

Hava Görüntülerinden 3 Boyutlu Haritalandırma ve Konumlandırma

Onur Eker

**YÜKSEK LİSANS TEZİ**

Elektrik Elektronik Mühendisliği Anabilim Dalı

Aralık 2020

3D Mapping and Localization on Aerial Images

Onur Eker

**MASTER OF SCIENCE THESIS**

Department of Electrical and Electronics Engineering

December 2020

Hava Görüntülerinden 3 Boyutlu Haritalandırma ve Konumlandırma

Onur Eker

Eskişehir Osmangazi Üniversitesi  
Fen Bilimleri Enstitüsü  
Lisansüstü Yönetmeliği Uyarınca  
Elektrik Elektronik Mühendisliği Anabilim Dalı  
Telekomünikasyon - Sinyal İşleme Bilim Dalı  
YÜKSEK LİSANS TEZİ  
Olarak Hazırlanmıştır

Danışman: Prof. Dr. Hakan Çevikalp

Bu tez TÜBİTAK tarafından 116E080 nolu proje kapsamında desteklenmiştir.

Aralık 2020

## ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Prof. Dr. Hakan Çevikalp danışmanlığında hazırlamış olduğum “**Hava Görüntülerinden 3 Boyutlu Haritalandırma ve Konumlandırma**” başlıklı tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 04/12/2020

Onur Eker

## ÖZET

Son yıllarda İnsansız Hava Araçları (İHA) görevlerini düşük maliyetlerle yerine getirebildikleri için gittikçe daha popüler hale gelmiştir. İnsansız Hava Araçları ile oluşturulmuş gerçek zamanlı bir ortomozaik, hem hızın hem de verimliliğin gerekli olduğu görevlerde yardımcı olabilir. Deprem, sel ve çığ gibi afet senaryolarında bir alana yapılan hızlı ve doğruluğu yüksek bir gözlem, ilk yardım ekipleri için önemlidir. Sayısal yüzey modelleri, ortomozaikler ve nesne tespit algoritmaları hızlı karar verme ve müdahale etmede önemli rol oynayabilir. Bir ortomozaik gözlemlenmek istenen bölgeye genel bir bakış sağlar. Operatörün ilgilenilen bölgeleri bulmasına ve nesne tespit algoritmaları da bu bölgelerde ilgilenilen nesnelere tespitine yardımcı olur. Bu çalışmada, İHA'nın kamera ve GPS verileri birleştirilerek gözlemlenen bir ortamı gerçek zamanlı olarak iki ve üç boyutlu haritalandıran monoküler SLAM tabanlı bir haritalama sistemi geliştirilmiştir. Geliştirilen sisteme oldukça hızlı çalışan derin öğrenme tabanlı bir nesne tespiti yöntemi eklenmiştir. Geliştirilen haritalama ve nesne tespit yönteminin performansı yeri gözlemleyen bir İHA'dan alınan görüntülerle hem tek İHA hem de çoklu İHA senaryosunda incelenmiştir.

**Anahtar Kelimeler:** ortomozaik, slam, nesne tespiti, insansız hava aracı.

## SUMMARY

Fast and accurate observation of an area in disaster scenarios such as earthquake, flood and avalanche is crucial for first aid teams. Digital surface models, orthomosaics and object detection algorithms can play an important role for rapid decision making and response in such scenarios. In recent years, Unmanned Aerial Vehicles (UAVs) have become increasingly popular because of their ability to perform tasks at lower costs. A real-time orthomosaic generated by using UAVs can be helpful for various tasks where both speed and efficiency are required. An orthomosaic provides an overview of the area to be observed, and helps the operator to find the regions of interest. Then, object detection algorithms help to identify the desired objects in those regions. In this study, a monocular SLAM based system, which combines the camera and GPS data of the UAV, has been developed for mapping the observed environment in real-time. A deep learning based state-of-the-art object detection method is adapted to the system in order to detect objects in real time and acquire their global positions. The performance of the developed method is evaluated in both single and multiple UAVs scenarios.

**Keywords:** orthomosaic, slam, object detection, unmanned aerial vehicle.

## TEŞEKKÜR

Öncelikle, görüntü işleme ve makine öğrenmesi üzerine sağlam temeller atmamı sağlayan ve bana ESOGÜ Bilgisayarlı Görü ve Makine Öğrenmesi (MLCV) araştırma laboratuvarında çalışma imkanı sağlayan danışmanım Prof. Dr. Hakan Çevikalp'e,

Bu çalışmayı 116E080 nolu 1001 projesi kapsamında destekleyen TÜBİTAK'a,

Yüksek lisans sürecinde bir arada çalıştığımız MLCV laboratuvarındaki dostlarıma,

Her zaman yanımda olan ve desteklerini benden hiçbir zaman esirgemeyen annem ve babama,

Bu yüksek lisans tezi sürecinde her türlü destek ve yardımlarını benden esirgemedikleri için minnettarım ve çok teşekkür ederim.

# İÇİNDEKİLER

|   | <u>Sayfa</u> |
|---|--------------|
| <b>ÖZET</b> . . . . .                           | <b>vi</b>    |
| <b>SUMMARY</b> . . . . .                        | <b>vii</b>   |
| <b>TEŞEKKÜR</b> . . . . .                       | <b>viii</b>  |
| <b>İÇİNDEKİLER</b> . . . . .                    | <b>ix</b>    |
| <b>ŞEKİLLER DİZİNİ</b> . . . . .                | <b>xi</b>    |
| <b>ÇİZELGELER DİZİNİ</b> . . . . .              | <b>xii</b>   |
| <b>1. GİRİŞ VE AMAÇ</b> . . . . .               | <b>1</b>     |
| <b>2. LİTERATÜR ARAŞTIRMASI</b> . . . . .       | <b>3</b>     |
| 2.1. Monoküler SLAM . . . . .                   | 3            |
| 2.1.1. Başlatma . . . . .                       | 3            |
| 2.1.2. Kamera konumlandırma . . . . .           | 5            |
| 2.1.3. Haritalama . . . . .                     | 6            |
| 2.1.4. Yeniden konumlandırma . . . . .          | 9            |
| 2.1.5. Döngü tespiti . . . . .                  | 9            |
| 2.2. Panorama ve Ortomozaik Üretimi . . . . .   | 9            |
| 2.3. Nesne Konumlandırma . . . . .              | 12           |
| <b>3. MATERYAL VE YÖNTEM</b> . . . . .          | <b>17</b>    |
| 3.1. Materyal . . . . .                         | 17           |
| 3.1.1. Orb öznitelikleri . . . . .              | 17           |
| 3.1.2. Yansıtmalı geometri . . . . .            | 19           |
| 3.1.3. Kamera matrisi ve kalibrasyonu . . . . . | 21           |
| 3.1.4. Epipolar geometri . . . . .              | 24           |
| 3.1.5. Ransac ve yığın dengeleme . . . . .      | 27           |
| 3.2. Yöntem . . . . .                           | 29           |
| 3.2.1. Küresel konum belirleme . . . . .        | 29           |
| 3.2.2. Yoğun nokta bulutu oluşturma . . . . .   | 33           |
| 3.2.3. Sayısal yüzey haritası . . . . .         | 34           |
| 3.2.4. Ortorektifikasyon . . . . .              | 35           |



## İÇİNDEKİLER (devam)

|  | <u>Sayfa</u> |
|--|--------------|
| 3.2.5. Nesne tespiti . . . . .           | 36           |
| 3.2.6. Ortomozaik . . . . .              | 38           |
| <b>4. BULGULAR VE TARTIŞMA . . . . .</b> | <b>39</b>    |
| 4.1. Nesne Tespiti . . . . .             | 39           |
| 4.2. Ortomozaik Üretimi . . . . .        | 40           |
| <b>5. SONUÇ VE ÖNERİLER . . . . .</b>    | <b>53</b>    |
| <b>KAYNAKLAR DİZİNİ . . . . .</b>        | <b>54</b>    |

## ŞEKİLLER DİZİNİ

| <u>Şekil</u>  | <u>Sayfa</u> |
|---|--------------|
| 2.1 Sol tarafta ORB-SLAM ile oluşturulan seyrek nokta bulutu, sağ tarafta ise LSD-SLAM ile oluşturulan yoğun nokta bulutu gösterilmektedir. . . . .               | 7            |
| 2.2 Döngü tespiti ve döngü kapatma. . . . .   | 10           |
| 2.3 Bölge önerici ağ ile nesne tespiti (Ren vd., 2015). . . . .   | 14           |
| 2.4 YOLO (Redmon vd., 2016). . . . .  | 15           |
| 3.1 FAST ile anahtar nokta çıkartma (Rosten ve Drummond, 2005). . . . .   | 17           |
| 3.2 İmge piramidi. . . . .  | 18           |
| 3.3 Dönüşümlerin hiyerarşi şeması. . . . .  | 21           |
| 3.4 Pinhole kamera modeli. . . . .  | 23           |
| 3.5 Lens gürültüleri. . . . .   | 23           |
| 3.6 Epipolar geometri. . . . .  | 24           |
| 3.7 3 boyutlu noktaların farklı kamera düzlemlerine yansımaları (Mouragnon vd., 2006). . . . .  | 27           |
| 3.8 Ransac ile eğri uydurma. Kırmızı noktalar aykırı değerleri, mavi noktalar ise aykırı olmayan değerleri göstermektedir. . . . .                                | 28           |
| 3.9 Haritalama ve konumlandırma yönteminin adımları. . . . .  | 30           |
| 3.10 İki nokta bulutunun dönüşüm parametrelerini hesapladıktan sonra birinci nokta bulutunun (a) ikinci nokta bulutuna (b) göre dönüşümü (c) verilmiştir. . . . . | 33           |
| 3.11 Örnek imge ve imgeye ait derinlik haritası. . . . .  | 34           |
| 3.12 İkili kd ağacıyla belirlenen yükseklik değerinin çok katmanlı ızgarada gösterimi. . . . .  | 35           |
| 4.1 Araba veri setine ait örnek imgeler. . . . .  | 40           |
| 4.2 Tek İHA senaryosunda oluşturulan ortomozaik görüntü. . . . .  | 42           |
| 4.3 Çoklu İHA senaryosunda oluşturulan ortomozaik görüntü. . . . .  | 43           |
| 4.4 Oluşturulan 3 boyutlu haritanın farklı açılardan görünümü. . . . .  | 44           |
| 4.5 Gerçek referans ile önerilen yöntemin konum eğrilerinin $x$ , $y$ ve $z$ düzleminde metre cinsinden ilk veri seti üzerindeki karşılaştırması. . . . .         | 46           |
| 4.6 Gerçek referans ile önerilen yöntemin ilk veri seti üzerinde üretilmiş yörüngeleri. . . . .   | 47           |
| 4.7 Tek İHA senaryosunda ikinci veri seti üzerinde oluşturulan ortomozaik görüntü. . . . .  | 48           |
| 4.8 İkinci veri setin oluşturulan üç boyutlu haritasının farklı bakış açılarından görünümü. . . . .   | 50           |
| 4.9 Gerçek referans ile önerilen yöntemin konum eğrilerinin $x$ , $y$ ve $z$ düzleminde metre cinsinden ikinci veri seti üzerindeki karşılaştırması. . . . .      | 51           |
| 4.10 Gerçek referans ile önerilen yöntemin ikinci veri seti üzerinde üretilmiş yörüngeleri. . . . .   | 52           |

## ÇİZELGELER DİZİNİ

| <u>Çizelge</u>  | <u>Sayfa</u> |
|---|--------------|
| 3.1 Darknet-53 mimarisi. . . . .  | 37           |
| 4.1 Önerilen yöntem, ORB-SLAM ve GPS'in ilk veri seti üzerindeki RPE sonuçları    | 45           |
| 4.2 Önerilen yöntem, ORB-SLAM ve GPS'in ilk veri seti üzerindeki APE sonuçları    | 45           |
| 4.3 Önerilen yöntem, ORB-SLAM ve GPS'in ikinci veri seti üzerindeki RPE sonuçları | 49           |
| 4.4 Önerilen yöntem, ORB-SLAM ve GPS'in ikinci veri seti üzerindeki APE sonuçları | 49           |

# 1. GİRİŞ VE AMAÇ

Günümüzde geliştirilen düşük güç tüketimine, yüksek işlem gücüne sahip tek kart bilgisayarlar ve düşük maliyetlerle üretilebilen İnsansız Hava Araçları (İHA) sayesinde havadan gözlem, haritalama, tespit ve takip görevleri üzerine yapılan çalışmalar oldukça hızlanmıştır. Düşük sensör maliyeti ve hafifliğinden dolayı kameralar İHA'larda bu tür görevlerin gerçekleştirilmesi için kullanılan en önemli sensör tipidir. Bu nedenle İHA'lar üzerinde uygulamaya yönelik bilgisayarlı görü çalışmaları araştırmacılar tarafından son zamanlarda çok sık çalışılmaktadır. İHA'dan elde edilen hava görüntüleri ile gözlemlenen alanı haritalamak, ortomozaik oluşturmak gözlemlenen alanın yapısını ve karakteristik özelliklerini elde etmemizi sağlar. Ortomozaik harita üretimi tarım, afet riski tespiti ve kentsel planlama gibi konularda gelecekte oldukça sık başvurulan bir yöntem olması konusunda umut vaat etmektedir. Bunun yanında gerçek zamanlı oluşturulan ortomozaik haritalar acil müdahale gerektiren durumlarda hız kazandırmakta ve operatörlere kolaylık sağlamaktadır.

Ortomozaik oluşturma işleminde Structure from Motion yöntemlerinden yararlanılabilir. Bu yöntemler temel olarak sırasıyla öznetelik çıkartma ve eşleştirme, görüntü hizalama ve yığın dengeleme (bundle adjustment) algoritmasıyla seyrek nokta bulutu üretimi, yoğun nokta bulutu ve mesh üretimi, ortomozaik adımlarından oluşmaktadır. Structure from Motion tabanlı yöntemler yüksek kalite ve doğrulukta haritalar oluştursa da haritalama işleminde kullanılacak tüm imgelerin önceden hazırlanmış olması gerekmekte ve haritalama işlemi oldukça uzun sürmektedir.

Haritalama işlemini gerçek zamanlı yapabilmek için SLAM (Simultaneous Localization and Mapping) yöntemleri kullanılmaktadır. Monoküler kamera tabanlı SLAM uygulamaları, sensör yapısı basit ve teknik olarak zor olması sebebiyle son zamanlarda robotik ve bilgisayarlı görü alanında çok sık çalışılan konulardan biri haline gelmiştir. SLAM, bilinmeyen ortamlarda navigasyon ve haritalama için kilit teknik olarak kabul edilmektedir. Bu tür sistemlere girdi sadece kameradan gelen görüntüler olduğu için bu yöntemler görsel SLAM (visual SLAM) olarak isimlendirilmektedir.

Bu tez kapsamında monoküler kamera kullanılarak uygulanan görsel SLAM, kameradan gelen görüntülerle kameranın konumunu tahmin ederken üçgenleme yaparak noktaların üç boyutlu konumunu belirlemektedir. Monoküler kameraların Stereo ve RGB-D kameralara göre bazı avantajları vardır. Düşük ağırlıklı ve düşük güç tüketimine sahip

olmalarının yanında ucuz olmaları monoküler kameraları diğer sensörlere göre tercih edilir kılmaktadır. Aynı zamanda, derinlik ölçen RGB-D kameralara karşın bu kameraların menzili teorik olarak sınırsızdır. Monoküler kameraların bu özelliği sayesinde monoküler SLAM sistemleri küçüklüğü veya büyüklüğü fark etmeksizin iç ve dış mekanlarda oldukça başarılı sonuçlar vermektedir. Ancak, haritalanan ortamın gerçek ölçeği bu tür kameralarla belirlenememektedir. Gerçek dünya ölçeğinde haritalama yapabilmek için hava görüntüleriyle haritalama işleminde İHA'nın GPS (Global Positioning System) verisinden yararlanılabilir.

Monoküler kamera tabanlı SLAM yöntemleri öznitelik tabanlı, doğrudan ve hibrit olmak üzere üçe ayrılmaktadır. Bu tez çalışmasında kullanılan öznitelik tabanlı monoküler SLAM yöntemiyle oluşturulan üç boyutlu nokta bulutu seyrek bir yapıya sahiptir. Seyrek nokta bulutu gözlemlenen bölgeyle ilgili yeteri kadar bilgiye ve anlamlı bir içeriğe sahip değildir. Ancak öznitelik tabanlı SLAM yöntemleri kamera konum tahminini yüksek doğrulukta yapabilmekte ve oldukça hızlı çalışmaktadırlar. Oluşturulan seyrek nokta bulutu yoğunlaştırılarak gözlemlenen bölgenin anlamlı bir haritası oluşturulabilir. Haritalama işleminde görüntü kalitesini ve imge birleştirmede doğruluğu artırmak için görsel bozuklukların giderilmesi gerekmektedir. Elde edilen yoğun derinlik haritası ve kamera konumu kullanılarak imgeler rektife edilebilir. Rektife edilen imgeler ve GPS verisi kullanılarak ortomozaik harita oluşturulabilir. Haritalama sırasında gözlemlenen bölgede ilgili nesnelere tespit edebilmek için son zamanlarda gittikçe popülerleşen, yüksek başarıma sahip ve gerçek zamanlı çalışan derin öğrenme tabanlı yöntemlerden yararlanılabilir. Bu çalışmada İHA'dan yere bakan bir kamerayla gözlemlenen bölgeyi 2 boyutlu ve 3 boyutlu haritalayan ve nesne tespiti yapan bir yöntem geliştirilmiştir. Gerçek dünya ölçeğinde ve gerçek zamanlı olarak haritalama işlemini yapabilmek için öznitelik tabanlı bir monoküler SLAM yöntemi ile İHA'nın GPS sensör verisi birleştirilmiştir. Öznitelik tabanlı monoküler SLAM yöntemi ile oluşturulan derinlik haritasını yoğunlaştırmak için içboyama tabanlı bir yoğunlaştırma yöntemi uygulanmıştır. Elde edilen yoğun derinlik haritası ve kamera konumları ile ortomozaik harita oluşturulmuştur.

Bu tez çalışması 4 ana bölümden oluşmaktadır. Bu bölümler sırasıyla; Literatür Araştırması, Materyal ve Yöntem, Bulgular ve Tartışma, Sonuçlar ve Öneriler bölümleridir. Haritalama ve konumlandırmanın genel tanımları, bu alanlarda yapılmış önemli çalışmalar Literatür Araştırması bölümünde incelenmiştir. Geliştirilen monoküler SLAM tabanlı haritalama ve konumlandırma yöntemi detaylı bir şekilde Materyal ve Yöntem bölümünde anlatılmıştır. Yöntemin tek ve çoklu İHA senaryosunda performansı hava görüntülerinden oluşan iki veri seti üzerinde test edilerek Bulgular ve Tartışma Bölümünde incelenmiştir. Yöntem ile elde edilen sonuçlar ve yöntemin genel değerlendirilmesi Sonuçlar ve Öneriler bölümünde verilmiştir.

## 2. LİTERATÜR ARAŞTIRMASI

Monoküler kamera ile üç boyutlu haritalama ve konumlandırma son yıllarda önem kazanan ve araştırmacıların yoğun olarak üzerinde çalışmakta olduğu bir problemdir. Bu başlık altında gerçek zamanlı haritalama ve konumlandırma için en önemli yöntemlerden biri olan monoküler SLAM, hava görüntülerinden üç boyutlu haritalandırma ve nesne konumlandırma yöntemleri incelenmiştir.

### 2.1 Monoküler SLAM

3 boyutlu haritalandırma ve konumlandırma probleminde kullanılan monoküler SLAM yöntemleri görüntüleri işleyiş biçimine göre öznitelik tabanlı (feature-based), doğrudan (direct) ve melez (hybrid) olmak üzere üç ana kategoriye ayrılmaktadır. Öznitelik tabanlı yöntemler, imgelerdeki öznitelikleri bularak bu öznitelikleri ardışık imgelerde eşleştirmeye çalışır. Doğrudan yöntemler ise imgelerden öznitelik çıkartmak yerine imgelerdeki piksellerin yoğunluk (intensity) bilgilerini kullanır. Melez yöntemler, öznitelik tabanlı ve doğrudan yöntemleri haritalama ve konumlandırma için birlikte kullanmaktadır. Monoküler SLAM yöntemleri üç temel aşamadan oluşmaktadır:

- Başlatma
- Kamera Konumlandırma
- Haritalama

#### 2.1.1 Başlatma

Monoküler SLAM algoritmaları başlangıç aşamasında, kameranın gözlemlenen sahneye göre konumu ve gözlemlenen ortamla ilgili hiçbir bilgiye sahip değildir. Bu nedenle, başlangıç aşamasında, referans bir koordinat sistemi tanımlanmalıdır. Referans bir koordinat sistemi tanımlamak için her monoküler SLAM yöntemi başlangıç aşamasında referans bir üç boyutlu harita (initial map) oluşturur. Daha sonrasında bilinmeyen bölgeler kamera tarafından tarandıkça başlangıç aşamasında referans düzlemde oluşturulan harita yeni üç boyutlu noktalarla güncellenir. Yüksek performanslı bir haritalama ve konumlandırma sistemi geliştirmek için başlangıç aşamasında oluşturulan referans koordinat sistemi doğru varsayımlar kullanılarak minimum hata ile tanımlanmalıdır.

İlk monoküler SLAM yöntemlerinden biri olan MonoSLAM (Davison vd., 2007) başlangıç işlemini düzlemsel bir yapıya bilinen uzaklığa koyulan kameradan, kullanıcının kamerayla düzlemsel yapı arasındaki mesafeyi girmesiyle tamamlar. Tek görüntüyle başlatma işlemini tamamlayan MonoSLAM gibi yöntemlerin yanı sıra başlangıç aşamasında iki görüntüden yararlanan SLAM yöntemleri de mevcuttur. Bu yöntemler gözlemlenen ortamın düzlemsel olduğu varsayımında bulunur. Başlatma için gerekli olan kamera konumunu Nister'in beş nokta algoritmasını (Nister, 2004) kullanarak homografi veya temel matris (essential matrix) ile hesaplar.

İki görüntüden yararlanarak başlangıç işlemini yapan SLAM yöntemlerinden biri olan PTAM (Parallel Tracking and Mapping) (Klein ve Murray, 2007), gözlemlenen ortamın düzlemsel olduğu varsayımında bulunarak homografi matrisi hesaplar. Homografi matrisini hesaplamak için başlangıç aşamasında kullanıcının sisteme iki anahtar kare (keyframe) vermesini ve bu iki anahtar kare arasında görüntülenen sahneye paralel bir şekilde yavaş ve düzgün geçişli bir öteleme hareketi yapmasını ister. SVO (Fast Semi-Direct Monocular Visual Odometry) (Forster vd., 2014) ise PTAM'e benzer şekilde düzlemsel varsayım ve homografi hesabı ile başlatma aşamasını tamamlar. SVO kullanıcıdan manuel anahtar kare alma işlemini ortadan kaldırmak ve başlangıç işlemini otomatik bir şekilde yapabilmek için ilk girdi kare ile o anki karenin takip edilen öznitelikleri arasındaki mesafenin medyanına bakar. Yöntem imgelerdeki FAST (Features from accelerated segment test) (Rosten ve Drummond, 2006) özniteliklerini bulur ve bunları KLT (Kanade-Lucas-Tomasi) (Lucas ve Kanade, 1981) öznitelik takipçisi ile ardışık karelerde takip eder. Eğer hesaplanan medyan değeri belirli bir eşik değerinin üstündeyse yeterli paralaksın sağlandığı varsayımı yapılarak homografi hesabı yapılır. ORB-SLAM (Mur-Artal vd., 2015) ise sezgisel bir başlatma yöntemi geliştirerek düzlemsellik ve küçük paralaks durumlarında temel matris hesaplamak yerine homografi matrisi hesaplayarak başlangıç aşamasını tamamlar.

LSD-SLAM (Large-Scale Direct Monocular SLAM) (Engel vd., 2014) ve DSO (Direct Sparse Odometry) (Engel vd., 2017) iki girdi imgeye ihtiyaç duymayan bir başlangıç yöntemi kullanmaktadır. Bu iki yöntem, başlangıç aşamasında düzlemsellik varsayımı yapmak yerine ortamın derinliğini rassal olarak belirler. Tek imge ile başlangıç işlemi yaparken ilgili piksellere (yüksek yoğunluk gradyanlarına sahip pikseller) yüksek varyans oluşturacak şekilde derinlik değerleri atanır. Bu yöntemle çok yüksek hataya sahip bir üç boyutlu başlangıç haritası elde edilir. Daha sonrasında yapılan kamera konum tahminleri de bu hatalı referans düzlemi nedeniyle yüksek hataya sahip olmaktadır. Ancak, aynı sahnenin birden fazla imge ile gözlemlenmesi sonucunda oluşturulan derinlik haritası kararlı bir çözüme yakınsamaktadır. Başlangıç aşaması, derinlik varyansı minimuma yakınsadığında tamamlanır. DDPTAM (Dense Piecewise Planar Tracking and Mapping)

(Concha ve Civera, 2015) yöntemi de LSD-SLAM'in başlangıç prosedürünü uygular. LSD-SLAM ve DSO'da olduğu gibi rassal derinlik haritası başlangıcında kararlı bir sonuca ulaşmak için sahneyi gözlemleyen birden fazla imgeye ihtiyaç duymaktadır.

### 2.1.2 Kamera konumlandırma

Kamera konumlandırma işleminde o anki güncel karenin kamera konumu, kameradan gelen görüntülerden oluşturulmuş referans harita bilgileri kullanılarak hesaplanır. Kameradan gelen görüntü ile referans harita arasındaki 2 boyutlu ve 3 boyutlu eşleştirmeler kullanılarak (imgeler arasında öznitelik eşleştirme veya öznitelik takibi yapılarak) kamera konumu elde edilebilir. Öznitelik takibi, imgedeki özniteliklerin bulunarak, ardışık kamera görüntülerinde lokal arama teknikleri kullanılarak takip edilmesiyle sağlanır. Öznitelik eşleştirme ise kameradan alınan her karede ayrı ayrı öznitelik çıkartılarak öznitelik tanımlayıcılarıyla (feature descriptors) belirlenen bir benzerlik metriğine göre imgeler arasındaki özniteliklerin eşleştirilmesiyle sağlanır.

PTAM özniteliklerin ölçek değişimine, gürültüye karşı gürbüz olmasını sağlamak ve kamera konum hesabını hızlı bir şekilde yapabilmek amacıyla kameradan alınan her imge için 4 seviyeli bir imge piramidi oluşturur. Piramidin her katmanında FAST öznitelikleri çıkartılır ve Shi-Tomasi skoru kullanılarak eşleştirme için kullanılacak öznitelikler belirlenir. PTAM, kamera konumunun altı parametre ile ifade edilmesini sağlayan SE(3) dönüşümünü kullanır. Kamera konumunun ifadesinin az parametre ile sağlanması, konum optimizasyon işlemlerini önemli ölçüde hızlandırmaktadır. PTAM'de konum tahmini, azalan sabit hız hareket modeli kullanılarak yapılır. Eğer kamera hareketinde yüksek hız varsa PTAM bunu tespit ederek hızlı kamera hareketlerinden kaynaklanan bulanıklığın konumlandırmaya olan etkisini azaltmak için piramidin sadece yüksek seviyelerinde işlem yapar.

ORB-SLAM ise 8 seviyeli bir imge piramidi oluşturur. Piramidin her katmanında PTAM'deki gibi FAST köşe öznitelikleri çıkartılır. İmgenin tümünde homojen bir dağılım sağlamak için her piramid katmanı ızgaralara bölünerek her ızgarada en az beş FAST köşe özniteliği çıkartılır. Daha sonrasında, çıkartılan her öznitelik için 256 bitlik ORB tanımlayıcıları hesaplanır. ORB-SLAM'de kamera konum hesaplama işlemi sabit hız hareket modeli kullanılarak yapılır. Sabit hız hareket modeli ani kamera hareketlerinde doğru sonuçlar üretememektedir. ORB-SLAM yaşanan bu problemi çözmek için imgelerde eşleştirilen öznitelik sayısını kontrol eder. Eğer eşleştirilen öznitelik sayısı belirli bir eşik değerinin altına düşerse, haritadaki üç boyutlu noktalar imge düzlemine yansıtılır, yansıtılan bölgelerde öznitelik araması yapılır. Bunun sonucunda eşleştirilen öznitelik sayısı artırılır ve ani kamera hareketleri sonucu yapılan yanlış hesaplamalar kısmen engellenir. ORB-SLAM büyük ölçekli alanlarda performansı artırmak için oluşturulan asıl üç boyutlu



nokta bulutunun (global harita) yanında ek olarak gözlemlenen sahneye ait yerel bir harita oluşturur. Konum grafi olarak oluşturulan yerel haritada düğümler anahtar kareleri ifade eder. O anki kare ile konum grafiında düğüm olarak ifade edilen karelerin birbirleriyle komşulukları ise kenar bilgisi olarak tutulur. Bu anahtar kareler ile yeni kare arasında sadece en yüksek eşleşmeye sahip, belirli bir eşik değerinin üstünde olan, öznitelikler tutulur. Böylece verimli, yüksek performanslı ve gürbüz bir kamera konum hesaplama işlemi sağlanmış olur. Bu özellikleri sayesinde ORB-SLAM kamera konum hesaplama işlemlerinde ve uzun zamanlı haritalama görevlerinde yüksek başarımlı sağlamaktadır.

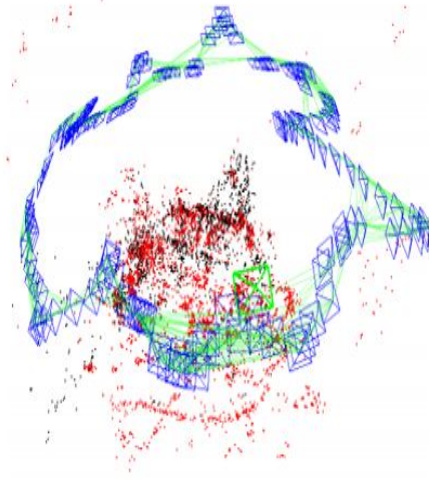
LSD-SLAM ise kamera konumunu hesaplamak için önceki aktif anahtar kareleri kullanılır. Aktif kareler, haritaya en yakın zamanda eklenen son kareleri ifade etmektedir. Kamera konumu, iteratif bir şekilde yeniden ağırlıklandırılarak fotometrik artık hatasını (photometric residual error) minimize eden Gauss-Newton optimizasyonu ile hesaplanır (Engel vd., 2013). Aykırı değerlerin etkilerini ortadan kaldırmak, gürültüyü azaltmak ve konumlandırmadaki doğruluğu artırmak için optimizasyon sırasında yüksek artıklara sahip ölçümlerin ağırlıkları azaltılır. SVO ise LSD-SLAM'e benzer bir şekilde Gauss-Newton yöntemini kullanarak fotometrik hatayı minimize etmeye çalışmaktadır. LSD-SLAM'den farklı olarak her yeni karenin konumunun bir önceki kare ile aynı olduğu varsayımını yapar. Daha sonra ardışık karelerde karşılıklı gelen piksel konumlarındaki derinlik ölçümleri üzerinden bir önceki kareye göre fotometrik hatayı azaltmaya çalışır.

### 2.1.3 Haritalama

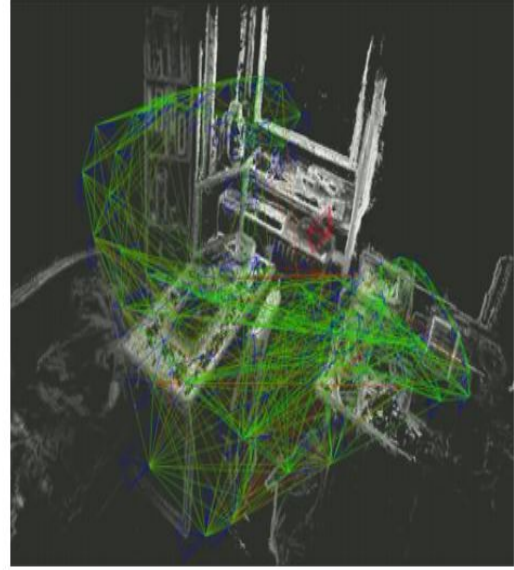
Monoküler SLAM yöntemlerinde haritalama işleminde temel amaç daha önce gözlemlenmemiş, bilinmeyen bir ortamın üç boyutlu nokta bulutlarıyla temsil edilmesini sağlamaktır. Temel olarak, nokta bulutlarıyla oluşturulan harita gözlemlenen bölgeyi yoğun (doğrudan yöntemler) veya seyrek (öznitelik tabanlı) bir nokta bulutu olarak ifade eder. Şekil 2.1, ORB-SLAM ile oluşturulan seyrek harita ve LSD-SLAM ile oluşturulan yoğun haritanın karşılaştırmasını göstermektedir.

PTAM'de seyrek nokta bulutu oluşturmak için, yeni bir anahtar kare elde edildiğinde anahtar kare ile eşleşmesi muhtemel olan haritadaki noktalar imge düzlemine geri yansıtılır ve öznitelikler eşleştirilir. Aynı noktaların haritaya tekrar eklenerek görsel bozulmalara sebep olmaması için eşleşen noktalar önceden gözlemlenmiş noktalar olarak işaretlenerek haritaya eklenmez. Yeni üç boyutlu harita noktaları ise girdi kare ile o kareye en yakın anahtar kare arasında üçgenleme yapılarak oluşturulur.

SVO harita oluşturma işleminde öncelikle ters derinlik parametreleme ile üç boyutlu noktaların derinlik değerlerini hesaplar. Yeni bir anahtar kare girdisi alındığında en yüksek



a) ORB-SLAM



b) LSD-SLAM

Şekil 2.1: Sol tarafta ORB-SLAM ile oluşturulan seyrek nokta bulutu, sağ tarafta ise LSD-SLAM ile oluşturulan yoğun nokta bulutu gösterilmektedir.

Shi-Tomasi skorlarına sahip özniteliklerle belirli sayıda derinlik filtreleri oluşturulur. Bu özniteliklere ait derinlik filtreleri, kamera merkezinden anahtar karedeki 2 boyutlu konumuna ilerleyen bir çizgi boyunca başlatılır. Daha sonrasında hesaplanması gereken üç boyutlu noktanın derinlik değeri, gözlemlenen sahnenin derinliğinin ortalamasına eşitlenir. Mevcut karede gözlemlenen ortalama derinlik dağılımı ile sınırlı olan epipolar arama düzlemi boyunca yeni karelerde eşleştirilecek öznitelikler aranır. Filtre yakınsadıkça derinlik belirsizliği azalır ve epipolar arama aralığı daralır. Eğer filtre ve arama aralığı sonsuza sapar veya başlangıç işleminin üzerinden uzun bir süre geçerse karelere ait üç boyutlu noktalar haritadan çıkartılır. Derinlik tahmini için kullanılan bu yöntem SVO'yu nispeten eşit derinlik dağılımlarına sahip ortamlarda çalışacak şekilde kısıtlar. REMODE (Regularized Monocular Depth Estimation) (Pizzoli vd., 2014) ise SVO yönteminin harita oluşturma modülünü temel almaktadır. Haritalama yaparken kamera konumlandırma işlemini SVO ile yapar. SVO'dan girdi olarak kamera konum hesaplamaları alır ve olasılıksal Bayes şeması (Vogiatzis ve Hernández, 2011) ile gözlemlenen sahnenin piksel başına yoğun üç boyutlu nokta haritasını oluşturur.

DT-SLAM (Deferred Triangulation for Robust SLAM) (Herrera vd., 2014) ise haritalama işlemi için kullanılan anahtar kareleri belirli ölçüde görsel değişiklik gözlemlendiğinde sisteme ekler. Anahtar kare eklemek için üç kriteri inceler; mevcut karenin yeterli sayıda yeni öznitelige sahip olması, belirlenen minimum eşik değerinin

üstündeki sayıda özniteliklerle üç boyutlu harita noktası oluşturulabilmesi ve önceden gözlemlenmiş üç boyutlu harita noktalarının gözlem açısının çok değişmiş olması. DT-SLAM ile üç boyutlu seyrek harita oluşturma işlemi, özniteliklerin üç boyutlu yer işaretlerine üçgenlenmesi, iki görünüm üçgenlemesi yöntemiyle yapılmaktadır.

LSD-SLAM harita oluşturma işleminde diğer monoküler SLAM yöntemlerine benzer şekilde yeni anahtar kareler seçerek bu anahtar karelerin minimum hatayla derinlik değerlerini hesaplayıp haritaya ekler. Mevcut karenin bir anahtar kare olup olmamasına bağlı olarak haritalama modülünün işlevleri iki ana kategoriye ayrılmaktadır. Eğer mevcut kare anahtar kare ise derinlik haritası oluşturma işlemi anahtar kare eklenmesi adımı ile gerçekleştirilir. Mevcut kare anahtar kare değilse derinlik haritası oluşturmak yerine derinlik haritası iyileştirilmesi yapılır. Sistem yeni bir anahtar kare elde ettiğinde, bir önceki anahtar kare ile oluşturulmuş derinlik haritası mevcut anahtar kare üzerine yansıtılır. Daha sonra, her derinlik değeri komşu derinlik değerlerinin ortalaması ile değiştirilerek mekansal düzenleme gerçekleştirilir ve varyans, komşu ölçümlerin minimum varyans değeri olarak seçilir. Son olarak, yeni eklenen anahtar karenin Sim(3) dönüşümü, haritadaki diğer anahtar karelere göre doğrudan bir görüntü hizalama yöntemiyle tahmin edilir ve hesaplanır.

ORB-SLAM harita oluşturmak için biri metrik diğer ikisi topolojik olmak üzere üç ayrı haritalama yöntemi kullanılmaktadır. Metrik haritada, konum dönüşümü hesabıyla yeni üç boyutlu noktaların yapısı hesaplanır. ORB-SLAM'de metrik harita, haritalanmamış bölgelerin kamera gözlemi sonucu oluşturulan 3 boyutlu yapısının sürekli olarak genişletildiği haritadır. Topolojik haritalar ise anahtar kareleri içeren öznitelik eşleştirilmesine göre oluşturulmuş kenarlarla bağlantılı yönsüz düğüm graflarıdır. İki topolojik haritadan birinde graflar anahtar kareler arasında olabildiğince çok kenar oluşturmaya çalışırken diğeri ise düğümlerin (anahtar karelerin) sadece en güçlü olan (en yüksek öznitelik eşleşmesine sahip) iki kenarını oluşturur. Üçgenlenmiş üç boyutlu noktalar haritaya dahil edilmeden önce, aykırı değerleri ayıklamak için, gözlemlendikleri tüm anahtar karelerde derinlik, geri yansıtma hatası ve ölçek tutarlılığı açısından kontrol edilir.

Monoküler SLAM uygulamalarında haritalandırma ve konumlandırma performansını gürbüzleştirmek amacıyla yukarıda anlatılan üç temel adıma ek olarak sıklıkla yeniden konumlandırma ve döngü tespiti olmak üzere iki ek yöntem daha uygulanmaktadır. Bu iki önemli teknik ile ilgili yapılan çalışmalar aşağıdaki başlıklarda incelenmiştir.

### 2.1.4 Yeniden konumlandırma

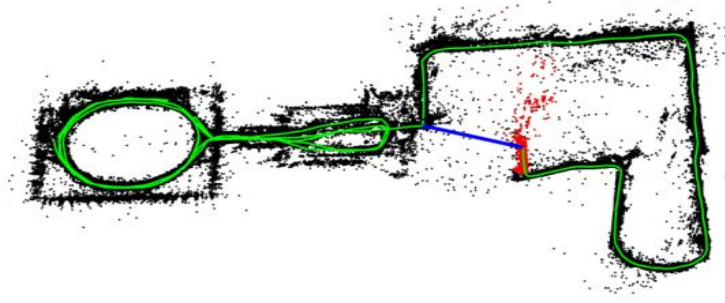
Hızlı kamera hareketleri, imgedeki gürültü gibi etkenler kamera konum takibinin kaybedilmesi, haritalamanın başarısız olması gibi durumlara yol açabilir. Kamera konum takibi kaybedildiği zaman, konum takibine ve haritalama işlemine devam edebilmek için kamera konumunun önceden oluşturulmuş harita üzerinden tekrar hesaplanması ve kameranın doğru bir şekilde yeniden konumlandırılması gerekmektedir. Bu işleme yeniden konumlandırma (relocalization) denir. Örneğin, ORB-SLAM yeniden konumlandırma işlemini, anahtar karelere ait öznitelikleri bag-of-words yöntemiyle tanıma veritabanında (recognition database) tutarak yapmaktadır (Gálvez-López ve Tardos, 2012). Yeniden konumlandırma işlemi sırasında veritabanı sorgusuyla mevcut kare ile eşleşen anahtar kareler bulunarak yeniden kamera konumu hesaplanır. Eğer SLAM yöntemlerinde yeniden konumlandırma sisteme dahil edilmemişse kamera konumu kaybedildiği zaman sistemin çalışması sonlanır.

### 2.1.5 Döngü tespiti

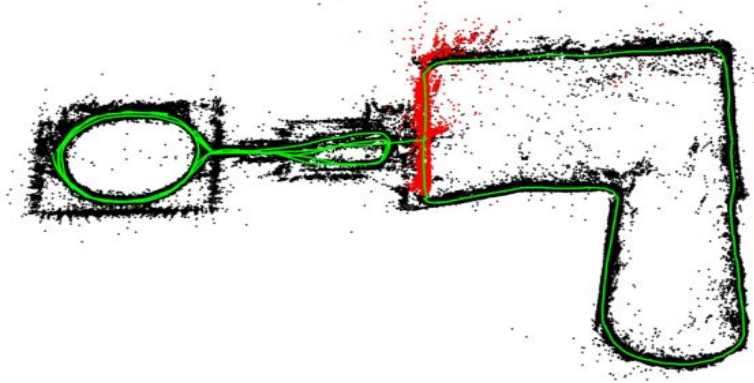
Döngü kapatma veya döngü tespiti, kameranın daha önce haritaladığı bir alana geri döndüğünü tespit edip haritadaki hataların minimize edilerek haritanın düzeltilmesi işlemidir. Eğer bir döngü bulduysa, kamera hareketi sırasında birikmiş hatalar hesaplanabilir ve düzeltilebilir. Döngü kapatma işlemiyle kaymalar sonucu oluşan konumlandırma hataları, özellikle büyük ölçekli alanlarda, önemli ölçüde azaltılabilir. Herrera vd., 2014'te BRIEF (Binary Robust Independent Elementary Features) (Calonder vd., 2010) tanımlayıcılarını ikili sözlük ağacında (binary vocabulary tree) önceki anahtar karelerde arar. Eğer bir döngü tespit edildiyse 2 boyutlu ve 3 boyutlu nokta eşleştirmeleriyle konum bulunur ve optimize edilir. ORB-SLAM ise ORB özniteliklerini yeniden konumlandırma işlemine benzer bir şekilde tanıma veritabanı aracılığıyla eşleştirip haritadaki tüm anahtar kareleri ve noktaları yığın dengeleme yöntemi ile optimize ederek döngü kapatma işlemini gerçekleştirir. Şekil 2.2, ORB-SLAM'in döngü kapatma işlemi öncesi ve döngü tespitinden sonra haritadaki hataların düzeltilerek iyileştirilen noktaların konumlarını göstermektedir.

## 2.2 Panorama ve Ortomozaik Üretimi

Hava görüntülerinden haritalama yapan klasik 2 boyutlu görüntü birleştirme yöntemleri art arda alınan karelerde öznitelik çıkartılması ve eşleştirilmesine dayanmaktadır. Bu yöntemler temel olarak iki imge düzlemi arasındaki hareketi tanımlayan homografi hesaplanmasına bağlıdır. Bu tür yöntemlerde hesaplamalar düzlemsel bir yüzeye sınırlandırıldığı için gözlemlenen bölgenin üç boyutlu yapısı elde edilemez.



a) Döngü kapatma öncesi



b) Döngü kapatma sonrası

Şekil 2.2: Döngü tespiti ve döngü kapatma.

Brown ve Lowe, 2007'de kamera hareketinin sadece optik eksenin etrafında olduğunu varsayarak sırasız (art arda toplanmamış) imgelerle panoramik görüntü üreten homografi tabanlı bir yöntem önermiştir. Yöntemdeki ana adımlar; öznitelik çıkartma, öznitelik uzayında eşleştirme, RANSAC (Fischler ve Bolles, 1981) uygulama, ardından homografi hesaplama ve yığın dengelemeden oluşmaktadır. Steedly vd., 2005'te videolardan alınan kamera görüntülerinin sırasız olmadığı, kamera görüntülerinin art arda toplanarak belirli bir kesişime sahip olduğu bilgisini kullanarak imgeleri daha verimli bir şekilde panoramik görüntü üzerinde birleştirmiştir. Agarwala vd., 2006'da homografi ve MRF (Markov Random Field) optimizasyonunu kullanarak gözlemlenen bir bölgenin çoklu bakış açılı panoramik görüntüsünü oluşturmuştur. Homografi tabanlı tüm bu yaklaşımların ortak noktası, kesintisiz ve görsel olarak anlamlı içeriğe sahip panoramalar veya kuş bakışı görüntüler elde etmeye çalışmaktır. Ancak bu yöntemler panoramik görüntü oluştururken coğrafi referanslama ve coğrafi referanslama hatalarını göz önüne almamaktadır. Aynı zamanda, ortomozaik üretim yöntemleri, girdi olarak alınan görüntülerden kamera konum tahmini yapmak için gürbüz ancak maliyetli öznitelik çıkartma ve eşleştirme algoritmaları kullanmaktadır. Bununla birlikte, sadece öznitelik eşleştirme işlemine dayanan görüntü birleştirme ve panoramik görüntü oluşturma yöntemleri, doğrudan İHA'lar üzerinde

uygulandığında hata birikimine ve yüksek hataya sahip haritalara yol açmaktadır. Bu problemler benzer bir şekilde, sadece panoramanın başlatıldığı ilk görüntü için coğrafi referans verildiğinde ve sonraki görüntüler bu referans görüntüye birleştirildiğinde de yaşanmaktadır. İHA uygulamalarında coğrafi referanslı ve yüksek doğruluğa sahip bir ortomozaik görüntü oluşturmak için gerekli olan kamera konum hesaplamalarına ek sensör ölçümleri dahil edilebilir. Örneğin, Yahyanejad ve Rinner, 2015'te ortomozaik problemini, coğrafi referanslama hatası ile ortomozaik üretimi arasında değiş tokuş bulan bir kayıp fonksiyonu tasarlayarak ele almıştır. Bunun için hava aracının IMU (Inertial Measurement Unit) ve GPS (Global Positioning System) ölçümlerini kullanmıştır. Ortomozaik üretiminde, kameranın içsel ve dışsal parametrelerini hesaplamak için GPS verilerine ek olarak manuel bir şekilde belirlenen yer kontrol noktalarından da yararlanılabilir.

Ortomozaik oluşturma işleminde 3 boyut bilgisini hesaplamak için sıklıkla Structure from Motion tabanlı yöntemler kullanılmaktadır. Ortomozaik oluşturmak için Structure from Motion kullanan birçok algoritma mevcuttur (Moulon vd., 2016; Wu, 2013; Furukawa vd., 2010; Furukawa ve Ponce, 2009). Bu yöntemler temel olarak aşağıda verilen adımlardan oluşmaktadır:

- Öznitelik çıkartma ve eşleştirme
- Görüntü hizalama
- Seyrek nokta bulutu üretimi
- Yoğun nokta bulutu üretimi
- Ortomozaik oluşturulma

Structure from Motion tabanlı yöntemlerde, ortomozaik görüntü elde etmek için haritalama işleminde kullanılacak tüm imgelerin önceden hazırlanmış olması gerekmektedir ve haritalama işlemi gerçek zamanlı yapılamamaktadır. Bu nedenle bu tür yöntemler, gerçek zamanlı ve artımlı (incremental) kullanım için uygun değildirler.

Structure from Motion algoritmalarından ayrı olarak 3 boyutlu haritalama ve konumlandırma için bu tez kapsamında da olduğu gibi SLAM yöntemleri kullanılmaktadır. Bu vd., 2016'da hava görüntülerinden 2 boyutlu ortomozaik haritalar oluşturmak için öznitelik tabanlı bir SLAM yöntemi ve hava aracının GPS verilerini kullanmıştır. Bu çalışmada 2 ve 3 boyutlu ortomozaik görüntü oluşturmak için gerekli olan kamera konumu hesaplama, seyrek nokta bulutu üretme işlemlerinde yüksek başarıma sahip, oldukça hızlı

çalışan ve gürbüz öznitelik tabanlı ORB-SLAM kullanılmıştır. Kern, 2018’de 2 ve 3 boyutlu ortomozaik görüntü üretmek için monoküler SLAM tabanlı bir yöntem kullanmıştır. Yöntem, bu tez kapsamında yapılan çalışmaya benzer şekilde monoküler SLAM yöntemi olarak ORB-SLAM’i kullanmıştır. Bu yöntemden farklı olarak tez kapsamında geliştirilen yöntemde, özgün bir veri seti üzerinde eğitilen derin öğrenme tabanlı bir nesne tespit yöntemi kullanılarak rektife edilen imgeler üzerinde nesne tespiti yapılmıştır. Bu tespitler üç boyutlu ve iki boyutlu haritalar üzerinde işaretlenerek tespit edilen nesnelerin gerçek dünya konumları hesaplanmıştır. Bunun yanında, yöntemde, mozaik işlemi sırasında örtüşme olan bölgelerdeki hücrelerin değerleri olasılıksal bir yaklaşım ile belirlenirken, tez kapsamında geliştirilen yöntemde ise hücrelerin değerleri minimum yükselti açısına göre belirlenmiştir. Ayrıca, yöntemde imge çiftleri üzerinden derinlik hesabı yapan bir yöntem kullanılırken, tez çalışmasında derinlik değerlerini hesaplamak için içboya yöntemi kullanılmıştır. Hinzmann vd., 2018’de hava görüntülerinden gerçek zamanlı haritalandırma görevini yapmak için KLT (Kanade-Lucas-Tomasi) öznitelik takipçisini ve hava aracının IMU ve GPS sensör verilerini birleştirmiştir. Elde edilen 3 boyutlu kamera konumu ile gözlemlenen alanın yoğun nokta bulutu ve sayısal yüzey modeli oluşturulmuştur. Elde edilen model ile ortomozaik görüntü, ızgara tabanlı ortomozaik yöntemi kullanılarak oluşturulmuştur. Bu yöntemde kamera konumlandırma için öznitelik takipçisinden yararlanılırken, tez kapsamında geliştirilen yöntemde kamera konumlandırma için monoküler SLAM kullanılmıştır. Tez çalışmasında, monoküler SLAM algoritmalarındaki konum ve harita iyileştirme, döngü tespiti gibi adımlardan faydalanılarak daha gürbüz bir konumlandırma ve haritalama performansı elde edilmesi hedeflenmiştir.

## 2.3 Nesne Konumlandırma

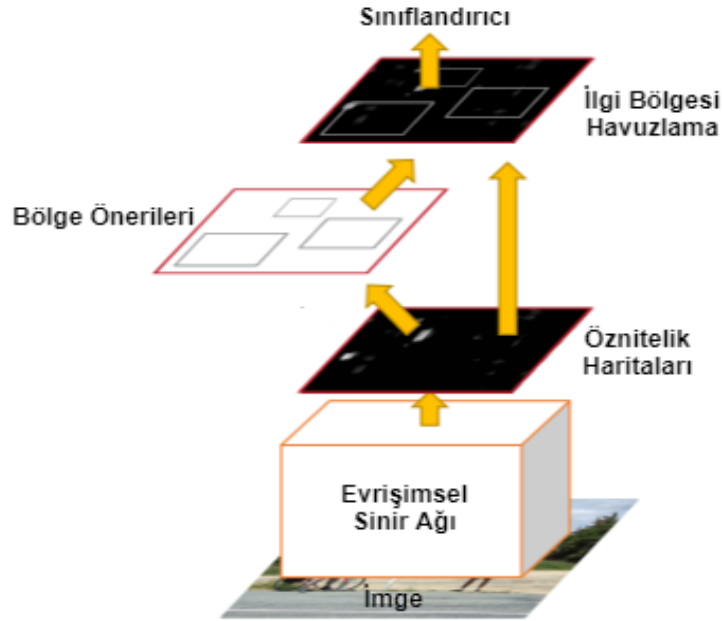
Nesne konumlandırma, bir nesne sınıfına ait örneğin imgede konumunun ve ölçeğinin bulunması problemidir. Nesne konumlandırma yöntemlerinde temel iki faktör vardır. Bunlardan biri tespit edilecek nesneyi betimleyen öznitelikler, diğeri ise nesnenin imgedeki konumunu belirleyen öğrenme algoritmasıdır. Derin öğrenmeden önce, HOG (Histogram of Oriented Gradients) (Dalal ve Triggs, 2005) ve öznitelik piramitleri gibi öznitelik çıkartma yöntemleri, nesne konumlandırma işlemi için yaygın olarak kullanılmıştır. Bu yöntemler genellikle genel nesne sınıflarına genişlemekte başarısız olmakta ve uygulamaların çoğu yüz veya yaya konumlandırmayla sınırlıydı. Derin öğrenme yöntemlerinin başarımı ve baştan başa eğitilebilme gibi özellikleri sayesinde, son zamanlarda derin öğrenme tabanlı yöntemler nesne konumlandırma probleminde sıklıkla kullanılmaktadır. Geliştirilen ekran kartları sayesinde derin öğrenme tabanlı yöntemler yüksek başarıma ulaşmanın yanında gerçek zamanlı çalışabilmektedir.

Derin öğrenme tabanlı yöntemlerin sınıflandırma probleminde gösterdiği başarılarından (Krizhevsky vd., 2012) sonra nesne konumlandırma problemine uygulayan ilk yöntemlerden biri olan R-CNN (Girshick vd., 2014), seçici arama (selective search) adı verilen bir teknik kullanarak girdi olarak verilen bir imgeden potansiyel ilgi bölgelerini çıkartır. Seçici arama, nesneyi gerçekten tanımaya çalışmak yerine, sezgisel bir yöntem kullanarak benzer pikselleri gruplar. Benzer pikseller genellikle aynı nesneye aittir. Bu nedenle, seçici aramanın sonuçlarının bir nesneyi içerme olasılığı çok yüksektir. Daha sonra, R-CNN, bu bölge önerilerini sabit boyutlu görüntülere dönüştürür, evrişimsel sinir ağlarını kullanarak bölgelere ait öznitelikleri çıkartır ve son olarak bu görüntüleri sınıflandırmak için DVM (Destek Vektör Makineleri) kullanır. Seçici aramayı kullanan eski yöntemlerin aksine, R-CNN, ikinci aşamasında tüm bölge önerilerinden öznitelik çıkartmak için HOG yerine evrişimsel sinir ağları kullanmıştır. Aday bölge önerisi benzerlik varsayımına bağlı olduğu için nesnenin tespit edilen konumu nesneyi yeterince iyi kapsayamamaktadır. Konumlandırmayı iyileştirmek için R-CNN derin öğrenme tabanlı nesne konumlandırma algoritmalarında sıklıkla kullanılan çevreleyici kutu regresyonu (bounding box regression) yöntemini nesneyi çevreleyen kutu koordinatlarına uygulamıştır. R-CNN seçici arama yöntemini kullandığı için baştan başa eğitilebilir bir ağ değildir. Aynı zamanda, önerilen aday bölgelerin tek tek ağdan geçirilip sınıflandırılması yöntemin oldukça yavaş çalışmasına sebep olmaktadır.

R-CNN’de her önerilen aday bölge için tekrarlanan evrişimi azaltmak yöntemin hızlanmasını sağlayabilir. Bu aday bölge önerilerinin tümü tek bir imgeden geldiğinden, evrişimsel sinir ağını imge üzerinde bir kez çalıştırmak ve çıkartılan öznitelik haritasını bölge önerileriyle paylaşmak R-CNN yöntemini çok daha hızlandırabilir. Ancak, farklı bölge önerileri birbirlerinden farklı boyutlara sahiptir. Bir imgeden öznitelik çıkartmak için aynı evrişimsel sinir ağı kullanıldığında, farklı boyutlardaki aday bölgeler farklı boyutlarda çıktı öznitelik haritalarına neden olur. Farklı boyutlara sahip bu öznitelik haritaları, sınıflandırma ve regresyon için aynı tamamen bağlı (fully connected) katmanların kullanılmasını engellemektedir. Çünkü tamamen bağlı katman sadece sabit boyutlu bir girdiyle çalışabilmektedir. Bu problemleri çözmek için Fast R-CNN (Girshick, 2015), ilgi bölgesi havuzlama (ROI Pooling) katmanı adı verilen bir katman önermiştir. Bu katmanın alt örnekleri, sabit boyutlu bir vektörde farklı boyutlarda bulunan öznitelik haritalarına sahiptir. Bu katman sayesinde, ilgi bölgesi ne kadar büyük veya küçük olursa olsun, sınıflandırma ve çevreleyici kutu regresyonu için aynı tamamen bağlı katmanlar kullanılabilir. Paylaşılan ortak bir öznitelik çıkartma ağı ve ölçekle değişmeyen ilgi bölgesi havuzlama katmanı ile Fast R-CNN, R-CNN’e benzer bir konumlandırma doğruluğuna ulaşabilmektedir. Bunun yanında yöntem, R-CNN’den yaklaşık 20 kat daha hızlı eğitim ve yaklaşık 200 kat daha hızlı değerlendirme hızına sahiptir.



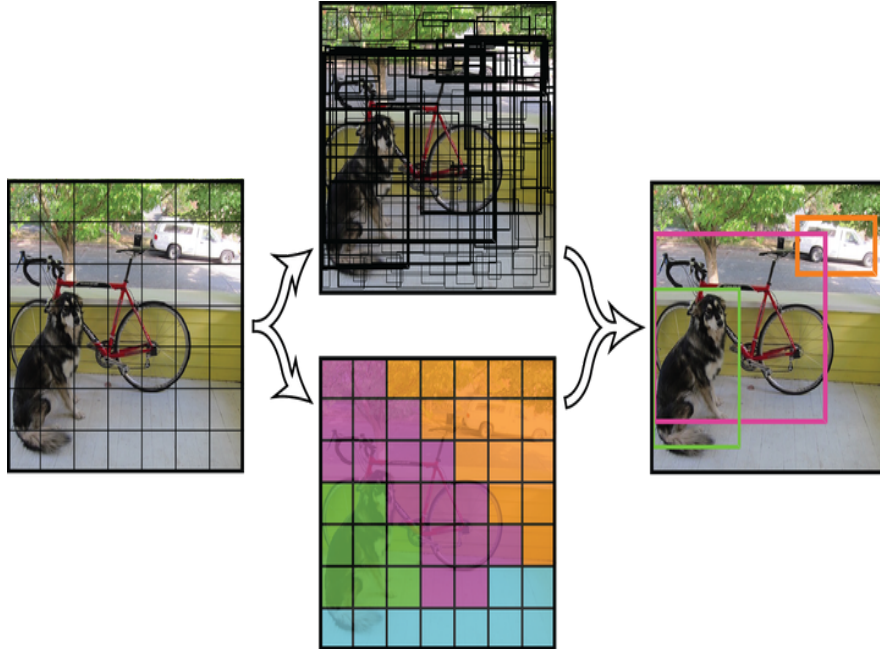
Faster R-CNN (Ren vd., 2015) ise nesne konumlandırma problemini iki aşamalı bir baştan başa eğitilebilir ağ yaklaşımı ile çözmeye çalışmıştır. Faster R-CNN’de önceki R-CNN yöntemlerinden farklı olarak seçici arama yapmak yerine bölge önerileri için bölge önerici ağ (Region Proposal Network) kullanılmıştır. Baştan başa eğitilebilir olan yöntemde bölge önerici ağ aday bölge üretme görevini yerine getirirken, bölge önerici ağ ile üretilen aday bölgeler üzerinde çevreleyici kutu regresyonu ve sınıflandırma işlemleri yapılmıştır. Faster R-CNN, bölge önerici ağ ile aday bölge önermek için bağlayıcı kutu (anchor box) adı verilen bir teknik önermiştir. Bağlayıcı kutular, farklı en-boy oranlarına ve ölçeklere sahip, aynı merkezi konumu paylaşan önceden tanımlanmış kutulardır. Faster R-CNN, her bir kayan pencere konumu için 3 ölçekte 3 farklı en-boy oranını kapsayan 9 adet bağlayıcı kutu kullanmaktadır. Farklı ölçeklerde tekrarlanan bu bağlayıcı kutular, aynı öznelik haritasının çıktılarını paylaşırken, ağa ölçek değişmezliği özelliğini kazandırır. Bu sayede bölge öneri ve tespit hızı artırılarak gerçek zamanlı çalışan gürbüz bir nesne konumlandırma yöntemi ortaya çıkmıştır. Faster R-CNN’de bölge önerici ağ kullanılarak yapılan iki aşamalı nesne tespiti Şekil 2.3’te gösterilmiştir.



Şekil 2.3: Bölge önerici ağ ile nesne tespiti (Ren vd., 2015).

R-CNN’den farklı olarak tek aşamalı önemli yöntemlerden biri olan YOLO (You Only Look Once) (Redmon vd., 2016), bölge öneri ve sınıflandırma işlemlerini aynı evrşimsel sinir ağında birlikte ele alır. Bölge önerilerine dayanan bir sınıflandırma problemi yerine nesne tespitini bir regresyon problemi olarak çözmeye çalışmaktadır.

Temel fikir, girdiyi bir  $S \times S$ 'lik ızgaraya bölmek ve her hücrenin çevreleyici kutu konumunu ve eğer nesnenin merkezi o hücreye düşerse güvenilirlik skorunu doğrudan hesaplamaktır. Şekil 2.4, ızgaralar üzerinden yapılan nesne tespiti işlemini göstermektedir. Tespit edilmek istenen nesneler farklı boyutlara sahip olabileceğinden, YOLO imgedeki her hücre için birden fazla sınırlayıcı kutu regresyonu uygulamaktadır. Eğitim sırasında, en yüksek kesişim değerine sahip regresör, gerçek referans etiketiyle karşılaştırılır. Böylece aynı konumdaki regresörler zaman içinde farklı ölçekleri kullanmayı öğrenirler. Her hücre, bir nesneyi içeren ızgara hücresinde (yüksek güvenilirlik skoruna sahip) sınıf olasılıklarını da tahmin etmektedir. Sonuç olarak YOLO, bir görüntüdeki tüm olası konumlar için sınıfları ve çevreleyici kutuları tek aşamada tahmin etmeye çalışır. YOLO'nun aksine Faster R-CNN, arka plan bölgelerini filtrelemek için bölge önerilerine güvenir, bu nedenle elde edilen nihai tahminler çok daha seyreklerdir. Daha sonraki YOLO versiyonları (Redmon ve Farhadi, 2017; Redmon ve Farhadi, 2018) başarımı artırmak için Faster R-CNN'de kullanılan bağlayıcı kutu yöntemini uygulamıştır. Bağlayıcı kutuların en-boy oranı ve ölçekleri eğitim veri setine ait örneklerin çevreleyici kutularına k-means kümeleme algoritması uygulanarak belirlenmiştir. Bunun yanında YOLOv3, girdi imgenin farklı çözünürlükleriyle ağı eğiterek ve farklı ölçeklerdeki öznitelik haritalarını tahmin için kullanarak (artık bağlantılar) küçük nesnelereki tespit başarımını yükseltmiştir.



Şekil 2.4: YOLO (Redmon vd., 2016).

SSD (Single Shot MultiBox Detector) (Liu vd., 2016) ise YOLO'da yaşanan sorunları çözmeye çalışan tek aşamalı bir nesne konumlandırma yöntemidir. Belirli bir öznitelik haritası verildiğinde, YOLO'da kullanılan sabit ızgaralar yerine SSD, Faster R-CNN yöntemine benzer şekilde farklı en boy oranlarına ve ölçeklere sahip bağlayıcı kutular kullanılmaktadır. Bağlayıcı kutuların kullanılması her hücre için algılanacak nesne miktarını artırmıştır. Aynı zamanda ağıın bu en boy oranı varsayımıyla üst üste binen küçük nesnelerin daha iyi ayırt edilmesi sağlanmıştır. SSD, özellikle küçük nesnelerin tespit başarımını yükseltmeyi hedefleyerek yüksek miktarda veri çoğaltma (data augmentation) yönteminden yararlanmıştır. Eğitim sırasında, imgelerdeki büyük nesnelerin büyük çevreleyici kutularını öğrenmek genellikle kolay olmaktadır. Bu nedenle kayıp fonksiyonunu domine eden bu kolay örneklerden kaçınmak için SSD, her bağlayıcı kutu için en yüksek kayıplı örnekleri seçmek amacıyla zor negatif madencilik (hard negative mining) tekniğini kullanmıştır.

Başka bir tek aşamalı nesne konumlandırma yöntemi RetinaNet (Lin vd., 2017) ortalama hassasiyeti (AP) iyileştirmek için odak kayıp fonksiyonu (focal loss) kullanılmaktadır. Odak kayıp fonksiyonu, zor negatif örneklerin kayıp fonksiyonuna daha fazla katkıda bulunmasını sağlayarak tespit edilmek istenen nesneye ait daha iyi ayırt edici öznitekler öğrenilmesini sağlamaktadır. Odak kayıp fonksiyonu, çapraz entropi kayıp fonksiyonuna ek bir güç parametresi (odaklanma parametresi) eklemiştir. Bu ek güç parametresi sayesinde güvenilirlik skoru yükseldikçe, kayıp değeri standart bir çapraz entropiden çok daha düşük olmaktadır. Böylelikle eğitim sırasında zor örnekler kayıp fonksiyonuna daha büyük kayıp değerleri sağlamaktadır.

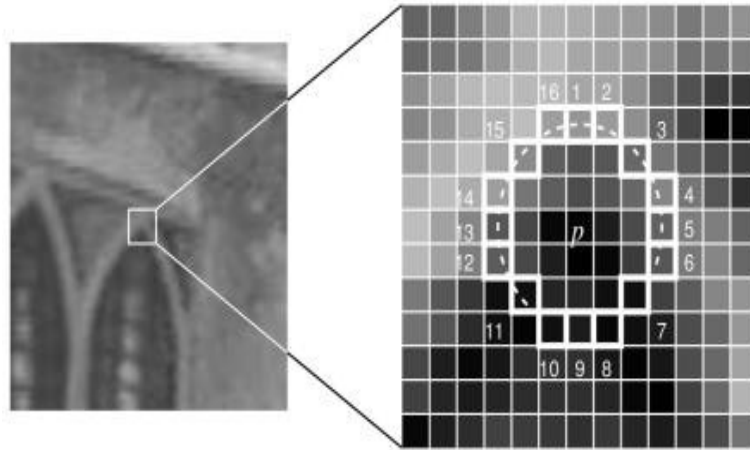
### 3. MATERYAL VE YÖNTEM

#### 3.1 Materyal

##### 3.1.1 Orb öznitelikleri

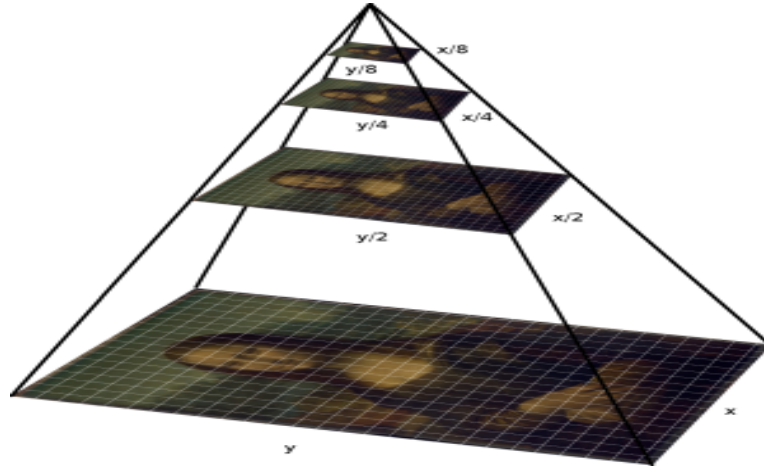
Bu başlıkta üç boyutlu haritalama ve konumlandırmada ORB-SLAM'in imgelerde öznitelik bulmak ve eşleştirmek için kullandığı ORB (Oriented FAST and Rotated BRIEF) (Rublee vd., 2011) anlatılmıştır. ORB, öznitelik çıkartmak için FAST özniteliklerini ve BRIEF tanımlayıcısını birleştirmiş ve performansı artırmak için bu iki yönteme yenilikler eklemiştir. ORB, temelde, öznitelikleri bulmak için FAST algoritmasını kullanır, ardından öznitelikler arasındaki en yüksek skora sahip  $N$  adet noktayı bulmak için Harris köşe tespitçisini (Harris, Stephens, vd., 1988) uygular. Aynı zamanda çok boyutlu öznitelikler üretmek için imge piramitlerini kullanır.

Yöntemde, öncelikle, imgede bulunan bir piksel  $p$ 'nin parlaklık (brightness) değeri, etrafındaki 4 piksellik yarıçapta bulunan 16 pikselin parlaklık değerleriyle karşılaştırılır (Şekil 3.1). Bu karşılaştırma işleminde çember üzerinde bulunan pikseller üç sınıfa ayrılır ( $p$ 'den daha açık,  $p$ 'den koyu veya  $p$ 'ye benzer). Eğer 8 pikselden daha fazla  $p$ 'den daha koyu veya parlak piksel varsa,  $p$  pikseli anahtar nokta olarak seçilir. FAST tarafından bulunan bu anahtar noktalar imgedeki köşe ve kenarların konumu hakkında bilgi verir.



Şekil 3.1: FAST ile anahtar nokta çıkartma (Rosten ve Drummond, 2005).

FAST algoritması ile bulunan özniteliklerin yön bileşeni bulunmamaktadır. Aynı zamanda, öznitelikler çok ölçekli bir yapıda değildir. Bu sebeplerden dolayı FAST öznitelikleri, ölçek değişmezliği (scale invariance) özelliğine sahip değildir. Bunu çözmek için ORB algoritmasında öncelikle imgenin farklı çözünürlüklerinin (imgeden daha yüksek ve daha düşük çözünürlüklü) bulunduğu çok ölçekli bir imge piramidi oluşturulur (Şekil 3.2). Piramitteki her seviye, imgenin önceki seviyeye göre ölçeklendirilmiş versiyonunu içermektedir. İmge piramidi oluşturulduktan sonra, piramitteki her imgede FAST ile anahtar noktalar tespit edilir. Her seviyede anahtar noktalar tespit edilerek, farklı ölçeklerde tespit edilen anahtar noktalar konumlandırılır. Bu sayede ORB, kısmi olarak ölçek değişmezliği özelliğine sahiptir.



Şekil 3.2: İmge piramidi.

Anahtar noktaların konumu belirlendikten sonra, her bir anahtar noktanın, o anahtar nokta etrafındaki yoğunluk seviyelerinin değişimine bağlı olarak yön bilgisi hesaplanır. Yoğunluk değişimini tespit etmek için yoğunluk merkezi (intensity centroid) kullanılır. Noktaların yönü hesapladıktan sonra, döndürme işlemi uygulanır ve ardından öznitelik tanımlayıcısı hesaplanarak, döndürme değişmezliği (rotation invariance) sağlanır. İmgede tespit edilen ve öznitelik tanımlayıcıları ile tutulan öznitelikler, sonraki imgelede hesaplanan tanımlayıcılarla karşılaştırmak ve anahtar noktaları eşleştirmek için kullanılırlar.

ORB, öznitelik tanımlayıcılarını hesaplamak için BRIEF kullanmaktadır. BRIEF, FAST algoritması tarafından bulunan tüm anahtar noktaları ikili öznitelik vektörüne dönüştürür. Öznitelik vektörleri, 1 ve 0'dan oluşmaktadır. Özetle, her bir anahtar nokta, 128 bitlik diziden oluşan bir öznitelik vektörüyle tanımlanır. İkili öznitelik vektörü, ikili öznitelik tanımlayıcısı olarak da isimlendirilmektedir.

BRIEF, öznitelik tanımlayıcısının yüksek frekanslı gürültüye karşı olan hassasiyetini önlemek için Gauss çekirdeği kullanarak imgeyi yumuşatır. Ardından, anahtar nokta etrafında tanımlanmış bir bölgede rastgele bir piksel çifti seçer. Rastgele seçilen piksel çiftindeki ilk piksel, sigma sapmasıyla anahtar noktanın etrafında ortalanmış bir Gauss dağılımından seçilir. Rastgele çiftteki ikinci piksel ise sigma sapmasıyla seçilen ilk piksel etrafında ortalanmış bir Gauss dağılımından çekilir. Eğer, ilk piksel ikinci pikselden daha parlaksa, BRIEF piksel için 1 değerini atar, değilse 0 değerini atar. Sonra anahtar noktaya ait 128 bitlik bir öznitelik tanımlayıcısı oluşturana kadar rastgele piksel çifti seçme ve değer atama işlemi tekrarlanır. Bir imgedeki her bir anahtar nokta için bu şekilde bir öznitelik vektörü oluşturulur. BRIEF, döndürme değişmezliği özelliğine sahip değildir. Bu nedenle ORB, rBRIEF (Rotation-aware BRIEF) yöntemini kullanarak anahtar noktalara döndürme değişmezliği özelliğini sağlar.

### 3.1.2 Yansıtımlı geometri

Bu başlıkta, bilgisayarlı görüde, 3 boyutlu düzlemdeki noktaların 2 boyutlu düzleme yansımalarını modellemek ve dönüşüm formülasyonlarını sağlamak için kullanılan yansıtımlı geometri anlatılmıştır. 3 boyuttaki tüm paralel doğruların yansımaları bir noktada buluşmayabilir. Örneğin, ufka doğru paralel olan iki doğru bir noktada buluşmayacaktır. Öklid geometrisi, sonsuzdaki noktaları ve doğruları ifade edememektedir. Bilgisayarlı görü uygulamalarında bir düzlemde bulunan noktaları (dünya koordinatları), aynı doğrudan geçen başka bir düzlemdeki noktalara (imge düzlemi) dönüşümünü sağlamak için yansıtımlı geometri kullanılır. Yansıtımlı geometride sonsuzdaki noktalar uzaydaki diğer noktalar ile aynı şekilde ele alınır. Bu ifadeleri matematiksel olarak sağlayabilmek için yansıtımlı geometride homojen koordinatlar kullanılmaktadır. Homojen koordinatlar sayesinde 2 boyutlu ve 3 boyutlu dönüşümlerde noktalar doğru, doğrular da nokta olarak ifade edilebilmektedir. Standart öklid uzayında, sütun vektör gösterimi kullanılarak, 2 boyutlu  $\mathbf{x}$  noktası aşağıdaki gibi temsil edilir.

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (3.1)$$

İki boyutlu  $\mathbf{x}$  noktasını homojen koordinatlar ile ifade etmek için, vektörün sonuna 1 değerine sahip ek bir boyut eklenir. Daha sonra, oluşturulan yeni vektör rastgele bir ölçeklendirme faktörü  $k_p$  ile ölçeklendirilir. 2 boyutlu gösterime geri dönmek için, ilk iki vektör girdisi üçüncü girdiye bölünür.

$$\mathbf{X} = k_p \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} k_p x \\ k_p y \\ k_p 1 \end{bmatrix} \quad (3.2)$$

3 boyutlu uzayda yansıtımalı geometri için 4 boyutlu homojen vektörlerin kullanılması gerekmektedir. Sonraki başlıklarda incelenecek olan 3 boyutlu uzaydaki yansıtma dönüşümleri  $4 \times 4$ 'lük matrisler olarak temsil edilir.

$$\mathbf{X} = k_p \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} k_p x \\ k_p y \\ k_p z \\ 1 \end{bmatrix} \quad (3.3)$$

Özetle, 2 boyutlu uzayda bulunan doğrular, 2 boyutlu uzaydaki noktalar gibi homojen bir vektör ile temsil edilebilir. 3 boyutlu uzayda bir nokta ise 4 boyutlu bir homojen vektör ile temsil edilir. Ayrıca 4 boyutlu bir homojen vektör ile düzlemleri de temsil edebiliriz. Homojen koordinatlar, sonsuzdaki noktaların sonlu koordinatlarla temsil edilmesine olanak sağlar ve dönüşüm formülasyonlarını basitleştirir. Bu dönüşümler, iki boyutlu düzlemde homografi adı verilen  $3 \times 3$ 'lük doğrusal dönüşüm matrisi ile sağlanır.


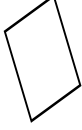
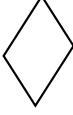


$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = k \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.4)$$

Burada  $k$  rastgele seçilen ölçeklendirme faktörünü belirtmektedir. Dönüşümü sekiz serbestlik derecesine sınırlandırmak için  $a_{33}$  değeri 1 ile değiştirilir. Bu değer, 0 dışında herhangi bir ölçeklendirme değerine sahip olabilir ve dönüşümü etkilemez. Şekil 3.3, ilk satırda en az serbestlik derecesi ile başlayan ve sonraki satırlarda serbestlik derecesi artan dönüşümlerin serbestlik derecelerini, koruduğu özellikleri ve matris ifadelerini göstermektedir. Bu ifadeler aynı zamanda dönüşümlerin hiyerarşi şeması olarak adlandırılmıştır (Hartley ve Zisserman, 2003). Örneğin, öteleme oryantasyonu korurken aynı zamanda şemada gösterilen diğer dönüşümlerin özelliklerine de sahiptir. Öteleme, homojen koordinatlarda aşağıdaki gibi ifade edilebilir.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.5)$$

Denklem 3.6 ise öteleme, döndürme ve deforme etme özellikleri olan altı serbestlik derecesine sahip afin dönüşüm için homojen koordinatlarla ifade edilen dönüşümü göstermektedir.

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.6)$$

| Dönüşüm    | Matris                  | Serbestlik Derecesi | Korunan Özellik | Şekil   |
|------------|-------------------------|---------------------|-----------------|---|
| Öteleme    | $[I   t]_{2 \times 3}$  | 2                   | Oryantasyon     |    |
| Öklit      | $[R   t]_{2 \times 3}$  | 3                   | Uzunluk         |    |
| Benzerlik  | $[sR   t]_{2 \times 3}$ | 4                   | Açı             |    |
| Afin       | $[A]_{2 \times 3}$      | 6                   | Paralellik      |   |
| Perspektif | $[H]_{2 \times 3}$      | 8                   | Düz çizgiler    |  |

Şekil 3.3: Dönüşümlerin hiyerarşi şeması.

### 3.1.3 Kamera matrisi ve kalibrasyonu

Kamera matrisi, gerçek dünyadaki 3 boyutlu noktaların imge düzlemindeki 2 boyutlu noktalara eşlenmesini tanımlar. Kamera matrisini ifade etmek için basitleştirilmiş ideal bir kamera modeli olan pinhole kamera modeli kullanılır. Pinhole kamera modeli, Şekil 3.4 üzerinden incelenebilir. Gerçek dünyada bulunan bir üç boyutlu  $\mathbf{X}$  noktasından yayılan, odak noktasından ve kameranın optik merkezinden geçen ışık imge düzlemi üzerindeki iki boyutlu  $\mathbf{x}$  noktasına yansıtılır. 3 boyutlu uzaydaki bir  $\mathbf{X}$  noktası ile bu noktaya imge düzleminde karşılık gelen 2 boyutlu  $\mathbf{x}$  noktası arasındaki ilişki aşağıdaki gibi ifade edilebilir.

$$(X, Y, Z) \rightarrow (x, y) \leftrightarrow (fX/Z, fY/Z) \quad (3.7)$$

Denklem 3.7 ile pinhole kamera modeli, bir nokta ve yansıtma düzlemi ile ifade edilir. Nokta ile düzlem arasındaki mesafe  $f$  odak uzaklığını temsil etmektedir. Aynı zamanda içsel



(kamera kalibrasyon matrisi) ve dışsal (öteleme ve döndürme matrisi) kamera matrisleri ile pinhole kamera modelinin yansıtma dönüşümü aşağıdaki gibi gösterilebilir.

$$\mathbf{x} = K[R|t]\mathbf{X} \quad (3.8)$$

İçsel kamera matrisi  $K$ , piksel birimleri olarak ifade edilen odak uzunlukları  $(f_x, f_y)$  ve genellikle imge merkezinde bulunan odak noktasından  $(c_x, c_y)$  oluşmaktadır. İçsel kamera matrisi, görüntülenen sahneye bağlı değildir. Bu nedenle, bir kere hesaplandıktan sonra sürekli olarak kullanılabilir. İçsel kamera matrisi  $K$  ile 3 boyutlu uzaydaki bir  $\mathbf{X}$  noktası ile bu noktaya imge düzleminde karşılık gelen  $\mathbf{x}$  noktası arasındaki ilişki aşağıdaki gibi düzenlenebilir.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (3.9)$$

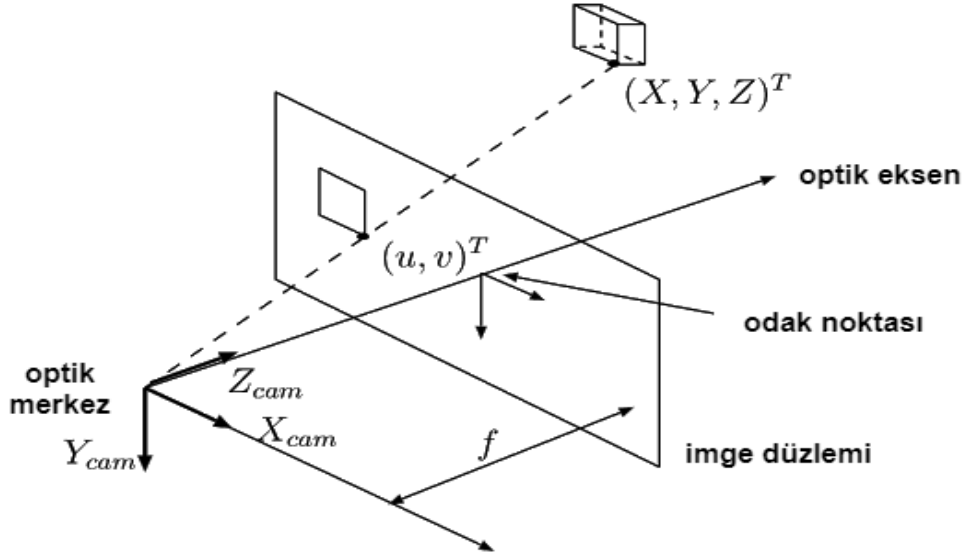
Dışsal kamera matrisi (döndürme ve öteleme matrisi)  $[R|t]$ , yansıtmalı ve homojen bir dönüşümün matris ifadesidir.  $3 \times 4$ 'lük yansıtma dönüşüm matrisi, dünya koordinatlarında temsil edilen 3 boyutlu noktaları imge düzlemindeki 2 boyutlu noktalara eşler ve 3 boyutlu noktaların kamera koordinatlarında temsil edilmesine olanak sağlar. Dışsal kamera matrisi, dışsal parametreler  $R$  ve  $t$  ile ifade edilir. Dışsal parametreler,  $3 \times 3$ 'lük döndürme matrisi olan  $R$  ve  $3 \times 1$ 'lik öteleme vektörü olan  $t$ 'den oluşmaktadır. Dışsal kamera matrisi homojen koordinatlarla aşağıdaki gibi temsil edilir.

$$\begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.10)$$

Son olarak, içsel ve dışsal matris denklemleri birleştirilerek (kamera matrisi), 3 boyutlu uzaydan 2 boyutlu imge düzlemine geçişi sağlayan  $\mathbf{x} = K[R|t]\mathbf{X}$  formu aşağıdaki gibi yazılabilir.

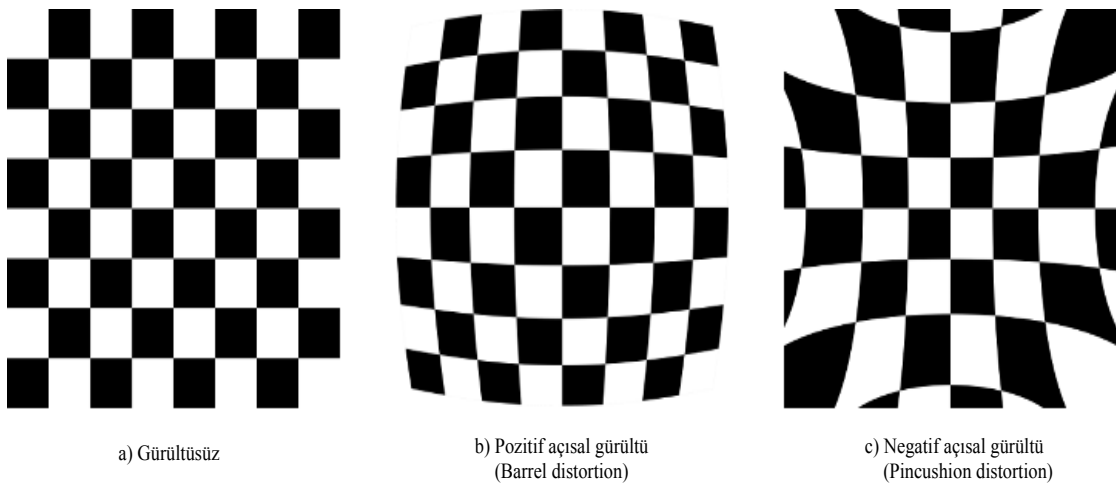
$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.11)$$

Kamera kalibrasyonu, içsel kamera matrisine ait parametreleri tahmin etme işlemidir. İçsel parametreler, kameranın odak uzaklığı, eğriliği, gürültüsü ve imge merkezi gibi özelliklerden oluşmaktadır. Kamera kalibrasyonu ile kameraya ait gürültü



Şekil 3.4: Pinhole kamera modeli.

parametreleri hesaplanarak imgeler gürültüsüz hale getirilir. Temel olarak, pozitif açısız ve negatif açısız olmak üzere iki tip gürültü vardır. Gürültü çeşitleri Şekil 3.5'te gösterilmektedir. Pozitif açısız gürültü, düz çizgilerin eğri görünmesine neden olur. Noktalar imge merkezinden uzaklaştıkça gürültü artar. Negatif açısız gürültüde ise noktalar imge merkezine yaklaştıkça gürültü artar. Kameranın gürültü katsayıları ve içsel kamera matrisi hesabını yaparak imgelerdeki lens gürültülerini ortadan kaldırmak için sıklıkla açık kaynak görüntü işleme kütüphanesi OpenCV (Bradski, 2000) kullanılır.



a) Gürültüsüz

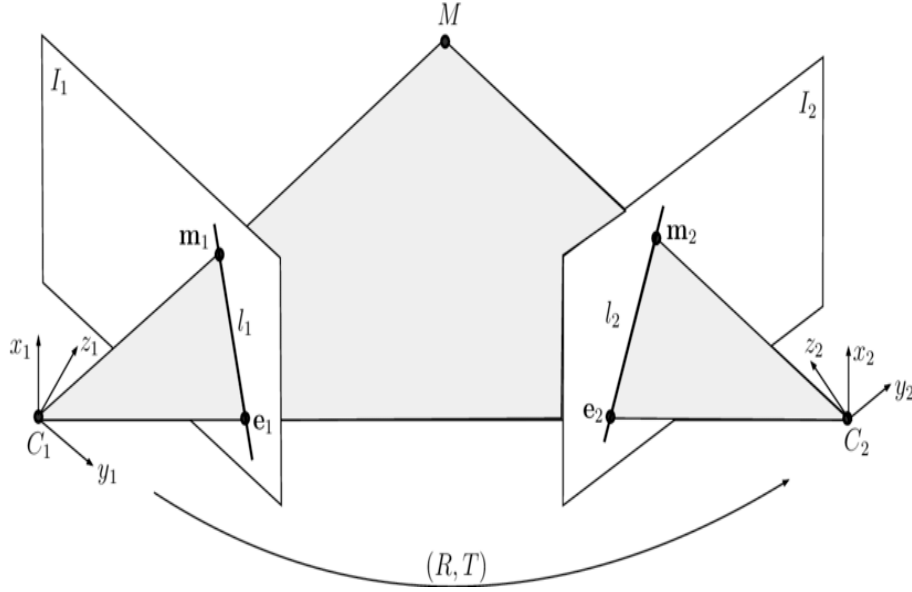
b) Pozitif açısız gürültü  
(Barrel distortion)

c) Negatif açısız gürültü  
(Pincushion distortion)

Şekil 3.5: Lens gürültüleri.

### 3.1.4 Epipolar geometri

Epipolar geometri, iki görünüm arasındaki yansıtma geometrisidir. Gözlemlenen sahnenin yapısından bağımsız olup yalnızca kameranın içsel ve dışsal parametrelerine bağlıdır. Epipolar geometri genellikle imgelerde karşılıklı gelen noktaları aramak ve bu noktaları temsil ettikleri 3 boyutlu noktalara eşlemek için kullanılır. Epipolar geometri, Şekil 3.6 üzerinden incelenebilir.



Şekil 3.6: Epipolar geometri.

Şekilde  $C_1$  ve  $C_2$  iki kameranın yansıtma merkezlerini,  $M$  gerçek dünyada bulunan 3 boyutlu noktayı,  $m_1$  ve  $m_2$  ise bu 3 boyutlu noktanın iki imge düzlemindeki yansımalarını göstermektedir. Kamera merkezlerinin ve 3 boyutlu noktanın kapsadığı düzleme epipolar düzlem, kamera merkezleri  $C_1$  ve  $C_2$ 'nin arasındaki doğruya temel çizgi adı verilir. İmge düzleminde bulunan ve epipolar düzlem ile kesişen  $l_1$  ve  $l_2$  doğruları epipolar doğru olarak adlandırılır. Gerçek dünyada bulunan  $M$  noktasının soldaki imge üzerindeki 2 boyutlu yansıması  $m_1$  noktası biliniyorsa, sağ imgedeki yansıması  $l_2$  doğrusu üzerinde olmalıdır. Bu kısıta epipolar kısıt adı verilir ve bu sayede nokta arama problemi tek boyuta indirgenir (Hartley ve Zisserman, 2003). Epipolar geometri ve epipolar kısıt ile iki görüş açısının göreceli konumları hesaplanabilir. Epipolar kısıtlamada imge düzleminde eşleşen noktalar arasındaki göreceli ilişkiyi ifade etmek için homografi matrisi veya temel matris kullanılır.

Kamera konum tahmini için kullanılan homografi, yansıtma düzleminde  $3 \times 3$ 'lük dönüşüm matrisi ile ifade edilir. Temel olarak homografi, kamera hareketini tanımlamak için

imgelerde karşılıklı gelen öznitelikleri kullanır. Denklem 3.12, gerçek dünyadaki 3 boyutlu  $\mathbf{X}$  noktasının iki farklı imgedeki yansıma noktalarını  $\mathbf{x}$  ve  $\mathbf{x}'$  olarak ifade etmektedir.

$$\mathbf{x} = H\mathbf{x}' \quad (3.12)$$

Aynı 3 boyutlu noktayı ifade eden  $\mathbf{x}$  ve  $\mathbf{x}'$  imge noktaları birbirleriyle bir homografi matrisi  $H$  ile ilişkilidir. Denklem 3.12, homojen koordinat sisteminde aşağıdaki gibi ifade edilebilir.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \quad (3.13)$$

Homografi matrisi hesaplanırken gözlemlenen bölgenin düzlemsel olduğu, tüm noktaların aynı düzlemde yer aldığı varsayımı yapılır. Homografi, bilinmeyen bir ortamda kamera hareketini dört belirsiz çözüme kadar yeniden oluşturabilir. Bunun için öncelikle  $h_{33}$ , 1'e eşitlenerek serbestlik derecesi 8'e düşürülür. Daha sonra  $x$  ve  $y$  noktaları için denklem aşağıdaki gibi ifade edilir.

$$x = \frac{h_{11}x' + h_{12}y' + h_{13}}{h_{31}x' + h_{32}y' + 1} \quad (3.14)$$

$$y = \frac{h_{21}x' + h_{22}y' + h_{23}}{h_{31}x' + h_{32}y' + 1}$$

Dörtten fazla karşılık gelen nokta sağlanırsa, en az hataya sahip homografi matrisini seçmek için en uygun optimizasyonun sağlanması gerekmektedir. Denklem 3.15, karşılıklı gelen dört nokta için çözülecek matrisin son halini göstermektedir. Elde edilen çözümlerden, kamera kalibrasyon matrisi ve kamera konum matrisi hesaplanır (Faugeras ve Lustman, 1988).

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1y'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2y'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2y'_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3x'_3 & -y_3y'_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3y'_3 & -y_3y'_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4x'_4 & -y_4y'_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4y'_4 & -y_4y'_4 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ x'_3 \\ y'_3 \\ x'_4 \\ y'_4 \end{bmatrix} \quad (3.15)$$

Epipolar geometride kamera konum tahmini için kullanılan başka bir matris, temel matristir. Temel matris, eğer kamera kalibrasyonu yapılmışsa, karşılıklı gelen imge

noktalarını ve epipolar kısıtları kullanarak kameralar arasındaki göreceli döndürme ve öteleme parametrelerini belirlemek için kullanılır. Homografi bir noktayı başka bir noktaya haritalarken, temel matris bir noktayı bir doğruya haritalar. Bir imgedeki noktanın temel matrisle çarpımı diğer imgedeki epipolar doğruyu verir.

$$E\mathbf{x} = l' \quad (3.16)$$

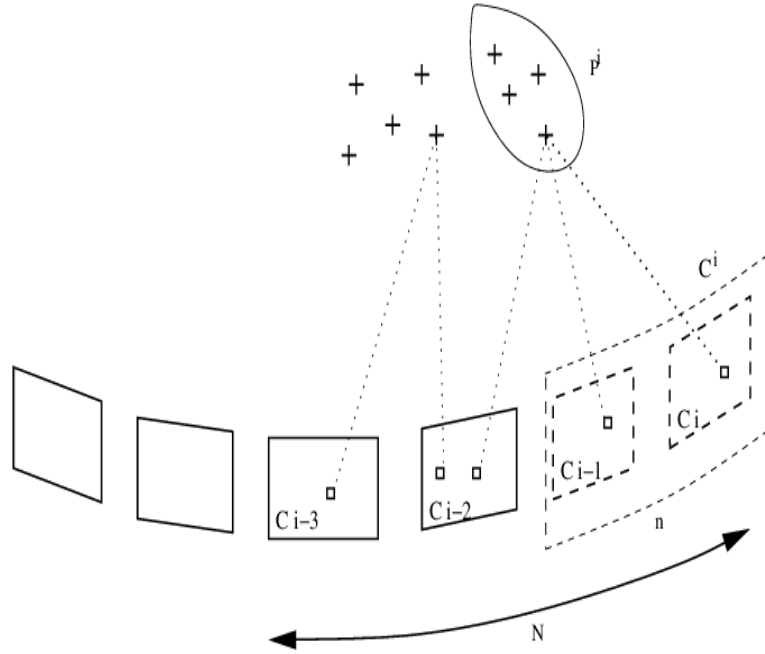
Burada  $E$  temel matrisi,  $\mathbf{x}$  gerçek dünyadaki 3 boyutlu noktanın imge düzlemindeki yansımasını,  $l'$  ise ikinci görüş açısındaki epipolar doğruyu ifade eder. Şekil 3.6 incelenerek  $\mathbf{x}^T l = 0$  olduğu gözlemlenebilir ve bu gözlem Denklem 3.16 ile birleştirilerek aşağıdaki denklem elde edilebilir.

$$\mathbf{x}^T E\mathbf{x} = 0 \quad (3.17)$$

Aynı sahneyi gözlemleyen iki imgede karşılıklı gelen imge noktaları verildiğinde, karşılıklı gelen tüm noktalar için epipolar kısıtlamayı karşılayan temel matrisi tahmin etmek Denklem 3.17'de gösterildiği gibi mümkündür. Bununla birlikte, tüm kısıtlamaları tam olarak karşılayan ideal bir temel matris, imge noktalarının maruz kaldığı gürültü nedeniyle hesaplanamamaktadır. Her kısıtlama ile ilgili hatanın ölçümüne bağlı olarak, belirli bir karşılık gelen imge noktaları kümesi için kısıtlamaları en uygun şekilde karşılayan temel matrisi belirlemek için sekiz nokta algoritması olarak bilinen toplam en küçük kareler problemi çözülür (Hartley ve Zisserman, 2003).

Temel matris ve homografi matrisi hesaplamasının yanında, imgelerden kamera konum ve hareket tahminini yapmak ve 3 boyutlu nokta oluşturmak için 2 boyutlu imge nokta karşılıklarından elde edilen 3 boyutlu noktaların üçgenlenmesi gerekmektedir. 3 boyutlu noktaları hesaplamak, kamera konum hesabını iyileştirmek için kullanılan yığın dengeleme işleminde de oldukça önemlidir (Şekil 3.7). Üçgenlenmiş 3 boyutlu noktalar, en az iki imgenin 2 boyutlu nokta eşleştirmelerinden geri yansıtılan ışınların kesişmesiyle belirlenir. İdeal koşullar altında eşleşen noktalardan geçen ışınlar tek bir 3 boyutlu noktada kesişmelidir. Ancak bu ışınlar, imgelerdeki gürültü, kamera modeli ve kalibrasyon hataları ve eşleşme belirsizliği nedeniyle tam olarak kesişmezler. Bu nedenle, tüm kesişen ışınlara en yakın mesafedeki nokta, 3 boyutlu noktanın yaklaşık konumu olarak kabul edilir. Üçgenlenmiş 3 boyutlu noktanın kesişen tüm ışınlara olan mesafesinin standart sapması 3 boyutlu noktanın doğruluğu hakkında bilgi vermektedir. Eğer standart sapma büyükse (noktanın konum belirsizliği büyükse), 3 boyutlu nokta, aykırı nokta olarak kabul edilir. Bu belirsizlik durumu özellikle kamera görüntüleri çok yakın aralıklarda çekildiğinde olur. Bunu engellemek için, 3 boyutlu noktaların ortalama belirsizliği belirli bir eşik değerinin altına düşene kadar kareler atlanır. Bu aralıklar dışında alınan karelere anahtar kare adı

verilir. Tez kapsamında geliştirilen yöntemde, haritalama sırasında gürültüyü ve işlem yükünü azaltmak için ORB-SLAM tarafından belirlenen anahtar kareler kullanılmıştır.

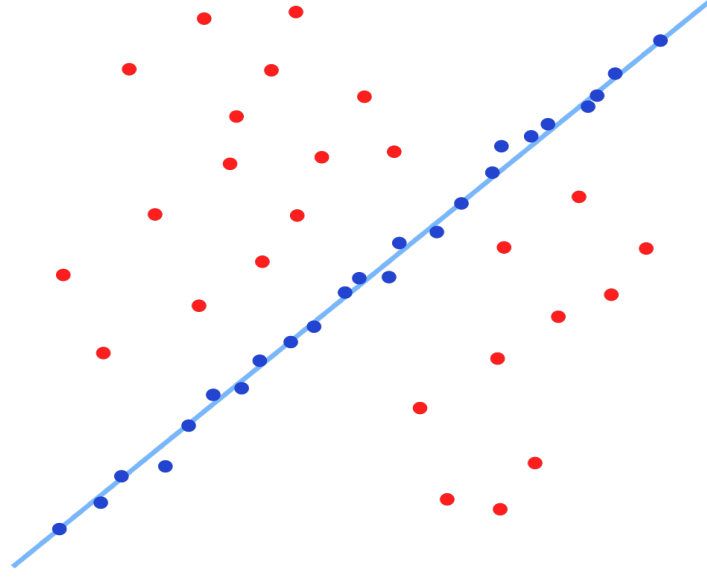


Şekil 3.7: 3 boyutlu noktaların farklı kamera düzlemlerine yansımaları (Mouragnon vd., 2006).

### 3.1.5 Ransac ve yığın dengeleme

Eşleşen noktalar arasında genellikle hatalı veri eşleştirmelerinden kaynaklanan aykırı değerler bulunmaktadır. Aykırı değerler, genellikle imge gürültüsü, bulanıklık ve öznitelik tespitçisinin veya tanımlayıcısının matematiksel modelinin hesaba katmadığı görüş açısı ve aydınlatma değişikliklerinden kaynaklanmaktadır. Kamera hareketinin ve kamera konumunun doğru tahmin edilebilmesi için aykırı değerlerin ortadan kaldırılması oldukça önemlidir. Random sample consensus (RANSAC) (Fischler ve Bolles, 1981), aykırı değerlerin mevcut olduğu durumlarda konum tahminini iyileştirmek için kullanılan standart bir yöntem olarak kabul edilmektedir. RANSAC algoritmasında temel fikir, rastgele örneklenmiş veri noktası kümelerinden model hipotezlerini hesaplamak ve diğer veri noktalarını kullanarak bu hipotezleri doğrulamaktır. Diğer verilerle en yüksek anlaşmayı (consensus) sağlayan hipotez, çözüm olarak seçilir. Şekil 3.8, en yüksek anlaşmayı sağlayarak uydurulmuş eğriyi, bu eğriye göre aykırı ve aykırı olmayan noktaları göstermektedir.

Kamera hareket tahmini için kullanılan model, önceki başlıklarda açıklanan ve  $[R|t]$  ile ifade edilen iki kamera konumu arasındaki göreceli hareket modelidir. Model tahmini



Şekil 3.8: Ransac ile eğri uydurma. Kırmızı noktalar aykırı değerleri, mavi noktalar ise aykırı olmayan değerleri göstermektedir.

için kullanılacak veri noktaları ise öznitelik eşleştirmeleridir. Bir hipotez için aykırı olmayan (inlier) değerler, noktadan epipolar doğruya olan uzaklık hesaplanarak bulunur (Fraundorfer ve Scaramuzza, 2012). Başlangıç aşamasında iki imge arasında eşleşen  $N$  adet öznitelikten bir  $A$  seti alınır. Bu setten  $s$  adet rastgele nokta seçilir. Seçilen bu rastgele noktalara doğrusal bir model uydurulur. Setteki bütün noktaların bu modele olan uzaklığı hesaplanır. Aykırı olmayan değerlerin (modele belirli bir uzaklığın altında bulunan) sayısı kaydedilir. Limit olarak belirlenen iterasyon sayısı tamamlanana kadar işlem devam eder. İterasyon sayısı  $N$ , rastgele belirlenen örnek kümelerden en az birinin aykırı değer içermemesini sağlayacak kadar yüksek seçilir.

$$N = \frac{\log(1 - p)}{\log(1 - (1 - \varepsilon)^s)} \quad (3.18)$$

Burada  $s$  model oluşturmak için kullanılan veri sayısını,  $\varepsilon$  veri noktaları içinde bulunan aykırı değerlerin yüzdesini,  $p$  ise çözümü başarılı kabul edebilmek için elde edilmesi gereken olasılık değerini ifade etmektedir. Minimum aykırı değere sahip model, çözüm olarak kabul edilir.

Haritalama ve konumlandırma işleminde imge sayısı, eşleşen öznitelikler, kamera parameterleri arttıkça, kümülatif hata giderek büyür ve bu da gürültülü haritalar ve konumlamalara sebep olur. Bu sorunu çözmek için sıklıkla yığın dengeleme (Triggs vd.,

1999) kullanılır. Yığın dengeleme, temelde, doğrusal olmayan bir optimizasyon algoritmasıdır. Genellikle dışsal kamera parametreleri,  $[R|t]$ , hesaplandıktan sonra kullanılır. Yığın dengeleme, yansıtma hatalarının karelerinin toplamını minimize ederek kamera konumunu ve harita noktalarını iyileştirir. Doğrusal olmayan yansıtma hatalarının minimizasyonu için Levenberg-Marquardt (More, 1978) algoritması kullanılarak öklit mesafesi en aza indirilmeye çalışılır (Denklem 3.19).

$$\min_{P_i, X_j} \sum_{ij} d(P_i X_j, x_{ij})^2 \quad (3.19)$$

## 3.2 Yöntem

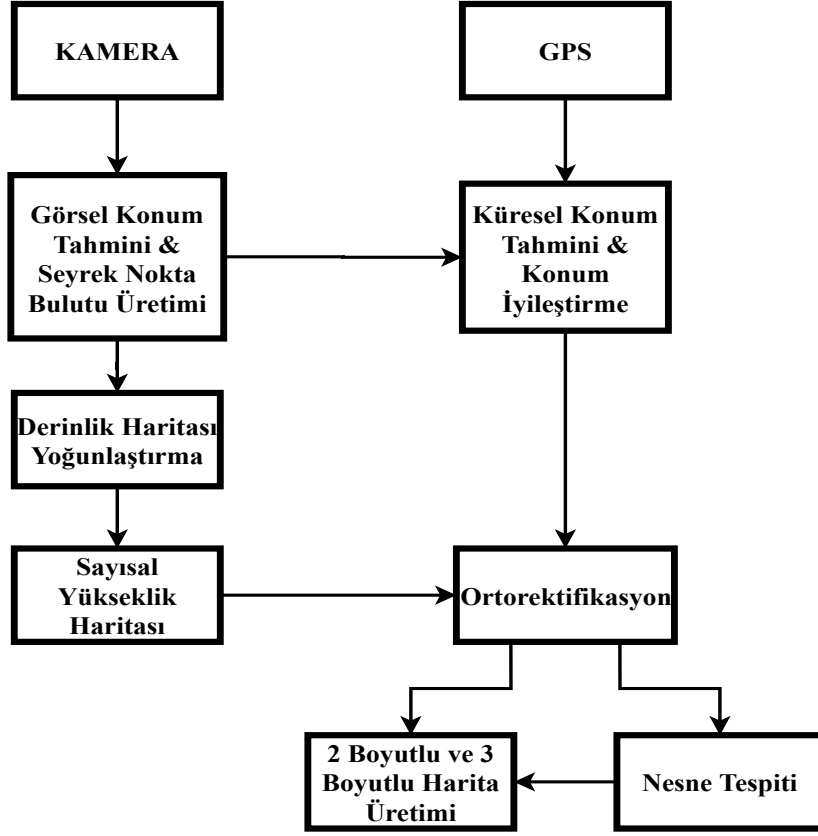
Tek ve çoklu insansız hava araçlarıyla yere bakan kameradan gözlemlenen bölgeyi gerçek dünya ölçeğinde 2 boyutlu ve 3 boyutlu haritalayan ve nesne tespiti yapan bir yöntem geliştirmek için birden fazla aşamaya ihtiyaç duyulmaktadır. Geliştirilen haritalama yöntemine ait adımlar Şekil 3.9’da gösterilmiştir. Haritalama ve konumlandırma sistemi için uygulanan adımlar sırasıyla şu başlıklarda incelenmiştir:

- Küresel Konum Belirleme
- Yoğun Nokta Bulutu Oluşturma
- Sayısal Yüzey Haritası
- Ortorektifikasyon
- Nesne Tespiti
- Ortomozaik

### 3.2.1 Küresel konum belirleme

Hava aracından yeri gözlemleyen bir kameradan alınan görüntülerle ortomozaik görüntü oluşturmak için sayısal yüzey modeline ve ortorektifikasyon işlemine ihtiyaç duyulmaktadır. Bu amaçla tez çalışması kapsamında geliştirilen yöntemde ilk olarak monoküler SLAM kullanılarak kamera konumu hesaplanmış ve seyrek nokta bulutu oluşturulmuştur. Monoküler SLAM yöntemi olarak seçilen ORB-SLAM, büyük ölçekli alanlarda ORB tabanlı gürbüz öznitelik takibi, konum grafi optimizasyonu ve döngü tespiti özellikleriyle yüksek doğrulukta kamera konum hesabı yapabilen oldukça hızlı çalışan bir yöntemdir. ORB-SLAM bu çalışma kapsamında kamera konumunu hesaplamak, seyrek nokta bulutunu üretmek ve bu hesaplamaları GPS verileriyle birleştirerek hesaplamaların





Şekil 3.9: Haritalama ve konumlandırma yönteminin adımları.

gerçek dünya ölçeğine dönüşümünü sağlamak için kullanılmıştır. Gerçek dünya ölçeğine dönüşüm sayesinde gözlemlenen bölgelerin daha gerçekçi ifade edilmesinin yanı sıra sisteme çoklu İHA ile haritalama yapabilme kabiliyeti kazandırmaktadır.

ORB-SLAM ile başlangıç aşamasında elde edilen ilk görüntüler üzerinde öznitelik çıkartma işlemi yapılır. Başlangıçta amaç, kameradan gelen imgeler arasındaki dönüşüm parametrelerini hesaplayarak ilk harita nokta kümesini üçgenlemek ve referans bir harita oluşturmaktır. Öznitelik çıkartma işleminde FAST köşeleri, 8 farklı ölçekte ve 1.2 ölçek faktörü ile oluşturulan imge piramidinden çıkartılır. Homojen bir dağılım sağlamak için her ölçek seviyesi ızgaralara bölünerek hücre başına en az 5 köşe çıkartılmaya çalışılır. Daha sonra, her bir hücredeki FAST köşeleri tespit edilir. Yeterli sayıda köşe bulunamazsa tespit eşiği uyarlanarak (eşik düşürülerek) daha fazla köşe tespiti yapılır. Eğer köşe içermeyen hücre varsa (yeterli öznitelik çıkartılacak görüntüler yoksa) hücre başına çıkartılmak istenen köşe miktarı azaltılır. Son olarak elde edilen FAST köşeleri için yön ve ORB tanımlayıcıları hesaplanır. ORB-SLAM, imgelerden elde edilen öznitelikleri kullanarak imgeler arasındaki dönüşüm parametrelerini hesaplamak için iki farklı yöntem kullanmaktadır. Bunların birincisi düzlemsel olmayan bölgelerde hesaplama yapabilmek için kullanılan temel matris,

ikincisi ise gözlemlenen bölgenin düzlemsel olduğu varsayımı yapılarak hesaplanan homografi matrisidir. Düzlemsel bölgeler için homografi matrisi ile hesaplama yapmak yerine temel matris hesabı da yapılabilir. Fakat temel matris düzlemsel bölgeler için hatalı sonuçlar üretecektir (Hartley ve Zisserman, 2003). Tercih edilecek yöntem simetrik transfer hatasına göre belirlenir.

$$R_H = \frac{S_H}{(S_H + S_F)} \quad (3.20)$$

Burada  $S_H$  homografi matrisinin skorunu,  $S_F$  ise temel matrisin skorunu belirtmektedir. Eğer  $R_H > 0.45$  ise homografi matrisi ile işleme devam edilir. Homografi seçilirse konum başlangıcı için elde edilen sonuçlara üçgenleme yapılarak üç boyutlu noktalar elde edilmeye çalışılır. Eğer düşük geri yansıtma hatasına sahip noktalar elde edilemediyse başlangıç yapılmaz. Başlangıç işlemi sonraki görüntüler üzerinden yapılmaya çalışılır. Temel matris üzerinden işlem yapılıyorsa sekiz nokta algoritması kullanılarak konum hesabı yapılır (Tan vd., 2013). Başlangıç aşaması tamamlandıktan sonra her yeni kare için ORB öznitelikleri çıkartılır. Çıkartılan öznitelikler önceki kareler ile eşleştirilir. Yeniden konumlandırma ve döngü kapatma için girdi karelere ait 2 boyutlu ve 3 boyutlu imge öznitelikleri kaydedilir. Bunun için bag-of-words kullanılarak imgelere ait öznitelikler bir öznitelik veritabanında tutulur (Gálvez-López ve Tardos, 2012). Fakat yüksek FPS ve uzun süreli haritalama işlemleri donanımına ait hafıza tüketimini yüksek oranda artıracığı için tüm karelere ait bilgileri tutmak yerine ORB-SLAM sadece anahtar kare olarak belirlediği karelerin özniteliklerini öznitelik veritabanında tutar. Ani hareket ve ışıklandırma değişiklikleri durumlarında özniteliklerin takibi başarısız olabilir. Bu durum yaşandığında ORB-SLAM haritalama ve konumlandırmaya devam edebilmek için bag-of-words yöntemiyle sakladığı önceki karelere ait öznitelikleri girdi kareye ait özniteliklerle eşleştirerek güncel kareye en yakın olan anahtar kareyi bulur. Daha sonra, PnP algoritması kullanılarak kamera konumu tekrar hesaplanır ve haritalandırma işlemine devam edilir.

Haritalama sırasında komşu anahtar kareler arasında yığın dengeleme yapılarak hesaplanan kamera konumları ve üç boyutlu noktalar iyileştirilmeye çalışılır. Gereksiz verilerden kurtulmak ve hesaplamaların doğruluğunu daha da artırmak için survival-of-the-fittest yöntemi kullanılır. Bir girdi karenin anahtar kare olarak belirlenebilmesi için aşağıdaki koşulları sağlaması gereklidir.

- Son anahtar kare ile güncel kare arasında 20 kare geçmiş olması
- Güncel karede en az 50 öznitelik tespit edilmesi
- Bir önceki anahtar kare ile güncel kare arasında eşleşen öznitelik sayısının belli bir eşik değerinden düşük olması (%90'dan daha az)

ORB-SLAM kamera konumu hesaplama işleminde anahtar kare seçimi, yığın dengeleme gibi teknikleri kullanarak oldukça başarılı sonuçlar elde etse de monoküler SLAM yöntemlerinde gerçek dünya ölçeği hesaplanamamaktadır. Dolayısıyla yöntem tarafından üretilen kamera konumu yerel bir koordinat sisteminde bulunmaktadır. Monoküler SLAM sistemiyle elde edilen kamera konumunun küresel koordinat sistemine dönüşümü hava aracının GPS verisi kullanılarak sağlanabilir. Bu dönüşüm, standart bir iki adet üç boyutlu nokta kümesi arasındaki dönüşüm parametreleri hesabı yapılarak sağlanabilir.

Yerel bir koordinat sisteminde hesaplanan kamera konumunun öncelikle gerçek dünya ölçeği hesaplanmalıdır. Gerçek dünya ölçeğini hesaplamak için başlangıç aşamasında ilk yirmi kare ölçek tahmini yapmak için kullanılır. Bu kareler ORB-SLAM ile üç boyutlu nokta bulutları oluşturulmuş, kamera konumu hesaplanmış ve GPS verisine sahip anahtar karelerdir. Bütün karelerin birbirleriyle arasındaki görsel ve küresel mesafe hesaplanır. Denklem 3.21’de gösterilen şekilde hesaplanan kareler arasındaki mesafelerin ölçeklerinin ortalaması alınarak ortalama ölçek hesaplanır.

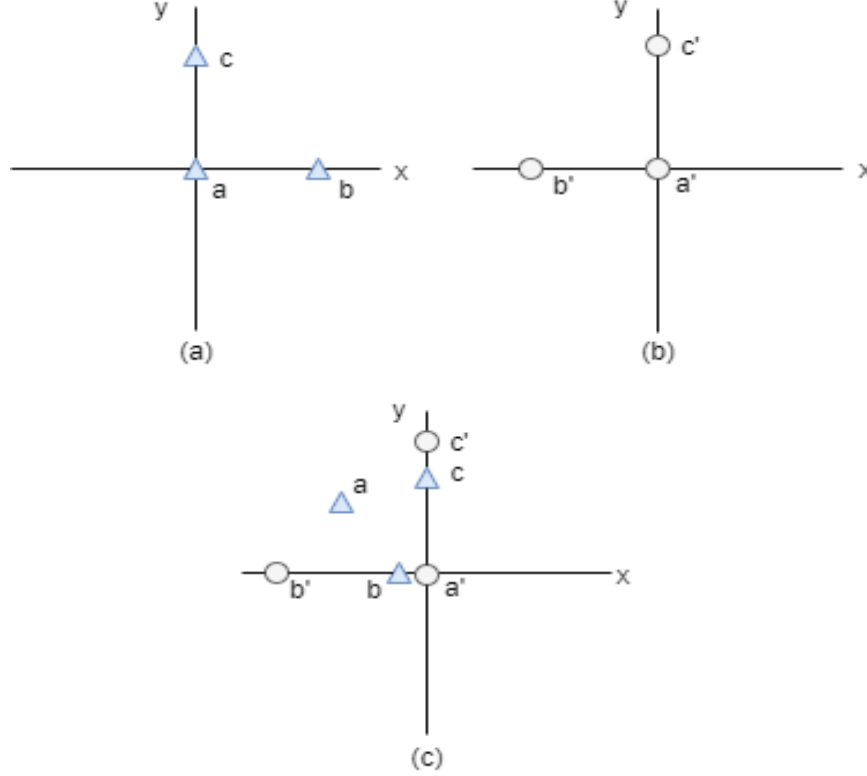
$$m_{gps}/m_{slam} \quad (3.21)$$

Burada  $m_{gps}$  iki karenin GPS konumları arasındaki mesafeyi,  $m_{slam}$  ise iki karenin ORB-SLAM ile hesaplanan görsel kamera konumları arasındaki mesafeyi ifade eder. Ölçek hesaplandıktan sonra, bu ölçek bilgisi kullanılarak sonraki kareler için kamera konum dönüşümü hesaplanır. Kamera konum dönüşümünü yapabilmek için ORB-SLAM ile elde edilen görsel konumlar ve GPS sensör verisi ile elde edilen küresel konumların iki üç boyutlu nokta modelleri arasında bir benzerlik dönüşümüne ihtiyaç vardır. İki nokta modeli arasındaki dönüşüm parametrelerini hesaplamak için en küçük kareler kestirimi (Least-squares estimation of transformation parameters between two point patterns) yöntemi kullanılmıştır (Umeyama, 1991). Yöntem girdi olarak verilen iki üç boyutlu nokta kümesi arasındaki dönme  $R$ , öteleme  $t$  ve ölçek  $c$ ’yi hesaplar. Benzerlik dönüşümü için elde edilmek istenen  $R$ ,  $t$  ve  $c$  değerleri minimum ortalama kare hatasını minimize eden değerlerdir. Denklem 3.22, dönüşüm parametrelerini hesaplamak için kullanılan formülü göstermektedir.

$$e^2(R, t, c) = \frac{1}{n} \sum_{i=1}^n \|y_i - (cRx_i + t)\|^2 \quad (3.22)$$

En küçük kareler kestirimi Şekil 3.10’da bir örnek ile gösterilmiştir. Şekilde verilen iki nokta bulutunun  $((a, b, c)$  ve  $(a', b', c')$ ) en küçük kareler kestirimi yöntemi kullanılarak hesaplanan benzerlik dönüşümü parametreleriyle (optimum dönme, öteleme ve ölçekleme)

yeni nokta kümesi elde edilir. Elde edilen yeni nokta kümesi, hesaplanan ölçek faktörüyle ölçeklendirilerek gerçek dünya ölçeğine göre kamera konumu hesaplanır.



Şekil 3.10: İki nokta bulutunun dönüşüm parametrelerini hesapladıktan sonra birinci nokta bulutunun (a) ikinci nokta bulutuna (b) göre dönüşümü (c) verilmiştir.

### 3.2.2 Yoğun nokta bulutu oluşturma

Bir önceki başlıkta anlatılan adımlarla kamera konumu küresel olarak belirlenmiştir. Kamera konumu ve ORB-SLAM ile hesaplanan seyrek nokta bulutu yoğunlaştırılarak 3 boyutlu yüzey bilgisi daha anlamlı hale getirilebilir. Küresel konumu belirlenmiş ve seyrek nokta bulutu oluşturulmuş imgelerin nokta bulutunu yoğunlaştırmak için oldukça hızlı çalışan seyrek nokta bulutu interpolasyonu kullanılmıştır. Seyrek nokta bulutu interpolasyonunu yapabilmek için derinlik haritasına ihtiyaç duyulmaktadır. Derinlik haritası oluşturmak için öncelikle ORB-SLAM ile hesaplanan 3 boyutlu noktaların derinlik değerleri imge düzleminde ait oldukları piksel konumlarına yazılır. Daha sonra imge düzlemine yansıtılmış bütün örneklem noktaları arasındaki boşlukları doldurmak için içboyama uygulanır. Kullanılan algoritma Navier-Stokes based Inpainting (Bertalmio vd., 2001), akışkanlar dinamiğine dayanır. Yoğun derinlik haritası oluşturmak için kullanılan içboyama algoritmasında temel ilke sezgiseldir. Yöntem, bilinen bölgelerden bilinmeyen

bölgelere olan kenarlar boyunca ilerler (çünkü kenarların sürekli olması amaçlanmıştır). Boyasız bölgenin sınırındaki gradyan vektörleri eşleştirilirken izofotları (aynı yoğunluğa sahip noktaları birleştiren çizgiler) devam ettirir. Boyasız piksel bu bölgedeki minimum varyansı azaltacak şekilde doldurulur. Şekil 3.11’de oluşturulan yoğun derinlik haritası gösterilmiştir.



Şekil 3.11: Örnek imge ve imgeye ait derinlik haritası.

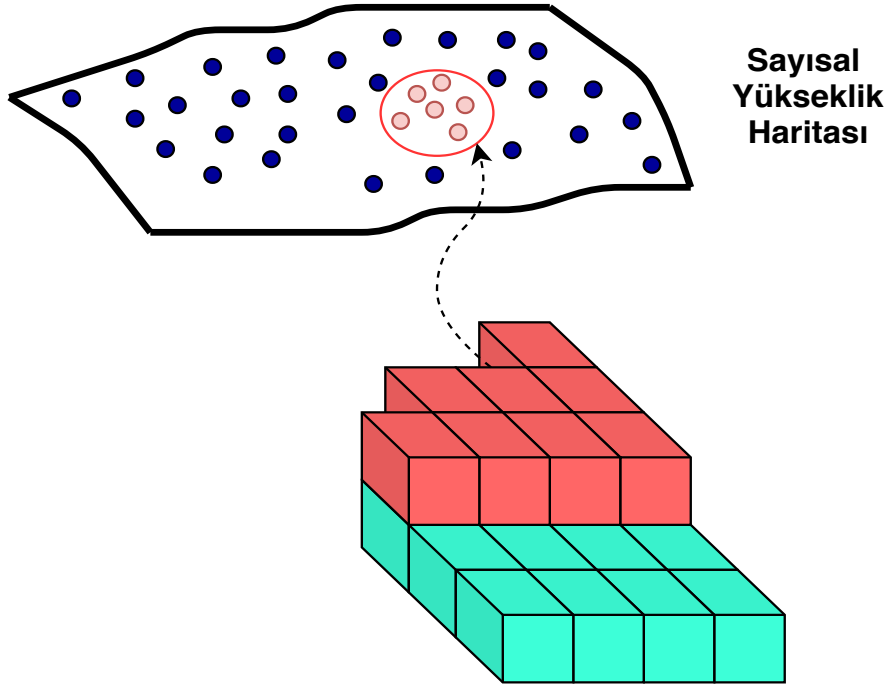
Yöntem gözlemlenen bölgedeki yapıları detaylı bir şekilde oluşturamamaktadır. Ancak yöntem ile oluşturulan derinlik haritası düşük gürültüye, homojen görünüme sahiptir ve yöntem oldukça hızlı çalışmaktadır. Yöntemin yüksek doğrulukta yoğun derinlik haritaları üretebilmesi için, ORB-SLAM tarafından hesaplanan seyrek nokta bulutu yüksek doğrulukta olmalıdır.

### 3.2.3 Sayısal yüzey haritası

İçboyama yöntemiyle oluşturulan yoğun nokta bulutu ile sayısal yüzey haritası oluşturmak için ızgara tabanlı bir yöntem kullanılmıştır (Hinzmann vd., 2018). Renk, yükseklik, nokta bulutu verilerini çoklu veri katmanlarıyla yönetmek için GridMap kütüphanesi (Fankhauser ve Hutter, 2016) kullanılmıştır. Kütüphane küresel koordinatlarla ifade edilen imge verilerinin hızlı bir şekilde manipülasyonunda ve küresel ortomozaik işleminde oldukça kolaylık sağlamaktadır.

Gözlemlenen yüzeyi yükseklik verileri içeren bir 3 boyutlu ızgara haritası ile temsil edebilmek için yoğun nokta bulutunun  $x$  ve  $y$  koordinatları, 2 boyutlu ikili bir kd ağacıyla yapılandırılmıştır. Tanımlanan interpolasyon yarıçapında, noktaların 20 piksel

yarıçapındaki, en yakın komşu noktalar hesaplanır. Daha sonra, interpolasyon yöntemi olarak ters mesafe ağırlıklandırma (inverse distance weighting) uygulanır. Ters mesafe ağırlıklandırma en yakın komşuların doğrusal ağırlıklı bir kombinasyonunu kullanarak hücrenin yüksekliğini belirler. Böylece hücre merkezine daha yakın olan noktalara daha yüksek ağırlık verilerek interpolasyon ile yükselti haritasında düzgün geçişli ve gürültüsüz yükselti elde edilir. Şekil 3.12’de kd ağacıyla belirlenen yükseklik değerinin çok katmanlı ızgara haritasına eklenmesi ile ilgili örnek gösterilmiştir.



Şekil 3.12: İkili kd ağacıyla belirlenen yükseklik değerinin çok katmanlı ızgarada gösterimi.

### 3.2.4 Ortorektifikasyon

Sayısal yüzey haritası oluşturulduktan sonra ortomozaik görüntüyü oluşturmak için gerekli olan bütün veriler elde edilmiştir. Ancak, ortomozaik görüntü oluşturma işleminden önce ilk aşamada hesaplanan kamera konum bilgisi ve bir önceki aşamada hesaplanan sayısal yüzey modeli kullanılarak imgelerde görüş açısı ve yüzey yapısı sebebiyle oluşan görsel bozulma rektifikasyon işlemi ile giderilmelidir. Ortorektifikasyon işlemi için artımlı ızgara tabanlı ortomozaik yöntemi kullanılmıştır (Hinzmann vd., 2018). İmgeye ait kamera konumu ve içsel kamera matrisi ile, her hücrenin görünür kamera konisi içinde olup olmadığı kontrol edilir. Kontrol işleminde ilk önce ilgili imgenin yükseklik katmanını içeren ızgara haritası, istenen yer örnekleme mesafesine göre yeniden boyutlandırılır. Yükseklik katmanının yeniden boyutlandırma işlemi bilineer interpolasyon ile yapılmıştır.

Daha sonra ızgaradaki her hücre için bir 3 boyutlu nokta oluşturulur. Oluşturulan noktalar imge düzlemine geri yansıtılarak gözlemlenen noktaların yeni konumları belirlenir. Belirlenen konumlar ızgara haritasında yeni bir imge katmanı oluşturularak katmanın hücrelerine yazılır. Yükseklik haritasındaki gürültülü veya hatalı konum tahmini nedeniyle, geri yansıtılan noktalar görüntü sınırları dışında tanımlanabilir. Bu gürültülü noktalar geçersiz olarak tanımlanır ve imge düzlemine geri yansıtılmaz.

### 3.2.5 Nesne tespiti

Rektifikasyon işlemi tamamlandıktan sonra imgenin çok katmanlı ızgara haritasında tutulan imge verisi üzerinde nesne tespiti yapılabilir. Bunun için kullanılan derin öğrenme tabanlı nesne tespit yöntemi YOLOv3, sisteme giriş olarak verilen ham görüntü piksellerini kullanarak nesneyi çevreleyen kutunun koordinatlarını ve sınıf olasılıklarını tahmin eder. YOLOv3 ağı Darknet-53 yapısını kullanmaktadır. Önceki YOLO versiyonlarından farklı olarak bu katmanlar arasındaki artık bağlantılar ve üç farklı ölçekte yapılan tespit sayesinde imgelerde bulunan küçük objelerin tespitinde başarımları artırılmıştır. Çizelge 3.1, Darknet-53 ağının mimarisini göstermektedir.

YOLOv3, imgeyi  $S \times S$ 'lik ızgaralara böler. Her ızgara hücresi sabit sayıda sınırlayıcı kutu tahmin eder ve her çevreleyici kutu ( $tx, ty, tw, th, \text{güvenilirlik skoru}$ ) beş parametreden oluşur. YOLOv3, nesne konumlarını doğrudan hesaplamak yerine referans kutularına olan ofseti ( $tx, ty, tw, th$ ) ve sınıf olasılığını temsil eden güvenilirlik skorunu tahmin eder. Bu strateji YOLOv3 için daha gürbüz bir eğitim prosedürü sağlar.

YOLOv3 için ağın çıkış tensörü boyutu  $N \times N \times (3 \times (4 + 1 + C))$  olmaktadır. Burada  $N \times N$ , ızgara hücrelerinin sayısını, 3 ise üç farklı ölçekte yapılan tespit sayısını,  $4 + 1$  ise çevreleyici kutu ofsetleri ve nesnelik (objectness) skorunu ve  $C$  ise sınıf sayısını göstermektedir. YOLOv3'ün kayıp fonksiyonunda ikili çapraz entropi kayıp fonksiyonu kullanılmıştır. YOLOv3 yönteminin kayıp fonksiyonu Denklem 3.23'te verilmiştir.

$$\begin{aligned}
& \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] + \\
& \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} \left[ \left( \sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left( \sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] + \\
& \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{noobj} (C_i - \hat{C}_i)^2 + \\
& \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2
\end{aligned} \tag{3.23}$$

Çizelge 3.1: Darknet-53 mimarisi.

| Katman Tipi                   | Filtre Sayısı | Filtre Boyutu    | Çıktı Boyutu     |    |
|-------------------------------|---------------|------------------|------------------|----|
| Konvolüsyonel                 | 32            | $3 \times 3$     | $256 \times 256$ |    |
| Konvolüsyonel                 | 64            | $3 \times 3 / 2$ | $128 \times 128$ |    |
| Konvolüsyonel                 | 32            | $1 \times 1$     |                  | ×1 |
| Konvolüsyonel                 | 64            | $3 \times 3$     |                  |    |
| Artık (Residual)              |               |                  | $128 \times 128$ |    |
| Konvolüsyonel                 | 128           | $3 \times 3 / 2$ | $64 \times 64$   |    |
| Konvolüsyonel                 | 64            | $1 \times 1$     |                  | ×4 |
| Konvolüsyonel                 | 128           | $3 \times 3$     |                  |    |
| Artık (Residual)              |               |                  | $64 \times 64$   |    |
| Konvolüsyonel                 | 256           | $3 \times 3 / 2$ | $32 \times 32$   |    |
| Konvolüsyonel                 | 128           | $1 \times 1$     |                  | ×8 |
| Konvolüsyonel                 | 256           | $3 \times 3$     |                  |    |
| Residual                      |               |                  | $32 \times 32$   |    |
| Konvolüsyonel                 | 512           | $3 \times 3 / 2$ | $16 \times 16$   |    |
| Konvolüsyonel                 | 256           | $1 \times 1$     |                  | ×8 |
| Konvolüsyonel                 | 512           | $3 \times 3$     |                  |    |
| Artık (Residual)              |               |                  | $16 \times 16$   |    |
| Konvolüsyonel                 | 1024          | $3 \times 3 / 2$ | $8 \times 8$     |    |
| Konvolüsyonel                 | 512           | $1 \times 1$     |                  | ×4 |
| Konvolüsyonel                 | 1024          | $3 \times 3$     |                  |    |
| Artık (Residual)              |               |                  | $8 \times 8$     |    |
| Ort. Örnekleme (Avg. Pooling) |               | Global           |                  |    |
| Bağlı (Connected)             |               | 1000             |                  |    |
| Softmax                       |               |                  |                  |    |

Burada ilk terim yanlış hücre merkezi tahminlerini, ikinci terim ise karekök işleminden dolayı daha büyük çevreleyici kutularla karşılaştırıldığında daha küçük çevreleyici kutulardaki düşük olan hataları düzenlemek için genişlik ve yükseklik hatalarını minimize etmeye çalışır. Üçüncü terim, bir nesne varsa güvenilirlik skorunu yükseltmeye, dördüncü terim ise bir nesne yoksa güvenilirlik skorunu en aza indirmeye çalışır ve beşinci terim sınıflandırmaya ait kayıptır.

Rektife edilmiş imgeler üzerinde YOLOv3 ile yapılan nesne tespiti sonucu çok katmanlı ızgara tabanlı harita sayesinde tespit edilen konumun yerel ve küresel



koordinatları elde edilir. Objenin küresel koordinatları, ilgili objenin gerçek dünya konumunun elde edilmesini sağlar.

### 3.2.6 Ortomozaik

Ortorektifikasyon ve nesne tespiti yapıldıktan sonra, elde edilen bütün veriler birleştirilerek yüksek çözünürlüklü bir ortomozaik oluşturulabilir. Ortorektifikasyon aşamasından alınan ilk imge ile küresel harita başlatılır. Başlatma işleminden sonra rektife edilmiş ve nesne tespiti yapılmış imgeler ile oluşturulan iki boyutlu ve üç boyutlu haritalar sürekli olarak güncellenir. Harita güncelleme işleminde gözlemlenen alanın önceden gözlemlenmemiş bir bölge veya haritada önceden gözlemlenmiş örtüşen bölge olması gibi iki durum vardır. Birinci durumda yeni imgeye ait veriler haritaya eklenerek harita güncellenir. İkinci durumda, küresel harita ile imgenin kesişime sahip olduğu bölgelerde, ızgara haritasındaki piksellerin renk değerleri ve yükseltilerinin belirlenmesi gerekmektedir. Örtüşen bölgelerdeki hücrelerin değeri minimum yükselti açısına göre belirlenir (Hinzmann vd., 2018). Küresel haritada, minimum yükselti açısına sahip olan hücrenin renk ve yükseklik değerleri ile küresel haritadaki hücre güncellenir.

Ortomozaik aşamasında örtüşme olan bölgelerde eklenecek olan yeni verilerde tespit edilen nesnelere çevreyici kutularının küresel koordinatlarında maksimum olmayan baskılama (non-maximum suppression) uygulanır. Bu şekilde aynı nesnenin haritada birden fazla çevreyici kutuyla ifade edilmesi engellenir.

## 4. BULGULAR VE TARTIŞMA

Geliştirilen haritalama ve konumlandırma sisteminin performansı, hem tekli hem de çoklu İHA senaryolarında test edilmiştir. Ek olarak, sistemde bulunan nesne tespiti için özgün araç veri seti ile ilgili eğitimler ve testler yapılmıştır. İlk önce nesne tespiti ile ilgili deneyler anlatılmış ve daha sonra haritalama ve konumlandırma sisteminin performansı değerlendirilmiştir.

### 4.1 Nesne Tespiti

Haritalama ve konumlandırma sisteminde hava görüntülerinden nesne tespiti işlemi için tespit edilmek istenen nesne araba olarak tercih edilmiştir. Hava görüntülerde araba tespiti için YOLOv3 özgün veri seti ile eğitilmiştir. YOLOv3 ağını eğitmek için, DJI matrice 600 Pro ve DJI Inspire 1 insansız hava araçları kullanılarak farklı hava koşullarında ve ölçeklerde elde edilen renkli dijital görüntülerden oluşan araba veri seti oluşturulmuştur (Saribaş vd., 2018). Veri seti havadan yere bakan bir kameradan çekilmiş yaklaşık 10.000 renkli dijital görüntüden oluşmakta ve yaklaşık 30.000 adet araba örneği içermektedir. Pozitif sınıfa ait veriler çevreleyici kutularla etiketlenerek oluşturulmuştur. Şekil 4.1, veri kümesinden pozitif örnekler içeren görüntüleri göstermektedir. Verilerin %80'i (yaklaşık 24 bin pozitif örnek) YOLOv3 dedektörünün eğitimi için ve kalan %20'si test için kullanılmıştır. YOLOv3 araba modelinin eğitimi NVIDIA Quadro P5000 GPU üzerinde yapılmıştır.

Eğitilen YOLOv3 araba modelinin performansını ölçmek amacıyla, nesne tespiti için en sık kullanılan metrik olan PASCAL VOC ölçütü kullanılmıştır. Bu metriğe göre, nesnenin konumu, tespit edilen nesne koordinatlarının ve gerçek referans konumlarının kesişim oranına göre yanlış veya doğru olarak sınıflandırılır. Bu kesişim aşağıda verilen formül ile hesaplanır.

$$\frac{alan|Q \cap R|}{alan|Q \cup R|} \quad (4.1)$$

Denklemden  $Q$  nesnenin gerçek referans konumunu,  $R$  ise algoritmanın tespit ettiği nesne konumunu ifade eder. Eğer hesaplanan oran %50'den daha yüksekse tespit edilen konum gerçek pozitif olarak değerlendirilir. Eğer oran %50'den daha düşükse tespit edilen konum yanlış pozitif olarak belirlenir. Daha sonra ortalama hassasiyet değeri doğruluk ve hassasiyet eğrisi ile hesaplanır. Eğitilen YOLOv3 araba modeli, araba veri seti üzerinde %83.4 ortalama hassasiyet değeri elde etmiştir (Saribaş vd., 2018).



Şekil 4.1: Araba veri setine ait örnek imgeler.

## 4.2 Ortomozaik Üretimi

Yöntemin haritalama ve konumlandırma performansını ölçmek ve değerlendirmek amacıyla iki farklı veri seti üzerinde deneyler yapılmıştır. Bu deneylerde yöntemin gözlemlenen bölgeyi 2 boyutlu ve 3 boyutlu haritalandırma kabiliyeti, hava aracını konumlandırmasındaki doğruluğu ve nesnelere tespit edip haritada gösterme başarımı ölçülmüştür.

İlk olarak Kern, 2018’de önerilen veri setinin bir kısmı ortomozaik üretiminde performans ölçümü için kullanılmıştır. Veri seti, yaklaşık 100 metrelik yükseklikten insansız hava aracı üzerinde bulunan yere bakan bir kameradan çekilen hava görüntülerinden oluşmaktadır. Kullanılan veri seti,  $1228 \times 1026$  boyutunda yaklaşık 1500 imgeden oluşmaktadır. Hava görüntüleri 10 FPS’te çekilmiştir. Her imgeye ait GPS verisi Exif formatında tutulmaktadır. Şekil 4.2’de İHA’dan alınan görüntülerle tek İHA senaryosuyla oluşturulmuş 2 boyutlu ortomozaik verilmiştir. Ortomozaikte eğitilen YOLOv3 araba modeli ile tespit edilmiş nesnelere gösterilmiştir. Harita üzerinde tespit

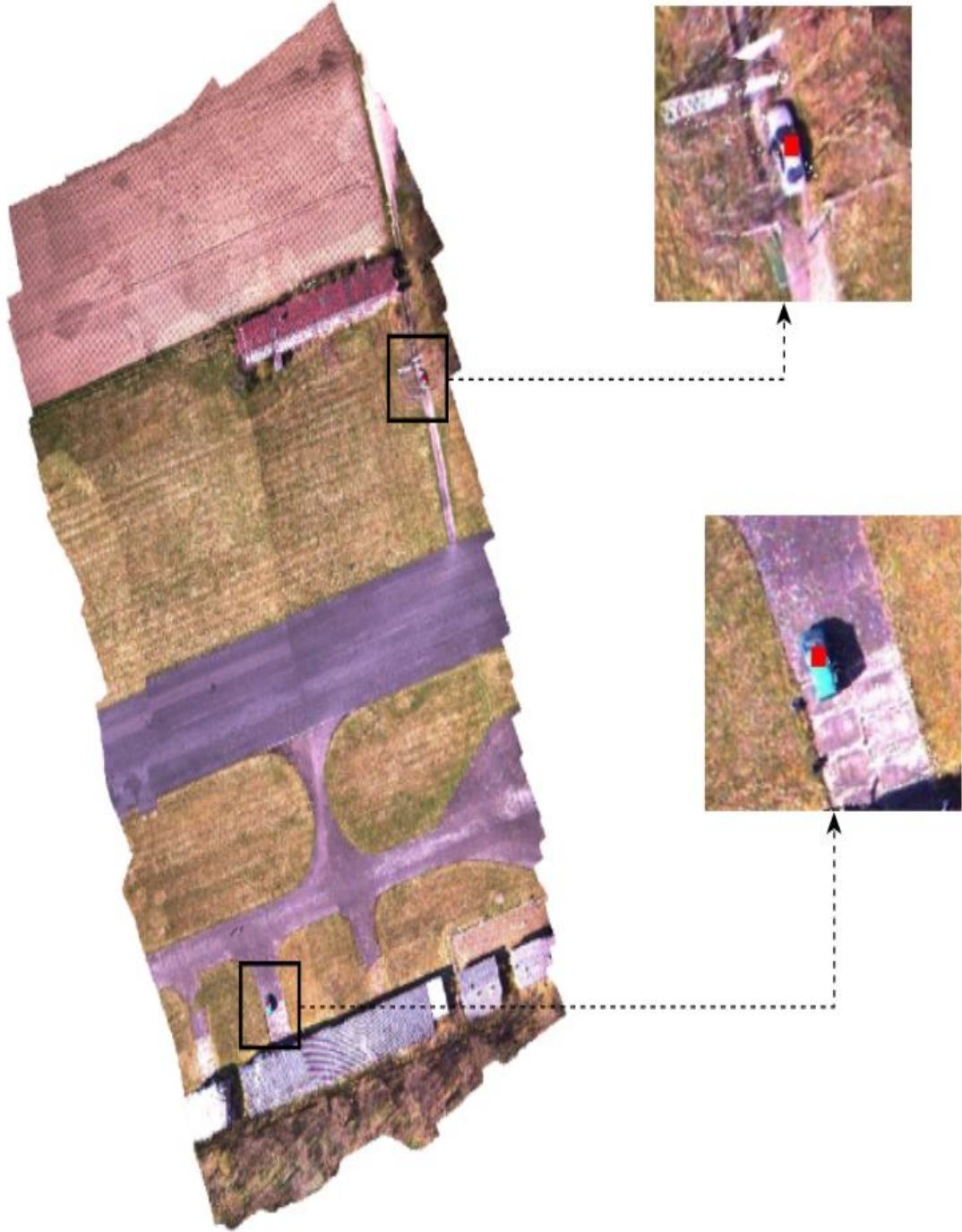
edilen nesnelere merkezi kırmızı kutu ile belirtilmiştir. Şekilde verilen ortomozaik artımlı ızgara tabanlı ortomozaik oluşturma yöntemi ile elde edilmiştir.

İyi bir ortomozaik yüksek çözünürlüğe, homojen görünüme ve düşük geometrik bozulmaya sahip olmalıdır. Gözlemlenen bölge oldukça detaylı bir biçimde haritalandırılmalı ve küresel ölçekte yüksek doğrulukta hizalanmalıdır. Ortomozaik üretiminde, küresel konum belirleme sırasında yapılan hatalı hesaplamalar yanlış görüntü hizalamalarına sebebiyet verebilmektedir. Haritalama ve konumlandırma işlemi sırasında SLAM ve GPS verilerinin birleştirilmesi küresel konum hesabını gürbüzleştirir de hatalı GPS verileri ve hatalı SLAM konum hesaplamaları görsel bozulmalara sebep olmaktadır. Oluşturulan ortomozaik görüntüde kesişim bölgelerinin birleşme noktalarında oluşan dikiş (seam) izleri gözlemlenebilir. Fakat oluşturulan yüksek çözünürlüklü haritada ortama ait yüzey bilgisi, gözlemlenen arazideki yapılar ve yollar doğru bir şekilde gözlemlenebilmektedir. Ayrıca, araba veri setiyle eğitilen YOLOv3 modeli görüntülenilen bölgedeki araçları doğru bir şekilde konumlandırmaktadır.

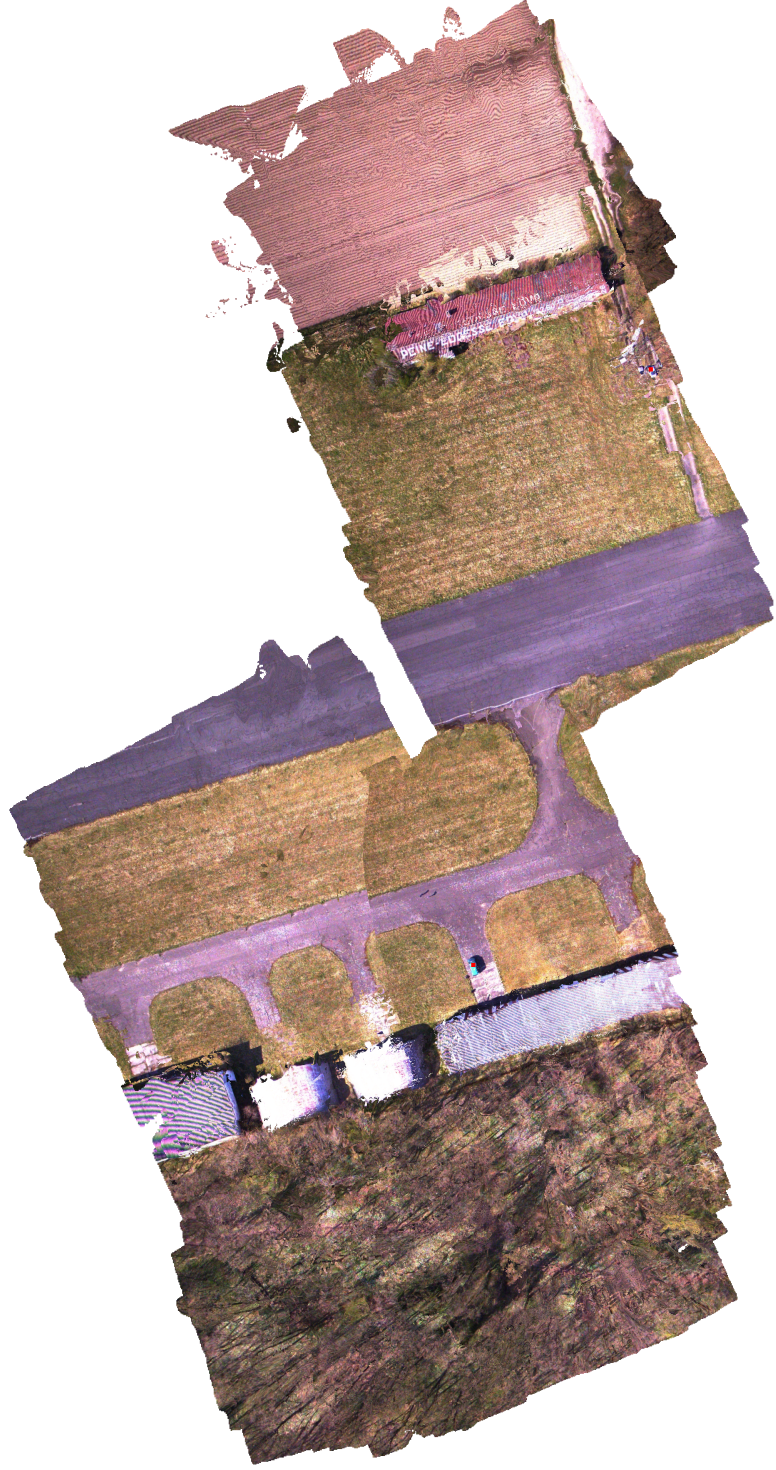
Şekil 4.3'te İHA'dan alınan görüntülerle çoklu İHA senaryosuyla oluşturulmuş ortomozaik verilmiştir. Yöntemi çoklu İHA senaryosunda test edebilmek için veri seti ikiye bölünmüştür. Veri seti ikiye bölünerek imgelerin iki farklı kaynaktan alındığı kabul edilmiştir. İki veri seti küçük bir kesişim içerecek şekilde bölünmüştür. Gözlemlenen bölgenin gerçek dünya ölçeğinde başarılı bir şekilde ölçeklendiği çoklu senaryoda daha iyi gözlemlenebilmektedir. Yöntem şekilde görüldüğü gibi farklı kaynaklardan alınan görüntüleri aynı düzleme başarılı bir şekilde haritalayabilmektedir. Gözlemlenen bölge, küresel koordinat sisteminde haritalandırılmıştır. Böylelikle konumlar arasındaki mesafeler gerçek dünya ölçeğinde ölçülebilmekte ve tespit edilen nesnelere gerçek dünya konumları elde edilebilmektedir.

Seyrek nokta bulutu interpolasyonu kullanılarak oluşturulan 3 boyutlu harita Şekil 4.4'te verilmiştir. Seyrek bulut interpolasyonu, keskin yükseklik geçişlerindeki bilgileri kurtarmakta başarısız olmaktadır. Bu, interpolasyonu yaklaşımının, 3 boyutlu haritalama için oldukça ayırt edici öznelikler kullanan öznelik tabanlı ORB SLAM'in seyrek nokta bulutu hesaplamasına dayalı olmasından kaynaklanmaktadır. Sonuç olarak, ORB-SLAM ile üretilen seyrek bulutun düşük çözünürlüğü belirsiz yükselti hesaplamalarına yol açar, bu da görsel görünümü bozar. Bununla birlikte, oluşturulan 3 boyutlu harita genel olarak düşük görüntüye sahip homojen bir yapıda ve gözlemlenen alanın yapısı ile ilgili yeterli bilgi içermektedir.

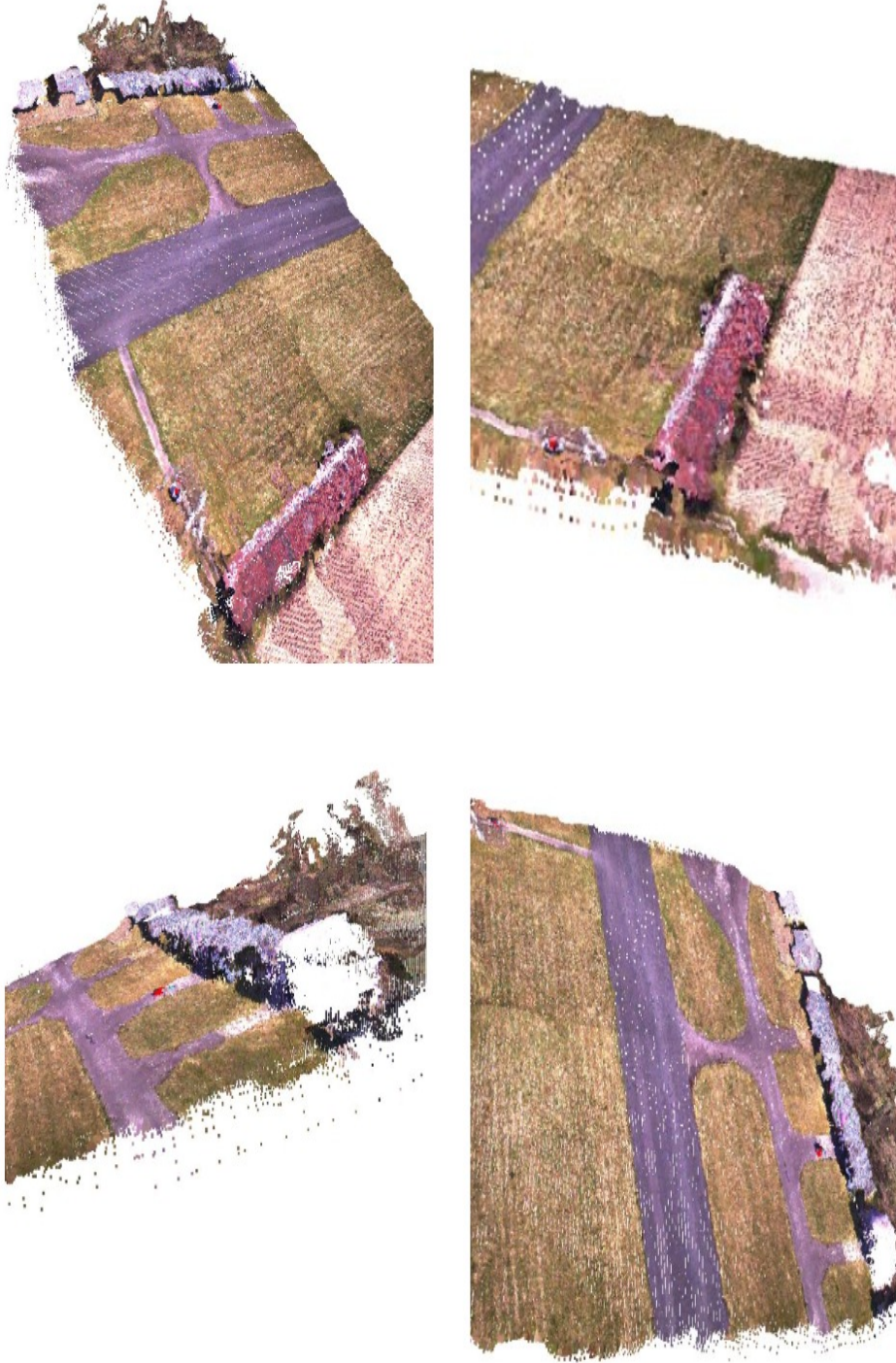
Önerilen yöntemin performansını değerlendirmek için gerçek referans veri olmadığından, yüksek doğrulukta çalışan bir haritalama yazılımı olan PhotoScan



Şekil 4.2: Tek İHA senaryosunda oluşturulan ortomozaik görüntü.



Şekil 4.3: Çoklu İHA senaryosunda oluşturulan ortomozaik görüntü.



Şekil 4.4: Oluşturulan 3 boyutlu haritanın farklı açılardan görünümü.

(Verhoeven, 2011) ile gerçek referans veri üretilmiştir. PhotoScan kullanılarak oluşturulan harita ve kamera konumları gerçek referans verileri olarak kabul edilmiştir. Kamera konum hesabının performansını değerlendirmek için görece konum hatası (relative pose error) (RPE) ve mutlak konum hatası (absolute pose error) (APE) metrikleri kullanılmıştır (Kummerle vd., 2009). APE, mutlak yörünge hatası olarak da adlandırılır. Konum tahminleri ve referans kamera konumları arasında doğrudan bir karşılaştırma yapar. APE yörüngenin küresel tutarlılığını test eder. RPE, kameranın öteleme ve dönme hareketlerini karşılaştırır ve metre başına öteleme ve dönme sapmasını hesaplar. Önerilen yöntem, ORB-SLAM ve GPS'in kamera konum hesaplama performansını karşılaştırmak için veri seti üzerinde yöntemlerin yörüngeleri oluşturulmuştur. Oluşturulan yörüngelerin RPE ve APE değerleri, Çizelge 4.1 ve Çizelge 4.2'de PhotoScan tarafından üretilen gerçek referans verileri ile karşılaştırılmıştır.

Çizelge 4.1: Önerilen yöntem, ORB-SLAM ve GPS'in ilk veri seti üzerindeki RPE sonuçları

|                 | <b>maks</b> | <b>ort</b> | <b>medyan</b> | <b>min</b> | <b>rmse</b> | <b>sse</b> | <b>std</b> |
|-----------------|-------------|------------|---------------|------------|-------------|------------|------------|
| <b>ORB-SLAM</b> | 1.12        | 0.27       | 0.25          | 0.02       | 0.35        | 47.32      | 0.22       |
| <b>GPS</b>      | 2.34        | 0.60       | 0.54          | 0.06       | 0.69        | 194.32     | 0.34       |
| <b>YÖNTEM</b>   | 0.96        | 0.22       | 0.17          | 0.01       | 0.29        | 35.08      | 0.19       |

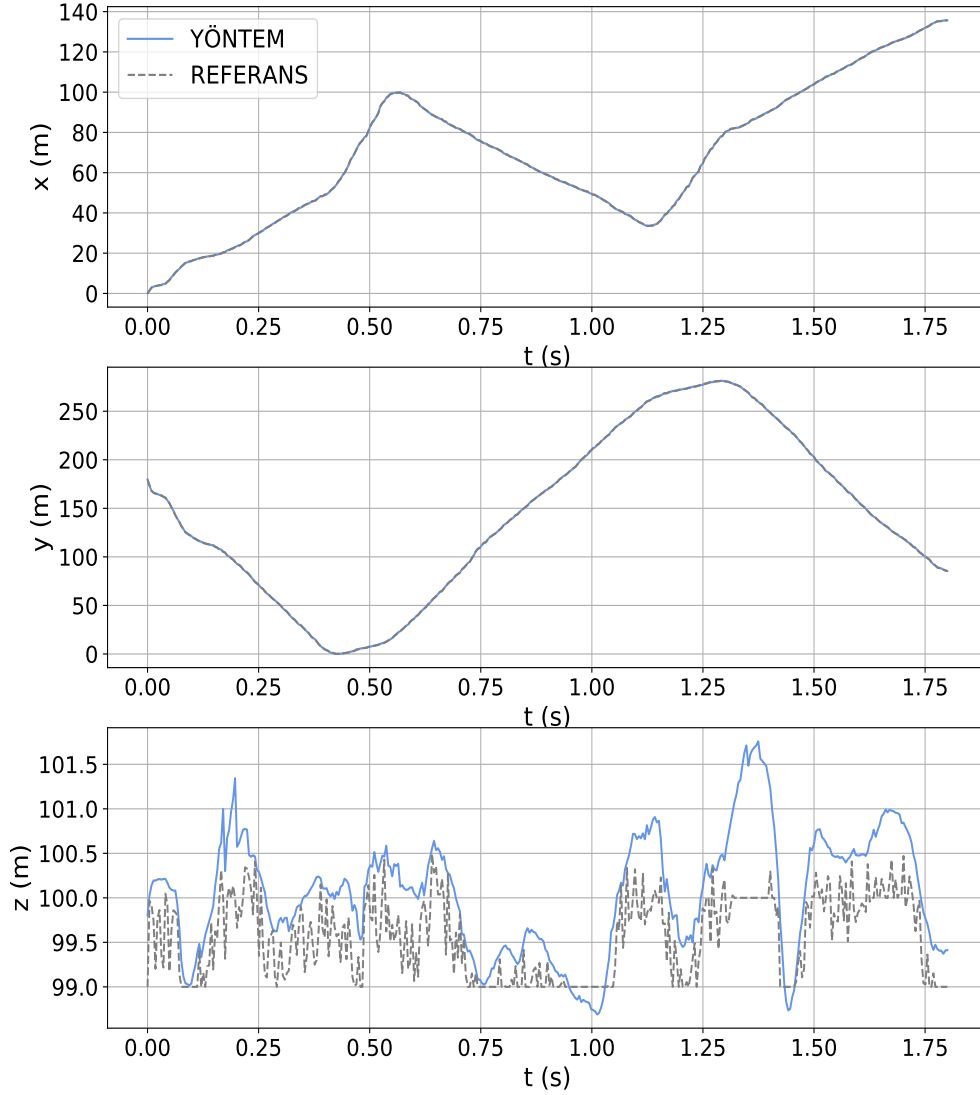
Çizelge 4.2: Önerilen yöntem, ORB-SLAM ve GPS'in ilk veri seti üzerindeki APE sonuçları

|                 | <b>maks</b> | <b>ort</b> | <b>medyan</b> | <b>min</b> | <b>rmse</b> | <b>sse</b> | <b>std</b> |
|-----------------|-------------|------------|---------------|------------|-------------|------------|------------|
| <b>ORB-SLAM</b> | 2.01        | 0.69       | 0.62          | 0.03       | 0.88        | 222.54     | 0.38       |
| <b>GPS</b>      | 2.78        | 1.12       | 1.11          | 0.08       | 1.21        | 595.42     | 0.46       |
| <b>YÖNTEM</b>   | 1.75        | 0.52       | 0.46          | 0.01       | 0.63        | 161.15     | 0.35       |

Tablodaki düşük standart sapma (std) değerleri (hem RPE hem de APE için 1 metrenin altında) kamera konum tahmininin düşük hataya sahip olduğunu göstermektedir. Ayrıca, önerilen yöntem, ORB-SLAM ve GPS'ten daha düşük hata değerleri elde etmiştir. Bu da önerilen yöntemin ORB-SLAM'in konum hesaplamasının gürbüzülüğünü artırdığını göstermektedir. Şekil 4.5, tek İHA senaryosunda üretilen ortomozaik için  $x$ ,  $y$  ve  $z$  düzlemine göre yörünge grafiklerini göstermektedir.  $x$  ve  $y$  düzlemlerinde, yöntem ile elde edilen kamera konumlarında gerçek referans konumlarına göre herhangi bir kayma veya hata gözlemlenmemiştir. Ancak, yükseklik arttığında,  $z$  düzleminde anlık bir kayma ve hata

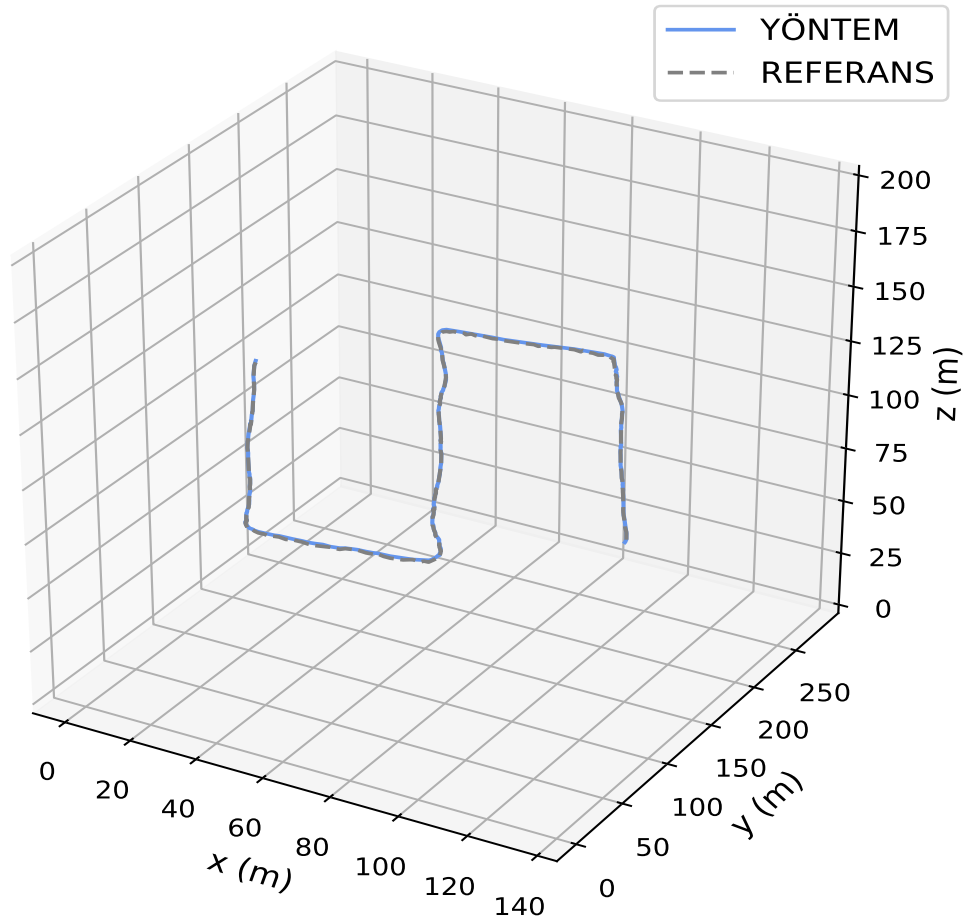


oluşturmuştur. Bu durumun temel sebebi, ORB SLAM'in tespit ve takip için yeteri kadar öznitelik bulundurmayan bölgelerde hatalı derinlik tahminidir. İHA'nın yöntem ile oluşturulan yörüngesi Şekil 4.6'da verilmiştir. Yöntemin kamera konumlandırmadaki gürbüzlüğü ve başarımı şekilden de gözlemlenebilmektedir. Yöntemle oluşturulan yörünge neredeyse gerçek referans yörüngesine eşittir. Veri seti için sistemin çalışma süresi yaklaşık 4 FPS sürmektedir.



Şekil 4.5: Gerçek referans ile önerilen yöntemin konum eğrilerinin  $x$ ,  $y$  ve  $z$  düzleminde metre cinsinden ilk veri seti üzerindeki karşılaştırması.

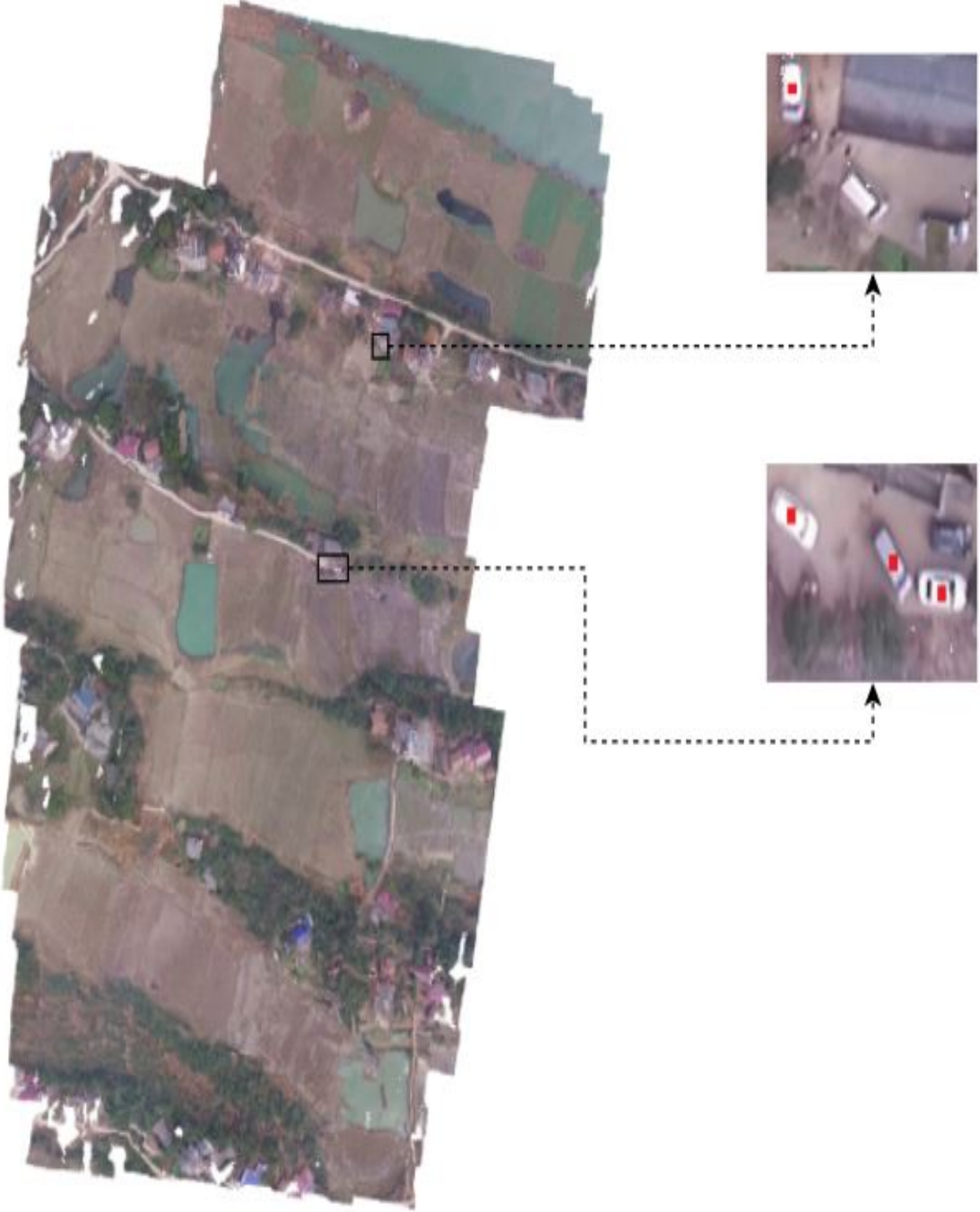
Yöntemin performansı ayrıca phantom3-village (Bu vd., 2016) veri setinde tek İHA senaryosunda değerlendirilmiştir. Veri seti, yere bakan bir GoPro Hero3+ kamera ile DJI Phantom3 kullanılarak yaklaşık 160 metre yükseklikte kaydedilmiştir. Veri seti yaklaşık



Şekil 4.6: Gerçek referans ile önerilen yöntemin ilk veri seti üzerinde üretilmiş yörüngeleri.

200 imgeden oluşmaktadır. Şekil 4.7, tek İHA senaryosunda phantom3-village veri seti üzerinde oluşturulan ortomozaik görüntüyü ve tespit edilen nesnelere göstermektedir. Oluşturulan harita, düzlemsel bölgeler ve binalar ile ağaçlar içeren alanlarda tatmin edici sonuçlar elde etmiştir. Haritanın oluşturulması büyük ölçüde öznitelik tabanlı SLAM yaklaşımına bağlı olduğundan yöntem yeteri kadar öznitelik bulundurmayan su bölgeleriyle ilgili bilgileri kurtarmakta başarısız olmaktadır. Bu problem, üretilen ortomozaik üzerinde bazı yanlış hizalamalar ile sonuçlanmıştır. İrtifa çok yüksek olduğundan, YOLOv3 modeli bazı araçları tespit edememiştir.

Seyrek bulut interpolasyonu kullanılarak oluşturulan 3 boyutlu harita Şekil 4.8'de verilmiştir. Phantom3-village veri seti 1 FPS'de toplanmıştır. Düşük FPS, ORB-SLAM'in derinlik haritası oluşturma performansını ciddi şekilde etkilemekte ve derinlik haritası interpolasyonunun gürültülü sonuçlar üretmesine neden olmaktadır. Haritalama ve konumlandırma sistemi, öznitelik tabanlı SLAM'ın doğası gereği, ayırt edici öznitelik



Şekil 4.7: Tek İHA senaryosunda ikinci veri seti üzerinde oluşturulan ortomosaik görüntü.

içermeyen bölgelerde 3 boyutlu yüzeyleri düzgün bir şekilde oluşturamamaktadır. Seyrek bulut interpolasyonu, önceki veri seti için yapılan deneylerde olduğu gibi keskin yükseklik geçişlerinde doğru sonuçlar üretmede başarısız olmuştur. Oluşturulan 2 boyutlu haritada görüldüğü gibi, yöntem aynı zamanda su bölgelerinin öznelik içermeyen yüzey yapısından dolayı su bölgeleri için harita noktaları oluşturmada başarısız olmaktadır.

Veri seti için gerçek referans verisi olmadığından, gerçek referans verileri birinci veri seti için yapıldığı gibi PhotoScan ile üretilmiştir. Veri seti üzerinde önerilen yöntemin, ORB-SLAM'in ve GPS'e ait konum yörüngeleri oluşturulmuştur. Yöntemlerin RPE ve APE değerleri, Çizelge 4.3 ve Çizelge 4.4'te gerçek referans verileri ile karşılaştırılmıştır. İlk veri setine benzer şekilde, önerilen yöntem hem APE hem de RPE metrikleri için en düşük hatayı elde etmiştir. Standart sapma (std) değerlerinin büyüklüğü, hatalı kamera konum tahminlerini göstermektedir.

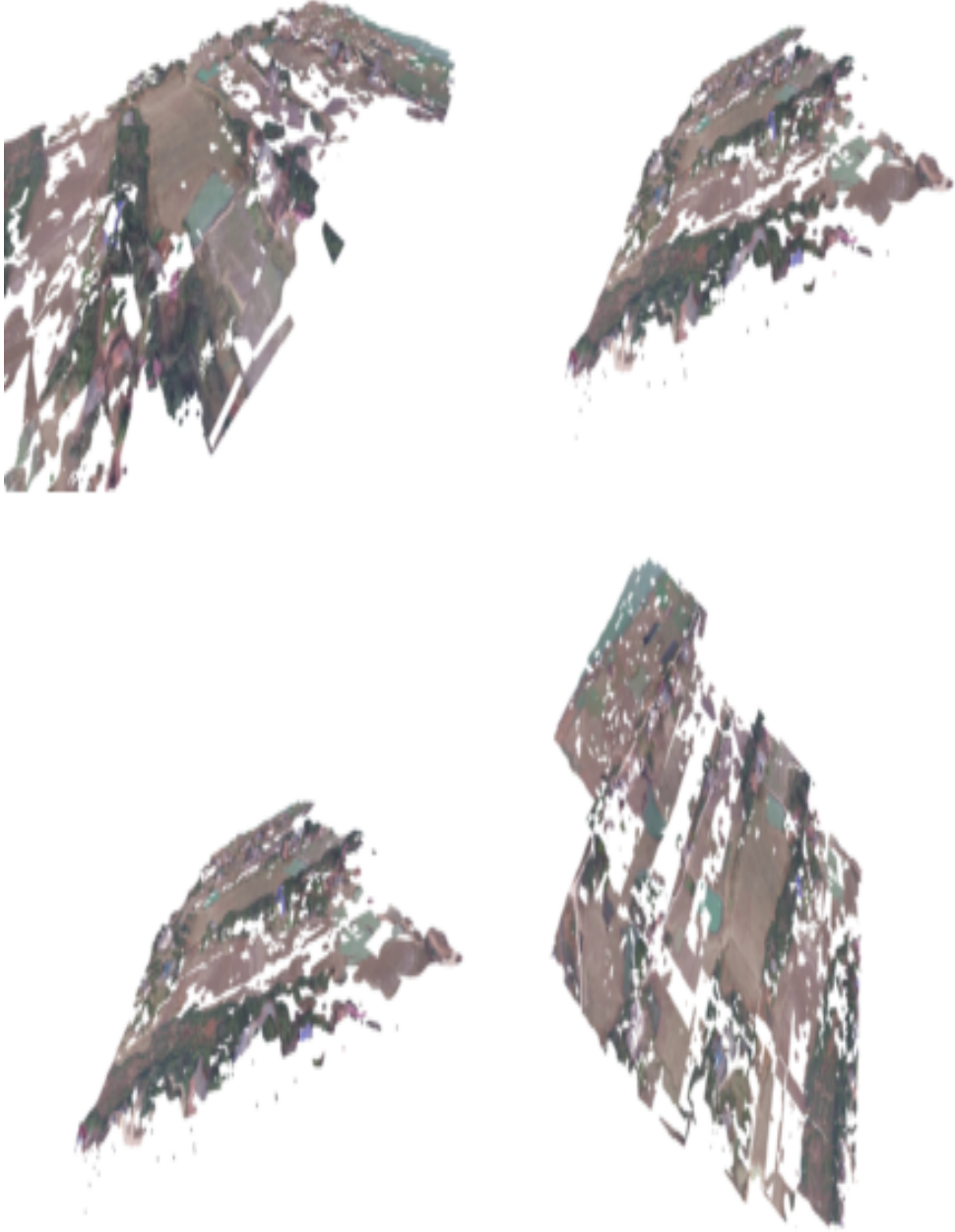
Çizelge 4.3: Önerilen yöntem, ORB-SLAM ve GPS'in ikinci veri seti üzerindeki RPE sonuçları

|                 | <b>maks</b> | <b>ort</b> | <b>medyan</b> | <b>min</b> | <b>rmse</b> | <b>sse</b> | <b>std</b> |
|-----------------|-------------|------------|---------------|------------|-------------|------------|------------|
| <b>ORB-SLAM</b> | 16.23       | 6.23       | 5.92          | 0.82       | 7.02        | 591.33     | 4.01       |
| <b>GPS</b>      | 25.95       | 11.23      | 10.68         | 1.32       | 9.42        | 1042.58    | 6.23       |
| <b>YÖNTEM</b>   | 14.68       | 5.63       | 4.81          | 0.73       | 6.49        | 486.70     | 3.21       |

Çizelge 4.4: Önerilen yöntem, ORB-SLAM ve GPS'in ikinci veri seti üzerindeki APE sonuçları

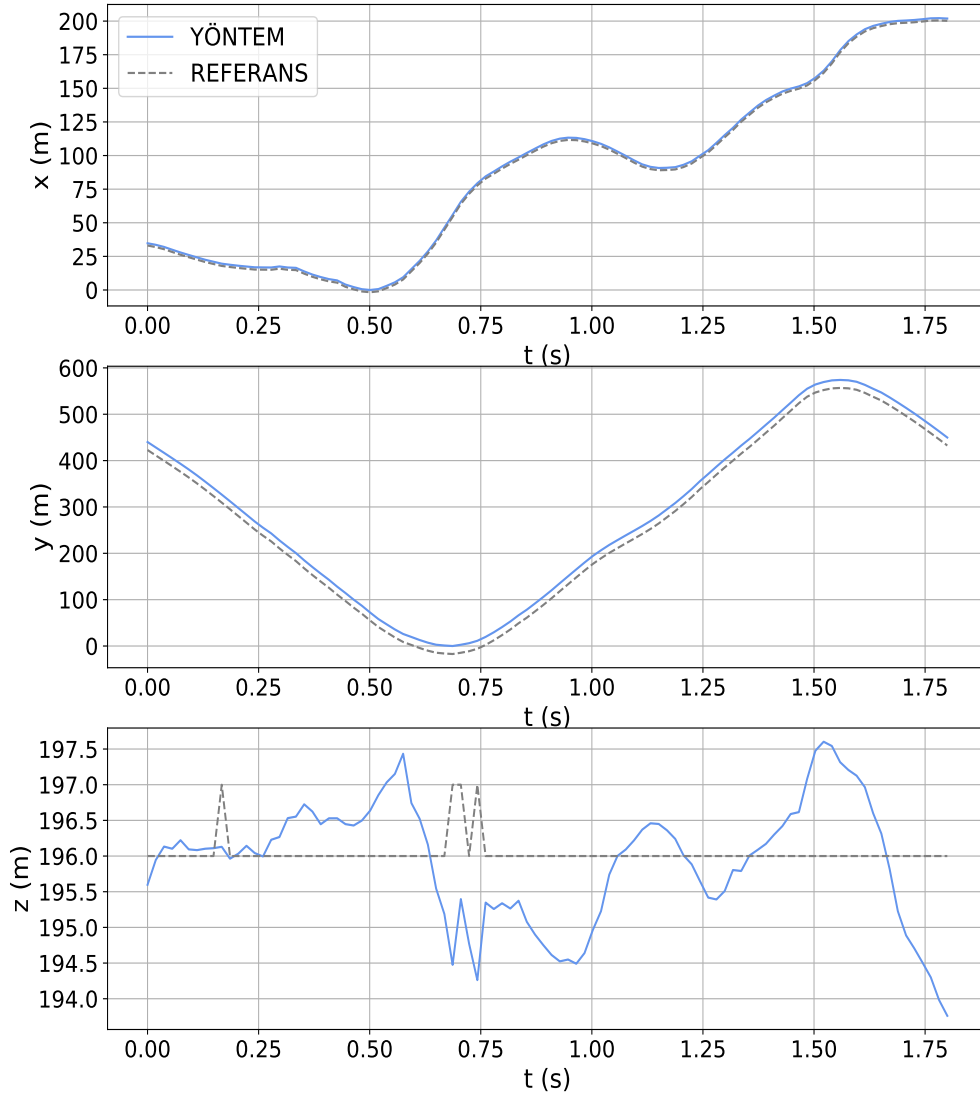
|                 | <b>maks</b> | <b>ort</b> | <b>medyan</b> | <b>min</b> | <b>rmse</b> | <b>sse</b> | <b>std</b> |
|-----------------|-------------|------------|---------------|------------|-------------|------------|------------|
| <b>ORB-SLAM</b> | 28.12       | 14.02      | 14.75         | 1.28       | 16.12       | 2304.39    | 8.01       |
| <b>GPS</b>      | 38.17       | 22.19      | 21.34         | 2.26       | 23.12       | 2884.42    | 9.54       |
| <b>YÖNTEM</b>   | 26.11       | 12.69      | 13.63         | 1.07       | 14.34       | 2165.06    | 6.68       |

Veri setinde görüntüler yüksek irtifalarda toplanmıştır (yaklaşık 160 metre). Yüksek irtifadan çekilmiş imgeler ORB-SLAM'in kamera konum tahmin performansını büyük ölçüde etkilemektedir. Şekil 4.9, phantom3-village veri setinde tek İHA senaryosunda üretilen ortomozaik için  $x$ ,  $y$  ve  $z$  düzlemine göre yörünge grafiklerini göstermektedir. Şekil 4.5'e benzer şekilde, yöntem  $x$  ve  $y$  düzlemlerinde herhangi bir kayma veya hata

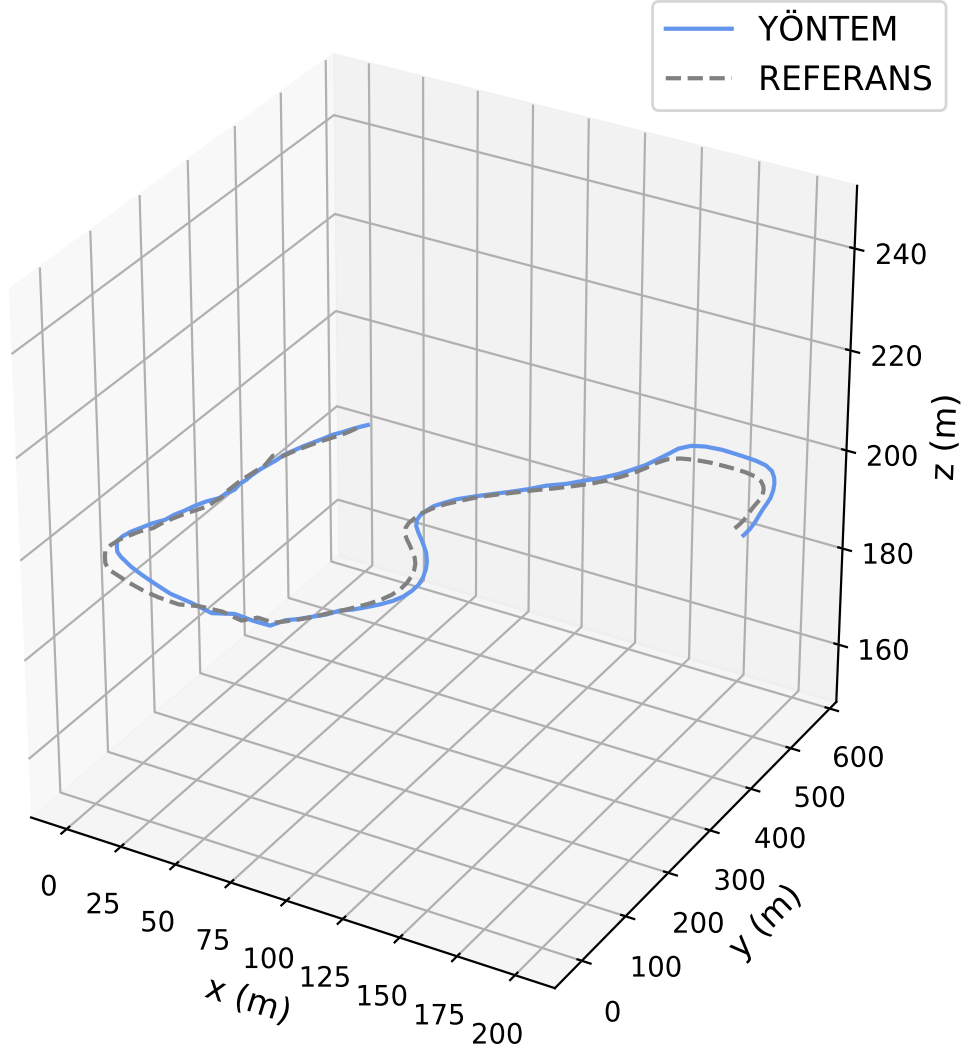


Şekil 4.8: İkinci veri setin oluşturulan üç boyutlu haritasının farklı bakış açılarından görünümü.

yaşamamıştır. Ancak,  $z$  düzleminde anlık kaymalar ve hatalar gözlemlenebilmektedir. Bu hatalar, su bölgeleri gibi öznitelik içermeyen bölgelerdeki hatalı derinlik tahminlerinden ve görüntülerin çok yüksek irtifalardan toplanmış olmasından kaynaklanmaktadır. İHA'nın oluşturulan yörüngesi Şekil 4.10'da verilmiştir. Veri seti için sistemin çalışma süresi yaklaşık 4 FPS sürmektedir.



Şekil 4.9: Gerçek referans ile önerilen yöntemin konum eğrilerinin  $x$ ,  $y$  ve  $z$  düzleminde metre cinsinden ikinci veri seti üzerindeki karşılaştırması.



Şekil 4.10: Gerçek referans ile önerilen yöntemin ikinci veri seti üzerinde üretilmiş yörüngeleri.

## 5. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, insansız hava aracının kamera ve GPS verileri birleştirilerek gözlemlenen bir bölgeyi gerçek zamanlı olarak iki ve üç boyutlu haritalandıran, nesne tespiti özelliğine sahip bir yöntem geliştirilmiştir. Yöntem, kamera konumlandırma ve seyrek nokta bulutu üretimi için gürbüz bir monoküler SLAM yöntemi olan ORB-SLAM'i kullanmaktadır. Kamera konumlandırmayı iyileştirmek ve gerçek dünya ölçeğinde haritalama yapabilmek amacıyla görsel konum hesaplamaları GPS verileriyle birleştirilmektedir. Yoğun nokta bulutu oluşturmak için ORB-SLAM ile üretilen seyrek nokta bulutu üzerine oldukça hızlı çalışan nokta bulutu interpolasyon algoritması uygulanmaktadır. Tez kapsamında önerilen yöntem artımlı ızgara tabanlı ortomozaik yöntemi ile anlamlı, yüksek çözünürlüklü haritalar oluşturabilmektedir. Aynı zamanda YOLOv3 nesne konumlandırma yöntemi ile gözlemlenen bölgelerdeki ilgili nesnelere haritalama sırasında tespit edilip harita üzerinde işaretlenmektedir. Yöntem oluşturulan 2 boyutlu ve 3 boyutlu ortomozaik haritayı küresel koordinat düzleminde ifade etmekte ve tespit edilen nesnelere gerçek dünya konumlarını belirleyebilmektedir. Derin öğrenme tabanlı YOLOv3 nesne konumlandırma yönteminin eğitimi için özgün bir araba veri seti kullanılmış ve başarıları ölçülmüştür. Haritalandırma performansını değerlendirmek için havadan çekilmiş görüntüler ve GPS verisine sahip iki veri seti üzerinde deneyler yapılmıştır. Gerçekleştirilen deneylerde önerilen yöntemin, düşük hata ve gerçek zamanlı çalışma değiş tokuşunda dengeli bir performansa sahip olduğu gözlemlenmektedir.



## KAYNAKLAR DİZİNİ

- Agarwala, A., Agrawala, M., Cohen, M., Salesin, D. ve Szeliski, R. (2006), Photographing long scenes with multi-viewpoint panoramas, *ACM SIGGRAPH 2006 Papers*, 853–861.
- Bertalmio, M., Bertozzi, A. L. ve Sapiro, G. (2001), Navier-stokes, fluid dynamics, and image and video inpainting, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, I–I.
- Bradski, G. (2000), The OpenCV Library, *Dr. Dobb's Journal of Software Tools*.
- Brown, M. ve Lowe, D. G. (2007), Automatic panoramic image stitching using invariant features, *International Journal of Computer Vision* 74.1, 59–73.
- Bu, S., Zhao, Y., Wan, G. ve Liu, Z. (2016), Map2DFusion: real-time incremental UAV image mosaicing based on monocular slam, *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4564–4571.
- Calonder, M., Lepetit, V., Strecha, C. ve Fua, P. (2010), Brief: Binary robust independent elementary features, *European conference on computer vision*. Springer, 778–792.
- Concha, A. ve Civera, J. (2015), DPPTAM: Dense piecewise planar tracking and mapping from a monocular sequence, *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 5686–5693.
- Dalal, N. ve Triggs, B. (2005), Histograms of oriented gradients for human detection, *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR '05)*. IEEE, 886–893.

## KAYNAKLAR DİZİNİ (devam)

- Davison, A. J., Reid, I. D., Molton, N. D. ve Stasse, O. (2007), MonoSLAM: Real-time single camera SLAM, *IEEE transactions on pattern analysis and machine intelligence* 29.6, 1052–1067.
- Engel, J., Koltun, V. ve Cremers, D. (2017), Direct sparse odometry, *IEEE transactions on pattern analysis and machine intelligence* 40.3, 611–625.
- Engel, J., Schöps, T. ve Cremers, D. (2014), LSD-SLAM: Large-scale direct monocular SLAM, *European Conference on Computer Vision*. Springer, 834–849.
- Engel, J., Sturm, J. ve Cremers, D. (2013), Semi-dense visual odometry for a monocular camera, *Proceedings of the IEEE international conference on computer vision*, 1449–1456.
- Fankhauser, P. ve Hutter, M. (2016), A universal grid map library: Implementation and use case for rough terrain navigation, *Robot Operating System (ROS)*. Springer, 99–120.
- Faugeras, O. D. ve Lustman, F. (1988), Motion and structure from motion in a piecewise planar environment, *International Journal of Pattern Recognition and Artificial Intelligence* 2.03, 485–508.
- Fischler, M. A. ve 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.

## KAYNAKLAR DİZİNİ (devam)

- Forster, C., Pizzoli, M. ve Scaramuzza, D. (2014), SVO: Fast semi-direct monocular visual odometry, *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 15–22.
- Fraundorfer, F. ve Scaramuzza, D. (2012), Visual odometry: Part ii: Matching, robustness, optimization, and applications, *IEEE Robotics & Automation Magazine* 19.2, 78–90.
- Furukawa, Y., Curless, B., Seitz, S. M. ve Szeliski, R. (2010), Towards internet-scale multi-view stereo, *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 1434–1441.
- Furukawa, Y. ve Ponce, J. (2009), Accurate, dense, and robust multiview stereopsis, *IEEE transactions on pattern analysis and machine intelligence* 32.8, 1362–1376.
- Gálvez-López, D. ve Tardos, J. D. (2012), Bags of binary words for fast place recognition in image sequences, *IEEE Transactions on Robotics* 28.5, 1188–1197.
- Girshick, R. (2015), Fast r-cnn, *Proceedings of the IEEE international conference on computer vision*, 1440–1448.
- Girshick, R., Donahue, J., Darrell, T. ve Malik, J. (2014), Rich feature hierarchies for accurate object detection and semantic segmentation, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580–587.
- Harris, C. G., Stephens, M., vd. (1988), A combined corner and edge detector., *Alvey vision conference*. 50. Citeseer, 10–5244.

## KAYNAKLAR DİZİNİ (devam)

- Hartley, R. ve Zisserman, A. (2003), *Multiple view geometry in computer vision*. Cambridge university press.
- Herrera, D. C., Kim, K., Kannala, J., Pulli, K. ve Heikkilä, J. (2014), DT-SLAM: Deferred triangulation for robust SLAM, *2014 2nd International Conference on 3D Vision*. IEEE, 609–616.
- Hinzmann, T., Schönberger, J. L., Pollefeys, M. ve Siegwart, R. (2018), Mapping on the fly: Real-time 3D dense reconstruction, digital surface map and incremental orthomosaic generation for unmanned aerial vehicles, *Field and Service Robotics*. Springer, 383–396.
- Kern, A. (2018), Real-time Photogrammetry using monocular SLAM for Unmanned Aerial Vehicles, M.Sc. thesis, Technische Universität Braunschweig, 85 p.
- Klein, G. ve Murray, D. (2007), Parallel tracking and mapping for small AR workspaces, *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE Computer Society, 1–10.
- Krizhevsky, A., Sutskever, I. ve Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, 1097–1105.
- Kummerle, R., Steder, B., Dornhege, C., Ruhnke, M., Grisetti, G., Stachniss, C. ve Kleiner, A. (2009), On measuring the accuracy of SLAM algorithms, *Autonomous Robots* 27.4, 387.

## KAYNAKLAR DİZİNİ (devam)

- Lin, T. Y., Goyal, P., Girshick, R., He, K. ve Dollár, P. (2017), Focal loss for dense object detection, *Proceedings of the IEEE International Conference on Computer Vision*, 2980–2988.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. ve Berg, A. C. (2016), Ssd: Single shot multibox detector, *European conference on computer vision*. Springer, 21–37.
- Lucas, B. D. ve Kanade, T. (1981), An iterative image registration technique with an application to stereo vision,
- More, J. J. (1978), The Levenberg-Marquardt algorithm: implementation and theory, *Numerical analysis*. Springer, 105–116.
- Moulon, P., Monasse, P., Perrot, R. ve Marlet, R. (2016), Openmvg: Open multiple view geometry, *International Workshop on Reproducible Research in Pattern Recognition*. Springer, 60–74.
- Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F. ve Sayd, P. (2006), Monocular vision based SLAM for mobile robots, *18th International Conference on Pattern Recognition (ICPR'06)*. IEEE, 1027–1031.
- Mur-Artal, R., Montiel, J. M. M. ve Tardos, J. D. (2015), ORB-SLAM: a versatile and accurate monocular SLAM system, *IEEE Transactions on Robotics* 31.5, 1147–1163.

## KAYNAKLAR DİZİNİ (devam)

- Nister, D. (2004), An efficient solution to the five-point relative pose problem, *IEEE transactions on pattern analysis and machine intelligence* 26.6, 756–770.
- Pizzoli, M., Forster, C. ve Scaramuzza, D. (2014), REMODE: Probabilistic, monocular dense reconstruction in real time, *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2609–2616.
- Redmon, J., Divvala, S., Girshick, R. ve Farhadi, A. (2016), You only look once: Unified, real-time object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.
- Redmon, J. ve Farhadi, A. (2017), YOLO9000: better, faster, stronger, *IEEE Society Conference on Computer Vision and Pattern Recognition*, 7263–7271.
- (2018), Yolov3: An incremental improvement, *arXiv preprint arXiv:1804.02767*.
- Ren, S., He, K., Girshick, R. ve Sun, J. (2015), Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in Neural Information Processing Systems*, 91–99.
- Rosten, E. ve Drummond, T. (2005), Fusing points and lines for high performance tracking, *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Ieee, 1508–1515.
- (2006), Machine learning for high-speed corner detection, *European conference on computer vision*. Springer, 430–443.

## KAYNAKLAR DİZİNİ (devam)

Rublee, E., Rabaud, V., Konolige, K. ve Bradski, G. (2011), ORB: An efficient alternative to SIFT or SURF, *2011 International conference on computer vision*. Ieee, 2564–2571.

Sarıbaşı, H., Çevikalp, H. ve Kahvecioğlu, S. (2018), Car detection in images taken from unmanned aerial vehicles, *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 1–4.

Steedly, D., Pal, C. ve Szeliski, R. (2005), Efficiently registering video into panoramic mosaics, *Tenth IEEE International Conference on Computer Vision (ICCV'05) 1*. IEEE, 1300–1307.

Tan, W., Liu, H., Dong, Z., Zhang, G. ve Bao, H. (2013), Robust monocular SLAM in dynamic environments, *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 209–218.

Triggs, B., McLauchlan, P. F., Hartley, R. I. ve Fitzgibbon, A. W. (1999), Bundle adjustment—a modern synthesis, *International workshop on vision algorithms*. Springer, 298–372.

Umeyama, S. (1991), Least-squares estimation of transformation parameters between two point patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 4, 376–380.

Verhoeven, G. (2011), Taking computer vision aloft—archaeological three-dimensional reconstructions from aerial photographs with photoscan, *Archaeological prospection* 18.1, 67–73.

## KAYNAKLAR DİZİNİ (devam)

Vogiatzis, G. ve Hernández, C. (2011), Video-based, real-time multi-view stereo, *Image and Vision Computing* 29.7, 434–441.

Wu, C. (2013), Towards linear-time incremental structure from motion, *2013 International Conference on 3D Vision-3DV 2013*. IEEE, 127–134.

Yahyanejad, S. ve Rinner, B. (2015), A fast and mobile system for registration of low-altitude visual and thermal aerial images using multiple small-scale UAVs, *ISPRS Journal of Photogrammetry and Remote Sensing* 104, 189–202.