

**ANKARAÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**ÜÇ BOYUTLU LAZER TARAYICI VE BİLGİSAYAR TEKNİKLERİ İLE
CİSİMLERİN SANAL ORTAMA AKTARILMASI**

Kurtuluş KÜLLÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA

2010

Her hakkı saklıdır.

ÖZET

Yüksek Lisans Tezi

ÜÇ BOYUTLU LAZER TARAYICI VE BİLGİSAYAR TEKNİKLERİ İLE CİSİMLERİN SANAL ORTAMA AKTARILMASI

Kurtuluş KÜLLÜ

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof.Dr. Baki KOYUNCU

Bu tezde ana amaç belirli büyüklüklerdeki nesnelerin makul bir süre zarfında 3 boyutlu (3B) modellerini çıkarabilecek, düşük maliyetli bir 3B tarayıcı inşa etmektir. Böyle bir sistemin yapımı sırasındaki değerlendirmelerde maliyet, isabetlilik, nesne büyüklükleri, nesne özellikleri ve zaman konularıyla ilgili ödünleşimler (tradeoffs) açıkça belirmektedir. Dolayısıyla, tasarım sürecinde bu ödünleşimler değerlendirilerek, laboratuvar ortamına getirilebilecek büyüklükte, yüzeyleri saydam veya parlak olmayan, hassas nitelikli kültürel değere sahip tarihi eserlerin ya da sanat eserlerinin, orta-yüksek derecede isabetli 3B modellerini oluşturabilecek, mümkün olduğunca düşük maliyetli, biçimli ışık kullanan optik bir tarayıcı yapılmasına karar verilmiştir.

Yapılan tarayıcı doğrusal bir Helyum-Neon (HE-NE) lazer ışık vericisi, iki standart web kamera, bir adım motoru, motor kontrol kartı ve bir kişisel bilgisayar kullanmaktadır. Nesneler, adım motoru ile bilgisayardan hareket ettirilen yuvarlak bir döner platform üzerine yerleştirilerek üzerlerine lazer ışık düşürülmekte ve kameralar bu şekilde görüntü almaktadır. Görüntülerde nesne yüzeyine düşen ve yansıyan ışığın gözlemlendiği resim ögesi (piksel) pozisyonları kullanılarak bilgisayar vasıtasıyla 3B sanal modeller oluşturulmaktadır. Bazı örnek nesneler üzerinde bu adımlar uygulanmış ve sonuç olarak oluşturulan 3B modellerin isabetliliği incelenmiştir.

Öncelikle, 3B tarayıcılar ile ilgili yapılan araştırmalar incelenmiş ve incelemeler sonucunda ortaya konan bir 3B tarayıcılar sınıflandırması Giriş bölümünde verilmiştir. Bu bölümde daha sonra kültürel miras nesnelerinin taranması ile ilgili büyük projeler özetlenmiştir. Kuramsal Temeller bölümünde, ortaya konan tasarıma dayanarak ihtiyaç duyulacak kuramlar, algoritmalar ve teknikler anlatılmıştır. Temin edilen donanımın ve Kuramsal Temeller bölümünde anlatılan tekniklerin uygulamada nasıl kullanıldığı Materyal ve Yöntem bölümünde irdelenmiştir. Bulgular bölümü bazı erken tarama sonuçları inceleyerek varılan kararları belirtmektedir. Son olarak Tartışma ve Sonuç kısmında sistem oluşturulurken karşılaşılan problemler, oluşturulan sistemin değerlendirilmesi ve daha da geliştirilmesi için yapılabilecek çalışmalar irdelenmektedir.

Mart 2010, 78 sayfa

Anahtar Kelimeler:3B tarayıcı, 3B model, biçimli ışık, üçgenleme, yeniden yapılandırma.

ABSTRACT

Master Thesis

CONSTRUCTION OF A 3-DIMENSIONAL LASER SCANNER AND OBJECT MODELLING BY USING IMAGE PROCESSING TECHNIQUES

Kurtuluş KÜLLÜ

Ankara University
Institute of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof.Dr. Baki KOYUNCU

The primary aim of this thesis is to construct a low cost 3 Dimensional (3D) scanner which can produce accurate 3D models of objects of a specified size range within a reasonable amount of time. Tradeoffs between issues regarding cost, accuracy, size range of objects, and time are directly evident in considerations towards building such a system. Hence, these tradeoffs were evaluated during the design stage and an optical 3D scanner based on structured light which is capable of scanning delicate cultural heritage objects that are of proper size to be brought to the laboratory environment and that do not have a transparent or shiny surface, and which can produce medium-high accuracy 3D models was decided to be constructed while keeping the cost as low as possible.

Constructed scanner uses a linear Helium-Neon (HE-NE) laser light transmitter, two standard web cameras, a stepper motor, motor control card and a PC. The laser beam is projected onto the surface of objects which are placed on a computer controlled revolving circular platform and cameras capture images of this scene. 3D virtual models are created with the PC by using positions of image pixels on which light reflected from the object surface can be observed. These steps were applied on some sample objects and accuracy of resulting 3D models was analysed.

Initially, research regarding 3D scanners were analysed and a classification of 3D scanners derived is given in Introduction section. Later in the same section large projects on scanning cultural heritage objects are summarised. In the Theoretical Fundamentals section, theories, algorithms, and methods necessary for the design are explained. Details of how acquired hardware and techniques explained in Theoretical Foundations section are used in practice are discussed in Materials and Methods section. Results section provides decisions made based on some early scanning results. Finally, Evaluation and Conclusions section provides a discussion of construction problems, an evaluation of the system created and suggestions for further work to improve the system.

March 2010, 78 pages

Key Words: 3D scanner, 3D model, structured light, triangulation, reconstruction.

TEŞEKKÜR

Çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarını esirgemeyerek akademik ortamda olduğu kadar beşeri ilişkilerde de engin fikirleriyle yetişme ve gelişmeye katkıda bulunan danışman hocam Sayın Prof. Dr. Baki KOYUNCU (Ankara Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı) başta olmak üzere, gerek bölümümüzde gerekse diğer bölümlerde veya üniversitelerde, sahip oldukları bilgi birikimlerini ve tecrübelerini paylaşarak bana şimdiye kadarki akademik yaşamımda katkıda bulunan ve bu noktaya gelmemde yardımcı olan tüm hocalarıma, çalışmalarım süresince maddi manevi desteklerini eksik etmeyen aileme ve dostlarıma içten teşekkür ederim.

Kurtuluş KÜLLÜ

Ankara, Mart 2010

İÇİNDEKİLER

ÖZET.....	ii
ABSTRACT	iii
TEŞEKKÜR	iv
SİMGELER DİZİNİ	vi
ŞEKİLLER DİZİNİ	vii
ÇİZELGELER DİZİNİ	ix
1. GİRİŞ	1
1.1 Tarayıcıların Sınıflandırılması	9
1.1.1 Uçuş süresi tarayıcıları	10
1.1.3 Biçimli ışık tarayıcıları	12
2. KURAMSAL TEMELLER.....	15
2.3 Yaklaşık Dönüşümün Hesaplanması.....	18
2.4 Aydınlik Resim Öğelerinin Belirlenmesi.....	23
2.6 Nokta Bulutundan 3B Modele	30
2.6.1 Yeniden yapılandırma algoritmalarının sınıflandırılması	30
2.6.2 Delaunay üçgenleme metodu.....	34
2.6.3 İlerleyen üçgenler (İÜ) ve top-pivotlama (TPA) algoritmaları.....	35
3. MATERYAL VE YÖNTEM.....	37
3.1 Döner Platform.....	38
3.2 Çizgi Lazer Kaynağı	43
3.3 Algılayıcı Kamera.....	43
3.4 Aydınlik Resim Öğesi Pozisyonlarının Elde Edilmesi	44
3.7 Yeniden Yapılandırma	52
3.7.1 Çekirdek üçgenin belirlenmesi.....	53
3.7.2 Verilen üç noktaya kürenin oturtulması.....	54
3.7.3 Kısmi yüzeyden döngüsel olarak topun pivotlanması	55
3.8 Yazılım	56
4. BULGULAR.....	62
5. TARTIŞMA VE SONUÇ.....	74
KAYNAKLAR	77
ÖZGEÇMİŞ.....	80

SİMGELERDİZİNİ

1B	Bir boyutlu
2B	İki boyutlu
3B	Üç boyutlu
3D	İng. Three Dimensional
BDT	Bilgisayar Destekli Tasarım
BPA	İng. Ball-Pivoting Algorithm, bkz. TPA
CCD	İng. Charge Coupled Device
EKK	En Küçük Kareler
HE-NE	Helyum-Neon, İng. Helium-Neon
İng.	İngilizce
İÜ	İlerleyen üçgenler
KÖM	Koordinat Ölçme Makineleri
KYM	Kırmızı-Yeşil-Mavi
LPT	İng. Line Print Terminal
MT	İng. Marching Triangles, bkz. İÜ
NURBS	Düzgün olmayan, rasyonel B-Spline, İng. Non-uniform Rational B-Spline
OAU	Ortalama Asgari Uzaklık
OP-AMP	İng. Operational Amplifier, İşlemsel kuvvetlendirici
PC	İng. Personal Computer
RGB	İng. Red-Green-Blue
SD	Sürekli Dalga
TPA	Top pivotlama algoritması

ŞEKİLLER DİZİNİ

Şekil 1.1 Bilgisayarda bir 3B model görünümü.....	2
Şekil 1.2 Bazı örnek 3B tarayıcı görüntümleri.....	3
Şekil 1.3 Kalkış platformu üzerinde Endeavour uzay mekiğinin nokta bulutu	4
Şekil 1.4 Stanford tavşanı adlı nokta bulutu ve yeniden yapılandırma ile oluşturulan yüzey modeli	4
Şekil 1.5 Örnek bir nesnenin 3B tarama ve 3B yazdırma ile oluşturulan kopyası.....	6
Şekil 1.6 Dijital Mikelanjelo projesi sonucunda oluşturulan sanal Davut heykeli modeli (Levoy vd. 2000).....	7
Şekil 1.7 Taranan çivi yazısı tabletlerin model görünümü (Kumar vd. 2003).....	7
Şekil 1.8 Çok detaylı bir eski Roma şehir maketinden tarama ile oluşturulan modelde Kolezyum vadisinin görüntüsü (Guidi vd. 2005).....	8
Şekil 1.9 Mikelanjelo'nun tarama ile oluşturulan Pietà heykeli modeli (Bernardini vd. 2002).....	8
Şekil 1.10 Temasa dayalı bir tarayıcı örneği.....	9
Şekil 1.11 Bir lazer mesafe ölçer	11
Şekil 1.12 Üçgenleme tabanlı tarama.....	12
Şekil 1.13 Çizgisel lazer ışık kullanan biçimli ışık tarama sistemi.....	13
Şekil 2.1 Genel üçgenleme kavramı.....	15
Şekil 2.2 Üçgen tabanlı tarama tekniğinin şematik modeli	17
Şekil 2.3 Dünya koordinatlarından (X_r, Y_r, Z_r) görüntü koordinatlarına $(X_l, Y_l, 1)$ fonksiyon T	17
Şekil 2.4 Görüntü koordinatlarından $(X_l, Y_l, 1)$ dünya koordinatlarına (X_r, Y_r, Z_r) fonksiyon $T - 1$	18
Şekil 2.5 Yeğnlik değerleri için bir histogram örneği.....	25
Şekil 2.6 Görüntülerde karşılaşılan optik etkiler.....	26
Şekil 2.7 Cismin kendi kendini örtmesi dolayısıyla oluşan gölge etkisi ve bir miktar çoklu yansıma	27
Şekil 2.8 Çevritin arka plandan ayrılması.....	28
Şekil 2.9 Çıkarılan profilin gri tonu fonksiyonu	29
Şekil 2.10 Çevrit çıkarımı algoritması	29
Şekil 2.11 Hoppe algoritmasıyla yeniden yapılandırma	33
Şekil 2.12 Bir düzlemde (2B) Delaunay Üçgenleme örneği.....	34
Şekil 2.13 Ardışık düzen olarak model edinimi.....	36
Şekil 3.1 Sistemin yukarıdan şematik görünümü.....	37
Şekil 3.2 Döner platform.....	39
Şekil 3.3 Motor haznesi açık (solda) ve kapalı (sağda) olarak	39
Şekil 3.4 Motor kontrolü için kullanılan ilk kartlar	40
Şekil 3.5 Adım motor sürücü kartı.....	40
Şekil 3.6 Op-amp devre şeması.....	41
Şekil 3.7 741 tipi OP-AMP entegre yonga.....	41
Şekil 3.8 Sürücü kart, OP-AMP devre ve bağlantıların görünümü.....	41
Şekil 3.9 Yeni platform yüzeyi	42
Şekil 3.10 Nesne yüzeyi ile lazer ışığın kesişim görüntüsü	45
Şekil 3.11 Şekil 3.10'dan elde edilen ikili görüntü	46
Şekil 3.12 Gerçekteki pozisyonlara koordinat uzayının yerleştirimi	47

Şekil 3.13 Derinlik kalibrasyonu görüntüleri.....	48
Şekil 3.14 Yükseklik kalibrasyonu görüntüleri.....	49
Şekil 3.15 Örnek bir nesnenin taramada alınan görüntüsü	51
Şekil 3.16 TPA algoritması.....	52
Şekil 3.17 Çekirdek üçgen bulma mantığı	53
Şekil 3.18 Üç nokta ve onlara eşit uzaklıktaki noktaların tanımladığı doğru	54
Şekil 3.19 Grafik arayüzün genel görünümü	57
Şekil 3.20 Tarama görüntüleri ve hafızaya kaydedilmesi.....	58
Şekil 3.21 Çıkarılan 3B nokta bilgileri ve kaydedilmesi	59
Şekil 3.22 Oluşturulan yazılımın önemli sınıfları ve kamusal elemanları	61
Şekil 4.1 İlk elde edilen nokta bulutları	62
Şekil 4.2 İyileştirmelerden sonra taranan cisimler ve oluşturulan nokta bulutları.....	63
Şekil 4.3 Kendi kendini örtme sonucu nokta bulutunda oluşan boşluk	65
Şekil 4.4 İki kameralı sistemin yukarıdan şematik görünümü	66
Şekil 4.5 a. Farklı kameralardan oluşturulan boşluklu nokta bulutları, b. ikisinin üst üste getirilmesiyle (küme birleşimi) oluşan nokta bulutu	67
Şekil 4.6 Oluşturulan nokta bulutunda TPA uygulanmasına örnek.....	69
Şekil 4.7 Farklı c değerleri için test grubu (i) sonuçları.....	71
Şekil 4.8 Farklı c değerleri için test grubu (ii) sonuçları.....	72
Şekil 4.9 Farklı c değerleri için test grubu (iii) sonuçları	73
Şekil 5.1 Örnek cisimlerden elde edilen modeller	76

ÇİZELGELER DİZİNİ

Çizelge 3.1 Kullanılan adım motorunun teknik özellikleri.....	38
Çizelge 3.2 Beşer adet derinlik ve yükseklik görüntülerinden çıkartılan 5'er resim ögesi pozisyonları ve yüzeyin z/y konumları	50
Çizelge 4.1 Gerçek cisimlerden ve nokta bulutlarından alınan bazı geometrik ölçümler.....	64
Çizelge 4.2 Oranlanmış nokta bulutu değerleri ve bağıl hatalar	65
Çizelge 4.3 Farklı bölge boyutlarıyla indirgenmiş nokta bulutlarındaki nokta sayıları $ P' $ ve OAU değerleri d'	69
Çizelge 4.4 Her testteki r değerleri	70

1. GİRİŞ

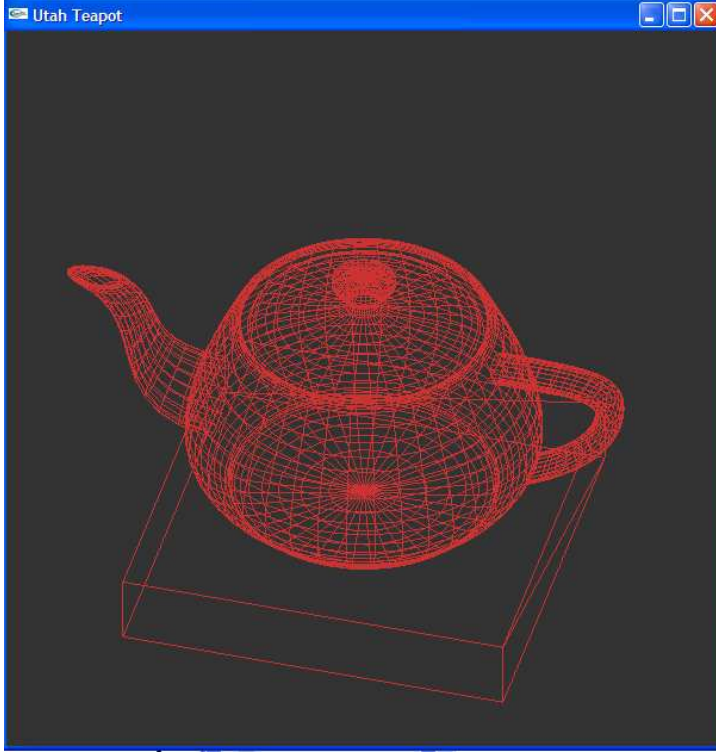
Bugün bilişim dünyasında çok geniş bir alan olan bilgisayar grafiklerinin yine çok genel bir tanımına göre bu alan bilgisayarda yazı veya ses olmayan her şeyi içermektedir. Aslına bakılırsa yazı bile ekranda gösterilebilmesi açısından bilgisayar grafiklerinin alanına girmektedir. Böylesine geniş bir alandaki araştırmaların daha özel alt gruplarda incelenmesi doğaldır. Geleneksel bir sınıflandırma 2 boyutlu (2B) ve 3 boyutlu (3B) grafik sınıflandırmasıdır. Bu iki alt alan sıklıkla birbirinin kullandığı tekniklerden faydalanır. Örneğin 3B uygulamalar içeriklerini standart bilgisayar ekranında gösterebilmek için sık sık 2B teknikler kullanırlar.

Bilgisayar grafikleri son yirmi yıllık süreçte tüm dünyada ana bilişim teknolojileri tekniklerinden birisi olarak yer etmiştir. Bilişim sistemleri gerçek veya hayali dünyaların sanal benzerlerini yaratmak için giderek daha fazla kullanılmaktadır. Bu benzetimler (simülasyonlar) karmaşık nesnelerin ve çevrelerin modellenmesini, işlenmesini gerektirmektedir. Bilgisayar grafiklerinin amaçlarından birisi verileri görselliğe çevirerek kullanıcıların karmaşık nesnelerle veya verilerle, doğal, insanlar için kolay yollarla etkileşimini sağlamaktır.

3B grafik alanında çok önemli bir kavram, nesne modelleridir. En genel tanımıyla model, bir nesnenin, sistemin veya kavramın kendisini ya da nasıl işlediğini göstermek amacıyla tasarlanmış plan, gösterim (genellikle daha küçük) veya tanımdır. Bu terim farklı alanlarda yer bulmaktadır ve verilen bu tanım da genel anlamı içermektedir. Bilgisayar Mühendisliği alanında 3B model, bir nesnenin 3B yüzeyinin matematiksel bir gösterimidir (Şekil 1.1).

Bilgisayar grafiklerinde 3B modellerin kullanımları aslen kişisel bilgisayarlarda 3B grafiklerin kullanımından eskidir. Bu modeller günümüzde tıbbi görüntüleme, eğlence endüstrisi, mimari ve üretim sektörü gibi birçok alanda kullanılmaktadır. Bilgisayar ortamındaki 3B geometrik modeller değişik amaçlar için kullanılabilir. Örneğin verilen bir 3B modeli, gerçekleştirme (rendering) işlemi ile 2B görüntüler halinde göstermek,

fiziksel bilgisayar benzetimlerinde kullanmak, hatta 3B yazdırma ile fiziksel olarak yaratmak mümkündür.



Şekil 1.1 Bilgisayarda bir 3B model görünümü

3B modellerin yaratılmasında ilk dönemlerdeki seçenek el ile tanımlama yöntemi idi. Genellikle Bilgisayar Destekli Tasarım (BDT) yazılımları kullanılarak yapılan el ile yaratma işlemi geleneksel heykeltıraşlığa benzetilebilir. İlk zamanlarda bu şekilde yaratılan modeller en iyi ihtimalle çok zaman harcanan ve kusurlu modeller olmaktadır. Ayrıca karmaşık şekilli ve ince geometrik detaylara sahip nesnelerin modellerini oluşturmak çok zor olduğu için bu tekniğin sınırları dışında kalabilmekteydi. Günümüzde ise 3D Studio Max, AC3D, Blender, vb. gibi bazıları genel kullanım amaçlı, bazıları da özel uygulama alanlarına yönelik 3B bilgisayar grafikleri yazılımları artık el ile tasarımın dışında kullanım kolaylıkları da sağlayan karmaşık paketler haline gelmiştir.

Bugün 3B modellerin otomatik olarak elde edilmeside mümkündür. Otomatik modellemede seçenekler yordamsal (procedural) modelleme ve taramadır. Yordamsal

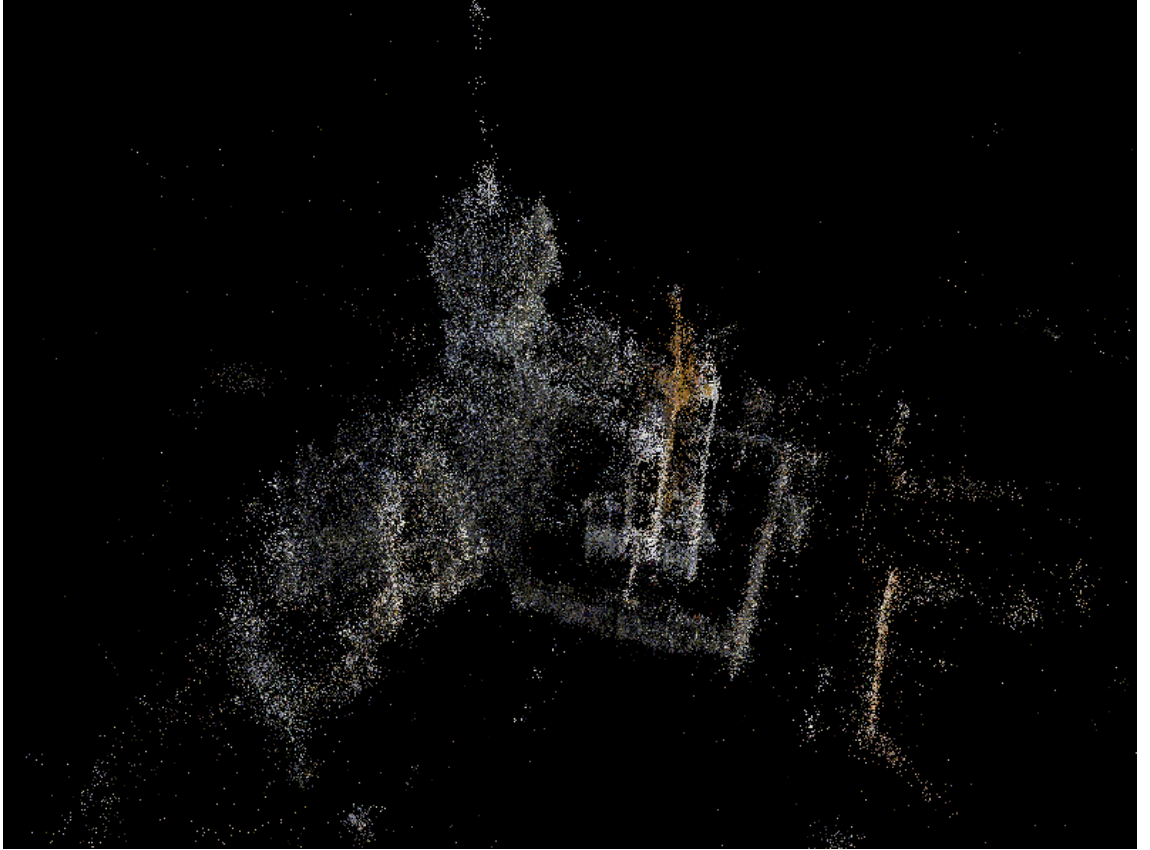
modelleme 3B modeller ve dokular (texture) oluşturmak için L-sistemler, fraktallar ve üretimsel (generative) modelleme gibi matematiksel yapıları uygulayan algoritmalar kullanan tekniklere verilen genel addır. 3B tarayıcılar ise 2B belge tarayıcıların taranan cismin dijital bir kopyasını oluşturmaya benzer bir şekilde gerçek bir nesneyi veya çevreyi çözümleyerek, o nesnenin veya çevrenin şekli hakkında veri toplayan cihazlardır (Şekil 1.2). Tarayıcı kullanılarak bir cisim hakkında toplanan veri cismin şeklinin yanı sıra doku, renk, vb. bilgiler de içerebilir (Curless 1999). Toplanan bu veriler nesnenin veya çevrenin birçok uygulamada faydalanılan üç boyutlu dijital modelinin oluşturulmasında kullanılabilir. Bu işlemin yaygın olarak kullanıldığı başlıca alanlar eğlence endüstrisi (sinema ve bilgisayar oyunları), endüstriyel tasarım, elektronik ticaret uygulamaları, tıp, eğitim, bilgisayar görüşü ve kültürel eserlerin dokümantasyonu olarak sıralanabilir.



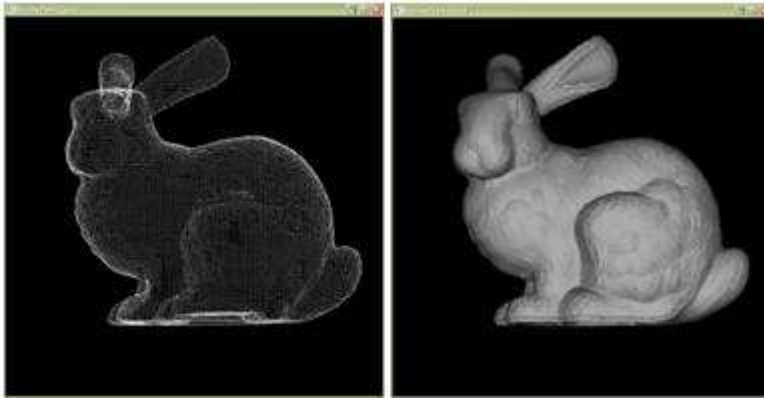
Şekil 1.2 Bazı örnek 3B tarayıcı görüntüleri

3B tarayıcıların yapımı için farklı teknolojiler mevcut olup, her teknolojinin kendine has sınırları, avantajları ve maliyeti vardır. Buna rağmen hepsinin ortak sonucu cismin yüzeyinden örneklemeler (sampling) yaparak 3B koordinat sisteminde bir “nokta bulutu” elde etmeleridir (Şekil 1.3). Nokta bulutu 3B uzayda tanımlı bir noktalar kümesidir. Veri olarak, bu bulutlar, taranan nesnenin görünür yüzeyini temsil etmektedirler ancak doğrudan bu şekilde 3B uygulamalarda kullanılamazlar. Görüntüleme veya fiziksel benzetim içeren uygulamalar gibi birçok kullanım için modellerin çokgen ağız yüzeyler (polygon meshes) ya da “Düzgün olmayan, rasyonel B-

Spline” (Non-Uniform Rational B-Spline, NURBS) gibi yüzey tanımlı şekilde olması tercih edilir. Dolayısıyla nokta bulutlarını genellikle yüzey tanımlı geometrik modellere dönüştürülmesi gerekir. Bu işleme yeniden yapılandırma (reconstruction) adı verilir (Şekil 1.4).



Şekil 1.3 Kalkış platformu üzerinde Endeavour uzay mekiğinin nokta bulutu



Şekil 1.4 Stanford tavşanı adlı nokta bulutu ve yeniden yapılandırma ile oluşturulan yüzey modeli

3B tarayıcılardan ve bunlardan elde edilen bilgisayar modellerinden yukarıda da dile getirildiği gibi birçok alanda faydalanılmaktadır. Aşağıda bir kaç alandaki kullanımları hakkında kısa bilgi verilmiştir. Verilen kesin, tam bir liste değildir. Tıp, yapı endüstrisi ve inşaat mühendisliği, kalite güvence, vb. daha birçok farklı daldan örnekler verilebilir.

Malzeme işleme ve üretimi: Lazer tarama, lazer teknolojisi kullanılarak bir yüzeyin örneklenmesi ya da taranması yöntemini tanımlar. Kullanılan lazerin gücünün ve tarama işleminin sonuçlarının farklı olduğu değişik kullanım alanları vardır. Genel olarak, düşük güce sahip lazerler yüzeyin etkilenmeyeceği işlemlerde kullanılır. Kullanılan lazerin gücü arttıkça yüzey ile etkileşim ve kullanım amacı da değişir. Biraz yüksek güç ile malzemenin biraz aşındırıldığı lazerle oyma, biraz daha yüksek güç kullanarak ise malzeme sıvılaştırılarak lazerle kaynak yapılabilir. Malzeme zarar görece kadar yüksek güçlerle de lazerle kesme işlemi gerçekleştirilir.

Eğlence sektörü: 3B tarayıcılar eğlence endüstrisinde hem filmler hem de video oyunları için 3B model yaratmada sıkça kullanılırlar. Gerçek dünyada bir eşi ya da benzeri olan bir modele gereksinim duyulduğunda, 3B modeli oluşturmak için bu gerçek nesneyi taramak, modelleme yazılımları kullanarak elle model oluşturmaktan çok daha hızlıdır. Hatta benzer bir gerçek nesne olmadığı durumlarda dahi çoğunlukla önce bir fiziksel model oluşturulup sonra bu modelin taranması yöntemine başvurulur.

Tersine mühendislik: Tersine mühendislik (Reverse Engineering) bir aygıtın, objenin veya sistemin yapısının, işlevinin veya çalışmasının, çıkarımcı bir akıl yürütme analiziyle keşfedilmesi işlemidir. Bu yöntem, genellikle orijinalin kopyalanmasını, onu bir araya getiren kısımları ve bir araya gelişlerini inceleyip anlayarak yeni bir aygıt, obje veya sistem üretilmesini hedefler. İnceleme ve üretim, çokgen ağyüzeyler, NURBS yüzeyleri şeklinde gösterilen dijital modeller üzerinde gerçekleştirilir.

Kültürel miras: Bugüne kadar, dokümantasyon ve inceleme amaçlarıyla tarihi yerleşkelerin ve nesnelerin tarandığı birçok araştırma gerçekleştirilmiştir. 3B tarama ve 3B yazdırma teknolojilerinin bir arada kullanılması gerçek nesnelerin otomatik olarak kopyalanmasına imkân sağlamaktadır. Bu işlem için geleneksel olarak uygulanan alçıya

koyma tekniğindeki en büyük problem tekniğin değerli ve hassas tarihi eserlere hasar vermesidir.Şekil 1.5'te solda görülen heykelcik 3B tarayıcı ile taranmış, elde edilen veriler yazılım ile işlenerek ekranda görülen model oluşturulmuştur. Oluşturulan bu model bir 3B yazdırma makinesine beslenerek sağda görülen reçine kopya elde edilmiştir.



Şekil 1.5 Örnek bir nesnenin 3B tarama ve 3B yazdırma ile oluşturulan kopyası

Tarihi eserlerin ve sanat eserlerinin 3B modellerinin elde edilmesinin bir kaç önemli amacı:

- 3B dijital kataloglar oluşturma,
- otomatik kopyalama,
- restorasyon planlama

olarak sıralanabilir. Dijital Mikelanjelo projesi (Şekil 1.6) (Levoyvd.2000), çivi yazısı tabletlerinin taranması (Şekil 1.7) (Kumar vd. 2003), yeniden doğan Roma projesi (Şekil 1.8) (Guidi vd. 2005) ve Pietà Projesi (Şekil 1.9) (Bernardini vd. 2002) büyük çaplı kültürel miras tarama çalışmalarının başlıca örnekleridir.

Bu ve benzer projelerin hepsinde hedeflenen detaylı tarama hassaslığı ve nesnelerin boyutları göz önüne alınarak, kullanılan tarayıcılar, konusunda uzman endüstriyel şirketlere yaptırılmıştır. Bu da tarama cihazlarının ve dolayısıyla projelerin maliyetinin çok yüksek rakamlara ulaşmasına sebep olmuştur.



Şekil 1.6Dijital Mikelanjelo projesi sonucunda oluşturulan sanal Davut heykeli modeli(Levoy vd. 2000)



Şekil 1.7Taranan çivi yazısı tabletlerin model görünümü (Kumar vd. 2003)



Şekil 1.8Çok detaylı bir eski Roma şehir maketinden tarama ile oluşturulan modelde Kolezyum vadisinin görüntüsü (Guidi vd. 2005)



Şekil 1.9Mikelanjelo'nun tarama ile oluşturulan Pietà heykeli modeli (Bernardini vd. 2002)

Bu çalışmadaki ana amaç 3B bir tarayıcı inşa ederek cisimlerin 3B sanal modellerini oluşturmaktır. 3B tarayıcılarla model elde etme işleminde çok farklı teknikler mevcuttur. İnşa edilecek 3B tarayıcının donanım parçalarının ve kullanılacak tekniğin belirlenmesinde taranacak nesnelerin özellikleri, tarama ortamı ve oluşturulacak modelin ne amaçla kullanılacağı büyük önem taşır. Bu çalışmada:

- laboratuvar ortamında taranabilecek boyutlarda,

- yüzeyi saydam veya parlak olmayan,
- fiziksel temas halinde zarar görebilecek

hassas tarihi eserlerin ya da sanat eserlerinin, 3B kataloglarda kullanıma yönelik sanal modellerini yaratmak amacıyla bir 3B tarayıcı inşa etmek hedeflenmektedir. Bu belirleme sonucunda araştırmanın yanıtlamayı planladığı ana soru istenilen model için yeterli (orta-yüksek derece) hassaslıkta, maliyeti makul olarak nitelendirilebilecek ve kolay temin edilebilir donanım parçaları kullanılarak, kullanımı kolay bir 3B tarama sisteminin tasarlanıp tasarlanamayacağıdır. Bu amaca yönelik doğru donanımları belirleyebilmek ve sistemin genel bir tasarımından bahsedebilmek için ilk adım çeşitli 3B tarama yöntemlerinin araştırılması olmuştur.

1.1 Tarayıcıların Sınıflandırılması

3B tarayıcılar tarama işleminin fiziksel dokunuş içerip içermemesine göre iki ana kategoriye ayrılır: temas içeren (ya da temasa dayalı) ve temas içermeyen. Temasa dayalı tarayıcılar, adından da anlaşılacağı üzere taranan cisme fiziksel olarak dokunarak şekil bilgisini elde eden tarayıcılardır (Şekil 1.10). Endüstride üretim ve montaj hatlarında kullanılan Koordinat Ölçme Makineleri (KÖM) temasa dayalı 3B tarayıcıların başlıca örnekleridir. Temasa dayalı tarayıcılar çok başarılı modeller meydana getirebilir, fakat nesneye fiziksel olarak dokundukları için yüzeyi değiştirebilme ya da zarar verebilme gibi bir dezavantajları vardır.



Şekil 1.10 Temasa dayalı bir tarayıcı örneği

Temas içermeyen tarayıcılar ise kendi içlerinde edilgen (pasif) ve etken (aktif) olarak iki alt grupta değerlendirilir. Edilgen tarayıcılar cismi veya ortamı tararken sadece

zaten ortamda var olan ışığın yansımalarını algılayan, kendisi herhangi bir ışınım yaymayan tarayıcılardır. Stereoskopik, fotometrik, ya da silüete dayalı tarayıcılar bu grupta yer alır.

Etken tarayıcıların ayırt edici özelliği genel olarak bir çeşit ışık yayma yaparak yansımaları algılamalarıdır. Işık yerine ses veya x-ışınlarıyla çalışan bazı çeşitleri de mevcuttur. Temasa dayalı olmayan etken tarayıcılarda en yaygın kullanılan tarama yöntemleri “uçuş süresi” (time-of-flight) ve “üçgenleme”(triangulation) yöntemleridir.

1.1.1 Uçuş süresi tarayıcıları

Bu 3B tarayıcılar nesneyi incelerken lazer ışık kullanan etken tarayıcılardandır. Lazer mesafe ölçerler bu sınıfın en yaygın örnekleridir (Şekil 1.11). Yüzeye olan mesafeyi, ışığın gidiş-gelişi ne kadar sürede gerçekleştirdiğini ölçerek hesaplarlar. Bir lazer ışığın etkinleştirildiği andan yansımanın algılayıcı tarafından görüldüğü ana kadar olan süre, t , ölçülür. Işık hızı, c , bilinen bir sabit olduğu için, gidiş-geliş süresi, t , ışığın kat ettiği mesafenin – ki bu da tarayıcı ile yüzey arasındaki mesafe d ’nin iki katıdır – hesaplanmasını sağlar.



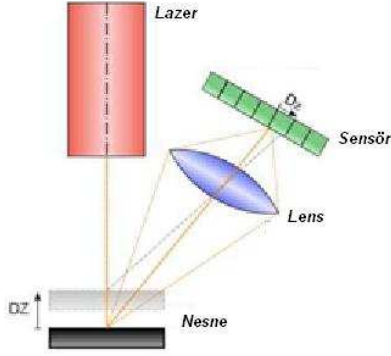
Şekil 1.11 Bir lazer mesafe ölçer

Yüzeye olan mesafe $d = (c \times t)/2$ ile hesaplanır. Işık hızı çok yüksek bir değer olduğu için (1mm'lik mesafeyi yaklaşık 3,3ps'de kat eder) t 'nin ölçülmesindeki hassasiyet uçuş süresi tarayıcılarının doğru çıkarım yapmalarına yönelik en önemli faktördür. Lazer mesafe ölçerler görüş açıları içerisinde yalnızca tek bir noktaya olan mesafeyi ölçebilirler. Dolayısıyla görüş açısının tamamını tarayabilmek için yönü değiştirilerek tek tek farklı noktalar taranır. Bunun için mesafe ölçerin kendisi döndürülebileceği gibi hareket ettirilebilir aynalardan oluşan bir sistem de kullanılabilir. Genellikle aynalar daha hafif oldukları ve daha isabetli bir şekilde daha kolay döndürülebilecekleri için ikinci yöntem tercih edilmektedir. Bugün kullanılan tipik bir mesafe ölçer saniyede 10.000 ~ 100.000 noktaya olan mesafeyi ölçebilir. Uçuş süresi tarayıcılarının en büyük avantajı kilometreler ile ifade edilebilecek çok uzak mesafelerden dahi çalışabilmeleridir. Bu özellikleri, uçuş süresi tarayıcılarını, binalar ve coğrafi şekiller gibi büyük yapıların taranmasına uygun kılmaktadır.

1.1.2 Üçgenleme tarayıcıları

Üçgenleme tarayıcıları da (üçgenleme tabanlı/üçgensel tarayıcılar) ortamı veya nesneyi araştırmak için genellikle lazer ışık kullanan etken tarayıcılardır. Bu yöntemle çalışan

tarayıcılar taranacak nesneye ışığı gönderip, ışığın cismin üzerine düştüğü noktanın yerini ayrı bir algılayıcı (sensor) ile saptar(Şekil 1.12). Işığın cismin üzerine düştüğü noktanın uzaklığıyla bağlantılı olarak bu noktanın kameranın görüş alanı içerisindeki yeri değişecektir. Bu yöntemle üçgenleme adı verilmesinin nedeni lazer nokta, kamera ve lazer ışık kaynağının bir üçgen oluşturmasıdır.



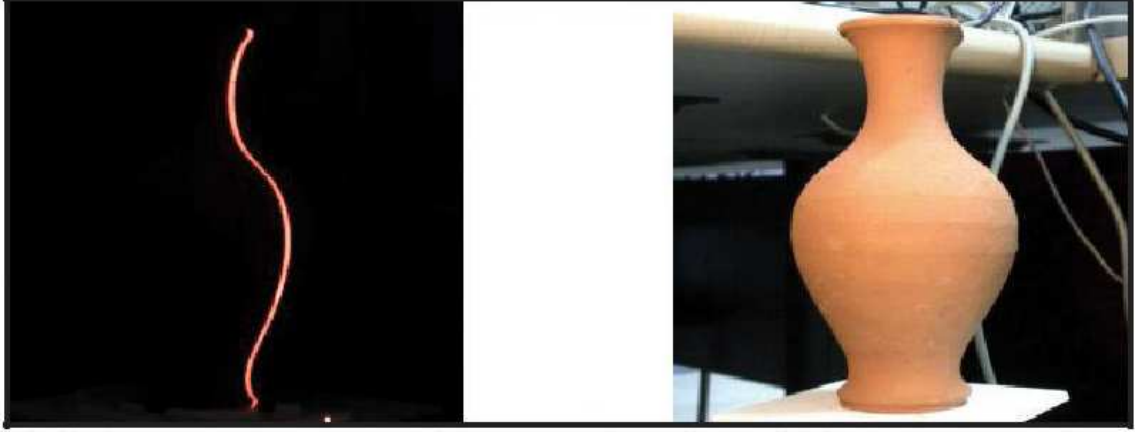
Şekil 1.12Üçgenleme tabanlı tarama

Üçgenleme tarayıcıları uçuş süresi tarayıcılarına kıyasla çok daha sınırlı menzile sahiptirler ancak kısa mesafeli taramalardaki hassaslıkları daha yüksektir. Uçuş süresi tarayıcılarında hassas mesafe ölçümü ancak milimetre seviyesinde kalabilmekteyken üçgenleme tarayıcılarında bu hassasiyet bir mikrometrenin($1\mu m$) 10 katı, yani $10^{-5}m$ seviyelerinde olabilmektedir. Daha da ötesi, üçgenleme tarayıcıları noktasal bir ışık ile bir seferde bir nokta taramak yerine çizgisel (1B) ya da düzlemsel (2B) bir ışık biçimi (çizgi lazer ya da paralel ve/veya kesişen lazer çizgiler serisi) kullanılarak tarama süresi açısından da geliştirilebilir. Bu şekilde geliştirilmiş üçgenleme tarayıcılarına biçimli ışık tarayıcıları adı verilmektedir.

1.1.3 Biçimli ışık tarayıcıları

Biçimli ışık kullanan tarayıcılar nesnenin üzerine belli bir biçimdeki ışığı yansıtır ve biçimdeki bozulmaları çözümler (Şekil 1.13). Kullanılan biçimlere örnek bir lazer çizgi (1B) veya paralel ve/veya kesişen lazer çizgi dizisi (2B) olabilir. Nesnenin üzerine düşürülen ışık biçimi üzerinden üçgenleme tarayıcılarındaki yöntemlerle derinlik bilgisi

elde edilebilir. Nokta lazer yerine çizgi lazer kullanımı işlemi pek deęiřtirmedięi veya karmařıklařtırmadıęı gibi tam tersine bir seferde birok noktayı (hatta btn grř aısını) tarayabilmek iřleme byk hız kazandırır.



řekil 1.13 izgisel lazer ıřık kullanan biimli ıřık tarama sistemi

zellikle 2B biimli ıřık kullanan teknolojiler zerinde arařtırmalar giderek yoęunlařmaktadır. Bu tip tarayıcılarda biimdeki bozulmaları incelemek ve doęru derinlik bilgilerini elde edebilmek olduka karmařık algoritmalar gerektirmektedir. Bunun sebebi benzer bozulmaların farklı řekillerde oluřabilmesidir. rnek olarak paralel dik lazer izgilerle tarama yapan bir sistem dřnelim. En basit durumlarda grntdeki lazer sırası gerek sıra olarak kabul edilebilir; en soldaki izgi birinci lazer, yanındaki ikinci lazer, vs. řeklinde. Ancak zerinde delikler veya ıkıntılar gibi ani derinlik deęiřimleri bulunan nesnelerin taranması durumlarında bu sıralama iře yaramayacaktır. Zira izgilerin grntye girmemesi veya sırasının farklı grntlenmesi gibi durumlar ortaya ıkabilir. ok izgili lazer genleme (Multi-stripe laser triangulation) yntemleri bunun gibi problemlerle bařa ıkabilecek algoritmalar gerektirmektedir.

Bu alıřmada geliřtirilmek istenen tarayıcı daha nce de belirtildięi gibi tarihi eserlerin laboratuvar ortamında taranmasında kullanılacaęı dřnlerek tasarlanmaktadır. Dolayısıyla taranan nesnelerin zarar grmemesi iin temas iermeyen, izgi lazer ıřık kullanan, etken bir biimli ıřık tarayıcısı tasarlanmasına karar verilmiřtir. Aslına

bakılırsa tarihi eserlerin taranmasında temasa dayalı olmayan tarayıcıların kullanılması şimdiye kadar yapılan çalışmalarda da tercih edilmiştir (Levoy vd. 2000, Kumar vd. 2003, Yıldırım vd. 2006). Bu çalışmalardan örnekler vererek tarayıcıların uygulamalarda kullanımına değinelim.

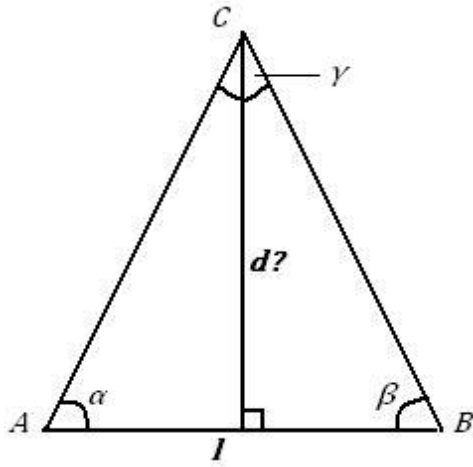
1999 yılında iki ayrı araştırma grubu Mikelanjelo'nun heykellerini taramaya başlamıştır. Stanford üniversitesinde Marc Levoy tarafından yönetilen grup, Cyberware şirketinin ürettiği üçgen tabanlı bir lazer tarayıcı kullanarak Mikelanjelo'nun (en önemlisi Davut heykeli olmak üzere) Floransa'da bulunan heykellerinin sanal modellerini elde etmiştir. Stanford'daki bu projeler için firmanın istenen özelliklere uygun ürettiği tarayıcı yaklaşık 7m yüksekliğindeki Davut heykelini zarar verilmeyecek güvenli bir mesafeden tarayabilecek kadar gelişmiş ve bir o kadar da maliyeti yüksek bir tarayıcı olmasına rağmen çalışma şekli yine üçgen tabanlı lazerli tarama teknolojisine dayanır. 0.25mm'de bir veri noktası alınarak yapılan tarama sanatçının keski izlerinin dahi görülebileceği kadar detaylı bir sonuç getirmiştir. Bu detaylı tarama çok fazla veri (yaklaşık 32 GB) ortaya çıkarmış ve işlenmesi beş ay kadar sürmüştür. Aynı dönemde IBM'den başka bir araştırma grubu da yine Floransa'daki Pietá'yı tarayarak hem geometrik hem de renksel detaylarıyla sanal ortama aktarmışlardır.

Bu çalışmaların yapıldıkları tarihlerden bugüne Kolezyum, Zeus heykeli ve Olimpos'un sanal modelleri, Dijital Minerva projesi ve sayılamayacak daha birçok çalışma hassas tarihi kültürel değerlerin zarar görmeden kitlelere hızlı ve kolay bir şekilde tanıtımlarının yapılabilmesini ve normalde gerçekleştirilemeyecek birçok bilimsel çalışmanın sanal kopyalar üzerinde gerçekleştirilmesini mümkün kılmıştır.

2. KURAMSAL TEMELLER

2.1 Üçgenleme Prensibi

Üçgenleme bugün navigasyon, meteoroloji, gök ölçüm, füze bilimi ve silah hedef belirleme gibi birçok alanda uygulaması olan, esasen kökeni eski Yunan'a kadar uzanan geometrik ve trigonometrik bir tekniktir. İşlem, özetle, bir noktanın yerinin, mesafeleri ölçmektense, belirli bir taban çizgisi ile bu çizgi üzerinde alınan iki noktadan noktaya uzanan doğruların yaptığı açılar ölçülmesine dayanır. Diğer bir deyişle, aranan nokta bir kenarı ve iki açısı bilinen bir üçgenin üçüncü köşesi olarak belirlenir (Şekil 2.1)(Hartley ve Sturm 1997).



Şekil 2.1 Genel üçgenleme kavramı

İstenen nokta C 'nin koordinatları ve o noktaya olan uzaklık d , bir kenarın uzunluğu, l , ve iki köşedeki açılar, α , β , ile hesaplanabilir.

$$l = \frac{d}{\tan \alpha} + \frac{d}{\tan \beta}$$

Dolayısıyla

$$d = l / \left(\frac{1}{\tan \alpha} + \frac{1}{\tan \beta} \right)$$

Yine aynı düşünce üzerinden farklı bir hesaplama ise şu şekilde özetlenebilir (Gibilisco ve Crowhurst 2007). ABC üçgeni üzerinden sinüs teoremine göre:

$$\frac{\sin \alpha}{BC} = \frac{\sin \beta}{AC} = \frac{\sin \gamma}{AB}$$

AB uzunluğu l bilindiği için üçgenin diğer kenarları $AC = l \times \frac{\sin \beta}{\sin \gamma}$ ve $BC = l \times \frac{\sin \alpha}{\sin \gamma}$ eşitliklerinden elde edilebilir. C noktasının AB doğru parçasına uzaklığı, yani d uzunluğu:

$$d = AC \times \sin \alpha = BC \times \sin \beta$$

Bir önceki eşitlikleri kullanarak:

$$d = \frac{l \times \sin \alpha \times \sin \beta}{\sin \gamma}$$

ABC üçgen olduğu için $\gamma = \pi - \alpha - \beta$ 'dir. Trigonometrik olarak $\sin \theta = \sin(\pi - \theta)$ olduğu için yukarıdaki denklem şu şekilde yazılabilir:

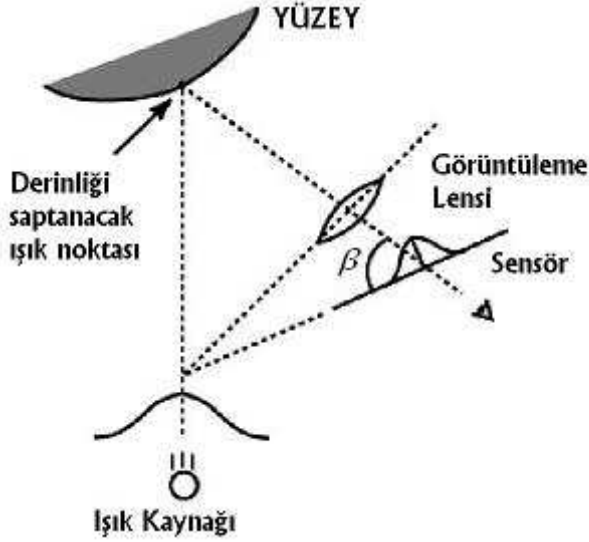
$$d = \frac{l \times \sin \alpha \times \sin \beta}{\sin(\alpha + \beta)}$$

Üçgenleme prensibinin temel hesaplaması olan bu eşitlik incelenirse taban doğru parçasının uzunluğu ve iki yandaki açılar bilindiği takdirde d uzunluğunun hesaplanabileceği görülür.

Gerçek cisimlerin 3B sanal modellerini oluşturmak amacıyla üçgenleme tekniği kullanan bir tarayıcıya şematik olarak bakılırsa (Şekil 2.2) ışık kaynağı, algılayıcı ve nesnenin, sırasıyla Şekil 2.1'deki A , B ve C noktalarına karşılık geldiği görülmektedir. Taban kenarının uzunluğu, yani algılayıcı ile ışık kaynağı arasındaki mesafe ve ışık kaynağının bulunduğu köşedeki açı bu cihazlar yerleştirilirken belirlenebilir. Üçgenleme prensibinin uygulanması için geriye sadece algılayıcının bulunduğu köşedeki açı β kalmaktadır.

Ölçülebildiği takdirde β hesaplanmak istenen yüzey uzaklığının bulunmasını sağlayacaktır. Üçgenleme tarayıcılarının çalışma mekanizmalarında bu noktada yapılacak önemli bir gözlem yatar. Yüzeyin uzaklığı değiştikçe β açısı değişirken algılayıcıdan elde edilen görüntüde ışığın belirttiği resim öğesinin (pikselin) yeri de orantılı olarak değişmektedir. Bu gözlemden yola çıkarak açının hesaplanması yerine

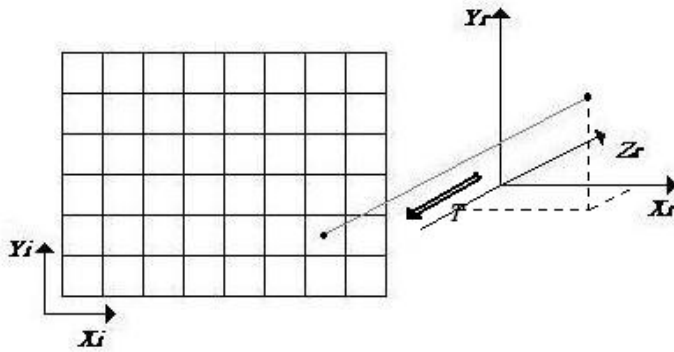
görüntüdeki resim ögesi pozisyonlarıyla yüzeyin konumu arasındaki ilişkinin doğasının belirlenmesi yüzeyin 3B konumunun belirlenmesini sağlar.



Şekil 2.2 Üçgen tabanlı tarama tekniğinin şematik modeli

2.2 Resim Ögesi Konumları ile 3B Koordinatlar Arasındaki İlişki

Tarama düzeneğinde ışık kaynağı ve algılayıcı yerleştirildikten sonra sabit kalmaktadır. Dolayısıyla algılayıcı ile elde edilen görüntü düşünülürse yansıyan ışığı gösteren resim öğelerinin 2B görüntü düzlemi üzerindeki konumları ile ışığın düştüğü noktanın 3B koordinatları arasında bir ilişki vardır. Bu ilişki, varış kümesinin üçüncü boyutu sabitlenmiş $T: \mathbb{R}^3 \rightarrow \mathbb{R}^3$ şeklinde bir fonksiyon olarak düşünülebilir (Şekil 2.3). İlişkiyi etkileyen etmenler de fonksiyonun parametrelerini oluşturmaktadır.

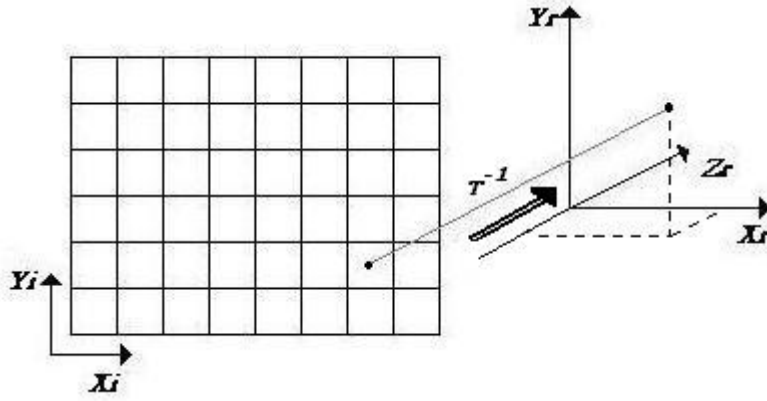


Şekil 2.3 Dünya koordinatlarından (X_r, Y_r, Z_r) görüntü koordinatlarına $(X_i, Y_i, 1)$ fonksiyon T

T fonksiyonunun karşılık geldiği algılayıcı ile görüntü alma işleminde nesne yüzeyine düşürülen ışığın hangi resim ögesine karşılık geleceğini gerçekte etkileyen sayılamayacak kadar fazla yapısal ve yardımcı parametre vardır. Bunlardan ilk akla gelenler:

- Nesne yüzeyinde ışık düşen noktanın konumu ve bu noktada yüzey normali,
- Işık kaynağı ve algılayıcının konumları ve yönelimleri,
- Kullanılan ışığın dalga boyu, ıraksama miktarı gibi yapısal özellikleri
- Nesne yüzeyinin rengi, saydamlığı gibi yapısal özellikleri
- Algılayıcının odak uzaklığı, mercek sapıncı gibi yapısal özellikleri olarak sayılabilir.

Tarayıcının işleyişinde ise görüntü alma işleminin tersi gerçekleştirilmektedir. Yani hesaplanmak istenen fonksiyon aslında T^{-1} 'dir (Şekil 2.4).



Şekil 2.4 Görüntü koordinatlarından $(X_i, Y_i, 1)$ dünya koordinatlarına (X_r, Y_r, Z_r) fonksiyon T^{-1}

T fonksiyonunun parametreleri T^{-1} fonksiyonu için de etkindir. Bu parametrelerin çoğunluğu sabit olmasına rağmen kuramsal açıdan tam olarak hesaplanması mümkün değildir. Bu yüzden sayısal yöntemlerle yaklaşık bir dönüşüm fonksiyonu $\Phi \cong T^{-1}$ hesaplanmaktadır.

2.3 Yaklaşık Dönüşümün Hesaplanması

Görüntü koordinatlarından gerçek dünya koordinatlarına gönderim yapan T^{-1} fonksiyonuna yaklaşan bir Φ fonksiyonu varsayalım (Teutsch 2007). En Küçük

Kareler(EKK) kistası kullanılarak bu fonksiyona bir yakınsama gerçekleştirilecektir.

$\Phi: \mathbb{R}^2 \rightarrow \mathbb{R}^3$, 3×3 'lük bir dönüşüm matrisi

$$\Phi = \begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix}$$

olarak tanımlanırsa 2B görüntü düzleminde tanımlanan homojen koordinatlara sahip

nokta $r = \begin{pmatrix} a \\ b \\ 1 \end{pmatrix}$ 'nin 3B karşılığı olan $p = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ 'ye dönüşümü

$$\Phi r = \begin{pmatrix} \phi_{11}a + \phi_{12}b + \phi_{13} \\ \phi_{21}a + \phi_{22}b + \phi_{23} \\ \phi_{31}a + \phi_{32}b + \phi_{33} \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = p$$

olarak yazılabilir.

ϕ_{ij} katsayı değerlerini hesaplamak için tarama sistemindeki ı ışık ile düzlem gibi basit bir geometrik şekle sahip ayarlama yüzeyi koordinatlarının bilindiği noktalarda kesiştirilerek bir örnek nokta eşleşmeleri kümesi

$$Z = \{(r_k, p_k) | r_k \in \mathbb{R}^2, p_k \in \mathbb{R}^3, T(p_k) = r_k\}$$

oluşturulduğunu düşünelim. Bu örnek eşleşmeler kümesi ve EKK kıstası ile ϕ_{ij} değerlerine dolayısıyla Φ fonksiyonuna bir yakınsama yapılır.

Işığın, koordinatları bilinen m farklı $p_i \in \mathbb{R}^3$ üzerine düşürülmesiyle karşılık gelen $r_i \in \mathbb{R}^2$ 'ler bulunmuş olsun, yani $|Z| = m$. Eğer d fonksiyonunun iki nokta arasındaki uzaklığı verdiği düşünülürse EKK ile yakınsama esnasında en aza indirgenecek kareler toplamı şöyle olur:

$$\begin{aligned} Sum &= \sum_{k=1}^m d(\Phi r_k, p_k)^2 \\ Sum &= \sum_{k=1}^m ((\phi_{11}a_k + \phi_{12}b_k + \phi_{13} - x_k)^2 + (\phi_{21}a_k + \phi_{22}b_k + \phi_{23} - y_k)^2 \\ &\quad + (\phi_{31}a_k + \phi_{32}b_k + \phi_{33} - z_k)^2) \end{aligned}$$

EKK kıstasına göre devam edildiğinde (Giordanov vd. 2003) kısmi türevlerin 0 olması için şu eşitlikler elde edilir:

$$\frac{\partial Sum}{\partial \phi_{11}} = 2 \sum_{k=1}^m a_k (\phi_{11}a_k + \phi_{12}b_k + \phi_{13} - x_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{12}} = 2 \sum_{k=1}^m b_k (\phi_{11} a_k + \phi_{12} b_k + \phi_{13} - x_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{13}} = 2 \sum_{k=1}^m (\phi_{11} a_k + \phi_{12} b_k + \phi_{13} - x_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{21}} = 2 \sum_{k=1}^m a_k (\phi_{21} a_k + \phi_{22} b_k + \phi_{23} - y_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{22}} = 2 \sum_{k=1}^m b_k (\phi_{21} a_k + \phi_{22} b_k + \phi_{23} - y_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{23}} = 2 \sum_{k=1}^m (\phi_{21} a_k + \phi_{22} b_k + \phi_{23} - y_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{31}} = 2 \sum_{k=1}^m a_k (\phi_{31} a_k + \phi_{32} b_k + \phi_{33} - z_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{32}} = 2 \sum_{k=1}^m b_k (\phi_{31} a_k + \phi_{32} b_k + \phi_{33} - z_k) = 0$$

$$\frac{\partial Sum}{\partial \phi_{33}} = 2 \sum_{k=1}^m (\phi_{31} a_k + \phi_{32} b_k + \phi_{33} - z_k) = 0$$

Bu eşitliklerden şu sonuç çıkarılabilir. Φ ile tahmin edilen noktalar Φr_k ve gerçek noktalar p_k arasındaki mesafelerin karelerinin toplamını en aza indirmeye çalışmak ayrı ayrı x, y ve z koordinatları arasındaki mesafelerin karelerinin toplamalarını en aza indirmeye çalışmaktan farksızdır. Diğer bir deyişle EKK kistasına göre yakınsama yapılırken farklı satırlardaki ϕ_{ij} katsayıları birbirinden bağımsız alınarak bunlara ayrı ayrı yakınsama uygulanabilir. Yukarıdaki eşitlikler yeniden düzenlendiğinde birbirinden bağımsız olarak düşünülebilecek 3 ayrı lineer eşitlik sistemi elde edilir. ϕ_{1j} katsayıları için ilk sistem L_1 :

$$\phi_{11} \sum_k a_k^2 + \phi_{12} \sum_k a_k b_k + \phi_{13} \sum_k a_k = \sum_k a_k x_k$$

$$\phi_{11} \sum_k a_k b_k + \phi_{12} \sum_k b_k^2 + \phi_{13} \sum_k b_k = \sum_k b_k x_k$$

$$\phi_{11} \sum_k a_k + \phi_{12} \sum_k b_k + \phi_{13} m = \sum_k x_k$$

ϕ_{2j} katsayıları için ikinci sistem L_2 :

$$\phi_{21} \sum_k a_k^2 + \phi_{22} \sum_k a_k b_k + \phi_{23} \sum_k a_k = \sum_k a_k y_k$$

$$\phi_{21} \sum_k a_k b_k + \phi_{22} \sum_k b_k^2 + \phi_{23} \sum_k b_k = \sum_k b_k y_k$$

$$\phi_{21} \sum_k a_k + \phi_{22} \sum_k b_k + \phi_{23} m = \sum_k y_k$$

Ve ϕ_{3j} katsayıları için üçüncü sistem L_3 :

$$\phi_{31} \sum_k a_k^2 + \phi_{32} \sum_k a_k b_k + \phi_{33} \sum_k a_k = \sum_k a_k z_k$$

$$\phi_{31} \sum_k a_k b_k + \phi_{32} \sum_k b_k^2 + \phi_{33} \sum_k b_k = \sum_k b_k z_k$$

$$\phi_{31} \sum_k a_k + \phi_{32} \sum_k b_k + \phi_{33} m = \sum_k z_k$$

halini alır. Bu sistemleri matris formuna dönüştürmek için

$$M = \begin{pmatrix} \sum_k a_k^2 & \sum_k a_k b_k & \sum_k a_k \\ \sum_k a_k b_k & \sum_k b_k^2 & \sum_k b_k \\ \sum_k a_k & \sum_k b_k & m \end{pmatrix}$$

ve

$$R_1 = X = \begin{pmatrix} \sum_k a_k x_k \\ \sum_k b_k x_k \\ \sum_k x_k \end{pmatrix}, R_2 = Y = \begin{pmatrix} \sum_k a_k y_k \\ \sum_k b_k y_k \\ \sum_k y_k \end{pmatrix}, R_3 = Z = \begin{pmatrix} \sum_k a_k z_k \\ \sum_k b_k z_k \\ \sum_k z_k \end{pmatrix}$$

olarak alınırsa her $L_i, i \in \{1, 2, 3\}$ kısaca

$$L_i; M \begin{pmatrix} \phi_{i1} \\ \phi_{i2} \\ \phi_{i3} \end{pmatrix} = R_i$$

şeklinde yazılabilir.

Hesaplanmak istenen deęerler ϕ_{ij} olduęu iin matris eřitlikleri yeniden dzenlenirse:

$$L_i: \begin{pmatrix} \phi_{i1} \\ \phi_{i2} \\ \phi_{i3} \end{pmatrix} = M^{-1} R_i$$

elde edilir. M^{-1} 'i hesaplamak iin nce $|M|$ (M matrisinin determinantı) bulunur:

$$|M| = m \sum_k a_k^2 \sum_k b_k^2 + 2 \sum_k a_k b_k \sum_k b_k \sum_k a_k - \sum_k a_k^2 \left(\sum_k b_k \right)^2 - m \left(\sum_k a_k b_k \right)^2 - \left(\sum_k a_k \right)^2 \sum_k b_k^2$$

Daha sonra M matrisinin eřarpan (cofactor) matrisi $C_M = \begin{pmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{pmatrix}$ 'i

oluřturan C_{ij} 'ler řu řekilde hesaplanır:

$$C_{11} = m \sum_k b_k^2 - \left(\sum_k b_k \right)^2$$

$$C_{12} = C_{21} = \sum_k a_k \sum_k b_k - m \sum_k a_k b_k$$

$$C_{13} = C_{31} = \sum_k a_k b_k \sum_k b_k - \sum_k a_k \sum_k b_k^2$$

$$C_{22} = m \sum_k a_k^2 - \left(\sum_k a_k \right)^2$$

$$C_{23} = C_{32} = \sum_k a_k b_k \sum_k a_k - \sum_k a_k^2 \sum_k b_k$$

$$C_{33} = \sum_k a_k^2 \sum_k b_k^2 - \left(\sum_k a_k b_k \right)^2$$

M 'nin determinantı ve eřarpan matrisinin devrięi (transpose) kullanılarak:

$$M^{-1} = \frac{1}{|M|} C_M^T$$

zerinden ϕ_{ij} 'ler hesaplanırsa ařaęıdaki eřitlikler yazılabilir:

$$\phi_{11} = \left(C_{11} \sum_k a_k x_k + C_{21} \sum_k b_k x_k + C_{31} \sum_k x_k \right) / |M|$$

$$\begin{aligned}
\phi_{12} &= \left(c_{12} \sum_k a_k x_k + c_{22} \sum_k b_k x_k + c_{32} \sum_k x_k \right) / |M| \\
\phi_{13} &= \left(c_{13} \sum_k a_k x_k + c_{23} \sum_k b_k x_k + c_{33} \sum_k x_k \right) / |M| \\
\phi_{21} &= \left(c_{11} \sum_k a_k y_k + c_{21} \sum_k b_k y_k + c_{31} \sum_k y_k \right) / |M| \\
\phi_{22} &= \left(c_{12} \sum_k a_k y_k + c_{22} \sum_k b_k y_k + c_{32} \sum_k y_k \right) / |M| \\
\phi_{23} &= \left(c_{13} \sum_k a_k y_k + c_{23} \sum_k b_k y_k + c_{33} \sum_k y_k \right) / |M| \\
\phi_{31} &= \left(c_{11} \sum_k a_k z_k + c_{21} \sum_k b_k z_k + c_{31} \sum_k z_k \right) / |M| \\
\phi_{32} &= \left(c_{12} \sum_k a_k z_k + c_{22} \sum_k b_k z_k + c_{32} \sum_k z_k \right) / |M| \\
\phi_{33} &= \left(c_{13} \sum_k a_k z_k + c_{23} \sum_k b_k z_k + c_{33} \sum_k z_k \right) / |M|
\end{aligned}$$

Bu denklemler elimizdeki bilinmeyen dokuz katsayının, ϕ_{ij} , bir örnek eşleşim kümesi $\mathbf{Z}$ 'deki noktalardan hesaplanmasını sağlar. Sonrasında bu katsayılar kullanılarak 2B görüntü düzlemine izdüşümünün koordinatları bilinen bir noktanın 3B uzaydaki koordinatları kestirilebilir. Kısacası, bilinen bir grup 3B koordinat ve onlara karşılık gelen 2B resim ögesi pozisyonlarına dayanarak, verilen resim ögesi pozisyonlarından yaklaşık 3B koordinatlar tahmini yapan bir fonksiyon oluşturulmuştur.

2.4 Aydınlik Resim Öğelerinin Belirlenmesi

Gerek Φ fonksiyonunun katsayılarının hesaplanmasında gerekse sonrasındaki tarama işleminde yüzey üzerine düşen ışığa karşılık gelen aydınlık resim öğelerinin belirlenmesi gerekmektedir. Bunun için elde edilen görüntülere yaygın olarak kullanılan bir dizi görüntü işleme (image processing) tekniği uygulanır.

Uygulanacak ilk işlem renkli görüntülerin yeşinlik (intensity) görüntülerine dönüştürülmesidir. Renkli görüntünün Kırmızı-Yeşil-Mavi (KYM, İng. RGB) renk uzayı kullanılarak tanımlandığını düşünelim. KYM görüntüler yeşinlik görüntülerine çevrilirken renk değerlerinin ağırlıklı toplamaları kullanılır. Eğer bir resim ögesi için K kırmızı değerini, Y yeşil değerini, M de mavi değerini belirtiyorsa bu işlem görüntüdeki her resim ögesi için bir

$$I = K \times w_K + Y \times w_Y + M \times w_M$$

değerinin hesaplanması anlamını taşır. Kullanılan w_K, w_Y, w_M ağırlıkları işlemin uygulanacağı görüntülerin renk dağılımlarına göre değiştirilebilir. Örneğin Matlab hesaplama program paketinde bulunan Görüntü İşleme Araç Kutusu (Image Processing Toolbox) dâhilinde bu işlemi yapan hazır fonksiyon *rgb2gray*

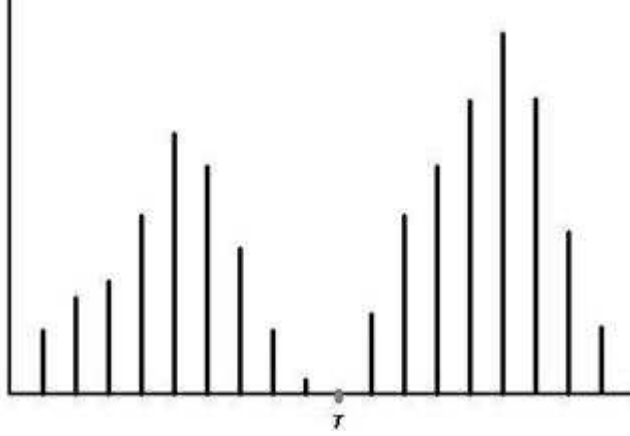
$$w_K = 0,2989, w_Y = 0,587, w_M = 0,114$$

değerlerini kullanmaktadır. Tarayıcı sistemde alınan görüntülerdeki renk dağılımı ve ayırt edilmek istenen ışığın özelliklerine göre ağırlıklar belirlenecektir.

Sıradaki işlem elde edilen yeşinlik görüntüsünden, kullanılan ışığın gözleendiği resim öğelerinin 1, geriye kalanların 0 ile gösterildiği ikili (binary) görüntünün elde edilmesidir. Bu dönüşüm görüntü işlemede çok yaygın olarak uygulanan bir dönüşümdür ve hemen hemen her örnekte eşikleme (thresholding) tekniği uygulanmaktadır. Bu işlem yeşinlik değerleri aralığında bir eşik değeri (T) belirlenmesine ve sahip olduğu değer eşik değerinden büyük olan resim öğeleri için değerin 1, küçük olanlar için ise değerin 0 ile değiştirilmesine dayanır. Diğer bir deyişle, yeşinlik görüntüsü $I(a, b)$ 'nin histogramı şekil 2.5'te verilen histogram, seçilen eşik de histogramda gösterilen T olursa, eşikleme sonucunda elde edilen ikili görüntü

$$B(a, b) = \begin{cases} 1 & I(a, b) \geq T \text{ ise} \\ 0 & I(a, b) < T \text{ ise} \end{cases}$$

şeklinde tanımlanır.



Şekil 2.5 Yeğlilik değerleri için bir histogram örneği

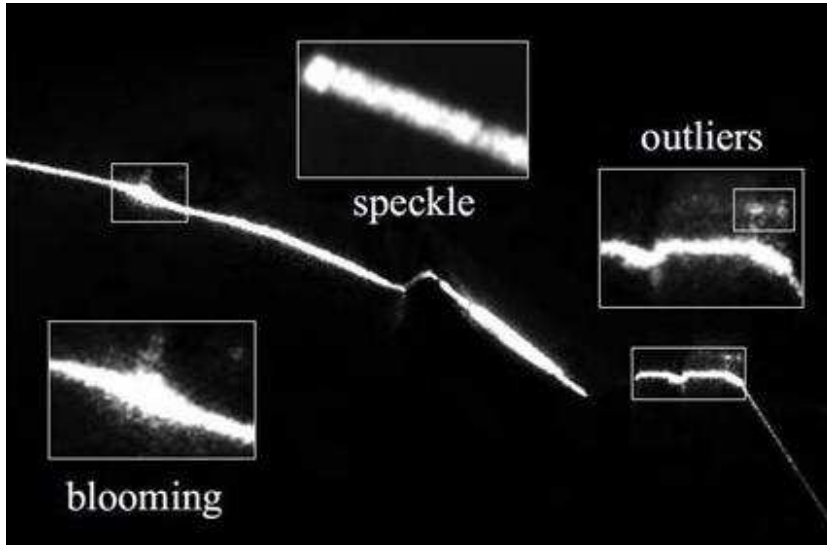
Geliştirilecek tarama sistemi için eşikleme işlemi sonucunda elde edilen ikili görüntüde değeri 1 olan resim öğeleri aydınlık noktaları göstermektedir. Bu basit eşikleme işleminin can alıcı noktası eşik değeri ve nasıl seçildiğidir. Elde edilen görüntüleri inceleyerek el ile seçim yapmak en basit yaklaşımdır. Çok farklı görüntüleri işleyecek sistemlerde eşikğin görüntüye göre otomatik belirlenmesi tercih edilmektedir. Tüm resim için sabit (küresel, İng. global) bir eşik değeri kullanmak yerine görüntünün farklı bölgeleri için farklı eşik değerleri hesaplayıp uygulayan (yerel eşikleme) algoritmalar da mevcuttur. Bu algoritmalar özellikle görüntüdeki bölgesel aydınlık etkileri ile başa çıkmak için uygundurlar. Buradaki çalışmada eşiklenecek görüntüler yeğlilik dağılımı açısından hemen hemen tamamen aynı özellikleri taşıyacaklardır. Dolayısıyla örnek birkaç görüntü incelenerek el ile eşik değeri seçilmesi daha uygundur. Yerel eşikleme algoritmaları da kullanılmayacaktır çünkü zaten görüntülerde ayırt edilmek istenen yüksek yeğlilikli resim öğeleri aynı yerel bölgelerde toplanacaktır. Bölüm 3.4’te tarama sisteminden alınan görüntüler üzerinde bu işlemlerin uygulanması gösterilmiştir.

2.5 Nokta Bulutundaki Hatalar

Üçgen tabanlı 3B bir tarayıcıdan elde edilen veri nesne yüzeyini temsil eden bir 3B noktalar kümesi şeklindedir. Bölüm 1’de de belirtildiği gibi bu 3B nokta kümesine nokta bulutu adı verilmektedir. Tarayıcıdan ilk olarak elde edilen nokta bulutu görüntüdeki gürültü ve başka sebeplerden ötürü hatalar içermektedir. Dolayısıyla hem nokta bulutu elde edilirken uygulanan ara işlemlerde hem de ilk oluşturulan nokta

bulutu üzerinde bu hataları gidermek için bazı yöntemler uygulanabilir. Uygulanabilecek yöntemlerden önce bahsi geçen bozulmalara yol açan gürültü ve etkenleri inceleyelim.

Birçok örnekte gözlemlenen bir olgu hatalı tarama verilerinin ana sebebinin istenmeyen yansımalar olmasıdır. Yüzeyin parlak kısımları kameraya çok fazla ışık gelmesine ve parıldama etkisi (blooming effect) adı verilen probleme yol açar. Hatalara sebep olan diğer sebeplerden bazıları yansıyan ışının algılayıcıya ulaşmaması, köşelerde kısmi yansıma olması, kendi kendini örten (self-occluding) cisimlerde çoklu yansımalar ve gölge etkileri olarak sayılabilir (Şekil 2.6). Bunların yanı sıra, 3B nokta bulutu, 2B görüntü işlemede örtüşme (aliasing) etkileri ve benek (speckle) etkisinden kaynaklanan gürültüler de içerir. Özet olarak, tarama işleminin ilk aşamalarında aşağıdaki etkiler nokta bulutu çıkarımını karmaşıktırarak sonuçlar doğurmaktadırlar:



Şekil 2.6 Görüntülerde karşılaşılan optik etkiler

Parıldama etkisi: Bu etki genel olarak tüm kameralar için geçerli olan bir görüntü bozulmasıdır. Problemin fiziksel kaynağı temel olarak merceklerin asla mükemmel odaklama yapamamasıdır. Normal koşullarda fark edilmeyen bozulmalar çok parlak bir ışık altında görünür hale gelir. CCD (Charge Coupled Device) algılayıcısına çok fazla ışık geldiğinde yongadaki hücreler üretilen enerjiyi saklayamaz ve fazla gelen enerji

komşu hücrelere aktarılır. Uygulamadaki sonuç parlak ışığın kendi doğal sınırları dışına taşması olarak gözlemlenir.

Benek etkisi: Bu etki lazer ışığın eşevreliliğinden (coherence) kaynaklanır. Lazer ışınımı sabit evre ilişkisi dolayısıyla çok küçük parçacıklar (dalga boyuyla aynı büyüklük mertebesindeki nesneler) tarafından dağıtıldığında girişime (interference) uğrar (Presnyakov ve Shchepinov 1997). Her nesnede çok küçük yüzey yapıları veya toz parçacıkları bulunduğu için hem yapıcı hem de yıkıcı girişim gözlemlenebilir (Şekil 2.6).

Gölge etkileri: Taranan nesne karmaşık bir geometriye sahip olduğunda bu etki görülebilir (Şekil 2.7). Delikler ve kendi kendini örten kısımlar kameranın lazer çizgiyi gözlemlemesini engelleyebilir. Sonuç olarak geometrik bilgi kaybına uğramış eksik bir nokta bulutuna yol açar.



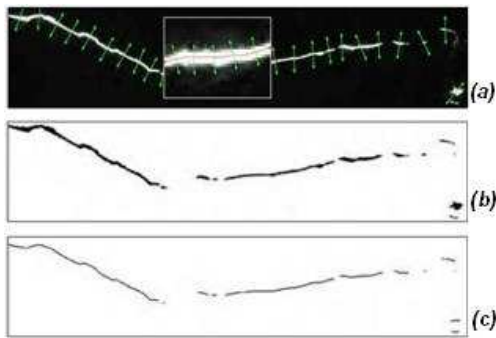
Şekil 2.7 Cismin kendi kendini örtmesi dolayısıyla oluşan gölge etkisi ve bir miktar çoklu yansıma

Örtüşme etkileri: Bu etkiler sürekli lazer çizginin görüntüdeki ayırık resim öğelerine gönderiminin doğrudan bir sonucudur. Buradaki negatif etki profil aradeğerlemesiyle (interpolation) ve alt resim öğesi aradeğerlemesiyle azaltılabilir.

Bu etkilerin birleşimi 2B görüntüde aykırı noktalara, boşluklara ve deliklere dolayısıyla da hatalı bir nokta bulutuna yol açar. 2B görüntüde boşlukların kapatılması iyi bir çözüm değildir çünkü bu boşluklar olması gereken öğeler de olabilirler. Ayrıca resim öğeleri arasındaki mesafe aynı olsa da 3B karşılıkları arasındaki mesafe öyle olmayabilir. Yine de, problemlerin birçoğuyla kamerayı, nesneyi ve lazeri istenmeyen yansımaları azaltmak için en uygun şekilde yerleştirerek başa çıkılabilir.

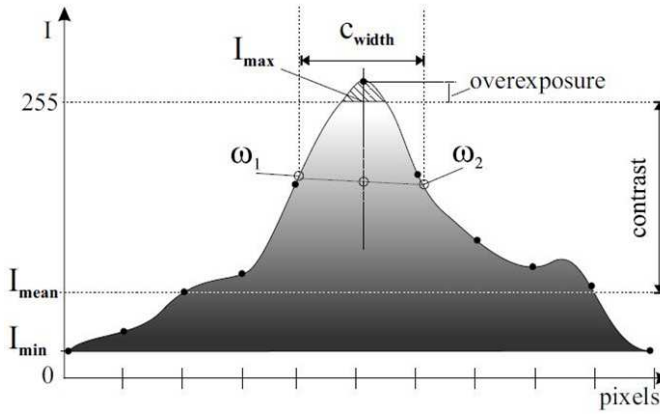
Curless ve Levoy (1995) tarafından önerilen yansıma miktarını zamana göre gözlemleyen bir yöntem özellikle parıldama ve benek etkilerinden kaynaklanan hataları azaltabilir. Ancak, yöntem bir çizgi için birden fazla görüntü gerektirmektedir. Bu da daha fazla zaman harcanmasına sebep olmaktadır.

Parıldama etkisi ve benek etkisi gibi sorunların sonuçları, genel olarak ikili görüntülerden çevritlerin (konturların, İng. contour) çıkarılmasında yayınık yansımaların (diffuse reflection) dağılım özelliklerini hesaba katan, Teutsch (2007) tarafından önerilen yöntemle de azaltılabilir. Bu yöntem çevrite dik yolları inceleyerek farklı profiller çıkartır (Şekil 2.8). Profillerin merkez bükülme noktaları arasında doğrusal aradeğerleme yapılarak resim ögesi boyutundan daha küçük hassaslıkla çevrit hesaplanır (Şekil 2.9). Algoritma ana hatlarıyla şekil 2.10'da verilmiştir.



Şekil 2.8 Çevritin arka plandan ayrılması.

a. İlk gri tonu görüntü, b. sabit bir eşik ile ikili görüntüye dönüştürülür, c. ve inceltilir.



Şekil 2.9 Çıkarılan profilin gri tonu fonksiyonu

Input: gray level image I containing the observed laser line and threshold t

Output: 2D subpixel positions of the contour line centers

calculate Binarization(I, t, B)

calculate Skeletonization(B, S)

calculate Laplace(I, L)

foreach pixel position p_i in S **do**

if p_i is foreground pixel **then**

 calculate nextNeighbor(p_i, n_i)

 define profile direction $\vec{d}_p$ perpendicular to $\overrightarrow{n_i - p_i}$

 calculate Mean(I, d_p)

foreach pixel position r_i on the profile **do**

if $I(r_i) \geq I_{mean}$ **then**

 calculate FirstZeroCrossing ($L, p(\omega_1)$)

 calculate FirstSecondCrossing($L, p(\omega_2)$)

 calculate centralPosition($p(\omega_1), p(\omega_2)$)

 break;

end

end

 set p_i to background

end

end

Şekil 2.10 Çevrit çıkarımı algoritması

Bunların yanı sıra genel olarak nesnelerin dijital modellemesinde halen aşılamamış sorunlar olduğu hatırlanmalıdır. Örneğin çok görüntü yansıtan veya şeffaf nesneler optik teknolojiler için zorluklar yaratmaktadır.

2.6 Nokta Bulutundan 3B Modele

Elde edilen nokta bulutları daha önce de belirtildiği gibi 3B uzayda tanımlanmış bir koordinatlar kümesidir. Veri olarak, bu bulutlar, taranan ya da dijitalleştirilen nesnenin görünür yüzeyini temsil etmektedirler ancak doğrudan bu şekilde çoğu 3B uygulamada kullanılamazlar (özellikle görüntüleme veya fiziksel benzetim içeren uygulamalarda). Bu verilerin birçok uygulamada kullanılabilmesi için 3B bir yüzey modeline dönüştürülmesi gerekir.

Nokta bulutundan model oluşturma işlemi yeniden yapılandırma olarak isimlendirilmektedir. Bu alanda çok sayıda algoritma vardır ve aktif bir araştırma konusudur. Değerlendirilen tüm algoritmaları tek bir sınıflandırmaya tabi tutmak mümkün değildir. Yine de bir sonraki bölümde bu algoritmalar için, kullandıkları metotlar, verilerin çeşidi ve gösterim şekli gibi kıstaslar bazında ayırık sınıflandırmalar verilmektedir.

2.6.1 Yeniden yapılandırma algoritmalarının sınıflandırılması

Öncelikle algoritmaları girdi verilerinin özelliklerine göre incelersek düzenlenmemiş nokta bulutları kullanan algoritmalar ve yapıli nokta bulutları kullanan algoritmalar olarak ikiye ayırabiliriz. Düzenlenmemiş veriler ile çalışan algoritmalar noktaların uzaydaki koordinatları haricinde bir bilgiye sahip değillerdir. Nesnenin geometrik yapısıyla ilgili herhangi bir varsayımda bulunmazlar, dolayısıyla, çokgenlerden oluşan bir yüzey yaratmadan önce noktaları birbirleriyle uyumluluklarına göre yapılandırırılar. Verilerin tekdüze (uniform) dağılmış olması başarıya ulaşmaları için önemlidir. Yapılı nokta bulutları kullanan algoritmalar ise diğerlerinin aksine noktalarla ilgili (kesme çizgileri gibi) ek bilgileri hesaba katabilirler.

Bir diğer sınıflandırma, algoritmaları, uzayı bölümlendirme yöntemlerine göre yüzeye dayalı ve hacme dayalı olarak iki sınıfa ayırmaktadır. Yüzeye dayalı algoritmalar oluşturulan yüzeylerin açık veya kapalı olmasını ayırt etmeyen algoritmalarlardır. Hacme

dayalı yöntemler ise özellikle kapalı yüzeylerle çalışırlar ve genellikle Delaunay üçgenlemesi tabanlıdırlar.

Yine bir ayrım da yöntemlerin yüzeyleri gösterme biçimi üzerinden yapılabilir. Parametrik gösterimi tercih eden yöntemler yüzeyleri, parametrik eşitliklerle tanımlanan yüzey parçalarıyla betimlerler. Çok sayıda parça bir araya getirilerek sürekli bir yüzey oluşturulur. B-Spline, Bezier eğrileri ve Coon parçaları parametrik gösterimin başlıca örnekleridir. Örtük gösterim kullanan metotlar ise yüzeydeki bütün noktalardan geçen sürekli bir fonksiyon bulmaya çalışırlar. Alfa şekilleri ve Crusts algoritması gibi, yüzeyleri nokta, kenar ve üçgenler gibi yapıların birleşimi olarak göstermeyi tercih eden basit gösterim algoritmaları da diğer bir gruba oluşturur.

Benimsenen gösterim tarzına göre sonuç olarak yaklaşık (approximated) yüzeyler veya aradeğer yüzeyleri oluşturulur. Yaklaşık yüzeylerde, yüzey bütün verilen noktaları içermez, sadece noktalara olabildiğince yakın geçen bir gösterim oluşturulur. Esas ağ yüzeyi tahmin etmek için bir uzaklık fonksiyonu kullanılabilir. Bu gruba eğmeye dayalı (warping based) yüzey yapılandırılmaları (verilen noktalara yaklaşmak için belirli bir başlangıç yüzeyini deforme ederek çalışan yöntemler) ve örtük yüzey uydurma algoritmaları da (mesela verilen noktalar için parça parça polinomsal fonksiyonlar uyduran) dâhil edilebilir. Ara değer yüzeyleri ise genellikle tam modeller gerekli olduğunda tercih edilirler. Verilen bütün noktaların oluşturulan yüzey üzerinde bulunduğu yöntemlerdir ve genellikle noktaların birleştirilmesinden oluşurlar.

Son olarak yeniden yapılandırma yöntemlerini varsayımlarına göre sınıflandırabiliriz. Bazı algoritmalar yüzeyin topolojik çeşidinin önceden bilindiğini varsayarlarken bir kısmı da yüzeyi yapılandırırken verilerin yapısından faydalanır. Örneğin, bu çalışmadaki gibi tarama işlemlerinde birim açı ile cisim döndürülüp birbiri ardına elde edilen imgeler arasındaki matematiksel ilişkiden yararlanılabilir. Başka yöntemlerde ise veri ile beraber cismin duruşu ile ilgili bilgiden faydalanılır. Mesela noktalar hacimsel verilerden elde edilmiş ise bu verilerdeki eğim, yeniden yapılandırmada kullanılabilecek yönelim bilgisini sağlar.

Kullanılacak yeniden yapılandırma yöntemini belirlemek amacıyla yukarıda verilen kategorilere göre bir inceleme yapılmıştır. Bazı sınıflandırmalarda seçim, sistemin çalışma prensipleri gereği zaten belli iken bazı sınıflandırmalarda, yeri geldiğinde ele alınan algoritmanın da daha iyi anlaşılması için, algoritmayı programa dökmek ve denemek gerekmiştir.

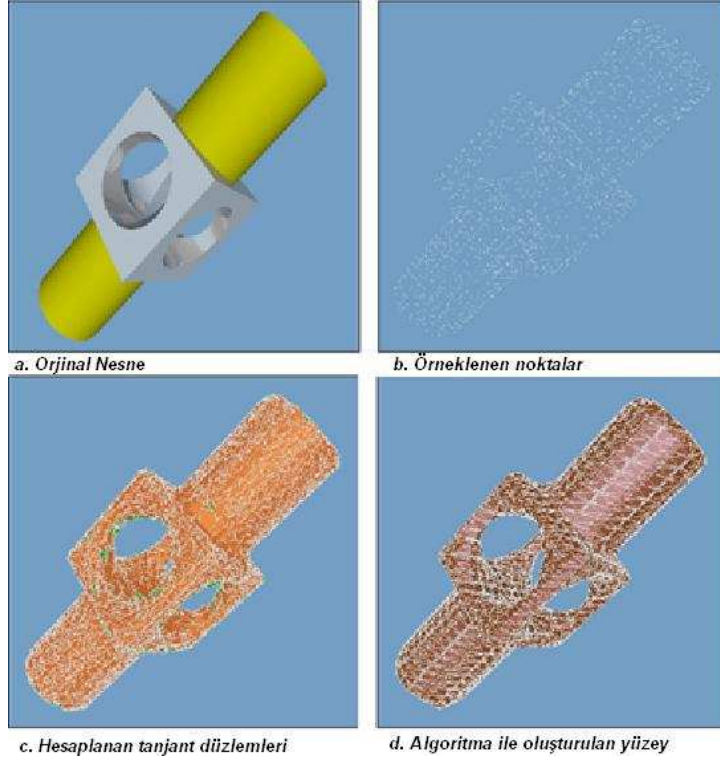
Örneğin ilk sınıflandırmayı, yani verinin düzensiz olması veya yapılı olmasını değerlendirirsek, ikinci seçenek, kullanılacak verilerin yapısının, dolayısıyla da tarama sisteminin daha önce anlatılan kısımlarının değiştirilmesini gerektirmektedir. Bu yüzden düzensiz veri kullanan algoritma tercih edilmiştir. Fakat uygulanacak yöntemin yüzeye dayalı mı yoksa hacme dayalı mı olacağını belirlemek için iki gruptaki algoritmaların daha iyi incelenmesi gerekmiştir.

Program hali ilk denenen algoritma, Hoppe vd. (1992) tarafından ortaya konulan, düzensiz veri ile çalışan, basit gösterim ile yaklaşık yüzeyler ortaya koyan bir algoritmadır ve aşağıda özetlenmiştir.

- Algoritma iki kısımdan ibarettir. İlk kısımda, verilere yakın bir bölge $D \subset \mathbb{R}^3$ üzerinde tanımlı, bilinmeyen esas yüzeye olan pozitif veya negatif mesafeyi tahmin eden bir fonksiyon, $f : D \rightarrow \mathbb{R}$, tanımlanır ve kullanılır. Elde edilen tahmini yüzeye $Z(f)$ dersek, ikinci kısımda da $Z(f)$ 'e yaklaşan basit bir yüzey oluşturulur.
- Mesafe fonksiyonunu tanımlarken yapılan kilit işlem, belirli bir yönelimdeki düzlemlerle verilen noktaları eşleştirme işlemidir. Bu *tanjant düzlemler* esas yüzeyin yerel lineer tahminleri gibidir. Tanjant düzlemlerinin ortaya konması oldukça basit olsa da bu düzlemler için, sonuçta elde edilecek yüzeyin doğru yönelimini sağlayacak küresel olarak uyum içerisinde olan yönelimler belirlemek algoritmanın önündeki önemli engellerden birisidir. Şekil 2.11'de görülebileceği gibi tanjant düzlemleri, doğrudan yüzeyi tanımlamaz. Bunun yerine yüzeye olan mesafe fonksiyonunu tanımlarlar. Şekil 2.11.d'de basit yüzeyli sonuç görülmektedir.

Hacme dayalı algoritmalar araştırılırken bunların genellikle Delaunay üçgenleme metoduna dayandığı, dolayısıyla çoğunun yaklaşık değer yüzeyleri değil aradeğer yüzeyleri oluşturduğu gözlemlenmiştir. Buradaki çalışmada geliştirilecek yazılımların

taranan cisimlerin yüzeyinden çıkarılan örnek noktalara yaklaşık yüzeyler oluşturmaları isabetli bir model olduğu için söz konusu değildir. Yani üçüncü kategorideki yaklaşık yüzeyler oluşturan yöntemler ilk başta elenmiştir.



Şekil 2.11 Hoppe algoritmasıyla yeniden yapılandırma

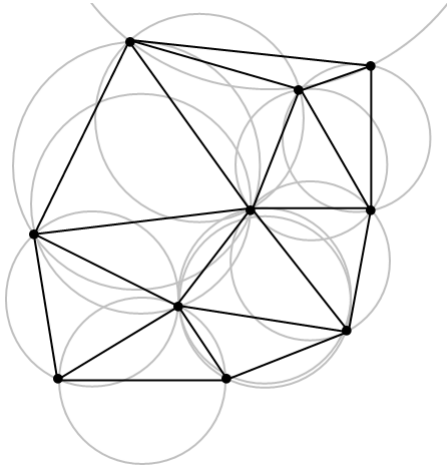
Sonuç olarak gerçekleştirilen ve test edilen algoritmalarla beraber yukarıdaki sınıflandırmaya göre karar verilerek düzenlenmemiş nokta bulutunu girdi olarak kabul eden, hacme dayalı, basit gösterimde, aradeğer yüzeyleri oluşturan ve elde edilecek görüntüler arasındaki yönelim bilgisine göre çıkarımlar yapan bir algoritmanın kullanılmasında karar kılınmıştır. Bölüm 2.6.3'te anlatılan Top-Pivotalama Algoritmasının (TPA, İng. Ball-Pivoting Algorithm, BPA) kullanılmasına karar verilmiştir. Algoritmanın programa dökülmesiyle ilgili detaylar bölüm 3.7'de anlatılmaktadır.

Nokta bulutlarını modellere dönüştürmede kullanılmasına karar verilen TPA ve anlaşılmasını kolaylaştıracak matematiksel Delaunay üçgenleme metodu ve benzer bir

yöntem olan İlerleyen Üçgenler (İÜ, İng. Marching Triangles, MT) aşağıda özetlenmiştir.

2.6.2 Delaunay üçgenleme metodu

Matematik ve geometride, verilen bir nokta kümesi, P için Delaunay üçgenlemesi, $DÜ(P)$, P 'deki herhangi bir nokta, herhangi bir üçgeni çevreleyen çemberin içerisinde bulunmayacak şekilde P 'nin bütün elemanlarının çizgilerle birleştirilmesidir (Şekil 2.12). Bu koşulu sağlayan bir şekilde tüm üçgenlerin tüm iç açılarından en küçük olanının maksimize edildiği garantilenmiş olur. Model oluşturmada bunun önemi, yassı ve zayıf üçgenlerden mümkün olduğunca kaçınılmış olunmasıdır.



Şekil 2.12 Bir düzlemde (2B) Delaunay Üçgenleme örneği

Bu tanım yalnızca 3B uzayda değil n -boyutlu uzayda geçerli olacak şekilde geliştirilebilir: n -boyutlu Öklit uzayında P noktalar kümesi için P 'nin elemanları arasında çizgiler çizilerek oluşturulan $DÜ(P)$ üçgenlemesinin Delaunay üçgenlemesi olması için, P 'deki hiçbir nokta $DÜ(P)$ 'deki herhangi bir simpleksi çevreleyen hiperküre içerisinde bulunmamalıdır.

Verilen bir nokta kümesi üzerinde Delaunay üçgenlemesini farklı yöntemlerle elde eden çeşitli (arttırılmalı/incremental, böl ve fethet / divide & conquer, vb.) algoritmalar mevcuttur. Hangi yöntem kullanılırsa kullanılsın bütün algoritmaların ortak noktası, bir

noktanın bir küre içerisinde olup olmadığını belirleyen operasyonların hızlı çalışmasına ve üçgenleri ve kenarları tutan veri yapılarının verimli olmasına bağlı olmalarıdır. 2B bir D noktasının, bir ABC üçgeni içerisinde olması durumunda aşağıdaki eşitsizlik doğru olacaktır (Determinant pozitif olacaktır).

$$\begin{vmatrix} A_x & A_y & A_x^2 + A_y^2 & 1 \\ B_x & B_y & B_x^2 + B_y^2 & 1 \\ C_x & C_y & C_x^2 + C_y^2 & 1 \\ D_x & D_y & D_x^2 + D_y^2 & 1 \end{vmatrix} = \begin{vmatrix} A_x - D_x & A_y - D_y & (A_x - D_x)^2 + (A_y - D_y)^2 \\ B_x - D_x & B_y - D_y & (B_x - D_x)^2 + (B_y - D_y)^2 \\ C_x - D_x & C_y - D_y & (C_x - D_x)^2 + (C_y - D_y)^2 \end{vmatrix} > 0$$

3B uzayda da benzer ancak daha fazla işlem içeren bir hesap yapılır. Buradaki işlem sayısı algoritmaların performansı için ciddi bir darboğaz oluşturduğundan ve nokta sayısı arttıkça performans giderek daha da kötüleştiğinden Delaunay üçgenlemesine yakınsayan yöntemler daha popülerdir.

2.6.3 İlerleyen üçgenler (İÜ) ve top-pivotlama(TPA)algoritmaları

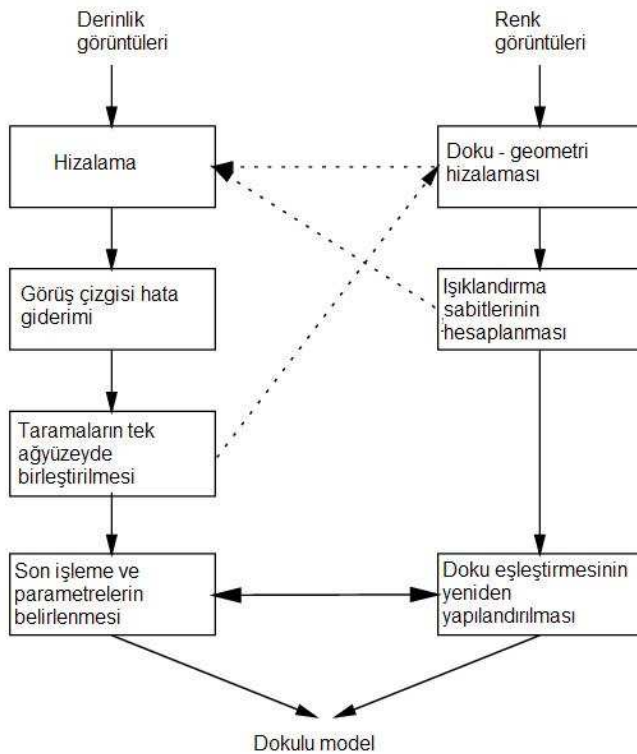
Delaunay üçgenlemesi optimal bir geometrik yapıdır ancak bir önceki kısmın sonunda da belirtildiği gibi hesaplanabilirliği zaman bakımından masraflıdır. İÜ, Delaunay üçgenlemesine yakın bir sonuç veren ve daha hızlı hesaplanabilir bir yaklaşımdır. Daha hızlı hesaplama yapılmasını sağlayan kısım, verilen bir üçgeni çevreleyen kürenin gerçek merkezi yerine üçgeni çevreleyen çemberin merkezinin kullanılmasıdır.

Yine Delaunay üçgenlemesine yakın bir sonuç vermeye çalışan TPA'da ise üç nokta yüzeyinde iken içi boş olması gereken kürenin boyutunu kullanıcının belirlemesine izin verilmektedir. Algoritmanın çalışma prensibi basitçe şöyle özetlenebilir:

- Alınan 3 nokta, i , j ve k ancak, belirlenen r yarıçaplı bir küre başka herhangi bir noktayı içermeyen kendilerine dokunursa geçerli bir üçgen oluşturur.
- Başlangıçta çekirdek olarak seçilen bir üçgenden yola çıkarak küre bir kenar üzerinde, başka bir noktaya temas edene kadar döndürülür. Yeni bir noktaya temas sağlandığında yeni bir üçgen elde edilmiş olur.
- Oluşan üçgen grubu içerisinde her kenar denenene kadar bu işlem devam eder. Sonrasında ise yeni bir çekirdek üçgen belirlenir.
- Tüm noktalar dâhil edildiğinde işlem sonlanır.

Bu yöntemin buradaki tez çalışmasında üç boyutlu tarama işleminden elde edilen nokta bulutu üzerinde uygulanmasının bir avantajı başlangıçta belirlenen küre yarıçapının (r) nokta bulutu elde edilirken ortaya çıkan örnekleme sıklığına bağlı olarak seçilebilmesidir. Genel olarak ise diğer yöntemlere oranla daha az zaman ve yer kullanarak yeteri kadar kaliteli sonuçlar vermesi yaygın kullanımının en büyük nedenidir.

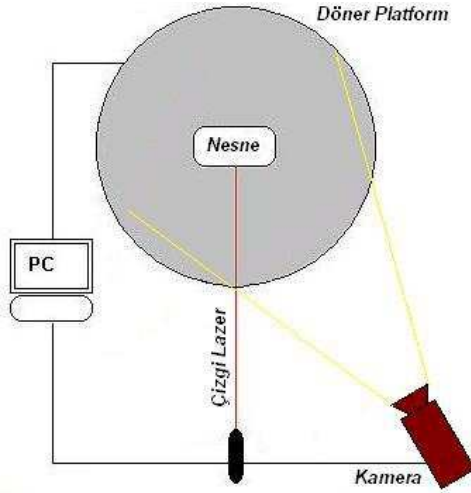
Her ne kadar çalışmadaki son aşama olsa da, noktaların birleştirilerek modele dönüştürülmesinden sonra yapılabilecek başka işlemler de vardır. Doku, renk, parlaklık gibi gerçekçi yüzey özellikleri sergileyen son bir model oluşturmak için lazer ışısız normal görüntülerin de alınarak işlenmesi nesnenin tam anlamıyla sanal bir kopyasına ulaşmak için gereklidir. Hatta derinlik bilgisi içeren mesafe görüntülerinin ve yüzey özellikleri elde edilecek normal görüntülerin işlenmesi birbiriyle iletişim kurarak yardımlaşan iki paralel işlem olarak uygulanabilir. Dolayısıyla, genel olarak bakıldığında model edinme işlemi birbiriyle iç içe boruhattı işlemleri olarak görülebilir (Şekil 2.13).



Şekil 2.13 Ardışık düzen olarak model edinimi

3. MATERYAL VE YÖNTEM

Başlangıçta inşa edilen tarayıcı donanım olarak hepsi standart bir bilgisayardan kontrol edilen bir döner platform, bir helyum-neon (HE-NE) çizgi lazer kaynağı ve bir kameradan oluşmaktadır. İkinci kameranın sisteme dâhil edilmesi sebepleriyle ve amacıyla birlikte bölüm 4’te anlatılmaktadır. Sistemin genel görünümü şekil 3.1’deki gibidir. Sistem yarıçapın 20cm, yüksekliğin de 40cm olduğu $\pi r^2 h$ boyutunda silindirik bir tarama hacmine sahiptir. Çizgi lazer ışık dikey olarak nesne üzerine düşürülür. Platform üzerine yerleştirilen nesne tüm yüzeyi görüntülenene kadar hassas bir adım motoru kontrol mekanizması aracılığıyla platformun merkezinden geçen dikey bir eksen etrafında döndürülür. Tarama işlemi aydınlanma ile ilgili problemlerin engellemesi için sadece lazer ışığın görülebildiği karanlık bir ortamda gerçekleştirilir. Alınan görüntülerin işlenmesi, 3B modelin oluşturulması ve ekranda gösterilmesi için kullanılan kişisel bilgisayarın işlem kapasitesi yüksek, bellek sıkıntısı yaşanmayacak kapasitede ve ileri seviye bir grafik işlemciye sahip olması öngörülmüştür.



Şekil 3.1 Sistemin yukarıdan şematik görünümü

3.1 Döner Platform

Döner platform, merkezdeki melez (hibrit) adım motorun kontrol kartları aracılığıyla bilgisayara bağlanmasıyla kontrol edilecektir. Döner masa düzeneğinin isabetli bir model çıkarılabilmesi için hata payını en aza indirecek yapıda olması gerekmektedir. Bunun içinsarsıntının ve sallanmanın olabildiğince engellenmesi, dönme hızının veya birim dönme açısının standart olması gibi arayışlar olacaktır.Sarsıntı dolayısıyla oluşacak bozulmaları yazılım ile gidermek için de bazı araştırmaların sonuçları umut vaat etmektedir (Blais vd. 2004).

Dönüş esnasında sabit bir açısal hızın korunması açısından adım motoru kullanılması tercih edilmiştir. Kullanılan motor %5 isabet ile 1.8° adım dönme açısına sahip melez, 4-aşamalı bir motordur (Çizelge 3.1). Bu motor, uyumlu kontrol kartları yardımıyla yarım adım dönme seçeneği de sağlamaktadır.

Çizelge 3.1Kullanılan adım motorunun teknik özellikleri

Technical Specifications	
Rated voltage (V)	2.5
Rated current (I)	4.5
Resistance (Ω)	0.56
Inductance (mH)	2.8
Detent torque (mNm)	100
Holding torque (mNm)	2200
Step angle accuracy (%)	5
Step angle	1.8
Insulation class	B

Oluşturulan döner platform(Şekil 3.2) üzerinde bulunan nesne tam bir tur döndürülene kadar her adımda üzerine düşen lazer çizgi ile görüntüsü alınacaktır. Nesnenin bütün çevresi için görüntüleme tamamlandığında üç boyutlu dijital modelin yaratılması için gerekli veriler elde edilmiş olacaktır.



Şekil 3.2 Döner platform

Platformun istenmeyen dikey hareketlerini önlemek ve sadece yatay düzlemde hareketini sağlamak için motor 50x50x10cm ebatlarında ahşaptan bir çerçeve (motor haznesi) içine yerleştirilmiştir (Şekil 3.3).

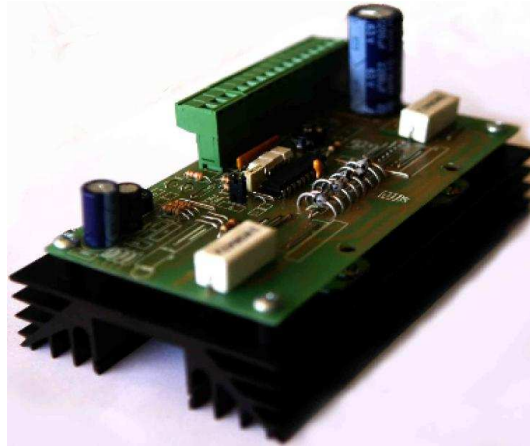


Şekil 3.3 Motor haznesi açık (solda) ve kapalı (sağda) olarak

Platformun tarama sisteminin genelini kontrol edecek bilgisayardan kontrol edilmesi başlangıçta, programlanabilir bir kontrol kartı aracılığıyla kendisine komut gönderilen bir sürücü kart vasıtasıyla sağlanmak istenmiştir. Bunun için temin edilen ilk elektronik kartların (Şekil 3.4) bağlantıları tamamlanıp motor kontrolü denemeleri yapılmaya başlandığı sırada bir arıza meydana gelmiştir. Bu arızanın yol açtığı sorun ancak yeni bir kontrol kartı temin edilerek giderilebilmiştir (Şekil 3.5).

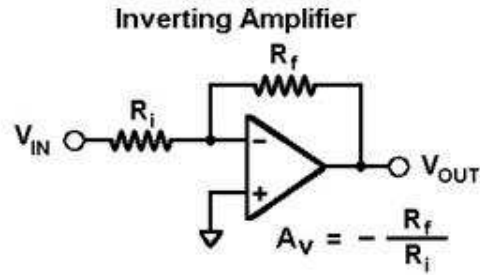


Şekil 3.4 Motor kontrolu için kullanılan ilk kartlar

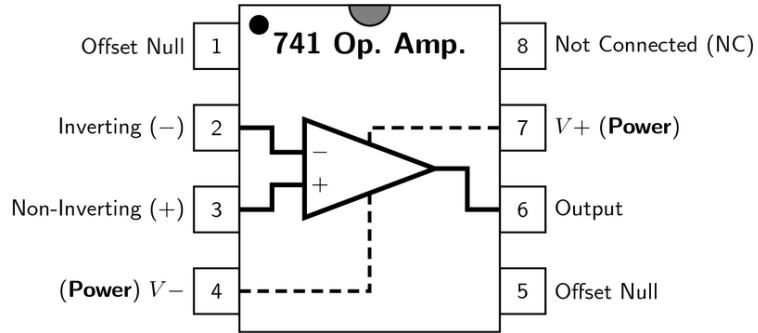


Şekil 3.5 Adım motor sürücü kartı

Temin edilen yeni kartı tetikleyebilen adım giriş sinyalinin bilgisayardan çıkan sinyalden farklı sınır değerlerine sahip olduğu gözlemlenmiştir. Bunun üzerine bilgisayar ve kart arasına bağlanarak adım sinyalini istenen seviyeye çekecek bir işlemsel kuvvetlendirici (operational amplifier, OP-AMP) devre hazırlanmıştır(Şekil 3.6). Bu şemaya göre oluşturulan devrede 741 tipi entegre OP-AMP kullanılmıştır (Şekil 3.7).

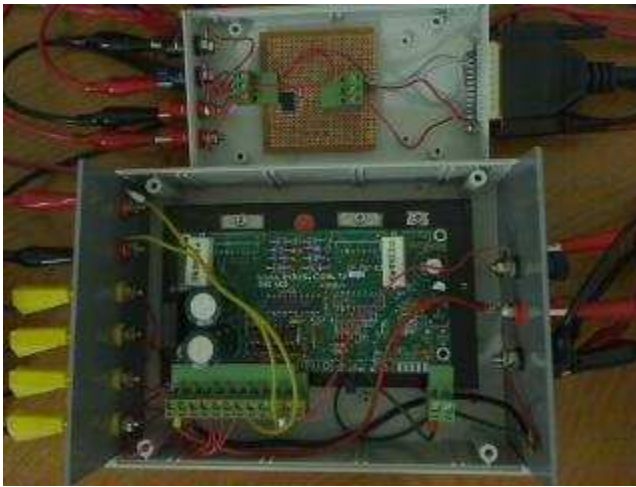


Şekil 3.6 Op-amp devre şeması



Şekil 3.7 741 tipi OP-AMP entegre yonga

Bilgisayardan gelen sinyaller OP-AMP devreden geçirildikten sonra sürücü karta iletilecek ve sürücü kart da motoru sürecektir şekilde bağlantılar yapılmıştır (Şekil 3.8). Kontrol yazılımında gerekli değişiklikler yapılmış ve düzenek test edilerek bilgisayarla platformun kontrolü yeniden sağlanmıştır.



Şekil 3.8 Sürücü kart, OP-AMP devre ve bağlantıların görünümü

Platformun bilgisayar kontrolü ile dairesel hareketi test edilirken karşılaşılan diğer bir problem dairesel yüzeyin dönme hareketinin her aşamasında eşit bir açısal hız sergileyememesi olmuştur. Hareket incelendiğinde problemin kaynağının dairesel yüzeyi oluşturan yüzeyin temas ettiği bilyelerle arasındaki sürtünme kuvveti olduğu saptanmıştır. Bu sürtünme kuvvetinin fazla olması motorun gücünün harekete net dönüşümünü etkilemiştir. Ayrıca dönüşün farklı aşamalarında sürtünme kuvvetinin değişmesi dairesel hareketin her aşamada eşit açısal hızla gerçekleşmesini engellemiştir.

Bu sorunu gidermek için yüzey olabilecek farklı malzemelerden farklı kalınlıklarda maddeler denenmiştir. Sürtünmeyi azaltmak amaç olmasına rağmen yüzeyin cisimleri taşıyabilecek, ağırlık altında eğilip bükülmeyecek olması da gerekir. Sonuç olarak 8mm kalınlığında kavak ağacından elde edilmiş kontra plakanın bilyelerle temas eden hizasına 3mm kalınlığında fiberglas yapıştırılarak bu yüzeyin kullanılmasına karar verilmiştir (Şekil 3.9). Bilyeler ve onlara temas edecek olan fiberglas yüzey yağlanmış, motor silikon malzeme ile iyice sabitlenmiş, şaftın üzerine yüzeyi ortadan desteklemesi için sıkı bir lastik geçirilmiştir. Sonra yüzey kenarlara yerleştirilen destekler sayesinde her bilyeden 1 mm yukarıda kalacak şekilde tutularak motor şaftına kuvvetli bir yapıştırıcı ile sabitlenmiştir. Tarama sırasında ışığın doğrudan veya dolaylı olarak üzerine düşebileceği platform yüzeyi ve arka plan ışığı olabildiğince az yansıtmaları için siyah renge boyanmıştır.



Şekil 3.9 Yeni platform yüzeyi

C# programlama dili ile paralel LPT portu (yazıcı portu) üzerinden kontrol kartına komut gönderen dolayısıyla da platforma dairesel hareket sağlayan bir program geliştirilmiştir. Bu program kullanılarak platform 0,9°'lik ve aradaki süre isteğe göre belirlenebilen adımlarla döndürülebilmektedir, bu da tam bir dönüşün 400 adıma bölünebilmesi anlamına gelir.

3.2 Çizgi Lazer Kaynağı

Biçimli ışık kullanılan 3B tarama işlemlerinde genellikle lazer tercih edilmektedir. Bunun başlıca sebeplerinden bazıları alternatiflerine oranla lazer ışığının mesafe artsa da iyi odaklanabilir olması ve tek dalga boyunun görüntü işleme teknikleri ile daha kolay ayırt edilebilir olmasıdır. Aşağıdaki maddeler kullanılan sürekli dalga (SD) kırmızı renk çizgi lazer diyotunun önemli teknik detaylarını özetler:

- Dalga boyu: $\lambda_1 = 635\text{nm}$
- Iraksama: $\theta_r = 0,3\text{mrad}$
- Çizgi kalınlığı: $k \leq 0,5\text{mm}$ 0,5m'den yakinken
 $\leq 1\text{mm}$ 3m'den yakinken
 $\leq 1,5\text{mm}$ 6m'den yakinken
- Kavis: $k_{av} = 1\text{mm}$

Teknik özelliklerden de görüldüğü gibi özellikle yakın mesafelerde lazer çizginin kalınlığı oldukça küçüktür. Yine de kalınlığı olabildiğince düşük tutmak için yerleştirme yapılırken ışık kaynağı, kameranın görüntüsüne girmeden nesneye ne kadar yaklaştırılabilirse yaklaştırılacaktır. Kameranın yerleşimi ve yönelimi göze alındığında bu mesafenin 1m'den yakın olacağı söylenebilir.

3.3 Algılayıcı Kamera

Görüntüleme cihazı için önemli olabilecek konuları irdelerken nasıl kullanılacağından biraz daha bahsedelim. Tarayıcının çalışması esnasında masanın her bir dönme açısı için 2 çeşit görüntü alınacaktır. Derinlik bilgisi elde etmeye yönelik, nesne üzerine lazer ışık

düşürülerek karanlıkta çekilmiş bir görüntü ile doku ve renk bilgisi elde etmeye yönelik lazer ışık olmayan en az bir (farklı ışıklandırma koşullarına gerek duyulabilir) görüntü alınacaktır. Bu amaçlara yönelik olarak görüntüleme cihazı için en önemli özellik farklı ışık koşulları altında yüksek çözünürlüklü, renk açısından hassas ve isabetli görüntü yakalayabilmesidir (Yıldırım vd. 2006).

Diğer bir yandan çalışmada başlangıçta belirtilen kıstaslardan bir diğeri maliyeti olabildiğince düşük tutma çabasıdır. Görüntüleme cihazlarında yüksek çözünürlüklerde kaliteli görüntü alma özelliği doğrudan cihazın fiyatını arttıran bir özelliktir. Bunun yanı sıra oluşturulan sistem çalışır hale getirildikten sonra kullanılan görüntüleme cihazının daha kaliteli bir tanesiyle değiştirilmesi sistemde pek bir değişiklik yapılmasını gerektirmemektedir. Bu sebeplerle başlangıç olarak maliyetin düşük tutulması için kamera kalitesinden ödün verilerek bugün kişisel bilgisayarlarda kullanılan cihazların seviyesinde ucuz bir web kamerası kullanılmıştır.

Platform ve hareket kontrol yazılımı tamamlandıktan sonra sisteme web kamerası ve çizgi lazer kaynağı da eklenmiş ve bölüm 2.3’de anlatılan yaklaşık dönüşümün hesaplanması için ayarlama görüntüleri alınmıştır. Ayarlama görüntüleri için çizgi lazerin farklı, bilinen noktalara yerleştirilen, saydam olmayan veya ışığı parlatmayan düz bir cisim üzerindeki görüntüleri kaydedilmiştir. Daha sonra bu görüntülerde aydınlık resim öğeleri belirlenerek EKK yönteminin girdileri olan nokta eşleşmeleri kümesi Z elde edilmiştir.

3.4 Aydınlık Resim Öğesi Pozisyonlarının Elde Edilmesi

Lazer ışığın nesne yüzeyiyle kesişmesi ile ortaya çıkan çevritinşekli yüzey topolojisine göre değişmiştir ve derinlikle ilgili bilgi taşır (Şekil 3.10). Bu bilgi ancak çevrit temiz bir şekilde arka plandan ayrılabilirse elde edilebilir. Bunun başarılması için bölüm 2.4’de anlatılan görüntü işleme teknikleri uygulanarak şekil 3.11’de görülen ikili görüntü elde edilir. Kullanılan ışık kırmızı renkli olduğu için renkli görüntü yeğnlik görüntüsüne çevrilirken kullanılan katsayılar

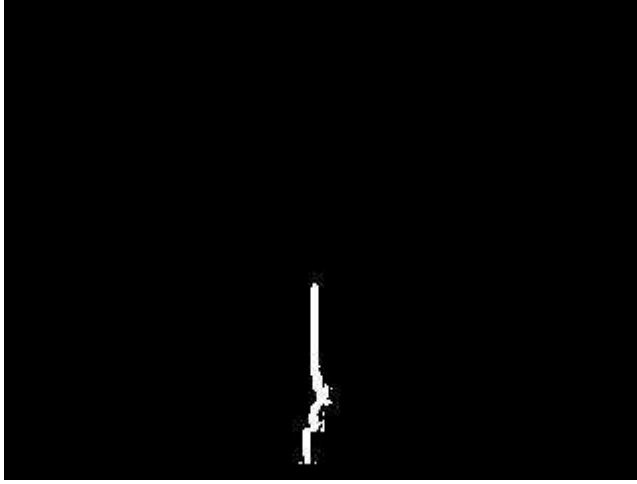
$$w_R = 1, w_Y = 0, w_M = 0$$

olarak belirlenmiştir. Yeğnlik görüntüsünden ikili görüntüyü elde ederken kullanılan eşik değeri ise nesne için alınan görüntüler incelenerek belirlenmektedir ve farklı nesneler taranırken farklı eşik değerleri kullanılabilir. Şekillerdeki örnekte 191 değeri kullanılmıştır.



Şekil 3.10 Nesne yüzeyi ile lazer ışığın kesişim görüntüsü

Aşağıdaki ikili şekilde de görülebileceği gibi uygulamadaki ilk problem görüntülenen çevritin bir resim ögesinden büyük bir genişliği olmasıdır. Nokta bulutunun oluşturulmasında yalnızca her satırda bir resim ögesine ihtiyaç duyulduğu için “inceltme” adı verilen bir teknik uygulanır. İnceltme işlemi çevrit tek resim ögesi kalınlığına indirgenene kadar kemirme (erosion) işleminin tekrar tekrar uygulanması esasına dayanır (Gonzales vd. 2002).



Şekil 3.11 Şekil 3.10'dan elde edilen ikili görüntü

3.5 Ayarlama İşlemi

Üçgenleme prensibinin uygulanmasından önce kameraların ve lazer ışık kaynağının yerleştirilmeleri yapılmalı ve görüntülerden 3B koordinatlara gönderim yapacak fonksiyon yaklaşımı yapılmalıdır. Bölüm 2.3’de anlatılanları özetlersek: Görüntü koordinatlarından gerçek dünya koordinatlarına gönderim yapan bir Φ fonksiyonu olduğu varsayılmaktadır (Şekil 2.4)(Teutsch 2007). Daha sonra EKK kıstası kullanılarak bu fonksiyona bir yakınsama gerçekleştirilir. Bu işleme başlarken gerçek dünyadaki pozisyonların 3B koordinatlarının belirlenmesi ile ilgili iki önemli varsayım yapılmaktadır:

- Dünya koordinatlarının merkezi, yani (0, 0, 0) noktası dairesel platform yüzeyinin merkezi olarak kabul edilmektedir.
- Dikey olarak nesne üzerine düşürülen çizgi lazerin tanımladığı düzlem 3B uzayın yz düzlemi olarak kabul edilmektedir (Şekil 3.12).

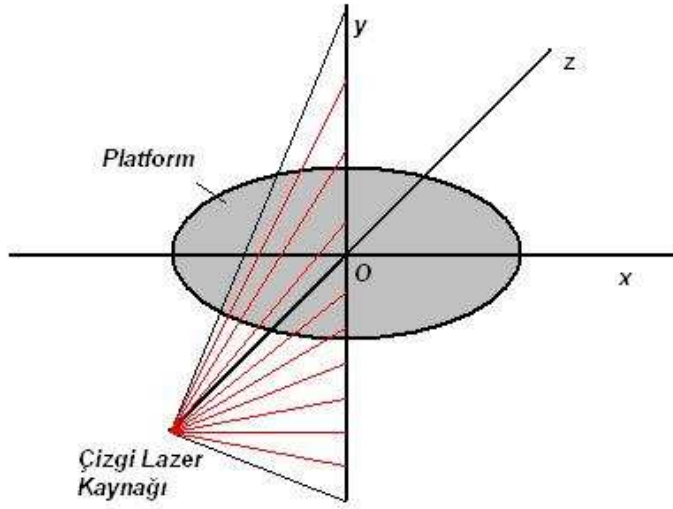
Bu varsayımların ışığında döner platformun belirli bir duruşunda alınan tek bir görüntüdeki aydınlık noktalara karşılık gelen 3B koordinatların tamamı aynı x değerine sahiptir ve bu değer 0’dır. Dolayısıyla yakınsama yapılacak fonksiyon

$$\Phi = \begin{pmatrix} 0 & 0 & 0 \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix}$$

halinde kabul edilir ve $r = \begin{pmatrix} a \\ b \\ 1 \end{pmatrix}$ 'ye uygulandığı zaman:

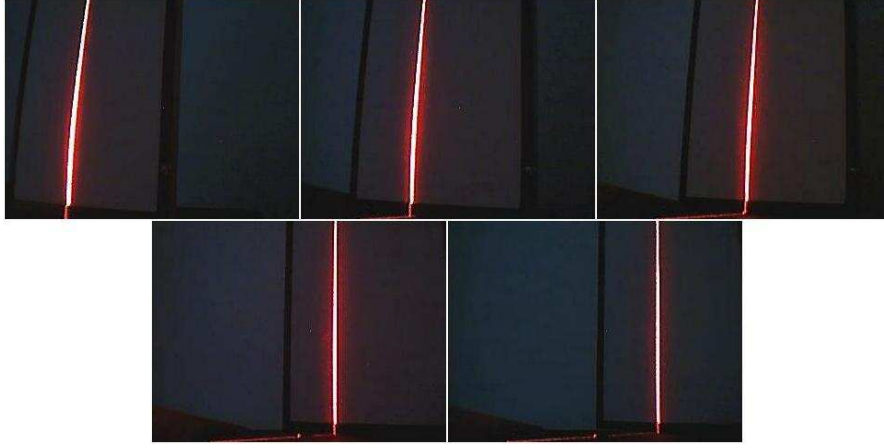
$$p = \Phi r = \begin{pmatrix} 0 \\ \phi_{21}a + \phi_{22}b + \phi_{23} \\ \phi_{31}a + \phi_{32}b + \phi_{33} \end{pmatrix}$$

3B noktası elde edilir. Geriye kalan ϕ_{ij} katsayı değerleri y eksenini boyunca yükseklik ayarlaması (ϕ_{2j}) ve z eksenini boyunca derinlik ayarlaması (ϕ_{3j}) ile hesaplanır.



Şekil 3.12 Gerçekteki pozisyonlara koordinat uzayının yerleştirimi

Derinlik ayarlaması şu şekildedir: Düz bir yüzey platform üzerine xy düzlemine paralel olacak şekilde yerleştirilir ve üzerine lazer ışık düşürülerek görüntü alınır. Bu düzlem z eksenini boyunca farklı ama koordinatları bilinen yerlere yerleştirilerek her konumdaki görüntü şekil 3.13'te görüldüğü gibi kaydedilir. Bu görüntüler daha sonra işlenerek EKK yöntemine girdiler bulunur ve buradan da 3B noktaların z koordinatını verecek olan ϕ_{3j} elde edilir.



Şekil 3.13 Derinlik kalibrasyon görüntüleri

Yükseklik ayarlaması da derinlik ayarlamasına benzer. Aradaki tek fark düz yüzeyin xz düzlemine paralel olacak şekilde y eksenı boyunca yerleştirilerek görüntü alınmasıdır.

Derinlik için lazer çizginin xy düzlemine paralel ve belirli $z_k, k \in \{1 \dots m\}$ derinliklerinde görüntülendiği m ayarlama resmi olsun. Ayrıca görüntü k 'da $(a_{lk}, b_{lk}), l \in \{1 \dots n_k\}$ ile gösterilen n_k aydınlık resim ögesi olduğunu varsayalım. Bu durumda bölüm 2.3'deki determinant:

$$\begin{aligned}
 |M| = & \sum_k \sum_l a_{lk}^2 \sum_k \sum_l b_{lk}^2 \sum_k n_k + 2 \sum_k \sum_l a_{lk} b_{lk} \sum_k \sum_l b_{lk} \sum_k \sum_l a_{lk} \\
 & - \sum_k \sum_l a_{lk}^2 \left(\sum_k \sum_l b_{lk} \right)^2 - \left(\sum_k \sum_l a_{lk} b_{lk} \right)^2 \sum_k n_k \\
 & - \left(\sum_k \sum_l a_{lk} \right)^2 \sum_k \sum_l b_{lk}^2
 \end{aligned}$$

olur. M matrisinin eşçarpan (cofactor) matrisi $C_M = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix}$ 'i oluşturan

c_{ij} 'ler:

$$\begin{aligned}
 c_{11} &= \sum_k \sum_l b_{lk}^2 \sum_k n_k - \left(\sum_k \sum_l b_{lk} \right)^2 \\
 c_{12} = c_{21} &= \sum_k \sum_l a_{lk} \sum_k \sum_l b_{lk} - \sum_k \sum_l a_{lk} b_{lk} \sum_k n_k
 \end{aligned}$$

$$C_{13} = C_{31} = \sum_k \sum_l a_{lk} b_{lk} \sum_k \sum_l b_{lk} - \sum_k \sum_l a_{lk} \sum_k \sum_l b_{lk}^2$$

$$C_{22} = \sum_k \sum_l a_{lk}^2 \sum_k n_k - \left(\sum_k \sum_l a_{lk} \right)^2$$

$$C_{23} = C_{32} = \sum_k \sum_l a_{lk} b_{lk} \sum_k \sum_l a_{lk} - \sum_k \sum_l a_{lk}^2 \sum_k \sum_l b_{lk}$$

$$C_{33} = \sum_k \sum_l a_{lk}^2 \sum_k \sum_l b_{lk}^2 - \left(\sum_k \sum_l a_{lk} b_{lk} \right)^2$$

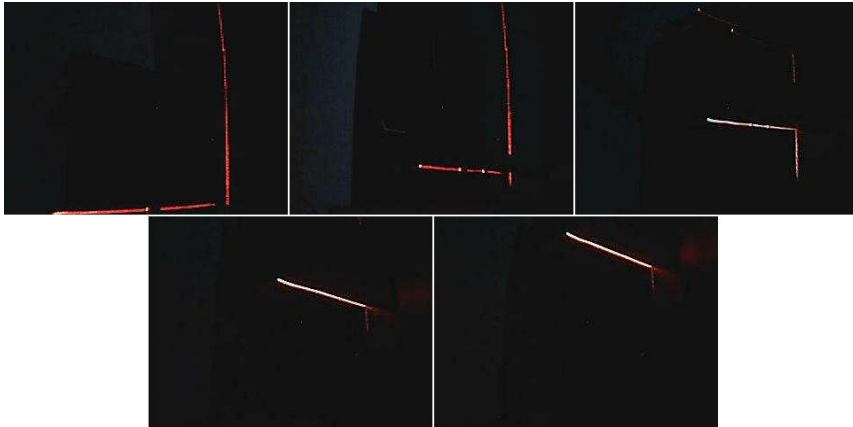
şeklindeir. Bunlar üzerinden ϕ_{3j} 'ler hesaplanırsa aşığıdaki eşitlikler yazılabilir:

$$\phi_{31} = \left(C_{11} \sum_k \sum_l a_{lk} z_k + C_{21} \sum_k \sum_l b_{lk} z_k + C_{31} \sum_k n_k z_k \right) / |M|$$

$$\phi_{32} = \left(C_{12} \sum_k \sum_l a_{lk} z_k + C_{22} \sum_k \sum_l b_{lk} z_k + C_{32} \sum_k n_k z_k \right) / |M|$$

$$\phi_{33} = \left(C_{13} \sum_k \sum_l a_{lk} z_k + C_{23} \sum_k \sum_l b_{lk} z_k + C_{33} \sum_k n_k z_k \right) / |M|$$

Bu denklemler bilinmeyen altı ϕ_{ij} 'den üçünün hesaplanmasını sağlar. Bu üç katsayı kullanılarak 2B görüntü düzleminde izdüşümünün koordinatları bilinen bir noktanın 3B uzayda z koordinatı kestirilebilir. Yukarıda da belirtildiğı gibi geriye kalan ϕ_{2j} 'ler de benzer fakat düz cismin xz düzlemine paralel farklı y koordinatlarına yerleştirilerek çekilmiş görüntülerinden (Şekil 3.14) hesaplanır.



Şekil 3.14 Yükseklik kalibrasyonu görüntüleri

Sayısal bir bakış açısı için basitleştirilmiş beşer ayarlama görüntüsü ile derinlik ve yükseklik ayarlamalarına bakalım. Örneği kısa ve sayıları küçük tutmak açısından her bir görüntüden sadece beş aydınlık resim ögesi pozisyonu çıkarıldığını düşünelim. Bu noktalar çizelge 3.2'deki noktalar olsun. Derinlik görüntüleri için düz yüzeyin yerleştirildiği derinlikler (z_k) sırasıyla -20cm, -10cm, 0cm, 10cm ve 20cm,yükseklik görüntüleri için y_k 'lar da 0cm, 9,5cm, 19,5cm, 29,5cm ve 40cm olarak ölçülmüştür.

Bu örnek noktalar ve karşılık geldikleri derinlikler ve yükseklikler kullanılarak:

$$\phi_{31} = 0,242778288$$

$$\phi_{32} = 0,010112454$$

$$\phi_{33} = -39,13946039$$

$$\phi_{21} = 0,018759503$$

$$\phi_{22} = -0,198793377$$

$$\phi_{23} = 43,39710988$$

katsayıları elde edilir.

Çizelge 3.2 Beşer adet derinlik ve yükseklik görüntülerinden çıkartılan 5'er resim ögesi pozisyonları ve yüzeyin z/y konumları

Görüntü No.	Derinlik kalibrasyonu görüntüleri					Yüzeyin z/y değeri
1	(73,51)	(67,132)	(66,143)	(65,192)	(65,196)	-20
2	(118,50)	(118,54)	(117,169)	(118,195)	(118,200)	-10
3	(165,56)	(165,57)	(164,102)	(164,105)	(163,124)	0
4	(201,79)	(200,139)	(199,176)	(199,199)	(199,222)	10
5	(228,63)	(229,143)	(229,158)	(230,182)	(230,183)	20
	Yükseklik kalibrasyonu görüntüleri					
1	(62,239)	(142,236)	(183,233)	(207,231)	(210,231)	0
2	(152,186)	(159,187)	(168,187)	(182,188)	(215,190)	9,5
3	(187,137)	(209,139)	(228,142)	(232,142)	(248,143)	19,5
4	(145,71)	(164,78)	(175,81)	(177,81)	(183,83)	29,5
5	(162,25)	(210,43)	(211,44)	(223,49)	(238,55)	40

3.6 Nokta Bulutlarının Oluşturulması

Kamera ve lazer ışık kaynağı sabitlenip ayarlama işlemi tamamlandıktan sonra artık Φ dönüşümü bilinmektedir. Bu dönüşüm kullanılarak 2B resim ögesi pozisyonları 3B

koordinatlara çevrilebilir. Bir nesnenin taranması esnasında nesne, döndürüldükçe lazer ışık yüzeyini tarayacak şekilde döner platform üzerine ortalanarak yerleştirilir. Bu şekilde bir pozisyon için kaydedilen görüntüden (Şekil 3.15) 2B resim ögesi pozisyonları çıkarılır. Görüntülerden resim ögesi pozisyonlarını çıkarabilmek için her görüntüye bölüm 3.4'te anlatılan görüntü işleme teknikleri uygulanır.



Şekil 3.15 Örnek bir nesnenin taramada alınan görüntüsü

Nesne yüzeyinin nokta bulutu olarak gösterimi için farklı görüntülerden elde edilen bilgilerin birleştirilmesi gerekmektedir. Bu birleştirme için, her görüntüden elde edilen noktalar y eksenini etrafında görüntünün sırasına göre doğru miktarda döndürülür. Diğer bir deyişle, eğer $P_i, t \in \{0 \dots 399\}$ görüntü t 'den elde edilen 3B noktalar kümesi ise sonuçta oluşturulan nokta bulutu P

$$P = \{R_i p_{ij} | i \in \{0 \dots 399\}, p_{ij} \in P_i\}$$

olarak yazılabilir. Burada R_i, y eksenini etrafında $(0,9 t)^\circ$ açı ile dönüşü simgeleyen matristir:

$$R_i = \begin{pmatrix} \cos(0,9 t) & 0 & \sin(0,9 t) \\ 0 & 1 & 0 \\ -\sin(0,9 t) & 0 & \cos(0,9 t) \end{pmatrix}$$

Daha genel düşünülürse, görüntü t üzerinde 2B bir resim ögesi noktası $r \in R_i$ çıkartıldığında, ona karşılık gelen $p \in P$ şu şekilde hesaplanır:

$$R_i \Phi r = p$$

3.7 Yeniden Yapılandırma

Bölüm 2.6’da nokta bulutlarından yüzey gösterimine sahip bir model elde etmek için kullanılan algoritmalarından ve bu çalışma için bunların içerisinde TPA algoritmasının kullanılmasına karar verildiğinden bahsedilmişti. Bu bölümde uygulamada TPA’nın programa dökülmesiyle ilgili detaylardan bahsedilecektir. TPA girdi olarak bir nokta bulutu alır ve çıktı olarak o noktaların birbirine bağlanmasıyla oluşturulmuş bir üçgen ağyüzey verir. Algoritma genel hatlarıyla sözde kod (pseudo-code) olarak şekil 3.16’da verilmiştir.

Algorithm $BPA(S, \rho)$

```
1. while (true)
2.   while ( $e_{(i,j)} = get\_active\_edge(\mathcal{F})$ )
3.     if ( $\sigma_k = ball\_pivot(e_{(i,j)}) \ \&\&$ 
        ( $not\_used(\sigma_k) \ || \ on\_front(\sigma_k)$ ))
4.       output_triangle( $\sigma_i, \sigma_k, \sigma_j$ )
5.       join( $e_{(i,j)}, \sigma_k, \mathcal{F}$ )
6.       if ( $e_{(k,i)} \in \mathcal{F}$ ) glue( $e_{(i,k)}, e_{(k,i)}, \mathcal{F}$ )
7.       if ( $e_{(j,k)} \in \mathcal{F}$ ) glue( $e_{(k,j)}, e_{(j,k)}, \mathcal{F}$ )
8.     else
9.       mark_as_boundary( $e_{(i,j)}$ )
10.  if ( $(\sigma_i, \sigma_j, \sigma_k) = find\_seed\_triangle()$ )
11.    output_triangle( $\sigma_i, \sigma_j, \sigma_k$ )
12.    insert_edge( $e_{(i,j)}, \mathcal{F}$ )
13.    insert_edge( $e_{(j,k)}, \mathcal{F}$ )
14.    insert_edge( $e_{(k,i)}, \mathcal{F}$ )
15.  else
16.    return
```

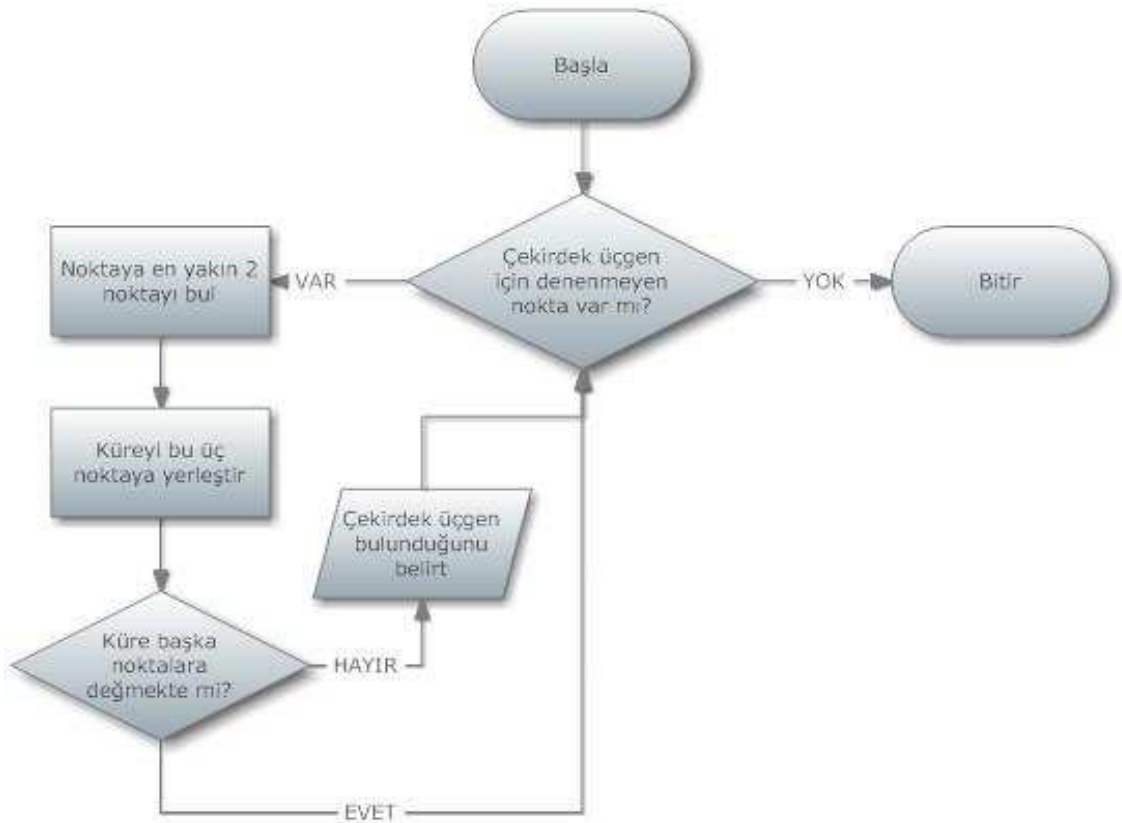
Şekil 3.16 TPA algoritması

Genellikle girdi olarak kabul edilecek nokta bulutları yüksek sayıda nokta içerirler. Algoritmada gerek çekirdek üçgenlerin belirlenmesinde gerekse varsayılan kürenin noktalara değip değmediğini belirlemede bütün noktalar değil yakın çevredeki noktalar değerlendirilmek istenecektir. Bunun sağlanması için “hacim ögesi” (ing. volume element, voxel) adı verilen birimlerden faydalanılır. Nasıl bir görüntü karesel resim ögelerinden oluşuyorsa 3B uzayın da kübik hacim ögelerinden oluştuğu düşünülebilir. TPA’da kullanılan kürenin kullanıcı tarafından belirlenen yarı çapı r iken hacim ögelerinin kenar uzunluğunu $2r$ alalım. Bu koşullarda, örneğin kürenin belirli bir pozisyonda hangi noktalara temas ettiği bulunmak istendiğinde yalnızca kürenin

merkezinin içinde bulunduğu hacim ögesine ve onun etrafındaki komşu hacim ögelerine (26 adet) bakmak yeterli olacaktır. Noktalar üzerinde yapılacak incelemeleri bu şekilde yerel bölgelerde sınırlandırmak yapılacak işlemlere harcanan süreyi oldukça kısaltmaktadır.

3.7.1 Çekirdek üçgenin belirlenmesi

Algoritmadaki önemli bir işlem çekirdek üçgenlerin belirlenmesidir. Bir çekirdek üçgenin özelliği, kürenin, üçgeni oluşturan üç noktaya değecek ama bu noktaların dışında hiçbir noktaya temas etmeyecek şekilde yerleştirilebilmesidir. Böyle bir üçgen bulmak için uygulanan adımlar şekil 3.17’deki akış diyagramında kısaca özetlenmiştir.

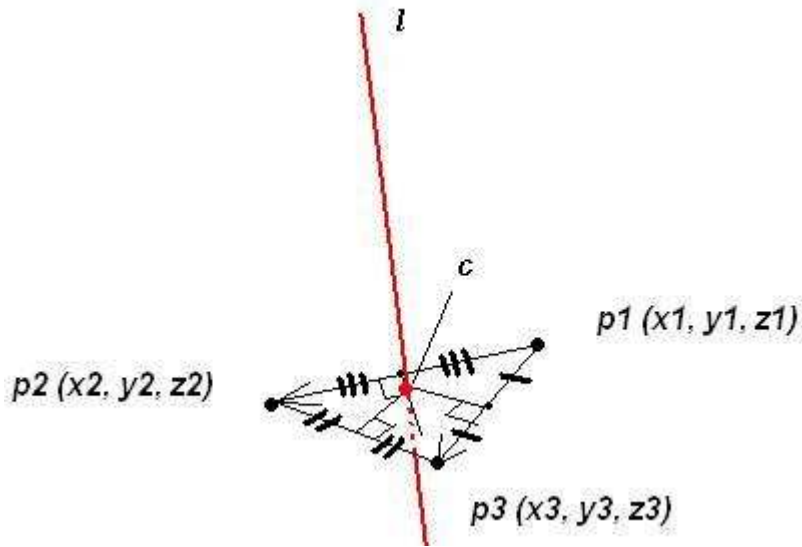


Şekil 3.17 Çekirdek üçgen bulma mantığı

Burada daha detaylı açıklama gerektiren bir alt işlem verilen 3 noktaya yarı çapı bilinen kürenin nasıl yerleştirildiğidir.

3.7.2 Verilen üç noktaya kürenin oturtulması

Bu işlemin anlatımı için şekil 3.18'deki 3 noktayı ele alalım. 3B uzayda verilen üç noktaya eşit uzaklıktaki noktalar l gibi bir doğru tanımlar. Yarıçapı r olan küre bu üç noktaya değmekte iken merkezi bu doğru üzerinde olacaktır. Bu sebeple ilk aranan, verilen üç noktadan bu doğru denkleminin nasıl elde edilebileceği olmuştur.



Şekil 3.18 Üç nokta ve onlara eşit uzaklıktaki noktaların tanımladığı doğru

Dikkat edilmesi gereken husus bu doğru ile üçgenin bulunduğu düzlemin kesiştiği noktanın, C 'nin özelliğidir. C ilk akla gelebileceği gibi üçgenin ağırlık merkezi değil, kenarların orta noktalarından çizilen dikey çizgilerin kesişim noktasıdır. Buradan yola çıkarak l doğrusunun şu şekilde bulunması düşünülmüştür:

- p_1 ve p_2 noktalarının orta noktası ve p_1 ve p_3 noktalarının orta noktası bu noktaların koordinatlarının ortalamaları alınarak kolaylıkla hesaplanabilir.
- Bulunan bu orta noktalar düzlemler üzerinde birer nokta ve sırasıyla $p_2 - p_1$ ve $p_3 - p_1$ vektörleri de düzlemlerin normal vektörleri olarak düşünülürse iki kenarın orta noktalarından geçen ve bu kenarlara dik olan iki düzlem elde edilmiştir. Bu iki düzlemin kesişim kümesi de l doğrusudur.

Elde edilen bu iki düzlem denklemi birlikte çözülmeye çalışılarak l doğrusunu veren

$$\begin{aligned}x &= x_c + tx_v \\l: y &= y_c + ty_v \\z &= z_c + tz_v\end{aligned}$$

şeklinde parametrik bir denklem elde edilebilir. Sonrasında yapılması gereken işlem ise bu doğru üzerinde hangi noktanın kürenin merkezine denk geleceğini hesaplamaktır. Bunun için de üçgen noktalarından herhangi biri seçilerek (örneğin p_1) o noktaya olan uzaklığın denkleminde yukarıdaki parametrik doğru denklemini de kullanılırsa:

$$(x_c + tx_v - x_1)^2 + (y_c + ty_v - y_1)^2 + (z_c + tz_v - z_1)^2 = r^2$$

gibi bir eşitliğe ulaşılacaktır. Burada tek bilinmeyen t parametresi için düzenlendiğinde ikinci dereceden bir denklem elde edilir. Bu ikinci dereceden denklemin çözülmesiyle doğru üzerinde bulunan ve p_1, p_2 ve p_3 noktalarına olan uzaklıkları r olan 2 noktayı verecek t 'ler bulunmuş olur.

Bulunan iki noktadan hangisinin kürenin merkezi olarak alınacağı ise tarama işlemindeki şu gerçekten yola çıkarak belirlenmektedir: Taranacak nesne, platform üzerine yerleştirilirken olabildiğince ortalanarak yerleştirildiği için y eksenini genellikle taranan yüzeyin arkasında kalmaktadır. Yüzeyden örneklenen noktalar üzerinde bir küre gezdirerek yüzeyi oluşturacak üçgenlerin belirlemesini sağlayan TPA için kürenin yüzeyin dışında kalması daha doğrudur. Bu sebeple bulunan 2 nokta arasında seçim yapılırken:

- Üçgenin merkezinden noktalardan ilkinde doğru olan vektör alınır;
- Bu vektörün, y ekseninden üçgenin merkezine gelen vektörle nokta çarpımı alınır;
- Elde edilen çarpım 0'dan büyük veya eşit ise birinci nokta, değil ise diğer nokta seçilir.

3.7.3 Kısmi yüzeyden döngüsel olarak topun pivotlanması

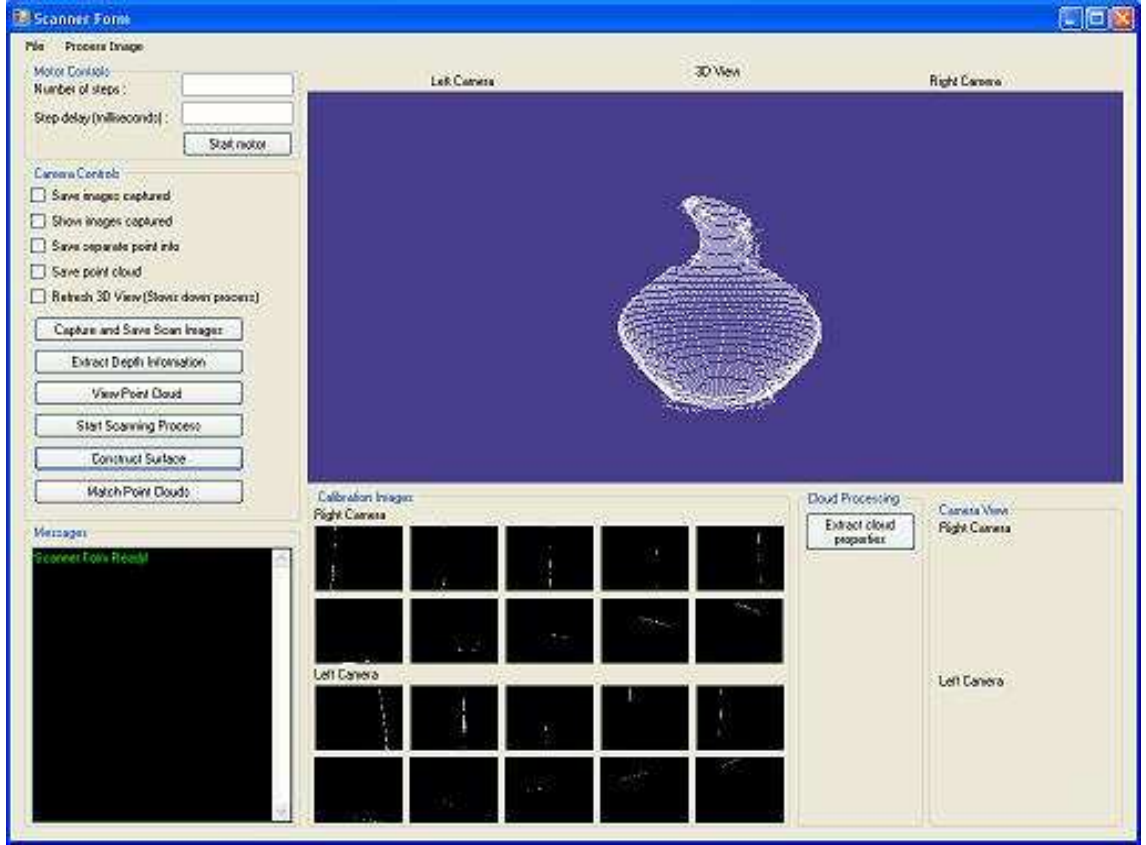
En az bir çekirdek üçgen elde edildiğinde kısmi bir yüzey elde edilmiştir. Algoritma bu noktadan itibaren kısmi yüzeyin kenarları üzerinde küreyi küçük açılarla döndürmeye başlar. Her adımda kürenin yeni bir noktaya temas edip etmediği kontrol edilir. Eğer küre hiçbir yeni noktaya temas etmeden yine aynı üçgene arka taraftan oturana kadar dönerse o kenar pivotlanmış kenar olarak kaydedilir ve pivotlanmamış diğer

kenarlardan biri seçilerek aynı işlem yapılır. Döndürme sırasında yeni bir noktaya temas ettiğinde ise pivotlanan kenarın uçlarındaki noktalarla yeni nokta yeni bir üçgen olarak kısmi yüzeye eklenir, topun üzerinde döndürüldüğü kenar pivotlanmış olarak kaydedilir ve pivotlanmamış diğer kenarlardan biri seçilerek aynı işlem tekrar eder. Pivotlanmamış kenar kalmadığında ise işlem yeni bir çekirdek üçgen bulmaya çalışır.

3.8 Yazılım

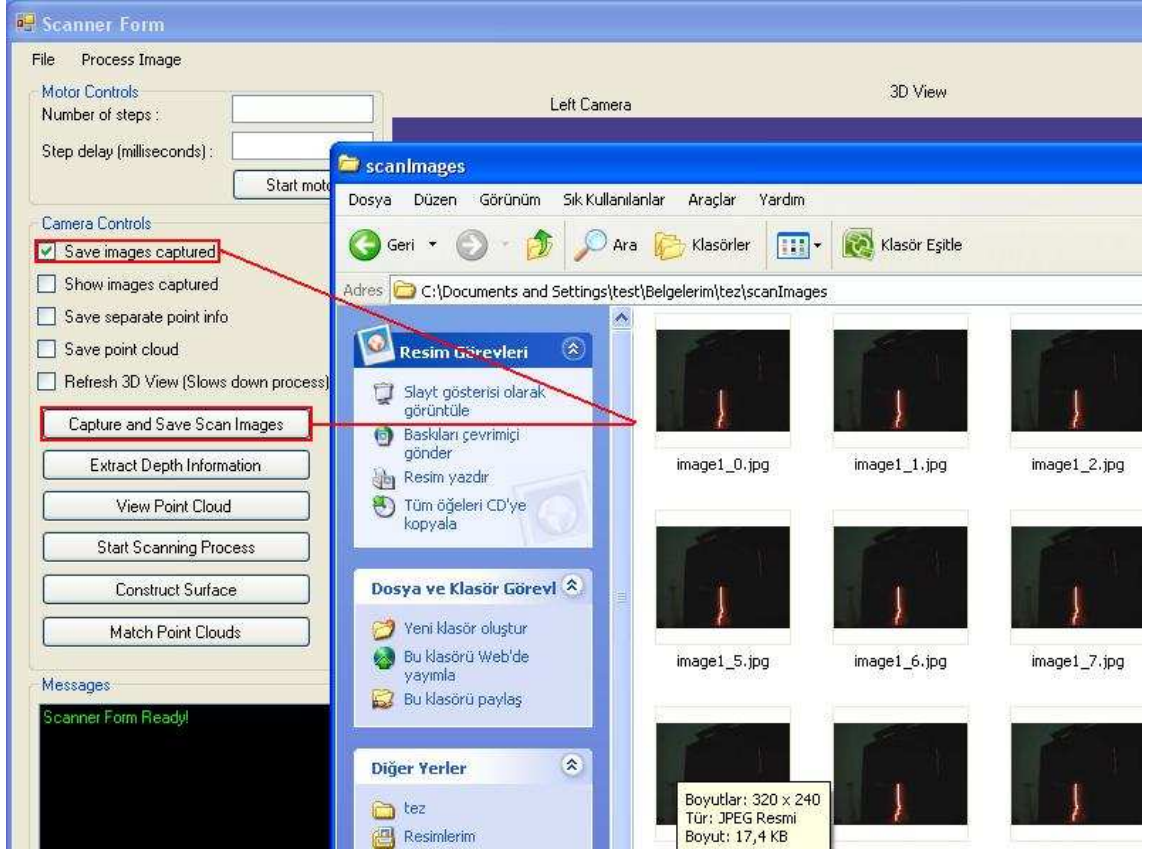
Tamamı C# programlama dili kullanılarak geliştirilen yazılım farklı işlevleri gerçekleştirmeye yönelik birbiriyle iletişim halindeki kısımlardan (modüllerden) oluşur. Bunlar:

- Döner masanın kontrolü için paralel LPT portu üstünden kontrol kartına komut gönderen ve adım motorunun hareketini sağlayan Motor Kontrol kısmı;
- Masanın hareketleriyle uyumlu bir şekilde otomatik olarak USB arayüze bağlı kameranın görüntü almasını sağlayan Kamera Kontrol kısmı;
- Ayarlama ve tarama süreçlerinde alınan görüntülere uygulanan ortak işlemleri kapsayan Görüntü İşleme kısmı;
- Ayarlama görüntülerinden çıkarılan resim ögesi pozisyonlarını kullanarak Φ fonksiyonunun katsayılarını hesaplayan Ayarlama kısmı;
- Tarama görüntülerinden gelen resim ögesi pozisyonlarını işleyerek her görüntü için 3B koordinatları hesaplayan Dönüşüm Uygulama kısmı;
- Her adımdaki görüntüden çıkarılan 3B koordinatları birleştirerek cismi simgeleyen bir 3B nokta bulutu oluşturan Rotasyon kısmı;
- 3B nokta bulutunu TPA ile üçgensel ağız yüzey gösterimine çeviren Yeniden Yapılandırma kısmı;
- Elde edilen görüntülerin, nokta bulutunun ve üçgensel ağız yüzey şeklindeki modelin bilgisayar ortamında gösterimini ve kullanıcı kontrollerini kullanım kolaylığı ve esnekliğini de göz önünde bulundurarak sağlayan Grafik Arayüz kısmı (Şekil 3.19).



Şekil 3.19 Grafik arayüzün genel görünümü

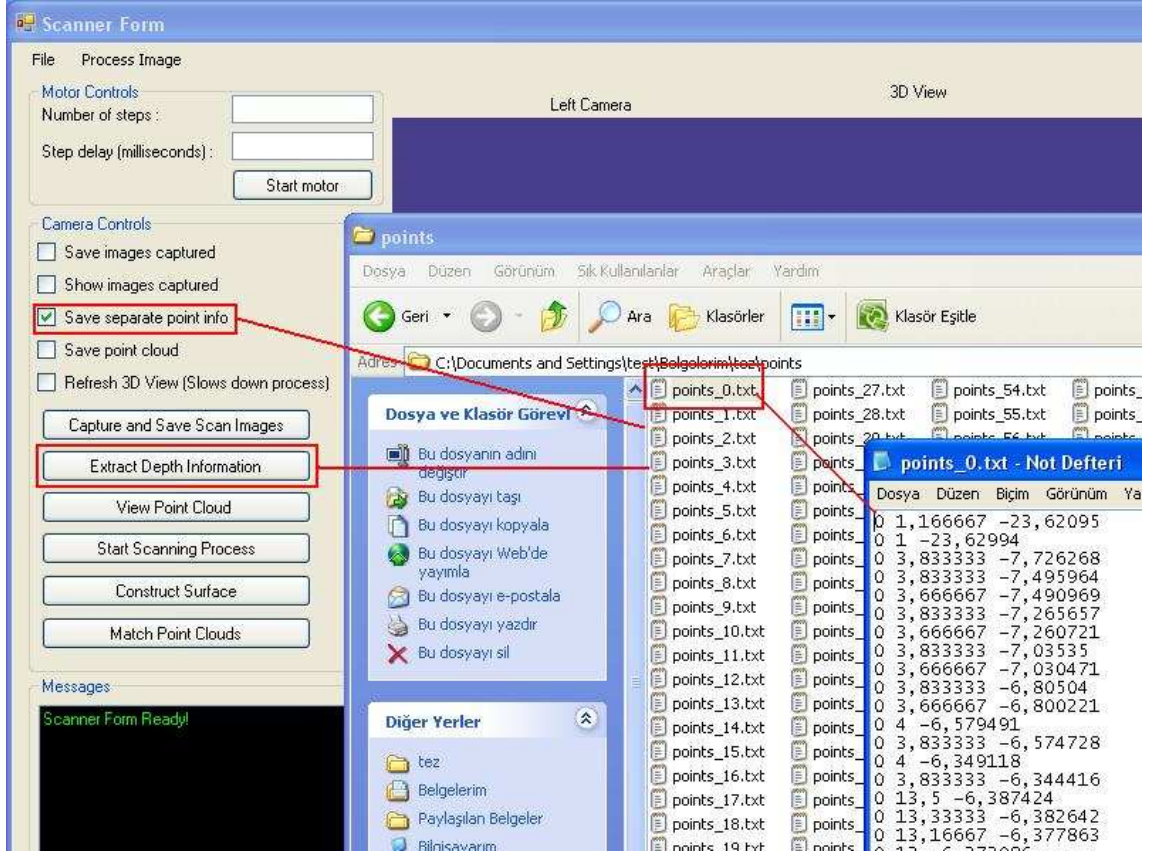
Yazılımdaki görüntü işleme kısmı yukarıda da belirtildiği gibi hem ayarlama görüntülerine hem de tarama görüntülerine uygulanan ortak algoritmaları barındırır. Bunlar bir görüntüyü renkli görüntüden yeşinlik görüntüsüne çeviren algoritma, eşikleme algoritması ve inceltme algoritmasıdır. Ayarlama kısmı girdi olarak kullanıcının sağladığı ayarlama görüntülerini alır, Görüntü İşleme biriminin hizmetlerinden yararlanarak Φ fonksiyonu için gerekli hesapları yapar ve bulunan Φ_{ij} değerlerini sistem değişkenleri olarak kaydeder. Standart bir tarama işleminde öncelikle Motor Kontrol ve Kamera Kontrol kısımları aracılığıyla otomatik şekilde her adımda platform döndürülüp, görüntü alınmaktadır. Alınan görüntüler kullanıcı tercih ettiği takdirde disk hafızasına kaydedilir (Şekil 3.20). Program içerisinde motor kontrolleri “inpout32.dll”, kamera kontrolleri de “user32.dll” ve “avicap32.dll” dinamik bağ kütüphaneleri (dynamic link libraries) aracılığıyla çalışmaktadır.



Şekil 3.20 Tarama görüntüleri ve hafızaya kaydedilmesi

Dönüşüm uygulama kısmı tercihe göre daha önce kaydedilen görüntü dosyaları üzerinde ya da o anda alınmakta olan görüntülerde işlem yapabilir. Yaptığı işlem Görüntü işleme biriminin imkânlarıyla resim öğelerini belirlemek ve Ayarlama biriminin hesapladığı 'yi bu resim öğesi pozisyonlarına uygulayarak her görüntü için 3B koordinat bilgileri elde etmektir. Bu ayrı ayrı veriler de yine kullanıcı istediği takdirde disk hafızasına kaydedilebilir (Şekil 3.21).

Yazılımın Rotasyon kısmı yine tercihe bağlı olarak kaydedilen ya da Dönüşüm Uygulama kısmından o anda kendisine gönderilmekte olan ayrı nokta bilgilerini adım numarasına göre doğru miktarlarda döndürerek taranan cisim simgeleyen 3B nokta bulutunu oluşturur. Oluşturulan nokta bulutunun ileride kullanılmak üzere disk hafızasına kaydedilmesi yine kullanıcının tercih edebileceği bir seçenektir.



Şekil 3.21 Çıkarılan 3B nokta bilgileri ve kaydedilmesi

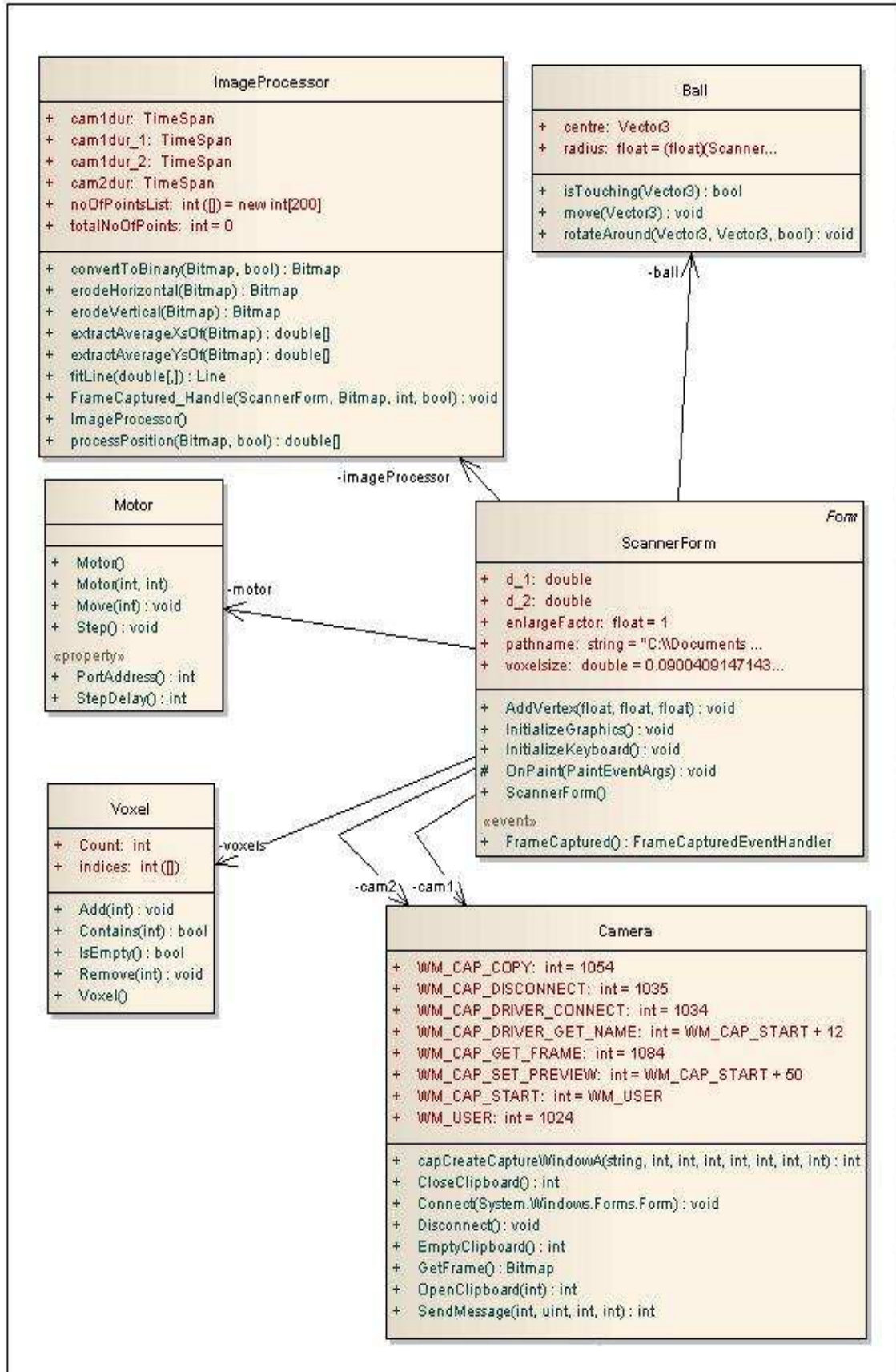
Grafik Arayüz birimi yukarıdaki şekillerde görüleceği gibi bir Windows Forms örneğidir. Bu form taramanın kontrolü için kullanılan işaret kutucukları ve düğmelerle, kullanıcıya bilgi sağlayan bir metin mesaj kutusu ve resim kutucukları barındırır. Bu kontrol ve bilgi öğelerinin yanı sıra en önemli öğesi şekil 3.19’da koyu mavi olarak görülen DirectX panelidir.

Oluşturulan nokta bulutları ve modeller bu panel içerisinde 3B bir sahne olarak gösterilmektedir. Kullanıcı klavye tuşlarıyla kolaylıkla modeli döndürebilmekte veya modele bakış açısını ve yerini değiştirebilmektedir. Microsoft’un DirectX SDK 9 paketi ile geliştirilen bu 3B gösterim, maalesef işlemler sırasında sürekli güncellendiği takdirde performansı kötü etkilemektedir. Örneğin tarama sırasında noktalar bulundukça burada görülmek istenirse tarama süresi kullanılan bilgisayarda 8-9 dakikaya ulaşırken 3B görüntü güncellenmediği zaman bu süre 3-4 dakika olmaktadır. Bu sebeple

işlemlersırasında 3B sahnenin güncellenip güncellenmeyeceđi yine kullanıcıya bir seçenek olarak sunulmuştur.

Şekil 3.19’da okuyucunun bu noktada dikkatini çekebilecek bir husus DirectX panelinin üzerindeki “Left Camera”, “Right Camera” yazıları ve “Match Point Clouds” düğmesi olabilir. Bu belirtici yazıların ve kontrol düğmesinin sebebi sistemin son halinde 2 kamera kullanılıyor olmasıdır. Bunun sebepleri ve amacı bölüm 4’de anlatılacaktır. Burada bilinmesi gereken DirectX panelinin aslında üç bölüme ayrılmış olmasıdır. Sol ve sağ kısımlarda farklı kameralardan elde edilen nokta bulutları ayrı ayrı gösterilebilmekte ve bunlar birleştirilerek ortada tek bir nokta bulutu oluşturulabilmektedir.

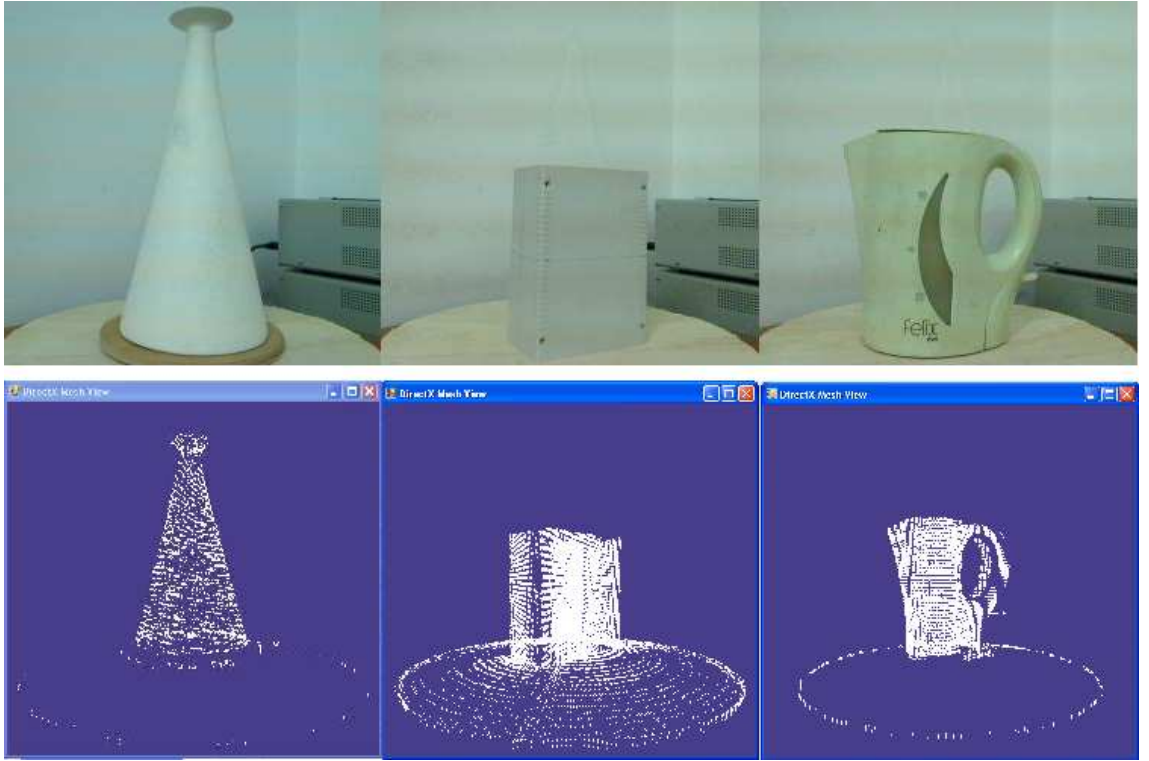
Yazılımın önemli kısımlarını ve bu kısımların kamusal (public) bileşenlerini gösteren genel bir sınıf (class) diyagramı şekil 3.22’de görülebilir.



Şekil 3.22 Oluşturulan yazılımın önemli sınıfları ve kamusal elemanları

4. BULGULAR

Sistemin çalışır hale getirildiği ilk zamanlarda taranan örnek cisimler ve elde edilen nokta bulutları Şekil 4.1’de verilmiştir. Şekilde de açıkça görüldüğü üzere başlangıçta elde edilen nokta bulutlarının kalitesi oldukça düşüktür. Alınan bu ilk sonuçlar incelenerek bazıları önceki kısımlarda belirtilen bazıları da yalnızca şekillere dikkat edilince fark edilebilen birçok iyileştirme yapılmıştır. Bunlardan bazıları, ayarlama görüntülerinin ve tarama görüntülerinin karanlıkta çekilmesi, zeminin ve arka planın koyu bir renge boyanması, lazer çizginin alt ve üst sınırlarının ve eşiklemede kullanılan eşik değerinin o an taranmakta olan nesneye göre ayarlanması olarak sayılabilir.



Şekil 4.1 İlk elde edilen nokta bulutları

Sayılan bu basit iyileştirmelerden sonra elde edilen nokta bulutlarında gözle görülür bir gelişme olmuştur. Şekil 4.2 bu iyileştirmelerden sonra taranan beş örnek cismi ve çıkarılan nokta bulutlarını göstermektedir.



Şekil 4.2 İyileştirmelerden sonra taranan cisimler ve oluşturulan nokta bulutları

Karşılaştırma amacıyla, hem gerçek nesnelerden hem de oluşturulan nokta bulutlarından bazı temel geometrik ölçümler alınmıştır. Bu ölçüm değerleri çizelge 4.1’de verilmektedir. Bu sonuçlar hem yükseklik hem de derinlik ayarlamaları için beşer ayarlama görüntüsü ile elde edilmiştir.

Çizelge 4.1 Gerçek cisimlerden ve nokta bulutlarından alınan bazı geometrik ölçümler

		Gerçek değerler (cm)	Noktabulutudeğerleri	Bağılhata
Konivazo	Çap	8,15	8,45726	0,0377
	Yükseklik	36,6	37,94335	0,0367
Kalemlik	Taban kenarı	6,5	6,71759	0,0335
	Yükseklik	15	15,49953	0,0333
Su ısıtıcısı	Dar çap	6,3	6,51421	0,034
	Geniş çap	8,1	8,42003	0,0395
	Yükseklik	23	23,87275	0,0379
Testi	En geniş çap	6,9	7,12828	0,0331
	Yükseklik	13,6	14,08976	0,036

Verilen bağıl hatalar nokta bulutu ölçümü ile gerçek ölçüm arasındaki farkın gerçek ölçüme bölümüyle, yani:

$$\text{bağıl hata} = \frac{\text{nokta bulutu ölçümü} - \text{gerçek ölçüm}}{\text{gerçek ölçüm}}$$

şeklinde hesaplanmıştır. Örneğin, koni vazonun çapı için bu hesap:

$$\frac{8,45726 - 8,15}{8,15} \cong 0,0377 = \%3,77$$

olarak hesaplanır. Hesap herhangi bir mutlak değer içermemektedir dolayısıyla negatif veya pozitif olabilmektedir. Bu özellik bize nokta bulutundaki noktaların koordinatlarının gerçek yüzeyin koordinatlarına göre daha büyük mü yoksa daha küçük mü hesaplandığını da göstermektedir.

Tarayıcı sistemi için ortalama bağıl hata çizelge 4.1’den $\bar{e} = \%3,57$ olarak hesaplanmıştır. Yaklaşık olarak bu ortalama bağıl hata her bir nokta bulutundaki her noktaya etkili olmaktadır. Bu büyüklükte hata yeniden oluşturulan cisim şeklinin çıplak göze görünümünü etkilememektedir. Yine de bu ortalama bağıl hata yardımıyla modeller matematiksel olarak iyileştirilebilir. Elde edilen 3B koordinatlar bu ortalama sapma hatasına göre oranlanarak nokta bulutundaki noktaların gerçek yüzeydeki karşılıklarına daha da yaklaştırılması düşünülmüştür.

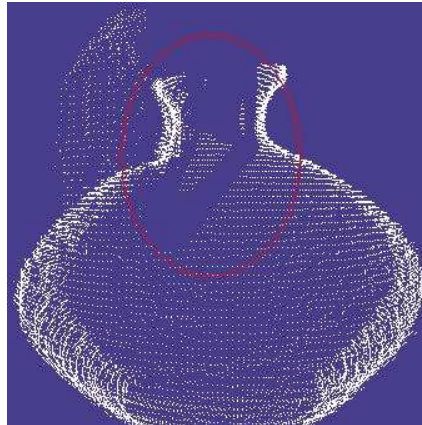
Bunun için nokta bulutlarındaki her nokta ($1 + \bar{\epsilon}$) değerine bölünerek sanal nesneler az miktarda küçültülmüştür. Bunun üzerine elde edilen sanal nesne ölçümleri ve bunlara karşılık gelen bağıl hata ölçümleri çizelge 4.2’de verilmiştir.

Çizelge 4.2 Oranlanmış nokta bulutu değerleri ve bağıl hatalar

		Gerçekdeğerler (cm)	Noktabulutudeğerleri	Bağılhata
Konivazo	Çap	8,15	8,16574	0,0019
	Yükseklik	36,6	36,63546	0,001
Kalemlik	Taban kenarı	6,5	6,48604	-0,0021
	Yükseklik	15	14,96527	-0,0023
Su ısıtıcısı	Dar çap	6,3	6,28967	-0,0016
	Geniş çap	8,1	8,12980	0,0037
	Yükseklik	23	23,04987	0,0022
Testi	En geniş çap	6,9	6,88257	-0,0025
	Height	13,6	13,60409	0,0003

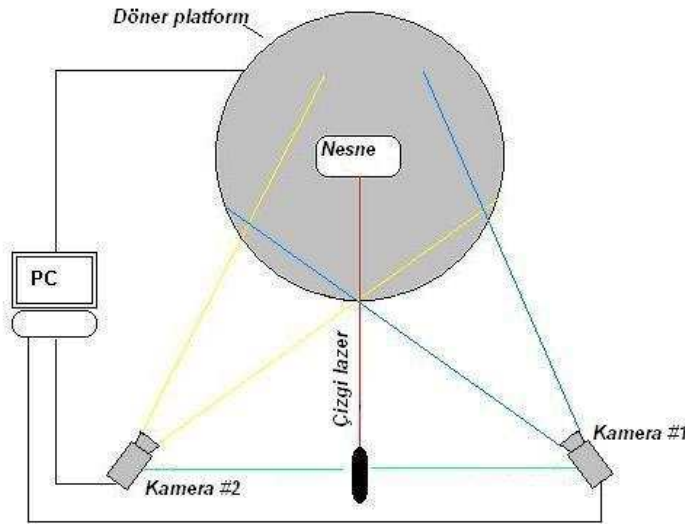
Matematiksel olarak da bekleneceği gibi noktalar ortalama bağıl hata payına göre oranlandığında bağıl hata değerleri de önceki ortalama kadar 0’a doğru gitmiştir. Yeni bağıl hata ortalaması $\bar{\epsilon}' = 6 \times 10^{-5}$ ’tir.

Bu incelemeler ve iyileştirmeler sırasında tek kameralı sistemin tam bir tarama yapmasını zorlaştıran önemli bir etken de belirlenmiştir. Kendi kendini örten ya da üzerinde ani derinlik değişimleri barındıran nesnelerde çizginin kısmen ya da tamamen kamera görüntüsü dışında kalabileceğine bölüm 2.5’te örnek bir şekilde (Şekil 2.7) beraber değinilmişti. Bu problemin sonuca etkisi nokta bulutlarında boşluklar olarak ortaya çıkmaktadır (Şekil 4.3).



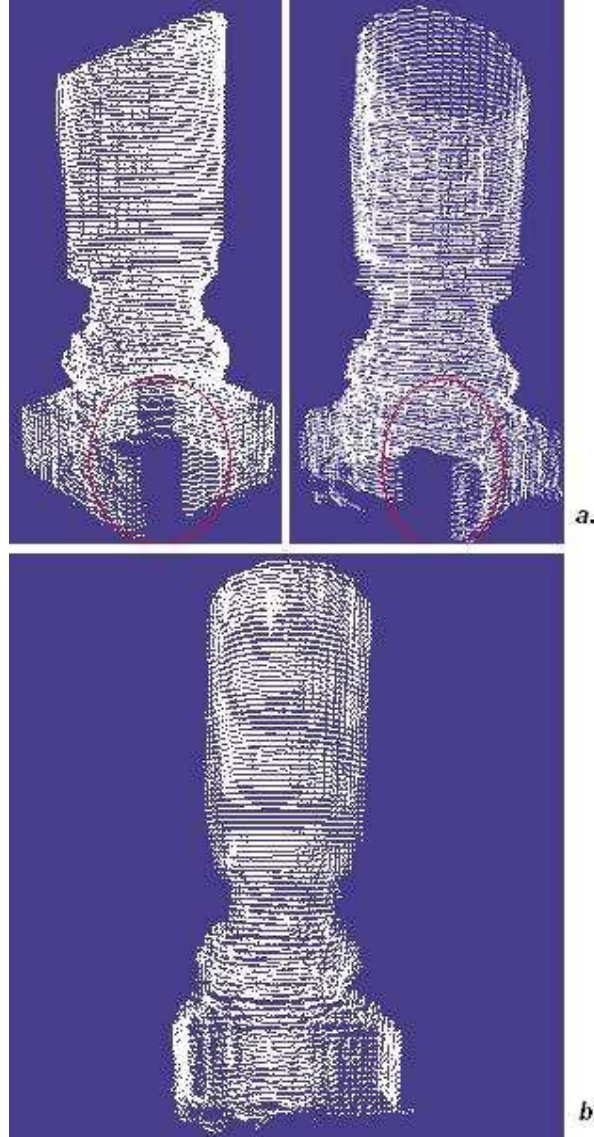
Şekil 4.3 Kendi kendini örtme sonucu nokta bulutunda oluşan boşluk

Bu problemin ortadan kaldırılması için sisteme ikinci bir kamera eklenmesine karar verilmiştir. Görüş açıları dikkatlice belirlenen iki ayrı kameradan elde edilen nokta bulutları birleştirildiğinde birbirleri üzerinde eksik olan kısımları tamamlamaları beklenmektedir. İkinci kamera sisteme eklendikten sonraki kurulum şematik olarak şekil 4.4’te verilmiştir.



Şekil 4.4 İki kameralı sistemin yukarıdan şematik görünümü

Tarama sistemlerinde nesneyi farklı açılardan ya da farklı zamanlarda taramak çoklu derinlik verilerine sebep olur ve genellikle kısmen örtüşen bu ayrı verileri en iyi dönüşümü hesaplayarak üst üste getirmek gerekmektedir. Bu işleme çakıştırma veya hizalama (registration) adı verilmektedir. Bu tezde oluşturulan sistemde iki kameradan elde edilen nokta bulutlarını hizalamak için karmaşık bir dönüşüm hesabına gerek duyulmamaktadır, çünkü her iki kamera da eşzamanlı görüntü almakta ve bu görüntüleri işleyen teknikler koordinat sistemiyle ilgili aynı varsayımları yapmaktadırlar. Dolayısıyla iki kameradan elde edilen nokta bulutları kendi kendini örtme probleminden kaynaklanan boşluklar dışında zaten hizalanmış durumdadırlar. Karmaşık herhangi bir hizalama işlemi yapılmadan sadece küme birleşimiyle oluşturulan nokta bulutu ve ayrı nokta bulutları şekil 4.5’te görülmektedir.



Şekil 4.5a. Farklı kameralardan oluşturulan boşluklu nokta bulutları,b. ikisinin üst üste getirilmesiyle (küme birleşimi) oluşan nokta bulutu

Matematiksel olarak, 1. kameradan elde edilen nokta bulutuna $\mathbf{P}_1$, 2. kameradan elde edilen buluta da $\mathbf{P}_2$ diyelim. $|\mathbf{P}_1| \neq |\mathbf{P}_2|$ olduğu hiçbir zaman unutulmamalıdır. İki nokta bulutunun birleşimi alındığında elde edilen nokta bulutu $\mathbf{P} = \mathbf{P}_1 \cup \mathbf{P}_2$ olur. Sadece küme birleşimi alınarak elde edilen böyle bir nokta bulutundaki noktalar iki gruba ayrılabilir:

- (i) Karşılık geldiği yüzey noktasının oldukça yakınında bir bölge için diğer nokta bulutundan gelen en az bir nokta bulunan noktalar,

- (ii) Karşılık geldiği yüzey noktasının yakın çevresi için boşluk dolayısıyla diğer nokta bulutundan gelen bir nokta bulunmayan noktalar.

Bu gruplar düşünülerek, aynı nokta bulutundan gelen iki yakın noktayı bir araya getirmeyecek boyutlarda küçük bölgeler içerisinde bir ortalama alma işlemi uygulanması düşünülmüştür. Böyle bir istatistiksel uygulama ile amaçlanan, farklı bulutlardan gelen, hemen hemen aynı bölgeyi temsil eden noktaların birleştirilerek, noktalar için hata paylarının ortalama etkisiyle azaltılması ve indirgenmiş bir nokta bulutu P' elde edilmesidir.

Bu istatistiksel uygulamadaki ana etken ortalama işlemi uygulanacak bölgelerin boyutudur. Aynı buluttan gelen ayrı ayrı örneklenmiş iki noktanın bir araya getirilmemesi için örnekleme mesafesi hakkında bilgi sahibi olmak gerekir. Malesef oluşturulan sistemde sabit ve kesin bir örnekleme mesafesinden bahsedebilmek mümkün değildir çünkü yüzeyden alınan iki örnek arasındaki mesafe yüzeyin o noktalarda dönme eksenine olan mesafesiyle değişmektedir. Bu yüzden nokta bulutlarının kendisinden elde edilen, örnekleme mesafesi yerine kullanılabilecek, ortalama asgari uzaklık (OAU) adı verilen bir değeri benimsenmiştir. Bu değerın hesaplanması için:

- Öncelikle nokta bulutu içindeki her nokta için kendisine en yakın nokta ile arasındaki mesafe hesaplanır;
- Sonra da bu en yakın mesafelerin aritmetik ortalaması alınır.

İstatistiksel uygulamada kullanılan bölge boyutları, nokta bulutlarının OAU değerleri d_1 ve d_2 'den daha küçük olana göre farklı katsayılarla belirlenerek denemeler yapılmıştır. Kullanılan örnek nokta bulutlarının özellikleri şu şekildedir:

$$|P_1| = 34447,$$

$$\bar{d}_1 = 4,62789 \times 10^{-2},$$

$$|P_2| = 33593,$$

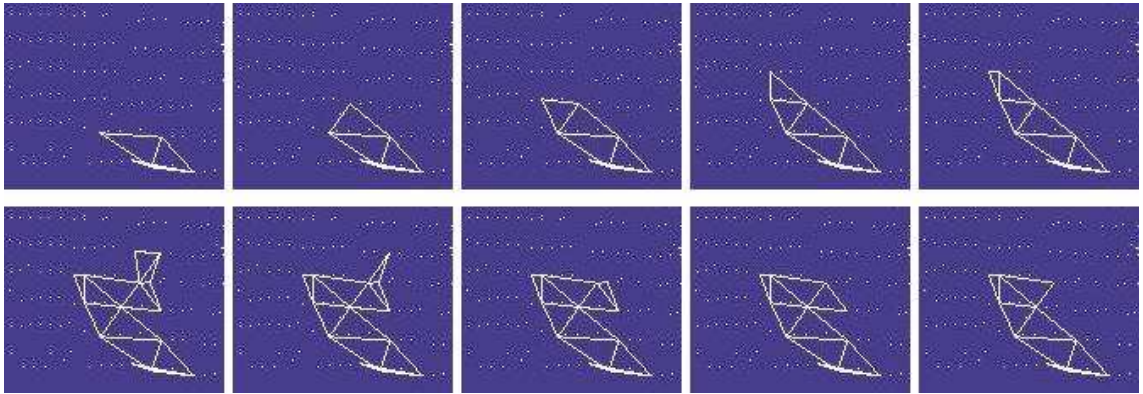
$$\bar{d}_2 = 4,80166 \times 10^{-2}$$

Bu deneme sonuçlarında elde edilen indirgenmiş nokta bulutlarının nokta sayılarının kümelerden büyük olana oranları ve OAU değerleri Çizelge 4.3’de verilmiştir.

Çizelge 4.3 Farklı bölge boyutlarıyla indirgenmiş nokta bulutlarındaki nokta sayıları $|P'|$ ve OAU değerleri $\bar{d}'$

$\frac{\text{bölge boyutu}}{\min(\bar{d}_1, \bar{d}_2)}$	$ P' $	$\frac{ P' }{\max(P_1 , P_2)}$	$\bar{d}'$
0,75	61711	1,79	$4,92673 \times 10^{-2}$
0,875	59072	1,72	$5,13618 \times 10^{-2}$
0,9375	57718	1,68	$5,24073 \times 10^{-2}$
0,96875	56998	1,66	$5,30786 \times 10^{-2}$
0,984375	56759	1,65	$5,32767 \times 10^{-2}$
1	56373	1,64	$5,3621 \times 10^{-2}$

İki kameradan alınan görüntülerle oluşturulan nokta bulutları birleştirildikten sonra bölüm 3.7’de anlatılan yeniden yapılandırma işlemi uygulanmaktadır. TPA’nın bir çekirdek üçgen bularak bu üçgenden döngüsel olarak ilerlerken yeni üçgenler buluşu şekil 4.6’da görülmektedir.



Şekil 4.6 Oluşturulan nokta bulutunda TPA uygulanmasına örnek

Daha önce de belirtildiği gibi bu çalışmada TPA kullanılmasının bir avantajı, topun yarıçapının noktalar elde edilirken ortaya çıkan örnekleme mesafesine bağlı olarak belirlenmesinin uygun olmasıdır. Bu noktada bir kez daha yukarıda belirtilen OAU değerleri kullanılarak, değişen yarı çaplar ile üç grup test yapılmıştır:

- (i) Bölge boyutu $\min(\bar{d}_1, \bar{d}_2)$ değerinin 2 katı alınarak indirgenmiş nokta bulutu üzerinde yarıçap (r) bölge boyutunun c katı alınarak

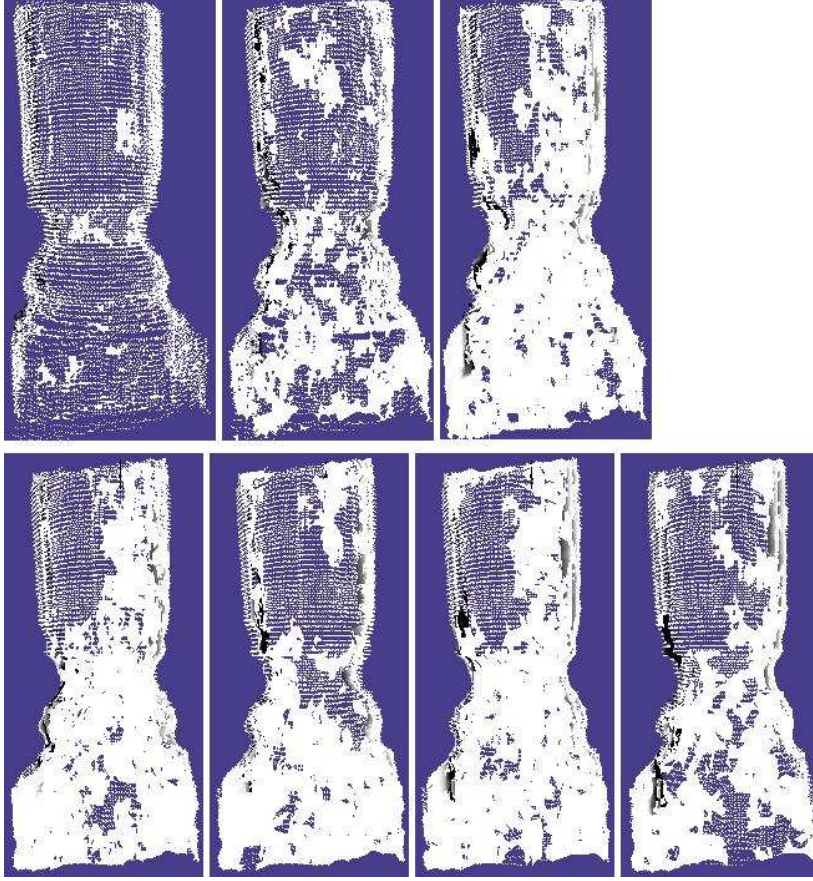
- (ii) (i) ile aynı şekilde indirgenen nokta bulutu ve r bu nokta bulutunun kendi OAU değeri $\bar{d}' = 7,64816 \times 10^{-2}$ 'nin c katı alınarak
- (iii) İndirgeme yapılmamış nokta bulutu ve r bu nokta bulutunun kendi OAU değeri $\bar{d}' = 4,49917 \times 10^{-2}$ 'nin c katı alınarak

Her test için bu şekilde hesaplanan r değerleri çizelge 4.4'te verilmiştir.

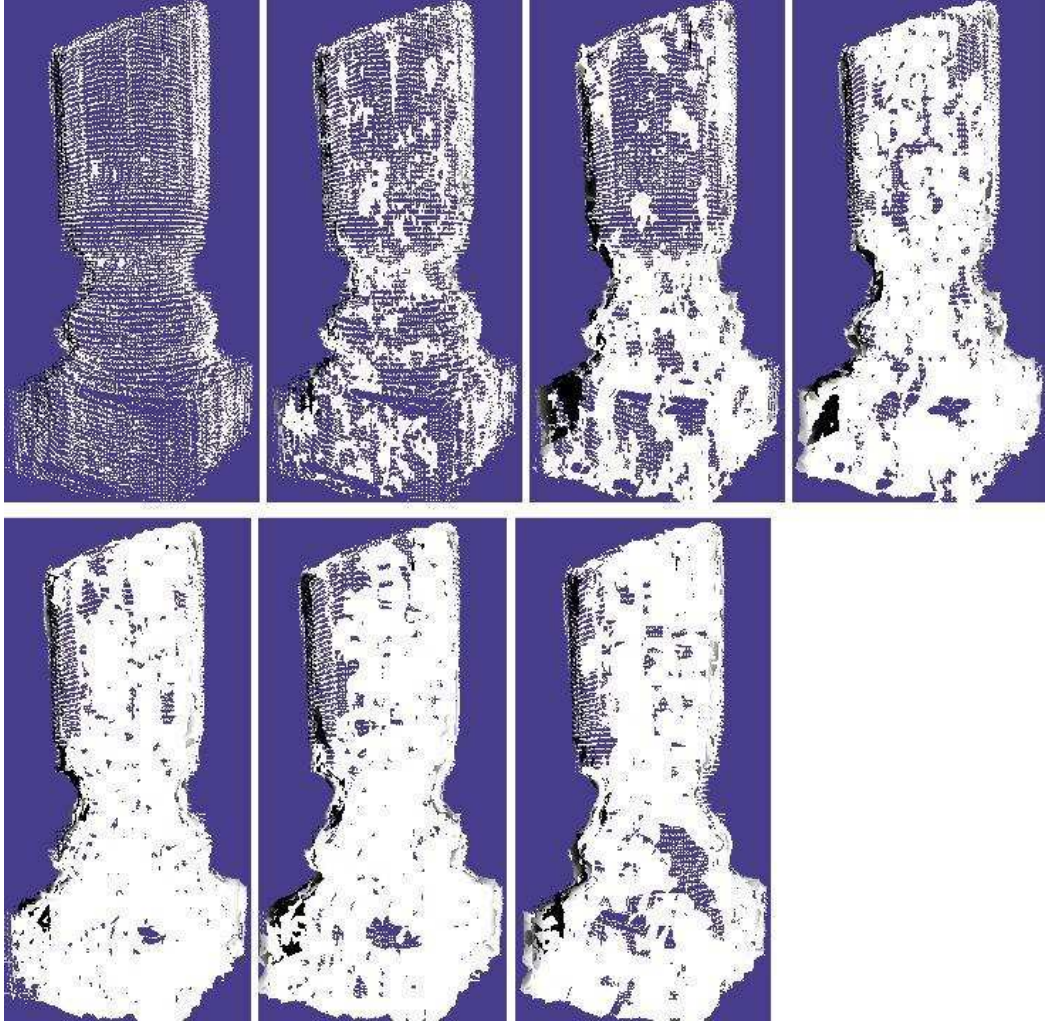
Çizelge 4.4 Her testteki r değerleri

c	i.	ii.	iii.
1	9.25578×10^{-2}	7.64816×10^{-2}	4.49917×10^{-2}
1.5	1.388367×10^{-1}	1.147224×10^{-1}	6.748755×10^{-2}
2	1.851156×10^{-1}	1.529632×10^{-1}	8.99834×10^{-2}
2.5	2.313945×10^{-1}	1.91204×10^{-1}	1.1247925×10^{-1}
3	2.776734×10^{-1}	2.294448×10^{-1}	1.349751×10^{-1}
3.5	3.239523×10^{-1}	2.676856×10^{-1}	1.5747095×10^{-1}
4	3.702312×10^{-1}	3.059264×10^{-1}	1.799668×10^{-1}

Aşağıda şekil 4.7-4.9'a kadar olan görüntüler sırasıyla test sonuçlarını göstermektedir. Bu sonuçlar oldukça basit bir TPA gerçekleştirilmesiyle elde edilmiş olup incelenirken dikkat edilmesi gereken ana nokta üçgenlerin sayısı ve oluşan yüzeyin kapladığı alandır. Her şekilde öncelikle soldan sağa doğru gidilip sonra alt satıra geçilerek artan c değerleri için sonuçlar yan yana konulmuştur.

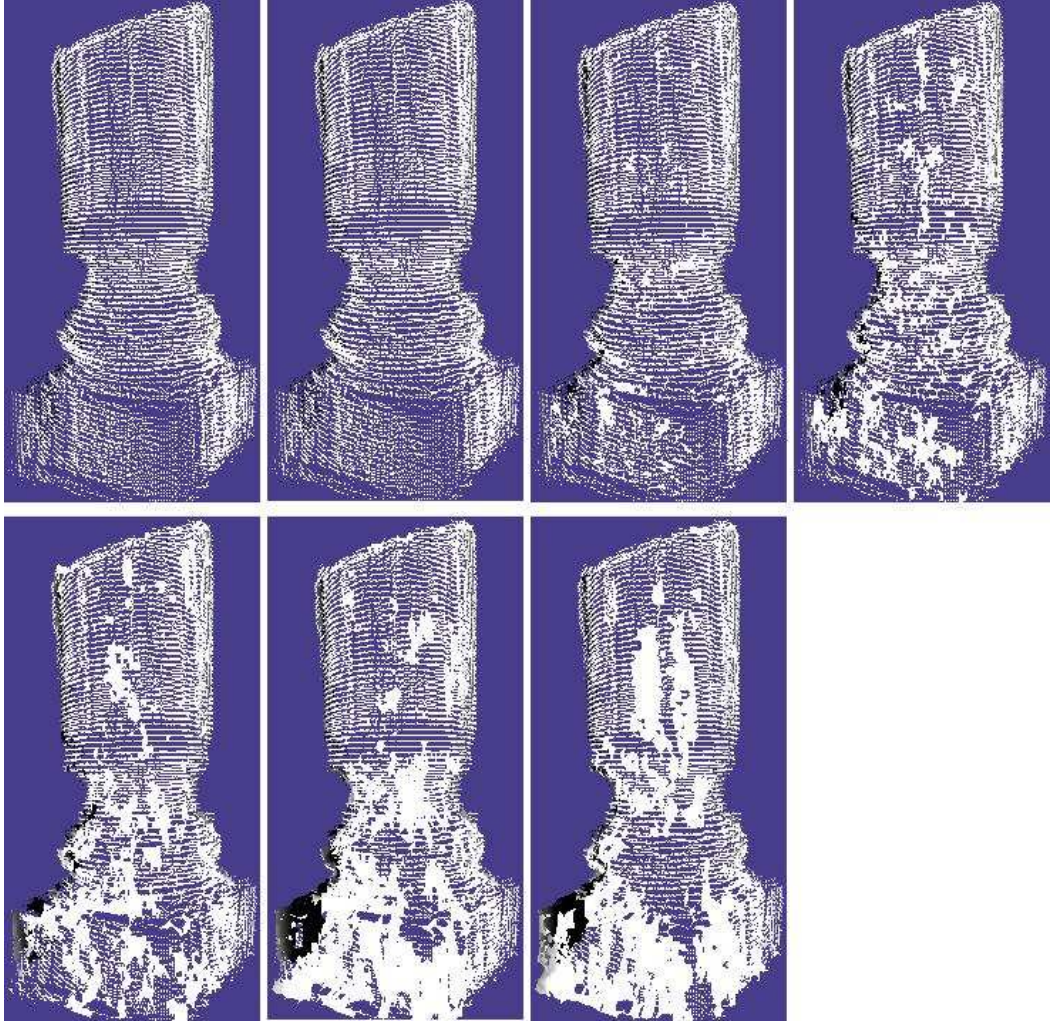


Şekil 4.7 Farklı c değerleri için test grubu (i) sonuçları



Şekil 4.8 Farklı c değerleri için test grubu (ii) sonuçları

Şekil 4.7’de topun yarıçapının bölge boyutunun yaklaşık 2,5 katı olduğu durumda en iyi sonuç elde edildiği görülebilir. Yarıçap 3,5 kat olarak alındığında da oluşturulan yüzey geniş bir alan kaplasa da bu kadar büyük bir topun pivotlanması ince yüzey detaylarını kaçırmaktadır. Şekil 4.8’de ise OAU değerinin 3 katı ile elde edilen beşinci görüntü en iyi sonucu göstermektedir. Bu iki test grubundan elde edilen sonuçlarda $2 \times \min(\bar{d}_1, \bar{d}_2)$ boyutunda bölgeler ile indirgenmiş bir bulutta, kullandığımız örnekleme mesafesi belirtecinin yaklaşık 2,5-3 katı bir top yarıçapı ile yeniden yapılandırmanın daha başarılı olduğu söylenebilir.



Şekil 4.9 Farklı c değerleri için test grubu (iii) sonuçları

Şekil 4.9’da ise çok az miktarda yeniden yapılandırılmış yüzey gözlemlenmektedir. Bunun ana sebebi indirgeme yapılmadığında OAU değerinin örnekleme mesafesi için kötü bir belirteç haline gelmesidir. İki nokta bulutunda yakın iki noktayı temsil eden noktalar bir araya geldiğinde aralarındaki mesafenin çok kısa olması, top yarıçapı OAU’ya göre belirlendiğinde sınırlı yeniden yapılandırmaya sebep olmaktadır.

5. TARTIŞMA VE SONUÇ

Genel olarak bu tezdeki amaç bir 3B tarayıcı inşa etmek ve bu tarayıcıyı gerçek nesnelerin 3B sanal modellerini çıkarabilmek için hazırlamaktır. İnşa edilecek tarayıcı için bir tasarım ortaya konmadan önce literatür araştırmaları ile benzer çalışmalar ve tarayıcıların yoğunlukla kullanıldığı projeler incelenmiştir. Bu incelemeler ve belirtilen bazı varsayımlar ışığında üçgenlemeye dayalı, biçimli ışık kullanan optik bir tarayıcı inşa edilmesine karar verilmiştir.

Çalışma prensibi seçilen tarayıcı için ilk tasarım gerçekleştirilmiş ve bu tasarımı meydana getirmek için gerekli donanım parçaları temin edilmiştir. Temin edilen donanım parçalarıyla ve yazılım kısımlarının da geliştirilmeye başlanmasıyla öncelikle döner platform mekanizması çalıştırılmış, sonrasında bir kamera ve çizgi lazer de sisteme eklenerek model çıkarımı için çalışmalara başlanmıştır. Bu noktadan sonra ağırlıklı olarak yazılıma dayalı olan çalışmalar sonucunda ilk nokta bulutları elde edilmiştir.

Ulaşılan ilk sonuçlar incelenerek bazıları fiziksel ortamda küçük değişiklikler gerektiren bazıları da yazılımda eklemeler gerektiren iyileştirmeler yapılmıştır. İyileştirmelerin ardından gerçeğe yakınlıkları ölçülmeye geçecek kalitede nokta bulutlarına ulaşıldığı kabul edilmiştir. Yapılan ölçümler ortalama %3-%4 civarında bir bağıl hata ortaya koymuştur. Bu seviyedeki hataların çıplak gözle modellere bakıldığı zaman belirgin olmadığı kanısına varılmıştır. Yine de basit matematiksel bir oranlama ile bu hatanın nasıl azaltılabileceği de belirtilmiştir.

Nokta bulutlarında dikkati çeken ve basit iyileştirmelerle giderilemeyecek olan bir başka problem de görülmüştür. Kamera görüntüsünde cismin bir kısmının lazer ışığının önüne geçmesi dolayısıyla tek kamera ile çıkarılan nokta bulutlarında boşluklar oluşması mümkündür. Bu boşlulardan kurtulmaya yönelik olarak sisteme ikinci bir

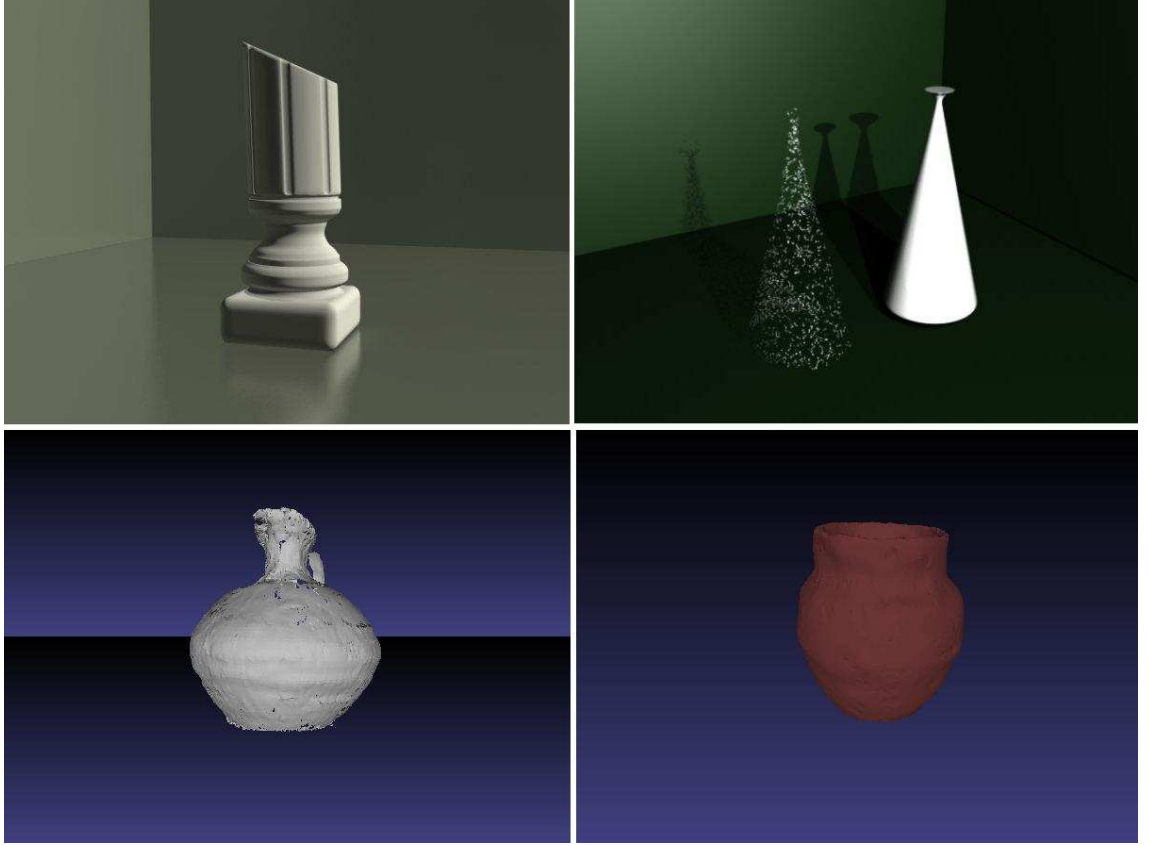
kamera eklenmiş ve iki kameradan ayrı ayrı elde edilen nokta bulutlarının birleştirilmesi değerlendirilmiştir.

Nokta bulutlarının birleştirilmesinde yerel bir istatistiksel ortalama metodunun sonuç olarak elde edilen nokta bulutlarına etkisi test edilmiştir. Bunun yanı sıra TPA ile nokta bulutlarından yüzey modelleri elde etmek için kullanılacak kavramsal topun yarıçapının nokta bulutlarının özellikleri incelenerek belirlenmesi değerlendirilmiş ve farklı yarıçap değerleri için yeniden yapılandırma sonuçları verilmiştir. Örnek cisimlerden tarama ve yeniden yapılandırma sonucunda elde edilen modeller şekil 5.1’de görülmektedir.

Hemen hemen her çalışmada olduğu gibi burada da tarayıcının ve elde edilen modellerin daha fazla geliştirilmesi mümkündür. Mümkün olan birçok geliştirmeden öncelikli olarak denenmesini düşündüklerimiz:

- Basit web kameraları yerine daha yüksek çözünürlükle daha kaliteli görüntü alabilen (dolayısıyla da maliyeti biraz daha yüksek) kameralar kullanılırsa hem düşey örnekleme sıklığı arttırılır hem de EKK kıstasında kullanılan örnek noktaların sayısı arttırılmış olur. EKK kıstasına göre kestirim yapılırken, başlangıçta kullanılan örneklenmiş noktalarının sayısının artmasıyla kestirimin iyileşmesi bilinen ve kabul edilen bir gerçektir (Giordanov vd. 2003). Dolayısıyla görüntülerin daha yüksek çözünürlükle alınmasının iki farklı yönden isabet oranını arttırması beklenir.
- Yine EKK kıstasında kullanılan örnek nokta sayısını arttırmanın diğer bir yolu daha fazla ayarlama görüntüsü kullanmaktır. Ayarlama cismi ne kadar farklı pozisyona yerleştirilerek görüntü alınır ve bunlar  fonksiyonuna yapılan yakınsamada kullanılırsa nokta bulutlarının isabetliliğinin de o kadar artması beklenecektir.
- Sonuçta ortaya çıkan hataların bir kısmı 3B görüntü alınması sırasındaki optik girişimlerden kaynaklanmaktadır. Bunlarla ilgili, bölüm 2.5’inson kısmında bahsi geçen, yakın zamanlı araştırmalarda ortaya konan başa çıkma yöntemleri sistemin çalışma zamanından ödün verildiği takdirde denenebilir.

Kullanılan algoritmaların programa dökülürken zaman ve hafıza harcamaları sürekli değerlendirilmiş olmasına rağmen son hallerinin optimum olduğu iddia edilmemektedir. Yazılan programın kontrol mekanizmaları ve kullanılan veri yapıları gözden geçirilip iyileştirilerek tarama süresi ve kullanılan hafıza miktarının azaltılması mümkün olabilir. Halen yukarıda belirtilenler gibi bazı geliştirmeler yapılabilecek olmasına rağmen meydana getirilen sistem denenen sınırlı sayıdaki cisim için kabul edilebilir seviyede geometrik bilgi çıkartabilmiştir. Tarama sonuçlarının yanı sıra tarama süresi de bir cisim için projede kullanılan bilgisayar ile yaklaşık 3-4 dakikadır ve bu süre de yeterli kalitede model çıkartabilen bir sistem için makul görülmüştür.



Şekil 5.1 Örnek cisimlerden elde edilen modeller

KAYNAKLAR

- Ahn, S.J., Rauh, W. and Recknagel, M. 1999. Geometric fitting of line, plane, circle, sphere and ellipse. In Proc. of 6th ABW-Workshop Optische 3D-Formerfassung, 1-8. Technical Academy Esslingen.
- Benveniste, R. and Ünsalan, C. 2006. Using color indices to overcome the scattering problem in line laser scanners. 14. IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SIU'2006).
- Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C. and Taubin, G. 1999. The Ball-Pivoting Algorithm for surface reconstruction. IEEE Transactions on Visualization and Computer Graphics, 5(4), 349-359.
- Bernardini, F. and Rushmeier, H. 2002. The 3D model acquisition pipeline. Computer Graphics Forum, 21(2); 149-172.
- Bernardini, F., Martin, I., Mittleman, J., Rushmeier, H. and Taubin, G. 2002. Building a digital model of Michelangelo's Florentine Pieta. IEEE Computer Graphics & Applications, 22(1), 59-67.
- Blais, F., Levoy, M. and Godin, G. 2004. Accurate 3D acquisition of freely moving objects. 2nd Int. Symp. On 3D Data Processing, Visualization and Transmission.
- Curless, B. and Levoy, M. 1995. Better optical triangulation through space time analysis. IEEE Int. Conf. On Computer Vision, 987-994.
- Curless, B. and Levoy, M. 1996. A Volumetric Method for Building Complex Models From Range Images. ACM SIGGRAPH, pp. 303-312, New York.
- Curless, B. 1999. From range scans to 3D models. Computer Graphics, 33(4).
- Davis, X. and Chan, J. 2001. A Laser Range Scanner Designed for Minimum Calibration Complexity. 3rd Int. Conf. on 3-D Digital Imaging and Modelling (3DIM'01), pp. 91.
- Dellepiane, M., Callieri, M., Ponchio, F. and Scopigno, R. 2007. Mapping highly detailed color information on extremely dense 3D models: the case of David's restoration. Proc. of Eurographics Conference.
- Gibilisco, S. and Crowhurst M.H. 2007. Mastering Technical Mathematics. McGraw-Hill Professional. p. 347. New York.
- Giordanov, F.R., Weir, M.D. and Fox, W.P. 2003. A First Course in Mathematical Modeling. Thomson Brooks/Cole, 114.
- Godding, R. and Luhmann, T. 1992. Calibration and accuracy of a multi-sensor online photogrammetric system. Int. Archives of Phot. & Remote Sensing, 29(B5), 24.
- Gonzales, R.C., Woods, R.E. and Eddins, S.L. 2004. Digital Image Processing using Matlab. Prentice Hall – Pearson. USA.

- Guidi, G., Micoli, L., Russo, M., Frischer, B., De Simone, M., Spinetti, A. and Carosso, L. 2005. 3D digitization of a large model of imperial Rome. In Proc. of 5th Int. Conf. on 3-D Digital Imaging and Modeling, 565-572.
- Hartley, R.I. and Sturm, P. 1997. Triangulation. *Computer Vision and Image Understanding*, 68, 146-157.
- Hilton, A., Stoddart, A.J., Illingworth, J. and Windeatt, T. 1996. Marching triangles: range image fusion for complex object modelling. In Proc. of Int. Conf. on Image Processing, 381-384.
- Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W. 1992. Surface reconstruction from unorganized points. *ACM SIGGRAPH Conference Proceedings*, 71-78.
- Kumar, S., Snyder, D., Duncan, D., Cohen, J. and Cooper, J. 2003. Digital preservation of ancient cuneiform tablets using 3D scanning. In Proc. of IEEE 4th Int. Conf. on 3-D Digital Imaging and Modeling, 326-333.
- Lavelle, J.P., Schuet, S.R. and Schuet, D.J. 2004. High speed 3D scanner with real-time 3D processing. *IEEE Int. Workshop on Imaging Systems and Techniques*.
- Levoy, M., Pulli, K., Curless, B., Rusinkiewicz, S., Koller, D., Pereira, L., Ginzton, M., Anderson, S., Davis, J., Ginsberg, J., Shade, J. and Fulk, D. 2000. The digital Michelangelo project: 3D scanning of large statues. *Proc. of the 27th Annual Conf. on Computer Graphics and Interactive Techniques*, 131-144.
- Meng, X.Q. and Hu, Z.Y. 2003. A new easy camera calibration technique based on circular points. *Pattern Recognition*, 36(5), 1155-1164.
- Presnyakov, Y.P. and Shchepinov, V.P. 1997. Use of the speckly effect to analyze vibrations of a rough surface. *Technical Physics*, 42(8), 919-922.
- Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. 2002. *Numerical Recipes in C++: The Art of Scientific Computing*. Cambridge University Press. Cambridge.
- Rocchini, C., Cignoni, P., Montani, C., Pingi, P. and Scopigno, R. 1999. A low cost optical 3D scanner based on structured light. In Proc. of Annual Conf. of European Assoc. For Computer Graphics (EUROGRAPHICS 2001), 299-308.
- Scopigno, R., Pingi, P., Rocchini, C., Cignoni, P. and Montani, C. 2000. 3D scanning and rendering cultural heritage artifacts on a low budget. *European Workshop on High Performance Graphics Systems and Applications*, Cineca, Bologna, Italy.
- Teutsch, C. 2007. Model-based analysis and evaluation of point sets from optical 3D laser scanners. *Magdeburger Schriften zur Visualisierung*, 1. PhD Thesis. Shaker Verlag.
- Triggs, B. and McLauchlan, P. 2000. Bundle adjustment – a modern synthesis. *Vision Algorithms: Theory and Practice*, 298-375. Springer Verlag.
- Wu, Y. and Hu, Z. 2006. Camera calibration and direct reconstruction from plane with brackets. *Journal of Mathematical Imaging and Vision*, 24(3), 279-293.

- Yıldırım, İ., Demir, B. ve Ertürk, S. 2006. E-Devlet için Kültür Varlıklarının 3-Boyutlu Modellenmesi ve Gösterimi. 14. IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SIU'2006).
- Zhang, Z. 1999. Flexible camera calibration by viewing a plane from unknown orientations. In Proc. Int. Conf. On Computer Vision (ICCV'99), 666-673.
- Zhang, Z. 2000. A flexible new technique for camera calibration. IEEE trans. Pattern Analysis and Machine Intelligence, 22(11), 1330-1334.
- Zhang, Z. and Huang, P. 2004. High-Resolution, Real-time 3D Shape Acquisition. In Proc. of Conf. on Computer Vision and pattern Recognition Workshop (CVPRW'04), 3, 28.

ÖZGEÇMİŞ

Adı Soyadı: Kurtuluş KÜLLÜ

Doğum Yeri: Gediz

Doğum Tarihi: 31/05/1982

Medeni Hali: Bekar

Yabancı Dili: İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise : Aydın Fen Lisesi (2000)

Lisans : The University of Edinburgh, School of Informatics,
B.Sc.Hon. Computer Science with Honours (2006)

Yüksek Lisans : Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar
Mühendisliği Anabilim Dalı (Şubat 2007 – Nisan 2010)

Çalıştığı Kurum/Kurumlar ve Yıl

Ankara İl Milli Eğitim Müdürlüğü

Bilgisayar- Sınav Hizmetleri Bölümü, Aday Memur 2006-2007

Ankara Üniversitesi, Mühendislik Fakültesi

Bilgisayar Mühendisliği Bölümü, Öğretim Görevlisi 2007-(devam etmekte)