Bilinmeyen 3B Nesnelerin Endüstriyel Robot Kolu İle Otonom Yüzey Modellemesi

Cansu Söyleyici

YÜKSEK LİSANS TEZİ

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Ağustos 2017

Autonomous Surface Modeling Of Unknown 3D Objects With An Industrial Robot Manipulator

Cansu Söyleyici

MASTER OF SCIENCE THESIS

Electrical and Electronics Engineering Department

August 2017

Bilinmeyen 3B Nesnelerin Endüstriyel Robot Kolu İle Otonom Yüzey Modellemesi

Cansu Söyleyici

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

Danışman: Doç. Dr. Metin Özkan

Tübitak 1001-Bilimsel ve Teknolojik Araştırma Projelerini Destekleme Programı tarafından 115E374 numaralı proje kapsamında desteklenmiştir.

Ağustos 2017

ONAY

Elektrik-Elektronik Mühendisliği Anabilim Dalı YÜKSEK LİSANS öğrencisi Cansu Söyleyici'ın YÜKSEK LİSANS tezi olarak hazırladığı "**Bilinmeyen 3B Nesnelerin Endüstriyel Robot Kolu İle Otonom Yüzey Modellemesi**" başlıklı bu çalışma, jürimizce lisansüstü yönetmeliğin ilgili maddeleri uyarınca değerlendirilerek oy birliği ile kabul edilmiştir.

Danışman : Doç. Dr. Metin Özkan

İkinci Danışman : -

Yüksek Lisans Tez Savunma Jürisi:

Üye : Doç. Dr. Metin Özkan

Üye : Doç. Dr. Serkan Günal

Üye : Yrd. Doç.Dr. Helin Dutağacı

Üye : -

Üye : -

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..... tarih ve ..... sayılı kararıyla onaylanmıştır.

Prof.Dr. Hürriyet ERŞAHAN
Enstitü Müdürü

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Metin Özkan danışmanlığında hazırlamış olduğum "**Bilinmeyen 3B Nesnelerin Endüstriyel Robot Kolu İle Otonom Yüzey Modellemesi**" başlıklı tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallara uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel etik ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. …/…/20…

Cansu Söyleyici

ÖZET

Günümüzde, gelişen teknolojiyle beraber nesnelerin 3B modellemesi mümkün hale gelmiş; endüstri, medikal, eğlence ve kültürel mirasların korunması gibi geniş uygulama alanlarına yayılmıştır. Bahsedilen bu kullanım alanları, yüksek kalitede ve doğrulukta 3B modellerin elde edilmesi ihtiyacını doğurmuştur.

Cisimlerin 3B modellemesinde, cismin yüzeyinden elde edilen derinlik ve/veya renk bilgisi kullanılmaktadır. Bu bilgi, algılayıcılar vasıtasıyla cismin tüm yüzeyinden veri toplanarak sağlanmaktadır. Veri toplama amacıyla, algılayıcının nesne yüzeyi boyunca dolaştırılmasına yüzey tarama denilebilir. Nesne yüzeyinden veri toplama süreci, elle ve otonom olmak üzere iki şekilde gerçekleştirilebilmektedir. Elle tarama, bu alanda tecrübeli ve yetenekli bir personel gerektirmektedir; çünkü karmaşık nesnelerin hangi açılardan taranacağı ve sonraki tarama noktasının ne olacağını belirlemek kolay bir iş değildir. Bu durumda, insan kaynaklı hatalar tamamlanmamış veya yanlış tamamlanmış modellerin elde edilmesine sebep olmakta, taramanın tekrar tekrar yapılması ise zaman kaybına yol açmaktadır. Otonom tarama ise, genellikle en az bir robot kolu ve algılayıcıdan oluşan bir sistem kullanılarak sonraki en iyi bakışın (Next Best View - NBV) hesaplanmasıyla yapılmaktadır.

Bu tez çalışması kapsamında; endüstriyel robot kolu, lazer profil algılayıcı ve döner tabla kullanılarak bilinmeyen nesnelerin 3B modelinin otonom olarak elde edilmesi amaçlanmaktadır. Uygulanan bu yöntem ile yaklaşık boyutları önceden bilinen herhangi bir nesne yüzeyinden ilk tarama ile veri alındıktan sonra NBV adayları hesaplanmaktadır. Sonra, hesaplanan NBV adaylarından ön tanımlı kriterlere göre en uygun olanı seçilmekte ve algılayıcı bu konuma götürülmektedir. 3B nesnenin yüzey modeli tamamlanana kadar algılayıcı için tarama noktalarının seçimi ve tarama süreci devam etmektedir.

Anahtar kelimeler: 3B sayısallaştırma, otonom tarama, sonraki en iyi bakış, görüş planlama, endüstriyel robot kolu, 3B nokta bulutu

SUMMARY

Today, 3-D modeling of objects becomes possible due to developing technology, spreads extensive application area such as industry, medical, entertainment and preserving cultural inheritance. Mentioned these applications create a need of obtaining 3D models with high quality and precision.

The depth and/or color information obtained from the surface of the object is used for 3-D modeling. This information is obtained by collecting data from the entire surface of the object by using sensors. Moving the sensor along the object surface for the purpose of data collection can be called surface scanning. Two different ways that are manual scanning and autonomous scanning can be followed to obtain 3D model of an object. Manual scanning requires an experienced and skilled person in this area; because it is not easy task to determine from which views the objects will be scanned and what the next scan point will be. In such a case, while man-made faults cause incomplete or inaccurate 3D models, scanning again and again causes loss of time. Autonomous scanning is generally made with calculation of Next Best Views (NBV) by using a system generally at least consisting of a robot manipulator and a sensor.

Within the scope of this thesis, it is planned to obtain 3D model of an object full autonomously using industrial robot manipulator, laser profile sensor and rotary table. By this method applied, NBV candidates are calculated after the data is acquired by the first scan on any object whose approximate dimensions are previously known. Then, the most suitable one is selected from the calculated NBV candidates according to the predefined criteria, and the sensor is taken to this position. The selection of the scan points and scanning process for the sensor continues until the surface model of the 3D object is completed.

Keywords: 3-D Digitization, automatic scanning, next best view, view planning, industrial robot manipulator, 3-D point cloud

TEŞEKKÜR

Tez konumun belirlenmesinde, tez sürecinde ve tezimin tamamlanmasında, bilimsel açıdan her türlü kaynak ve materyali sağlayan, benden desteğini hiçbir zaman esirgemeyen ve yol gösteren danışman hocam Doç. Dr. Metin ÖZKAN'a teşekkür ederim.

Lisans ve yüksek lisanstaki öğrencilik yaşamım boyunca her zaman desteğini hissettiğim, değerli fikir ve görüşleriyle yol gösteren hocam Prof. Dr. Osman PARLAKTUNA'ya teşekkür ederim.

Tezim boyunca; her türlü fikir, görüş, düşünce ve yönlendirmesiyle yardımını esirgemeyen hocam Yrd. Doç. Dr. Helin DUTAĞACI'na teşekkür ederim.

Tezimin tamamlanmasında en büyük destek veren hocalarımdan biri olan, tezimin ilerlemesinde çok büyük yardımı olan, her zaman büyük bir sabırla yardım eden hocam Araş. Gör. Kaya TURGUT'a çok teşekkür ederim.

Ders aşamasında olduğu gibi tez aşamasında da her türlü yardım ve desteğini benden esirgemeyen, duyduğu güvenle beni cesaretlendirip ilerlememe yardımcı olan hocam Araş. Gör. Sezgin SEÇİL'e çok teşekkür ederim.

Lisans ve yüksek lisans hayatımda, mantıklı düşünceleri ve ileri görüşlülüğü ile her zaman destek veren arkadaşım Samet ALBAYRAK'a teşekkür ederim.

Yüksek lisans ders aşamamda aynı ofisi paylaştığım, araştırmalarımda yardımcı olan ve her türlü desteğini veren veren arkadaşım Çağrı Mete ŞENLER'e teşekkür ederim.

Her konuda manevi destek sağlayan ve değerli fikirleriyle ve görüşleriyle yol gösteren hocam Yrd. Doç. Dr. Kemal KESKİN'e teşekkür ederim.

Tüm yaşamım boyunca her zaman olduğu gibi yüksek lisans boyunca da arkamda olan, hiçbir zaman maddi ve manevi desteğini esirgemeyen aileme teşekkür ederim.

115E374 numaralı proje kapsamında 1001 – Bilimsel ve Teknolojik Araştırma Projelerini Destekleme Programı ile maddi destek sağlayan TÜBİTAK'a teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET . . . . . **vi**

SUMMARY . . . . . **vii**

TEŞEKKÜR . . . . . **viii**

ŞEKİLLER DİZİNİ . . . . . **xi**

ÇİZELGELER DİZİNİ . . . . . **xv**

SİMGELER VE KISALTMALAR DİZİNİ . . . . . **xvi**

1. GİRİŞ VE AMAÇ **1**

2. LİTERATÜR ARAŞTIRMASI **3**

3. MATERYAL VE YÖNTEM **20**

4. 3B YÜZEY MODELLEME YÖNTEMİ **21**

4.1. Filtreleme İşlemi 23

4.1.1. Voksel ızgara filtresi 23

4.1.2. Yarıçap aykırı değer filtresi 24

4.1.3. Kırpma filtresi 26

4.1.4. Yumuşatma filtresi 27

4.2. Üçgen Örgü Yapısı İle Yüzey Oluşturma 27

4.3. Sınırlar Üzerinde Yapılan İşlemler 29

4.3.1. Sınırların bulunması 29

4.3.2. Sınırların sıralanması 32

4.3.3. Sınırların sınıflandırılması 33

4.4. Yüzey Eğilimi Tahmini 40

4.5. NBV Adaylarının Hesaplanması 51

4.5.1. NBV adaylarının yönelim hesabı 51

4.5.2. NBV adaylarının pozisyon hesabı 55

4.6. Kalite Kriteri 58

4.6.1. Örnekleme yoğunluğu 63

4.6.2. Ortalama yüzey normalleri 63

4.6.3. Sınır kenarlarının miktarı 64

4.7. NBV Seçimi 65

5. 3B YÜZEY MODELLEMEDE SINIFLANDIRMA 66
5.1. Nokta Özellik Histogramları 66
5.2. Hızlı Nokta Özellik Histogramları 69
5.3. FPFH Histogram Karşılaştırması İle Sınıflandırma 70

6. UYGULAMA ORTAMININ TASARIMI 75
6.1. Sistem Mimarisi 75
6.1.1. Endüstriyel robot kolu 76
6.1.2. Lazer profil algılayıcı 78
6.1.3. Döner tabla 79
6.1.4. Kamera 80
6.2. Endüstriyel Robot Kollarının Programlaması İçin Arayüz Tasarımı 81
6.3. Döner Tabla Etrafında Lokal Bir Koordinat Ekseni Oluşturma 83

7. BULGULAR VE TARTIŞMA 95

8. SONUÇ VE ÖNERİLER 98

KAYNAKLAR DİZİNİ 100

ŞEKİLLER DİZİNİ

Şekil		Sayfa
4.1	NBV Algoritması İçin Akış Diyagramı	22
4.2	Modellemede Kullanılan Nesneler: (a) Fil (b) Endüstriyel Parça (c) Çanak	23
4.3	Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Voksel Izgara Filtresi Uygulanmadan Önceki (a,b,c) ve Uygulandıktan Sonraki Nokta Bulutları (d,e,f)	24
4.4	Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Yarıçap Aykırı Değer Filtresi Uygulanmadan Önceki (a,b,c), Uygulandıktan Sonraki Nokta Bulutları (g,h,i) ve İki Nokta Bulutunun Farkı (d,e,f)	25
4.5	Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Kırpma Filtresi Uygulanmadan Önceki (a,c,e) ve Uygulandıktan Sonraki Nokta Bulutları (b,d,f)	26
4.6	Köşe, Kenar, Yüz ve Yüzey Yapıları	27
4.7	Manifold Olan ve Olmayan Yapılar (a) Manifold Yapı Örneği (b) Manifold Olmayan Yapı Örneği	28
4.8	Üçgen Örgü Yapısı (a) Fil İçin (b) Endüstriyel Nesne İçin (c) Çanak İçin	29
4.9	Örgü Yapısında Yarı Kenar Gösterimi	31
4.10	Yarı kenar, zıt yarı kenar ve okların dönüş yönleri: (a) Yarı kenar ve zıt yarı kenar, (b) Yarı kenarlar (İç taraf), (c) Zıt yarı kenarlar (dış taraf), (d) Yarı kenar için okların dönüş yönü, (e) Zıt yarı kenar için okların dönüş yönü	31
4.11	Ön Yüzlerinden Taranan Nesnelerin En Uzun Sınırları: (a) Fil İçin (b) Endüstriyel Parça İçin (c) Çanak İçin	33
4.12	100 Noktalı Bir Sınırın Alt Kısmı ve Noktaların İndisleri	35
4.13	İlk Sınır Zincir Vektörünün Oluşturulması	35
4.14	İleri Yönde Vektör Zincirinin Oluşturulması	35
4.15	Geri Yönde Vektör Zincirinin Oluşturulması	36
4.16	Aşamalı Olarak İleri Ve Geri Yönde Vektör Eklenmesi: (a) İlk Aşama (b) İkinci Aşama (c) Üçüncü Aşama (d) Dördüncü Aşama	37
4.17	Fil Nesnesi İçin Sınıflandırılmış Sınırlar: Alt Sınırlar (a, b, c), Üst Sınırlar (d, e, f), Sağ Sınırlar (g, h) ve Sol Sınır (i)	41
4.18	Endüstriyel Parça İçin Sınıflandırılmış Sınırlar: Alt Sınırlar (a, b), Üst Sınırlar (c, d, e, f), Sağ Sınırlar (g, h) ve Sol Sınır (i)	42
4.19	Çanak Nesnesi İçin Sınıflandırılmış Sınırlar: Alt Sınır (a), Üst Sınırlar (b, c), Sağ Sınırlar (d, e) ve Sol Sınırlar (f)	43
4.20	Üç Nesne İçin Sınıflandırılan Sınırlar: (a) Fil Nesnesi İçin (b) Endüstriyel Nesne İçin (c) Çanak Nesnesi İçin	43

4.21 Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Sınırlayıcı Kutu İçine Düşen Noktalar ve İlgili Sınırlar: Fil Nesnesinin Sağ Sınırları ve Sınırlayıcı Kutu İçine Düşen Noktalar (a, b), Endüstriyel Parçanın Sağ Sınırları ve Sınırlayıcı Kutu İçine Düşen Noktalar (c, d), Çanak Nesnesinin Üst Sınırları ve Sınırlayıcı Kutu İçine Düşen Noktalar (e, f) 44
4.22 Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Bölge Genişletme Algoritmasıyla Elde Edilen Noktalar: Fil Nesnesinin Sağ Sınırları İçin (a, b), Endüstriyel Parçanın Sağ Sınırları İçin (c, d), Çanak Nesnesinin Üst Sınırları İçin (e, f) 47
4.23 Bölge Genişletme Algoritmasında Düzlük Parametresinin Öneminin Fil Nesnesi Üzerinde Açıklanması: Fil Nesnesinin Alt Sınırlarının Uygun Parametre Değeri İle Sonuçları (a, b), Fil Nesnesinin Alt Sınırlarının Uygun Olmayan Parametre Değeri İle Sonuçları (c, d) 48
4.24 Nesnelere Ait Yüzey Eğilim Tahminleri: Fil Nesnesine, Endüstriyel Parçaya ve Çanak Nesnesine Ait Tahmin Edilen Eğriler (a, b, c, d, i, j, k, l) ve Eğrilerin Başka Açılardan Görselleri (e, f, g, h, m, n, o, p) 52
4.25 Sınırların Orta Noktalarının Sınırın Sınıfına Göre Uzatılması: (a) Alt Sınır İçin (b) Üst Sınır İçin (c) Sağ Sınır İçin (d) Sol Sınır İçin 53
4.26 Sınırın Uzatılmış Orta Noktasının Yüzey Eğilim Eğrilerine Oturtturulması (a) Alt Sınır İçin (b) Üst Sınır İçin (c) Sağ Sınır İçin (d) Sol Sınır İçin 53
4.27 Sınırların Yönleri ve Yüzey Eğilim Eğrisinin Eğriye Oturtturulan Nokta Üzerinden Yüzey Normali (a) Alt Sınır İçin (b) Üst Sınır İçin (c) Sağ Sınır İçin (d) Sol Sınır İçin 54
4.28 Tarama Aday Yolunun Nesnenin Maksimum ve Minimum Noktaları Doğrultusunda Uzatılması 57
4.29 Fil Nesnesine Ait Bazı Sınırlar İçin Hesaplanan NBV Adaylarının Pozisyon ve Yönelimlerinin Farklı Açılardan Görselleri: Üst Sınırı İçin (a, b), Fil Nesnesinin Alt Sınırı İçin (c, d) 59
4.30 Endüstriyel Parça ve Çanak Nesnesinin Bazı Sınırları İçin NBV Adaylarının Pozisyon ve Yönelimlerinin Farklı Açılardan Görselleri: Endüstriyel Parça Üst Sınırlarından Biri (a, b) ve Diğeri (c, d) İçin, Çanak Sağ Sınırlarından Biri İçin (e, f) 61
4.31 Voksel Yapısının Minimum Noktası, Merkezi ve Maksimum Noktasının Görseli 62

5.1 Nokta Özellik Histogramı Hesabında *r* Yarıçapı ve Bu Yarıçap İçine Düşen k-komşuluğundaki Noktaların Bulunması (Rusu vd., 2009) 67
5.2 Nokta Özellik Histogramında uvw Çerçevesinin Hedef Nokta Üzerinde Belirlenmesi ve Kaynak Nokta Üzerinde Taşınması (Rusu, 2009) 67

5.3 Hızlı Nokta Özellik Histogramı Hesabında *r* Yarıçapları ve Bu Yarıçaplar İçine Düşen k-komşuluğundaki Noktaların Bulunması (Rusu vd., 2009) 70
5.4 Küp Üzerinde Köşe, Kenar ve Düz Yüzey Bölgeleri 71
5.5 Silindir Nokta Bulutunun Üstten ve Yandan Görselleştirilmesi (a) Silindir Nokta Bulutunun Üstten Görselleştirilmesi (b) Silindir Nokta Bulutunun Yandan Görselleştirilmesi 71
5.6 Küre Nokta Bulutunun Üstten ve Yandan Görselleştirilmesi (a) Küre Nokta Bulutunun Üstten Görselleştirilmesi (b) Küre Nokta Bulutunun Yandan Görselleştirilmesi 72
5.7 Her Bir Nokta Bulutu İçin Hızlı Nokta Değer Histogramları (a) Köşe Nokta Bulutu İçin (b) Kenar Nokta Bulutu İçin (c) Düz Yüzey Nokta Bulutu İçin (d) Küre Nokta Bulutu İçin (e) Silindir Nokta Bulutu İçin 73

6.1 Sistem Mimarisi 76
6.2 Kawasaki RS05L Endüstriyel Robot Kolu 77
6.3 Kawasaki RS05L Endüstriyel Robot Kolu Denetleyicisi 77
6.4 Micro-Epsilon ScanControl 2900-100 Lazer Profil Algılayıcı ve Ölçüm Koordinatları 78
6.5 Micro-Epsilon ScanControl 2900-100 Lazer Profil Algılayıcı için Ölçüm Aralıkları 79
6.6 Standa 8MR190-2- 4247 Döner Tabla 79
6.7 Standa 8MR190-2- 4247 Döner Tabla Sürücüsü 80
6.8 Kamera ve Lens (a) Kamera (b) Lens 81
6.9 Döner Tabla Etrafında Lokal Koordinat Ekseni Oluşturmak İçin Kullanılan Örüntü 86
6.10 Döner Tabla Etrafında Lokal Koordinat Ekseni Oluşturmak İçin Kullanılan Örüntünün CAD çizimi 86
6.11 Örüntü Üzerine Işın Düşürülmesi 87
6.12 Örüntü Üzerine Işın Düşürülmesi 87
6.13 İki Profil İçin Ham Veri Gösterimi (a) İlk Profil Verisi İçin (b) İkinci Profil Verisi İçin 88
6.14 İki Profil İçin RANSAC Algoritmasının Sonuçları (a) İlk Profil Verisi İçin (b) İkinci Profil Verisi İçin 90
6.15 Kenar Noktaların RANSAC İle Bulunan Çizgiye İzdüşümünün Alınması (a) İlk Profil Verisi İçin (b) İkinci Profil Verisi İçin 91
6.16 İlk ve İkinci Profil Verisinin Aynı Şekil Üzerinde Gösterimi 91
6.17 Belirlenen Örüntüdeki Dairenin Merkez ve Döner Tabla Merkez Koordinatlarının Gösterimi 92
6.18 Örüntüdeki Daire Merkezinin Birim Vektörleri 93
6.19 Tamamlanmış 3B model: (a) Tamamlanmış Modelin Önden Görünümü (b) Tamamlanmış Modelin Üstten Görünümü 93

6.20 Her Tarama İçin Elde Edilen Nokta Bulutları: (a) İlk Tarama İçin (b) İkinci Tarama İçin (c) Üçüncü Tarama İçin (d) Dördüncü Tarama İçin 94

ÇİZELGELER DİZİNİ

Çizelge **Sayfa**

4.1 Lazer Algılayıcının Bakış Eksenine (s) Göre Sınırların Sınıflandırılması 34

SİMGELER VE KISALTMALAR DİZİNİ

Simge veya Kısaltma	Tanım	Sayfa Numarası
2B	2 Boyutlu	78
3-D	3-Dimensional (3 Boyutlu)	vii
3B	3 Boyutlu	vi
ADC	Analog-Digital Converter (Analog-Dijital Dönüştürücü)	80
API	Application Programming Interface (Uygulama Program Arayüzü)	81
CAD	Computer Aided Design (Bilgisayar Destekli Tasarım)	4
CCD	Charge Coupled Device (Yüklenme İliştirilimiş Araç)	8
CMOS	Complementary Metal Oxide Semiconductor (Tamamlayıcı Metal Oksit Yarı İletken)	80
DOF	Degree of Freedom (Serbestlik Derecesi)	8
FIFO	First In First Out (İlk Giren İlk Çıkar)	68
FOV	Field of View (Görüş Alanı)	8
FPFH	Fast Point Feature Histogram (Hızlı Nokta Özellik Histogramı)	66
GKLT	Guided Kanade-Lucas-Tomasi (Kılavuzlu Kanade-Lucas-Tomasi)	12
GLM	General Linear Model (Genel Lineer Model)	10
ICP	Iterative Closest Point (İteratif En Yakın Nokta)	7
IG	Information Gain (Bilgi Kazancı)	15
IRC	Infra Red Cut (Kızılötesi Işık Kesici)	80
LCD	Liquid Crystal Display (Sıvı Kristal Ekran)	10

Simge veya Kısaltma	Tanım	Sayfa Numarası
MLS	Moving Least Squares (Hareketli En Küçük Kareler)	27
MVC	Mass Vector Chain (Kütle Vektör Zinciri)	5
NBS	Next Best Scan (Sonraki En İyi Tarama)	15
NBV	Next Best View (Sonraki En İyi Görüş)	vi
PC	Personal Computer (Kişisel Bilgisayar)	81
PCL	Point Cloud Library (Nokta Bulut Kütüphanesi)	21
PFH	Point Feature Histogram (Nokta Özellik Histogramı)	66
PVS	Probabilistic Voxel Space (Olasılıksal Voksel Uzayı)	18
RANSAC	Random Sample Consensus (Rastgele Örnek Oy Birliği)	88
ROI	Region of Interest (İlgi Bölgesi)	89
SFS	Shape From Shading (Gölgeden Şekil)	12
SPFH	Simplified Point Feature Histogram (Basitleştirilmiş Nokta Bulut Histogramı)	69
TCP	Transmission Control Protocol (Transmisyon Kontrol Protokolü)	81

1. GİRİŞ VE AMAÇ

Son zamanlarda, gelişen teknolojiyle birlikte 3B ölçüm yapan algılayıcıların üretimleri artmaktadır. Üretimdeki bu artış ve algılayıcıların hassasiyetlerinde meydana gelen olumlu gelişmeler, bu cihazların kullanımının geniş alanlara yayılmasını sağlamıştır. 3B algılayıcılar ile nesnelerin sayısallaştırılması işlemi 3B yüzey modelleme olarak ifade edilmektedir. Canlı veya cansız bütün nesnelerin 3B yüzey modelini elde etmek mümkündür.

3B modelleme; medikal alanda, mimari alanda, eğlence sektöründe, endüstri alanında ve kültürel mirasların korunmasında kullanılmaktadır. Medikal alanda elde edilen 3B insan modeli insan sağlığıyla ilgili işlemlerde kullanılırken, mimari alanda 3B modelleme evlerin iç ve dış tasarımının şekillendirilmesinde kullanılabilmektedir. 3B oyunlar ve animasyonlar eğlence sektörünün birer parçasını oluşturmaktadır. Eşi benzeri bulunmayan kültürel eserlerin korunması amacıyla 3B modelleri çıkarılmaktadır. 3B modellemenin en önemli alanlarından biri olan endüstride ise tersine mühendislik, replikasyon ve kalite kontrol gibi işlemler için 3B modeller kullanılmaktadır. Diğer alanlarda olduğu gibi endüstriyel alanda bahsedilen bu uygulamalar, yüksek kalitede ve doğrulukta 3B modellerin elde edilmesi ihtiyacını doğurmuştur.

3B modelin elde edilmesi için nesne yüzeyinden algılayıcı ile veri toplanması gerekmektedir. Nesne yüzeyinden veri toplanması için elle tarama ve otonom tarama olmak üzere iki çeşit yol vardır. Elle tarama, bir el tarayıcısı ile taranacak nesnenin etrafından veri alınmasıyla elde edilen tarama çeşididir. Bu tarama, basit nesneler için kolaylık sağlasa da karmaşık nesneler için tecrübeli ve yetenekli bir personel gerektirmektedir; çünkü karmaşık nesnelerin hangi açılardan taranacağı ve sonraki tarama noktasının ne olacağını belirlemek karışık bir işlemdir. Böyle bir durumda, tamamlanmamış veya yanlış tamamlanmış modellerin elde edilmesi taramanın tekrar yapılmasını gerektirmekte ve taramanın tekrar tekrar yapılması ise zaman kaybına yol açmaktadır. Otonom tarama ise, genellikle farklı mekanizmalar içeren bir sistem ile yapılan taramadır. Bu tarama, robot kolu kullanılarak yapılanlar ve robot kolu kullanılmadan yapılanlar olarak iki kısma ayrılmaktadır. Robot kol kullanılarak yüzey modellemesi yapan sistemlerde genellikle robot kolu algılayıcıyı yüzey modellemesi yapılan nesne etrafında hareket ettirmektedir, fakat robot kolunun yüzey modellemesi yapılan nesneyi algılayıcı etrafında hareket ettirdiği sistemler de mevcuttur. Robot kol kullanılmadan yüzey modellemesi yapan sistemlerde ise döner tabla, nesne yüzeyinden veri alacak şekilde yerleştirilen algılayıcılar gibi mekanizmalar bulunmaktadır.

Fakat, her ikisi için de ortak problem sonraki en iyi bakışın ne olacağı, yani algılayıcının nesne yüzeyinden hangi bakış ile veri alması gerektiğidir. Bu problem literatürde de sıkça geçmektedir. Otonom taramanın, elle taramaya göre üstünlükleri vardır; çünkü otonom tarama elle taramaya göre, daha kısa zamanda, az bakış noktası ile yüksek doğrulukta model elde edilmesini sağlamaktadır.

Bu tez çalışması kapsamında; endüstriyel robot kolu, lazer profil algılayıcı ve döner tabladan oluşan bir sistem kullanılarak bilinmeyen nesnelerin 3B modellerinin otonom çıkarılması amacıyla bir yöntem uygulanmıştır. 3B modelleme için uygulanan bu yöntem ile yaklaşık boyutları önceden bilinen bir nesne üzerinden ilk tarama ile veri alındıktan sonra, nesnenin yüzeyinden alınan bu verilere birtakım işlemler uygulanarak NBV adayları hesaplanmaktadır. Daha sonra, hesaplanan NBV adaylarından belirli kıstaslara göre en uygun olanın seçilmesi ve elde edilen parçalı 3B modelin belirli sonlandırma kriterlerini sağlayana kadar bu aşamaların yinelemeli olarak devam etmesi gerekmektedir. Sonlandırma ölçütleri sağlandıktan sonra 3B modelleme işleminin tamamlanması ve nesnenin 3B modelinin yüksek kalitede ve doğrulukta elde edilmesi hedeflenmektedir.

Tez çalışmasının içeriği şu şekilde sıralanabilir: Literatürün detaylı olarak incelendiği Bölüm 2'de 3B modelleme ile ilgili geçmişten günümüze yapılmış çalışmalardan bahsedilmektedir. Bölüm 3'te tez kapsamında kullanılan materyal ve yöntemden kısaca bahsedilmektedir. Bölüm 4'te, bir önceki bölümde kısaca bahsedilmiş olan 3B modelleme yönteminin alt aşamaları ile detaylandırılmaları yapılmıştır. Bu bölüm, nesne yüzeyinden veri toplandıktan sonra elde edilen nokta bulutu üzerinde uygulanan filtrelemeyi, üçgen örgü yapısı ile yüzey oluşturmayı, sınırların bulunması, sıralanması, sınıflandırılması gibi nesne sınırları üzerinde yapılan işlemleri, yüzey eğilim tahminleri, NBV adaylarının bulunması, NBV adaylarının seçimi için kalite kriterleri ve NBV seçimi gibi önemli başlıkları içermektedir. Bu bölümde, yazılımsal detaylar ve algoritmalar ilgili başlıklar altında verilmiştir. Bölüm 5, 3B modellemeye yeni bir bakış açısı kazandıracak olan nesne yüzeyindeki noktaların bazı özellikler kullanarak sınıflandırılmasını içermektedir. Bu başlık altında, nokta özellik histogramlarından, hızlı nokta özellik histogramlarından, bunların farklarından ve bu yöntemlerle nokta bulut verilerinin nasıl sınıflandırılabileceğinden bahsedilmektedir. Bölüm 6'da, yöntemin uygulanacağı donanımları içeren sistem mimarisinden, robot kollarının programlanmasını kolaylaştıracak bir arayüz tasarımından ve döner tabla etrafında oluşturulan lokal bir koordinat ekseni ve gerekli koordinat eksenleri arası dönüşümlerden bahsedilemektedir. Bölüm 7'de, bulgular ve tartışma verilmiştir. Son bölüm olan Bölüm 8'de uygulamadan çıkarılan sonuçlardan bahsedilerek ilerideki çalışmalar için öneriler verilmektedir.

2. LİTERATÜR ARAŞTIRMASI

3B modelleme, dünya üzerinde bulunan canlı veya cansız herhangi bir varlığın matematiksel temsilinin 3B algılayıcılar yardımı ile dijital ortama aktarılması olarak tanımlanabilir. 3B algılayıcıların hassasiyetlerinin artması, 3B modelleme konusunda yapılan çalışmalarda hızlı bir artışa neden olmaktadır. 3B modelleme konusunda araştırmalar devam etmekte olup özellikle bilinmeyen nesneler için 3B modellemenin otonom yapılması tam olarak çözülmüş bir problem değildir. 3B modeli elde edilen nesnenin yaklaşık boyutlarının belirlenmesi, nesnenin taranacağı sonraki en iyi bakışın planlanması, robotun istenilen pozisyona hareketi, eğer kullanılıyorsa döner tablanın kaç derece açıyla hangi yöne döneceği ve taramada kullanılan donanımlar arası koordinat dönüşümleri başlıca problemler olarak sayılabilir.

3B modellemede yukarıda da bahsedildiği üzere en büyük sorunlardan biri nesnenin üzerinden veri alınacak sonraki bakışın, başka bir deyişle bir sonraki en iyi görüş noktasının belirlenmesidir. Sonraki görüş noktasının doğru bir şekilde belirlenmesi, tarama süresinin kısalmasını ve daha yüksek kalitede bir model elde edilmesini sağlamaktadır. Elle yapılan taramada sonraki görüş noktasının belirlenmesi nesneyi tarayan kişinin yetenek ve tecrübesine bağlı olduğu için etkili bir yöntem değildir. Bu sorunun üstesinden gelmek için çözüm robot kollarıyla yapılan tarama yani otonom taramadır. Literatürde sonraki en iyi görüş noktasının planlanmasına dair pek çok çalışma bulunmasına rağmen bunu otonom olarak gerçekleştiren çok az sayıda çalışma bulunmaktadır. Yani, endüstriyel robot kolu kullanımı için nesnelerin 3B modellenmesi hala ucu açık bir problem olup üzerinde araştırmalar devam etmektedir.

3B modelleme problemi, aslında bir sonraki en iyi görüş noktası problemi olduğundan dolayı, yapılan ilk çalışmalar sadece NBV hesaplama ile alakalı olup robot kolları ve kullanımı yaygın olmadığı için uygulamalar genellikle benzetim ortamında yapılmıştır. Son yıllardaki çalışmalarda ise, hesaplanan en iyi görüş noktaları; robot kolları, 3B algılayıcılar ve ekstra donanımların olduğu fiziksel sistemlerde denenmiştir.

Sonraki en iyi görüş noktası hesaplamanın ilk çalışmalarından biri olan ve Connolly (1985) tarafından yapılan çalışma bir ortamın tamamlanmış modelini elde etmek için ardışık derinlik imgelerinin sırasının ne olduğunu bulmak için yapılmıştır. Bu çalışmada, sonraki en iyi görüşe karar vermek için parçalı sekizli ağaç yapısı kullanan iki yöntem sunulmuştur. İlk algoritma, Planetarium algoritması olup NBV hesaplanacak alanda bir küre tanımlayıp küre

üzerinde enlemler ve boylamlar şeklinde örnekler almaktadır. Her örnekleme noktası için sadece dolu ve görünmeyen alanlar düşünülerek gizli bir çizgi şeklinde bakış açıları üretilir. Görünmeyen alanı en çok kavrayan bakış açısı NBV olarak seçilmektedir. İkinci algoritma ise Normal algoritması olup sekizli ağaç yapısındaki yüz bilgisini kullanmaya dayanmaktadır ve küp şeklinde altı yüze sahip bir yapı oluşturulmaktadır. Her altı yönde de görünmeyen/boş yüzlerin alanları sayılarak NBV yönü, üç tane yüzün en yüksek değerde birleştiği küpün köşe noktası olarak seçilir. İlk yöntem, zaman açısından maliyetli olsa da basit, açık ve daha az bakış açısı seçme eğiliminde olan bir yöntemdir. İkinci yöntem ise, daha basit ve hızlıdır. İki algoritmanın ortak dezavantajı ise NBV hesaplanan nesnenin engelli alanlarını ya da başka bir deyişle tıkalı bölgelerini dikkate almamasıdır.

Maver ve Bajcsy (1991), tek bir resim ile hacimsel model elde etmeyi bir problem olarak görmüşlerdir. Derinlik imgelerindeki nesnenin tıkalı bölgelerinden kaynaklı karışıklıkların çözülmesi için algılayıcının farklı şekillerde konumlandırılması gerektiğine ihtiyaç duyulduğunu söylemişlerdir. Bu yüzden, tıkalı bölge etrafındaki dar bölgeden alınan verinin kullanılması ile bir dizi farklı bakış noktaları elde edilmesini sağlayan bir yöntem sunmuşlardır. İlk veri rastgele alındıktan sonra tıkalı bölgeler poligonlar ile temsil edilmektedir. Bakış açıları hesaplandıktan sonra sonraki bakış açıları da hesaplanmalıdır. Yeni ve eski bakış açıları kullanılarak bir histogram elde edilmektedir. NBV ise gerekli tarama sayısını en az olarak bulan yön olarak hesaplanmıştır. Maver ve Bajcsy (1992) ve Maver ve Bajcsy (1993) tarafından yapılan çalışmalar ise bu çalışmaları taban alarak elde edilen daha gelişmiş çalışmalardır.

Pito (1995) sunduğu yöntemde derinlik ölçen kamera kullanarak herhangi bir nesnenin CAD modelini çıkarmak amacıyla NBV problemine çözüm üretmiştir. Çalışmanın amacı, yeni alınan resmin önceki resim ile çakışmasını ve üzerine kaydedilmesini garanti ederken nesne yüzeyinden maksimum veri almak için kameranın pozisyonunun ne olduğunu bulmaktır. Bu yüzden her taramadan sonra görüş alanı içinde kalan bölge, görülen hacim ve görülmeyen hacim (boş hacim) olarak sınıflandırılmıştır. Algoritma, nesne hakkında herhangi bir geometriyle ya da topolojiyle sınırlama yapmamıştır. Bu çalışma kapsamında yazar tarafından konumsal uzay adı verilen, derinlik kamerasının tarayabileceği yüzeyin ve taranması gereken yüzeyin ne kadar olduğunu elde eden bir kavram sunulmuştur. Konumsal uzayın bir alt kümesi nesne ile kamera arasındaki göreceli hareketi kullanarak parametrelerle ifade edilebilmektedir. Bir sonraki en iyi bakış açısına karar vermek için ya kameranın ya da taranan nesnenin konumsal uzay ifadesi hareket parametresiyle bağlantılı eksen boyunca parametre uzayında ötelenir. Bir önceki tarama ile yeteri çakışmayı sağlayacak kadar bilgiye sahip yeni imgeyi garanti ederken kameranın tarayabileceği taranmamış hacim miktarını maksimize eden bir amaç fonksiyonu kullanılmaktadır. Önerilen yöntem, üçgenleme prensibi ile çalışan bir silindirik tarayıcı ve

döner tabla kullanılan bir düzenek üzerinde bir kupanın modellenmesi ile denenmiştir. Pito (1999), 1995 yılında sunduğu çalışmayı geliştirerek yöntemini daha kapsamlı hale getirmiştir. Algoritma, önceki çalışma ile aynı düzenek üzerinde kupa, telefon ve heykelcik üzerinde denenmiş ve görünmeyen bölgeler hakkında edinilen bilgiyi arttırma, önceki tarama ve şimdiki tarama arasındaki çakışmayı sağlama, taranan nesnenin geometrisi ve topolojisi hakkında herhangi bir varsayıma sahip olmama, geniş bir potansiyel tarama verisi içinde NBV seçerek bunun matematiksel hesap açısından maliyetinin makul olması, algoritmanın kendini sonlandırılabilmesi ve genellenebilir olması kısıtları sağlanmaya çalışılmıştır.

Yuan (1995) tarafından otomatik 3B modelleme için bir mekanizma sunulmaktadır. Yöntem, yüzey görülebilirliği hesabı ile başlar. Daha sonra, kütle vektör zinciri (MVC-Mass Vector Chain) ve nesnenin kapalı model olması ilişkisi yorumlanmış olup kapalı nesneler için MVC değerinin sıfır çıkması gerektiği belirtilmiştir. Sunulan sistemin daha önce veri alınmamış yüzeylerden veri alacak şekilde bakış açısı elde etmesi, yüzey için hesaplanan MVC yardımı ile olmuştur. NBV ise hesaplanan MVC vektörünün tersi yönünde seçilmelidir. Yöntemin çok karmaşık nesneler için uygun olmadığı belirtilir iken böyle bir durumda maksimum iterasyon sayısının kullanılması gerektiği vurgulanmaktadır .

Banta ve Abidi (1996), 3B model elde etmek için NBV hesabında algılayıcı pozisyon ve yönelimini otomatik olarak belirleyen bir yöntem sunmuşlardır. Geliştirilen sistem ile modelin görünen yüzeylerinin ve kameranın bakış açısından engelli olan alanların yeniden yapılandırılması gerçekleştirilmektedir. Algoritma, beş adımda çalışmaktadır. İlk adımda, en son alınan resimden elde edilen bilgiler yüzey modeli ve tıkalı bölge modeli olarak ayrıştırılmaktadır. İkinci adımda, tıkalı bölge modelindeki pozisyonlar ve bu yüzlerin yönelimleri not edilmektedir. Üçüncü adımda, kendi küme merkezinden en uzakta olan vokselin merkezi küme merkezlerinin ortalama uzaklığının yarısı olana kadar yinelemeli olarak voksel yüzleri kümelere bölünmektedir. Dördüncü adımda, en büyük kümenin yüzey normali bakış doğrultusu olarak seçilmektedir. Son adımda ise, NBV pozisyonu olarak en büyük kümenin merkezinden geçen bakış doğrultusu üzerindeki ışın üzerinden seçilmektedir. Algoritma, yapay nesnelerin derinlik imgeleri üzerinde test edilmiştir. Banta vd. (2000), Banta ve Abidi (1996) tarafından sunulan çalışmayı geliştirerek nesne etrafından alınmış minimum sayıda resim kullanarak 3B model elde etmeye çalışmışlardır. Bu çalışmada, C++ dilinde bir ara yüz tasarımı yapılarak 3B modellemeye ait kontrol paneli, imge paneli, model paneli ve istatistik paneli kullanıcı tarafından kontrol edilebilir bir duruma getirilmiştir. Algoritma, girdi olarak Schilling Titan II robot kolunun ucuna monte edilmiş lazer derinlik algılayıcısı ile alınmaktadır. 3B modelleme, verilen doğruluk eşik değerine ulaşana kadar devam etmektedir. Birkaç nesnenin 3B modellemesi yapılarak yöntem test edilmiştir.

Papadopoulos-Orfanos ve Schmitt (1997) nesnelerin 3B modellenmesinde farklı bir probleme değinmişlerdir. Modelleme aşamasında veri alımı için derinlik ölçen bir lazer algılayıcının kullanılması ve kullanılan algılayıcının bakış açısının küçük olması, algılayıcı ve taranan nesnenin birbirine çok yakın olmasına neden olarak çarpışma ihtimalinin doğmasına sebep olmuştur. Bu yıla kadar yapılan çalışmalar çok büyük oranlarda sadece modelleme ile uğraşırken, bu çalışmada çarpışmadan kaçınma ile uğraşılmıştır. Verilerin düzgün örneklenmesi, çalışma uzayının doğru bir şekilde boş, gölge ve tıkalı olarak işaretlenmesini sağlamaktadır. Sonrasında, planlama problemi iki aşamadan oluşan hiyerarşik bir yaklaşım ile çözülmektedir. İlk seviye olan alt seviyede tek bir bakıştan alınan verilerin dijitalleştirilmesi ve çarpışmasız yol planlama yapılmaktadır. Diğer seviye olan üst seviyede ise, sorun NBV seçimidir.

Garcia vd. (1998) tarafından yapılan bir çalışmada hedef görüş alanında bulunan ve önceden hakkında hiçbir bilgiye sahip olunmayan bir nesnenin yüzeyini gözlemleyen ve yüzeyinden veri almayı sağlayan bakış açılarının kümesi bu görüş alanını gören küre şeklinde ve hareket edebilme yeteneğine sahip bir derinlik algılayıcısı kullanılarak yapılmıştır. Görülebilirlik analizi gibi zaman açısından maliyetli bir yöntem kullanılması yerine iki aşamadan oluşan bir algoritma sunulmuştur. İlk aşama, tıkalı kenarları taban alan nesne yüzeyinden veri almadan sorumlu iken; ikinci aşama ise, nesnenin kendisini tıkamasından kaynaklı ilk aşamada alınmayan veriler için delikleri bulmak amacıyla görülebilirlik analizi uygulamaktır.

Wong vd. (1999) tarafından yapılan çalışmanın amacı, her bir potansiyel bakış açısında hacimsel model için bilinmeyen bilgilerin miktarını ölçen bir amaç fonksiyonunu optimize ederek NBV hesaplamaktır. Potansiyel bakış açıları ya nesne etrafına eş dağılımlı olarak yerleştirilir ya da doluluk ızgarası modelinin yüzey normallerinden hesaplanır. İlk yöntem olan optimal metot yöntemi, küre şeklindeki bir yapıya eş dağılımlı olarak bakış açılarının yerleştirilmesi ile yapılmaktadır. Bu yöntem, bakış açılarının yeterince yoğun olması durumunda potansiyel bakış açılarının yüzeyi yeterince kavrayacağı mantığına dayanmaktadır. Bu yöntemin tek dezavantajı çok sayıda bakış açısı olmasından kaynaklı hesaplama yüküdür. İkinci yöntem olan yüzey normalleri yöntemi ise, bilinmeyen görülebilir voksellerin yüzey normallerinin toplamını içermektedir. Hesaplanan altı tane yüzey normali sekiz tane bakış açısı vermek için kullanılmaktadır. Başka bir deyişle, bu yöntem her bir voksel doluluk ızgarasının sekiz köşesini bakış açısı olarak seçer. Bu yöntemin tek dezavantajı ise çerçeve ve kupa gibi kendi kendini engelleyen nesnelerin modellenmesi sırasında bakış açılarının bilinmeyen köşelerden gelebilmesidir. Bu dezavantajı elimine etmek amacıyla optimal metot ve yüzey normalleri metodunun birlikte kullanıldığı uyarlamalı bir yöntem sunulmuştur. Bu yöntem yüzey normalleri metodunun bilinmeyen bölgeden bakış açısı elde edilmesi durumunda diğer bakış açısı optimal metot

ile seçilmektedir. Her yeni veride amaç fonksiyonu ile uyarlamalı metot kullanılarak optimal pozlar hesaplanır ve her yeni bakıştan alınan veriler ile model güncellenir. Algoritma bu şekilde, içi veri ile dolan voksellerin sayısı artana kadar devam eder ve final modeli elde edilir. Yöntem, farklı karmaşıklık derecelerine sahip sekiz nesne üzerinde denenmiştir.

Callieri vd. (2004) doğru ve tamamlanmış 3B nesne modelleri elde etmek için RoboScan adını verdikleri döner tabla, robot kolu ve 3B lazer algılayıcıdan oluşan bir sistem oluşturmuşlardır. Sundukları yöntem ise dört aşamadan oluşmaktadır. İlk aşama, kaba şekil elde etme aşaması olarak adlandırılmıştır. Bu aşamada, nesneye ait hiçbir veri olmadığı varsayımından dolayı sınırlayıcı bir hacim elde etme ihtiyacı doğmuştur. Modellenen nesnenin bir yüzü taranacak ve döner tabla kullanılmayacak ise bu hacim düzlemsel olarak, modellenen nesnenin bütün yüzleri taranacak ve dolayısıyla döner tabla kullanılacak ise hacim silindirik olarak seçilmektedir. Bu hacim türlerinden birine karar verildikten sonra, kullanıcı tarafından, bakış noktalarının sayısı ve yoğunluğu seçilerek alınan tarama ile elde edilen veri, nesne ile ilgili kaba şekli oluşturmaktadır. İkinci adımda, ilk dijital modelin elde edilmesi için ilk adımda alınan derinlik verileri belli bir oranda çakışma şartını sağlıyorsa ICP algoritması ile birleştirilmektedir. Üçüncü adım, veri alınmayan bölgelerin, yani model üzerindeki deliklerin tespit edilmesidir. Bu aşamada, deliklerdeki eksik veriyi tamamlamak için algılayıcı pozisyonunun uygun olarak planlanması ve algılayıcının hesaplanan noktalardan veri alması gerekmektedir. Planlama algoritması, kullanıcı tarafından tanımlanan sınırlayıcı kutunun içindeki bir noktanın merkezinde bulunan, küresel bir yüzey üzerinde yer alan bir dizi aday bakış açısını test ederek başlamaktadır. Alınan her resim için deliklerin yüzdesi hesaplanarak bakış açıları sıralanmaktadır. Bu algoritma, sıralı bakış açılarından elde edilen veride büyük bir değişim olmayana kadar veya iterasyon sayısı maksimum seviyeye ulaşana kadar devam etmektedir. Son aşamada ise, elde edilen model üzerinde deliklerin hala yüzeyde bulunma ihtimaline karşı kullanıcı, tarayıcının manuel hareketi ile nesne yüzeyinden veri toplar ve üçgen örgü yapısı oluşturularak nesnenin final modeli elde edilmektedir.

Chen ve Li (2005) bilinmeyen nesnelerin aktif görüş sistemleri ile 3B modellenmesi için yeni bir yöntem önermişlerdir. Bu yöntemde, taranan nesnenin bölgesel özelliklerini gösteren yüzey eğilimi kullanılmaktadır. Eğilim analizine yönelik önceki yaklaşımlar genellikle regresyon yüzeylerini üç boyutlu yorumlamak için polinom denklemleri üretmeye odaklanmış olsa da, bu bildiride nesne yüzeyinin bilinmeyen alanını tahmin etmek için yeni bir matematiksel model önerilmektedir. İlk veri rastgele alındıktan sonra elde edilen parçalı model kullanılarak iki aşamada NBV için karar verilmektedir. İlk aşama, keşif yönüne karar vermek iken ikinci aşama yüzey eğilimi kullanarak algılayıcı pozisyonuna karar vermektir. Keşif yönüne karar vermek ise üç aşamadan oluşmaktadır. İlk

aşamada, nesnenin sınırları parçalı model üzerinden bulunmaktadır. İkinci aşamada nesneye ait elde edilen sınır noktaları bir sıralama fonksiyonu ile sıralanmaktadır. En yüksek sıra değerine sahip olan nokta keşif yönü olarak seçilmektedir. Keşif yönüne karar verildikten sonra yüzey eğilimi bulunur. Algılayıcının görülebilirlik, odaklanma, görüş alanı (FOV), bakış açısı, çözünürlüğü gibi bazı parametrelerle birlikte yüzey eğilimi de kullanılarak algılayıcının pozisyon ve yönelimi bulunmaktadır. Elde edilen verilerin gürültülü olması sınır bulurken sonuçları çok fazla etkileyeceğinden alçak geçiren bir filtre kullanılarak veriler işlenmektedir. Yöntem, projektör ve CCD kameradan oluşan yapılandırılmış ışık sistemi ve 6-DOF robot kolu kullanılarak farklı iki nesne üzerinde test edilmiştir.

Li vd. (2005) sundukları makalede nesnelerin 3B modellenmesi için otomatik ve verimli bir yöntem kullanmışlardır. Yöntemde, ilk bakış açısı rastgele seçildikten sonra bakış açısı planlamanın ilk aşaması olan ve yüzeyin şekli hakkında yaklaşık bilgi veren yüzey eğilimi hesaplaması yapılarak ikinci aşaması olan keşif yönünün karar verilmesine geçilmektedir. Bu aşamada yüzey türünün sınıflandırılması için Gauss eğriliği ve ortalama eğriliği kullanılmaktadır. Yüzey türü sınıflandırıldıktan sonra her yama (patch) için ikinci dereceden eğriler elde edilmekte ve benzer eğriler birleştirilmektedir. Sınırlar boyunca en fazla veri alımına olanak veren algılayıcı pozu NBV olarak seçilmektedir. Sonraki en iyi bakış açısına karar verildikten sonra, bakış açısı planlamanın üçüncü aşaması olan algılayıcının sonraki bakış açısına konumlaması yapılmaktadır. Bu konumlama; algılayıcının odaklanması, görülebilirliği, bakış açısı, çözünürlüğü ve görüş alanı gibi parametreler göz önünde bulundurularak yapılmaktadır. Daha sonra, ardışık iki ölçüm arasındaki bakış açılarından alınan hacimlerdeki değişimin analizi ile karar verilen sonlandırma ölçütünün sağlanıp sağlanmadığı kontrol edilmektedir. Veri kümesinin hacim hesabı için üçgen ağ yapısının yüzey integrali kullanılmaktadır. Bu şekilde algoritma sonlandırılma ölçütü sağlanana kadar devam etmektedir. Algoritma, kamera ve projektör kullanılan bir sistemde bazı nesnelerin kullanılmasıyla test edilmiştir. Kullanılan nesneler ile ilgili yüzey integrali, integral değişimi ve hesaplanma süreleri tablo haline getirilmiştir.

Li ve Liu (2005) tarafından yazılmış başka bir makalede ise nesnelerin 3B modellenmesi için bilgi entropisi tabanlı bir yöntem sunulmuştur. Yöntemde, nesne enine kesitlere ayrılarak bu kesitlerin yeniden birleştirilmesi kapalı B-Spline eğrileri kullanılarak yapılmaktadır. B-Spline model yapısının seçimi için bir bilgi kriteri geliştirilmiştir. Seçilen B-Spline modeline bağlı olarak, bilgi entropisi o modelin belirsizlik ölçütünü belirlemekte kullanılıp, alınacak yeni ölçümler için bilgi kazanıcını tahmin etmekte bu ölçüt analiz edilmektedir. Nesne hakkındaki bilgi kazancı tahmin edildikten sonra bilgi kazancının bakış açısının uzayına gönderimi (mapping) yapılmaktadır. Maksimum bilgi kazancına sahip olan nokta NBV olarak seçilmektedir. Bilgi kazancının maksimum olduğu nokta görünmez olana kadar ya da gürültü şeklinde görünene kadar algoritma çalışmaya devam etmektedir.

Sunulan yöntemin literatüre katkıları; NBV problemi çözümünde kullanılan parametrelerin orijinal olması, B-Spline modelinin Bayesian istatistiği çerçevesinde yeniden yapılandırılması ve her yeni veri alındığında B-Spline modelinin daha uyum sağlayan bir yapıda kullanılması için geliştirilmiş Bayesian bilgi kazancı ölçütünün kullanılması olarak sıralanabilir. Algoritma, robot kolu ucuna takılmış bir projektör ve kamera sistemi kullanılarak birkaç nesne üzerinde test edilmiştir. Her nesne için ilk NBV, diğer NBV'ler, nesnelerin karmaşıklığı ve doğruluk oranları görselleştirilmiş, model karmaşıklığı ve B-Spline modelinin karmaşıklığı tablo haline getirilmiştir.

Wenhardt vd. (2006) tarafından yapılan çalışmanın amacı, 2B noktalar kullanarak nesnelerin 3B modellenmesi için en uygun bakış noktalarını seçmektir. İlk olarak, alınan noktaların üzerindeki gürültünün fazla olmasından dolayı Kalman filtresi kullanılarak nesnenin geometrisi hakkında daha iyi tahminler yapılabilmektedir. Bu aşama, ileriki aşamalarda elde edilecek olan kamera pozisyonunun etkisini ve dolayısıyla ölçümün kalitesini arttırmaktadır. Sonraki aşama, bakış açısının değerlendirilmesi ve seçilmesi aşamasıdır. Önceden hakkında bilgi sahibi olunan nesneler için bilgi entropisi kullanılmakta olup burada böyle bir durum söz konusu olmadığı için şartlı entropi olarak adlandırılan tahmin edilen entropi, bakış açısının değerlendirilmesi için kullanılmaktadır. Bulunan optimal noktanın NBV olarak seçilmesi için gözlemlenen 3B noktaların imge içinde görülebilir yani kameranın bakış açısı içinde kalması anlamına gelen görülebilirlik ve bakış noktasının robot tarafından erişebilir olması anlamına gelen erişebilirlik kısıtlarının sağlanası gerekmektedir. Sunulan yöntem, Firewire kamera ve kamerayı hareket ettiren robot kolu kullanılarak bir kalibrasyon örüntüsünün modellenmesi ve eğme kolu (tilting arm) ve döner tabla kullanılarak herhangi bir nesnenin modellenmesi yapılarak iki farklı şekilde test edilmiştir.

Wenhardt vd. (2007) tarafından yapılan çalışma 2006 yılında yine Wenhardt vd. (2006) tarafından yapılan çalışmanın gelişmiş versiyonu olarak söylenebilmektedir. Bu çalışmada, 3B modellemenin, sensörün hareketleri ile birlikte olasılıksal durum hesabı ile alakalı olduğu ve NBV seçimi için durum tahminlerinin belirsizliklerinin kullanıldığı vurgulanmıştır. Tahmin belirsizlikleri için Gauss matrisinin kovaryansının determinantı ile bağlantılı olan entropi D-optimalite, yine bu matrisin özdeğeri ile bağlantılı olan E-optimalite ve izi ile bağlantılı olan T-optimalite kullanılmaktadır. Yani, önceki çalışmadan bu çalışmanın farkı şartlı entropi kullanmak yerine D-optimalite, E-optimalite ve T-optimalite kullanılmasıdır. Testler ise önceki çalışma ile aynı donanımlar kullanılarak yapılmıştır.

He ve Li (2006), 3B otonom nesne modellemesini bakış açılarını planlama, göreceli nesne-algılayıcı pozisyonunun konfigürasyonu, nesne yüzeyinden veri toplanması, alınan

verinin ortak bir referans koordinat ekseni çerçevesinde kaydedilmesi ve final modelin elde edilmesi olarak beş adımda özetlemişlerdir. Algılayıcı pozisyonlarının bulunması ve konfigürasyonu kadar önemli olan başka bir husus ise 3B modellemenin sonlandırılmasıdır. Sunulan yöntem kendi kendine sonlandırma kararı verilebilmektedir. Algoritma, rastgele ilk ölçümün yapılması ile başlamaktadır. Sonra, Maver'in yöntemi kullanılarak NBV hesabı yapılmaktadır. Daha sonra, alınan ölçümlerle elde edilen veriler parçalı modele eklenerek tamamlanmış model elde edilene kadar devam etmektedir. Her bakış açısından elde edilen hacimlerin hesaplanması ve arka arkaya olan bakış açılarından elde edilen hacimlerdeki değişimin analiz edilmesi ile sonlanma ölçütünün sağlanıp sağlanmadığı kontrol edilmektedir. Veri bulutunun hacmi üçgen örgü yapısının yüzey integralinin hesaplanması ile bulunmaktadır. Yüzey integrali ise Gauss teoremi ile hesaplanmaktadır. İki ardışık bakış noktasından elde edilen hacim arasındaki fark belli bir eşik değerinden düşük ise algoritma sonlanmaktadır. Sunulan yöntem, iş tablası üzerine yerleştirilmiş bir nesne, bir kamera ve LCD projektör kullanılarak test edilmiştir.

Zhou vd. (2008) tarafından yapılan, nesnelerin 3B otonom modelleme, çalışmasının amacı nesne hakkında minimum bilgiye sahip olarak optimal NBV hesabı yapmaktır. İlk önce lazer tabanlı görüş sisteminin görüş alanına karar verilmiştir. Sonra, görüş planlama stratejisi oluşturulmuştur. Bu strateji, bakış noktası 1'den veri alındıktan sonra bakış noktası 2'nin lazer algılayıcı ile veri alabileceği aralıkta olması ve görünmeyen alanı en çok gören bakış açısı olması şartını sağlaması mantığına dayanmaktadır. Yani, görünmeyen alandan en çok veri alan görüş noktası NBV olarak seçilmektedir.

He vd. (2009), Zhou vd. (2008) tarafından yapılan çalışmaları bir aşama daha geliştirerek otomatik 3B modelleme için yüzey eğilimi ve görüş sisteminin sınırlarını kullanan bir yaklaşım öne sürmüşlerdir. İlk önce, Zhou vd. (2008) tarafından yapılan çalışmada da olduğu gibi lazer ve CCD kamera kullanarak etkili görüş alanının minimum ve maksimum değerlerini bulmuşlardır. Alınan ölçüm için Genel Lineer Model (GLM) kullanılarak yüzey eğilimi hesaplanmaktadır. Sonra, algılayıcının gördüğü yüzeyin sınırları ve hesaplanan nesne yüzey eğilimi arasındaki ilişkinin kullanılmasıyla aday bakış noktaları elde edilmektedir. Aday noktalar içinden, en fazla görülmeyen alanı gören, başka bir ifade ile, nesne yüzeyinden en fazla veri toplayabilecek olan bakış açısı adayı NBV olarak seçilmektedir. Yöntem konkav nesnelerin modellenmesi haricinde etkili olarak çalışmaktadır.

He vd. (2010) tarafından yapılan başka bir çalışma ise, tekrardan Zhou vd. (2008) tarafından yapılan çalışmadaki yöntemin geliştirilmesi ile oluşturulmuştur. Görüş sisteminin maksimum ve minimum sınırları belirlendikten sonra bunla bağlantılı olarak parçalı nesne modelinin yüzeyindeki sınır bilgisini temel alan görüş sisteminin görsel bölge limitleri elde

edilmektedir. Elde edilen bu görsel bölge limitlerine göre görüş sisteminin dönüş açısı ve ötelenme mesafesinin aralığı belirlenmektedir. Yani, ilgili görülebilir bölge alanı ile bilgi elde edilmiş olmaktadır. Bakış açıları içinden en geniş görülebilir alana sahip aday NBV olarak seçilmektedir. Bu çalışma da konkav nesneler dışında etkili olarak çalışmakta olup yapılan testler bunu göstermektedir.

Ding vd. (2010) tarafından yapılan çalışmada; Zhou vd. (2008), He vd. (2009) ve He vd. (2010) tarafından yapılan çalışmaların ortak eksik yönü olan nesnenin kendi kendini engellediği durumlar için çözüm üretilmiştir. İki CCD kamera ve lazerden oluşan bir sistem ile görüş sisteminin görülebilir bölge alanının limitlerini elde edildikten sonra sensör planlama stratejisi ve tıkalı bölge planlama stratejisini içeren planlama aşamasına geçilmektedir. Sensör planlama stratejisi görünen yüzeyin limitlerine dayanmaktadır. Tıkalı bölge planlama stratejisi ile ise önceden alınan yüzey bilgisi ile tıkalı bölge üzerinde kontur tahmini yapılır ve görüş sisteminin görülebilir alanı ile tıkalı bölgenin ölçümü için NBV pozisyonuna karar verilmektedir. Sunulan yöntem, tıkalı bölgelere sahip olan kupa ve fil nesneleri üzerinde test edilerek yöntemin verimliliği gösterilmiştir.

Vásquez-Gómez vd. (2009) bilinmeyen bir nesnenin derinlik kamerası ile 3B modellenmesi için bir NBV algoritması geliştirmişlerdir. Yöntem, hacimsel olup voksel işaretlemesi kullanılmaktadır. Her bir voksel, bir etiket (dolu voksel, boş voksel, tıkalı voksel, işaretlenmemiş voksel veya tıkalı vokselin sınırındaki voksel), normal değerleri (voksel yapısı içindeki noktaların yüzey normallerinin ortalaması) ve kalite değeri (noktanın yüzey normali ile algılayıcının bakış açısı arasındaki açının kosinüs değeri) olmak üzere üç özellik saklamaktadır. Bakış açısı adaylarının hesaplanmasından sonra makul olan bakış açılarının değerlendirilmesi fayda fonksiyonu kullanılarak yapılmaktadır. Fayda fonksiyonu; görünmeyen bölgelerden veri alımını arttıran ve önceki taramalarla çakışma sağlayan alan faktörünü, şimdiki görüş noktası ve aday görüş noktası arasındaki mesafeye karşılık gelen navigasyon faktörünü, aday bakış açısı için önceki ve şimdiki tarama ile oluşan kesişim bölgesindeki dolu voksellerin kalitesini belirleyen kalite faktörünü ve nesnenin kendi kendini engellediği tıkalı bölgelerdeki sorunu çözen tıkanıklık faktörünü içermektedir. Daha sonra, yinelemeli hiyerarşik strateji, ideal ışın izleme çözünürlüğü ve çok çözünürlüklü strateji aşamalarından geçen arama stratejisi aşaması ile en iyi görüşe karar verilmektedir. Bu çalışmanın literatüre ana katkıları; navigasyon mesafesini kullanan fayda fonksiyonu ve hiyerarşik mozaik döşeme (tessellation) ve çok çözünürlüklü ışın yollama ile yeni bir arama stratejisi olarak sıralanmıştır. Derinlik kamerası ile alınmış veriler için benzetim ortamında algoritma test edilmiştir.

Trummer vd. (2010) tarafından yapılan çalışmanın amacı herhangi bir nesnenin 3B modellemesindeki doğruluğu optimize edecek şekilde NBV planlaması yapmaktır. O

zamana kadar yapılan çalışmalarda hesaplanan NBV, hep derinlik kamerası için olup bu çalışmada farklılık unsuru yaratan yoğunluk kamerası kullanılmasıdır. Pasif kamera ile bilinmeyen nesnelerin modellenmesinde Wenhardt vd. (2006) tarafından yapılan çalışma NBV planlaması için istatiksel E-kriterine değinen tek çalışma olup özdeğerlerden gelen bilgilerin burada kullanılmaması eksiklik olarak görülmektedir ve planlama optimizasyonu bütün olası kamera pozisyonlarını değerlendiren yorucu bir metot olduğu söylenebilmektedir. Bu yüzden Trummer vd. (2010) tarafından yapılan çalışma genişletilmiş E-kriterini kullanarak doğruluk optimizasyonu için gerçek zamanlı olarak NBV hesaplamaktadır. Kamera parametrelerini kullanan GKLT (Guided Kanade-Lucas-Tomasi) izleme yöntemi NBV planlama için kullanılmaktadır. Bu sayede 3B modelleme ve NBV planlaması yapan birleşik bir yöntem ortaya çıkmıştır. Alınan her 3B veri için belirsizlik ve kovaryans matrisi hesaplanmaktadır. Kapalı bir model elde etmek ve NBV hesaplamak için, yani optimal bir çözüm için, genişletilmiş E-kriteri kullanılmıştır. Son olarak, final planlama aşaması, izleme ve modellemeden gelen güncel bilgiler doğrultusunda gerçek zamanlı olarak gerçekleşmektedir.

Krainin vd. (2011) sundukları yöntemde diğer çalışmalardan farklı bir yol izlemiş olup robot koluyla algılayıcıyı hareket ettirmek yerine nesneyi hareket ettirerek nesnenin 3B modelini çıkarmak için bir sistem geliştirmişlerdir. Robot kolunun kamera önünde nasıl hareket edeceğine karar vermek amaçlı bilgi kazancı tabanlı NBV kullanılmıştır. Bilgi kazancını maksimize edecek şekilde robot kolu nesnenin en belirsiz olan yüzeylerinden veri almaktadır. NBV hesabında hacimsel yöntem kullanılmakta olup vokseller bilinmeyen ve boş olarak işaretlenmektedir. Robot kolunun nesne yüzeyinden veri almasını engelleyen tutucu kısımdan veri alınmasını sağlamak için sunulan yöntem ne zaman ve ne şekilde nesneyi bırakıp tutması gerektiğini tamamlanmış 3B model etmek için hesaplanmaktadır. Bilgi kazancı belli bir değerin altına düştüğü zaman robot kolu nesneyi bırakarak başka bir yerden kavrar ve robot kolunun nesneyi engellediği kısımlardan bu sayede veri alınmış olmaktadır. Algoritma, robot kolu ve algılayıcı olarak bir derinlik kamerası kullanılan bir sistem üzerinde farklı nesneler için denenmiştir.

Fang ve He (2012) tarafından bilinmeyen nesnelerin modellenmesi için aktif ve pasif görüşlerin bir arada kullanıldığı ve bakış açılarına insan sezgilerine benzer bir şekilde karar verilen yeni bir yöntem sunulmuştur. İlk olarak yöntem, binoküler görsel sistemde (sağ ve sol iki kameradan oluşan bir sistem) görsel yüzeyin limitlerini vermiş olup NBV ise konturun tahmin edilmesi ile elde edilmektedir. İkinci olarak, modelleme aşamasındaki tıkalı bölgeler için SFS (Shape from Shading – Gölgeden Şekil) yöntemi veya görsel yüzeyin limitleri ile tıkalı bölgelerin konturları tahmin edilmektedir. Son aşamada ise, modeli tam olarak tamamlamak için derinlik imgesindeki tıkalı bölgeyi doldurmak amacıyla ek olarak bulunan bakış noktası seçilmektedir. Derinlik imgeleri binoküler görüş

sistemleri ile alınarak algoritma işletilmiş ve algoritmanın etkinliği tıkalı bölgeler içeren bir nesnenin modellenmesi ile kanıtlanmıştır.

Torabi ve Gupta (2011) yaptıkları bir çalışmada nesnenin durağan halde olduğu 9-DOF mobil bir robot kolu sistemin kullanıldığı 3B modelleme sistemi sunmuşlardır. Burada, 9 serbestlik derecesinin 6'sı ucunda lazer bir algılayıcı bulunan robot koluna aitken kalan 3 derece robot kolunun üzerinde bulunduğu mobil platforma aittir. O zamana kadar yapılan çalışmalar içinde bilindiği kadarıyla bu tarzda yapılan ilk sistem olarak sunulmuştur. Önceden sadece sınırlayıcı kutu ile nesnenin konumu ve yaklaşık boyutu dışında nesneye ve ortama ait bir bilginin bulunmadığı varsayılmaktadır. Sunulan sistem basitçe üç modülden oluşmaktadır. Bunlar; nesnenin tarama, kayıtlama ve güncellenmesini içeren 3B modelleme modülü, tarayıcının öbür pozuna karar veren bakış planlayıcı modülü ve çarpışmasız yol planı yapan yol planlayıcı modülüdür. Planlayıcı modülü, biri modelleme diğeri keşif için olan iki farklı NBV hesabının birleştirilmesi ile oluşturulmaktadır. Çarpışmasız yol planı modülünde çarpışma tespiti robot kol için ve mobil platform için ayrı ayrı hesaplanmaktadır. Hedef noktanın sayısı sıfır olana kadar algoritma işlemektedir. Mobil platformun 4 farklı pozisyonundan alınan 34 tarama ile algoritma test edilmiştir.

Torabi ve Gupta (2012) tarafından sunulan başka bir çalışmada ise donanım olarak 6-DOF robot kolu ve ucuna monte lazer derinlik algılayıcısı kullanılmıştır. Burada amaçlanan, nesnenin orijinal konumunda kalması ve hiçbir şekilde hareket ettirilmeden 3B modelinin elde edilmesidir. Sınırlayıcı kutu ile nesnenin yaklaşık konumu ve boyutu dışında ortam hakkında başka bir bilgi bulunmamaktadır. Tüm planlama üç ana başlıktan oluşmaktadır. Bunlardan birincisi nesnenin modellemesi için NBV bulma, ikincisi ortamın keşfi için NBV bulma ve sonuncusu çarpışmasız yol bulmak için algılayıcı tabanlı yol planlamadır. En iyi modeli elde etmek için NBV algoritması görüş alanı, çakışma ve nesnenin tıkalı olması gibi ana kısıtları içermektedir. Eğer bulunan NBV ulaşılabilir ise algılayıcı tabanlı yol planlayıcı istenen görüşe yol planlamasını yapar ve nesne yüzeyinden veri toplar. Çalışma uzayı bilinmediği için bazı aşamalarda keşif görüş planlayıcısı ulaşılabilirlik hakkında veri toplamak için kullanılmaktadır. Bu şekilde algoritma nesne modeli tamamlanana kadar veya planlayıcı daha fazla aşama bulamayana kadar devam etmektedir. Algoritmanın verimliliğini test etmek için 6-DOF robot kolu ve onun ucuna monte lazer tarayıcı kullanarak bazı nesneler için 3B modelleme yapılmıştır.

Khalfaoui vd. (2012) yaptıkları çalışma ile minimum bilgiye sahip olarak otomatik tarama gerçekleştiren bir yöntem sunmuşlardır. Sunulan yöntem, veri alımı sırasında sınırlayıcı kutunun değerlendirilmesine dayanır. Nesnenin sınırlayıcı kutu dahilinde yeterince kapalı olduğu kabullenmesi altında sınırlayıcı kutudan tarama pozisyonu elde

edilmektedir. Tarama için hedef noktalar yüzlerin merkezleri, kenarların orta noktaları ve köşeler olarak seçilmektedir. Böylece, tarayıcının parametrelerine bağlı olan bir *d* mesafesi kadar nesnenin uzağına algılayıcı konumlandırılmaktadır. Her veri alma aşamasında şu üç adım uygulanmaktadır: Sınırlayıcı kutu ve pozisyon tanımlamaları, tarayıcının konumlandırılması ve veri alınması ve sınırlayıcı kutunun boyutlarının güncellenmesi. Buradaki veri alımı rastgele sayıdaki pozisyonları baz almakta olup her yeni alınan veri öncekinden bağımsızdır. Ardışık iki taramada birbirini bağlayan tek bilgi sınırlayıcı kutunun boyutu olup taranan kısımların geometrisi değildir. Böylece zaman açısından maliyetli hesaplamalar olmadığı için yöntem basit bir haldedir. Önceki tarama ve şimdiki taramayı çakıştırmak için pozisyonlama sisteminin koordinatlarını kullanmaktadır. Daha sonra yeni alınan verinin kayıtlaması yapılmaktadır. Yöntem; 6-DOF robot kolu, bir döner tabla ve 3B algılayıcı kullanılan bir sistem ile test edilmiştir. Ayrıca, otomatik taramanın el ile taramaya kıyasla 2 ile 5 kat arası hız kazandırdığı bu çalışma ile kanıtlanmıştır.

Karaszewski vd. (2012) kültürel eserlerin yüksek çözünürlükte 3B modellenmesi için bir yöntem öne sürmüşlerdir. Sistem; NBV modülü, ters kinematik ile çarpışma kontrol modülü ve döner tabla ile endüstriyel kaynak robotu kullanılan fiziksel bir sistem ile gerçeklenmiştir. Hesaplanan NBV adaylarından uygun olanı seçilerek nokta yoğunluğu az olan bölgeler kalmayana kadar ya da bulunan bakış açısı ölçüm sistemi tarafından erişilir olmayana kadar devam etmektedir. Bu yöntemin diğer yöntemlere göre üstünlükleri, en iyi pozisyon ve yönelimi veren sonraki bakış açısının hesaplanması ve ters kinematik ve çarpışma tespiti için esnek yazılım modüllerinin oluşturulmasıdır. Sistemin en büyük avantajlarından biri ise, donanımdan bağımsız bir yapı oluşturularak istenilen donanımın istenilen şekilde değiştirilebilmesidir. Yöntem, Fanuc marka LRMate 200i model endüstriyel robot kolu, Kawasaki marka FS020N model endüstriyel robot kolu, döner tabla ve tarayıcı kullanılarak farklı boyutlardaki yaklaşık 40 kültürel eser üzerinde test edilmiştir. Kültürel eserler 3B modellendiği için zaman bir parametre olarak kullanılmayarak amaç yüksek çözünürlükte (milimetre karede 10 bin nokta) model elde etmek olmuştur. Taranan nesnelerin boyutları, çözünürlükleri ve erişebilirlik dereceleri tablo haline getirilmiştir.

Vasquez-Gomez vd. (2013) tarafından yapılan çalışmada NBV bulmaktan ziyade NBV'yi makul bir sürede bulmak amaçlanmaktadır. NBV adaylarının üretildiği ve bir amaç fonksiyonu ile değerlendirildiği arama tabanlı yaklaşımlar yüksek çözünürlüklü modeller üretmek için zaman açısından çok maliyetlidir. Burada sunulan metotta, hesaplama süresini azaltan hiyerarşik ışın izleme ile sekizli ağaç yapısı 3B modelleme için kullanılmıştır. Yapılan uygulamalarda, hesaplama zamanının en az bir derece azaldığı; bunun da daha çok kısıt kullanılarak daha fazla aday üretilmesine imkan sağladığı vurgulanmıştır. Sunulan yöntem, benzetim ortamında ve gerçek dünyada test edilerek verimliliği zaman cinsinden kanıtlanmıştır.

Khalfaoui vd. (2013) yaptıkları çalışmada taranacak nesne hakkında önceden minimum bilgiye sahip olarak 3B modelleme yapmayı amaçlamışlar ve sonraki en iyi bakışı hesaplamak için bir yöntem ileri sürmüşlerdir. Bu yöntemle, bakış açılarının otomatik olarak bulunmasının yanı sıra 3B modelleme süresinin de azaltılması planlanmıştır. Algoritma nesne üzerinde az görülen (barely visible) alanları iyi görülebilir (well visible) yapmayı amaçlayarak tasarlanmıştır. Önerilen yöntem iki aşamada gerçekleşmektedir. İlk aşama, görülebilirlik kontrolü ile hedef noktaların belirlenmesidir. İkinci aşama, ortalama kayma yöntemi (mean shift) ile bakış açısının hesaplanmasıdır. Yöntem için tek kısıt, ince disk veya kâğıt gibi çok ince ve keskin kenarları olan nesnelerin kullanılmamasıdır. Önerilen metot, KUKA marka KR16 model endüstriyel robot kolunun ucuna monte edilmiş projeksiyon tarayıcısı (projection scanner) kullanılarak oluşturulmuş bir sistem ile zorluk derecesi azdan çoğa değişen nesneler üzerinde denenmiştir. Elle tarama ve otomatik tarama arasındaki farkı göstermek için tamamlanma yüzdesi, yapılan ölçüm sayısı ve tarama zamanı parametreleri tablo haline getirilerek karşılaştırılma yapılmıştır.

Kriegel vd. (2013) yaptıkları çalışmada, bilinmeyen bir nesnenin modellemesini gerçek zamanlı olarak yapmayı amaçlamışlardır. Aslında bu çalışmanın temeli Kriegel vd. (2011) ve Kriegel vd. (2012) tarafından yapılan iki farklı çalışma ile yapılmıştır. Kriegel vd. (2011) tarafından yapılan çalışmada yüzey tabanlı bir yaklaşım kullanılmış olup Kriegel vd. (2012) tarafından yapılan çalışmada önceki çalışmaların eksikleri bulunmuş olup yüzey modeli, hacimsel model, karmaşık geometrili nesnelerin modellenmesi, delik tespiti, 6-DOF arama uzayı, algılayıcı belirsizliği, yol planlama ve keşif içeren bir sistem sunulmuştur. Bu çalışmada ise 3B modelleme problemi çözülürken, bilinmeyen çevrenin keşfi ve taranacak nesnenin 3B modellemesi eş zamanlı olarak yapılmaktadır. Nesnenin şekline bağlı olarak tarama yolları belirlenirken çarpışmasız yol planı, nesne için istenilen model kalitesine ulaşana kadar yapılmaktadır. Önerilen yöntemin diğer birçok yönteme göre üstünlüğü vardır. Bunlar; gerçek zamanlı olması, üçgen ağ kalitesinin kullanılması, nesne seçimi hakkında herhangi bir kısıt olmaması, tarama adaylarını belirlerken silindir/küre modelinin kullanılması yerine üçgen ağ yapısının kullanılması, lazer tarayıcıdan gelen verilerle eş zamanlı olarak olasılıksal uzay güncellemesinin yapılması, yüzey tabanlı ve hacim tabanlı yöntemin aynı anda kullanılması, bilgi kazancı (IG-Information Gain) ve yüzey kalitesinin sonraki en iyi taramayı (NBS-Next Best Scan) seçerken kıstas olarak kullanılması, yüzey kalitesinin sonlandırılma ölçütü olarak kullanılması şeklinde sıralanabilir. Algoritma, KUKA marka KR16 model endüstriyel robot kolunun ucuna monte edilmiş ScanControl 2700-100 Micro-Epsilon lazer tarayıcı kullanılarak kültürel eser nesneleri, gündelik kullanılan nesneler ve endüstriyel nesneler üzerinde denenmiştir. Kullanılan bütün bu nesneler için tarama sayısı, dakika cinsinden algoritmanın toplam çalışma süresi, ortalama göreceli nokta yoğunluğu, nesnenin tahmin edilen kavranma yüzdesi, nesnenin alt kısmını

kullanarak hesaplanan gerçek tamamlanma yüzdesi, nesnenin tamamlanma yüzdesi ve milimetre cinsinden koordinat hataları tablo halinde verilmiştir.

Vasquez-Gomez vd. (2014), yukarıda bahsedilmiş olan ve Vásquez-Gómez vd. (2009) tarafından yapılan çalışmanın eksiklilerini gidermek üzere yapılmış bir çalışma olup yine amaç etkili bir NBV hesabı yapmaktır. Sunulan algoritma, bakış açısı adaylarının kümesinin belirlenip her bakışın en iyi olup olmadığına karar veren arama tabanlı bir algoritmadır. Yapılan çalışmanın literatüre katkıları; makul olan bakış açılarının değerlendirilmesi için kullanılan fayda fonksiyonunun çok kapsamlı olması ve birçok sorunu aşması, bakış açısı adaylarının hesaplanmasındaki zamanı azaltan verimli arama stratejisi ve pozisyonlanma hatalarını minimize etmek için bakış açısı adaylarının tekrar değerlendirilmesidir. Modellemenin yapıldığı çevre voksel yapısı ile saklanmakta olup voksellerin etiketleri ve taşıdığı özellikler Vásquez-Gómez vd. (2009) tarafından yapılan çalışma ile aynıdır. Bakış açısı adaylarının hesaplanmasından sonra makul olan bakış açılarının değerlendirilmesi fayda fonksiyonu kullanılarak yapılmaktadır. Bu çalışmanın, Vásquez-Gómez vd. (2009) tarafından yapılan çalışmadan farkı algoritmanın pozisyonlama hatalarına karşı güvenilir olması için tekrar değerlendirilmesi yapılmaktadır. Daha sonra bütün bakış açıları sıralanarak en iyi olanı NBV olarak seçilmektedir. NBV, yeni bilgi sağlamayıncaya dek algoritma çalışmaktadır. Algoritma, pozisyonlama hatası olduğu kabul edilerek ve edilmeyerek iki ayrı durum için ve karmaşıklık dereceleri değişen nesneler için benzetim ortamında test edilmiştir. Bu iki durum için bakış açısı, kalite ve uzaklık parametreleri tablo haline getirilmiştir.

Aleotti vd. (2014) robotun bilinmeyen nesneleri ve ortamı 3B derinlik verisi kullanarak keşfi ile ilgili bir yöntem sunmuşlardır. Sunulan yöntemin amacı derinlik ölçen algılayıcıların fiziksel limitlerini aşmaktır. Bu yüzden birbirini tamamlayan karakteristik özelliklere sahip iki algılayıcı robot koluna monte halde kullanılmaktadır. İlk algılayıcı, orta menzilli bir algılayıcı olup ortamın bilinmediği durumda keşfin ilk aşamasında kullanılmaktadır. İkinci algılayıcı ise kısa menzilli olup ortamdaki nesneler keşfedildikten sonra NBV planlama için kullanılmaktadır. NBV planlama ortamın hacimsel gösterimi ile yapılmaktadır. Her nesnenin tamamlanmış modeli bu nesnelere ait orta ve kısa menzilli algılayıcılardan alınan veriler ile global kayıtlama kullanılarak elde edilmektedir. Ardışık kayıtlama yaklaşımından ziyade global kayıtlamanın daha iyi sonuçlar verdiği, ortamda bulunan birden fazla nesnenin modellenmesi ile deney ortamında kanıtlanmıştır. Deneyler; 6-DOF robot kolu, bir kısa menzilli algılayıcı ve bir uzun menzilli algılayıcı kullanılarak yapılmıştır.

Singh vd. (2015) tarafından sunulan çalışmanın amacı nesne hakkında minimum bilgiye sahip olarak 3B model elde etmektir. Sunulan yöntem, ulaşılamayan noktalardan

kaçınan uyarlamalı ortalama kaydırma algoritması kullanarak bilinmeyen nesnenin en görülebilir yüzeyine karar vermektedir. Aynı zamanda sunulan bu yöntem, nesne döner tabla üzerinde dönerken minimum sayıda görülebilir yüzey parçası kullanarak nesnenin modelini gerçek zamanlı olarak üretebilmektedir. Algılayıcı tarafından alınan veri görülebilirlik analizi ile yeterince görülebilir ve kısmen görülebilir olarak ayrılmakta ve elde edilen bu parçalar potansiyel NBV hesabında kullanılmaktadır. Yöntem, kısmen görülebilir yüzeylerin kavranmasını amaçladığı için en fazla tanımlayıcıya sahip olan küme NBV olarak seçilmektedir. Nesnenin 3B modelinin tamamlanması tamamen nesnenin karmaşıklığına bağlıdır. Burada Kinect ve lazer algılayıcılar kullanılarak gerçek nesnelerin 3B modellenmesi yapılmasına rağmen yöntem bütün algılayıcı türleri için uygulanabilir olmakla beraber yöntemin sağlamlığı diğer yöntemlerle bazı parametreler cinsinden karşılaştırılmasıyla kanıtlanmıştır.

Zhang vd. (2015) otonom bakış planlama için NBV hesabını ana problem olarak görmüşlerdir. MVC yöntemini NBV yönünü hesaplayan klasik metotlardan biri olarak görmüş olup NBV için hem yönelim hem de yön bulmak için görsel servo tabanlı bir yöntem sunmuşlardır. Görüntü tabanlı servo sisteminin kullanım amacı robotun uç noktasının pozisyonunun kontrolü iken, MVC yönteminin kullanım amacı Jacobian verisinin boyutunu azaltmaktır. İlk veri, ilk bakışın elle ayarlanması ile alındıktan sonra elde edilen nokta bulutu gürültüyü kaldırma, nokta bulutunu boyut olarak küçültme ve üçgensel ızgaralama yapmak için ön aşamaya sokulmaktadır. Her küçük üçgen ızgara kendi yüzeyinin vektörünü oluşturmakta olup MVC ilk bakış için bu yüzeylerin toplamı olarak hesaplanmaktadır. Böylece, ikinci bakış yönü için gereken değerler hesaplanmış olmaktadır. İkinci bakış için, pozisyona karar vermede ilk bakıştan alınan veriler üzerinden çıkarılan özellikler kullanılmaktadır. Robotun uç noktasının ikinci bakış açısına hareketi için görsel servolama kullanılmaktadır. Lazer algılayıcı ile veri alımı yapıldıktan sonra ilk taramada da uygulanan gürültüyü kaldırma, nokta bulutunu boyut olarak küçültme ve üçgensel ızgaralama ön aşamaları burada da uygulanarak önceden alınan veri ile bu taramada alınan veri çakıştırılmaktadır. Yüzeylerin toplamından elde edilen MVC tekrardan üçüncü bakış için yönelim hesabında kullanılmaktadır. Tarama aşaması sonlanana kadar bütün taramalar için bu döngü devam etmekte olup MVC değeri yeterince küçüldüğünde tarama işlemi sonlanmaktadır. Yöntem, konkav yüzeylerdeki küçük delikler hariç 3B modellemeyi yeterince iyi bir şekilde tamamlamaktadır.

Isler vd. (2016) tarafından bilgi kazancı tabanlı aktif hacimsel 3B modelleme çalışması yapılmıştır. Buradaki bilgi kazancı, gerçek zamanlı elde edilen olasılıksal ve hacimsel harita tabanlı NBV adaylar kümesinden hesaplanmaktadır. Görülebilirlik ve nesnenin görülmeyen taraflarından veri almayı içeren formülasyonlar sunularak değerlendirilmesi yapılmıştır. Bunlar, aynı zamanda robotun hareketinin maliyeti ile amaç

fonksiyonunda birleştirilmiştir. Burada, nesnenin yeni kısımlarını keşfetmeyi amaçlayan amaç fonksiyonunu optimize eden bakış NBV olarak seçilmektedir. Bu çalışmanın literatüre ana katkısı bilgi kazancı hesabında yeni formülasyonların kullanılmış olmasıdır. Sunulan bu yöntem hem benzetim ortamında hem de gerçek dünyada test edilmiştir. Gerçek dünyada yapılan test düzeneği o ana kadar yapılan çalışmalardan farklı olup mobil bir robotun üstünde robot kolu ve robot kolunun ucuna monte bir algılayıcı içermektedir. Sunulan yöntem, diğer robotik platformlara ve 3B modelleme problemlerine uygun olup açık kaynaklı olarak kullanılma sunulmuştur.

Rink ve Kriegel (2016) yaptıkları çalışma ile otonom modellemeye farklı bir bakış açısı getirmiş olup Monte Carlo kayıtlama yöntemi kullanmışlardır. Özellikle, lazer tarayıcılardan elde edilen sağlam, yüksek doğrulukta ve yoğunlukta veriler üzerinde çok sayıda poz tahmin teknikleri uygulanabilmektedir. Fakat tekniklerin çoğu lazer tarayıcılar için akıcı bir şekilde çalışmamaktadır. Bu boşluğu doldurmak amaçlı Monte Carlo Poz Tahmini kullanılmaktadır. Eğer bir ön bilgi varsa, bu yöntem öbür yöntemler kadar iyi çalışmakta olup veri alımı sırasında poz tahmini yapmaktadır. Otonom modelleme aşamasında, modelleme için lazer tarayıcı ile veri alma, üçgen örgü yapısı ve PVS güncelleme, NBV hesaplama ve yol planlama aşamaları sürerken her seferinde elde edilen modelin istenilen kaliteye ulaşılıp ulaşılmadığı kontrol edilmektedir. Model için istenilen kaliteye ulaşıldığı anda nesnenin tıkalı bölgelerinden veri alınacak şekilde tekrar pozisyonlaması yapılmaktadır. İlgili bölgeden özellik çıkarmak için lazer algılayıcı ile ölçüm yapılmaktadır. Senkron bir şekilde, partikül filtresi yinelemeli olarak gelen özelliklere örnekleme ve ağırlıklandırma yapmaktadır. Kayıtlama için lazer taramadan sonra tahmin edilen poz uygun hale getirilir. Monte Carlo ile kayıtlama yapıldıktan sonra otonom modelleme model tamamlanana kadar devam etmektedir. Poz tahmini için önceden bir bilgiye sahip olmanın etkisi incelenerek önceki yöntemler ve ICP algoritması ile karşılaştırılma yapılmaktadır. Altı serbestlik derecesine sahip bir robot kolu ve ucuna monte bir lazer profil algılayıcı kullanılarak yöntem üç farklı nesne üzerinde denenmiştir. Bu nesneler için yüksek kalitede 3B modeller otonom üretilerek yöntemin doğruluğu kanıtlanmıştır .

Tez kapsamında Kriegel vd. (2013) tarafından sunulan yöntemin uygulamasının yapılması planlanmıştır. Bunun sebebi, diğer makalelere göre daha kapsamlı olması, tez kapsamında gerçekleştirilecek amaçların makalede sunulan amaçlarla benzer olması ve makalede kullanılan lazer profil algılayıcının proje kapsamındaki kullanılan lazer algılayıcıya çok benzer olmasıdır. Ayrıca, üçgen ağ yapısının yanı sıra voksel yapısının kullanılması, yani yüzey tabanlı yaklaşımın yanında hacimsel yaklaşımın kullanılması, sistemin gerçek zamanlı olarak çalışması, tarama adaylarının belirlenirken sabit bir silindir veya küre yapısı kullanmak yerine üçgen ağ yapısının kullanılması, taranacak nesne

hakkında herhangi bir kısıt olmaması ve yüzey kalitesinin NBS seçiminde ve sonlandırma kriteri olarak kullanılması gibi avantajlar bu makalenin uygulama için seçilmesinde etkili olmuşlardır.

3. MATERYAL VE YÖNTEM

Bu bölümde, tez çalışması kapsamında kullanılan materyallerden ve yöntemlerden kısaca bahsedilmektedir. Çalışmada kullanılan materyaller hakkında detaylı bilgi, uygulama ortamının tasarımından bahsedilen Bölüm 6'nın Sistem Mimarisi alt başlığında, yöntem hakkındaki detaylı bilgi ise Bölüm 4'teki 3B Yüzey Modelleme Yöntemi ana başlığında verilmiştir.

3B yüzey modelleme yönteminin gerçeklenmesi için kullanılan materyaller, yani elektronik mühendisliğindeki adıyla donanımlar, endüstriyel robot kolu, lazer profil algılayıcı ve döner tabladır. Bu donanımların tek başlarına kullanılması mümkün değildir. Endüstriyel robot kolunun kullanımı ve kontrolü için bir iş istasyonu bilgisayar ve bir robot denetleyicisi bulunmaktadır. Robot kolunun programlanması için KRTerm adı verilen robot koluna özgü bir yazılım platformu ve yine robot koluna özgü AS programlama dilinin kullanılması gerekmektedir. İlerleyen bölümlerde de bahsedileceği üzere bu ihtiyaç farklı bir arayüz tasarımı ile mümkün kılınmıştır. Lazer profil algılayıcı, iş istasyonu bilgisayar üzerinden yazılan bir program ile kontrol edilebilmektedir. Bu program, Visual Studio ortamında C++ dili ile oluşturulan lazer profil algılayıcıya özgü kütüphanelerin kodlarını içermektedir. Endüstriyel robot kolu ile benzer şekilde, döner tablanın kullanımı için yine iş istasyonu bilgisayar ve döner tablanın denetleyicisinden faydalanılmaktadır. Döner tablanın kontrolü, yine Visual Studio ortamında C++ dili ile oluşturulan döner tablaya özgü kütüphanelerin kullanımı ile yapılmaktadır.

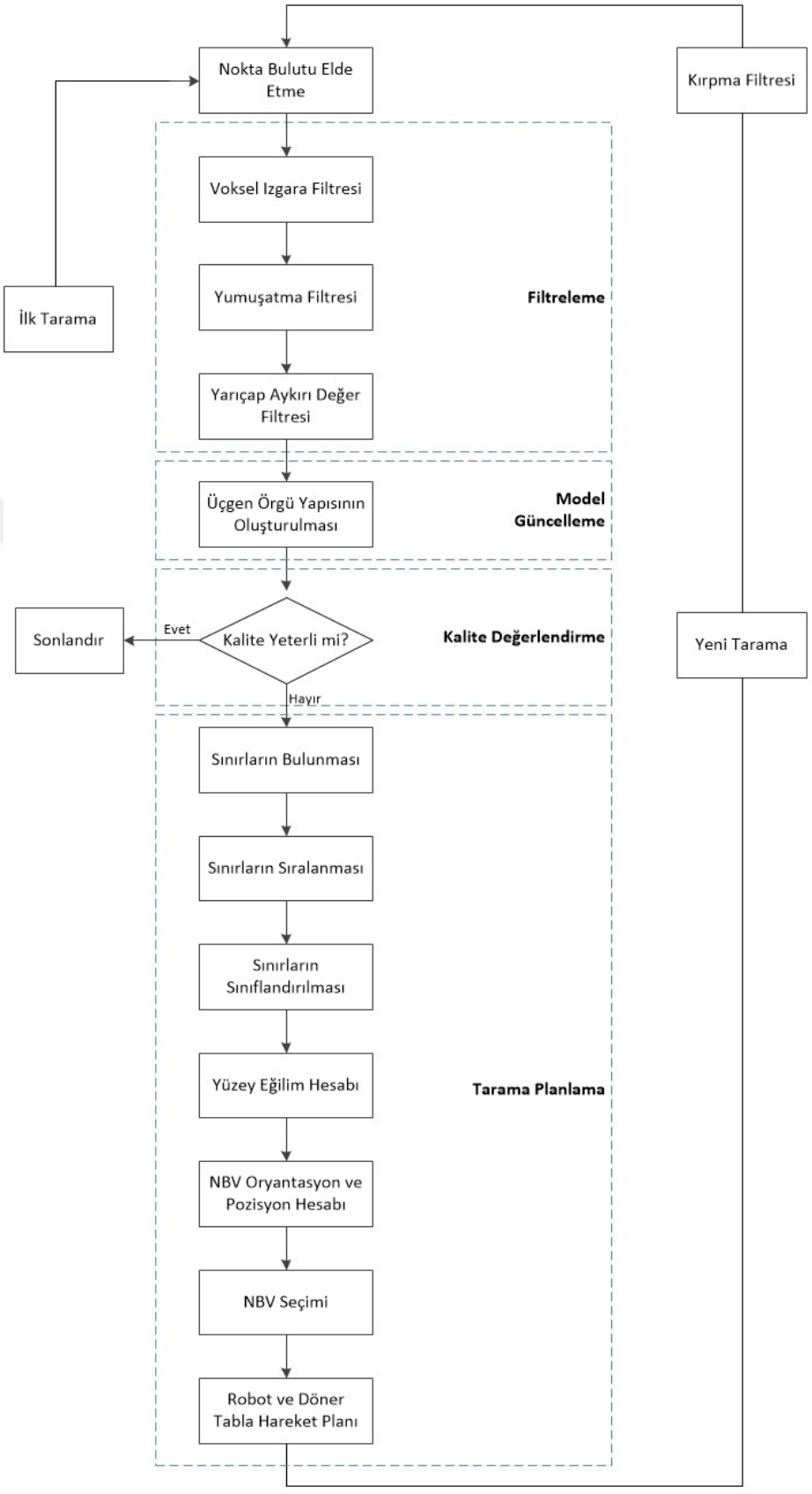
3B yüzey modelleme yöntemi, yukarıdaki donanımların kullanılması ve literatür araştırmasını içeren Bölüm 2'de referans olarak verilen makalenin gerçeklenmesi ile mümkün kılınmaktadır. Bu yöntem, gerçeklenen makale ile bire bir aynı olmamakla döner tabla kullanımı gibi farklılıklar içermektedir. Yaklaşık konumu ve boyutları bilindiği varsayılan bir nesne üzerinden ilk tarama ile veri alındıktan sonra sırasıyla filtreleme, üçgen örgü yapısı ile yüzey oluşturma, sınırların bulunması, sıralanması ve sınıflandırılması, yüzey eğilim tahmini, NBV adaylarının hesaplanması ve kalite kriterlerinin değerlendirilmesi ve NBV seçimi işlemleri gerçekleştirilmektedir. Bu yöntem, yinelemeli olarak devam etmekte olup sonlanma kriteri sağlandığı anda 3B modelleme işlemi tamamlanmış olmaktadır.

4. 3B YÜZEY MODELLEME YÖNTEMİ

Nesnelerin 3B modellemesinde temel problem sonraki en iyi bakışın (NBV) belirlenmesidir; çünkü NBV, nesne üzerinden veri alınacak sonraki noktayı belirlemektedir. Bu bölümde, sonraki en iyi bakışın bulunması için kullanılan yöntemden bahsedilmektedir.

3B nesne yüzey modellemede ilk aşama, nesne yüzeyinden ilk tarama nokta bulut verisinin alınmasıdır. İlk tarama verisini almak için, algılayıcı bakış pozisyonunun sisteme dışarıdan elle verilmesiyle 3B yüzey modelleme işlemi başlamaktadır. İlk tarama ile nesne yüzeyinden veri alındıktan sonra, NBV algoritması uygulanmadan önce nokta bulutu üzerinde NBV algoritmasının daha etkili ve verimli çalışmasını sağlamak amacıyla birtakım filtreler uygulanmaktadır. Uygulanan bu filtreler sayesinde nokta bulutundaki gürültülü noktalardan doğabilecek sıkıntılar önlenmektedir. Daha sonra, gerçeklenmek üzere seçilen makaledeki yöntem NBV algoritması olarak uygulanmaktadır. Bahsedilen bu algoritma, filtrelenmiş nokta bulutu üzerinde üçgen örgü yapısı ile yüzey oluşturarak başlamaktadır. Elde edilen bu parçalı modelin final model olup olmadığının bazı ölçütlerin kullanımı ile kontrol edilmesi gerekmektedir. Eğer, 3B parçalı nesne modeli belli ölçütleri sağlıyor ise, akış sonlanarak nesnenin 3B final modeli elde edilmiş olmaktadır. Aksi takdirde akış şu şekilde devam etmektedir: Elde edilen üçgen örgü yapısı ile nesneye ait sınırlar bulunarak, sırasıyla sınırların uzunluk olarak sıralanması ve alt, üst, sağ ve sol olarak sınıflandırılması yapılmaktadır. Sınıflandırılan sınırlar için yüzey eğilim tahminleri yapılarak NBV adayları için pozisyon ve yönelim hesabı yapılmaktadır. Sınıflandırılmış sınırlar içinden en çok nokta içeren sınıra ait NBV adayı NBV olarak seçilmektedir. Daha sonra, seçilen NBV'ye göre, endüstriyel robot kolu ve döner tabla için hareket planlaması yapılmaktadır. Bulunan pozisyona, lazer profil algılayıcının veri alması için, endüstriyel robot kolunun ve döner tablanın hareketi sağlandıktan sonra nesne yüzeyinden yeni nokta bulutu alınmaktadır. Bu aşamada elde edilen nokta bulutu ile önceki nokta bulutu birleştirilerek filtreleme ile başlayan 3B yüzey modelleme yönteminin aşamaları, sırayla ve yinelemeli olarak devam etmektedir. Elde edilen 3B modelin belirli ölçütleri sağlamasında kadar devam eden bu işlemler için Nokta Bulut Kütüphanesi'nin (PCL) ilgili fonksiyonları kullanılmaktadır.

NBV algoritması için akış diyagramı Şekil 4.1'de gösterilerek bloklarda yapılan işlemler ilgili başlıklarda detaylandırılmaktadır. Bahsedilen akış diyagramında mavi kesikli çizgiler ile gösterilen kutular ilgili blokların genel başlıklarını temsil etmektedir.



Şekil 4.1. NBV Algoritması İçin Akış Diyagramı .

NBV algoritması üç farklı nesne üzerinde uygulanmaktadır. Bu nesnelerden biri fil şeklinde, diğeri endüstriyel bir parça ve bir değeri çanak şeklinde olup Şekil 4.2 ile gösterilmiştir.



(a) (b) (c)

Şekil 4.2: Modellemede Kullanılan Nesneler: (a) Fil (b) Endüstriyel Parça (c) Çanak

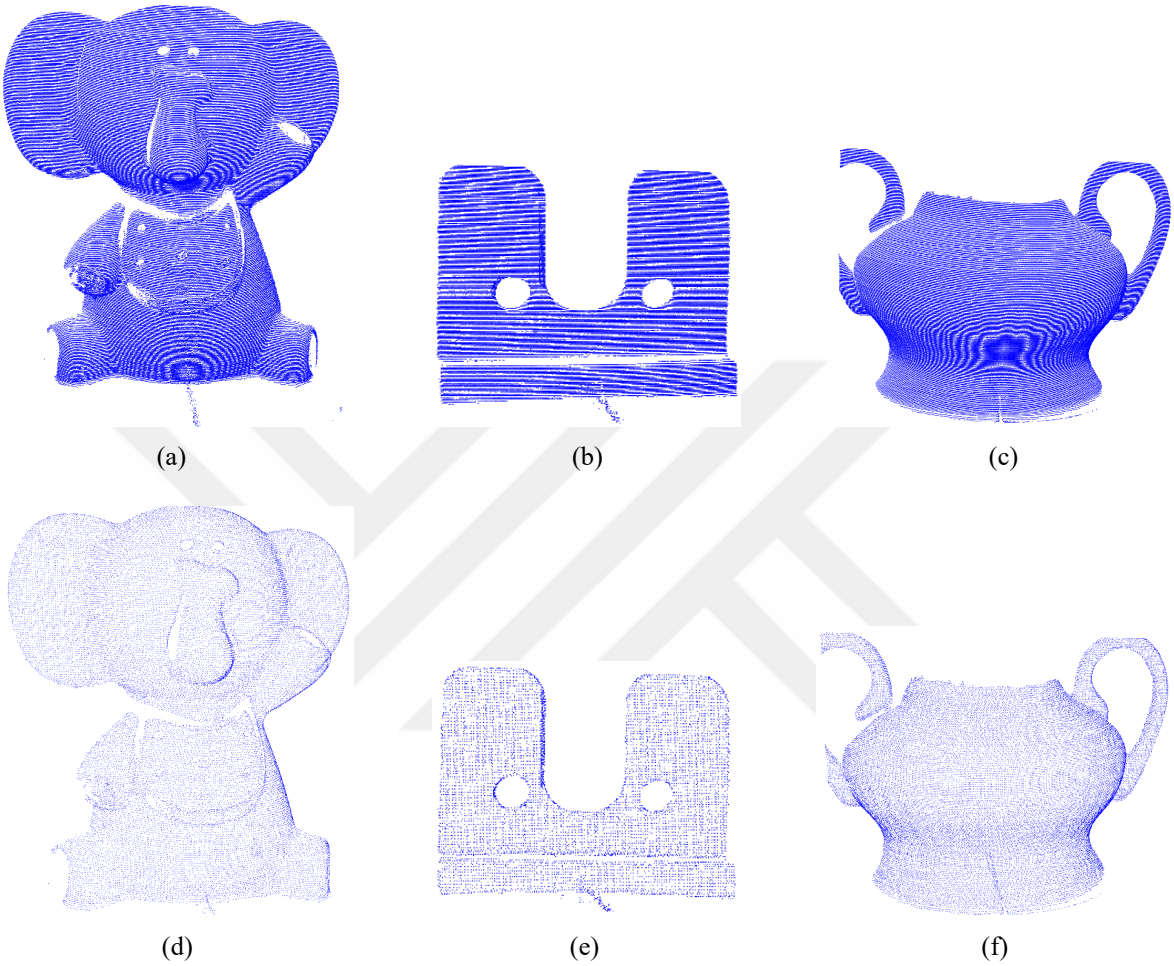
4.1 Filtreleme İşlemi

Nokta bulutlarında filtreleme; yanlış ölçümlerden kaynaklı gürültülü noktaları elemek, tekrar eden veya birbirine çok yakın olan noktaları elemek suretiyle nokta yoğunluğunu azaltmak ve belirli bölge içinde kalan noktalar üzerinden işlem yaparken kullanılmayan noktaları nokta bulutundan silmek gibi amaçlarla kullanılabilmektedir. Bu amaçları gerçekleştirmek için voksel ızgara filtresi, yarıçap aykırı değer filtresi, kırpma filtresi ve yumuşatma filtresi kullanılmaktadır. Öncelikle, her yeni alınan veri için kırpma filtresi uygulanmaktadır. Daha sonra ise önceki nokta bulutuyla birleştirilen nokta bulutu üzerine sırasıyla; voksel ızgara filtresi, yumuşatma filtresi ve yarıçap aykırı değer filtreleri uygulanmaktadır.

4.1.1 Voksel ızgara filtresi

Vokselleştirilmiş ızgara yapısı kullanılarak nokta bulutundaki nokta sayısını azaltmak için kullanılan bir filtredir. Nokta sayısını azaltma işlemi her bir vokseli bir nokta ile temsil ederek yapılmaktadır. Eğer bir vokselin içerisine birden fazla nokta düşüyorsa, o noktaları vokselin merkezindeki tek bir nokta temsil etmektedir. Bu sayede vokselin ayrıt büyüklüğü ile doğru orantılı olarak nokta bulutunun yoğunluğu azalmaktadır. Ayrıca, nokta bulutunda bölgeler arası yoğunluk farkı varsa bu filtre ile daha eşit bir dağılım elde edilmektedir. Bu filtrenin kullanılmasıyla nokta yoğunluğu azaldığından nokta bulutu

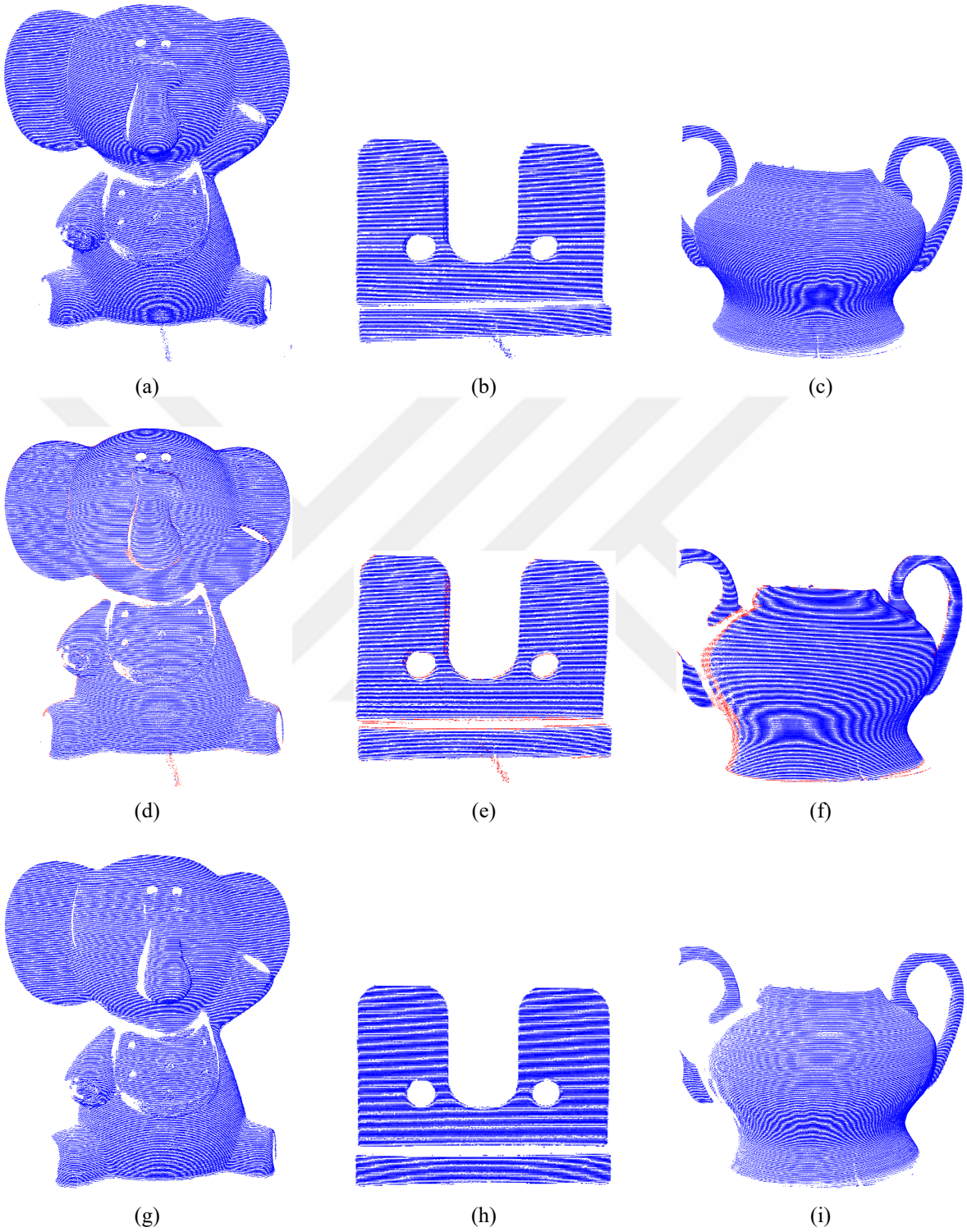
üzerinde gerçekleştirilecek olan işlemler için hız kazanılması en önemli avantajlardan biridir. Şekil 4.3'te fil, endüstriyel parça ve çanak nesneleri için voksel filtresi uygulanmadan önceki ve uygulandıktan sonraki nokta bulutları görselleştirilmiştir.



Şekil 4.3: Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Voksel Izgara Filtresi Uygulanmadan Önceki (a,b,c) ve Uygulandıktan Sonraki Nokta Bulutları (d,e,f)

4.1.2 Yarıçap aykırı değer filtresi

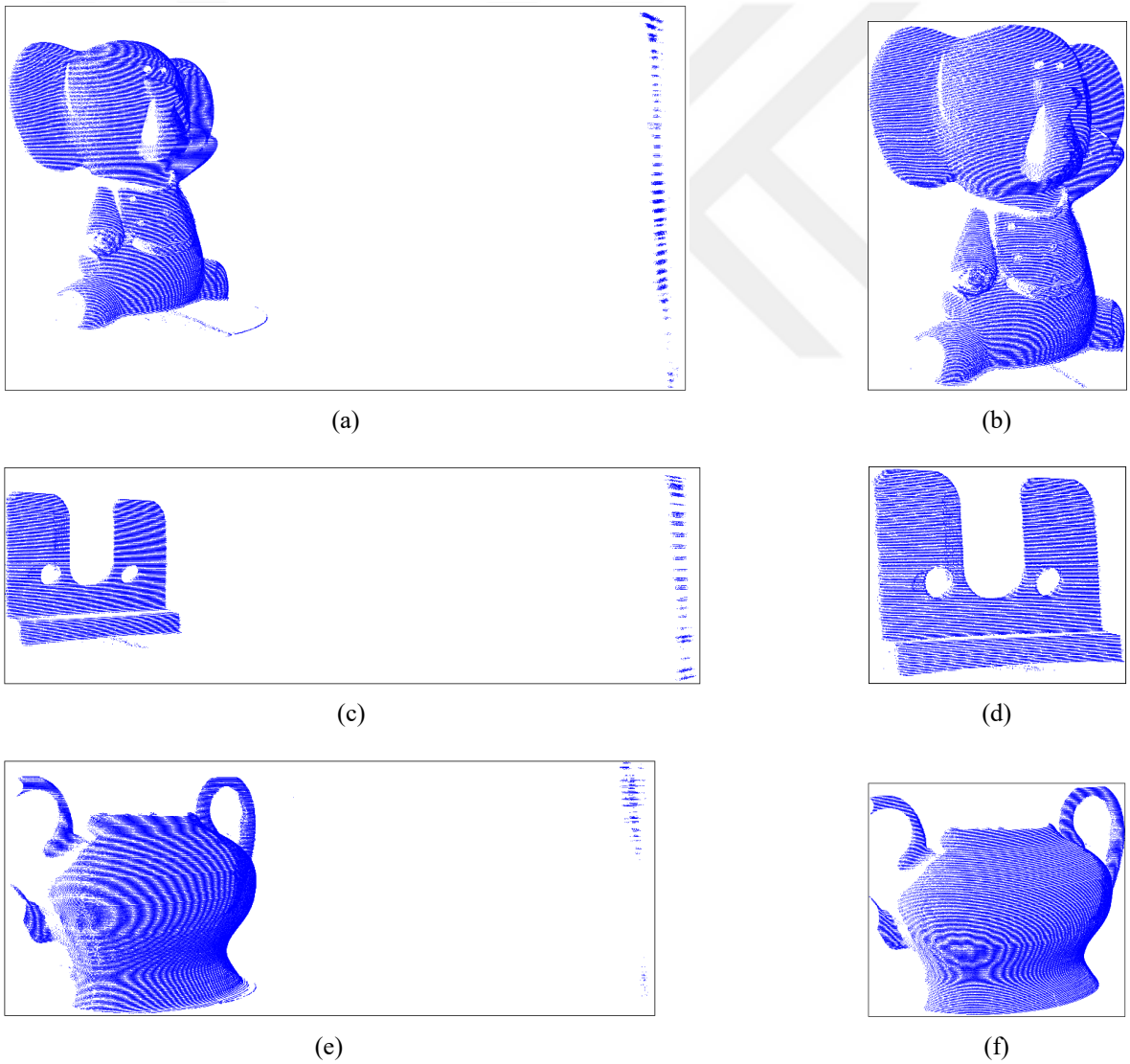
Nesne yüzeyinden kaynaklı yansıma veya algılayıcının ölçüm aralığının dışına çıkılması gibi sebepler yanlış ölçümlere neden olabilmektedir. Yanlış ölçümlerden kaynaklı gürültülü noktaları nokta bulutundan kaldırmak amacıyla yarıçap aykırı değer filtresi kullanılmaktadır. Filtre, belli bir yarıçap içerindeki nokta sayısına bakar ve nokta sayısı olması gereken minimum nokta sayısından az ise o noktayı gürültü kabul ederek nokta bulutundan siler. Şekil 4.4'te fil, endüstriyel parça ve çanak nesneleri için yarıçap aykırı değer filtresi uygulanmadan önceki nokta bulutları, uygulandıktan sonraki nokta bulutları ve kırmızı renk ile bu iki nokta bulutunun farkları görselleştirilmiştir.



Şekil 4.4: Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Yarıçap Aykırı Değer Filtresi Uygulanmadan Önceki (a,b,c), Uygulandıktan Sonraki Nokta Bulutları (g,h,i) ve İki Nokta Bulutunun Farkı (d,e,f)

4.1.3 Kırpma filtresi

Verilen kesim değerlerine göre nokta bulutu üzerinde kırpmalar yapmak amacıyla kullanılan bir filtredir. Bu filtre, lazer algılayıcının ölçümle elde ettiği, nesnenin arka planında kalan noktaları ve yanlış ölçümden kaynaklı noktaları kırpmak için kullanılmaktadır. Filtreleme; *x*, *y* ve *z* yönlerinde ayrı ayrı yapılabileceği gibi, istenilen herhangi iki yönde ya da bütün yönlerde yapılabilir. Filtreleme yapılmak istenen yönlerde alt ve üst sınır değerleri ayarlandıktan sonra, alt değerin altında ve üst değerin üstünde kalan noktalar nokta bulutundan silinmektedir. Şekil 4.5'te fil, endüstriyel parça ve çanak nesneleri için kırpma filtresi uygulanmadan önceki ve uygulandıktan sonraki nokta bulutları görselleştirilmiştir.



Şekil 4.5: Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Kırpma Filtresi Uygulanmadan Önceki (a,c,e) ve Uygulandıktan Sonraki Nokta Bulutları (b,d,f)

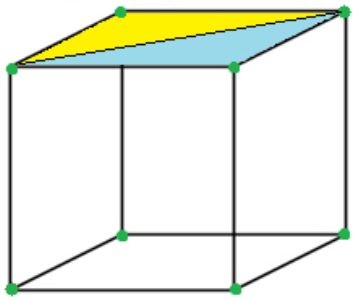
4.1.4 Yumuşatma filtresi

Ölçüm hatalarından meydana gelen düzensiz noktaların, örnekleme işlemi ile nokta bulutundan silinmesini sağlayan ve daha düzgün ve pürüzsüz bir model elde etmek amacıyla kullanılan bir filtredir. Filtreleme işlemi, MLS (Moving Least Square) algoritması ile yapılmaktadır. Özellikle, birden fazla nokta bulutunun üst üste çakışması ile elde edilen nokta bulutlarındaki örtüşme hataların bu filtre ile eliminasyonu sağlanmaktadır. Bu filtrenin kullanılması daha düzgün bir üçgen örgü yapısının elde edilmesini sağlamaktadır, fakat işlem yükü fazla olduğu için özellikle gerçek zamanlı işlemler yapılırken zaman kaybına sebep olmaktadır.

4.2 Üçgen Örgü Yapısı İle Yüzey Oluşturma

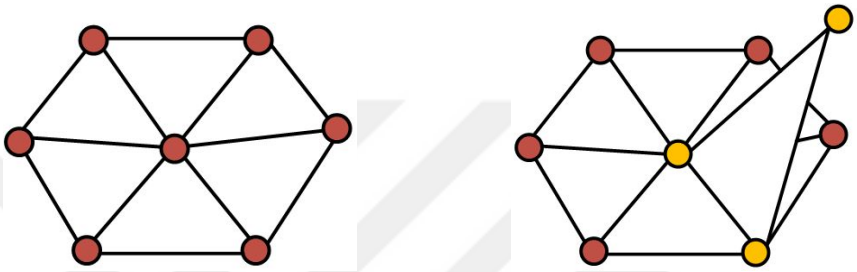
Üçgen örgü yapısı, yüzeyleri temsil etmek için kullanılan bir metottur. Birçok geometri algoritmasında, üçgen örgü yapısının herhangi bir matematiksel yapısı olmayan üçgenler topluluğu olarak düşünülmesine rağmen, her üçgen örgü yapısı parçalı doğrusal yüzey temsilinin bir bölümü olarak ifade edilmektedir (Botsch vd., 2010). Üçgen örgü yapısı geometrik modellemelerde ve bilgisayar grafiklerinde yaygın olarak kullanılmaktadır (Fabio, 2003).

Üçgen örgü yapısı; köşe (vertice), kenar (edge) ve yüz (face) yapılarını içinde barındırmaktadır. Köşe, her biri *x*, *y* ve *z* koordinatlarını içeren 3B noktalar kümesinin her bir elemanı; kenar, iki köşenin birleşmesiyle oluşan yapı; yüz, üçgen yapısı kullanıldığı için üç kapalı kenardan oluşan yapı; yüzey ise benzer özelliklere sahip yüzlerin birleşmesiyle oluşan yapı olarak tanımlanmaktadır (Anonim, 2017a). Bu yapılar bir şekil üzerinde örneklendirilmek istenirse, Şekil 4.6'da gösterilen yapıda, yeşil noktalar köşeleri, siyah çizgiler kenarları, siyah çizgilerin arasında oluşan sarı ve mavi bölgeler yüzleri, sarı ve mavi yüzün birlikte oluşturduğu yapının tamamı ise yüzeyi oluşturmaktadır.



Şekil 4.6: Köşe, Kenar, Yüz ve Yüzey Yapıları

Bir yüzeyi temsil eden üçgen örgü yapıları birbirlerini kesmez ve süreklidir. Bu yapılarda, 3B modelleme yönteminin bu aşamadan sonraki aşaması olan sınır analiz işlemlerini yapabilmek için manifold olma özelliğinin mutlaka sağlanması gerekmektedir. Bu özelliğin sağlanabilmesi için her kenarın bir veya en fazla iki yüz tarafından paylaşılması gerekmektedir. Şekil 4.7'de manifold olma özelliğini sağlayan ve sağlamayan yapılar görselleştirilmiştir. Bu şekildeki ilk yapı manifold olma özelliğini sağlarken, ikinci yapı sarı renk ile gösterilen köşe noktalarının oluşturduğu üçgen yüzünden bir kenarın üç yüz tarafından paylaşılmasına sebep olmakta ve bu özelliği sağlanmasına engel olmaktadır.



(a) (b)

Şekil 4.7: Manifold Olan ve Olmayan Yapılar (a) Manifold Yapı Örneği (b) Manifold Olmayan Yapı Örneği

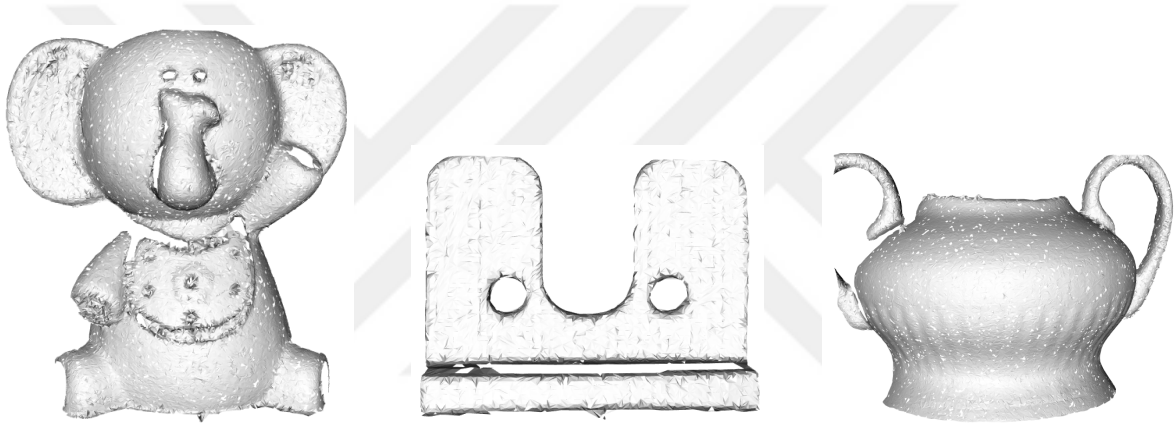
Üçgen örgü yapısı üretmek için kullanılan algoritmalar; aç gözlü üçgenleme algoritması (greedy triangulation algorithm), varyasyonel üçgenleme algoritması (variational triangulation algorithm) ve artımlı üçgenleme algoritması (incremental triangulation) olarak sıralanabilmektedir (Botsch vd., 2010). Tez kapsamında kullanılan üçgenleme algoritması ise aç gözlü üçgenleme algoritmasıdır.

Üçgen örgü yapısı elde etmek için, açık kaynak kodlu PCL içinde bulunan açgözlü yüzey üçgenleme algoritması kullanılmıştır. Bu algoritma, ilk noktadan başlayarak mümkün tüm saçak noktaların (fringe points) genişletilmesiyle devam etmektedir ve bir ya da daha fazla taramayla elde edilmiş olan organize olmamış nokta bulut yapısı (unorganized point cloud structure) için kullanılmaktadır (Marton vd., 2009). Tez çalışması kapsamında da, birden fazla tarama ile birlikte organize olmamış nokta bulut yapısı için kullanılmıştır. Ayrıca, bölgeler arası yumuşak geçişlerin olması durumunda algoritma daha verimli çalışmaktadır. Yani, bu algoritma çalıştırılmadan önce Filtreleme İşlemi başlığında bahsedilen yumuşatma filtresi kullanıldığında üçgen örgü yapısı daha düzgün oluşturulacaktır.

Açgözlü yüzey üçgenleme algoritması şu şekilde çalışmaktadır: Üçgenlemeye başlanacak nokta belirlenip, o nokta etrafında belirli bir yarıçap komşuluğunda (R_n) olan

noktalar kullanılarak bir normal bulunur. Daha sonra normale dik bir düzlem oluşturulur. Üçgenlemeye başlanacak noktanın etrafında başka bir yarıçap komşuluğunda (R_m) olan noktaların bu düzleme iz düşümü alınarak, üçgenlemeye başlanan nokta ile ona en yakın komşu nokta arasındaki uzaklık belirlenir (Burada en yakın uzaklığı kullanmasından ötürü, algoritma açgözlü olarak adlandırılmaktadır). Daha sonra en yakın uzaklık ile komşuluk büyüklüğünü denetleyen μ parametresi çarpılarak yeni bir yarıçap değeri (R_r) elde edilir. Bu yarıçap içerisinde bulunan komşu noktalar uzaklıklarına göre en yakın olandan başlanarak üçgen örgü yapısına dahil edilirler (Marton vd., 2009).

Yukarıdaki açıklamalar ve parametreler kullanılarak sırasıyla fil, endüstriyel nesne ve çanak için elde edilen üçgen örgü yapıları Şekil 4.8 ile görselleştirilmiştir.



(a) (b) (c)

Şekil 4.8: Üçgen Örgü Yapısı (a) Fil İçin (b) Endüstriyel Nesne İçin (c) Çanak İçin

4.3 Sınırlar Üzerinde Yapılan İşlemler

Sınırlar üzerinde yapılan işlemler; sınırların bulunması, sıralanması ve sınıflandırılmasıdır. Bu işlemler, sayılan sıra ile yapılmaktadır.

4.3.1 Sınırların bulunması

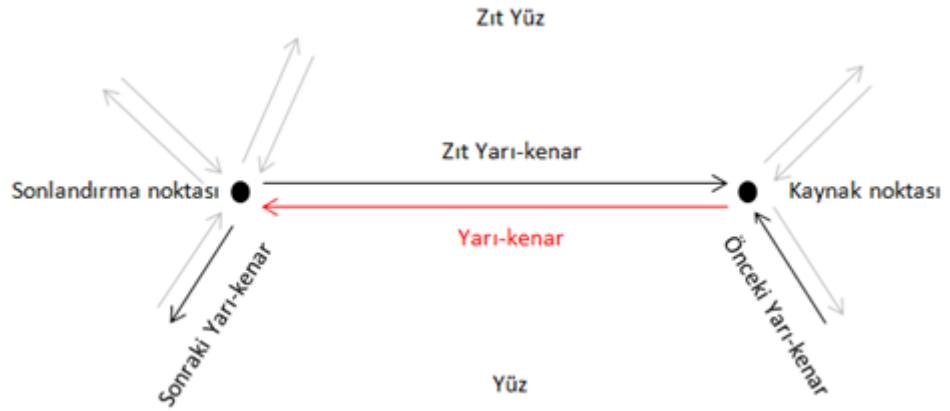
Cismin modellenmemiş yüzey bölgelerinin saptanmasında, sınır tespiti yöntemleri büyük önem arz etmektedir. Sınırların bulunmasında amaç, sınırdaki üçgen örgü yapısının analizini yaparak, cisim yüzeyinde nokta bulutu temin etme gerekliliği olan bölgeleri tespit etmektir; çünkü tamamlanmış bir modelde sınır olmaması beklenmektedir. Yani, sınırların var olması modelin eksik olduğu anlamına gelmekle beraber bazı bölgelerin yeniden taranması gerektiği anlamına da gelmektedir.

Üçgen yapısından sınır olup olmadığını anlamak için yüz kontrolü yapmak gerekir. Üçgen Örgü Yapısı İle Yüzey Oluşturma başlıklı Bölüm 4.2'de bahsedilen manifold olma şartını sağlayan bir üçgen örgü yapısında, eğer bir kenar iki tane yüzün ortak kenarı ise, orada sınır yoktur. Ancak, kenar sadece bir yüze ait ise, orada sınır olduğu anlamına gelmektedir.

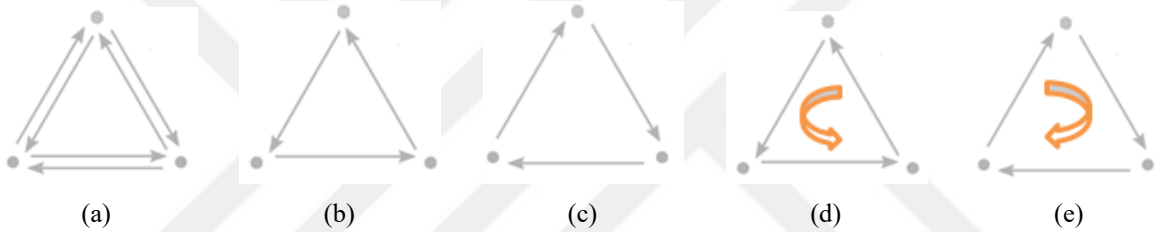
Sınır tespitinde, PCL içerisinde bulunan fonksiyonlar kullanılmaktadır. Sınır tespitinde kullanılan algoritmanın daha iyi anlaşılması açısından bazı kavramlardan bahsetmek faydalı olacaktır. Üçgen örgü veri yapısı için yüz tabanlı veri yapısı (face-based data structure), kenar tabanlı veri yapısı (edge-based data structures), yarı kenar tabanlı veri yapısı (half egde-based data structure) ve yönlendirilmiş kenarlı veri yapısı (directed-edge data structure) olmak üzere dört çeşit veri yapısı mevcuttur (Botsch vd., 2010). Açgözlü üçgenleme algoritması ile elde edilmiş olan üçgen örgü yapısı, yüz tabanlı veri yapısı ile ifade edilmektedir. Yani, üçgenleri oluşturan kenarlar yönsüz olarak hafızada tutulmaktadır. Fakat sınırların bulunması için yapılacak olan analizde kenarların yönlü olması gerekmektedir.

Yarı kenar tabanlı veri yapısı, bu ihtiyacı karşılayan bir model sunmaktadır. PCL içerisinde, üçgen örgünün, yarı kenar formunda tanımlandığı yarı kenar veri yapısı mevcuttur (Sälzle, 2012). Yarı kenar tabanlı veri yapısı, üçgen örgü yapısındaki bir kenarın farklı yönlü olan iki kenarla temsil edilmesinde kullanılmaktadır. Böylece örgü yapısına ait tüm bilgi yarı kenarlarda saklanmaktadır. Bu veri yapısında, sonraki yarı kenar (saat yönünün tersi doğrultusunda), önceki yarı kenar, yarı kenarın işaret ettiği köşe noktası, zıt yarı kenar yapısı ve komşuluktaki yüz bilgilerine ulaşılmaktadır (Botsch vd., 2010). Her üçgen örgü yapısındaki bu bilgiler belirli indeks numaraları ile kodlanırlar. Geçersiz yüz indeksine (-1) sahip yüzeylerin yarı kenarları sınır yarı kenar ve bu kenarlara bağlı köşe noktaları da sınır noktası olarak kabul edilmektedir. Yüzlerin yönelimleri, yüzü çevreleyen yarı kenarlardan belirlenmektedir. Bu veri yapısı Şekil 4.9'daki gibi görselleştirilmiştir.

Şekil 4.9'da görüldüğü üzere her bir kenar zıt yönlü iki kenar ile temsil edilmektedir. Her bir vektör önceki yarı kenar, sonraki yarı kenar, sonlandırma noktası, kaynak noktası, zıt yarı kenar ve yüz ile ilişkilidir. Üçgen örgü yapısında, sınır tespiti yaparken yarı kenar yapısını veya zıt yarı kenar yapısını kullanmak fark etmeyecektir; çünkü her iki kenar yapısıyla geçersiz indekse sahip yüze ulaşılabilmektedir. Ayrıca, yarı kenar yapısında ya da zıt yarı kenar yapısında okları takip ederek başlanılan noktaya dönülmesi gerekmektedir. Sadece dönüş yönünde farklılık olup yönü zıttır. Şekil 4.10'da bu durumun görsel olarak açıklanması, yarı kenar veri yapısı ve zıt yarı kenar veri yapısının görselleştirilmesi yapılmıştır.



Şekil 4.9. Örgü Yapısında Yarı Kenar Gösterimi .



Şekil 4.10: Yarı kenar, zıt yarı kenar ve okların dönüş yönleri: (a) Yarı kenar ve zıt yarı kenar, (b) Yarı kenarlar (İç taraf), (c) Zıt yarı kenarlar (dış taraf), (d) Yarı kenar için okların dönüş yönü, (e) Zıt yarı kenar için okların dönüş yönü

PCL kullanılarak sınır bulma sürecinde, ilk aşama, önceden de bahsedildiği gibi, yüz tabanlı veri yapısı ile tanımlanan üçgen örgü yapısını yarı kenar tabanlı veri yapısına çevirmektir. Daha sonra, ilk indeksten (0. indeks) başlayıp son indekse (yarı kenar üçgen örgü yapısının boyutu) kadar giden bir döngü ile sınır analizi yapılmaktadır.

Eğer yarı kenar sınırsa ve daha önce ziyaret edilmemiş ise, analize başlanılan sınırın indeksi aynı zamanda sınırın bitişindeki elemanın indeksine atanmıştır; çünkü sınır başladığı yerde bitecektir. Böylece, sınırın nerede sonlandığı bu indeksten kontrol edilebilir. Sınır olan ve ziyaret edilmemiş yarı kenar indeksini bulunduktan sonra yarı kenar vektörünün kaynak köşe noktasının indeksi elde edilmiştir. Aslında, burada önemli olan yarı kenar yapısından ziyade köşe noktalarıdır. Çünkü algoritma sonlandığında çıktı olarak sadece köşe noktaları kalacaktır. Yarı kenar vektörünün doğduğu köşe noktasının indeksi kullanılıp nokta bulutunda hangi nokta olduğu bulunarak depolanmaktadır. Bu şekilde tüm sınırlar bulunduktan sonra algoritma sonlanır. Bu algoritma, üçgen örgü üzerindeki tüm

sınırları döndürmektedir. Özellikle, ilk taramalardan sonra, model eksik olduğundan dolayı sınır sayısının çok fazla çıkması olağan bir durumdur. Daha kolay anlaşılması açısından Algoritma 1'de sınır bulma ile ilgili algoritma gösterilmiştir.

Algoritma 1 Üçgen Örgü Yapısından Sınırların Bulunması

```
Yüz tabanlı veri yapısını yarı kenar tabanlı veri yapısına çevir
for ilk indeksten yarı kenar üçgen örgü yapısının boyutuna kadar git do
    if yarı kenar üçgen örgü yapısı sınırsa ve daha önce ziyaret edilmemişse then
        dıştaki yarı kenar yapısının indeksini bul
        son indekse dıştaki yarı kenar yapısının indeksini ata
        do
            yarı kenar vektörünün doğduğu köşe noktasının indeksini bul
            bulduğun indeksten nokta bulutunda karşılık gelen noktayı bul
            Sınır noktalarını boundaryCloud vektörüne doldur
        while noktanın indeksi son noktanın indeksine eşit olmadıkça
            boundaryCloud vektörlerini boundaryCloudVector vektörüne doldur
    end if
end for
return tüm sınırlar
```

4.3.2 Sınırların sıralanması

Sınırların sıralamasında amaç, sınırları büyükten küçüğe sıralamaktır; çünkü büyük sınırlar üzerinden yapılacak işlemlerde nokta bulutuna olacak bilgi kazancı daha fazla olacaktır. Sınırların çevresi, bulunan her bir sınır için noktalar arası Öklid uzaklıkları toplanarak bulunmaktadır. Bulunan bu değerlere göre sınır zincirleri sıralanmaktadır.

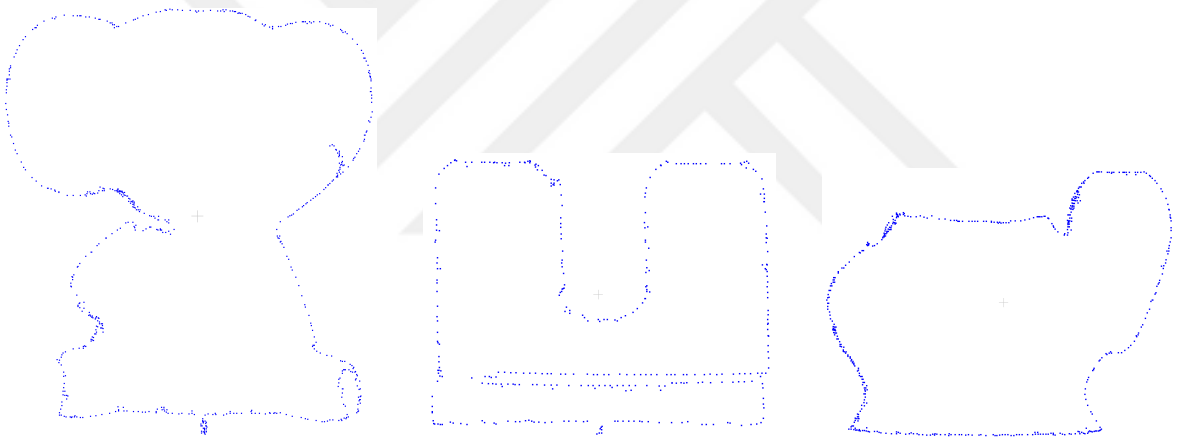
Daha kolay anlaşılması açısından Algoritma 2'de sınır bulma ile ilgili algoritma gösterilmiştir.

Filtrelenmiş nokta bulutu üzerindeki üçgen örgü yapısından sınırlar bulunduktan ve uzunluklarına göre sıralandıktan sonra ön yüzlerinden taranan fil, endüstriyel parça ve çanak nesnelerine ait en uzun sınırlar Şekil 4.11'de görselleştirilmiştir. Görselleştirilen bu sınırlardan endüstriyel nesneye ait olanın orta kısmından yeteri kadar veri alınamadığı için o bölge de sınıra dahil olmuştur, çanak nesnesinin ise kulplarından biri lazer profil algılayıcının ölçüm aralığının dışına çıkması ve o bölgeden az veri alınması sebebiyle sınır

Algoritma 2 Sınırların Sıralanması

for ilk indeksten sınır vektörünün boyutuna kadar git **do**
  for ilk indeksten uzaklığı hesaplanan sınır vektörünün nokta sayısına kadar git **do**
    sınır vektöründeki şimdiki ve sonraki nokta arasındaki Öklid uzaklığını hesapla
    uzaklık değişkenine hesaplanan Öklid uzaklığını ekle
  end for
  hesaplanan uzaklık değişkenini vektöre doldur
end for
uzaklık değişkenlerine göre sınırları sırala
return tüm sıralanmış sınırlar

bulunamamıştır. Bir sonraki aşama olan sınırların alt, üst, sağ ve sol olarak sınıflandırılması için bu aşamada elde edilmiş olan en uzun sınırlar kullanılmaktadır.



(a) (b) (c)

Şekil 4.11: Ön Yüzlerinden Taranan Nesnelerin En Uzun Sınırları: (a) Fil İçin (b) Endüstriyel Parça İçin (c) Çanak İçin

4.3.3 Sınırların sınıflandırılması

Uygulaması gerçekleştirilen makalede (Kriegel vd., 2013), tarama noktaların tespitinde yüzey eğilim tahmini kullanılmaktadır. Bu metodun uygulanması için, sınırların sınıflandırılmasına ihtiyaç duyulmaktadır. Elde edilmiş nesneye ait üçgen örgü yapısının sıralanmış sınırları; alt sınır, üst sınır, sağ sınır ve sol sınır olmak üzere 4 farklı tipte sınıflandırmaya tabi tutulacaktır. Sınıflandırmanın temeli, sınır üzerindeki ardışık noktaların x ve z eksenleri (robot taban eksenine göre x bileşeni yatay ekseni ve z bileşeni dikey ekseni ifade etmektedir) ile yaptığı açıların hesaplanmasına dayanmaktadır.

Sınırlar için yapılabilecek olan 4 farklı sınıflandırma şu şekilde özetlenebilir:

- Eğer ardışık noktalar arası vektör ile x ekseni arasındaki açı değeri -45° ve 45° arasındaysa sınır alt sınır,
- Eğer ardışık noktalar arası vektör ile x ekseni arasındaki açı değeri 135° ve 225° arasındaysa sınır üst sınır,
- Eğer ardışık noktalar arası vektör ile z ekseni arasındaki açı değeri -45° ve 45° arasındaysa sınır sağ sınır,
- Eğer ardışık noktalar arası vektör ile z ekseni arasındaki açı değeri 135° ve 225° arasındaysa sınır sol sınır olarak sınıflandırılmaktadır.

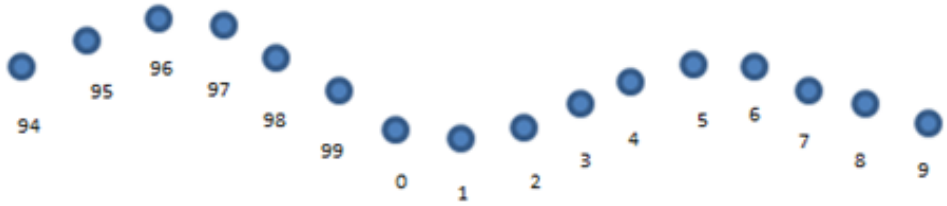
Çizelge 4.1, bu dört sınıflandırma türünün gösterimi için verilmiştir.

Çizelge 4.1: Lazer Algılayıcının Bakış Eksenine (s) Göre Sınırların Sınıflandırılması

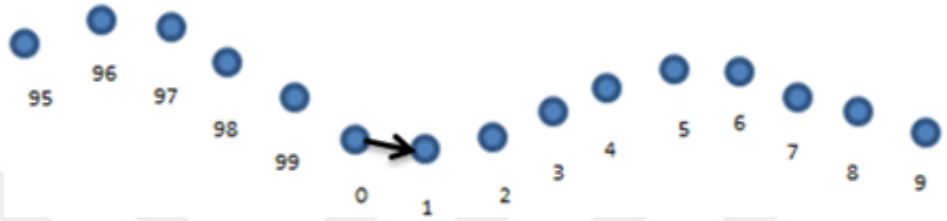
Açı	$s = x$ ekseni	$s = z$ ekseni
-45° ve 45° arası	alt sınır	sağ sınır
135° ve 225° arası	üst sınır	sol sınır

Sınırların sınıflandırılmasında uygulanan algoritma şu şekilde özetlenebilir. Önceki aşamada elde edilmiş olan en uzun sınır vektör zincirindeki noktaların ilk indekse sahip noktasından başlayıp indeks sayısını arttırarak, yani; ileri yönde giderek elde edilen vektör zinciri ile sınır yönünün alt, üst, sağ ya da sol sınıflarından hangisine ait olduğu kontrol edilebilmektedir (İndeksler, saat yönünün tersinde artıyor kabul edilmektedir). Fakat sadece ileri yöndeki vektör zincirini kullanarak yapılan bir sınıflandırma, aynı yönde ve birleşik olan sınırların ayrımış gibi elde edilmesine sebep olacağı için, geri yönde de vektör zincirinin devam ettirilmesi gerekmektedir. Böylece sınırın yönü kontrol edilirken, hem ileri yöndeki hem de geri yöndeki noktaları kullanarak bir vektör zinciri oluşturmak daha doğru sonuçların elde edilmesini sağlamaktadır. İleri ve geri yönde oluşturulan vektör zincirinin daha iyi anlaşılması için Şekil 4.12'de mavi ile gösterilen 100 noktalı bir sınırın alt parçası ve altlarında yazan indisler ile durum özetlenebilmektedir.

İlk vektör şekil 4.13'te gösterildiği gibi ilk indeks ve sonraki indeks arasında (yani, 0. ve 1. indeksler arasında) oluşturulur.

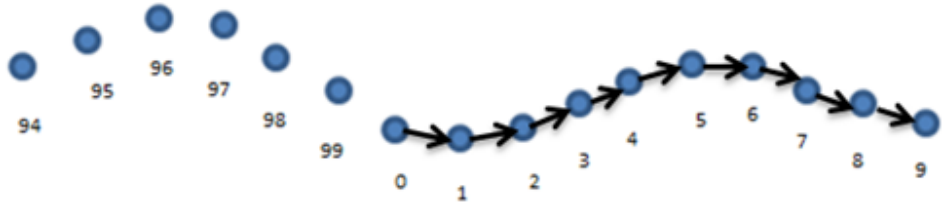


Şekil 4.12. 100 Noktalı Bir Sınırın Alt Kısmı ve Noktaların İndisleri .



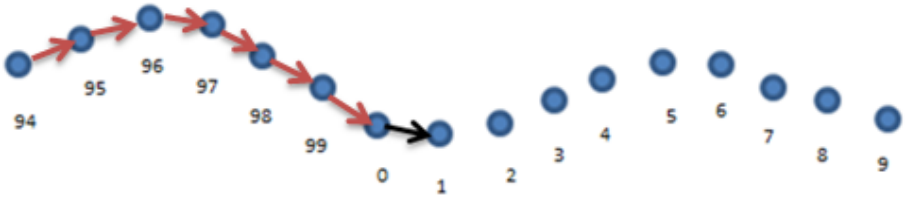
Şekil 4.13. İlk Sınır Zincir Vektörünün Oluşturulması .

İleri yönde vektör zinciri oluştururken indeks sayısı arttırılmakta ve belirli şartların (açı ve ceza parametresi) sağlanıp sağlanmadığı kontrol edilerek vektör zincirine yeni noktalar eklenmektedir. Şekil 4.14'te birazdan bahsedilecek olan açı şartlarının sağlandığı ve ceza parametresinin aşılmadığı kabul edilerek ileri yönde oluşturulan vektör zinciri siyah oklar ile gösterilmiştir.



Şekil 4.14. İleri Yönde Vektör Zincirinin Oluşturulması .

Geri yönde vektör zinciri oluştururken indeks sayısı azaltılır, fakat negatif olarak kullanılmaz. İlk indeksten önceki indeks sınır noktasının boyutuna eşittir. Şekil 4.15'te birazdan bahsedilecek olan açı şartlarının sağlandığı ve ceza parametresinin aşılmadığı kabul edilerek geri yönde oluşturulan vektör zinciri kırmızı ok ile gösterilmiştir.



Şekil 4.15. Geri Yönde Vektör Zincirinin Oluşturulması .

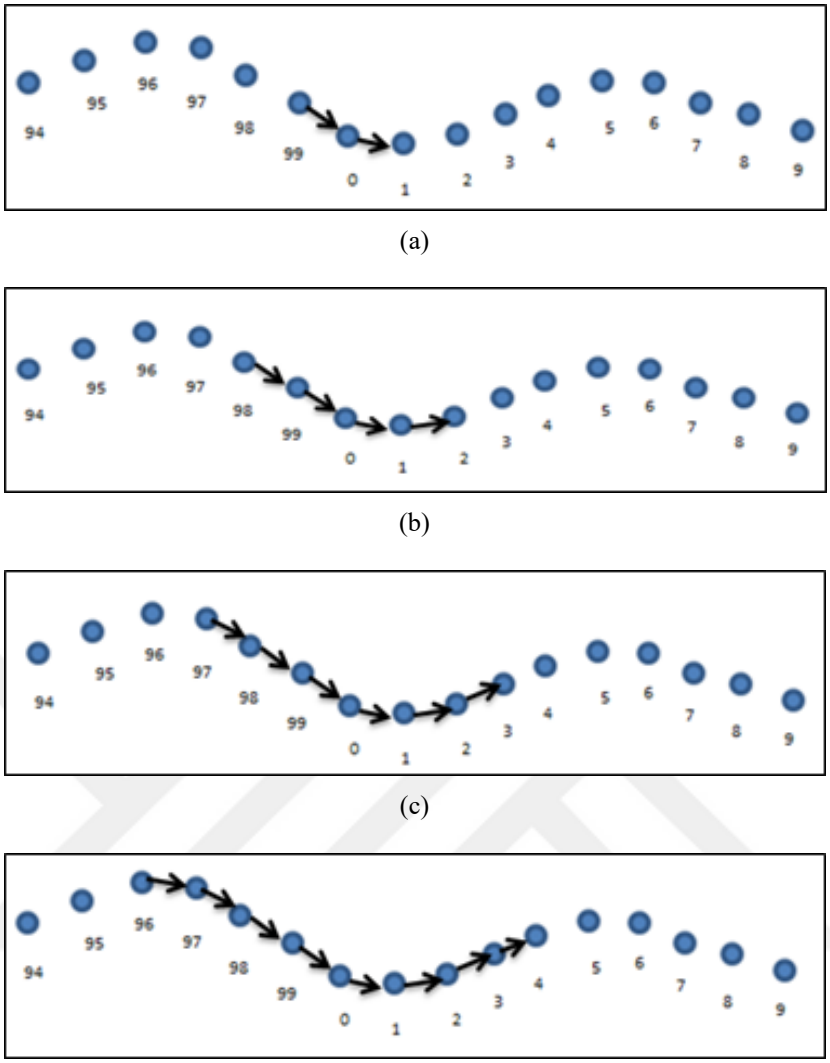
Tekrardan, ileri ve geri yöndeki vektörler için açı şartlarının sağlandığı ve ceza parametresinin aşılmadığı kabul edilip aşamalı olarak bu vektörlerin oluşturması Şekil 4.16'da gösterilmiştir.

İleri ve geri yöndeki noktaların kullanılmasıyla bir vektör zinciri elde edilirken, sınırın alt, üst, sağ ve sol sınıflarından hangisine ait olduğunu belirlemek için sırasıyla lazer profil algılayıcıya ait x ve z koordinat ekseni ile vektör arasındaki açı değeri kontrol edilmektedir. Bu açı aynı zamanda sınırın sonlanıp sonlanmadığını kontrol etmek için de kullanılmaktadır. Açı, iç çarpımın tersi kullanılarak, s algılayıcının bakış yönü ve v ardışık noktaların oluşturduğu vektörü temsil etmek üzere Denklem 4.1'deki gibi hesaplanmaktadır:

$$\alpha = \arccos\left(\frac{s.v}{|s|.|v|}\right) \tag{4.1}$$

Denklem 4.1 kullanılarak hesaplanan x ekseni ile vektör arasındaki açı değeri -45° ve 45° arasında ise sınır alt sınır, 135° ve 225° arasında ise üst sınır olarak sınır sınıflandırılır. Benzer şekilde, z ekseni ile vektör arasındaki açı -45° ve 45° arasında ise sınır sağ sınır, 135° ve 225° arasında ise sol sınır olarak sınır sınıflandırılır. Görüldüğü üzere, sınıflandırmada açı değerleri için 45° ve katları olan -45°, 135°, 225° açılar kullanılmıştır. Bunun sebebi, sınırların yanlışlıkla ait olmadığı sınıfa girmesini önlemektir, çünkü 45° eksenleri tam ortalayarak bölen açı değeridir ve bu şekilde yanlış sınıflandırmanın önüne geçmektedir.

Sınıflandırma aşamasında yeni nokta ekleme işlemi belirli sayıda ve arka arkaya açı değerini sağlamayan nokta geldiğinde sonlandırılmaktadır. Aslında bu işlem, sınırların son noktalarının bulunması ya da başka bir deyişle sınırın baştan ve sondan sınırlandırılması işlemidir ve ceza sayısı denilen parametreyle kontrol edilmektedir. Eğer açı değeri, belli bir eşik değerinden büyükse ceza parametresi arttırılmaktadır. Sonraki aşamada, uygun bir açı değeri gelirse bu parametre sıfırlanmaktadır. Eğer ceza parametresi belli bir eşik değerini



Şekil 4.16: Aşamalı Olarak İleri Ve Geri Yönde Vektör Eklenmesi: (a) İlk Aşama (b) İkinci Aşama (c) Üçüncü Aşama (d) Dördüncü Aşama

aşarsa, devam eden noktalara bakmaya gerek yok anlamına gelmektedir. Bu işlemler ileri ve geri yönde oluşturulan vektörler için ayrı ayrı yapılmaktadır. Böylece, sınırın ilk ve son noktaları bulunarak sınır sınırlandırılmış ve sınıflandırılmış olur. Ceza parametresi hem ileri yöndeki vektör zinciri hem de geri yöndeki vektör zinciri için 3 olarak kullanılmaktadır. Bu işlem, nokta bulutundaki tüm noktalar ziyaret edilene kadar devam eder ve tüm noktalar ziyaret edilip bir sınıfa atandıktan sonra algoritma sonlanmaktadır.

Algoritma sonlandıktan sonra bahsedilen alt, üst, sağ ve sol sınıflara ait birçok sınır elde edilir. Sınırlar için önemli olan nokta, sınırların uzunluğudur, çünkü sınırın uzunluğu çok kısa ise yüzey eğiliminin hesaplanması zorlaşacaktır. Bu yüzden, sınırı sınıflandırırken sınıfı oluşturan nokta sayısı 10'dan az ise vektör zinciri sınıflandırmaya alınmaz, 10'dan fazla nokta

içeren vektör zincirleri için sınıflandırma işlemi gerçekleştirilir. Algoritma ile ilgili detaylar Algoritma 3'te verilmiştir.

Algorithm 3 Sınırların Sıralanması

ρ_{xf}: ileri yöndeki vektörlerin x ekseni için ceza parametresi
ρ_{zf}: ileri yöndeki vektörlerin z ekseni için ceza parametresi
ρ_{xb}: geri yöndeki vektörlerin x ekseni için ceza parametresi
ρ_{zb}: geri yöndeki vektörlerin z ekseni için ceza parametresi
ρ_{xt}: x ekseninde ceza parametresi için eşik değeri
ρ_{zt}: z ekseninde ceza parametresi için eşik değeri
α_x: x ekseni ile ardışık noktaların oluşturduğu vektör arasındaki açı
α_{xt}: x ekseni ile ardışık noktaların oluşturduğu vektör arasındaki açı için eşik değeri
α_z: z ekseni ile ardışık noktaların oluşturduğu vektör arasındaki açı
α_{zt}: z ekseni ile ardışık noktaların oluşturduğu vektör arasındaki açı için eşik değeri
for indeks 0'dan nokta bulutunun boyutuna kadar ilerle **do**
  if tüm noktalar ziyaret edilmediyse **then**
    if x açısı ileri yönlü vektör için -45° ve 45° veya 135° ve 225° arasında **then**
      for şimdiki noktanın indisinden nokta bulutunun boyutuna kadar ilerle **do**
        if α_x açısı ileri yönlü vektör için -45° ve 45° arasında ve $\rho_{xf} < \rho_{xt}$ **then**
          sınırı alt sınır olarak ekle
        else
          $\rho_{xf} = \rho_{xf} + 1$
        end if
        if α_x açısı geri yönlü vektör için -45° ve 45° arasında ve $\rho_{xb} < \rho_{xt}$ **then**
          sınırı alt sınır olarak ekle
        else
          $\rho_{xb} = \rho_{xb} + 1$
        end if
      end for
    else
      if α_x açısı ileri yönlü vektör için 135° ve 225° arasında ve $\rho_{xf} < \rho_{xt}$ **then**
        sınırı üst sınır olarak ekle
      else
        $\rho_{xf} = \rho_{xf} + 1$
      end if
      if α_x açısı geri yönlü vektör için 135° ve 225° arasında ve $\rho_{xb} < \rho_{xt}$ **then**
        sınırı üst sınır olarak ekle
      else

ρ_{xb}=ρ_{xb}+1
end if
if ρ_{xf}>3 ve ρ_{xb}>3 ise **then**
döngüyü sonlandır
end if
end if
else
for şimdiki noktanın indisinden nokta bulutunun boyutuna kadar ilerle **do**
if α_z açısı ileri yönlü vektör için -45° ve 45° veya 135° ve 225° arasında **then**
if α_z açısı ileri yönlü vektör için -45° ve 45° arasında ve ρ_{zf}< ρ_{zt} **then**
sınırı sağ sınır olarak ekle
else
ρ_{zf}=ρ_{zf}+1
end if
if α_z açısı geri yönlü vektör için -45° ve 45° arasında ve ρ_{zb}< ρ_{zt} **then**
sınırı sağ sınır olarak ekle
else
ρ_{zb}=ρ_{zb}+1
end if
else
if α_z açısı ileri yönlü vektör için 135° ve 225° arasında ve ρ_{zf}< ρ_{zt} **then**
sınırı sol sınır olarak ekle
else
ρ_{zf}=ρ_{zf}+1
end if
if α_z açısı geri yönlü vektör için 135° ve 225° arasında ve ρ_{zb}< ρ_{zt} **then**
sınırı sol sınır olarak ekle
else
ρ_{zb}=ρ_{zb}+1
end if
if ρ_{zf}>3 ve ρ_{zb}>3 **then**
döngüyü sonlandır
end if
end if
end for
end if
end for
return tüm sınıflandırılmış sınırlar

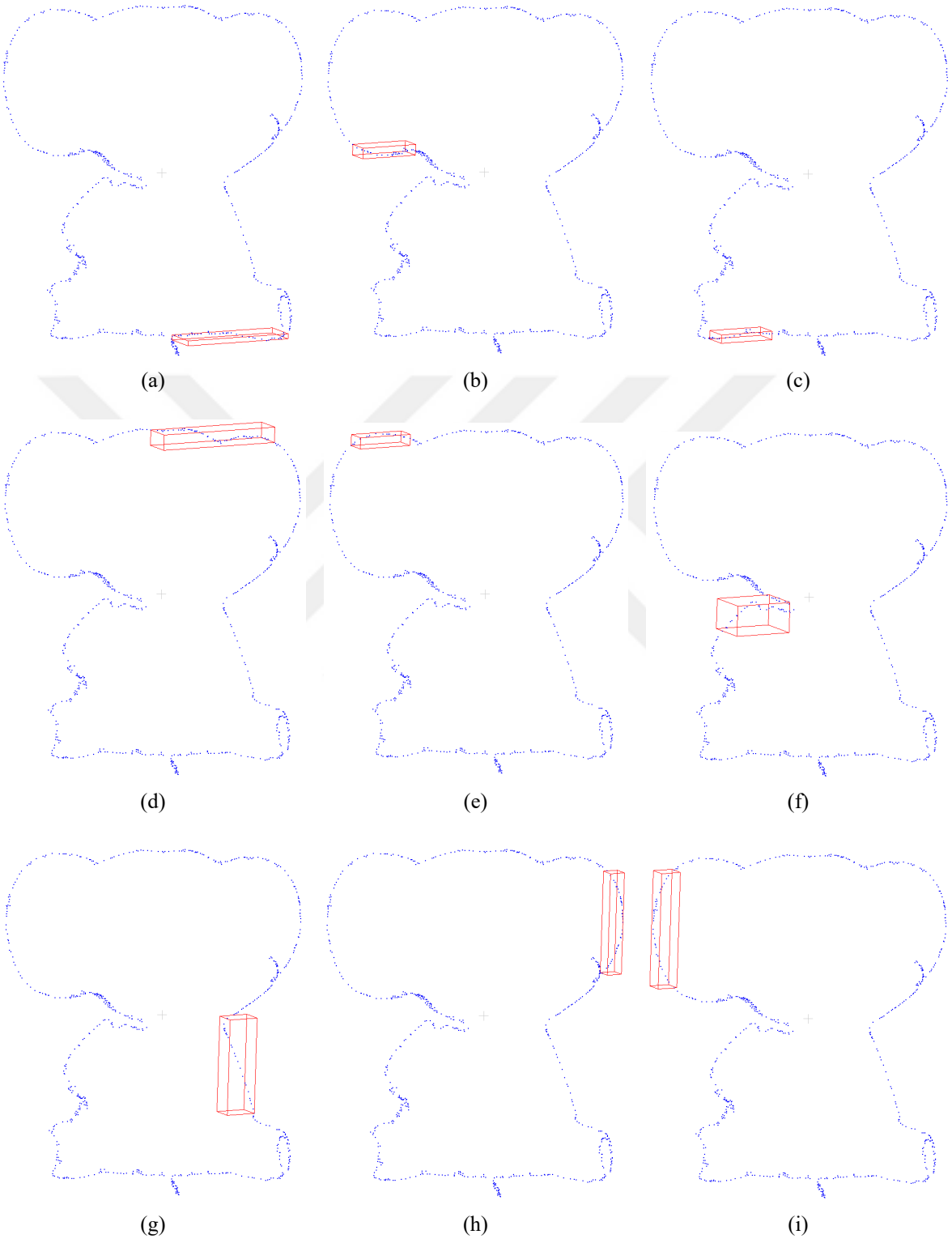
Yukarıda verilen algoritma ile; fil nesnesi, endüstriyel parça ve çanak nesnesi için filtrelenmiş nokta bulutlarından elde edilen sınırların sıralanmasıyla bulunan en uzun sınırlar sağ, sol, alt ve üst olarak sınıflandırılmıştır. Fil nesnesine, endüstriyel parçaya ve çanak nesnesine ait sınıflandırılmış nesne sınırları sırasıyla Şekil 4.17, Şekil 4.18 ve Şekil 4.19 ile gösterilmiştir. Sınırların gösterimi yapılırken mavi nokta bulutu ile gösterilmiş ilgili nesnenin en uzun sınırı üzerinde kırmızı kutular kullanılmıştır.

Özetle, bu üç nesne için herbirine ait sınıflandırılan sınırların ilgili nesne üzerinde gösterimi Şekil 4.20'deki gibidir. Bu şekiller üzerinde hiçbir sınıfa ait olmayan noktalar göze çarpmaktadır. Bu noktalar için, ön yüzden bakıldığında açı şartlarının sağlanıp ceza parametresinin aşılmaması gerektiği düşünülse de, yan yüzden bakıldığında noktalar arası keskin değişimler olduğu gözlenmektedir. Bu durum, noktaların oluşturduğu vektörler arası açının 45° yerine daha yüksek açı değerinin alınmasıyla aşılabilir. Fakat, bu durum sınırların yanlış sınıflandırılabilme ihtimalini doğurmakta ve yanlış sınıflanmış sınırlar yüzünden NBV adaylarının yanlış hesaplanmasının tüm süreci etkilemektedir. Bu yüzden, böyle bir riske girilmemek adına noktaların oluşturduğu vektörler arası açı değeri 45° olarak kullanılmaya devam edilmektedir.

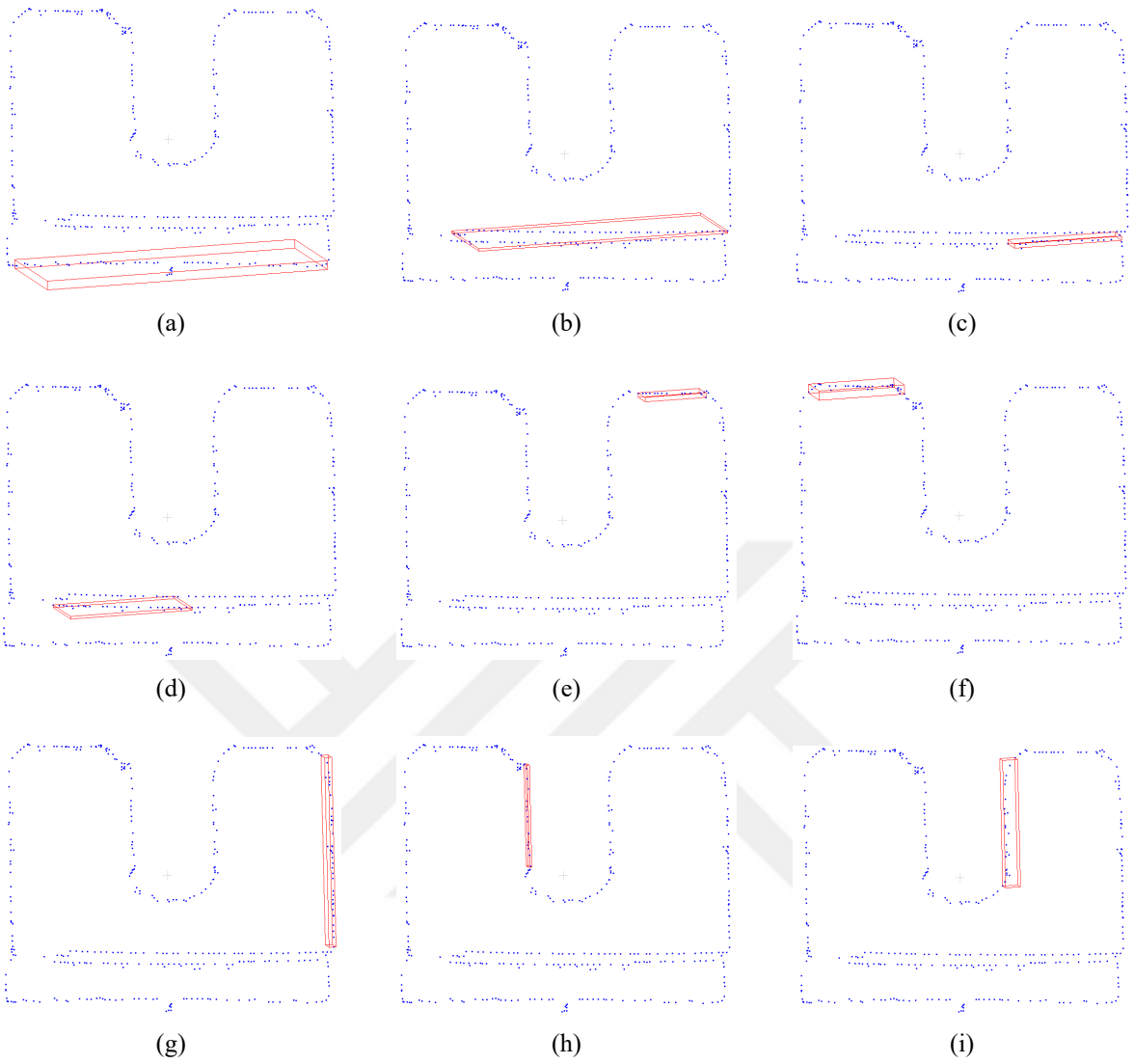
4.4 Yüzey Eğilimi Tahmini

Uygulaması yapılan makalenin en önemli noktalarından biri, bir sonraki tarama için algılayıcı pozisyonlarının belirlenmesinde yüzey eğilim tahmininin kullanılmasıdır. Yüzey eğilimi tahmini, nesnenin mevcut yüzey nokta bulutunu kullanarak yüzey eğilimi hesaplamak ve nokta bulutu bulunmayan yüzey şekli hakkında yaklaşık olarak tahminler yapabilmektir. Burada yüzeyler geometrik fonksiyonlar olarak temsil edilmektedir. Sınırların sınıflandırılmasıyla elde edilen alt, üst, sağ ve sol sınırlar bu aşamada kullanılmaktadır. Uygulaması yapılan makalede, yüzey eğiliminin basit nesnelerde 3B modelleme için kullanılabileceği, karmaşık nesneler için ise bunun mümkün olmadığı ve her sınır için ayrı ayrı yüzey eğilimin kullanılması gerektiğinden bahsedilmektedir. Tez kapsamında modellenecek nesneler de karmaşık olduğundan bütün sınırlar için yüzey eğilimi hesaplanmaktadır.

İlk olarak; alt, üst, sağ ve sol olarak sınıflandırılmış sınırları kapsayan yönlendirilmiş bir sınırlayıcı kutu (oriented bounding box) bulunmaktadır. Sınırlayıcı kutunun bir ayrıtı ile sınır vektör zincirinin ilk noktası, diğer bir ayrıtı ile sınır vektör zincirinin son noktası sınırlandırılırken, diğer iki ayrıtı ise belli bir oranda genişletilmektedir. Bu genişletilme yapılmadan önce elde edilen yönlendirilmiş sınırlayıcı kutular Şekil 4.20 ile aynı olduğu için burada tekrar görselleştirilmemiştir. Sınırlayıcı kutu



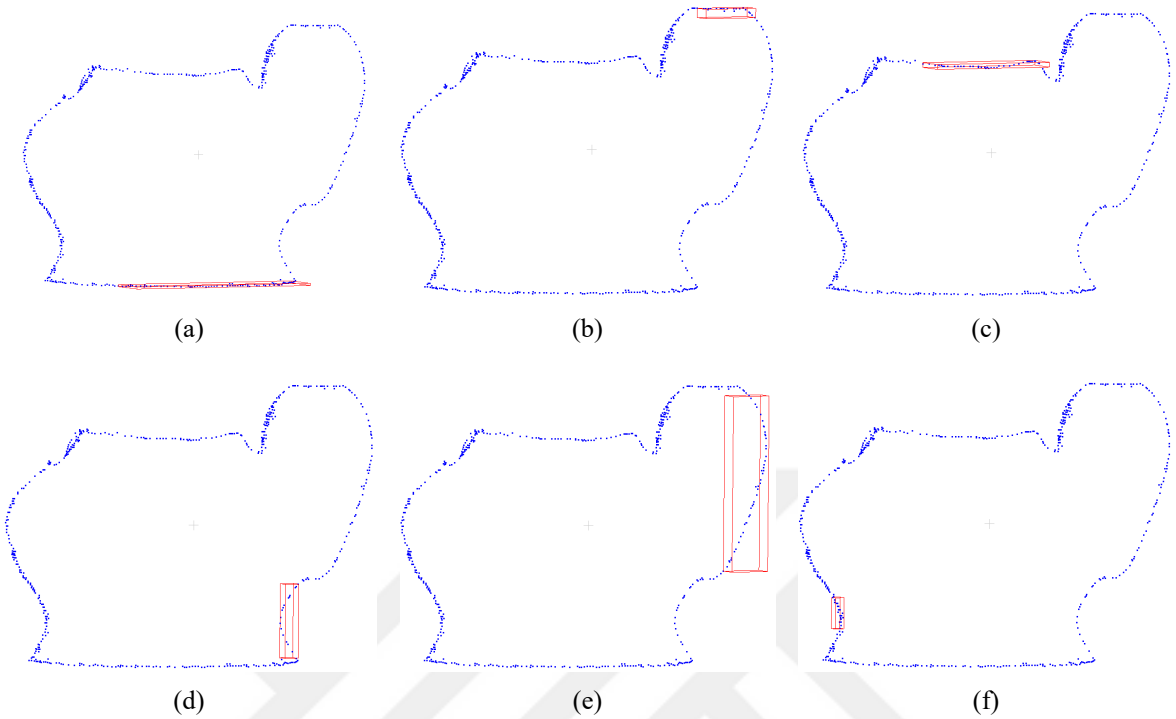
Şekil 4.17: Fil Nesnesi İçin Sınıflandırılmış Sınırlar: Alt Sınırlar (a, b, c), Üst Sınırlar (d, e, f), Sağ Sınırlar (g, h) ve Sol Sınır (i)



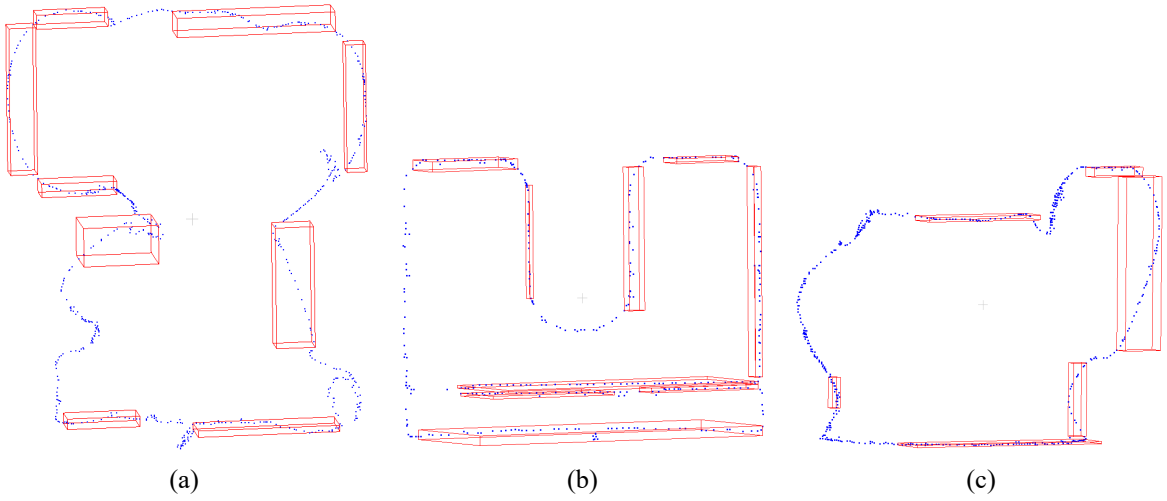
Şekil 4.18: Endüstriyel Parça İçin Sınıflandırılmış Sınırlar: Alt Sınırlar (a, b), Üst Sınırlar (c, d, e, f), Sağ Sınırlar (g, h) ve Sol Sınır (i)

genişletildikten sonra sınırlayıcı kutunun içine düşen noktalar yeşil renk ile, sınırlar ise kırmızı renk ile igili nokta bulutu üzerinde, fil nesnesi ve endüstriyel parçada sağ sınırlar ve çanak nesnesinde üst sınırlar için Şekil 4.21'deki gibi görselleştirilmiştir.

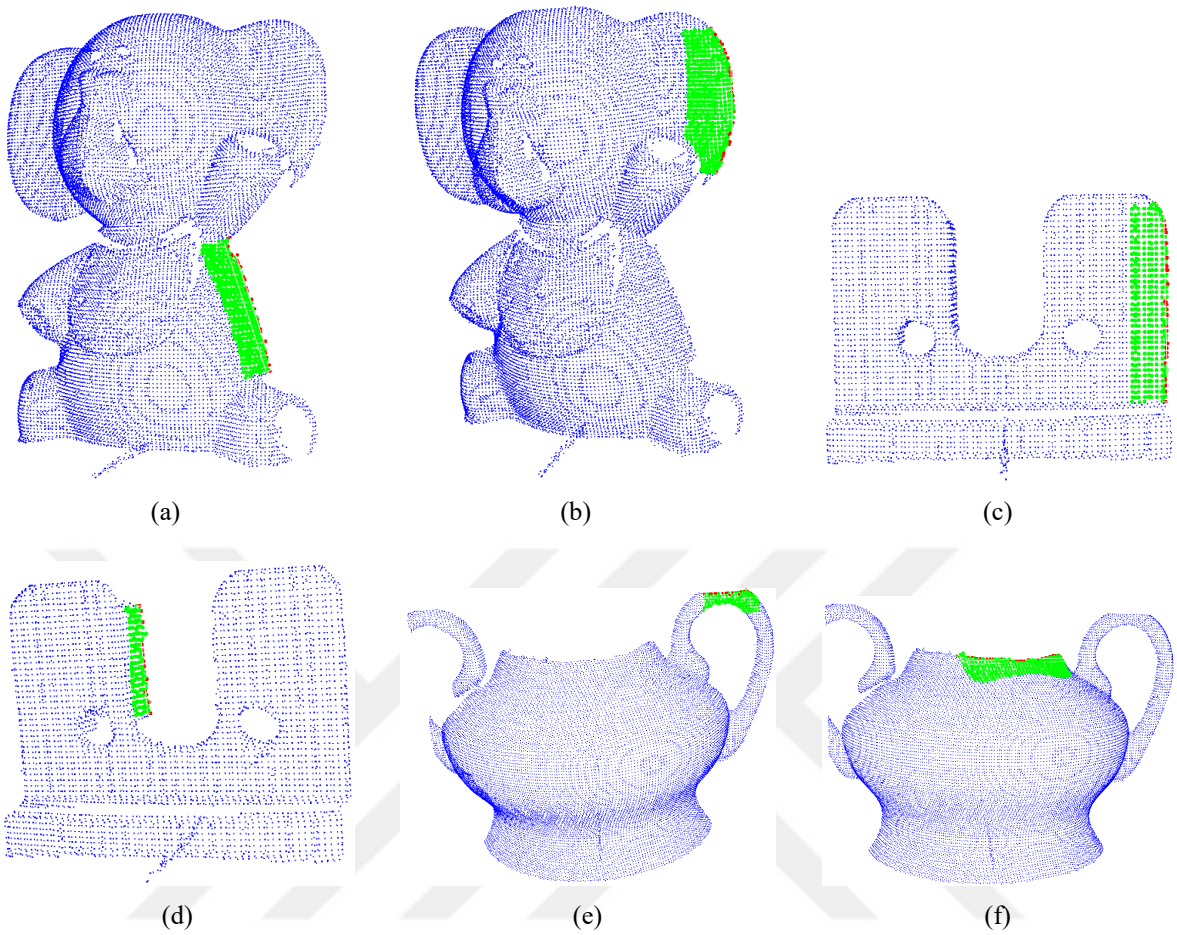
Daha sonra, elde edilen genişletilmiş ve yönlendirilmiş bu sınırlayıcı kutunun içine düşen noktalar bulunmaktadır. Bu noktalar için bölge genişletme (region growing) algoritması uygulanmaktadır. Aslında, segmentasyonun bir parçası olan bölge genişletme algoritmasının kullanım amacı yüzey eğilimi hesabında ikinci dereceden bir eğri oluşturmak için girdi olarak sınırın karakteristiğini yansıtan yakın noktalar kümesini elde etmektir.



Şekil 4.19: Çanak Nesnesi İçin Sınıflandırılmış Sınırlar: Alt Sınır (a), Üst Sınırlar (b, c), Sağ Sınırlar (d, e) ve Sol Sınırlar (f)



Şekil 4.20: Üç Nesne İçin Sınıflandırılan Sınırlar: (a) Fil Nesnesi İçin (b) Endüstriyel Nesne İçin (c) Çanak Nesnesi İçin



Şekil 4.21: Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Sınırlayıcı Kutu İçine Düşen Noktalar ve İlgili Sınırlar: Fil Nesnesinin Sağ Sınırları ve Sınırlayıcı Kutu İçine Düşen Noktalar (a, b), Endüstriyel Parçanın Sağ Sınırları ve Sınırlayıcı Kutu İçine Düşen Noktalar (c, d), Çanak Nesnesinin Üst Sınırları ve Sınırlayıcı Kutu İçine Düşen Noktalar (e, f)

Bölge genişletme algoritmasının amacı, düzlük kısıtı bakımından birbirine yakın noktaların birleştirilmesini sağlamaktır. Bu yüzden bu algoritmanın çıktısı, aynı yüzeyde olduğu düşünülen noktaların kümesidir. Algoritma, noktaların normalleri arasındaki açıların karşılaştırılmasına dayanmaktadır. İlk olarak, tüm noktalar eğrilik değerlerine göre sıralanmaktadır; çünkü minimum eğrilik değerine sahip olan noktadan bölge genişletilmeye başlatılmaktadır. Bunun sebebi ise, minimum eğriliğe sahip olan noktanın düz bir bölgede olmasıdır. Bu sayede, en düz bölgeden başlatılan bölge genişletme algoritması toplam küme sayısını azaltmaya imkan tanımaktadır (Anonim, 2017c).

Minimum eğrilik değerine sahip olan noktadan bölge genişletmeye başlanarak şu aşamalar sırasıyla gerçekleştirilmektedir:

1. Bölge genişletme yapılan noktaya tohum nokta adı verilmek üzere seçilen nokta tohum kümesine eklenmektedir.
2. Her tohum nokta için algoritma komşu noktaları bulmaktadır.
    2.1. Her komşu noktanın normali ve tohum noktanın normali arasındaki açı değeri hesaplanır. Eğer bu açı değeri belli bir eşik değerin altında ise şimdiki bölgeye eklenir.
    2.2. Her komşu noktanın eğrilik değeri test edilir. Eğer, eğrilik değeri belli bir eşik değerinin altında ise bu nokta tohum nokta olarak eklenir.
    2.3. Şimdiki tohum nokta, tohum noktaları kümesinden kaldırılır.
3. Tohumlar kümesi boş olduğunda, bölgenin genişletilmiş olup bu işlemler başka bir bölge için baştan başlatılmaktadır.

Yukarıda maddeler halinde verilen aşamalar Algoritma 4 ile detaylı olarak verilmiştir.

Burada bölge genişletme algoritmasının kullanım amacı, girdileri ve çıktıları yukarıda bahsedilenden biraz daha farklıdır. Normalde bölge genişletme ile amaçlanan bölütleme olduğu için tüm nokta bulutu girdi olarak verilmiş ve çıktı olarak tüm nokta bulutundaki bölütlenmiş nokta kümeleri elde edilmiştir. Bu çalışma kapsamında ise, girdi olarak yönlendirilmiş sınırlayıcı kutu içine düşen noktalar verilmiş ve tek bir noktadan bir kez yapılan bölge genişletmeyle sınırın karakteristiğini yansıtan yakın noktalar kümesi çıktı olarak elde edilmiştir.

Sınırın karakteristiğini yansıtan yakın noktalar kümesini elde etmek amacıyla sınırın merkezinden, yani orta noktasından başlayan bölge genişletme, yönlendirilmiş sınırlayıcı kutunun sınırları içinde bulunan, düzlük kısıtını sağlayan ve birbirine yeterince yakın olan komşu noktaların eklenmesiyle yinelemeli olarak devam eder. Sınırdaki nokta sayısının tek ya da çift olmasına göre Denklem 4.2'deki eşitlikler kullanılarak sınırın merkezi bulunmaktadır. Bölge genişletmede; minimum küme sayısı, maksimum küme sayısı, komşu sayısı, düzlük eşik değeri ve eğrilik eşik değeri olarak adlandırılan beş parametre bulunmaktadır. Bu parametrelerden minimum küme sayısı ve maksimum küme sayısı daha çok bölütleme ile alakalı olup, bölge genişletme kısmının performansını en çok düzlük eşik değeri ve eğrilik eşik değeri parametreleri etkilemektedir. Bu iki parametrenin uygun bir şekilde ayarlanması, bölge genişletme algoritmasının çıktısı olan aynı yüzeyde olduğu düşünülen noktaların kümesinin doğru olarak elde edilmesini sağlamaktadır. Bölge genişletme algoritması uygulandıktan sonra elde edilmiş olan noktalar, fil nesnesi ve

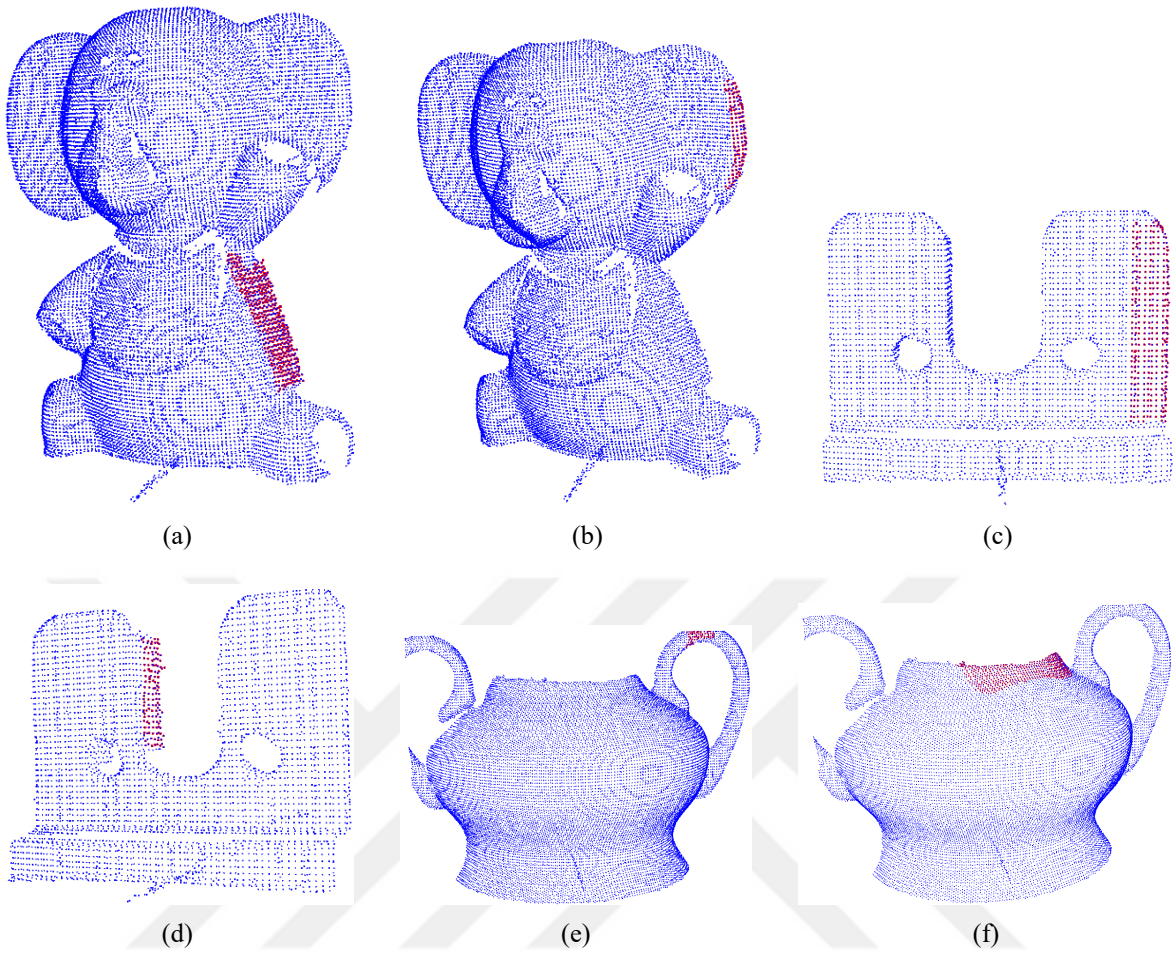
Algoritma 4 Bölge Genişletme Algoritması

while bölge genişletmeye uygun noktalar kümesi boş değilken **do**
şimdiki bölgeyi sıfırla
şimdiki tohum noktasını sıfırla
nokta bulutunda minimum eğriliğe sahip noktayı bölge genişletmeye uygun noktalar kümesine ekle
şimdiki tohum noktasına minimum eğriliğe sahip noktayı ekle
şimdiki bölgeye minimum eğriliğe sahip noktayı ekle
bölge genişletmeye uygun noktalar kümesinden minimum eğriliğe sahip noktayı çıkar
for ilk değişken 0'dan şimdiki tohum noktasının boyutuna kadar ilerlesin **do**
tohum noktasının en yakın komşularını bul
for ikinci değişken 0'dan en yakın komşuların boyutuna kadar ilerlesin **do**
şimdiki komşu noktayı belirle
if bölge genişletmeye uygun noktalar kümesi şimdiki komşu noktayı içeriyor ve tohum nokta ve komşu noktanın normalleri arasındaki açı eşik değerden küçük **then**
komşu noktayı şimdiki bölgeye ekle
bölge genişletmeye uygun noktalar kümesinden komşu noktayı çıkar
if komşu noktanın eğimi eşik değerden küçük **then**
komşu noktayı tohum noktalar kümesine ekle
end if
end if
end for
end for
şimdiki bölgeyi global bölgeler listesine ekle
end while
return nokta bulutu üzerindeki genişletilmiş bölgelerin kümesi

endüstriyel parçada sağ sınırlar ve çanak nesnesinde üst sınırlar için kırmızı renk ile Şekil 4.22'de ilgili nokta bulutu üzerinde görselleştirilmiştir.

$$M_{vektorzinciri} = \begin{cases} (n_{sinir}+1)/2, & \text{sınırdaki nokta sayısı tek ise} \\ n_{sinir}/2, & \text{sınırdaki nokta sayısı çift ise} \end{cases} \tag{4.2}$$

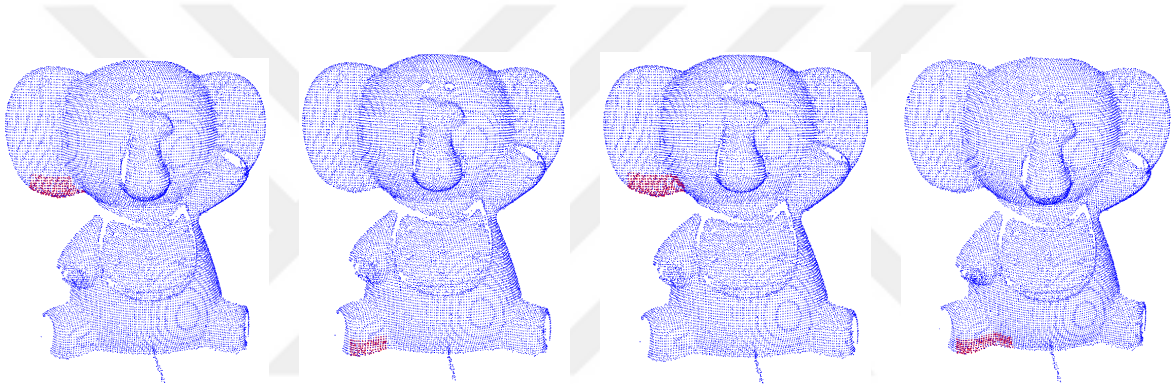
Bölge genişletmede düzlük eşik değerinin önemi fil nesnesinin alt iki sınırı ile Şekil 4.23 üzerinde gösterildiği gibi açıklanabilir. Bu şekil üzerinde, aynı sınırlar için diğer tüm parametreler sabit tutularak düzlük eşik değerleri uygun olan ve olmayan değerler ile bölge genişletme uygulanmıştır. Uygun olan değerler kullanılarak yapılan bölge genişletme, Şekil



Şekil 4.22: Fil, Endüstriyel Parça ve Çanak Nesneleri İçin Bölge Genişletme Algoritmasıyla Elde Edilen Noktalar: Fil Nesnesinin Sağ Sınırları İçin (a, b), Endüstriyel Parçanın Sağ Sınırları İçin (c, d), Çanak Nesnesinin Üst Sınırları İçin (e, f)

4.23a ve Şekil 4.23b'de görüldüğü üzere, farklı yüzeyleri birbirinden ayırmıştır. Aynı sınırlar için uygun olmayan parametre değeri ile bölge genişletme uygulanmış, Şekil 4.23c ve Şekil 4.23d'de görüldüğü üzere farklı yüzeyler aynı yüzeyin parçasıymış gibi elde edilmiştir. Bölge genişletmenin amacı farklı özelliklere sahip farklı yüzeyleri ayırmak olduğundan, bu tarz parametre seçimleri dikkatle yapılmalıdır. Diğer parametreler için de bu şekilde, farklı değerler için denemeler yapılmış olup, bölge genişletmenin en önemli parametresi düzlük eşik değeri olarak saptanmıştır. Burada amaç bölütleme olmadığı için, minimum küme sayısı ve maksimum küme sayısı gibi parametreler sadece işlem hızının yavaş ya da hızlı olmasında etkilidir. Her ne kadar bu parametreler bölge genişletmeyi etkilemese de çalışma gerçek zamanlı olacağı için hesaba katılmaları gerekmektedir. Bölge genişletmede düzlük eşik değerinin önemi fil nesnesinin alt iki sınırı ile Şekil 4.23 üzerinde gösterildiği gibi açıklanabilir. Bu şekil üzerinde, aynı sınırlar için diğer tüm parametreler sabit tutularak düzlük eşik değerleri uygun olan ve olmayan değerler ile bölge genişletme

uygulanmıştır. Uygun olan değerler kullanılarak yapılan bölge genişletme, Şekil 4.23a ve Şekil 4.23b'de görüldüğü üzere, farklı yüzeyleri birbirinden ayırmıştır. Aynı sınırlar için uygun olmayan parametre değeri ile bölge genişletme uygulanmış, Şekil 4.23c ve Şekil 4.23d'de görüldüğü üzere farklı yüzeyler aynı yüzeyin parçasıymış gibi elde edilmiştir. Bölge genişletmenin amacı farklı özelliklere sahip farklı yüzeyleri ayırmak olduğundan, bu tarz parametre seçimleri dikkatle yapılmalıdır. Diğer parametreler için de bu şekilde, farklı değerler için denemeler yapılmış olup, bölge genişletmenin en önemli parametresi düzlük eşik değeri olarak saptanmıştır. Burada amaç bölütleme olmadığı için, minimum küme sayısı ve maksimum küme sayısı gibi parametreler sadece işlem hızının yavaş ya da hızlı olmasında etkilidir. Her ne kadar bu parametreler bölge genişletmeyi etkilemese de işlem süresini kısaltmak adına hesaba katılmaları gerekmektedir.



(a) (b) (c) (d)

Şekil 4.23: Bölge Genişletme Algoritmasında Düzlük Parametresinin Öneminin Fil Nesnesi Üzerinde Açıklanması: Fil Nesnesinin Alt Sınırlarının Uygun Parametre Değeri İle Sonuçları (a, b), Fil Nesnesinin Alt Sınırlarının Uygun Olmayan Parametre Değeri İle Sonuçları (c, d)

Yüzey eğilimi hesaplanırken, sadece sınırdaki noktaların kullanılması yerine genişletilmiş bölge içerisindeki noktalar kullanılmaktadır. Bölge genişletme algoritması ile noktalar fazla genişletilmemelidir, çünkü bu şekilde yapılan bir hesaplama doğru bir yüzey eğilimi hesabına engel olmaktadır. Sınır bölgesindeki noktaları, ikinci dereceden bir denkleme yakınsamak için, algılayıcı bakış doğrultusu y kabul edilerek Denklem 4.3'teki gibi uyarlanmıştır.

$$y = f(x_{vi}, z_{vi}) = a_1 x_{pi}^2 + a_2 x_{pi} z_{pi} + a_3 z_{pi}^2 + a_4 x_{pi} + a_5 z_{pi} + a_6 \tag{4.3}$$

İkinci dereceden bir eğri kullanılmasının nedeni düşük dereceli olması, dolayısıyla hızlı olması ve bilinmeyen alanın yönünün içeri veya dışarı olması fark etmeksizin çok fazla

değişikliğe gerek kalmadan iyi sonuçlar vermesidir. Denklem 4.3'teki a_1, a_2, a_3, a_4, a_5 ve a_6 katsayılarını bulmak için öncelikle, Denklem 4.4'te verilen X matrisi sınır bölgesindeki noktaların *x* ve *z* koordinatları kullanılarak doldurulmaktadır.

$$X = \begin{bmatrix} x_{p1}^2 & x_{p1}z_{p1} & z_{p1}^2 & x_{p1} & z_{p1} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{pn}^2 & x_{pn}z_{pn} & z_{pn}^2 & x_{pn} & z_{pn} & 1 \end{bmatrix} \tag{4.4}$$

Bakış açılarını içeren *y* vektörü ise Denklem 4.5'teki gibi doldurulmaktadır.

$$y = \begin{bmatrix} y_{p1} & y_{p2} & y_{p3} & \dots y_{pn} \end{bmatrix}^T \tag{4.5}$$

Bilinmeyen katsayılarından oluşan A matrisi ise Denklem 4.6'daki gibidir.

$$A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \end{bmatrix}^T \tag{4.6}$$

Çözüme ulaştıran denklem ise Denklem 4.7'de verilmiştir.

$$y = XA \tag{4.7}$$

Golub-Reinsch tekil değer ayrışımı yöntemi kullanarak, A matrisi yaklaşık olarak Denklem 4.8'deki gibi yakınsanabilir.

$$\hat{A} = (X^T W X)^{-1} X^T W y \tag{4.8}$$

Denklem 4.8'deki W matrisi yapılan *n* tane gözlemin ağırlıklarını temsil etmektedir ve bu matris Denklem 4.9 ile gösterilen şekilde doldurulmaktadır. Bu matrisin amacı, sınıra uzak noktaların sınırın özelliklerini yansıtmayacağını düşünerek, sınıra yakın noktalara daha

çok ağırlık verilmesini, diğer noktalara ise uzaklıklarıyle ters orantılı olarak daha az ağırlık verilmesini sağlamaktır.

$$w_i = \begin{cases} 1, & V_i \text{ sınırda ise} \\ \|V_i - V_{sinir}\|, & \text{diğer} \end{cases} \tag{4.9}$$

Denklem 4.9'da V_i sınır olup olmadığı kontrol edilen noktayı temsil ederken, V_{sinir} ise sınır olup olmadığı kontrol edilen noktaya en yakın sınır noktasını temsil etmektedir. V_i vektöründeki her bir nokta için o noktaya en yakın olan sınır noktasının uzaklığı kd-ağacı algoritması kullanılarak bulunmaktadır. Uzaklık 0 ise, kontrol edilen nokta sınır noktasıdır ve diyagonal olarak elde edilecek olan W matrisinde 1 ile doldurulmaktadır. Eğer sınırda olup olmadığı kontrol edilen nokta sınır noktası üzerinde değilse, yani sınırda olup olmadığı kontrol edilen nokta ve ona en yakın olan sınır noktasının uzaklığı 0'dan farklı ise diyagonal olan W matrisi uzaklık kadar fark ile doldurulmaktadır. Bu noktalar arası uzaklık farkı, değişim metre cinsinden olduğu için 1'den çok küçük değerler çıkmaktadır. Eğer birim metre yerine milimetre cinsinden ise 1000 ile bölünerek normalize edilebilir. Normalize işlemi yapılmadığı takdirde sınırdaki noktaların etkisi, daha içerideki noktaların etkisinden az çıkacak ve yüzey eğilimi istenilen şekilde elde edilmeyecektir. Noktaların birimine göre edilen w matrisinin doldurulmasından sonra Denklem 4.10'daki gibi diyagonal bir matris elde edilmektedir.

$$W = \begin{bmatrix} w_{11} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & w_{nn} \end{bmatrix} \tag{4.10}$$

Sınırdaki noktalar için ağırlıklandırma kullanılmak istenmediği takdirde W matrisinin birim matris olarak kullanılmasıyla ya da hiç kullanılmamasıyla A matrisi elde edilebilir. Bu çalışma için, ağırlıklandırma kullanılmadan Denklem 4.7'de A matrisi çekilerek yüzey eğilim eğrisinin parametreleri bulunmuştur. Fil nesnesinin sağ, sol, yukarı ve aşağı sınırlarından birer tanesi; endüstriyel nesnenin üst sınırlarından iki tanesi ve çanak nesnesinin sağ sınırlarından iki tanesi için yüzey eğilim hesabı Şekil 4.24 ile gösterildiği gibi farklı açılardan mavi nokta bulutu üzerinden kırmızı noktalar ile görselleştirilmiştir.

Şekil 4.24'te Şekil 4.24a, Şekil 4.24b, Şekil 4.24c ve Şekil 4.24d fil nesnesine ait sırasıyla üst, alt, sol ve sağ sınırlarına ait yüzey eğilim tahminlerini; Şekil 4.24e, Şekil 4.24f, Şekil 4.24g ve Şekil 4.24h aynı sıra ile farklı açılardan görselleştirilmiş yüzey eğilim

tahminlerini göstermektedir. Benzer şekilde, endüstriyel parça ve çanak nesnesine ait sırasıyla üst ve sağ sınırlar Şekil 4.24i, Şekil 4.24j, Şekil 4.24k ve Şekil 4.24l ile görselleştirilirken, aynı sıra ile farklı açıdan görseller 4.24m, Şekil 4.24n, Şekil 4.24o ve Şekil 4.24p ile verilmiştir.

Kağıt, CD ve levha gibi bazı nesneler keskin değişimlere sahiptir. Modellenen nesnelerde keskin değişimlerin olması durumunda tahmin edilen eğri nesnenin yaklaşık şeklini belli etmemektedir. Bu duruma örnek olarak bu çalışma kapsamında 3B modellenen endüstriyel nesnenin üst sınırları verilebilir (Şekil 4.24i, Şekil 4.24j ve Şekil 4.24l).Bu durumun benzeri, Şekil 4.24l ile gösterilen çanağın sağ sınırlarının birinde de mevcuttur. Yumuşak geçişlerin olduğu nesnelerde ise yüzey tahmini etkili ve verimli olarak çalışmaktadır. Bu duruma örnek olarak Şekil 4.24k verilebilir. Yani özetle; 3B modellenen nesnedeki değişimlerin yumuşak olması, nesne sınırlarının karakteristiğini yansıtan ikinci dereceden eğrilerin daha doğru tahmin edilmesini ve dolasıyla, hesaplanan NBV adaylarına ait poziyon ve yönelim bilgilerinin daha doğru elde edilmesini sağlamaktadır.

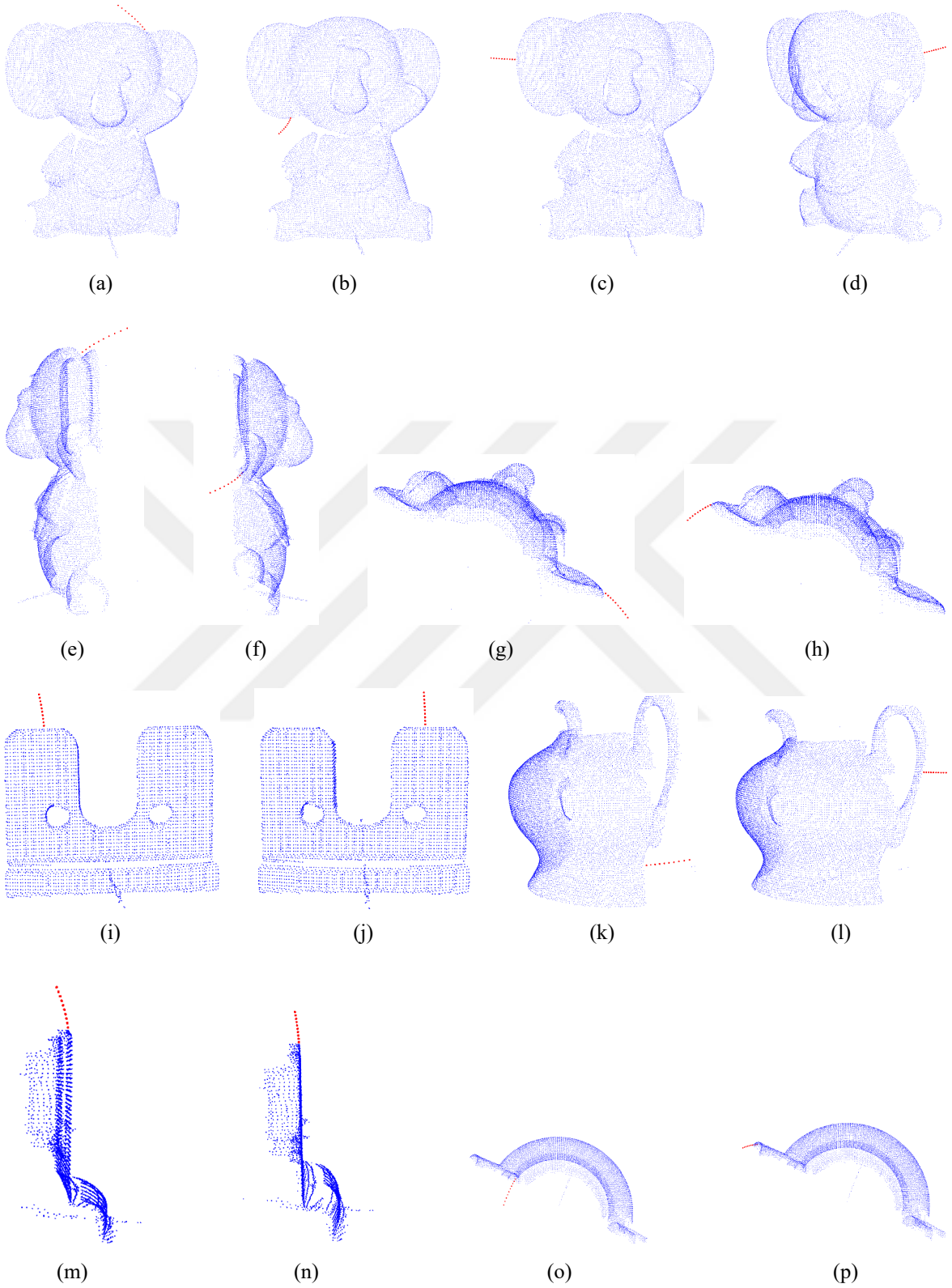
Yüzey eğiliminin hesabı ile tahmin edilen ikinci dereceden eğrilerin Denklem 4.3'te de bahsedilen a_1, a_2, a_3, a_4, a_5 ve a_6 katsayılarından oluşan A matrisi sonraki en iyi görüş adaylarının pozisyonunun ve yöneliminin belirlenmesinde kullanılmaktadır. Bundan sonraki aşama, bütün sınırlar için kalite kriterine bakmaksızın NBV adaylarının pozisyon ve yönelim hesabının yapılmasıdır. NBV adaylarından NBV seçimi ise daha sonraki aşamada yapılmaktadır.

4.5 NBV Adaylarının Hesaplanması

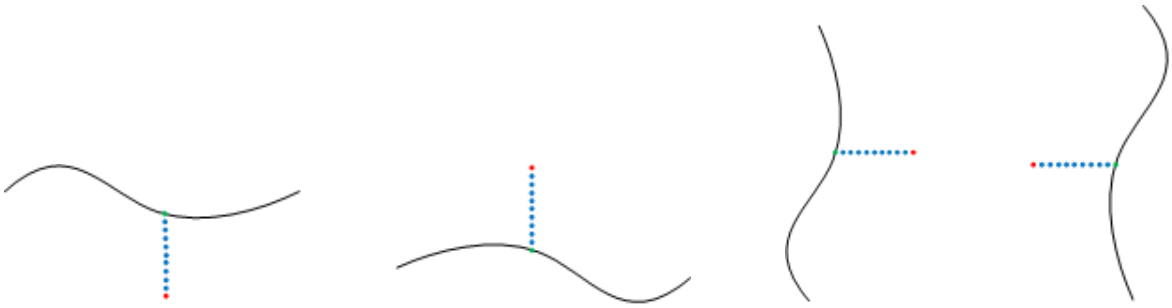
NBV adaylarının hesaplanması kapsamında NBV adaylarının pozisyon hesabının yapılması ve yönelim hesabının yapılması bulunmaktadır.

4.5.1 NBV adaylarının yönelim hesabı

Bu bölümde NBV adayları için yönelim hesabı yapılmaktadır. İlk olarak hangi sınır için yönelim hesabı yapılacaksa o sınırın orta noktası bulunmaktadır. Daha sonra, orta nokta; sınırın sınıfı göz önüne alınarak sağda ise sağa, solda ise sola, aşağıda ise aşağıya ve yukarıda ise yukarıya doğru 10 birim uzatılmaktadır. Uzatılan orta nokta görselleştirmek açısından Şekil 4.25 ile gösterilmektedir. Bu şekil üzerinde sınırlar siyah sürekli çizgi ile, sınırların orta noktası yeşil nokta ile, ötelenmiş noktalar mavi noktalar ile ve ötelenmiş noktaların son noktası kırmızı nokta ile görselleştirilmiştir. Bu sınır ölçümle alınmış herhangi bir nesneye sınır değildir, konseptin daha iyi açıklanması açısından verilmiştir.



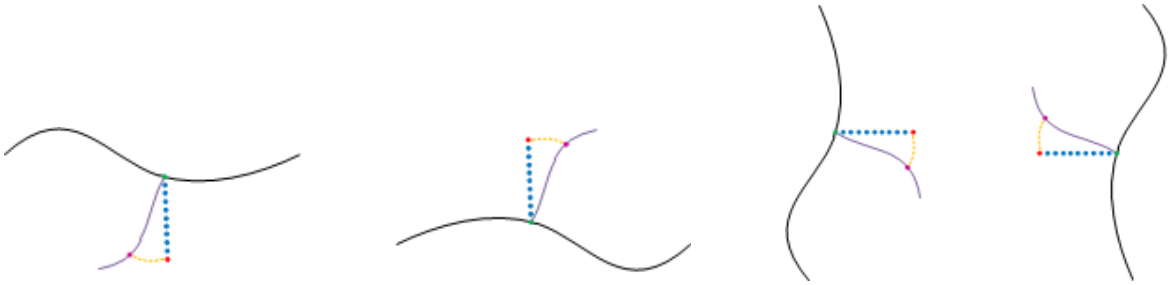
Şekil 4.24: Nesnelere Ait Yüzey Eğilim Tahminleri: Fil Nesnesine, Endüstriyel Parçaya ve Çanak Nesnesine Ait Tahmin Edilen Eğriler (a, b, c, d, i, j, k, l) ve Eğrilerin Başka Açılardan Görselleri (e, f, g, h, m, n, o, p)



(a) (b) (c) (d)

Şekil 4.25: Sınırların Orta Noktalarının Sınırın Sınıfına Göre Uzatılması: (a) Alt Sınır İçin (b) Üst Sınır İçin (c) Sağ Sınır İçin (d) Sol Sınır İçin

Kırmızı ile gösterilen sınır doğrultusunda 10 birim ötelenmiş noktanın x ve z değerleri aynı kalarak y değerleri Denklem 4.3 kullanılarak yeniden hesaplanmaktadır. Yani, noktanın yüzey eğilim eğrisine oturtturulduğu söylenilebilir. Bu ifade Şekil 4.26 ile açıklanabilir. Yukarıdaki şekilde kullanılan renkler ve figürler aynı kalmak kaydı ile, pembe çizgi tahmin edilen ikinci dereceden yüzey eğilim eğrisi ve pembe nokta bu eğri üzerine oturtturulmuş noktayı temsil etmektedir. Bu tahmin edilen eğri ve eğriye oturtturulan nokta hesapla elde edilmiş değildir, konseptin daha iyi açıklanması açısından verilmiştir.



(a) (b) (c) (d)

Şekil 4.26: Sınırın Uzatılmış Orta Noktasının Yüzey Eğilim Eğrilerine Oturtturulması (a) Alt Sınır İçin (b) Üst Sınır İçin (c) Sağ Sınır İçin (d) Sol Sınır İçin

İkinci dereceden eğri üzerinde oturtturulmuş nokta için, yüzey bilgileri ile tahmin edilmiş bilinmeyen bölgenin parametreleri kullanılarak o yüzeyin normali Denklem 4.11'e göre hesaplanmaktadır. Bu denklemdeki matrisin ikinci satırının -1 olmasının sebebi

algılayıcının bakış yönünü pozitif y ekseninin olmasıdır ve yüzey noktaları buna zıt yöndedir.

$$n_i = \begin{bmatrix} \frac{\partial f}{\partial xpi} \\ -1 \\ \frac{\partial f}{\partial zpi} \end{bmatrix} = \begin{bmatrix} 2a_1 x_{pi} + a_2 z_{pi} + a_4 \\ -1 \\ a_2 x_{pi} + 2a_3 z_{pi} + a_5 \end{bmatrix} \tag{4.11}$$

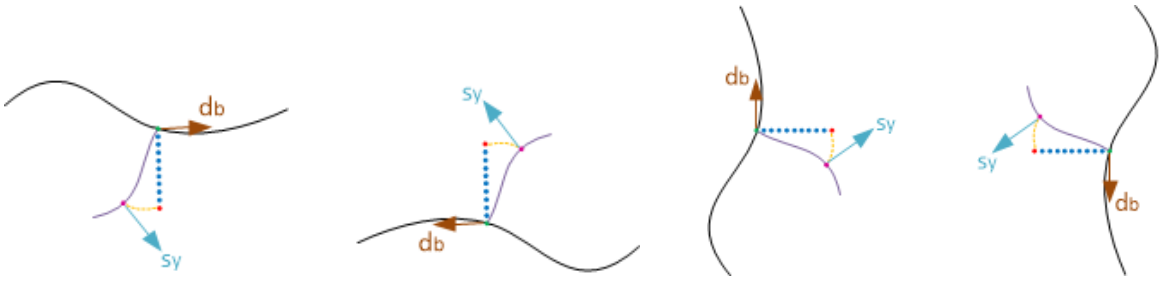
İkinci dereceden eğrinin yüzey normalinin (n_i) negatifi algılayıcının bakış açısını (s_y) temsil etmektedir. Bu ilişki Denklem 4.12'deki gibi verilmektedir.

$$s_y = -n_i \tag{4.12}$$

Algılayıcının bakış açısının z ekseninin hesaplanmasında Denklem 4.13'te verilen sınır yönü (d_b) kullanılmaktadır. Bu denklemde v_1 normalize edilmiş sınır vektör zincirinin ilk noktasını, v_m ise son noktasını temsil etmektedir.

$$d_b = dir(v_1, v_m) \tag{4.13}$$

Sınır yönü (d_b) kahverengi ok ile ve hesaplanan yüzey normali (n_i) açık mavi ok ile Şekil 4.27'de gösterilmiştir.



(a) (b) (c) (d)

Şekil 4.27: Sınırların Yönleri ve Yüzey Eğilim Eğrisinin Eğriye Oturtturulan Nokta Üzerinden Yüzey Normali (a) Alt Sınır İçin (b) Üst Sınır İçin (c) Sağ Sınır İçin (d) Sol Sınır İçin

Algılayıcının bakış açısının *x* yönelimini temsil eden s_x Denklem 4.14'teki vektör çarpımı ile hesaplanmaktadır. Bu değer elde edilirken, algılayıcı bakış açısının *x* bileşeni normalize edilerek kullanılmıştır.

$$s_x = d_b \times s_y \tag{4.14}$$

Daha sonra, algılayıcının bakış açısının *z* yönelimini temsil eden s_z, Denklem 4.15'teki gibi algılayıcının yöneliminin normalize edilmiş *x* bileşeni ve *y* bileşeninin vektör çarpımı ile hesaplanmaktadır. Bulunan *z* bileşeni de normalize edilmiştir.

$$s_z = s_x \times s_y \tag{4.15}$$

İlgili eşitlikler ile NBV adayları için ayrı ayrı normalize edilmiş s_x, s_y ve s_z değerleri bulunarak yönelim hesabı yapılmıştır. Modellenen nesneler üzerinde bazı sınırlar için bulunan yönelimlerin görselleştirilmesi NBV adaylarının pozisyon hesabından sonra yapılmıştır.

4.5.2 NBV adaylarının pozisyon hesabı

Algılayıcının yönelim hesabından sonra pozisyonu da hesaplanmaktadır. Nesnenin tamamı algılayıcının görüş alanı içerisinde olmayan algılayıcılar için tek bir bakış noktası tanımlamak yerine iki bakış noktasından oluşan bir yol bulunmaktadır. Bu yol, nesnenin bilinen yüzey uzunluğu kadar olan doğrusal bir hareket ile temsil edilmektedir. Bu yüzden, bir sonraki en iyi görüş yerine bir sonraki en iyi tarama olarak bir adlandırma yapmak daha doğru olabilmektedir. Bu çalışma kapsamında lazer profil algılayıcı kullanıldığı için tarama, çizgi şeklindedir. Tarama boyunca sabit yönelim kullanılarak sadece algılayıcının pozisyonu değiştirilmektedir.

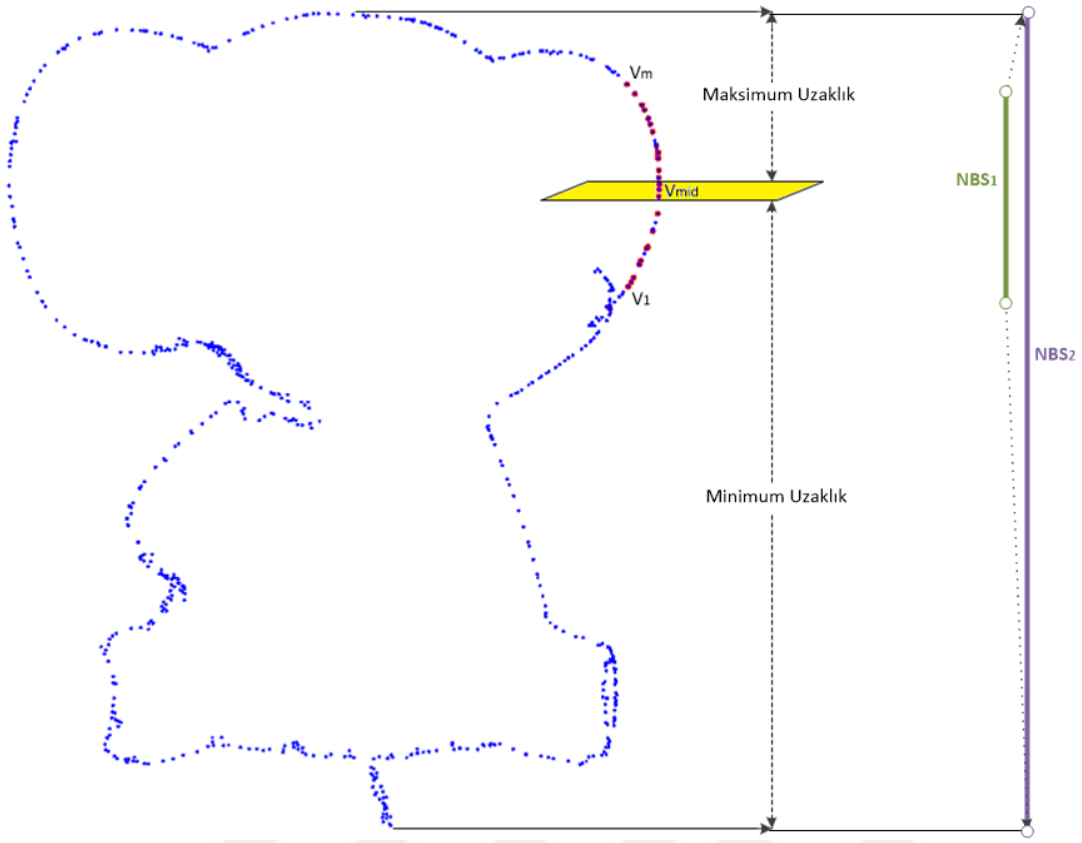
NBV adaylarının pozisyon hesabında öncelikle, yönelim hesabında da kullanılmış olan, sınırın orta noktasının yüzey eğilim eğrisi boyunca ötelenmesiyle elde edilen nokta bulunmuştur. Bu sayede sınırın orta noktası yüzey eğilimi doğrultusunda 5 ile 10 nokta arası ötelenmiş olmaktadır. Ötelemedeki bu sayılar, herhangi bir hesap ile yapılmamakta olup denemeler sonucu elde edilmiştir. Bu sayıların az seçilmesi tarama esnasında çok veri alınmamasına sebep olabileceği gibi, çok seçilmesi de veri alınacak yüzeyden fazla uzaklaşılıp istenilen şekilde veri alınmasına engel olabilmektedir.

Daha sonra, nesnenin bilinen yüzey uzunluğunun başlangıç (v_1) ve bitiş (v_m) noktalarının içerildiği bir vektör tanımlanarak bu sınır vektör zincirin orta noktası (v_{mid}) bulunmaktadır. Sonra, sınır yönünü (d_b) normal yönü olarak kabul eden ve v_{mid} noktasını içeren bir düzlem tanımlanmaktadır. Düzlem formülü Denklem 4.16'daki gibidir.

$$ax + by + cz + d = 0 \tag{4.16}$$

Denklem 4.16'da *a* katsayısı yüzey normalinin *x* bileşenine, *b* katsayısı yüzey normalinin *y* bileşenine, *c* katsayısı yüzey normalinin *z* bileşenine karşılık gelmektedir. Buradan *d* parametresi bulunabilmektedir.

Katsayıları hesaplanan düzlem, nokta bulut kümesini iki parçaya ayırmaktadır. İlk parça, düzlemin bir tarafında kalan parça, ikincisi ise düzlemin diğer tarafında kalan parçadır. İşaretli uzaklık hesaplaması yapıldığı için, düzlemin bir tarafında kalan parçaların düzleme uzaklığı alındığında pozitif, diğer tarafında kalan parçaların düzleme uzaklığı alındığında negatif değerler çıkmaktadır. Yani, girdi olarak tüm nokta bulutu verilmiş ve nokta bulutunun düzleme pozitif olan en büyük ve negatif olan en küçük değeri sırasıyla maksimum ve minimum uzaklık değerleri olarak elde edilmiştir. Düzlem normali yönünde, maksimum ve minimum uzaklıklar dikkate alınarak tarama adayının yol uzunluğu uzatılmıştır. Sadece sınırı taramak yerine, bu şekilde uzatılmış bir yol taranmasının sebebi bir tarama ile mümkün olabildiğince fazla veri almaktır, aksi takdirde; daha fazla taramanın yapılması, planlama ve hareket için daha çok zaman harcanması anlamına gelmektedir. Bu anlatılanlar Şekil 4.28 ile özetlenmiştir. Bu şekil üzerinde fil nesnesine ait en uzun sınır mavi renk ile nesnenin sağ sınırlarından biri kırmızı renk ile görselleştirilmiştir. Sağ sınırın ilk noktası v_1 ile, son noktası v_m ile ve orta noktası v_{mid} ile işaretlenmiştir. Sınırın orta noktasından geçen ve yukarıda *a*, *b*, *c* ve *d* parametreleri hesaplanan düzlem sarı renk ile görselleştirilmiştir. Nokta bulutunun filtrelenmediği ya da en üstteki ve en alttaki noktalarının değişmediği varsayımı altında düzlemin altındaki en uzak nokta minimum uzaklığı ve düzlemin üstündeki en uzak nokta maksimum uzaklığı bulmak için kullanılmaktadır. Normalde yeşil renkteki gibi elde edilen tarama adayının pozisyonu (NBS_1), bulunan bu maksimum ve minimum uzaklıklar dikkate alınarak konumu değiştirilmeden mor renkteki tarama aday pozisyonu (NBS_2) elde edilmiştir. Bu sayede tarama uzunluğu arttırılmış olup tek taramada alınabilecek maksimum veri sayısına erişilmektedir. Bu şekil sadece görsellik amaçlı olup gerçek nokta bulutunda maksimum ve minimum uzaklık noktaları, düzlem ve sonraki en iyi görüş için pozisyonlar farklılık gösterebilmektedir.



Şekil 4.28: Tarama Aday Yolunun Nesnenin Maksimum ve Minimum Noktaları Doğrultusunda Uzatılması

Algılayıcının taraması için hesaplanan yol, algılayıcının üreticiler tarafından tavsiye edilen optimum görüş uzaklığı kadar ötelenerek görüş noktasının pozisyonu hesaplanmaktadır. Bu çalışma kapsamında kullanılan ScanControl 2900-100 lazer profil algılayıcının tavsiye edilen görüş uzaklığı 190 mm - 290 mm aralığındadır.

Özetle, yüzey eğilim eğrisi boyunca ötelenmiş sınırın orta noktası, yukarıda da bahsedildiği gibi öncelikle daha fazla veri almak amacıyla düzlemin altında ve üstünde kalan en uzak noktalar kadar ötelenmiş ve tek bir nokta yerine iki noktadan oluşan bir yol elde edilmiştir. Daha sonra ise algılayıcının optimal ölçüm uzaklığı kadar algılayıcının görüş yönü doğrultusunda tekrar ötelenerek NBV adayının final pozisyonu hesaplanmıştır. NBV adayının pozisyonunu oluşturan noktalar Denklem 4.17 ile gösterilen şekilde hesaplanmıştır. Bu denklemde sp_1 ve sp_2 sırasıyla tarama yolunun başlangıç ve bitiş noktalarını, t_p yüzey eğilim eğrisi boyunca sınırın ötelenmiş orta noktasını, d_b sınırın yönünü gösteren vektörü, u_{max} ve u_{min} sırasıyla düzlemin altında ve üstünde kalan düzleme en uzak noktaların uzaklıklarını, d_s algılayıcının optimal ölçüm uzaklığını ve s_y ise algılayıcının görüş yönünün y bileşenini temsil etmektedir.

$$sp_1 = t_p + d_b.u_{max} - d_s.s_y$$
$$sp_2 = t_p + d_b.u_{min} - d_s.s_y \quad (4.17)$$

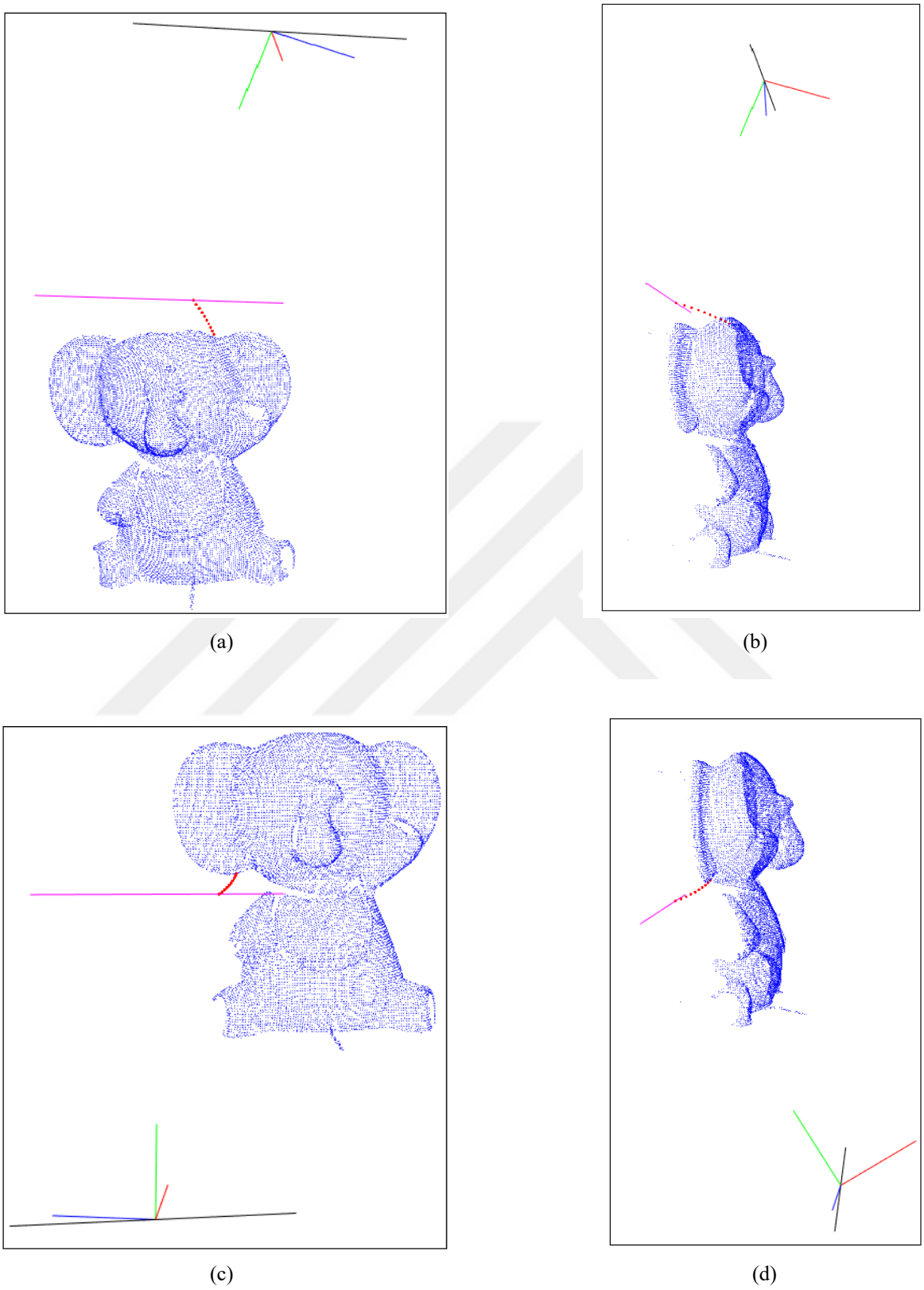
NBV adaylarının poziyon ve yönelim hesabı fil nesnesinin üst, alt, sol ve sağ sınırlarından birer tanesi için, endüstriyel parçanın üst sınırlarından ikisi için ve çanak nesnesinin sağ sınırlarından iki tanesi için yapılarak Şekil 4.29, Şekil 4.30, Şekil 4.30 ve Şekil 4.30 ile görselleştirilmiştir. Şekiller üzerindeki mavi noktalar nesneye ait nokta bulutunu, kırmızı noktalar daha önceden de verilen yüzey eğiliminin tahmin edilen eğrileri, pembe çizgiler görüş alanı uzaklığı kadar ötelenmemiş bakış pozisyonunu, siyah çizgiler görüş alanı uzaklığı kadar ötelenmiş bakış pozisyonunu; yani lazer profil algılayıcının ölçüm alacağı uzaklığa sahip NBV adayının pozisyonu temsil etmektedir. Yönelim kısmında ise; kırmızı renkteki eksen yönelimin *x* eksenini, yeşil renkteki eksen *y* eksenini ve mavi renkteki eksen *z* eksenini temsil etmektedir. Bulunan yol boyunca algılayıcı sabit yönelim ile ölçüm alma işlemini gerçekleştirmelidir.

NBV adayları için pozisyon ve yönelim hesabı yapılmasıyla 3B modellenen nesne için NBV aday kümesi elde edilmektedir. Bundan sonra yapılması gereken aşamalar, NBV adaylar kümesi içinden bir adayı seçmek ve NBV için gerekli robot ve döner tabla hareket planının yapılmasıdır. Burada bahsedilen tüm işlemler sonlanma kriteri sağlanana kadar yapılmalıdır.

4.6 Kalite Kriteri

Tamamlanmayı ve kaliteyi gösteren lokal özellikler önemlidir; çünkü modelin tamamlanması ve yüksek kalitede bir 3B model elde edilmesi istenmektedir. Yüzey özelliklerin güncellenmesinde üç farklı ölçüt vardır. Bunlar; örnekleme yoğunluğu (sampling density), ortalama yüzey normalleri (average surface normals) ve sınır kenarlarının miktarıdır (amount of border edges). Her bir voksel için bu özelliklerin hesaplanması gerektiğinden ilk olarak nokta bulut kümesinin, sekizli ağaç yapısı kullanılarak voksellerine ayrılması gerekmektedir. Vokseller, sekizli ağaç yapısı ile elde edildikten sonra, voksellerin görselleştirilmesi için voksellerin merkezleri ve PCL'in ModelCoefficients kütüphanesi kullanılarak elde edilen küp katsayıları kullanılmıştır.

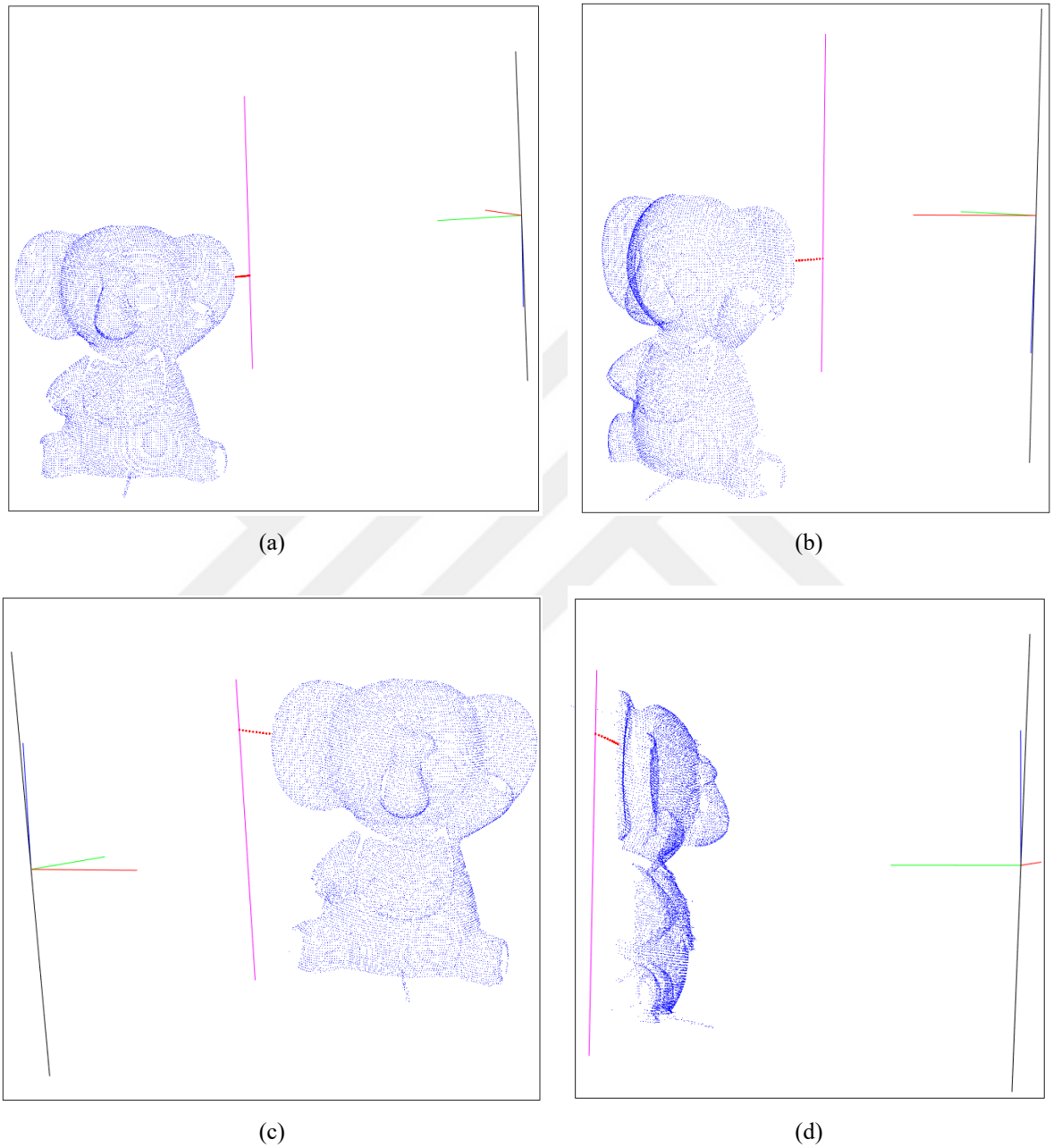
Voksellerin boyutu çözünürlüğe bağlı olarak değişmektedir. İstenilen voksel boyutuna göre çözünürlük ayarlanabilmektedir . Vokselleri çok küçük seçmek işlem yükünü ve dolayısıyla otonom taramada geçen süreyi arttıracağı için kullanılmamaktadır. Aynı şekilde, vokselleri çok büyük seçmek ise nesnenin lokal özelliklerini çok iyi



(a) (b)

(c) (d)

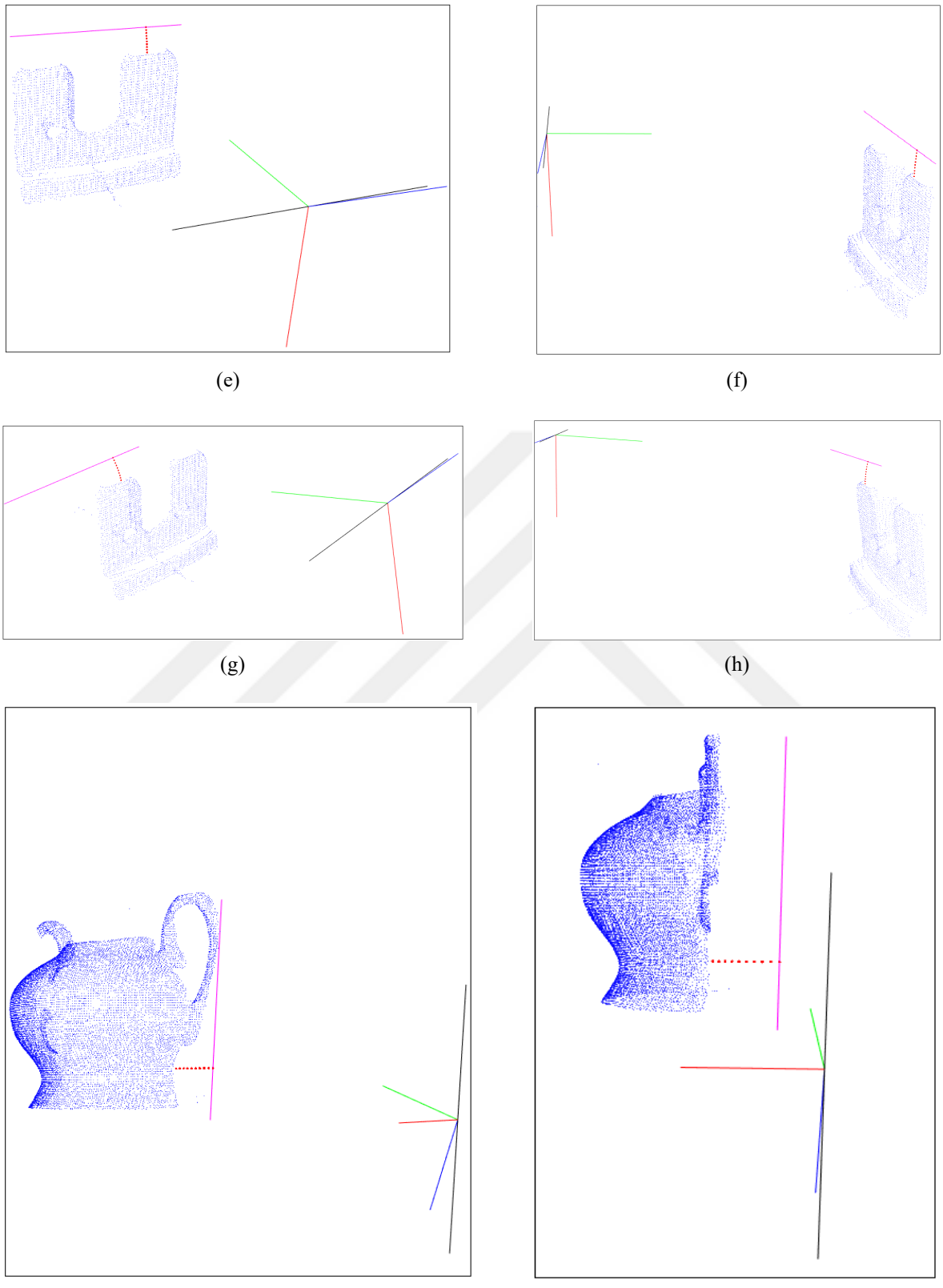
Şekil 4.29: Fil Nesnesine Ait Bazı Sınırlar İçin Hesaplanan NBV Adaylarının Pozisyon ve Yönelimlerinin Farklı Açılardan Görselleri: Üst Sınırı İçin (a, b), Fil Nesnesinin Alt Sınırı İçin (c, d)



(a) (b)

(c) (d)

Şekil 4.30: Fil Nesnesine Ait Bazı Sınırlar İçin Hesaplanan NBV Adaylarının Pozisyon ve Yönelimlerinin Farklı Açılardan Görselleri: Fil Nesnesinin Sağ Sınırı İçin (a, b), Fil Nesnesinin Sol Sınırı İçin (c, d)

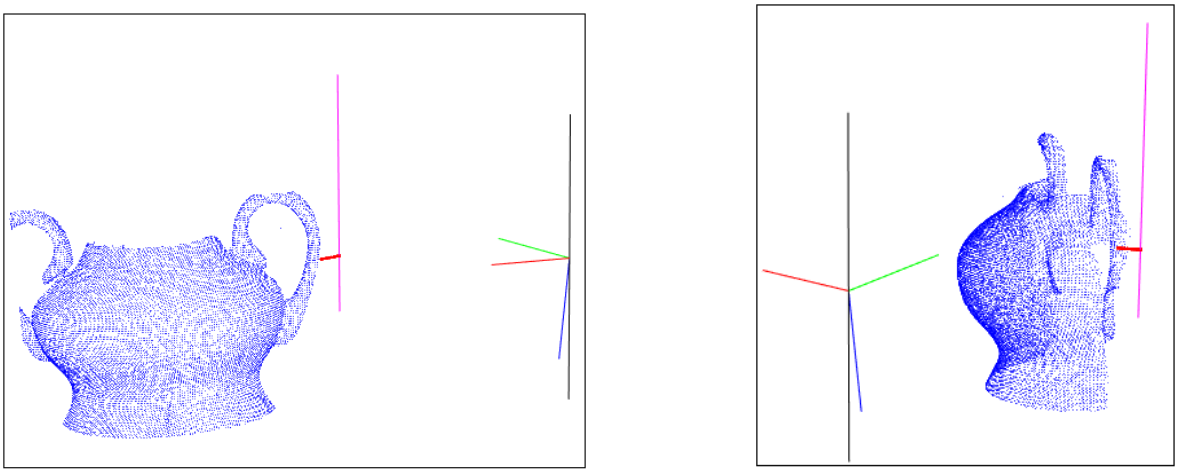


(e) (f)

(g) (h)

(i) (j)

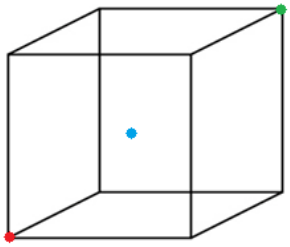
Şekil 4.30: Endüstriyel Parça ve Çanak Nesnesinin Bazı Sınırları İçin NBV Adaylarının Pozisyon ve Yönelimlerinin Farklı Açılardan Görselleri: Endüstriyel Parça Üst Sınırlarından Biri (a, b) ve Diğeri (c, d) İçin, Çanak Sağ Sınırlarından Biri İçin (e, f)



(k) (l)

Şekil 4.30: Çanak Nesnesine Ait Bazı Sınırlar İçin Hesaplanan NBV Adaylarının Pozisyon ve Yönelimlerinin Farklı Açılardan Görselleri: Çanak Nesnesinin Sağ Sınırlarından Diğeri İçin (a, b)

yansıtmayacağı ve kalite ölçütlerini istenilen şekilde gerçeklemeyeceği için kullanılmamaktadır. Yukarıda bahsedilen ölçütlerin uygulanması, bütün voksellerin içinde bulunan noktalara ayrı ayrı erişim ile mümkün olmaktadır. Noktalara erişim için, bilinen voksel merkezinden vokselin minimum ve maksimum noktaları bulunmaktadır. Şekil 4.31'da kırmızı nokta vokselin minimum noktasını, yeşil nokta vokselin maksimum noktasını ve mavi nokta ise vokselin merkezini temsil etmektedir.



Şekil 4.31: Voksel Yapısının Minimum Noktası, Merkezi ve Maksimum Noktasının Görseli

R_n normal yarıçapı, R_r ise noktalar arası minimum uzaklığı gösteren yarıçap olmak üzere aşağıda kalite ölçütlerin nasıl hesaplandığı anlatılmaktadır.

4.6.1 Örnekleme yoğunluğu

N_{act} değişkeninin R_n normal yarıçapı içine düşen noktalar olarak tanımlandığı ve N_{max} değişkeninin ise R_r yarıçapı içine düşebilecek mümkün olan en fazla nokta sayısı olduğu düşünülerek Denklem 4.18'deki gibi yoğunluk hesaplanmıştır. Bu değer 0 ve 1 arasında çıkmaktadır. N_{max} değişkeni ise Denklem 4.19'daki gibi hesaplanmıştır.

$$d(v) = \frac{N_{act}}{N_{max}} \tag{4.18}$$

$$N_{max} = \frac{\sqrt{2}\pi R_n^2}{R_r^2} \tag{4.19}$$

Daha sonra her bir voksel için hesaplanan yoğunluk değerleri, örnekleme yoğunluğu hesaplanan voksel içindeki nokta sayısına bölünerek ortalama yoğunluk, yani örnekleme yoğunluğu bulunmuş olur. Algoritma 5, örnekleme yoğunluğu hesaplanmasını içeren algoritmayı göstermektedir.

Algorithm 5 Örnekleme Yoğunluğunun Hesaplanması

```
for indeks1:0'dan voksel sayısına kadar ilerle do
    toplamYoğunluk değişkenini sıfırla
    toplamOrtalamaYoğunluk değişkenini sıfırla
    octree yapısını örnekleme yoğunluğu hesaplanan voksel için çağır
    for indeks2:0'dan vokselin içindeki nokta sayısına kadar ilerle do
        Octree yapısından arama noktasını bul
        if Octree yapısı için yarıçaplı arama yapılacaksa then
            Nact değişkenine arama yarıçapındaki nokta sayısını ata
            toplamYoğunluk+ =(Nact/Nmax)
        end if
    end for
    toplamOrtalamaYoğunluk = toplamYoğunluk / vokseldeki nokta sayısı
end for
```

4.6.2 Ortalama yüzey normalleri

Voksel içine düşen her bir nokta için R_n yarıçapında normal değerleri hesaplanarak toplanmaktadır. Bulunan bu yüzey normalleri, ortalama yüzey normali hesaplanan voksel içindeki nokta sayısına bölünerek o voksel için ortalama yüzey normalinin değeri bulunmuş olur. Bu işlem bütün vokseller için ayrı ayrı uygulanarak her bir voksel için ortalama yüzey

normali elde edilmiştir. Algoritma 6, ortalama yüzey normallerinin hesaplanmasını içeren algoritmayı göstermektedir.

Algoritma 6 Ortalama Yüzey Normallerinin Hesaplanması

```
for indeks1:0'dan voksel sayısına kadar ilerle do
    Sekizli ağaç yapısını ortalama yüzey normali hesaplanan voksel için çağır
    for indeks2:0'dan voxelin içindeki nokta sayısına kadar ilerle do
        toplamNormaller değişkenini sıfırla
        toplamOrtalamaNormaller değişkenini sıfırla
        vokseldeki nokta bulutunu girdi yaparak normal hesapla
        for indeks3:0'dan hesaplanan normallerin boyutuna kadar ilerle do
            Normallerin toplamını bul
        end for
    end for
    toplamOrtalamaNormaller =toplamNormaller/hesaplanan normallerin boyutu
end for
```

4.6.3 Sınır kenarlarının miktarı

Gerçeklenen makalede her voksele düşen sınır kenarlarının sayısının, voksel içerindeki sınır kenarlarına oranı ile sınır kenarlarının miktarı bulunmuştur. Burada ise, benzer bir yaklaşımla bu oran her voksele düşen sınır noktalarının sayı olarak toplamının o voksel içerisindeki toplam nokta sayısına bölümü ile hesaplanmıştır. Tarama sayısı arttıkça, bu oran 0'a yaklaşmalıdır, çünkü tamamlanmış bir nesnede sınır kalmaması beklenmektedir. Algoritma 7, sınır kenarlarının miktarının hesaplanmasını içeren algoritmayı göstermektedir.

Algorithm 7 Sınır Kenarlarının Miktarının Hesaplanması

```
for indeks1:0'dan voksel sayısına kadar ilerle do
    sayaç değişkenini 0'a ata
    sekizli ağaç yapısını sınır kenarlarının miktarı hesaplanan vokel için çağır
    for indeks2:0'dan sınır nokta bulutunun boyutuna kadar ilerle do
        arama noktasını sınır noktasının indeks2li indekse sahip elemanına ata
        sekizli ağaç yapısı ile en yakın k-komşuluğu kodunu çağır
        if en yakın komşuluk uzaklığı 0'a eşitse then
            sayaç değişkenini 1 arttır
        end if
    end for
```

sınırKenarlarınınMiktarı =sayaç/sınır nokta bulutunun boyutu
end for

4.7 NBV Seçimi

Bütün sınıflandırılmış sınırlar için NBV adaylarının pozisyon ve yönelimini hesapladıktan sonra içlerinden birinin tarama ile veri alınması amacıyla seçilmesi gerekir. Bu işlem NBV seçimi olarak adlandırılmaktadır. NBV seçiminin tüm sınırlar için yapılması, yapılan uygulama gerçek zamanlı olduğu için işlem yükünü arttırarak süreci yavaşlatmaktadır. Bu yüzden, NBV seçim işleminin, sınıflandırılmış sınırlar bulunduktan sonra yapılması, kullanılmayacak sınırlara ait yapılacak olan gereksiz hesaplamaları ortadan kaldırmaktadır. Çünkü, NBV seçim kriteri bu hesaplardan bağımsız olarak gerçekleştirilmektedir.

NBV seçim işlemi sınıflandırılmış sınırlar üzerinden sınırdaki nokta sayısı ile yapılmaktadır. Nokta sayısı en fazla olan sınırın uzun bir sınır olduğu varsayımı altında bu seçim yapılmaktadır; çünkü lazer profil algılayıcı tarafından alınan noktalar arası uzaklık birbirine yakın olmaktadır. Ayrıca, nokta bulutu zaten filtrelenmiş olduğu için noktalar arası ani sıçramaların olmaması beklenmektedir. Bu şartlar altında, NBV seçimi için noktalar arası uzaklığın kullanılması makul bir seçimdir.

NBV seçiminde başka ölçütler de kullanılabilir. Bunlardan biri, noktalar arası toplam Öklid uzaklığı fazla olan sınır ile hesaplanan bir sonraki en iyi görüş noktası olarak kullanılması olabilir. Başka bir ölçüt ise, nesne üzerinden alınan bilinmeyen bölgedeki veri miktarını, yani bilgi kazancını, maksimize edecek sınır ile hesaplanan bir sonraki en iyi görüş noktasının kullanılması olabilir. Fakat şu anki mevcut sistem için, nokta sayısı fazla olan sınır üzerinden hesaplanan bir sonraki en iyi görüş noktasının kullanılması yeterli bir ölçüt olup diğer ölçütler ileriye yönelik çalışmalarda eklenebilir.

5. 3B YÜZEY MODELLEMEDE SINIFLANDIRMA

Bilinmeyen nesnelerin 3B modellenmesi çalışmasına yenilik katabilecek en önemli ölçütlerden biri sınıflandırmadır. Burada, sınıflandırma ile amaçlanan, cismin elde edilen modelinin yüzeyindeki nokta verisinin bazı sınıflara ayrılmasıyla o bölgenin tekrar taranıp taranmaması gerektiğine dair bir ölçüt elde edilmesidir.

Aslında sınıflandırma ölçütünün çıkış sebebi, taranan nesnelerin homojen bir şekle sahip olmaması yani; girintilere ve çıkıntılara sahip olmasıdır. Nesneler, dış bükey (konveks) ve iç bükey (konkav) olarak iki ana sınıf ile temsil edilebilmektedir (Düz yüzeyler hariç). Özellikle, iç bükey yüzeye sahip olan nesnelerde, nesne bu özelliği gereği, algılayıcının veri almasını engelleyebilmekte veya algılayıcı tarafından diğer bölgelere nispeten daha az nokta verisi edinimine sebep olabilmektedir. Bu nedenlerden kaynaklı, nesneye ait elde edilen nokta bulutu eksik olarak tamamlanmış veya tamamlanamamış olabilmektedir. Bu sebeple, bir yenilik ölçütü olarak, sınıflandırmanın da eklenmesi, modellenen nesneye ait daha yüksek kalitede bir nokta bulutu elde edilmesini sağlayabilmektedir.

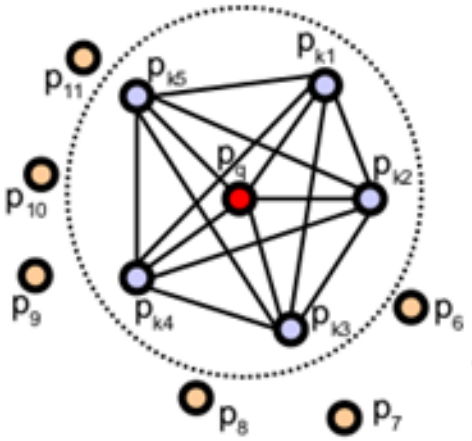
Nesne yüzeyinden elde edilmiş nokta bulutlarının sınıflandırılmasını sağlamak amacıyla Hızlı Nokta Özellik Histogramları (FPFH – Fast Point Feature Histogram) kullanılmaktadır. FPFH kavramının daha iyi anlaşılması açısından, onun önceki aşaması olan Nokta Özellik Histogramları'ndan (PFH – Point Feature Histogram) bahsettikten sonra bu kavramdan bahsetmek daha faydalı olacaktır.

5.1 Nokta Özellik Histogramları

Nokta özellik histogramları, 3B bir nokta bulutu kümesindeki bir *p* noktası etrafında lokal geometrik özellikleri tanımlayan çok boyutlu tanımlayıcılar olarak tanımlanmaktadır. Nokta özellik histogramlarının hesaplanması, 3B nokta koordinat değerlerini ve yüzey normallerini temel aldığı için düzgün yapılmış bir ölçümden elde edilen nokta bulut kümesi ve uygun şekilde hesaplanmış normallerin kullanılması farklı sınıflar için daha ayırt edici histogramların elde edilmesini sağlamaktadır (Rusu vd., 2009).

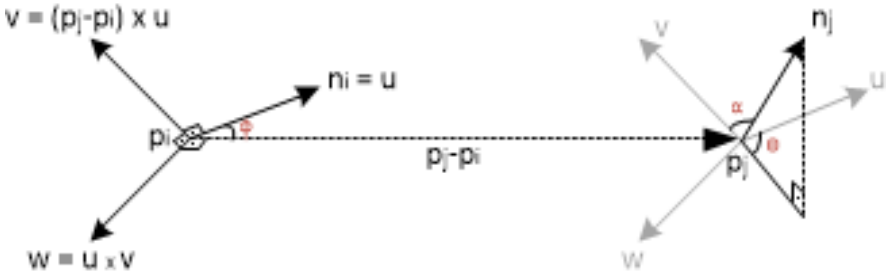
Nokta özellik histogramlarının hesaplanması iki ana aşama ile gerçekleşmektedir. İlk aşama, nokta özellik histogramı hesaplanan noktayı merkez alan bir *r* yarıçapı belirleyerek bu yarıçap içine düşen k-komşuluğundaki noktaların bulunmasıdır. Bu ifadenin

görsel hali Şekil 5.1'deki gibidir. Bu şekil üzerinde, p_q ile gösterilen, nokta özellik histogramı hesaplanan noktaya; p_{k1},p_{k2},p_{k3},p_{k4} ve p_{k5} ile gösterilen noktalar ise *r* yarıçapına düşen k-komşuluğundaki noktalara karşılık gelmektedir.



Şekil 5.1: Nokta Özellik Histogramı Hesabında *r* Yarıçapı ve Bu Yarıçap İçine Düşen k-komşuluğundaki Noktaların Bulunması (Rusu vd., 2009)

İkinci aşama ise daha fazla matematiksel işlem gerektiren bir aşamadır. Yukarıdaki şekilde gösterilen *r* yarıçapı içine düşen her bir nokta çiftinin birbiri ile olan bağlantısı ağ yapısı ile gösterilmiştir. İşte bu ağ yapısı ile gösterilen her bir nokta çifti için, nokta özellik histogramı hesaplanan nokta kaynak nokta (p_i) ve komşuluğundaki nokta hedef nokta (p_j) olmak üzere kaynak nokta etrafında uvw çerçevesi Denklem 5.1, 5.2 ve 5.3'te verilen formüller ile oluşturulduktan sonra bu çerçeve hedef nokta üzerine Şekil 5.2'deki gibi taşınmaktadır.



Şekil 5.2: Nokta Özellik Histogramında uvw Çerçevesinin Hedef Nokta Üzerinde Belirlenmesi ve Kaynak Nokta Üzerinde Taşınması (Rusu, 2009)

$$u = n_s \tag{5.1}$$

$$v = u \times \frac{p_j - p_i}{\left\| p_j - p_i \right\|} \tag{5.2}$$

$$w = u \times v \tag{5.3}$$

Denklem 5.4, 5.5 ve 5.6'da verilen açılara karşılık gelen değerler bulunduktan sonra bu aşama da tamamlanmış olmaktadır.

$$\alpha = v.n_j \tag{5.4}$$

$$\phi = u.\frac{p_j - p_i}{\left\| p_j - p_i \right\|} \tag{5.5}$$

$$\theta = arctan(w.n_j, u.n_j) \tag{5.6}$$

Nokta özellik histogramlarının ayırt edici gücü farklı sınıflara ait histogramlar için farklı değerler verip vermemesiyle ölçülebilmektedir. Rusu vd. (2009) tarafından yapılan çalışmada, nokta özellik histogramlarının ayırt edici gücünü göstermek amacıyla gri seviyede karmaşıklık matrisi kullanılmıştır. Bu matris; silindir, küre, düz yüzey, kenar ve köşe sınıflarını ayrı ayrı içeren ortalama nokta özellik histogramların farklarının alınmasıyla elde edilmiştir. Gri seviyelerdeki gözle görülür farklılık yöntemin etkinliğini göstermektedir.

Nokta özellik histogramların hesabında, bir yarıçap içerisine düşen komşuluk başka bir komşuluk içine de düşebilmektedir. Bu da aynı matematiksel hesaplamaların birden fazla kez yapılması anlamına gelmekte ve süre bazında hesaplama maliyetini arttırmaktadır. Bu değerlerin geçici bir hafızada tutulması bu maliyeti azaltmaktadır. Geçici dolarak tutulan bu değerlerin hafızayı aşması durumunda ilk giren ilk çıkar (First In First Out – FIFO) mantığı kullanılarak bu sorunun önüne geçilmektedir. Geçici hafızada tutma ve ilk giren ilk çıkar mantığının daha etkin kullanılması için sekizli ağaç yapısı kullanılarak noktaların indekslemesi yapılmış ve süre bazındaki maliyet bir nebze de olsa azaltılmıştır. Bütün bunlara rağmen nokta özellik histogramları gerçek zamanlı uygulamalarda kullanılabilecek kadar hızlı sonuçlar üretmemektedir; çünkü her bir nokta için histogramı oluşturan 125 boyutlu tanımlayıcılar kullanılmıştır.

5.2 Hızlı Nokta Özellik Histogramları

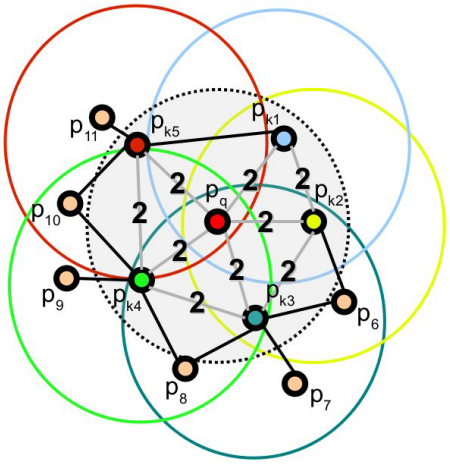
Nokta özellik histogramların gerçek zamanlı uygulamalarda kullanılabilecek kadar hızlı olmaması matematiksel işlemlerde iyileştirmeler yapma ihtiyacı doğurmuştur. Bu amaçla hızlı nokta histogramları ortaya çıkmıştır. Her ne kadar bu özellik iç ortamlarda 3B yeniden yapılandırma (3D registration) sorununu çözmek için ortaya çıksa da, bu çalışma kapsamında farklı geometrik şekillerin farklı hızlı nokta özellik histogramlarına sahip olması özelliği kullanılmaktadır.

Nokta özellik histogramları gibi hızlı nokta özellik histogramları da iki ana aşamadan oluşmaktadır. İlk aşamada, bir *r* yarıçapı içinde k-komşuluğuna düşen noktaları belirledikten sonra, hızlı nokta özellik histogramı hesaplanan nokta ile bu yarıçap dahilinde direk bağlantısı olan komşu noktaların kullanılmasıyla basitleştirilmiş nokta özellik histogramı (SPFH – Simplified Point Feature Histogram) elde edilmektedir. İkinci aşamada ise *r* yarıçapı içinde k-komşuluğuna düşen noktalar için, o noktaları merkez alan yine *r* yarıçapında k-komşuluğuna düşen noktalar bulunarak başka bir basitleştirilmiş nokta özellik histogramı elde edilmektedir. Bütün komşu noktalar için ayrı ayrı elde edilen basitleştirilmiş nokta özellik histogramları, komşu nokta ve hızlı nokta özellik histogramı hesaplanan noktanın arasındaki mesafe ile ağırlıklandırılarak ilk aşamada elde edilen basitleştirilmiş nokta özellik histogramı ile toplanmış ve hızlı nokta özellik histogramı elde edilmiş olmaktadır. Bu ifadenin matematiksel gösterimi Denklem 5.7'de gösterildiği gibidir.

$$FPFH(p) = SPH(p) + \frac{1}{k}\sum_{i=1}^{k}\frac{1}{\omega_k}SPH(p_k) \tag{5.7}$$

Şekil 5.3'te p_q ile gösterilen nokta, hızlı nokta histogramı hesaplanan noktaya karşılık gelmekte ve etrafında oluşturulan *r* yarıçapı kesikli çizgilerle gösterilmektedir. Kırmızı, açık mavi, sarı, koyu mavi ve yeşil yarıçaplar ise p_q noktasının direk komşuları olan p_{k1},p_{k2},p_{k3},p_{k4} ve p_{k5} noktalarını merkez alan *r* yarıçaplarını göstermektedir. Kenarlar üzerinde 2 ile gösterilen sayılar ise aynı kenarın bu hesaplamada iki kez kullanıldığını göstermektedir.

Nokta özellik histogramı ile hızlı nokta özellik histogramı arasındaki temel farklar şu şekilde sıralanabilir:



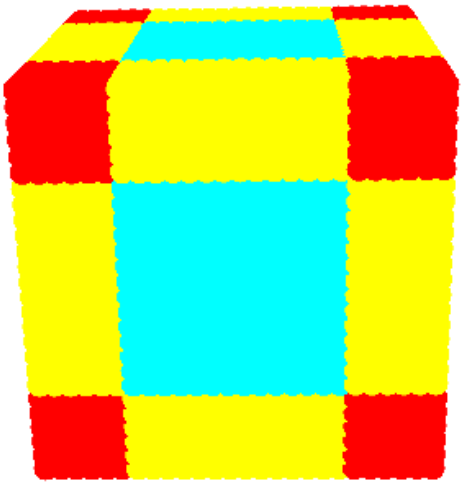
Şekil 5.3: Hızlı Nokta Özellik Histogramı Hesabında *r* Yarıçapları ve Bu Yarıçaplar İçine Düşen k-komşuluğundaki Noktaların Bulunması (Rusu vd., 2009)

- Nokta özellik histogramında *r* yarıçapında k-komşuluğundaki tüm noktalar birbirleri ile bağlanarak hesaplamalara dahil olurken, hızlı nokta özellik histogramında *r* yarıçapında k-komşuluğundaki direk bağlantısı olan noktalar hesaplamalara dahil olmaktadır.
- Nokta özellik histogramıında *r* yarıçapında hesaplamalar yapılırken, hızlı nokta özellik histogramında ise komşu noktanın *r* yarıçapı da hesaba katıldığı için yarıçap *2r*'ye kadar genişleyebilmekte, bu da lokal bazı özelliklerin kaybolmasına neden olabilmektedir.
- Nokta özellik histogramında ağırlıklandırma olmazken, hızlı nokta özellik histogramında basitleştirilmiş nokta özellik histogramı ve ağırlıklandırılan basitleştirilmiş nokta özellik histogramının birleştirilmesi sonucu bazı komşulukların iki kez katkı sağlamış olması mümkün olabilmektedir.

5.3 FPFH Histogram Karşılaştırması İle Sınıflandırma

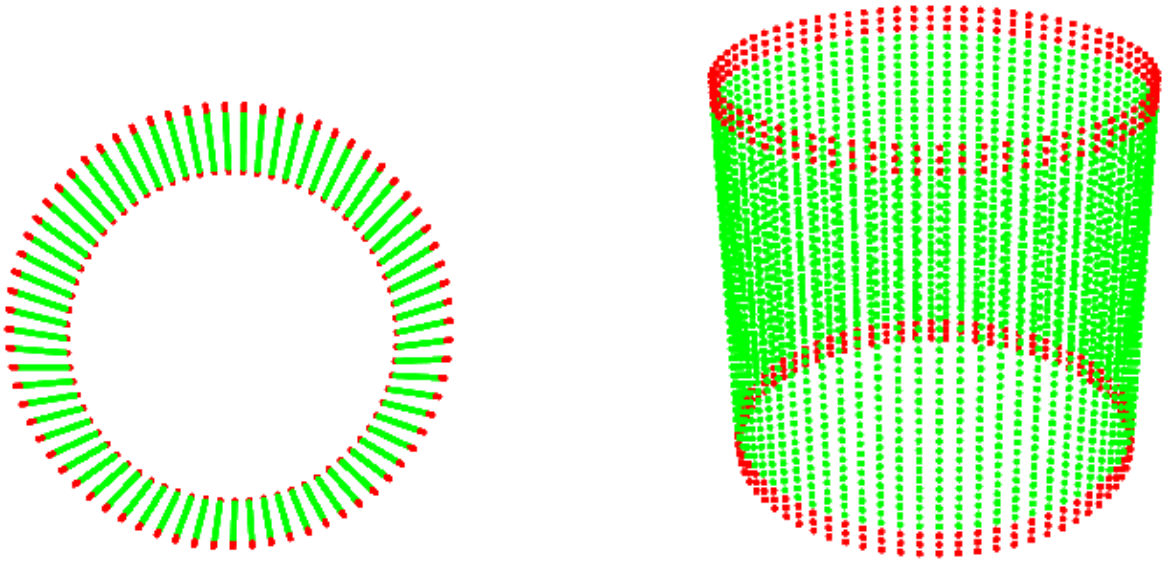
Hızlı nokta özellik histogramların karşılaştırılmasında amaç; köşe (corner), kenar (edge), düz yüzey (plane), küre (sphere) ve silindir (cylinder) sınıflarına ait her bir nokta için içlerinde benzer histogramlar elde edilmesi beklentisiyle farklı bu 5 sınıf için farklı histogramların elde edilip edilmediğini kontrol etmektir.

Şekil 5.4'te bir küpün parçalanmasıyla elde edilen köşe, kenar ve düz yüzey sırasıyla kırmızı, sarı ve mavi renk ile görselleştirilmiştir. Yine bu şekilden görüldüğü üzere, kenarlar ve köşeler tek bir nokta veya tek sıra nokta dizisi olarak değil bölge olarak seçilmiştir. Bunun amacı, bu bölge için bir tolerans aralığı oluşturmaktır.



Şekil 5.4: Küp Üzerinde Köşe, Kenar ve Düz Yüzey Bölgeleri

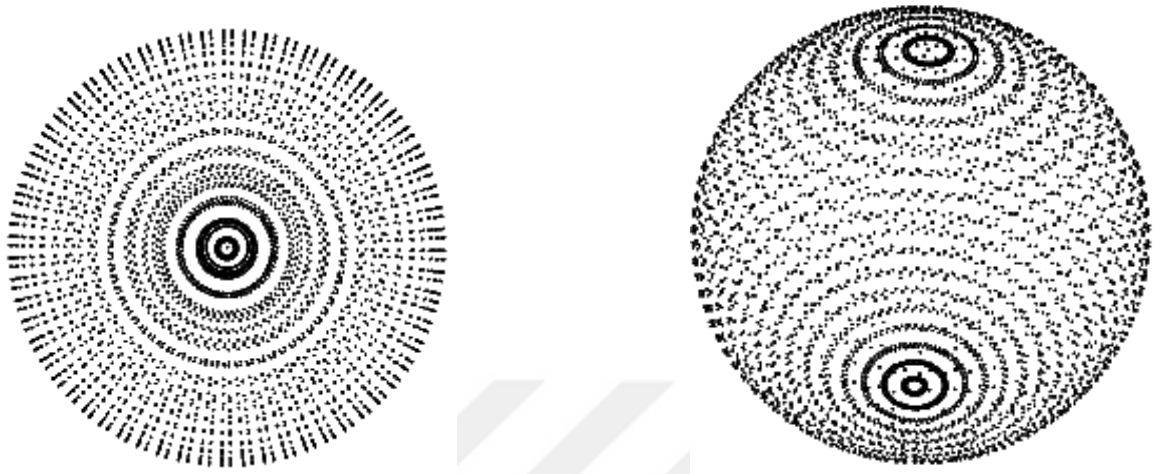
Sınıflardan bir diğeri yarıçapı 1 olan silindir Şekil 5.5'te üstten ve yandan görselleştirilmiştir. Burada, küp sınıfında da olduğu gibi, bir tolerans aralığı oluşturmak amacıyla sınır bölgeleri çıkarılmıştır. Yeşil renk ile gösterilen silindirin sınır bölgeleri çıkarıldıktan sonra silindirin karakteristiğini yansıtan noktalar yeşil renk ile görselleştirilmiştir.



(a) (b)

Şekil 5.5: Silindir Nokta Bulutunun Üstten ve Yandan Görselleştirilmesi (a) Silindir Nokta Bulutunun Üstten Görselleştirilmesi (b) Silindir Nokta Bulutunun Yandan Görselleştirilmesi

Yarıçapı 2 olan diğer bir sınıf kürenin üstten ve yandan görselleştirilmesi Şekil 5.6'da verilmiştir.



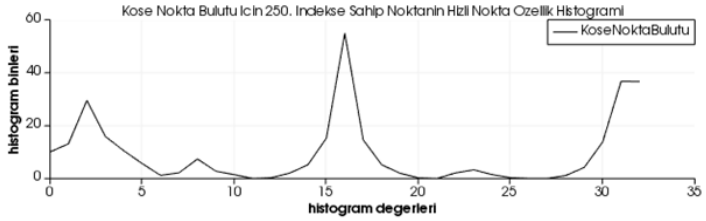
(a) (b)

Şekil 5.6: Küre Nokta Bulutunun Üstten ve Yandan Görselleştirilmesi (a) Küre Nokta Bulutunun Üstten Görselleştirilmesi (b) Küre Nokta Bulutunun Yandan Görselleştirilmesi

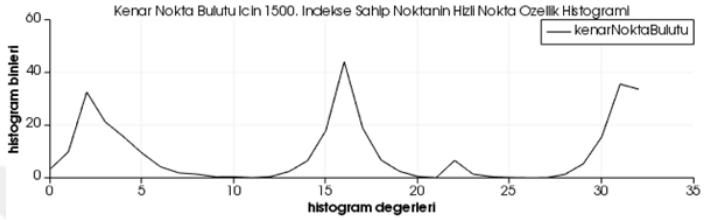
Köşe nokta bulutunda 250 numaralı, kenar nokta bulutunda 1500 numaralı, düz yüzey nokta bulutunda 400 numaralı, silindir nokta bulutunda 1000 numaralı, küre nokta bulutunda 1500 numaralı indekslere sahip noktaların hızlı nokta özellik histogramları Şekil 5.7'deki gibi görselleştirilmiştir. Burada görüleceği üzere, her bir sınıf farklı histogramlara sahiptir. Bu farklılık, sınıflandırma yaparken ayırt edici gücü göstermektedir. Yani, histogramlar ne kadar farklı olursa sınıfların ayırt edilmesi o kadar kolay olacaktır.

Sınıflandırma yapmak için, üretilen ve FPFH değerleri elde edilen her sınıftaki noktalar eğitim seti (training set) ve test seti (test set) olarak ayrılmaktadır. Bu ayrılma işlemi gerçekleştirilirken, eğitim ve test setleri elde edilmeye çalışılan nokta bulutunun nokta sayısının 2/3'ü eğitim seti, 1/3'ü ise test seti olarak ayrılmaktadır. Yani, test ve eğitim setleri aynı nokta bulutu üzerinden seçilmektedir.

Veriler, her 5 sınıf için eğitim seti ve test seti olarak ayrıldıktan sonra, eğitim setindeki noktaların hepsi için FPFH değerleri bir araya getirilerek büyük bir histogram kümesi elde edilmektedir. Eğitim setindeki noktaların hangi sınıfa ait olduğu bilindiğinden onların etiket atama işlemi tarafımızdan yapılmaktadır. Büyük bir histogram kümesi oluşturma işlemi test seti için de ayrıca yapılmaktadır. Test setindeki her bir noktanın FPFH değerleri, eğitim setindeki her bir noktanın FPFH değerleri ile teker teker karşılaştırılarak



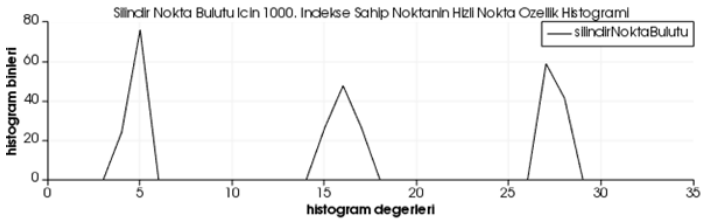
(a)



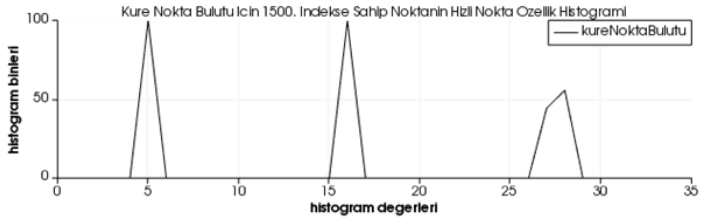
(b)



(c)



(d)



(e)

Şekil 5.7: Her Bir Nokta Bulutu İçin Hızlı Nokta Değer Histogramları (a) Köşe Nokta Bulutu İçin (b) Kenar Nokta Bulutu İçin (c) Düz Yüzey Nokta Bulutu İçin (d) Küre Nokta Bulutu İçin (e) Silindir Nokta Bulutu İçin

farklarının mutlak değerleri alınarak toplanmıştır. Bu değerin minimum olduğu noktanın etiketine bakılarak test setindeki noktanın etiketine o etiket atanmaktadır. Bu şekilde bütün test setindeki noktalar için etiketler elde edilmektedir.

Test noktaları için etiket atama işlemi tamamlandıktan sonra, noktaların doğru sınıflandırılıp sınıflandırılmadığına dair bir doğruluk hesabı yapmak gerekmektedir. Nokta bulutları tarafımızdan oluşturulduğu için noktaların kenar, köşe, düz yüzey, silindir veya küre sınıflarından hangisine ait olduğu bilinmektedir. Bu yüzden, test seti için bilinen etiketler olarak bir etiket seti oluşturumaktadır. Test setine ait, bilinen etiket seti ile az önce bahsedilen fark yöntemi ile elde edilmiş etiket seti karşılaştırılmaktadır. Bu sınıflarda aynı olan etiketler doğruluk değerini arttırırken, yanlış olanlar doğruluk değerini düşürmektedir. Bu şekilde yapılan test için doğruluk oranı yüzde 98 çıkmıştır.

Doğruluk oranı hesabında en önemli unsur, normal hesaplamak için kullanılan yarıçap ile FPFH değerlerini hesaplarken kullanılan yarıçap parametrelerinin düzgün bir şekilde ayarlanmasıdır. Aksi takdirde bu değişimden doğruluk oranı büyük ölçüde değişeceği gibi silindir ile küre sınıflarında veya kenar ve düz yüzey sınıflarında büyük bir karışıklık olabilecektir. Zaten parametrelerdeki önem referans verilen bildiri tarafından da vurgulanmıştır.

6. UYGULAMA ORTAMININ TASARIMI

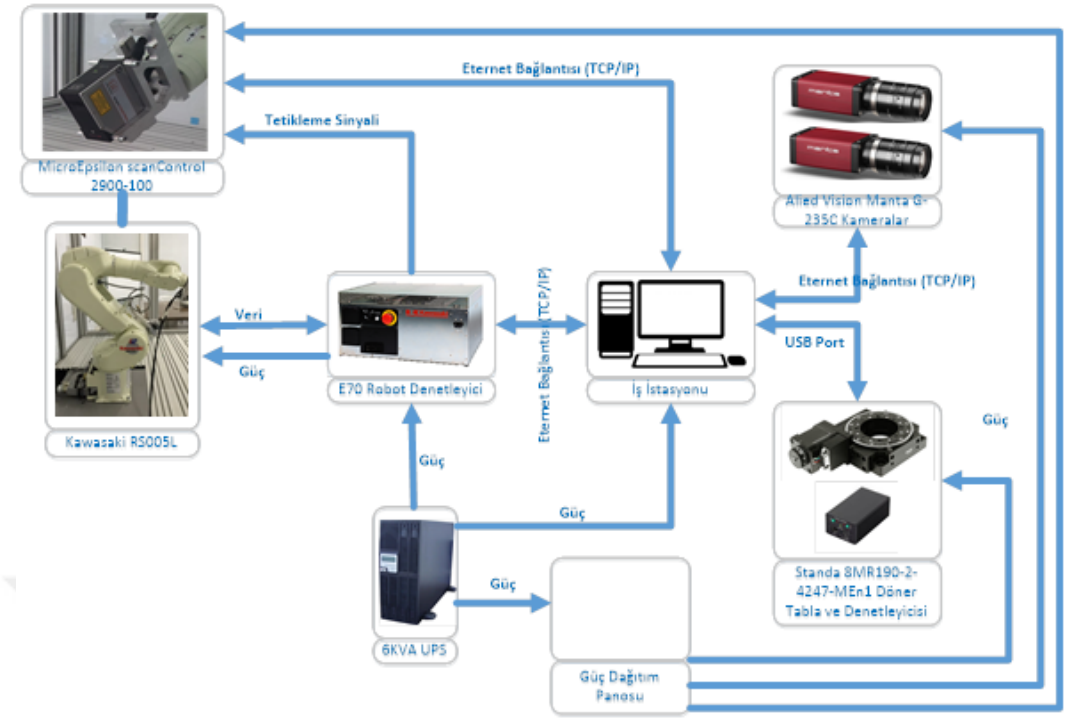
NBV algoritmasının uygulanacağı ortamın tasarımı; sistem mimarisi, endüstriyel robot kollarının programlaması için arayüz tasarımı ve döner tabla etrafında lokal koordinat sisteminin oluşturulması olmak üzere üç ana kısımdan oluşmaktadır. Sistem mimarisi, NBV algoritmasının uygulanmasında kullanılan donanımları içermekte olup donanımlar hakkındaki teknik özellikler bu başlık altında detaylı olarak verilmektedir. Endüstriyel robot kollarının programlaması için arayüz tasarımının amacı, NBV algoritmasının işletileceği sistem için bir uygulama alt yapısı oluşturmaktır. Döner tabla etrafında lokal koordinat sisteminin oluşturulma amacı ise algılayıcıya göre tanımlı olan nokta bulutundaki verilerin döner tablaya göre tanımlanmasıyla anlamlı bir 3B nesne modelinin elde edilmesidir. Bu üç ana konu ile ilgili detaylı anlatımlar ilgili başlıklar altında verilmiştir.

6.1 Sistem Mimarisi

Bilinmeyen nesnelerin 3B modellenmesi için oluşturulan yazılımının, bir sistem donanımı üzerinde uygulanması gerekmektedir. 3B modelleme amacıyla daha önce yapılan çalışmalarda farklı donanımlar içeren platformlar sunulmuştur. Suppa vd. (2007) tarafından yapılan çalışmada mekanik ve elektriksel tasarım içeren çok amaçlı bir görüş sistemi oluşturularak birden fazla algılayıcının kombinasyonu kullanılmıştır. Elde edilen bu görüş sistemi bir robot kolu veya bir insansı robota monte edilebilmektedir. Borangiu vd. (2008) tarafından yapılan bir çalışmada ise bir robot kolu, robot kolunun ucuna monte edilmiş bir lazer tarayıcı ve döner tabla kullanılarak esnek bir platform oluşturulmuştur. Bu çalışmalar dışında, Literatür Araştırması ana başlığı altında 3B modelleme için çeşitli NBV algoritmalarının uygulandığı platformlardaki donanımlardan detaylı olarak bahsedilmiştir.

Bilinmeyen nesnelerin 3B modellenmesindeki sistem mimarisi Şekil 6.1'de gösterildiği gibidir.

Sistemde, lazer profil algılayıcıdan yüzey modeli oluşturulacak olan nesnenin yüzeyinden nokta bulutunun elde edilmesinde faydalanılmaktadır. Robot kol ise, lazer profil algılayıcının planlanan konum ve yönelime getirilmesini sağlamaktadır. Döner tabla, robotun erişemediği nesne yüzeylerine algılayıcının taşınması için nesnenin robotun görebileceği gibi döndürülmesini sağlamaktadır. Güç dağıtım panosu, lazer profil algılayıcının, kameraların ve döner tablanın ihtiyaç duyduğu güç beslemesini sağlamak üzere tasarlanmıştır. Hızlı hesaplama ve depolama kaynaklarına sahip iş istasyonu ise,



Şekil 6.1. Sistem Mimarisi .

sistem yazılımının üzerinde koştuğu bilgisayardır. Robot denetleyicisi, kameralar, döner tabla ve lazer profil algılayıcı bu bilgisayara bağlıdır. Kameralar, bu çalışmada kullanılmamaktadır, fakat ileriye yönelik çalışmalarda modelleme süresini kısaltmak üzere nesnenin siluetinin elde edilmesinde kullanılması planlanmaktadır.

6.1.1 Endüstriyel robot kolu

Bu çalışma kapsamında, endüstriyel robot kolunun kullanım amacı çalışma uzayında algılayıcıya hareket yeteneği sağlamaktır. Genel olarak NBV algoritmalarının uygulandığı sistemlerde, bu çalışma kapsamında da olduğu gibi robot kolunun ucuna algılayıcı monte edilmektedir. Fakat, bazı çalışmalarda 3B modellenecek nesnenin robot kolu ile tutularak algılayıcı etrafında veri almak amacıyla dolaştırıldığı da görülmektedir.

Algılayıcıya hareket yeteneği sağlayan endüstriyel robot kolu Kawasaki firması tarafından üretilen RS005L modeline sahip robot koludur. 6 serbestlik derecesine sahip bu robot kolu endüstride; montaj, malzeme taşıma, malzeme kaldırma ve contalama gibi küçük veya orta yükte ağırlığa sahip işlerde kullanılabilmektedir. Ekstra donanımlar dışında 37 kg ağırlığa sahip olan endüstriyel robot kolu; yere, tavana veya duvara monte edilebilme özelliklerine sahiptir. Birinci, ikinci ve üçüncü eksenler için 300°/s, dördüncü ve beşinci eksenler için 460°/s ve son eksen olan altıncı eksen için 740°/s maksimum hıza erişmek

mümkündür. Hareket aralığı olarak bakıldığında ise; birinci eksen ±180°, ikinci eksen +135°'den -80°'ye kadar, üçüncü eksen +118°'den -172°'ye kadar, dördüncü ve altıncı eksen ±360° ve beşinci eksen ±145° hareket aralığına sahiptir. 5 kg yük taşıma kapasitesi, 903 mm yatayda erişebilirlik, 1484 mm dikeyde erişebilirlik, 0.03 mm tekrar edilebilirlik, 9300 mm/s maksimum hıza sahip olma ve Ethernet veya RS232 ile haberleşmenin sağlanabilmesi bu robot kolunun diğer özellikleri olarak söylenebilmektedir. Şekil 6.2'de kullanılan endüstriyel robot kolunun görseli verilmiştir.



Şekil 6.2: Kawasaki RS05L Endüstriyel Robot Kolu

Robot kolu, E70 adı verilen bir denetleyici ile kontrol edilebilmektedir. Robotun hareketi için el terminalini kullanılma veya robota hareketi öğretme yoluna gidilebildiği gibi bu robota özgü olan AS programlama dili kullanılarak da robotun hareketi programlanabilmektedir. Şekil 6.3'te ise robot denetleyicisi gösterilmektedir.



Şekil 6.3: Kawasaki RS05L Endüstriyel Robot Kolu Denetleyicisi

6.1.2 Lazer profil algılayıcı

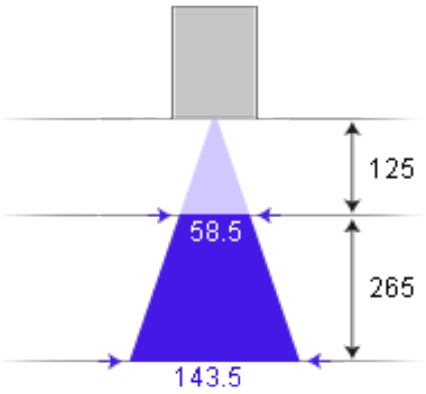
Bu çalışma kapsamında, robot kolunun altıncı eksenine monte halde bulunan lazer profil algılayıcının kullanım amacı 3B modelleme için nesne yüzeyinden veri alımını sağlamaktır. Kullanılan bu algılayıcı Micro-Epsilon firması tarafından üretilmiş olup ScanControl 2900-100 modeline sahiptir.

Lazer profil algılayıcı, üçgenleme prensibine göre çalışmaktadır. Tarama yapılan yüzeye lazer optik sistem ile lazer ışını yollanıp algılayıcı tarafından bu yansımanın algılanmasıyla ölçüm alınan yüzey üzerindeki noktaların 2B olarak x ve z koordinatları elde edilmektedir. Sağlıklı bir ölçüm alınabilmesi için bahsedilen x ve z koordinatlarının belli bir aralıkta olması gerekmektedir. Lazer profil algılayıcı için modelinden de anlaşılacağı üzere z ekseninde 100 mm ölçüm aralığı bulunmaktadır. Bu ölçüm aralığının başlangıç mesafesi 190 mm, bitiş mesafesi ise 290 mm'dir. Başka bir ifadeyle, bu algılayıcı 190 mm'den yakın ve 290 mm'den uzak mesafedeki nesne yüzeylerinden veri alımını gerçekleştirememektedir. Şekil 6.4'te kullanılan bu algılayıcının görseli ve ölçüm koordinatları verilmiştir. Lazer profil algılayıcı için ölçüm aralıkları Şekil 6.5'te verilmiştir.



Şekil 6.4: Micro-Epsilon ScanControl 2900-100 Lazer Profil Algılayıcı ve Ölçüm Koordinatları

Lazer profil algılayıcı ile elde edilen her bir profil verisi üzerinde 160, 320, 640 veya 1280 adet nokta elde edilebilmektedir. 12 μm çözünürlük, yüzde ±0.16 lineerlik ve Ethernet veya RS422 ile haberleşmenin sağlanabilmesi ise bu algılayıcının diğer özellikleri olarak söylenebilmektedir.

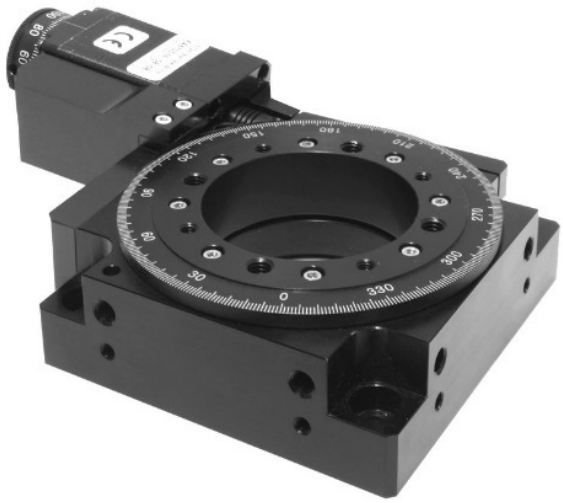


Şekil 6.5: Micro-Epsilon ScanControl 2900-100 Lazer Profil Algılayıcı için Ölçüm Aralıkları

6.1.3 Döner tabla

Bu çalışma kapsamında, döner tablanın kullanım amacı robotun hareket uzayının kısıtlı olması sebebiyle lazer profil algılayıcının nesne yüzeyinden veri alamadığı bölgelere erişimi sağlamaktır. Bu sayede 3B modellenen nesnenin tüm yüzeylerinden veri almak mümkün hale gelmektedir.

Standa firması tarafından üretilen 8MR190-2- 4247 model döner tabla; yüksek kararlılıklı olma, sıfır geri tepme, dakikada maksimum 8 tur dönebilme, 0.01 derece çözünürlüklü, dakikada maksimum 8 tur dönebilme özelliklerine ve 0.65 kg ağırlığa sahip olan bir donanımdır. Hareket yeteneğini step motorlarla sağlayan döner tabla bulunduğu pozisyonu kontrol etmek için enkoder kullanmaktadır. Şekil 6.6'da döner tablanın görseli verilmiştir.



Şekil 6.6. Standa 8MR190-2- 4247 Döner Tabla .

Döner tabla 8SMC4-USB model step motor denetleyicisi ile kontrol edilebilmektedir. Döner tablanın denetleyicisi ise Şekil 6.7'de verilmiştir.



Şekil 6.7: Standa 8MR190-2- 4247 Döner Tabla Sürücüsü

6.1.4 Kamera

Lazer profil algılayıcının, ölçüm mesafesi ve aralığı çok küçüktür. Bundan dolayı, lazer profil algılayıcı hareket etmeden tek seferde nesnenin tümünü kavrayamamaktadır. Bu da nesnenin yaklaşık boyutunun belirlenememesine neden olmaktadır. Böyle bir durumda, sisteme nesneyi üstten ve yandan gören iki kameranın monte edilmesi, nesnenin siluetinin elde edilmesini sağlayacak ve çalıştırılan yönteme hız kazandıracaktır. Bu çalışma kapsamında kamera, sistem mimarisine monte edilmekte olup 3B modelleme için kullanımı ileriye yönelik bir çalışma olarak planlanmıştır.

Kameralar, Alied Vision marka, Manta G-235C modeldir. Bu kameralarda, görüntü sensörü olarak Sony IMX174, 1/1.2" CMOS kullanılmaktadır ve C-Mount, renkli, 1936 x 1216 boyut, 50 fps, 12 bit ADC, IRC filtre özelliklerine sahiptir. Her bir kamera için, Pentax B1218A, 1", 12.5 mm odak uzaklık, F1.8 İris aralığı olan lens kullanılmıştır. Kamera üzerinde bulunan gömülü bilgisayar, görüntü ön işleme ve bazı filtrelemeleri yapabilmektedir. Gig-E Ethernet bağlantısı üzerinden görüntü aktarımı yapılabilmekte ve yapılandırma ayarları alınabilmektedir. Kamera ve lens Şekil 6.8 ile görselleştirilmiştir.



(a) (b)

Şekil 6.8: Kamera ve Lens (a) Kamera (b) Lens

6.2 Endüstriyel Robot Kollarının Programlaması İçin Arayüz Tasarımı

Endüstriyel robot kollarının programlaması için arayüz (API) tasarlamanın amacı, modüler formda sunucu-istemci ilişkisi kurarak eski tarz, zor ve pahalı programlama yerine daha esnek, kullanışlı ve kolay programlama sağlamaktır. API tasarımından bahsetmeden önce robot kolu ile ilgili bazı kavramlardan, API tasarımından önce robot-algılayıcı sistemindeki veri alım sürecinden ve robot kolunun programlanmasındaki zorluklardan bahsetmek API tasarımının neden yapıldığının daha iyi anlaşılması açısından faydalı olacaktır.

Secil vd. (2015) tarafından sunulan hâlihazır robot-algılayıcı sisteminde; robot kolunun programlanma çerçevesi, sunucu programı ve istemci programı olmak üzere iki başlığı içermektedir. Sunucu programı, robot denetleyicisi üzerinde çalışmakta olup Kawasaki robotlarına özgü AS programlama dili ile programlanmaktadır. Sunucu programı herhangi bir bağlantı isteği olup olmadığını kontrol etmek için sürekli dinleme halindedir ve istek olduğu anda TCP bağlantısı kurar. İstemci tarafından tetikleme mesajı geldiğinde, sunucu robotun hareketini başlatmaktadır. Aynı zamanda sunucu, robotun uç noktasının pozisyon ve yönelim bilgisini istemciye yollamaktadır. Buna ek olarak, lazer profil algılayıcı TCP bağlantısı vasıtasıyla aldığı veriyi istemciye yollar. İstemci programı ise, robot kolu ve algılayıcıya Ethernet kablosu ile bağlı olan PC tarafında olup C/C++ dilleri ile kodlanarak lazer profil algılayıcı tarafından taranan nokta bulutunun görselleştirilmesi veya işlenmesi gibi işlemlerde kullanılmaktadır.

Bilindiği üzere, robot kolları farklı amaçlarla birçok firma tarafından üretilmektedir. Her firmanın kendi ürettiği robota özgü bir programlama dili kullandığı bilinmektedir. Bu duruma örnek olarak; Kuka marka robotların KRL, Kawasaki marka robotların AS ve ABB marka robotların RAPID dilini kullandığı bilinmektedir. Bu durumda her marka robot kolu

için farklı programlama dillerini öğrenmek karşılaşılan sorunlardan bir tanesidir. Ayrıca; robot denetleyicisindeki bilgisayar kapasitesinin kısıtlı olması, robot denetleyicideki bilgisayarın kod için hata ayıklamaya imkan vermemesi ve aynı anda çalışan program sayısını kısıtlaması diğer göze çarpan zorluklar olarak söylenebilir (Secil vd., 2016). Bu sorunları aşmak, robotların programlanmasını kolaylaştırmak ve robotun programlanmasındaki zorluklardan kaynaklı zaman kaybını en aza indirmek adına API adı verilen bir Uygulama Programlama Arayüzü tasarımı yapılmıştır. Bu tasarım ile bir PC tarafından C/C++ dilinde robotun programlanması mümkün hale gelmektedir.

API tasarımı için hem sunucu tarafının (robot tarafının) hem de istemci tarafının (PC tarafının) ayrı ayrı kodlanması gerekmiştir. Sunucu tarafı robotun kendine özgü AS programlama dili ile kodlanırken istemci tarafı nesne tabanlı C++ dili ile kodlanmıştır.

API tasarımında sunucu kısmı paralel çalışan Bağlantı Kurma (Connection Establishment, CE), Sistem Arayüzü (System Interface, SI) ve Komut Yorumcusu (Command Interpreter, CI) aşamalarından oluşmaktadır. Bağlantı Kurma aşaması; bağlantı isteklerini dinler, istek geldiği zaman bağlantının kurulmasını sağlar ve son olarak tüm bağlantıların kesilmesini sağlar. Bağlantı Kurma aşaması çalışmadan diğer iki aşamanın çalışması mümkün değildir. Sistem arayüzü aşaması; Ethernet bağlantısı kurulan istemciden komutları almak ve istemciye cevap göndermek üzere görevlendirilmiştir. Komut Yorumcusu aşaması ise, gelen mesajı ayrıştırarak mesajdaki komutu anlamak, bu komut ile istenen görevi yapmak ve istemciye gönderilecek mesajların uygun formata getirilmesini sağlamaktan sorumludur. Ayrıca, aşamaların ve aşamalardaki bazı işlemlerin tetiklenmesini sağlamak ya da aşamanın/işlemin aktif veya pasif durumda olduğunu anlamak amacıyla sinyaller kullanılmıştır.

API tasarımında istemci kısmı PC tarafında nesne tabanlı olan C++ dili ile kodlanmış olup ara yüzün esnek bir yapıya sahip olması adına çok yüzlülük (polymorphism) kullanılmıştır. Bu tasarımda dört soyut sınıf kullanılmıştır. Bu sınıflardan birincisi; robotla bağlantı kurmayı, mesaj almayı ve mesaj göndermeyi içeren bağlantı sınıfıdır. İkinci ve üçüncü sınıflar sırasıyla mesajları uygun formata göre paketlemeyi ve parçalamayı sağlayan sınıflardır. Son sınıf ise robotun yönetilmesi ile ilgili komutları içeren bir sınıftır.

API tasarımı ile ilgili detaylı bilgilere Secil vd. (2016) tarafından yazılan bildiri ile ulaşılabilir.

Yukarıda bahsedilen API tasarımı ile lazer profil algılayıcıdan veri almak ve döner tablayı kontrol etmek mümkün değildir, çünkü bu tasarım sadece robot ve robot ile ilgili

komutları içermektedir. Benzer yazılımların lazer profil algılayıcı ve döner tabla için oluşturulması bilgisayar kontrollü robot kolu, lazer profil algılayıcı ve döner tabladan oluşan bir sistemin oluşturulmasını sağlamıştır. Bu sayede, yazılan test programı ile robot kolunun lazer profil algılayıcıya hareket imkanının vermesinin yanı sıra eş zamanlı olarak nesne yüzeyinden veri toplaması mümkün hale gelmiş olup nesnenin arka kısımlarından veri toplanması amacıyla döner tablanın hareketine olanak sağlanmıştır. Sıradaki aşama, döner tablanın dönmesiyle lazer profil algılayıcıdan alınan verilerin üst üste gelmesi problemini çözen döner tablaya göre lokal bir koordinat ekseninin oluşturulmasıdır.

6.3 Döner Tabla Etrafında Lokal Bir Koordinat Ekseni Oluşturma

Geometrik yöntemlerle döner tabla etrafında lokal bir koordinat ekseni oluşturmanın amacının döner tablayı baz alan lokal bir koordinat ekseni oluşturarak algılayıcıyı baz alan noktaların döner tablayı baz alan noktalara dönüşümünün sağlanmasıdır. Buradaki bu dönüşümün yapılmaması durumunda nesne yüzeyinden alınan noktalar algılayıcıyı baz aldığı için ve döner tabla döndükten sonra alınan bu noktaların döner tablaya göre yeri değişmesine rağmen algılayıcıya göre sabit kalması ölçülen noktaların üst üste binmesine sebep olmaktadır. Bu durumda elde edilen 3B nesne modeli bir anlam ifade etmemektedir.

Geometrik yöntemlerle lokal koordinat ekseni oluşturma konusunda literatürde araştırma yapıldığında sıklıkla robot kalibrasyonu kavramı ile karşılaşılmaktadır. Robot kalibrasyonu bazı geometrik ilişkiler kullanarak robotun kendini pozisyonlaması olarak adlandırılabilir. Aslında bu çalışma kapsamında yapılan, robot kalibrasyonundan ziyade robot kalibrasyonunun alt basamaklarından biri olan geometrik yöntemlerin kullanılmasıdır. Tez kapsamında yapılan bu çalışma geliştirilirse robot kalibrasyonu için de kullanılabilir. Bu sebeplerden dolayı aşağıda bahsedilen literatür çalışmalarında sıklıkla robot kalibrasyonu kavramı geçmektedir.

Liang vd. (1988) insanı baz alarak yapılmış görüş ve hareketin koordinasyonu anlamına gelen robot el-göz koordinasyonunu (robot hand-eye coordination) sağlamak için dinamik, gerçek zamanlı ve uyarlamalı bir yöntem sunmuşlardır. Sunulan bu yöntem; global hareket kontrolü, eldeki nesnenin hassas ölçümü ve ince hareket kontrolü olarak adlandırılan üç aşamadan gerçekleşmektedir. Yöntemde, lazer ışınının nesne üzerine düşürülmesiyle iki kesişim noktası elde edilmesi gerektiği gerçeği kullanılmaktadır. Yöntemin uygulama kısmı küp ver daire üzerinde yapılmıştır. Bu her iki nesne için hizalanma analizi, rotasyon ve öteleme bulunmaktadır. Özellikle daire olan nesnenin (civatanın) hizalanmasında kullanılan çemberlerin kesişimi ile civatanın merkezinin

belirlenmesi önemli bir yaklaşım olup sunulan çalışmada da benzer bir bakış açısı kullanılmaktadır.

Filz vd. (1995) robotu kalibre etmek için robotun uç noktasının pozisyonunun ve yöneliminin önemli bir etken olduğunu belirlemişler ve robotun uç noktasının yönelimini bulmak için sundukları yöntemde birbirine dik üç ayna içeren köşe şeklinde bir küpün geometrisinin yanı sıra lazer ışınları ve görüntü işleme teknikleri kullanmışlardır. Robotun uç noktasının yöneliminin bulunması için sunulan bu yöntem o zamana kadar yapılan çalışmalar içinde en yüksek çözünürlük ve doğruluğa sahip sonuçları vermiş olan yöntemdir.

Bley vd. (2001), endüstriyel robot kolunun ucuna montelenmiş lazer üçgenleme algılayıcısı kullanarak gerçek ortam ve benzetim ortamındaki kalibrasyon farklarını ortadan kaldırmak amacıyla hücre kalibrasyon yöntemi kullanmışlardır. Hücreler için geometrik yaklaşımlar uygulanan bu yöntemle, bahsedilen bu iki ortam arasındaki pozisyon ve yönelim değişimleri bulunabilmektedir.

Lei vd. (2004) yaptıkları çalışmada 6-DOF bir robot kolu için CCD kamera ve kalibrasyon cihazı kullanarak geometri tabanlı bir kalibrasyon yapılmasını sağlamışlardır. Kalibrasyon cihazı olarak sadece üç tane lazer ışını yollayan bir donanım robot kolunun ucuna monte halde bulunmakta olup dünya koordinat sisteminden robot koordinat sistemine geçişte bazı geometrik analizlerden faydalanılmaktadır. Sunulan bu yöntemin bir arada çalışan çoklu robot sistemleri için basit, hızlı ve düşük maliyetli bir yöntem olması yazarlar tarafından vurgulanmaktadır.

Wang vd. (2009) sanal bir nokta kullanarak robot kalibrasyonu yapmışlardır. Sunulan yöntem el-göz kalibrasyonu ve robot kalibrasyonu olmak üzere iki aşamadan oluşmaktadır. Yöntem, sabitlenmiş noktanın mesafe bilgisini ölçmek için robotun uç noktasına monte bir algılayıcı kullanılarak benzetim ortamı için test edilmiştir.

Hu vd. (2012) bir robot kolunun kinematik kalibrasyonu için bir yöntem sunmuşlardır. Robotun uç noktasına bağlı bir lazer işaretçiden düz bir zemine yollanan ışının bir kamera tarafından algılanmasıyla elde edilen çizgi-düzlem keşişim geometrik ilişkisi yöntemde kullanılmıştır. Yöntemin amacı, kinematik parametrelerin yanlış hesabından doğacak olan robotun uç noktasının koordinatının da yanlış elde edilmesinin önüne geçmektir. Sunulan yöntem hem gerçek ortamda hem de benzetim ortamında gerçeklenerek maliyet bakımında düşük olduğu ve herhangi bir robot kolunda uygulanabilecek kadar esnek olduğu vurgulanmıştır.

Nieves vd. (2012) endüstriyel robot kolunun otomatik kalibrasyonu için tamamen yeni bir yöntem geliştirmişledir. Robotun uç noktasına bağlı bir lazer, bilgisayar tabanlı bir ara yüz ve iki adet konuma duyarlı algılayıcıdan oluşan yeni taşınabilir donanım kullanılan bir sistem de bu yöntem için önerilmektedir. Sunulan bu yöntemin robot kalibrasyonu için sadece kolay ve hızlı olmadığı aynı zamanda etkili ve makul olduğu yazarlar tarafından iddia edilen bir gerçektir.

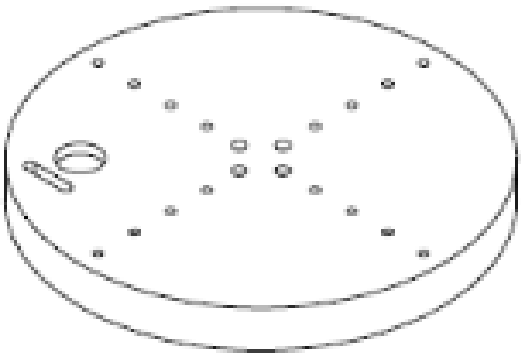
Huang vd. (2015) de robot el-göz kalibrasyonunu sağlamak için yöntem önermişlerdir. Onlar da diğer sistemlere benzer olarak robot kolu ve robot kolunun ucuna monte lazer işaretçi kullanmışlardır. Sunulan metot ile rotasyon matrisi ve öteleme vektörü yüksek bir hassasiyet ile bulunmuştur.

Wagner vd. (2015) robot kolunun ucuna monte 2B lazer profil algılayıcı kullanan bir sistem için yeni bir kalibrasyon yöntemi önermişlerdir. Önerilen yöntem bir pini referans noktası olarak kullanmaktadır. Algılayıcı ve robotun çıkıntısı arasındaki rotasyon ve öteleme ile ilgili ilişki bu pinin kullanılmasıyla elde edilmektedir. Burada, otomatik hale getirilmiş bir kendi kalibrasyonunu yapma işlemi gerçekleşmektedir. Sunulan yöntemin testleri sırasında iki endüstriyel robot kolu ve lazer profil algılayıcı kullanılmaktadır. Birkaç nesnenin lazer ile taraması yapılarak yüksek çözünürlükte sonuçlar elde edilmektedir.

Liang vd. (1988) ve Wagner vd. (2015) tarafından yapılan çalışmalar tez kapsamında geliştirilen yöntem için önem arz eden iki çalışmadır. Geliştirilen yöntemin Liang vd. (1988) tarafından yapılan çalışma ile benzerliği benzer geometrik yaklaşımlar ve varsayımlar kullanılmasıdır. Bunlar; lazer ışınının nesne üzerine düşürülmesiyle iki kesişim noktası elde edilmesi gerektiği gerçeği ve özellikle çember şekli üzerinde uygulanan geometrik yaklaşımlar olarak sıralanabilir. Geliştirilen yöntemin Wagner vd. (2015) tarafından yapılan çalışma ile benzerlikleri ise, ilk olarak benzer lazer profil algılayıcıların kullanılması (Wagner vd. (2015) ScanControl 2600-100 kullanılırken, tez kapsamında ScanControl 2900-100 kullanılmaktadır), ikinci olarak referans örüntülerin kullanılması (Wagner vd. (2015) bir pini sabitlenmiş referans noktası olarak kullanırken tez kapsamında bir çember ve kenarları yuvarlatılmış bir dikdörtgen referans olarak kullanılmaktadır) ve son olarak yöntemin test edilmesi aşamasının bir nesnenin taranması ile yapılması olarak sıralanabilmektedir.

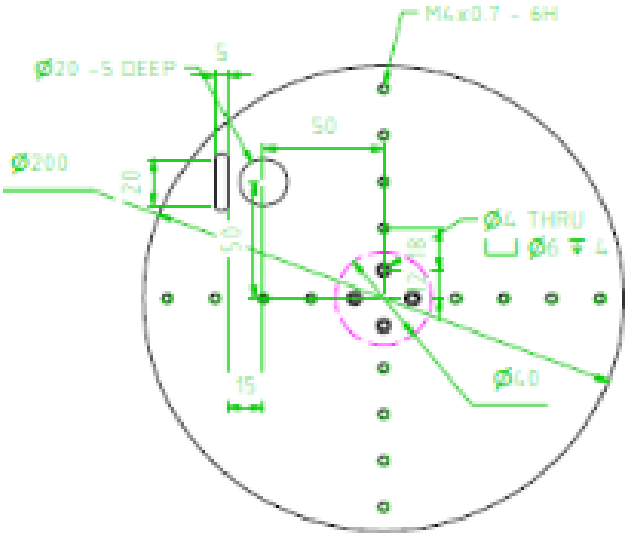
Çalışma kapsamında uygulanan yöntem, özel tasarlanmış bir örüntü üzerinde lazer profil algılayıcı ile yapılan ölçüm değerlerine ve robotun ölçüm gerçekleştiği andaki uç noktasının robotun tabanına göre pozisyon ve yönelim değerlerine ihtiyaç duymaktadır.

Bu yöntemin uygulanmasında, derinliğe sahip bir daire ve dikdörtgenden oluşan örüntü kullanılmaktadır. Örüntüdeki daire; pozisyonu, dikdörtgen ise yönelimi bulmak için kullanılmaktadır. Şekil 6.9'da bu örüntünün çizim hali görselleştirilmiştir.



Şekil 6.9: Döner Tabla Etrafında Lokal Koordinat Ekseni Oluşturmak İçin Kullanılan Örüntü

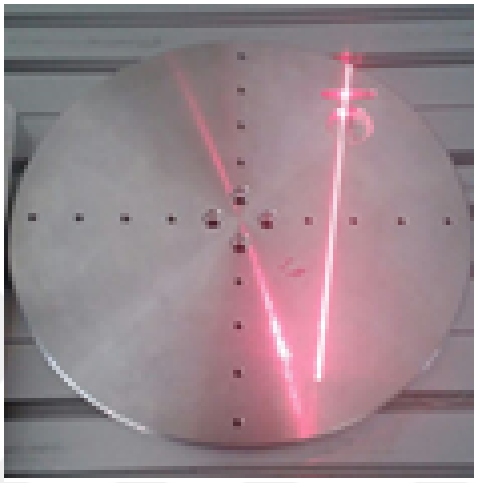
Şekil 6.10'da kullanılan örüntünün CAD çizimi görselleştirilmiştir.



Şekil 6.10: Döner Tabla Etrafında Lokal Koordinat Ekseni Oluşturmak İçin Kullanılan Örüntünün CAD çizimi

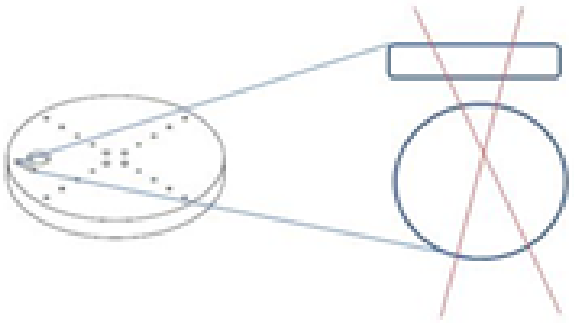
Yöntemin uygulanmasında, öncelikli olarak endüstriyel robot kolunun ucuna monte lazer profil algılayıcı kullanılarak, iki farklı doğrultuda profil verisi alınmaktadır. Şekil 6.11'de, döner tabla üzerindeki örüntünün üzerine algılayıcı ışınının düşmüş hali

görülmektedir. Bahsi geçen şekilde yansımadan dolayı iki profil varmış gibi dursa da örüntü üzerinde bir profil ölçümü görselleştirilmiştir. Her profil verisi için 1280 adet nokta, lazer profil algılayıcı üzerindeki koordinat sistemine göre tanımlı olarak kaydedilmektedir. Lazer algılayıcı üzerindeki koordinat sistemine göre tanımlı bu noktalar, ölçüm anında robottan temin edilen robot eklem açıları da kullanılarak, robot tabanlı koordinat sistemine göre dönüştürülmektedir. Parça üzerinde, örüntü ölçekli olarak yerleştiğinden bir dönüşümle parça merkezinin koordinatı da hesaplanabilir.



Şekil 6.11: Örüntü Üzerine Işın Düşürülmesi

Yöntem iki aşamada gerçekleştirilmektedir. Öncelikle örüntüdeki daire merkezinin koordinatı bulunmaktadır. Daha sonra, daire yanındaki dikdörtgen kullanılarak örüntünün yönelimi bulunmaktadır. Yukarıda da bahsedildiği gibi iki adet farklı doğrultudaki profil verisi Şekil 6.12'de görüldüğü gibi alınmaktadır.

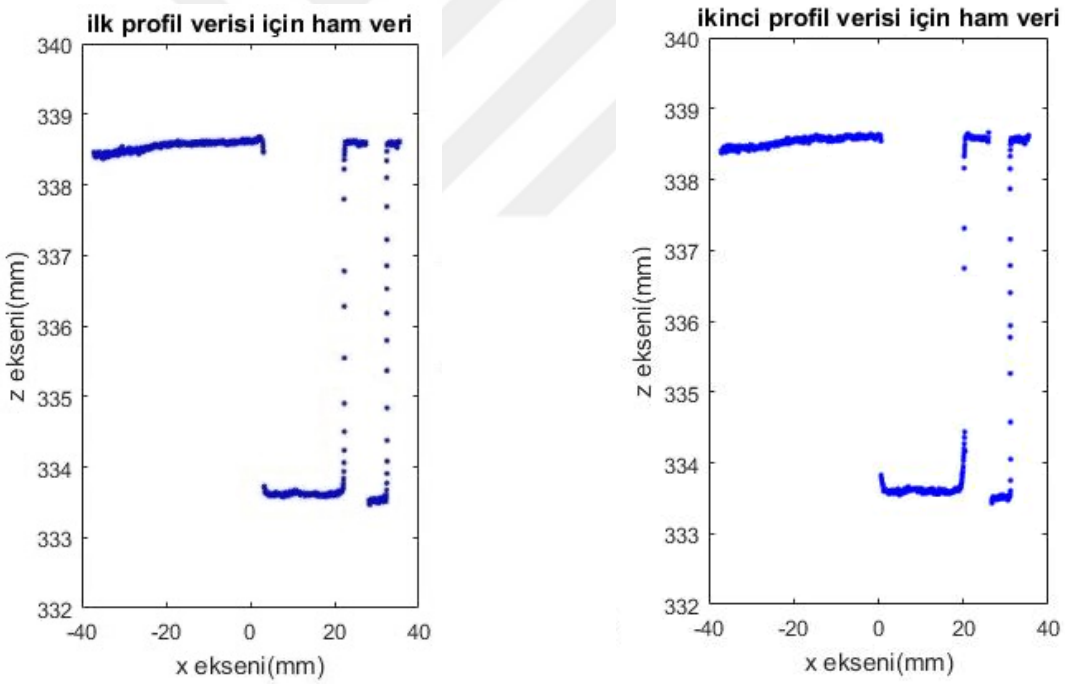


Şekil 6.12: Örüntü Üzerine Işın Düşürülmesi

İki profil verisinin alınmasından sonra döner tabla etrafında lokal koordinat ekseni oluşturma işleminin devamı aşamalar halinde anlatılmıştır.

Adım 1: Bu adımda profil verilerinin *x* ve *z* koordinatlarının doğruluğunun test edilmiştir. Profil verilerinin, algılayıcı koordinat sistemine göre *y* bileşenleri sıfırdır. *X* bileşenleri, profil doğrultusundadır. *Z* bileşeni ise, noktanın algılayıcıya dik uzaklık doğrultusundadır. Elde edilen ham veri Matlab ortamında görselleştirilmiştir.

Lazer profil verilerinin dört yerde büyük bir değişime uğraması beklenmektedir. Bu değişim, *z* ekseninde olacaktır; çünkü metal nesnenin üzerindeki dairenin ve dikdörtgenin *z* bileşenleri genel nesne yüzeyinin *z* koordinatından 5 mm daha aşağıdadır. Onun dışında, lazer profil verisinden gelen noktaların *z* koordinatlarında sapmalar olsa da büyük değerlerde olmayacaktır. Şekil 6.13'te çapraz şekilde alınan iki tarama için ham veriler gösterilmiştir.



(a) (b)

Şekil 6.13: İki Profil İçin Ham Veri Gösterimi (a) İlk Profil Verisi İçin (b) İkinci Profil Verisi İçin

Geometrik örüntü yüzeyindeki keskin derinlik değişimlerine karşılık gelen kenarları bulmak için noktaların alt noktalar ve üst noktalar olarak sınıflandırılması ihtiyacı doğmuştur. Bu yüzden, bu noktaların sınıflandırılması için ikinci aşamada bahsedilen RANSAC algoritması kullanılmıştır.

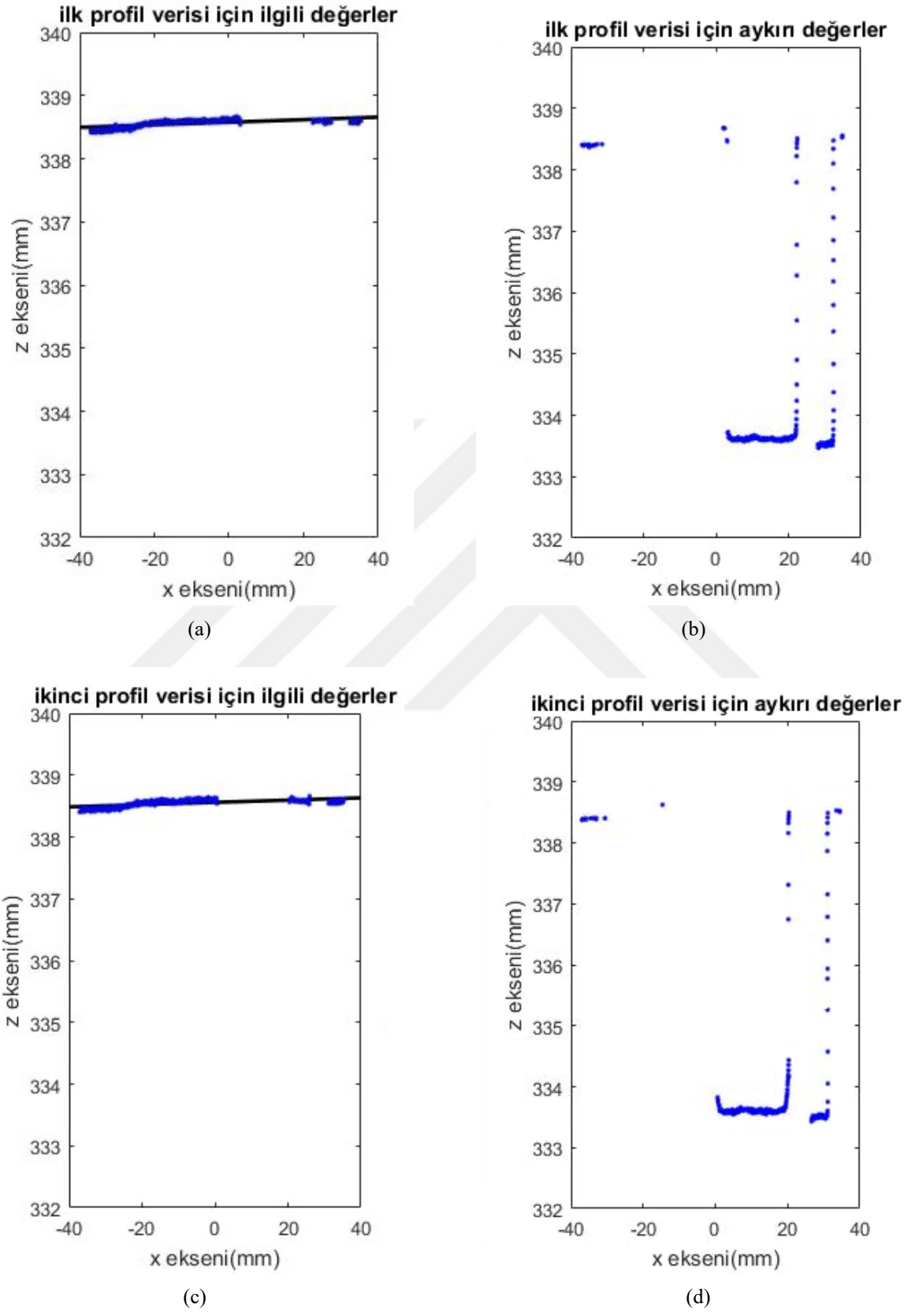
Adım 2: Bu aşama RANSAC algoritması kullanılarak noktaların alt ve üst noktalar olarak sınıflandırılması yapılmıştır. RANSAC (Random sample consensus) algoritması aykırı değerler (outlier) içeren bir veri kümesinin matematiksel model parametrelerini ölçmek için kullanılan bir yöntemdir. 1981 yılında Fischler ve Bolles tarafından yayınlanan bu yöntem deterministik olmayan bir algoritma olup rassal olmasından kaynaklı olasılıksal sonuçlar üretir. En basit bir örnek, iki boyutlu veri kümesine bir çizgi oturtmaktır (Anonim, 2017b).

RANSAC algoritması burada noktaları sınıflandırmak amacıyla kullanılmıştır. Noktaları sınıflandırmanın amacı bir adet ana sınıf belirleyerek geri kalan noktaları elemektir. Yani, ilgi alanını (ROI) elde ederek işlemleri bu noktalar üzerinden yapmaktır. RANSAC algoritması ham veri üzerinden çalışmaktadır. Algoritma, ilgili ve aykırı noktaları (inliers and outliers) ayırarak istenilen noktalar üzerinde işlem yapma olanağı tanımaktadır. Daha doğru ve güvenilir sonuçlar elde edilmesi için RANSAC algoritması arka arkaya iki kere uygulanmıştır. Her iki profil verisi için, RANSAC algoritmasının art arda uygulanmasından sonra elde edilen sonuçlar Şekil 6.14'te gösterilmektedir. Aynı şekil üzerinde oturtulan çizgi de gösterilmektedir.

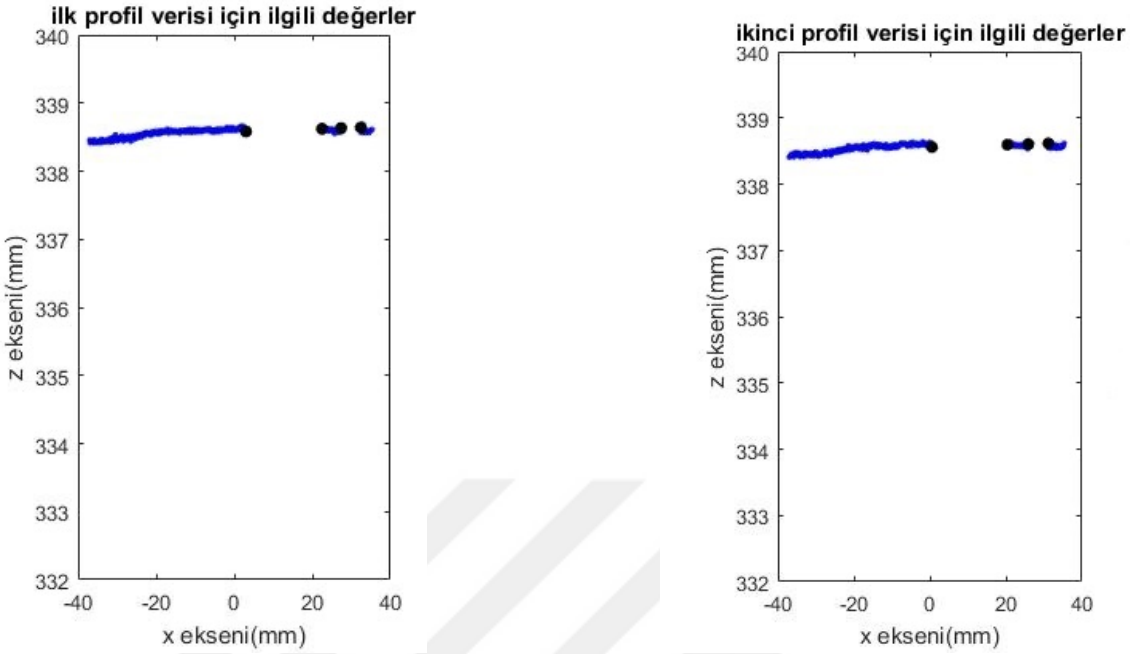
Adım 3: Bu aşamada, ilk adımda bahsedilen geometrik örüntü yüzeyindeki keskin derinlik değişimlerine karşılık gelen kenarlar bulunmaktadır. Her profil verisi, iki tane dairenin kenar noktası ve iki tane dikdörtgenin kenar noktası olmak üzere dört adet kenar noktası içermektedir. Buradaki kenar noktaları ilgili noktaların x koordinatlarındaki sıçramanın kontrol edilmesi ile bulunmaktadır. Ardışık iki nokta arası sıçrama eğer önceden tanımlı bir eşik değerinden büyük ise o noktanın kenar olduğu kabul edilmektedir. Daha sonra bulunan kenar noktaların RANSAC algoritması ile bulunan çizgiye izdüşümü gürültünün etkisini azaltmak amacıyla alınmaktadır. Her iki ölçüm için izdüşümü alınmış noktalar şekil 6.15'te gösterilmektedir.

Adım 4: Bu adımda izdüşümü alınan kenar noktaların 2B lazer profil algılayıcı koordinat sisteminden 3B robot koordinat sistemine dönüşümü yapılmaktadır. Bu dönüşüm halihazırda referans verilen tez çalışması ve bildiri kullanılmıştır (Secil, 2015), (Seçil vd., 2014). Burada, rotasyon matrisi, ZYZ Euler açılarını kullanmıştır. Dönüşüm matrisinde, ölçüm alındığı andaki robot eklem açıları kullanılmıştır. Her iki profil verisi için elde edilen daire ve dikdörtgeni kesen kenar noktaları, robotun koordinat sistemine göre dönüştürülmüş ve 3B noktalar Şekil 6.16'daki gibi aynı şekil görselleştirilmiştir.

Adım 5: Bu aşamada trilaterasyon kullanarak dairenin merkezini bulunmuştur. Her iki profil için toplamda 8 köşe noktası (İkişer nokta ilk ve ikinci profilde daire için, ikişer nokta ilk ve ikinci profilde dikdörtgen için) bulunmuştur. Birinci profilde daireyi kesen iki

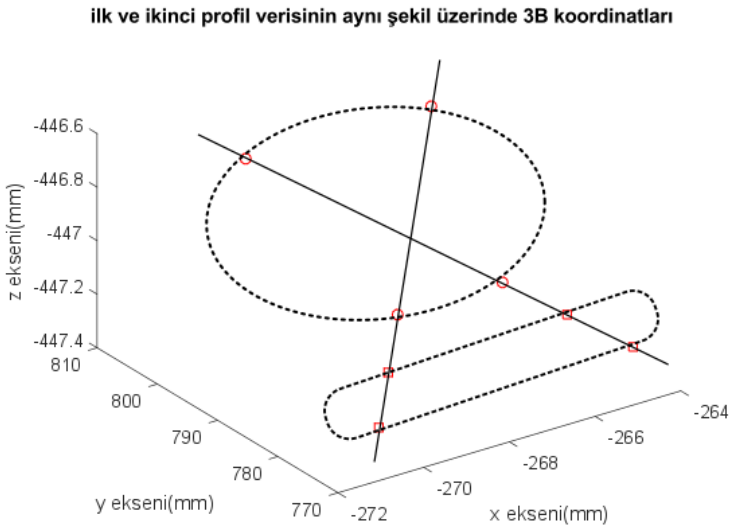


Şekil 6.14: İki Profil İçin RANSAC Algoritmasının Sonuçları (a) İlk Profil Verisi İçin (b) İkinci Profil Verisi İçin



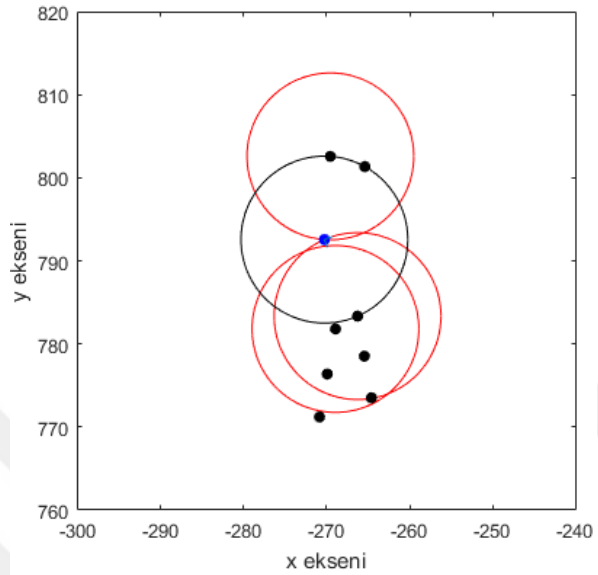
(a) (b)

Şekil 6.15: Kenar Noktaların RANSAC İle Bulunan Çizgiye İzdüşümünün Alınması (a) İlk Profil Verisi İçin (b) İkinci Profil Verisi İçin



Şekil 6.16: İlk ve İkinci Profil Verisinin Aynı Şekil Üzerinde Gösterimi

nokta ile ikinci profilde daireyi kesen iki nokta kullanılarak, yani trilaterasyon yöntemi ile dairenin merkezi hesaplanmıştır. Şekil 6.17'de 3B noktalar kullanılarak elde edilen dairenin merkezi ve nesnenin merkezi 2B olarak gösterilmiştir.

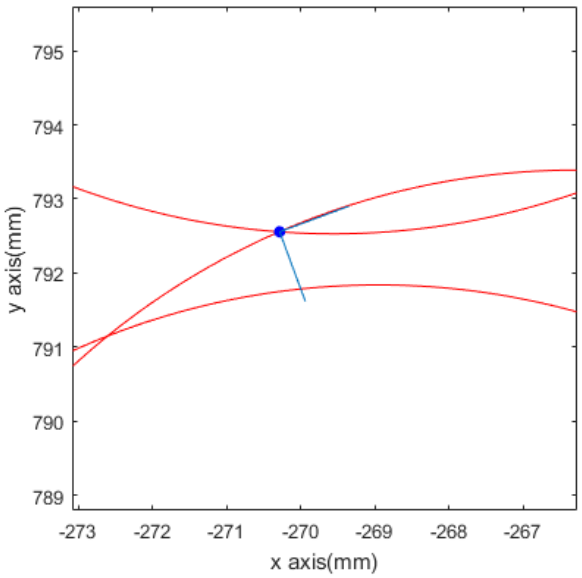


Şekil 6.17: Belirlenen Örüntüdeki Dairenin Merkez ve Döner Tabla Merkez Koordinatlarının Gösterimi

Şekil 6.17'deki kırmızı daireler, tespit edilen köşe noktaları merkezli 20 mm çaplı daireleri; siyah daire ise kırmızı dairelerin kesişim noktası merkezli 20 mm çaplı daireyi göstermektedir. Bu hesaplamalarla elde edilen siyah daire, örüntüdeki dairedir.

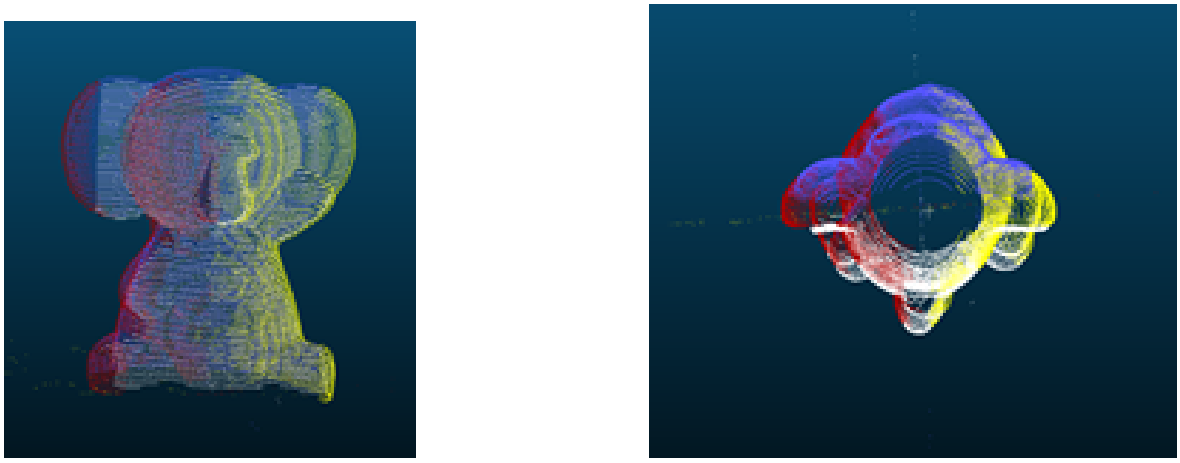
Adım 6: Bu adımda hesaplanan örüntü dairesinin merkezinin örüntüdeki dikdörtgen şekline iz düşümünün alınmasıyla lokal koordinat eksenini belirlenmiştir. Daire merkezinin dikdörtgenleri kesen noktaların oluşturduğu çizgi üzerine izdüşümü alınması ile oluşan vektör, lokal eksende x koordinat vektörünü vermektedir. 90 derece dik olan ve sağ el kuralını sağlayan vektör y koordinat vektörünü vermektedir. Aynı zamanda, bu y vektörü, dikdörtgeni kesen iki farklı profilin noktalarından geçen çizgi doğrultusundadır. Z eksenini bulmak için ise x ve y eksenlerinin vektörel çarpımı kullanılmaktadır. Şekil 6.18'te dikdörtgeni kesen iki farklı profilin noktalarından geçen çizgi ve daire merkezinin bu çizgi üzerindeki izdüşümü kolaylık açısından 3 boyutta yapılmasına rağmen 2 boyutlu olarak gösterilmiştir.

Bahsedilen adımlar ile döner tabla etrafında lokal bir koordinat ekseni belirlenerek yöntem bir nesnenin NBV algoritması kullanılmadan 3B modellenmesi ile test edilmiştir. Döner tablanın 0, 90, 180 ve 270 derecelik dönüş açıları için robot kolunun, dolayısıyla



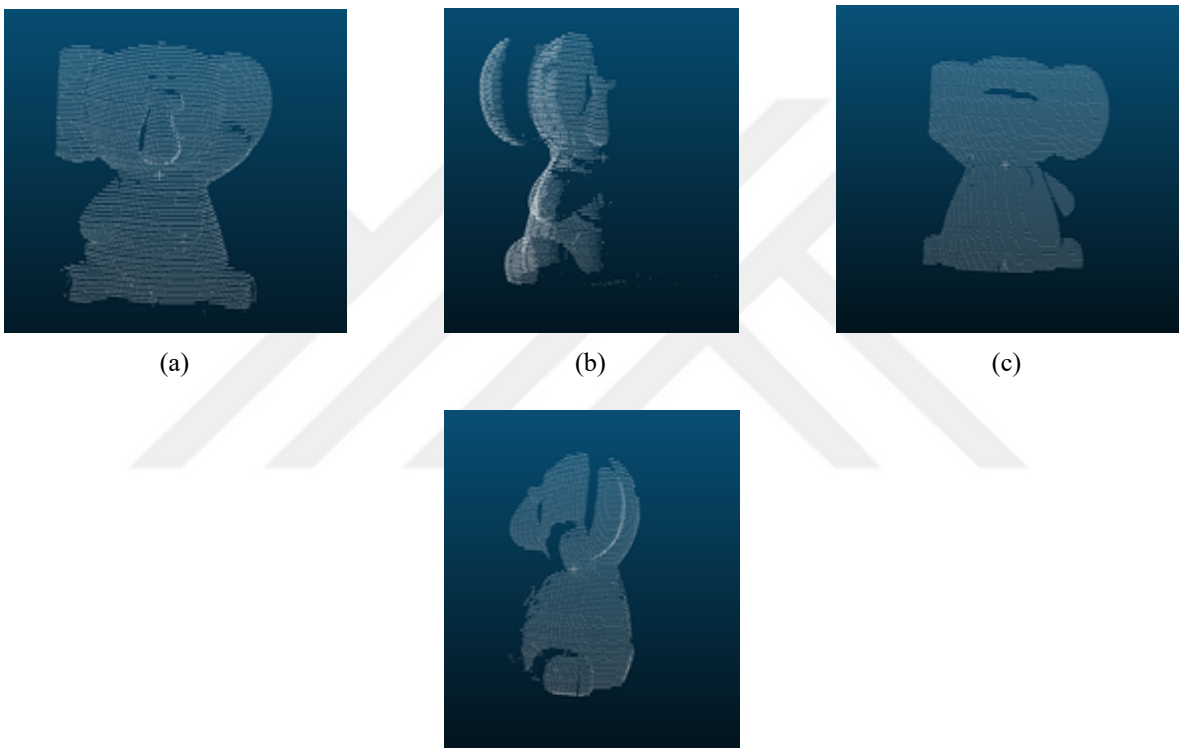
Şekil 6.18: Örüntüdeki Daire Merkezinin Birim Vektörleri

lazer profil algılayıcının lineer olarak yukarı-aşağı hareketleri ile silindirik bir tarama yapılmıştır. Döner tablanın bahsedilen açı pozisyonları için alınan nokta bulutları Şekil 6.20'de görselleştirilmiştir. Şekil 6.19'de ise her taraması farklı renkle gösterilen tamamlanmış model üstten ve yandan görselleştirilmiştir.



(a) (b)

Şekil 6.19: Tamamlanmış 3B model: (a) Tamamlanmış Modelin Önden Görünümü (b) Tamamlanmış Modelin Üstten Görünümü



(a) (b) (c)

(d)

Şekil 6.20: Her Tarama İçin Elde Edilen Nokta Bulutları: (a) İlk Tarama İçin (b) İkinci Tarama İçin (c) Üçüncü Tarama İçin (d) Dördüncü Tarama İçin

7. BULGULAR VE TARTIŞMA

Bu çalışma kapsamında, bilinmeyen nesnelerin endüstriyel robot kolu, lazer profil algılayıcı ve döner tabla kullanılarak 3B modellenmesi gerçeklenmiş ve birisi endüstriyel parça diğer ikisi gündelik nesne olmak üzere üç nesne üzerinde test edilmiştir. Bu nesneler, literatürde taranan çalışmalara bakılarak seçilmiştir. Çalışmalarda, endüstriyel nesnelerin yanı sıra evlerde kullanılan gündelik nesnelerin de modellendiği gözlemlenmiştir. Endüstriyel parça ve çanak nesnesi bu amaçla seçilmiştir. Fil nesnesinde kullanım amacı, yumuşak geçişlerin olmasından kaynaklı yüzey modelleme yönteminin daha verimli çalışacağı düşüncesidir. Bu nesnelerin yüzey özelliklerinden bahsetmek gerekirse; fil nesnesi ve çanak porselen, endüstriyel parça ise metalden oluşmaktadır. Fil nesnesinin dışı bir boya ile kaplıdır, bu yüzden lazer profil algılayıcının yolladığı lazer ışınları dağılmadığı için düzgün ölçüm yapmak mümkün olmuştur. Fakat, endüstriyel parça ve çanak nesnesinde, lazer profil algılayıcının ölçüm almak amacıyla yolladığı lazer ışınları nesne yüzeyindeki yansımadan dolayı dağılmış, doğru ölçüm almayı önlemiştir. Bu sorunun önüne geçmek adına matlaştırıcı sprey adı verilen ve nesne yüzündeki yansımayı önlemek için nesne yüzeyinde bir çok ince katman oluşturan kimyasal bir madde kullanılmıştır. Bu şekilde nesne yüzeyinden alınan verilerden elde edilen test sonuçları ve nesnelere ait elde edilen 3B nokta bulut modelleri ilgili bölümde detaylandırılmıştır. 3B modelleme için kullanılan yöntemden de detaylı olarak ilgili bölümlerde bahsedilmiştir. Bu bölümde ise, uygulanan yöntemin artı yönlerinden bahsedildikten sonra çalışmanın eksiklikleri ve çalışmaya eklenebilecekler ile ilgili detaylar verilecektir.

Uygulanan bu yöntemin literatürde geçen bir çok yönteme göre üstünlüğü vardır. İlk olarak, yöntemin üçgen ağ yapısı kullanması ve sonraki NBV'nin belirlenmesinde üçgen ağ yapısının analiz edilmesidir. Liteartürde kullanılan çalışmalar genellikle silindir veya küre üzerinden bakış açılarını belirlemektedir. Bu da, belli noktalar dışında bakış noktalarının kullanılmayacağı anlamına gelmektedir. Fakat bu sistemde, üçgen örgü yapısını baz alarak hesaplanan herhangi bir bakış açısı kullanılabilmektedir.

İkinci olarak, işlemin gerçek zamanlı olarak çalışmasından bahsedilebilir. Noktaların alınması, alınan noktalar üzerinde yapılan hesaplamalar, robot ve döner tablanın hareket etmesi ve yeni nokta bulutunun alınması döngüsünün tamamı gerçek zamanlı olarak işletilmektedir. Yani bir simülasyon ya da çevrim dışı bir işlem gerçekleşmemektedir.

Üçüncü olarak, taranan nesne hakkında herhangi bir kısıtlama olmaması söylenebilmektedir. Döner tablanın taşıyamayacağı ve robotun hareket uzayını tamamen dolduracak çok büyük nesneler dışında bütün nesnelerin 3B modeli elde edilebilir. Sadece nesnenin şekli taramanın kalitesini etkileyebilir.

Dördüncü olarak, yüzey kalitesinin NBS seçiminde ve sonlandırma kriteri olarak kulanılması söylenebilir. Bu sayede elde edilen modelin belli bir kaliteye ulaştığı garanti edilmektedir; çünkü maksimum tarama sayısı veya alınan nokta bulutunun önceki nokta bulutu ile çakışma oranına bakmayarak sadece kalite kriterlerini ölçüt almaktadır.

Buraya kadar bahsedilen avantajların hepsi yöntemin avantajları olarak söylenebilir. Donanım sisteminin avantajları olarak bakıldığında ise, endüstriyel robot kolu otonomluğu sağlamakta, lazer profil algılayıcı profil başına 1280 nokta alarak yüksek hassasiyette model elde edilmesini sağlamakta ve döner tabla ise endüstriyel robot kolunun çalışma uzayı dışında kalan NBV noktalarından tarama alabilmek için gerekli hareket manevrasını sağlamaktadır.

Bu çalışmaya otonomluğu arttıracak ve elde edilen model kalitesinin daha da iyileştirilmesini sağlayacak eklemeler yapılabilir. İlk olarak, sistem donanımında kullanılan fakat 3B modelleme için kullanılmayan kameranın sisteme dahil edilmesinden bahsedilebilir. Bu çalışmada, nesnenin yaklaşık pozisyon ve boyutunun bilindiği varsayılmıştır; yani ilk tarama kullanıcı tarafından elle alınmaktadır. Nesnenin üstünden ve yanından bakan iki kameranın 3B modellemeye dahil edilmesiyle, nesnenin yaklaşık boyutu ve konumu da otonom olarak belirlenebilir.

İkinci olarak, 3B Yüzey Modellemede Sınıflandırma başlığında geçen sınıflandırma yönteminin 3B nesne modelleme yöntemine eklenmesi olabilir. Bu sayede elde edilen son nokta bulutunun düz bölümlerinin tekrar taranmaya ihtiyacı olmadığı varsayımı ile kenar ve köşe gibi girinti çıkıntısı olan bölgelerin tekrar taranması ile kalitenin arttırılması sağlanabilir. Sınıflandırmada başka bir yaklaşım alanı ise, sınıflandırmanın bir kalite kriteri olarak eklenmesi ve her taramada elde edilen nokta bulutunun sınıflandırılmasıyla bilgi kazancına bir ağırlık verilmesi olabilir.

Üçüncü olarak, yukarıdaki sınıflandırmadan farklı olarak nokta bulutunun değil nesnelerin sınıflandırılması yapılabilir. Örneğin, endüstriyel parçalarda keskin değişimlerin olduğu ve günlük kullanılan nesnelerin daha yumuşak geçişlere sahip olduğu varsayımı altında, taranan nesnenin hangi sınıfa ait olduğu bilindiği takdirde taramanın farklı bir şekilde yapılması sağlanabilir. Böylece, farklı bakış noktalarından veri alınabilir ve 3B model kalitesi artabilir.

Dördüncü olarak, nesneye ait genel bir eğrilik yüzdesi çıkarıp işlemler sırasında adaptif parametre kullanımı olabilir. Örneğin, bütün nesnelerin modellenmesi için sınırların sınıflandırılması esnasında ardışık noktaların oluşturduğu vektör zincirindeki vektörler arası açı değeri 45° olarak kullanılmaktadır. Bu da, girinti çıkıntısı fazla olan nesneler için sınıflandırılmamış noktaların fazlalığı anlamına gelmektedir; çünkü noktalar arası değişim 45°'den fazla olabilmektedir. Belki de böyle bir durumda 45° yerine 50° kullanmak sınırların daha doğru sınıflandırılmasını sağlayabilmektedir. Nitekim, fil nesnesi üzerinde açı değeri 50° olarak denenmiş, fil nesnesinin sol tarafında sınıflandırılmaya dahil edilmeyen bir sınır bulunmuştur. Bu aslında, bütün süreci etkileyebilir bir adım olabilir. Fakat, eğrilik yüzdesi gibi bir bilgi mevcut olmadığı için nesne üzerinde sınırların yanlış bulunmaması adına açı değeri 45° olarak bırakılmıştır.

Beşinci olarak, sistem yazılımına delik taraması eklenebilir. Delik taraması, küçük bir oran da olsa kesinlikle yüzey kalitesini arttıracaktır. 3B model belli bir yüzey kalitesine ulaştığında, deliğin/deliklerin nesne üzerindeki oranı veya alanı bulunabilir. Bu sayede, bir kere veya delik sayısı/oranı/alanı çok fazla ise birden fazla kez delik taraması yapılarak yüzey kalitesi belli bir oranda arttırılabilir.

Altıncı olarak, NBV seçiminde farklı ölçütlerin kullanılması ileriye dönük bir çalışma olarak eklenebilir. Bu çalışma kapsamında, sınıflandırılmış sınırlardan en çok nokta verisine sahip olanı seçilerek NBV hesabı bu sınır üzerinden yapılmıştır. Bu seçim ölçütü yerine, sınıflandırılmış sınırlardan noktalar arası toplam Öklid uzaklığı en çok olan kullanılabilir. Başka bir NBV seçim ölçütü ise, bilinmeyen bölgedeki bilgi kazancını maksimize edecek bakışın kullanılması olabilir. Bu şekilde farklı seçim kriterleri eklenebilir.

Yedinci olarak, taramanın verimliliğini test etmek amacıyla otonom tarama ve elle tarama bazı parametreler cinsinden karşılaştırılabilir. Bu parametreler modelin tamamlanma süresi, tamamlanma yüzdesi, delik yüzdesi/oranı, en iyi bakış noktalarının sayısı, yüzey modelinin kalitesi olabilir.

8. SONUÇ VE ÖNERİLER

Bu tez çalışması kapsamında endüstriyel robot kolu, lazer profil algılayıcı ve döner tabla kullanımı ile bilinmeyen nesnelerin otonom 3B modellerinin çıkarılması amaçlanmıştır. Endüstriyel robot kolunun uç noktasına monte halde bulunan lazer profil algılayıcı ile kullanıcı tarafından nesne üzerinden ilk veri alındıktan sonra sonraki en iyi görüş noktası algoritmasının çıktı olarak ürettiği pozisyon ve yönelim bilgisine göre endüstriyel robot kolu lazer profil algılayıcıyı hareket ettirerek veri alma işlemi gerçekleşmektedir. Eğer bu pozisyon ve yönelim bilgisi, endüstriyel robot kolu tarafından erişilemeyecek bir bölgede ise döner tabla, hareketi ile o noktayı robot kolunun hareket uzayına taşımaktadır. Böylece 3B modellenen nesnelerin bütün yüzeyinden veri almak mümkün hale gelmiştir. Bahsedilen bu aşamalar model tamamlanana kadar devam etmiş ve sonuç olarak nesnelerin 3B yüzey modeli elde edilmiştir.

Bu tez çalışması ile; endüstriyel robot kolu, lazer profil algılayıcı, döner tabla ve iki adet kameradan oluşan donanımsal bir sistem kurulmuştur. Ayrıca, modüler olarak bir yazılım sistemi de oluşturulmuştur. İleriye yönelik yapılacak çalışmalara göre var olan donanımsal ve yazılımsal sistem bozulmadan yeni donanımın ve yazılımın entegrasyon işlemi gerçekleştirilebilmektedir.

Bu çalışmanın literatürdeki çalışmalara göre bazı üstünlükleri bulunmaktadır. Bunlardan biri NBV'nin belirlenmesinde silindirik veya küresel bakış açısı yerine üçgen ağ yapısının analizi ile elde bakış açısının kullanılmasıdır. Bu sayede, kısıtlı bakış açısı kümesi yerine daha esnek bir yapı kullanılmış, bir bakış ile daha fazla veri alımı sağlanmıştır. Diğer bir üstünlük NBV pozisyon ve yönelim hesabında tahmin edilen yüzey eğilim eğrisinin kullanımıdır; çünkü yüzey eğilim eğrisi nesnenin devamını tahmin ederek bilinmeyen bölgelerden veri edimini sağlamaktadır. Başka bi üstünlük döner tabla kullanımıdır. Endüstriyel robot kolunun kısıtlı hareket uzayı bu sayede genişleyerek nesnenin bütün yüzeyinden veri almak mümkün hale gelmiştir. Nesnenin şekli hakkında bir kısıt olmaması ve tüm çalışmanın simülasyon veya çevrim dışı olarak değil gerçek zamanlı olarak işletilmesi diğer üstünlükler olarak söylenebilmektedir.

Bu çalışma, bilinmeyen nesnelerin 3B modellenmesinde temel bir çalışma olduğu için ileriye yönelik çalışmalarda ise dikkate alınabilecek birçok husus bulunmaktadır. Bunlardan biri, sisteme entegre fakat kullanılmayan iki adet kameranın sisteme dahil edilmesiyle nesnenin yaklaşık boyut ve konumu elde etmek olabilir. Bu sayede, kullanıcı

tarafından elle yapılan ilk veri alımı da otonom hale gelmektedir. Başka bir ileriye yönelik çalışma, nesne yüzeyinden alınan nokta bulutularının sınıflandırılmasıyla bazı bölgelerin tekrar taranma ihtiyacının olup olmadığı analizinin yapılmasıdır. Başka bir sınıflandırma yaklaşımı ise, modellenecek nesnelerin keskin geçişli nesneler ve yumuşak geçişli nesneler olarak sınıflandırılması olabilir. Bu farklı sınıflara dair farklı NBV yaklaşımları sunulabilir. Başka bir ileriye yönelik çalışma ise, şu anki mevcut NBV algoritmasının etkinkiliği arttırmak için nesnenin eğrilik durumuna göre değişken parametrelerin kullanımı olabilir. Bu sayede, farklı karakteristik özelliklere sahip bütün nesneler ya da bütün sınırlar için aynı parametrelerin kullanımından kaynaklı sorunlar önlenebilir. Ayrıca sisteme delik taramasının eklenmesi yüzey kalitesini arttıracak ileriye dönelik bir çalışma olabilir. Ek olarak, NBV seçiminin farklı kriterlere göre yapılması da ileriye dönük çalışmalardan biri olup 3B nesne modellemenin performansını etkileyecek faktörlerden biridir. Son olarak ise, bilinmeyen nesnelerin 3B modellenmesinin elle taramaya üstünlüklerini göstermek amacıyla modelin tamamlanma süresi, tamamlanma yüzdesi, delik yüzdesi/oranı, en iyi bakış noktalarının sayısı, yüzey modelinin kalitesi gibi parametreler cinsinden karşılaştırmalar da yapılabilir.

KAYNAKLAR DİZİNİ

Aleotti, J., D. L. Rizzini, R. Monica ve S. Caselli (2014). "Global registration of mid-range 3D observations and short range next best views". İn: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3668–3675. DOI: 10.1109/IROS.2014.6943077.

Anonim (2017a). Polygon Mesh. http://www.wikiwand.com/en/Polygon_mesh/. erişim tarihi 10-Temmuz-2017.

— (2017b). Random Sample Consensus. http://www.wikiwand.com/en/Polygon_mesh/. erişim tarihi 10-Temmuz-2017.

— (2017c). Region Growing. http://pointclouds.org/documentation/tutorials/region_growing_segmentation.php/. erişim tarihi 10-Temmuz-2017.

Banta, J. E. ve M. A. Abidi (1996). "Autonomous placement of a range sensor for acquisition of optimal 3-D models". İn: Proceedings of the 1996 IEEE IECON. 22nd International Conference on Industrial Electronics, Control, and Instrumentation. **volume** 3, 1583–1588 vol.3. DOI: 10.1109/IECON.1996.570630.

Banta, J. E., L. R. Wong, C. Dumont ve M. A. Abidi (2000). "A next-best-view system for autonomous 3-D object reconstruction". İn: IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans 30.5, pp. 589–598. ISSN: 1083-4427. DOI: 10.1109/3468.867866.

Bley, H., M. Bernardi, C. Franke ve U. Seel (2001). "Process-based assembly planning using a simulation system with cell calibration". İn: Proceedings of the 2001 IEEE International Symposium on Assembly and Task Planning (ISATP2001). Assembly and Disassembly

in the Twenty-first Century. (Cat. No.01TH8560), pp. 116–121. DOI: 10.1109/ISATP.2001.928976.

Borangiu, T., A. Dogar ve A. Dumitrache (2008). “Constraints-based motion planning for an automatic, flexible laser scanning robotized platform”. İn: 2008 IEEE International Conference on Automation, Quality and Testing, Robotics. **volume** 2, pp. 65–69. DOI: 10.1109/AQTR.2008.4588795.

Botsch, Mario, Leif Kobbelt, Mark Pauly, Pierre Alliez ve Bruno Lévy (2010). Polygon Mesh Processing. Natick, Massachusetts: A K Peters, Ltd.

Callieri, M., A. Fasano, G. Impoco, P. Cignoni, R. Scopigno, G. Parrini ve G. Biagini (2004). “RoboScan: an automatic system for accurate and unattended 3D scanning”. İn: Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004. Pp. 805–812. DOI: 10.1109/TDPVT.2004.1335398.

Chen, S. Y. ve Y. F. Li (2005). “Vision sensor planning for 3-D model acquisition”. İn: IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 35.5, pp. 894–904. ISSN: 1083-4419. DOI: 10.1109/TSMCB.2005.846907.

Connolly, C. (1985). “The determination of next best views”. İn: Proceedings. 1985 IEEE International Conference on Robotics and Automation. **volume** 2, pp. 432–435. DOI: 10.1109/ROBOT.1985.1087372.

Ding, H., B. He ve X. Zhou (2010). “A Novel Occlusion Planning Method for Unknown 3D Objects Automatic Reconstruction”. İn: 2010 International Conference on Artificial Intelligence and Computational Intelligence. **volume** 1, pp. 463–467. DOI: 10.1109/AICI.2010.103.

Fabio, R. (2003). “From Point Cloud to Surface:The Modeling and Visualization Problem”. İn: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. **volume** XXXIV-5/W10.

Fang, Wei ve Bingwei He (2012). “Automatic view planning for 3D reconstruction and occlusion handling based on the integration of active and passive vision”. İn: 2012 IEEE International Symposium on Industrial Electronics, pp. 1116–1121. DOI: 10.1109/ISIE.2012.6237245.

Filz, K. M., M. Vincze ve J. P. Prenninger (1995). “Camera system to detect the orientation of a corner cube in real time”. İn: Proceedings of 1995 IEEE International Conference on Robotics and Automation. **volume** 2, 1713–1718 vol.2. DOI: 10.1109/ROBOT.1995.525520.

Garcia, M. A., S. Velazquez, A. D. Sappa ve L. Basanez (1998). “Autonomous sensor planning for 3D reconstruction of complex objects from range images”. İn: Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146). **volume** 4, 3085–3090 vol.4. DOI: 10.1109/ROBOT.1998.680899.

He, B.W., Zhou Xiao Long ve Y.F. Li (2010). “The research of an automatic object reconstruction method based on limit visible region of the laser-scanning vision system”. İn: Robotics and Computer-Integrated Manufacturing 26.6, pp. 711–719. DOI: 10.1016/j.rcim.2010.03.011. URL: https://doi.org/10.1016/j.rcim.2010.03.011.

He, Bingwei ve Y. F. Li (2006). “A Self-Termination Judgment Method in 3D Object Automatic Measurement and Inspection”. İn: 2006 6th World Congress on Intelligent Control and Automation. **volume** 1, pp. 5008–5012. DOI: 10.1109/WCICA.2006.1713341.

He, Bingwei, Xiaolong Zhou ve Y. F. Li (2009). “A view planning method for automatic 3D modeling based on the trend surface and limit region”. İn: 2009 IEEE Instrumentation and Measurement Technology Conference, pp. 803–808. DOI: 10.1109/IMTC.2009.5168560.

Hu, J. S., J. J. Wang ve Y. J. Chang (2012). “Kinematic calibration of manipulator using single laser pointer”. İn: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 426–430. DOI: 10.1109/IROS.2012.6385531.

Huang, C., D. Chen ve X. Tang (2015). “Robotic Hand-Eye Calibration Based on Active Vision”. İn: 2015 8th International Symposium on Computational Intelligence and Design (ISCID). **volume** 1, pp. 55–59. DOI: 10.1109/ISCID.2015.246.

Isler, S., R. Sabzevari, J. Delmerico ve D. Scaramuzza (2016). “An information gain formulation for active volumetric 3D reconstruction”. İn: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 3477–3484. DOI: 10.1109/ICRA.2016.7487527.

Karaszewski, M., R. Sitnik ve E. Bunsch (2012). “On-line, collision-free positioning of a scanner during fully automated three-dimensional measurement of cultural heritage objects”. İn: Robotics and Autonomous Systems 60.9, pp. 1205–1219. DOI: 10.1016/j.robot.2012.05.005. URL: https://doi.org/10.1016/j.robot.2012.05.005.

Khalfaoui, Souhaiel, Antoine Aigueperse, Ralph Seulin, Yohan Fougerolle ve David Fofi (2012). “Fully automatic 3D digitization of unknown objects using progressive data bounding box”. İn: Three-Dimensional Image Processing (3DIP) and Applications II. Ed. Atilla M. Baskurt ve Robert Sitnik. SPIE. DOI: 10.1117/12.909164. URL: https://doi.org/10.1117/12.909164.

Khalfaoui, Souhaiel, Ralph Seulin, Yohan Fougerolle ve David Fofi (2013). “An efficient method for fully automatic 3D digitization of unknown objects”. İn: Computers in Industry 64.9, pp. 1152–1160. DOI: 10.1016/j.compind.2013.04.005. URL: https://doi.org/10.1016/j.compind.2013.04.005.

Krainin, M., B. Curless ve D. Fox (2011). “Autonomous generation of complete 3D object models using next best view manipulation planning”. İn: 2011 IEEE International

Conference on Robotics and Automation, pp. 5031–5037. DOI: 10.1109/ICRA.2011.5980429.

Kriegel, S., T. Bodenmüller, M. Suppa ve G. Hirzinger (2011). “A surface-based Next-Best-View approach for automated 3D model completion of unknown objects”. İn: 2011 IEEE International Conference on Robotics and Automation, pp. 4869–4874. DOI: 10.1109/ICRA.2011.5979947.

Kriegel, S., C. Rink, T. Bodenmüller, A. Narr, M. Suppa ve G. Hirzinger (2012). “Next-best-scan planning for autonomous 3D modeling”. İn: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2850–2856. DOI: 10.1109/IROS.2012.6385624.

Kriegel, Simon, Christian Rink, Tim Bodenmüller ve Michael Suppa (2013). “Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects”. İn: Journal of Real-Time Image Processing 10.4, pp. 611–631. DOI: 10.1007/s11554-013-0386-6. URL: https://doi.org/10.1007/s11554-013-0386-6.

Lei, Sun, Liu Jingtai, Sun Weiwei, Wu Shuihua ve Huang Xingbo (2004). “Geometry-based robot calibration method”. İn: Robotics and Automation, 2004. Proceedings. ICRA ’04. 2004 IEEE International Conference on. **volume** 2, 1907–1912 Vol.2. DOI: 10.1109/ROBOT.2004.1308102.

Li, Y. F. ve Z. G. Liu (2005). “Information entropy-based viewpoint planning for 3-D object reconstruction”. İn: IEEE Transactions on Robotics 21.3, pp. 324–337. ISSN: 1552-3098. DOI: 10.1109/TRO.2004.837239.

Li, Y F, B He, S Chen ve P Bao (2005). “A view planning method incorporating self-termination for automated surface measurement”. İn: Measurement Science and Technology 16.9, pp. 1865–1877. DOI: 10.1088/0957-0233/16/9/020. URL: https://doi.org/10.1088/0957-0233/16/9/020.

Liang, P., J. Feng ve S. Hackwood (1988). “A general framework for robot hand-eye coordination”. İn: Proceedings. 1988 IEEE International Conference on Robotics and Automation, 1082–1087 vol.2. DOI: 10.1109/ROBOT.1988.12204.

Marton, Zoltan Csaba, Radu Bogdan Rusu ve Michael Beetz (2009). “On Fast Surface Reconstruction Methods for Large and Noisy Datasets”. İn: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Kobe, Japan.

Maver, J. ve R. Bajcsy (1991). “Multiple view planning”. İn: [1991 Proceedings] 6th Mediterranean Electrotechnical Conference, 1201–1204 vol.2. DOI: 10.1109/MELCON.1991.162057.

— (1992). “Occlusions and the next view planning”. İn: Proceedings 1992 IEEE International Conference on Robotics and Automation, 1806–1811 vol.2. DOI: 10.1109/ROBOT.1992.220118.

— (1993). “Occlusions as a guide for planning the next view”. İn: IEEE Transactions on Pattern Analysis and Machine Intelligence 15.5, pp. 417–433. ISSN: 0162-8828. DOI: 10.1109/34.211463.

Nieves, E., N. Xi, B. Du ve Y. Jia (2012). “A reflected laser line approach for industrial robot calibration”. İn: 2012 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), pp. 610–615. DOI: 10.1109/AIM.2012.6266035.

Papadopoulos-Orfanos, D. ve F. Schmitt (1997). “Automatic 3-D digitization using a laser rangefinder with a small field of view”. İn: Proceedings. International Conference on Recent Advances in 3-D Digital Imaging and Modeling (Cat. No.97TB100134), pp. 60–67. DOI: 10.1109/IM.1997.603849.

Pito, R. (1995). A Solution to the Next Best View Problem for Automated CAD Model Acquisition of Free-Form Objects Using Range Cameras. techreport. Department of Computer **and** Information Science, University of Pennsylvania.

Pito, R. (1999). “A solution to the next best view problem for automated surface acquisition”. İn: IEEE Transactions on Pattern Analysis and Machine Intelligence 21.10, pp. 1016–1030. ISSN: 0162-8828. DOI: 10.1109/34.799908.

Rink, C. ve S. Kriegel (2016). “Streaming Monte Carlo Pose Estimation for Autonomous Object Modeling”. İn: 2016 13th Conference on Computer and Robot Vision (CRV), pp. 156–163. DOI: 10.1109/CRV.2016.21.

Rusu, R. B., N. Blodow ve M. Beetz (2009). “Fast Point Feature Histograms (FPFH) for 3D registration”. İn: 2009 IEEE International Conference on Robotics and Automation, pp. 3212–3217. DOI: 10.1109/ROBOT.2009.5152473.

Rusu, Radu Bogdan (2009). “Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments”. phdthesis. Computer Science department, Technische Universitaet Muenchen, Germany.

Secil, S. (2015). “Endüstriyel Robot Kolları İçin Otomatik Programlama Yöntemi Geliştirme”. mathesis. Eskisehir Osmangazi Universitesi.

Secil, S., K. Turgut, O. Parlaktuna ve M. Ozkan (2015). “Simple programming scheme for industrial robot manipulators: A case study for feasibility proof”. İn: 2015 International Symposium on Innovations in Intelligent SysTems and Applications (INISTA), pp. 1–7. DOI: 10.1109/INISTA.2015.7276722.

Secil, S., C. Soyleyici, M. Ozkan, O. Parlaktuna, H. Dutagaci, K. Turgut ve M. Parlaktuna (2016). “Design of an application programming interface (API) for commercial industrial robots”. İn: 2016 IEEE 14th International Conference on Industrial Informatics (INDIN), pp. 225–230. DOI: 10.1109/INDIN.2016.7819163.

Seçil, S., K. Turgut, O. Parlaktuna ve M. Özkan (2014). “3-D visualization system for geometric parts Using a laser profile sensor and an industrial robot”. İn: 2014 IEEE

International Symposium on Robotics and Manufacturing Automation (ROMA), pp. 160–165. DOI: 10.1109/ROMA.2014.7295880.

Singh, M. K., K. S. Venkatesh ve A. Dutta (2015). “A new next best view method for 3D modeling of unknown objects”. İn: 2015 Third International Conference on Image Information Processing (ICIIP), pp. 516–519. DOI: 10.1109/ICIIP.2015.7414827.

Sälzle, Martin (2012). Half-edge Data Structure. http://pointclouds.org/blog/nvcs/martin/index.php/. erişim tarihi 10-Temmuz-2017.

Suppa, M., S. Kielhofer, J. Langwald, F. Hacker, K. H. Strobl ve G. Hirzinger (2007). “The 3D-Modeller: A Multi-Purpose Vision Platform”. İn: Proceedings 2007 IEEE International Conference on Robotics and Automation, pp. 781–787. DOI: 10.1109/ROBOT.2007.363081.

Torabi, L. ve K. Gupta (2011). “An autonomous six-DOF eye-in-hand system for in situ 3D object modeling”. İn: The International Journal of Robotics Research 31.1, pp. 82–100. DOI: 10.1177/0278364911425836. URL: https://doi.org/10.1177/0278364911425836.

— (2012). “An autonomous 9-DOF mobile-manipulator system for in situ 3D object modeling”. İn: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4540–4541. DOI: 10.1109/IROS.2012.6386271.

Trummer, M., C. Munkelt ve J. Denzler (2010). “Online Next-Best-View Planning for Accuracy Optimization Using an Extended E-Criterion”. İn: 2010 20th International Conference on Pattern Recognition, pp. 1642–1645. DOI: 10.1109/ICPR.2010.406.

Vasquez-Gomez, J. I., L. E. Sucar ve R. Murrieta-Cid (2013). “Hierarchical Ray Tracing for Fast Volumetric Next-Best-View Planning”. İn: 2013 International Conference on Computer and Robot Vision, pp. 181–187. DOI: 10.1109/CRV.2013.42.

Vasquez-Gomez, J. Irving, L. Enrique Sucar, Rafael Murrieta-Cid ve Efrain Lopez-Damian (2014). “Volumetric Next-best-view Planning for 3D Object Reconstruction with Positioning Error”. İn: International Journal of Advanced Robotic Systems 11.10, p. 159. DOI: 10.5772/58759. URL: https://doi.org/10.5772/58759.

Vásquez-Gómez, J. I., E. Löpez-Damian ve L. E. Sucar (2009). “View planning for 3D object reconstruction”. İn: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4015–4020. DOI: 10.1109/IROS.2009.5354383.

Wagner, M., P. Heß, S. Reitelshöfer ve J. Franke (2015). “Self-calibration method for a robotic based 3D scanning system”. İn: 2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA), pp. 1–6. DOI: 10.1109/ETFA.2015.7301508.

Wang, W., A. Li ve D. Wu (2009). “Robot calibration by observing a virtual fixed point”. İn: 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1351–1355. DOI: 10.1109/ROBIO.2009.5420879.

Wenhardt, S., B. Deutsch, J. Hornegger, H. Niemann ve J. Denzler (2006). “An Information Theoretic Approach for Next Best View Planning in 3-D Reconstruction”. İn: 18th International Conference on Pattern Recognition (ICPR’06). **volume** 1, pp. 103–106. DOI: 10.1109/ICPR.2006.253.

Wenhardt, S., B. Deutsch, E. Angelopoulou ve H. Niemann (2007). “Active Visual Object Reconstruction using D-, E-, and T-Optimal Next Best Views”. İn: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7. DOI: 10.1109/CVPR.2007.383363.

Wong, L. M., C. Dumont ve M. A. Abidi (1999). “Next best view system in a 3D object modeling task”. İn: Computational Intelligence in Robotics and Automation, 1999. CIRA ’99. Proceedings. 1999 IEEE International Symposium on, pp. 306–311. DOI: 10.1109/CIRA.1999.810066.

Yuan, Xiaobu (1995). “A mechanism of automatic 3D object modeling”. İn: IEEE Transactions on Pattern Analysis and Machine Intelligence 17.3, pp. 307–311. ISSN: 0162-8828. DOI: 10.1109/34.368196.

Zhang, L., J. Zuo, X. Yao, X. Zhang ve L. Shuai (2015). “A robot visual servo-based approach to the determination of next best views”. İn: 2015 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 2654–2659. DOI: 10.1109/ICMA.2015.7237906.

Zhou, Xiaolong, Bingwei He ve Y. F. Li (2008). “A New View Planning Method for Automatic Modeling of Three Dimensional Objects”. İn: Intelligent Robotics and Applications. Springer Berlin Heidelberg, pp. 161–170. DOI: 10.1007/978-3-540-88513-9_18. URL: https://doi.org/10.1007/978-3-540-88513-9_18.