Izgara Tabanlı Haritalama Kullanımı ile Robotik Denetleme Yöntemi

Cansu Akdeniz

YÜKSEK LİSANS TEZİ

Elektrik Elektronik Mühendisliği Anabilim Dalı

Nisan 2022

Robotic Inspection Method by Using Grid-Based Mapping

Cansu Akdeniz

MASTER OF SCIENCE THESIS

Department of Electric Electronic Engineering

April 2022

Izgara Tabanlı Haritalama Kullanımı ile Robotik Denetleme Yöntemi

Cansu Akdeniz

Eskişehir Osmangazi Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Elektrik Elektronik Mühendisliği Anabilim Dalı
Kontrol ve Kumanda Sistemleri Bilim Dalında
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır.

Danışman: Doç. Dr. Metin ÖZKAN

Nisan 2022

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Metin Özkan danışmanlığında hazırlamış olduğum "**Izgara Tabanlı Haritalama Kullanımı ile Robotik Denetleme Yöntemi**" başlıklı YÜKSEK LİSANS 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. 21/04/2022

Cansu Akdeniz

İmza

ÖZET

Endüstriyel üretim süreçlerinde, üretilen parçaların ve ürünlerin kalite güvencesi, kalitenin gerçekleşme durumuna yönelik bilgi gerektirir. Bu bilgi, kalite denetlemesi ile sağlanabilir. Uygun sensör, robot ve teknikler kullanılarak, kalite denetleme süreci otomatik hale getirilebilir. Böylece, denetim süreci sürdürülebilir, güvenilir ve kayıt altına alınabilir olacaktır.

Bu tez çalışmasının amacı, üretim süreci esnasında çok sayıda parçanın birbirine monte edilmesi ile elde edilen yapıların, kalite denetimlerini otomatik olarak gerçekleştirmek üzere yöntem geliştirmektedir. Bu yapıları oluşturan parçalardan, dışarıdan görülebilen ve montajı yapılmamış olanların tespit edilmesi beklenmektedir. Bu amaca yönelik olarak, ızgara tabanlı haritalama kullanımıyla, robotik denetleme yöntemi geliştirilmesi ve benzetim ortamında gösterimi gerçekleştirilmektedir. Önerilen yöntem üç kısımdan oluşmaktadır. Öncelikle, yapıyı oluşturan denetimi yapılacak parçalar ızgara tabanlı haritada etiketlenir. Buna inanç haritası denilmektedir. Daha sonra, ortamda sabitlenmiş ve robot kol ucuna monte edilmiş derinlik kameralarından alınan nokta bulutları ile inanç haritası üzerinde doluluk haritalaması yapılır. Son aşamada, inanç harita etiketleri ile ızgara doluluk bilgilerinin ilişkileri değerlendirilmektedir. Bu değerlendirme neticesinde, yapıda montajı yapılmamış parçaların tespiti gerçekleştirilmektedir. Benzetim ortamında yapılan deneysel çalışmalarla yöntemin etkinliği gösterilmiştir.

Anahtar Kelimeler: Robotik denetleme, ızgara tabanlı haritalama, derinlik sensörü

SUMMARY

In industrial production processes, the quality assurance of manufactured parts and products requires knowledge of the realization of quality. This information can be provided by quality control. By using appropriate sensors, robots and techniques, the quality inspection process can be automated. Thus, the audit process will be sustainable, reliable, and recordable.

The aim of this thesis study is to develop a method to automatically perform quality inspections of structures obtained by assembling many parts together during the production process. It is expected that the parts that make up these structures, which can be seen from the outside and not assembled, will be determined. For this purpose, robotic inspection method is developed and demonstrated in a simulation environment by using grid-based mapping. The proposed method consists of three parts. First, the parts to be inspected that make up the structure are labelled on the grid-based map. This is called a belief map. Then, occupancy mapping is performed on the belief map with point clouds taken from depth cameras fixed in the environment and mounted on the end of the robot arm. In the final stage, the relationships between belief map labels and grid occupancy information are evaluated. As a result of this evaluation, the parts that have not been assembled in the structure are determined. The effectiveness of the method has been demonstrated by experimental studies carried out in the simulation environment.

Keywords: Robotic inspection, grid-based mapping, depth sensor

İÇİNDEKİLER

Sayfa

ÖZET ..... **vi**

SUMMARY ..... **vii**

TEŞEKKÜR ..... **viii**

İÇİNDEKİLER ..... **ix**

ŞEKİLLER DİZİNİ ..... **xi**

ÇİZELGELER DİZİNİ ..... **xiii**

SİMGELER VE KISALTMALAR DİZİNİ ..... **xiv**

1. GİRİŞ VE AMAÇ **1**

2. LİTERATÜR ARAŞTIRMASI **4**

2.1. Robotik Denetleme 4

2.2. Izgara Tabanlı Haritalama Yaklaşımları 7

2.2.1. Haritalama yapıları 7

2.2.2. Haritalama Teknikleri 10

2.3. Haritalamada Kullanılan Sensörler 12

3. MATERYAL VE YÖNTEM **14**

3.1. Deneysel Çalışma Ortamının Oluşturulması 14

3.1.1. Robot İşletim Sistemi (ROS) 15

3.1.2. Benzetim Yazılımı (Gazebo) 16

3.1.3. Derinlik Sensörleri 17

3.1.4. Endüstriyel Robot Kolları 19

3.1.5. 3B Doluluk Izgarası Haritalaması için Yazılım Kütüphanesi (OctoMap) 21

3.2. Izgara Tabanlı Haritalamaya Dayalı Robotik Denetleme Yöntemi 25

İÇİNDEKİLER (devamı)

Sayfa

3.2.1. İnanç Haritası (Belief Map) Yaratma (I. AŞAMA) ……...……...…..... 27
3.2.2. Doluluk Haritası (Occupancy Map) Yaratma (II. AŞAMA) ……..... 28
3.2.3. Değerlendirme (III. AŞAMA) …….……. 29

4. UYGULAMA ……. 31
4.1 Deneysel Ortamın Oluşturulması …….……..... 31
4.2 Denetimi Yapılacak Yapılar …….. 33
4.3 Derinlik Kamerası …… 35
4.4 İnanç Haritası Oluşturulması ...……..... 41
4.5 Doluluk Haritası Oluşturulması …...……..... 50
4.6 Denetleme ……. 52

5. BULGULAR VE TARTIŞMA ……..……. 58

6. SONUÇ VE ÖNERİLER ……..……. 65

KAYNAKLAR DİZİNİ ……..……. 66

ŞEKİLLER DİZİNİ

Şekil **Sayfa**

2.1 Serbest (gölgeli beyaz) ve dolu (siyah) hücreleri depolayan bir oktree örneği 9
3.1 Gazebo' da Oluşturulan Ortam Görüntüsü 15
3.2 Yapılandırılmış Işık Teknolojisi Kullanan 3B Sensörler 18
3.3 UR10e Robot Kolu 20
3.4 En yaygın oktree ve düğüm sınıflarının UML diyagramı 23
3.5 Izgara Tabanlı Haritalamaya Dayalı Robotik Denetleme Yönteminin Uygulama Aşamaları 26
3.6 İnanç Haritasının Oluşturulması 27
3.7 Doluluk Haritasının Oluşturulması 28
4.1 Benzetim Ortamında Koordinat Sistemleri 31
4.2 Gazebo Benzetim Programında Test Ortamları 33
4.3 Üçgen Kirişli Yapı (Truss) 34
4.4 Elektrikli Bisiklet İskeleti 34
4.5 Kinect XBOX 360 35
4.6 Kamera üzerindeki koordinat sistemleri: (üst) Ön çapraz görünüm, (orta) önden görünüm, (alt) üstten görünüm 36
4.7 Ortam-1' e ait Kamera Görüntüleri 38
4.8 Ortam-1'deki Kameralardan Elde Edilen Nokta Bulutlarının Rviz Görüntüsü 39
4.9 Ortam-2' e ait Kamera Görüntüleri 40
4.10 Ortam-2'deki Kameralardan Elde Edilen Nokta Bulutlarının Rviz Görüntüsü 41
4.11 Ortamlarda Kullanılan Yapılar 42
4.12 İnanç Haritası Oluşturulması için İşlem Basamakları 43
4.13 Seçilen Dört Farklı Parçanın CloudCompare Görüntüsü 44
4.14 Ortamlarda Kullanılan Yapılara Ait Parçaların Üçgen Örgü (Mesh) Modellerinin Octovis Görüntüsü 46
4.15 Yapıların Octovis Görüntüleri 49

ŞEKİLLER DİZİNİ (devamı)

Şekil **Sayfa**

4.16 Haritalama Akış Diyagramı…… 50
4.17 Ortam-2'de Kullanılan Kamera K1'den Alınan Nokta Bulutlarının Rviz Görüntüsü …… 51
4.18 Ortam-2'de Kullanılan Kamera K1'den Alınan Nokta Bulutlarıyla Oluşturulmuş Doluluk Haritası Görüntüsü …… 52
4.19 Denetleme İşlem Basamakları …… 53
4.20 Denetlemesi Yapılacak Yapının **(a)** İnanç Haritası ve **(b)** 4 Kamera için Oluşturulmuş Doluluk Haritası …….. 54
4.21 İnanç Haritası ve Dört Kamera için Oluşturulan Doluluk Haritasının Birlikte Görselleştirilmesi …… 55
5.1 Eksik Parçaların Birleştirilmiş Haritada Olması Gerektiği Yerler ……..… 59
5.2 İnanç Haritası ve Bir Kamera için Oluşturulan Doluluk Haritasının Birlikte Görselleştirilmesi …… 60
5.3 Bir Kamera için Eksik Parçaların Birleştirilmiş Haritada Olması Gerektiği Yerler …….. 63

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

4.1. Kameranın Teknik Özellikleri ……..……..... 37
4.2. İnanç Haritası Oluşturma İşlem Süreleri ……..…….. 47
4.3. Dört Kamera için Denetim Sonuçları ……..…….. 57
5.1. Bir Kamera için Denetim Sonuçları ……..……. 62

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
2B	İki Boyutlu
3B	Üç Boyutlu
RGB	Red-Green-Blue (Kırmızı-Yeşil-Mavi)
RGB-D	Red-Green-Blue-Depth(Kırmızı-Yeşil-Mavi-Derinlik)
ROS	Robot İşletim Sistemi
ToF	Time-of-Flight
CMOS	Complementary Metal OxideSemiconductor
UML	UnifiedModeling Language
STL	Stereolitografi Dosya Uzantısı
PCD	Point Cloud Data
VRML	Sanal Gerçeklik Modelleme Dili
XML	Genişletilebilir İşaretleme Dili
SDF	Spatial Data Format

1. GİRİŞ VE AMAÇ

Robotik, insanların hayatını kolaylaştırmak için insan müdahalesi olmadan otomatik olarak çalışabilen makinelerin geliştirilmesi ve insanların yavaş veya zayıf olduğu yerlerde çalışmaya devam etmesi için yapılan birçok çalışmanın ana konusu olmuştur. Robotlar, üretim süreçlerinde, sürdürülebilir kalite, hızlı ve düşük maliyetli üretim gibi önemli avantajlar sağlamaktadır. Robotik sistemlerde kullanılan, günümüzde KUKA, Universal Robots, FANUC gibi çok sayıda endüstriyel robot markası bulunmaktadır. Bu robotlar, geliştirilen sistemlerde kalite kontrolünden, kaynak işleri, montajdan lojistiğe kadar çok farklı uygulamada kullanılmaktadır.

Teknolojinin gelişiminin de etkisiyle robotlar hayatın birçok noktasında insanların en büyük destekçisidir. Robotlar çok farklı işlevleri yerine getirmek için kullanılabilir çünkü robotların verilen görevleri yerine getirmesi programlanmasıyla doğrudan bağlantılıdır. Kullanılan yere ve kullanılacak amaca göre şekillendirilebilirler. Robotların gelişmesiyle ve yaşamda yer edinmesiyle alınan riskler büyük ölçüde azaltılmıştır. Bu riskleri en aza indirmek için literatürde yapılmış birçok çalışma vardır. Fakat, son zamanlarda en fazla üzerinde durulan konu endüstriyel robotların üretimde ve denetimde aktif olarak kullanılmasını hedefleyen çalışmalardır. Robotların üretimde ve denetimde kullanılması hedeflenirken riski azaltmanın yansıra, kaliteyi arttırma ve endüstride insan gücünü azaltmada amaçlanıyor. Literatürde bununla ilgili birçok çalışma vardır.

Son zamanlarda endüstride robot kullanımı oldukça yaygındır. İnsanların yaptığı işlerden ayrı olarak robotlar, fiziksel olarak yorucu, bilek gücüne dayanan ve insan hayatı için riskli olan işleri yapmaktadırlar. Özellikle kalitede ve denetimde insan hatasını azaltmak için robot kullanımına büyük önem verilmiştir. Endüstride robot kullanımı hem robotların verimliliği hem de rekabet gücü açısından büyük fayda sağlamaktadır. Ayrıca, robotlar görevi tamamlarken, işgücü kapasitesini de arttırmaktadırlar. Fakat robotik kullanımında ortamdaki diğer etkenler de göz önünde bulundurulmalıdır ve robotlar bu etkenler göz önünde bulundurularak programlanmalıdır. Robotlardan kalite ve denetimde hatasız

çalışmaları ve hedeflerini en verimli şekilde gerçekleştirilmesi beklenir. Bu planlama yapılırken robotlar sensörlerle desteklenir. Ayrıca, robotlara gerekli yazılımlarla insansı özelliklerde kazandırılmaya çalışılır. Kullanılan sensör sayısı ve çeşidi yapılacak çalışmaya göre değişiklik gösterirken seçilen sensörler çoğunlukla robotların hedeflenen görevi tam olarak yerine getirmesi planlanarak seçilir. Denetimde ve kalitede robot kullanımı insan çalışmasının yerini almaz, ancak çalışanların verimliliğini arttırmak, işyerlerinde yaralanma riskini azaltmak, üretimde insanların erişemeyeceği ya da fark edemeyeceği insan hatalarını en aza indirmek için bu alanlarda büyük fayda sağlar.

Robotların istenilen görevi yerine getirmesinde en önemli faktörlerden biri sensörlerdir. Sensörler robota hassas algılama becerisi kazandırıyor. Sensör teknolojisi geliştikçe, denetim uygulamaları daha çeşitli hale gelir. Şu anda robotik denetleme için kullanılan algılama sistemleri, kuvvet sensörleri ve çift dokunma sensörlerinin yanı sıra 2B (iki boyutlu) görüntü ve ışık çizgilerini içerir. Ayrıca, son zamanlarda görüntü işleme aracılığıyla da hassas algılama yapmak mümkündür. Hedeflenen işi yapmak için uygun sensör kullanımı denetimde en önemli faktördür çünkü denetimi hassas olarak gerçekleştirmek uygun sensör kullanımıyla mümkündür. Son zamanlarda gelişen sensör teknolojisi sayesinde, robotik sistem üretimde, taşımada, montajda vb. birçok alandaki sorunları robotik denetim ile çözer. Denetim faaliyetlerinde robotların kullanılmasının temel nedeni kalite kontroldür. Robotik denetim robotların tutarlılığına yardımcı olurken daha fazla öngörülebilirlik sağlar. Tutarlılık ve öngörülebilirlik, üretim ve süreç planlamasına yardımcı olur.

Üretimin temel bileşenlerinden bir tanesi olan kalite, günümüzde önemi giderek vurgulanan başlıklardan biri olmuştur. Kalite kontrolünün otomasyonu ve üretimle bütünleştirilmesi canlı bir araştırma ve uygulama konusu haline gelmiştir. Bu çalışmanın amacı da üretim süreci esnasında çok sayıda parçanın birbirine monte edilmesi ile elde edilen yapıların, kalite denetimlerini otomatik şekilde hayata geçirmek üzere yeni bir yöntem geliştirmek olarak belirlenmiştir. Bu çalışmada yapıları oluşturan parçalardan, dışarıdan görülebilen ve montajı yapılmamış olanların tespit edilmesi beklenmektedir. Bu hedef doğrultusunda, ızgara tabanlı haritalama kullanımıyla, robotik denetleme yönteminin kavramsal ispatı gerçekleştirilmektedir. Bu çalışmada ızgara tabanlı haritalama yapısı olarak Octomap kullanıldı. Octomap kütüphanesi, eşleme algoritmaları sunan bir sekizli ağaç (Octree) tabanlı 3B (üç boyutlu) haritalama yaklaşımıdır. Octomap, bellek tüketimini

minimumda tutarken tam ve boş alanın verimli ve olasılıklı güncellemelerine izin verir. Bu nedenle son zamanlarda en çok tercih edilen haritalama yapılarından biridir. Octomap kütüphanesi voksel yapısına dayalı haritalama yapar. Açık kaynaklı yazılım olduğu için çeşitli eklemelere ve düzenlemelere uygundur. Çalışmada, kütüphaneye çeşitli eklemeler yapılarak ve Octomap veri yapısına dayalı yeni bir algoritma geliştirilerek robotik denetleme yapılması amaçlanmıştır. Bu algoritmada, ortama yerleştirilen çok parçalı yapıyı oluşturan parçaların üçgen örgü (mesh) modelleri kullanılmakta ve renklendirme seçeneğiyle her bir parça farklı renkle işaretlendi ve inanç haritası (Belief map) oluşturuldu. Daha sonra, sensör verilerinin kullanımı ile haritadaki voksellerin doluluk olasılıkları oluşturulmaktadır. Her bir parçanın etiketine sahip voksellerin dolu olanlarının voksel sayısına oranı parçanın bulunup bulunmadığı bilgisini sağlamaktadır. Böylece, robotik denetleme ile hangi parçaların yapı üzerinde monte edildiği saptanmaktadır.

Bölüm 2 ‘de bu tez konusu ile ilişkili literatürdeki çalışmaların özeti verilmektedir. Bölüm 3’ de bu çalışmayı yaparken kullanılan materyal ve yöntemler ayrıntılı olarak açıklanmıştır. Önerilen yöntem benzetim ortamında yapılan uygulama ile test edilmekte ve değerlendirilerek elde edilen bulgular Bölüm 4’te verilmektedir. Son olarak, bu çalışmanın genel olarak değerlendirilmesi sonuç ve öneriler başlığı altında Bölüm 5’te verilmektedir.

2. LİTERATÜR ARAŞTIRMASI

Literatürde, robotik sistemlerle ilgili çok sayıda çalışma bulunmaktadır. Bunlardan bir kısmı, üretim süreçlerinde kalite kontrolü gerçekleştirmek üzere robotik denetleme sistemlerini kapsamaktadır. Robotik denetleme çalışmalarına yönelik yayınlar, uygun sensörler oluşturulan ortamları ve geliştirilen denetleme yöntemlerin sunumunu yapmaktadır. Bu kısımda, tez çalışması ile ilişkili literatürdeki çalışmaların özeti verilmektedir. Bu kapsamda, robotik denetleme yöntemleri, denetleme amaçlı kullanılan sensörler ve ızgara tabanlı haritalama çalışmaları verilmektedir.

2.1. Robotik Denetleme

Robotların birçok görev için kullanılması amaçlansa da günümüzde daha çok insanların çalışmasının zor olduğu çalışma ortamlarında kullanılırlar. Özellikle, robotların endüstride yaygın kullanılmasıyla robotik denetleme, endüstriyel sektörün önemli bir parçası olmuştur. Robotik denetleme, yol planlaması, hareket kontrolü ve görev planlaması gibi birçok farklı yönden ele alınabilir. Denetimi otomatikleştirmek için robotların kullanımı maliyeti düşürür ve üretkenliği arttırır. Bu yüzden denetim görevlerinin robotik sistemlerle sağlanması endüstride önemli katkılar sağlar. Literatürde, robotik kolları kullanılarak endüstriyel alanlarda denetleme görevlerine yönelik çeşitli çalışmalar bulunmaktadır.

Al Khawli vd. (2020), emek yoğun çalışma gerektiren üretim süreçlerinde, kalite denetimini otomatikleştirmeyi hedeflemişlerdir ve lazer profil sensörünün endüstriyel bir robot kola monte etmişlerdir. Amaçları, 6 serbestlik dereceli robot kola monte edilmiş olan bir lazer profil sensörden sağlanan ölçümleri kaydetmekti. Bu çalışmada kayıt hem takım konumlandırma ve hizalamada yüksek düzeyde doğruluk gerektiren çeşitli otomatikleştirilmiş üretim süreçlerine hem de esneklik ve doğruluk gerektiren kalite kontrol sistemleri için önemli bir adım olan 6 serbestlik dereceli kalibrasyona dayanmaktadır. Araştırma, keskin bir nesne kullanılarak yapılan kalibrasyon ve düzlemsel kısıtlamalar olmak üzere iki kalibrasyon prosedürünü içermektedir. Yapılan bu çalışmada hem yinelemeli hem de optimizasyon çözücülerinden tahmin edilen kalibrasyon prosedürlerinin çözümü

kapsamlı bir şekilde tartışılmaktadır. Ek gürültü seviyelerine sahip iki prosedür için sanal veriler üreten bir simülasyon platformu uygulayarak, altı boyutlu pozlar tahmin edilerek temel gerçekle karşılaştırılır. Sonuç olarak, denetim doğruluğunu araştırmak için bir robotik denetim sistemine kalibrasyon prosedürleri uygulanmıştır. Simülasyon ve deneysel sonuçlar benzer eğilimleri göstermekte ve keskin bir alet kullanılarak kalibrasyon prosedürü ile daha yüksek bir inceleme doğruluğunun elde edilebileceğini göstermektedir.

Otonom çok robotlu optik denetim sistemleri, süreç içi izleme, kalite kontrolü ve hat içi ölçümler elde etmek için giderek daha fazla uygulanmaktadır. Statik ve dinamik ortamlar için yol planlama ve robotik koordinasyon için çok sayıda yöntem geliştirilmiş ve farklı alanlara uygulanmıştır. Bununla birlikte, bu yaklaşımlar, hat içi optimizasyonun hızlı hesaplama gereksinimleri, robotik uç eyleyici oryantasyonlarındaki benzersiz özellikler ve karmaşık büyük ölçekli serbest biçimli ürün yüzeyleri nedeniyle otonom çoklu robot optik inceleme sistemi için çalışmayabilir. Bir çalışmada, çoklu robot denetiminin koordineli hareket planlaması için yeni bir görev tahsis metodolojisi önermektedir (Liu vd., 2021). Bu çalışmada iki konu üzerinde durulmuştur. İlk olarak, robotlar arasında verimli ve dengeli bir ölçüm ataması elde etmek için yerel bir sağlam denetim görevi tahsisi önerilmiştir. İkinci olarak çarpışmasız yol planlaması ve koordineli hareket planlaması, robotik koordinat uzayında dinamik arama ve çatışan robotlarda yerel yolların bozulması yoluyla geliştirilir. Önerilen yaklaşımın robotlar ve ortamlar arasındaki çarpışma riskini azaltabileceğini, robotlar arasındaki çatışmaları çözebileceğini ve denetim döngüsü süresini önemli ölçüde ve tutarlı bir şekilde azaltabileceğini göstermektedir.

Çevrenin yeterince yapılandırılmamış ve öngörülemez olduğu demiryolu vagonlarının bakımında, küçük bileşenlerin tespiti ve lokalizasyonu zordur. Otonom sistemlerin tanıtımında önemli bir güçlük olmaya devam etmektedir. Yapılan bir çalışmada endüstriyel robotların demiryolu kuplörlerinin elektrik başlığını denetlemesine izin veren bir sistemin tasarımı ve değerlendirilmesi sunulmaktadır (Vithanage vd., 2019). Çoklu sensörler ve aktif uyumlu kol ucu takımlarından oluşan geliştirilmiş sistem, gürültü ve belirsizliklere karşı dayanıklıdır. Sistemin başarılı bir biçimde doğrulanması, hem çalışma ortamındaki gerçek bir kuplör hem de ticari sınıf bir endüstriyel robot ve bir sahte kuplör içeren bir deneysel kurulum kullanılarak elde edilir.

Kompozit havacılık parçalarının tahribatsız muayenesi (NDT) için denetimi hızlandırma gereksinimi, birçok üretici için ortaktır. Endüstride karmaşık kavisli yüzeylerin yaygınlığı, bu denetimlerde NDT problarının yerleştirilmesi için altı eksenli robotların kullanımı için önemli bir motivasyon sağlar. Bazı firmaların ortak yürüttüğü bir çalışmayla protip bir robotik NDK sistemi üretildi (Mineo, vd., 2015). Prototip sistem, büyük zaman tasarrufu ile karmaşık geometrili kompozit bileşenleri inceleme yeteneğine sahiptir. İki altı eksenli robotik kol, aşamalı dizi ultrasonik test (PAUT) problarını taşıyan uç eyleyicileri dağıtır. Parça verilerinin ilk yüklenmesinden, parçanın kapsamlı taranmasından veri analizine kadar robotik incelemenin tüm yönlerini kontrol etmek için kullanımı kolay bir grafik kullanıcı arayüzü (GUI) geliştirilmiştir. PAUT ayarlarının optimizasyonu için gerçek zamanlı B-taraması ve ilgi çekici noktalara geri dönmeyi sağlamak için robotik manipülatörlerin harici kontrolü gibi birçok benzersiz özellik, denetim sürecinin kullanışlılığını artırır.

Başka bir çalışmada, yeraltı kentsel gaz boru hatlarının boru içi denetimi için robotik bir sistem sunulmuştur (Choi ve Ryew, 2002). Bu robot, boru hattı ağlarının görsel ve Tahribatsız Muayenesi (NDT) için mobil bir platform olarak kullanılmak üzere geliştirilmiştir. Robot sisteminin önünde ve arkasında sırasıyla iki aktif sürüş aracı yer alarak tasarlanmıştır. Kontrol modülü gibi pasif modüller ve diğer isteğe bağlı modüller, aktif araçlar arasında bağlantılıdır. Esnek tekerlekli ayak mekanizmaları, uyum kontrollü direksiyon mekanizması gibi diğerlerinden üstün birçok karakteristik özelliğe sahiptir. Direksiyon mekanizması sayesinde robotun sürüş yönü boyunca yuvarlanması engellenir ve uyumluluğunun kontrol edilmesi sağlanır. Bu özellikler robota, boru hattı ağlarının karmaşık yapılarında son derece kısıtlı alan içinde mükemmel hareket kabiliyeti sağlar.

Otomatikleştirilmiş bir görüş sistemi ve özellikle geliştirilmiş bir grafik kullanıcı ara yüzü ile robotik bir çalışma hücresinin entegre edilmesiyle elde edilen, bileşenlerin özel görsel denetimli ultrasonik denetimini gerçekleştirmek için başka bir yaklaşım sunulmuştur (Khan vd., 2020). Hareketten yapıyı temel alan görüş sistemi, parçaların 3B modellerinin oluşturulmasına olanak tanır. Ayrıca bu çalışma, optimum görüntü yakalama için dört farklı takım yolunu karşılaştırır. Ortaya çıkan optimum 3B modeller, kullanıcının ultrasonik inceleme için herhangi bir ilgi noktasını seçmesini sağlamak için robotik inceleme hücresinin sanal ikiz ortamında kullanılır. Bu, genellikle çok zaman alan bir aşama olan bir

parça üzerindeki belirli konumları değerlendirmek için çevrimdışı robot yol planlama ve parça yönlendirme ihtiyacını ortadan kaldırır.

Tünellerin otonom denetimi içinde bazı çalışmalar yapılmış ve bir çalışmada robotik araç geliştirilmiştir (Loupos vd., 2018). Bu robotik araç, robotlu bir üretim sepetli vinci, yüksek hassasiyetli bir robotik kol, gelişmiş bilgisayar görüş sistemleri, bir 3B lazer tarayıcı ve bir ultrasonik sensörden oluşur. Tünellerin otonom denetimi, robotik aracın ve bilgisayarlı görme alt sisteminin gelişmiş yeteneklerini gerektirir. Yeraltı boşluklarında ve uzun doğrusal yollarda robot lokalizasyonu, beş tonluk bir vinç aracına monte edilmiş bir robotik ucun mm hassas konumlandırmasının yanı sıra zorlu bir görevdir. Ayrıca, inceleme sürecini destekleyen 2B ve 3B görme görevleri, zayıf ve değişken aydınlatma koşulları, düşük dokulu kaplama yüzeyleri ve yüksek doğruluk ihtiyacı ile başa çıkmalıdır. Bu çalışmada geliştirilen robotik araç denenerek test edilmiş ve verimli sonuçlar elde edilmiştir.

2.2. Izgara Tabanlı Haritalama Yaklaşımları

2.2.1. Haritalama yapıları

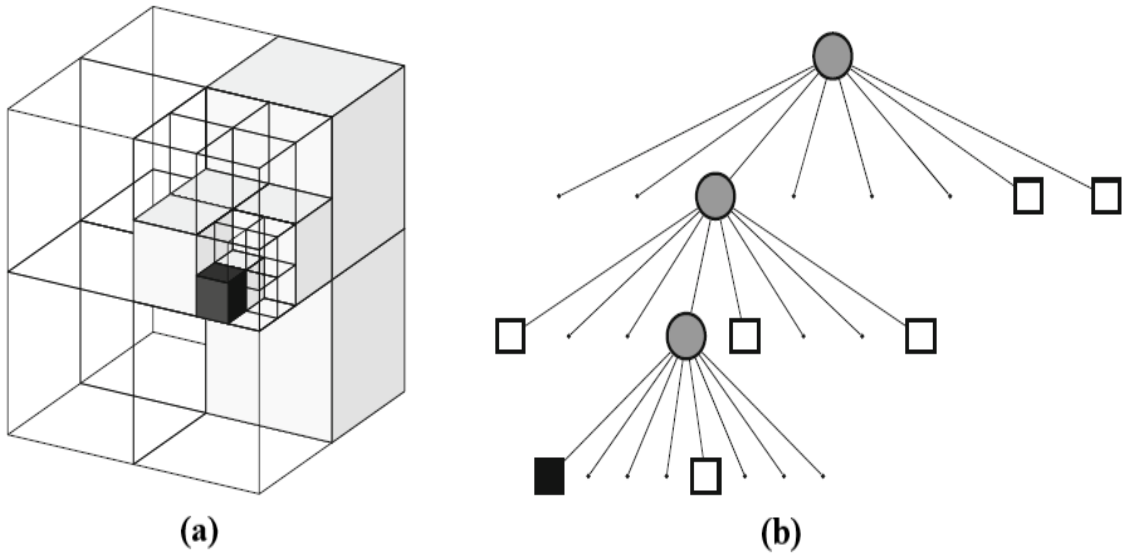
Bu tez çalışmasında, robotik denetleme için önerilen yöntemin temelini, ızgara tabanlı haritalama oluşturmaktadır. Izgara tabanlı haritalamaya yönelik yöntemler ve yazılım araçları bulunmaktadır.

Izgara tabanlı haritalamada, ortam iki ya da üç boyutlu ızgara adı verilen kare ya da küplerle temsil edilir. Ancak, haritalama uzayını sabit boyutlu kare veya küpler yerine, sekizli ağaç (octree) ile temsil etmek işlem ve depolama açısından daha verimlidir. Bu nedenle, sekizli ağaç yapısı ızgara tabanlı haritalamada yaygın olarak kullanılmaktadır. Haritalamada, ızgaralarla temsil edilen uzayda ızgaraların temsil ettiği alanın bir nesneye ait olması durumunda dolu olarak tanımlanmaktadır. Böylece, ızgaraların dolu ya da boş olması robotun hareket edebileceği alanları tanımlaması nedeniyle kullanılmaktadır. Harita çıkarımı, olasılıksal doluluk bilgisi kullanılarak gerçekleştirilir. Hücreler, robotun navigasyonu için çok önemli olan işgal (doluluk, boşluk) bilgilerini içerir. Robotlar hücre doluluk bilgilerini kullanarak çevrelerini haritalayabilir veya engellerle karşılaşmadan

hedeflerine ulaşan navigasyon algoritmaları uygulayabilir. Bir bilgisayarda uygulanması ve işlenmesi nispeten kolaydır ve zor problemleri çözmek için oldukça faydalıdır. Fakat bazı dezavantajlara sahiptir. Özellikle robot navigasyonunda ızgara tabanlı haritalarda kullanıldıklarında özellikle büyük haritalarda hafıza gereksinimi ve zaman yönünden bazı problemlere neden olabilmektedir. Bu bölümde literatürde var olan ızgara tabanlı haritalama yaklaşımlarından kısaca bahsedilecektir.

3B modeller, birçok robotik uygulamanın önemli bir parçasıdır. Ortamın önceden bilinmediği veya ortamın yalnızca bir kısmının bilindiği uygulamalarda, 3B modelin bilinmeyen alanı verimli bir şekilde ele alabilmesi önemlidir. Yol planlama, keşif ve yeniden yapılanmanın tümü bu kategoriye girer. 3B ortamı modellemek için birçok farklı yöntem kullanılabilir: nokta bulutları, yükseklik haritaları, çok katmanlı yüzey haritaları, sekizler ve işaretli mesafe alanları. Bu yöntemler sayesinde tüm dolu ve boş alanlar temsil edilebilir. Ama yalnızca sekizler ve işaretli mesafe alanları bilinmeyen alanı temsil eder. Ayrıca, nokta bulutları, yükseklik haritaları ve çok seviyeli yüzey haritaları, rastgele 3B ortamları temsil edemez.

OctoMap, sekizli ağaç (octree) yapısına dayanan en popüler haritalama çerçevelerinden biridir (Hornung vd., 2012). Octomap kütüphanesi ROS içerisinde bulunan, özellikle C++ için veri yapıları ve eşleme algoritmaları sunan bir sekizli ağaç tabanlı 3B haritalama yaklaşımıdır. Sekizli ağaç 3B uzamsal alt bölümlere yönelik hiyerarşik bir veri yapısıdır (Meagher, 1982; Wilhelms ve Van Gelder 1992). Bir sekizli ağaç içindeki her düğüm, genellikle bir voksel olarak adlandırılan kübik bir hacimde bulunan alanı temsil eder. Bu hacim, Şekil 2.1' de gösterildiği gibi, belirli bir minimum voksel boyutuna ulaşılana kadar yinelemeli olarak sekiz alt hacme bölünür. Minimum voksel boyutu, sekizli ağaç çözünürlüğünü belirler. Bir sekizli ağaç hiyerarşik bir veri yapısı olduğundan, eğer iç kısım daha kaba bir alt bölüm elde etmek için ağaç herhangi bir seviyede kesilebilir. OctoMap yapısı, haritalamaya odaklanmayan araştırmacılar için hazır, güvenilir ve verimli bir 3B haritalama çerçevesi ve hazır haritalama çözümü sağlar.



Şekil 2.1 Serbest (gölgeli beyaz) ve dolu (siyah) hücreleri depolayan bir oktree örneği. **(a)** Hacimsel model, **(b)**Temsili ağaç modeli (Hornung vd., 2012).

OctoMap yapısı UFOMap denilen bir uzantı sunan çalışma yapılmıştır (Duberg ve Jensfelt, 2020). UFOMap yapısı da Octomap gibi sekizli ağaç veri yapısını kullanılır. Octomap yapısı bazı uzantılar eklenerek haritadaki üç durumun da yani bilinmeyen, serbest ve dolu olma durumu, açık olarak temsil edilmesiyle ilgili bir çalışmadır. Octomap yapısından farklı olarak bilinmeyen alanı da temsil eder. Octomap yapısı bilinmeyen alan modellenmez sadece dolu ya da boş olma durumu belirlenir. Fakat bu çalışma bilinmeyen alanı da temsil etmemizi sağlar. Ayrıca, bu çalışma bellek açısından Octomap yapısından daha verimlidir.

Octomap yapısı bazı iyileştirmeler sunan başka bir çalışma daha yapılmıştır. Bu çalışmaya SkiMap denilmiştir. Octomap yapısında, haritalama yapılırken alan büyüdükçe hafıza ve zaman açısından bazı sıkıntılar meydana gelir. Bu yüzden hafıza ve zaman yönünden iyileştirmeler yapılarak daha geniş alanların daha kısa sürede haritalanması hedeflenir ve bununla ilgili çeşitli çalışmalar yapılmıştır. Özellikle robot navigasyonun da haritalama alanında, haritanın doğruluğunu sürekli olarak iyileştirmek için sensörlerden optimize edilmiş pozların gerçek zamanlı olarak alınıp haritaya entegre edilmesi için bu çalışma yapılmıştır (Gregorio ve Stefano, 2017). Octomap yapısından farklı olarak vokselleri gruplandırmaya dayalı bir yöntem sunulmuştur.

2.2.2. Haritalama teknikleri

Işın İzleme (Ray Tracing) ve Işın Dökümü (Ray Casting)

Octomap ve Octomap' i baz alarak yapılan çalışmalar genel olarak, düğümlerin (nodeların) dolu mu boş mu olduğunu ışın izleme yöntemi ile belirler. Işın izleme, nokta bulutu sensör ölçümlerini entegre etmek için kullanılır. Bir nokta bulutunu entegre ederken, işgal edilmesi gereken noktaların işgal edilmesi ve sensör orijini ile nokta bulutlarının her noktası arasındaki tüm noktaların serbest bırakılması önemlidir.

Işın izleme yönteminin en temel formu ışın dökümü yöntemidir. Işın dökümü yüksek kaliteli görüntüler oluşturmak için tercih edilen bir algoritmadır. Fakat hesaplama yönünden oldukça dezavantajlıdır. Bu dezavantajı en aza indirmek için bazı çalışmalar yapılmıştır ve iki strateji geliştirilmiştir. Hızlı ve basit voksel geçiş algoritması kullanılmıştır (Amanatides J. ve Woo A. 1987). Birden fazla vokselde bulunan nesnelerle çoklu ışın kesişimlerinin ortadan kaldırılması hedeflenir. Bu önerilen algoritmayla nesneler için çoklu kesişim testleri ortadan kaldırılmıştır.

Semantic Map

Robotik alanında, haritalamadaki genel sorun ortamda bulunan katı cisimlerden ya da ortam haritasının değişken olmasından kaynaklanır. Bu noktada robotun bulunduğu ortamı öğrenebilmesi ve ortamdaki konumunu belirleyebilmesi çok önemlidir. Semantik haritalama sayesinde ortamdaki katı cisimler haritada işaretlenebilir. Literatürde Octomap' i Semantik OctoMap gösterimine genişleten ve derinlik veri setini kullanarak GPS (Küresel Konumlama Sistemi) haritalama performansı ile karşılaştıran bir çalışmada vardır. Bu çalışmada, sınıflandırılmış nokta bulutlarını OctoMap ile ilişkilendirerek bir etiketleme işlemi yapılır ve çıkarılan harita herhangi bir çözünürlükle çıkarılabilir (Jadidi vd. 2017). Fakat bu çalışmada komşu konumların aynı semantik etiketini paylaşması muhtemeldir. Bu durum haritada bozulmalara yol açar.

Recurrent-Octomap, uzun vadeli üç boyutlu (3-D) Lidar verilerinden öğrenilen yeni bir semantik haritalama yaklaşımı sunar. Mevcut semantik haritalama yaklaşımlarının çoğu, anlamsal haritaların üç boyutlu iyileştirilmesinden (yani anlamsal gözlemlerin kullanılması) ziyade, tek çerçevelerin anlamsal anlayışını geliştirmeye odaklanır. Üç boyutlu semantik harita iyileştirmesi için en yaygın olarak kullanılan yaklaşım, bir Markov-zincir modelini izleyen ardışık tahmin olasılıklarını birleştiren "Bayes güncellemesi" dir. Bunun yerine, bir sınıflandırıcıdan gelen tahminleri birleştirmek yerine, anlamsal özellikleri birleştirmek için bir öğrenme yaklaşımı öneren bir çalışma yapılmıştır (Sun vd., 2018). Bu çalışma sayesinde uzun vadeli yerelleştirme ve haritalama da semantik hafıza korunmuştur ve bu da semantik haritalamayı doğrulamıştır.

Semantik haritalama ile ilgili bir başka çalışmada nesne algılama ve tanımayı gerçekleştirmek için bir nesne algılama modülü kullanılarak yapılmıştır ve kameranın tam konumunu elde etmek için bir yerelleştirme modülü kullanılmıştır. Ortamın anlamsal haritalarını elde etmek için iki modül birbirine entegre edilmiştir. Ayrıca, çerçevenin hesaplama verimliliğini artırmak için The Fast Line Rasterization Algoritmasına dayalı gelişmiş bir Octomap oluşturulmuştur (Zhang vd., 2018). Bu çalışmada haritalama oldukça hızlı gerçekleştirilmiştir.

Eşleştirme

Haritalama yaparken en önemli konulardan biri de ortamın kameralarla görüntülenmesidir. Bazı ortamları haritalamak için tek bir kamera yeterli olurken bazı ortamlarda, örneğin mobil robotların kullanıldığı ya da birden fazla robotun kullanıldığı alanlarda, tüm alanı tek bir kamera ile görüntülemek zordur. Bu yüzden tüm çalışma uzayının haritalanması için birden fazla kamera kullanılması gerekir. Birden fazla kamera kullanıldığı durumlarda ortamın 3 boyutlu modellenebilmesi için kameralardan alınan görüntülerin eşleştirilmesi gerekir. Bununla ilgili bazı çalışmalar yapılmıştır. Fakat en yaygın kullanılan Doluluk Izgara Haritalaması yaklaşımdır. Bu yaklaşım kullanılarak haritalama yapmayı hedefleyen bir çalışmada harita birleştirme için yeni bir yöntem sunuluyor (Yue vd., 2018). Bu çalışmada haritaların birleştirilmesinden önce, harita özellikleri ve belirsizlikleri açıkça modelleniyor ve bütünleştiriliyor. Harita eşleştirme için ise yüksek seviyeli yapısal ve düşük seviyeli voksel özelliklerini içerecek şekilde iki seviyeli olasılıklı harita eşleştirme

algoritması geliştiriliyor. Yapılan çalışmayla öteleme ve döndürme tahminin doğruluğu da test edilmiştir.

2.3. Haritalamada Kullanılan Sensörler

Robotun belirlenen ortamda hedeflenen işleri yapmasına yardımcı olan çevresel birimler sensörlerdir. Sensörler fiziksel ortam ile endüstriyel amaçlı elektrik ve elektronik cihazları birbirine bağlayan köprü görevindedirler. Sensörler haritalamada kullanılan en önemli elemanlardır. Haritalama işlemi kullanılan sensörlere ve yöntemlere göre farklılık gösterir. Literatür incelendiğinde robotik alanında farklı amaçlar için kullanılan çeşitli sensörlere rastlanılmaktadır. Bu bölümde haritalama için en yaygın kullanılan bazı sensörlerin incelenmesi amaçlanmıştır.

Son zamanlarda, RGB-D (Kırmızı-Yeşil-Mavi-Derinlik) sensörlerinin piyasaya sürülmesiyle bu alandaki araştırmalar giderek daha popüler hale geldi ve RGB-D sensörlerine dayalı haritalama yöntemleri de yaygınlaştı. Bu konuda en önemli faktör RGB-D sensörün görüş ve derinlik verisi sağlayabilmesidir. RGB-D sensörlerde üretilen derinlik görüntüsü üzerinden çeşitli hesaplamalar yapılarak 3 boyutlu nokta bilgisi elde edilebilir. Bu sensörler, bir tane kızılötesi sensörü, bir tane derinlik sensörü ve bir tanede RGB (Kırmızı-Yeşil-Mavi) kameradan oluşmaktadır. Kızılötesi (IR) sensör sayesinde nesneye kızılötesi dalga boyunda ışık göndererek derinlik haritasını oluşturur ve derinlik sensörü sayesinde de bu ışık algılanır. Ayrıca RGB-D gibi Time-of-flight (ToF) kameralar derinlik haritası üretir. ToF kameraların her görüntüye ait piksel mesafesini ölçerek kaydetmesi sayesinde herhangi bir ek algoritma olmadan 3 boyutlu sahneler yakalanır. RGB-D sensörler kırmızı, yeşil ve mavi olmak üzere 3 adet CMOS sensör içerir. Her sensörün 8' lik bir veri aktarımı vardır ve elde edilen görüntü toplam 24 bit görüntü oluşturulur. Bu sensörler ilk olarak oyunlarda kullanılmak için geliştirilmiştir. Fakat günümüzde robotik alanında da yaygın olarak kullanılırlar. Özellikle RGB-D sensörü olarak Kinect sensörü kullanılarak birçok çalışma yapılmıştır (Endress vd., 2014; Miseikis vd., 2016).

Haritalamada lazer ve sonar mesafe sensörleri de önemli bir yere sahiptir. Mesafe sensörleri herhangi bir fiziksel temas söz konusu olmadan bir nesnenin yakınlığını belirlemede kullanılır. Haritalamada özellikle çarpışma durumlarından kaçınmak için

kullanılırlar. Literatürde lazer veya sonar mesafe sensörleri kullanan yöntemler ve tek veya çoklu kameraya dayalı görsel haritalama yöntemleri bulunmaktadır (Lee ve Lee, 2016; Misnan vd.,2012).

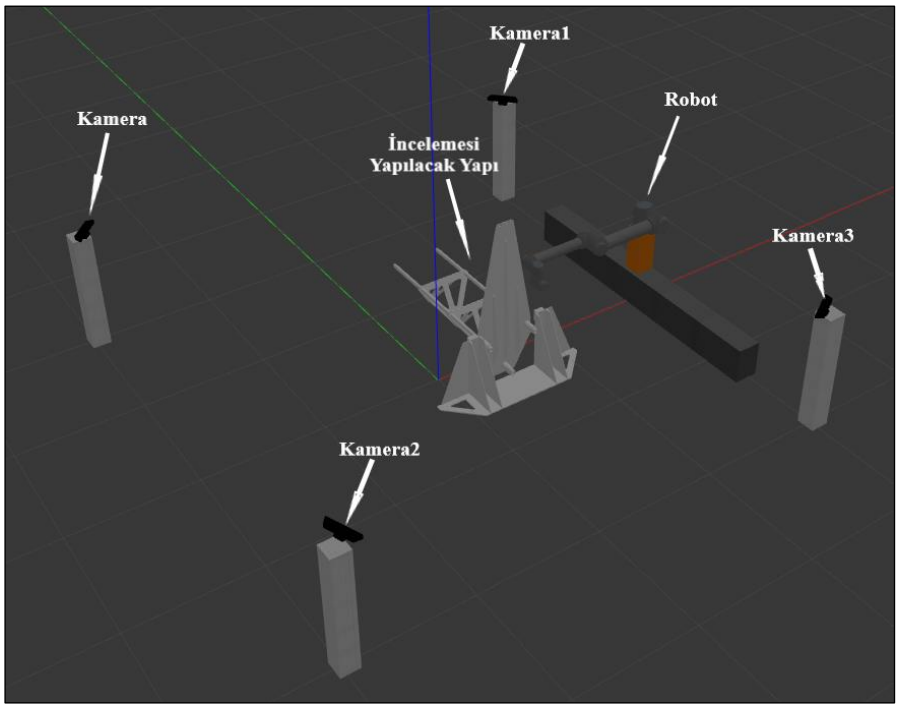
3. MATERYAL VE YÖNTEM

Önerilen yöntemde, ızgara tabanlı haritalama, derinlik kameraları ve endüstriyel robot kol kullanılmaktadır. Izgara tabanlı haritalama için sekizli ağaç (octree) tabanlı 3B haritalama yapısı OctoMap kullanılmaktadır (Hornung vd., 2012). Ayrıca, deneysel çalışmaların yürütülmesinde, 3B benzetim ortamlarından faydalanılmaktadır. Bu tez çalışmasında benzetim ortamı olarak GAZEBO kullanılmaktadır (Anonim2, 2014). Benzetim ortamı ve kontrol programları, robot işletim sistemi (ROS, **R**obot **O**perating **S**ystem) üzerinde çalıştırılmaktadır (Anonim1, 2020).

Önerilen yöntem, ızgara tabanlı haritada olması beklenen parçaların etiketlenmesi, derinlik kamerasından alınan veriler ile haritada dolu ve boş ızgaraların belirlenmesi ve sonrasında etiketlerle doluluk değerini karşılaştırılması ile denetleme sağlanmaktadır.

3.1. Deneysel Çalışma Ortamının Oluşturulması

Önerilen yöntemin geliştirilmesi sürecinde, deneysel bir sistemden verilerin elde edilmesine ihtiyaç vardır. Bu nedenle, deneysel çalışma ortamı oluşturulmuştur. Oluşturulan çalışma ortamı örneği Şekil 3.1' de gösterilmiştir. Bu ortamda, benzetim yazılımları, ara katman yazılımlar ve haritalamada kullanılan açık kaynak kodlu kütüphanelerden faydalanılmıştır. Benzetim ortamı, ROS ve Gazebo programları kullanılarak gerçekleştirilmiştir. Deneysel çalışma ortamı SDF (**S**patial **D**ata **F**ormat) formatında oluşturulmuştur. SDF, XML (Genişletilebilir İşaretleme Dili) formatında modellerin tanımlanmasında kullanılan bir formattır. Gazebo programı ise, 3B benzetim ortamı SDF formatında tanımlanmış deneysel ortamın modellenmesini gerçekleştirmektedir. Model, RVIZ (Anonim3, 2015) adlı 3 boyutlu görselleştirme aracı ile görselleştirilmekte, oluşturulan ortamın uygunluğu ve kameralardan elde edilen bilgiler kontrol edilebilmektedir.



Şekil 3.1 Gazebo' da Oluşturulan Ortam Görüntüsü

3.1.1. Robot İşletim Sistemi (ROS)

ROS, açık kaynaklı bir robotik ara katman yazılım paketidir. Donanım soyutlaması, düşük seviyeli cihaz kontrolü, yaygın olarak kullanılan işlevlerin uygulanması, süreçler arasında mesaj geçişi ve paket yönetimi dahil olmak üzere birçok hizmet sunar (Wikipedia, 2021). Ayrıca, birden çok bilgisayarda kod almak, oluşturmak, yazmak ve çalıştırmak için araçlar ve kitaplıklar sağlar. ROS gerçek zamanlı bir çerçeve değildir, ancak ROS' u gerçek zamanlı kodla entegre etmek mümkündür.

ROS robotla ilgili tüm süreçleri ve varlıkları ayrı düğümler olarak kabul eder ve düğümler arasında iletilen mesajlarda bir grafik mimarisi vardır. ROS aynı zamanda uluslararası bir topluluktan gelen yerleşik robot modüllerinin geniş bir koleksiyonudur. Mesaj türlerinin ve ağ iletişiminin soyutlanması, robotik sistemlerin tek modüllerinin farklı bilgisayar sistemlerinde yürütülmesine izin verir. Ayrıca, modüler mimari, düğümleri gerekli mesaj protokolüne uygunluk öncülüne göre değiştirilebilir hale getirir. Bu, geliştiricilerin bir robotun belirli işlevlerini çok az çabayla değiştirmesine izin verdiği için araştırma amaçlı önemli bir husustur.

Koddaki yinelemeli değişiklikler test ve doğrulama gerektirir, ancak her test için gerçek sistem deneyleri yapmak pek mümkün değildir. İyi bir şekilde çalışmayan kod, örneğin uçan robotları test ederken, donanıma zarar verebilir ve hatta insanları tehlikeye atabilir. Bu nedenle simülasyon odaklı geliştirme ve test etme vazgeçilmez bir gereklilik haline gelmiştir. ROS, Gazebo ve Webots gibi simülasyon ortamlarına entegre edilebilir olmasıyla oldukça kullanışlı bir programdır.

3.1.2. Benzetim Yazılımı (Gazebo)

Gazebo, robot sistemleri için benzetim ortamıdır. ROS ile uyumludur. ROS altyapısını kullanan programlar ile mesajlaşarak, denetim programları tarafından yönetilebilir. Gazebo ile çeşitli robotlar, sensörler ve 3-boyutlu nesnelerin benzetimleri yapılabilmektedir. Gazebo ayrıca aydınlatma, yerçekimi, atalet vb. için fiziksel bir motor kullanır. Robotu zor veya tehlikeli senaryolarda robota zarar vermeden değerlendirmeye ve test etmeye olanak sağlar.

Gazebo veri tabanında birçok hazır modeller bulundurduğu gibi kullanıcıların kendi modelini XML formatında tasarlanması için de uygundur. Gazebo temelinde istemci sunucu ilişkisiyle çalışmaktadır. Sunucu (gzserver) kısmı fiziksel işlemleri yaparken, istemci (gzclient) kısmı kullanıcının etkileşimini ve simülasyonun görselleştirilmesini sağlar.

Gazebo' nun özelliklerini aşağıdaki gibi sıralayabiliriz (Anonim2, 2014);

Dinamik Simülasyon: ODE, Bullet, Simbody ve DART dahil olmak üzere birden fazla yüksek performanslı fizik motoruna erişim sağlar.

Robot Modelleri: PR2, Pioneer2 DX, iRobotCreate ve TurtleBot dahil olmak üzere birçok robot hazır olarak kullanılabilir. Kullanıcı kendi robot modelini oluşturmak isterse SDF kullanarak oluşturabilir.

Gelişmiş 3B Grafikler: Gazebo OGRE kütüphanesini kullanarak, yüksek kaliteli aydınlatma, gölgeler ve dokular dahil ortamların gerçekçi bir şekilde işlenmesini sağlar.

TCP/IP ile Veri İletme: Uzak sunuculardа çalıştırılan Gazebo benzetim yazılımlarına soket tabanlı mesaj ileterek ara yüz oluşturulabilir.

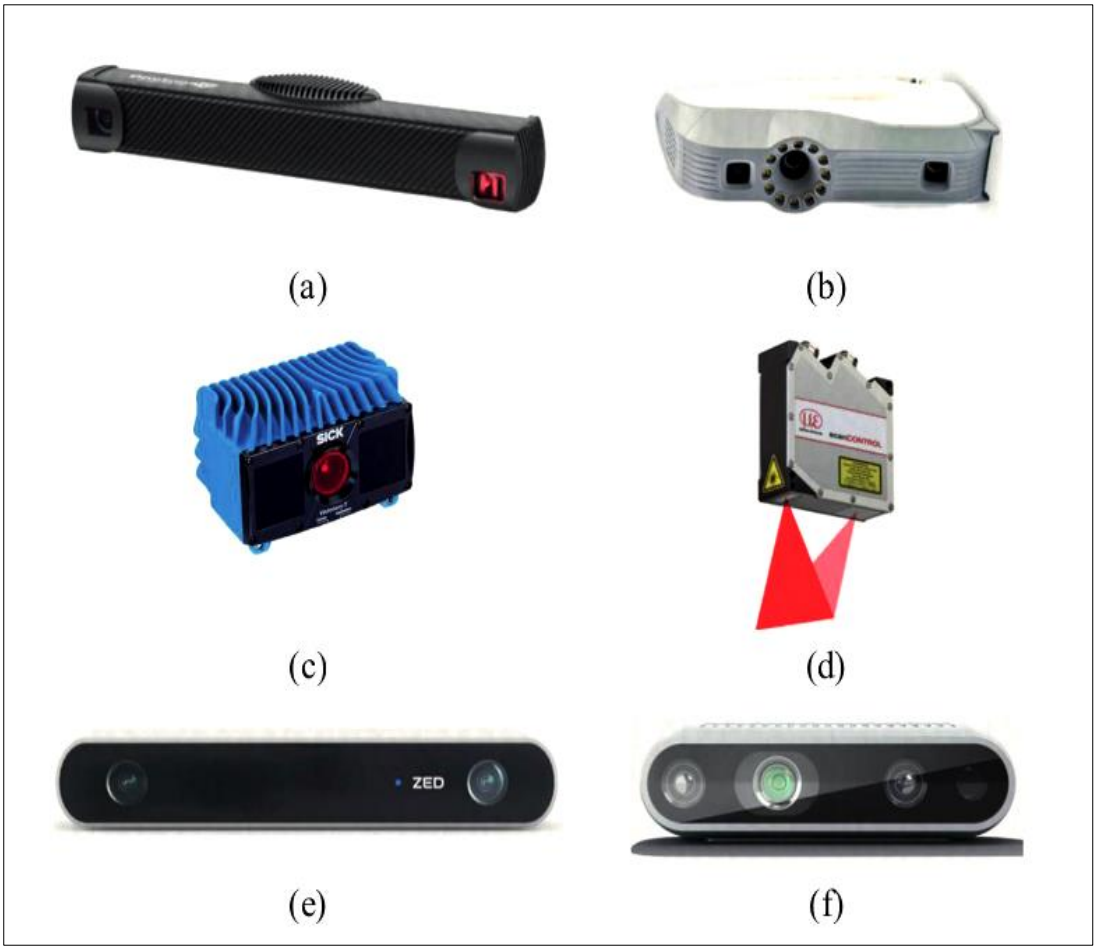
Sensörler ve Gürültü: Lazer mesafe sensörlerinden, 2B/3B kameralardan, Kinect tarzı sensörlerden, kontak sensörlerinden ve daha fazlasından isteğe bağlı olarak gürültülü sensör verileri oluşturulmasına olanak sağlar.

Eklentiler (Plugins): Robot, sensör ve çevre kontrolü için özel eklentiler geliştirilebilir. Eklentiler, Gazebo' nun API' sine doğrudan erişim sağlar.

Komut Satırı Araçları: Kapsamlı komut satırı araçları, simülasyonun iç gözlemini ve kontrolünü kolaylaştırır.

3.1.3. Derinlik Sensörleri

Robotlar ortamda sensörden aldığı verilere göre hareket eder. İhtiyaca ve çalışmanın amacına göre birçok farklı çeşit sensör vardır. Robotik denetleme amaçlı olarak, özellikle derinlik sensörleri yaygın kullanılmaktadır. Sensörler farklı derinlik algılama teknikleri kullanarak nokta bulutlarını sağlarlar (Özkan ve Yazıcı, 2021). Derinlik sensörleri (Şekil 3.2) çalışma yöntemlerine göre aşağıdaki gibi gruplandırılabilir.



Şekil 3.2 Yapılandırılmış Işık Teknolojisi Kullanan 3B Sensörler. **(a)** PhoXi 3B Tarayıcı, **(b)** Artec Eva El Tipi Tarayıcı, **(c)** Uçuş Süresi Kullanan 3B Sensör, **(d)** Lazer Üçgenleme Kullanan 3B Sensör: Mikro Epsilon Profil Sensörü, **(e)** Gömülü Stereo Kamera: Stereo Labs ZED Kamera, **(f)** Aktif IR Stereo Kamera: Intel RealSense (Özkan ve Yazıcı, 2021)

Yapılandırılmış Işık Tekniği Kullanan 3B Sensörler: Bu sensör tiplerinde yapılandırılmış ışık projektörü ve 2B görüntü kamerası kullanılır. Son zamanlarda kullanılan 3B yazıcılarda bu teknoloji ile çalışmaktadır. Bu sensörler ortama yansıtılan ışık örüntüsüyle çalışan trigonometrik üçgenleme kullanır. Işık örüntüsü projektör ile yansıtılır. Yüzeyin geometrisine göre yansıyan ışık örüntüsü, kamera ya da kameralarda görüntü oluşturur. Oluşan bu görüntüler sayesinde nokta bulutları elde edilir.

Lazer Üçgenleme Tekniği Kullanan 3B Sensörler: Yapılandırılmış ışık tekniği kullanan sensörlerden farklı olarak yapılandırılmış ışık yerine lazer kullanılır. Yansıtılan lazer ortama

yerleştirilen kamera tarafından algılanır. Yansıtılan lazerin yüzeyden yansıma açısı uzaklığı belirlemeye yardımcı olur.

Uçuş Süresi Kameraları: Lazer ya da LED ile sağlanan ışık, bir yüzeye gönderildiğinde yüzeyden yansıyarak gelen ışığın seyahat süresi ölçülerek uzaklık hesaplaması yapılır.

Gömülü Stereo Sensörler: İnsan görüş sisteminin çalışmasına benzer olarak çalışan pasif stereo kameralardır. Ortama sinyal göndermeden ortamda var olan ışığın yüzeyden yansıyarak kamerada oluşturduğu 2B görüntüden 3B veri elde edilir. Herhangi bir yüzeyden yansıyan ışık kameralarda oluşturduğu görüntülerde aynı piksellere düşmez. Bu pikseller arasındaki uzaklığa göre ışığın yansıdığı noktanın kameraya olan mesafesi belirlenir. Fakat bu işlem yapılırken noktanın görüntülerde hangi piksellere geldiği bulunması gerekir. Bunun için çeşitli algoritmalar kullanılır. Bu sensörler az enerji harcarlar, fakat ortamdaki doğal ışığı etkileyen gürültülerden çok fazla etkilenirler.

Aktif IR Stereo Kameralar: Çalışma tekniği pasif stereo kameralar ile aynıdır. Pasif stereo kameralardan farklı olarak ölçüm yapılacak yüzeye IR örüntü yansıtılmaktadır. Bu örüntü, sensör üzerinde yer alan IR projektör aracılığıyla yansıtılmaktadır.

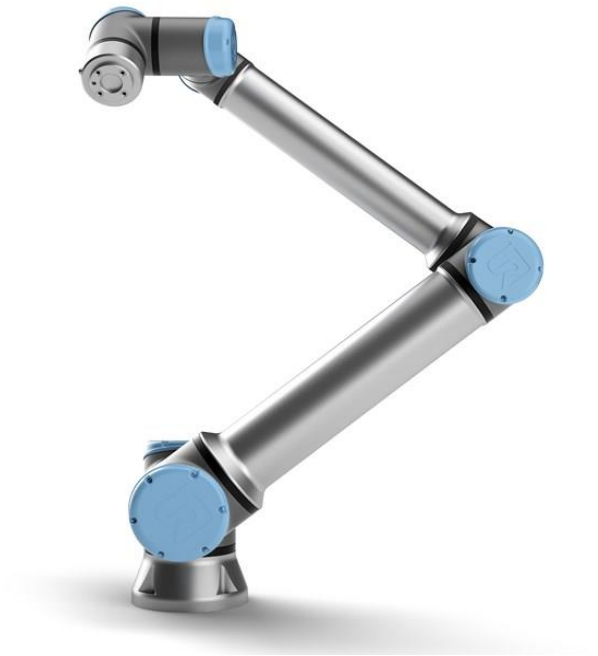
3.1.4. Endüstriyel Robot Kolları

Teknolojinin ilerlemesiyle birlikte robot sistemlerinin, endüstriyel üretim başta olmak üzere birçok alanda insanlara desteği artmaktadır. Bugüne kadar sensör ve eyleyici mekanizmalarının gelişmesiyle birlikte robotik sistemler iki kategoriye ayrıldı: hizmet robotları ve endüstriyel robotlar. Endüstriyel robotlar, tüm işçilik maliyetlerini düşürebilir, sürdürülebilir kalite ve üretimde esneklik sağlayabilir. Standart operatörlere kıyasla çeşitli üretim faaliyetlerini daha hızlı ve daha kaliteli gerçekleştirebilirler. Bu nedenle endüstriyel robotlar, endüstriyel alanda rekabeti ve üretim kalitesini artırmak için vazgeçilmez bir unsurlardandır.

Otomatik olarak kontrol edilebilen, yeniden programlanabilen ve üç veya daha fazla eksen üzerinde hareket edebilen endüstriyel robotlar, endüstri 4.0 devriminin en önemli araçlarından biridir. Bu bakımdan birçok ülke sanayisini bu devrime uygun olarak

değiştirmeye isteklidir. Özellikle imalatta otomasyon teknolojisini hızla geliştirmeye ve çağdaşlaştırmaya devam ediyor. Tüm bu kolaylıklarından dolayı son zamanlarda endüstride de robot kullanımı oldukça yaygın hale gelmiştir.

Endüstriyel robotlar günümüzde oldukça önemlidir. Özellikle insan gücünün yetersiz olduğu işlerde robotlar devreye girer. Ağır ürünlerin paketlenmesinden ince işçilik gerektiren birçok alanda rahatlıkla kullanılabilirler. Fakat robot kullanımında dikkat edilmesi gereken önemli unsurlar vardır. Bunlardan en önemlisi onların da aslında bir makine olduğu ve bundan dolayı hata yapılma olasılığının her zaman için var olmasıdır. Günümüzde, bu hata oranını azaltmak ve insanlarla iş birliği içinde çalışabilmeleri için robot hareket planlama ve haritalama gibi çeşitli alanlarda çalışmalar yapılıyor. Bizde bu çalışmada haritalama yaparken bazı iyileştirmeleri yaparak robotların üretimde hata olmaksızın kullanılmasını sağlanmaya çalıştık ve çalışmamızda benzetim ortamında Universal Robots firmasına ait UR10e cobot robotunun modelini kullandık (Şekil 3.3). Bu cobot 33,5 kg kablo dahil olmak üzere oldukça düşük ağırlığa sahip altı eksenli bir robot koludur. UR10e 12,5 kg taşıma kapasitesine ve 1300 mm erişim yarıçapına sahiptir.



Şekil 3.3 UR10e Robot Kolu (Kaynak: https://www.universal-robots.com/products/ur10-robot/)

3.1.5. 3B Doluluk Izgarası Haritalaması için Yazılım Kütüphanesi (OctoMap)

Octomap kütüphanesi, özellikle robotik için uygun olan C++' da veri yapıları ve haritalama algoritmaları sağlayan 3B doluluk ızgara haritalama yaklaşımıdır. Octomap, sekizli ağaca dayalı bir 3B haritalama yapısı kullanır. Bu bölümde ilk olarak sekizli ağaç yapısından sonrasında Octomap, Octomap veri yapısından ve doluluk ızgarası haritalamadan bahsedilmiştir.

Sekizli Ağaç

Sekizli ağaç, 3B'da uzamsal alt bölümün hiyerarşik bir veri yapısıdır. Sekizli ağaçtaki her düğüm, kübik hacimde (genellikle voksel olarak adlandırılır) bulunan alanı temsil eder. Hacim, belirli bir minimum voksel boyutuna ulaşılana kadar tekrar tekrar sekiz alt hacme bölünür (Şekil 2.1). Minimum voksel boyutu, sekizli ağacın çözünürlüğünü belirler. Sekizli hiyerarşik bir veri yapısı olduğundan, iç düğümler uygun şekilde korunursa, ağaç daha kaba alt bölümler elde etmek için herhangi bir düzeyde kesilebilir.

İkili bir ağacın 1 boyutlu bir alanı iki parçaya böldüğü gibi, bir sekizli ağaçta 3B alanı, her bir octantın bir düğüm ile temsil edildiği 8 alt birime böler. Sekizli ağaç düğümleri ek verileri depolamak için genişletilebilir. Bu ek veriler sayesinde hassas haritalama oluşturulabilir. Örneğin vokseller sıcaklık gibi çevresel verileri veya renk bilgisini depolayabilir. Her bir düğüm renk bilgisini kırmızı-yeşil-mavi (RGB) renk profili için tutar. Tüm yapı tek bir küpten oluşur ve çarpışma bilgileri için boş iç küpler kullanılır. Her bir alt birimde tutulan nispeten az miktarda veri olduğu için bellek açısından oldukça verimlidir.

Octomap

Octomap, haritalama yapılan alanda ve çözünürlükte maksimum esneklik sağlamak için ağaç tabanlı bir yapı kullanır. Sekizli ağaca dayalı bir 3B haritalama yapısıdır. Ölçeklenebilirliği sağlamak ve sensör gürültüsünü işlemek için olasılıklı doluluk tahminini gerçekleştirir. Sadece dolu alanlar değil boş ve bilinmeyen alanlarda temsil edilir. Ayrıca, sıkıştırma yöntemi elde edilen modelin kompaklığını da sağlayabilir. Octomap, aşağıda bahsedilen gereksinimleri karşılayacak biçimde tasarlanmıştır:

Tam 3B model: Harita, önceden herhangi bir varsayım olmaksızın herhangi bir ortamı modellemektedir. Ortamdaki bilinmeyen alanlar harita üzerinde örtük olarak kodlanmıştır. Güvenli robot navigasyonu için boş alan ve doğrudan alan arasındaki ayrım önemli olsa da bilinmeyen alanlar hakkındaki bilgiler, örneğin bağımsız arama ortamları için, çok önemlidir.

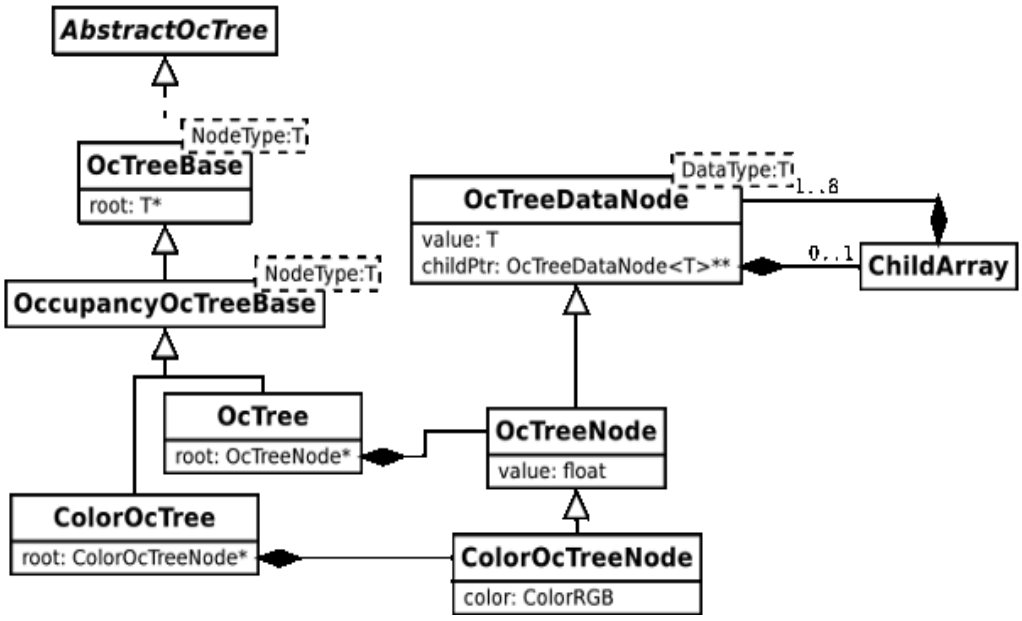
Güncellenebilirlik: Yeni bilgiler veya sensör okumaları herhangi bir zamanda eklenebilir. Modelleme ve güncelleme olasılıklı olarak yapılır. Bu, sensör gürültüsü veya dinamik nesnelerin neden olduğu çevresel dinamik değişiklikler nedeniyle ölçüm sonuçlarını hesaplayacaktır. Aynı şekilde, birden fazla bot aynı haritaya katkıda bulunabilir ve yeni alanlar keşfedildiğinde önceden kaydedilmiş haritaları genişletebilir.

Esnek: Haritanın kapsamının önceden bilinmesine gerek yoktur. Bunun yerine, haritayı gerektiği gibi dinamik olarak genişletir. Haritanın çözünürlüğü çok yüksektir, bu nedenle yerel plancılar yüksek çözünürlüğü kullanabilirken, en iyi planlamacılar kaba haritalar kullanabilir. Ayrıca kaba genel bakışlardan ayrıntılı yakın çekimlere kadar etkili bir görselleştirme sağlayabilir.

Kompakt: Haritalama etkin bir şekilde bellek ve diskte saklanır. Bant genişliği sınırlamaları altında bile, gelecekte robotlar arasında kolay değişim için sıkıştırılmış dosyalar oluşturulabilir.

Octomap Veri Yapısı

Çerçevemizdeki en yaygın sekizli ağaç ve düğüm türleri Şekil 3.4’ de bir UML (Unified Modeling Language) diyagramı olarak özetlenmiştir. Temel sekizli ağaç işlevi OcTreeBase' de ve temel düğüm işlevi OcTreeDataNode' da uygulanır. OcTreeDataNode, düğümde depolanan veriler üzerinde modellenirken, OcTreeBase, düğüm türü üzerinde modellenir. OccupancyOcTreeBase, ağaç uygulamasına tarama ekleme (Scan Insertions) ve ışın dökümü (Ray Casting) gibi doluluk eşleme işlevselliği ekler. Temel OccupancyOctree sınıfı sekizli ağaç, düğümleri için OcTreeNode kullanan OccupancyOcTreeBase' den türetilir. Bu yapı, örneğin düğümleri özel verilerle genişletmek veya sekizli ağaca yeni işlevler eklemek için çerçevemizin farklı seviyelerde esnek uzantılarına izin verir.



Şekil 3.4 En Yaygın Oktree ve Düğüm Sınıflarının UML Diyagramı (Hornung vd., 2012)

Bu çalışmada, Octomap 'in ColorOcTreeNodes yapısını kullanan ColorOcTree uygulamasını kullanıldı. Bu düğümler, bir doluluk tahmininin yanı sıra renk depolar.

Kütüphane ile ilgili değişiklikler C++ dilinde yazılan kodlarla sağlanabilir. Bu yazılan kodlarda istenilen durumları yapmak için dosyalar .ot veya .bt dosya formatlarında kaydedilebilir. Octomap her iki formatıda destekler. .ot (Octree) dosya formatı OctoMap formatında bir son doluluk haritası içerir ve düğümler, tam (olasılıklı) doluluk bilgilerini depolar. .bt (Bonsai ağacı veya ikili ağaç) dosyaları, yalnızca maksimum olasılık bilgisiyle bir doluluk haritasını kodlar, her düğüm ya boştur ya da doludur. Hem .ot hem de .bt dosyaları Octovis' te görselleştirilebilir (Anonim4, 2017).

Doluluk Izgarası Haritalama

Doluluk ızgarası metodu ilk olarak Elfes (1989) tarafından resmileştirilmiştir. Bu gösterimle, ortamın sürekli uzayları ayrıklaştırılır, böylece ortam bir ızgara veya çok boyutlu matris (2B veya 3B) konfigürasyonunda temsil edilir. Hücre olarak da adlandırılan matrisin her bir öğesi, dolu, boş veya keşfedilmemiş bir ortam konumunu temsil eder. Doluluk ızgara gösterimi başlangıçta ortamları 2 boyutlu bir ızgarada haritalamak için önerildi, ancak Elfes'

ten sonra Moravec (1996) bu temsili haritanın oluşturulması için kullanılan sensör tipini de yenileyerek 3B ayrık bir konfigürasyona genişletti. Octomap kütüphanesi de 3B Doluluk Izgara yaklaşımını kullanan haritalama yaklaşımını uygular. Doluluk ızgarası haritalamasın da düğümlerde doluluk olasılığı depolanır.

Robotik sistemler genel olarak sensör gürültüsüyle ve geçici veya kalıcı değişen ortamlarla başa çıkmak zorundadır. Bu gibi durumlarda tek başına doluluk bilgisini etkilemek yeterli değildir çünkü bu bazı haritalamalar da hafıza yönünden verimsiz olabilir. Bunun yerine doluluğun olasılıksal olarak modellenmesi gerekir. Doluluk ızgarasının temel fikri, her bir ortamdaki o konumdaki bir engelin varlığını temsil eden ikili rasgele değişkenlerin eşit aralıklı verimli olarak bir ortam haritasını temsil etmektir. Doluluk Izgara algoritmaları, ortamı ızgaralara böler ve rastgele değişkenler için yaklaşık sonsal tahminleri hesaplar.

Bu işlemleri yaparken aynı zamanda düğümlerin dolu mu boş mu olduğunu olasılıklı hesaplar ve doluluk boşluk durumunu ona göre belirler (Wikipedia, 2021). Olasılık tahminini eşitlik denklem 3.1 de verilen formül taban alınarak hesaplanır;

$$P = (m|z_{1:t}, x_{1:t}) \tag{3.1}$$

Formülde m haritayı, $z_{1:t}$ 1'den t'ye kadar olan ölçüm setini ve $x_{1:t}$ 1'den t'ye kadar olan robot pozlarını ifade eder. 2B haritalarda genellikle iki boyutu temsil etmek için iki indeks kullanılır. Izgara hücreleri i indeksi ile gösterilirse, $p(m_i)$, i hücresinin doluluk olasılığını gösterir. Denklem 3.1' de iki indeksli yazılacak olursa;

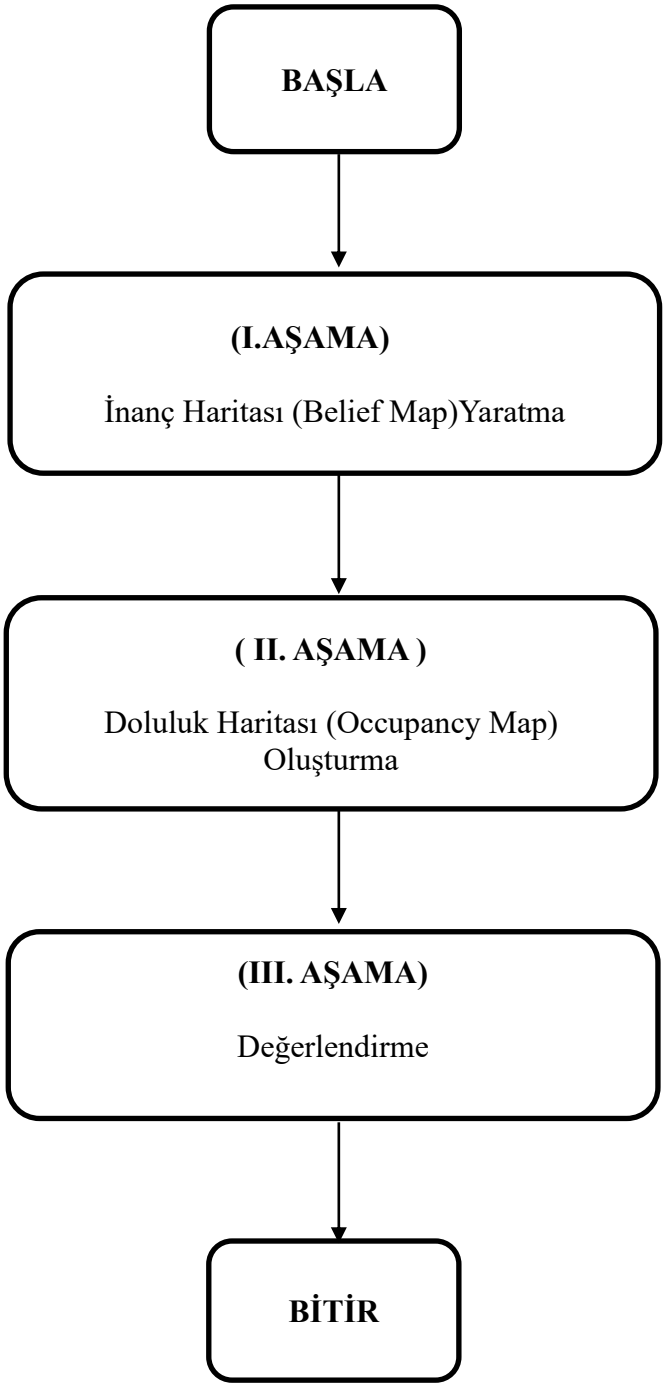
$$P = (m_i|z_{1:t}, x_{1:t}) \tag{3.2}$$

elde edilir.

Doluluk ızgara haritalama yaklaşımıyla, hücrelerin dolu mu boş mu olduğu olasılıklı olarak değerlendirilirken her ızgara hücre birbirinden bağımsız olarak değerlendirilir. Bu işlemler yapılırken çevreden alınan farklı görüntüler taban alınır ve oluşturulan harita temsili bir haritadır.

3.2. Izgara Tabanlı Haritalamaya Dayalı Robotik Denetleme Yöntemi

Robotik denetim için önerilen yöntem, Şekil 3.5' de görüldüğü gibi 3 aşamadan oluşmaktadır. Birinci aşamada, inanç haritası (Belief map) yaratılmaktadır. Bu kısımda, üretim sürecindeki yapıda montajı yapılması beklenen parçaların katı modelleri bu haritanın oluşmasında kullanılmaktadır. Haritada bu parçalar etiketlenmektedir. İkinci aşamada, doluluk haritası (Occupancy Map) oluşturulmaktadır. Derinlik algılayıcılardan sağlanan verilerle haritadaki ızgaraların dolu ya da boş olma olasılıkları belirlenmektedir. Son aşama ise, inanç haritası yaratılırken etiketlenen ızgaraların, ikinci aşamada belirlenen doluluk bilgilerinin değerlendirmesi yapılmaktadır. Bu değerlendirme sonucunda, yapıda monte edilmiş olması beklenen parçaların gerçekten monte edilip edilmediği belirlenmektedir. İzleyen alt başlıklarda her aşamanın detaylı anlatımı verilmektedir.

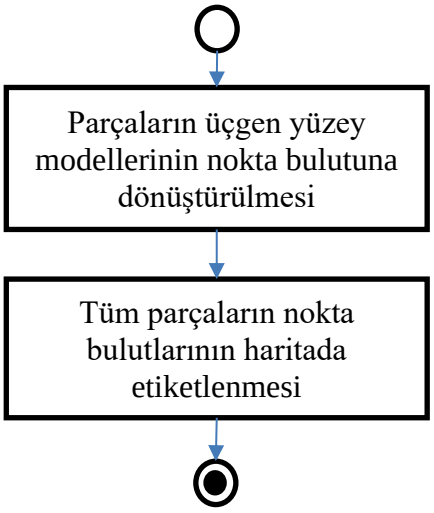


Şekil 3.5 Izgara Tabanlı Haritalamaya Dayalı Robotik Denetleme Yönteminin Uygulama Aşamaları

3.2.1. İnanç Haritası (Belief Map) Yaratma (I. AŞAMA)

Haritalamada, ortamın hacimsel temsilinde sekizli ağaç (octree) veri yapısı kullanılmaktadır. Sekizli ağaç veri yapısı ile oluşturulan harita, $\mathcal{M} = (\mathcal{N}, \mathcal{L})$ ile temsil edilebilir. Burada, $\mathcal{N}$ bir düğüm kümesi ve $\mathcal{L}$ bir yaprak kümesini temsil etmektedir. $\mathcal{N} = (n_1, \cdots, n_k)$, k düğüm sayısı, n_i ise düğümlerdir. Her bir düğüm $n_i = (\alpha_i, \rho_i, \mathcal{N}_i')$ olarak ve her bir yaprak $\ell_i = (\alpha_i, \rho_i)$ olarak tanımlanabilir. Burada, $\alpha_i \in \mathbb{Z}$ düğüm veri yapısında etiketleme amaçlı kullanılan tam sayı bir değer, ρ_i düğümün temsil ettiği hacmin doluluk oranını, $\mathcal{N}_i'$ ise çocuk düğümleri gösteren işaretçiyi temsil etmektedir.

İnanç haritası Şekil 3.6' da görüldüğü gibi iki aşamada oluşmaktadır. İlk aşamada, denetlemesi yapılacak parçaların üçgen örgü (mesh) ile yaratılmış üçgen yüzey modelleri nokta bulutunu dönüştürülmektedir. Bu dönüştürme sonucunda, bir yapıyı oluşturan tüm parçaların nokta bulutları $\mathcal{P}_i^c$, $i \in \mathbb{Z}$ ve $i \in [1, n_c]$ ile temsil edilmektedir. Burada, n_c parça sayısıdır. Her bir parça için nokta bulutundaki nokta yoğunluğu parça büyüklüğüne bağlı olarak seçilebilir.



Şekil 3.6 İnanç Haritasının Oluşturulması

İkinci aşamada ise, her bir nokta bulutunun haritada etiketlemesi yapılmaktadır. Bu işlem Algoritma 1'de verilmektedir.

Algoritma 1. Tüm parçaların nokta bulutlarının haritada etiketlenmesi

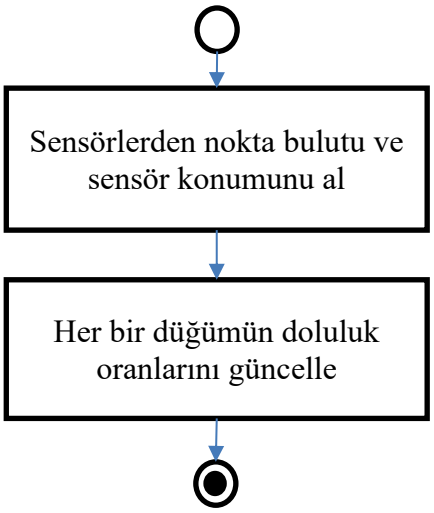
Girdi:$\{\mathcal{P}_1^c, \cdots, \mathcal{P}_{n_c}^c\}$

Çıktı:$\mathcal{M}_B$: İnanç haritası

for $i := 1\ to\ n_c$ **do**
    for $\forall p \in \mathcal{P}_i^c$ **do**
        $herhangi\ bir\ l = (\alpha, \rho) \in\ \mathcal{M}_B\ bul$ s.t. $l\ p'yi\ kapsar$
        $\alpha \leftarrow i$
        $\rho \leftarrow \emptyset$
    endfor
endfor

3.2.2. Doluluk Haritası (Occupancy Map) Oluşturma (II. AŞAMA)

Doluluk haritası, derinlik sensörlerinden elde edilen veriler ile düğümler $n_i = (\alpha_i, \rho_i, \mathcal{N}_i')$ ve yaprakların $\ell_i = (\alpha_i, \rho_i)$ doluluk olasılıklarının güncellenmesi ile oluşturulmaktadır. Bu süreçte, etiketleme amaçlı kullanılan veri α_i değiştirilmemektedir. Ancak, doluluk olasılığı ρ_i güncellenmektedir.



Şekil 3.7 Doluluk Haritasının Oluşturulması

Sensörlerden alınan nokta bulutları $\mathcal{P}_i^s$, $i \in \mathbb{Z}$ ve $i \in [1, n_s]$ ile temsil edilmektedir. Ayrıca, $S_i \in \mathbb{R}^3$, $i \in \mathbb{Z}$ ve $i \in [1, n_s]$ ise, i. her bir nokta nokta bulutunu sağlayan sensörün pozisyonudur. Burada, n_s ortamda kullanılan sensör sayısıdır.

Algoritma 2. Düğümlerin doluluk olasılıklarının güncellemesi

Girdi: :$\{\mathcal{P}_1^s, \cdots, \mathcal{P}_{n_s}^s\}$, $S_i, i \in [1, n_s]$: Sensör orijini, $\mathcal{M}_B$: İnanç haritası

Çıktı: $\mathcal{M}_O$: Doluluk haritası

Tanım: $(p \in \mathcal{P}_i^s, S_i) \xrightarrow{\rho} \{L_f, L_o\}$ işlemi verilen $p, S_i \in \mathbb{R}^3$ nokta çifti için L_f, L_o kümelerini oluşturmaktadır. L_o, p noktasını kapsayan hacmi temsil eden düğümü n_i içermektedir. L_f ise, $\overline{pS_i}$ doğru parçasını kapsayan hacimleri temsil eden düğümleri n_i içermektedir.

$\mathcal{M}_o \leftarrow \mathcal{M}_B$
for $i := 1$ to n_c **do**
    for $\forall p \in \mathcal{P}_i^s$ **do**
        $(p, S_i) \xrightarrow{\rho} \{L_f, L_o\}$
    for $\forall n \in L_f, \mathcal{M}_o$ **do**
        $\rho \leftarrow boş$
    endfor
    for $\forall n \in L_f, \mathcal{M}_o$ **do**
        $\rho \leftarrow dolu$
    endfor
endfor

3.2.3. Değerlendirme (III. AŞAMA)

Bu aşamada, haritadaki etiketleme amaçlı kullanılan veri α_i ve doluluk olasılığı ρ_i değerlerinin analizi ile parçaların olup, olmadığı tespit edilmektedir. Bunun için Algoritma 3' de verilen algoritma kullanılmaktadır. $\mathcal{C} = \{c_i | \forall i \in [1, n_c]\}$ bir sayaç kümesi, c_i ise sayaçları belirtmektedir. Her bir sayaç c_i, i. parça için dolu düğümlerin sayısını tutmaktadır. $\mathrm{T} = \{\tau_i | \forall i \in [1, n_c]\}$ ise, eşik kümesidir. Her bir τ_i eşik, parçanın olup olmadığının değerlendirmesinde kullanılmaktadır.

Algoritma 3. Değerlendirme

Girdi: $\mathcal{M}_o$, T

Çıktı: $\mathcal{E} = \{e_i | \forall i \in [1, n_c]\}$: Değerlendirme sonucu, $e_i \in \{mevcut, tanımsız\}$,

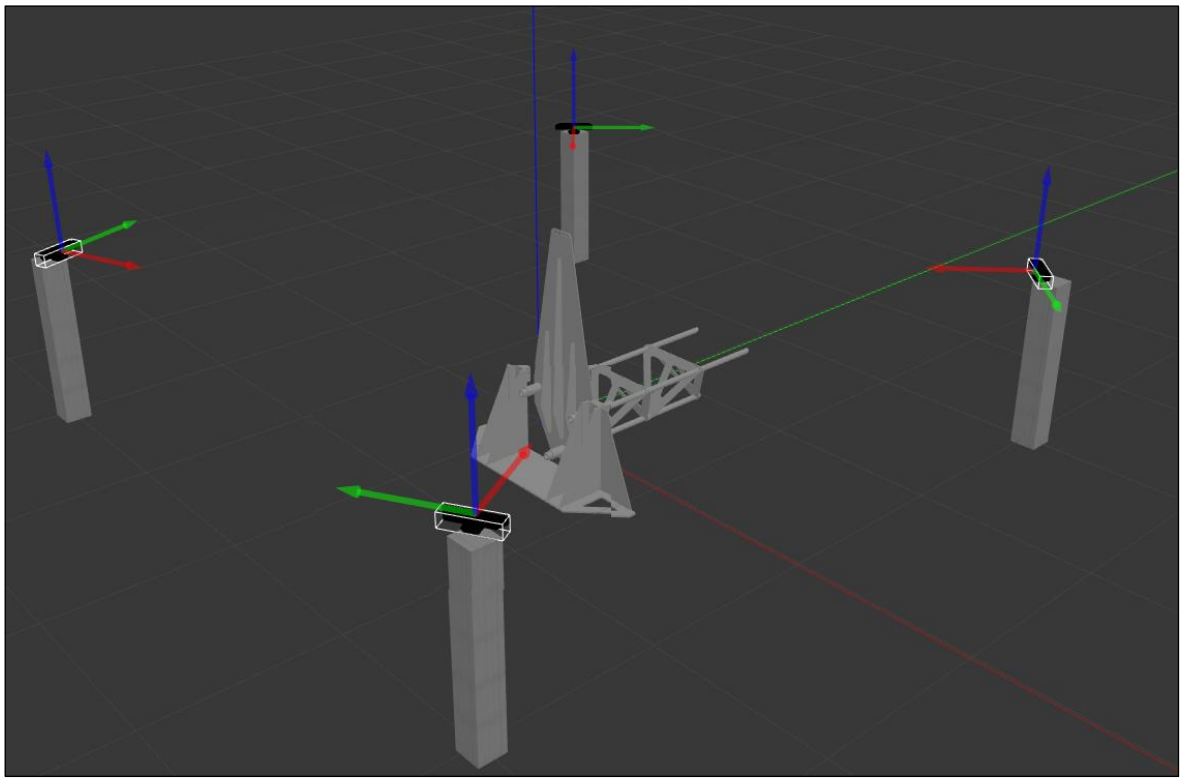
for $i := 1$ *to* n_c **do**
    $e_i \leftarrow 0$
endfor
for $\forall n \in L_f, \mathcal{M}_o$ **do**
    for $i := 1$ *to* n_c **do**
        if $\alpha == i$ **then**
            if $\rho == dolu$ **then**
                $c_i \leftarrow c_i + 1$
            endif
            break
        endif
    endfor
endfor
for $i := 1$ *to* n_c **do**
    if $c_i \geq \tau_i$ **then**
        $e_i \leftarrow mevcut$
    else
        $e_i \leftarrow tanımsız$
    endif
endfor

4. UYGULAMA

Yöntemin uygulanabilirliği ve performansını göstermek üzere deneysel çalışmalar yapılmıştır. Deneysel ortam Gazebo simülasyon ortamı kullanılarak oluşturulmuştur. Benzetim ortamına denetimi yapılacak yapılar yerleştirilmiş ve ortamda bu yapıların sensörler tarafından algılanması hedeflenerek oluşturulmuştur. Bu bölümde oluşturulan deneysel ortamdan, kullanılan yapılardan, sensörlerin teknik özelliklerinden ve uygulama adımlarından ayrıntılı olarak bahsedilecektir.

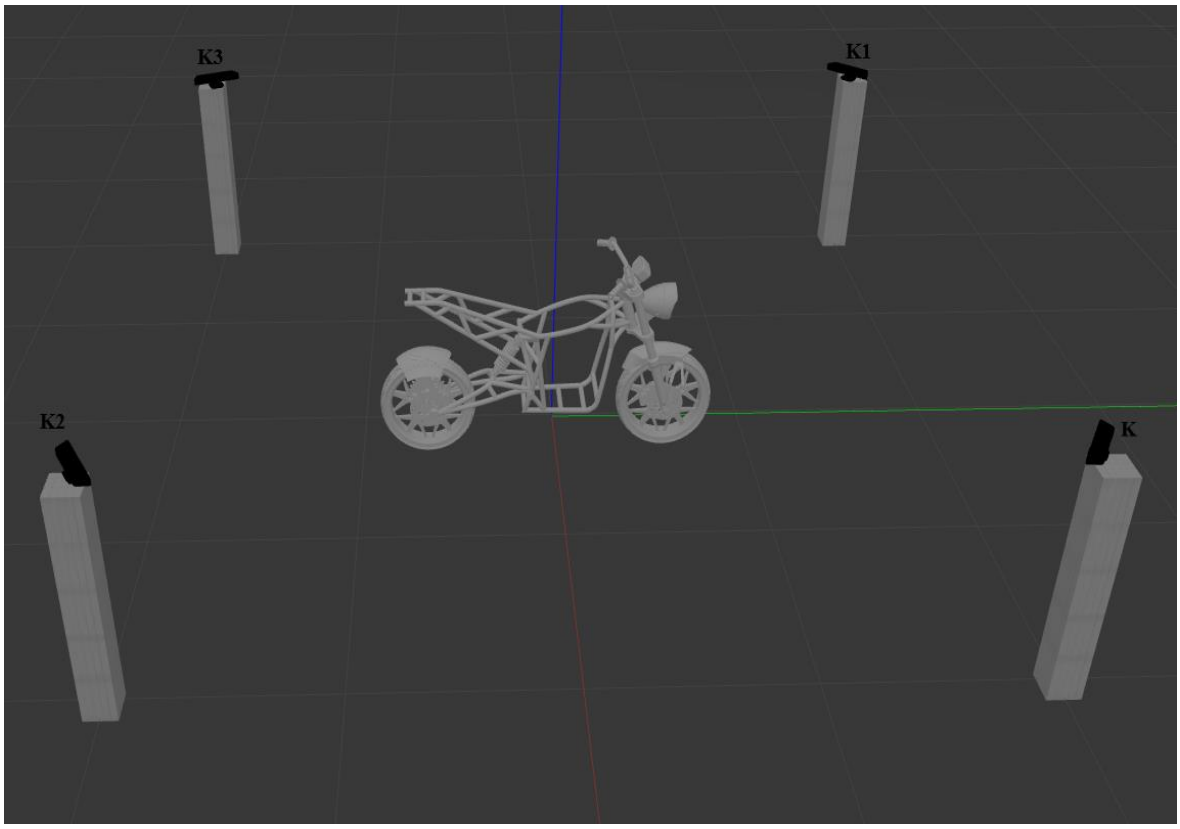
4.1. Deneysel Ortamın Oluşturulması

Bu çalışmada deneysel ortam Şekil 4.1' de gösterilen test ortamı referans alınarak Gazebo simülasyon programında tasarlanmıştır.

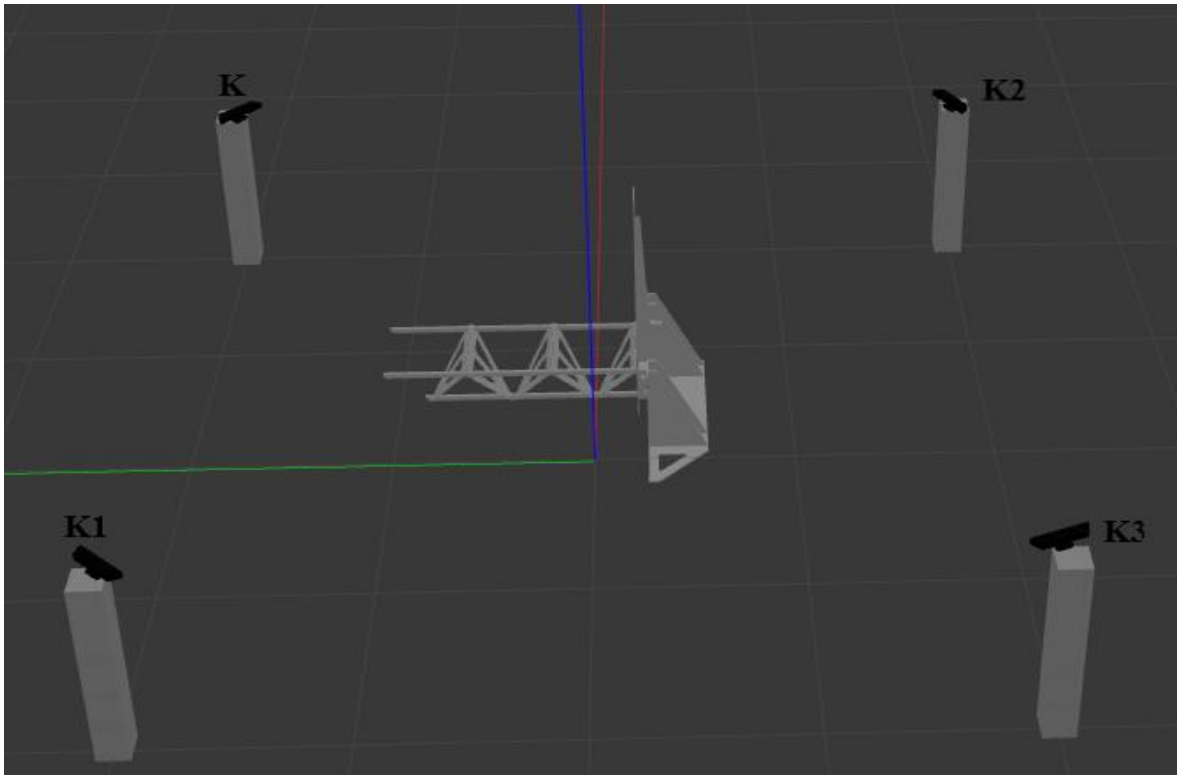


Şekil 4.1 Benzetim Ortamında Koordinat Sistemleri

Gazebo benzetim ortamı, farklı formatta 3B model ekleme yapmak için uygundur. Bu çalışmada, Gazebo benzetim programının bu özelliğini kullanarak iki farklı deney için ortama, iki ayrı yapıya ait parçalar eklendi. Oluşturulan bu ortamlara göre, denetleme görevi gerçekleştirildi. Benzetim ortamında, yapıların yüklenmiş hali Şekil 4.2.' de gösterilmiştir. Benzetim ortamlarında kullanılan kameralar K, K1, K2 ve K3 olarak isimlendirilerek gösterilmiştir. Ayrıca, Gazebo benzetim ortamına kamera ekleme, robota hareket kabiliyeti sağlama gibi özellikler modele eklenen eklentiler (plugins) ile sağlanır.



(a) Ortam-1



(b) Ortam-2

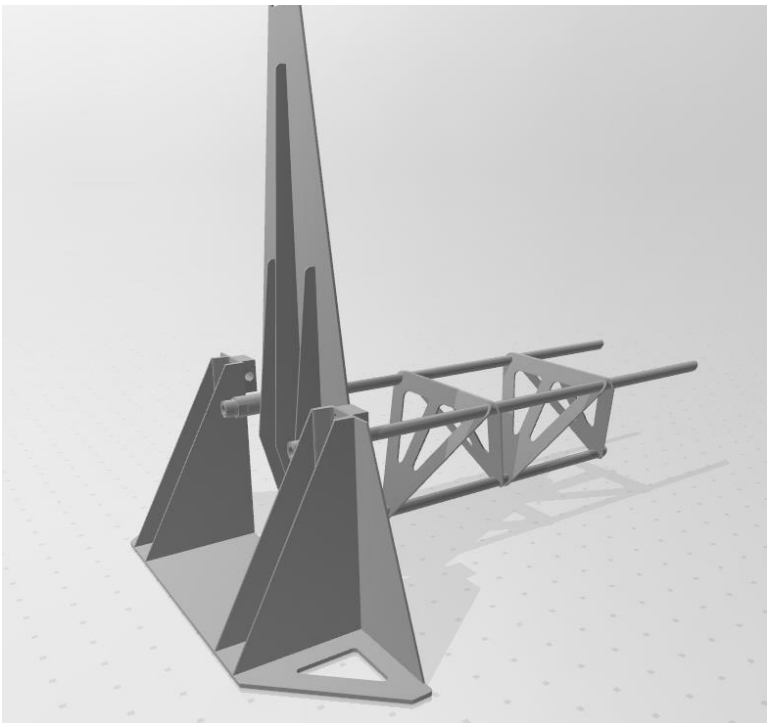
Şekil 4.2 Gazebo Benzetim Programında Test Ortamları

Gazebo benzetim ortamı nesneler eklemek için iki mekanizma sunmaktadır. Birincisi, oluşturma penceresinin üzerinde bulunan basit şekiller kümesidir. Hazır kullanılacak modeller bu şekilde eklenebilir. İkincisi, model veri tabanıdır. Ortama eklenecek modelin .sdf ve .config dosyaları oluşturularak Gazebo benzetim ortamına eklenir. Bu çalışmada, benzetim ortamı oluşturulurken ilk olarak, Gazebo benzetim ortamının Building Editor' u kullanılarak ortama yerden 1,5 m yüksekliğine sahip dört duvar oluşturuldu. Oluşturulan bu duvarların üzerine Kinect kameralar yerleştirilerek ortam görüntülenmesi sağlandı. Sonrasında ortama denetlemesi yapılacak yapı eklenerek .world uzantılı dosya olarak kaydedildi.

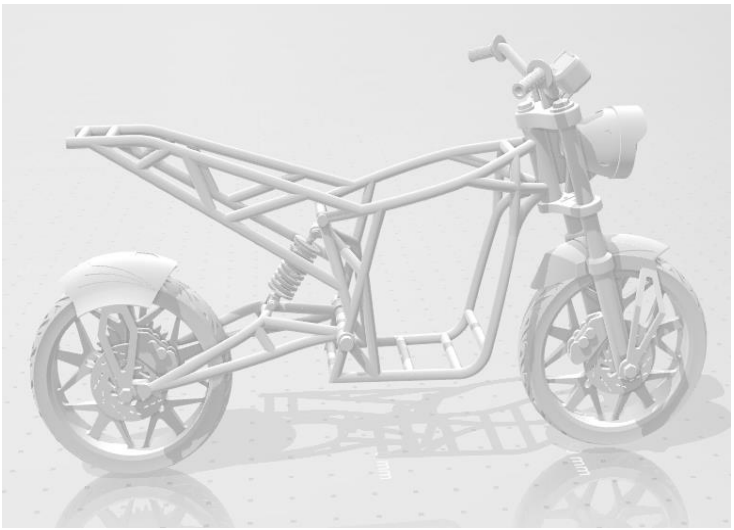
4.2. Denetimi Yapılacak Yapılar

Bu çalışmada, denetimi yapılmak üzere iki yapı kullanılmıştır. Bunlardan birisi Şekil 4.3' de görüldüğü gibi bir üçgen kirişli bir yapıdır. Bu yapı, 52 adet parçanın birbirine montajından oluşmaktadır. Bu parçalardan en küçüğü yaklaşık 45 mm en büyüğü 1375 mm 'dir. Diğer bir yapı ise Şekil 4.4' de gösterilmektedir. Elektrikli bir bisikletin iskeletini oluşturan bu yapı ise, 31 adet parçadan oluşmaktadır. Bu parçalardan en küçüğü yaklaşık 0,7

mm en büyüğü 120 mm ‘dir. Bu çalışmada yapılara ait parçaların denetimi gerçekleştirileceği için ortam oluşturulurken bu yapılara ait parçalar ortama ayrı ayrı eklenmiştir.



Şekil 4.3 Üçgen Kirişli Yapı (Truss) (Kaynak: https://grabcad.com/library/non-metallic-truss)



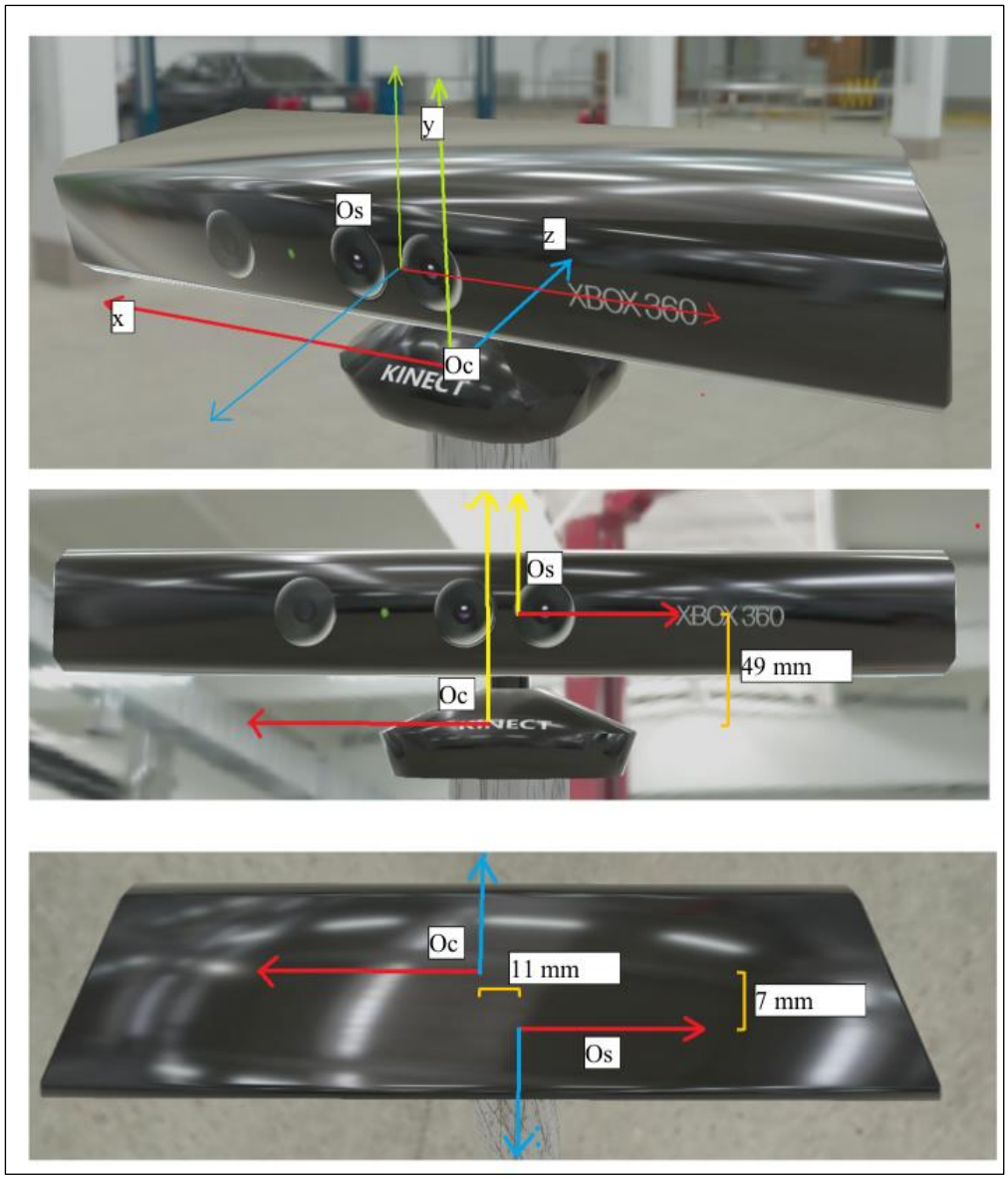
Şekil 4.4 Elektrikli Bisiklet İskeleti (Kaynak: https://grabcad.com/library/electric-bike-skeleton-2)

4.3. Derinlik Kamerası

Robotik denetimin yapılabilmesi için robot çalışma uzayına derinlik sensörleri eklenmiştir. Bu çalışmada benzetim ortamında dört tane Kinect XBOX 360 sensörü kullanılmıştır. Bu sensörler (Şekil 4.5) bir tane RGB kamera bir tane de derinlik sensörü içerir. Derinlik sensörü, bir kızılötesi projektör ve bir tek renkli CMOS sensörü, aydınlatma koşullarına bakılmaksızın alanı 3 boyutlu "görmek" için birlikte çalışır. Şekil 4.2' de görüldüğü gibi kameralar haritalama yapılacak alanın çevresine yerleştirilir. Çevreye yerleştirilen kameralar yerden 1,5 m yüksekliğinde 0,15 m genişliğinde bir yere sabitlenmiştir. Bu mesafeler ortama göre ayarlanabilir. Ayrıca, ortamda kullanılan kameraların koordinat sistemleri Şekil 4.6' da gösterildiği gibidir.



Şekil 4.5 Kinect XBOX 360



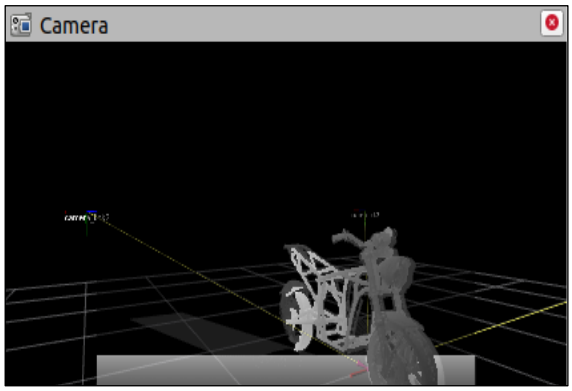
Şekil 4.6 Kamera üzerindeki koordinat sistemleri: (üst) Ön çapraz görünüm, (orta) önden görünüm, (alt) üstten görünüm

Oluşturulan deneysel çalışma ortamında kullanılan kameranın teknik özellikleri Çizelge 4.1 ‘de verilmiştir. Gazebo ortamında kameraya ait SDF dosyasında teknik özellikleri göz önünde bulundurularak oluşturulan ortama göre değişiklikler yapılabilir.

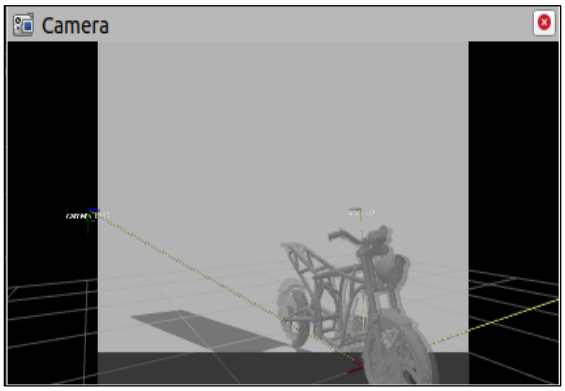
Çizelge 4.1 Kameranın Teknik Özellikleri

Cihaz	Kinect XBOX 360
RGB Kamera	640 x 480 @30 FPS
Derinlik Kamerası	320 x 240
Min / Max Derinlik Mesafesi	40 cm / 4,5 m
Yatay Görüş Alanı	57 Derece
Dikey Görüş Alanı	43 Derece
Fiziksel Eğim Aralığı	± 27 Derece

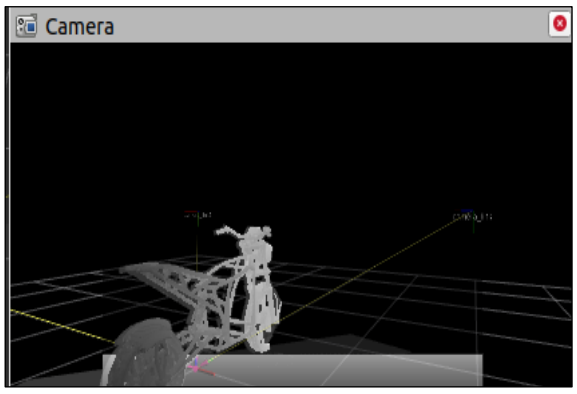
Ortam-1 ‘deki kameralardan alınan görüntüleri Şekil 4.7’ de gösterilmiştir.



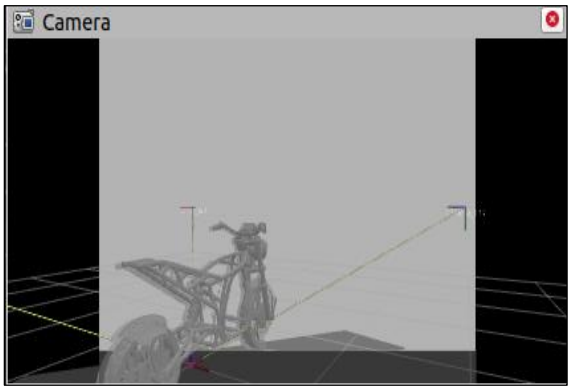
(a)



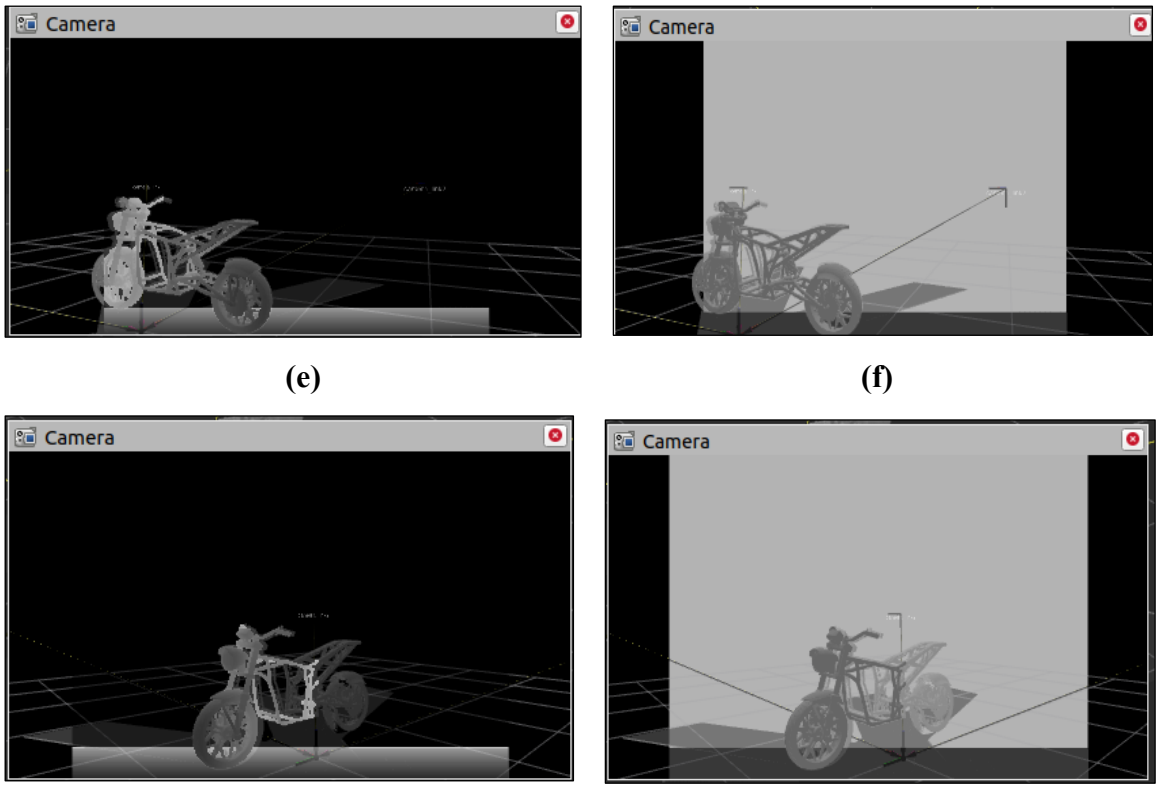
(b)



(c)

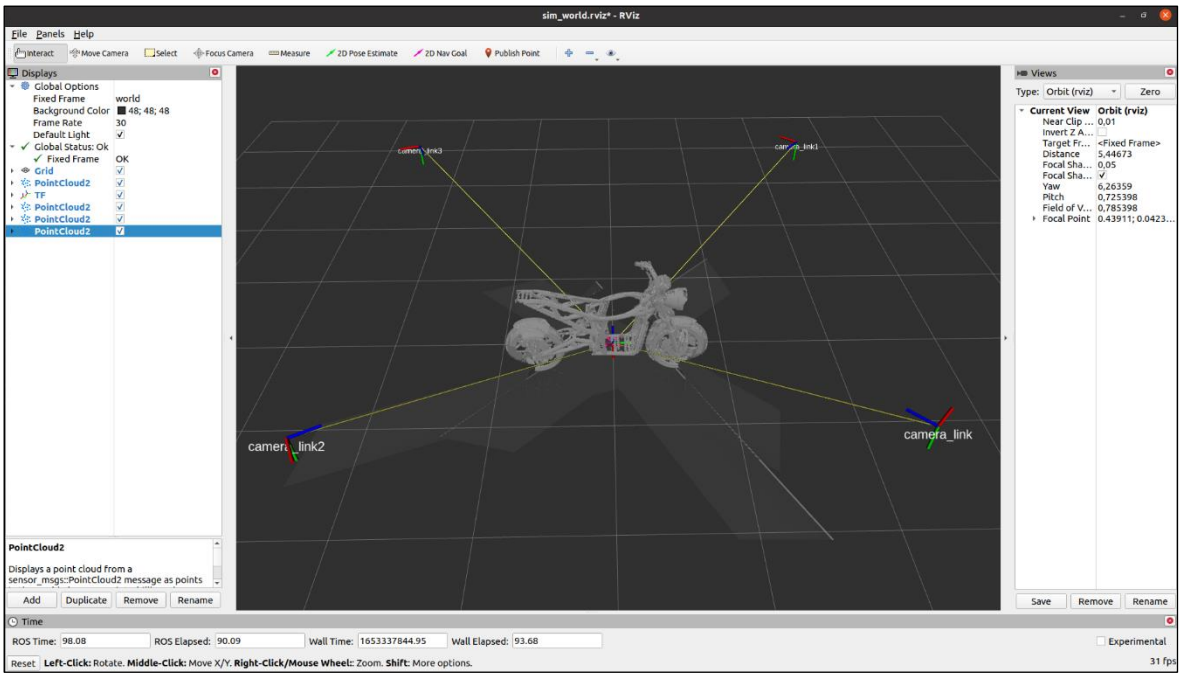


(d)



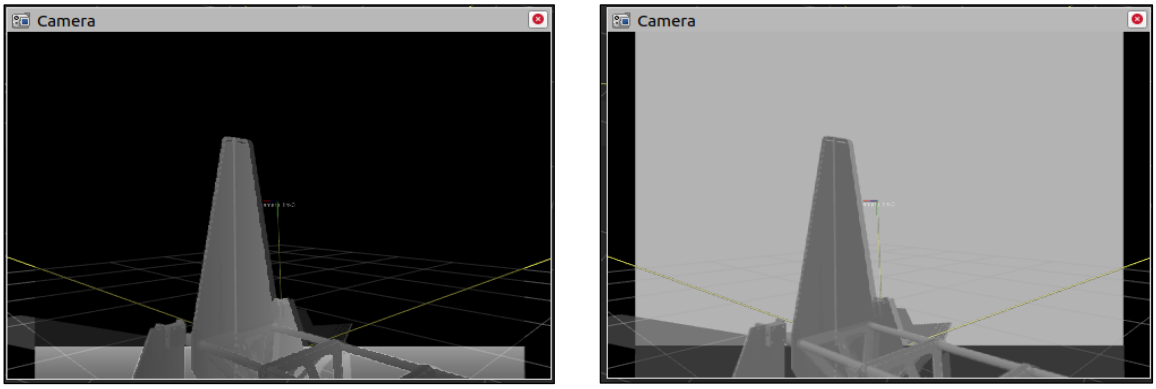
Şekil 4.7 Ortam-1' e ait Kamera Görüntüleri **(a)** K-Depth, **(b)** K-RGB, **(c)** K1-Depth, **(d)** K1-RGB, **(e)** K2-Depth, **(f)** K2-RGB, **(g)** K3-Depth ve **(h)** K3-RGB

Şekil 4.7' da verilen görüntülerden (a),(c),(e) ve (g) ortamdan alınan derinlik görüntülerdir. (b),(d),(f) ve (h) RGB görüntüleridir. Ayrıca ortamda kullanılan kameralar sabittir. Hareketli sensör kullanılmamıştır. Ortam-1'deki dört kameradan elde edilen nokta bulutlarının Rviz görüntüsü Şekil 4.8' de gösterilmiştir.

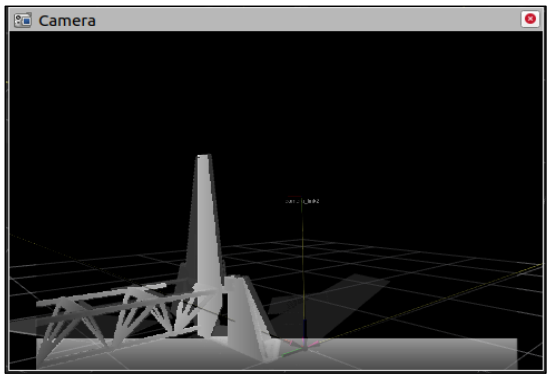


Şekil 4.8 Ortam-1'deki Kameralardan Elde Edilen Nokta Bulutlarının Rviz Görüntüsü

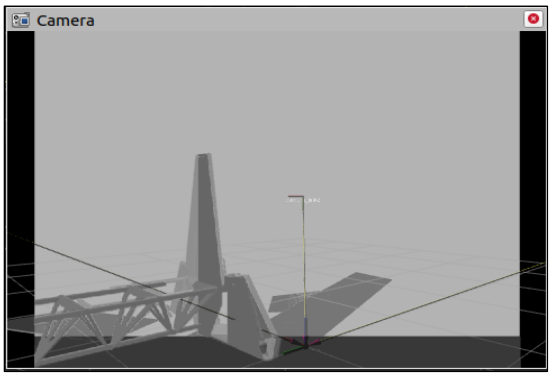
Ortam-2 'deki kameralardan alınan RGB ve derinlik görüntüleri ise Şekil 4.9' de verilmiştir.



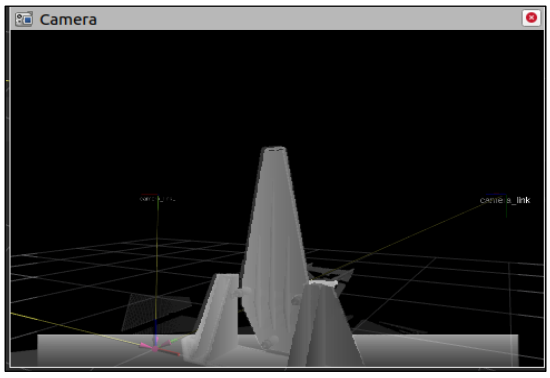
(a) **(b)**



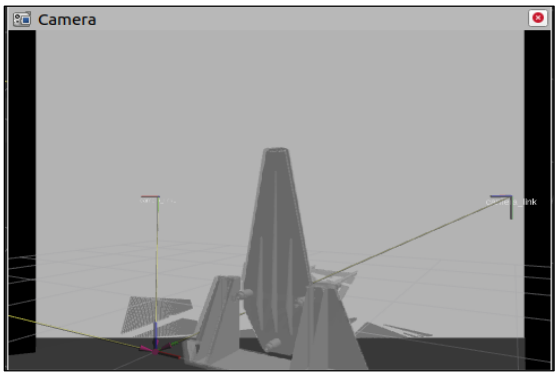
(c)



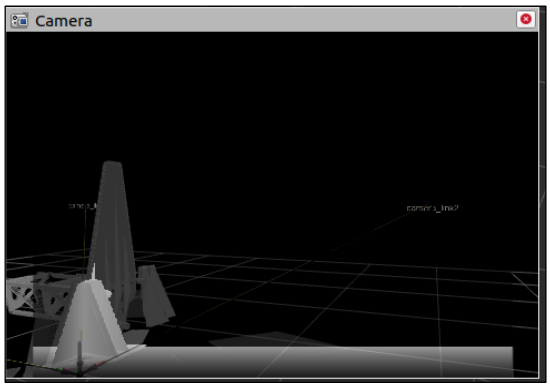
(d)



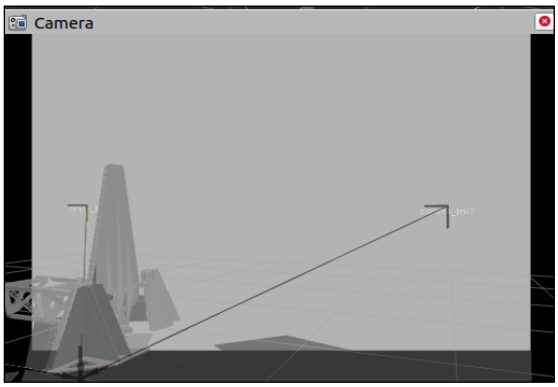
(e)



(f)



(g)

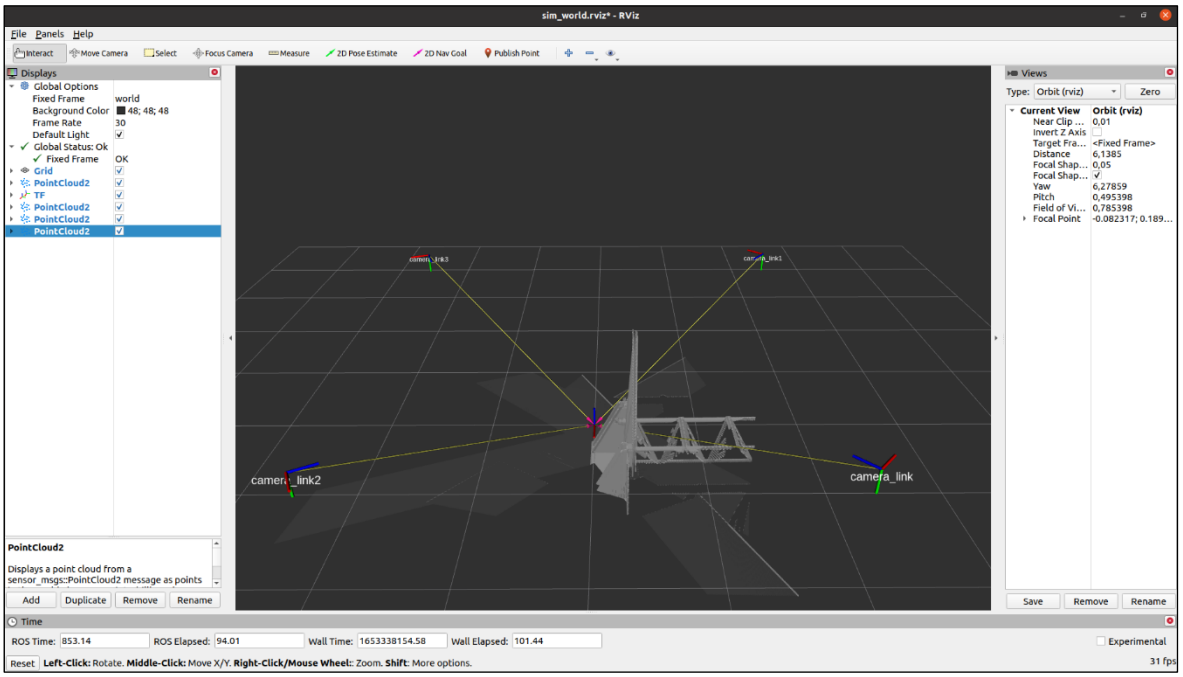


(h)

Şekil 4.9 Ortam-2' ye ait Kamera Görüntüleri **(a)** K-Depth, **(b)** K-RGB, **(c)** K1-Depth, **(d)** K1-RGB, **(e)** K2-Depth, **(f)** K2-RGB, **(g)** K3-Depth ve **(h)** K3-RGB

Şekil 4.9' da verilen görüntülerden (a),(c),(e) ve (g) ortamdan alınan derinlik görüntülerdir. (b),(d),(f) ve (h) RGB görüntüleridir. . Ayrıca ortamda kullanılan kameralar

sabittir. Hareketli sensör kullanılmamıştır. Ortam-2'deki dört kameradan elde edilen nokta bulutlarının Rviz görüntüsü Şekil 4.10' da gösterilmiştir.



Şekil 4.10 Ortam-2'deki Kameralardan Elde Edilen Nokta Bulutlarının Rviz Görüntüsü

4.4. İnanç Haritası Oluşturulması

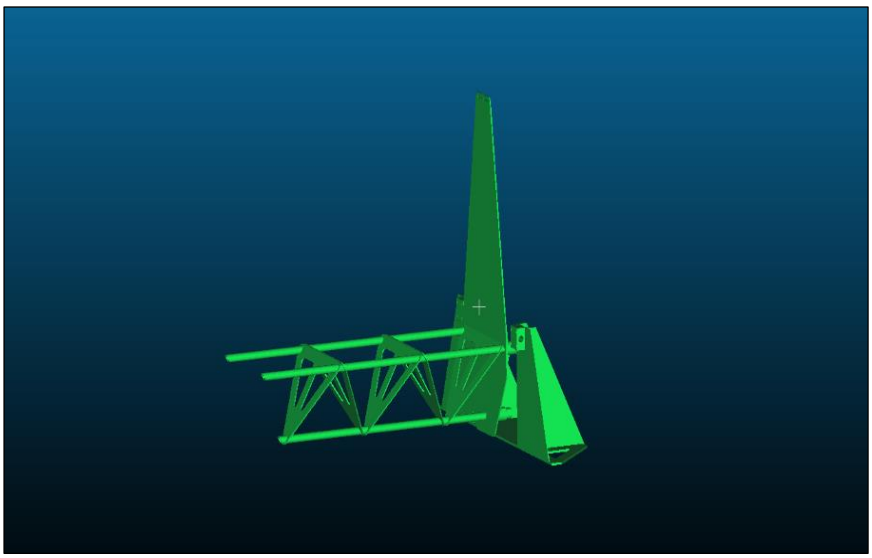
Bu çalışmada haritalama yapısı olarak Octomap kütüphanesi kullanıldı (Hornung vd., 2021). Çalışmada, simülasyon ortamına yapılar yerleştirilerek bu yapılara ait parçaların haritada işaretlenmesi sonucunda robotik denetim hedeflenmiştir. Bu parçalara ait üçgen örgü (mesh) modelleri kütüphaneye aktararak haritalama yapılır. Fakat kullanılan kütüphanenin veri yapısında bazı değişiklikler yapılması gerekir. Octomap kütüphanesi daha önceki bölümlerde de bahsedildiği gibi C++ için veri yapıları ve eşleme algoritmaları sunan bir sekizli ağaç tabanlı 3B haritalama yaklaşımıdır. Bu kütüphane yapısı çeşitli eklemelere ve düzenlemelere uygundur. Sınıf eklentisi ile istenilen veri yapısı, her bir düğüm ile eşlenebilmektedir. Böylece, her düğüm istenen içerikli veri yapısını da kapsamaktadır. Fakat bu çalışmada kütüphanede halihazırda mevcut olan ColorOcTreeNodes adlı soyut sınıftan türetilmiş ColorOcTree sınıfı kullanılmıştır. Bu sınıf her birisi 0-255 aralığında değer alabilen kırmızı (R), yeşil (G) ve mavi (B) kanallarında tanımlı renk bilgisini düğümler ile eşleyebilmektedir. Bu çalışmada, bu renk kanallarındaki değerler etiketleme amaçlı

kullanılmıştır. Her bir parçaya ait hacimler, Octomap haritasına taşındığında vokseller ile temsil edildiği için, her parçayı temsil eden vokseller için tek bir etiket ataması yapılacaktır. Böylece, haritada herhangi bir vokseli temsil eden düğümle eşleştirilmiş etiket, hangi parçaya ait olduğunu belirtecektir.

Çalışmada iki farklı simülasyon ortamı oluşturulmuştur ve ortamlara farklı sayıda parça içeren iki yapı yerleştirilmiştir. Ortamlarda kullanılan yapıların CloudCompare programında görüntülenmiş şekli Şekil 4.11' da gösterilmiştir.



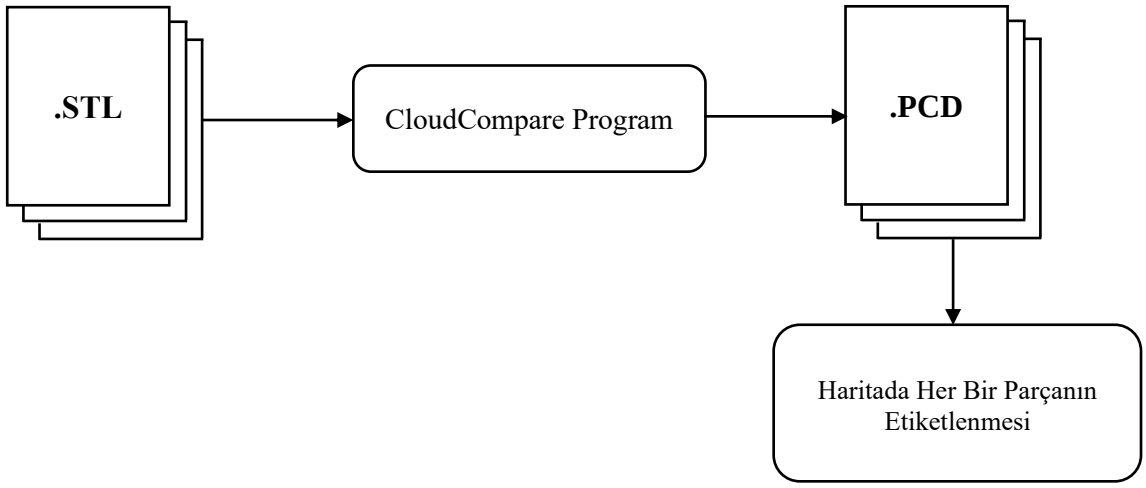
(a) Ortam-1 için Kullanılan Yapı



(b) Ortam-2 için Kullanılan Yapı

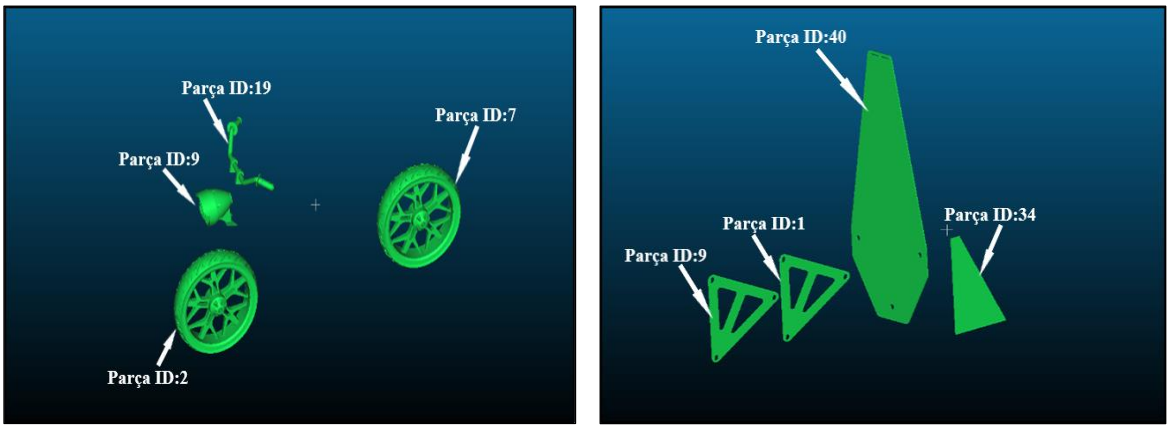
Şekil 4.11 Ortamlarda Kullanılan Yapılar

Bu yapılarda kullanılan parçaların modelleri, her birinin boyutu ve şekli farklı .stl uzantılı dosyalarda bulunmaktadır. Parçaların etiketlenmesi için Şekil 4.12' de gösterilen işlem basamakları tek tek uygulanmıştır.



Şekil 4.12 İnanç Haritası Oluşturulması İçin İşlem Basamakları

İlk olarak ortamda kullanılan yapılara ait üçgen örgü (mesh) modelleri .pcd formatına dönüştürülmesi gerekir. Bu çalışmada bu dönüşümü yapmak için CloudCompare programı kullanıldı. Nokta bulutuna dönüştürme süreci, iki farklı parametreye göre yapılabilmektedir. Birim alandaki nokta sayısı tanımlanabilir ya da toplam nokta sayısı tanımlanabilir. Bu çalışmada, birim alandaki nokta sayısı kullanılmıştır. Bu programa alternatif olarak diğer programlarda kullanılabilir. Ortamda kullanılan yapılara ait farklı 4 tane parça seçilerek bu dosyalar CloudCompare programıyla .pcd uzantılı dosyaya dönüştürüldü. Seçilen parçaların birbirine göre konumları CloudCompare programında ayrı ayrı eklenerek görüntülendiğinde, görüntüleri Şekil 4.13' deki gibidir.



Şekil 4.13 Seçilen 4 Farklı Parçanın CloudCompare Görüntüsü

Parçalara ait üçgen örgü (mesh) modelleri .stl uzantılı dosya formatından .pcd formatına dönüştürdükten sonra geliştirilen algoritma sayesinde, üçgen örgü (mesh) modellerinin etiketlemesi gerçekleştirildi. Kütüphane halihazırda, kırmızı (red-R), yeşil (green-G) ve mavi (blue-B) olmak üzere her biri 0-255 (8-bit) arasında değer alan 255x255x255 farklı renk, dolayısıyla etiket kodu elde edilebilecek şekilde etiketlenmiştir. Böylece, etiketlemede $I_R I_G I_B$ ile belirtilen 24 bitlik sayı kullanılabilir. Öyle ki, $I_R \in [0,255]$, $I_G \in [0,255]$ ve $I_B \in [0,255]$ olur. Çalışmada amaç parçaları renklendirmek değil, parçaları renklendirerek her bir parçaya farklı ID tanımlama ve bu sayede parçaların haritada yerlerinin belirlenmesini sağlamaktır. Parçalara farklı ID' ler tanımladıktan sonra tüm parçalar tek bir .ot uzantılı dosyaya kaydedilir. Bu dosya formatı kodlanmış tüm olasılıkları saklar. Bu çalışmada da parçaların hem doluluk hem de renk bilgisi kullanılacağı için .ot uzantılı dosyaya kaydedilmiştir. Kaydedilen .ot uzantılı dosya Octovis' te görüntülenerek test edilir.

Önerilen yöntemin algoritması Algoritma 4' de verilmektedir. Algoritmaya göre, $\mathcal{R}$ çözünürlüklü harita oluşturulmaktadır. Harita oluşturulurken ortamda bulunan parçalara ait üçgen örgü (mesh) modellerinin, nokta bulutuna dönüştürülmüş .pcd uzantılı dosyaları $\mathcal{F}_j (j = 1,2 \dots \dots \dots \mathcal{N})$olarak temsil edilmektedir ve $\mathcal{N}$ ortamda kullanılan parça sayısını temsil ediyor. Ortam-1 için parçalara ait dosya sayısı 31' dir ve Ortam-2 için parçalara ait dosya sayısı 52'dir. Ayrıca, bu parçaları renklendirmek için kütüphanenin color node kullanılmıştır. Parçalara ID tanımlayarak oluşturulan nihai harita $\mathcal{M}_B$ olarak ifade edilmiştir.

Algoritma 4. ID Tanımlama

Girdi: $\mathcal{F}_j(j = 1,2 \dots \dots \dots \mathcal{N}),\ \mathcal{R}$

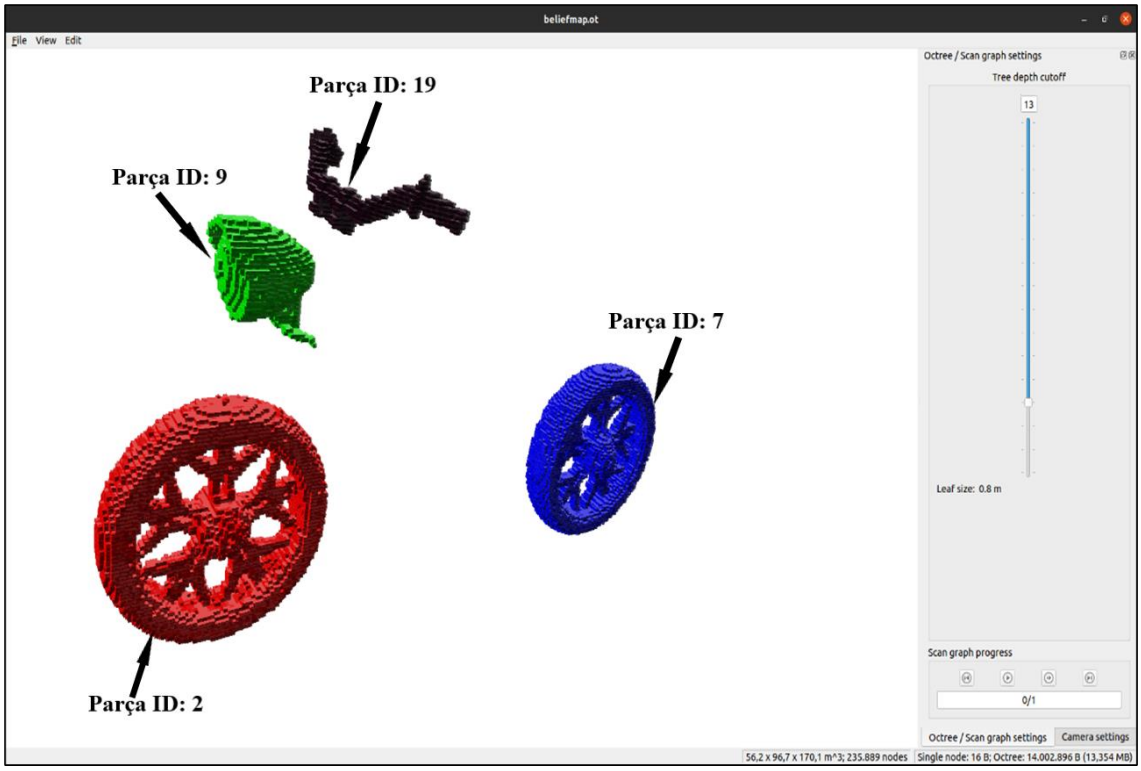
Çıktı: $\mathcal{M}_B$: Parçaların renklendirilerek yüklendiği son harita

$\mathcal{M}_B.initialize(\mathcal{R})$

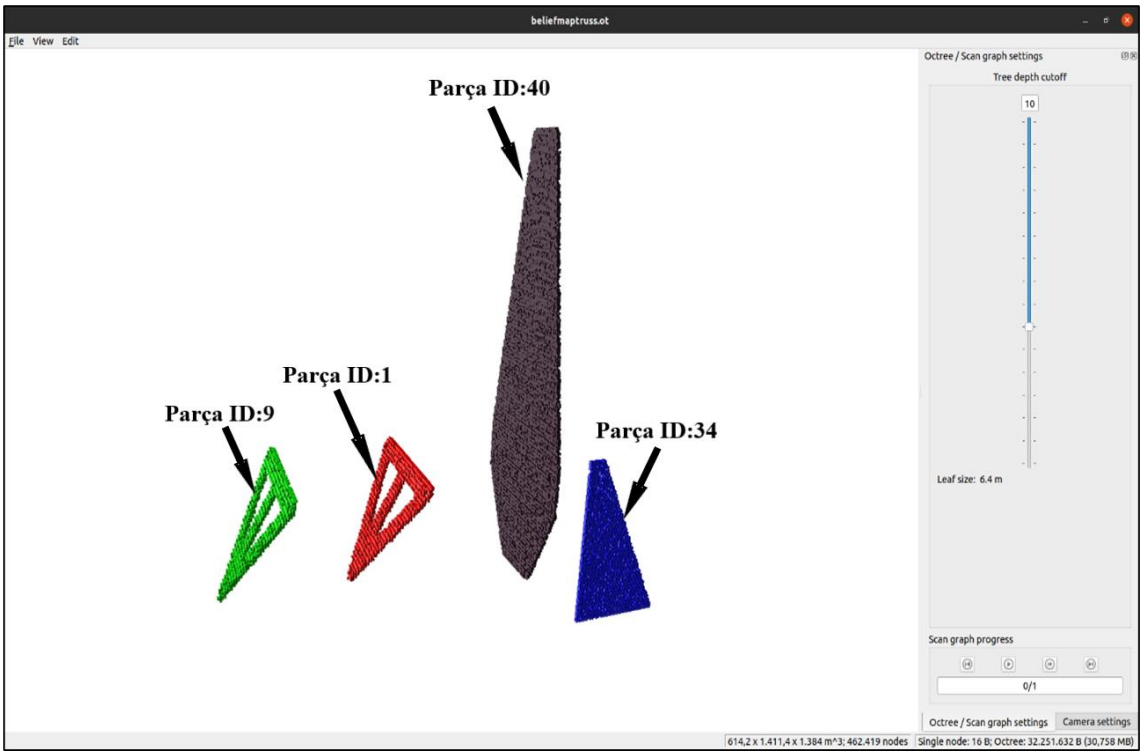
$I_R, I_G, I_B \leftarrow 0$

for $j := 1\ to\ \mathcal{N}$ **do**
    $\mathcal{P}$. reset ();
    $\mathcal{P}.loadPCDfile(\mathcal{F}_j)$
    for $\forall p \in \mathcal{P}$ **do**
        $\mathcal{M}_B$.updateNode (p_i)
        updateColor(I_R, I_G, I_B)
        I_R++
        if $I_R > 255$ **then**
        $I_R = 0$
        I_G++
            if $I_G > 255$ **then**
            $I_G = 0$
            I_B++
            endif
        endif
    endfor
endfor

Şekil 4.10' da gösterilen parçaların .pcd formatı elde edildikten sonra geliştirilen Algoritma 4 uygulanır. İlk aşamada her bir ortam için sadece seçilen dört parça üzerinden uygulama yapılmıştır. Seçilen dört parça için, Octovis görüntüsü Şekil 4.14' de görüldüğü gibidir. Bu parçaların programda işleme süresi parçaların düğüm (node) sayıları Çizelge 4.2' de gösterilmiştir.



(a) Ortam-1'de Kullanılan Yapıya Ait Parçaların Octovis Görüntüsü



(b) Ortam-2'de Kullanılan Yapıya Ait Parçaların Octovis Görüntüsü

Şekil 4.14 Ortamlarda Kullanılan Yapılara Ait Parçaların Üçgen Örgü (Mesh) Modellerinin Octovis Görüntüsü

Çizelge 4.2 İnanç Haritası Oluşturma İşlem Süreleri

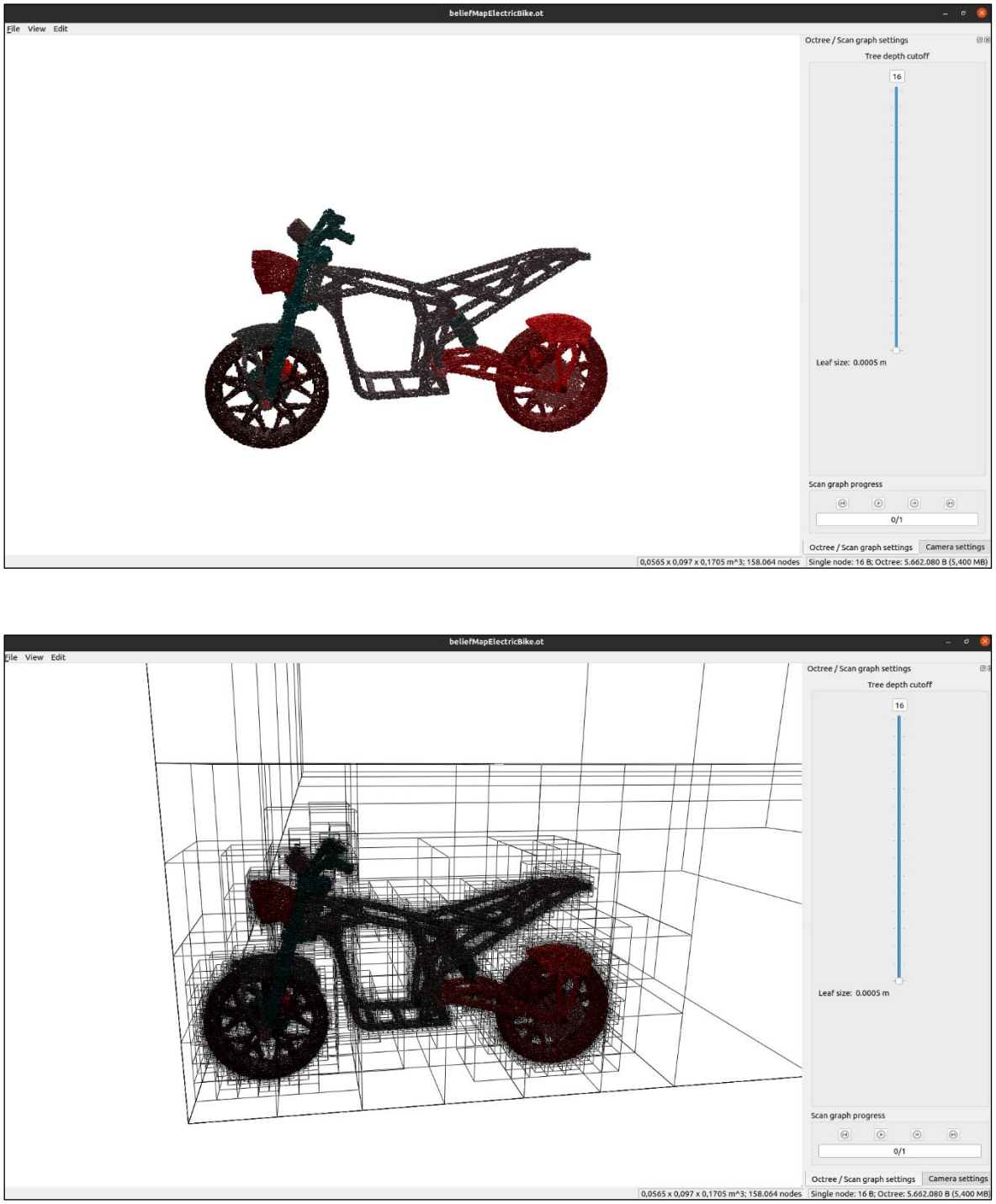
(a) Şekil 4.1 (a) ‘da Verilen Yapıdan Seçilen Parçalar

Parça ID Değerleri	***Nokta Sayısı***	***Düğüm (Node) Sayısı***	***Parçaların Kütüphaneye Aktarılma İşlem Süresi (s)***
2	31.434	95.977	7,71
7	31.353	95.703	7,65
9	9.093	27.739	7,61
19	5.521	16.497	7,12
		235.889	7,62

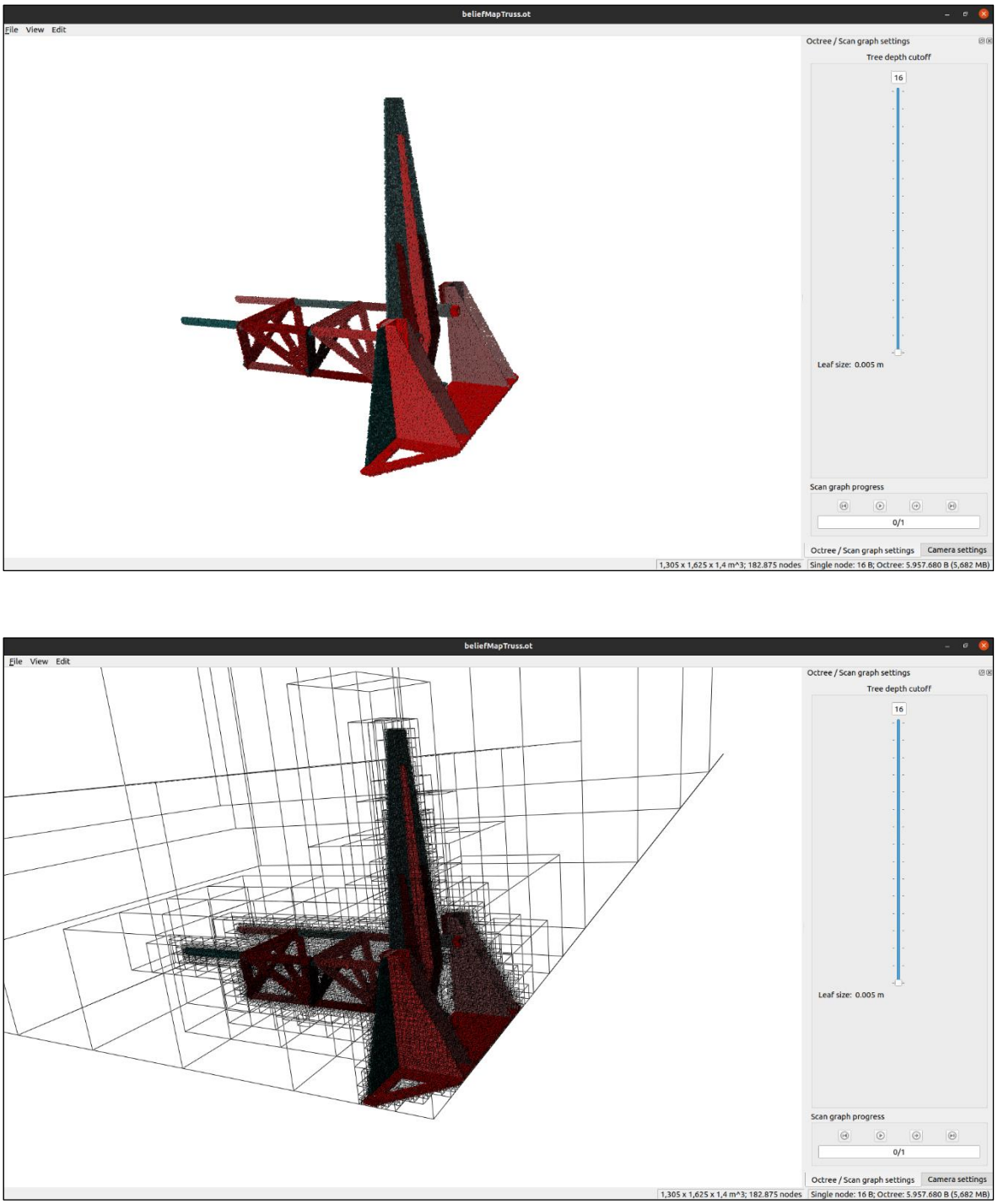
(b) Şekil 4.1 (b) ‘de Verilen Yapıdan Seçilen Parçalar

Parça ID Değerleri	***Nokta Sayısı***	***Düğüm (Node) Sayısı***	***Parçaların Kütüphaneye Aktarılma İşlem Süresi (s)***
1	8.130	51.345	7,82
9	8.125	51.196	7.58
34	9.015	56.578	7.35
40	48.849	303.320	7,51
		462.419	7,71

Bahsedilen tüm adımlar Ortam-1 ve Ortam-2’ de kullanılan yapılardaki tüm parçalar için de gerçekleştirilerek uygulama yapıldı. Ortamda kullanılan tüm parçalara farklı ID’ ler tanımlandı ve kütüphaneye aktarıldı. Bunun octovis görüntüsü Şekil 4.15‘deki gibidir.



(a) Ortam-1’ de Kullanılan Yapının Octovis Görüntüsü

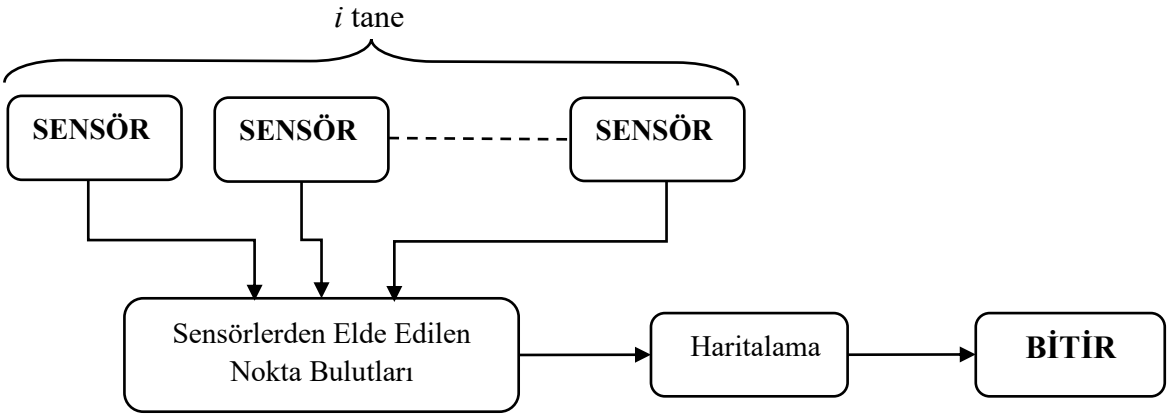


(b) Ortam-2' de Kullanılan Yapının Octovis Görüntüsü

Şekil 4.15 Yapıların Octovis Görüntüleri

4.5. Doluluk Haritasının Oluşturulması

Önceki bölümlerde de bahsedildiği gibi robotik denetlemede en önemli araçlar arasında sensörler vardır. Bu çalışmada, derinlik sensörleri kullanılmaktadır. Denetleme gerçekleştirilecek yapının algılanması için derinlik sensörlerinden faydalanılmaktadır. Sensör verileri, ızgara tabanlı doluluk haritalarının oluşmasını sağlamaktadır. Bu bölümde, Şekil 4.16' da gösterildiği gibi adımlar takip edilerek yapılan haritalamadan ve baz alınan yaklaşımdan bahsedilecektir.



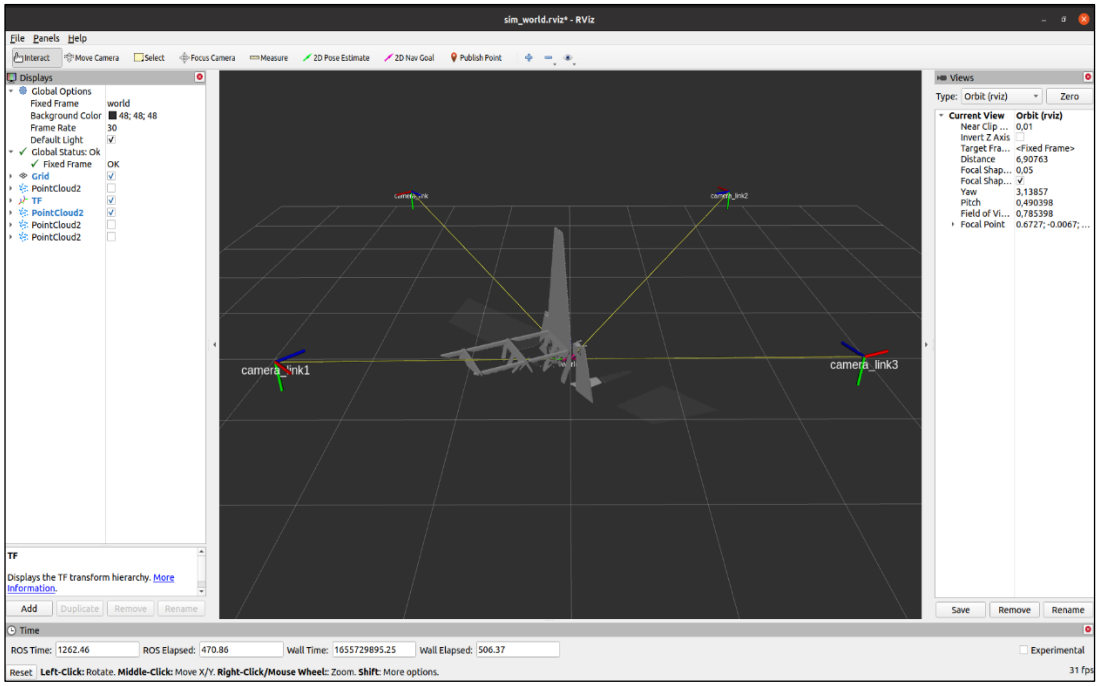
Şekil 4.16 Haritalama Akış Diyagramı

Yapılan haritalamada ortamda bulunan parçalara ait CAD modellerin nokta bulutu verileri kullanılıyor. Parçalara ait nokta bulutları derinlik kameraları aracılığıyla sağlanmaktadır. Nokta bulutları, $\{\mathcal{P}_1^S \cdots \mathcal{P}_{nf}^S \ \mathcal{P}_{nf+1}^S \cdots \mathcal{P}_{nm}^S\}$ ile ifade edilebilir. Burada, n_f ortama sabitlenmiş sensörlerin sayısı ve , n_m ise hareket halindeki sensörlerin sayısı olarak tanımlanmıştır. $\mathcal{P}_1^S$ ise, her bir sensörden sağlanan nokta bulutunu temsil etmektedir. Haritalama süreçlerinde, haritanın güncelleme periyotlarında her bir kameradan elde edilmiş nokta bulutları olarak değerlendirilebilir.

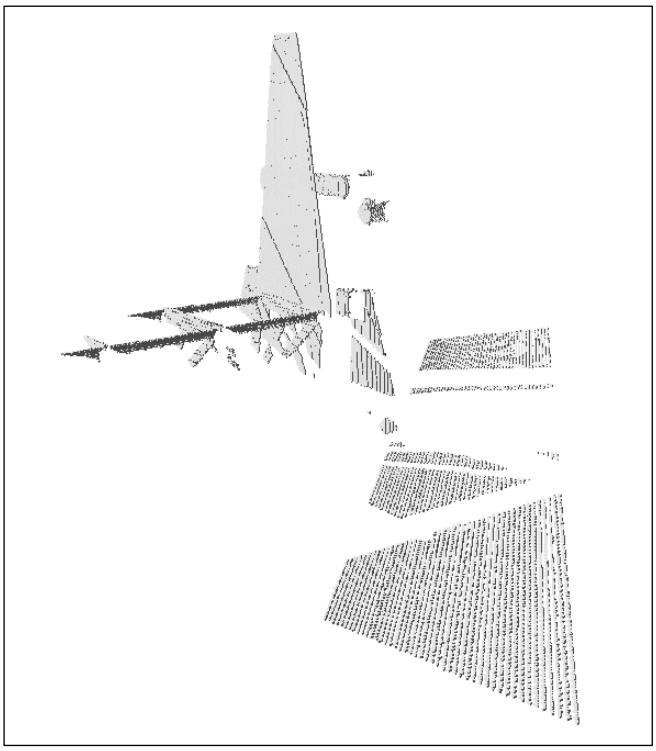
Algoritma 2'de verilen algoritma ile, sensörlerden sağlanan nokta bulutları $\{\mathcal{P}_1^S \cdots \mathcal{P}_{nf}^S \ \mathcal{P}_{nf+1}^S \cdots \mathcal{P}_{nm}^S\}$ haritadaki doluluk ızgaralarının değerlerinin oluşturulmasında kullanılmaktadır. Etiketleme ile elde edilen inanç haritası $\mathcal{M}_B$, sensör

verileri ile oluşturulan doluluk haritası $\mathcal{M}_O$ haline dönüşmektedir $\mathcal{M}_B \xrightarrow{mapping} \mathcal{M}_O$. $\mathcal{M}_O$, hem etiketlenmiş vokselleri hem de doluluk voksellerini kapsamaktadır.

Örnek olarak Ortam-2' de kullanılan kamera K1' den gelen nokta bulutlarının Rviz görüntüsü ve aynı kameradan gelen nokta bulutlarıyla oluşturulmuş doluluk haritasının Octovis görüntüsü Şekil 4.17 ve Şekil 4.18' de verilmiştir.



Şekil 4.17 Ortam-2'de Kullanılan Kamera K1'den Alınan Nokta Bulutlarının Rviz Görüntüsü



Şekil 4.18 Ortam-2'de Kullanılan Kamera K1'den Alınan Nokta Bulutlarıyla Oluşturulmuş Doluluk Haritası Görüntüsü

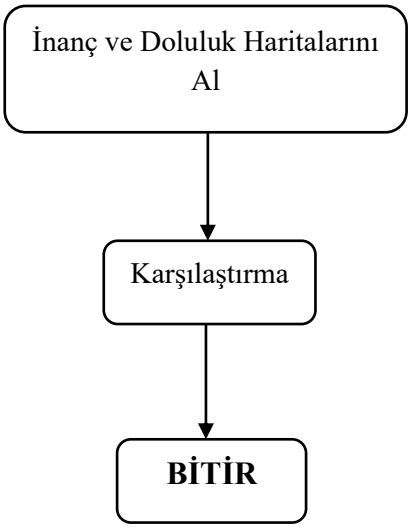
4.6. Denetleme

Bu bölümde denetleme sonuçları ve bu sonuçların nasıl elde edildiğinden bahsedilecektir. İlk olarak, doluluk olasılığı hesaplanırken tüm düğümlere tek tek bakılarak dolu olma durumu belirlenir. Tüm düğümlere bakıldığında renk bilgisine göre değerlendirme yapılır,

- Eğer renk bilgisi yok ama sensörden gelen bilgi dolu ise boş olarak işaretlenir.
- Eğer renk bilgisi var ise sensörden gelen bilgi ile karşılaştır her iki durumda da dolu ise haritaya eklenir.

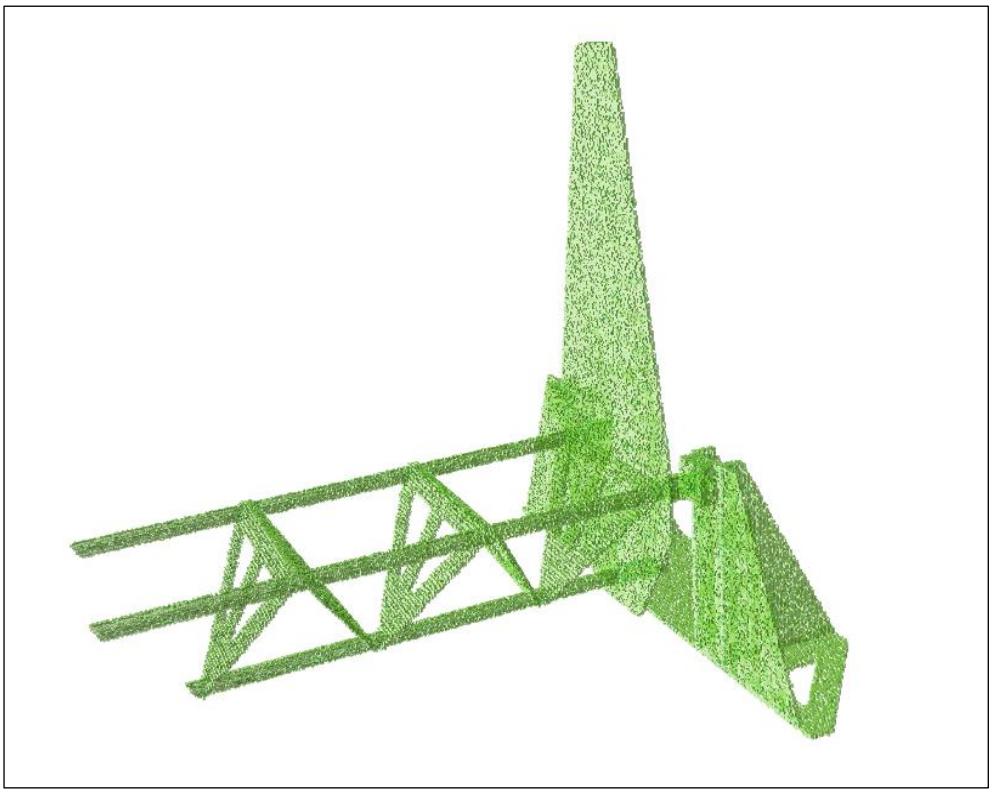
Bu yapılan işlem sonucunda doluluk haritası oluşturulurken aynı zamanda her bir parçanın doluluk olasılığı da hesaplanır. Haritadaki etiketleme amaçlı kullanılan veri α_i ve doluluk olasılığı ρ_i eşik üzerinde olan yaprakların sayıları elde edilmektedir. Aynı etiketli yapraklarda dolu olanların sayısı hesaplanarak parçanın bulunup bulunmadığı tespit

edilmektedir. Böylece parçanın ortamda bulunup bulunmadığı olasılıksal olarak da belirlenmiş olur.

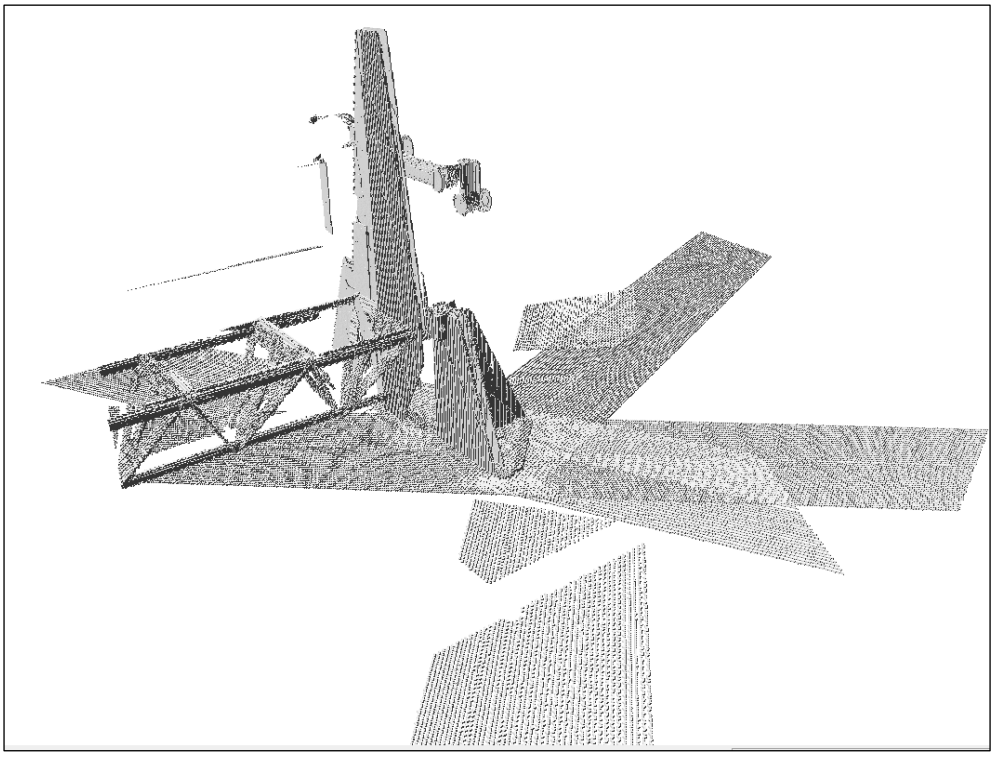


Şekil 4.19 Denetleme İşlem Basamakları

Denetleme yapılırken ilk olarak, Algoritma 1 uygulanarak dört kamera ile oluşturulmuş benzetim ortamında kullanılan parçaların inanç haritası (Şekil 4.20 (a)) elde edilir. Sonrasında Algoritma 2 uygulanarak doluluk haritası (Şekil 4.20 (b)) elde edilir. İnanç ve doluluk haritasının birleştirilerek düğümlerin doluluk olasılıkları hesaplanır.



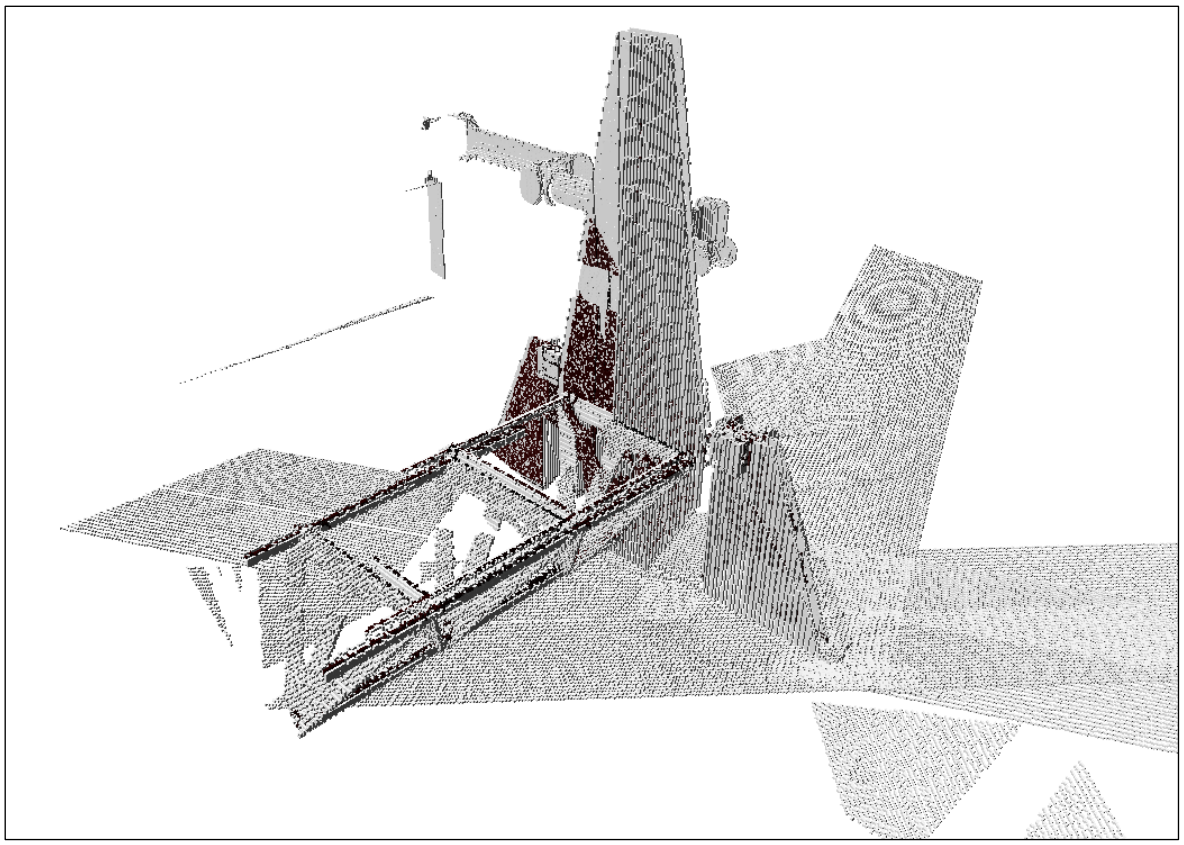
(a) Ortam-2’ de Kullanılan Yapının İnanç Haritası



(b) Ortam-2’ de Kullanılan Yapının 4 Kamera için Oluşturulmuş Doluluk Haritası

Şekil 4.20 Denetlemesi Yapılacak Yapının **(a)** İnanç Haritası ve **(b)** 4 Kamera için Oluşturulmuş Doluluk Haritası

İnanç ve doluluk haritası oluşturulduktan sonra inanç ve doluluk haritalarının birleştirilerek (Şekil 4.21) yukarıda bahsedildiği gibi her bir parça için ayrı ayrı tüm düğümlere bakılır ve doluluk olasılıkları hesaplanır. Bu adım için Algoritma 3 uygulanmıştır ve bu yapı için elde edilen denetleme sonuçları Çizelge 4.3' de verilmiştir.



Şekil 4.21 İnanç Haritası ve Dört Kamera için Oluşturulan Doluluk Haritasının Birlikte Görselleştirilmesi

İnanç haritasındaki toplam yaprak düğüm sayısı 139830 'dır. Ayrıca, doluluk haritasında 14337'si renkli 153511'si beyaz olmak üzere toplam 293348 yaprak düğüm vardır.

Çizelge 4.3 Dört Kamera için Denetim Sonuçları

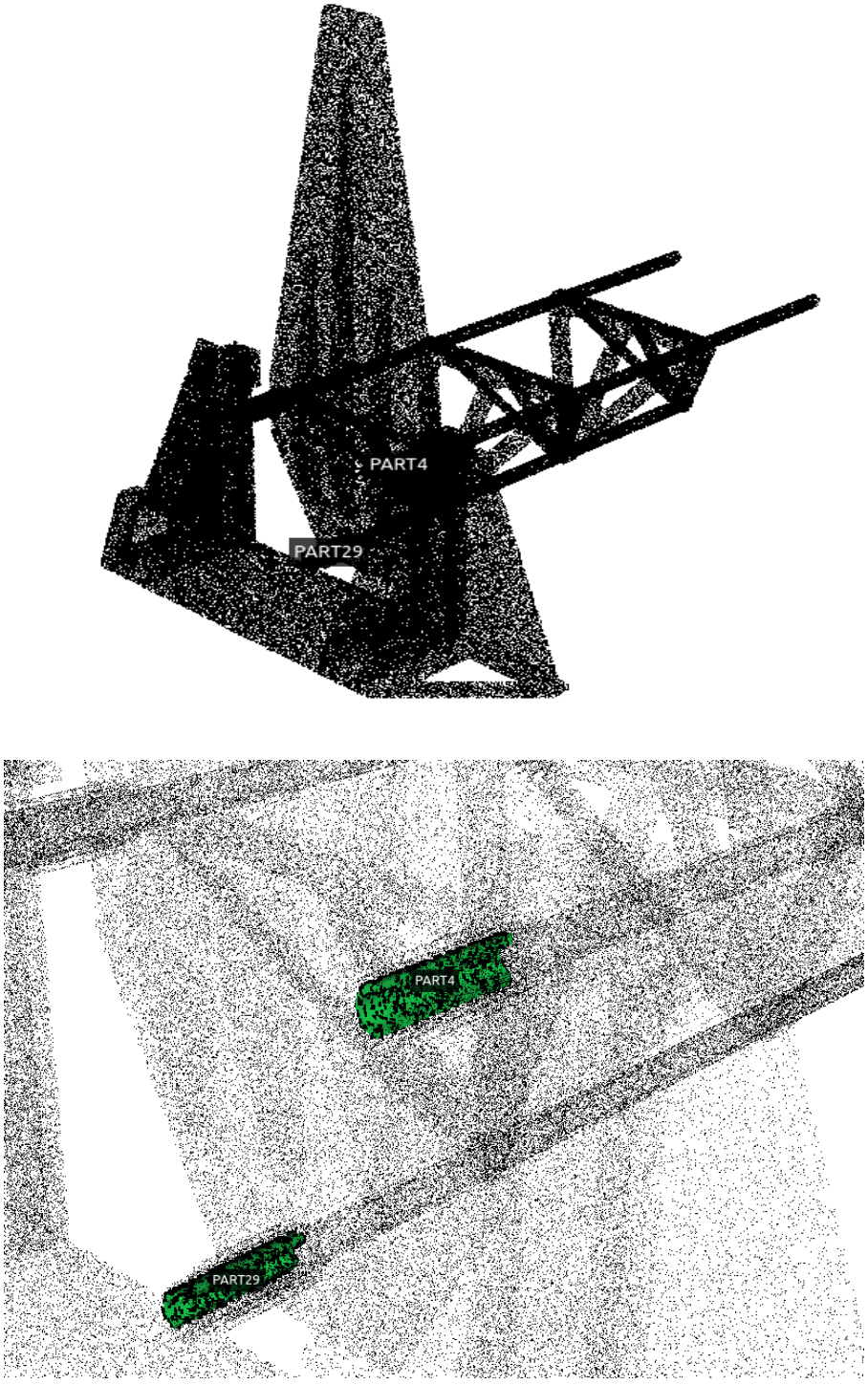
Parça ID Değerleri	***Doluluk Tahmini için Bakılan Toplam Düğüm Sayısı***	***Dolu Olarak İşaretlenen Düğüm Sayısı***	***Doluluk Olasılığı***	***Parçanın Ortamda Var Olma Durumu***
0	4648	71	%1	Mevcut
1	4667	63	%1	Mevcut
2	1918	67	%3	Mevcut
3	132	33	%25	Mevcut
4	184	0	%0	Eksik
5	621	9	%1	Mevcut
6	1928	117	%6	Mevcut
7	534	48	%8	Mevcut
8	1904	111	%5	Mevcut
9	4632	50	%1	Mevcut
10	1947	189	%9	Mevcut
11	539	63	%11	Mevcut
12	4685	51	%1	Mevcut
13	18111	302	%1	Mevcut
14	71	9	%12	Mevcut
15	520	18	%3	Mevcut
16	216	3	%1	Mevcut
17	4075	580	%14	Mevcut
18	4310	631	%14	Mevcut
19	157	11	%7	Mevcut
20	155	13	%8	Mevcut
21	575	59	%10	Mevcut
22	4046	585	%14	Mevcut
23	78	25	%32	Mevcut
24	890	71	%7	Mevcut
25	5131	155	%3	Mevcut

Parça ID Değerleri	***Doluluk Tahmini için Bakılan Toplam Düğüm Sayısı***	***Dolu Olarak İşaretlenen Düğüm Sayısı***	***Doluluk Olasılığı***	***Parçanın Ortamda Var Olma Durumu***
26	649	132	%20	Mevcut
27	1892	100	%5	Mevcut
28	1906	110	%5	Mevcut
29	213	0	%0	Eksik
30	399	14	%3	Mevcut
31	4723	130	%2	Mevcut
32	1972	180	%9	Mevcut
33	3314	241	%7	Mevcut
34	5136	406	%7	Mevcut
35	413	20	%4	Mevcut
36	158	5	%3	Mevcut
37	164	19	%11	Mevcut
38	1953	115	%5	Mevcut
39	795	15	%1	Mevcut
40	27685	7056	%25	Mevcut
41	1931	130	%6	Mevcut
42	959	24	%2	Mevcut
43	151	23	%15	Mevcut
44	985	57	%5	Mevcut
45	155	15	%9	Mevcut
46	5063	33	%0	Mevcut
47	5124	5	%0	Mevcut
48	647	47	%7	Mevcut
49	1918	27	%1	Mevcut
50	397	13	%3	Mevcut
51	4454	2086	%46	Mevcut

5. BULGULAR VE TARTIŞMA

Önerilen yöntem farklı boyutlarda parçalar içeren yapı kullanılarak ve benzetim ortamı oluşturularak test edilmiştir. Ortamda kullanılan yapıları oluşturan parçalardan montajı yapılmamış olanların tespit edilmesi hedeflenmektedir. Bu hedef doğrultunda yapılan çalışma sonucu oluşturulan inanç haritası ve doluluk haritası sonuçları değerlendirilerek doluluk haritasında hem renk hem de doluluk bilgisinin var olduğu uygulama sonucu test edilmiştir.

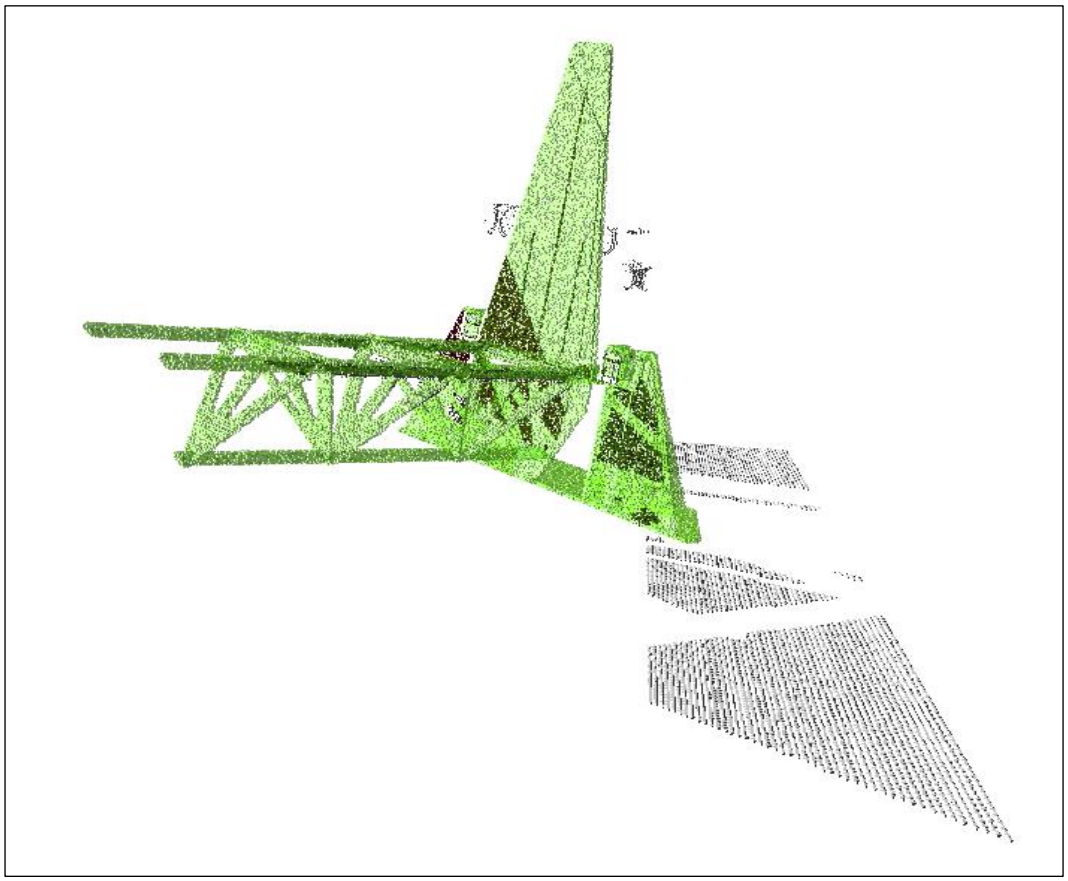
Elde edilen denetleme sonuçları Çizelge 4.3’ de verilmiştir. Genel olarak sonuçlar incelendiğinde yapılan çalışma sayesinde inanç haritası oluşturularak her parçanın ortamdaki yeri işaretlenmiştir. Doluluk haritası ile de kameralardan alınan verilerle parçaların ortamda var olup olmadığı olasılıksal olarak hesaplanmıştır. Çizelge 4.3’e baktığımızda iki tane parçanın ortamda olmadığını doluluk olasılıklarına bakarak söyleyebiliriz. Bu parçalardan birinin ID’ si 4 diğerinin 29 olduğu görülmektedir. Bu parçalarda dolu olarak işaretlenen düğüm içermediği denetleme sonuçlarında açıkça görülmektedir. Şekil 5.1’ de eksik olan parçaların haritadaki yeri gösterilmektedir. Ayrıca, doluluk olasılığı 0 olup ortamda mevcut olan ID değerleri 46 ve 47 olan parçalarda mevcuttur. Bunların ortamda mevcut olma durumu çizelgede de görüldüğü gibi dolu olarak işaretlenmiş düğümler içermelerinden kaynaklanmaktadır.



Şekil 5.1 Eksik Parçaların Birleştirilmiş Haritada Olması Gerektiği Yerler

Yapılan bu çalışmada ortamda kullanılan kamera sayısı doluluk olasılığını doğrudan etkiler. Bu çalışmada dört kamera kullanılmıştır. Fakat, bu farkı gözlemlemek içinde tek kameradan da denetim sonuçları elde edilmiştir. Tek kameradan alınan nokta bulutlarıyla oluşturulan doluluk haritası Şekil 4.18’ de görüldüğü gibidir. Ayrıca, Şekil 5.2’ de inanç

haritası ile bir kamera için oluşturulmuş doluluk haritasının birlikte görselleştirilmesi verilmiştir. Tek kamera için denetim sonuçları Çizelge 5.1’ de verilmiştir.



Şekil 5.2 İnanç Haritası ve Bir Kamera için Oluşturulan Doluluk Haritasının Birlikte Görselleştirilmesi

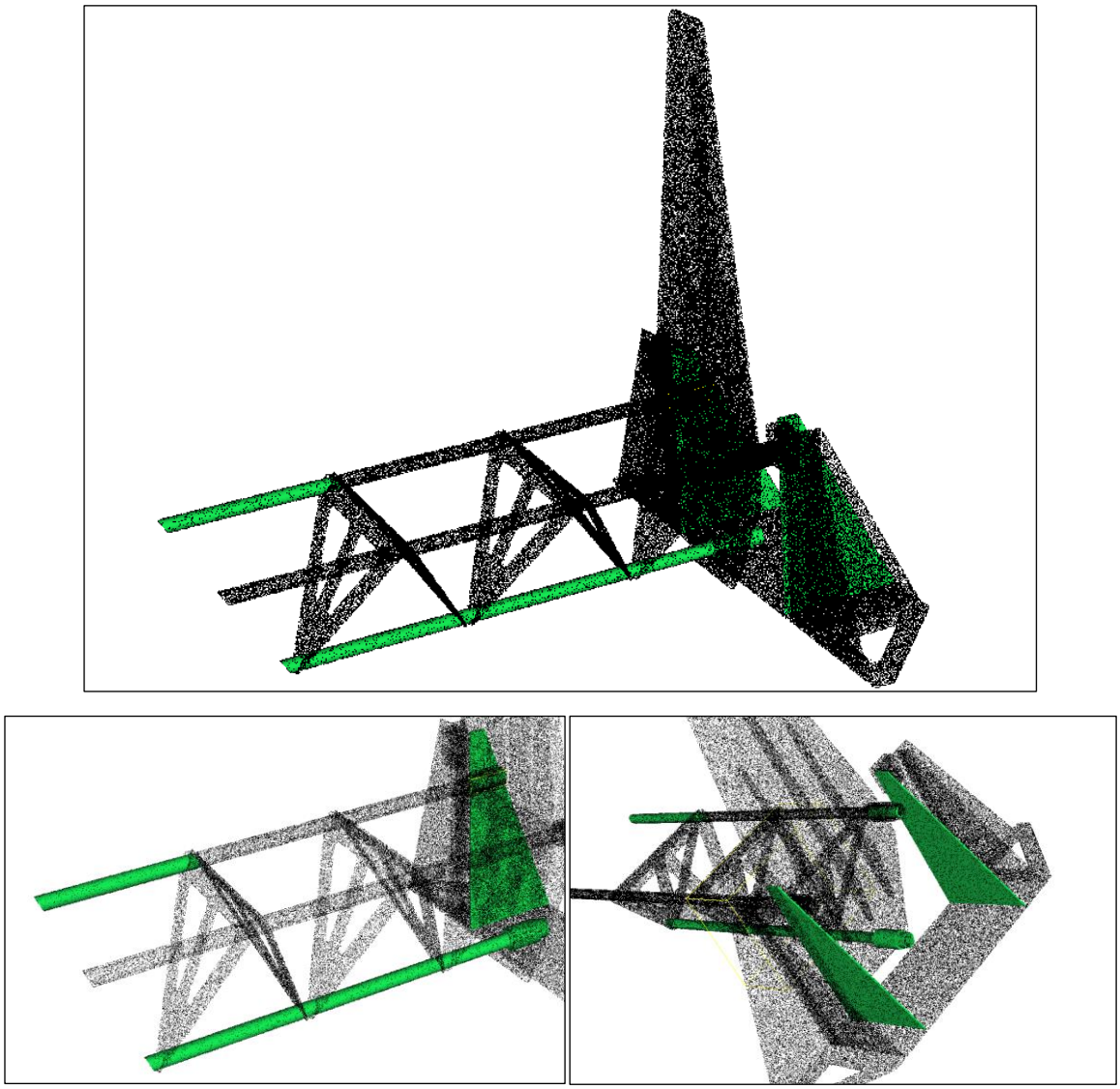
İnanç haritasındaki toplam yaprak düğüm sayısı yapı aynı olduğu için aynıdır. Ayrıca, doluluk haritasında 5786’sı renkli 37732’ si beyaz olmak üzere toplam 177562 yaprak düğüm vardır.

Çizelge 5.1 Bir Kamera için Denetim Sonuçları

Parça ID Değerleri	***Doluluk Tahmini için Bakılan Toplam Düğüm Sayısı***	***Dolu Olarak İşaretlenen Düğüm Sayısı***	***Doluluk Olasılığı***	***Parçanın Ortamda Var Olma Durumu***
0	4648	14	%0	Mevcut
1	4667	23	%0	Mevcut
2	1918	67	%3	Mevcut
3	132	20	%15	Mevcut
4	184	0	%0	Eksik
5	621	0	%0	Eksik
6	1928	117	%6	Mevcut
7	534	46	%8	Mevcut
8	1904	0	%0	Eksik
9	4632	14	%0	Mevcut
10	1947	104	%5	Mevcut
11	539	38	%7	Mevcut
12	4685	32	%0	Mevcut
13	18111	7	%0	Mevcut
14	71	5	%7	Mevcut
15	520	4	%0	Mevcut
16	216	2	%0	Mevcut
17	4075	342	%8	Mevcut
18	4310	631	%14	Mevcut
19	157	0	%0	Eksik
20	155	0	%0	Eksik
21	575	16	%2	Mevcut
22	4046	386	%9	Mevcut
23	78	16	%20	Mevcut
24	890	22	%2	Mevcut
25	5131	0	%0	Eksik

Parça ID Değerleri	***Doluluk Tahmini için Bakılan Toplam Düğüm Sayısı***	***Dolu Olarak İşaretlenen Düğüm Sayısı***	***Doluluk Olasılığı***	***Parçanın Ortamda Var Olma Durumu***
26	649	13	%2	Mevcut
27	1892	49	%2	Mevcut
28	1906	0	%0	Eksik
29	213	0	%0	Eksik
30	399	5	%1	Mevcut
31	4723	107	%2	Mevcut
32	1972	53	%2	Mevcut
33	3314	241	%7	Mevcut
34	5136	61	%1	Mevcut
35	413	8	%1	Mevcut
36	158	0	%0	Eksik
37	164	8	%4	Mevcut
38	1953	38	%1	Mevcut
39	795	0	%0	Eksik
40	27685	2515	%9	Mevcut
41	1931	28	%1	Mevcut
42	959	13	%1	Mevcut
43	151	12	%7	Mevcut
44	985	0	%0	Eksik
45	155	0	%0	Eksik
46	5063	31	%0	Mevcut
47	5124	0	%0	Eksik
48	647	8	%1	Mevcut
49	1918	0	%0	Eksik
50	397	0	%0	Eksik
51	4454	690	%15	Mevcut

Çizelge 5.1 incelendiğinde birçok parçanın eksik olduğu tespit edilmiştir. Bu parçaların haritadaki yerleri Şekil 5.3’ de gösterilmiştir.



Şekil 5.3 Bir Kamera için Eksik Parçaların Birleştirilmiş Haritada Olması Gerektiği Yerler

Dört kameradan elde edilen denetleme sonuçları ile bir kameradan elde edilen denetleme sonuçları karşılaştırıldığında, kamera sayısının artırılması ile daha hassas denetleme gerçekleştirileceğini söyleyebiliriz. Çizelgelerde de görülebileceği gibi doluluk olasılığı kameralardan gelen verilere göre elde edilir ve kamera sayısı arttıkça doluluk olasılığı da artar.

Bu çalışma yapılırken işaretlenmiş voksellerin sensörlerden gelen verilerle oluşturulan haritalama sonucunda karşılaştırma yapıldığında bir sorunla karşılaşıldı ve oluşturulan iki haritanın tam olarak eşleşmediği fark edildi. Bu durum, nokta yoğunluğundan ya da kalibrasyon hatasından kaynaklı olabileceği gözlemlenmiştir.

- *Nokta Yoğunluğu:* Sensörlerden gelen nokta yoğunluğu ile etiketlenen parçaların nokta yoğunluğundan kaynaklı eşleşmeme durumu ortaya çıkar.

- *Kalibrasyon Hatası:* Voksel boyutu büyütülürse kalibrasyon hatası yok edilebilir. Fakat bu defada iç içe geçmiş parçaların renk bilgisinde hata meydana gelebilir.

Bu durumlar göz önünde bulundurularak değerlendirme yapıldığında da çözünürlüğün yüksek tutulma durumunun da bazı dezavantajları meydana gelir. Eğer parça yeterince büyükse sorun yok ama küçükse küçük parçalar yok sayılacak. Böylece etiketleme işlemi hassas bir şekilde yapılmamış olup ve denetleme hataları meydana gelecektir.

6. SONUÇ VE ÖNERİLER

Üretimde robot kullanımı yaygın olduğu gibi son zamanlarda denetimin de robotlar tarafından yapılması oldukça önemli bir yere sahiptir. Bu çalışma ile üretimde çok sayıda parçanın birbirine monte edilmesiyle oluşturulan yapıların, kalite denetimlerini otomatik gerçekleştiren bir yöntem önerilmiştir. Önerilen yöntem iki farklı yapıya ait parçalar kullanılarak ve iki farklı simülasyon ortamı oluşturularak test edilmiştir. Ortamda kullanılan yapıları oluşturan parçalardan montajı yapılmamış olanların tespit edilmesi hedeflenmektedir.

Önerilen yöntem üç kısımdan oluşmaktadır. Birinci kısım, yapıyı oluşturan denetimi yapılacak parçalar ızgara tabanlı haritada etiketlenir. Bu etiketleme ile inanç haritası adı verilen bir harita oluşturulur. İkinci kısımda, ortamda sabitlenmiş derinlik kameralarından alınan nokta bulutları ile inanç haritası üzerinde doluluk haritalaması yapılır. Son kısımda, inanç harita etiketleri ile ızgara doluluk bilgileri arasındaki ilişkiler değerlendirilmektedir. Bu değerlendirme sonucunda yapıya monte edilmeyen parçalar belirlenir. Yöntemin etkinliği, simüle edilmiş bir ortamda gerçekleştirilen deneysel çalışmalarla kanıtlanmıştır.

Yapılan deneysel çalışmalar ile kamera verisi arttıkça doluluk olasılığının da arttığı gözlemlenmiştir. Böylece ilerleyen çalışmalarda ortamda hareketli algılayıcılar kullanılarak daha hassas denetleme yapılması hedeflenebilir.

KAYNAKLAR DİZİNİ

Al Khawli, T., Anwar, M., Gan, D., Islam, S., 2020, Integrating Laser Profile Sensor to An Industrial Robotic Arm for Improving Quality Inspection in Manufacturing Processes, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, Vol. 235, p. 4-17.

Amanatides, J., Woo, A., 1987, A Fast Voxel Traversal Algorithm for Ray Tracing, Proceedings of Euro Graphics,Vol. 87,

Anonim1, 2020, ROS Installation Options, http://wiki.ros.org/ROS/Installation, erişim tarihi: 29.05.2021.

Anonim2, 2014, GAZEBO, http://gazebosim.org/, erişim tarihi: 06.01.2022

Anonim3, 2018, Rviz, http://wiki.ros.org/rviz, erişim tarihi: 25.03.2022

Anonim4, 2017,Octovis, http://wiki.ros.org/octovis, erişim tarihi: 21.05.2022

Choi, H.R., Ryew, S.M., 2002, Robotic System with Active Steering Capability for Internal Inspection of Urban Gas Pipelines, Mechatronics, Vol. 12, p. 713-736

Duberg, D., Jensfelt, P., 2020, UFOMap: An Efficient Probabilistic 3D Mapping Framework That Embraces the Unknown, IEEE Robotics and Automation Letters, Vol.5, No.4,p. 6411-6418

Elfes, A., 1989, Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation, PhD Thesis, Carnegie Mellon University, Pensylvania, USA.

Endress, F., Hess, J., Sturm, J., Cremers, D., Burgard, W., 2014, 3-D Mapping With an RGB-D Camera, IEEE Transactions on Robotics, Vol.30, No. 1,p. 177-187

KAYNAKLAR DİZİNİ (Devamı)

Gregorio, D.D., Stefano, L.D, 2017, SkiMap: An Efficient Mapping Framework for Robot Navigation, IEEE International Conference on Robotics and Automation (ICRA), Singapore, p. 2569-2576

Hornung, A., Wurm, K.M., Bennewitz, M., Stachniss, C., Burgard, W., 2012, Octomap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees, Autonomous Robots, Vol. 34, p. 189-206.

Hornung, A., 2021, OctoMap - An Efficient Probabilistic 3D Mapping Framework Based on Octrees, https://github.com/OctoMap/octomap, erişim tarihi:29.05.2021.

Jadidi, M.G., Gan, L., Parkison, S.A., Li, J., Eustice, R.M., 2017, Gaussian Processes Semantic Map Representation, ArXiv:1707.01532v1,

Khan, A., Mineo, C., Dobie, G., Macleod, C., Pierce, G., 2020, Vision Guided Robotic Inpection for Parts in Manufacturing and Remanufacturing Industry, Journal of Remanufacturing, Vol. 11, p. 49-70.

Lee, E., Lee, S., 2016, Development of Underwater Terrain's Depth Map Representation Method based on Occupancy Grids with 3D Point Cloud from Polar Sonar Sensor System, 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Xian, China, p. 497-500.

Liu, Y., Zhao, W., Lutz, T., Yue, X., 2021, Task Allocation and Coordinated Motion Planning for Autonomous Multi-Robot Optical Inspection Systems, arXiv:2106.08164,

Loupos, K., Doulamis, A.D., Stentoumis, C., Protopapadaksi, E., Makantasis, K., vd.,2017, Autonomous Robotic System for Tunnel Structural Inspection and Assessment, International Journal of Intelligent Robotics and Applications, Vol. 2, p. 43-66.

KAYNAKLAR DİZİNİ (Devamı)

Meagher, D., 1982, Geometric Modeling Using Octree Encoding, Computer Graphics and Image Processing,Vol. 19, No. 2,p. 129–147

Mineo, C., Pierce, S.G., Wright, B., Cooper, I., Nicholson, P.I., 2015, PAUT Inspection of Complex-Shaped Composite Materials Through Six DOFs Robotic Manipulators, Insight - Non-Destructive Testing and Condition Monitoring, Vol. 57, No. 3, p. 161-166

Miseikis, J., Glette, K., Elle, O.J.,Torresen, J., 2016, Multi 3D Camera Mapping for Predictive and Reflexive Robot Manipulator Trajectory Estimation, Proceedings of 2016 IEEE Symposium Series on Computational Intelligence (SSCI), p. 1-8.

Misnan, M.F., Arshad, N.M., Razak, N.A., 2012, Construction Sonar Sensor Model of Low Altitude Field Mapping Sensors for Application on a UAV, IEEE 8th International Colloquium on Signal Processing and its Applications, p. 446-450.

Moravec, P.H.,1996, Robot Spatial Perception by Stereoscopic Visionand 3D Evidence Grids, The Robotics Institute Carnegie Mellon University Pittsburgh, Pennsylvania

Özkan, M., Yazıcı, A., 2021, Otonom Robot Uygulamaları, Nobel Yayınevi, s. 51-82.

Sun, L., Yan, Z., Zaganidis, A., Zhao, C., Duckett, T., 2018, Recurrent-OctoMap: Learning State-Based Map Refinement for Long-Term Semantic Mapping With 3-D-Lidar Data, IEEE Robotics and Automation Letters, Vol. 3, No. 4, p. 3749-3756.

KAYNAKLAR DİZİNİ (Devamı)

Vithanage, R.K.W., Harrison, C.S., De Silva, A.K.M., 2019, Autonomous Rolling-Stock Coupler Inspection Using Industrial Robots, Robotics and Computer Integrated Manufacturing, Vol. 59, p. 82-91

Wikipedia, 2021, Robot Operating System, https://en.wikipedia.org/wiki/Robot_Operating_System, erişim tarihi: 06.01.2022

Wikipedia, 2021, Occupancy Grid Mapping, https://en.wikipedia.org/wiki/Occupancy_grid_mapping, erişim tarihi: 23.12.2021

Wilhelms, J.,Van Gelder, A., 1992, Octrees for Faster Isosurface Generation. ACM Transactions on Graphics, Vol. 11, No.3, p. 201–227.

Yue, Y., Senarathne, P. G. C. N., Yang, C., Zhang, J., Wen, M., vd., 2018, Hierarchical Probabilistic Fusion Framework for Matching and Merging of 3-D Occupancy Maps, IEEE Sensors Journal, Vol. 18, No. 21, p. 8933-8949.

Zhang, L., Wei, L., Shen, P., Wei, W., Zhu, G., vd., 2018, Semantic SLAM Based on Object Detection and Improved Octomap, IEEE Access, Vol. 6, p. 75545-75559.