



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



3D LIDAR NOKTA BULUTU İŞLEMEDE
SINIR GÖZETİMLİ VOKSEL TABANLI BİR
SEGMENTASYON YÖNTEMİ
GELİŞTİRİLMESİ

Ali SAĞLAM

DOKTORA TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Kasım-2020
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Ali SAĞLAM tarafından hazırlanan “3D Lidar Nokta Bulutu İşlemede Sınır Gözetimli Voksel Tabanlı Bir Segmentasyon Yöntemi Geliştirilmesi” adlı tez çalışması 12/11/2020 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda DOKTORA TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Prof. Dr. Erkan ÜLKER

.....

Danışman

Dr. Öğr. Üyesi Nurdan BAYKAN

.....

Üye

Prof. Dr. İsmail Rakıp KARAŞ

.....

Üye

Doç. Dr. Murat CEYLAN

.....

Üye

Dr. Öğr. Üyesi Gür Emre GÜRAKSIN

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

Bu tez çalışması TÜBİTAK tarafından 119E012 nolu proje ve ÖYP Kurum Koordinatörlüğü tarafından 2016-ÖYP-061 nolu proje ile desteklenmiştir.

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.



Ali SAĞLAM

Tarih: 24.11.2020

ÖZET

DOKTORA TEZİ

3D LIDAR NOKTA BULUTU İŞLEMEDE SINIR GÖZETİMLİ VOKSEL TABANLI BİR SEGMENTASYON YÖNTEMİ GELİŞTİRİLMESİ

Ali SAĞLAM

Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Nurdan BAYKAN

2020, 122 Sayfa

Jüri

Prof. Dr. İsmail Rakıp KARAŞ
Prof. Dr. Erkan ÜLKER
Doç. Dr. Murat CEYLAN
Dr. Öğr. Üyesi Gür Emre GÜRAKSIN
Dr. Öğr. Üyesi Nurdan BAYKAN

İç ve dış mekânlarda bulunan yapı ve nesneler Lidar (ışık algılayan ve mesafe ölçen) sistemler ile taranarak nokta bulutu halinde, üç boyutlu (3D) ve renkli olarak dijital ortamlara aktarılabilir. Lidar taramasıyla elde edilen, yapı ve nesneler hakkında detaylı bilgi sağlayan bu 3D nokta bulutu verisinin elemanları olan noktalar, organize edilmiş bir veri yapısı içerisinde konumlandırılmamış olarak düzensiz bir şekilde gelmektedir. Günümüzde Lidar teknolojisindeki gelişmeler, nokta bulutu verilerinin kalitesini artırmasının (daha az konum hatası ve daha yüksek çözünürlüklü olarak) yanında, çok yüksek miktarlarda düzensiz veri yığınına da getirmiştir. Çok yüksek boyuttaki bir veriden benzer özellikteki ve konumsal yakınlığı olan verileri gruplayarak, işlenecek veri sayısını düşürmekle birlikte veriden daha anlamlı bilgiler çıkarılmasını da sağlayan işleme segmentasyon denilmektedir. Segmentasyon, 3D nokta bulutu işlemeyi de kapsayan bilgisayarlı görme alanında büyük miktarda veri ile uğraşmayı gerektiren uygulamalar için oldukça yüksek bir öneme sahiptir. Segmentasyon işleminin de karmaşık veriler üzerinde istenilen özelliklerde ve sürede sonuç vermesi, bilgisayarlı görme alanında ayrı bir uğraş konusu olmuştur.

Tez çalışmasında, 3D nokta bulutu işlemede, uygulamanın başarısını önemli oranda etkileyen segmentasyon işleminin daha başarılı ve hızlı bir şekilde yapılabilmesi için yeni bir voksel tabanlı segmentasyon yöntemi geliştirilmiştir. Geliştirilen yöntem, yüzeylerdeki lokal nokta gruplarının oluşturdukları düzlemsel yapıların eğim açıları ve ağırlık merkezleri gibi basit geometrik özelliklerini kullanarak segmentasyon işlemini gerçekleştirebilmiştir.

Tez kapsamında, literatürde kullanılan veri setlerinin özellikleri dikkate alınarak, benzer şekilde bir adet iç ve iki adet dış mekânsal ortam, bir karasal Lidar sistemi ile taranarak üç farklı 3D nokta bulutu verisi temin edilmiştir. Elde edilen ham nokta verileri, oluşturulan veri setinin kullanım amacına göre indirgeme, kırpma ve gürültü giderme gibi ön işlemlerden geçirildikten sonra, segmentasyon referans segmentleri de hazırlanarak üç adet veri seti oluşturulmuştur. Tez kapsamında hazırlanan veri setlerine ek olarak, literatürden de iki adet daha segmentasyon veri seti temin edilmiş ve böylece toplam beş adet veri seti segmentasyon karşılaştırmasında kullanılmıştır.

Veri setlerinin temin edilmesinin ardından, yöntemlerin nicel değerler üzerinden karşılaştırması aşamasına kadar olan geliştirme ve iyileştirme aşamaları iki ayrı koldan eş zamanlı olarak ilerlemiştir. Bunlardan birisi sekiz dallı ağaç (octree) organizasyonu ile veriyi vokselleme tekniğinin, düzlem özelliği göstermeyen vokseller için yeniden düzlem uydurma ön işleminin ve geliştirilen segmentasyon yönteminin kodlanması aşamalarıdır. Diğer ise karşılaştırma için literatürde başarı göstermiş segmentasyon

yöntemlerinin belirlenmesi, bunların temin edilmesi veya yeniden kodlanması ve nicel karşılaştırma için doğruluk ve $F1$ skor değerleri hesaplama yöntemlerinin kodlanması aşamalarıdır. Bütün bu geliştirme, iyileştirme ve kodlama aşamaları tamamlandıktan sonra uygulanan yöntemlerin tez kapsamında kullanılan veri setleri üzerindeki segmentasyon çıktılarının doğruluk ve $F1$ skor sonuçları alınarak, başarı ve çalışma süresi açısından karşılaştırma analizleri yapılmıştır. Geliştirilen yöntem, 0.81 ortalama doğruluk değeri ve 0.69 ortalama $F1$ skor değeri ile literatürde bulunan ve benzer şekilde noktaların geometrik özelliklerini kullanarak segmentasyon yapan diğer yöntemlere göre segmentasyon başarısı ve hız açısından üstünlük elde etmiştir. Tez kapsamında ayrıca, nokta bulutundaki noktaların renk değerlerinin farklılıkları da belirli etki oranlarında segmentasyona dâhil edilmiş ve renk kalitesi yüksek iç mekân verisinde başarı arttırılmıştır.

Tez kapsamında daha sonra, geliştirilen yöntemin farklı parametre değerleri ile literatürden alınan yüksek miktarda noktadan oluşan bir iç mekân anlamsal segmentasyon (semantic segmentation) veri seti (S3DIS) üzerindeki ham nokta bulutu sınıflandırmasında ara işlem olarak kullanımı da incelenmiştir. Sınıflandırma işlemi için, öncelikle geliştirilen yöntemle segmentasyon yapılarak veri segmentlere ayrılmış ve her segmentten bir özellik vektörü çıkarılmıştır. Daha sonra da, bu özellik vektörleri kullanılarak sınıflandırma yapılmıştır. Segmentasyon tabanlı sınıflandırma işlemi, Destek Vektör Makinesi (DVM) ve Rastgele Orman (RO) olarak iki farklı sınıflandırıcı ile ayrı ayrı uygulanmıştır. Sınıflandırma işlemlerinin sonuçları da noktaların sınıf etiketlerinin doğruluk ve $F1$ skor değerleri üzerinden karşılaştırılmıştır. Karşılaştırma sonuçlarına göre, ham nokta bulutundaki noktaların sınıflandırma başarıları DVM için 0.76 doğruluk ve 0.48 $F1$ skor iken RO için 0.83 doğruluk ve 0.70 $F1$ skor olmuştur. Sonuçlara bakıldığında kullanılan veri ve özellik setlerine göre RO sınıflandırıcısı DVM sınıflandırıcısından daha iyi sonuç vermiştir.

Anahtar Kelimeler: Çok boyutlu ikili arama ağacı, düzlem uydurma, lidar, nokta bulutu işleme, nokta bulutu segmentasyonu, normal vektör, sekiz dallı ağaç, vokselleme.

ABSTRACT

PhD THESIS

DEVELOPING A BORDER CONSTRAINED VOXEL-BASED SEGMENTATION METHOD IN 3D LIDAR POINT CLOUD PROCESSING

Ali SAĞLAM

**Konya Technical University
Institute of Graduate Studies
Department of Computer Engineering**

Advisor: Asst. Prof. Dr. Nurdan BAYKAN

2020, 122 Pages

Jury

**Prof. Dr. İsmail Rakıp KARAŞ
Prof. Dr. Erkan ÜLKER
Assoc. Prof. Dr. Murat CEYLAN
Asst. Prof. Dr. Gür Emre GÜRAKSIN
Asst. Prof. Dr. Nurdan BAYKAN**

Structures and objects located indoors and outdoors can be scanned with Lidar (Light Detecting and Range) systems and transferred to digital environments as point clouds in three-dimensional (3D) and colored. The points, which are the elements of this 3D point cloud data obtained by Lidar scanning and providing detailed information about the structures and objects, come in an irregular form without being located in an organized data structure. Nowadays, developments in Lidar technology have not only improved the quality of point cloud data (with less location error and higher resolution), but also brought enormous amounts of irregular data. The segmentation process is the process that reduces the number of data by grouping data with similar characteristics and spatial proximity, and provides more meaningful information from the data. Segmentation is of great importance for applications that require dealing with a lot of data in the field of computer vision, including 3D point cloud processing. The ability of the segmentation process to yield results on complex data in the desired features and time has been a separate challenge in the field of computer vision.

In this thesis, a novel voxel-based segmentation method was developed by focusing on the segmentation of point clouds in order to make the segmentation process more successful and faster, which significantly affects the success of the application. The developed method was able to perform the segmentation process by using simple geometric features such as the inclination angles and barycenters of the planar structures formed by the local point groups on the surfaces.

Within the scope of the thesis, considering the characteristics of the data sets used in the literature, similarly, three different 3D point cloud data were obtained by scanning one indoor and two outdoor environments with a terrestrial Lidar system. After the raw point data obtained were pre-processed such as reduction, clipping and noise removal according to the intended use of the data set, the segmentation reference segments of them were also prepared and three data sets were created. In addition to the data sets prepared within the scope of the thesis, two segmentation data sets were obtained from the literature and thus, a total of five data sets were used in segmentation comparison.

After the data sets had been obtained, the development and improvement stages were proceeded simultaneously in two separate branches up to the stage of comparing the methods over quantitative values. One of them is the stages of coding the data voxelization technique with the eight-branch tree (octree) organization, the refitting pre-process for voxels that do not show planar feature, and the developed segmentation method. The other is the stages of determining the segmentation methods that have succeeded

in the literature for comparison, obtaining or recoding them, and coding the accuracy and $F1$ score values calculation methods for quantitative comparison. After all these development, improvement and coding stages were completed, the accuracy and $F1$ score results of the segmentation outputs of the applied methods on the data sets used within the scope of the thesis were obtained and comparison analyzes were made in terms of success and working time. The method developed has 0.81 average accuracy value and 0.69 average $F1$ score value in an advantage in terms of segmentation success and speed compared to other methods using geometric properties of points similarly in the literature. Within the scope of the thesis, the differences in the color values of the points in the point cloud were also included in the segmentation at certain effect rates, and success was increased in indoor data with high color quality.

In the scope of the thesis, the developed segmentation method was also examined as an intermediate process in raw point cloud classification with different segmentation parameter values on an indoor semantic segmentation dataset (S3DIS) consisting of a large amount of points obtained from the literature. For the classification process, the data was divided into segments by segmentation with the method developed first and feature vector was extracted from each segment. Then, classification is made using these feature vectors. The segmentation-based classification process was applied with two different classifiers as Support Vector Machine (SVM) and Random Forest (RF), separately. The results of the classification processes were compared on the accuracy and $F1$ score values of the class labels of the points. According to the comparison results, the classification successes of the points in the raw point cloud were 0.76 accuracy and 0.48 $F1$ score for SVM, while 0.83 accuracy and 0.70 $F1$ score for RF. Looking at the results, the RF classifier gives better results than the SVM classifier according to the data and feature sets used.

Keywords: kd-tree, lidar, normal vector, octree, plane fitting, point cloud processing, point cloud segmentation, voxelization.

ÖNSÖZ

Yüksek Lisans ve Doktora öğrenimim boyunca danışmanlığımı yapan, bu süreçteki bütün akademik çalışmalarımın yürütülmesinde bilgisi, tecrübesi, öngörüsü ve pozitif enerjisi ile hep yanımda olan değerli danışman hocam sayın Dr. Öğr. Üyesi Nurdan BAYKAN'a;

Tez izleme komitemde bulunan ve çok önemli fikir ve yönlendirmeleri ile çalışmama katkı sağlayan Prof. Dr. Erkan ÜLKER ve Doç. Dr. Murat CEYLAN'a;

Çalışmalarımın gerçekleştirilmesinde sağlamış olduğu imkânlar için Konya Teknik Üniversitesi ÖYP Kurum Koordinatörlüğü'ne ve 1002 – Hızlı Destek Programı kapsamında maddi destek sağlayan Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK)'na;

Destek ve yardımları ile çalışmalarımındaki kritik konuların çözümünde önemli rol oynayan ve TÜBİTAK-1002 projesinde görev almış olan Dr. Öğr. Üyesi Ömer Kaan BAYKAN ve Arş. Gör. Hasan Bilgehan MAKİNECİ'ye;

Veri alımında kullanılan cihazların temininde ve veri analizlerinde yapmış oldukları katkılardan dolayı Konya Teknik Üniversitesi, Harita Mühendisliği Bölümü'ne ve bölümdeki Fotogrametri laboratuvar ekibine;

Bugünlere gelmemde büyük pay sahibi olan annem Ayşe SAĞLAM ve babam Mehmet SAĞLAM'a;

Her zaman ilgi, anlayış ve destekleriyle yanımda olan değerli eşim Duygu SAĞLAM'a ve hayatıma renk katan mutluluk kaynağım kızlarım Emine Beyza SAĞLAM ve Zeynep Berra SAĞLAM'a teşekkür ederim.

Ali SAĞLAM
KONYA-2020

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	vi
ÖNSÖZ	viii
İÇİNDEKİLER	ix
SİMGELER VE KISALTMALAR	xi
1. GİRİŞ	1
1.1. Tezin Amacı.....	4
1.2. Tezin Önemi	4
1.3. Tezin Organizasyonu	7
2. KAYNAK ARAŞTIRMASI	8
2.1. Kümeleme Tabanlı Segmentasyon Yöntemleri	8
2.2. Model Uydurma Tabanlı Segmentasyon Yöntemleri	10
2.3. Bölge Genişletme Tabanlı Segmentasyon Yöntemleri	13
3. MATERYAL VE YÖNTEM.....	18
3.1. Önışlem Metotları	18
3.1.1. Kd-tree (çok boyutlu arama ağacı)	18
3.1.2. Octree (sekiz dallı ağaç) organizasyonu ile vokselleme ve indeksleme	21
3.1.3. Temel Bileşen Analizi (TBA) ve normal vektör	27
3.2. Sınıflandırıcılar	30
3.2.1. Destek Vektör Makinesi (DVM)	30
3.2.2. Rastgele Orman (RO)	32
3.3. Nicel başarı ölçümleri	33
3.3.1. Doğruluk (accuracy)	34
3.3.2. $F1$ skor ($F1$ score).....	34
3.4. Veri Setleri	35
3.4.1. Tez kapsamında hazırlanan segmentasyon veri setleri	35
3.4.2. Literatürden alınan segmentasyon veri setleri	43
3.4.3. Literatürden alınan sınıflandırma veri seti (S3DIS)	44
4. GELİŞTİRİLEN YÖNTEM	46
4.1. Geliştirilen Segmentasyon Yöntemi	46
4.1.1. Vokselleme sonrası lokal özelliklerin belirlenmesi	47
4.1.2. Kenar ve köşe vokselleri için yeniden düzlem uydurma (refitting).....	49
4.1.3. Komşu vokseller arasındaki bağlantıların ağırlıklandırılması	53
4.1.4. Vokselleri birleştirerek bölge genişletme	58
4.2. Geliştirilen Yöntemin Sınıflandırmada Ara İşlem Olarak Kullanılması	66

5. DENEYSEL SONUÇLAR VE TARTIŞMA	70
5.1. 3D Nokta Bulutu Segmentasyon Sonuçları	72
5.1.1. Tanımlanan geometriksel ağırlık ölçümüne göre segmentasyon sonuçları ..	72
5.1.2. Geliştirilen yöntemin konumsal özellik kullanarak segmentasyon sonuçları ve literatür ile karşılaştırılması	75
5.1.3. Geliştirilen yöntemde konum ve renk özelliği kullanarak segmentasyon sonuçları.....	82
5.2. 3D Nokta Bulutu Sınıflandırma Sonuçları.....	87
6. SONUÇLAR VE ÖNERİLER.....	93
6.1. Sonuçlar	93
6.2. Öneriler	96
KAYNAKLAR	98
EKLER	105
ÖZGEÇMİŞ	120

SİMGELER VE KISALTMALAR

Simgeler

α	: Düzlem ile teğet vektörü arasındaki açı (β açısının 90° 'ye tümleyeni)
β	: Normal vektör ile teğet vektör arasındaki açı
c	: Refitting (yeniden düzlem uydurma) parametresi
C_{DVM}	: DVM düzenleyici parametresi
c_{renk_etki}	: Renk etkisi
c_g	: Gürültü giderme patametresi
$\vec{d}$	: Teğet vektörü
d_{renk}	: Renk farklılığı
$d_{geometrik}$	: Geometrik farklılık
d_k	: k . en yakın nokta
d_{k_ort}	: En yakın noktaların ortalama uzaklığı
e	: Öz vektör
E	: Bağlantı listesi
ε	: Düzlem standart sapması eşik değeri
k	: Komşu nokta sayısı
K	: Küme sayısı
λ	: Öz değer
n	: Toplam voksel sayısı
$\vec{n}$	: Normal vektör
$ \vec{n} $	: Normal vektör skaler büyüklüğü
π	: Pi sayısı
S	: Segment kümesi
$\bar{\sigma}$	: Ortalama standart sapma
σ_{max}	: Olabilecek maksimum standart sapma
θ	: İki normal vektörün fark açısı
τ_{max}	: Geliştirilen segmentasyon yönteminin açılal parametresi
v	: Voksel
V	: Voksel kümesi
w	: Bağlantı ağırlık değeri
W	: Ağırlık değeri listesi
X	: Ağırlık merkezi

Kısaltmalar

2D	: 2 Boyutlu
2.5D	: 2.5 Boyutlu (derinlik değerine sahip 2D)
3D	: 3 Boyutlu
ALS	: Airborne Laser Scanning
DVM	: Destek Vektör Makinesi
FH	: Felzenszwalb ve Huttenlocher
FPFH	: Fast Point Feature Histogram
GPS	: Global Positioning System
HT	: Hough Transform
INS	: Inertial Navigation System

K-NN	: K-Nearest Neighbors
LCCP	: Local Convex Connected Patches
Lidar	: Light Detection and Ranging
LNy	: Yerel Nokta Yoğunluğu
LPD	: Local Points Density
MLS	: Mobile Laser Scanning
NDT	: Normal Distribution Transformation
OF1	: Overall <i>F1</i> skor
ORG	: Octree-based Region Growing
PCA	: Principal Component Analysis
PCL	: Point Cloud Library
RANSAC	: Random Sample Consensus
RF	: Random Forest
RG	: Region Growing
RGB	: Red Green Blue
RO	: Rastgele Orman
SVM	: Support Vector Machine
S3DIS	: The Stanford Large-Scale 3D Indoor Spaces
SS	: Standart Sapma
SVGS	: Supervoxel and Graph-based Segmentation
TBA	: Temel Bileşen Analizi
TIN	: Triangular Irregular Network
TLS	: Terrestrial Laser Scanning
VCCS	: Voxel Cloud Connectivity Segmentation
VIS	: Voxel Incremental Segmentation

* Tez kapsamında sayılar ve sayı birimleri SI (The International System of Units – Fransızca Le Système International d’Unités) sistemine uygun olarak verilmiştir.

1. GİRİŞ

Geleneksel bilgisayarlı görme sistemlerinde, ışık sensörleriyle elde edilen görüntü verisinin en küçük elemanı olan piksellerin konumları, yatay ve dikey indeksli olarak düzenli 2D yapı (x ve y eksenleri) içerisinde elde edilmektedir. Lazer teknolojisinin gelişmesi ile piksellere derinlik (z eksenindeki konum uzaklığı) değeri de ilave edilerek yine x ve y eksenleri düzenli bir 2D yapı içerisinde temsil edilen 2.5D görüntüler elde edilebilmektedir. Lidar (ışık tespit eden ve mesafe ölçen - light detection and ranging) sistemler ışın (lazer veya ışık) gönderdikten sonra, gönderdiği ışını yansıtan nokta yüzeyin geri dönüş süresine göre mesafesini ölçmektedir. Günümüzde çok gelişmiş olan bu Lidar sistemler kısa sürede ve çok yoğun çözünürlükte art arda gönderdiği ışınlar ile bulunduğu ortamı 360° tarayarak ortamı temsil eden milyonlarca noktasal veri oluşturabilmektedir (Polat ve Uysal, 2016). Işınlar ile elde edilen mesafe bilgileri, nokta atışının yönü, yüksek doğruluklu inersiyal referans sistemi (INS - Inertial Navigation System) ve küresel konumlandırma uydu sistemi (GPS - Global Positioning System) kullanılarak noktasal verilere, 3D konumsal uzaydaki x, y ve z eksenlerinde konum değerleri atanmaktadır (Ekercin ve Üstün, 2004). INS ve GPS sayesinde farklı açılardan taranan nesnelerin noktasal verileri birleştirilerek dijital ortamda daha çok 3D bilgi içeren nokta bulutu verileri de oluşturulabilmektedir (Ural ve Shan, 2012; Su ve ark., 2016). Lidar sistemler ile elde edilen bu noktasal veriler, oluşturdukları 3D nokta bulutu içerisinde düzenli indekslenmiş (organize edilmiş) ve ayrık bir yapıda konumlanmamışlardır. Bu yüzden bu verilerin işlenmesi için verilerin organizasyonunu sağlayan ayrı yöntemler (veri yapıları) kullanılmaktadır.

Günümüzde nokta bulutları üç değişik yöntemle elde edilmektedir (Li ve ark., 2017). Hava araçları vasıtası ile yapılan havasal lazer tarama (Airborne Laser Scanning – ALS) yöntemi ile gökyüzünden, yeryüzü taranarak noktasal veri elde edilmektedir. Bu şekilde veri elde etmenin avantajı geniş bir alanın hızlı bir şekilde taranabilmesi iken, dezavantajı ise görüş açısının sınırlı olmasıdır (Aijazi ve ark., 2013; Polat ve Uysal, 2016). Yeryüzünde belirli noktalara sabitlenmiş olan araçlar vasıtası ile yapılan bölgesel lazer tarama (Terrestrial Laser Scanning – TLS) yöntemi de farklı açılardan alınan noktasal verilerin birleştirilerek 3D nokta bulutu elde edilmesini sağlamaktadır. Bu yöntemin avantajı bir alanın istenilen açılardan daha yüksek ve doğru örneklemeler ile taranabilmesi iken, dezavantajı ise tarayıcının alana yakın olduğu için dar bir görüş açısına sahip olması ve daha çok örnekleme yaptığı için çok yüksek yoğunluklarda

verinin elde edilmesiyle nokta bulutu işleminin zorlaşmasıdır (Li ve ark., 2016; Babahajiani ve ark., 2017; Mayr ve ark., 2017). ALS ve TLS yöntemi birlikte kullanılarak iki yöntemin de bazı avantajlarından faydalanılarak dış mekân kentsel tarama (outdoor urban scanning) yapılabilir (Golovinskiy ve Funkhouser, 2009). Bunların haricinde bir diğer tarama yöntemi ise taşınabilir olan mobil lazer tarama (Mobile Laser Scanning – MLS) yöntemidir. Bu yöntemin TLS yönteminden farkı, kolay taşınabilmesi ve dolayısıyla hareket halinde iken veri elde etme işleminin gerçekleştirilebilmesidir. Bu yöntem hem dış mekân (outdoor) hem de iç mekân (indoor) için kullanılabilir (Zhu ve ark., 2010; Jamali ve ark., 2016; Li ve ark., 2017).

Şehir, orman ve arazilerin; ve bunların içerisinde yer alan nesne ve yapıların yüzeylerinin Lidar sistemler ile taranmasıyla elde edilen 3D nokta bulutları, değişen yoğunluklarda ve düzensiz yapıda dağılmış nokta örüntüleridir (Wang ve Tseng, 2010; Vo ve ark., 2015; Vosselman ve ark., 2017; Xu ve ark., 2018c). Tarama işlemlerinde genellikle, noktaların lazer ışınları ile konumunu belirledikten sonra bu noktaları, sistemle bütünleştirilmiş kamera ile alınan 3D fotoğraf ile karşılaştırarak noktalara öznitelik olarak konum değerlerinin yanında renk değerleri de katabilen ışık algılama ve mesafe ölçme (Lidar – Light Detection And Ranging) sistemleri kullanılmaktadır. Bir Lidar sistem ile alınmış olan nokta bulutunda, tarayıcı ile arasındaki mesafeye göre farklı yoğunluklarda kümelenmiş olarak yer alan noktaların her birisi, konumsal (3D koordinatlar) ve radyometrik (spektral) (RGB renkler, parlaklık veya lazer yansıma yoğunluğu gibi) değerlere sahip olmaktadır (Wang ve Tseng, 2011; Barnea ve Filin, 2013).

Bir nokta grubu içerisindeki noktaların konumsal değerlerine ve bu değerlerin 3D uzaydaki kovaryans dağılımlarına bakılarak yakınlık, süreklilik, düzlemsellik, doğrusallık, küresellik ve saçılım gibi nokta grubunun oluşturduğu geometrik şekil hakkında bir çıkarım yapılmasını sağlayacak birçok bilgi elde edilebilmektedir (Rusu ve ark., 2009; Blomley ve ark., 2016; Poux ve Billen, 2019). Bu çıkarımlardan yola çıkılarak nesne tespiti (Pu ve ark., 2011; Serna ve Marcotegui, 2014), yol tespiti (Clode ve ark., 2004; Boyko ve Funkhouser, 2011), şehir haritalama (Guindon ve ark., 2004; Haala ve ark., 2008), yeşil alan haritalama (Wurm ve ark., 2014), yapıların analizi (Pu ve Vosselman, 2009), şehir modelleme (Lafarge ve Mallet, 2012), orman analizi (Morsdorf ve ark., 2004; Mayr ve ark., 2017), afet yönetimi (Laefer ve Pradhan, 2006), kültürel miras analizi (Lozes ve ark., 2015) ve karmaşık 3D ortamların yorumlanması (Boulch ve ark., 2013) gibi gerçek hayattaki problemlerin çözümü kolaylaşmaktadır.

Bir uzayda (2D ya da 3D) yer alan ve bu uzay içerisinde konumsal yakınlığı olan ve öznitelik olarak benzer özelliklere sahip elemanların homojen gruplara yani segmentlere (bölütlere) ayrılmasına “segmentasyon” (segmentation) denilmektedir (Pal ve Pal, 1993; Lezoray ve Charrier, 2009; Sağlam, 2016). 2D ve 3D görüntülerin işlenmesinde önemli bir yeri olan segmentasyon konusu her zaman uğraş gerektiren bir problem olmuştur. 3D nokta bulutu verileri ile çalışan bazı bilgisayarlı görme, robotik, uzaktan algılama ve fotogrametri uygulamalarındaki yoğun işlem yükü ve/veya daha anlamlı bilgi çıkarımları gerektiren karmaşık verilerin yorumlanmasında segmentasyon işlemi bir ön veya ara işlem olarak yaygın bir şekilde kullanılmaktadır (Weinmann ve ark., 2015a; Sağlam ve Baykan, 2017).

3D uzayda konumsal olarak bağlantılı olan ve benzer özelliklere sahip noktaların segmentler halinde gruplandığı segmentasyon aşamasının, daha sonraki aşamalar için birçok faydası bulunmaktadır (Xu ve ark., 2018c). Başlıca faydalarından en önde geleni, sonraki aşamalarındaki işlemlerde, segmentlerdeki verilerin birlikte değerlendirilmesi sayesinde veri boyutunun önemli miktarda düşürülerek işlem yükünün azaltılmasıdır. Bununla beraber segmentler kendi içerisindeki homojen yapıyı korumakta ve nesneleri birbirinden ayıran sınırlarını belirlemektedir. Ayrıca, bir segmentteki noktaların 3D uzayda oluşturduğu birliktelik, sonraki uygulama aşamaları için, düzlemsellik seviyesi, doğrusalılık seviyesi, düzlemin uzaydaki yönü, noktaların dağılım şekli, segment büyüklüğü ve nokta yoğunluğu gibi üst düzey geometrik özellikler sağlamaktadır (Lari ve Habib, 2014; Yang ve ark., 2015; Xu ve ark., 2018c).

Lokal nokta gruplarının oluşturduğu lokal yüzey eğimleri de bir çok segmentasyon yönteminde kullanılan başlıca özelliklerdendir (Papon ve ark., 2013; Stein ve ark., 2014; Verdoja ve ark., 2017; Xu ve ark., 2017; Sağlam ve ark., 2020). Lokal nokta gruplarının belirlenmesinde literatürde farklı teknikler kullanılmaktadır. Bunlardan en yaygın olanları en yakın komşu noktaların belirlenmesi (Rabbani ve ark., 2006), vokselleme (Stein ve ark., 2014) ve üçgenleme (triangulation) (Sacchi ve ark., 1999) teknikleridir.

Yapılan tez çalışmasında, 3D nokta bulutu işlemede uygulamanın başarısını önemli oranda etkileyen segmentasyon işleminin daha başarılı ve hızlı bir şekilde yapılabilmesi için yeni bir voksel tabanlı segmentasyon yöntemi geliştirilmiştir. Geliştirilen yöntem, önce octree yapısı kullanarak vokselleme yapmakta daha sonra nokta bulutundaki nesnelerin kenar ve köşelerinde kalıp, segmentasyon sonucunu etkileyen vokselleri tespit ederek bu voksellere düzlemsel özellik kazandırmaktadır. Böylelikle 3D

nokta bulutu verisi kullanan uygulamalarda, daha iyi segmentasyon sonucu elde edilmesi hedeflenmiştir. Geliştirilen yöntem içerisinde, vokseller içerisinde yer alan lokal nokta gruplarının oluşturduğu lokal yüzeylerin özellikleri dikkate alınmıştır. Bu lokal yüzeyler arasındaki geometrik ve/veya renk farklılıklarına göre yüzeyler segmentlenmekte ve bunun için de yöntem içerisinde geliştirilen geometrik farklılık ölçümü ve τ_{max} açısal eşik değeri parametresi kullanılmaktadır. Elde edilen sonuçlar hem nicel hem de görsel olarak, geliştirilen yöntemin 3D nokta bulutu segmentasyon işlemlerinde kullanılabileceğini ortaya koymuştur.

1.1. Tezin Amacı

Ortam analizi, nesne sınıflandırma, modelleme, haritalama, navigasyon ve sahne eşleştirme gibi üst seviye 3D nokta bulutu uygulamalarında gerekli olan ve uygulamanın başarısını önemli oranda etkileyen segmentasyon işleminin daha başarılı ve hızlı bir şekilde yapılabilmesi için tez kapsamında, literatürde mevcut olan segmentasyon yöntemlerinin analiz edilmesi, hem başarı hem de hız bakımından etkili yeni bir 3D nokta bulutu segmentasyonu yönteminin geliştirilmesi ve literatürdeki yöntemlerle nicel değerler ve görselleştirilmiş sonuçlar üzerinden karşılaştırılması amaçlanmıştır. Geliştirilen segmentasyon yönteminin başarısının artırılması için de lokal yüzey eğimlerinin değişim miktarını ölçen uygun bir ağırlık ölçüm formülü ve düzlem özelliği göstermeyen kenar ve köşelerde kalmış nokta gruplarının doğru segmentlenebilmesi için bir iyileştirme yöntemi geliştirilmesi amaçlanmıştır. Ayrıca, geliştirilen segmentasyon yönteminin gürültü olarak tanımlanan aykırı verilere karşı kararlı olması da tezin amaçları arasında yer almaktadır. Bununla beraber geliştirilen yöntemin sınıflandırma işleminde bir ara işlem olarak kullanılarak, sınıflandırma işlemine olan etkisinin ölçülmesi de tezin amaçları arasında yer almaktadır.

1.2. Tezin Önemi

Günümüzde, Lidar teknolojisindeki gelişmeler sayesinde fiziki ortamlar ve bu ortamlarda yer alan nesne ve yapılar taranarak yüksek kalitede 3D nokta bulutu verileri elde edilebilmektedir. Bu veriler haritacılık, mimari, restorasyon, tarım, endüstri, ormancılık ve robotik gibi çeşitli alanlardaki bazı problemlerin çözümü için kullanılmaktadır. Bu problemlerin çözümü için de büyük miktarlarda ve düzensiz halde

gelen 3D ham nokta bulutu verilerinin analiz edilmesi ve anlamlı hale dönüştürülmesi için etkili veri işleme tekniklerine ve bilgisayarlı görme yöntemlerine ihtiyaç duyulmaktadır. İhtiyaç duyulan yöntemlerden birisi de sınıflandırma, modelleme ve nesne tespiti gibi birçok üst düzey işlemde ara işlem olarak kullanılan segmentasyon işlemidir. Segmentasyon işlemi aşamasında, benzer özelliklere sahip çok sayıdaki veri yığını, tek bir grup halinde temsil edilerek veri sayısı önemli miktarda indirgenmektedir. Diğer taraftan, grupların içerisindeki nokta verilerinin dağılım şekline göre oluşan verinin dağılım yönleri, ağırlık merkezi, hacimsel büyüklüğü, yoğunluğu ve yüzey yönü gibi, segmentasyon sonrası aşamalar için daha fazla anlam ifade eden, üst düzey özellikler de çıkarılabilmektedir. Konumsal olarak bağlantılı noktaları, benzerliklerini dikkate alarak gruplayan segmentasyon işlemi de öncesinde gürültü giderme, örnekleme, komşuluk belirleme, lokal yüzey özellikleri çıkarma gibi bazı veri ön işleme tekniklerine ihtiyaç duyabilmektedir. Bütün bu sayılan ihtiyaçlar doğrultusunda tez kapsamında, sonraki çalışmalar için de faydalı olabilecek 3D nokta bulutu verisi ön işleme teknikleri açıklanmış, literatürdeki segmentasyon yöntemleri analiz edilmiş, yeni bir başarılı ve hızlı segmentasyon yöntemi geliştirilmiştir.

Tez kapsamında geliştirilen segmentasyon yöntemi, vokselleme tabanlı bir yöntemdir ve içerisinde süper vokselleme oluşturma aşamasını içermemektedir. İlk aşama olarak octree organizasyonu kullanılarak düzensiz yapıdaki nokta bulutu voksellenebilmekte ve aynı aşama içerisinde bir taraftan da vokselleme içerisindeki noktaların ağırlık merkezleri, yüzey normaleri ve eğer kullanılacaksa ortalama renk değerleri hesaplanmaktadır. Voksellene aşamasından sonra, nesnelerin kenar ve köşelerinde kalmış ve bundan dolayı da düzlemsel özellik göstermeyen vokselleri tespit etmek için düzlem standart sapmaları için bir eşik değeri bulma formülü geliştirilmiştir. Daha sonrasında da tespit edilen bu voksellere daha uygun bir düzlem uydurmak için tez kapsamında geliştirilen K-ortalama kümeleme yöntemi ile yeniden düzlem uydurma tekniği uygulanmıştır.

Bitişik vokseller arasındaki tüm bağlantılar belirlenirken, bu bağlantılara bir ağırlık değeri atanması için tez kapsamında yeni bir yöntem önerilmiştir. Bu yöntemde, lokal düzlemlerin eğimsel farklılıkları kullanılmış ve bu farklılığın hesaplanması için normal vektör ve ağırlık merkezi gibi basit geometriksel özellikleri kullanan yeni bir formül önerilmiştir. Bu yöntemde göre, ağırlık değeri atanan bütün bağlantılar bir bağlantı listesinde toplanmakta ve daha sonra bu listedeki bağlantılar ağırlık değerlerine göre artan şekilde sıralanmaktadır. Ayrıca, ağırlık değeri atanmasında eğer renk farklılıkları da kullanılacaksa, bağlantının uçlarındaki iki vokselin içerisindeki noktaların

ortalama renk deęerleri arasındaki Öklit renk uzaklıęı da geometrik farklılıkla aynı aralıęa ölçeklendikten sonra, renk etkisi için belirlenen etki oranı derecesinde aęırlık hesaplamasına dâhil edilebilmektedir. Bölge genişletme aşamasının başlangıcında her bir voksele büyüme noktası özelliğine sahip bir segment gibi davranılmaktadır. En az aęırlıęa sahip olan bağlantıdan başlayarak sırayla bağlantıların her iki ucundaki segmentler birleştirme işlemi için deęerlendirilmeye alınmakta ve birleştirilen segmentler de böylece genişlemektedirler.

Tez kapsamında geliştirilen segmentasyon yönteminin literatürdeki yöntemlerden temel farkı, iki segment arasındaki sınırı oluşturan voksellerin karşılıklı olarak birebir eşleştirilmesidir. Segmentlerin birleştirilebilmesi için aranan kriter ise eşleştirilmiş vokseller arasındaki tüm bağlantıların aęırlık deęerlerinin, segmentasyon için verilen açısal parametre olan τ_{max} deęerinden küçük veya eşit olmasıdır. Bu aşama, yüzey özellięi göstermeyen yani tez kapsamında “kalıntı” olarak isimlendirilen voksellere hassasiyet göstermektedir ve segmentasyon işlemini yanlış yönlendirecek şekilde kolayca etkilenebilmektedir. Bu nedenle, geliştirilen yöntem içerisinde, kalıntı vokseller bağlantı aęırlıklandırılması ve bölge genişletme aşamalarında ayrı olarak deęerlendirmektedir. Geliştirilen yöntemin dięer bölge genişletme yöntemlerine kıyasla avantajı, segmentasyon derecesini (az veya aşırı segmentasyon) belirlemek için vokselle büyüklüęü haricinde, birleştirmenin olup olmayacağına karar verilmesini saęlayan eşik deęeri olarak yalnızca bir parametre deęiřiklięine ihtiyaç duymasısıdır.

Geliştirilen yöntem ve tekniklerin yanı sıra, tezin literatüre saęladığı bir dięer katkı da test edilen yöntemlerin segmentasyon başarılarının (doęruluk ve $F1$ skor) ve işlem sürelerinin kıyaslanmasıdır. Yöntemleri deęerlendirebilmek için tez kapsamında bir iç mekân ve iki dış mekân olmak üzere üç adet segmentasyon veri seti hazırlanmış ve ilaveten iki adet dış mekân veri seti de literatürden temin edilmiştir. Yöntemlerin nicel ve görsel segmentasyon sonuçları veri setleri üzerinde karşılaştırılmış ve tüm sonuçlar sonraki yapılacak çalışmalara yardımcı olmak amacıyla şekil, çizelge ve grafikler ile gösterilmiştir.

Tez kapsamında segmentasyon yöntemlerinin incelenmesi ve segmentasyon işlemi üzerine yapılan geliřtirmeler haricinde, geliştirilen segmentasyon yönteminin ham 3D nokta bulutu sınıflandırması gibi üst düzey bir uygulama içerisinde farklı parametre deęerleri ile kullanımını test etmek için de bir uygulama yapılmıştır. Sınıflandırmada önışlem olarak yapılan segmentasyon uygulamasında, literatürden temin edilen, karmařık bir yapıda çok miktarda noktadan oluşan bir iç mekân anlamsal (semantik) segmentasyon

veri seti kullanılmıştır. Ayrıca uygulama için iki ayrı sınıflandırma yöntemi olarak Destek Vektör Makinesi (DVM) (SVM - Support Vector Machine) ve Rastgele Orman (RO) (RF - Random Forest) kullanılmıştır. Yapılan uygulamanın sonuçları da yine hem görsel olarak hem de nicel başarı değerleri (doğruluk ve $F1$ skor) ile sunulmuştur. Elde edilen sonuçlar, 3D nokta bulutu sınıflandırma işlemi üzerine çalışan veya bu işlemleri kullanan araştırmacılar için bir yol gösterici olabilme niteliği göstermektedir.

1.3. Tezin Organizasyonu

Tez çalışması altı bölümden oluşmaktadır.

İlk bölümde 3D nokta bulutu hakkında giriş niteliğinde bilgiler verilmiş ve kullanım alanlarından bahsedilmiştir. Bunun devamında da segmentasyon işleminin tanımı ve 3D nokta bulutu uygulamalarındaki öneminden bahsedilmiştir. Tez çalışmasında geliştirilen segmentasyon yönteminde kullanılan vokselleme işleminin ve voksel yapılarının avantajlarından ve dezavantajlarından da devamında bahsedilmiştir. Tezin amacı ve önemine ilişkin yer alan konular da bu bölüm içerisinde yer almaktadır.

İkinci bölümde literatürde yer alan segmentasyon yöntemleri kategoriler altında incelenmiş ve tezin literatüre olan katkısı hakkında genel bilgiler verilmiştir.

Kullanılan yöntemler ve veri setleri üçüncü bölümde açıklanmıştır. Veri setlerinden tez kapsamında elde edilenler de alt bölümlerde detaylı bir şekilde üçüncü bölümde açıklanmıştır.

Geliştirilen segmentasyon yönteminin aşamaları ve segmentasyon işleminin üst düzey bir işlem olan ham nokta bulutu sınıflandırma işleminde bir ara işlem olarak kullanılması üzerine bir uygulamanın aşamaları teorik olarak dördüncü bölümde yer almaktadır.

Geliştirilen segmentasyon yönteminin, kullanılan segmentasyon veri setleri üzerindeki nicel ve görsel test sonuçları ve literatürdeki yöntemlerle karşılaştırılması beşinci bölümde sunulmuştur. Ayrıca, geliştirilen segmentasyon yönteminin sınıflandırma veri seti üzerinde farklı parametre değerleri ile ara işlem olarak kullanıldıktan sonra, bazı sınıflandırıcılar ile elde edilen sınıflandırma başarı sonuçları da bu bölümde yer almaktadır.

Son bölümde tez kapsamında yapılan çalışmalar ve elde edilen sonuçlar genel olarak değerlendirilmekte ve yapılabilecek sonraki çalışmalar için bazı öneriler sunulmaktadır.

2. KAYNAK ARAŞTIRMASI

Birçok gerçek hayat probleminin 3D nokta bulutu uygulamaları ile çözümünde bir ara işlem olarak yer alan segmentasyon aşaması, ham nokta verilerinin gruplanarak indirgenmesinde, yüzey sınırlarının belirlenmesinde ve sonraki aşamalar için daha anlamlı bilgiler sağlanmasında çok önemli bir rol oynamaktadır (Rabbani ve ark., 2007; Schumann ve ark., 2007; Rusu ve ark., 2008; Rau ve ark., 2015). Literatürde yapılan son yıllardaki bazı çalışmalarda, 3D nokta bulutu segmentasyonu probleminin çözümü için geliştirilen yöntemler genel olarak üç kategori altında incelenmiştir (Vo ve ark., 2015; Xu ve ark., 2018c). Tez içerisindeki kaynak araştırması bölümünde, alt bölümler olarak yer alan bu kategoriler; kümeleme tabanlı segmentasyon yöntemleri, model uydurma tabanlı segmentasyon yöntemleri ve bölge genişletme tabanlı segmentasyon yöntemleri olarak isimlendirilmektedirler.

2.1. Kümeleme Tabanlı Segmentasyon Yöntemleri

Veri madenciliğinde, bir veri yığınındaki elemanların denetimsiz (eğitimsiz) olarak sınıflandırılması işlemi olarak tanımlanan “kümeleme” kavramı, veri yığını içerisindeki elemanlardan homojen gruplar oluşturmak olarak da tanımlanabilmektedir (Han ve ark., 2012). Küme sayısı, denetimli sınıflandırmadaki önceden belirli olan sınıf sayısının aksine, kümeleme algoritmasının çalışması süresince veriye ve istenilen homojenlik seviyesine (kümeleme parametresine) göre değişebilmektedir (Xu ve Wunsch, 2005). Kümeleme algoritmaları genelde yinelemeli (iterasyonlu) olarak çalışmaktadır. Her yinelemede (iterasyonda) küme merkezleri optimize edilerek kümeler yeniden tanımlanır ve böylece her iterasyonda istenilen sonuca daha çok yaklaşılmaktadır. Veri madenciliği alanında uygulanan çoğu kümeleme işlemi olduğu gibi, 3D uzayda bulunan nokta verilerinin kümelenmesinde de noktalar, küme içerisindeki elemanların konumsal yakınlığına bakılmaksızın, elemanların diğer özniteliklerine göre kümeleme sürecinde yinelenen her işlem aşamasında yeniden tanımlanan kümeler atanırlar. Böylelikle, istenilen küme sayısı kadar ve/veya istenilen homojenliği sağlayan kümeler oluşturuluncaya kadar işlem devam etmektedir. İstenilen seviyeye gelememe ihtimaline karşılık maksimum iterasyon sayısı ile de işlem süresi sınırlandırılmaktadır. Ancak, özelliklerin ve parametrelerin optimizasyonu için işlemin

defalarca tekrarlanması durumundan kaynaklanan fazla işlem süresi gereksinimi, kümeleme yaklaşımının eksikliği olarak görülmektedir.

2002 yılında Filin, 3D uzaydaki noktaları, lokal yüzeylerin özelliklerini kullanarak, noktaları kullanıcı tarafından tanımlanmış olan kategorilere tek iterasyonda atayan bir lokal yüzey kümeleme yöntemi geliştirmiştir (Filin, 2002). Bu kategoriler, lokal yüzeylerin zeminden yüksekliğine ve bu lokal yüzeyi oluşturan noktaların yüzey için uydurulan düzlem üzerindeki standart sapmasına göre tanımlanmıştır. 2003 yılında Morsdorf ve arkadaşları, bir milli park alanından elde edilen 3D nokta bulutu verisindeki ağaçları segmentlemişlerdir. Boyutları 3 metreye 3 metre uzunluğunda olan farklı geometrik özelliklere sahip 6 adet kategori tanımlamışlardır. Segmentlenmiş olan nokta bulutu içerisinde ağaçları tanımlayabilmek için, önce bu 6 kategoriye göre K-Ortalama (K-Means) kümeleme algoritması kullanarak segmentler oluşturmuşlardır. Yapılan çalışmada, havadan tarama yapan lazer tarayıcı ile elde edilen 3D nokta bulutunun tarama doğruluğunu bu kümeler sayesinde ölçmüşlerdir (Morsdorf ve ark., 2003).

2008 yılında Biosca ve Lerma, lokal yüzey kümelemesi için Olasılık C-Ortalamları (Possibilistic C-Means) ve Bulanık C-Ortalamları (Fuzzy C-Means) algoritmalarını birlikte kullanmışlardır. Küme merkezlerini ise bir noktanın komşuları ile arasındaki uzaklıkları, lokal yüzey normalleri ve lokal yüzeylerin bir orijin (referans) noktasına olan uzaklıklarına göre optimize etmişlerdir (Biosca ve Lerma, 2008). Yöntem, kümelerin başlangıç özelliklerinin tanımlanmasına ihtiyaç duymamaktadır. Kümelerin üyelik katsayıları iterasyonlu olarak optimize edilmektedir.

2009 yılında Yao ve arkadaşları, ham nokta bulutundan başlangıç segmentleri elde etmek için Ortalama Kaydırma (Mean Shift) algoritmasını kullanarak noktaların konumsal değerlerine göre kümeleme yapmışlardır (Yao ve ark., 2009). Daha sonra bu segmentleri Normalleştirilmiş Graf Kesme (Normalized Graph-Cut) yöntemi ile nesne çıkarma işleminde oluşturulması gereken graf yapısı için kullanmışlardır. 2014 yılında Lari ve Habib, bir 3D nokta bulutu içerisinde yer alan nesneleri düzlemsel ve doğrusal/silindirik yapılar için tanımlanan özelliklere göre kümeleyerek, segmentler oluşturmak için iki adet kümeleme yaklaşımı ortaya koymuşlardır (Lari ve Habib, 2014). Bu yaklaşımlarda, önce her noktanın belirli bir lokal alan içerisindeki en yakın komşularıyla birlikte oluşturduğu yüzeysel yapının düzlemsel ve doğrusal/silindirik derecesini temsil eden değerleri belirlenmektedir. Daha sonra da bu lokal yüzeyler segmentasyon parametrelerinin yeniden yapılandırılmasıyla bu özelliklere göre kümelenebilirler.

2019 yılında Poux ve Billen sekiz dallı ağaç (octree) organizasyonu ile noktaları vokseller içerisinde lokal gruplara ayırmış ve oluşturulan bu gruplardan iki adet özellik seti elde etmişlerdir. Bunlardan ilki düşük seviye şekil tabanlı geometrik özellik seti ve diğeri ise bağlantı ve ilişkileri tanımlayan özellik setidir. Bu özelliklerin benzerliğine göre vokselleri bağlı gruplar halinde kümeledikten sonra, bu kümeleri semantik etiketleme için kullanmışlardır (Poux ve Billen, 2019).

2.2. Model Uydurma Tabanlı Segmentasyon Yöntemleri

Model uydurma tabanlı segmentasyon yöntemlerine göre, bir nokta grubunu tanımlanmış bazı özel geometrik modellere uydurma (fitting) teknikleri kullanılarak segmentasyon işlemi yapılmaktadır. Literatürde, bu tekniklerden en yaygın olarak kullanılanları Hough Dönüşümü (Hough Transform - HT) (Borrmann ve ark., 2011) ve Rastgele Örneklem Konsensüsü (Random Sample Consensus - RANSAC) (Fischler ve Bolles, 1981) teknikleridir.

HT tekniği ilk zamanlarda, görüntü işleme alanında bir görüntüden sadece çizgileri ve daireleri tespit etmek için kullanılmaktaydı. Günümüzde ise HT, son yıllardaki genişletilmiş hali ile düzlemler, silindirler ve küreler gibi 3D şekilleri tespit edebilmek için kullanılmaktadır (Hulik ve ark., 2014). Bu teknik, konumsal uzaydaki şekilsel parametreleri yüksek hesaplama maliyetiyle optimize etmeye çalışmaktadır. RANSAC ise iterasyonlu olarak ve rastgele örnek noktalar seçerek bu noktaları, az sayıda aykırı değere sahip olacak şekilde, farklı şekil türlerine uydurmaya çalışmaktadır. Bu algoritmada, ilk önce rastgele ve az olacak şekilde örnek noktalar seçilmekte ve seçilen noktalar farklı şekiller için test edilerek en fazla uyan aday şekiller belirlenmektedir. Daha sonra da bu aday şekiller, her iterasyonda değişmekte olan şekil parametreleriyle tüm noktalar için test edilmektedir (Schnabel ve ark., 2007). HT ve RANSAC teknikleri her ne kadar tanımlanan şekil özellikleri için başarılı ve aykırı değerlere karşı dayanıklı olsa da, tanımlanması zor şekillerden oluşan bir verinin segmentasyonunda başarısız olabilmekte ve çok fazla işlem süresi gerektirmektedirler.

Marshall ve arkadaşları, yüzeylerdeki noktaların dağılımlarını kullanarak, nesneleri, en küçük kareler yöntemi ile hesaplanan silindir, küre, simit ve koni şekillerinden en yakın olan modele uyduran bir segmentasyon yöntemi geliştirmişlerdir (Marshall ve ark., 2001). Aynı yıl Vosselman ve Dijkman, kentsel mekânların inşası için zemini ve düzlemsel olan bina yüzeylerini segmentlemek için 3D HT tekniğini

kullanmışlardır (Vosselman ve Dijkman, 2001). Rabbani ve Heuvel, bir nokta bulutundaki silindirik özellik gösteren nesnelerin etkili bir şekilde segmentlenebilmesi için HT tekniği üzerinde değişiklikler yapmışlardır (Rabbani ve Heuvel, 2005). 2017 yılında Overby ve arkadaşları, lazer tarayıcılar ile havadan elde ettikleri 3D nokta bulutu içerisindeki binaları modelleyebilmek için HT tekniğini kullanmışlardır (Overby ve ark., 2017). Örnek olarak alınan noktalara şekil uydurmak için önceki iterasyonlarda değerlendirilmiş olan noktaları kullanmamışlardır. Çalışmalarında ayrıca, kesişen düzlemleri de belirleyerek çatıların çokgen yapılarını modellemişlerdir.

RANSAC tekniğini kullanan çalışmalardan, Chen ve arkadaşları, çatıların düzlemsel özellik gösteren kısımlarını segmentlemek için orijinal RANSAC algoritmasında bazı değişikliklere gitmişlerdir. Algoritmanın değiştirilmiş halinde, aykırı değerler çalışmayı etkilememekte ve konumsal yakınlık korunarak segmentasyon derecesi dengelenebilmektedir (Chen ve ark., 2014). Schnabel ve arkadaşları, RANSAC tekniğini yüzeylerin bazı önemsiz kısımlarından taviz vererek, kullanımı basit ve hızlı olacak şekilde değiştirmişlerdir. (Schnabel ve ark., 2007). Her noktanın, bulunduğu lokal yüzey normalini hızlı bir şekilde, yaklaşık olarak hesaplayabilen bir normal hesaplama tekniği kullanmışlar ve nokta bulutlarını octree yapısında oktanların içerisindeki nokta sayılarını kullanarak ölçeklendirmişlerdir.

Düzlem segmentasyonu, birçok 3D lidar nokta bulutu işlemi içerisinde önemli ve temel bir yer tutmaktadır. Bu işlem için en bilinen yöntemlerden birisi de RANSAC tekniğinin kullanılmasıdır (Li ve ark., 2017). Li ve arkadaşları, normal dağılım dönüşümü (Normal Distribution Transformation – NDT) hücrelerini kullanarak RANSAC algoritmasını genişletmişlerdir. Bu yöntemle göre dağınık halde olan nokta bulutu verileri NDT hücreleri ile temsil edilmektedir. Bu NDT hücrelerinin her birisi düzlemsel ve düzlemsel olmayan yüzeyler olarak sınıflandırılmaktadır. Her iterasyonda düzlemlerin doğru bir şekilde tespit edildiğini doğrulamak için aynı düzlem üzerinden örnek olarak az sayıda NDT hücresi seçilmektedir. Yöntemin başarısı üç adet iç mekân 3D nokta bulutu üzerinde test edilmiş ve test sonuçları geliştirilen metodun klasik RANSAC metoduna göre daha fazla doğruluk ve güvenilirlik başarısı sağladığını göstermiştir (Li ve ark., 2017).

HT ve RANSAC teknikleri haricinde, literatürde lokal yüzeyleri basit geometriksel özelliklerine göre, tanımlanmış olan genel şekilsel yapılar içerisinde gruplayan yöntemler de mevcuttur. Lari ve Habib 3D nokta bulutu üzerindeki noktaların lokal komşulukları üzerinden Temel Bileşen Analizlerini (TBA) (Principal Component

Analysis – PCA) yaparak; düzlemsel bir yapı üzerinde mi yoksa doğrusal/silindirik bir yapı üzerinde mi bulunduklarını tespit etmeye çalışmışlardır (Lari ve Habib, 2014). Bunu yaparken kümeleme de yaptıklarından, yapılan çalışma aynı zamanda kümeleme tabanlı segmentasyon yöntemleri kategorisine de girmektedir. Düzlemsel bir yapı üzerinde bulunduğu tespit edilen noktaların segmentleme için gereken özniteliklerini adaptif bir silindir komşuluğu tanımlaması ile hesaplarken, doğrusal/silindirik bir yapı üzerinde bulunduğu tespit edilen noktaların segmentleme özneliği olarak, noktaların yönlerini ve pozisyonlarını kullanmışlardır. Daha sonra bu öznitelikleri, geliştirdikleri sıralı bir segmentasyon yöntemi için kullanarak; noktaların üzerlerinde bulunduğu yapıları birbirlerinden ayırmışlardır. Çalışmalarında kullandıkları lokal yüzeylerin geometriksel özellikleri, 3D nokta bulutu işlemede birçok çalışmanın kapsamında da kullanılmıştır (Nurunnabi ve ark., 2012; Yang ve Dong, 2013; Lozes ve ark., 2015; Su ve ark., 2016; Xu ve ark., 2016; Li ve ark., 2017).

Geometrik modellere uydurma (fitting) teknikleri kullanılarak yapılan çalışmalardan hariç literatürde aynı zamanda, sınıflandırma yöntemlerini kullanarak çalışan model uydurma tabanlı segmentasyon yöntemleri de mevcuttur. Yang ve Dong, 2013 yılında MLS dış mekân kentsel nokta bulutları üzerinde uygulanmış olan şekil tabanlı bir segmentasyon yöntemi geliştirmişlerdir. Bu yöntemde ilk olarak, noktaların geometrik özelliklerini elde etmek için kullanılan komşuluk yöntemi için en uygun komşu sayısını hesaplamışlar ve noktaları geometrik özelliklerine göre Destek Vektör Makinesi (DVM) (Support Vector Machine – SVM) ile sınıflandırmışlardır. İkinci olarak, sınıflandırılan nokta bulutlarını segmentlemek için bir kurallar dizisi tanımlamışlardır. Aşırı segmentasyon problemini çözmek için de segmentler için normalize edilmiş kesme (the normalized cut) yöntemi üzerinde bir benzerlik ölçütü önermişlerdir. Son olarak, segmentasyon işleminden elde edilen segmentler topolojik olarak bağlantılarına göre doğrusal, düzlemsel ve küresel gibi anlamlı olan geometrik yapılar içerisinde birleştirilmiştir. Önerilen yöntem büyük boyutta iki farklı MLS dış mekân kentsel nokta bulutu üzerinde test edilmiştir. Sınıflandırma için nokta bulutu içerisinde direk benzeri nesneleri çıkarmışlar ve 105 milyon noktadan oluşan bir veri seti üzerinde %95 başarı sonucu elde etmişlerdir. Bütün işlemler için toplam 59 dakika civarı işlem zamanı harcanmıştır ve bunun %60'ını da noktaların geometrik özelliklerini tanımlamak almıştır (Yang ve Dong, 2013).

Wu ve arkadaşları süper voksel tabanlı bir segmentasyon yöntemi kullanarak sokak ışıklarının direklerini ayırmışlardır (Wu ve ark., 2017). Bir MLS sistemden alınan

verileri kullanarak uygulanan yöntem beş aşamadan oluşmuştur. Bunlar ön işleme, lokalleştirme, segmentasyon, özellik çıkartma ve sınıflandırmadır. İlk önce, ham halde bulunan nokta bulutu verisini yörüngeler boyunca segmentlere ayırmışlar ve zemin üzerinde bulunan noktaları kaldırmışlardır. Geri kalan kısımları ise süper vokseller içerisine segmentlemişlerdir. Daha sonra da direk nesnesine benzer olan nesneleri daha doğru tespit edebilmek için güçlü bir lokalleştirme tabanlı segmentasyon yöntemi önermişlerdir. Son olarak bu segmentler direk özelliklerine göre DVM ve RO sınıflandırma metotları kullanılarak sınıflara ayrılmıştır. Çalışmada nokta bulutu üzerinden sokak ışığı direklerini çıkarmada önemli bir başarı sağlamışlardır.

2.3. Bölge Genişletme Tabanlı Segmentasyon Yöntemleri

Bölge genişletme tabanlı yöntemler, “başlangıç büyüme noktaları” (seed) diye adlandırılan ve çözünürlüğü kullanıcı tarafından belirlenmiş olan büyüme noktalarından yayılmak üzere, oluşturulan segmentleri aşamalı olarak genişleterek segmentasyon yapmaktadırlar. Bu başlangıç büyüme bölgesi, 3D nokta bulutundaki bir nokta (Rabbani ve ark., 2006; Nurunnabi ve ark., 2012), bir voksel (Vo ve ark., 2015; Su ve ark., 2016), bir süper voksel (Papon ve ark., 2013; Xu ve ark., 2018c) veya bir alt segment (Vieira ve Shimada, 2005; Wang ve Tseng, 2011) olabilmektedir. Bu kategorideki yöntemler, genişletilmekte olan segmentlerin, yüzeylerin sınırlarını ihlal etmemesini dikkate almakta ve bu sebeple önceden tanımlanabilen geometrik şekil özellikleri ile sınırlandırılmamaktadırlar. Genellikle noktaların, voksellerin veya süper voksellerin komşuluk bağlantılarını ve bu bağlantılar üzerine kurulan çizge (graf) yapısını kullanmaktalar ve genelde açgözlü (greedy) yani olabilecek her durum için bir sonraki adımı belirli olan çalışma prensibiyle çalışmaktadırlar. Bu nedenle, özellikle vokseller üzerinde çalışılıyorsa, bu kategorideki yöntemler diğer kategorilerdeki yöntemlere göre oldukça hızlı bir şekilde segmentasyon işlemini tamamlayabilmektedirler (Wang ve Tseng, 2011; Vo ve ark., 2015). Bölge genişleterek segmentleme yaklaşımı, son yıllarda 3D nesnelerin yüzeylerini segmentleme alanında en çok kullanılan yaklaşım olmuştur. Bu yaklaşımın yaygın şekilde kullanılmasından ve bu nedenle daha fazla kaynak bulunmasından dolayı hem tez kapsamında geliştirilen yöntem hem de tez kapsamında karşılaştırılma yapılan diğer yöntemler bölge genişletme yaklaşımını kullanmaktadırlar.

2002 yılında Gorte, hava araçlarına bağlı lazer tarayıcılarla elde edilen 3D nokta bulutlarını segmentlemek için basit bir bölge genişletme yöntemi geliştirmiştir (Gorte,

2002). Bu yöntemde, çatıların yüzeyleri ve binalar arasındaki boşluklarda kalan zemin kullanılarak, önce düzlemsel yüzeyleri segmentleyen ilk segmentasyon yapılmaktadır. İlk segmentasyon işleminden sonra, düzlemsel yapıdaki segmentler üzerinde bir üçgensel düzensiz ağ (Triangular Irregular Network - TIN) veri yapısı oluşturulmakta (triangulation) ve düzlemler bu üçgensel yapıdaki komşuluğa göre birleştirilerek, amaçlanan asıl segmentler elde edilmektedir. Benzer şekilde, Vieira ve Shimada karmaşık şekilleri içeren bir 3D nokta bulutunu ilk başta 3D geometriksel özelliklere göre farklı segmentlere ayırmışlardır (Vieira ve Shimada, 2005). Daha sonra da bu segmentleri, eğim ve gürültü derecelerinin tahmini hesaplamasını yaptıktan sonra, büyüme bölgeleri olarak kullanmışlar ve her biri için bir düzlem uydurduktan sonra belirledikleri kriterlere göre bu segmentleri birleştirmişlerdir.

Nokta seviyesinde yapılan segmentasyon yöntemleri arasında çok bilinen bir yöntem olarak, Rabbani ve arkadaşları, ilk önce nokta bulutunda büyüme noktaları belirlemişler ve bu noktaların genişlemesiyle ortaya çıkan alt segmentler oluşturmuşlardır. Aralarında konumsal bağlantısı olan bu alt segmentlerden fazla bir değişkenlik göstermeyenleri bulmak için değişme kısıtı kullanan ve “Region Growing” (RG) yöntemi olarak isimlendirilen bir nokta bulutu segmentasyon yöntemi geliştirmişlerdir (Rabbani ve ark., 2006). Her biri bir büyüme noktası olabilecek bu noktaların ayrı ayrı hepsi için lokal komşuları ile oluşturduğu yüzey normalleri hesaplanmaktadır. Bu büyüme noktaları, lokal yüzey normalleri dikkate alınarak iterasyonlu olarak birleştirilmekte ve böylece amaçlanan sonuç segmentler oluşturulmaktadır.

Graf araçlarının ve tekniklerinin kullanımı da bölge genişletme tabanlı segmentasyon yöntemlerinde oldukça yaygındır. 2009 yılında, Golovinskiy ve Funkhouser, bir graf kesme (graph-cuts) segmentasyonu yaklaşımı olan minimum kesme (min-cut) yöntemini (Wu ve Leahy, 1993; Shi ve Malik, 2000; Boykov ve Funka-Lea, 2006) kullanarak 3D nokta bulutu segmentleme yapmışlardır (Golovinskiy ve Funkhouser, 2009). Yaptıkları çalışmada 3D graf yapısını oluşturmak ve graf düğümleri arasındaki hat (bağlantı) ağırlıklarını atamak için k-En Yakın Komşular (k-Nearest Neighbors – k-NN) yöntemini kullanmışlardır. Çalışma için Kanada'nın Ottawa şehrindeki 6 km²'lik bir alandan bölgesel ve havasal lazer tarayıcılar ile alınan verilerin birleştirilmesiyle elde edilmiş 3D nokta bulutu veri setinden faydalanmışlardır. Nokta bulutu içerisindeki nesneleri arka plandan ayırmak için interaktif ve otomatik olmak üzere iki yöntem geliştirmişlerdir. İnteraktif yöntemi, doğrulama verisi (ground-truth) elde

etmek için kullanmışlardır. Bu şekilde yaklaşık 100 milyon noktadan oluşan ve içerisinde yaklaşık 1000 adet nesne içeren 300 bin m² civarı bir alanın doğrulama verisini elde etmişlerdir. İnteraktif yöntemle elde edilen bu doğrulama verileri, otomatik yöntemin başarısını test etmek için kullanılmıştır. Bu çalışmada, veri setinde noktaların radyometrik (renksel) değerleri olmadığı için sadece konumsal değerleri kullanılmıştır. Nesneleri oluşturan nokta kümelerinin şekilsel özellikleri de göz ardı edilmiştir.

Graf araçları da bazı temel kümeleme algoritmaları gibi bölge genişletme tabanlı segmentasyon uygulamalarında kullanılmıştır. 2010 yılında, Zhu ve arkadaşları k-NN ile 3D graf yapısı oluşturan bir 3D nokta bulutu kümeleme algoritması önermişlerdir (Zhu ve ark., 2010). İşlem zamanını düşürmek için zeminin düz olduğunu kabul ederek ön işlem olarak zemini nesnelerden ayırmışlardır. Daha sonra da, 2004 yılında Felzenszwalb ve Huttenlocher (FH) tarafından sunulan graf tabanlı ve hızlı bir segmentasyon algoritması olan FH algoritmasını (Felzenszwalb ve Huttenlocher, 2004) kullanarak nesneleri birbirinden ayırmışlardır. Çalışmalarında noktaların radyometrik özelliklerini kullanmayıp, sadece birbirlerine olan konum (uzamsal) uzaklıklarını kullanmışlardır. Aynı yıl, Schoenberg ve arkadaşları, renk özellikleri de olan 3D nokta verisi üzerinde FH algoritmasını uygulamışlardır (Schoenberg ve ark., 2010). FH algoritmasındaki hat ağırlıkları olarak noktaların konumsal Öklit uzaklıklarını, renk değerleri farklılıklarını ve bulundukları yüzeylerin açısallık farklılıklarını birlikte kullanmışlardır. Algoritmanın başarısını sadece karayolu segmentasyonu üzerinde test etmişlerdir. Yine aynı yıl Strom ve arkadaşları da benzer olarak FH algoritmasındaki renk farklılıklarına yüzeylerin açısallık farklılıklarını da dâhil ederek FH algoritmasına ilaveler yapmışlardır (Strom ve ark., 2010). Algoritma başarısı üzerindeki değerlendirme, kullanılan veri setinin doğrulama verileri olmadığı için yalnızca görsel olarak yapılmıştır. 2016'da da, Su ve arkadaşları, silindir şeklindeki yapıları (kanalları ve boruları) ve düzlemleri (duvarları) segmentlemek için graf araçlarını kullanan bir vokselleştirme yöntemi önermişlerdir (Su ve ark., 2016). Çalışmalarında, konumsal yakınlık, yön ve eğrilik özelliklerine göre vokselleri birleştiren bir birleştirme kriteri tanımlamışlardır.

Bölge genişletme için kullanılan bir diğer teknik de octree organizasyonudur. Wang ve Tseng, 3D nokta bulutu segmentasyonu için böl ve birleştir prensibi ile çalışan "Voxel Incremental Segmentation" (VIS) yöntemini geliştirmişlerdir (Wang ve Tseng, 2011). Yönteme göre ilk önce, nokta bulutu hiyerarşik olarak bölünmektedir. Bunun için octree yapısındaki her en alt seviye oktanın (leaf) temsil ettiği nokta grubu, belirlenen kritere uygun bir düzlemsel özelliğe sahip olana veya belirlenen vokselle büyüklüğüne

ulaşana kadar veri hiyerarşik olarak bölünmektedir. Daha sonra da, her biri düzlemsel özellik gösteren bu nokta grupları (vokseller içerisindeki düzlem özelliği göstermeyen küçük gruplar hariç) düzlem standart sapmalarına ve düzlem eğim açlarına (normal vektörlerine) göre iki aşamalı olarak birleştirilmektedir. İlk aşamada, birleştirildiğinde düzlem standart sapmasına göre yine bir düzlem özelliği gösterebilen düzlemler birleştirilmekte; ikinci aşamada ise birleştirildiğinde belirlenen eğim kriterine göre aynı yüzeye ait olabilecek (normal vektörleri arasındaki açı eğim kriterine uyan) düzlemler birleştirilmektedir. Bu çalışmaya benzer olarak daha sonra Xiao ve arkadaşları bir düzlem segmentasyonu yöntemi sunmuşlardır (Xiao ve ark., 2013). Bu yöntemde göre, 3D nokta bulutu ilk olarak octree bölünme tekniği ile “alt-pencereler” olarak isimlendirilen hücresel yapılara bölünmektedir ve bu alt-pencereler düzlemsel, düzlemsel olmayan ve dağınık olarak üç farklı etiket ile etiketlenmektedir. Verilen nokta sayısı parametresine göre alt-pencerelerin büyüklüğü belirlenmektedir. Bu etiketlenmiş alt-pencereler de daha sonra etiketleri ve geometrik özellikleri dikkate alınarak birleştirilmektedirler. Bu yöntemin VIS yönteminden başlıca farkı, bu yöntemde nokta bulutu, belirlenen nokta sayısına ulaşana kadar alt-pencerelere bölünürken; VIS algoritmasında ise veri, belirlenen düzlemsellik kriteri sağlanana veya belirlenen hacime (voksel büyüklüğüne) ulaşana kadar bölünmektedir. Daha sonra, VIS yöntemini temel alarak 2015 yılında Vo ve arkadaşları, vokselleri birleştirirken yüzey sınırlarını da koruyan "Octree-based Region Growing" (ORG) yöntemini sunmuşlardır (Vo ve ark., 2015). 3D nokta bulutunun düzlemsel yapıdaki yüzeylere bölünmesine ve bölge genişletme aşamalarına bakıldığında, ORG yönteminin VIS yöntemine çok benzediği dikkat çekmektedir. ORG yönteminin VIS yönteminden en belirgin farkı, ORG yönteminin düzlemsel yüzeyleri iki aşamada değil, tek bir aşamada birleştirmesidir.

Süper voksel tabanlı bölge genişletme yaklaşımı da literatürde yaygın bir şekilde kullanılmaktadır. Papon ve arkadaşları, vokseller arasındaki bağlantıları kullanan ve "Voxel Cloud Connectivity Segmentation" (VCCS) adlı ve süper voksel tabanlı bir aşırı-segmentasyon (over-segmentation) yöntemi sunmuşlardır (Papon ve ark., 2013). Yöntemde, belirlenen çözünürlükte seçilen büyüme noktalarının (büyüme için seçilen voksellerin) genişlemesiyle oluşan süper voksellerin yüzey sınırlarını ihlal etmemesi için aşırı-segmentasyon yapılmaktadır. Ayrıca, voksel özellikleri için Rusu ve arkadaşlarının (Rusu ve ark., 2009) tanımladığı hızlı nokta özellikleri histogramının (Fast Point Feature Histogram – FPFH) 33 ögesini (lokal yüzeylerdeki noktaların dağılımından elde edilen geometrik özellikler) ve renk bilgilerini kullanmışlardır. Benzer şekilde, Stein ve

arkadaşları çalışmalarında, süper vokseller içerisindeki vokselle bağlantıları için bir kriter kullanan “Local Convex Connected Patches” (LCCP) olarak isimlendirilen süper vokselle tabanlı bir yöntem geliştirmişlerdir (Stein ve ark., 2014). VCCS yöntemindeki voksel bağlantılarının belirlenmesinde bazı değişiklikler yaparak süper vokseller oluşturmuşlardır. Süper vokselleri içbükey, dışbükey ve düz olarak etiketlemişler ve sonra da rastgele seçilmiş bir vokselden başlayarak vokselleri birleştirme kriterlerine göre genişletmişlerdir. Daha sonra, Xu ve arkadaşları, voksel bağlantılarını ve nokta gruplarının geometrik özelliklerini kullanarak geliştirdikleri "Supervoxel and Graph-based Segmentation" (SVGS) adlı voksel tabanlı bir segmentasyon yöntemi için insan algı sistemine dayalı bazı gruplama kuralları tanımlamışlardır (Xu ve ark., 2017; Xu ve ark., 2018b). SVGS yöntemi ile LCCP yöntemi arasındaki fark, SVGS yönteminin, süper vokseller arasında bir graf yapısı oluşturması ve daha sonra bir süper voksel segmentleme yöntemi öneren ve bunları sınıflandırma, kayıt (registration) ve modelleme amaçlı birleştiren başka bir çalışmada (Ben-Shabat ve ark., 2018) kullanılan kriteri (Felzenszwalb ve Huttenlocher, 2004) kullanarak süper vokselleri birleştirmesidir. Bu yöntemde, süper vokseller arasındaki bağlantılar öncelikle ağırlık değerlerine (bağlantılar graftaki hatlar olarak düşünülürse) göre artan sırada sıralanmaktadır. Daha sonra minimum ağırlık değerine sahip bağlantıdan başlayarak, kullanılan kriterlere göre birleştirme için değerlendirilme yapılarak, bağlantının uçlarındaki süper voksellerin birleştirilip birleştirilmeyeceğine karar verilmektedir.

Bölge genişletme tabanlı yöntemler kategorisine dâhil olan yöntemlerin diğer kategorilerdeki yöntemlere göre en zayıf noktası, aykırı verilere karşı daha hassas yani bunlardan daha çok etkilenecek durumda olmalarıdır (Xu ve ark., 2018c). Ayrıca belirli şekillere sahip nesne ve yapıların tespit edilmesinde, genelde sadece lokal yüzey özellikleri temel alan bölge genişletme yaklaşımı, lokal özelliklere takılarak daha geniş alandaki asıl şekilsel yapıyı göz ardı edebilmektedir.

3. MATERYAL VE YÖNTEM

3.1. Önışlem Metotları

3.1.1. Kd-tree (çok boyutlu arama ağacı)

Nokta bulutu içerisinde yer alan noktaların 3D uzayda yakın komşuları ile oluşturduğu lokal düzlemsel yapılar, segmentasyon işleminde ayırt ediciliği yüksek özellikler sağlamaktadır (Babahajiani ve ark., 2017; Poux ve Billen, 2019). Noktaların arasındaki konumsal komşuluk (yakınlık) yapısı ve bu yapıya göre komşu olan noktaların 3D uzayda oluşturduğu lokal düzlemlerin eğim yönleri de literatürde nokta bulutu segmentasyonu için sıklıkla kullanılmaktadır (Xu ve ark., 2018b). Nesneler, yapısını oluşturan en düşük seviyedeki elementine kadar, 3D konumsal özelliğe sahip olarak gerçek hayatta konumlanmışlardır ve düz, kavisli veya karmaşık yüzeylerden oluşmaktadırlar. Yüzeylerin bu özellikleri, 3D nokta bulutu segmentasyonu aşamasında farklı nesnelere ait noktaları farklı segmentler içerisinde gruplamada ayırt edici özellik olarak kullanılmaktadır.

Nesneler lazer tarayıcı ile tarandığında, lazer ışınlarının çarptığı nesne yüzeylerinin 3D uzaydaki noktasal karşılıkları, sayısal olarak dijital ortama aktarılmaktadır. Tarama sonrasında dijital ortama aktarılan sayısal nokta özelliklerinin bir araya gelmesi, nokta bulutu verisini oluşturmaktadır. Bir noktanın en yakın komşu noktalar ile beraber 3D uzayda oluşturduğu geometrik yapı, nesnenin o bölgedeki lokal yüzeyi hakkında bilgi verebilmektedir. Bu lokal yüzey küçük bir bölgeyi temsil ettiğinde genelde (lokal yüzey eğer nesnenin kenar veya köşe gibi değişken kısımlarında yer almıyorsa) küçük bir düzlemsel yapıya benzemektedir. Bu durumda, lokal yüzeyler için 3D uzayda matematiksel olarak ifade edilebilen bir düzlem uydurulabilmektedir ve bu düzlemin yönü “normal vektör” veya “yüzey normali” olarak ifade edilebilen bir vektör (bir düzlemin üç boyutlu uzayda yönünü gösteren birim vektör) ile temsil edilebilmektedir. Birbirine bitişik iki lokal yüzeyin normalleri arasındaki açısal fark o bölgedeki lokal eğim açısını vermektedir. Bu eğim açısı 3D nokta bulutu segmentasyonunda geometrik bir temel özellik olarak yaygın bir şekilde kullanılmaktadır (Rabbani ve ark., 2006).

Literatürde yapılmış olan çalışmalarda, lokal yüzey normallerinin hesaplanması için gerekli olan lokal komşu noktaların belirlenmesi ve yüzey normallerinin

hesaplanması işlemleri genelde iki farklı yöntem ile yapılmaktadır. Birincisi her nokta için belirli sayıdaki veya belirli bir yarıçap içerisindeki komşu noktaları bulmak ve her noktanın komşu noktalar ile oluşturduğu nokta grubunu kullanarak yüzey normalini hesaplamaktır. Bu yöntemde, komşu noktaları bulmak için “çok boyutlu ikili arama ağacı” (kd-tree) gibi çok boyutlu verileri hızlı arama için düzenleyen veri yapıları kullanılmaktadır.

Kd-tree (çok boyutlu arama ağacı) yapısı; tek boyutlu veriler üzerinde arama işlemi gerçekleştiren ikili arama ağacını (binary search tree) temel alarak, çok boyutlu veriler üzerinde arama gerçekleştiren bir veri yapısıdır (Bentley, 1975; Arya ve Mount, 1993). Bu teknik ile bir verinin aranmasında harcanan zaman karmaşıklığı, doğrusal aramadaki $O(n)$ yerine $O(\log n)$ olmaktadır (Friedman ve ark., 1977). Kd-tree tabanlı yaklaşıma göre, 3D nokta bulutu içerisindeki noktalar Öklit uzayındaki 3D konumlarına göre hiyerarşik olarak alt dallara ayrılmaktadır (Lari ve Habib, 2014; Lozes ve ark., 2015). Dallanma, yapraklardaki nokta sayısı bire indiğinde veya belirli bir değere veya altına düştüğünde sonlandırılabilir.

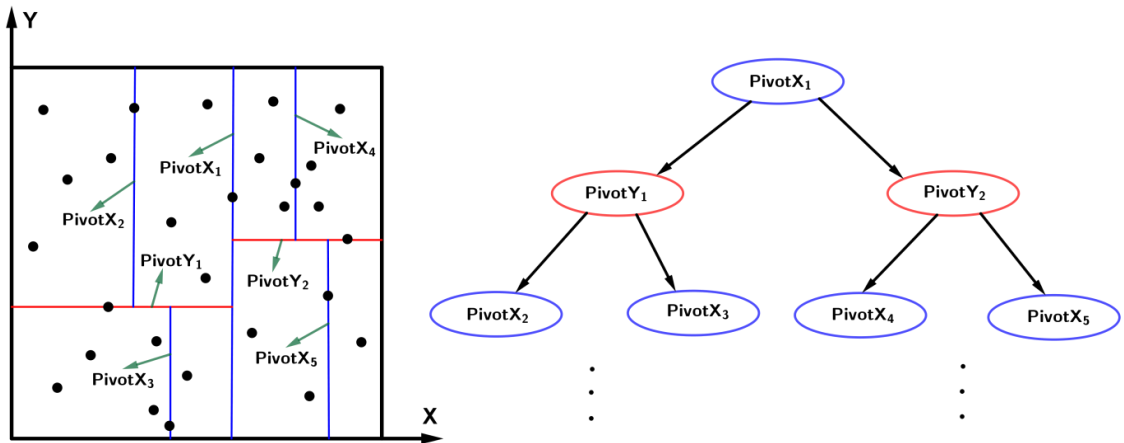
Çok boyutlu bir veride, veri elemanları “vektör değerleri” olarak adlandırılan birden fazla konum değerine sahiptir. 3D nokta bulutlarında da noktalar, x-y-z eksenleri üzerindeki x, y ve z değerleri olarak adlandırılan üç adet konum (uzamsal) vektör değerine sahiptir. Kd-tree yapısı organize edilirken, veri elemanları “pivot” değeri olarak isimlendirilen bir eksenindeki konum değerine göre (x, y veya z) iki alt gruba ayrılmaktadır. Alt gruplardaki elemanlar da kendi içlerinde aynı veya başka bir eksenindeki seçilen pivot değerine göre alt gruplara dallanmakta ve her iterasyonda dallanma sonlanana kadar bu böyle devam etmektedir. Örneğin; 3D nokta bulutu için oluşturulan bir kd-tree organizasyonu için, noktalar önce x ekseninde seçilen bir pivot değerine göre iki alt gruba ayrılır. Daha sonra, alt gruplar da kendi içlerinde algoritmaya göre (örneğin sırayla veya nokta dağılım şekline göre) seçilen y eksenindeki (sırayla seçildiği varsayılırsa) pivot değerine göre alt gruplara ve onlar da z eksenindeki pivot değerine göre alt gruplara ayrılır (ve daha sonra tekrar başa dönerek, x ekseninden tekrar başlar).

Kd-tree yapısının bilgisayar üzerinde uygulanmasında ise, her bölünmede ağaç veri yapısında bölünmüş noktalar için yeni bir bellek alanı oluşturmaktansa, bellekte var olan elemanların, belirlenen pivot değerine eşit veya daha küçük olanlar bir arada diğerleri kendi aralarında birlikte tutulacak şekilde yerleri değişmesi bellek açısından çok büyük bir verimlilik sağlayacaktır. Bu yer değişikliği için, yer değişikliği ile sıralama yapabilen Quicksort (hızlı sıralama) (Cormen ve ark., 2009) gibi bir sıralama algoritması

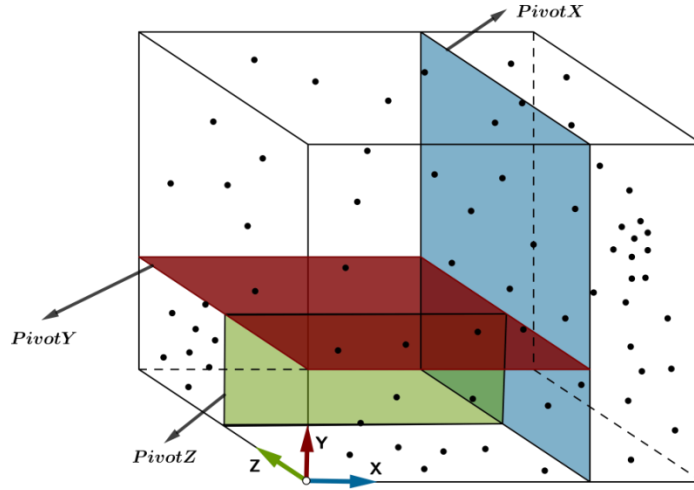
kullanılabilmektedir. Böylece ağaçtaki düğümlerin, temsil ettikleri elemanlar için ayrı bir bellek alanı tutmak yerine, temsil ettiği elemanların bellekteki başlangıç ve bitiş adreslerini tutması yeterlidir.

Pivot değeri belirlemek için ikiye ayrılacak olan nokta grubunun, belirlenen eksenindeki koordinat değerlerinin ortalaması (ağırlık merkezi), o eksenindeki ortanca elemanın koordinat değeri, o eksenindeki en küçük ve en büyük değerlerin ortalaması (hacimsel orta nokta) veya bellekte tutulma sırasına göre ortadaki elemanın koordinat değeri gibi farklı yöntemler ile seçilebilmektedir. Şekil 3.1’de, bölünme ekseninin sırayla seçildiği ve pivot elemanın bellekteki sırada direk ortadaki elemanın seçildiği bir kd-tree organizasyonu örneğinin 2D temsili ve Şekil 3.2’de bölünme aşamasının 3D gösterimi görülmektedir.

Çok boyutlu yapıya sahip 3D Lidar nokta bulutu verilerinin kd-tree veri yapısı ile hiyerarşik yapılara bölünerek, bir ağaç yapısı elde edilmesine literatürde sık rastlanmaktadır. Literatürde kd-tree yapısı çoğunlukla bir noktanın en yakın komşularını aramada ve konumsal değerleri verilen bir noktayı nokta bulutu içerisinde aramada kullanılmaktadır (Golovinskiy ve Funkhouser, 2009; Connor ve Kumar, 2010; Lari ve Habib, 2014).



Şekil 3.1. Çok boyutlu ikili arama ağacı organizasyonunun 2D temsili



Şekil 3.2. 3D veri için çok boyutlu arama ağacı bölünmesinin gösterimi

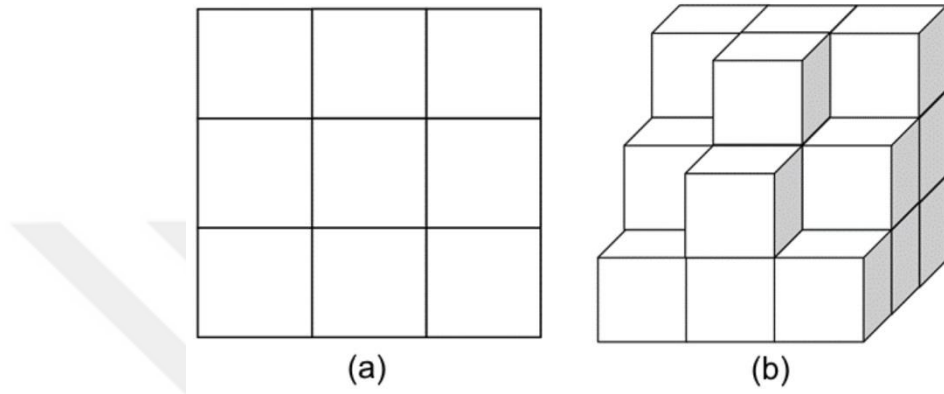
Tez kapsamında kd-tree, veri setlerinin hazırlanmasında gürültü giderme aşamasında ve noktaların yoğunluk miktarının analizi kısmında komşulukların hesaplanması için kullanılmıştır. Ayrıca başarı değerlerinin hesaplanmasında, konum değerleri bir veriden (sonuç nokta bulutu verisi gibi) alınan bir noktanın başka bir veri (referans nokta bulutu verisi gibi) içerisinde aranması amacıyla da kullanılmıştır.

3.1.2. Octree (sekiz dallı ağaç) organizasyonu ile vokselleme ve indeksleme

Literatürde yapılmış olan çalışmalarda, lokal komşu noktaların belirlenmesi ve yüzey normallerinin hesaplanması işlemlerinde kullanılan bir diğer yöntem ise, noktaları eşit büyüklüklerde kübik yapılar (vokseller) içerisinde örnekleyerek lokal gruplar (komşuluklar) oluşturmak ve içerisindeki bu grupları kullanarak her voksel için yüzey normalini hesaplamaktır (Vo ve ark., 2015). Bu vokselleme işlemi, lokal yüzey normallerinin hesaplanması için lokal grupları oluşturmak, lokal yüzeylerin aralarındaki komşulukları bir indeksleme sistemine göre organize edilmiş yapı üzerinde belirlemek ve gürültü verilerini lokal gruplar içerisinde bastırmak gibi birçok ihtiyacı karşılamaktadır.

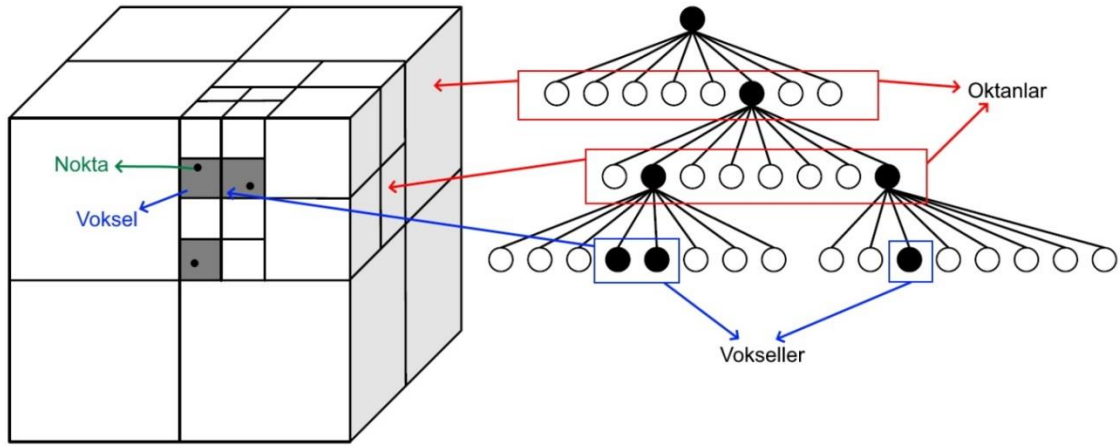
“Sekiz dallı ağaç” (octree) organizasyonu, verilerin hiyerarşik bir yapı içerisinde organize edilerek indeksli düzende örneklenmesinde yaygın olarak kullanılan bir vokselleme tekniğidir. 3D nokta bulutu verilerini basit bir şekilde üç boyutlu dizi içerisinde örneklemek yerine octree organizasyonu ile örneklemenin nedeni, nokta bulutunu kapsayan üç boyutlu hacim içerisindeki voksel büyüklüğündeki her boş alanın indeksini tutmak için de bellekte adres alanı ayrılmasının önüne geçmektir (Jain ve ark., 1999).

“Voksel” (voxel) kavramı, 3D nokta bulutunda eşit hacimlerde ve eşit kenarlara sahip hücresel yapıları ifade etmektedir (Aijazi ve ark., 2013; Su ve ark., 2016; Xu ve ark., 2017). Bir ızgara yapısı içerisine alınmış 2D bir veride Şekil 3.3 (a)’daki gibi ızgara çizgileri içerisindeki her bir bölgeye “piksel” (hücre) denilmektedir. Şekil 3.3 (b)’deki gibi ızgara yapısı içerisine alınmış 3D bir veride de bu hücrelerin karşılığı “voksel” olarak tanımlanmaktadır.



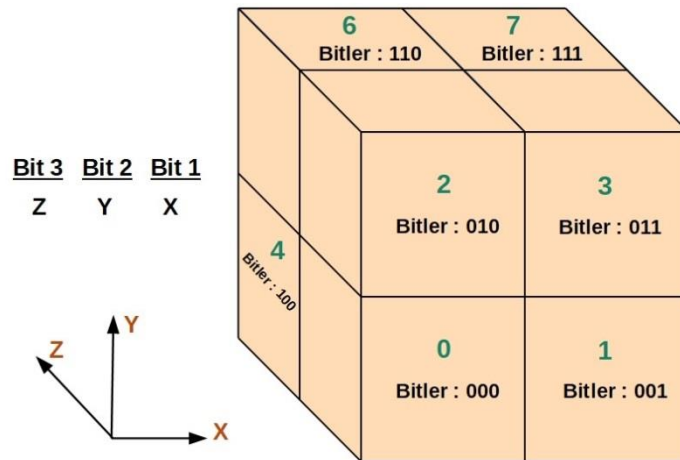
Şekil 3.3. 2D konumsal uzayındaki pikseller (a) ve 3D konumsal uzayındaki vokseller (b)

3D yaklaşımda, nokta bulutu kübik yapı olarak düşünülüp, tüm veri voksellere bölünerek, işlem yapılmaktadır. Ancak, bir nokta bulutu verisinde çeşitli nesneler olduğu gibi ayrıca boşluklar da bulunmaktadır. Tüm veri kübik yapıda tek bir hücre içine alındıktan sonra, veri içinde bulunan her boşluk için de ayrı bir hücre oluşturmak aşırı miktarda bellek tüketmektedir. Bu sorunu aşmak için hiyerarşik bir bölünme yöntemi olan “sekiz dallı ağaç” (octree) yapısı kullanılmaktadır. Bunun için octree yapısı ile her adımda (alt seviyede), içerisinde nokta bulunan hücreler sekiz eşit hücreye bölünmektedir. Bu hücreler octree yapısı içerisinde “oktan” olarak isimlendirilmektedir ve bölünen her hücre sekiz adet oktanı sahip olmaktadır. Bölünmeler sonucunda, küçülerek istenilen büyüklüğe ulaşmış ve içerisinde nokta bulunan hücrelerin karşılığı olan oktanlar da vokselleri temsil etmektedir (Dybedal ve ark., 2019). Şekil 3.4’te hücrelerin bölünmesiyle oluşan oktanlar ve vokseller örnekler ile gösterilmektedir.



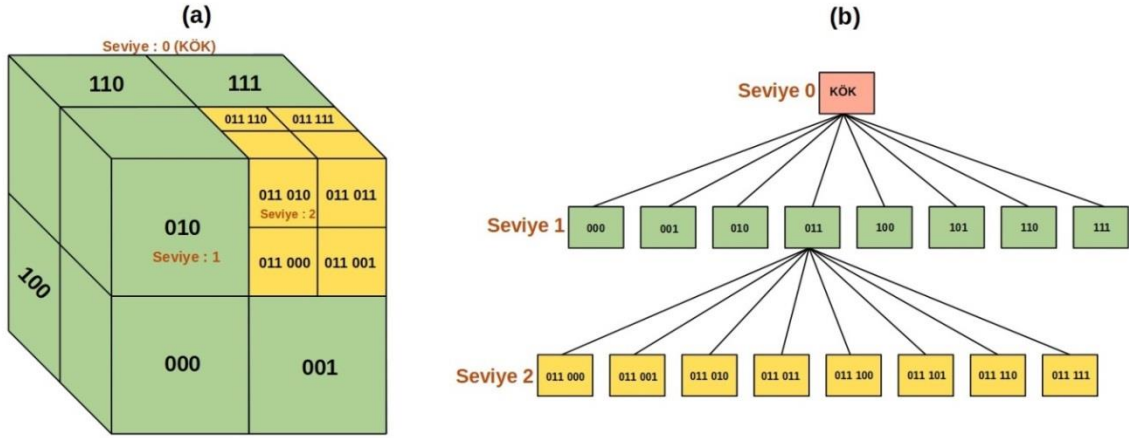
Şekil 3.4. Hücrelerin octree yapısı ile bölünmesinden oluşan oktanlar ve vokseller

Hücreler, nokta dizisinin bellekteki başlangıç ve bitiş indekslerini tutmaktadır. Eğer bitiş indeksi başlangıç indeksinden küçük olursa yani içinde nokta olmayan boş bir hücre olursa o hücre için bellekte veri yapısı oluşturulmayıp, ebeveyn hücreden onu gösteren bağlantıya boş (null) değer atanmaktadır. Bu sayede boş alanlar için bellekte fazladan yer kaplayacak vokseller oluşturulmamaktadır. Bölünmenin son seviyesinde, içerisinde nokta bulunan hücreler (vokseller) octree yapısındaki yaprakları temsil etmektedir (Lohmann, 1998). Her hücre daha sonra komşu hücrelerin bulunabilmesi için alt seviyeye doğru bit düzeyinde kaydırılarak, Morton sırasına (Morton, 1966) göre numaralandırılmaktadır (Samet, 1989; Jiang, 1996). Morton sırasına göre 3 bit (ikili sayı) ile “oktan” olarak isimlendirilen 8 hücre indekslenilmektedir. Her bir bit, hücrenin bir eksene göre konumunu göstermektedir. Şekil 3.5’te Morton sırasına göre numaralandırılmış oktanlar görülmektedir.



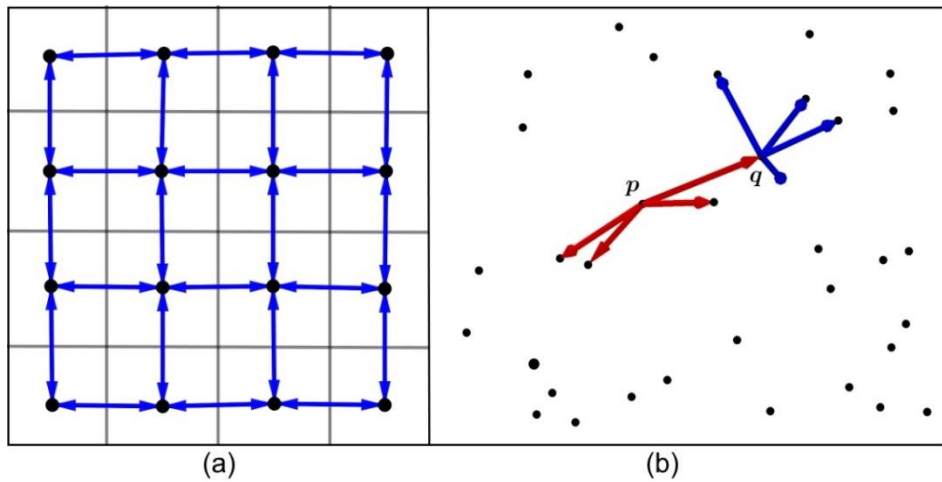
Şekil 3.5. Morton sırasına göre indekslenmiş bir sekizli bölünme

Şekil 3.6 (a)'da bir hücrenin octree organizasyonu ile bölünmesi ve hücrelerin ikilik (bitset) düzende numaralandırılması, Şekil 3.6 (b)'de ise octree yapısının seviye seviye ağaç diyagramı görülmektedir. Bitset indeksi “101” olan hücre arka-alt-sağ köşede kaldığı için figürde görülmemektedir.



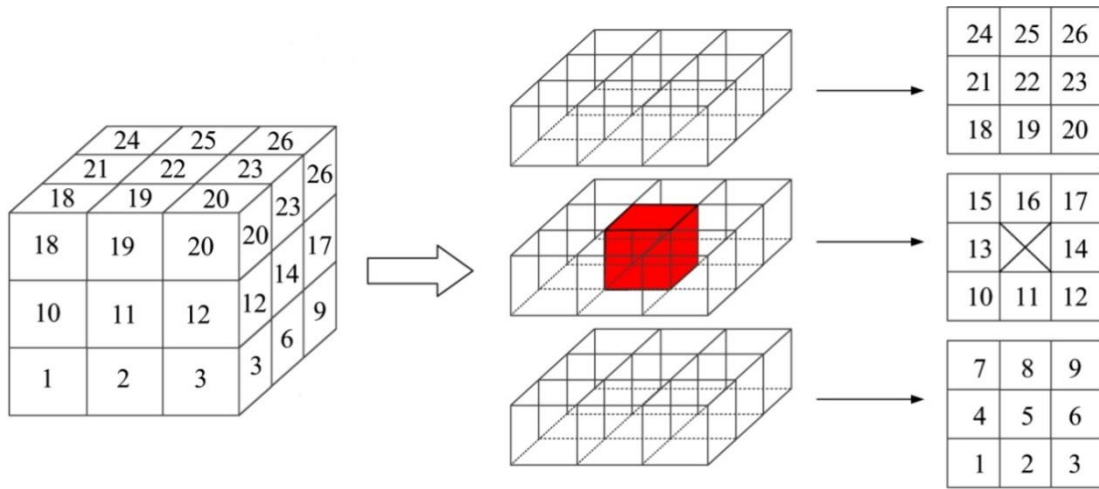
Şekil 3.6. Hücrelerin seviye seviye bölünmesinin octree yapısı ile gösterimi

Oluşturulan octree yapısı ile noktalar vokseller içerisinde gruplanarak, düzenli bir yapıda olması sağlanmıştır. Düzenli yapı içerisinde yer alan bir vokselin, komşu voksellerinin tespit edilmesi; dağınık bir bulut içerisinde yer alan bir noktanın komşu noktalarının elde edilmesinden çok daha hızlı olmaktadır. Çünkü düzenli yapıdaki vokseller indekslenmiştir ve bu indeksler kullanılarak bir vokselin komşu vokselleri hesaplanabilmektedir (Şekil 3.7 (a)). Nokta bulutu içerisindeki noktaların ise bir düzenli yapı içerisindeki konumunu temsil eden bir yer indeksi bulunmamaktadır (Şekil 3.7 (b)).



Şekil 3.7. Düzenli bir yapı içerisindeki komşuluk (a) ve düzensiz bir yapı içerisindeki komşuluk

Tez kapsamında, bir vokselin komşu vokselleri olarak $3 \times 3 \times 3$ komşuluk yapısı değerlendirilmiştir (Cheng ve ark., 2015). Yani, kübik yapıda olan voksellerin yüz, kenar ve köşe komşuluklarına bakılarak 26 komşusu aranmaktadır (Şekil 3.8). Voksel değerleri Şekil 3.8'deki gibi olan bir nokta bulutunda, ortadaki vokselin komşulukları örnekteki gibi alınmaktadır.

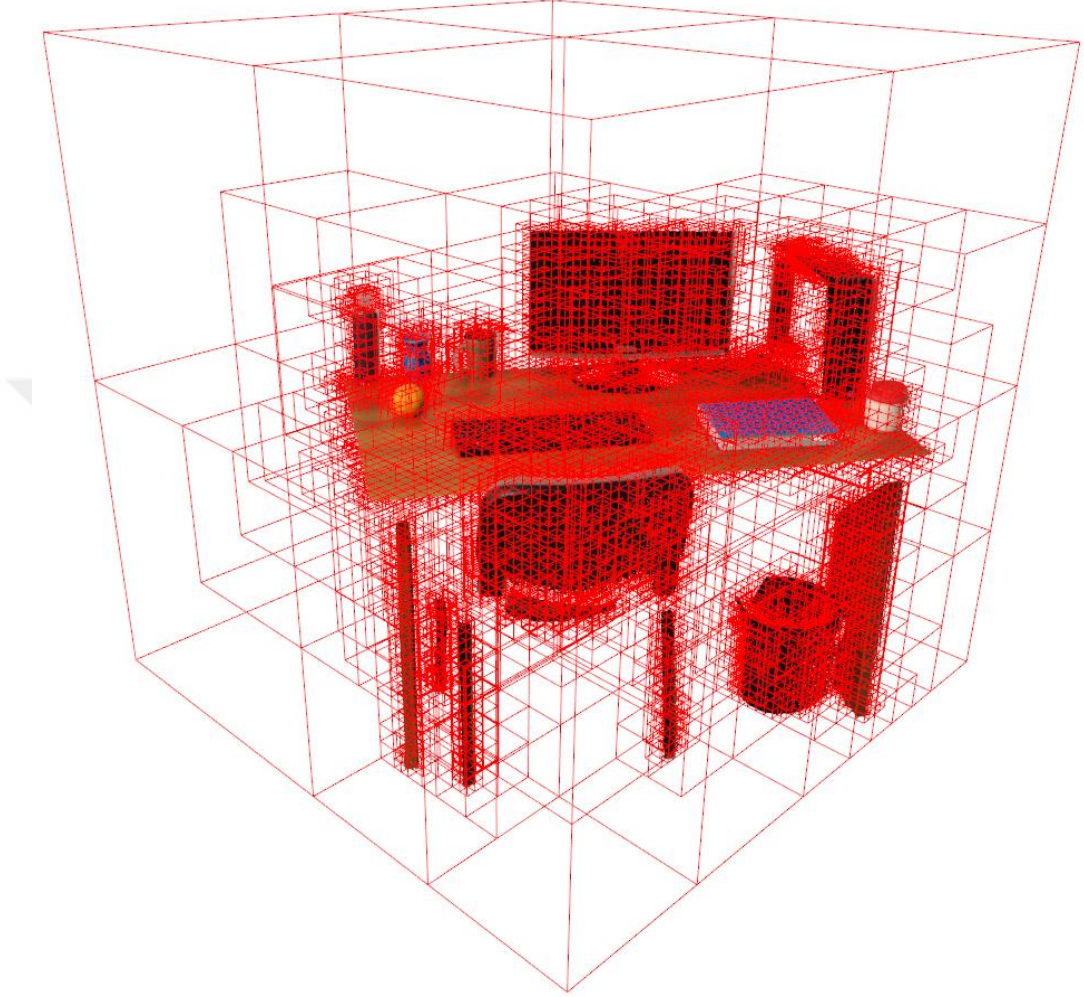


Şekil 3.8. Bir vokselin 26'lık komşuluk düzenine göre komşuları

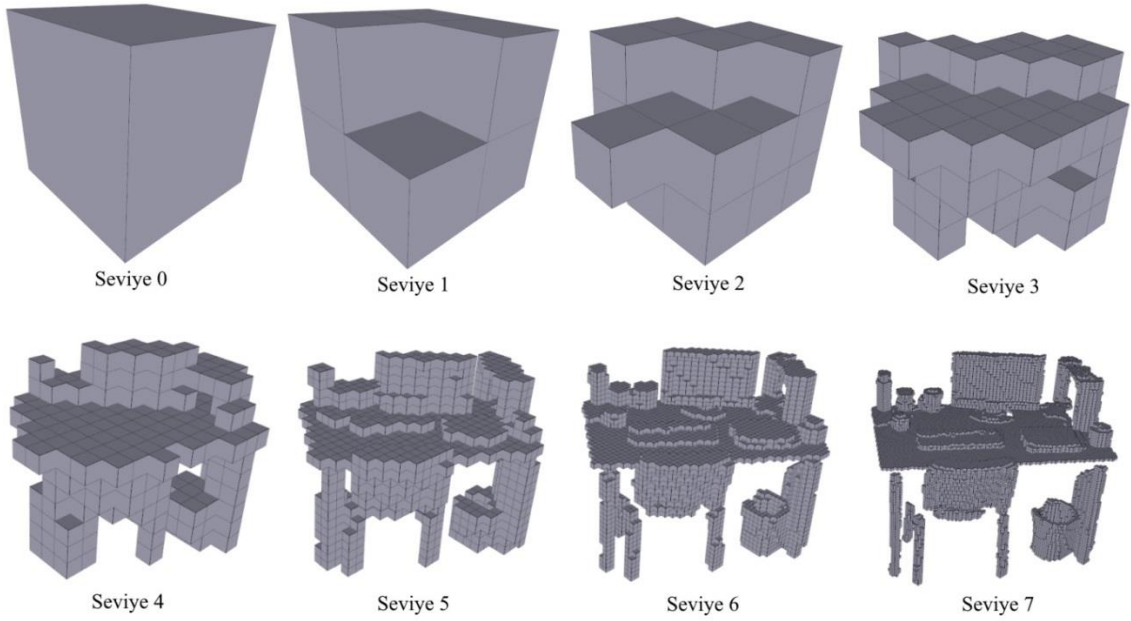
Octree yapısında, bir vokselin komşu voksellerini bulabilmek için indeks değerlerine bitisel düzeyde bakılmaktadır. Her 3 bit, bir hücrenin ebeveyn hücresinin kaçınıcı indeksindeki çocuğu olduğunu ve Morton sıralaması da göz önünde bulundurulduğunda ebeveyn hücresine göre konumunu göstermektedir.

Bir vokselin komşu vokselleri aranmadan önce, ilk olarak o vokselin (işlem yapılan voksel) indeks değeri çözülerek x-y-z olarak üç ayrı eksen üzerindeki ayrık değeri elde edilmektedir. Daha sonra, bu değerlere göre $3 \times 3 \times 3$ komşuluk yapısı içerisinde sırasıyla o voksele komşu vokseller taranmaktadır. Taranan bir komşu vokselin de x-y-z olarak ayrık eksen değerleri yine Morton sırasına göre tek bir indeks değerine kodlanmaktadır. İlk önce, işlem yapılan voksel ile taranan vokselin indeks değerleri bitisel düzeyde karşılaştırılarak octree yapısı içerisindeki ata (kök) hücre bulunmaktadır. Daha sonra da ata hücreden başlayıp aranan hücrenin indeks bitleri üçer üçer kaydırılarak taranan voksele ulaşılmaktadır. Üst hücreden (üst seviyeden) alt hücrelere (alt seviyelere) doğru bitleri kaydırarak ilerleme esnasında, yol üzerindeki bir çocuk hücre boş değer gösteriyorsa aranan voksel yok demektir ve arama boş değer döndürerek sona ermektedir.

Şekil 3.9’da gerçek hayattan alınmış bir nokta bulutunun octree ile voksellenmesiyle oluşan bölünme konumları ve Şekil 3.10’da aynı verinin farklı seviyelerde voksellenmesi gösterilmektedir.



Şekil 3.9. Gerçek tarama ile alınmış bir nokta bulutunun octree ile bölünme konumları



Şekil 3.10. Gerçek taramadan alınmış bir nokta bulutunun farklı seviyelerde vokselleneşmesi

Tez kapsamında octree; veri setlerinin hazırlanmasında, iç mekândan alınan verinin indirgenmesinde ve geliştirilen segmentasyon yönteminde bir ön aşama olarak kullanılmıştır.

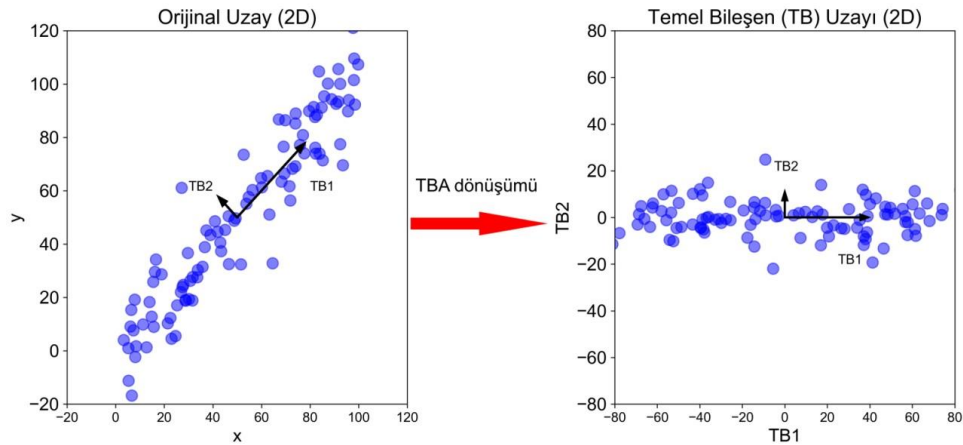
3.1.3. Temel Bileşen Analizi (TBA) ve normal vektör

Temel Bileşen Analizi (TBA) (Principle Component Analysis – PCA), çok boyutlu uzaydaki bir verinin daha düşük boyutlu bir uzaya izdüşümünü, varyansı maksimize edecek şekilde bulma yöntemidir. Uzayda bir noktalar kümesi için, tüm noktalara ortalama uzaklığı en az olan “en uygun doğru” seçilir. Daha sonra bu doğruya dik olanlar arasından yine en uygun doğru seçilerek, bu adımlar, yeni bir boyutun varyansı belirli bir eşiğin altına inene kadar tekrarlanır. Bu sürecin sonunda elde edilen doğrular, bir doğrusal uzayın tabanlarını oluşturur. Bu taban vektörlerine “temel bileşen” denir.

TBA, çok boyutlu bir uzayda yer alan verinin boyutunu, veri içerisindeki elemanların uzaydaki dağılım şekillerini mümkün olduğunca koruyarak verinin daha az boyut ile ifade edilmesini sağlayan bir dönüşüm tekniğidir (Yazar ve ark., 2009; Abdi ve Williams, 2010; Jolliffe ve Cadima, 2016). Bu analiz kullanılarak, eldeki veri daha az sayıda değişkenle ifade edilebilecek en iyi dönüşüm amaçlanmaktadır. Bu dönüşüm ile elde edilen değişkenler orijinal değişkenlerin temel bileşenleri olarak adlandırılmaktadır. Temel bileşenler, varyans (değişim) değeri büyükten küçüğe doğru olacak şekilde

sıralanmaktadır ve ilk temel bileşendeki öz vektör, verideki dağılımın en fazla olduğu yönü gösterirken karşılığı olan öz değer de bu yöndeki dağılım miktarı hakkında bilgi vermektedir.

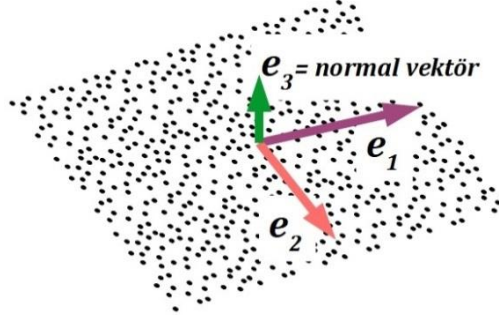
TBA tekniğinde, çok boyutlu uzaydaki bir verinin, verideki değişkenlerin varyansının maksimum olacağı daha düşük boyutlu bir uzaydaki izdüşümü elde edilmeye çalışılmaktadır. Verinin ilk bulunduğu uzayda, tüm noktalara ortalama uzaklığı en az olan doğru, dağılımın en fazla olduğu yön olarak seçilmektedir. Daha sonra bu doğruya dik olan doğrular arasında en uygun olanı seçilmekte ve belirlenen boyut sayısına veya o adımda seçilen en uygun yöndeki varyans değeri belirlenen eşik değerinin altına inene kadar bu işlem devam etmektedir. İşlemin sonunda elde edilen doğrular, dönüştürülen uzayın yeni eksenlerini oluşturmaktadır. Şekil 3.11’de, bir verinin 2D orjinal uzayından 2D temel bileşen (TB) uzayına aktarılmasını gösteren bir örnek yer almaktadır. TB uzayının eksenleri temel bileşenlerin (TB1 ve TB2) yönleri ile temsil edilmektedir.



Şekil 3.11. Bir verinin TBA dönüşümü ile temel bileşen (TB) uzayında temsil edilmesi

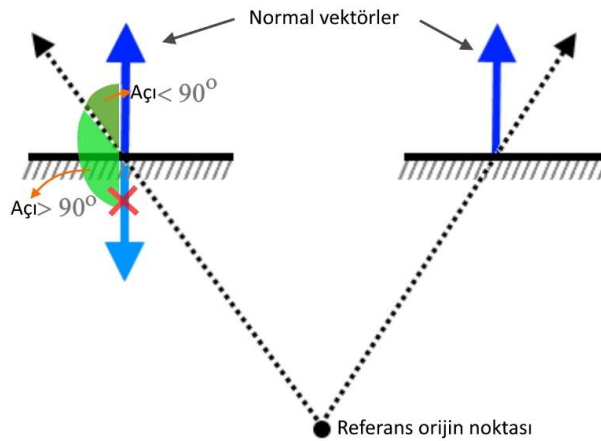
TBA, makine öğrenmesi alanında sınıflandırma, kümeleme ve gürültü giderme gibi alanlarda boyut indirgemek için kullanılmaktadır. Bu teknik her ne kadar boyut indirgeme için sıklıkla kullanılsa da, 3D nokta bulutu işleme alanında kullanıldığı gibi, sadece temel bileşenlerini elde etmek için de kullanılabilir. Bir voksel içerisindeki noktalar için uygun bir düzlem uydurulabilmesi ve uydurulan düzlemler arasındaki eğim derecelerini hesaplayabilmek için, bu noktalara TBA uygulanmaktadır (Shakarji, 1998; Rabbani Shah, 2006). 3D bir uzayda bulunan nokta bulutu için TBA hesaplanması sonrasında temel dağılım yönlerini gösteren üç adet özvektör (e_1 , e_2 ve e_3) ve noktaların bu yönlerdeki varyansını sırasıyla gösteren üç adet öz değer ($\lambda_1 \geq \lambda_2 \geq \lambda_3$) elde

edilmektedir. En düşük öz değere karşılık gelen öz vektör, bu nokta grubunun düzlemsel eğim yönünü yani “normal” vektörünü göstermektedir. Şekil 3.12’de noktaların en az değişim yönü olan e_3 vektörü, noktaların düzlemsel olarak yönünü göstermektedir.



Şekil 3.12. TBA sonrası elde edilen öz vektörlerin yönleri

İki yüzeyin normalleri birbirine paralel olsa da zıt yönleri gösterebilirler. Normalleri aynı yöne döndürmek için, bir referans orijin noktası ile voksellerin ağırlık merkezleri arasındaki doğruların birim vektör değerleri kullanılabilmektedir (Lari ve Habib, 2014; Stein ve ark., 2014; Verdoja ve ark., 2017; Xu ve ark., 2018b; Sağlam ve ark., 2020). Bir vokselin birim normal vektörü ile orijin noktasından vokselin ağırlık merkezine olan doğrunun birim vektörü arasındaki açısal fark, 90° açısından daha yüksekse (veya düşüksen), vokselin normal değerlerinin işareti değiştirilir. Bu şekilde, normaller arasındaki açısal farklar 90° 'ye kadar çıkabilir, çünkü Şekil 3.13'te görüldüğü gibi önemli olan vektörlerin eğimlerinin yönleri değil, eğimlerin doğrultusudur. İki vektör birbirine paralel ancak zıt yönlerde ise, bu işlem sonrasında açısal farkları sıfır olmaktadır.



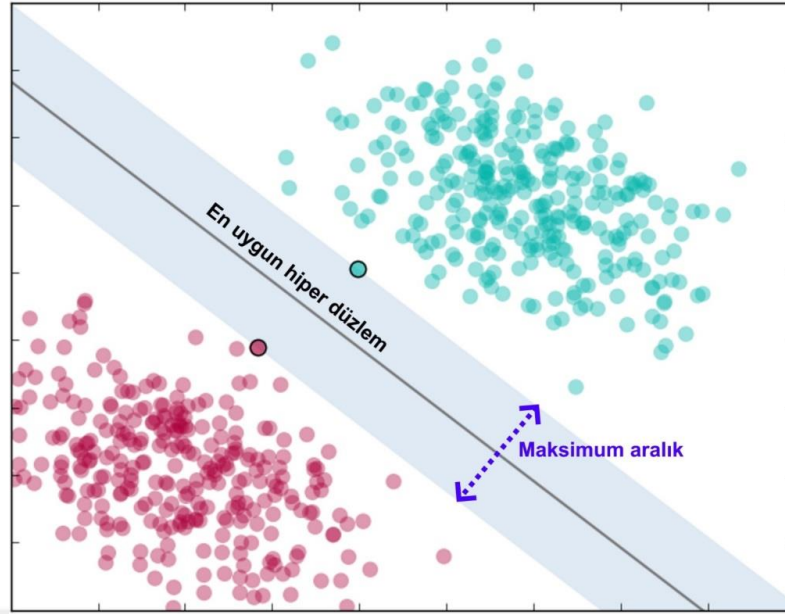
Şekil 3.13. Bir referans orijin noktasına göre normal vektör yönünün ayarlanması

3.2. Sınıflandırıcılar

Sınıflandırma, bir verideki elemanların önceden tanımlanmış bilgilere ve bulunduğu duruma göre ait olduğu kategoriye atanmasıdır. Veri madenciliğinde kullanılan sınıflandırma algoritmalarının (sınıflandırıcıların) genel çalışma mantığına göre, ilk başta örnek veriler (eğitim verileri) ile eğitilmekte ve daha sonra bu eğitilmiş sınıflandırıcı (eğitilmiş model) ile yeni verileri sınıflandırmaktadır. Oluşturulan modelin performansı, öğrenme kümesinde yer almayan veriler (test verisi) ile test edilerek de ölçülmektedir. Her sınıflandırıcının kendisine özgü model yapısı vardır ve eğitilmesi de sınıflandırıcının yapısına ve çalışma şekline göre değişmektedir (Duda ve ark., 2001). Yapılan tez çalışmasında, geliştirilen segmentasyon yönteminin bir ara işlem olarak uygulanmasının ardından, 3D nokta bulutu verisinin sınıflandırma başarıları da test edilmiştir.

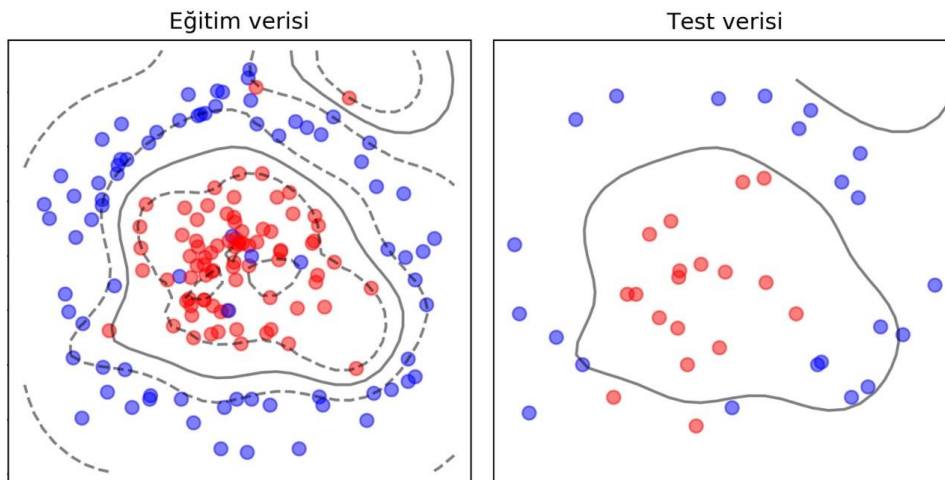
3.2.1. Destek Vektör Makinesi (DVM)

Destek Vektör Makinesi (DVM) (SVM – Support Vector Machine) sınıflandırıcısı; sınıflandırma, regresyon veya aykırı değer tespiti gibi diğer görevler için kullanılabilen yüksek veya sonsuz boyutlu bir özellik uzayında bir hiper düzlem (özellik uzayının boyutunun bir alt boyutundaki ayırıcı düzlem) bulur. Son sınıflandırma, herhangi bir sınıfın hiper düzleme en yakın olan eğitim verisinden hiper düzleme olan mesafesini maksimize ederek ve sınıflandırıcının genelleme hatasını en aza indirerek elde edilmektedir (Cristianini ve Shawe-Taylor, 2000; Hastie ve ark., 2009; Zhang ve ark., 2013). DVM sınıflandırıcısının asıl çalışma mantığı ikili sınıflandırma üzerine kurulmuştur. Daha sonra çoklu sınıflandırma için genişletilmiştir. İkili sınıflandırmadaki mantık, iki sınıfı ayıracak en uygun düzlemi bulmaktır. Bu düzlemin uygunluk derecesi, düzlemin her iki sınıfa olan uzaklığının eşit olmasıyla ifade edilmektedir (Burges, 1998). Şekil 3.14’te örnek bir veride DVM sınıflandırıcısı için bulunan en uygun hiper düzlem görülmektedir.



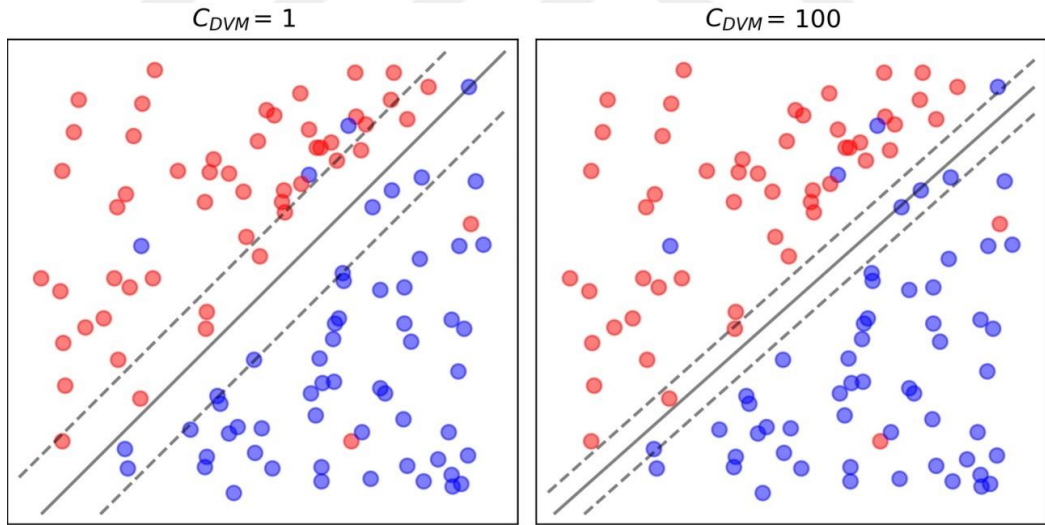
Şekil 3.14. Örnek bir veride DVM için bulunan en uygun hiper düzlem

DVM sınıflandırıcısının parametrelerinden birisi olan çekirdek işlevi (kernel function) ayırıcı düzlemin şeklini belirtmektedir. Çekirdek işlevi doğrusal (linear) veya doğrusal olmayan olarak tercih edilebilmektedir (Busuttil ve ark., 2004). Doğrusal olmayan çekirdek işlevine örnek olarak sigmoid, poly (çoklu) ve RBF (radial basis function – radyal temel işlevi) verilebilir. Bir diğer parametresi olan “gamma” değeri, doğrusal olmayan düzlemin çekirdek katsayısını (kernel coefficient) belirtmektedir ve verideki özellik sayısına göre otomatik olarak belirlenebilmektedir. Şekil 3.15’te RBF çekirdek işlevi ile eğitim verisi üzerinde bulunan hiper düzlemin test verisinde uygulanmasına bir örnek gösterilmektedir.



Şekil 3.15. RBF doğrusal olmayan çekirdek işlevi ile eğitim verisi üzerinde bulunan hiper düzlemin test verisinde uygulanması

Bir diğer parametre olan C_{DVM} parametresi, düzenleyici parametredir ve pozitif olması gerekmektedir. Varsayılan değeri olan 1 gibi bir değerle kullanılabilirken değer yükseldikçe daha sıkı kurallarla çalışmaktadır. Yüksek C_{DVM} değeri eğitim verisini sınıflandırırken daha hassas düzlemler seçerek daha fazla eğitim başarısı sağlarken, tamamen eğitim verisine göre şekillendiği için test verisi üzerinde daha başarısız sonuç verebilmektedir. C_{DVM} parametresinin değeri küçüldükçe verileri daha fazla genelleyerek hiper düzlem oluşturmakta ve eğitim verisinden farklı gelen test verilerine karşı daha genelleyici olmaktadır. Diğer bir ifade ile eğitim verisinin ezberlenmesini sağlamaktadır. Çok küçük C_{DVM} değerinde ise yeterli eğitim sağlanmamış olmakta ve kabaca bir sınıflandırma yapmaktadır. Şekil 3.16'da farklı C_{DVM} değerleri için bulunan en iyi doğrusal hiper düzlem için örnekler görülmektedir. Şekilde C_{DVM} değerinin 1 olduğu durumda daha çok veri elemanının yer aldığı geniş bir alandaki veri elemanlarının uzaydaki konumlarına göre kabaca bir hiper düzlem bulurken, C_{DVM} değerinin 100 olduğu durumda daha az veri elemanının yer aldığı dar bir alandaki veri elemanlarının uzaydaki konumlarına göre hiper düzlem bulmaktadır.

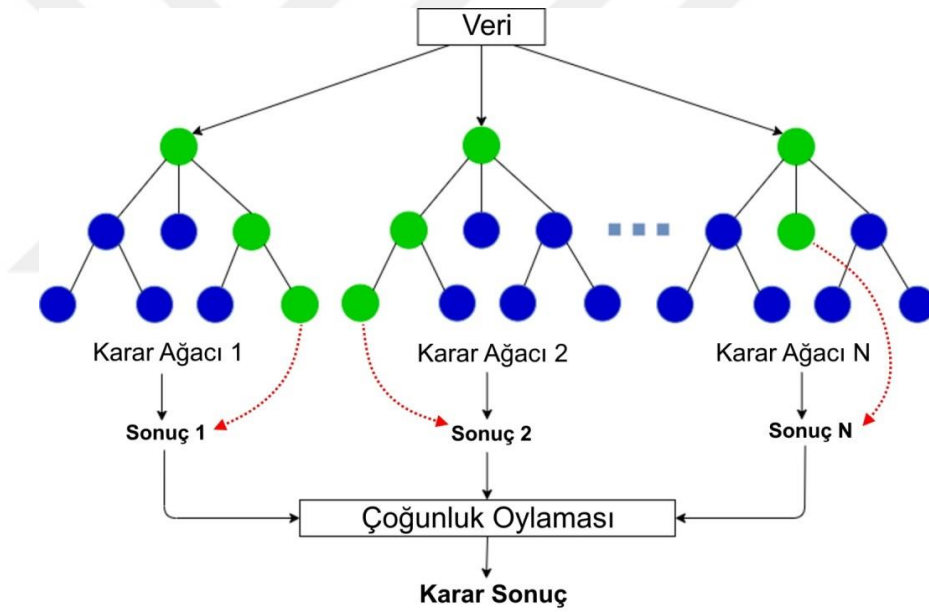


Şekil 3.16. Farklı C_{DVM} parametre değerleri için bulunan en iyi doğrusal hiper düzlem için örnekler

3.2.2. Rastgele Orman (RO)

Rastgele Orman (RO) (RF – Random Forest) sınıflandırıcısı, rastgele bir orman oluşturmak için kullanıcı tanımlı ağaç sayısı kadar rastgele oluşturulmuş karar ağacının bir araya gelmesiyle oluşmaktadır. Her karar ağacı, birbirinden bağımsız olarak kullanıcı

tanımlı değişken sayısına göre bölünerek üretilir ve verilerin farklı şekilde seçilmiş alt kümeleriyle eğitilir. Bu karar ağaçlarının bir araya gelmesi eğitim modelini oluşturmaktadır. Modelin eğitilmesinde, tahminlerin hatasını belirlemek için dâhili bir çapraz doğrulama (cross-validation) tekniği kullanılmaktadır. Test aşamasında, gelen yeni verilerinin sınıfı modeldeki karar ağaçlarının oy çokluğuna göre belirlenmektedir (Breiman, 2001; Belgiu ve Dragut, 2016). Şekil 3.17’de RO sınıflandırıcısında yer alan karar ağaçlarının bir temsili görülmektedir. Şekilde, verinin her alt kümesi için eğitim aşamasında ayrı ayrı oluşturulmuş olan karar ağaçlarının, test aşamasında gelen bir verinin karar ağaçları üzerindeki uygun olan sınıfa yönlendirilmesi görülmektedir. Son aşamada, karar ağaçları tarafından yapılan sınıf oylamasında, tanımlı her sınıf için o sınıf için karar ağaçlarından çıkan oy sayısına göre bir olasılık değeri alınmaktadır ve gelen veri son karar olarak en çok olasılık değerine sahip olan sınıfa atanmaktadır.



Şekil 3.17. RO sınıflandırıcısının çalışma yönteminin ve içerisinde yer alan karar ağaçlarının temsili

3.3. Nicel başarı ölçümleri

Doğruluk (accuracy) ve $F1$ skor ($F1$ score) ölçümleri veri sınıflandırma işlemlerinde başarı ölçütü olarak çok yaygın olarak kullanılmaktadır. Bu iki ölçüm de kategorilenen (etiketlenen) veriler ile gerçek (olması gereken) etiketlerin karşılaştırılması üzerine başarıyı ölçmektedir. Sınıflandırma işleminde, tanımlanan kategorilere özel etiketler atandığı için bu ölçümler kolayca yapılabilirdiği gibi, etiket (karşılıklı olarak farklı

etiketleri) eşitleme gibi yöntemler kullanılarak kümeleme ve segmentasyon işlemlerinde de kullanılabilir (Awrangjeb ve Fraser, 2014; Vo ve ark., 2015; Sağlam ve Baykan, 2019).

3.3.1. Doğruluk (accuracy)

“Doğruluk” (accuracy) ölçümü, kategorilenen (etiketlenen) bir verideki, doğru etiketlenen verilerin toplam test edilen veriye olan oranını göstermektedir (Sokolova ve Lapalme, 2009). Doğruluk değeri hesaplanması, bütün veri bazında yapıldığı için kategoriler, içerdiği eleman sayısına göre sonucu daha fazla etkilemektedir. Eğer yapılan işlemde, çok elemana sahip kategoriler (sınıf, küme, segment vb.) uygulamada daha fazla öneme sahip ise bu ölçüm uygun bir yöntemdir. Eğer, her kategori eşit öneme sahipse, az sayıda elemana sahip kategoriler sonuca az etki edeceğinden, diğer bir ifade ile çok sayıda elemana sahip kategoriler az sayıda olanları bastıracağından, bu ölçüm amaca uygun olmayacaktır. Denklem 3.1’de doğruluk hesaplamasının basitçe hesaplanması görülmektedir. Doğruluk değeri [0-1] aralığında yer almakta ve değer yüksek olması başarının da yüksek olduğunu ifade etmektedir. İstenirse [0-100] aralığına ölçeklenerek, başarı % (yüzde) olarak da ifade edilebilmektedir.

$$\text{Doğruluk} = \frac{\text{Doğru etiketlenen eleman sayısı}}{\text{Test edilen eleman sayısı}} \quad (3.1)$$

3.3.2. F1 skor (F1 score)

“F1 skor” (F1 score) ölçümü her kategori için ayrı ayrı yapılan bir ölçümdür. “Tahmin” (precision) ve “geri çağırma” (recall) ölçümlerini kullanmaktadır (Van Rijsbergen, 1979; Sağlam ve Baykan, 2019). Örneğin bir kategorinin etiketi A ise, A kategorisi için hesaplanan tahmin (precision) ölçümü Denklem 3.2’de hesaplandığı şekilde, A kategorisinde olup A olarak doğru etiketlenen eleman sayısının A olarak etiketlenen (A kategorisinden olduğu tahmin edilen) toplam eleman sayısına oranıdır. A kategorisi için hesaplanan geri çağırma (recall) ölçümü ise Denklem 3.3’te hesaplandığı şekliyle, A kategorisindeki doğru etiketlenen eleman sayısının A kategorisindeki toplam eleman sayısına oranıdır. A kategorisi için F1 skor başarı değeri de Denklem 3.4’teki gibi tahmin ve geri çağırma değerlerinin harmonik ortalaması alınarak hesaplanmaktadır.

Tahmin, geri çağırma ve $F1$ skor değerleri [0-1] aralığında yer almakta ve değerlerin yüksek olması başarının da yüksek olduğunu ifade etmektedir.

$$A_{\text{Tahmin}} = \frac{\text{A kategorisindeki doğru (A olarak) etiketlenen eleman sayısı}}{\text{A olarak etiketlenen toplam eleman sayısı}} \quad (3.2)$$

$$A_{\text{Geri çağırma}} = \frac{\text{A kategorisindeki doğru (A olarak) etiketlenen eleman sayısı}}{\text{A kategorisindeki toplam eleman sayısı}} \quad (3.3)$$

$$A_{F1 \text{ skor}} = 2 \times \frac{A_{\text{Tahmin}} \times A_{\text{Geri çağırma}}}{A_{\text{Tahmin}} + A_{\text{Geri çağırma}}} \quad (3.4)$$

Kategorilenen verinin geneli hakkında bir $F1$ skor değeri almak gerekirse, bütün kategorilerin $F1$ skor değerinin ortalaması alınabilmekte veya literatürde sıklıkla kullanıldığı gibi bütün kategorilerin tahmin değerlerinin ortalaması ve geri çağırma değerlerinin ortalamasının Denklem 3.5'teki gibi harmonik ortalaması alınarak "Genel $F1$ skor" (overall $F1$ score – OF1) değeri hesaplanabilmektedir.

$$OF1 = 2 \times \frac{\text{Ortalama tahmin} \times \text{Ortalama geri çağırma}}{\text{Ortalama tahmin} + \text{Ortalama geri çağırma}} \quad (3.5)$$

OF1 ölçümünde bütün kategorilerin tahmin ve geri çağırma ölçüm değerleri hesaplamaya eşit olarak etki yaptıkları için, bir kategorileme (sınıflandırma, kümeleme, segmentasyon vb.) işleminde eğer küçük kategoriler de büyük kategoriler kadar öneme sahip ise OF1 ölçümünün kullanılması doğruluk ölçümüne kıyasla daha uygundur. Tez içerisinde, bir verinin geneli üzerindeki $F1$ skor değeri verildiğinde aslında o verinin OF1 değeri kastedilmektedir.

3.4. Veri Setleri

3.4.1. Tez kapsamında hazırlanan segmentasyon veri setleri

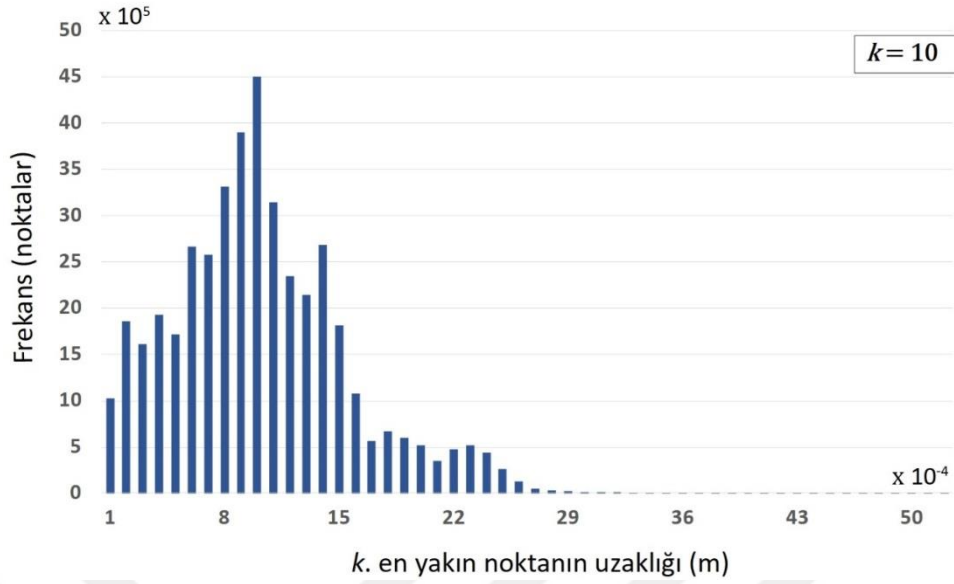
Geliştirilen yöntemin başarısını doğrulamak ve segmentasyon sonuçlarının literatürle karşılaştırılmasının yapılması için tez kapsamında üç adet nokta bulutu veri seti alınmıştır. Bu amaçla, bir FARO® Laser Scanner Focus3D X 330 sistemi kullanılarak birisi iç ve ikisi dış mekân olmak üzere üç farklı mekânda tarama gerçekleştirilmiş ve her

mekâna ait 3D nokta bulutu verileri elde edilmiştir (Info, 2015). Sistemin nokta tarama yoğunluğu parametresi olarak, her mekân için 10 m mesafeden 1 m² alan içerisinde 27,777 (27 bin 777) nokta (0.006 m aralıkla) (6 mm) tarayabilecek yoğunluk değeri ayarlanmış ve 360° tarama sonrasında ham nokta bulutları renkli olarak elde edilmiştir. Segmentasyon veri setlerinin hazırlanması (Vo ve ark., 2015) ve (Xu ve ark., 2018c) çalışmalarındaki karşılaştırma veri setlerinin hazırlanma şekilleri göz önünde bulundurularak hazırlanmıştır. Ham veri üzerinde veri indirgeme ve gürültü giderme ön işlemlerinden sonra referans segmentleri hazırlanacak kısımlar, segmentasyon veri setini hazırlamak üzere ham veri içerisinde çıkarılmıştır. Son olarak da, test edilecek algoritmalarından bağımsız ve manuel olarak belirlenen referans yüzey segmentlerindeki noktalar birbirlerinden ayrıştırılmışlardır. Karşılaştırmada kullanılacak olan referans veri setleri ve her veri setinde yapılan işlemler, ayrı ayrı alt başlıklar halinde aşağıda açıklanmaktadır.

3.4.1.1. Ofis Nesneleri veri seti (Örnek 1)

İç mekân veri seti olarak Konya Teknik Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi'ndeki bir ofiste tarama yapılarak veri alınmıştır. Ofiste, farklı yüzey özelliklerine sahip çeşitli nesneler tarama öncesinde farklı nesne yüzeyleri segmentasyonuna uygun olarak yerleştirilmişlerdir.

Tarama sonrası gelen ham veri üzerinde 43,145,120 (43 milyon 145 bin 120) adet nokta bulunmaktadır. İç mekân verisi için bu nokta sayısı oldukça fazla olduğundan, nokta sayısının azaltılması gerekmiştir. Kullanılan tarama aralığı iç mekân için çok yüksektir ama küçük detayların alınmasını sağlamıştır. İç mekân veri alımı, dış mekân veri alımına göre kısıtlı bir alan içerisinde yapıldığı ve nesneler yakından tarandığı için noktalar çok yüksek yoğunlukta bulunmaktadır. Bunun için tarama sonrasında, iç mekân verisinde çözünürlük azaltmak ve aynı zamanda uzak noktalar ile yakın noktalar arasındaki yoğunluğu eşitlemek için veri indirgeme işlemi uygulanmıştır. İç mekân verisi, farklı şekildeki nesnelerin segmentasyonunun sonuçlarının test edilmesinde kullanılacağı için, yoğunluğu eşitleyerek veriyi doğal nokta bulutundaki noktaların yoğunluklarının düzensizliğinden (tarama cihazına yakın nesnelerdeki noktaların uzak nesnelere göre daha yoğun olması) bağımsız bir şekilde segmentlemek ve işlem süresini düşürmek için yapaylaştırmak amaçlanmıştır. Bunun için ilk önce ham verideki noktaların en yakın k 'inci nokta ile arasındaki mesafelerin Şekil 3.18'deki histogramı çıkarılmıştır.



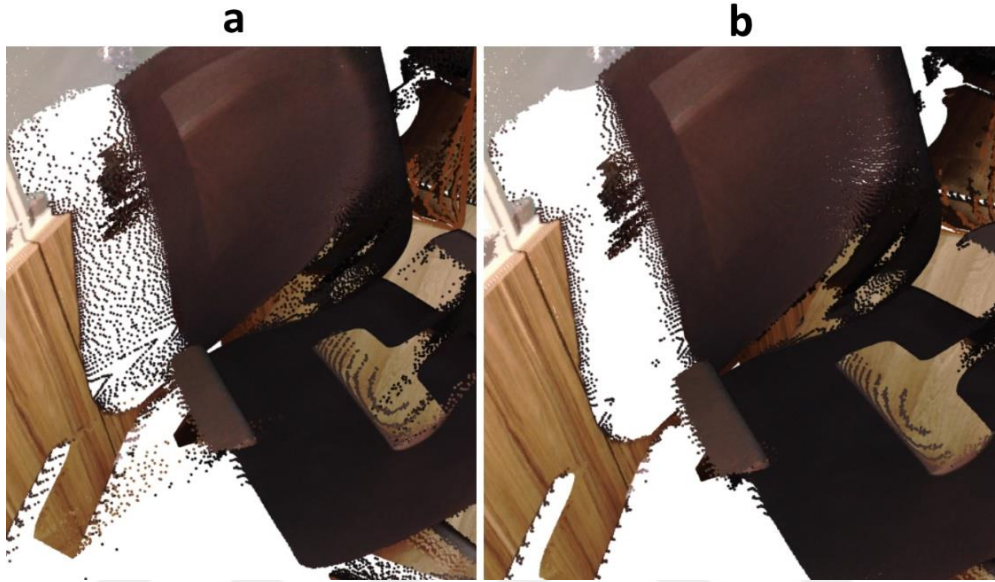
Şekil 3.18. İç mekân ham verisinin k . en yakın nokta uzaklığı histogramı

Şekil 3.18'e bakıldığında noktaların neredeyse tamamının 10. en yakın komşusu ($k = 10$ için) 25×10^{-4} m (2.5 mm) mesafenin altında bir uzaklıkta bulunmaktadır. Bu yüzden kenar uzunluğu 25×10^{-4} m olan vokselleme işlemi yapılmıştır ve vokseller içerisindeki noktaların ortalama değeri tek bir nokta olarak kaydedilmiştir. Böylece veri sayısı 13,700,111'e (13 milyon 700 bin 111) düşürülmüş, uzaktaki ve yakındaki noktaların yoğunlukları büyük oranda eşitlenmiştir.

İç mekân içerisindeki diğer bir sorun ise boşluğa dağılan çok fazla gürültülü verinin bulunmasıdır. İç mekândaki nesneler farklı türlerde malzemeler içerdiğinden ve yakın mesafeden yoğun çözünürlükte tarama yapıldığından, bazı nesneler üretildikleri maddelerin lazeri emmesi nedeniyle, kendisine gelen lazeri geç yansıtmakta ve tarandığı konumu gerçekte olandan farklı bir yer olarak göstermektedir. Bunun için de indirgenmiş veri üzerinde Denklem 3.6'daki koşulu sağlayan noktalar gürültü nokta olarak değerlendirilerek, indirgenmiş veriden çıkarılmıştır. Denklemde d_k ilgili noktanın en yakın k . noktaya olan uzaklığını, $d_{k_{ort}}$ ise bütün noktaların d_k değerinin ortalamasını ifade etmektedir. Bu işlemde k değeri literatürdeki bazı çalışmalarda tercih edildiği gibi 10 olarak uygulanmıştır (Vo ve ark., 2015; Xu ve ark., 2018c). Gürültü gidermenin derecesini ayarlayan c_g sabiti ise 3 olarak uygulanmıştır. Bu işlemin sonucunda da veri setindeki nokta sayısı 13,656,534'e (13 milyon 656 bin 534) düşmüştür. Şekil 3.19 (a)'da örnek bir gürültülü veri ve Şekil 3.19 (b)'de ise gürültü giderme işlemi uygulanmış hali

görülmektedir. Şekil 3.20’de ise iç mekân verisinin veri indirgeme ve gürültü giderme aşamalarından sonraki hali iki farklı açıdan gösterilmektedir.

$$d_k > c_g \cdot d_{k_{ort}} \quad (3.6)$$



Şekil 3.19. (a) Örnek bir gürültülü veri ve (b) gürültü giderme işlemi uygulanmış hali

Son olarak, segmentasyon işlemlerinin nicel başarı ve karşılaştırma sonuçlarını alabilmek için, iç mekân verisinden nesnelerin yer aldığı kısım alınmış ve veri seti için gereksiz kısımlar manuel olarak temizlenmiştir. İç mekân verisinin son halinden alınan ve temizlenen kısım, “Ofis Nesneleri” (Örnek 1) olarak isimlendirilen segmentasyon veri setinin “orijinal” kısmını oluşturmakta ve 382,217 (382 bin 217) nokta içermektedir. Daha sonra, orijinal veride, yüzeyleri manuel olarak birbirinden bağımsız 26 segmente ayrılmış nesneleri içeren, veri setinin “referans” kısmı oluşturulmuştur. Şekil 3.21 (a)’da Ofis Nesneleri veri setinin orijinal verisi ve Şekil 3.21 (b)’de ise referans verisi gösterilmektedir.

Lokal Nokta Yoğunluğu (LNY) (Local Points Density - LPD) (noktalar / m²) nokta bulutu verilerinin yoğunluk dağılımlarını gösteren, literatürde yaygın olarak kullanılan bir ölçüm yöntemidir (Lari ve Habib, 2014; Vo ve ark., 2015; Xu ve ark., 2018c). Noktaların LNY dağılımı, nokta işleme algoritmalarında voksel büyüklüğü belirleme gibi işlemlerde veri hakkında fikir vermektedir. Bir noktanın LNY değeri Denklem 3.7’deki gibi ölçülmektedir. Denklemdaki k değişkeni komşu nokta sayısını

temsil ederken, d_k^2 bütün verideki noktaların en uzak komşularının ortalamasını temsil etmektedir.



Şekil 3.20. İç mekân verisinin veri indirgeme ve gürültü giderme aşamalarından sonraki hali

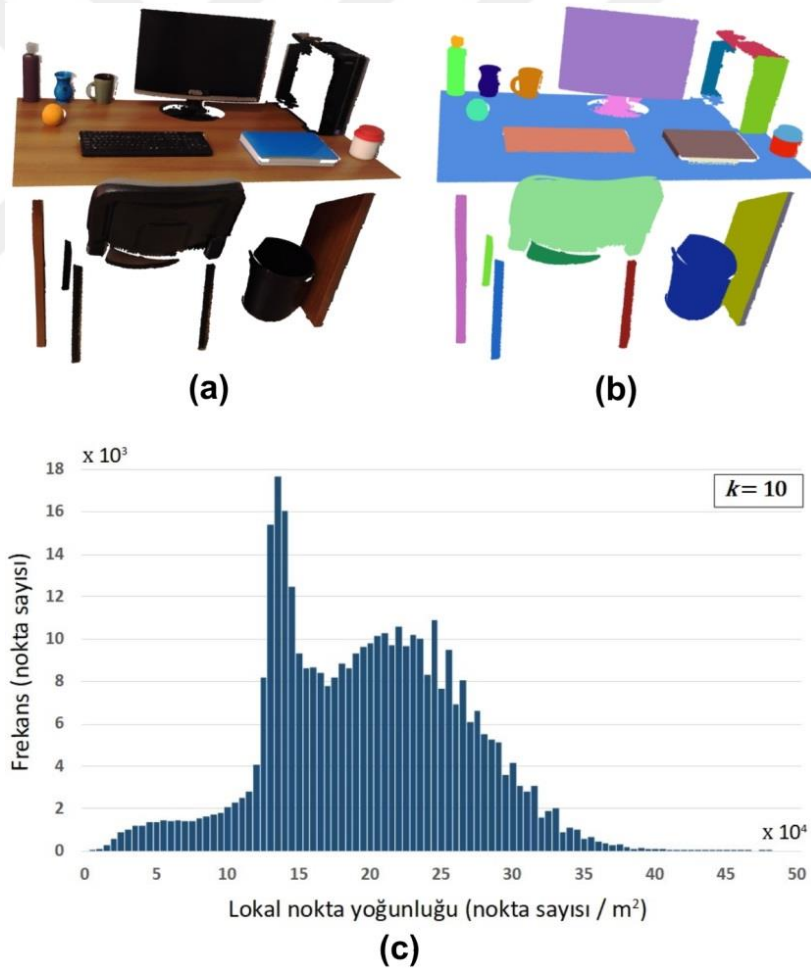
$$\text{LNY} = \frac{k}{\pi \cdot d_k^2} \quad (3.7)$$

Lokal nokta yoğunluğu (LNY), bir nokta bulutu ile ilgili istatistiksel bilgiler sağlayabilmektedir (Lari ve Habib, 2014; Vo ve ark., 2015; Xu ve ark., 2018c). Veriye göre değişiklik gösteren bazı nokta bulutu işleme algoritmaları için kullanıcıların voksel boyutu, komşu nokta sayısı (k), komşuluk yarıçapı ve süper vokselizasyon vb. işlemler

için beslenme alanı (belirlenecek büyüme noktalarının maksimum büyüme alanı) gibi bazı parametreleri belirleyebilmesine yardımcı olmaktadır.

3D uzayda bir düzlem yönü verebilmesi için, bir vokselin içerisindeki noktaların sayısının en az üç olması gerekmektedir. Bu bağlamda, voksel büyüklüğü neredeyse tüm voksellerin en az üç noktaya sahip olmasını sağlayacak kadar büyük, nokta bulutundaki gerekli ayrıntıları yakalayacak kadar da küçük olmalıdır.

Örnek 1 veri seti için voksel boyutu (voksel kenar uzunluğu) 0.015 m (15 mm) olarak belirlenirse, 1 m² alan içerisinde yaklaşık olarak 0.43×10^4 (4300) adet voksel yer alabilmektedir. Her vokselde en az üç nokta olması gerektiği dikkate alındığında, 0.43×10^4 değeri 3 ile çarpıldığında noktaların LNY değeri en az yaklaşık olarak 1.3×10^4 (13000) olmalıdır. Şekil 3.21 (c)'ye bakıldığında, hemen hemen tüm noktaların LNY değerleri 1.3×10^4 (13000) değerinden büyüktür.



Şekil 3.21. Ofis Nesneleri (Örnek 1) veri setinin orijinal verisi (a), referans verisi (b) ve LNY dağıtımı (c)

3.4.1.2. Cami veri seti (Örnek 2)

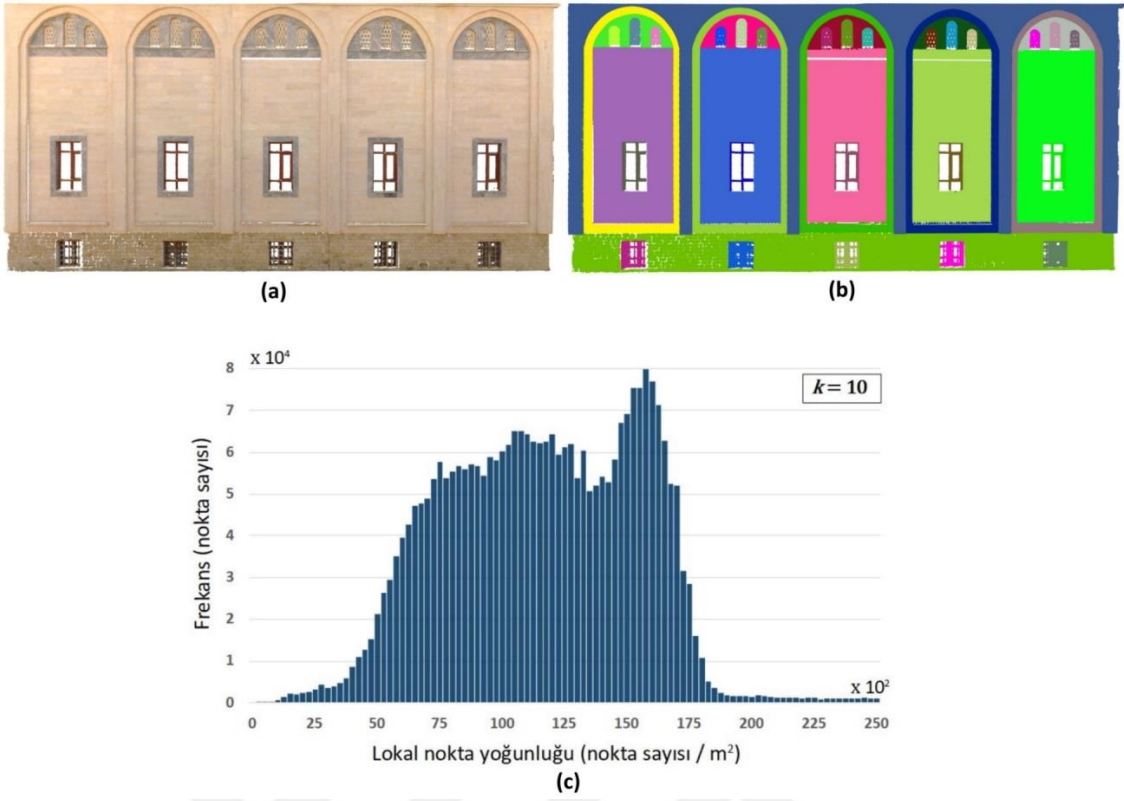
Dış mekân veri setlerinden ilki olan “Cami” nokta bulutu, Selçuk Üniversitesi Kampüs Camii'nin arka cephe taramasından elde edilmiştir. Bu veri seti, yöntemlerin gerçek hayattaki uygulamalardaki verimliliğini görmek için taramadan gelen doğal halinde bırakılmıştır, yani iç mekân verisinden farklı olarak veri indirgeme ve gürültü giderme işlemleri yapılmamıştır. Verinin taramadan gelen ham hali 24,115,659 (24 milyon 115 bin 659) noktadan oluşmakta ve Şekil 3.22’de gösterilmektedir.



Şekil 3.22. Cami nokta bulutunun taramadan gelen ham hali

Alınan “Cami” nokta bulutu ham verisinden segmentasyon veri seti için önemsiz küçük parçalar manuel olarak temizlenmiştir. Böylelikle nokta sayısı 3,069,150 (3 milyon 069 bin 150) olan Cami (Örnek 2) veri setinin projede kullanılacak olan orijinal verisi elde edilmiştir. Orijinal veri (a), orijinal veriye ait 42 segmentten oluşan referans veri (b) ve orijinal verinin LNY dağılımı (c) Şekil 3.23’de gösterilmektedir.

Örnek 2 veri seti için vokselle boyutu 0.03 m (30 mm) olarak belirlenirse, 1 m² alan içerisinde yaklaşık olarak 11×10^2 (1100) adet vokselle yer alabilmektedir. Her vokselde en az üç nokta olması gerektiği dikkate alındığında, noktaların LNY değeri en az 33×10^2 (3300) olmalıdır. Şekil 3.23 (c)’ye bakıldığında, hemen hemen tüm noktaların LNY değerleri 33×10^2 (3300) değerinden büyüktür.



Şekil 3.23. Cami (Örnek 2) veri setinin orijinal verisi (a), referans verisi (b) ve LNY dağıtımı (c)

3.4.1.3. Tren Garı veri seti (Örnek 3)

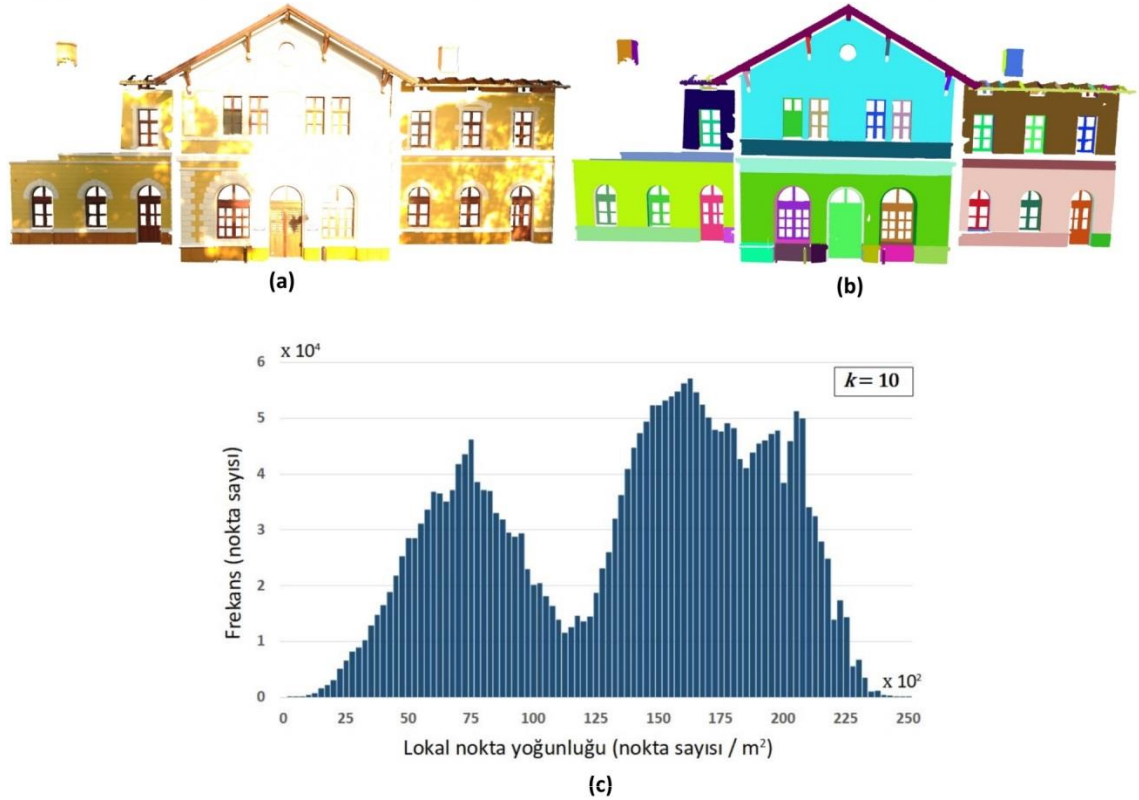
Dış mekân veri setlerinden ikinci veri seti “Tren Garı” nokta bulutu, Konya iline ait eski tren garının ön cephesinden alınmıştır ve Cami nokta bulutunda olduğu gibi hiçbir indirgeme ve gürültü giderme işlemi yapılmadan doğal olarak bırakılmıştır. Verinin taramadan gelen ham hali 6,111,144 (6 milyon 111 bin 144) noktadan oluşmakta ve Şekil 3.24’te gösterilmektedir.



Şekil 3.24. Tren Garı nokta bulutunun taramadan gelen ham hali

Cami veri setindeki gibi Tren Garı ham veri setinden de segmentasyon veri seti için önemsiz küçük parçalar manuel olarak temizlenmiştir. Böylelikle Tren Garı (Örnek 3) orijinal veri setindeki nokta sayısı 2,742,237 (2 milyon 742 bin 237) olarak elde edilmiştir. Orijinal veri (a), orijinal veriye ait 106 segmentten oluşan referans veri (b) ve orijinal verinin LNY dağılımı (c) Şekil 3.25'te gösterilmektedir.

Örnek 3 veri seti için vokselle boyutu 0.045 m (45 mm) olarak belirlenirse, 1 m² alan içerisinde yaklaşık olarak 5×10^2 (500) adet vokselle yer alabilmektedir. Her vokselde en az üç nokta olması gerektiği dikkate alındığında, noktaların LNY değeri en az 15×10^2 (1500) olmalıdır. Şekil 3.25 (c)'ye bakıldığında, hemen hemen tüm noktaların LNY değerleri 15×10^2 (1500) değerinden büyüktür.

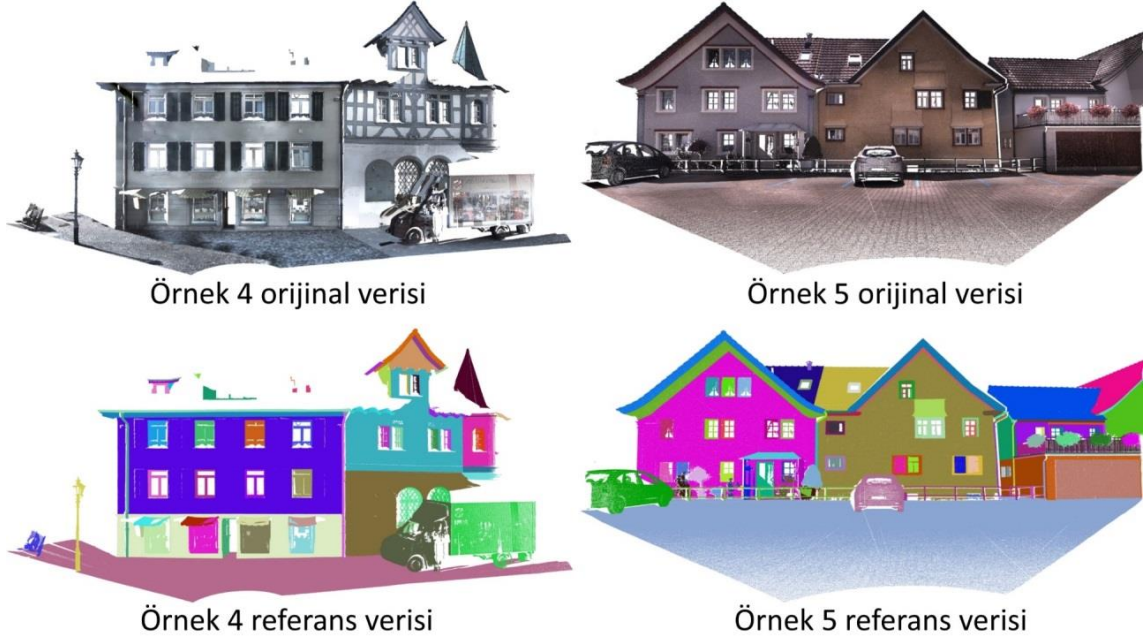


Şekil 3.25. Tren Garı (Örnek 3) veri setinin orijinal verisi (a), referans verisi (b) ve LNY dağılımı (c)

3.4.2. Literatürden alınan segmentasyon veri setleri

Tez kapsamında alınan segmentasyon veri setlerine ek olarak iki adet ek segmentasyon veri seti de (Şekil 3.26'da görülen Örnek 4 ve Örnek 5) literatürden temin

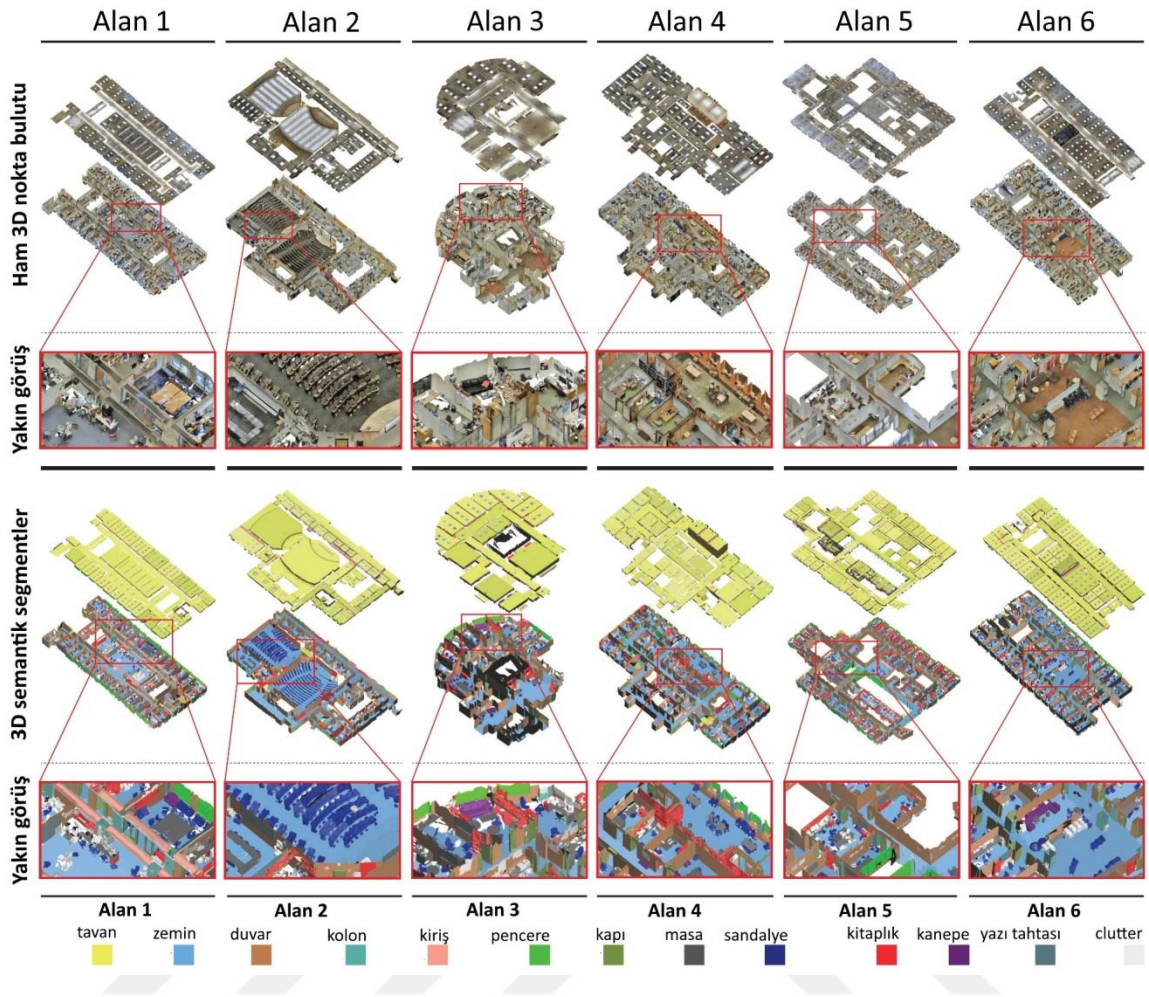
edilerek test amaçlı kullanılmıştır (Xu ve ark., 2018a; Xu ve ark., 2018c). Örnek 4 ve Örnek 5'in segment sayısı sırasıyla 66 ve 101'dir.



Şekil 3.26. Literatürden alınan Örnek 4 ve Örnek 5 segmentasyon veri setleri

3.4.3. Literatürden alınan sınıflandırma veri seti (S3DIS)

Literatürden temin edilen “The Stanford Large-Scale 3D Indoor Spaces (S3DIS) - Semantik (anlamsal) Segmentasyon Veri Seti”, referans verilere de sahip olan Alan1, Alan2, Alan3, Alan4, Alan5 ve Alan6 olarak adlandırılmış olan toplam 6 adet iç mekân nokta bulutundan oluşmaktadır (Armeni ve ark., 2016). Veri setindeki noktalar konumsal değerlere (x , y ve z) (XYZ) ve renk değerlerine (kırmızı, yeşil ve mavi) (RGB) sahiptir. Veri setinin genel bir görünümü (ham noktalar ve sınıf etiketlerine göre renklendirilmiş referans semantik segmentleri) Şekil 3.27’de ve alanlara göre nokta sayıları Çizelge 3.1’de listelenmiştir. Çizelge 3.2’de ise alanlara ve sınıflara göre veride yer alan semantik segment sayıları listelenmektedir.



Şekil 3.27. Literatürden alınan S3DIS sınıflandırma veri seti

Çizelge 3.1. S3DIS veri seti nokta sayıları

	Alan 1	Alan 2	Alan 3	Alan 4	Alan 5	Alan 6
Nokta sayısı	44,026,810	47,294,447	18,662,173	43,682,507	78,649,818	41,353,055

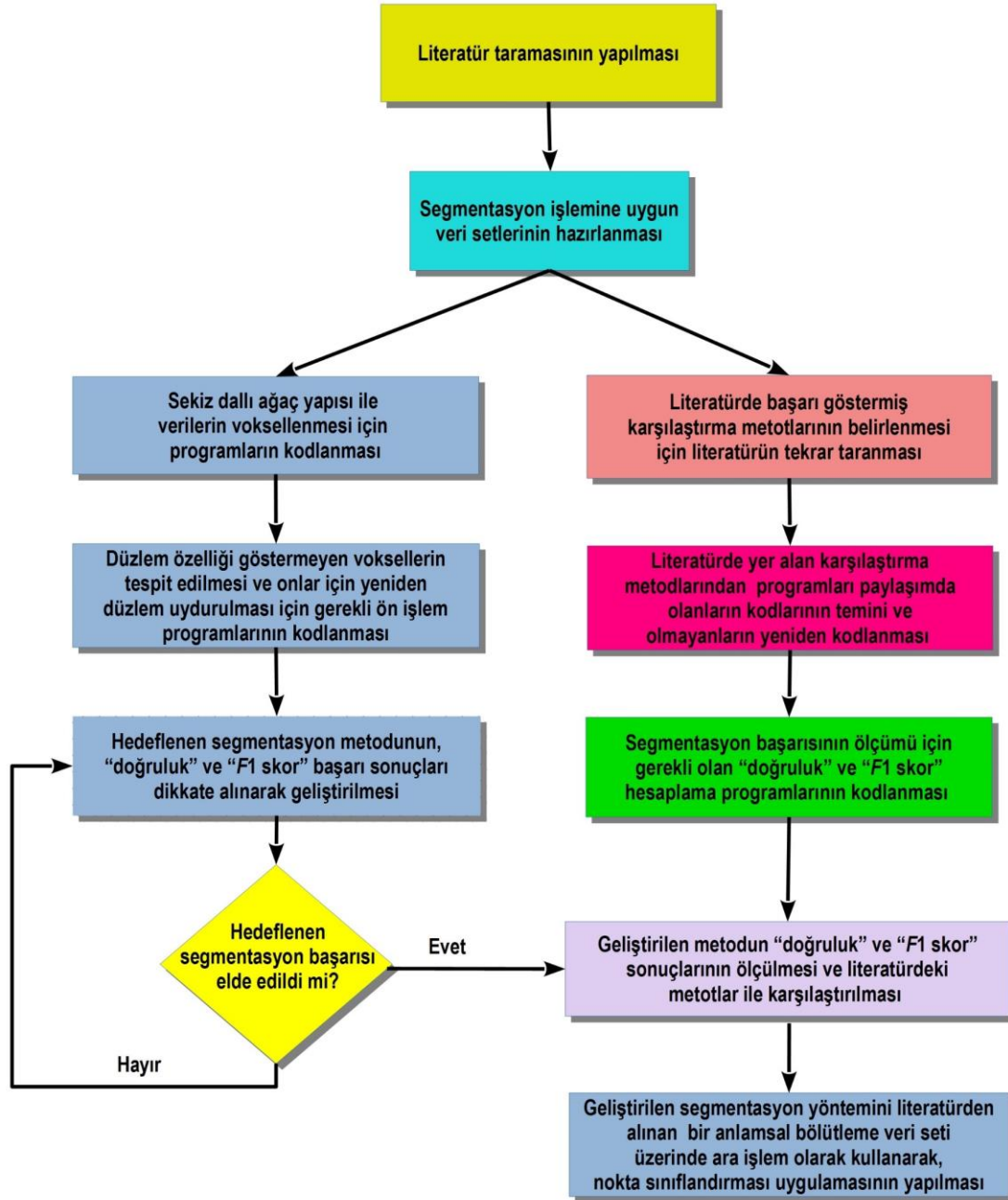
Çizelge 3.2. S3DIS veri setindeki sınıflara göre semantik segment sayıları

Alan	Zemin	Tavan	Duvar	Kiriş	Kolon	Penc.	Kapı	Sand.	Masa	Kitp.	Kanp.	Yazı Taht.	Toplam
1	56	45	235	62	58	30	87	156	70	91	7	28	925
2	82	51	284	12	20	9	94	546	47	49	7	18	1,219
3	38	24	160	14	13	9	38	68	31	42	10	13	460
4	74	51	281	4	39	41	108	160	80	99	15	11	963
5	77	69	344	4	75	53	128	259	155	218	12	43	1,437
6	64	50	248	69	55	32	94	180	78	91	10	30	1,001
Toplam	391	290	1,552	165	260	174	549	1,369	461	590	61	143	6,005

4. GELİŞTİRİLEN YÖNTEM

4.1. Geliştirilen Segmentasyon Yöntemi

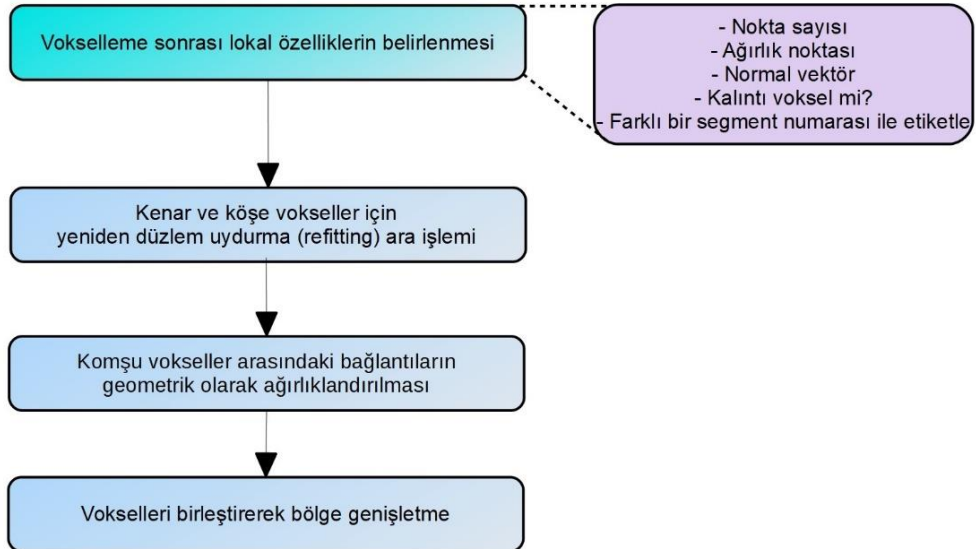
Tez kapsamında yapılan çalışmalar, geliştirilen segmentasyon yönteminin geliştirme aşamaları ile birlikte akış şeması halinde Şekil 4.1’de görülmektedir.



Şekil 4.1. Tez kapsamında yapılan çalışmaların akış şeması

4.1.1. Vokselleme sonrası lokal özelliklerin belirlenmesi

Veri setlerinin hazırlanmasında olduğu gibi, geliştirilen metodun çalışmasında da ön işlem olarak vokselleme işlemi gerekmektedir. Çünkü geliştirilen metod vokseller üzerinde işlem yapmakta ve segmentasyon işlemi yaparken aslında vokselleri segmentlemektedir. Vokselleme işlemi sırasında veya sonrasında, voksellerin içerisindeki noktalar hakkında nokta sayısı, ağırlık noktası, normal vektör (eğim yönü) ve ortalama renk değerleri gibi bazı nitelikler hesaplanmaktadır. Vokselleme aşamasında, Bölüm 3.1.2’de belirtildiği gibi octree veri yapısında her oktanın içerisinde kalan noktaların bellekteki başlangıç ve bitiş adresleri tutulmaktadır. Octree organizasyonundaki bölünmenin en uç noktasındaki oktanlar (leaf) olan vokseller de bu şekilde hem içerisindeki noktaların sayısını tutmakta hem de o noktalara ulaşabilmektedir. Bu sayede vokselleme aşamasında, bir voksel tespit edildiğinde içerisindeki noktaların sayısı, ağırlık merkezi, TBA hesaplaması ile elde edilen normal vektör dâhil temel bileşenleri ve ortalama renk değerleri de hesaplanarak, vokseli temsil eden veri yapısı içerisinde tutulabilmektedir. Şekil 4.2’de, vokselleme sonrası lokal özelliklerin belirlenmesi aşamasının, geliştirilen metodun ana aşamalarının içerisindeki yeri şema halinde görülmektedir.

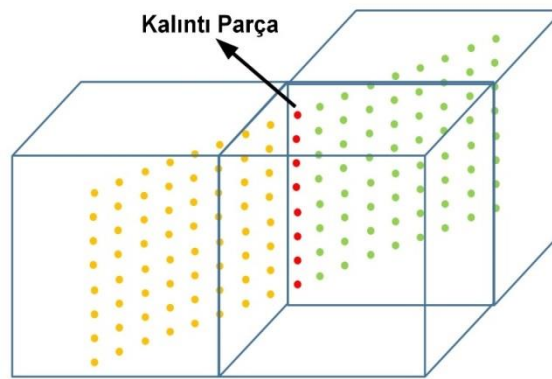


Şekil 4.2. Lokal özelliklerin belirlenmesi aşamasının geliştirilen metodun ana aşamalarının içerisindeki yeri

Bir nokta grubundan düzlem bilgisi elde edebilmek için o nokta grubunda en az üç adet nokta bulunması gerekmektedir. Nokta gruplarını temsil eden voksellerden üç

adetten daha az nokta sayısına sahip olan vokseller “kalıntı” (residual) olarak isimlendirilmektedir (Wang ve Tseng, 2011; Vo ve ark., 2015).

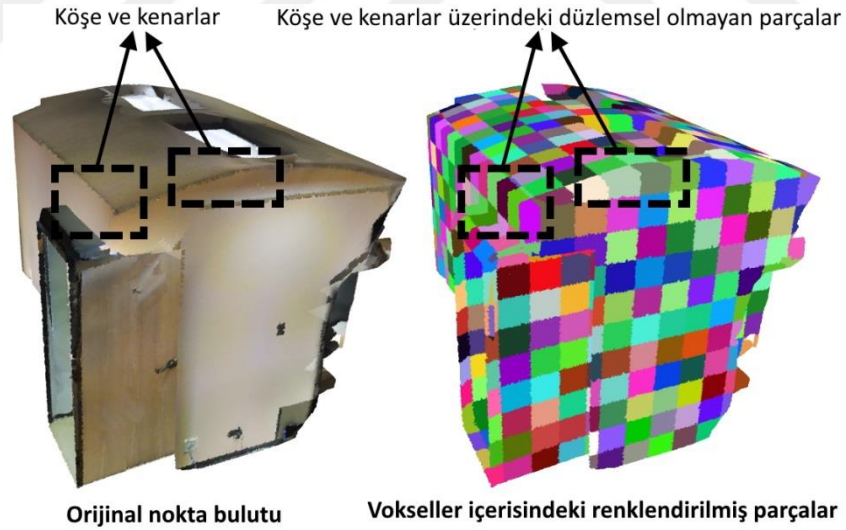
Vokselleme aşamasında, bazı vokseller içerisindeki nokta sayısı üç veya daha fazla olmasına rağmen düzlem bilgisi verememektedir. Bunun nedeni, voksel içerisinde kalan noktaların Şekil 4.3’te görüldüğü gibi bir doğru üzerinde arka arkaya dizilmiş olmalarıdır. Bu şekilde nokta dizilimine sahip vokseller de kalıntı voksel olarak değerlendirilmektedir. Voksellerin bu şekilde doğrusal bir özellik gösterip göstermediği ise TBA hesaplanması sonrasında ortanca öz değer ile en büyük öz değer arasındaki orana bakılarak tespit edilebilmektedir. Voksel yapıları eşit kenarlara sahip olduğundan voksel içerisinde kalan düzlemsel özelliğe sahip bir lokal yüzeyin de karesel bir yapıya sahip olması beklenmektedir. Yani 3D uzaydaki üç öz değerden en büyük ile ortancanın öz değerlerinin birbirine yakın olması beklenmektedir. Bu bağlamda ortanca öz değer en büyük öz değere oranı 1’e yaklaştıkça yapı kare yapısına sahip tam bir düzlemi ifade etmektedir. Bu oran 0’a yaklaştığında ise yapı bir çizgiyi temsil etmektedir ve bu durumda yapı belirli bir düzlem yönüne sahip olmayacaktır. Bu yapılar tespit edilirken (ortanca özdeğer / en büyük özdeğer) oranı 0’a yakın olan voksellerin belirlenmesi gerekmektedir. Bu yakınlığı belirleyen eşik değeri olarak da, tez kapsamında yapılan çalışmalarda kabaca 0.2 (2/10) değeri kullanılmış ve bu değer altında kalan vokseller “kalıntı” olarak etiketlenmiştir. Eşik değeri olarak 0.2 kullanılmıştır; çünkü eşik değeri için 0 ya da $[0, 0.2]$ aralığında kullanıldığında taramalardan alınan noktalarındaki olabilecek hafif dağılımdan dolayı kalıntı vokseller tespit edilemeyebilmektedir. Bu işlem, segmentasyon sonucunu çok etkilemediği gibi eşik değer belirlenmesindeki hassas değer farklılıkları da sonucu neredeyse hiç etkilemeyecektir. Bu yüzden eşik değeri için 0’a yakın yaklaşık bir değer belirlenmesi yeterli olmaktadır.



Şekil 4.3. Voksel içerisinde doğrusal dizilim gösteren kalıntı parça

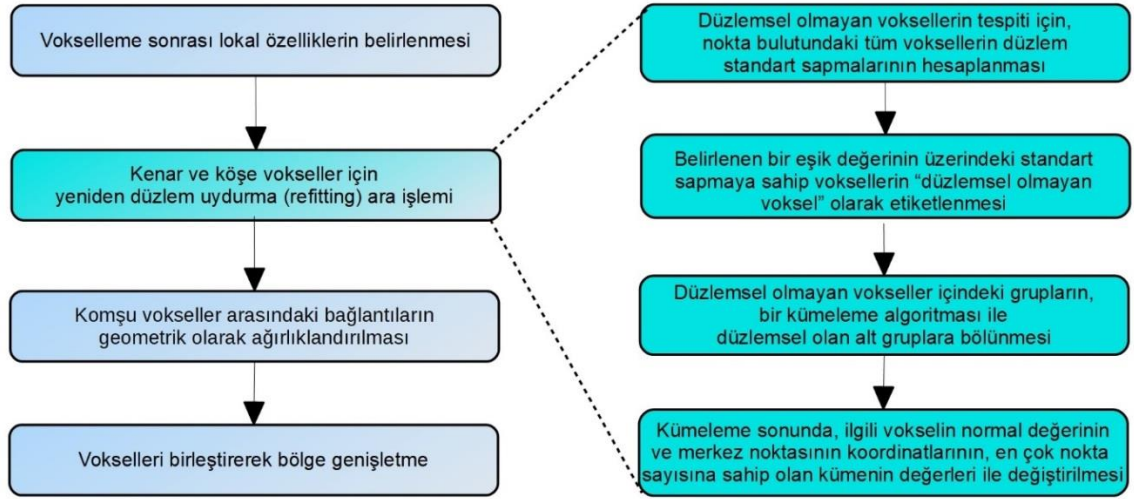
4.1.2. Kenar ve köşe vokselleri için yeniden düzlem uydurma (refitting)

Tez kapsamında geliştirilen segmentasyon yöntemi vokselleme tabanlı bir yöntemdir ve segmentasyon işlemi için vokseller içerisindeki nokta gruplarının (parçaların) düzlemsel eğim özelliklerini kullanmaktadır. Ancak vokselleme işlemi, bütün veriyi kapsayan kübik şeklindeki hücrenin, hiyerarşik olarak eşit büyüklükteki alt hücrelere bölünmesi ile yapıldığından; vokselleme aşamasından sonra, nokta bulutu verisinde nesnelerin kenar ve köşe kısımlarını içine almış olan vokseller olabilmektedir. Bu voksellerde bulunan ve 3D uzayda düzlem özelliği göstermeyen (düzlemsel olmayan) nesnelerin kenar ve köşe kısımlarına ait parçalardan dolayı sorunlar ortaya çıkabilmektedir. Bu düzlemsel olmayan parçalar, farklı düzlemlere ait nokta gruplarından oluşmaktadır. Bu durum o parçaların gerçekte olmayan bir düzlem özelliği göstermesini sağlamakta ve vokselleme tabanlı bir segmentasyon yönteminin sonucunu etkileyerek, yanlış sonuçlar elde edilmesine sebep olabilmektedir. Böyle bir soruna örnek olarak, literatürden alınmış bir nokta bulutu verisi Şekil 4.4'te S3DIS kütüphanesindeki (Armeni ve ark., 2016) bir ofisi temsil eden nokta bulutu üzerinde gösterilmektedir.



Şekil 4.4. Köşe ve kenarlardaki düzlemsel olmayan parça örnekleri

Yapılan tezde, bu sorunun çözümü için önce düzlemsel olmayan parçaların tespit edilmesini ve daha sonra bu parçaların belirli bir düzleme ait özellik kazanmasını sağlayan bir “yeniden düzlem uydurma” (refitting) yöntemi geliştirilmiştir. Refitting ara işleminin geliştirilen yöntemin ana aşamalarının içerisindeki yeri Şekil 4.5'te şema halinde gösterilmektedir.



Şekil 4.5. Kenar ve köşe vokseller için refitting işleminin geliştirilen metodun ana aşamalarının içerisindeki yeri

“Yeniden düzlem uydurma” yönteminin ilk aşaması olan düzlemsel olmayan nokta gruplarının tespiti için, nokta bulutundaki tüm grupların (voksellerin içerisinde yer alan nokta gruplarının) normal vektör değerleri alınmıştır. Bunun için, yönü TBA ile belirlenmiş olan normal vektör doğrultusundaki düzlemler üzerinde, nokta gruplarının standart sapmaları kullanılmıştır. Nokta gruplarındaki noktaların, o grup için elde edilen normal vektör doğrultusundaki değerleri hesaplanarak, “düzlem standart sapmaları” hesaplanabilmektedir. Veride bulunan bütün voksellerdeki grupların düzlem standart sapmaları hesaplandıktan sonra, belirlenen bir eşik değerinin (ε) üzerindeki standart sapmaya sahip nokta grupları “düzlemsel olmayan parça” olarak etiketlenmiştir. Bu eşik değeri için kullanılacak uygun parametre veriye göre çok değişkenlik gösterebileceği için, tez kapsamında bu parametrenin belirlenmesi için de bir yöntem geliştirilmiştir. Standart sapma eşik değeri için kullanılan ε değerini parametre olarak kullanmak yerine, ε değerinin belirlenmesini sağlayan $[-1, +1]$ arasında bir değer olan c parametresi kullanılmaktadır. Bunun için de bütün nokta gruplarının düzlem standart sapmalarının ortalaması olan $\bar{\sigma}$ değeri alınmıştır. c parametre değeri 0 olarak verildiğinde, ε eşik değeri $\bar{\sigma}$ ortalama standart sapma değerine eşit olmaktadır (Denklem 4.1). c değeri en alt sınır değeri olarak -1 olarak belirlenirse ε değeri 0 olur (Denklem 4.2) ve bütün parçalar “düzlemsel olmayan parça” olarak etiketlenir. c değeri üst sınır değeri olarak +1 olarak belirlenirse ε değeri Denklem 4.6’daki gibi hesaplanan σ_{max} değerine eşit olur (Denklem 4.3) ve bütün parçalar “düzlemsel parça” olarak etiketlenmektedir. ε eşik değerinin c parametresinin 0’dan küçük ve 0’dan büyük olması durumlarına göre hesaplanması Denklem 4.4 ve 4.5’teki gibi yapılmaktadır. Çalışma kapsamında c parametresi için Şekil

4.6'daki histogramlardan ve deneylerden yola çıkılarak 0.05 (5/100) değeri kullanılmıştır. Yeniden düzlem uydurma işlemi sonunda önemli bir başarı artışı sağlayan Cami ve Tren Garı verileri üzerindeki nokta gruplarının standart sapma dağılımları ve kullanılan c parametre değerinin konumu Şekil 4.6'da histogram üzerinde gösterilmektedir.

$$\varepsilon = \bar{\sigma} \Leftarrow c = 0 \quad (4.1)$$

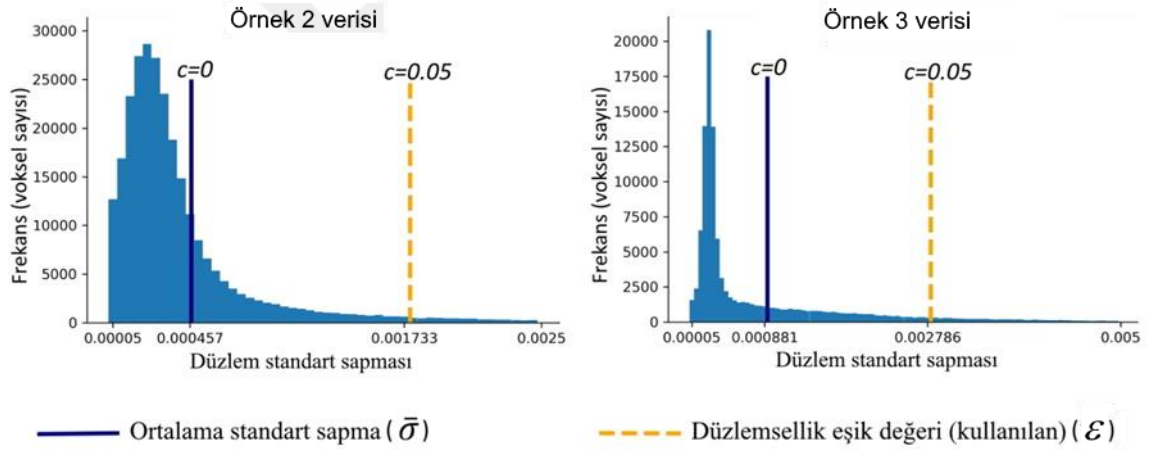
$$\varepsilon = 0 \Leftarrow c = -1 \quad (4.2)$$

$$\varepsilon = \sigma_{max} \Leftarrow c = +1 \quad (4.3)$$

$$\varepsilon = \bar{\sigma} \cdot (1 + c) \Leftarrow c < 0 \quad (4.4)$$

$$\varepsilon = \bar{\sigma} + c \cdot (\sigma_{max} - \bar{\sigma}) \Leftarrow c > 0 \quad (4.5)$$

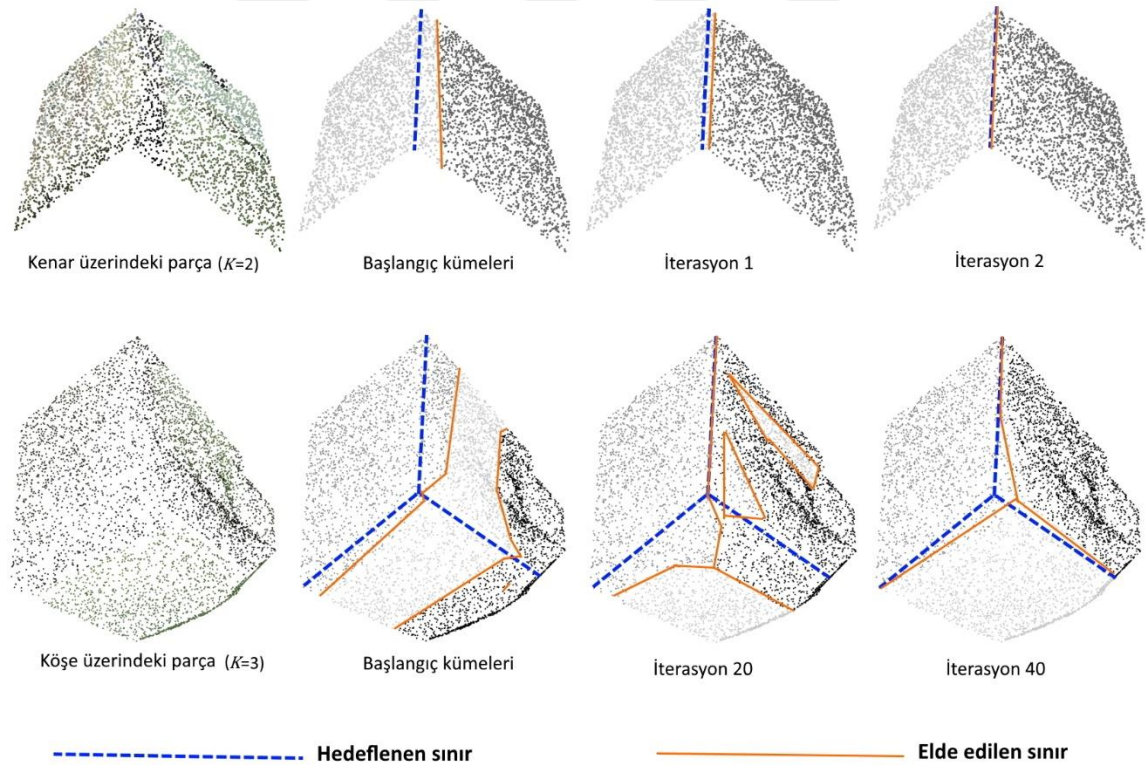
$$\sigma_{max} = \frac{\sqrt{3} \cdot \text{voksel boyutu}}{2} \quad (4.6)$$



Şekil 4.6. Cami ve Tren Garı verilerinin düzlem standart sapma dağılımları

Belirlenen düzlemsel olmayan nokta gruplarının, segmentasyon yöntemi için düzlemsel bir özellik verebilmesi (düzlem standart sapmasının belirlenen eşik değerinin altında olması) için bu nokta grupları, bir kümeleme algoritması ile düzlemsel olan alt parçalara bölünmektedir. Kümeleme algoritması için tez kapsamında K-Ortalamalar (K-Means) algoritması (Jain, 2010) kullanılmıştır. K-Ortalamalar algoritması, parametre olarak küme sayısını önceden belirleyen bir K parametresi kullanmaktadır. Tez kapsamında metot çalıştırılırken, küme sayısı için K parametresi ilk başta 2 olarak belirlenmiştir (kenarlara düşen nokta grupları için). Eğer bu ilk aşamadaki kümeleme sonunda bütün alt parçaların düzlem standart sapmaları verilen eşik değerine eşit veya altına düşmezse yani hala düzlemsel olmayan nokta grubu kaldıysa, K parametresi 3

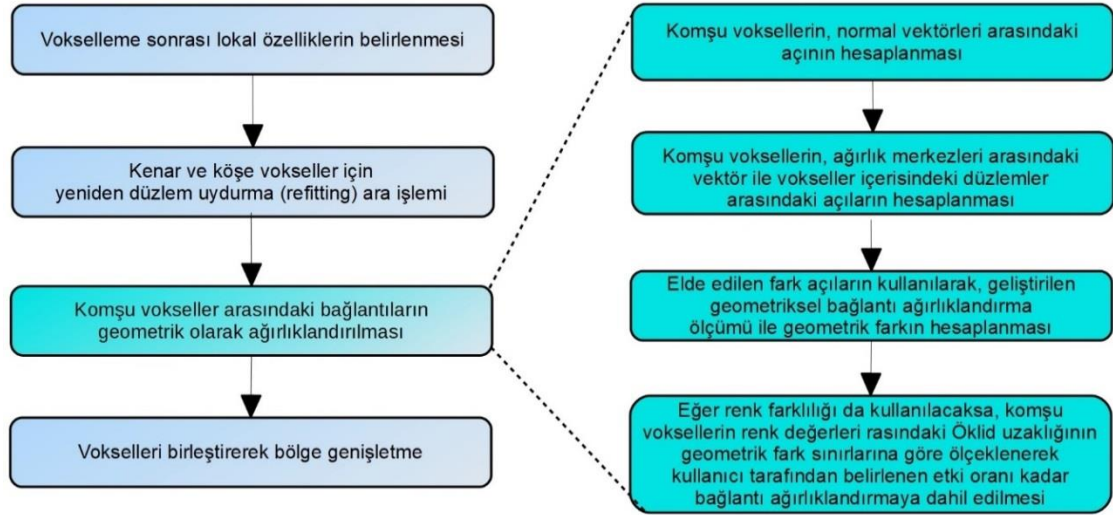
olarak belirlenip (köşelerde kalan parçalar için) yeniden kümeleme yapılmaktadır. K-Ortalama ile kümeleme işleminde, başlangıç kümeleri olarak en yüksek öz değeri veren öz vektör kullanılmıştır. Noktalar bu öz vektör doğrultusundaki konumlarına göre sıralanmış K sayısına eşit bir şekilde bölünerek başlangıç kümeleri oluşturulmuştur. Şekil 4.7’de nokta gruplarının düzlem kümesi için örnekleri görülmektedir. Test aşamalarında, K-Ortalamlar için maksimum iterasyon sayısı 40 olarak belirlenmiş ve her kümeleme işleminde en az 2 iterasyon çalıştırılması (başlangıç kümelerinin ayarlanması hariç) zorunlu kılınmıştır. En son olarak, ilgili vokselin ilk baştaki normal vektör ve ağırlık merkezi koordinat değerleri, kümeleme sonrasındaki en çok nokta sayısına sahip olan kümenin (alt parçanın) değerleri ile değiştirilmektedir. Böylece, bu voksel artık belirli bir nesneye ait olan lokal bir yüzeyi temsil etmektedir. Bununla birlikte, segmentasyonda eğer renk farklılığı da kullanılacaksa, vokselin renk değerleri de (içerisindeki noktaların renk değerlerinin ortalaması) aynı kümeye (alt parçaya) ait olan noktaların renk değerlerinin ortalaması ile değiştirilmektedir.



Şekil 4.7. K-ortalamlar ile düzlem kümeleme örnekleri

4.1.3. Komşu vokseller arasındaki bağlantıların ağırlıklandırılması

Bağlantı ağırlıklandırma aşamasının geliştirilen metodun ana aşamalarının içerisindeki yeri Şekil 4.8’de şema halinde gösterilmektedir.



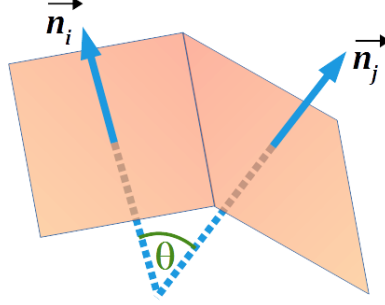
Şekil 4.8. Bağlantı ağırlıklandırma aşamasının geliştirilen metodun ana aşamalarının içerisindeki yeri

Önceki aşamalarda, nokta bulutu verisi voksel yapısına getirilip, bütün voksellerin normal vektörleri hesaplanmış (Bölüm 4.1.1) ve vokseller içinde düzlem olmayan parçalar yeniden düzlem uydurma (refitting) ile düzlemsel parça haline getirilmiştir (Bölüm 4.1.2). Bu aşamada ise önce komşu voksellerin normal vektörlerinin arasındaki ve komşu vokseller için uydurulan teğet vektörü ile düzlemler arasındaki açısal farkları hesaplanmaktadır. Daha sonra, bu açısal farklara göre, komşu vokseller arasındaki bağlantılar ağırlıklandırılmaktadır. Böylece bir sonraki aşamada bu ağırlık değerlerine göre aynı özellik gösteren vokseller birleştirilerek, en iyi segmentasyon yapılması amaçlanmıştır. Bunun için ilk önce aralarındaki açı farkı hesaplanacak normal vektörlerin skaler çarpımları hesaplanmakta ve birim vektöre ölçeklenmektedir (Tseng ve ark., 2007). Denklem 4.7’de $\vec{n}_i$ ve $\vec{n}_j$ normal vektörlerinin arasındaki θ eğim açısının kosinüs değeri elde edilmektedir. Denklemde $|\vec{n}_i|$ ve $|\vec{n}_j|$ sırasıyla $\vec{n}_i$ ve $\vec{n}_j$ normal vektörlerinin skaler olarak büyüklük değerlerini temsil etmektedir.

$$\cos \theta_{ij} = \frac{\vec{n}_i \cdot \vec{n}_j}{|\vec{n}_i| \cdot |\vec{n}_j|} \quad (4.7)$$

Denklem 4.7’den $\cos \theta_{ij}$ değeri elde edildikten sonra Denklem 4.8’deki gibi ters kosinüs hesaplaması ile Şekil 4.9’daki θ eğim açısı hesaplanmaktadır.

$$\theta = \cos^{-1}(\cos \theta_{ij}) \quad (4.8)$$



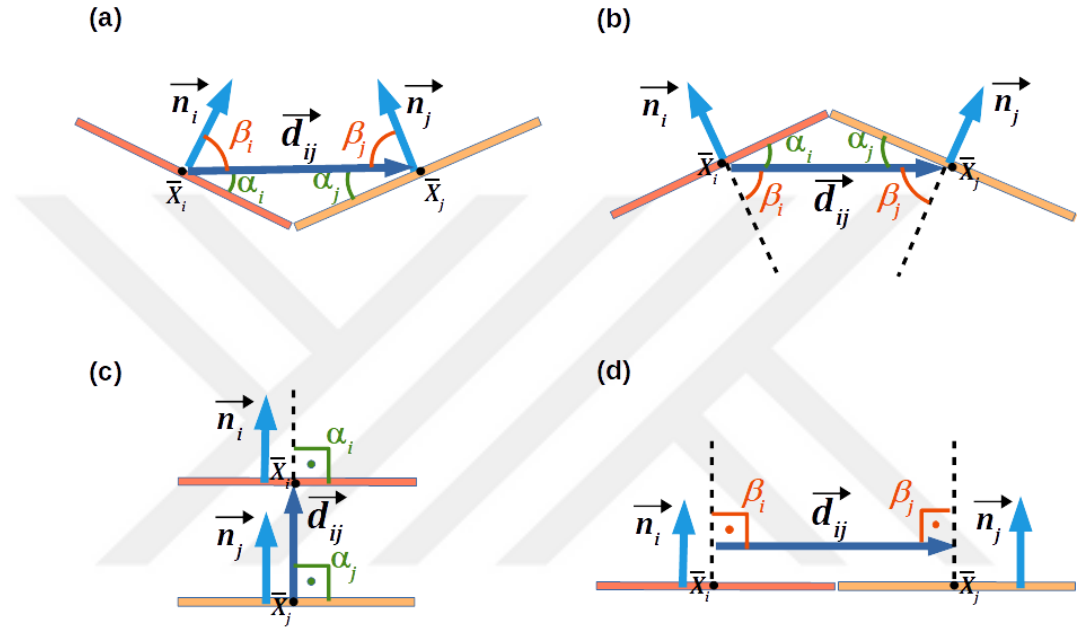
Şekil 4.9. İki normal vektör arasındaki fark açısı

Normal vektörler arasındaki açı iki yüzey eğimi arasında açısal bir fark değeri vermektedir ve belirlenen bir eşik değerine göre sadece bu açı kullanılarak da yüzey farklılıkları üzerinden farklı yöntemlerle segmentasyon yapılabilmektedir. Ancak bu açı, iyi bir segmentasyon işlemi yapılabilmesi için tek başına yeterli olmamaktadır. Çünkü silindirik özellik gösteren ama düzlem açıları farklı komşu düzlemlerin de birlikte gruplanması gerekebilmektedir. Bunun için normal vektör değerleri arasındaki θ açısı (Şekil 4.9’da verilmiş olan) dışında yeni bir özellik çıkarılması gerekmektedir.

Komşu voksellerin içerisindeki nokta gruplarının, merkez noktaları arasındaki yönü belirleyen vektör, literatürde sıklıkla kullanılan bir açısal özellik çıkarma yöntemidir (Verdoja ve ark., 2017; Xu ve ark., 2018b). Birbirine komşu olan v_i ve v_j voksellerinin içerisindeki noktaların merkezleri $\vec{X}_i$ ve $\vec{X}_j$ olarak düşünülürse, $\vec{d}_{ij}$ vektörü de bu merkez noktaları arasındaki birim teğet vektörünü temsil etmektedir. Her iki düzleme ait $\vec{n}_i$ ve $\vec{n}_j$ normal vektörler ile $\vec{d}_{ij}$ merkez vektör arasındaki sırası ile β_i ve β_j açılarının tümler açıları olan sırası ile α_i ve α_j açıları hesaplanmaktadır. Şekil 4.10 (a)’da iç bükey ve Şekil 4.10 (b)’de dış bükey iki düzlem arasındaki β_i ve β_j açıları ve bunların tümler açıları α_i ve α_j açıları görülmektedir. Bu α_i ve α_j açıları teğet vektörü olan $\vec{d}_{ij}$ ile düzlemler arasındaki açısal farkları ifade etmektedir.

Şekil 4.10 (c)’de birbirine tamamen paralel ama birlikte bir düzleme konumlandırılmayacak iki düzlem arasındaki β_i ve β_j açıları 0° (derece - degree) olduğu

için α_i ve α_j açıları 90° 'dir. Bu sonuç da, bu iki düzlem arasındaki α_i ve α_j açılarının maksimum derecede olduklarını ve normal vektörler arasındaki θ fark açısı 0° olsa bile α_i ve α_j açıları sayesinde bu iki düzlemin aynı gruba ait olmaması gerektiği tespit edilmektedir. Şekil 4.10 (d)'de hem birbirine paralel hem de birlikte bir düzleme konumlandırılabilir iki komşu düzlem görülmektedir. Bu durumda hem θ açısı hem de α_i ve α_j açıları 0° olmaktadır ve bu iki düzlem iyi bir birleşme özelliği göstermektedir.



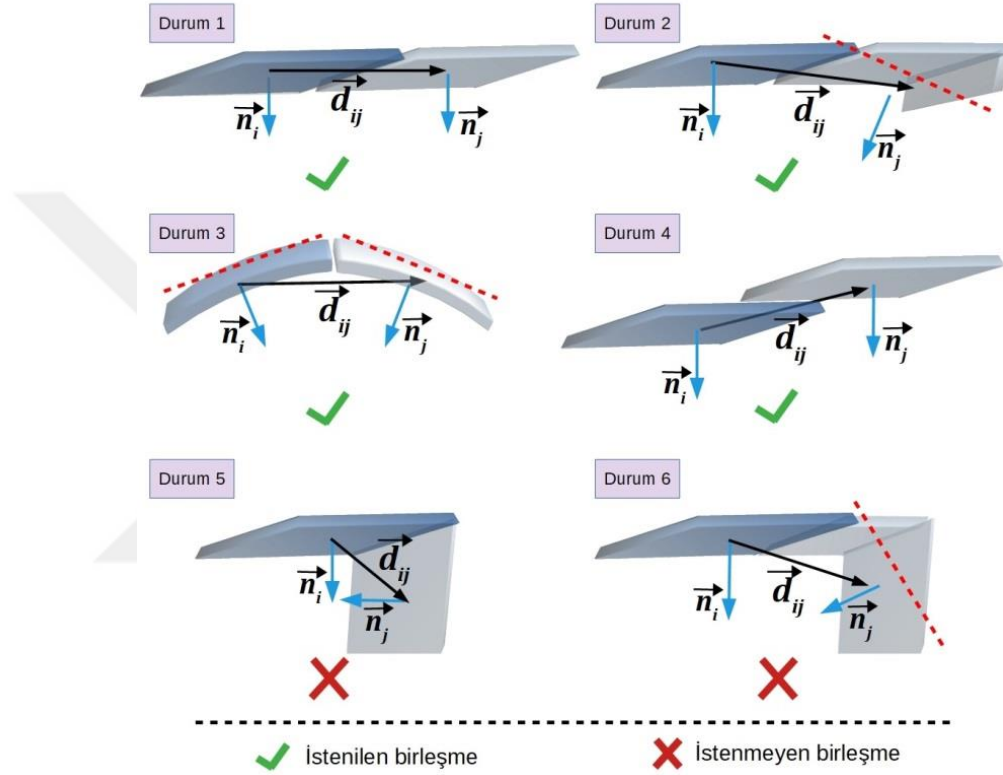
Şekil 4.10. (a) İç bükey düzlemler, (b) dış bükey düzlemler, (c) paralel ama beraber bir düzlem oluşturmayan düzlemler ve (d) hem paralel hem de beraber bir düzlem oluşturan düzlemler

Tez kapsamında geliştirilen segmentasyon yöntemi, komşu lokal yüzeyleri temsil eden bitişik vokseller arasındaki bağlantının geometriksel ağırlık ölçümünü yani iki vokseller arasındaki geometriksel farklılığı kullanarak, bitişik vokselleri birleştirip segmentasyon işlemi yapmaktadır. Birleştirme işlemi için, bir eşik açısı (τ_{max}) değeri kullanılmaktadır. Bu değer kullanıcı tarafından $[0-90]^\circ$ değer aralığında parametre olarak girilmektedir. Segmentasyon işleminde kullanılan ağırlık ölçüm değerinin bu eşik açısından küçük veya eşit olması istenmektedir.

Segmentasyon işleminde bitişik voksellerin geometriksel özelliklerine göre lokal yüzeylerin yapılarında görülebilecek geçiş olasılık durumları Şekil 4.11'de gösterilmektedir. Komşu vokseller arasındaki bağlantılar için atanan ağırlık (farklılık) değerleri, geliştirilen metottaki voksellerin birleştirme aşamasında, voksellerin veya

voksel gruplarının (segmentlerin) birleştirilip birleştirilmemesinde karar verici nitelikte olmaktadır.

Bir bağlantının “ağırlık değeri” olarak tanımlanan $w(v_i, v_j)$ ifadesi, v_i and v_j vokselleri arasındaki bağlantı için sayısal bir değeri temsil etmektedir. Ağırlık değerinin hesaplanmasında, Şekil 4.11’de görülen, yüzeyler arasındaki farklı geçiş olasılıkları dikkate alınmıştır.



Şekil 4.11. İki lokal yüzeyin farklı geçiş olasılıkları için istenen ve istenmeyen birleşmeler

Şekil 4.11’deki geçiş olasılıkları göz önüne alındığında, Denklem 4.9’daki ağırlık değeri ölçümü tez kapsamında geliştirilmiştir. Denklem 4.9’da, $w(v_i, v_j)$ ifadesi v_i and v_j vokselleri arasındaki bağlantının ağırlık değerini, $\bar{\alpha}$ ifadesi ise α_i and α_j açılarının ortalaması olan $(\alpha_i + \alpha_j)/2$ değerini göstermektedir.

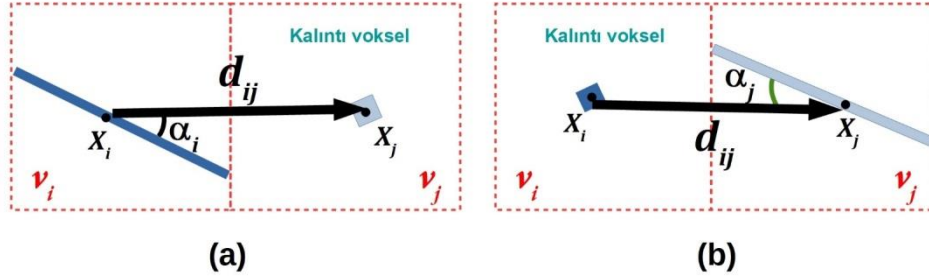
$$w(v_i, v_j) = \min(\theta, \bar{\alpha}) \quad (4.9)$$

Şekil 4.11’deki Durum 3 düşünüldüğünde, eğer birleştirme için sadece Şekil 4.9’da verilen θ açısı dikkate alınırsa ve kullanıcı tarafından $[0-90]^\circ$ değer aralığında

parametre olarak girilen eşik açısı (τ_{max}), $\bar{\alpha}$ ve θ arasında ise bu iki yüzeyin birleştirilmeyeceği görülmektedir. Hâlbuki iyi bir segmentasyon olması için şekle göre bu iki yüzeyin birleştirilmesi istenmektedir. Aynı şekilde Durum 4'e bakıldığında, eğer sadece $\bar{\alpha}$ açısal değeri dikkate alınır ve eşik açısı (τ_{max}), $\bar{\alpha}$ ve θ arasında ise bu iki yüzey birleştirilmeyecektir. Oysa Durum 4'e göre bu yüzeylerin de birleştirilmesi istenmektedir. Buna benzer durumlar dikkate alınarak Denklem 4.9'daki birleştirme kriteri geliştirilmiştir.

Tezde önerilen metot içerisinde, komşu iki vokselin ikisi de kalıntı ise bu vokseller birleştirilmemektedir. Bir kalıntı voksel sadece kalıntı olmayan voksel ile ve yalnızca bir defa birleşebilmektedir. Eğer v_i ve v_j voksellerinden, v_j vokseli kalıntı ise bu iki komşuluk arasındaki hat ağırlığı derecesi α_i olarak ölçülmektedir (Şekil 4.12 (a)). Tam tersi durum için ise α_j olarak hesaplanmaktadır (Şekil 4.12 (b)). Böylelikle, eğer komşu iki vokselde biri kalıntı ise; Denklem 4.9'da verilen vokseller arası ağırlık değeri hesaplaması yerine Denklem 4.10'daki hesaplama kullanılmaktadır.

$$\begin{aligned} w(v_i, v_j) &= \alpha_i \Leftarrow v_j \text{ kalıntı} \\ w(v_i, v_j) &= \alpha_j \Leftarrow v_i \text{ kalıntı} \end{aligned} \quad (4.10)$$

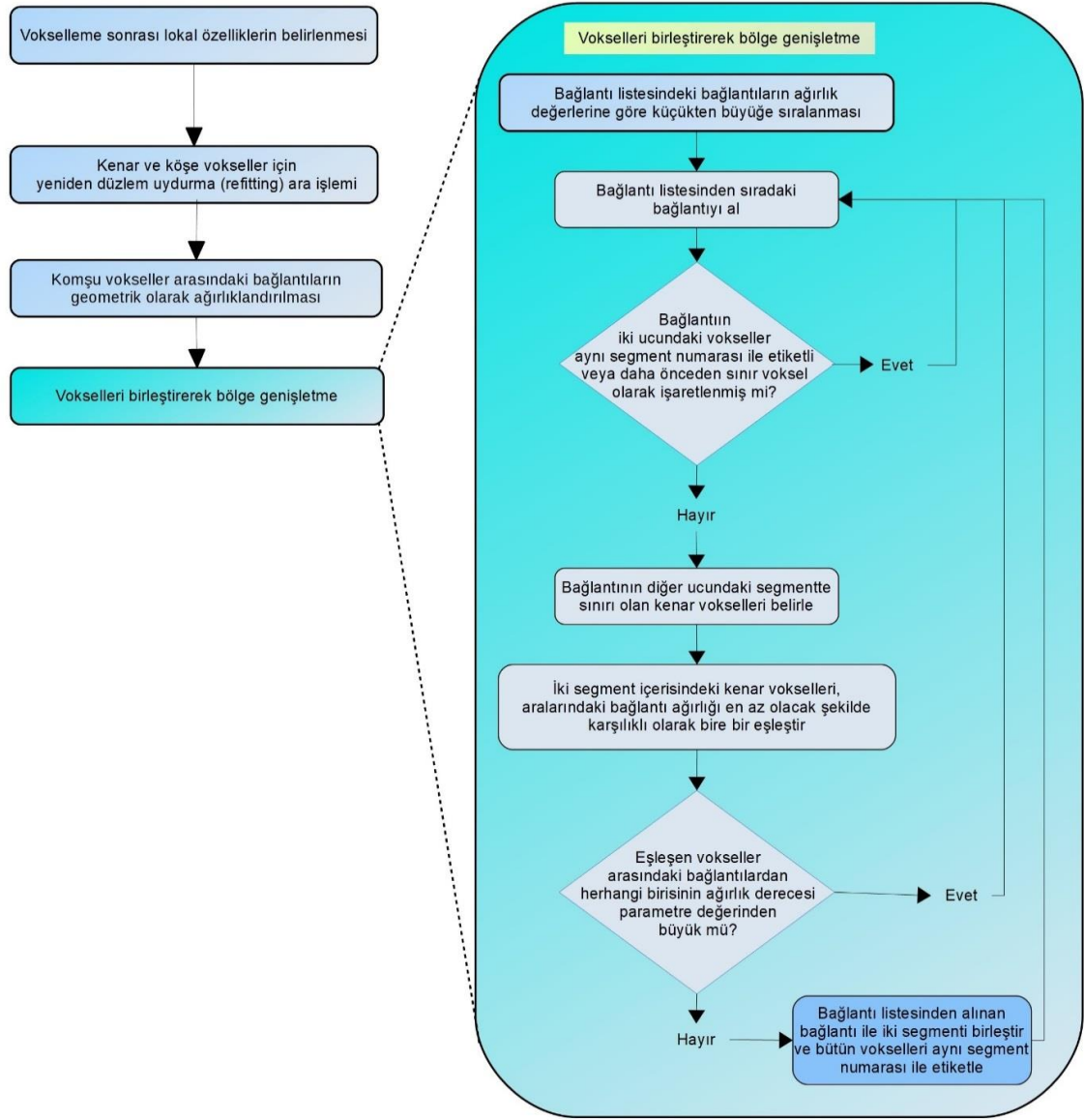


Şekil 4.12. Kalıntı ve kalıntı olmayan vokseller arasındaki açısal farklılıklar

Buraya kadar açıklanan bağlantı ağırlıklandırma kısmı yalnızca geometrik farklılık içindir. Eğer nokta bulutu verisindeki renk farklılığı da kullanılacaksa, iki voksele ait olan renk değerlerinin Öklid uzaklığı da geometrik farklılıkta olduğu gibi $[0-90]^\circ$ aralığına ölçeklenip, kullanıcı tarafından belirlenen etki oranı kadar bağlantı ağırlıklandırmaya bu aşamada dahil edilecektir. Bütün bağlantılar ağırlıklandırıldıktan sonra bağlantılar ve karşılıkları olan ağırlık değerleri liste halinde tutularak bir sonraki aşamada kullanılmak üzere hazır hale getirilmektedir.

4.1.4. Vokselleri birleştirerek bölge genişletme

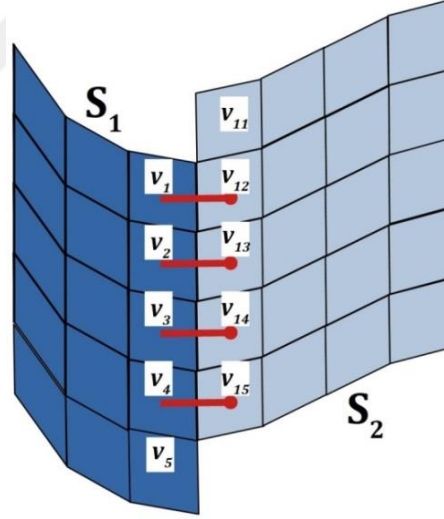
Vokselleri birleştirerek bölge genişletme aşamasının, geliştirilen metodun ana aşamalarının içerisindeki yeri ve iç adımları Şekil 4.13'te şema halinde görülmektedir.



Şekil 4.13. Vokselleri birleştirerek bölge genişletme aşamasının geliştirilen metodun ana aşamalarının içerisindeki yeri ve iç adımları

Tezde geliştirilen metoda göre, her bir voksel ilk başta bir segment niteliği taşımaktadır ve bu segmentlerin yüzey eğimlerinin sürekliliği dikkate alınarak, segmentlerin birleştirilip büyütülmesi ile amaç daha büyük ve anlamlı segmentler oluşturulmasıdır. Önceki işlem aşamasında bütün vokseller arasındaki bağlantı ağırlıkları hesaplanmış ve bir bağlantı listesine eklenmişti. Bu aşamada bağlantı listesi ağırlık

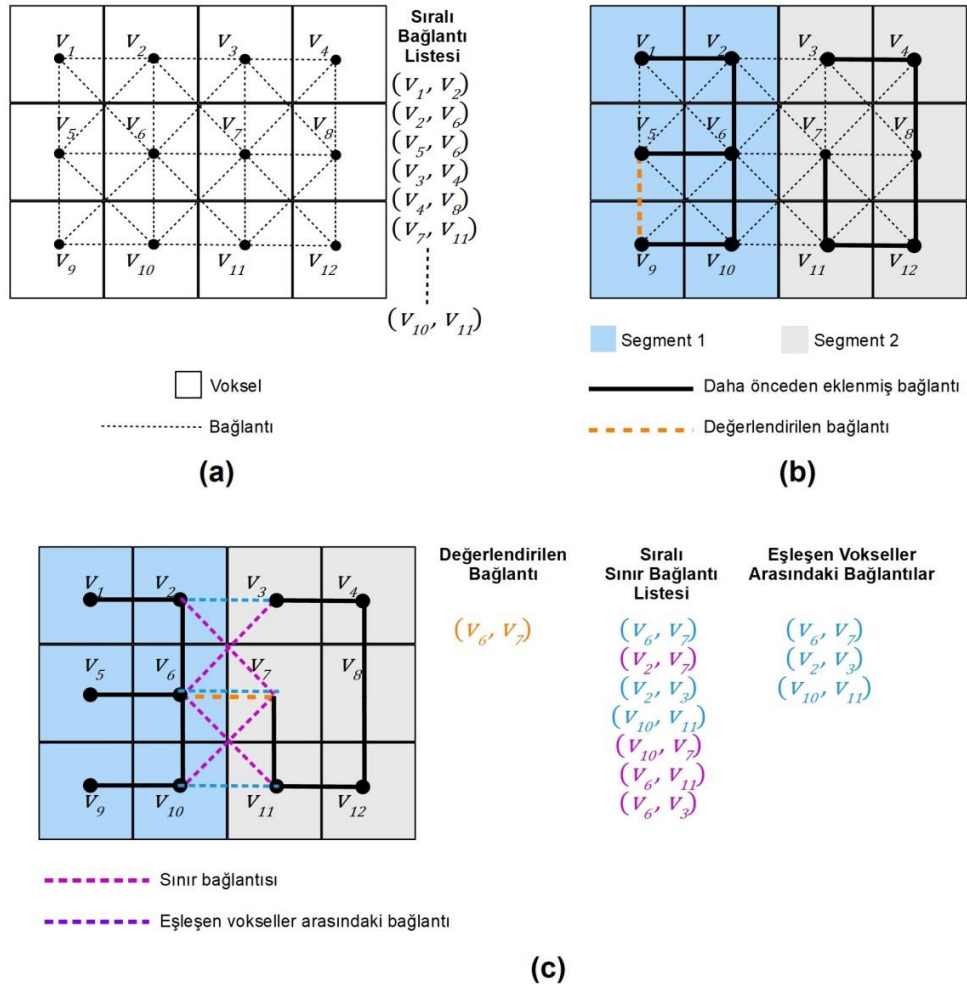
değerlerine göre küçükten büyüğe doğru sıralanmaktadır. Daha sonra, en küçük ağırlıklı bağlantıdan başlanarak, kullanıcı tarafından parametre olarak girilen $[0-90]^\circ$ aralığındaki bir τ_{max} açısal eşik değerine göre sırayla birleştirilmektedir. Bir önceki aşamada elde edilen komşu vokseller arasındaki ağırlık (değişim) değerleri $[0-90]^\circ$ derece arasında olacağı için, eşik değeri de minimum 0° , maksimum 90° derece olabilmektedir. Başlangıçta her birisi farklı etiket değerine sahip olan vokseller, birleştikleri vokseller veya vokal grupları ile aynı etiket değerine sahip olmaktadır. Geliştirilen metoda göre, birden fazla voksele sahip vokal grupları birleştirilirken, sadece değerlendirilen bağlantının ağırlığına değil, iki grup arasındaki komşu vokseller arasındaki bağlantılara da bakılmaktadır. Değerlendirilmeye alınan iki vokal grubu arasındaki komşu vokseller arasında, her bir vokal, kendisine en yakın olan karşı gruptaki vokal ile birebir eşleşmektedir. Eşleşen vokseller arasındaki bağlantıların tamamı segmentasyon işlemi için belirlenen bir açısal parametre değeri olan τ_{max} değerinden küçük veya eşit ise bu iki vokal grubu birleştirilmekte (Şekil 4.14), aksi durumda birleştirme olmamaktadır.



Şekil 4.14. Komşu voksellerin birebir eşleşmesi

Bölge genişletme aşaması, geliştirilen metot içerisindeki en kritik aşamadır ve literatürdeki diğer metotlarla arasındaki en büyük farklılığı bu aşamada göstermektedir (Saglam ve ark., 2020). Bölge büyütme aşamasını daha detaylı ve Şekil 4.14'teki ifadeler ile açıklamak gerekirse, başlangıçta her bir vokal seti (segment) içerisinde yalnızca bir vokal vardır ve ilk baştaki $S = \{S_1, S_2, \dots, S_n\}$ segment kümesi, bütün vokselleri içeren $V = \{v_1, v_2, \dots, v_n\}$ vokal kümesine karşılık gelmektedir (n = toplam vokal sayısı).

Birleştirme işleminin başlangıcında, bitişik vokseller arasındaki ağırlıklandırılmış bağlantılardan oluşan bağlantı listesindeki bağlantılar, ağırlıklarına göre artan şekilde sıralanmaktadır. Bu işleme bir örnek gösterim Şekil 4.15 (a)'da verilmektedir. En düşük ağırlıklı bağlantıdan başlayarak, sırayla listeden alınan bağlantıların her iki ucundaki voksellere sahip segmentler birleştirilmek amacıyla değerlendirmeye alınmaktadır. Eğer bu iki vokselle aynı segmente ait ise (aynı segment numarası ile etiketlenmişse) listedeki bir sonraki bağlantı değerlendirmeye alınmaktadır. Şekil 4.15 (b)'de değerlendirme sırası gelen (v_5, v_9) bağlantısının her iki ucundaki vokselle Segment 1'e ait olduğu için diğer birleştirme kriterlerine bakılmaksızın dikkate alınmayacak ve liste sırasındaki bir sonraki bağlantı değerlendirmeye alınacaktır.



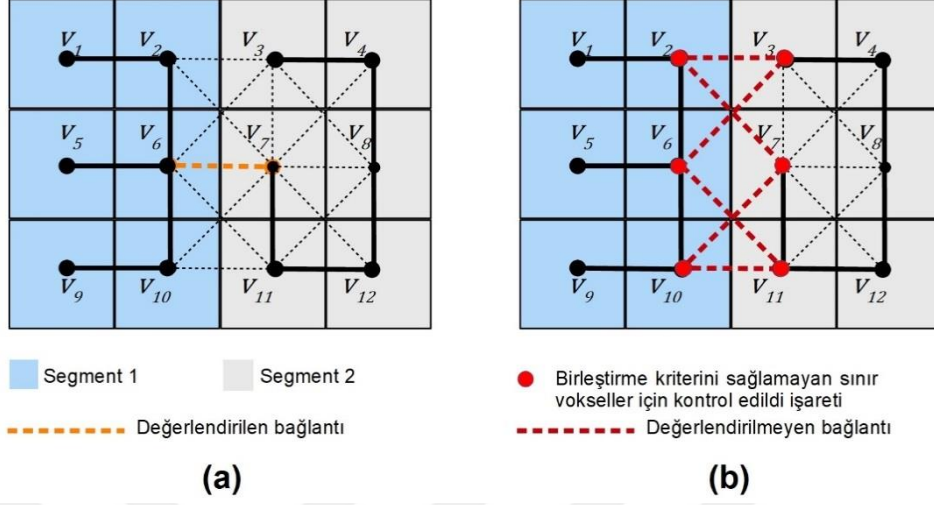
Şekil 4.15. Bütün bitişik vokseller arasındaki bağlantıların ağırlık değerine göre sıralanmış bağlantı listesi örneği (a), aynı segment içerisindeki vokseller arasındaki bağlantıya örnek (b) ve ağırlık değerlerine göre sıralanmış sınır bağlantı listesindeki eşleşen vokseller arasındaki bağlantılara örnek (c)

Değerlendirme aşamasında, bir segmentin diğer segmentteki voksellere bitişik olan ve “sınır vokselleri” olarak adlandırılan vokseller, geliştirilen yöntemde dikkate alınmaktadır. Şekil 4.14’te, (v_i, v_j) bağlantısı ($v_i \in S_1$ ve $v_j \in S_2$) birleştirme için dikkate alındığında, $\{v_1, v_2, v_3, v_4, v_5\} \in S_1$ ve $\{v_{11}, v_{12}, v_{13}, v_{14}, v_{15}\} \in S_2$ sınır voksel kümelerindeki vokseller, en küçük ağırlık önceliğine göre karşı segmentteki bitişik voksellerden birisi ile eşleştirilmektedir.

Sınır vokselleri belirleme aşamasının yazılımsal olarak kodlanmasında, iki bitişik segmentten daha küçük (içerdiği voksel sayısı az) olanı, sınır voksellerin tespiti için tarandığında, programın çalışması önemli ölçüde hızlanmıştır. Küçük olan segmentteki sınır vokseller bulunduğundan sonra, bu voksellerin indeksleri ile komşu vokselleri bulma yöntemi kullanılarak, diğer segmentteki bitişik sınır vokselleri, büyük olan segmenti taramadan kolayca bulunabilmektedir. Sınır vokselleri bulunduğundan sonra, her iki segment içerisindeki sınır voksellerinden karşı segmentteki bitişik sınır vokseller arasındaki tüm bağlantılar bir listeye eklenmektedir (Şekil 4.15 (c)). Bu liste, tez kapsamında “sınır bağlantı listesi” olarak isimlendirilmektedir. Sınır bağlantı listesi, ağırlık değerlerine göre artan şekilde sıralandıktan sonra, en küçük ağırlıklı olandan başlayarak bağlantılar tek tek eşleştirilmeye başlanmaktadır. Şekil 4.15 (c)’de sıralı bağlantı listesindeki sıraya göre eşleşme sırası (v_2, v_7) bağlantısının uçlarındaki voksellere geldiğinde, v_7 vokseli daha önceden v_6 ile eşleştiği için liste sırasındaki bir sonraki bağlantının uçlarındaki vokseller eşleştirilmek üzere değerlendirmeye alınmaktadır.

Bir bağlantı, birleştirme kriterini sağlıyorsa, bağlantının iki ucundaki vokseller, bu voksellerin başka herhangi bir voksel ile eşleşmemesi için işaretlenmektedir. Eğer bağlantının ucundaki vokseller önceden işaretlenmişse veya voksellerden herhangi biri kalıntı voksel ise, bağlantının sonundaki vokseller eşleştirilmemekte ve sınır bağlantı listesindeki bir sonraki bağlantı dikkate alınmaktadır. Çiftler arasındaki her bağlantı, birleştirme kriterini sağlıyorsa, iki segment birleştirilmekte ve ardından işaretler kaldırılmaktadır. Herhangi bir çift arasındaki bağlantı, birleştirme kriterini sağlamıyorsa, iki segment birleştirilmemekte ve bu iki segmentin tekrar değerlendirilmemesi için işaretler kaldırılmamaktadır. Bu işaret detayı, uygulamanın çalışma aşamasında segmentasyon sürecini önemli ölçüde hızlandırmaktadır. Şekil 4.16 (a)’daki (v_6, v_7) bağlantısının değerlendirmeye alındığı ve birleştirme kriterini sağlamadığı farzedilirse, $\{v_2, v_3, v_6, v_7, v_{10}, v_{11}\}$ sınır vokselleri Şekil 4.16 (b)’deki gibi işaretlendiği için

$\{(v_2, v_3), (v_2, v_7), (v_6, v_3), (v_6, v_{11}), (v_{10}, v_7), (v_{10}, v_{11})\}$ bağlantıları bundan sonra birleştirme için değerlendirmeye alınmayacaktır.

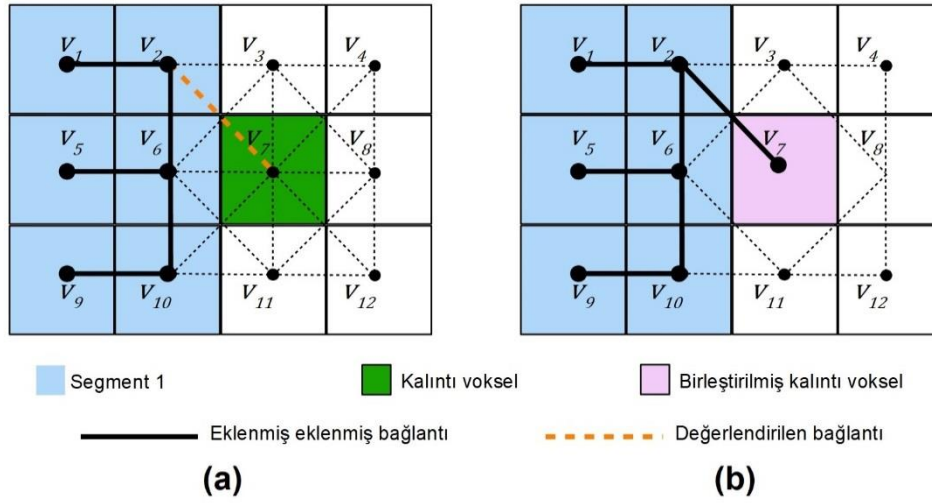


Şekil 4.16. Birleştirme için değerlendirmeye alınan bir bağlantının (a) birleştirme kriterini sağlamadığı durumda işaretlenen sınır vokseller ve bir daha değerlendirmeye alınmayacak olan bağlantılar

Şekil 4.14'e bakıldığında, eşleştirme işleminin sonucunun $E_{1,2} = \{(v_1, v_{12}), (v_2, v_{13}), (v_3, v_{14}), (v_4, v_{15})\}$ bağlantı kümesi olduğu görülmektedir. Önerilen yöntemde, $E_{1,2}$ bağlantı kümesine karşılık gelen $W_{1,2} = \{w(v_1, v_{12}), w(v_2, v_{13}), w(v_3, v_{14}), w(v_4, v_{15})\}$ ağırlık değeri kümesindeki bütün değerler, S_1 ve S_2 segmentlerinin birleştirilmesi işlemi için τ_{max} eşik değeri tarafından değerlendirilmektedir. Eğer $W_{1,2}$ kümesindeki herhangi bir ağırlık değeri τ_{max} değerinden büyükse S_1 ve S_2 segmentleri birleştirilmemekte; aksi durumda ise birleştirilmektedir. Eşik değeri $[0-90]^\circ$ aralığında olabilmektedir; çünkü bağlantı ağırlıkları, bağlantıların ağırlıklandırılması aşamasında açıklanan ağırlık hesaplamalarına göre (Denklem 4.9) bu aralıkta olabilmektedir.

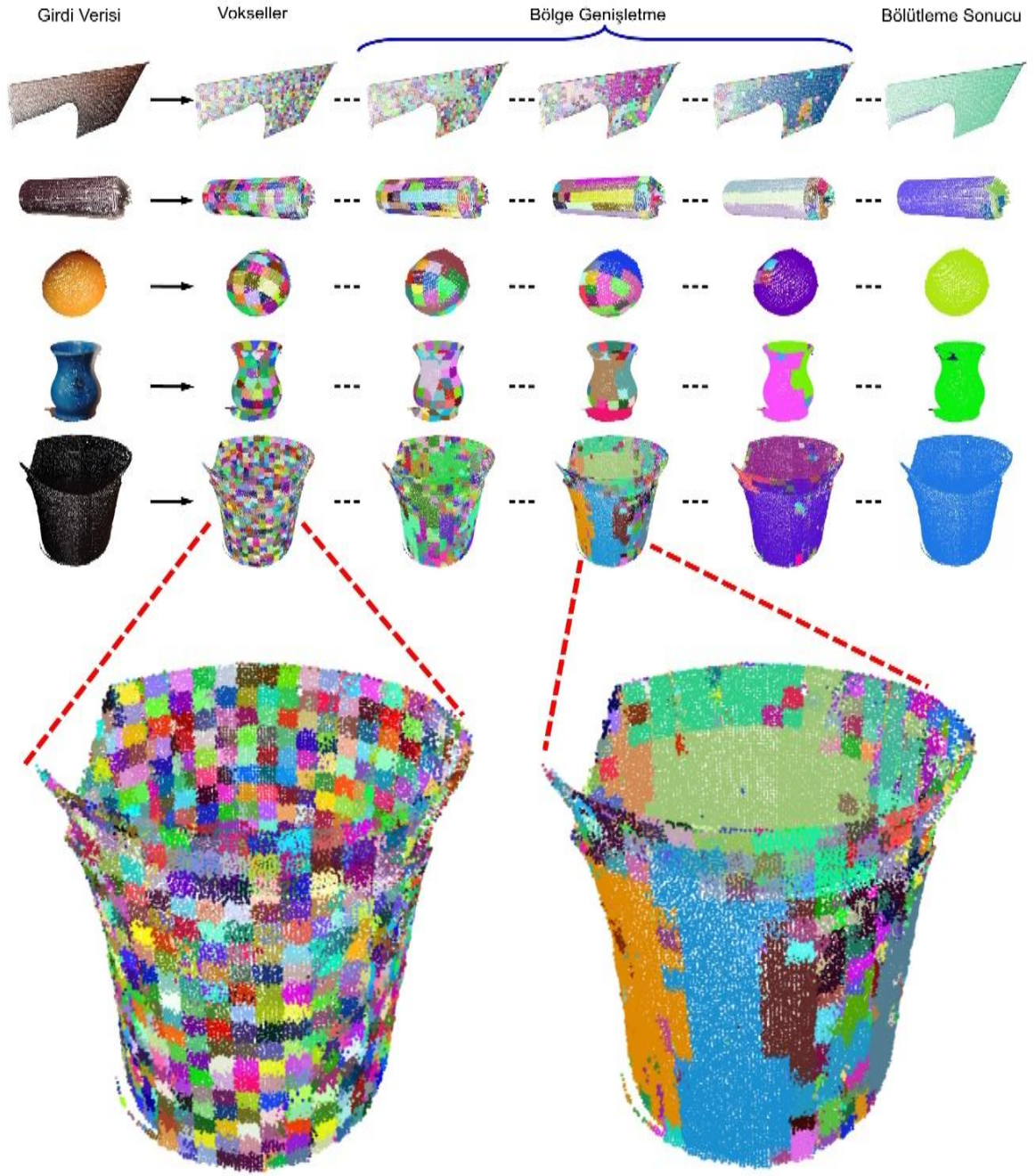
İki segmentin birleştirilip birleştirilmeyeceği değerlendirildikten sonra, algoritmanın başında oluşturulan ve bitişik vokseller arasındaki tüm bağlantıları içeren sıralı bağlantı listesine göre, bir sonraki bağlantının iki ucundaki segment çifti değerlendirmeye alınmaktadır. Eğer bağlantının her iki ucundaki vokselle de kalıntı vokselle ise birleştirme olmamaktadır. Bir kalıntı vokselle, bir defaya mahsus olmak üzere koşulsuz şekilde bir kalıntı olmayan segment ile birleştirilmektedir. Birleştirme işleminde ağırlığı düşük olan bağlantılardan başladığı için kalıntı voksellerin de koşulsuz olarak birleştiği segment kendisine en yakın olan segment olacaktır. Kalıntı olmayan bir vokselle bir

segment ile birleştirildikten sonra, tekrar başka bir segment ile birleştirilemeyecek şekilde işaretlenmektedir (Şekil 4.17).



Şekil 4.17. Bir kalıntı vokselin ilk birleşmeden sonra bir daha birleşmemek üzere işaretlenmesi

Şekil 4.18’de geliştirilen yöntemin, segmentasyon işlemindeki bölge büyütme aşamasının farklı şekillerdeki gerçek nesneler üzerindeki bazı safhaları görülmektedir. Şekil 4.19’da ise vokselleri birleştirerek bölge genişletme aşamasının sözde kodu gösterilmektedir. Şekil 4.20’de ise Örnek 1 veri setine uygulanan geliştirilen segmentasyon yönteminde bölge genişletme aşamasındaki bazı ara safhaları görülmektedir. Şekil 4.20’de görüldüğü gibi, bölge genişletme işlemi aralarındaki bağlantı ağırlıkları en düşük olan komşu voksellerden başladığı için segmentasyon işlemi de ilk önce en düz yapıya sahip yüzeylerden başlamaktadır.



Şekil 4.18. Geliştirilen yöntemin bölge büyüme aşamasındaki bazı safhaları

Girdi: V kümesindeki başlangıçta hiç birisi işaretlenmemiş olan vokselleri temsil eden S segment kümesi ve bütün bitişik vokseller arasındaki E bağlantı listesi

E kümesindeki bağlantıları ağırlık değerlerine göre küçükten büyüğe sırala

Sıralanmış E listesindeki en küçük ağırlık değerine sahip olan ilk bağlantıyla başla

for $(v_i, v_j) \in E \wedge w(v_i, v_j) \leq \tau_{max} \wedge (v_i \text{ veya } v_j \text{ işaretlenmemiş ise})$

S_1, v_i vokselini içeren segment olsun

S_2, v_j vokselini içeren segment olsun

if $(S_1 = S_2)$

continue

$V_1 \leftarrow S_1$ segmentindeki S_2 segmentine bitişik olan sınır vokseller

$V_2 \leftarrow S_2$ segmentindeki S_1 segmentine bitişik olan sınır vokseller

V_1 ’deki vokseller ile V_2 ’deki vokselleri, düşük ağırlık değerine sahip bağlantılara öncelik vererek bire bir eşleştir

$E_{1,2} \leftarrow$ eşleşen voksel çiftleri arasındaki bağlantılar

$W_{1,2} \leftarrow E_{1,2}$ ’deki bağlantılarla aynı sıradaki ağırlık değerleri

Eşleşen voksel çiftlerini işaretle

if $(\exists W_{1,2} > \tau_{max})$

continue

İşaretlenen voksel çiftlerinin işaretini kaldır

if $(|S_1| > |S_2|)$

$S_1 \leftarrow S_1 \cup S_2$

$S \leftarrow S - S_2$

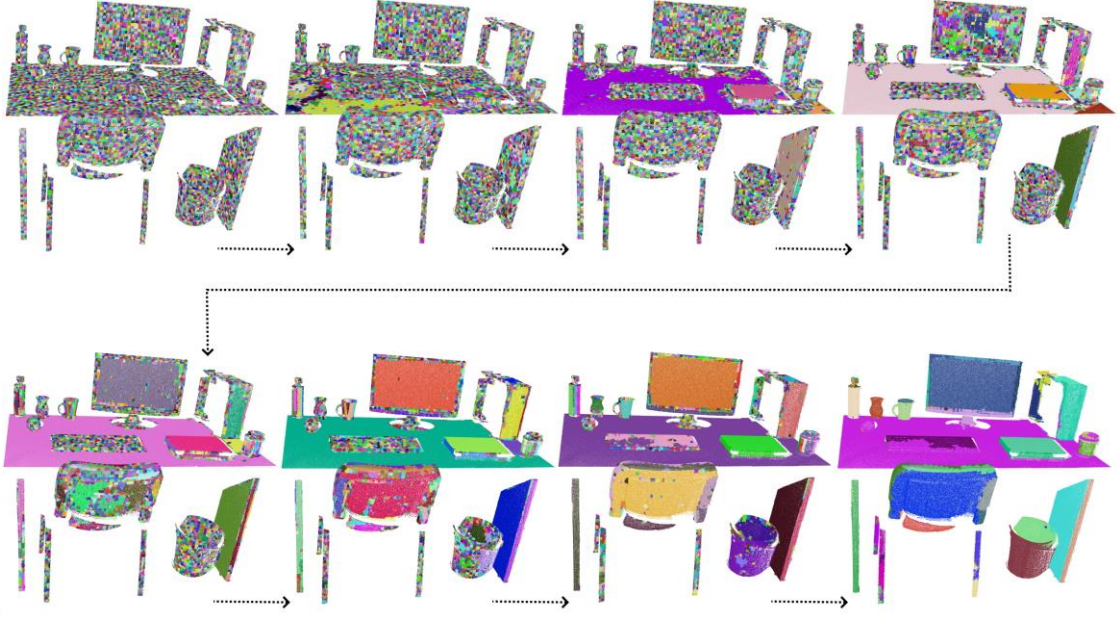
else

$S_2 \leftarrow S_1 \cup S_2$

$S \leftarrow S - S_1$

end

Şekil 4.19. Vokselleri birleştirerek bölge genişletme aşamasının sözde kodu



Şekil 4.20. Örnek 1 veri setinin bölge genişletme aşamasındaki bazı ara safhaları

4.2. Geliştirilen Yöntemin Sınıflandırmada Ara İşlem Olarak Kullanılması

3D nokta bulutlarından anlamlı bilgi toplamak, uzaktan algılama, robotik ve bilgisayarlı görme alanlarında çok önemli bir konudur (Weinmann ve ark., 2015b). Nokta bulutu sınıflandırması; nesne algılama (Pu ve ark., 2011; Serna ve Marcotegui, 2014), kentsel haritalama (Guindon ve ark., 2004; Haala ve ark., 2008), bitki örtüsü haritalama (Wurm ve ark., 2014), bina yapılarını tanımlama (Vanegas ve ark., 2012), şehir modelleme (Lafarge ve Mallet, 2012) ve karmaşık 3D sahnelerin yorumlanması (Boulch ve ark., 2013) gibi birçok uygulama için anahtar bir işlemdir.

Segmentasyon aşaması sınıflandırma işleminde, kullanılan sınıflandırıcının doğru ve hızlı bir sınıflandırma yapabilmesine yardımcı olan önemli bir ara işlemdir (Vosselman ve ark., 2017). Segmentasyon, benzer veri öğelerini benzerliklerine göre gruplandırarak harcanan gereksiz işlem süresini azaltmakta ve sonraki aşamalar için daha ayırt edici özellikler sağlamaktadır. Segmentasyon aşamasından sonraki gelen süreç olan özellik çıkarma aşamasında segmentlerden, tanımlanan özellikler çıkarılarak her segment için bir özellik vektörü elde edilmektedir. Sınıflandırma uygulamasında, özellik çıkarma aşamasından sonra devreye giren sınıflandırıcı, bu özellik vektörlerine göre eğitim ve test işlemlerini gerçekleştirmektedir. Tez kapsamında, geliştirilen yöntem bir ara işlem olarak kullanılarak, sınıflandırma uygulaması da yapılmıştır. Sınıflandırma için Bölüm 3.4.3'te

verilen sınıflandırma veri seti (S3DIS) kullanılmıştır. Çizelge 4.1’de, tez kapsamında yapılan sınıflandırma uygulamalarında kullanılan özellikler ve isimleri verilmiştir. Sınıflandırma işleminde girdi olarak kullanılmak üzere, tez kapsamında geliştirilen segmentasyon yöntemi sonucunda elde edilen her segment için eğer sadece konum bilgisi kullanılacaksa toplamda 15 boyutlu, eğer $L*a*b^*$ (tezdeki çalışmalarda kullanılan) olarak 3 tane renk değeri de kullanılırsa toplamda 18 boyutlu bir özellik vektörü elde edilmektedir.

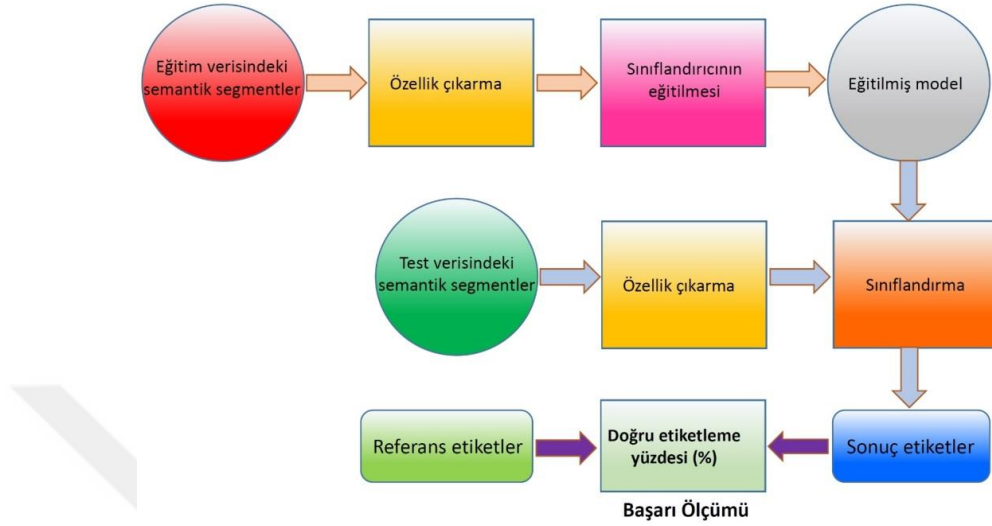
Tez kapsamında geliştirilen segmentasyon yönteminin, yapılacak bir sınıflandırmada işleminde ara işlem olarak kullanılmasında elde edilecek başarıyı test etmek amaçlanmıştır. Ancak segmentasyon yönteminin etkisinin ortaya konması için, öncelikle segmentasyon yapılmadan, sınıflandırma veri setinde (S3DIS) hazır olarak verilen referans segmentlerden belirlenen özellikler çıkarılarak, bu özellikler ile sadece sınıflandırma işlemi yapılmıştır. Daha sonra ise, geliştirilen yöntem ile önce segmentasyon yapılmış, segmentlenmiş her alt segmentten belirlenen özellikler çıkarılmış ve segmentlere ait özelliklerle sınıflandırma işlemi yapılmıştır.

Çizelge 4.1. Tez kapsamındaki sınıflandırma uygulamalarında kullanılan özellik seti

Özellik İsmi	Açıklama
Nokta Sayısı	Segmentte yer alan noktaların sayısı
Ortalama Z	Ortalama z değeri (ortalama yükseklik)
Uzunluk	Maksimum ve minimum z değerlerinin farkı
Öz değerler	λ_1, λ_2 ve λ_3
Ölçekli öz değerler	$\frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3}, \frac{\lambda_2}{\lambda_1 + \lambda_2 + \lambda_3}$ ve $\frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}$
Normal vektör değerleri	n_x, n_y ve n_z
Noktaların öz vektörlere göre standart sapmaları	σ_1, σ_2 ve σ_3 (sırasıyla e_1, e_2 ve e_3 öz vektörleri yönündeki standart sapmaları)
Ortalama renk	Noktaların ortalama renk değerleri (eğer renk özelliği kullanılacaksa)

Segmentasyon yapılmadan gerçekleştirilen sınıflandırma işleminde, S3DIS veri seti ilk önce eğitim ve test verisi olarak ayrılmıştır. Eğitim verisine ait hazır olan referans segmentlerden Çizelge 4.1’de verilen özellikler çıkarılarak, her semantik segment için bir özellik vektörü oluşturulmuştur. Eğitim verisinden çıkarılan bu özellik vektörleri ve veri setinde hazır olarak belirlenmiş sınıf etiketleri kullanılarak, sınıflandırıcı eğitilmiştir. Daha sonra test verisindeki her bir semantik segment için de aynı özellikler çıkarılarak,

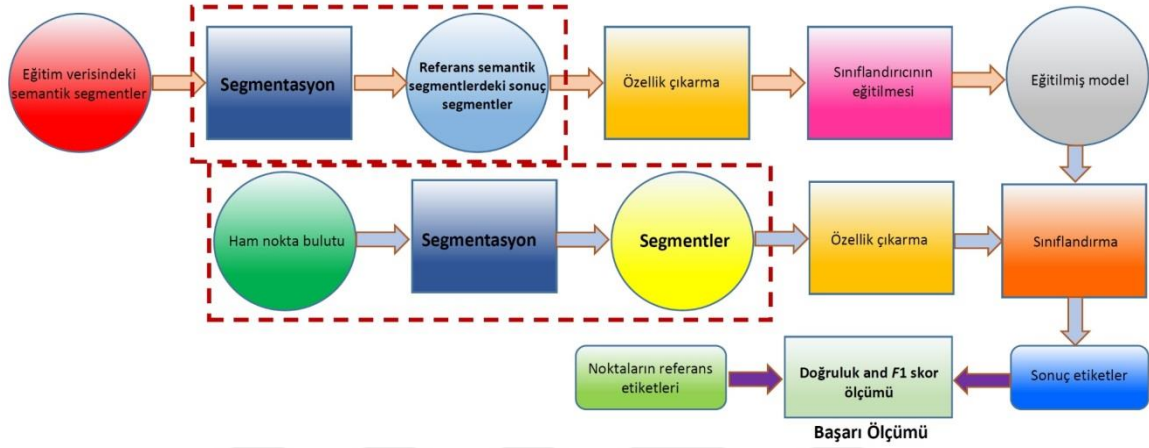
sınıflandırma işlemi yapılmıştır. En son olarak da test verisine ait sınıf etiketleri kullanılarak sınıflandırmanın başarısı segment bazında doğru etiketleme yüzdesi (%) olarak ölçülmüştür. Bu uygulamadaki aşamalar şema halinde Şekil 4.21’de görülmektedir.



Şekil 4.21. Referans semantik segmentlerin sınıflandırılması

Segmentasyon yapılmadan gerçekleştirilen sınıflandırmanın ardından, geliştirilen yöntemin başarısını test etmek için, S3DIS verisi önce geliştirilen yöntem ile segmentlenmiş ve ardından da sınıflandırma işlemi yapılmıştır. Bu çalışma için de, daha önce ayrılmış olan eğitim ve test verileri kullanılmıştır. Çalışma için öncelikle, eğitim verisindeki bütün referans semantik segmentler, geliştirilen segmentasyon yöntemi ile ayrı ayrı segmentlenerek alt segmentler oluşturulmuştur. Çünkü tez içerisindeki çalışmalarda ham nokta bulutu segmentlenirken geometriksel yüzey özelliklerine ve/veya renk özelliklerine göre segmentlenmektedir. Veri setinde var olan referans semantik segmentler ise nesnelerin anlamsal yapılarına göre yüzey ve renk farklılığı gözetmeksizin kenarları el ile (manuel) belirlenmiş anlamlı segmentlerdir. Eğitim modelinin doğrudan referans segmentler ile eğitilmesi, test aşamasında yüzey eğim ve/veya renk farklılıklarına göre sınırları belirlenmiş olan segmentler için modelin çok fazla yanlış sınıflandırma yapmasına sebep olacaktır. Geliştirilen yöntemle alt segmentler oluşturulduktan sonra, her bir alt segmentten yine Çizelge 4.1’de verilen özellikler çıkarılarak, bir özellik vektörü elde edilmiştir. Daha sonra her bir alt segmente karşılık gelen özellik vektörleri ile alt segmentin ait olduğu referans semantik segmentin sınıf etiketi birlikte sınıflandırıcıya verilerek, eğitim yapılmıştır. Test için ise; test verisinde de segmentasyon yapılmış, alt segmentler oluşturularak bu segmentlere ait özellikler

çıkarılmıştır. Çıkarılan özellikler yine referans segmente ait sınıf etiketleri ile sınıflandırıcıya verilerek, test edilmiştir. Sınıflandırma sonucunda etiketlenen segmentlerin etiket değerleri, segmentlerin içerisinde yer alan noktalara atanmış ve test verisindeki noktaların referans sınıf etiketleri ile karşılaştırılarak, nokta bazında bir başarı ölçümü yapılmıştır. Bunun için de doğruluk ve $F1$ skor ölçümleri kullanılmıştır. Bu uygulamadaki aşamalar şema halinde Şekil 4.22’de görülmektedir.



Şekil 4.22. Ham nokta bulutundaki noktaların segmentasyon yardımı ile sınıflandırılması

5. DENEYSEL SONUÇLAR VE TARTIŞMA

Tez kapsamında yapılan bütün uygulamalar, ÖYP proje desteğiyle satın alınan 2,8 GHz Intel i7-7700K CPU ve 16 GB RAM'e sahip bir dizüstü bilgisayar ve TÜBİTAK-1002 proje desteğiyle satın alınan 3,6 GHz Intel i9-9900K CPU ve 64 GB RAM'e sahip bir masaüstü bilgisayar kullanılarak gerçekleştirilmiştir. Tezdeki hız karşılaştırmaları için daha yüksek hıza sahip olan masaüstü bilgisayar (Intel i9-9900K CPU) kullanılmıştır.

Tez kapsamında geliştirilen yöntem hem 3D nokta bulutunda yalnızca segmentasyon işlemi olarak uygulanarak, segmentasyon sonuçları literatürdeki algoritmalarla karşılaştırılmış; hem de 3D nokta bulutu verilerinin sınıflandırılmasında bir ara işlem olarak, verilerin segmente edilmesi için uygulanıp sonuçlar sınıflandırıcı başarıları açısından elde edilmiştir. Geliştirilen yöntemin segmentasyon sonuçları literatürde yer alan RG (Rabbani ve ark., 2006), VIS (Wang ve Tseng, 2011), ORG (Vo ve ark., 2015), LCCP (Stein ve ark., 2014) ve SVGS (Xu ve ark., 2017) yöntemleriyle karşılaştırılmıştır. VIS ve ORG yöntemlerinin kodları açık erişimde olmadığı için tez kapsamında yeniden kodlanmıştır. RG ve LCCP yöntemlerinin kaynak kodları, PCL (Point Cloud Library) kütüphanesinden (Rusu & Cousins, 2011) temin edilmiştir. SVGS yönteminin kaynak kodları, bu yöntemin sunulduğu çalışmanın ilk yazarının kendi kişisel web sayfasında (Xu, 2019) yayınlanmaktadır ve kodlar bu web sayfasından temin edilmiştir. Geliştirilen segmentasyon yöntemi dâhil, karşılaştırma için kullanılan tüm segmentasyon yöntemleri C++ programlama dili kullanılarak gerçekleştirilmiştir. Segmentasyon işlemlerinde, TBA ile temel bileşenlerin hesaplanması ve 3D nokta bulutu verilerinin (ham nokta bulutu, segmentasyon çıktıları vb.) gösterim işlemleri için PCL kütüphanesi (Rusu ve Cousins, 2011) kullanılmıştır.

Geliştirilen segmentasyon yönteminin, 3D nokta bulutu sınıflandırmasında ara işlem olarak kullanımında ise, segmentlerden öznitelik çıkarma ve bu özniteliklere göre segmentlerin sınıflandırılması işlemleri Python programlama dili kullanılarak gerçekleştirilmiştir. Eğitim ve test işlemlerinde kullanılan sınıflandırıcılar (DVM ve RO), scikit-learn kütüphanesinde (Pedregosa ve ark., 2011) mevcut olan programlar üzerinden çalıştırılmıştır.

Tez kapsamında gerçekleştirilen segmentasyon uygulamalarındaki sonuçlar, hem renklendirilmiş görseller hem de nicel başarı parametreleri üzerinden değerlendirilmiştir. Nicel değerlendirme için “doğruluk” (accuracy) ve “genel $F1$ skor” (overall $F1$ score) ölçümleri kullanılmıştır. Bu ölçümler, sınıflandırma gibi tanımlanmış etiketler

kullanılarak sonucu alınan uygulamalarda sıklıkla ve kolaylıkla kullanılabilmekteyken, segmentasyon uygulamasında kullanılması için bazı ek işlemler gerektirmektedir. Segmentasyon veri setlerindeki referans segmentlerin ve segmentasyon işlemi sonucunda elde edilen sonuç segmentlerin etiket numaraları belirli tanımlamalar ile yapılmadığı için, bunların birebir karşılaştırılmaları mümkün olamamaktadır. Bu problemin üstesinden gelebilmek için kullanılan en yaygın çözümlerden birisi, karşılaştırma öncesinde referans ve sonuç segmentlerdeki etiketlerin karşılıklı olarak eşleştirilmesi işleminin gerçekleştirilmesidir.

Tez kapsamında gerçekleştirilen referans ve sonuç segmentlerin eşleştirilmesi işlemi, literatürdeki bir 3D nokta bulutu segmentasyon çalışmasında (Vo ve ark., 2015) kullanılan yöntem (Awrangjeb ve Fraser, 2014) ile sağlanmıştır. Bu eşleştirme yöntemi iki aşamadan oluşmaktadır. İlk aşamada, referans ve sonuç etiketleri arasında, etiketlenmiş segmentler örtüşen nokta sayılarına göre karşılaştırılmakta ve karşılıklı olarak birbirini en fazla kapsayan segmentler eşleştirilmektedir. İkinci aşamada, birinci aşamada karşılıklı olarak birbirini en fazla kapsadığı bir segment bulamadığı için herhangi bir karşı segment ile eşleşmemiş olan segmentlerden, kendisini en fazla kapsayan ilk iki segmentteki eşleşmemiş olanı en fazla kapsayan ilk iki segment arasında yer alan kısımlar karşılıklı eşleştirilmektedir. İkinci aşamanın sonunda da her hangi bir karşı segment ile eşleştirilmemiş olan segmentler eşleşmemiş olarak kalmaktadır (Awrangjeb ve Fraser, 2014).

Eşleştirme işleminin sonunda her hangi bir karşı segment ile eşleşmemiş olarak kalan segmentler doğruluk (accuracy), tahmin (precision) ve geri çağırma (recall) ölçüm sonuçlarını düşürmektedir. Tahmin ve geri çağırma ölçümü her referans segment için ayrı ayrı yapıldığından, eğer bir referans segment herhangi bir sonuç segment ile eşleşmemişse, bu referans segmentin tahmin ve geri çağırma sonuçları 0 olarak değerlendirilmektedir. Bu durum da, tahmin ve geri çağırma sonuçlarının referans segment sayısı bazında ortalamasını düşürdüğünden, $OF1$ ölçüm değeri de düşmektedir.

Tez çalışmasındaki bütün segmentasyon uygulanmalarında, içerdiği nokta sayısı 10'dan küçük olan segmentler istenmeyen segmentler olarak belirlenerek renklendirilmiş görsel segmentasyon çıktılarında gürültü olarak gözükmemeleri için dahil edilmemişlerdir.

5.1. 3D Nokta Bulutu Segmentasyon Sonuçları

5.1.1. Tanımlanan geometriksel ağırlık ölçümüne göre segmentasyon sonuçları

Geliştirilen yöntemin, kullanılan veri setleri üzerindeki ilk değerlendirilmesinde, yöntem içerisinde önerilen Denklem 4.9'daki ağırlık ölçümü Çizelge 5.1'deki θ , α_1 ve α_2 açıları kullanılarak üretilen ağırlık ölçümleri ile birlikte segmentasyon başarı değerleri, en iyi parametre aralıkları ve bunların standart sapmaları bazında analiz edilmiştir. Bu analizdeki amaç, Şekil 4.11'de tanımlanan farklı yüzey geçişlerine çözüm olarak en uygun dengeyi sağlayacak ağırlık ölçümünü belirlemektir. Çizelge 5.1'de, Denklem 4.9'daki ağırlık ölçümü W_{14} olarak adlandırılmıştır.

Çizelge 5.1. Geliştirilen segmentasyon yönteminde bağlantı ağırlık ölçümü için tanımlanan farklı geometrik ağırlık ölçümleri

Ağırlık adı	Ağırlık ölçümü
W_1	θ
W_2	$\frac{\alpha_1 + \alpha_2}{2}$
W_3	$\frac{\theta + \alpha_1 + \alpha_2}{3}$
W_4	$\frac{\theta + \frac{\alpha_1 + \alpha_2}{2}}{2}$
W_5	$\max(\alpha_1, \alpha_2)$
W_6	$\min(\alpha_1, \alpha_2)$
W_7	$\max(\theta, \alpha_1, \alpha_2)$
W_8	$\min(\theta, \alpha_1, \alpha_2)$
W_9	$\max(\theta, \min(\alpha_1, \alpha_2))$
W_{10}	$\min(\theta, \max(\alpha_1, \alpha_2))$
W_{11}	$\frac{\theta + \max(\alpha_1, \alpha_2)}{2}$
W_{12}	$\frac{\theta + \min(\alpha_1, \alpha_2)}{2}$
W_{13}	$\max\left(\theta, \frac{\alpha_1 + \alpha_2}{2}\right)$
W_{14}	$\min\left(\theta, \frac{\alpha_1 + \alpha_2}{2}\right)$

Burada her bir ağırlık ölçümüne göre hesaplanan değer yani ağırlık ölçüm hesabından elde edilen değer, tez kapsamında önerilen ve segmentasyon işlemi için belirlenen bir açısal parametre değeri olan τ_{max} değeri ile karşılaştırılmakta ve karşılaştırma sonucuna göre iki yüzeyin birleştirilip birleştirilmeyeceğine karar verilmektedir (Denklem 5.1).

$$\begin{aligned} W_i \text{ için ağırlık ölçüm değeri sonucu} &\leq \tau_{max} \text{ ise yüzeyler birleşecek} \\ W_i \text{ için ağırlık ölçüm değeri sonucu} &> \tau_{max} \text{ ise yüzeyler birleşmeyecek} \end{aligned} \quad (5.1)$$

Çizelge 5.1’de tanımlanan bağlantı ağırlık ölçümleri kullanılarak, τ_{max} parametresinin farklı veri setlerine göre daha kolay belirlenebilmesi de önemlidir. Geliştirilen yöntemin en iyi doğruluk ve $F1$ skor sonucunu veren τ_{max} parametre değerleri ve standart sapmaları (SS) Çizelge 5.2 ve Çizelge 5.3’te verilmiştir.

Çizelge 5.2. Geliştirilen segmentasyon yönteminin ağırlık ölçümlerine ve veri setlerine göre en iyi doğruluk sonucunu verdiği τ_{max} parametre değerleri

Ağırlık adı	Veri Seti					SS
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5	
W_1	60°	60°	80°	60°	60°	8,0000
W_2	45°	30°	40°	25°	40°	7,3485
W_3	50°	50°	50°	45°	40°	4,0000
W_4	55°	55°	50°	50°	40°	5,4772
W_5	50°	55°	45°	35°	30°	9,2736
W_6	25°	20°	40°	10°	15°	10,2956
W_7	75°	80°	70°	75°	60°	6,7823
W_8	25°	25°	40°	15°	20°	8,3666
W_9	25°	60°	70°	75°	60°	17,4929
W_{10}	45°	60°	40°	25°	25°	13,1909
W_{11}	65°	45°	65°	55°	45°	8,9443
W_{12}	50°	50°	50°	55°	25°	10,6771
W_{13}	75°	75°	80°	75°	55°	8,7178
W_{14}	35°	30°	30°	20°	20°	6,0000

Çizelge 5.1’de tanımlanan bağlantı ağırlık ölçümleri kullanılarak, geliştirilen yöntemin veri setleri üzerinde Çizelge 5.2 ve Çizelge 5.3’te verilen en iyi τ_{max} parametre değerleri uygulanarak elde edilen doğruluk sonuçları Çizelge 5.4’te ve $F1$ skor sonuçları Çizelge 5.5’te ortalama ve standart sapma (SS) değerleri ile birlikte verilmiştir. Çizelgelerdeki her bir sonuç, geliştirilen yöntemdeki τ_{max} açısal parametresi için $[5-90]^\circ$ aralığındaki 5’er derecelik artışla uygulanmasında elde edilen en iyi sonuç değerleridir.

Çizelge 5.3. Geliştirilen segmentasyon yönteminin ağırlık ölçümlerine ve veri setlerine göre en iyi $F1$ skor sonucunu verdiği τ_{max} parametre değerleri

Ağırlık adı	Veri Seti					SS
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5	
W_1	65°	80°	40°	40°	30°	18,5472
W_2	25°	30°	25°	20°	15°	5,0990
W_3	35°	50°	30°	35°	25°	8,3666
W_4	40°	60°	40°	30°	25°	12,0000
W_5	25°	55°	30°	35°	25°	11,1355
W_6	25°	20°	5°	10°	10°	7,3485
W_7	75°	85°	50°	50°	40°	17,0294
W_8	15°	25°	10°	10°	5°	6,7823
W_9	65°	80°	50°	35°	30°	18,6011
W_{10}	35°	45°	30°	25°	25°	7,4833
W_{11}	40°	65°	45°	35°	25°	13,2665
W_{12}	45°	60°	35°	25°	30°	12,4097
W_{13}	65°	85°	65°	45°	35°	17,4356
W_{14}	25°	30°	20°	20°	15°	5,0990

Çizelge 5.4. Geliştirilen segmentasyon yönteminin ağırlık ölçümlerine göre doğruluk sonuçları

Ağırlık adı	Veri Seti					Ortalama Doğruluk	SS
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5		
W_1	0.7673	0.7579	0.8309	0.6434	0.8071	0.7613	0.0646
W_2	0.8563	0.8665	0.7913	0.6501	0.7556	0.7840	0.0785
W_3	0.8237	0.8552	0.7773	0.6444	0.8372	0.7876	0.0761
W_4	0.8258	0.8315	0.8057	0.6426	0.8341	0.7879	0.0733
W_5	0.7997	0.8828	0.7609	0.6732	0.7294	0.7692	0.0703
W_6	0.8338	0.8216	0.7700	0.5930	0.6985	0.7434	0.0890
W_7	0.8182	0.8437	0.8332	0.6578	0.8397	0.7985	0.0709
W_8	0.8431	0.8584	0.7838	0.6361	0.7538	0.7750	0.0793
W_9	0.7793	0.7575	0.8264	0.6381	0.8287	0.7660	0.0696
W_{10}	0.7944	0.8223	0.8405	0.7031	0.8131	0.7947	0.0481
W_{11}	0.7890	0.8307	0.8062	0.6485	0.8296	0.7808	0.0680
W_{12}	0.8056	0.7805	0.7885	0.6199	0.8032	0.7595	0.0704
W_{13}	0.7784	0.8036	0.8355	0.6417	0.8208	0.7760	0.0698
W_{14}	0.7926	0.8770	0.8707	0.6913	0.7967	0.8057	0.0673

Çizelge 5.5. Geliştirilen segmentasyon yönteminin ağırlık ölçümlerine göre genel $F1$ skor sonuçları

Ağırlık adı	Veri Seti					Ortalama $F1$	SS
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5		
W_1	0.7934	0.7071	0.8061	0.6147	0.4829	0.6808	0.1205
W_2	0.7504	0.7593	0.7491	0.6167	0.5679	0.6887	0.0803
W_3	0.8286	0.8137	0.8137	0.6272	0.5824	0.7331	0.1059
W_4	0.7956	0.7970	0.8214	0.6259	0.5755	0.7231	0.1016
W_5	0.7559	0.7703	0.7761	0.6284	0.5636	0.6989	0.0867
W_6	0.7135	0.7648	0.7014	0.5812	0.5125	0.6547	0.0931
W_7	0.8098	0.7800	0.8161	0.6253	0.5724	0.7207	0.1016
W_8	0.7155	0.7545	0.7227	0.6063	0.4917	0.6581	0.0971
W_9	0.8138	0.7390	0.8147	0.6263	0.5625	0.7113	0.1013
W_{10}	0.7761	0.7495	0.7880	0.6345	0.5092	0.6915	0.1061
W_{11}	0.8033	0.7721	0.8236	0.6314	0.5755	0.7212	0.0991
W_{12}	0.7923	0.7316	0.8072	0.6176	0.5533	0.7004	0.0993
W_{13}	0.8089	0.7716	0.8147	0.6250	0.5553	0.7151	0.1054
W_{14}	0.7540	0.7629	0.7928	0.6028	0.5364	0.6898	0.1012

Geliştirilen yöntem içerisinde önerilen ağırlık ölçümü olan W_{14} ağırlığı doğruluk karşılaştırmasında en iyi ortalama sonucu vermiştir. Bunun dışında veri setlerine göre en iyi doğruluk ve en iyi $F1$ skor sonucu veren τ_{max} parametre değerleri düşük standart sapma ile birbirine yakın olmakta ve bu da parametre değerinin uygulanan veriye göre daha kolay uyarlanmasını sağlamaktadır. Bundan dolayı, Bölüm 4.1.3'te yer alan kullanım gerekçeleriyle, tez çalışmasında geliştirilen yöntem içerisinde ağırlık ölçümü olarak W_{14} ağırlık ölçümü kullanılmıştır. Geliştirilen yöntemin W_{14} ağırlık ölçümünde farklı parametre değerleri kullanıldığında bütün veri setleri için alınan sonuçlarının renklendirilmiş olarak gösterimleri Ek 1'de yer almaktadır.

5.1.2. Geliştirilen yöntemin konumsal özellik kullanarak segmentasyon sonuçları ve literatür ile karşılaştırılması

Tez kapsamında geliştirilen yöntemin segmentasyon başarısının değerlendirilmesi için yöntemin doğruluk ve $F1$ skor sonuçları, literatürde yer alan RG, VIS, ORG, LCCP ve SVGS yöntemlerinin sonuçları ile karşılaştırılmıştır. Karşılaştırmada kullanılan bu yöntemlerde 3D nokta bulutundaki noktaların renk özellikleri kullanılmadığından, yöntemlerin karşılaştırılmasında noktaların sadece konum özellikleri kullanılmıştır. Ayrıca nokta bulutundaki nesnelerde kenar ve köşelerde bulunan voksellerin doğru segment edilebilmeleri için tez kapsamında geliştirilen refitting (yeniden düzlem

uydurma) işleminin etkisinin değerlendirilmesi için de karşılaştırma işlemi, refitting işlemini içerip içermemesine göre “refitting olmadan” ve “refitting ile” diye ayrı ayrı değerlendirilmiştir.

Karşılaştırmada RG yöntemindeki arama yarıçapı ile VIS, ORG, LCCP, SVGS ve geliştirilen yöntemdeki vokselsel boyutu, veri setine göre aynı değerler ile uygulanmıştır (Çizelge 5.6). Vokselsel boyutları, hem voksellerin içerisinde kalan noktaların lokal yüzey bilgisi verebilmesine hem de nesne ve yapıların amaçlanan detaylarını belirleyebilmesine imkan sağlaması gözetilerek belirlenmiştir. VIS ve ORG yöntemlerinde kullanılan düzlemsellik tolerans değerleri, yüzeydeki noktaların uydurulan düzlemsel yapıya göre sapmalarına ve gürültü oranına göre değişkenlik göstermektedir. Daha uzaktan taranan nokta bulutları daha fazla nokta sapmasına ve gürültü oranına sahip olmaktadır. RG, VIS, ORG ve geliştirilen yöntemin açısal eşik değerini belirleyen parametreleri $[5-75]^\circ$ aralığındaki 5° ’lik eşit aralıklara sahip değerler ile test edilmiştir. LCCP yönteminin dışbükeylik eşiğini belirleyen parametresi 2’şerli artışlarla $[2, 30]$ aralığında test edilirken, SVGS yönteminin kullandığı graf kesme eşiğini belirleyen parametresi $[0.1-0.8]$ aralığındaki 0.05 miktarında eşit aralıklara sahip değerler üzerinden test edilmiştir. RG yönteminin en yakın komşu sayısı parametre değeri Örnek 1 veri setindeki noktaların yoğunluk miktarındaki yükseklik nedeniyle Örnek 1 veri seti için 100, diğer veri setleri için ise kaynak kodunda varsayılan değer olan 30 olarak belirlenmiştir.

Karşılaştırmada metotlar için kullanılan parametre değerleri, literatürdeki çalışmalardan ve yapılan deneysel çalışmalar sonucunda en iyi segmentasyon sonuçlarını sağlayacak şekilde belirlenmiştir (Rabbani ve ark., 2006; Wang ve Tseng, 2011; Xu ve ark., 2017; Xu ve ark., 2018c). Yapılan deneysel çalışmalarda parametrelerin farklı kombinasyonlarının başarı sonuçlarına bakılarak parametre değerleri belirlenmiştir. Karşılaştırılan her yöntem için yöntemdeki bir parametrenin değeri, belirli bir aralık içinde belirlenen miktarda eşit aralıklara sahip farklı değerler test edilerek, doğruluk ve $F1$ skor ölçüm sonuçları elde edilmiştir. Karşılaştırma aşamasında yöntemin belirlenen parametresi için en iyi sonucu veren parametre değerinin doğruluk ve $F1$ skor sonuçları kullanılmıştır. Test edilen yöntemlerin, karşılaştırmada kullanılmak üzere tez çalışmasında belirlenen parametre değerleri Çizelge 5.6’da görülmektedir.

Çizelge 5.6. Karşılaştırılan yöntemlerinin test için kullanılan parametre değerleri

Yöntem	Parametre / Veri seti	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
RG	Arama yarıçapı	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	En yakın komşu sayısı	100	30	30	30	30
	Açısal eşik (Doğruluk)	70°	20°	60°	70°	20°
	Açısal eşik ($F1$ skor)	65°	25°	40°	65°	40°
VIS	Voksel boyutu	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	Düzlemsellik	0.004	0.005	0.01	0.02	0.02
	Ortak düzlemsellik	0.008	0.01	0.02	0.03	0.03
	Açısal eşik (Doğruluk)	45°	5°	25°	25°	20°
	Açısal eşik ($F1$ skor)	30°	15°	20°	15°	20°
ORG	Voksel boyutu	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	Düzlemsellik	0.004	0.005	0.01	0.02	0.02
	Açısal eşik (Doğruluk)	35°	35°	10°	35°	25°
	Açısal eşik ($F1$ skor)	30°	30°	15°	10°	15°
LCCP	Voksel boyutu	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	Beslenme yarıçapı	0.03 m	0.06 m	0.09 m	0.25 m	0.25 m
	Dışbükeylik (Doğruluk)	12	18	10	10	10
	Dışbükeylik ($F1$ skor)	10	18	10	12	12
SVGS	Voksel boyutu	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	Beslenme yarıçapı	0.03 m	0.06 m	0.09 m	0.25 m	0.25 m
	Graf büyüklüğü	0.06 m	0.12	0.18 m	0.5 m	0.5 m
	Kesme eşiği (Doğruluk)	0.5	0.3	0.5	0.35	0.35
	Kesme eşiği ($F1$ skor)	0.5	0.2	0.45	0.35	0.35
Geliştirilen Yöntem (refitting olmadan)	Voksel boyutu	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	τ_{max} (Doğruluk)	35°	40°	35°	20°	25°
	τ_{max} ($F1$ skor)	25°	40°	20°	20°	15°
Geliştirilen Yöntem (refitting ile)*	Voksel boyutu	0.015 m	0.03 m	0.045 m	0.1 m	0.1 m
	τ_{max} (Doğruluk)	35°	30°	20°	20°	20°
	τ_{max} ($F1$ skor)	25°	30°	15°	20°	15°

* Refitting ara işlemindeki c parametre değeri tüm veri setleri için 0.05 olarak kullanılmıştır.

Çizelge 5.6 aynı zamanda, her yöntem için belirli aralıkta farklı değerler ile test edilen ilgili parametrenin en iyi doğruluk ve en iyi $F1$ skor sonuçlarını veren değerlerini de göstermektedir. Geliştirilen yöntemin refitting işlemi ile uygulanmış versiyonunda refitting işlemi için gerekli c parametresi için bütün veri setlerinde 0.05 değeri kullanılmıştır.

Karşılaştırılan yöntemlerin Çizelge 5.6'daki parametre değerlerine göre doğruluk sonuçları Çizelge 5.7'de ve $F1$ skor sonuçları Çizelge 5.8'de veri setine göre ortalamaları ve standart sapmaları (SS) ile birlikte verilmektedir. Çizelgelerdeki sonuçlara bakıldığında, geliştirilen yöntemin yüksek doğruluk ve $F1$ skor değerleri ile birlikte düşük standart sapma değerleri ile en iyi segmentasyon başarısını sağladığı görülmektedir. Ek 2'de yöntemlerin en iyi doğruluk ve Ek 3'de en iyi $F1$ skor sonucu veren parametre değerlerinin renklendirilmiş görsel sonuçları gösterilmektedir. Şekil 5.1'de ise yöntemlerin, farklı değerleri ile test edilen parametre değerlerine göre grafiksel olarak karşılaştırılmaları görülmektedir.

Çizelge 5.7. Karşılaştırılan yöntemlerin veri setlerine göre doğruluk sonuçları

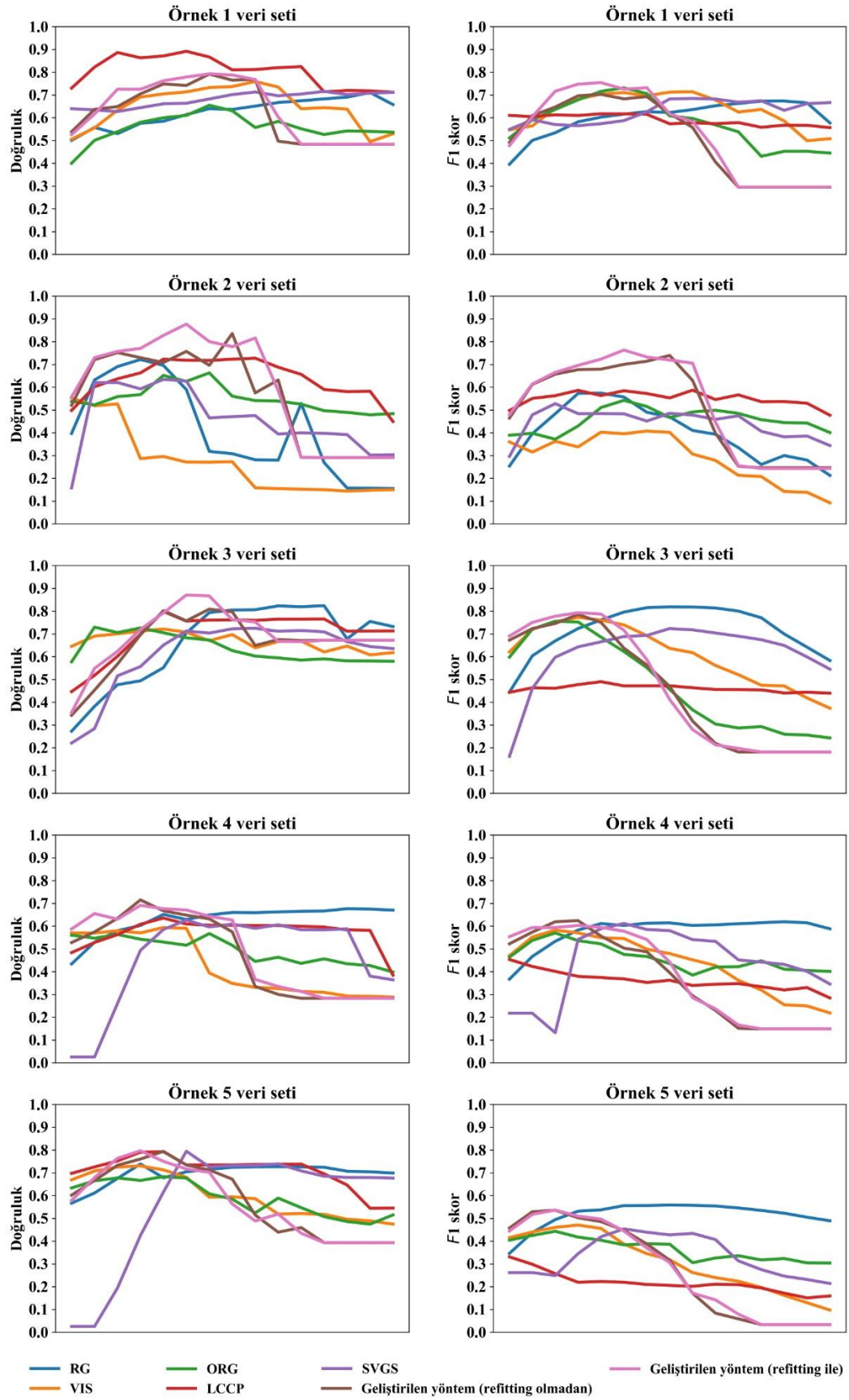
Yöntem	Veri Seti					Ortalama Doğruluk	SS
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5		
RG	0.7107	0.7215	0.8236	0.6747	0.7388	0.7339	0.0495
VIS	0.7589	0.5526	0.7213	0.594	0.7302	0.6714	0.0821
ORG	0.6546	0.6629	0.7296	0.5673	0.6831	0.6595	0.0529
LCCP	0.8916	0.7274	0.7984	0.6355	0.7928	0.7691	0.0849
SVGS	0.7129	0.6343	0.7245	0.6266	0.7948	0.6986	0.0624
Geliştirilen yöntem (refitting olmadan)	0.7915	0.8354	0.8087	0.7151	0.7943	0.7890	0.0401
Geliştirilen yöntem (refitting ile)	0.7926	0.877	0.8707	0.6913	0.7967	0.8057	0.0673

Çizelge 5.8. Karşılaştırılan yöntemlerin veri setlerine göre F1 skor sonuçları

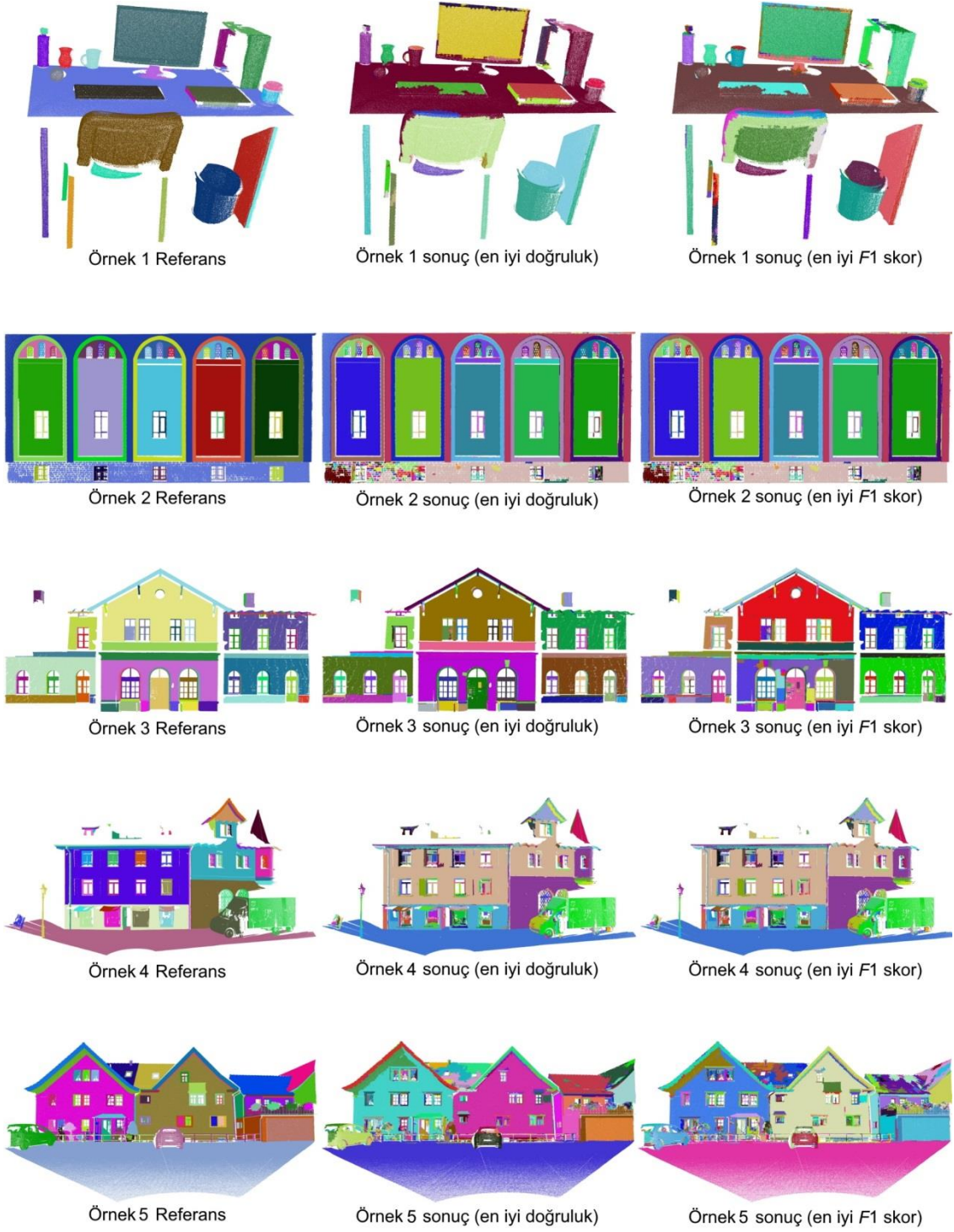
Yöntem	Veri Seti					Ortalama F1 skor	SS
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5		
RG	0.6734	0.5741	0.8187	0.6194	0.5586	0.6488	0.0939
VIS	0.7104	0.4074	0.7724	0.5827	0.4706	0.5887	0.1381
ORG	0.7312	0.5421	0.7558	0.57	0.4434	0.6085	0.1182
LCCP	0.6172	0.5869	0.4901	0.4534	0.3311	0.4957	0.1020
SVGS	0.6848	0.5278	0.7235	0.6117	0.4539	0.6003	0.0991
Geliştirilen yöntem (refitting olmadan)	0.7025	0.7394	0.7854	0.6241	0.5356	0.6774	0.0884
Geliştirilen yöntem (refitting ile)	0.754	0.7629	0.7928	0.6028	0.5364	0.6898	0.1012

Geliştirilen segmentasyon yönteminin veri setleri üzerindeki en iyi doğruluk ve F1 skor parametre değerleri ile verdiği segmentasyon sonuçları Şekil 5.2’de gösterilmektedir.

Karşılaştırılan yöntemlerinin, segmentasyon sonucundaki başarı değerlerinin yanında segmentasyon işlemini gerçekleştirdiği işlem süresinin azlığı da bir diğer başarı ölçütüdür. Çizelge 5.9 ve Çizelge 5.10’da en iyi doğruluk ve F1 skor sonucunu veren parametre değerleri ile karşılaştırılan segmentasyon yöntemlerinin işlem süreleri (dosyadan nokta bulutu verilerinin okunması hariç tüm aşamaları içeren) saniye (sn) olarak gösterilmektedir. Sürelerin geneline bakıldığında, geliştirilen yöntem segmentasyon işlemini diğer yöntemlere kıyasla hızlı bir şekilde gerçekleştirmektedir. Geliştirilen yöntem içerisinde refitting ara işlemi gerçekleşmediği zaman tüm veri setlerinde en hızlı çalışma performansını göstermektedir. Örnek 1, Örnek 4 ve Örnek 5 veri setlerinde refitting ara işlemi kullanıldığında çalışma hızında az da olsa bir düşüş olmakla beraber, refitting kullanıldığında da en hızlı olanlara yakın olmaktadır.



Şekil 5.1. Karşılaştırılan segmentasyon yöntemlerinin farklı parametre değerlerine ve veri setlerine göre grafiksel olarak karşılaştırılması



Şekil 5.2. Geliştirilen segmentasyon yönteminin veri setleri üzerindeki en iyi doğruluk ve $F1$ skor parametre değerleri ile verdiği segmentasyon sonuçları

Çizelge 5.9. Karşılaştırılan yöntemlerin veri setlerine göre en iyi doğruluk sonucunu veren parametre değerleri ile çalışma süreleri (sn)

	Veri Setleri				
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
Yöntem /Nokta Sayıları	382,217	3,069,150	2,742,237	3,306,260	2,350,862
RG (nokta seviye)	7.270	54.432	31.887	268.779	111.197
VIS (voksel seviye)	0.751	13.383	10.451	12.353	8.256
ORG (voksel seviye)	0.105	2.948	1.267	1.033	1.013
LCCP (voksel seviye)	0.106	2.756	1.196	0.784	0.814
SVGS (voksel seviye)	0.313	7.981	3.578	2.379	2.644
Geliştirilen yöntem (refitting olmadan) (voksel seviye)	0.078	2.429	1.007	0.503	0.566
Geliştirilen yöntem (refitting ile) (voksel seviye)	0.205	2.550	1.153	0.796	0.796

Çizelge 5.10. Karşılaştırılan yöntemlerin veri setlerine göre en iyi *F1* skor sonucunu veren parametre değerleri ile çalışma süreleri (sn)

	Veri Setleri				
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
Yöntem / Nokta sayısı	382,217	3,069,150	2,742,237	3,306,260	2,350,862
RG (nokta seviye)	7.334	49.002	46.897	269.225	80.743
VIS (voksel seviye)	0.553	21.274	8.423	9.234	8.256
ORG (voksel seviye)	0.104	2.954	1.272	1.037	1.003
LCCP (voksel seviye)	0.107	2.756	1.196	0.786	0.819
SVGS (voksel seviye)	0.313	8.041	3.524	2.379	2.644
Geliştirilen yöntem (refitting olmadan) (voksel seviye)	0.079	2.429	0.990	0.503	0.557
Geliştirilen yöntem (refitting ile) (voksel seviye)	0.204	2.550	1.126	0.796	0.897

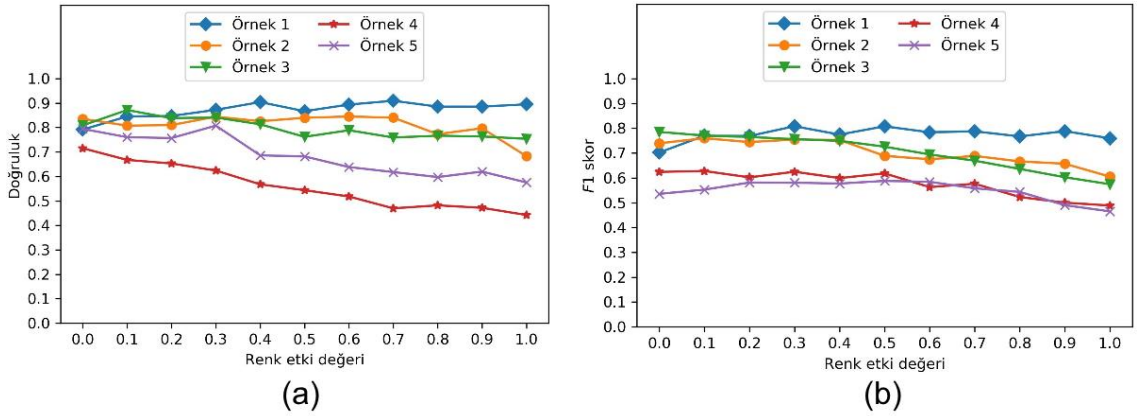
Çizelge 5.9 ve Çizelge 5.10'daki işlem süreleri incelendiğinde dikkat çeken bir diğer durum ise RG yönteminin diğer yöntemlere kıyasla segmentasyon işlemini çok uzun bir işlem süresi ile gerçekleştiriyor olmasıdır. RG yönteminin fazla işlem süresi gerektirmesi, en belirgin sebebi ile RG yönteminin segmentasyon işlemini voksel seviyesinde değil, nokta seviyesinde gerçekleştirmesinden kaynaklanmaktadır.

5.1.3. Geliştirilen yöntemde konum ve renk özelliği kullanarak segmentasyon sonuçları

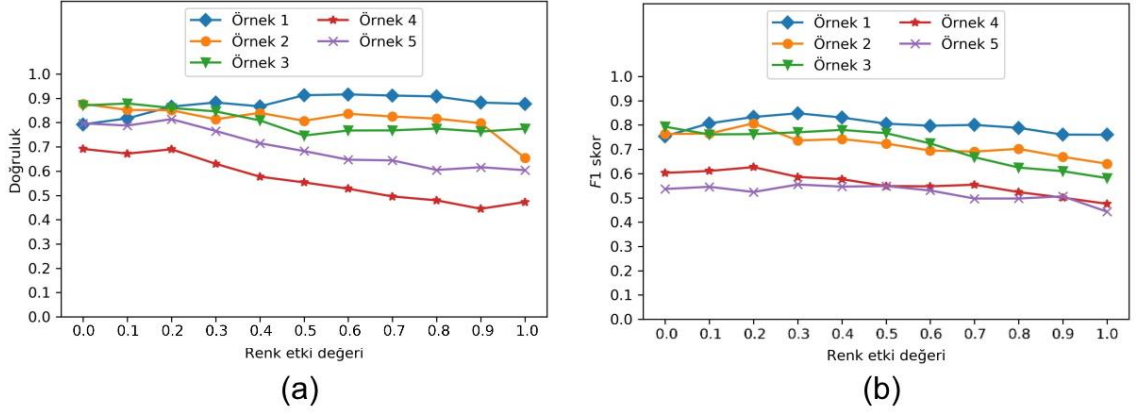
Geliştirilen yöntemin, sadece konum özellikleri kullanarak segmentasyon sonuçları elde edilip, literatürdeki yöntemlerle karşılaştırılmasından sonra, 3D nokta bulutundaki noktaların renk değerleri de dikkate alınarak segmentasyon işlemi gerçekleştirilmiştir. Bunun için, konumsal özelliklere ek olarak renk bilgisi de dikkate alınmış ve bağlantı ağırlık ölçümü Denklem 5.2'deki gibi güncellenmiştir. Denkleme, v_i ve v_j vokselleri arasındaki ağırlık değeri $w(v_i, v_j)$ hesaplanırken, voksellerin içerisindeki noktaların ortalama renk değerleri geometrik değer aralığına ([0-90]) ölçeklenmiş Öklid uzaklığı d_{renk} , renk etki oranı olan c_{renk_etki} kadar ağırlık değerine dahil olmaktadır. Renk etki oranı arttıkça ağırlık ölçümü olan W_{14} ile hesaplanan geometrik farklılığı $d_{geometrik}$ değerinin etki oranı düşmektedir.

$$w(v_i, v_j) = d_{renk} \cdot c_{renk_etki} + d_{geometrik} \cdot (1 - c_{renk_etki}) \quad (5.2)$$

Yapılan segmentasyon işlemlerinde renk değerinin etki oranının belirlenmesi için, geliştirilen yöntemin refitting olmadan ve refitting ile RGB renk uzayında farklı renk etkilerine göre segmentasyon başarıları test edilmiştir. Test işlemleri RGB renk uzayında yapılmıştır; çünkü alınan nokta bulutu verilerinde renk bilgisi RGB renk uzayında gelmektedir. RGB renk uzayında farklı renk etkilerine göre segmentasyon başarılarının refitting olmadan Şekil 5.3 (a), (b)) ve refitting ile elde edilen (Şekil 5.4 (a), (b)) doğruluk ve F1 skor değerleri grafiksel olarak gösterilmektedir.



Şekil 5.3. Geliştirilen yöntemin (refitting olmadan) farklı renk etki değerleri ile RGB uzayındaki renk katkısı ile doğruluk (a) ve F1 skor (b) değerlerinin grafiksel olarak gösterimi



Şekil 5.4. Geliştirilen yöntemin (refitting ile) farklı renk etki değerleri ile RGB uzayındaki renk katkısı ile doğruluk (a) ve $F1$ skor (b) değerlerinin grafiksel olarak gösterimi

Şekil 5.3 ve Şekil 5.4'te Örnek 2 ve Örnek 5 veri setlerinde bazı etki oranlarında ufak bir artış olsa da genel olarak renk etkisi uygulandığında başarı sonucu düşmüştür. İç mekândan alınarak hazırlanan Örnek 1 veri setinde ise başarı renk etkisi kullanıldığında genel olarak artmıştır. Bu durum, Örnek 1 veri setindeki renk kalitesinin ve nesneler arasındaki renk ayırt ediciliğinin yüksek olduğunu göstermektedir.

Yapılan işlemler sonucunda test edilecek renk etki değeri (c_{renk_etki}) 0.5 olarak belirlenmiştir. Bu değer kullanılarak, renk etkisinin veri setlerindeki segmentasyon işlemine katkısı test edilmiştir. Bu test işleminde amaçlanan, veri setindeki renk kalitesinin ve segmentlenecek yüzeylerdeki renk ayırt ediciliğinin derecesini ölçmektir.. Renk etkisi oranı 0.5 olarak yani %50 oranında belirlendiğinde, geometrik farklılığın etkisi de 1'e tamlayan şekilde 0.5 yani %50 oranında olmaktadır. İlk olarak, vokseller içerisindeki noktaların RGB (Red-Green-Blue) (kırmızı, yeşil, mavi) renk değerlerinin ortalaması alınmış ve farklılık ölçümünde bu ortalama RGB değerlerinin Öklid uzaklıkları geometrik ölçümde olduğu gibi [0-90] aralığına ölçeklenmiştir. Ölçekleme aşamasında, en küçük ve en büyük RGB değerleri bulunarak en küçük değer 0 ve en büyük değer 90 olarak alınmıştır. Ancak en büyük ve en küçük değerler bulunurken, verideki genel renk değişimine göre fazla aykırı renk değerlerine sahip nesnelerin ölçeklemeyi etkilememesi için literatürdeki uygulamalar örnek alınarak, bütün Öklid farklılıkları sıralanmış, baş ve son %5'lik kısımdaki değerler hesaplama dâhil edilmemiştir (Shah, 2006).

Renk bilgisinin 0.5 etki oranında dikkate alınarak yapılan segmentasyon işleminin refitting olmadan elde edilen sonuçlarının doğruluk ve $F1$ skor değerleri Çizelge 5.11 ve Çizelge 5.12'de; refitting ile elde edilen sonuçların doğruluk ve $F1$ skor değerleri Çizelge

5.13 ve Çizelge 5.14’te verilmiştir. Renk bilgisinin etkisinin görülebilmesi için çizelgelerde renk bilgisi olmadan ve renk bilgisi kullanıldığında elde edilen sonuçlar birlikte sunulmuştur.

Çizelge 5.11. Geliştirilen yöntemin (refitting olmadan) renk etkisi olmadan ve 0.5 renk etki oranı kullanılarak gerçekleştirdiği segmentasyonun doğruluk sonuçları

	İç mekân veri	Veri Seti			
		Dış mekân veriler			
Yöntem	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
Geliştirilen Yöntem (renk olmadan)	0.7915	0.8354	0.8087	0.7151	0.7943
Geliştirilen Yöntem (RGB, $c_{renk_etki} = 0.5$)	0.8667	0.8449	0.7617	0.5436	0.6592
Geliştirilen Yöntem (HSV, $c_{renk_etki} = 0.5$)	0.7502	0.6151	0.7285	0.5986	0.6887
Geliştirilen Yöntem (L*a*b*, $c_{renk_etki} = 0.5$)	0.9077	0.7151	0.6905	0.5764	0.6518

Çizelge 5.12. Geliştirilen yöntemin (refitting olmadan) renk etkisi olmadan ve 0.5 renk etki oranı kullanılarak gerçekleştirdiği segmentasyonun F1 skor sonuçları

	İç mekân veri	Veri Seti			
		Dış mekân veriler			
Yöntem	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
Geliştirilen Yöntem (renk olmadan)	0.7025	0.7394	0.7854	0.6241	0.5356
Geliştirilen Yöntem (RGB, $c_{renk_etki} = 0.5$)	0.8079	0.6888	0.7259	0.6182	0.5879
Geliştirilen Yöntem (HSV, $c_{renk_etki} = 0.5$)	0.7475	0.6373	0.6901	0.5560	0.5234
Geliştirilen Yöntem (L*a*b*, $c_{renk_etki} = 0.5$)	0.8162	0.6618	0.7135	0.5728	0.5958

Çizelge 5.13. Geliştirilen yöntemin (refitting ile) renk etkisi olmadan ve 0.5 renk etki oranı kullanılarak gerçekleştirdiği segmentasyonun doğruluk sonuçları

	İç mekân veri	Veri Seti			
		Dış mekân veriler			
Yöntem	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
Geliştirilen Yöntem (renk olmadan)	0.7926	0.8770	0.8707	0.6913	0.7967
Geliştirilen Yöntem (RGB, $c_{renk_etki} = 0.5$)	0.9126	0.8071	0.7465	0.5535	0.6827
Geliştirilen Yöntem (HSV, $c_{renk_etki} = 0.5$)	0.7551	0.6733	0.6881	0.5693	0.6677
Geliştirilen Yöntem (L*a*b*, $c_{renk_etki} = 0.5$)	0.8942	0.7042	0.6851	0.5716	0.6245

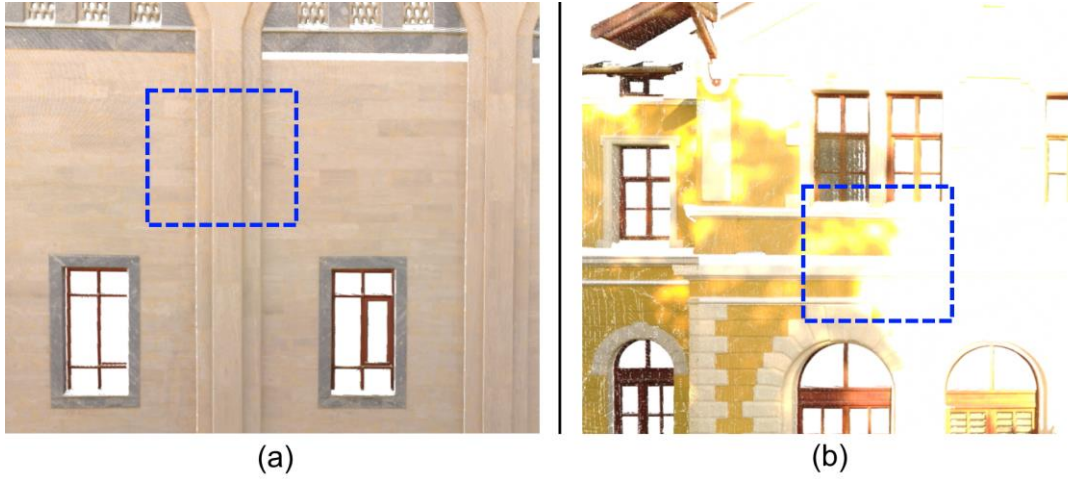
Çizelge 5.14. Geliştirilen yöntemin (refitting ile) renk etkisi olmadan ve 0.5 renk etki oranı kullanılarak gerçekleştirdiği segmentasyonun $F1$ skor sonuçları

Yöntem	İç mekân veri	Veri Seti			
		Dış mekân veriler			
	Örnek 1	Örnek 2	Örnek 3	Örnek 4	Örnek 5
Geliştirilen Yöntem (renk olmadan)	0.7540	0.7629	0.7928	0.6028	0.5364
Geliştirilen Yöntem (RGB, $c_{renk_etki} = 0.5$)	0.8056	0.7234	0.7664	0.5484	0.5485
Geliştirilen Yöntem (HSV, $c_{renk_etki} = 0.5$)	0.7306	0.6149	0.6941	0.5196	0.5103
Geliştirilen Yöntem ($L^*a^*b^*$, $c_{renk_etki} = 0.5$)	0.8408	0.6850	0.7218	0.5724	0.5719

Çizelgeler incelendiğinde, dış mekândan (Örnek 2-5) alınan verilerdeki renk bilgisi gün ışığından ve yansılardan dolayı segmentasyon işlemine dikkate değer bir katkı yapmadığı gibi çoğunlukla başarıyı da düşürmüştür. Bunun dışında, binalardaki kısımlar arasındaki renk farklılıkları, amaçlanan segmentler için yeterli ayırt edici özelliği sağlamamaktadır. Farklı yüzeyler aynı renkte olabilirken (Şekil 5.5 (a)), aynı yüzeye ait noktalar gölge ve lekeler gibi harici sebeplerden farklı renk değerlerine sahip olabilmektedirler (Şekil 5.5 (b)). Diğer taraftan, iç mekândan alınan Örnek 1 veri setindeki renk kalitesi yüksek olduğundan, sonuç segmentler arasındaki renk farklılıkları yüksek derecede ayırt edici özellik sağlamaktadır.

Segmentasyon yönteminde renk bilgisi kullanıldığında dış mekândan alınmış olan veri setlerinin segmentasyon başarılarının çoğunlukla artmadığı görülmüştür. Bu nedenle, renk bilgisi kullanıldığında segmentasyon başarısı artan iç mekân Örnek 1 veri seti için farklı renk etki oranları ile farklı renk uzaylarındaki segmentasyon başarı sonuçları test edilmiştir. Çizelge 5.15'te geliştirilen yöntemin refitting ile farklı renk etkisi oranlarında RGB, HSV ve $L^*a^*b^*$ renk uzayındaki segmentasyon sonuçları görülmektedir. Çizelge 5.15'te görüldüğü gibi, RGB uzayındaki renk değerleri 0.6 ile en iyi doğruluk sonucunu verirken $L^*a^*b^*$ uzayındaki renk değerleri 0.3 etki oranı ile en iyi $F1$ skor sonucunu vermektedir.

Şekil 5.6 (a)'da Örnek 1 veri seti üzerinde refitting ile en iyi doğruluk sonucunu veren 0.6 RGB renk etkisinin segmentasyon çıktısı ve Şekil 5.6 (b)'de refitting ile en iyi $F1$ skor sonucunu veren 0.3 $L^*a^*b^*$ renk etkisinin segmentasyon çıktısı yer almaktadır.

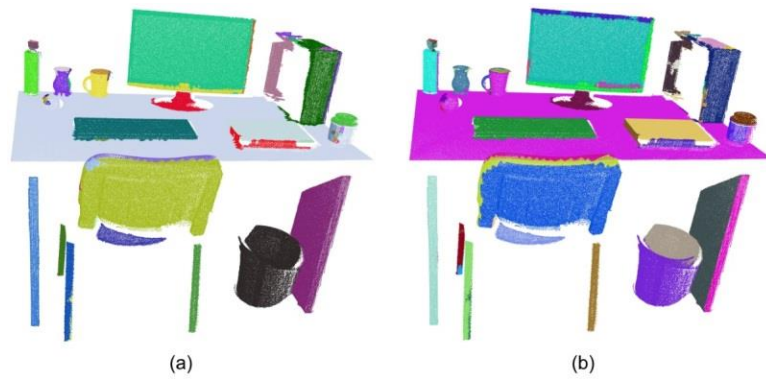


Şekil 5.5. Dış mekân verilerindeki yüzeyler arasındaki ayırt edici olmayan renk farklılığı (a) ve yansımadan dolayı aynı yüzey içerisinde gerçekleşen renk geçişi (b) örnekleri

Çizelge 5.15. Geliştirilen segmentasyon yönteminin (refitting ile) farklı renk etkisi oranlarında ve farklı renk uzayındaki renk değerleri için Örnek 1 veri seti için verdiği başarı sonuçları

Renk etki oranı	Doğruluk			F1 skor		
	Renk Uzayı			Renk Uzayı		
	RGB	HSV	L*a*b*	RGB	HSV	L*a*b*
0	0.7926	0.7926	0.7926	0.754	0.754	0.754
0.1	0.8173	0.8176	0.8646	0.8068	0.7721	0.8114
0.2	0.8658	0.7859	0.8386	0.8330	0.7450	0.8282
0.3	0.8827	0.7908	0.8679	0.8483	0.7388	0.8511**
0.4	0.8667	0.7944	0.8944	0.8307	0.7589	0.8332
0.5	0.9126	0.7551	0.8942	0.8056	0.7306	0.8408
0.6	0.9159*	0.7445	0.8431	0.7970	0.7266	0.8428
0.7	0.9113	0.7243	0.8595	0.8003	0.6934	0.8451
0.8	0.9076	0.7444	0.8872	0.7883	0.6773	0.7819
0.9	0.8824	0.7466	0.8256	0.7604	0.6713	0.7908
1.0	0.8772	0.7440	0.8850	0.7598	0.6705	0.7809

*: En iyi doğruluk sonucu, **: En iyi F1 skor sonucu



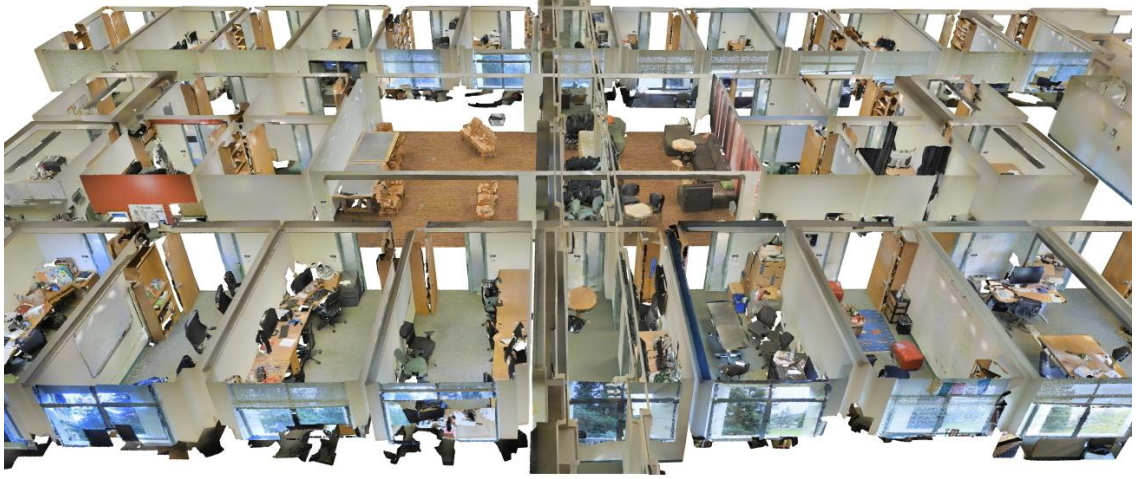
Şekil 5.6. En iyi doğruluk sonucunu veren 0.6 RGB renk etkisi (a) ve en iyi F1 skor sonucunu veren 0.3 L*a*b* renk etkisi ile Örnek 1 segmentasyon çıktıları

5.2. 3D Nokta Bulutu Sınıflandırma Sonuçları

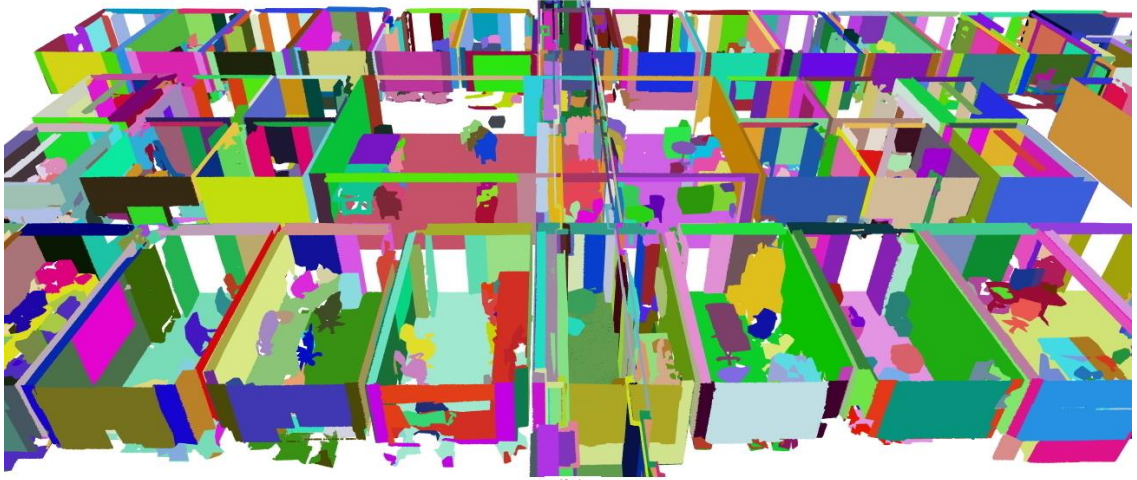
Geliştirilen yöntem ile 3D nokta bulutu verilerinde segmentasyon yaparak sonuçlar elde edildikten sonra, nokta bulutu verilerinde sınıflandırma yapıldığında ara işlem olarak kullanılmasında sınıflandırma başarılarına etkisi de incelenmiştir.

Sınıflandırıcı olarak test edilen DVM (Destek Vektör Makinesi) (SVM – Support Vector Machines) ve RO (Rastgele Orman) (RF – Random Forest) sınıflandırıcılarına ait parametreler, literatür incelenerek ve en iyi sonucu verecek şekilde farklı değerler test edilerek belirlenmiştir. DVM sınıflandırıcısı $C_{DVM} = 7$ ve RBF çekirdeği ile kullanılmıştır. RBF kernelindeki eğrilik derecesini belirleyen Gamma değeri için ise scikit-learn kütüphanesindeki özellik sayısını ve veri değerlerinin varyansını kullanarak otomatik hesaplama yapan ‘scale’ özelliği kullanılmıştır. RO sınıflandırıcısındaki ağaç sayısı da 200 olarak belirlenmiştir. Diğer parametreler kullanılan scikit-learn kütüphanesindeki varsayılan değerler ile kullanılmıştır. DVM sınıflandırmasında, eğitim ve test verisindeki özellik değerleri, her özellik için eğitim verisindeki minimum ve maksimum değerler kullanılarak [0-1] aralığına ölçeklenmiştir. Ayrıca, segmentlerdeki noktaların mevcut olan RGB renk değerlerinin ortalaması L*a*b* renk uzayında karşılık gelen değerlere dönüştürülmüştür.

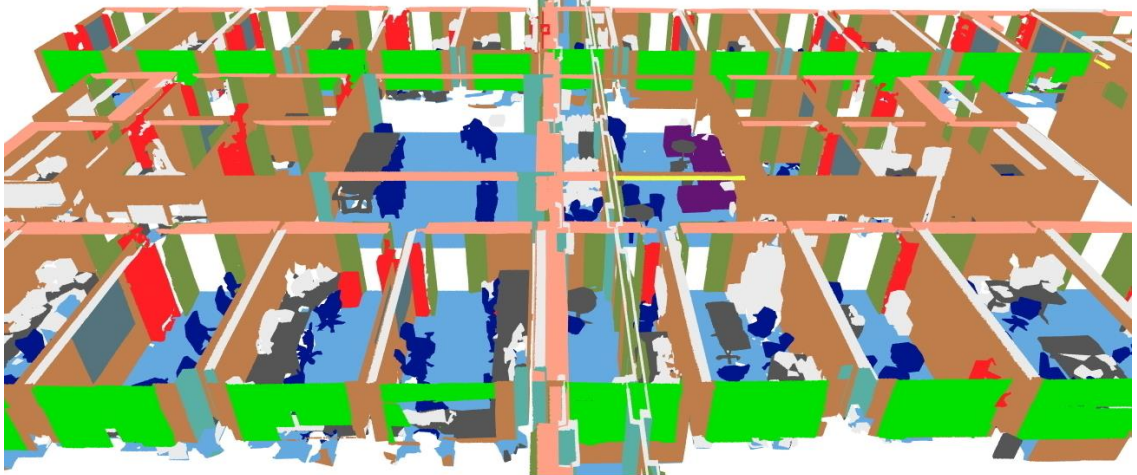
Tez çalışmasındaki sınıflandırma çalışmalarında, Bölüm 3.4.3’te verilmiş olan S3DIS veri seti kullanılmıştır. Veri setindeki Alan 1, Alan 2, Alan 3, Alan 4 ve Alan 5 isimli alanlar eğitim verisi olarak kullanılmış ve Alan 6 ise test verisi olarak kullanılmıştır. Şekil 5.7 (a)’da Alan 6’nın orijinal RGB değerleri, Şekil 5.7 (b)’de rastgele renklendirilmiş referans semantik segmentlerin renk değerleri ve Şekil 5.7 (c)’de tanımlanan sınıflar için atanmış özel renk değerleri ile renklendirilmiş noktalar görülmektedir. Şekilde büyük çoğunluğunu tavan sınıfının oluşturduğu alanı kapatan noktalar, odaların içerisindeki nesneleri göstermek amacıyla görsellerden kaldırılmıştır.



(a)



(b)



tavan zemin duvar kolon kiriş pencere kapı masa sandalye kitaplık kanepe yazı tahtası clutter

(c)

Şekil 5.7. S3DIS semantik segmentasyon veri setinde yer alan Alan 6'nın orijinal RGB renkli hali (a), rastgele renklendirilmiş referans semantik segmentleri (b) ve tanımlanan sınıflara göre etiketlenmiş noktalar (c)

Tez kapsamında ilk olarak, referans semantik segmentlerin tez kapsamında tanımlanmış olan özellik vektörleri kullanılarak DVM ve RO sınıflandırıcıları ile sınıflandırma yapılmış, böylece tanımlanan özelliklerin ve sınıflandırıcıların başarısı ölçülmüştür. Bu uygulamada, herhangi bir segmentasyon işlemi gerçekleştirilmemiştir. Çizelge 5.16'da, test verilerinin (Alan 6) doğru bir şekilde sınıflandırılmış referans semantik segmentlerin sayısının toplam semantik segment sayısına olan oranları yüzde olarak, hem DVM hem de RO sınıflandırıcılar için sınıf bazında ve genel olarak gösterilmektedir.

Çizelge 5.16. Test verisindeki (Alan 6) sınıfların referans semantik segmentlere göre sınıflandırılma başarılarının doğru sınıflandırılan segment sayısı üzerinden doğruluk değerleri

Sınıf adı	DVM	RO
tavan	0.9524	1.0000
zemin	1.0000	0.9796
duvar	0.9636	0.9960
kiriş	0.9265	0.9853
kolon	0.9630	1.0000
pencere	0.6129	0.7097
kapı	0.9785	0.9892
masa	0.8182	0.8571
sandalye	0.9888	0.9665
kanepe	0.1111	0.4444
kitaplık	0.6111	0.8778
yazı tahtası	0.9655	0.9310
Genel	0.9060	0.9515

Çizelge 5.16'ya bakıldığında, RO sınıflandırıcısının 12 sınıf içinden 9 sınıfta DVM sınıflandırıcısından daha iyi sonuç verdiği görülmektedir. Sınıflandırmada kullanılan S3DIS verisinde sınıflar (özellikle nesneler) çok karmaşıktır. Bu nedenle, % 90'ın üzerindeki doğruluk değerleri deneylerde iyi bir sonuç olarak görülüp, hem DVM hem de RO'nun, test verisindeki semantik segmentlerin genelinde başarı sağladığı kabul edilmiştir.

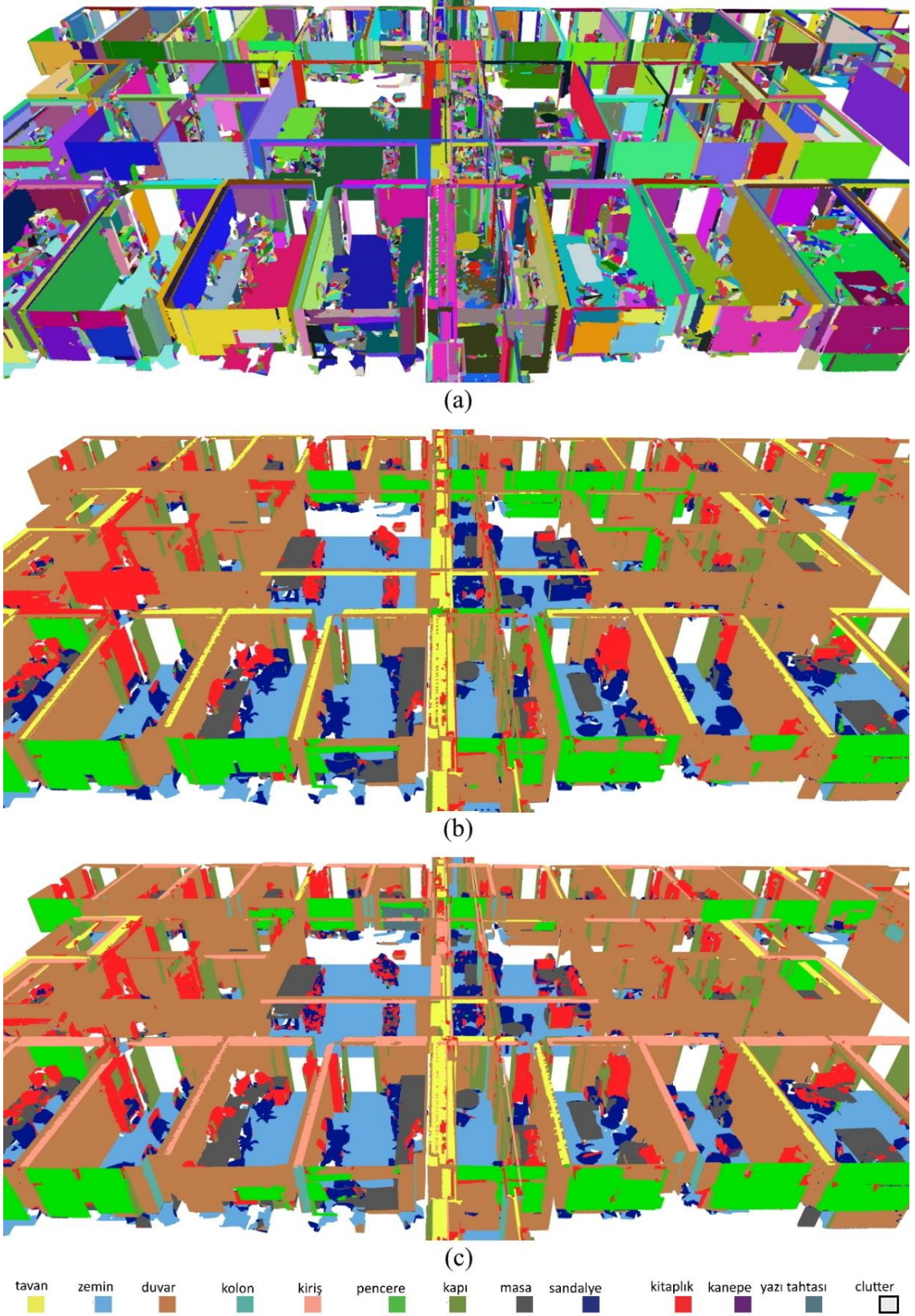
Referans semantik segmentlere ait özelliklerin DVM ve RO ile sınıflandırılıp, sonuçlar elde edildikten sonra; tez kapsamında geliştirilen yöntem ile segmentasyon yapılarak elde edilen özelliklerin DVM ve RO ile sınıflandırılması yapılmıştır. Bu amaçla yapılan uygulamalarda, voksel büyüklüğü parametresi S3DIS verisinin nokta yoğunluğu ve nesnelerdeki yakalanacak detay arasındaki denge gözetilerek 0.05 m olarak kullanılmıştır. Refitting aşamasındaki c parametre değeri için segmentasyon deneylerinde

kullanıldığı gibi yine 0.05 değeri kullanılmıştır. Geliştirilen segmentasyon yönteminin açısal parametresi olan τ_{max} $[5-45]^\circ$ aralığında 5° 'lik artışlarla test edilmiştir. Başarı ölçümü olarak ise segmentasyon karşılaştırmalarında kullanıldığı gibi doğruluk ve genel $F1$ skor ölçümü kullanılmıştır. Çizelge 5.17'de geliştirilen yöntem ile segmentasyon sonucunda elde edilen özellik vektörleri kullanılarak yapılan sınıflandırma işleminin, S3DIS veri setindeki farklı parametre ve sınıflandırıcılara göre başarı sonuçları görülmektedir. Çizelge 5.17'de τ_{max} segmentasyon parametresi 40° 'ye çıktığında veri artık segmentlenmemekte ve 45° 'ye geldiğinde benzer değerlerde çok düşük bir sınıflandırma başarısı vermektedir. Bu yüzden $[0-90]^\circ$ aralığında olabilen τ_{max} parametre değerinin 45° 'den sonraki değerleri test edilmemiştir

Çizelge 5.17. Geliştirilen yöntem ile yapılan segmentasyon sonucunda elde edilen özellik vektörleri ile yapılan sınıflandırma sonuçları

Segmentasyon parametresi (τ_{max})	Doğruluk		Genel $F1$ skor	
	DVM	RO	DVM	RO
5°	0.7580	0.8299	0.4813	0.6982
10°	0.6868	0.8243	0.4411	0.6660
15°	0.7000	0.7327	0.4426	0.6249
20°	0.6640	0.7502	0.4112	0.5889
25°	0.5540	0.6747	0.4074	0.5671
30°	0.5177	0.5048	0.3490	0.4579
35°	0.3936	0.4808	0.3150	0.4287
40°	0.2447	0.1474	0.2690	0.2415
45°	0.2049	0.1771	0.2188	0.2219

Çizelge 5.17'deki sonuçlara bakıldığında, aşırı segmentasyon (düşük τ_{max} parametresi) yapıldığında daha iyi başarı sonuçları elde edilmiştir. Diğer taraftan, RO sınıflandırıcısı hem doğruluk hem de genel $F1$ skor değeri açısından kullanılan S3DIS 3D semantik segmentasyon veri setinde DVM sınıflandırıcısından daha iyi sonuç vermiştir. Şekil 5.8 (a)'da, Alan 6'nın renklendirilmiş segmentasyon çıktısı, Şekil 5.8 (b)'de DVM ve Şekil 5.8 (c)'de RO sınıflandırıcıları ile yapılan en iyi sınıflandırmanın sınıf etiketlerine göre renklendirilmiş sınıflandırma sonuçları görülmektedir.



Şekil 5.8. Alan 6 ham verisinden elde edilmiş ve rastgele renklendirilmiş sonuç segmentleri ($\tau_{max}=5^\circ$) (a), tanımlanan sınıflara göre renklendirilmiş DVM (b) ve RO (c) sınıflandırıcılarının sonuçları

Çizelge 5.18’de S3DIS veri setindeki her sınıf için sınıflandırma işleminde en iyi sonuçların alındığı $\tau_{max}= 5^\circ$ için tahmin, geri çağırma ve $F1$ skor değerleri gösterilmektedir. Sonuçlara bakıldığında DVM sınıflandırıcısı “kiriş”, “kolon”, “kanepe” ve “yazı tahtası” sınıfları için bir sınıflandırma yapamamış, RO sınıflandırıcısı ise “kanepe” ve “yazı tahtası” sınıfları için çok düşük bir başarı sonucu vermiştir.

Çizelge 5.18. Sınıflandırıcıların, geliştirilen yöntemle segmentasyon ile sınıflandırma işlemindeki sonuçları (segmentasyon açılal parametresi olarak $\tau_{max}= 5^\circ$)

DVM				RO		
Sınıf adı	Tahmin	Geri çağırma	$F1$ skor	Tahmin	Geri çağırma	$F1$ skor
tavan	0.9073	0.9588	0.9323	0.9380	0.9617	0.9497
zemin	0.9876	0.9454	0.9660	0.9910	0.9614	0.9760
duvar	0.7122	0.8784	0.7866	0.7730	0.9225	0.8411
kiriş	0.0000	0.0000	0.0000	0.9176	0.4621	0.6146
kolon	0.0000	0.0000	0.0000	0.7709	0.3187	0.4510
pencere	0.5711	0.5116	0.5397	0.7961	0.5726	0.6661
kapı	0.7079	0.6204	0.6613	0.8355	0.6897	0.7556
masa	0.8802	0.6587	0.7535	0.8660	0.8017	0.8326
sandalye	0.4839	0.7535	0.5893	0.7160	0.7544	0.7347
kanepe	0.0000	0.0000	0.0000	0.8316	0.1630	0.2726
kitaplık	0.4068	0.5715	0.4753	0.5986	0.7339	0.6594
yazı tahtası	0.0000	0.0000	0.0000	0.6101	0.0635	0.1151
Ortalama	0.4714	0.4915	0.4753	0.8037	0.6171	0.6557

6. SONUÇLAR VE ÖNERİLER

6.1. Sonuçlar

Bir verideki veri elemanlarını uzamsal yakınlıklarına ve özniteliklerinin benzerliklerine göre gruplara ayıran segmentasyon işlemi, bilgisayarlı görme alanında oldukça öneme sahiptir. Bu işlem sınıflandırma, modelleme ve haritalama gibi birçok üst düzey uygulama içerisinde ara işlem olarak kullanılmaktadır. Çeşitli aşamalardan oluşan üst düzey görüntü işleme uygulamalarında, segmentasyon işlemi sonraki adımların sonuçlarını önemli derecede etkilemektedir. Gün geçtikçe kullanım alanı daha da genişleyen 3D Lidar nokta bulutu verilerinin segmentasyonu ile büyük miktarlarda ve düzensiz olarak elde edilen nokta bulutu verileri indirgenerek, işlenmesi kolay ve daha anlamlı hale dönüştürülmektedir. Yüksek boyutlarda elde edilen bu noktasal verilerin segmentasyon işlemi ile boyutunun azaltıldıktan sonraki uygulama aşamalarında, benzer özellikteki noktalar birlikte değerlendirilerek gereksiz veri temizlenmektedir. Veri miktarının azaltılmasının yanında, sınıflandırma gibi üst düzey işlemler için daha anlamlı ve ayırt edici özelliklere sahip özellikler elde edilmektedir.

Tez kapsamında bir yersel lazer tarayıcı kullanılarak, bir iç mekân ve iki dış mekân olmak üzere üç adet nokta bulutu verisi alınmıştır. Bu nokta bulutu verileri, tezde yer alan veri indirgeme ve gürültü giderme teknikleri ile düzenlenmiş ve segmentasyon karşılaştırması için uygun olan kısımları manuel olarak kırpılmıştır. Daha sonra bu verilere ait referans verileri (karşılaştırma için ground-truth verileri) manuel olarak hazırlanmış ve test için hazır hale getirilmiştir.

3D nokta bulutu segmentasyonun faydalarının yanında, doğru bir şekilde ve az işlem zamanında segmentasyon yapılabilmesi de bilgisayarlı görme alanında ayrıca bir problem olarak görülmektedir. Bu problemin çözümüne katkıda bulunulması amacıyla tez kapsamında, literatürde var olan 3D nokta bulutu segmentasyonu yöntemleri ve segmentasyon için gerekli temel nokta bulutu işleme araçları incelenmiş ve hem başarı hem de hız yönünden literatürdeki karşılaştırılan yöntemlere göre üstünlük sağlayan yeni bir segmentasyon yöntemi geliştirilmiştir. Geliştirilen yöntem, hem tez kapsamında temin edilen hem de dışarıdan temin edilen veriler üzerinde uygulanarak mevcut yöntemlerle karşılaştırılmıştır. Sonuçlar görsel olarak ve kullanılan nicel başarı ölçümleri (doğruluk ve $F1$ skor) ile sunulmuştur.

Tez kapsamında geliştirilen segmentasyon yöntemi, segmentlenecek ham nokta bulutu verilerini ilk önce octree organizasyonu ile hiyerarşik ve düzenli bir yapı içerisinde

örnekleyerek (vokselleyerek) hem işlenecek veri miktarını azaltmakta hem de lokal yüzey özelliklerinin elde edilmesini sağlayan nokta komşuluklarını (vokseller içerisindeki nokta grupları) belirlemektedir. Nokta komşulukları sayesinde lokal yüzeyler için TBA işlemi ile düzlemlerin yönleri (lokal yüzey normalleri) elde edilmektedir. Geliştirilen yöntem, geometriksel yüzey farklılıklarını hesaplamak için voksellerin normal ve ağırlık merkezi değerlerini kullanmakta ve bu değerler sayesinde iki bitişik yüzey arasındaki açısal farkları hesaplamaktadır. Bu açısal farklar kullanılarak ağırlık (fark) değeri elde eden bir ağırlıklandırma yöntemi de tez kapsamında önerilmiştir. Bununla beraber, kenar ve köşelerde kalmış düzlemsel olmayan vokseller (nokta grupları) için de düzlem özellikleri kazandıran bir ara işlem olan refitting (yeniden düzlem uydurma) yöntemi de tez kapsamında geliştirilmiştir. Geliştirilen segmentasyon yöntemi “refitting ile” ve “refitting olmadan” olarak ayrı ayrı test edilmiş ve başarı sonuçları alınmıştır. Geliştirilen yöntemin literatürde yer alan beş adet yöntem (RG, VIS, ORG, LCCP ve SVGS) ile karşılaştırılmasında, kıyaslanan bütün yöntemler noktaların konumsal özniteliklerini kullanarak geometriksel yüzey şekillerine göre segmentasyon yapmakta ve başarılar buna göre karşılaştırılmaktadır.

Sonuçlara bakıldığında geliştirilen segmentasyon yöntemi, kullanılan segmentasyon veri setleri üzerindeki ortalama segmentasyon başarısında hem refitting işlemi ile beraber uygulandığında hem de refitting işlemi olmadan uygulandığında doğruluk ve $F1$ skor olarak literatürde karşılaştırılan yöntemlere göre daha iyi sonuç vermiştir. Bununla beraber, refitting işleminin segmentasyon için başarıya etkisine bakıldığında, refitting uygulandığında işlem süresi az da olsa artmakta ancak buna rağmen refitting ile elde edilen 0.8057 doğruluk ve 0.6898 $F1$ skor ile refitting yapılmadan elde edilen başarının arttığı görülmektedir. Geliştirilen yöntem voksel boyutu parametresinin yanında refitting işlemi kullanılmadığında sadece bir parametre değerine (τ_{max}) ihtiyaç duymaktadır. Refitting ara işlemi kullanıldığında ise belirlenmesi gereken c parametre değerinin tez kapsamında kullanılan tüm veri setlerinde 0.05 olarak belirlenebileceği tez çalışması içerisinde grafiklerle gösterilmiştir. Bu yüzden, c parametresi için belirlenen 0.05 değeri, belirlenmesi gereken bir parametre değeri olarak değil; refitting işleminde kullanılan bir sabit olarak düşünülebilir.

Karşılaştırma sonuçlarına göre literatürdeki diğer yöntemlerin sonuçları değerlendirilecek olursa, RG yöntemi literatürdeki diğer yöntemlere göre yüksek $F1$ skor başarısı verse de nokta seviyesinde işlem yaptığı için diğer yöntemlere göre oldukça fazla işlem süresi harcamaktadır. LCCP yöntemi doğruluk değerlerinde geliştirilen yöntemin

sonuçlarına yakın olacak kadar yüksek başarı sağlasa da $F1$ skor değerlerinde diğer yöntemlere göre düşük başarı değeri vermiştir. Bu durum da LCCP yönteminin, kullanılan parametre değerleriyle büyük segmentlerin bölütlenmesinde başarılıken küçük segmentlerin bölütlenmesinde başarısız olduğunu göstermektedir. VIS ve ORG yöntemleri düzlemsel yapıdaki büyük segmentlerde gürültüye karşı oldukça dirençli olmasıyla birlikte sınırları korunması gereken bazı detayları yakalayamamaktadır. VIS ve ORG yöntemlerinde açısal eşik değerinin yanında belirlenmesi gereken ve belirli bir değer aralığında bulunmayan düzlemsellik parametresinin ayarlanması veriye göre oldukça hassasiyet göstermektedir. Aynı şekilde SVGS yönteminde de voksel boyutu ve beslenme yarıçapı bilgileri gerektiren süper voksel oluşturma ara işleminin parametre değerlerinin haricinde belirlenmesi gereken graf büyüklüğü ve kesme eşik değeri parametrelerinin dengeli bir şekilde belirlenmesi de uygulama kısmında problem olmaktadır.

Geliştirilen yöntem içerisinde, noktaların yalnızca konumsal değerlerinin kullanılmasının yanında, ayrıca renk değerleri de belirlenen etki oranlarında segmentasyona dâhil edilmiştir. Yapılan çalışmalarda, dış mekân verilerinde amaçlanan segmentlerin renk bilgisi, segmentasyon üzerinde başarıya katkı sağlamazken, iç mekândan alınmış nesne segmentasyonu için kullanılan veri setinde başarıyı artırmıştır. Tez kapsamında “Örnek 1” olarak isimlendirilen iç mekan nesne segmentasyonu veri setindeki doğruluk başarısı RGB renk uzayında 0.6 (%60) renk etkisi ile 0.7926’dan 0.9159’a, $F1$ skor başarısı ise $L^*a^*b^*$ renk uzayında 0.3 (%30) renk etkisi ile 0.754’ten 0.8511’e çıkarılmıştır.

Bunlara ek olarak, literatürden geniş ve karmaşık nesneler içeren iç mekân alanlarından taranarak elde edilmiş yüksek boyutlarda bir semantik segmentasyon veri seti (S3DIS) temin edilmiş ve geliştirilen yöntem bir ara aşama olarak kullanılarak bu verinin üzerinde sınıflandırma uygulaması yapılmıştır. S3DIS veri seti 6 adet alandan oluşmakta; tez çalışması kapsamında seçilen ilk 5 adet alan eğitim verisi olarak kullanılırken son alan ise test için ayrılmıştır. Uygulama sonuçlarına göre, kullanılan yöntem düşük segmentasyon açısal parametre değeri ($\tau_{max}=5^\circ$) ile aşırı segmentasyon yaptığı daha iyi sonuç vermiştir. Sınıflandırma aşamasında DVM ve RO sınıflandırıcıları ayrı ayrı denenmiştir. Noktasal olarak ayrıca bir işlem yapmadan doğrudan referans semantik segmentlerin tez kapsamında tanımlanan özellikleri kullanılarak sınıflandırma yapıldığında DVM sınıflandırıcısı 0.906 doğruluk başarısı verirken RO sınıflandırıcısı 0.9515 doğruluk başarısı vermiştir. Tez çalışmasındaki

sınıflandırma işlemlerinde asıl amaç olan ham nokta bulutundaki noktaların segmentasyon yardımı ile sınıflandırılmasında ise DVM sınıflandırıcısı 0.758 doğruluk ve 0.4813 *F1* skor başarısı verirken, RO sınıflandırıcısı 0.8299 doğruluk ve 0.6982 *F1* skor başarısı sağlamıştır.

6.2. Öneriler

Segmentasyon işlemi birçok gerçek hayat probleminin çözümü için geliştirilen 3D nokta bulutu uygulamalarında ara işlem olarak kullanılmakta ve istenilen sonuçlara ulaşmada önemli oranda etkili olmaktadır. Kentsel yapıların modellerinin oluşturulmasında ALS ve MLS ile elde edilen nokta bulutları içerisinde çatıların, binaların, yolların ve zeminin belirlenmesi işlemlerinde, haritacılıkla ilgili uygulamalarda segmentasyon çok kritik bir ara işlem olarak kullanılabilir. Çatıların, binaların ve bir bina içerisindeki duvar, zemin, pencere ve merdiven gibi yapıların 3D modellerinin oluşturulmasında da 3D nokta bulutu segmentasyonu çok önemli bir ara işlem olarak kullanılabilir. Bunların yanında, bir oda içerisindeki nesnelerin dijital ortamda anlamlandırılarak tespit edilmesinde segmentasyon işlemi ham nokta bulutu verisini daha anlamlı hale getirerek, yeni özellikler elde edilmesini sağlamaktadır. Endüstriyel çalışmalarda boru gibi belirli şekilsel özelliklere sahip yapıların tespit edilmesi ve modellenmesinde, segmentasyon işlemi üzerinde çok durulan bir problem olmuştur. Ormancılıkla ilgili arazi tespiti uygulamalarında ALS ve TLS ile elde edilen nokta bulutları içerisinde ağaçlık alanların belirlenmesinde de segmentasyon işlemi kullanılabilir. Bireysel olarak ağaç yapılarının belirlenerek belirli bir alandaki ağaç sayılarının tespit edilmesi de segmentasyon işleminin kullanım alanları arasında yer almaktadır. Ayrıca, yapılardan elde edilen 3D nokta bulutlarının segmentasyon işlemi sonrasında elde edilen segmentlerinin kültürel mirasın korunması amacıyla dijital ortamlara aktarılması ve analiz edilmesi de 3D nokta bulutu segmentasyonunun kullanım alanlarından birisidir.

Tez kapsamında, 3D Lidar nokta bulutu segmentasyonu üzerine yapılmış çalışmalar hakkında kapsamlı bilgiler karşılaştırmalı ve özet olarak verilmiştir. Bunların yanında yeni bir segmentasyon yöntemi geliştirilerek literatürde öne çıkan yöntemler ile karşılaştırılmıştır. Geliştirilen segmentasyon yöntemi içerisinde kullanılan temel nokta bulutu işleme teknikleri ve yöntemin aşamaları detaylı bir şekilde anlatılmıştır. Geliştirilen yöntem içerisinde noktaların öz nitelikleri farklı formlarda kullanılarak segmentasyon işlemine etkisi araştırılmış ve son olarak da daha üst düzey bir işlem olan

sınıflandırma işleminde ara işlem olarak kullanılmıştır. Geliştirilen yöntemin sonraki çalışmalarda farklı veriler üzerinde kullanımında, tez kapsamında kullanılmış veri setleri üzerindeki en iyi parametre değerleri ilk olarak denenerek, parametre ayarlama aşaması daha hızlı gerçekleştirilebilecektir.

Tez kapsamında elde edilen bilgiler bu konuda araştırma yapmak isteyen araştırmacılar için alt yapı niteliğinde bilgiler taşımaktadır. Yüzeylerin lokal özelliklerini kullanarak tez kapsamında geliştirilen segmentasyon yöntemini temel alarak, nokta bulutu segmentasyonu üzerine daha sonra yapılacak çalışmalarda yüzeylerin daha geniş alandaki farklı geometriksel özelliklerinin dikkate alınabileceği farklı yöntemler geliştirilebilir.

Geliştirilen segmentasyon yöntemi, nokta bulutu işleme alanındaki sınıflandırma, modelleme ve nesne algılama gibi farklı uygulamalarda ara işlem olarak kullanılarak; bu tip uygulamalar üzerinde geliştirme çalışmaları yapılabilir. Bilgisayarlı görme alanında birçok alanda kullanılan sınıflandırma işlemleri için segmentlerin kendi aralarındaki ilişkiler ve topolojik bağlantılar da kullanılarak daha başarılı sınıflandırma işlemleri gerçekleştirilebilir. Böylece sınıflandırma işleminin kullanıldığı nesne tespiti, haritalama, ormancılık, şehir planlama ve iç mekân analizi gibi gerçek hayat problemlerinin çözümü amacıyla yapılabilecek olan çalışmalarda da tez kapsamında geliştirilen yöntem temel alınarak farklı uygulamalar geliştirilebilir.

Nokta bulutunun doğrudan semantik segmentasyonunun yapılmasında, son zamanlarda kullanımı yaygınlaşmakta olan derin öğrenme yöntemleri de kullanılabilir. Çok çekirdekli grafik kartlarındaki gelişmelerden sonra 2D görüntülerde oldukça yaygınlaşmış olan CNN (Convolutional Neural Networks – konvolüsyonel sinir ağları), son zamanlarda 3D nokta bulutu üzerinde de semantik segmentasyon ve nesne tespiti amacıyla kullanılmaya başlanmıştır. 3D nokta bulutu üzerinde çalışan CNN algoritmaları, nokta veya voksel gibi farklı seviyelerde işlem yapabilmektedir. Ayrıca, nokta bulutunun 2D bir düzleme olan izdüşümünü kullanarak daha hızlı işlem yapan CNN algoritmaları da mevcuttur. Literatürde CNN ile semantik segmentasyon yapan çalışmalar arasında, CNN işlemini danışmansız segmentasyon işlemi ile birlikte kullanan yöntemler de mevcuttur. Derin öğrenme yöntemlerinin veriye göre parametre ayarı gerektirmeden, doğrudan referans verileri kullanarak model oluşturma gibi avantajları varken, eğitim için fazla işlem yükü gerektirmesi gibi dezavantajları da mevcuttur.

KAYNAKLAR

- Abdi, H. ve Williams, L. J., 2010, Principal component analysis, *Wiley Interdisciplinary Reviews-Computational Statistics*, 2 (4), 433-459.
- Aijazi, A. K., Checchin, P. ve Trassoudaine, L., 2013, Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation, *Remote Sensing*, 5 (4), 1624-1650.
- Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M. ve Savarese, S., 2016, 3D Semantic Parsing of Large-Scale Indoor Spaces, *2016 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr)*, 1534-1543.
- Arya, S. ve Mount, D. M., 1993, Algorithms for fast vector quantization, *[Proceedings] DCC '93: Data Compression Conference*, 381-390.
- Awrangjeb, M. ve Fraser, C. S., 2014, An Automatic and Threshold-Free Performance Evaluation System for Building Extraction Techniques From Airborne LIDAR Data, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7 (10), 4184-4198.
- Babahajiani, P., Fan, L. X., Kamarainen, J. K. ve Gabbouj, M., 2017, Urban 3D segmentation and modelling from street view images and LiDAR point clouds, *Machine Vision and Applications*, 28 (7), 679-694.
- Barnea, S. ve Filin, S., 2013, Segmentation of terrestrial laser scanning data using geometry and image information, *Isprs Journal of Photogrammetry and Remote Sensing*, 76, 33-48.
- Belgiu, M. ve Dragut, L., 2016, Random forest in remote sensing: A review of applications and future directions, *Isprs Journal of Photogrammetry and Remote Sensing*, 114, 24-31.
- Ben-Shabat, Y., Avraham, T., Lindenbaum, M. ve Fischer, A., 2018, Graph based over-segmentation methods for 3D point clouds, *Computer Vision and Image Understanding*, 174, 12-23.
- Bentley, J. L., 1975, Multidimensional binary search trees used for associative searching, *Communications of the ACM*, 18 (9), 509-517.
- Biosca, J. M. ve Lerma, J. L., 2008, Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods, *Isprs Journal of Photogrammetry and Remote Sensing*, 63 (1), 84-98.
- Blomley, R. d., Jutzi, B. ve Weinmann, M., 2016, 3D semantic labeling of ALS point clouds by exploiting multi-scale, multi-type neighborhoods for feature extraction. 6th GEOBIA Conference.
- Borrmann, D., Elseberg, J., Lingemann, K. ve Nuchter, A., 2011, The 3D Hough Transform for Plane Detection in Point Clouds: A Review and a new Accumulator Design, *3d Research*, 2 (2).
- Boulch, A., Houllier, S., Marlet, R. ve Tournaire, O., 2013, Semantizing Complex 3D Scenes using Constrained Attribute Grammars, *Computer Graphics Forum*, 32 (5), 33-42.
- Boyko, A. ve Funkhouser, T., 2011, Extracting roads from dense point clouds in large scale urban environment, *Isprs Journal of Photogrammetry and Remote Sensing*, 66 (6), 2-12.
- Boykov, Y. ve Funka-Lea, G., 2006, Graph cuts and efficient N-D image segmentation, *International Journal of Computer Vision*, 70 (2), 109-131.
- Breiman, L., 2001, Random forests, *Machine Learning*, 45 (1), 5-32.
- Burges, C. J. C., 1998, A tutorial on Support Vector Machines for pattern recognition, *Data Mining and Knowledge Discovery*, 2 (2), 121-167.

- Busuttill, S., Abela, J. ve Pace, G. J., 2004, Support vector machines with profile-based kernels for remote protein homology detection, *Genome Inform*, 15 (2), 191-200.
- Chen, D., Zhang, L. Q., Mathiopoulos, P. T. ve Huang, X. F., 2014, A Methodology for Automated Segmentation and Reconstruction of Urban 3-D Buildings from ALS Point Clouds, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7 (10), 4199-4217.
- Cheng, L., Wu, Y., Wang, Y., Zhong, L. S., Chen, Y. M. ve Li, M. C., 2015, Three-Dimensional Reconstruction of Large Multilayer Interchange Bridge Using Airborne LiDAR Data, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8 (2), 691-708.
- Clode, S., Kootsookos, P. ve Rottensteiner, F., 2004, The Automatic Extraction of Roads from LIDAR data, *The International Society for Photogrammetry and Remote Sensing's Twentieth Annual Congress*.
- Connor, M. ve Kumar, P., 2010, Fast Construction of k-Nearest Neighbor Graphs for Point Clouds, *Ieee Transactions on Visualization and Computer Graphics*, 16 (4), 599-608.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L. ve Stein, C., 2009, Introduction to algorithms, *Cambridge, Mass.*, MIT Press, p.
- Cristianini, N. ve Shawe-Taylor, J., 2000, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, *Cambridge*, Cambridge University Press, p.
- Duda, R. O., Hart, P. E. ve Stork, D. G., 2001, Pattern classification, *New York*, Wiley, p.
- Dybedal, J., Aalerud, A. ve Hovland, G., 2019, Embedded Processing and Compression of 3D Sensor Data for Large Scale Industrial Environments, *Sensors*, 19 (3).
- Ekerçin, S. ve Üstün, B., 2004, Uzaktan Algılamada Yeni Bir Teknoloji: Lidar, *Jeodezi ve Jeoinformasyon Dergisi*, 91, 34-38.
- Felzenszwalb, P. F. ve Huttenlocher, D. P., 2004, Efficient graph-based image segmentation, *International Journal of Computer Vision*, 59 (2), 167-181.
- Filin, S., 2002, Surface clustering from airborne laser scanning data, *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34, 119-124.
- Fischler, M. A. ve Bolles, R. C., 1981, Random Sample Consensus - a Paradigm for Model-Fitting with Applications to Image-Analysis and Automated Cartography, *Communications of the ACM*, 24 (6), 381-395.
- Friedman, J. H., Bentley, J. L. ve Finkel, R. A., 1977, An Algorithm for Finding Best Matches in Logarithmic Expected Time, *ACM Trans. Math. Softw.*, 3 (3), 209-226.
- Golovinskiy, A. ve Funkhouser, T., 2009, Min-cut based segmentation of point clouds. 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops 2009: 39-46.
- Gorte, B., 2002, Segmentation of TIN-structured surface models, *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*.
- Guindon, B., Zhang, Y. ve Dillabaugh, C., 2004, Landsat urban mapping based on a combined spectral-spatial methodology, *Remote Sensing of Environment*, 92 (2), 218-232.
- Haala, N., Peter, M., Kremer, J. ve Hunter, G., 2008, Mobile LiDAR mapping for 3D point cloud collection in urban areas—A performance test, *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*.
- Han, J., Kamber, M. ve Pei, J., 2012, Data Mining: Concepts and Techniques, 3rd Edition, *USA*, Elsevier, p.

- Hastie, T., Tibshirani, R. ve Friedman, J. H., 2009, The elements of statistical learning : data mining, inference, and prediction, *New York, NY, Springer*, p.
- Hulik, R., Spanel, M., Smrz, P. ve Materna, Z., 2014, Continuous plane detection in point-cloud data based on 3D Hough Transform, *Journal of Visual Communication and Image Representation*, 25 (1), 86-97.
- Info, F. P., 2015, FARO Laser Scanner Focus3D X 330 Features, Benefits & Technical Specifications,
<https://faro.app.box.com/s/8ilpeyxcuitnczqgsrgp5rx4a9lb3skq/file/441668110322>:
- Jain, A. K., Murty, M. N. ve Flynn, P. J., 1999, Data clustering: A review, *Acm Computing Surveys*, 31 (3), 264-323.
- Jain, A. K., 2010, Data clustering: 50 years beyond K-means, *Pattern Recognition Letters*, 31 (8), 651-666.
- Jamali, A., Anton, F., Rahman, A. A. ve Mioc, D., 2016, 3D Indoor Building Environment Reconstruction using Least Square Adjustment, Polynomial Kernel, Interval Analysis and Homotopy Continuation, *3rd International Geoadvances Workshop*, 42-2 (W1), 103-113.
- Jiang, Y. H. D., 1996, Set operations between linear octrees, *Computers & Geosciences*, 22 (5), 509-516.
- Jolliffe, I. T. ve Cadima, J., 2016, Principal component analysis: a review and recent developments, *Philosophical Transactions of the Royal Society a-Mathematical Physical and Engineering Sciences*, 374 (2065).
- Laefer, D. F. ve Pradhan, A. R., 2006, Evacuation route selection based on tree-based hazards using light detection and ranging and GIS, *Journal of Transportation Engineering*, 132 (4), 312-320.
- Lafarge, F. ve Mallet, C., 2012, Creating Large-Scale City Models from 3D-Point Clouds: A Robust Approach with Hybrid Representation, *International Journal of Computer Vision*, 99 (1), 69-85.
- Lari, Z. ve Habib, A., 2014, An adaptive approach for the segmentation and extraction of planar and linear/cylindrical features from laser scanning data, *Isprs Journal of Photogrammetry and Remote Sensing*, 93, 192-212.
- Lezoray, O. ve Charrier, C., 2009, Color image segmentation using morphological clustering and fusion with automatic scale selection, *Pattern Recognition Letters*, 30 (4), 397-406.
- Li, L., Yang, F., Zhu, H. H., Li, D. L., Li, Y. ve Tang, L., 2017, An Improved RANSAC for 3D Point Cloud Plane Segmentation Based on Normal Distribution Transformation Cells, *Remote Sensing*, 9 (5).
- Li, Z. Q., Zhang, L. Q., Tong, X. H., Du, B., Wang, Y. B., Zhang, L., Zhang, Z. X., Liu, H., Mei, J., Xing, X. Y. ve Mathiopoulos, P. T., 2016, A Three-Step Approach for TLS Point Cloud Classification, *Ieee Transactions on Geoscience and Remote Sensing*, 54 (9), 5412-5424.
- Lohmann, G., 1998, Volumetric image analysis, Wiley, p.
- Lozes, F., Elmoataz, A. ve Lezoray, O., 2015, PDE-Based Graph Signal Processing for 3-D Color Point Clouds Opportunities for cultural heritage, *Ieee Signal Processing Magazine*, 32 (4), 103-111.
- Marshall, D., Lukacs, G. ve Martin, R., 2001, Robust segmentation of primitives from range data in the presence of geometric degeneracy, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23 (3), 304-314.

- Mayr, A., Rutzinger, M., Bremer, M., Elberink, S. O., Stumpf, F. ve Geitner, C., 2017, Object-based classification of terrestrial laser scanning point clouds for landslide monitoring, *Photogrammetric Record*, 32 (160), 377-397.
- Morsdorf, F., Meier, E., Allgöwer, B. ve Nüesch, D., 2003, Clustering in airborne laser scanning raw data for segmentation of single trees, *ISPRS Working group III/3 Workshop 3-D Reconstruction from Airborne Laserscanner and InSAR Data*.
- Morsdorf, F., Meier, E., Kotz, B., Itten, K. I., Dobbartin, M. ve Allgower, B., 2004, LIDAR-based geometric reconstruction of boreal type forest stands at single tree level for forest and wildland fire management, *Remote Sensing of Environment*, 92 (3), 353-362.
- Morton, G. M., 1966, A Computer Oriented Geodetic Data Base and a New Technique in File Sequencing, International Business Machines Company, p.
- Nurunnabi, A., Belton, D. ve West, G., 2012, Robust Segmentation for Multiple Planar Surface Extraction in Laser Scanning 3D Point Cloud Data, *21st International Conference on Pattern Recognition (Icpr 2012)*, 1367-1370.
- Overby, J., Bodum, L., Kjems, E. ve Ilsøe, P. M., 2017, Automatic 3D building reconstruction from airborne laser scanning and cadastral data using hough transform, *XX ISPRS Congress, Commission III*, 1-6.
- Pal, N. R. ve Pal, S. K., 1993, A Review on Image Segmentation Techniques, *Pattern Recognition*, 26 (9), 1277-1294.
- Papon, J., Abramov, A., Schoeler, M. ve Worgotter, F., 2013, Voxel Cloud Connectivity Segmentation - Supervoxels for Point Clouds, *2013 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr)*, 2027-2034.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. ve Duchesnay, E., 2011, Scikit-learn: Machine Learning in Python, *Journal of Machine Learning Research*, 12, 2825-2830.
- Polat, N. ve Uysal, M., 2016, Hava lazer tarama sistemi, uygulama alanları ve kullanılan yazılımlara genel bir bakış, *Afyon Kocatepe Üniversitesi Fen Ve Mühendislik Bilimleri Dergisi*, 16, 679 - 692.
- Poux, F. ve Billen, R., 2019, Voxel-based 3D Point Cloud Semantic Segmentation: Unsupervised Geometric and Relationship Featuring vs Deep Learning Methods, *Isprs International Journal of Geo-Information*, 8 (5).
- Pu, S. ve Vosselman, G., 2009, Knowledge based reconstruction of building models from terrestrial laser scanning data, *Isprs Journal of Photogrammetry and Remote Sensing*, 64 (6), 575-584.
- Pu, S., Rutzinger, M., Vosselman, G. ve Elberink, S. O., 2011, Recognizing basic structures from mobile laser scanning data for road inventory studies, *Isprs Journal of Photogrammetry and Remote Sensing*, 66 (6), 28-39.
- Rabbani Shah, T., 2006, Automatic reconstruction of industrial installations using point clouds and images, 62, *Delft, Nederlander Commissie voor Geodesie*, p.
- Rabbani, T. ve Heuvel, F. V. D., 2005, Efficient Hough transform for automatic detection of cylinder in pint clouds, *ISPRS WG III/3, III/4, V/5 Workshop "Laser scanning 2005"*.
- Rabbani, T., van den Heuvel, F. ve Vosselman, G., 2006, Segmentation of point clouds using smoothness constraint, *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences - Commission V Symposium 'Image Engineering and Vision Metrology'*, 36, 248-253.

- Rabbani, T., Dijkman, S., van den Heuvel, F. ve Vosselman, G., 2007, An integrated approach for modelling and global registration of point clouds, *Isprs Journal of Photogrammetry and Remote Sensing*, 61 (6), 355-370.
- Rau, J. Y., Jhan, J. P. ve Hsu, Y. C., 2015, Analysis of Oblique Aerial Images for Land Cover and Point Cloud Classification in an Urban Environment, *Ieee Transactions on Geoscience and Remote Sensing*, 53 (3), 1304-1319.
- Rusu, R. B., Marton, Z. C., Blodow, N., Dolha, M. ve Beetz, M., 2008, Towards 3D Point cloud based object maps for household environments, *Robotics and Autonomous Systems*, 56 (11), 927-941.
- Rusu, R. B., Blodow, N. ve Beetz, M., 2009, Fast Point Feature Histograms (FPFH) for 3D Registration, *IEEE International Conference on Robotics and Automation*, Vols 1-7, 1848-1853.
- Rusu, R. B. ve Cousins, S., 2011, 3D is here: Point Cloud Library (PCL), *2011 IEEE International Conference on Robotics and Automation (Icra)*.
- Sacchi, R., Poliakoff, J. F., Thomas, P. D. ve Hafele, K., 1999, Curvature estimation for segmentation of triangulated surfaces, *Second International Conference on 3-D Digital Imaging and Modeling (Cat. No.PR00062)*, 536-543.
- Saglam, A. ve Baykan, N. A., 2017, Sequential image segmentation based on minimum spanning tree representation, *Pattern Recognition Letters*, 87, 155-162.
- Saglam, A. ve Baykan, N. A., 2019, Continuous time threshold selection for binary classification on polarized data, *Pamukkale University Journal of Engineering Sciences-Pamukkale Üniversitesi Muhendislik Bilimleri Dergisi*, 25 (5), 596-602.
- Saglam, A., Makineci, H. B., Baykan, N. A. ve Baykan, O. K., 2020, Boundary constrained voxel segmentation for 3D point clouds using local geometric differences, *Expert Systems with Applications*, 157.
- Sağlam, A., 2016, Minimum yayılan ağaç tabanlı sıralı görüntü bölütleme, Yüksek Lisans Tezi, *Selçuk Üniversitesi*.
- Samet, H., 1989, Applications of Spatial Data Structures: Computer Graphics, Image Processing, and GIS, Addison-Wesley, p.
- Schnabel, R., Wahl, R. ve Klein, R., 2007, Efficient RANSAC for point-cloud shape detection, *Computer Graphics Forum*, 26 (2), 214-226.
- Schoenberg, J. R., Nathan, A. ve Campbell, M., 2010, Segmentation of Dense Range Information in Complex Urban Scenes, *IEEE International Conference on Intelligent Robots and Systems (Iros 2010)*.
- Schumann, C., Oeltze, S., Bade, R., Preim, B. ve Peitgen, H.-O., 2007, Model-free Surface Visualization of Vascular Trees, *EuroVis07: Joint Eurographics - IEEE VGTC Symposium on Visualization*.
- Serna, A. ve Marcotegui, B., 2014, Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning, *Isprs Journal of Photogrammetry and Remote Sensing*, 93, 243-255.
- Shah, T. R., 2006, Automatic Reconstruction of Industrial Installations Using Point Clouds and Images, *Niederlander Commissie voor Geodesie*, p.
- Shakarji, C. M., 1998, Least-squares fitting algorithms of the NIST algorithm testing system, *Journal of Research of the National Institute of Standards and Technology*, 103 (6), 633-641.
- Shi, J. B. ve Malik, J., 2000, Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (8), 888-905.
- Sokolova, M. ve Lapalme, G., 2009, A systematic analysis of performance measures for classification tasks, *Information Processing & Management*, 45 (4), 427-437.

- Stein, S. C., Schoeler, M., Papon, J. ve Worgotter, F., 2014, Object Partitioning using Local Convexity, *2014 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr)*, 304-311.
- Strom, J., Richardson, A. ve Olson, E., 2010, Graph-based Segmentation for Colored 3D Laser Point Clouds, *Ieee/Rsj 2010 International Conference on Intelligent Robots and Systems (Iros 2010)*, 2131-2136.
- Su, Y. T., Bethel, J. ve Hu, S. W., 2016, Octree-based segmentation for terrestrial LiDAR point cloud data in industrial applications, *Isprs Journal of Photogrammetry and Remote Sensing*, 113, 59-74.
- Tseng, Y. H., Tang, K. P. ve Chou, F. C., 2007, Surface reconstruction from LiDAR data with extended snake theory, In: *Energy Minimization Methods in Computer Vision and Pattern Recognition, Proceedings*, Eds, p. 479-+.
- Ural, S. ve Shan, J., 2012, Min-Cut Based Segmentation of Airborne Lidar Point Clouds, *12th Isprs Congress, Technical Commission LII*, 39-B3, 167-172.
- Van Rijsbergen, C. J., 1979, Information retrieval, *Newton, MA, USA*, Butterworth-Heinemann, p. 208p.
- Vanegas, C. A., Aliaga, D. G. ve Benes, B., 2012, Automatic Extraction of Manhattan-World Building Masses from 3D Laser Range Scans, *Ieee Transactions on Visualization and Computer Graphics*, 18 (10), 1627-1637.
- Verdoja, F., Thomas, D. ve Sugimoto, A., 2017, Fast 3d Point Cloud Segmentation Using Supervoxels with Geometry and Color for 3d Scene Understanding, *2017 Ieee International Conference on Multimedia and Expo (Icme)*, 1285-1290.
- Vieira, M. ve Shimada, K., 2005, Surface mesh segmentation and smooth surface extraction through region growing, *Computer Aided Geometric Design*, 22 (8), 771-792.
- Vo, A. V., Linh, T. H., Laefer, D. F. ve Bertolotto, M., 2015, Octree-based region growing for point cloud segmentation, *Isprs Journal of Photogrammetry and Remote Sensing*, 104, 88-100.
- Vosselman, G. ve Dijkman, S., 2001, 3D building model reconstruction from point clouds and ground plans, *International Archives of Photogrammetry and Remote Sensing*, 34, 37-43.
- Vosselman, G., Coenen, M. ve Rottensteiner, F., 2017, Contextual segment-based classification of airborne laser scanner data, *Isprs Journal of Photogrammetry and Remote Sensing*, 128, 354-371.
- Wang, M. ve Tseng, Y. H., 2010, Automatic Segmentation of Lidar Data into Coplanar Point Clusters Using an Octree-Based Split-and-Merge Algorithm, *Photogrammetric Engineering and Remote Sensing*, 76 (4), 407-420.
- Wang, M. A. ve Tseng, Y. H., 2011, Incremental segmentation of lidar point clouds with an octree-structured voxel space, *Photogrammetric Record*, 26 (133), 32-57.
- Weinmann, M., Jutzi, B., Hinz, S. ve Mallet, C., 2015a, Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers, *Isprs Journal of Photogrammetry and Remote Sensing*, 105, 286-304.
- Weinmann, M., Urban, S., Hinz, S., Jutzi, B. ve Mallet, C., 2015b, Distinctive 2D and 3D features for automated large-scale scene analysis in urban areas, *Computers & Graphics-Uk*, 49, 47-57.
- Wu, F., Wen, C. L., Guo, Y. L., Wang, J. J., Yu, Y. T., Wang, C. ve Li, J., 2017, Rapid Localization and Extraction of Street Light Poles in Mobile LiDAR Point Clouds: A Supervoxel-Based Approach, *IEEE Transactions on Intelligent Transportation Systems*, 18 (2), 292-305.

- Wu, Z. ve Leahy, R., 1993, An Optimal Graph-Theoretic Approach to Data Clustering - Theory and Its Application to Image Segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15 (11), 1101-1113.
- Wurm, K. M., Kretzschmar, H., Kummerle, R., Stachniss, C. ve Burgard, W., 2014, Identifying vegetation from laser data in structured outdoor environments, *Robotics and Autonomous Systems*, 62 (5), 675-684.
- Xiao, J. H., Zhang, J. H., Adler, B., Zhang, H. X. ve Zhang, J. W., 2013, Three-dimensional point cloud plane segmentation in both structured and unstructured environments, *Robotics and Autonomous Systems*, 61 (12), 1641-1652.
- Xu, R. ve Wunsch, D., 2005, Survey of clustering algorithms, *IEEE Transactions on Neural Networks*, 16 (3), 645-678.
- Xu, Y., Tuttas, S. ve Stilla, U., 2016, Segmentation of 3D outdoor scenes using hierarchical clustering structure and perceptual grouping laws, *2016 9th Iaprr Workshop on Pattern Recognition in Remote Sensing (Prirs)*.
- Xu, Y., Hoegner, L., Tuttas, S. ve Stilla, U., 2017, Voxel- and graph-based point cloud segmentation of 3D scenes using perceptual grouping laws, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-1/W1, 43-50.
- Xu, Y., Yao, W., Tuttas, S., Hoegner, L. ve Stilla, U., 2018a, Building-Segmentation-Reference-Dataset, <https://github.com/Yusheng-Xu/Building-Segmentation-Reference-Dataset>: [2020].
- Xu, Y., 2019, SVGS, <https://github.com/Yusheng-Xu/VGS-SVGS-Segmentation>:
- Xu, Y. S., Tuttas, S., Hoegner, L. ve Stilla, U., 2018b, Voxel-based segmentation of 3D point clouds from construction sites using a probabilistic connectivity model, *Pattern Recognition Letters*, 102, 67-74.
- Xu, Y. S., Yao, W., Tuttas, S., Hoegner, L. ve Stilla, U., 2018c, Unsupervised Segmentation of Point Clouds From Buildings Using Hierarchical Clustering Based on Gestalt Principles, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11 (11), 4270-4286.
- Yang, B. S. ve Dong, Z., 2013, A shape-based segmentation method for mobile laser scanning point clouds, *Isprs Journal of Photogrammetry and Remote Sensing*, 81, 19-30.
- Yang, B. S., Dong, Z., Zhao, G. ve Dai, W. X., 2015, Hierarchical extraction of urban objects from mobile laser scanning data, *Isprs Journal of Photogrammetry and Remote Sensing*, 99, 45-57.
- Yao, W., Hinz, S. ve Stilla, U., 2009, Object extraction based on 3d-segmentation of LiDAR data by combining mean shift with normalized cuts: two examples from urban areas, *2009 Joint Urban Remote Sensing Event, Vols 1-3*, 1216-1221.
- Yazar, I., Yavuz, H. S. ve Çay, M. A., 2009, Temel Bileşen Analizi Yönteminin Ve Bazı Klasik Ve Robust Uyarlamalarının Yüz Tanıma Uygulamaları, *Eskişehir Osmangazi Üniversitesi Mühendislik ve Mimarlık Fakültesi Dergisi*, 22, 49-63.
- Zhang, J. X., Lin, X. G. ve Ning, X. G., 2013, SVM-Based Classification of Segmented Airborne LiDAR Point Clouds in Urban Areas, *Remote Sensing*, 5 (8), 3749-3775.
- Zhu, X. L., Zhao, H. J., Liu, Y. M., Zhao, Y. P. ve Zha, H. B., 2010, Segmentation and Classification of Range Image from an Intelligent Vehicle in Urban Environment, *Ieee/Rsj 2010 International Conference on Intelligent Robots and Systems (Iros 2010)*, 1457-1462.

EKLER

EK-1 Geliştirilen yöntemde farklı τ_{max} değerleri kullanıldığında elde edilen sonuçların renklendirilmiş olarak gösterimleri











EK-2 Karşılaştırılan segmentasyon yöntemlerinin en iyi doğruluk sonucunu veren parametre değerlerinin segmentasyon sonuçlarının renklendirilmiş olarak gösterimleri



RG



VIS



ORG



LCCP



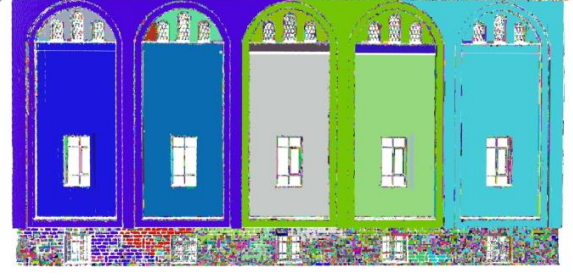
SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



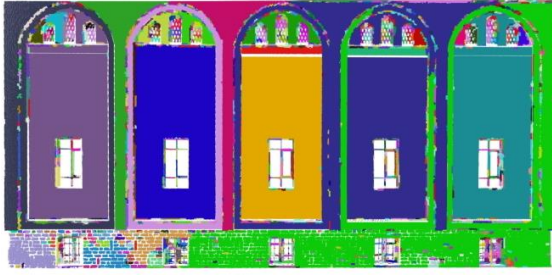
VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



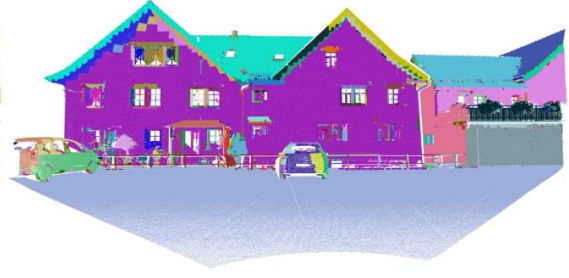
SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler

EK-3 Karşılaştırılan segmentasyon yöntemlerinin en iyi $F1$ skor sonucunu veren parametre değerlerinin segmentasyon sonuçlarının renklendirilmiş olarak gösterimleri



RG



VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



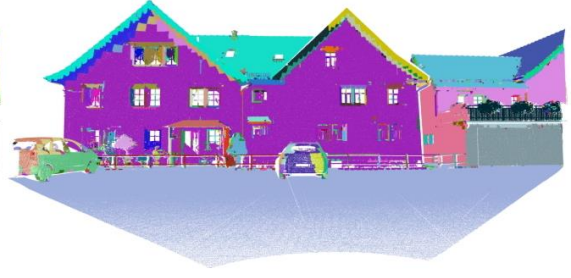
SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler



RG



VIS



ORG



LCCP



SVGS

Geliştirilen Yöntem
(refitting olmadan)Geliştirilen Yöntem
(refitting ile)

Referans Segmentler

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Ali SAĞLAM
Uyruğu : T.C.
Doğum Yeri ve Tarihi : Ankara - 09.10.1983
Telefon : 0 (505) 393 91 30
E-Posta : asaglam@ktun.edu.tr, alisaglam666@gmail.com

EĞİTİM

Derece	Adı	İlçe	İl	Bitirme Yılı
Lise	: Ankara Aydınlikevler Lisesi, Keçiören, Ankara			2000
Üniversite	: Çukurova Üniversitesi, Bilgisayar ve Öğretim Teknolojileri Öğretmenliği			2004
Üniversite	: Hoca Ahmet Yesevi Uluslararası Türk-Kazak Üniversitesi, Bilgisayar Mühendisliği			2012
Yüksek Lisans	: Selçuk Üniversitesi, Bilgisayar Mühendisliği			2016
Doktora	: Konya Teknik Üniversitesi, Bilgisayar Mühendisliği			2020

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2004-2012	MEB	Öğretmen
2012-2014	Karabük Üniversitesi	Araştırma Görevlisi
2014-2018	Selçuk Üniversitesi	Araştırma Görevlisi
2018-Devam	Konya Teknik Üniversitesi	Araştırma Görevlisi

UZMANLIK ALANI

Görüntü İşleme, Graf Teorisi, Veri Madenciliği, Makine Öğrenmesi

YABANCI DİLLER

İngilizce

YAYINLAR

1. **A. Saglam**, N.A. Baykan (2015). An efficient object extraction with graph-based image segmentation. *2015 4th International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, 86-91. Doi: 10.1109/ACSAT.2015.29 (**Yüksek Lisans tezinden yapılmıştır**)

2. **A. Saglam**, N.A. Baykan (2016). Parallelization of a Hierarchical Graph Based Image Segmentation using OpenMP. *International Journal of Applied Mathematics, Electronics and Computers*, 4(Special Issue-1), 6-11. Doi: 10.18100/ijamec.271038 **(Yüksek Lisans tezinden yapılmıştır)**
3. **A. Saglam**, N.A. Baykan (2017). Sequential image segmentation based on minimum spanning tree representation. *Pattern Recognition Letters*, 87, 155-162. Doi: 10.1016/j.patrec.2016.06.001 **(SCI-Exp) (Yüksek Lisans tezinden yapılmıştır)**
4. M. Karakoyun, **A. Saglam**, N.A. Baykan, A.A. Altun (2017). Non-locally color image segmentation for remote sensing images in different color spaces by using data-clustering methods. *5th International Conference on Advanced Technology Sciences (ICAT2017)*, 6-12.
5. **A. Saglam**, N.A. Baykan (2017). Roof detection on satellite images. *2017 25th Signal Processing and Communications Applications Conference (SIU)*, Doi: 10.1109/SIU.2017.7960413
6. **A. Saglam**, N.A. Baykan (2017). Effects of color spaces and distance norms on graph-based image segmentation. *2017 3rd International Conference on Frontiers of Signal Processing (ICFSP)*, 130-135. Doi: 10.1109/ICFSP.2017.8097156 **(Yüksek Lisans tezinden yapılmıştır)**
7. N.A. Baykan, **A. Saglam** (2018). Fast k-means color image clustering with normalized distance values. *Selcuk University Journal of Engineering, Science and Technology*, 6(2), 175-187. Doi: 10.15317/Scitech.2018.124 **(TR-Index)**
8. **A. Saglam**, N.A. Baykan (2018). A satellite image classification approach by using one dimensional discriminant analysis. *GeoInformation For Disaster Management (Volume XLII-3/W4)*, 42(3-W4), 429-435. Doi: 10.5194/isprs-archives-XLII-3-W4-429-2018
9. **A. Saglam**, N.A. Baykan (2018). User interactive object extraction with sequential image segmentation. *2018 26th Signal Processing and Communications Applications Conference (SIU)*. Doi: 10.1109/SIU.2018.8404215 **(Yüksek Lisans tezinden yapılmıştır)**
10. **A. Saglam**, N.A. Baykan (2019). Continuous time threshold selection for binary classification on polarized data. *Pamukkale University Journal of Engineering Sciences*, 25(5), 596-602. Doi: 10.5505/pajes.2018.26125 **(ESCI) (TR-Index)**
11. **A. Saglam**, N.A. Baykan (2019). Evaluating the attributes of remote sensing image pixels for fast k-means clustering. *Turkish Journal Of Electrical Engineering Computer Sciences*, 27(6), 4188-4202. Doi: 10.3906/elk-1901-190 **(SCI-Exp)**
12. **A. Saglam**, H.B. Makineci, N.A. Baykan, Ö.K. Baykan (2020). Boundary constrained voxel segmentation for 3D point clouds using local geometric differences. *Expert Systems with Applications*, 157, 113439. Doi: 10.1016/j.eswa.2020.113439 **(SCI-Exp) (Doktora tezinden yapılmıştır)**

PROJELER**1. TÜBİTAK 1002**

Proje No : 119E012

Proje Adı : Lidar 3D Nokta Bulutu Verilerinin Konum ve Renk Özelliklerine
Göre Bölütlenmesi

Proje Yürütücüsü: Dr. Öğr. Üyesi Nurdan BAYKAN

Projedeki Görevi : Bursiyer

Süre : 15.08.2019 – 15.08.2020

