü}), \text{DoG}, 'same') \quad (53)$$

Harris köşede elde edilen, yüksek skorlu pikseller, yüksek skorlu komşu piksellere sahip olacağı için, komşu piksellerin köşe noktası olarak seçilmesi önlenmelidir. Kabarcık özelliklerinde ise görüntüden çıkarılan maksimum özellik sayısının sınırlanması ve özelliklerin görüntüde dengeli bir şekilde dağılması gerekir. Bunun için, Harris köşe detektöründe her bir yinelemede seçilen en yüksek skorlu piksel etrafında ve kabarcık detektöründe Gauss'ların Farkı filtresinin çıkışına Maksimum Olmayanı Baskılama (*non-maximum suppression*) yöntemi uygulanır (Neubeck and Gool 2006). Şekil 15'teki gibi, blob maksimum, blob minimum, köşe maksimum ve köşe minimum olmak üzere dört sınıftan birine ait olan anahtar noktaları elde edilir. Özellik eşleme kısmında, yalnızca aynı sınıfa ait özellikler arasında eşleştirme bulunacağı için, görsel odometri algoritmasının hesaplama süresinde de azalır. Tam bu noktada özellikler lokasyon, değer ve sınıf bilgilerini içermektedir.





**Şekil 15.** Köşe ve blob özellikleri

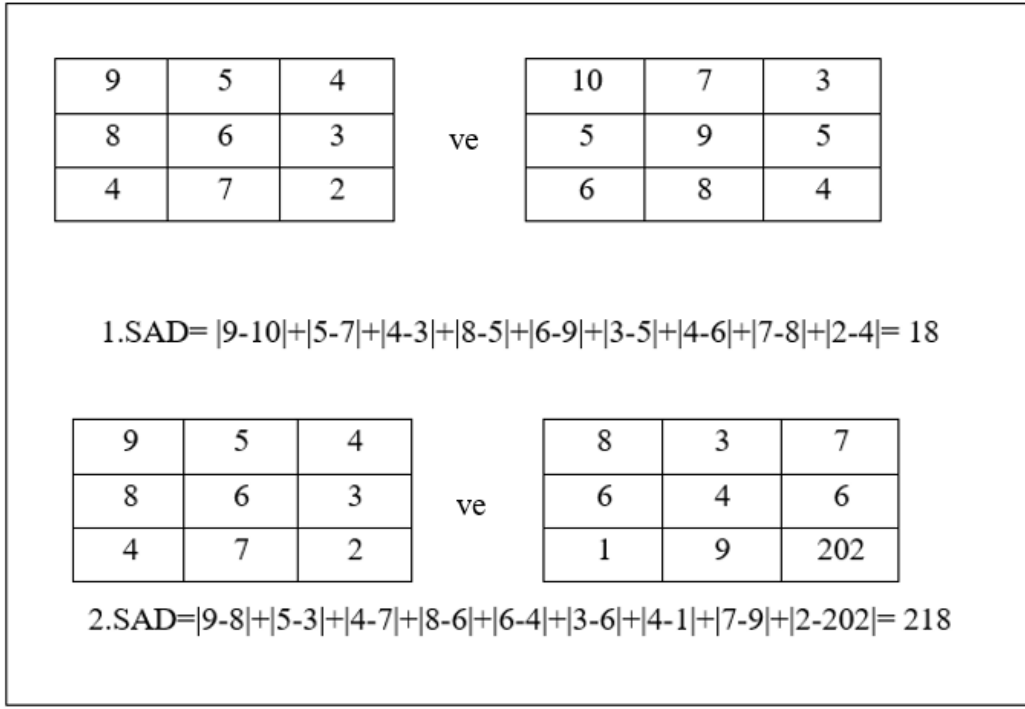
Kabarcık ve Harris köşe detektörleri özelliklerin tespitini yapabilmektedir, fakat özelliklerin tanımını yapamamaktadır. İki görüntü arasındaki özellikleri takip edebilmek, eşleştirebilmek için özellik tanımlayıcılarına ihtiyaç duyulmaktadır. Özellik tanımlayıcıları, görüntülerin gradyanlarındaki her bir özellik noktasının etrafındaki  $11 \times 11$ 'lik bloktan 16 konum kümesi kullanılarak oluşturulmuştur. Görüntü gradyanları, denklem (54) ve denklem (55)'te gösterilen  $5 \times 5$ 'lik Sobel kernellerinin görüntüler ile konvolüsyonu sonucunda oluşur. Sırasıyla ilk kernel x yönünde ve ikinci kernel y yönündeki gradyanları hesaplamaktadır.

$$\text{sobel}_{\text{kernel } x} = \begin{bmatrix} 1 & 2 & 0 & -2 & -1 \\ 4 & 8 & 0 & -8 & -4 \\ 6 & 12 & 0 & -12 & -6 \\ 4 & 8 & 0 & -8 & -4 \\ 1 & 2 & 0 & -2 & -1 \end{bmatrix} \quad (54)$$

$$\text{sobel}_{\text{kernel } y} = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 2 & 8 & 12 & 8 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ -2 & -8 & -12 & -8 & -2 \\ -1 & -4 & -6 & -4 & -1 \end{bmatrix} \quad (55)$$

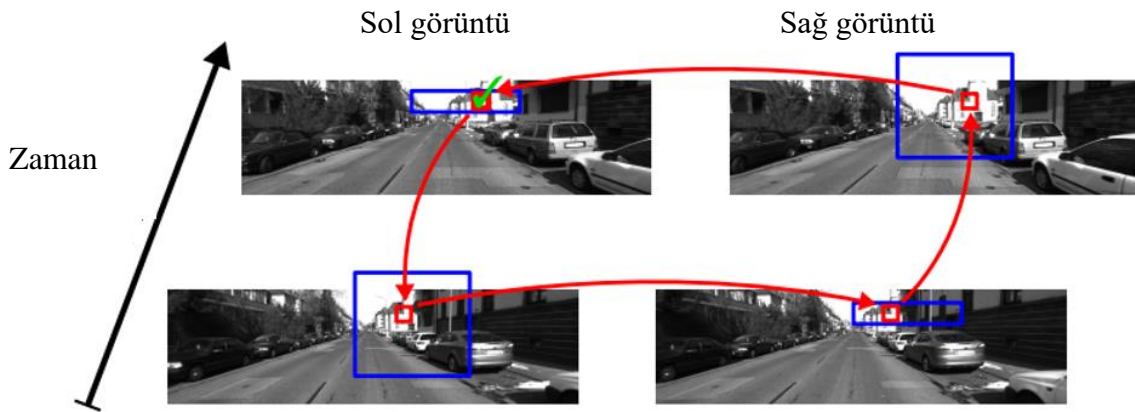
## Özellik Eşleştirme

Görüntülerden çıkarılan özellik noktaları arasında eşleşmeleri bulmak için kullanılan genel yöntemlerden biri Mutlak Farkların Toplamı (*Sum of Absolute Differences-SAD*) ve Farkların Karelerinin Toplamı (*Sum of Squared Differences-SSD*) yöntemleridir. Birinci görüntüde bir sınıfa ait olan piksel ve ikinci görüntüde aynı sınıfa ait olan piksel arasındaki mutlak fark veya farkın karesi alınarak bir değer hesaplanır. Hesaplanan değer ne kadar küçükse iki özellik birbirine o kadar benzerdir demektir. Şekil 16'da SAD yöntemine ait bir örnek bulunmaktadır.



**Şekil 16.** SAD örneği

1.SAD < 2.SAD olduğu için ilk karşılaştırılan kareler birbirine daha çok benzemektedir. SAD yöntemi basit ve hızlıdır, ancak aykırı değerleri içerir. Bu aşamada, aykırı değerlerin bazıları dairesel eşleme ile elenir. Dairesel eşleme, her özelliğin birbirini izleyen iki karenin sol ve sağ görüntü kareleri arasında eşleşmesi gerektiği anlamına gelir. Şekil 17’de gösterildiği gibi mevcut t anında sol görüntüdeki özellikler, sırasıyla t – 1 anında sol görüntü karesi, t – 1 anında sağ görüntü karesi, t anında sağ görüntü karesi ve tekrar t anında sol görüntü karesinde bitecek şekilde bir daire dizisi oluşturularak takip edilir ve eşleştirilir.



**Şekil 17.** Dairesel Eşleme

Özellik, belirtilen dizinin tüm karelerinde doğru şekilde eşleşirse, daire kapatılır ve dizinin son karesindeki özellik ilk karedeki özelliikle çakışır. Eğer özellik dizide bulunan dört karenin herhangi birisinde eşleşmezse aykırı değer olarak kabul edilir ve elenir

## Özellik Seçimi

Kamera hareketinin tahmini (*egomotion*) için sadece bir özellik alt kümesini kullanmak, algoritmanın hesaplama süresini önemli ölçüde azaltır. Bu nedenle çıkarılan ve eşleşen özelliklerin hepsini kullanmak yerine, bir kısmı kullanılır. Kamera hareketinin kesin olarak hesaplanması hem uzak hem de yakın özelliklerin kullanılmasını ve özelliklerin görüntü üzerinde eşit olarak dağıtılmasını gerektirir. Bunun için görüntü dikdörtgenlere (*bucket*) bölünür ve her bir dikdörtgen başına kullanılacak olan özellik sayısı belirlenir. Mittal (2018) görüntüyü  $50 \times 50$  piksel boyutunda dikdörtgenlere bölmüştür. Bu çalışmada, yer referansını daha doğru takip etmek için  $40 \times 40$  lık dikdörtgenler kullanılmıştır. Dikdörtgen boyutu küçüldükçe, görüntü daha fazla dikdörtgenlere bölünmüş olacağı için daha fazla sayıda özellik seçilmiş olur ve yer referansını kestirmedeki doğruluk artar. Fakat bu durum algoritmanın yavaşlamasına sebep olur. Bu tezde dikdörtgen boyutunu değiştirerek yapılan deneylerde, algoritma hızı ve yer referansını kestirmedeki doğruluk en uygun değer için  $40 \times 40$  lık dikdörtgenler olduğu saptanmıştır. Görüntü dikdörtgenlere bölündüğünde, her bir dikdörtgende sınırlı sayıda özellik korunur ve diğerleri atılır. Dikdörtgenler için tutulacak özelliklerin seçimi aşağıdaki maddelere göre yapılmıştır.

1. Dört ayırt edici sınıfa ait olan özellikler (blob maksimum, blob minimum, çizgi maksimum ve çizgi minimum) seçilmiştir.
2. Giriş görüntüsünün blob ve çizgi özelliklerinin çıkarılmasıyla oluşturulan görüntüdeki özelliklerin piksel değerleri en güçlüden en zayıfa doğru sıralanmıştır.
3. En güçlü özelliğin tutulması sağlanmıştır.
4. Adım 2 ve 3 bütün özellikler için tekrar edilmiştir.
5. Algoritmanın sonunda özelliklerden ilk  $n$  tanesinin seçilmesi sağlanmıştır (Cvišić and Petrovic 2015).

Bu çalışmada her bir dikdörtgen başına bir özellik seçilmiştir yani  $n = 1$ 'dir. Seçilen  $n$  sayısı arttıkça algoritma, yer referansını daha yüksek doğrulukla kestirir fakat kamera hareketi tahmini için geçirilen süre uzar ve algoritma yavaşlar.

## Kamera Hareketinin Tahmini

Kamera hareketinin tahmini veya kameranın poz kestirimi (*egomotion*), yer referansını kestirmek için gereken son adımdır. Burada kullanılan poz terimi, kameranın konum ve yönelimini belirtmektedir. Kameranın pozunu, bir dizi 3B nokta ve bu noktalara görüntülerde karşılık gelen 2B projeksiyonları verildiğinde kestirilebilir. Bu aşama için üçgenleme, P3P ve aykırı değerleri elemek için M-SAC algoritması kullanılmıştır.

## Üçgenleme

Bölüm 2’de bahsedilen lineer üçgenleme ile her iki görüntüde birbirine denk gelen noktalara karşılık gelen 3B noktaları bulunur.

$$P = \text{linearTriangulation}(\text{pts1}, \text{pts2}, P1, P2) \quad (56)$$

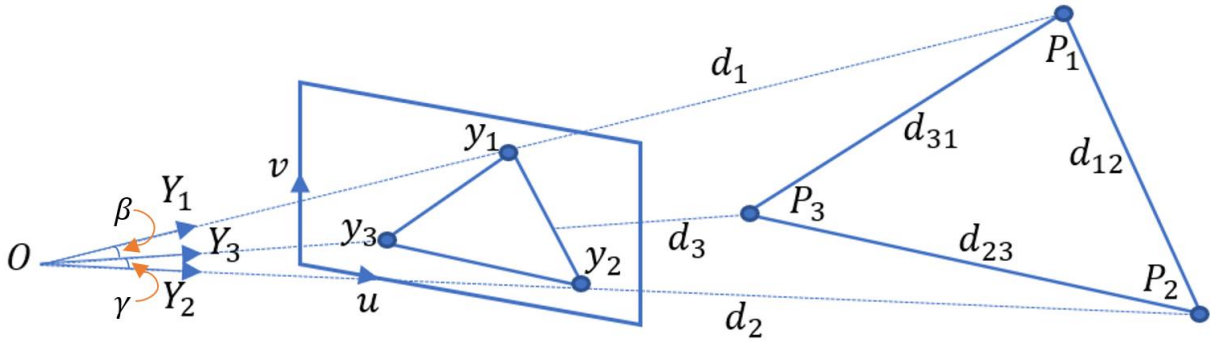
Denklem (56)’da verilen linearTriangulation fonksiyonunda pts1 ve pts2 sırasıyla görüntü 1 ve görüntü 2’deki birbirine denk gelen noktaların homojen koordinatları, P1 ve P2 sırasıyla birinci ve ikinci görüntülerin projeksiyon matrisleridir. Fonksiyonun çıkışı ise 3B’lu P noktasıdır.

## P3P Algoritması

Perspektif n noktası (PnP) dünyada n tane 3B nokta ve bunların görüntülerde karşılık gelen 2B projeksiyonları verildiğinde kalibre edilmiş bir kameranın pozunu tahmin etme problemidir. P3P probleminde  $n = 3$  olarak belirlenir ve P3P fonksiyonu denklem (57)’de temsil edilmiştir.

$$[R, t] = \text{P3P}(P_{1:3}, y_{1:3}) \quad (57)$$

Kalibre edilmiş kamera olduğu için, kameranın iç parametreleri belirlidir. Şekil 18’de 3B’de bulunan  $P_1, P_2$  ve  $P_3$  dünya noktalarının görüntü düzlemindeki iz düşümleri sırasıyla  $y_1, y_2$  ve  $y_3$  ile gösterilmektedir.  $P_1$  ve  $P_2$  noktaları arasındaki uzaklık  $d_{12}$ ,  $P_3$  ve  $P_1$  noktaları arasındaki uzaklık  $d_{31}$ ,  $P_2$  ve  $P_3$  noktaları arasındaki uzaklık  $d_{23}$ ,  $P_1, P_2$  ve  $P_3$  noktalarının optik merkez  $O$ ’ya olan uzaklıkları sırasıyla  $d_1, d_2$  ve  $d_3$  ile ifade edildiklerinde,  $d_1, d_2$  ve  $d_3$  mesafeleri denklem dizisi (58)-(60)’taki gibi kosinüs teoreminden hesaplanabilir. Denklem dizisi (58)-(60)’ta  $d_1$  ve  $d_3$  doğruları arasındaki açı  $\beta$ ,  $d_2$  ve  $d_3$  doğruları arasındaki açı  $\gamma$ ,  $d_1$  ve  $d_2$  doğruları arasında kalan açı ise  $\alpha$  olarak temsil edilmiştir.



Şekil 18. P3P Geometrisi (Carceroni 1997)

$$d_{12}^2 = d_1^2 + d_2^2 + 2d_1d_2 \cos \alpha \quad (58)$$

$$d_{31}^2 = d_1^2 + d_3^2 + 2d_1d_3 \cos \beta \quad (59)$$

$$d_{23}^2 = d_2^2 + d_3^2 + 2d_2d_3 \cos \gamma \quad (60)$$

$i = 1:3$  olmak üzere, 3B noktalar  $P_i = (x_i, y_i, z_i)$  ve bu noktalara karşılık gelen homojen 2B görüntü noktaları  $y_i = (u_i, v_i, 1)$  olmak üzere denklem (61) kameranın rotasyon (R) ve translasyonunu (t) bulmak için kullanılır.

$$\lambda_i y_i = R P_i + t \quad (61)$$

Denklem (61)'de  $\lambda_i$ , kamera merkezi O'dan her bir 3B noktaya uzaklıklar olan  $d_1, d_2$  ve  $d_3$  ile aynı anlama gelmektedir. Dolayısıyla denklem dizisi (58)-(60)'ta bilinmeyenler olan  $d_1, d_2$  ve  $d_3$  mesafeleri bulunduğunda  $\lambda_1, \lambda_2$  ve  $\lambda_3$  ölçek faktörleride bulunmuş olur. Denklem (61)'de bilinmeyen olarak kalan ifadeler  $\lambda_i, R, t$  'den sadece R ve t 'ye indirgenmiş olur. P3P probleminde geri kalan adımlar Persson and Nordberg (2018)'deki gibi yapılmaktadır.

### **RANSAC ve MSAC**

Aynı 3B'lu noktanın farklı görüntü kareleri boyunca iz düşümü olan 2B'lu görüntü noktaları, görüntü karşılıkları (*image correspondences*) olarak adlandırılır. Bu görüntü karşılıklarında aykırı noktalar (*outliers*) varsa, P3P algoritması hatalara eğimli hale gelir. RANSAC (Fischler and Bolles 1981) algoritması hataya sebep olan aykırı değerleri elemek için P3P algoritması ile birlikte kullanılır.

RANSAC algoritmasında, görüntü nokta karşılıklarından örneklem alınır ve bu örnekleme bulunan noktaların uyumlu noktalar (*inliers*) olduğu kabul edilir. P3P algoritması, 3 adet nokta karşılığı ile ilgilendiği için, uyumlu noktalar olduğu kabul edilen sette bulunan noktalardan 3 adet nokta karşılığı seçilir. Seçilen bu 3 adet nokta karşılığı ile kalibre edilmiş kameranın pozunu tahmin edilir. Daha sonra, uyumlu noktalar oldukları kabul edilen örneklemin içinde kalan diğer noktaların, tahmin edilen pozla ne kadar tutarlı olduğuna bakılır.

$$e = \frac{\text{aykırı nokta sayısı}}{\text{toplam nokta sayısı}} \quad (62)$$

Denklem (62)'daki aykırı nokta olasılığı e önceden belirlenen bir değerdir ve denklem (60) uyumlu nokta olasılığını göstermektedir.

$$\text{uyumlu nokta olasılığı} = 1 - e \quad (63)$$

Denklem (63), sadece bir adet uyumlu nokta olasılığıdır. Nokta karşılıklarından alınan örneklemelerin hepsinin uyumlu nokta olduğu varsayıldığı için, s seçilen örneklem sayısı olmak üzere denklem (64)'deki p, bütün örneklem için uyumlu nokta olasılığını göstermektedir.

$$p = (1 - e)^s \quad (64)$$

Denklem (65)'de  $1 - p$ , sadece bir iterasyonda yani rastgele seçilen bir örneklem kümesinde başarısızlığı gösterir.

$$1 - p = 1 - (1 - e)^s \quad (65)$$

Toplam iterasyon sayısı T ile gösterilirse, denklem (66) T iterasyondaki başarısızlıktır.

$$1 - p = (1 - (1 - e)^s)^T \quad (66)$$

Denklem (62)-(66) dizisinde e, p, s parametreleri bilirken, tek bilinmeyen denklem (63)'teki toplam RANSAC iterasyon sayısı T'dir.

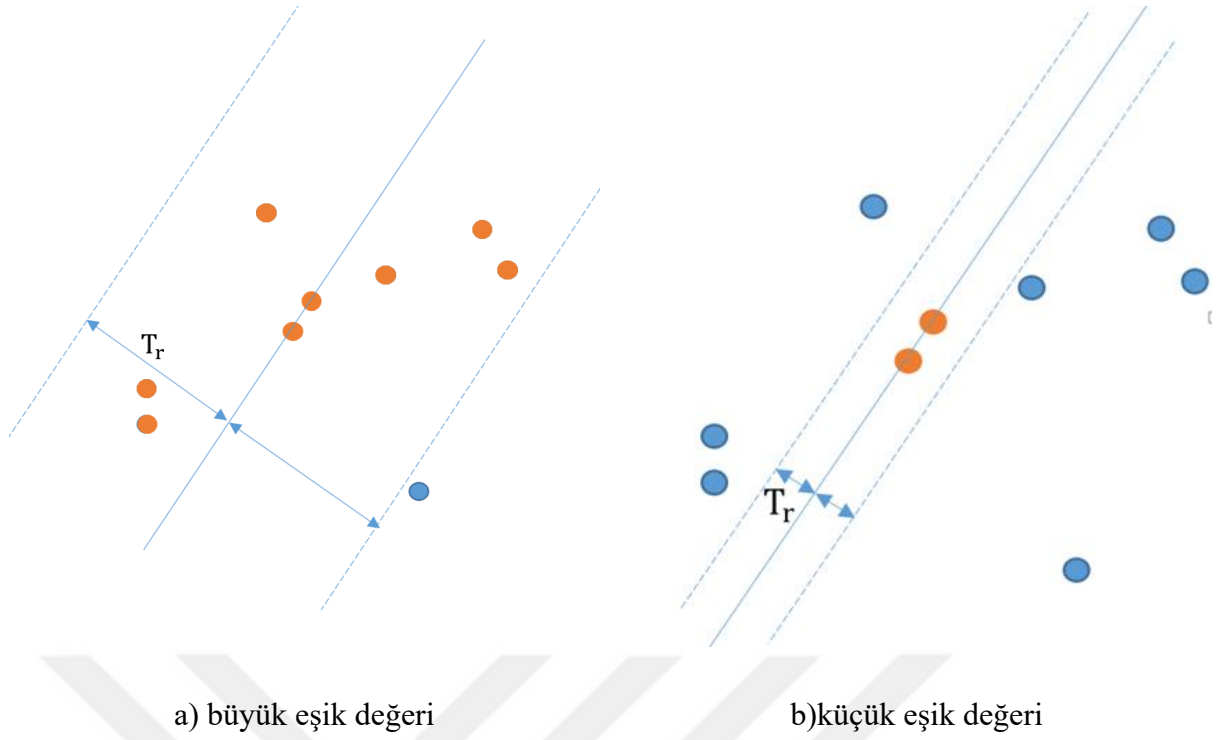
Denklem (67), T'nin nasıl bulunacağını gösterir.

$$T = \frac{\log(1 - p)}{\log(1 - (1 - e)^s)} \quad (67)$$

İterasyon sayısı T'yi bulmaktaki amaç, p olasılıkla, en az bir kez hatasız seçim yapmak için kaç kere tekrar yapılması gerektiğini bulmaktır.

Örneğin aykırı değer olasılığı  $e = 0,1$  ve toplam nokta sayısı  $s = 3$  olarak belirlenirse,  $p = 0,9$  olasılıkla en az bir kez hatasız seçim yapmak için gerekli olan iterasyon sayısı  $T = 2$  bulunur.

Hata Tolerans Eşiği ( $T_h$ ), bir veri örneğinin bir modelle uyuşup uyuşmadığını belirlemek için RANSAC tarafından kullanılır. RANSAC ile ilgili sorunlardan biri, Şekil 19-a'daki gibi eğer uyumlu değerleri dikkate almak için  $T_h$  eşiği çok yüksek ayarlanmışsa, veri örneğinde bulunan bütün noktalar uyumlu değer olarak kabul edilir veya Şekil 19-b'deki gibi  $T_h$  eşiği çok düşük ise uyumlu noktalar, aykırı değerlermiş gibi kabul edilebilir. Bu durum, RANSAC algoritmasının başarısız olmasına neden olur.



**Şekil 19.** RANSAC hata tolerans eşikğine göre uyumlu noktaların belirlenmesi

RANSAC maliyet fonksiyonu  $e_r$  hata değerini göstermek üzere denklem (68)'deki gibidir.

$$C = \sum_i \rho(e_{r_i}^2) \quad (68)$$

Denklem (68)'de Şekil 19'u açıklayan  $\rho(\cdot)$ , denklem (69)'daki gibi tanımlanır. Burada,  $T_h$  eşik değeridir.

$$\rho(e_{r_i}^2) = \begin{cases} 0 & e_{r_i}^2 < T_h^2 \\ \text{sabit} & e_{r_i}^2 \geq T_h^2 \end{cases} \quad (69)$$

Torr and Zisserman (2000), bu sorunun çözümü için denklem (70) ve denklem (71)'deki gibi yeni bir maliyet fonksiyonu önermişler ve yönteme MSAC (M kestiricisi ve konsensüs) ismini vermişlerdir.

$$C_2 = \sum_i \rho_2(e_{r_i}^2) \quad (70)$$

$$\rho_2(e_{r_i}^2) = \begin{cases} e_{r_i}^2 & e_{r_i}^2 < T_h^2 \\ T_h^2 & e_{r_i}^2 \geq T_h^2 \end{cases} \quad (71)$$

Denklem (71)'de aykırı değerlere hala sabit bir ceza verildiği görülmektedir fakat uyumlu noktalar verilere ne kadar iyi uyduklarına göre değerlendirilir.

## ARAŞTIRMA BULGULARI ve TARTIŞMA

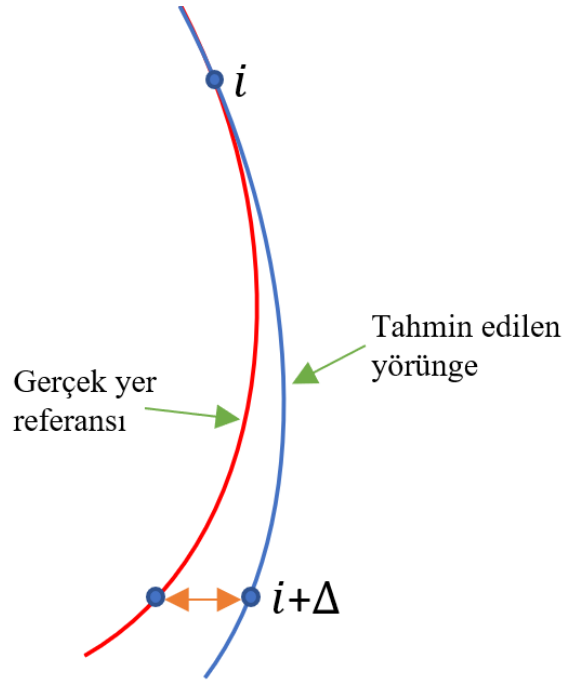
Bu bölümde, görsel odometri algoritmasının deneysel değerlendirilmesinin sonuçları sunulmuştur. Deneyler, erişime açık KITTI veri seti (Geiger *et al.* 2013) üzerinde gerçekleştirilmiştir. Deneyler MATLAB 2019b ortamında çalıştırılmıştır. MATLAB (*Matrix Laboratory*), MathWorks şirketi tarafından geliştirilmiş, bilim insanları tarafından yaygın bir şekilde kullanılan sayısal hesaplama yazılımı ve programlama dilidir. Deneyler, Intel Core i7-7500U CPU @2.70 GHz işlemci ve 8 GB RAM kapasiteli Windows 10 Pro işletim sistemi kullanan dizüstü bilgisayar ile yapılmıştır.

### Doğruluk Belirleme

Görsel odometri algoritmasının doğruluğunu test etmek için Sturm *et al.* (2012) tarafından yapılmış olan çalışmada bulunan iki hata metriği kullanılmıştır. Görsel odometri ile kestirilen pozların dizisi  $P_1, \dots, P_n$  ve yer referansında bulunan pozların dizisi  $Q_1, \dots, Q_n$  ile gösterilsin.

İlk hata metriği sabit bir zaman aralığı  $\Delta$ 'da kestirilen yörüngenin yerel doğruluğunu ölçen göreceli poz hatasıdır (*Relative Pose Error-RPE*). Şekil 20'de gösterildiği gibi, RPE  $i$ 'inci ve  $i+\Delta$ 'ıncı görüntüler arasındaki gerçek ve tahmin edilen poz arasındaki hatayı belirler. Göreceli poz hatası, yörüngedeki sapma hatasına karşılık gelir.





**Şekil 20.** Göreceli poz hatası (Sturm *et al.* 2015)

Göreceli poz hatası  $i$ . anda denklem (72)'deki gibi tanımlanır.

$$E_i = (Q_i^{-1}Q_{i+\Delta})^{-1}(P_i^{-1}P_{i+\Delta}) \quad (72)$$

$\Delta$ 'yı belirlemenin iki yolu vardır.  $\Delta = 1$  seçilirse, görüntü başına sapma (*drift*) hatası bulunur.  $\Delta$ , saniyedeki görüntü sayısına (*frame per second- FPS*) eşit olarak seçilirse, saniye başına sapma hatası bulunur. Bu tezde saniye başına sapma hatası bulunmuştur. KITTI veri setinde, bir saniyede 10 kare görüntü alındığı için,  $\Delta=10$  olarak alınmıştır.

Denklem (72)'de bulunan hatanın translasyonel bileşeninin karesel ortalama hatası (RMSE) denklem (73)'teki gibi olur.

$$\text{RMSE}(E_{1:n}, \Delta) = \left( \frac{1}{m} \sum_{i=1}^m \|\text{trans}(E_i)\|^2 \right)^{1/2} \quad (73)$$

$\text{trans}(E_i)$ , Göreceli poz hatası  $E_i$ 'nin translasyonel bileşenidir. Rotasyon hataları, kamera hareket ettiğinde translasyon hareketi olarak göründüklerinden dolayı genellikle açıkça hesaplanmaz. Denklem (73)'te  $m$  değeri kaçınıcı görüntüde durulacağını belirtmektedir.  $n$  adet görüntü içeren bir dizide,  $m = n - \Delta$ 'dır.

İkinci hata metriği Sturm *et al.* (2012) tarafından yapılmış olan çalışmada verilen metriğin bir versiyonu olan ve Aladem (2017) tarafından kullanılan metriktir. Bu metrik denklem (74)'te verilmiştir. İkinci metrik, mutlak translasyon hatasıdır (*absolute translation error-ATE*). Görsel odometri ile kestirilen yörünge ve  $x$ - $z$  düzleminde hareket eden aracın yer referansından elde edilen yörünge arasındaki fark hesaplanır.

$$T_i = ((Q_i^x - P_i^x)^2 + (Q_i^z - P_i^z)^2)^{1/2} \quad (74)$$

$Q^x, Q^z$ , yer referansın  $x, z$  translasyon bileşenleri ve  $P^x, P^z$  kestirilen pozun  $x, z$  translasyon bileşenleridir. Karesel ortalama hata (RMSE) denklem (75)'teki gibi olur.

$$\text{RMSE}(T_{1:n}) = \left( \frac{1}{n} \sum_{i=1}^n T_i^2 \right)^{1/2} \quad (75)$$

## KITTI Veri Seti

KITTI veri seti Geiger *et al.* (2013) tarafından LIDAR verilerini ve GPS konum ölçümlerini içeren stereo gri tonlama ve renkli görüntü setlerini içerir. Veri setini, bir otomobilin üzerine yerleştirilmiş bir adet OXTS RT 3003 ataletsel navigasyon sistemi (GPS/IMU), bir adet Velodyne HDL-64E lazer tarayıcı, iki adet 1,4 Megapiksel Point Grey Flea 2 (FL2-14S3M-C) siyah beyaz kamera, iki adet Point Grey Flea 2 (FL2-14S3C-C) renkli kamera ve dört adet Edmund Optics NT59-917 4-8 mm değişken odaklı lens kullanılarak, Almanya'nın Karlsruhe şehrinde farklı trafik senaryolarında aracı sürerek toplamışlardır. KITTI veri setindeki görüntüler  $1241 \times 376$  çözünürlükte dirler. Görüntüler gri tonlamalı olarak ve önceden görüntü düzeltme ve mercek bozukluğunu düzeltme işlemlerinden geçirilerek sunulmuştur. Veri setinin zorlu yönlerinden bazıları dinamik hareketli nesnelerin (araçlar, bisikletliler, yayalar) varlığı, araç hareket halindeyken farklı aydınlatma ve gölge koşulları, bazı veri dizilerinde köşe çıkarımının zor olduğu yeşillik alanların ve yeterli detayların bulunmadığı otoban varlığıdır.

KITTI veri setinde, ilk 11 (00-10) veri dizisi kendisine ait yer referans pozlarını içeren poses.txt klasörünü bulundurmaktadır. Bu klasörler,  $N$ , dizide bulunan görüntü sayısı olmak üzere,  $N \times 12$ 'lik poz bilgisinden oluşurlar.  $i$ 'inci görüntünün poz bilgisi  $i$ 'inci satırda yer almaktadır. Her satır, denklem (76)'da verilen  $4 \times 4$ 'lük homojen poz matrisinin ilk üç satırının denklem (77)'deki gibi tek bir satıra yazılmasıyla elde edilmiştir.

$$[R|t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (76)$$

$$r_{11} \quad r_{12} \quad r_{13} \quad t_x \quad r_{21} \quad r_{22} \quad r_{23} \quad t_y \quad r_{31} \quad r_{32} \quad r_{33} \quad t_z \quad (77)$$

Bu tezde yer referansı çizdirilirken  $x$  (sağ) ve  $z$  (ileri) translasyon bileşenleri  $t_x$  ve  $t_z$  kullanılmıştır.

KITTI veri setinde, bütün veri dizileri kendisine ait kalibrasyon dosyalarını (calib.txt) içermektedir. Kalibrasyon dosyasında, kameranın iç parametre değerleri ve stereo kameralar arasındaki uzaklık (*base-line*) değeri bulunur. Kamera indeksi  $P_{i \in \{0,1,2,3\}}$  olmak üzere,  $i = 0$  gri ölçek sol kamerayı,  $i = 1$  gri ölçek sağ kamerayı,  $i = 2$  renkli sol kamerayı ve  $i = 3$  renkli sağ kamerayı temsil eder. Özellik çıkarımında parlaklık, piksellerin yoğunluk değerleri için önemli bir bilgi olduğu için gri ölçekli görüntüler kullanılır ve  $P_{i \in \{0,1\}}$  gri ölçek kamera değerleri dikkate alınır. Kalibrasyon dosyasındaki her bir satır, denklem (78)'de verilen iç parametre matrisinin tek bir satıra yazılmasıyla elde edilir.

$$P_{i \in \{0,1\}} = \begin{bmatrix} f_u & 0 & c_x & -f_u b_x \\ 0 & f_v & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (78)$$

Denklem (78)'de verilen matrisin ilk üç sütunu iç parametre matrisidir.  $f_u$  ve  $f_v$  sırasıyla u ve v yönlerindeki odak uzaklığını,  $c_u$  ve  $c_v$  sırasıyla u ve v yönlerindeki bakış noktası değerlerini temsil etmektedir. Son sütunda verilen kameralar arasındaki uzaklık  $b_x$ 'i bulmak için,  $b_x = \left| -f_u b_x / f_u \right|$  işleminin yapılması gerekir.

### KITTI Veri Seti Değerlendirmesi

Deneyle, KITTI veri setinin 04, 05, 07, 09 ve 10 numaralı dizileri ile yapılmıştır. MATLAB ortamında, tezde sunulan algoritma ve Mittal (2018) ile deneysel sonuçlar elde edilmiştir. Bahsi geçen her veri setine ait olan yer referansları elde edilen sonuçlar ile birlikte çizdirilmiştir. Tezde sunulan algoritma ile elde edilen sonuçlar, Mittal (2018) ile elde edilen sonuçlar ile karşılaştırılmıştır. Daha sonra, her iki algoritma ile elde edilen sonuçların, RPE ve ATE hata değerleri verilerek kıyaslama yapılmıştır.

Şekil 21 ve Şekil 22'de, örnek olarak KITTI veri setinin 07 numaralı dizisine ait 121. stereo görüntü çifti verilmiştir.

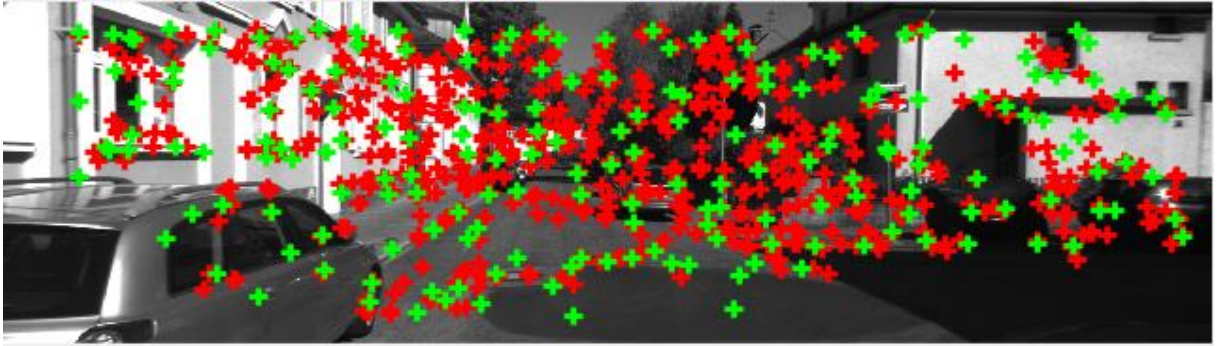


Şekil 21. KITTI- dizi 07'de bulunan sağ kameradan alınan 121.görüntü



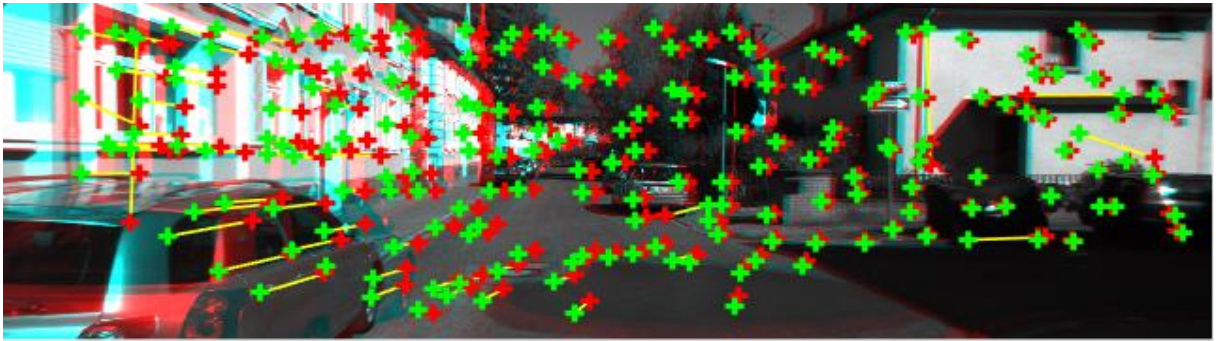
**Şekil 22.** KITTI- dizi 07’de bulunan sol kameradan alınan 121.görüntü

Şekil 23, KITTI- dizi 07’ye ait 121. sol kamera görüntüsünde  $40 \times 40$ ’lık dikdörtgenler kullanılarak en güçlü özelliklerin seçiminin yapıldığını göstermektedir.



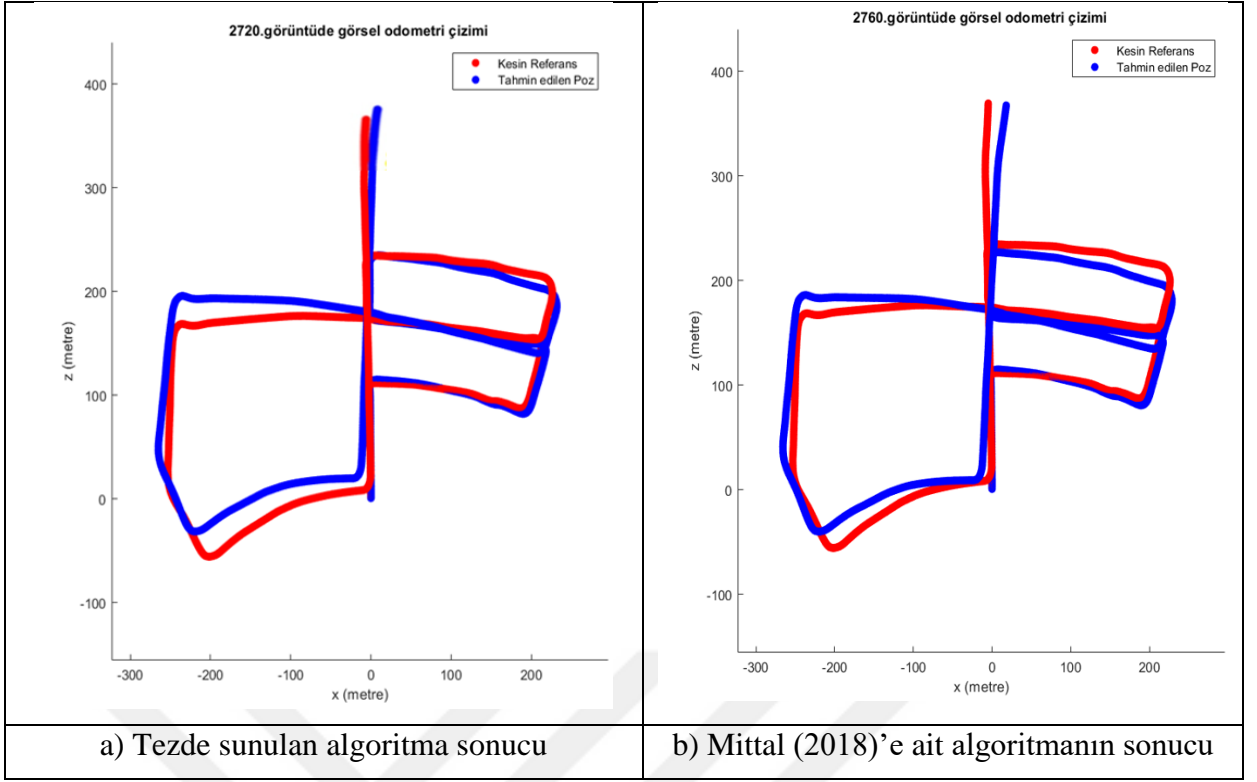
**Şekil 23.** 121. Görüntüde dikdörtgenler kullanarak özellik seçimi; Kırmızı işaretler zayıf özellikleri, yeşil işaretler güçlü özellikleri temsil etmektedir.

Şekil 24,  $t$  (121. görüntü) ve  $t - 1$  (120. görüntü) anlarında özellik eşleşmesini göstermektedir.

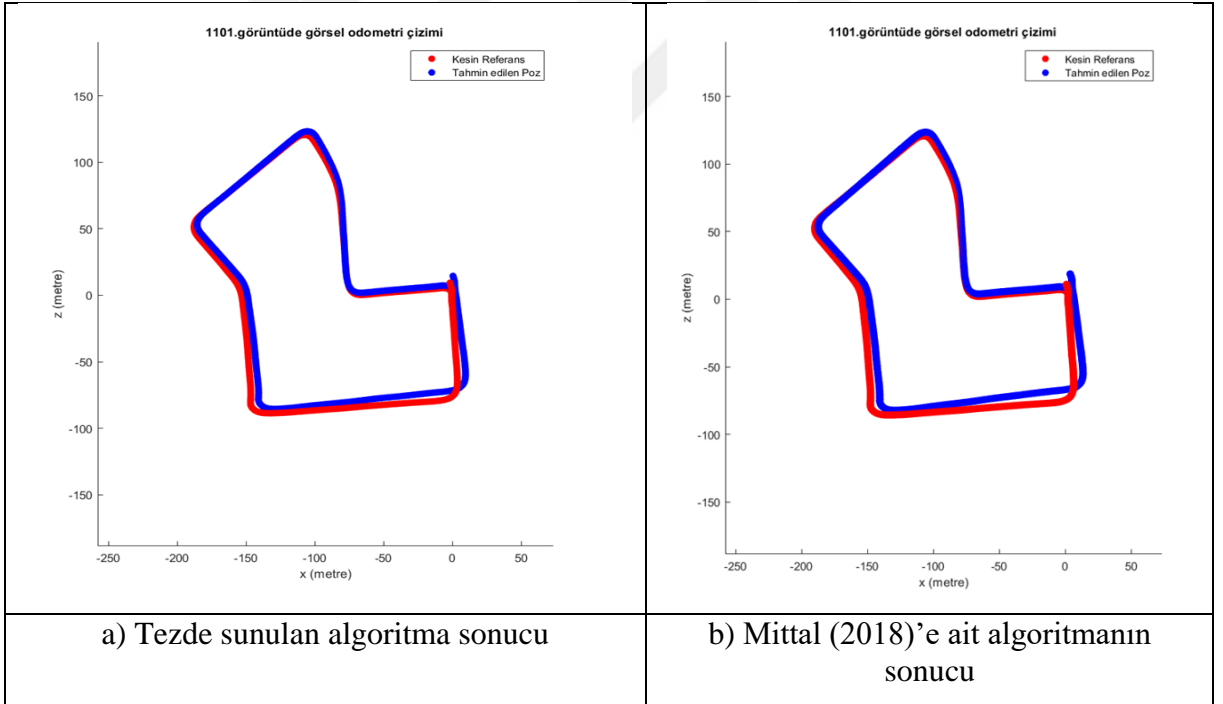


**Şekil 24.** Sol kamerada 120. ve 121. görüntüler arasında özellik eşleşmesi; Yeşil işaretler  $t$  anını ve kırmızı işaretler  $t - 1$  anını temsil etmektedir.

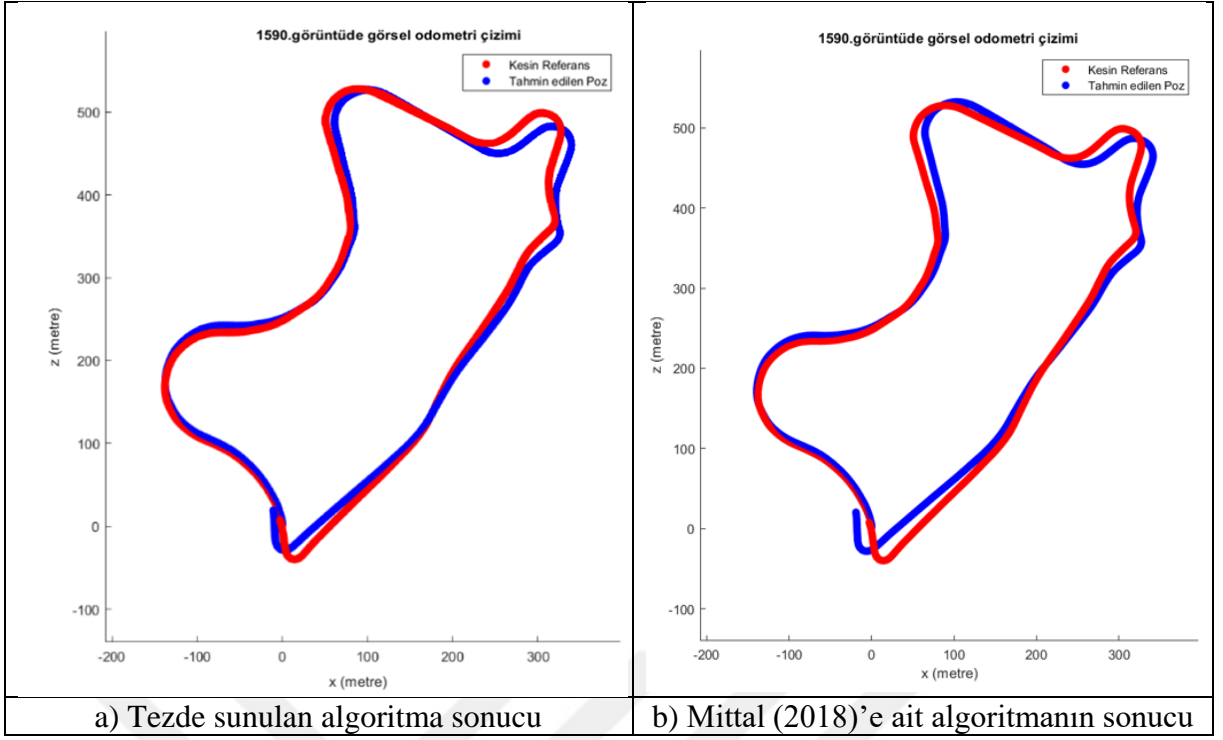
Bu tezde sunulan algoritma ve Mittal (2018) algoritma MATLAB ortamında, KITTI’nin 04, 05, 07, 09 ve 10 numaralı dizileri üzerinde çalıştırılmıştır. Elde edilen deneysel sonuçlar Şekil 25-29’da verilmiştir.



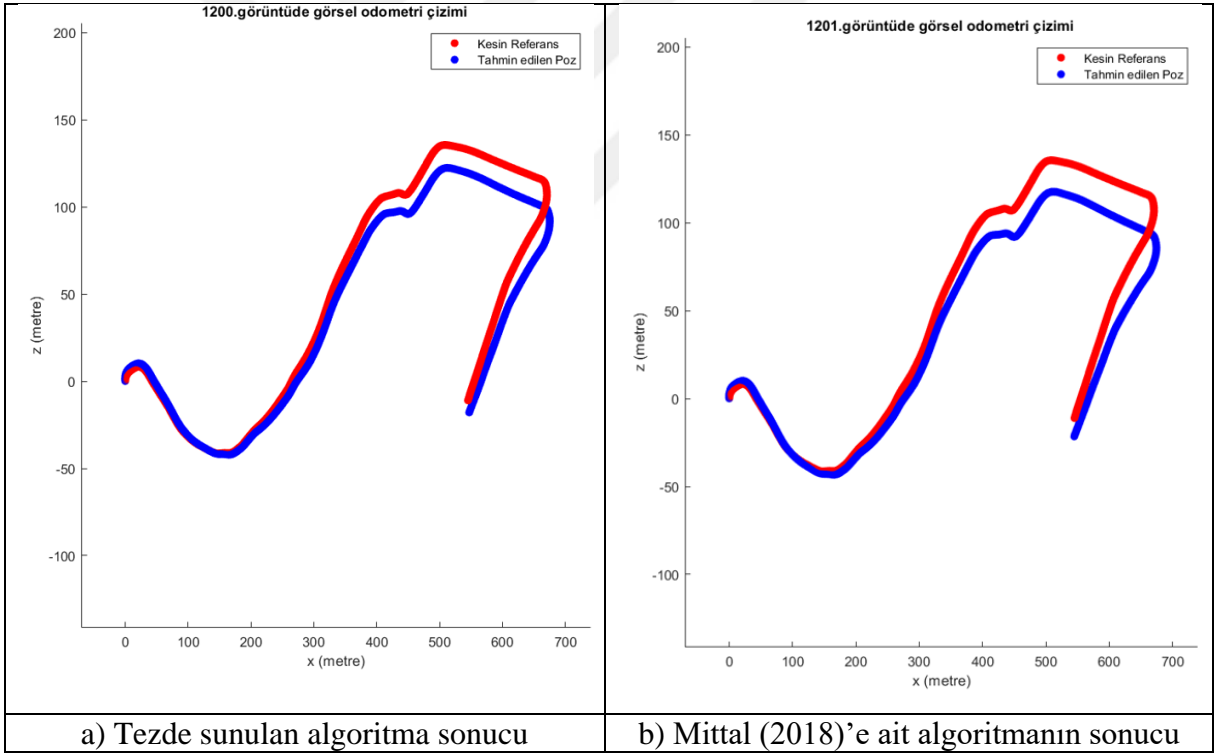
Şekil 25. KITTI dizi-05



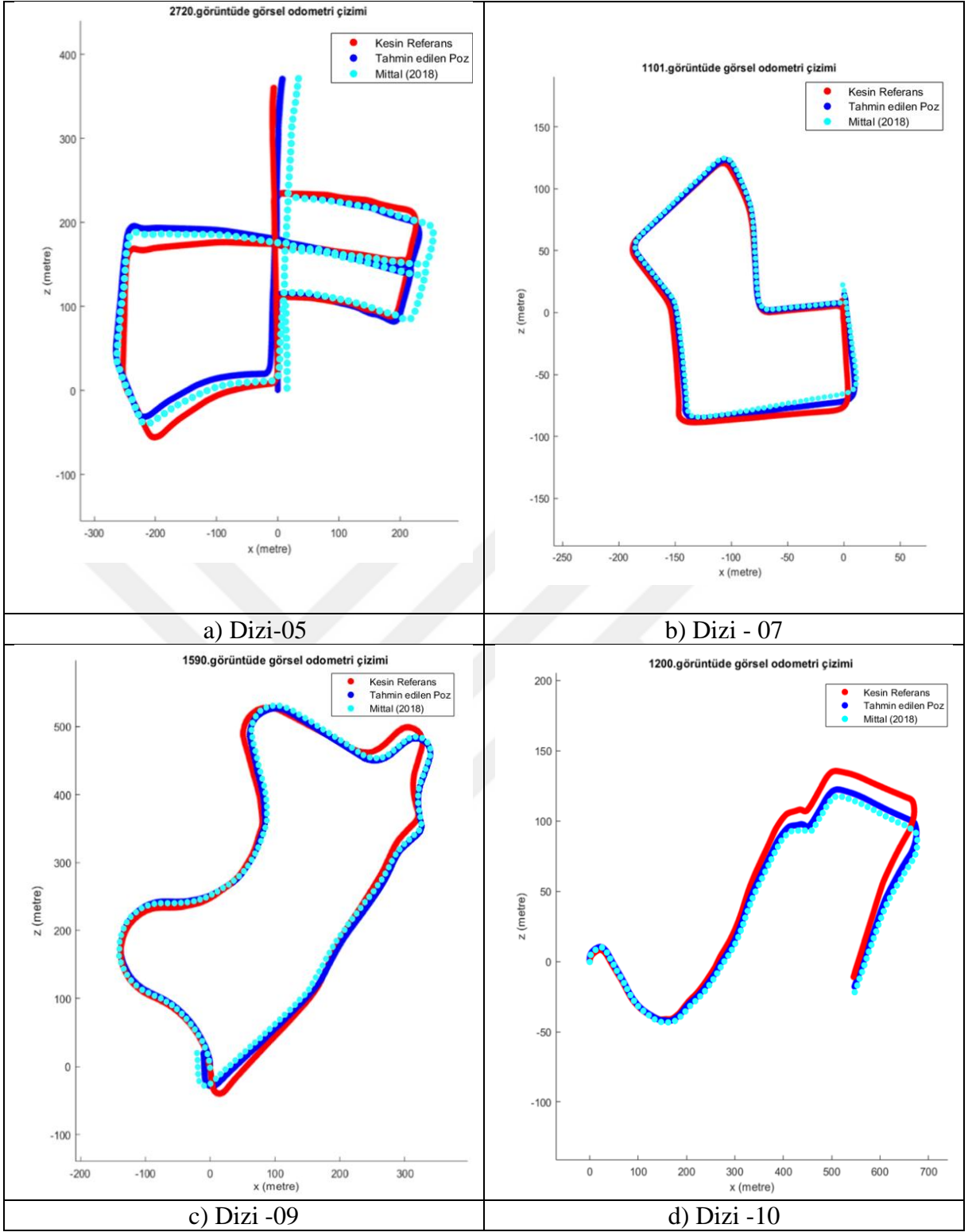
Şekil 26. KITTI dizi-07



Şekil 27. KITTI dizi-09



Şekil 28. KITTI dizi-10



**Şekil 29.** Kesin yer referansının, tezde sunulan algoritma ve Mittal (2018)'e ait algoritma sonuçlarının birlikte çizdirilmesi

Tablo 1, dizi 05, dizi 07, dizi 09 ve dizi 10 için elde edilen RPE ve ATE hataları ile her bir dizinin kaç km uzunluğunda olduğunu ve kaç tane görüntü içerdiğini göstermektedir. Tabloda, her iki algoritma için RPE – ATE metriklerine göre elde edilen hatalar nümerik olarak karşılaştırılmıştır. KITTI veri setinin bahsi geçen bütün dizilerinde, veri seti oluşturulurken saniyede 10 kare görüntü kaydedildiğinden RPE hesaplanırken  $\Delta = 10$  olarak alınmıştır.



**Tablo 1.** KITTI veri setinde görsel odometri sonuçları ve hata karşılaştırması

Dizi	Uzunluk [km]	Görüntü Sayısı	Algoritma	RMSE(RPE) [m/s]	RMSE (ATE) [m]
KITTI 05	2,2046	2760	Tez	0,1384	12,5961
			Mittal (2018)	0,1402	12,7599
KITTI 07	0,6944	1100	Tez	0,1173	3,9000
			Mittal (2018)	0,1198	3,9832
KITTI 09	1,7025	1590	Tez	0,2567	5,6567
			Mittal (2018)	0,2806	6,1833
KITTI 10	0,9178	1200	Tez	0,1210	4,9562
			Mittal (2018)	0,1328	5,4395

Tablo 1’de elde edilen RMSE(RPE) ve RMSE(ATE) hata karşılaştırmalarına göre, bu tezde sunulan algoritmanın yer referansını takip etmede daha başarılı olduğu görülmektedir. Başarının artmasına, görüntüyü daha fazla sayıda dikdörtgenlere (*bucket*) bölerek daha fazla sayıda özellik tutulması etken olmuştur. Ayrıca Mittal (2018)’in çalışmasında zaten önemli bir avantaj olan özellik seçimi kısmında kullanılan seçici yöntemler bu tez çalışmasında da aynen korunmuştur.

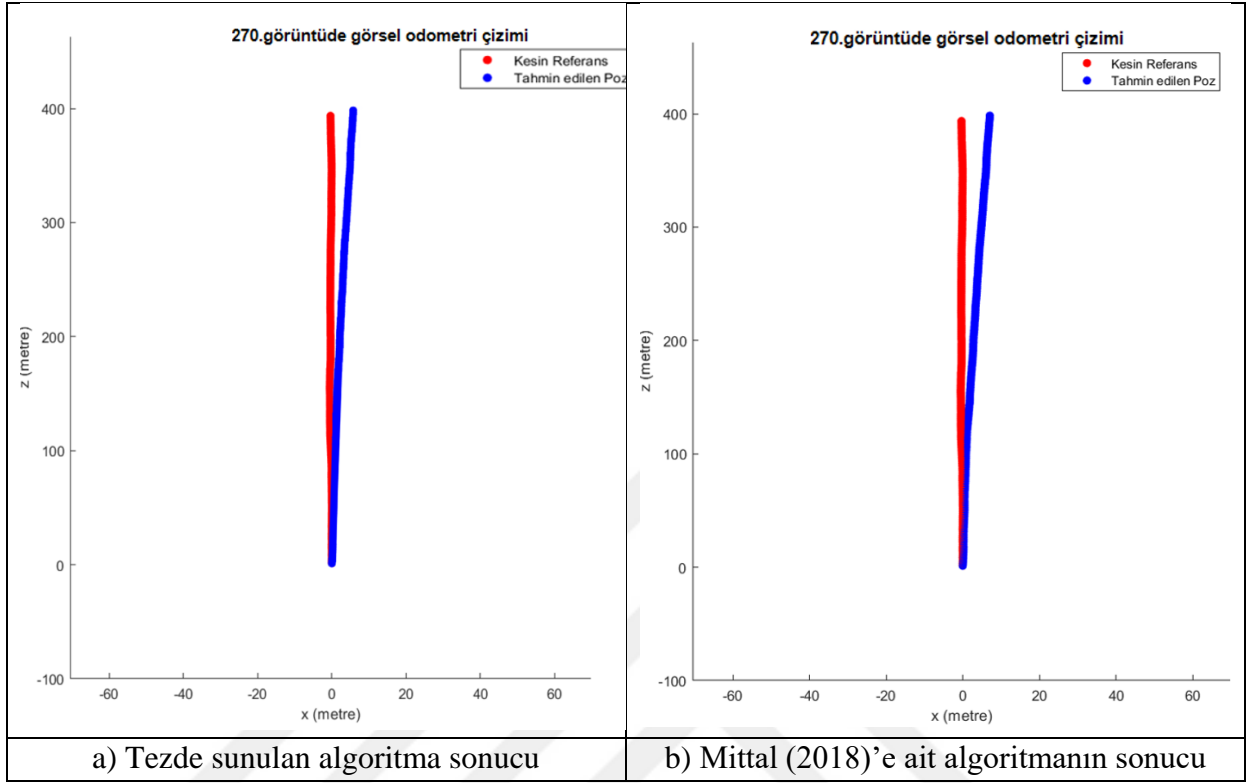
Ancak, araç hareket halindeyken farklı aydınlatma ve gölge koşulları, köşe çıkarımının zor olduğu yeşillik alanların ve yeterli detayların bulunmadığı sokakların varlığı gibi etkenler görüntülerde ayırt edici özelliklerin sayısının az olmasına neden olur. Bu durum, özellik çıkarımını zorlaştırır ve özellik takibinde yeteri kadar noktanın elde edilememesine yol açar ve yer referansından sapma artar. Bu durumun bir örneği olarak KITTI’nin 04 numaralı dizisine ait olan 270. görüntü Şekil 30’da verilmiştir.



**Şekil 30.** KITTI-04 dizisinde 270. görüntü



Bu tezde sunulan algoritma ve Mittal (2018)'e ait algoritma MATLAB ortamında, KITTI'nin 04 numaralı dizisi üzerinde çalıştırılmıştır. Elde edilen deneysel sonuç Şekil 31'de verilmiştir.



Şekil 31. KITTI dizi-04

Tablo 2'de RPE ve ATE hata karşılaştırmaları verilmiştir. Bu dizide, kestirilen poz bilgileri ile elde edilen yörünge'nin yer referansından sapma miktarının fazla olduğu görülmektedir. Bunun nedeni, yukarıda belirtildiği gibi KITTI- dizi 04'te ayırt edici özelliklerin az sayıda bulunmasıdır.

Tablo 2. KITTI dizi-04 hata karşılaştırması

Dizi	Uzunluk [km]	Görüntü Sayısı	Algoritma	RMSE(RPE) [m/s]	RMSE (ATE) [m]
KITTI 04	0.3936	270	Tez	0,1821	1,6124
			Mittal (2018)	0,1852	1,6398

## SONUÇ ve ÖNERİLER

Bu tezde, bir stereo kamera görsel odometri algoritmasında yer alan unsurların etkileri ve değerlendirilmesi sunulmuştur. Görsel odometri, görsel girdiden hareket yörüngesini tahmin etme işlemini yapan güçlü bir tekniktir. Otonom araçlarda, kamera ve görsel odometri merkezi bir rol oynamaktadır. Bu tez kapsamında, bilgisayarlı görme ve çoklu görüş geometrisindeki temel kavramlar sunulmuş ve bir görsel odometri algoritmasının uyarlaması ayrıntılı olarak tartışılmıştır. Özellik tabanlı algoritma, Harris köşe ve Gauss'ların Farkı kabarcık özelliklerini algılamaya ve onları üçgenleyerek izlemeye dayanır. Görüntü boyunca özellik dağıtımı için Maksimal Olmayanı Bastırma, Mutlak Farkların Toplamı, özellik eşleşmesi ve takibi için dairesel eşleme, izleme sırasında özellik seçimini hızlandırmak için bucket kullanımı gösterilmiştir. Algoritma daha sonra Perspektif 3 Nokta (P3P) yöntemiyle kameranın pozunu kestirmek için kameranın rotasyonunu ve translasyonunu kestirmiştir. Tezde sunulan algoritmanın aykırı değerleri elemek için Rastgele Örnek Konsensüs (RANSAC) algoritmasının yerine M-kestiricisi Örnek ve Konsensüs (M-SAC) algoritmasını kullanması, çalışmanın ayırt edici bir özelliğidir.

Tezde sunulan görsel odometri algoritması, otonom sürüş algoritmalarını değerlendirmek için yaygın olarak kullanılan KITTI verisetinde değerlendirilmiştir. Algoritma, döngü kapama (loop-closure) aşaması içermemesine rağmen, döngü kapamanın olduğu KITTI 07 ve KITTI 09 numaralı dizilerde rekabetçi bir şekilde performans gösterebilmiştir. Tezde sunulan algorithmadan elde edilen veriler, erişime sunulan Mittal (2018) çalıştırılmasıyla elde edilen veriler ile karşılaştırılmıştır. Bu karşılaştırmaya göre, yer referansını takip etmede başarının arttığı tespit edilmiştir.

Bu çalışmada ve literatürde yapılan çalışmalar incelendiğinde özellik tabanlı görsel odometri algoritmalarının kentsel alanlarda daha başarılı olduğu gözlemlenmiştir. Bunun nedeni özellik çıkarımı ve takibi için ayırt edici özelliklerin kentsel alanlarda daha fazla olmasıdır. Aynı zamanda, dikey ve yatay çizgi özelliklerinin kentsel alanlarda fazla olmasından dolayı, görüntülerden çizgi özelliklerinin çıkarımının daha başarılı olması beklenir. Gelecek çalışmalarda görsellerden çizgi çıkarımı yapılabilir ve Perspektif n Çizgi (*Perspective n Lines-PnL*) algoritmasıyla kameranın hareket kesitirimi hesaplanabilir. Ayrıca, doğrudan görsel odometri yöntemleri özellik tabanlı görsel odometri yöntemlerine göre özelliklerin az olduğu ortamlarda bile güçlü kamera hareketi tahmini yapabilmektedir. Bu yüzden ayırt edici

özelliklerin az olduđu otoban ve ağaçlarla çevrili yollarda özellik tabanlı görsel odometri yöntemleri yerine, doğrudan (görünüşe dayalı) görsel odometri yöntemleri tercih edilebilir. Görsel odometride, hem doğrudan hemde özellik tabanlı yöntemlerin bir kombinasyonu olan hibrit algoritmalarından yararlanılabilir. Hibrit yöntemler ile, görüntülerde bulunan güçlü özellikler ve tüm görüntünün piksel yoğunluk bilgileri birleştirilerek veri setinin farklı özelliklerini kapsayan güçlü bir algoritma kurulması sağlanabilir.

Tezde sunulan görsel odometri algoritması için veri setinin özelliklerine göre önerilen algoritmaların derin öğrenme yöntemleri ile entegrasyonu daha faydalı olabilir. Derin öğrenme algoritmaları kullanarak veri setinin özelliklerine göre hangi görsel odometri yönteminin daha uygun olduđu tespit edilip, o veri setine uygun görsel odometri yönteminin seçilmesi sağlanabilir.

Gelecek çalışmalarda otonom bir araç üzerine yerleştirilmiş stereo kamera ile farklı senaryolar için yeni veri setleri oluşturulup, yukarıda bahsi geçen görsel odometri yöntemlerinin oluşturulan veri setleri üzerinde denenmesi planlanmaktadır.

## KAYNAKLAR

- Aladem, M. 2017. Robust real-time visual odometry for autonomous ground vehicles (Doctoral dissertation)
- Aqel, M. O. A., Marhaban, M. H., Saripan, M. I., & Ismail, N. B. 2016. Review of visual odometry: types, approaches, challenges, and applications. SpringerPlus, 5(1). doi:10.1186/s40064-016-3573-7
- Bay, H., Tuytelaars, T., & Van Gool, L. 2006. Surf: Speeded up robust features. In European conference on computer vision (pp. 404-417). Springer, Berlin, Heidelberg.
- Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Siegwart, R. 2016. The EuRoC micro aerial vehicle datasets. The International Journal of Robotics Research, 35(10), 1157-1163.
- Cai, G.-S., Lin, H.-Y., & Kao, S.-F. 2019. Mobile Robot Localization using GPS, IMU and Visual Odometry. Paper presented at the 2019 International Automatic Control Conference (CACS).
- Carceroni, R. L., & Brown, C. M. 1997. Numerical methods for model-based pose recovery. Technical report 659, 1997. Rapport technique. University of Rochester, Computer Science Department, New York. Ref Type: Report.
- Cheng, Y., Maimone, M., & Matthies, L. 2005. Visual odometry on the Mars exploration rovers. Paper presented at the 2005 IEEE International Conference on Systems, Man and Cybernetics.
- Chiodini, S., Giubilato, R., Pertile, M., & Debei, S. 2020. Retrieving Scale on Monocular Visual Odometry Using Low Resolution Range Sensors. IEEE Transactions on Instrumentation and Measurement.
- Chotrov, D., Uzunova, Z., Yordanov, Y., & Maleshkov, S. 2018. Mixed-Reality Spatial Configuration with a ZED Mini Stereoscopic Camera. Paper presented at the 2018 Technologies and Education for a Smart World
- Cognard, T., 2014. Stereo camera.JPG CC-BY-SA-3.0. Own work, <https://commons.wikimedia.org/w/index.php?curid=33843774> (28.07.2020).
- Collins, R. 2007. Introduction to Computer Vision Computer Vision I CSE/EE486. Penn State University, Lecture 19: Essential and Fundamental Matrices, [http://www.cse.psu.edu/~rtc12/CSE486/lecture19\\_6pp.pdf](http://www.cse.psu.edu/~rtc12/CSE486/lecture19_6pp.pdf) (21.05.2020).
- Corke, P. 2017. Image Formation, Robotics vision and control: Fundamental algorithms in MATLAB® second, completely revised (Vol. 118): Springer, 319-356.
- Cvišić, I., & Petrović, I. 2015. Stereo odometry based on careful feature selection and tracking. In 2015 European Conference on Mobile Robots (ECMR) (pp. 1-6). IEEE..
- de la Escalera, A., Izquierdo, E., Martín, D., Musleh, B., García, F., & Armingol, J. M. 2016. Stereo visual odometry in urban environments based on detecting ground features. Robotics and Autonomous Systems, 80, 1-10.
- Delmas, P., & Gee, T. 2019. Stereo camera visual odometry for moving urban environments. Integrated Computer-Aided Engineering, 26(3), 243-256. doi:10.3233/ica-190598
- Dias, N., & Laureano, G. 2019. Accurate Stereo Visual Odometry Based on Keypoint Selection. Paper presented at the 2019 Latin American Robotics Symposium (LARS), 2019 Brazilian Symposium on Robotics (SBR) and 2019 Workshop on Robotics in Education (WRE).
- Engel, J., Koltun, V., & Cremers, D. 2018. Direct Sparse Odometry. Ieee Transactions on Pattern Analysis and Machine Intelligence, 40(3), 611-625. doi:10.1109/tpami.2017.2658577

- Fanani, N., Ochs, M., Bradler, H., & Mester, R. 2016. Keypoint trajectory estimation using propagation based tracking. Paper presented at the 2016 IEEE Intelligent Vehicles Symposium (IV).
- Fanani, N., Stürck, A., Ochs, M., Bradler, H., & Mester, R. 2017. Predictive monocular odometry (PMO): What is possible without RANSAC and multiframe bundle adjustment? *Image and Vision Computing*, 68, 3-13.
- Feng, J., Zhang, C., Sun, B., & Song, Y. 2017. A fusion algorithm of visual odometry based on feature-based method and direct method. Paper presented at the 2017 Chinese Automation Congress (CAC).
- Fischler, M. A., & Bolles, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381-395.
- Geiger, A., Lenz, P., & Urtasun, R. 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. Paper presented at the 2012 IEEE Conference on Computer Vision and Pattern Recognition.
- Geiger, A., Ziegler, J., & Stiller, C. 2011. Stereoscan: Dense 3d reconstruction in real-time. In 2011 IEEE intelligent vehicles symposium (IV) (pp. 963-968). Ieee.
- Hartley, R., & Zisserman, A. 2003. *Epipolar Geometry and the Fundamental Matrix. Multiple view geometry in computer vision: Cambridge university press. United Kingdom*, 239-259.
- Hata, K., & Savarese, S. 2018. CS231 Course Notes 3: Epipolar Geometry: Stanford Universty, [https://web.stanford.edu/class/cs231a/course\\_notes/03-epipolar-geometry.pdf](https://web.stanford.edu/class/cs231a/course_notes/03-epipolar-geometry.pdf) (11.18.2019).
- He, M., Zhu, C. Z., Huang, Q., Ren, B. S., & Liu, J. T. 2020. A review of monocular visual odometry. *Visual Computer*, 36(5), 1053-1065. doi:10.1007/s00371-019-01714-6
- Janai, J., Güney, F., Behl, A., & Geiger, A. 2017. Computer vision for autonomous vehicles: Problems, datasets and state-of-the-art. arXiv preprint arXiv:1704.05519.
- Kitani, K. 2017. Triangulation 16-385 *Computer Vision: Carnegie Mellon The Robotic Institute* [http://www.cs.cmu.edu/~16385/s17/Slides/11.4\\_Triangulation.pdf](http://www.cs.cmu.edu/~16385/s17/Slides/11.4_Triangulation.pdf) (17.06.2020)
- Kocic, J., Jovicic, N., & Drndarevic, V. 2018. *Sensors and Sensor Fusion in Autonomous Vehicles*.
- Kornia Developers 2019. <https://kornia.readthedocs.io/en/latest/governance.html>
- Kos, T., Markežic, I., & Pokrajcic, J. 2010. Effects of Multipath Reception on GPS Positioning Performance. In M. Grgic, J. Bozek, & S. Grgic (Eds.), *Proceedings Elmar-2010* (pp. 399-402).
- Kuutti, S., Fallah, S., Katsaros, K., Dianati, M., McCullough, F., & Mouzakitis, A. 2018. A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications. *IEEE Internet of Things Journal*, 5(2), 829-846. doi:10.1109/jiot.2018.2812300
- Li, A., Ruan, X. G., Huang, J., Zhu, X. Q., & Wang, F. 2019. Review of vision-based Simultaneous Localization and Mapping.
- Li, R. H., Wang, S., Long, Z. Q., Gu, D. B., & Ieee. 2018. UnDeepVO: Monocular Visual Odometry through Unsupervised Deep Learning. In 2018 Ieee International Conference on Robotics and Automation (pp. 7286-7291).
- Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2), 91-110.
- Lu, X. X. 2018. A Review of Solutions for Perspective-n-Point Problem in Camera Pose Estimation. Paper presented at the *Journal of Physics: Conference Series*.
- Maklouf, O., & Adwaib, A. 2014. Performance Evaluation of GPS\INS Main Integration Approach. *World Acad Sci Eng Technol Int J Mech Aerosp Ind Mechatron Eng*, 8(2), 476-484.

- Mellish, B., 2005. How a pinhole camera works. Own work, <https://commons.wikimedia.org/wiki/File:Pinhole-camera.png> (28.07.2020).
- Mirabdollah, M. H., & Mertsching, B. 2015. Fast techniques for monocular visual odometry. Paper presented at the German Conference on Pattern Recognition.
- Mittal, M. 2018. Stereo Odometry SOFT-MATLAB Implementation of Visual Odometry using SOFT algorithm. <https://github.com/Mayankm96/Stereo-Odometry-SOFT> (04.03.2020)
- Mohamed, S. A. S., Haghbayan, M.-H., Westerlund, T., Heikkonen, J., Tenhunen, H., & Plosila, J. 2019. A Survey on Odometry for Autonomous Navigation Systems. *IEEE Access*, 7, 97466-97486. doi:10.1109/access.2019.2929133
- Moravec, H. P. 1980. Obstacle avoidance and navigation in the real world by a seeing robot rover. Retrieved from
- Morvan, T., 2018. Projective Geometry, <http://epixea.com/research/multi-view-coding-thesis.html> (20.07.2020).
- Mur-Artal, R., & Tardós, J. D. 2017. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5), 1255-1262.
- Neubeck, A., & Van Gool, L. 2006. Efficient non-maximum suppression. In 18th International Conference on Pattern Recognition (ICPR'06) (Vol. 3, pp. 850-855). IEEE.
- Nister, D. 2003. An efficient solution to the five-point relative pose problem. In 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings. (Vol. 2, pp. II-195). IEEE.
- Nistér, D., Naroditsky, O., & Bergen, J. 2004. Visual odometry. Paper presented at the Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.
- Nordmann (norro), A., 2008. CC-BY-SA-3.0. Own illustration, [https://commons.wikimedia.org/wiki/File:Aufnahme\\_mit\\_zwei\\_Kameras.svg](https://commons.wikimedia.org/wiki/File:Aufnahme_mit_zwei_Kameras.svg) (28.07.2020).
- Noureldin, A., Karamat, T. B., & Georgy, J. 2013. Fundamentals of inertial navigation, satellite-based positioning and their integration: Springer Science & Business Media.
- Pereira, F., Luft, J., Ilha, G., Sofiatti, A., & Susin, A. 2017. Backward Motion for Estimation Enhancement in Sparse Visual Odometry.
- Persson, M., & Nordberg, K. 2018. Lambda twist: An accurate fast robust perspective three point (P3P) solver. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 318-332).
- Poddar, S., Kottath, R., & Karar, V. 2018. Evolution of visual odometry techniques. arXiv preprint arXiv:1804.11142.
- Ren, S., He, K., Girshick, R., & Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Paper presented at the Advances in neural information processing systems.
- Rhee, J., & Seo, J. 2019. Low-Cost Curb Detection and Localization System Using Multiple Ultrasonic Sensors. *Sensors*, 19(6), 1389. doi:10.3390/s19061389
- Rosten, E., & Drummond, T. 2006. Machine learning for high-speed corner detection. Paper presented at the European conference on computer vision.
- Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. 2011. ORB: An efficient alternative to SIFT or SURF. Paper presented at the 2011 International conference on computer vision.
- Scaramuzza, D. 2019. Vision Algorithms for Mobile Robotics. Universty of Zurich UZH. Lecture 02 - Image Formation 1: perspective projection and camera models. [http://rpg.ifi.uzh.ch/docs/teaching/2019/02\\_image\\_formation\\_1.pdf](http://rpg.ifi.uzh.ch/docs/teaching/2019/02_image_formation_1.pdf) (14.12.2019)
- Scaramuzza, D. 2019. Vision Algorithms for Mobile Robotics. Universty of Zurich UZH. Lecture 07 - Multiple-view geometry.

[http://rpg.ifi.uzh.ch/docs/teaching/2019/07\\_multiple\\_view\\_geometry\\_1.pdf](http://rpg.ifi.uzh.ch/docs/teaching/2019/07_multiple_view_geometry_1.pdf)

(14.12.2019)

- Scaramuzza, D., & Fraundorfer, F. 2011. Visual Odometry [Tutorial]. *IEEE Robotics & Automation Magazine*, 18(4), 80-92. doi:10.1109/mra.2011.943233
- Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. 2011. Perception. *Introduction to autonomous mobile robots: MIT press, France*, 101-262.
- Sturm, J., Engelhard, N., Endres, F., Burgard, W., & Cremers, D. 2012. A benchmark for the evaluation of RGB-D SLAM systems. Paper presented at the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems.
- Sturm, J., Steinbrücker, F., Engel, J., Kerl, C., Cremers, D. 2015. Dense Localization and Mapping. Computer Vision Group TUM <http://jsturm.de/publications/data/sturm12pcl.pdf> (01.08.2020)
- Taketomi, T., Uchiyama, H., & Ikeda, S. 2017. Visual SLAM algorithms: a survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications*, 9(1). doi:10.1186/s41074-017-0027-2
- Tomasi, C., & Kanade, T. 1991. Detection and tracking of point features.
- Torr, P. H., & Zisserman, A. 2000. MLESAC: A new robust estimator with application to estimating image geometry. *Computer vision and image understanding*, 78(1), 138-156.
- Von Gioi, R. G., Jakubowicz, J., Morel, J.-M., & Randall, G. 2008. LSD: A fast line segment detector with a false detection control. *Ieee Transactions on Pattern Analysis and Machine Intelligence*, 32(4), 722-732.
- Von Gioi, R. G., Jakubowicz, J., Morel, J.-M., & Randall, G. 2012. LSD: a line segment detector. *Image Processing On Line*, 2, 35-55.
- Wang, D. B., Liang, H. W., Zhu, H., & Zhang, S. 2014. A Bionic Camera-Based Polarization Navigation Sensor. *Sensors*, 14(7), 13006-13023. doi:10.3390/s140713006
- Wen, S., Zhao, Y., Zhang, H., Lam, H. K., & Manfredi, L. 2020. Joint optimization based on direct sparse stereo visual-inertial odometry. *Autonomous Robots*, 1-19.
- Woo, A., Fidan, B., & Melek, W. W. 2018. Localization for Autonomous Driving, *Handbook of Position Location: Theory, Practice, and Advances, Second Edition*, 1051-1087.
- Yoon, S.-J., & Kim, T. 2019. Development of Stereo Visual Odometry Based on Photogrammetric Feature Optimization. *Remote Sensing*, 11(1), 67.
- Yousif, K., Bab-Hadiashar, A., & Hoseinnezhad, R. 2015. An Overview to Visual Odometry and Visual SLAM: Applications to Mobile Robotics. *Intelligent Industrial Systems*, 1(4), 289-311. doi:10.1007/s40903-015-0032-7
- Yurtsever, E., Lambert, J., Carballo, A., & Takeda, K. 2020. A Survey of Autonomous Driving: Common Practices and Emerging Technologies. *IEEE Access*, 8, 58443-58469. doi:10.1109/access.2020.2983149
- Zekavat, R., & Buehrer, R. M. 2019. Wireless positioning systems: Operation, application, and comparison.
- Zhang, F., Stahle, H., Chen, G., Simon, C. C. C., Buckl, C., & Knoll, A. 2012. A sensor fusion approach for localization with cumulative error elimination.
- Zhang, Y., Zhang, H., Wang, G., Yang, J., & Hwang, J. N. 2019. BUNDLE ADJUSTMENT FOR MONOCULAR VISUAL ODOMETRY BASED ON DETECTIONS OF TRAFFIC SIGNS. *IEEE Transactions on Vehicular Technology*.
- Zheng, F., Tsai, G., Zhang, Z., Liu, S., Chu, C.-C., & Hu, H. 2018. Trifo-VIO: Robust and efficient stereo visual inertial odometry using points and lines. Paper presented at the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

## ÖZGEÇMİŞ

Kişisel Bilgiler	
<b>Adı Soyadı:</b>	Nurgüneş ÖNAL
<b>Doğum tarihi:</b>	10 Şubat 1992
<b>Doğum Yeri:</b>	Erzurum
<b>Uyruğu:</b>	T.C.
<b>Adres:</b>	Atatürk Üniversitesi Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği Anabilim Dalı
<b>Tel:</b>	0538 707 74 07
<b>E-mail:</b>	nurgunes.onal@atauni.edu.tr
Eğitim	
<b>Lise:</b>	Erzurum Anadolu Lisesi
<b>Lisans:</b>	Atatürk Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü
Yabancı Dil Bilgisi	
<b>İngilizce:</b>	İyi
Üye Olunan Mesleki Kuruluşlar	
	Elektrik Mühendisleri Odası