LİDAR VERİSİNDEN BİNA ÇATI DÜZLEMLERİNİN OTOMATİK ÇIKARIMI VE MODELLEMESİ

AUTOMATIC EXTRACTION AND MODELING OF BUILDING ROOF PLANES FROM LiDAR DATA

MURAT GÜLER

PROF. DR. MUSTAFA TÜRKER

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim - Öğretim ve Sınav Yönetmeliğinin

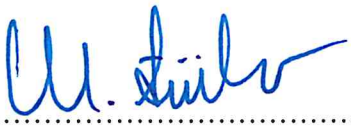
Geomatik Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2018

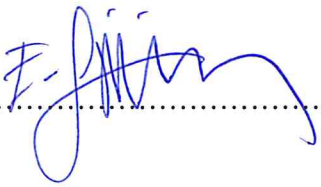
MURAT GÜLER'in hazırladığı **"LiDAR Verisinden Bina Çatı Düzlemlerinin Otomatik Çıkarımı ve Modellemesi"** adlı bu çalışma aşağıdaki jüri tarafından **GEOMATİK MÜHENDİSLİĞİ ANABİLİM DALI'** nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. Erhan TERCAN
Başkan ..... 

Prof. Dr. Mustafa TÜRKER
Danışman ..... 

Doç. Dr. Cevdet Coşkun AYDIN
Üye ..... 

Doç. Dr. Füsun BALIK ŞANLI
Üye ..... 

Dr. Öğr. Üyesi Emre SÜMER
Üye ..... 

Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından **YÜKSEK LİSANS TEZİ** olarak onaylanmıştır.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU
Fen Bilimleri Enstitüsü Müdürü

YAYINLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

☒ **Tezimin/Raporumun tamamı dünya çapında erişime açılabilir ve bir kısmı veya tamamının fotokopisi alınabilir.**
(Bu seçenekle teziniz arama motorlarında indekslenebilecek, daha sonra tezinizin erişim statüsünün değiştirilmesini talep etseniz ve kütüphane bu talebinizi yerine getirse bile, tezinin arama motorlarının önbelleklerinde kalmaya devam edebilecektir.)

☐ **Tezimin/Raporumun ..... tarihine kadar erişime açılmasını ve fotokopi alınmasını (İç Kapak, Özet, İçindekiler ve Kaynakça hariç) istemiyorum.**
(Bu sürenin sonunda uzatma için başvuruda bulunmadığım taktirde, tezimin/raporumun tamamı her yerden erişime açılabilir, kaynak gösterilmek şartıyla bir kısmı ve ya tamamının fotokopisi alınabilir)

☐ **Tezimin/Raporumun ..... tarihine kadar erişime açılmasını istemiyorum, ancak kaynak gösterilmek şartıyla bir kısmı veya tamamının fotokopisinin alınmasını onaylıyorum.**

☐ **Serbest Seçenek/Yazarın Seçimi**

23 / 05 / 2018

M. Güler

Murat GÜLER

ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

23/05/2018



MURAT GÜLER

Aileme, dostlarıma
ve sevgili eşim Elif'e

ÖZET

LiDAR VERİSİNDEN BİNA ÇATI DÜZLEMLERİNİN OTOMATİK ÇIKARIMI VE MODELLEMESİ

MURAT GÜLER

Yüksek Lisans, Geomatik Mühendisliği Bölümü

Tez danışmanı: Prof.Dr. Mustafa TÜRKER

Mayıs 2018, 149 sayfa

Üç boyutlu (3B) şehir modellerinde şehri oluşturan temel unsur olan binaların modellenmesi LiDAR (Light Detection and Ranging) nokta bulutu verisinin en yaygın uygulama alanları arasında yer almaktadır. Bu tez çalışmasında, İzmir ili Bergama ilçesi kent merkezinden seçilen üç alana ait 3B hava LiDAR nokta bulutu verilerinden otomatik bina çatı düzlemlerinin çıkarılması ve modellenmesi amaçlanmıştır. Bunun için ilk olarak, yer filtrelemesi işlemi yapılmıştır. Yer filtrelemesi sonucu elde edilen çıplak zemin noktaları ham veriden çıkarıldıktan sonra, geriye kalan LiDAR noktaları üzerinde sınıflama işlemi yapılarak bina sınıfı çıkarılmıştır. Sonra, çıkarılan bina sınıfına Bölge Büyüme Bölütleme (Region Growing Segmentation) algoritması uygulanarak, her bir binaya ait nokta bulutu kümesi ayrı olarak tespit edilmiştir. Tespit edilen her bir binaya ait nokta bulutu kümesine 3 Boyutlu Rastgele Örnek Uzlaşımı (RANSAC: 3D RANdom SAmple Consensus) algoritması uygulanarak bina çatılarının düzlemsel yüzeyleri otomatik olarak çıkarılmıştır.

Bina çatısı düzlemlerinin çıkarılmasından sonra, çatı düzlemleri noktalarında bulunan gürültü DBSCAN (Density Based Spatial Clustering of Applications with Noise) algoritması

kullanılarak belirlenmiş ve silinmiştir. Gürültünün giderilmesinden sonra, bina çatısı düzlemi noktalarından sınır geçirilmiştir. Son olarak, Douglas-Peucker algoritması ile bina çatısı düzlemi sınırının sadeleştirilmesi işlemi yapılmıştır.

Elde edilen bina çatısı modelleri incelendiklerinde, en başarılı sonuçlar üçüncü test alanında elde edilmiştir. Üçüncü test alanında, diğer test alanlarından daha başarılı sonuçların elde edilmesinin sebebi, DBSCAN algoritması aynı parametre değerlerinin, daha az sayıda bina üzerine uygulanmasından dolayı gürültünün daha başarılı bir şekilde tespit edilmesi ve RANSAC algoritmasının bu test alanında daha başarılı sonuç vermesidir. Birinci ve ikinci test alanları sonuçları karşılaştırıldıklarında, birinci test alanında elde edilen sonuçların ikinci test alanında elde edilen sonuçlara göre daha başarılı oldukları görülmüştür. Bunun nedeni, birinci test alanında nokta yoğunluğunun düşük olmasından dolayı gürültü noktalarının otomatik tespiti işlemi daha başarılı bir şekilde gerçekleşmiştir.

Anahtar Kelimeler: LiDAR, 3B Şehir Modeli, DBSCAN, Otomatik Çatı Düzlemi Çıkarımı, Bölütleme, Geriçatım, RANSAC, Douglas-Peucker

ABSTRACT

AUTOMATIC EXTRACTION AND MODELING OF BUILDING ROOF PLANES FROM LiDAR DATA

MURAT GÜLER

Master of Sciences, Department of Geomatics Engineering

Supervisor: Prof. Dr. Mustafa TÜRKER

May 2018, 149 pages

Being as main factor on 3D city models; building modelling is among the most common field of applications of LiDAR (Light Detection And Ranging) point cloud data. In this study, automatic extraction and modeling of building roof planes from the data of 3D airborne LiDAR point cloud dataset of three pilot areas selected from the city center of Bergama / İzmir province is aimed. First, ground filtering process was carried out. Building class was extracted through the classification of the remaining LiDAR points after bare ground points -obtained from ground filtering process- removed from raw data. Following this step, Region Growing Segmentation algorithm was applied on the extracted. Building class and the point cloud of each building was detected separately. Next, the planar surfaces of the building roofs were automatically extracted by applying the 3D RANSAC (3D RANdom SAmple Consensus) algorithm to point cloud of each detected building.

After extracting the planar surfaces of the building roofs, the noise points on each building roof plane were identified using the DBSCAN (Density Based Spatial Clustering of Applications with Noise) algorithm and removed from the roof plane points. After removing the noise points,

the boundary line was extracted from the points of the building roof plane. As the last step, building roof plane border lines were simplified by using Douglas-Peucker algorithm.

When the obtained roof plane models were analyzed it was observed that the best results belong to test field #3. The main reason for having better results in the third test field is that the DBSCAN algorithm is more successful in detecting noise on fewer buildings using the same parameter values. Also, the RANSAC algorithm was more successful in this test field compared to other fields. When the results of test fields #1 and #2 were compared, it was noted that the results of test field #1 were better than the results of test field #2. The main reason for this is that due to lower point density in test field #1 the automatic detection of noise points was performed more successfully.

Keywords: LiDAR, 3D City Models, DBSCAN, Automatic Roof Plane Extraction, Segmentation, Reconstruction, RANSAC, Douglas-Peucker

TEŞEKKÜR

Bu çalışmanın gerçekleştirilmesinde, değerli bilgilerini benimle paylaşan, kıymetli zamanını ayırıp engin bilgi ve deneyimi ile beni yönlendiren danışman hocam Prof. Dr. Mustafa TÜRKER'e teşekkürü bir borç biliyor ve şükranlarımı sunuyorum. Sabırla ve büyük bir ilgiyle bana faydalı olabilmek için elinden gelenden fazlasını sunan, her sorun yaşadığımda yanına çekinmeden gidebildiğim, güler yüzünü ve samimiyetini benden esirgemeyen ve gelecekteki mesleki hayatımda da bana verdiği değerli bilgilerden faydalanacağımı düşündüğüm kıymetli hocam Doç. Dr. Füsun BALIK ŞANLI'ya müteşekkirim. Tez süresi boyunca yardımlarından ve gösterdikleri yakın ilgiden dolayı Arş. Gör. Dr. Engin TUNALI ve Arş. Gör. Yüksel BOZ'a teşekkür ederim. Tez süresince birlikte çok fazla vakit geçirdiğim, hayatında çok daha başarılı olacağına inandığım kıymetli arkadaşım Mehmet BÜYÜKDEMİRCİOĞLU'na sonsuz teşekkürlerimi sunarım. Ayrıca çalışma süresince bilgi, görüş ve yardımlarını esirgemeyen tüm değerli hocalarıma, bugüne kadarki süreçte derslerde ve projelerde kazandırdıkları bilgi ve deneyim için teşekkür ederim.

Benden hiçbir zaman desteğini esirgemeyen bu hayattaki en büyük şansım olan annem Herdem GÜLER ve babam Kerem GÜLER'e, çalışmamda desteğini ve bana olan güvenini benden esirgemeyen abim Emrah GÜLER'e ve son olarak ilgi ve anlayışıyla her zaman yanımda olan eşim Elif GÜLER'e teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET i
ABSTRACT iii
TEŞEKKÜR v
İÇİNDEKİLER vi
ÇİZELGELER viii
EŞİTLİKLER ix
ŞEKİLLER x
SİMGELER VE KISALTMALAR xiii
1. GİRİŞ 1
1.1. Tezin Amacı 2
1.2. Geçmiş Çalışmalar 3
1.3. Çalışma Alanı ve Veri 5
1.4. Tezin Organizasyonu 7
1.5. Kullanılan Yazılımlar 8
2. LiDAR NOKTA BULUTU VERİSİNİN İŞLENMESİ 9
2.1. LiDAR Nokta Bulutu Veri Yapısı 9
2.2. Las Format Tanımı 10
2.3. Zemin Filtreleme 11
3. YÖNTEM 14
3.1. Yöntemin Genel Adımları 14
3.2. Ön İşlemler 16
3.2.1. Indeks Oluşturma 18
3.2.2. Gürültü Noktalarının Kaldırılması 18
3.2.3. Örtüşen Alanların Belirlenmesi 19
3.2.4. Zemin ve Nesne Ayrımı 21
3.2.5. Yüksek Bitki Örtüsü (Ağaç) ve Binaları Sınıflandırma 24
3.2.6. Bina Sınıfının Elde Edilmesi 25
3.2.7. Nokta Bulutu Meta Verisi 26
3.2.8. Nokta Bulutunu ASCII Metin Formatına Dönüştürme 26

3.3. Bölütleme 27
3.4. Bina Düzlemlerinin Çıkarılması 29
3.5. Gürültünün Ayıklanması 36
3.6. Çatı Düzlemi Sınırlarının Çıkarılması 46
3.7. Çatı Düzlemi Sınırlarının Sadeleştirilmesi 48
4. BULGULAR 50
4.1. Çatı Düzlemlerinin Çıkarılması 50
4.2. Gürültünün Ayıklanması 58
4.3. Çatı Düzlemi Sınırlarının Çıkarılması ve Sadeleştirilmesi 62
5. SONUÇLAR 73
KAYNAKLAR 77
EKLER 80
ÖZGEÇMİŞ 130

ÇİZELGELER

Sayfa

Çizelge 2.1. ASPRS standart LiDAR nokta sınıfları 10
Çizelge 3.1. Veriye en uygun parametrenin belirlenmesi için yapılan denemeden bir kesit ... 23
Çizelge 3.2. Görsel olarak yapılan denemeler sonucunda öngörülen en uygun parametre değerleri 23
Çizelge 3.3. Bölge Büyüme Bölütleme algoritması parametreleri 28
Çizelge 3.4. RANSAC algoritmasına ait parametreler ve değerleri 30
Çizelge 3.5. Test alanlarına uygulanan DBSCAN algoritması parametreleri 37

EŞİTLİKLER

Sayfa

Eşitlik 3.1. 28
Eşitlik 4.1. 50

ŞEKİLLER

Sayfa

Şekil 1.1. Test alanları hava LiDAR nokta bulutu verisi. (a) Test alanı #1, (b) Test alanı #2, (c) Test alanı #3 7
Şekil 2.1. İlerleyen TIN Yoğunlaştırma filtreleme yöntemi a) Çalışma alanının karelajlara bölünmesi b) Üçgen sınırında bulunan noktalardan bir noktanın, noktadan üçgen kenarlarına olan açı eşik değer parametresinin gösterimi 12
Şekil 2.2. İlerleyen TIN Yoğunlaştırma filtreleme yöntemi işlem adımları. a) Tohum noktalarının seçimi b) Arazinin kaba yaklaşımı için tohum noktalarndan üçgenleme yapılması c) Arazinin gerçek yapısına yakın bir gösterim elde etmek için araziye dahil edilen daha fazla nokta ile aşamalı yoğunlaştırma işlemi. 13
Şekil 3.1. Yöntemin Akış Şeması 16
Şekil 3.2. Test alanı #2'ye ait LiDAR verisinin gösterimi 17
Şekil 3.3. Test alanı #2'ye ait LiDAR verisinin yükseklik değerlerine göre renkli gösterimi . 17
Şekil 3.4. Veriye ait gürültü noktalarının bir kısmının profileden görüntüsü 18
Şekil 3.5. Gürültüden arındırılmış nokta bulutunun profilden gösterimi 19
Şekil 3.6. Farklı şeritten gelen gelen veri olup olmadığının "Max overlap count " seçeneği ile incelenmesi 20
Şekil 3.7. Google Earth'te test 2#'ye ait alanın gösterimi 20
Şekil 3.8. Zemine ait nokta bulutunun yükseklik değerlerine göre renkli gösterimi 22
Şekil 3.9. Zemin ait olmayan nokta bulutunun yükseklik değerlerine göre renkli gösterimi .. 23
Şekil 3.10. Yüksek Bitki Örtüsü (Ağaç) ve Binaların Sınıflandırması 25
Şekil 3.11. Elde edilen bina sınıfının yükseklik değerlerine göre gösterimi 25
Şekil 3.12. Elde edilen bina sınıfının yükseklik değerlerine göre farklı açıdan gösterimi 26
Şekil 3.13. Bölge Büyüme Bölütleme algaritmasının uygulanması ile bölütlenmiş 26 adet bina nokta bulutu kümesi. Her bir renk bir binayı temsil etmektedir. 29
Şekil 3.14. Test alanı #3, birinci binaya ait RANSAC sonuçları a) orjinal bina nokta bulutu b) ilk düzlem noktaları c) ikinci düzlem noktaları d) üçüncü düzlem noktaları e) dördüncü düzlem noktaları.. 31
Şekil 3.15. Test alanı #3, birinci binaya ait; elde edilen dört düzlem noktalarının bir arada gösterimi 32

Şekil 3.16. Test alanı #3, ikinci binaya ait RANSAC sonuçları a) orjinal bina nokta bulutu b) ilk düzlem noktaları c) ikinci düzlem noktaları d) üçüncü düzlem noktaları e) dördüncü düzlem noktaları 33
Şekil 3.17. Test alanı #3, ikinci binaya ait; elde edilen dört düzlem noktalarının bir arada gösterimi 34
Şekil 3.18 Test alanı #3, üçüncü binaya ait RANSAC sonuçları a) orjinal bina nokta bulutu b) ilk düzlem noktaları c) ikinci düzlem noktaları d) üçüncü düzlem noktaları e) dördüncü düzlem noktaları 35
Şekil 3.19. Test alanı #3, üçüncü binaya ait; elde edilen dört düzlem noktalarının bir arada gösterimi 36
Şekil 3.20. Test alanı #3, birinci bina; a) birinci b) ikinci c) üçüncü d) dördüncü düzlem'e ait gürültü noktalarının gösterimi 39
Şekil 3.21. Test alanı #3, ikinci bina; a) birinci b) ikinci c) üçüncü d) dördüncü düzlem'e ait gürültü noktalarının gösterimi 41
Şekil 3.22. Test alanı #3, üçüncü bina ; a) birinci b) ikinci c) üçüncü d) dördüncü düzlem'e ait gürültü noktalarının gösterimi 43
Şekil 3.23. Test alanı #3'e ait; a) birinci b) ikinici c) üçüncü ve d) tüm binaların, gürültüden ayıklanmış düzlemlerinin bir arada gösterimi 45
Şekil 3.24. Test alanı #3'e ait; a) birinci b) ikinci ve c) üçüncü bina düzlemlerine sınır fonksiyonunun uygulanması ve bu düzlemlerin bir arada gösterilmesi sonrası oluşan sonuç görüntü 47
Şekil 3.25. Test alanı #3'e ait; a) birinci b) ikinci ve c) üçüncü bina düzlemlerine douglas-peucker algoritmasının uygulanması sonrası oluşan sonuç görüntü 49
Şekil 4.1. Test alanı #1'de yer alan #1, #2, #8 ve #11 nolu binalara ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi 54
Şekil 4.2. Test alanı #1'de yer alan #3 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi 54
Şekil 4.3. Test alanı #1'de yer alan 15 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi 55
Şekil 4.4. Test alanı #1'de yer alan #17 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi 55

Şekil 4.5. Test alanı #2'de yer alan #1, #2 ve #3 nolu binalara ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi 57
Şekil 4.6. Test alanı #2'de yer alan #11 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi 57
Şekil 4.7. DBSCAN algoritması ile test alanı #1'de yer alan #5, #7, #10, #12 ve #24 numaralı binalara ait çatı düzlemlerinde bulunan gürültü noktalarının tespit edilmesi 59
Şekil 4.8. DBSCAN algoritması ile test alanı #2'de yer alan alan #5, #7, #10, #12 ve #24 numaralı binalara ait çatı düzlemlerinde yer alan gürültü noktalarının tespit edilmesi 61
Şekil 4.9. Çatı modellemesi amacıyla test alanı #1'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarılı örnekleri 67
Şekil 4.10. Çatı modellemesi amacıyla test alanı #1'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarısız örnekleri 68
Şekil 4.11. Çatı modellemesi amacıyla test alanı #2'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarılı örnekleri. 70
Şekil 4.12. Çatı modellemesi amacıyla test alanı #2'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarısız örnekleri 72
Şekil 4.13. Binalar üzerine ağaçların sarkması 72

SİMGELER VE KISALTMALAR

Simgeler

m	Metre
cm	Santimetre
mm	Milimetre

Kısaltmalar

3B	3 Boyutlu
ASPRS	American Society for Photogrammetry and Remote Sensing
ASCII	American Standart Code for Information Interchange
DBSCAN	Density Based Spatial Clustering of Applications with Noise
EVLRs	Extended Variable Length Records
VLRs	Variable Length Records
HGK	Harita Genel Komutanlığı
LiDAR	Lazer Tarama ve Ölçüm (Light Detection and Ranging)
PCL	Nokta Bulutu Kütüphanesi (Point Cloud Library)
RANSAC	Rastgele Örnek Uzlaşımı (RANdom SAmple Consensus)
SYM	Sayısal Yüzey Modeli
TIN	Üçgenlenmiş Düzensiz Ağ (Triangulated Irregular Network)
UTM	Merkator Projeksiyonu (Universal Transverse Mercator)
WGS84	Dünya Jeodezik Sistemi 1984 (World Geodetic System 1984)

1.GİRİŞ

Turizm bilgi sistemleri, kent pazarlama, cep telefonu sağlayıcıları için anten konumlarının planlanması, çoklu taşıma türleri içeren navigasyon sistemleri, mikro iklim ve gürültü yayılımı simülasyonları, bilgisayar oyunları gibi birçok potansiyel uygulama geçmişten beri vardır. Fakat bu uygulamaların veri bakımından çok farklı gereksinimleri olduğu için tüm ihtiyaçlara cevap veren tek bir 3B şehir modeli temsili yoktur [1].

3B şehir modelleri, telekomünikasyon ve şehir plancıları arasında gitgide popülerlik kazanmaktadır. Her geçen gün hızla büyüyen şehirler yoluyla yayılan hava ve gürültü kirliliğinin analizi ve emlak vergilerinin tahmini, 3B şehir modellerinin diğer uygulamaları arasında yer almaktadır [2]. Ayrıca, güvenlik hizmetleri, kent planlaması ve bunun gibi birçok uygulama alanlarında güncel 3B şehir modellerine ihtiyaç duyulmaktadır [3], [4].

Bir şehri oluşturan ana unsur bina olduğu için, binalar 3B şehir modellemesinde en çok talep gören nesnelerdir [5]. Güncel bina bilgisi, 3B şehir modelleri, afet ve iletişim yönetimi, kamu kurumları ve karar destek sistemlerinde önemli bir role sahiptir [6].

Kentsel alanlar, genellikle yol ve bina gibi topoğrafik unsurların inşası, yıkımı ya da genişletilmesinde insan faaliyetleri nedeniyle hızla değişmektedir [4]. Ayrıca, kentsel alanlarda yapılaşmanın hızlı bir şekilde artmasından dolayı bina nesneleri güncel olmayabilir. Bu nedenle, 3B topoğrafik nesneleri güncel olarak belirlemek ve çıkarmak için hızlı ve doğru veri toplama teknolojilerine ihtiyaç duyulmaktadır

Bu bağlamda, 3B kent verisi elde etmek için geleneksel fotogrametriye karşın LİDAR (Light Detection and Ranging) teknolojisini kullanmanın avantajları olarak yüksek hız, yüksek düşey doğruluk, yüksek yoğunluk ve düşük maliyet ön plana çıkmaktadır. LİDAR bina, ağaç, zemin, yol ve enerji hatlarının özellik çıkarımı ve 3B geriçatımında kullanılmaya uygun bir veri toplama yöntemidir [5].

Yarı-otomatik ve otomatik bina çıkarımı, kent uygulamalarında LİDAR verisinden yararlı bilgiler elde etmek için son yıllarda birçok araştırmacının temel çalışma alanı arasında yer almıştır. Kentsel alanlarda toplanan LİDAR nokta bulutu verisi, araba, bina vs. insan yapımı nesneler ile çıplak arazi, ağaç gibi doğal yüzeylerden yansıyan noktalardan oluşur. Farklı

yansıtıcı yüzeylerden gelen noktaların modellenebilmesi için bunların birbirinden ayrılması gerekmektedir.

Binaları LiDAR verilerinden modellemek için kabul görmüş yöntem, filtreleme olarak adlandırılan çıplak zemini belirlemek ve bunu takiben bina çıkarımı yapmaktır. Kısaca bina çıkarımı işlemi, çıplak zemin noktalarının ham veriden çıkarılmasından sonra geriye kalan (zemin üstü) LiDAR noktalarının alt kümesi üzerinde yapılır [7], [8].

Genel olarak bina geriçatımı için işlem adımları 3'e ayrılabilir [9]. Bunlar;

- Bina tespiti,
- Bireysel çatı düzlemleri için düzlemsel yüzeylerin belirlenmesi ve
- Model oluşturmadır.

Bu tez çalışmasında bina modellerinin belirlenebilmesi için nokta bulutu verisinden otomatik düzlemsel yüzeylerin çıkarılması sağlanmıştır. Böylece, bina geriçatımı için bu çalışmada yukarıda belirtilen ilk iki adım gerçekleştirilmiştir.

1.1. Tezin Amacı

Bu tez çalışmasının amaçları şu şekildedir:

- Yalnız hava LiDAR nokta bulutu verisi kullanılarak bina çatı düzlemlerini modellemek.
- Bina çatı düzlemlerini, 3B LiDAR nokta bulutu verisinden modellemek için yer filtrelemesi yaparak zemin üstü nesnelerden bina sınıfı çıkarımı yapmak.
- Bölge Büyüme Bölütleme algoritması uygulanarak her bina nokta bulutu kümelerini tespit edip bu kümelere RANSAC algoritması uygulayarak binalarda bulunan düzlemsel yüzeyleri çıkarmak.
- Bireysel çatı düzlemlerini modellemedeki başarızlığı en aza indirmek için, çıkarılan her bir düzleme ait noktalarda mevcut olan gürültüleri belirleyerek çatı modellemesini gerçekleştirmek.

1.2. Geçmiş Çalışmalar

Çalışmanın bu bölümünde, LİDAR verisinden düzlemsel yüzeylerin belirlenmesi için geçmiş yıllarda yapılmış olan ilgili çalışmalar incelenmiş ve bu çalışmalarda kullanılan yöntemlerden bahsedilmiştir.

Geçmişten günümüze kadar yapılmış çalışmalardan, 2001yılında [2] tarafından yapılmış olan bir çalışmada, bina çatılarında düzlem noktalarını çıkarmak için Hough dönüşümünü kullanılmıştır. Düzlem çatı yüzeyleri başlangıç segmentleri uzatılarak ve birleştirilerek elde edilmiştir. Geliştirilen algoritma bulunan noktaların aynı çatı düzleminde olup olmadığını kontrol etmediği için yapılan bu çalışmada ek kriterler kullanılmıştır.

2003 yılında [10] tarafından yapılmış olan bir çalışmada, Bölge Büyüme Bölütleme algoritması kullanılmıştır. Bu işlem, çatı düzlemlerini belirlemek için sayısal yüzey modeli (SYM) normal vektör varyasyonlarının analizi ile gerçekleştirilmiştir.

2003 yılında [11] tarafından yapılmış olan bir çalışmada, otomatik olarak bina bölgeleri tespit edilmiştir. Daha sonra, eğrilik tabanlı bölütleme tekniği uygulanarak çatı düzlemleri tespit edilmiştir. Çokyüzlü yapı modelleri oluşturmak için çatı düzlemleri gruplandırılarak çatı düzlem sınırlarının şekilleri belirlenmiştir. Ayrıca bu çalışmada, yeniden yapılandırılmış modellerin geometrik kalitesini arttırmak için geriçatım sürecine hava görüntülerini entegre etme konusunu ele alınmıştır.

2004 yılında [12] tarafından yapılmış çalışmada, bina yüzeylerini çıkarmak için bölge büyüme bölütleme algoritması ile en küçük kareler yüzey analizi uygulamışlardır. Değişen pencere boyutları ve lazer kaynaklı noktaların şekilleri ile yüzey analizi yapan en küçük kareler, bina çatı detaylarının belirlenmesinde anahtar rol oynamıştır. Bölütleme işlemi başarılı sonuçlar gösterse de, ağaçların çatı düzleminin büyük bir kısmının üzerine sarkması ve tüm lazer verilerinin binanın çatısına ulaşamama durumunun, bölütlemede başarısız sonuçlara neden olduğu söylenmiştir.

2007 yılında [9] tarafından yapılmış olan bir çalışmada, parametre uzayında kümelenerek başlatılan bölütleme algoritması, görüntü eşleştirme ve lazer taraması ile elde edilen nokta bulutları üzerinde uygulanarak güvenilir sonuçlar elde edilmiştir.

Bölge Büyüme Bölütleme algoritmaları bazen çok anlaşılır değildir ve homojen olarak uygulanmazlar. Bu yüzden 2007 yılında [13] tarafından yapılmış olan bir çalışmada, LİDAR

verisinde 3B bina çatı düzlemlerini otomatik tespit etmek için 3B Hough dönüşümü ve RANSAC algoritması kullanılmıştır. Hough-dönüşümü bölütleme parametreleri değerlerine çok duyarlı olduğundan dolayı, bina düzlemlerinin bölütlenmesi için genişletilmiş RANSAC algoritması kullanılmıştır. Genişletilmiş yaklaşımın çok zayıf nokta yoğunluğu ve farklı bina karmaşıklığı seviyeleri için çok tatmin edici sonuçlar verdiği gösterilmiştir.

2011 yılında [14] tarafından yapılan bir çalışmada, lazer tarama verilerinden çatı düzlemlerinin belirlenmesi için, çatı düzlemi bölütleme ve çatı modeli geriçatımı işlemlerini içeren yeni bir yaklaşım geliştirilmiştir. Bölütleme, çatıların geometrik yapılarının belirlenmesinde etkiliyken, geriçatmanın çatılardaki bitişiklik ve bütünlüğün daha iyi belirlenmesinde etkili olduğu belirtilmiştir. Burada kullanılan multiphase level set yöntemi iki adet lazer verisi üzerinde test edilmiştir. Çalışma sonucunda, çatı yapılarını temsil edecek yeterli yoğunlukta veri noktası yoksa yapılacak bölütleme işleminin nesneleri iyi bir şekilde ifade etmeyeceği belirtilmiştir. Çalışmanın devamında, önerilen yöntemin farklı çatı yapılarını ve karmaşık binaları içeren veriler ile test edileceği ifade edilmiştir.

2012 yılında [15] tarafından yapılmış olan bir çalışmada, yalın yer yüzeyi veya herhangi bir bina vb. detayın çıkarımı için, diğer metodlara göre daha etkili olan RANSAC algoritması uygulanarak binaları ve arazi yüzeyini tespit eden olası en iyi düzlemler belirlenmiştir. RANSAC algoritması ile tatmin edici sonuçlar elde edildiği söylenmiştir.

2014 yılında [16] tarafından yapılmış olan bir çalışmada, 3B yersel LİDAR nokta bulutu verisinde yer alan düzlem, koni, silindir ve küre gibi şekillerin yüzeylerinin RANSAC algoritması ile otomatik çıkarımı yapılmıştır. Burada yapılan işlemde doğruluğu incelemek için elle ve otomatik çıkarılan yüzeyler karşılaştırılmıştır. Yapılan incelemeler sonucunda 3B yersel LİDAR nokta bulutu verisinden otomatik yüzey çıkarım işleminde RANSAC algoritmasının oldukça etkili olduğu belirtilmiştir. RANSAC algoritmasının düzlem yüzeyi içeren nesnelerde düzlem çıkarmada başarılı olabileceği söylenmiştir.

2015 yılında [17] tarafından göl sınırlarının otomatik olarak belirlenmesi için LiDAR verileri kullanılarak yapılmış olan bir çalışmada, su yüzeyi RANSAC algoritması kullanılarak çıkarılmıştır. Göl alanlarının düzlemsellik özelliği göstermesinden dolayı bu alanların RANSAC algoritmasyla kolay bir şekilde çıkarıldığı söylenmektedir.

2016 yılında [6] tarafından yapılmış olan bir çalışmada, yalnız LİDAR verisi kullanılarak çatıların 3B modellemesi yapılmıştır. Bu çalışmada ayrıca, görüntüler kullanılarak oluşturulan 3B çizgilerin dahil edilmesiyle daha kaliteli sonuçların üretilebileceği söylenmiştir.

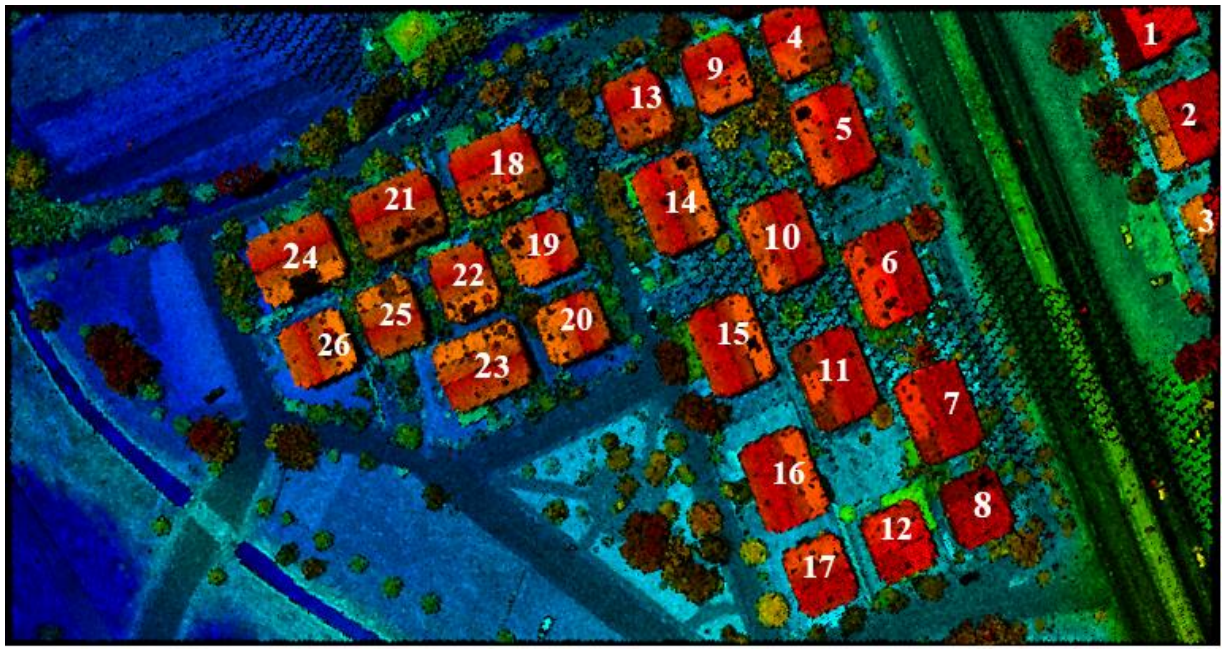
En güncel çalışmalardan olan 2017 yılında [18] tarafından yapılmış olan bir çalışmada, Bölge Büyüme algoritmasının aşırı bölütleme ve tutarsızlık gibi problemlerini gidermek için, farklı nokta bulutu modellerine kendinden uyarlanır (self-adaptive) bölütleme algoritması uygulanarak %96 seviyesinde bölütleme doğruluk oranı elde edilmiştir.

Bu tez çalışmasında, diğer çalışmalardan farklı olarak, yalnız 3B hava LiDAR nokta bulutu verisine Bölge Büyüme Bölütleme, RANSAC, DBSCAN ve Douglas-Peucker algoritmaları uygulanarak çatı düzlemleri çıkarılmış ve modellenmiştir.

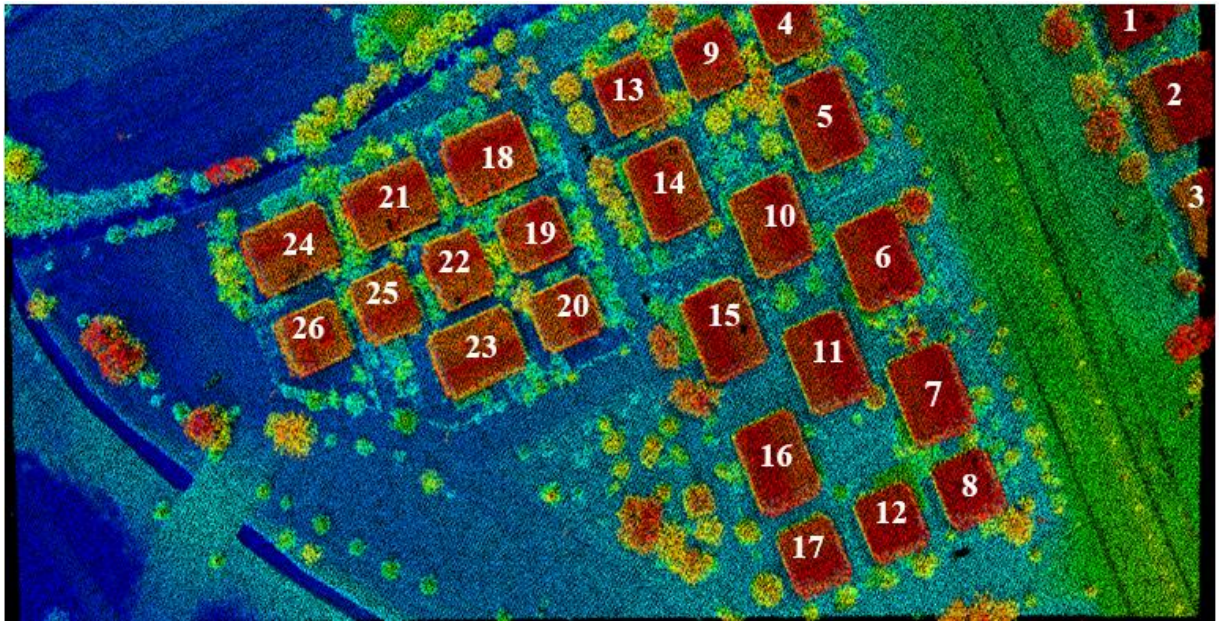
1.3. Çalışma Alanı ve Veri

Bu tez çalışmasında kullanılan İzmir ili Bergama ilçesine ait hava LiDAR verileri Harita Genel Komutanlığı (HGK) tarafından temin edilmiştir. Çalışmada bina ve ağaç detaylarını içeren test alanları seçilmiştir. Seçilen test alanlarından Şekil 1.1. (a)' da görüntüsü verilen birinci test alanı (Test alanı #1), 2600 metre uçuş yüksekliğinde Riegl firmasının LMS-Q1560 Lidar sistemi tarafından, Şekil 1.1. (b)'de görüntüsü verilen ikinci test alanı (Test alanı #2), 1200 metre uçuş yüksekliğinde Riegl firmasının LMS-Q1560 Lidar sistemi tarafından ve Şekil 1.1. (c)' de görüntüsü verilen üçüncü test alanı (Test alanı #3), 1200 metre uçuş yüksekliğinde Optech firmasının Pegasus HA-500 LiDAR sistemi tarafından çekilmiştir [19] .

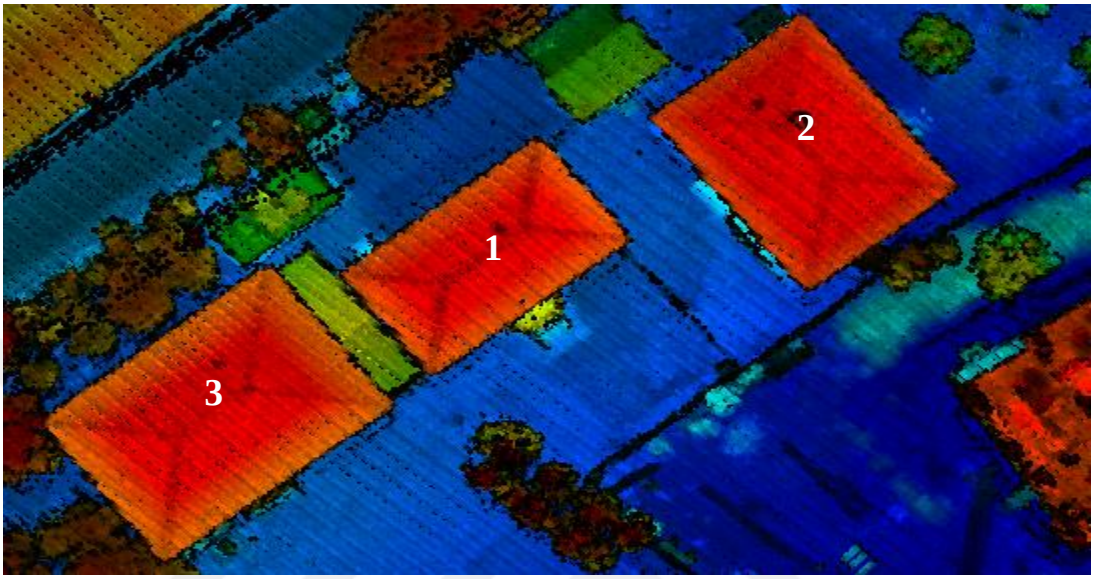
Test alanı #1'in koordinatları yaklaşık olarak, UTM 6 Derece WGS-84 Zone; 35 projeksiyon bilgisinde; X:513639.680 m - Y:4327874.751 m (Kuzey doğu) ve X: 513530.404 m – Y: 4327649.925 m (Güney batı)' dır. Test alanı #2'nin koordinatları yaklaşık olarak, UTM 6 Derece WGS-84 Zone; 35 projeksiyon bilgisinde; X:513639.313 m - Y:4327873.985 m (Kuzey doğu) ve X: 513530.237 m – Y: 4327649.844 m (Güney batı) ve Test alanı #3'ün koordinatları yaklaşık olarak, UTM 6 Derece WGS-84 Zone; 35 projeksiyon bilgisinde; X:516922.621 m - Y:4328832.454 m (Kuzey doğu) ve X: 516831.428 m – Y: 4328760.999 m (Güney batı)' dır.


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

a)


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

b)



c)

Şekil 1.1. Test alanları hava LiDAR nokta bulutu verisi. (a) Test alanı #1, (b) Test alanı #2, (c) Test alanı #3.

1.4. Tezin Organizasyonu

Bu tez çalışması 5 temel bölümden oluşmaktadır. Tezin 1. bölümünü giriş teşkil etmekte olup bu bölümde, binaların 3B şehir modellemesindeki önemi hakkında açıklamalarda bulunulmuş, hava LİDAR verisinden düzlemsel yüzeylerin belirlenmesi için geçmişte yapılmış olan çalışmalar hakkında bilgi verilmiş ve bu tez çalışmasının amaçları belirtilmiştir. Tezin 2. bölümünde, LiDAR nokta bulutu verisinin işlenmesi hakkında bilgi verilmiştir. Tezin 3. bölümünde, çalışmada kullanılan yöntemler anlatılmıştır. Tezin 4. bölümünde ise, test alanlarında gerçekleştirilen uygulamalardan elde edilen bulgular verilmiştir. Son olarak tezin 5. bölümünde, elde edilen bulgular kapsamında bu tez çalışması neticesinde varılan sonuçlar değerlendirilerek çalışma sonlandırılmıştır.

1.5. Kullanılan Yazılımlar

Bu tez çalışmasında LAStools, CMake, Microsoft Visual Studio, ve MATLAB yazılımları kullanılmıştır. İndeks oluşturma, gürültü giderme, örtüşen alanları belirleme, zemin ve nesne ayrımı (filtreleme), ağaç ve binaları sınıflandırma, bina sınıfını elde etme, nokta bulutu meta verisini inceleme ve nokta bulutunu ASCII metin formatına dönüştürme işlemleri Lastools yazılımı ile yapılmıştır. CMake yazılımı ile, Bölge Büyüme Bölütleme algoritmasının C++ kodu uygulanabilir duruma getirilmiştir. Nokta bulutu kütüphanesindeki uygulanabilir duruma getirilen bu C++ kodu Microsoft Visual Studio ortamında bina sınıfı nokta bulutu verisine uygulanmıştır. MATLAB ortamında ise, bölütlenmiş bina nokta bulutu kümelerinin her birine 3B RANSAC algoritması uygulanarak çıkarılan düzlem noktalarında, gürültüler tespit edilmiştir. MATLAB ortamında ayrıca, çatı düzlem sınırları çıkarılarak bu sınırlara sadeleştirme işlemi uygulanmıştır.

2.LiDAR NOKTA BULUTU VERİSİNİN İŞLENMESİ

2.1. LiDAR Nokta Bulutu Veri Yapısı

Algoritmalar tarafından işlenen en temel elemanlar veri olarak tanımlanmaktadır. Algoritmaların işleyeceği verilerin düzenlenmesi amacıyla, verinin veya bilginin bellekte tutulma şeklini veya düzenini gösteren temel veri yapıları vardır.

Las formatı tanımında kullanılan veri yapıları şunlardır [20]:

- char
- unsigned char
- short
- unsigned short
- long
- unsigned long
- long long
- unsigned long long
- float
- double
- string

LiDAR verisini depolamak için las formatı Amerika Fotogrametri ve Uzaktan Algılama Birliği (ASPRS: American Society for Photogrammetry and Remote Sensing) tarafından ikili (binary) formatta geliştirilmiştir. Las formatı; kullanıcılar, üretici, yazılım ve donanım arasında ortak kullanım imkanı veren bir format durumundadır. Önceleri ASCII (American Standard Code for Information Interchange) formatında saklanan veriler, LAS format ile beraber daha etkili ve verimli bir depolama formatına erişmiştir [21].

2.2. Las Format Tanımı

Las dosya formatı, bir Genel Başlık Bloğu, Değişken Uzunluk Kayıtları (VLRs: Variable Length Records), Nokta Veri Kaydı ve Genişletilmiş Değişken Uzunluk Kaydı (EVLRs: Extended Variable Length Records) bölümlerinden oluşur [20].

Her bir LAS dosyası, Genel Başlık Bloğundaki LiDAR ölçümlerinin meta verilerini içerir ve ardından kaydedilen her bir lazer darbesi için ayrı kayıtlar içerir [21].

Her LAS dosyasının Başlık Bloğu kısmında, uçuş tarihi, uçuş süresi, kaydedilen nokta sayısı, geri dönüş sayısı, uygulanılan ölçek faktörü gibi LiDAR ölçümleri ile ilgili öznitelik bilgiler depolanır.

VLRs projeksiyon bilgileri, meta veriler, dalgaboyu paket bilgisi ve kullanıcı uygulama verileri gibi değişken veri türleri içerir [20].

Noktalara Ait Veri Kayıtları ise; dönen sinyalin şiddeti (intensity), dönüş sayısı ve kaçıncı dönüş olduğu, nokta koordinatları (x,y,z), ve LiDAR'a özgü diğer öznitelik bilgileri (uçuş hattı, yüzey kategorisi vb.) içermektedir [22].

LiDAR nokta bulutu ile zemin filtrelemesi, gürültü giderme, bina ve ağaçları sınıflandırma vs. gibi bir takım işlemler yapılırken verinin ASPRS standart LiDAR Nokta Sınıfları, Nokta Veri Kayıt Formatına bakılarak öğrenilebilinir [20]. Çizelge 2.1.'de Nokta Veri Kayıt Formatı 6-10 arası olanlar için sınıflandırma değerleri ve anlamları gösterilmiştir.

Çizelge 2.1. ASPRS Standart LiDAR Nokta Sınıfları
(Nokta Veri Kayıt Formatı 6-10 arası olanlar)

Sınıflandırma Değeri	**Anlamı**
0	Sınıflandırılmamış
1	Sınıflandırılmamış
2	Zemin
3	Düşük Bitki Örtüsü
4	Orta Bitki Örtüsü
5	Yüksek Bitki Örtüsü
6	Bina
7	Gürültü
8	Rezerve
9	Su
10	Ray
11	Yol Yüzeyi

12	Rezerve
13	Tel
14	Tel
15	Elektrik Direği
16	Tel Yapısı Konektörü
17	Köprü Güvertesi
18	Yüksek Gürültü
16-63	Rezerve
64-255	Kullanıcı Tanımlı

2.3. Zemin Filtreleme

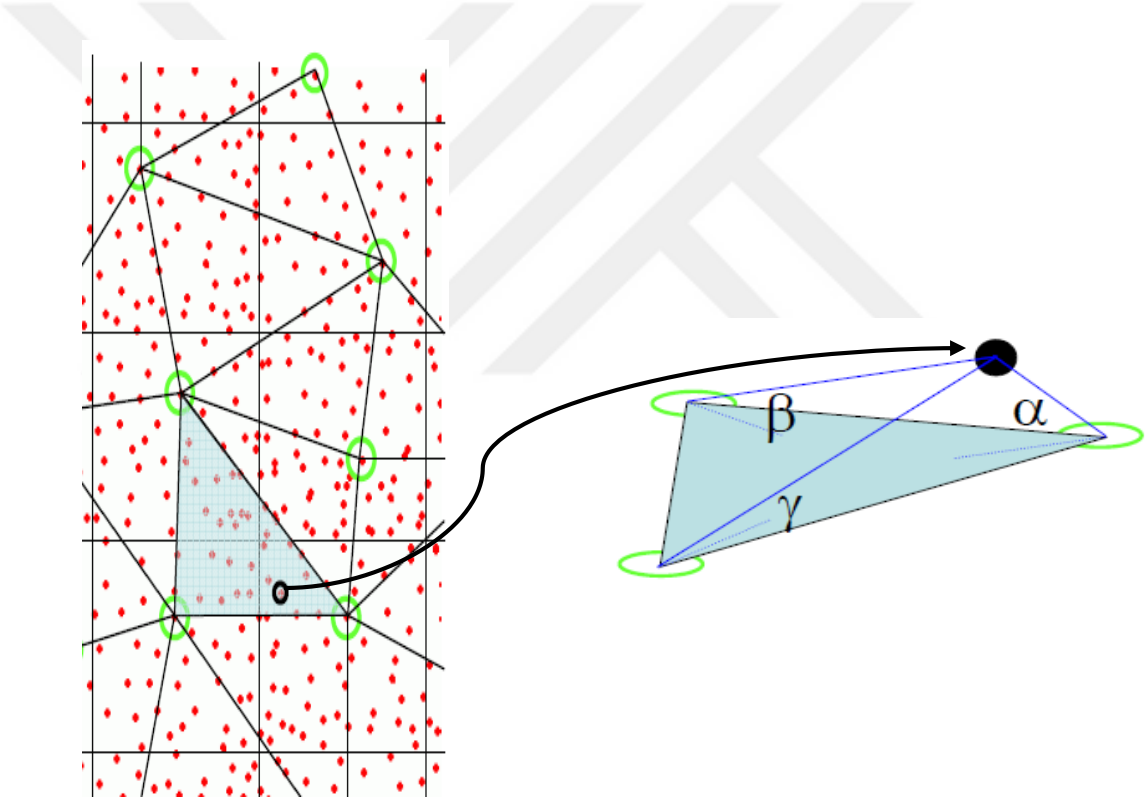
Çalışmanın bu kısmında, bu tez çalışmasında LAStools yazılımı "lasground" modülünde kullanılan, İlerleyen Üçgenlenmiş Düzensiz Ağ Yoğunlaştırma (Progressive Triangular Irregular Network (TIN) Densification) filtreleme yöntemi anlatılmıştır [23]. Bu yöntemde kaba bir TIN, tohum noktalarından (seed points) oluşturulup aşamalı olarak yoğunlaştırılır.

"lasground" modülünde kullanılan filtreleme yöntemi aşamaları şu şekildedir:

- Parametre Kestrimi: TIN yoğunlaşması için parametreler (TIN yüzeyine mesafe ve düğümlere olan açılar) veriden üretilir. Veriden elde edilen istatistikler, medyan değerlerin hızlı hesaplanmasını sağlamak için, yüzey normal açıları ve yükseklik farkının ayrı ayrı histogramları şeklinde toplanır. Medyan değerlere dayalı parametre eşik değerleri, histogramdan tahmin edilerek yinelemeli işlemde kullanılır.
- Tohum Noktalarının Seçimi: Şekil 2.1. (a)'da gösterildiği üzere, çalışma alanı karelajlara bölünerek her karelaj içerisinde en az bir yer noktası alma garantisi olacak şekilde filtreleme işlemi başlatılır. Her karelaj içerisinde en az bir yer noktası almak için karelaj boyutu, çalışma alanındaki en büyük nesne boyutundan daha büyük olmalıdır. Filtreleme işlemi yapılmadan önce, zeminden aşağıda bulunan (zemin olmayan noktalar) birtakım hatalardan dolayı ortaya çıkan gürültü noktalarının giderilmesi gerekir. Karelaj boyutu belirlendikten sonra karelajlar içerisinde yükseklik değeri en küçük olan noktalar (Şekil 2.1. (a)'da yeşil renk çember içine alınan noktalar) seçilir. Gürültü giderilmezse en düşük yüksekliğe sahip nokta yanlış olabilmektedir.
- TIN'in Yoğunlaştırılması: Karelajlar içersindeki en düşük yüksekliğe sahip noktalardan TIN oluşturulur. Sonrasında, her aşamada, hesaplanan eşik değer parametrelerine dayalı

kriterler karşılanırsa, üçgen sınırında bulunan noktalardan bir nokta her seferinde üçgenlemeye dahil edilir. Eşik değer parametreleri olan "*üçgen ile noktanın birleştirildiği kenarlar arasındaki açı*" ve "*nokta ile yakınında bulunan üçgen düzlemine olan mesafe*" her iterasyonda yeniden hesaplanır. Üçgen sınırı dahilindeki seçilen tek nokta (Şekil 2.1. (b)), oluşan üçgene belirtilen mesafeden yakın ve açılar belli bir sınırın altında ise yer noktası olarak kabul edilerek üçgen sıklaştırılır.

- Zemin veya nesne olarak tüm noktalar sınıflandırılana kadar TIN yoğunlaştırma işlemi devam eder [24], [25], [26], [27].



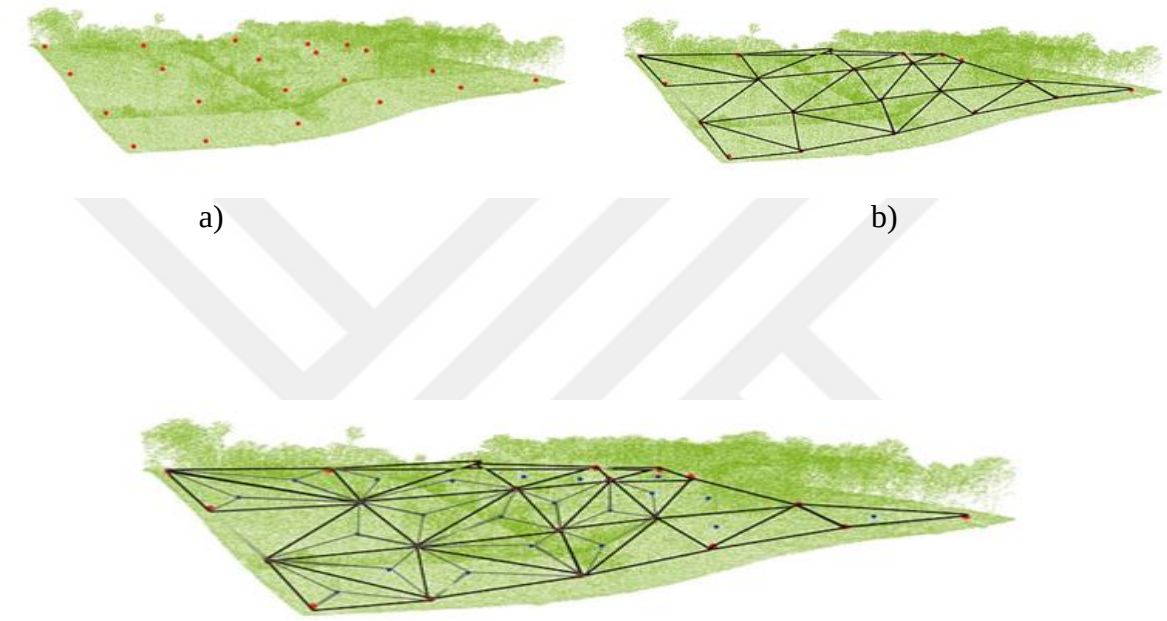
a) b)

Şekil 2.1. İlerleyen TIN Yoğunlaştırma filtreleme yöntemi a) Çalışma alanının karelajlara bölünmesi b) Üçgen sınırında bulunan noktalardan bir noktanın, noktadan üçgen kenarlarına olan açı eşik değer parametresinin gösterimi.

Kısaca, burada anlatılan filtreleme işleminin adımları şunlardan oluşmaktadır.

- Tüm veri kullanılarak başlangıç parametrelerinin hesaplanması.
- Tohum noktalarının seçimi (Şekil 2.2. (a))

- TIN'in oluşturulması (Şekil 2.2. (b)) ve aşamalı yoğunlaştırılması (Şekil 2.2 (c))
    a) TIN'de bulunan noktalardan her aşamada parametre hesaplanması
    b) Belirtilen eşik değerine göre TIN'e noktaların eklenmesi
- TIN'e nokta eklenmediğinde aşamalı sürecin durdurulması.



a) b)

c)

Şekil 2.2. İlerleyen TIN Yoğunlaştırma filtreleme yöntemi işlem adımları. a) Tohum noktalarının seçimi b) Arazinin kaba yaklaşımı için tohum noktalarndan üçgenleme yapılması c) Arazinin gerçek yapısına yakın bir gösterim elde etmek için araziye dahil edilen daha fazla nokta ile aşamalı yoğunlaştırma işlemi.

3. YÖNTEM

3.1. Yöntemin Genel Adımları

Yöntem, Şekil 3.1.'deki akış şemasında gösterildiği üzere altı temel aşamadan oluşmaktadır. Bunlar:

1) LAStools Yazılımı ile Ön İşlemler,
2) Bölge Büyüme Bölütleme Algoritması ile Bölütleme,
3) RANSAC Algoritması ile Düzlem Çıkarılması,
4) DBSCAN Algoritması ile Düzlemde Gürültü Ayıklama,
5) Bina Çatı Düzlem Sınırı Çıkarılması,
6) Douglas-Peucker Algoritması ile Bina Çatı Düzlem Sınırı Sadeleştirmesi dir.

Çalışmada, RANSAC algoritması ile otomatik bina çatı düzlemlerinin çıkarılması ve modellenmesi amaçlanmıştır.

LAStools yazılımı ile ön işlemler aşamasında, "lasindex" modülü ile indeks oluşturularak verinin olduğu adrese daha hızlı bir şekilde ulaşıp daha hızlı sonuç almak amaçlanmıştır. Başarılı sonuçlar almak için "lasnoise" modülü ile tek başına olan noktaları ayıklamak amaçlı gürültüler giderilerek veri sonraki aşamalara hazırlanmıştır.

Veri hakkında daha fazla bilgi elde etmek için verinin projeksiyon bilgileri "lasoverlap" modülünde sisteme tanıtılarak, verinin Google Earth uydu görüntüsünden incelenmesi yapılmış ve veride farklı şeritler arasında örtüşme ya da boşluklar olup olmadığı incelenmiştir.

"lasground" modülünde yer alan İlerleyen TIN Yoğunlaştırma filtreleme yöntemi ile LiDAR nokta bulutu üzerinde, LiDAR nokta bulutundan yer noktalarının belirlenmesi amaçlı, LiDAR noktaları zemin ve zemin olmayan noktalar şeklinde sınıflandırılmıştır. Filtreleme işlemi ile zemin ve zemin olmayan noktaların sınıflandırılmasının ardından, "lasclassify" modülü kullanılarak zeminden en az verilen değer kadar yukarı mesafedeki çatı ve bitki örtüsü komşu noktalarının, standart sapma değeri ile bulunması sağlanarak bina ve bitki örtüsü sınıflandırılması yapılmıştır. Sonrasında sınıflandırılmış olan tüm veriden (zemin, zemin olmayan, ağaç ve bina) "las2las" modülü kullanılarak bina sınıfı elde edilmiştir. Çalışma alanlarına ait nokta bulutunun, verideki toplam nokta sayısı, dönüş numarasına göre nokta

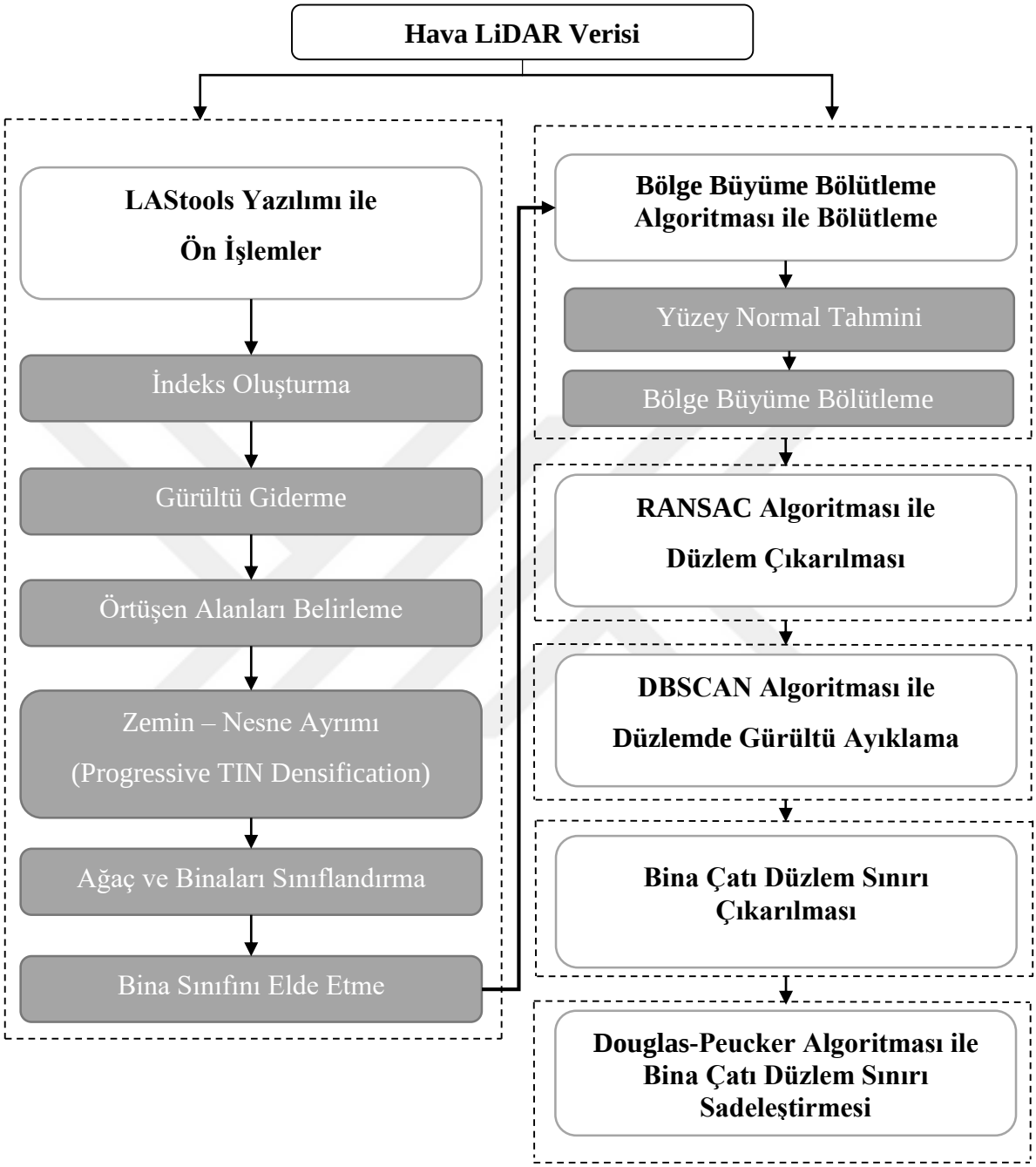
sayıları, sınıflandırmaya göre nokta sayıları, nokta yoğunluğu ve nokta sıklığı gibi las dosyasına kaydedilen meta-verisi “lasinfo” modülü ile incelenerek tüm alanların metre kareye düşen ortalama nokta sayısı elde edilmiştir. Sonraki aşamalarda girdi olarak kullanılmak üzere bina sınıfına ait “.las” uzantılı üç boyutlu nokta bulutu x, y ve z koordinatları, ASCII metin formatında dönüştürülerek ön işlem adımları tamamlanmıştır.

Veri üzerinde ön işlemler gerçekleştirildikten sonra, ticari ve araştırma amaçlı ücretsiz açık kaynak Nokta Bulutu Kütüphanesi (PCL: Point Cloud Library) tarafından elde edilen Bölge Büyüme Bölütleme algoritması C++ kodları ile “Visual Studio” yazılımında bina sınıfı nokta bulutuna bölütleme işlemi uygulanarak tek bina kümelerine ait nokta bulutu elde edilmiştir.

Bölütleme sonrası elde edilen nokta bulutu kümelerinin tamamına ayrı bir şekilde 3B RANSAC algoritması uygulanarak her bir binayı temsil eden çatı düzlemlerine ait noktaların çıkarılması sağlanmıştır.

Çıkarılan düzlemlere ait nokta bulutlarında düzlemde olması beklenmeyen gürültü noktaları DBSCAN algoritması ile kaldırılmıştır.

Yöntem, sonrasında MATLAB ortamında bina sınırlarının çıkarılması adımlarını takip ederek sınırı temsil eden noktalara, bina sınırı sadeleştirilmesi amaçlı Douglas-Peucker algoritmasının uygulanması ile sona ermektedir.



Şekil 3.1. Yöntemin Akış Şeması.

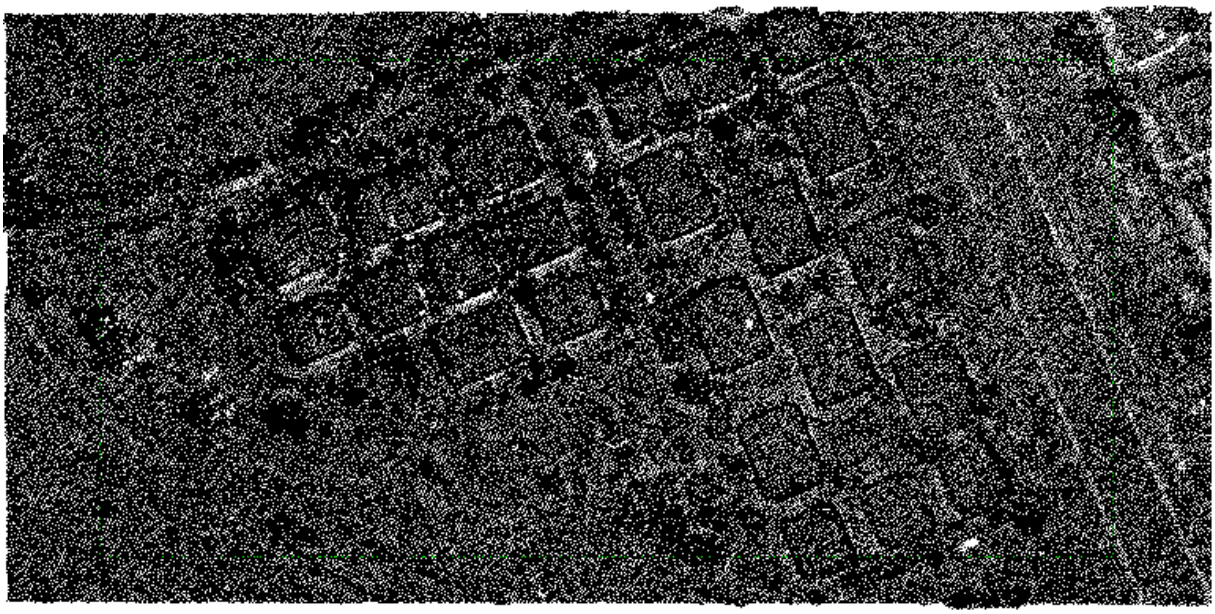
3.2. Ön İşlemler

Ön işlemler, şu aşağıdaki adımlardan oluşmaktadır:

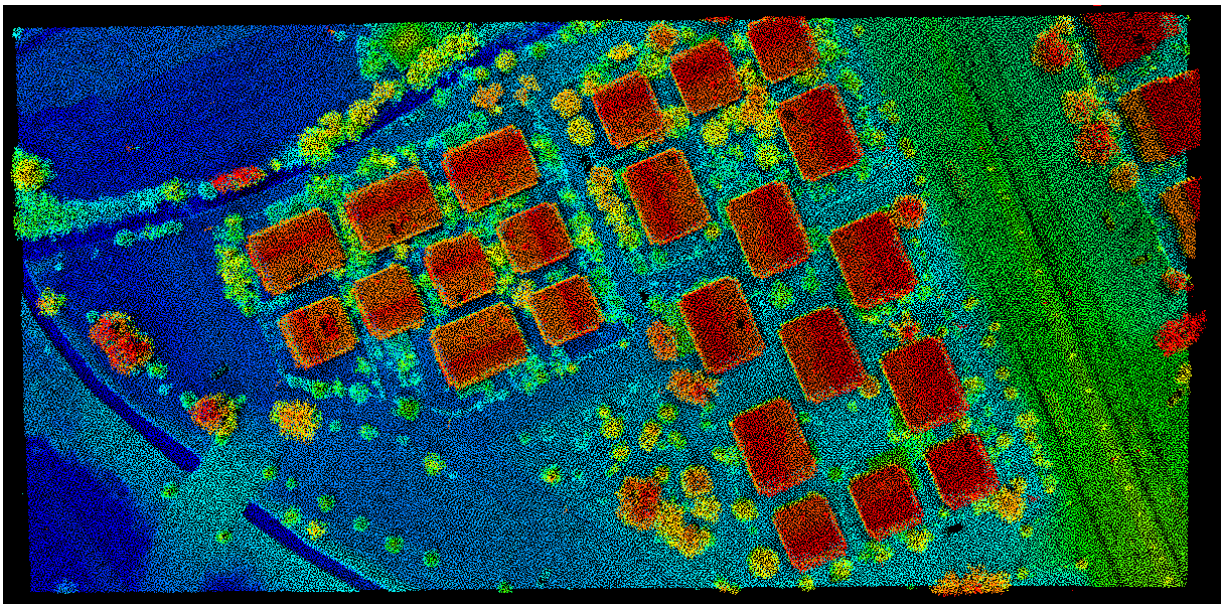
- İndeks oluşturma,
- Gürültü giderme,
- Örtüşen alanları belirleme,

- Zemin ve nesne ayrımı,
- Ağaç ve binaları sınıflandırma,
- Bina sınıfını elde etme.

Ön işlemler, Şekil 3.2. ve Şekil 3.3.'te gösterilen test alanı #2 üzerinden gerçekleştirilmiştir.



Şekil 3.2. Test alanı #2'ye ait LiDAR verisinin gösterimi.



Şekil 3.3. Test alanı #2'ye ait LiDAR verisinin yükseklik değerlerine göre renkli gösterimi.

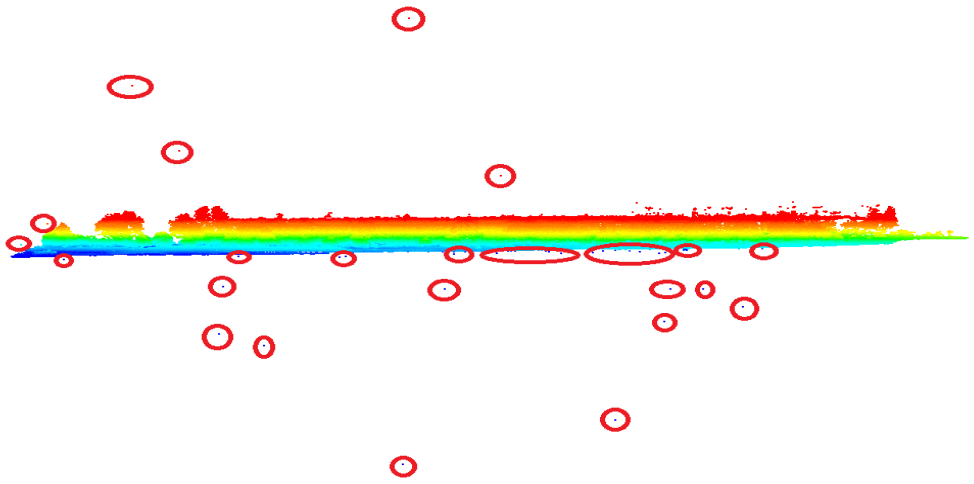
3.2.1. İndeks Oluşturma

Verinin indeksli olması, verinin olduğu adrese daha hızlı bir şekilde ulaşılıp, her işlemde daha hızlı sonuç alınmasına olanak sağlamaktadır. İndeks oluşturma işlemi "LAStools" yazılımında "lasindex" modülü kullanılarak yapılmıştır. Mekansal sorgulamalar "lasindex" modülü ile hızlandırılmaktadır [28].

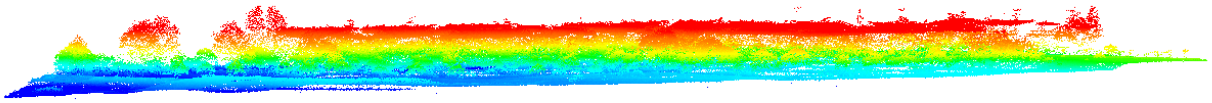
3.2.2. Gürültü Noktalarının Kaldırılması

Verinin sınıflandırılamamış olarak gelmesi durumunda veriyi yapılacak işleme göre hazırlamak gerekir. Dolayısıyla bu çalışmada, veri görüntülendiğinde gürültüler içerdiği ve herhangi bir sınıflandırma içermediği görülmüş olup bu doğrultuda filtreleme yapılmadan önce veri gürültüden arındırılmıştır. Bunun için "LAStools" yazılımının "lasnoise" modülü kullanılmıştır [29]. Bu modül belli kriterlere göre izole olmuş noktaları aramaktadır. Çalışmada "*step_xy*" , "*step_z*" ve "*isolated*" girdi parametreleri için sırasıyla 1,1 ve 5 değerleri kullanılmıştır. Verilen değerler sonucunda gürültü olarak elde edilen veriler, "remove noise points" seçeneği ile veriden arındırılmıştır.

Nokta yoğunluğuna bağlı olarak izole noktaların ayıklanması amaçlanan bu çalışmada, kullanılan parametre değerlerine göre, 3x3x3 karelaj etrafında 27 hücrenin her bir hücresinin x, y ve z boyutu 1x1x1 metre birim boyutunda olup 5 veya daha az başka noktaya sahip tüm noktalar gürültü olarak belirlenerek kaldırılmıştır (Şekil 3.4, Şekil 3.5). Verilere bu modül uygulanırken tüm test alanları için aynı parametre değerleri girilmiştir.



Şekil 3.4. Veriye ait gürültü noktalarının bir kısmının profilden görüntüsü.



Şekil 3.5. Gürültüden arındırılmış nokta bulutunun profilden gösterimi.

3.2.3. Örtüşen Alanların Belirlenmesi

Büyük alanlara ait LiDAR verisi genellikle şeritler halindedir. Bu sebeple çalışma alanının hangi bölgesinde kaç şeridin örtüştüğü incelenmiştir. Bir noktanın diğer bir nokta üzerine düşüp düşmeyeceği belli bir örnekleme aralığı içerisinde olacağı için bu çalışmada karelajlaması yapılan LiDAR noktalarının 1x1 metre kare piksel (veya raster) içerisinde örneklenmesinin yapılması amacıyla "*pixel/step size*" parametresi değeri 1 girilmiştir.

"lasoverlap" modülünden elde edilen örtüşme raster verisinde, örtüşme sayıları için bir renk aralığı kullanılır [30]. Tek şerit dışındaki birden çok şerit olan verilerde, belirtilen metre karelik her piksel üzerinde kaç şeritten gelen nokta varsa onlar farklı renkte renklendirilir. Örtüşme raster verisinde en fazla örtüşen alanların gösterimi için üst sınırın kırmızı renk olduğu LAStools'un eğitici amaçlı sitesinde var olan yedi şeritli verisinde yapılan çalışma ile deneysel olarak elde edilmiştir. LAStools'un eğitici amaçlı sitesinde var olan yedi şeritli verisinde "Max overlap count" seçeneğine her defasında farklı değerler verildiğinde elde edilen örtüşme raster verisinde farklı renklerde, üç farklı renk oluştuğu için bu alanda en fazla üç şerit örtüştüğü bilgisi çıkarılmıştır. Dolayısıyla "Max overlap count " seçeneğine 3 değeri verildiğinde üç şeridin örtüştüğü alan kırmızı olarak görülmüştür.

Bu çalışmada ise "Max overlap count " için 5 değeri kullanıldığında açık mavi tonlarında tek bir renk görülmüştür. Bu veri için tek renk görülmesi tek bir şeritten gelen veri olduğu anlamını taşımaktadır ve bundan sonra verilecek değerlerde de tek bir şerit olduğu için tek bir renk görülür. Bir şerit olduğu için 1 değerini verdiğimizde kırmızı ve tek renk olarak görüntü elde edilmektedir.

Yapılan çalışmada tüm test alanları için aynı parametreler girilerek aynı sonuçlara erişilmiştir. Test alanı #2 Şekil 3.6'da görüldüğü gibi tek bir şeritte yer almaktadır.





“Max overlap count ” 5 “Max overlap count ” 1

Şekil 3.6. Farklı şeritlerden gelen gelen veri olup olmadığının “Max overlap count ” seçeneği ile incelenmesi

Veriye “lasoverlap” modülünde projeksiyon bilgileri eklenerek işlem sonucunda üretilen“.kml” uzantılı dosya ile bu alan Google Earth’de görsel olarak incelenmiştir (Şekil 3.7).



Şekil 3.7. Google Earth’te test 2#’ye ait alanın gösterimi.

3.2.4. Zemin ve Nesne Ayrımı

LiDAR noktalarını zemin (Şekil 3.8.) ve zemin olmayan noktalar (Şekil 3.9.) şeklinde sınıflandırmak için "lasground" modülü kullanılarak zemin noktaları ayrılmış ve kodlanmıştır (etiketlenmiştir). Bu kodlamanın detayları ASPRS LAS şartnamesi 1.4-R13 versiyonunda görülebilir [20]. "LAStools" yazılımında sınıflandırmanın bu şartnameye uyularak yapıldığı görülmektedir [31].

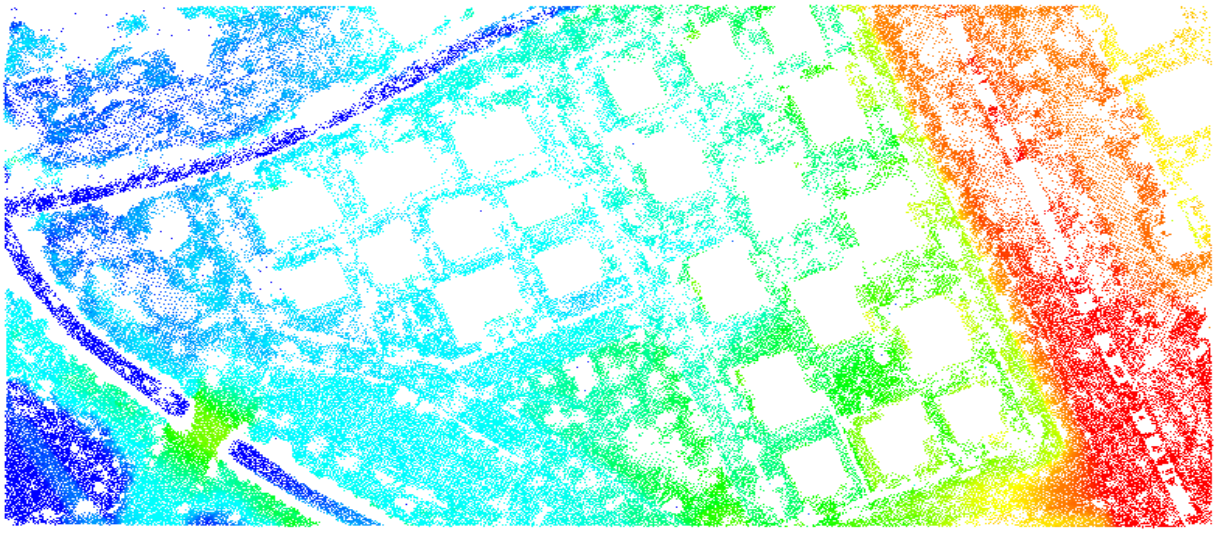
Bu aşamada, kullanılan "lasground" modülünde uygulanan filtreleme yöntemi İlerleyen TIN Yoğunlaştırma olup bu yöntem, LiDAR nokta bulutları üzerinde LiDAR nokta bulutundan yer filtrelemesinde (yer noktalarının belirlenmesinde) kullanılır [23].

"lasground" modülünün temel parametreleri şunlardır:

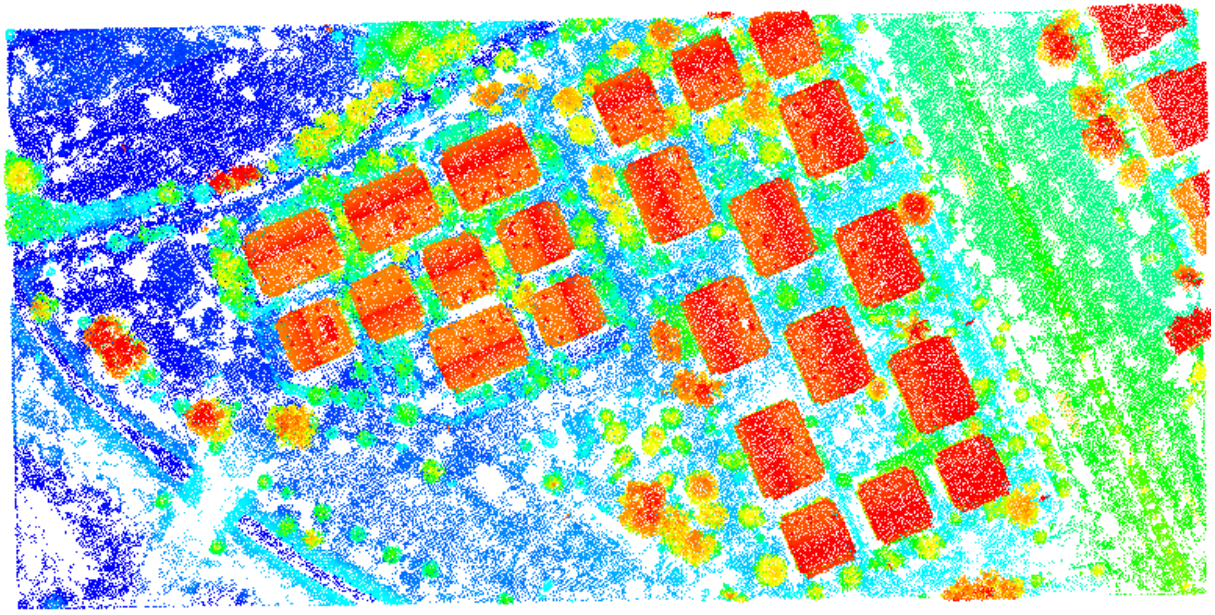
- "*step size*" parametresi: Bu parametre değeri, filtrelenmiş alandaki en büyük nesnenin boyutu büyüklüğünde olmalıdır [32]. Bu yüzden çalışmaya başlamadan önce yaklaşık olarak verideki en büyük nesnenin uzunluğunun tespit edilmesi önemlidir.
- "*spike*" parametresi: Filtreleme sürecinde bu parametre önemli rol oynamaktadır. Bu parametre, LiDAR noktalarının araziye kabul edildiği en kaba TIN üzerindeki mesafeyi açıklaması bakımından, sonuçları önemli ölçüde etkilemektedir. Bu parametre gürültüyü ayıklar, aksi taktirde en düşük nokta yanlış olabilir.
- "*bulge*" parametresi: TIN'de değişiklikler yapılarak TIN'e noktalar dahil edilirken TIN'in ne kadar yükselmesine izin verileceğini belirler. "bulge" parametresi, 5 metreden büyük "*step size*" parametresi için, bu parametrenin onda biri aksi halde beşte biri kadardır.
- "*standart sapma*" parametresi: Arazi yüzeyine yakın olan nesnenin kaldırılması için daha küçük bir "standart sapma" değeri kullanmanın iyi sonuçlar verdiği görülmektedir.
- "*maksimum offset*" parametresi: Bütün işlemler yapıldıktan sonra, son aşamada bu parametre kullanılarak hangi noktaların mevcut zemine dahil edileceği ayarlanmaktadır.

"lasground" modülü temelde "*step size*" parametresine dayandığı için bu parametrenin, verideki en büyük nesnenin büyüklüğünden küçük olması durumunda yer filtrelemesinde gerekli doğru sonuç elde edilemeyebilinir. Şöyle ki şehir veya kasaba için en büyük nesnenin bina olması beklenmektedir. Buralarda "*step size*" parametresi değerinin bina büyüklüğünden küçük olması, filtreleme işleminin sonucunu olumsuz etkilemektedir. Bu tez çalışmasında kullanılan test alanlarının tümünde "*step size*" parametre değeri verideki en büyük nesneler olan binalar dikkate alınarak belirlenmiştir [33].

Çalışmanın bu aşamasında Çizelge 3.1'de görüldüğü gibi en uygun parametrelerin belirlenmesi için çok sayıda deneme yapılmıştır. Bunun için "*step size*", "*standart sapma*" ve "*bulge*" parametreleri değerleri sabit tutulmuş "*maksimum offset*" ve "*spike*" parametresi sabit tutulan parametrelere göre değiştirilmiştir. Değiştirmede "*maksimum offset*" parametresi için 0.01'lik artırım "*spike*" parametresi için ise 0.05'lik artırım yapılmıştır. Kullanılan yazılımın eğitici amaçlı sitesinde benzer çalışma alanı için "*step size*" parametresi değerinin 10'a kadar artırıldığı önerilmektedir. Dolayısıyla, "*step size*" parametre değeri için 10, "*standart sapma*" parametresi değeri için 1 ve "*bulge*" parametre değeri için ise "*step size*" parametresi değerinin onda biri olan 1 değeri sabit alınıp "*maksimum offset*" ve "*spike*" parametresi değiştirilmiştir. Her parametre seti ile elde edilen sonuçların değerlendirilmeleri, sonuç yüzeylerinin gölgeli rölyef görüntüsü üzerinden yapılarak en uygun parametre değerleri Çizelge 3.2'de görüldüğü gibi belirlenmiştir. Filtreleme işlemi son dönüş noktaları üzerinden gerçekleştirilerek sonuç görüntüler Şekil 3.8. ve Şekil 3.9.'da görülmektedir.



Şekil 3.8. Zemine ait nokta bulutunun yükseklik değerlerine göre renkli gösterimi.



Şekil 3.9. Zemine ait olmayan nokta bulutunun yükseklik değerlerine göre renkli gösterimi.

Çizelge 3.1. Veriye en uygun parametrenin belirlenmesi için yapılan denemeden bir kesit.

STEP(m)	STDDEV(cm)	OFFSET(m)	SPIKE(m)
5	1	0.01	0.20
5	1	0.01	0.35
5	1	0.01	1.00
7	1	0.02	0.25
7	1	0.02	0.50
7	1	0.02	1.00
10	1	0.05	0.30
10	1	0.05	1.00
10	1	0.05	3.00

Çizelge 3.2. Görsel olarak yapılan denemeler sonucunda öngörülen en uygun parametre değerleri.

STEP(m)	STDDEV(cm)	OFFSET(m)	SPIKE(m)	BULGE(m)
10	1	0.05	1	1

Tüm test alanları için, öngörülen en uygun parametre değerleri girilmiştir.

3.2.5. Yüksek Bitki Örtüsü (Ağaç) ve Binaları Sınıflandırma

Bu aşamada yüksek bitki örtüsü ve binaları sınıflandırma işlemi için "lasclassify" modülü kullanılmıştır. Ancak bu modül kullanılmadan önce bu işlemin yapılabilmesi için "lasheight" modülü ile zeminin üzerindeki her bir noktanın yüksekliğinin hesaplanması gerekmektedir. Filtreleme aşamasında "compute height" seçeneği ile zeminin üzerindeki her bir noktanın yüksekliği hesaplandığı için "lasheight" modülüne gerek kalmadan yüksek bitki örtüsü ve binaları sınıflandırma işlemi "lasclassify" modülü kullanılarak yapılmıştır [34]. "lasclassify" modülü, metre cinsinden zeminden en az verilen değer kadar yükseklikte bulunan komşu noktaları bulmaya çalışmaktadır. Çok düşük nokta yoğunluğu olan verilerde genellikle "lasclassify" modülünün başarısız olduğu görülmektedir.

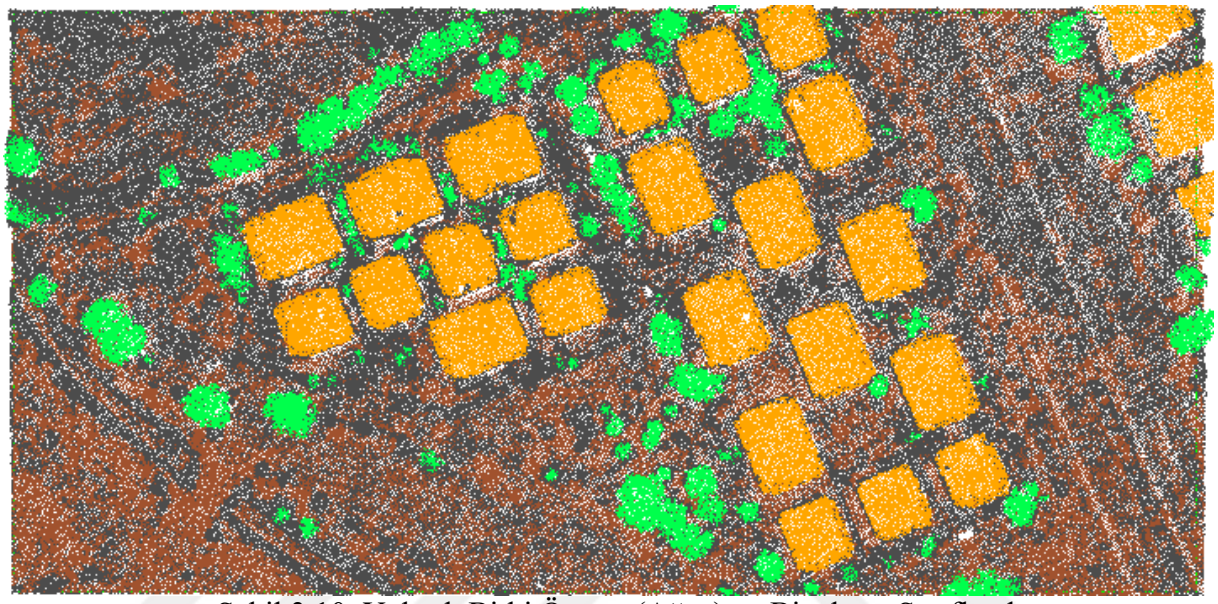
"lasclassify" modülünün, temel parametreleri şunlardır:

- "*ground offset*" parametresi: Bu yükseklik eşik değeri parametresi ile zeminden, girilen değer kadar yükseklikteki komşu noktaların bulunması sağlanmaktadır.
- "*search area size (step)*" parametresi: Bu parametre ile metre cinsinden nxn karelej hücre boyutu tanımlanmaktadır.
- "*building planarity*" parametresi: Bu parametre metre cinsinden girilen standart sapmaya kadar, karelaj hücre noktalarının potansiyel çatılar olduğu belirtilmektedir.
- "*forest ruggedness*" parametresi: Bu parametre metre cinsinden girilen standart sapmaya kadar, karelaj hücre noktalarının potansiyel bitki örtüsü olduğu belirtilmektedir.

Bu çalışmada sınıflandırmanın iyileştirilmesi ve uygun parametre değerlerinin belirlenmesi için çok sayıda denemeler yapılmıştır. Tüm test alanında, "*search area size*" prametresi için 1,2 ve 3, "*building planarity*" ve "*forest ruggedness*" parametresi için 0.1 artışla 0.1'den 1.0'a kadar ve son olarak "*ground offset*" parametresi için 1,2,3,4,5' değerleri kullanılmıştır. Bu işlem sonucunda, "*search area size*", "*building planarity*", "*forest ruggedness*", "*ground offset*" parametreleri için en uygun değerler sırasıyla birinci test alanı için 2, 0.3, 0.1, 3 ikinci ve üçüncü test alanları için ise 2, 0.7, 0.5, 4 olarak belirlenmiştir.

Yukarıdaki açıklamalar ışığında bina ve ağaçların bulunması için "ground offset" parametresi belirlenirken çalışma alanındaki en alçak binanın yüksekliği göz önüne alınmıştır. Bunun sebebi komşu noktaların bulunmasının bu mesafeden sonra dikkate alınmasıdır. Diğer parametreler

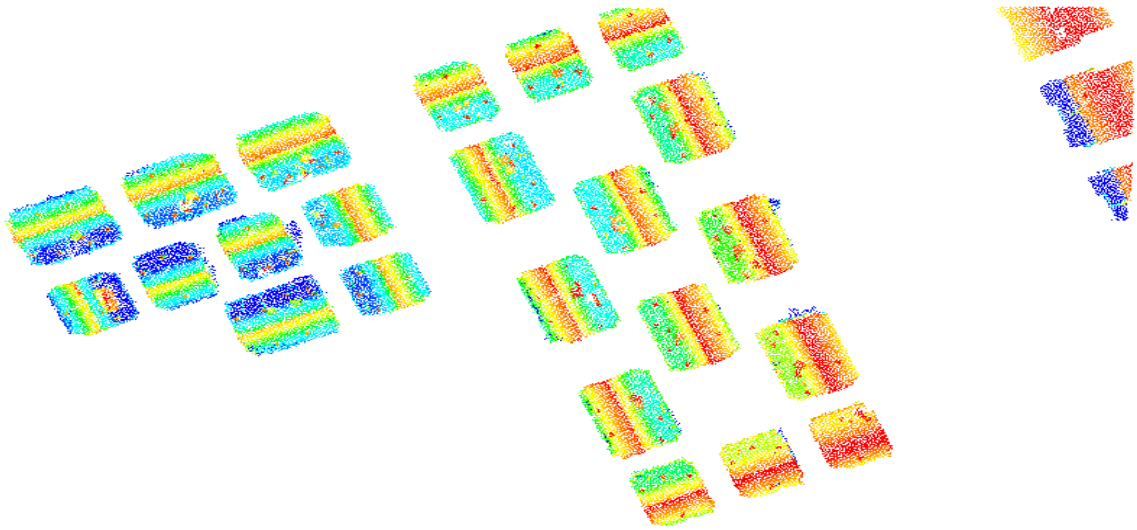
sabit tutulduğunda, “building planarity” için girilen standart sapma değeri artırıldığında çatıya daha fazla nokta dahil olduğu saptanmıştır. Bu modülde işlem yaparken ağaçların bina üzerine sarkması ve arazide yükseklik farkının olması sıkıntılar doğurabilmektedir. Yapılan işlemin sonucu Şekil 3.10.’da gösterilmiştir.



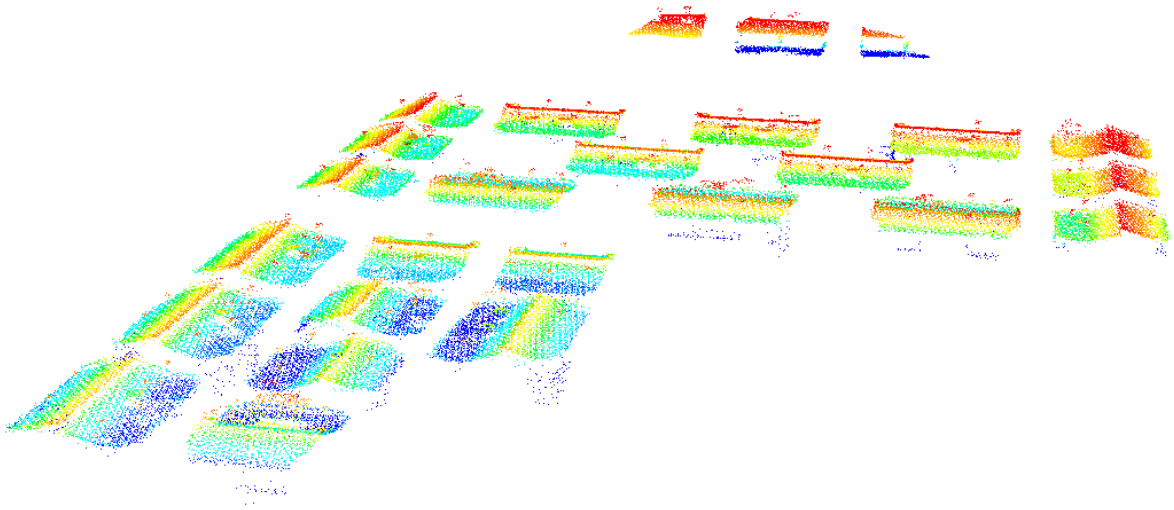
Şekil 3.10. Yüksek Bitki Örtüsü (Ağaç) ve Binaların Sınıflandırması.

3.2.6. Bina Sınıfının Elde Edilmesi

Çalışmanın bu aşamasında “las2las” modülünün içinde yer alan “filter” kısmı kullanılarak sınıflandırılmış olan tüm veriden, yalnız bina sınıfının çıkarılması işlemi yapılmıştır [35]. Elde edilen bina sınıfları, üstten Şekil 3.11.’de ve diğer bir açıdan Şekil 3.12’de gösterilmektedir.



Şekil 3.11. Elde edilen bina sınıfının yükseklik değerlerine göre gösterimi.



Şekil 3.12. Elde edilen bina sınıfının yükseklik değerlerine göre farklı açıdan gösterimi.

3.2.7. Nokta Bulutu Meta Verisi

Çalışma alanına ait nokta bulutunun, las dosyasına kaydedilen meta-verisi **"**lasinfo" modülü kullanılarak incelenmiştir. "Lasinfo" modülüne "-cd" (compute density) seçeneği eklenerek metre kareye düşen ortalama nokta sayısı (nokta yoğunluğu) hesaplanmıştır [36]. Bu modül tüm test alanları ham verilerinin yalnız son dönüşleri için uygulandığında, metre kare başına nokta yoğunluğu test alanı #1 için 3.42, test alanı #2 için 18.77 ve test alanı #3 için 22.63 olarak bulunmuştur.

3.2.8. Nokta Bulutunu ASCII Metin Formatına Dönüştürme

LAStools yazılımında "las2txt" modülü kullanılarak, bina sınıfına ait ".las" uzantılı üç boyutlu nokta bulutunun bölütleme aşamasında girdi olarak kullanılması için x,y ve z koordinatları, aralarına virgül bırakılacak şekilde ASCII metin formatında dönüştürülmüştür [37].

3.3. Bölütleme

Ticari ve araştırma amaçlı ücretsiz açık kaynak yazılımı olan PCL tarafından elde edilen Bölge Büyütme Bölütleme algoritması C++ kodları, tüm test alanlarına uygulanmıştır. Algoritma, eğrilik ve yüzey normalleri arasındaki açı değerleri kriterlerini göz önüne alarak bir bina kümesine ait olan benzer özellikteki noktaları gruplandırmaktadır [38].

Bir bina kümesine ait benzer özellikteki noktalar gruplandırılırken, öncelikle bütün noktaların yüzey normallerinin hesaplanması gerekmektedir. Bu çalışmada her noktada yüzey normallerinin doğrudan hesaplanması, yüzeye teğet bir düzlemin normalini tahmin etme problemi ile benzetilmektedir. Dolayısıyla bu çalışmada yüzey normal tahminine yönelik olan çözüm, sorgu noktasının en yakın komşularının Kovaryans Matrisi ya da Temel Bileşenler Analizi öz vektörleri ve öz değerlerinin analizini yapmaktır [39].

Sorgu noktasının k-komşuluğunda bulunan (k mesafesine küçük ve eşit nokta seti) nokta setinin belirlenmesi için, sorgu noktasının bir yarıçap (r) kadar tüm k komşularını belirlemek yerine, sorgu noktasının k en yakın komşularının belirlenmesi (K-Search: K-Arama) seçimi yapılmıştır [40].

Yüzey normallerinin hesaplanmasının ardından C++ kodu bölütleme işlemini gerçekleştirmektedir. Bölge Büyüme Bölütleme algoritmanın amacı, pürüzsüzlük kısıtlaması bakımından birbirine yakın olan noktaları bir araya getirmektir [38]. Yüzey normalinin hesabına dayalı bu algoritmanın çıktısı, girilen parametre değerlerine göre aynı pürüzsüz yüzeye sahip noktalar kümesini oluşturmaktır. Bu algoritmanın işleyişi, nokta normalleri arasındaki açıların karşılaştırılmasına dayanmaktadır.

Algoritmanın takip ettiği işlem adımları şu şekildedir:

1- En küçük eğrilik değerine sahip nokta, düz/pürüzsüz alanda yer almaktadır. Bundan dolayı, noktalar eğrilik değerlerine göre sıralanmaktadır. Noktaların eğrilik değerleri, kovaryans matrisinin öz değerleri ile Eşitlik 3.1.'te görüldüğü gibi hesaplanmaktadır. Böylece, bölge büyüme bölütleme işlemi en küçük eğrilik değerine sahip olan noktadan başlamak suretiyle yapılmaktadır.

2- Seçilen en küçük eğriliğe sahip nokta, tohum (seed) denilen kümeye eklenir.

3- Algoritma her tohum noktası için komşu noktalarını bulur.

4- Her komşu noktasının normali ile mevcut olan tohum noktasının normali arasındaki açı test edilir. Açı belirtilen eşik değerden az olur ise nokta mevcut bölgeye eklenir.

5- Her komşu noktası eğrilik değeri için test edilir. Eğrilik eşik değerden küçük ise o zaman bu nokta tohum noktalarına eklenir.

6- Son olarak mevcut tohum noktası tohum noktalarından kaldırılır.

Tohum kümesi boşalırsa, bu algoritmanın bölgeyi büyüttüğü ve işlemin baştan tekrarlandığı anlamına gelir.

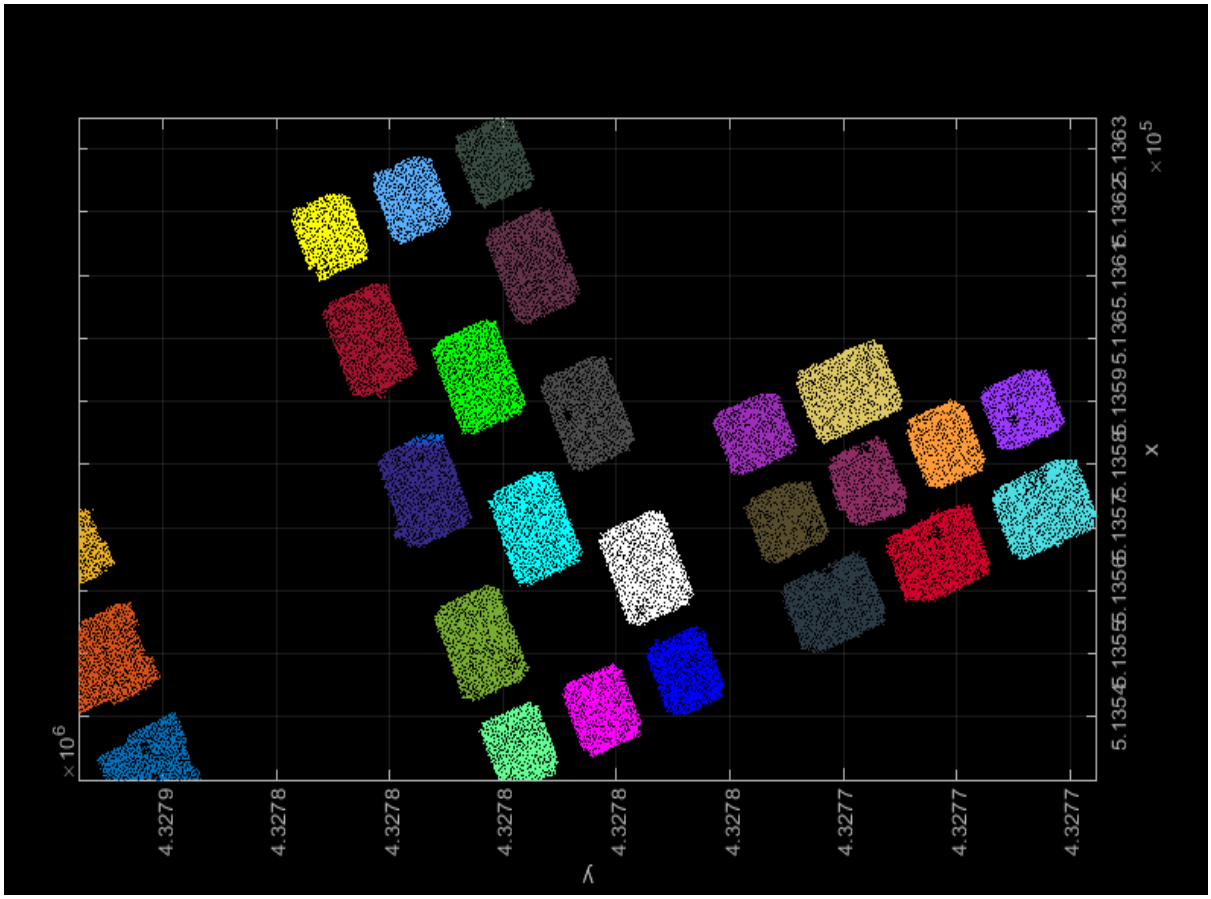
$$\sigma = \frac{\lambda 0}{\lambda 0 + \lambda 1 + \lambda 2} \tag{3.1}$$

Bu eşitlikte σ yüzey eğriliğini λ0, λ1, λ2 ise kovaryans matrisinin öz değerlerini temsil etmektedir.

Uygulamada yüzey normallerine dayalı Bölge Büyüme Bölütleme algoritmasının uygun parametrenin belirlenmesi için çok sayıda demene yapılmıştır. Tüm test alanları için, "*K-Search*" parametresine 2,3,4,5,10,20,30,50,100, "*Number Of Neighbours*" parametresine 4,6,8,15,22,30, "*Angle Threshold (derece cinsinden)*" parametresine 0.3,0.5,1,3,3.3,3.7,4,5.5 ve "*Curvature Threshold*" parametresine 0.5,1,3 değerleri verilmiştir. Bu işlem sonucunda, "*K-Search*", "*Number Of Neighbours*", "*Angle Threshold*", ve "*Curvature Threshold"* parametreleri için tüm sonuçlar incelendiğinde, en uygun değerler sırasıyla 50,22,3 derece ve 1 olarak (Çizelge 3.3.) belirlenmiştir. Sonuç olarak, bina sınıfına ait tüm binalar Şekil 3.13.'te gösterildiği gibi otomatik olarak ayrı bir şekilde çıkarılmıştır.

Çizelge 3.3. Bölge Büyüme Bölütleme algoritması parametreleri.

Test Alanı	#1	#2	#3
Yüzey Normali Tahmini	Değer		
Arama Yöntemi	k-d tree	k-d tree	k-d tree
K-Arama	50	50	50
Bölge Büyütme Bölütleme	Değer		
En Düşük Küme Büyüklüğü	50	500	5000
En Yüksek Küme Büyüklüğü	750	15000	10800
Arama Yöntemi	k-d tree	k-d tree	k-d tree
Komşu Sayısı	22	22	22
Açı Eşik Değeri (derece)	3	3	3
Eğrilik Eşik Değeri	1	1	1



Şekil 3.13. Bölge Büyüme Bölütleme algaritmasının uygulanması ile bölütlenmiş 26 adet bina nokta bulutu kümesi. Her bir renk bir binayı temsil etmektedir.

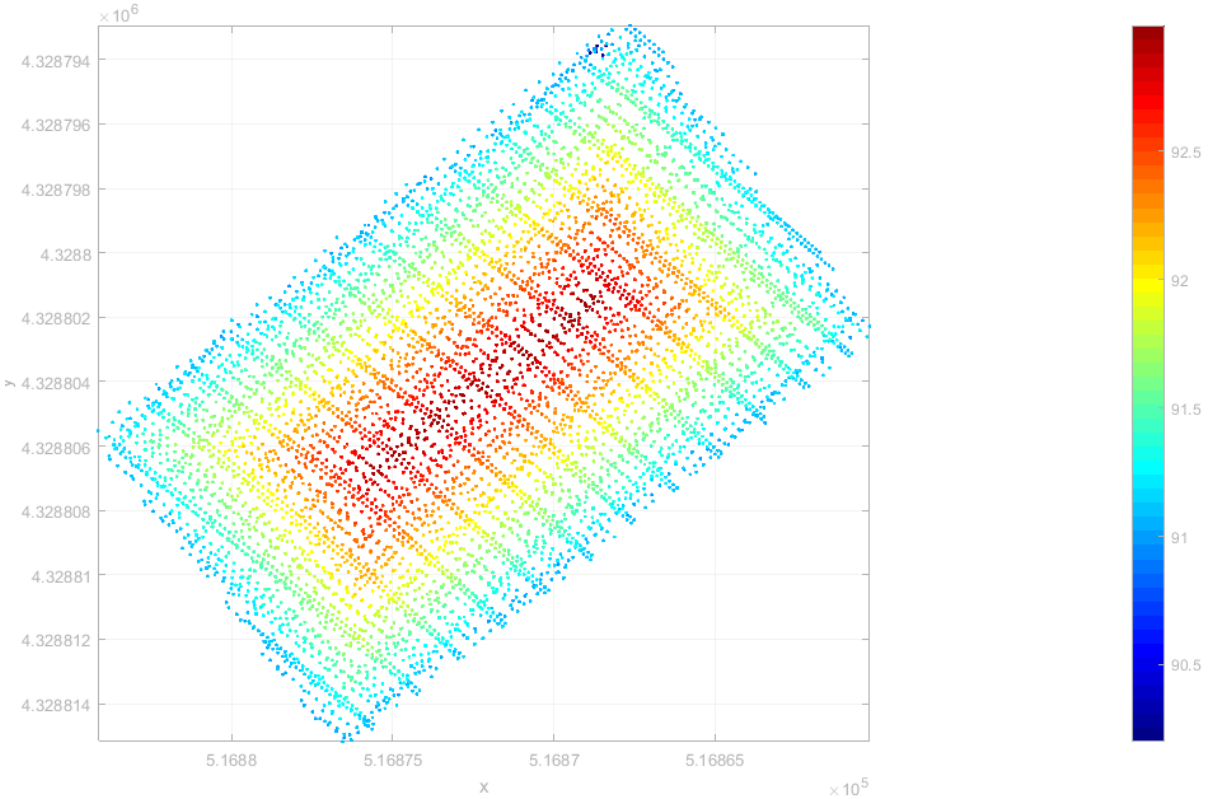
3.4. Bina Düzlemlerinin Çıkarılması

Çalışmanın bu aşamasında, tüm test alanları için ASCII metin formatına dönüştürülmüş olan bölütlenmiş bina nokta bulutu kümelerinin her birine Tim ZAMAN [41] tarafından yazılan 3B RANSAC algoritması uygulanmıştır. Sonuç olarak, bir binayı temsil eden düzlemlere ait düzlem noktalarının çıkarılması sağlanmıştır (Şekil 3.14., Şekil 3.15., Şekil 3.16., Şekil 3.17., Şekil 3.18. ve Şekil 3.19.). Bölütlenmiş her bir bina nokta bulutu kümesine ait düzlem noktaları, "*eşik değer*" parametresi mesafe değeri sınırları içerisinde bulunan tüm noktalar, bir düzlem belirtir yaklaşımıyla elde edilmiştir. Çıkarılan bir düzleme ait olan noktalar, algoritma tarafından bir sonraki düzlem çıkarımında kullanılmaması için LiDAR bina nokta bulutu kümesinden çıkarılmış ve düzlem olarak kayıt edilmiştir. Sonrasında kalan LiDAR bina kümesi nokta bulutuna sahip noktalar, RANSAC algoritması aynı şekilde uygulanarak diğer düzlemler elde

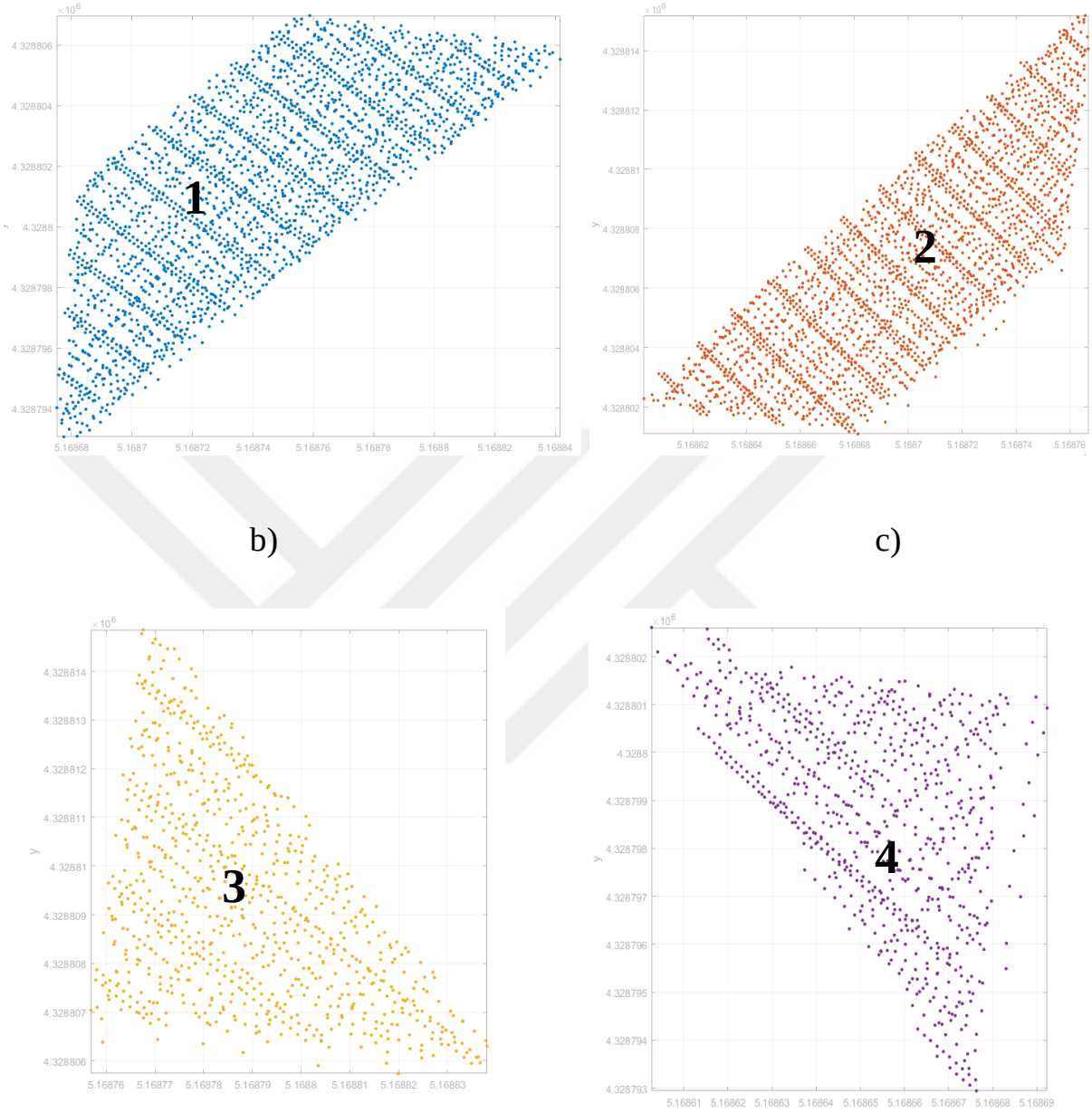
edilerek kayıt altına alınmıştır. Düzlem noktaları, bütün test alanlarındaki tüm bina nokta bulutlarına aynı parametre değerleri girilerek elde edilmiştir (Çizelge 3.4.). Bu değerlerden "*eşik değer*" parametresi deneme yanılma yöntemi ile tüm test alanlarında bulunan tüm verilere en uygun olacak şekilde, 0.10 metre olarak belirlenmiştir.

Çizelge 3.4. RANSAC algoritmasına ait parametreler ve değerleri.

Parametreler	Değer
Modele uyması için gereken en az veri noktası sayısı (no)	3
Algoritmada izin verilen en fazla yineleme sayısı (k)	168
Bir veri noktasının bir modele uyup uymadığını belirlemek için eşik değeri (t)	0.10 metre
Bir modelin verilere iyi uyduğunu ileri sürmek için gereken yakın veri noktalarının sayısı (d)	Veriye göre değişiklik göstermektedir.



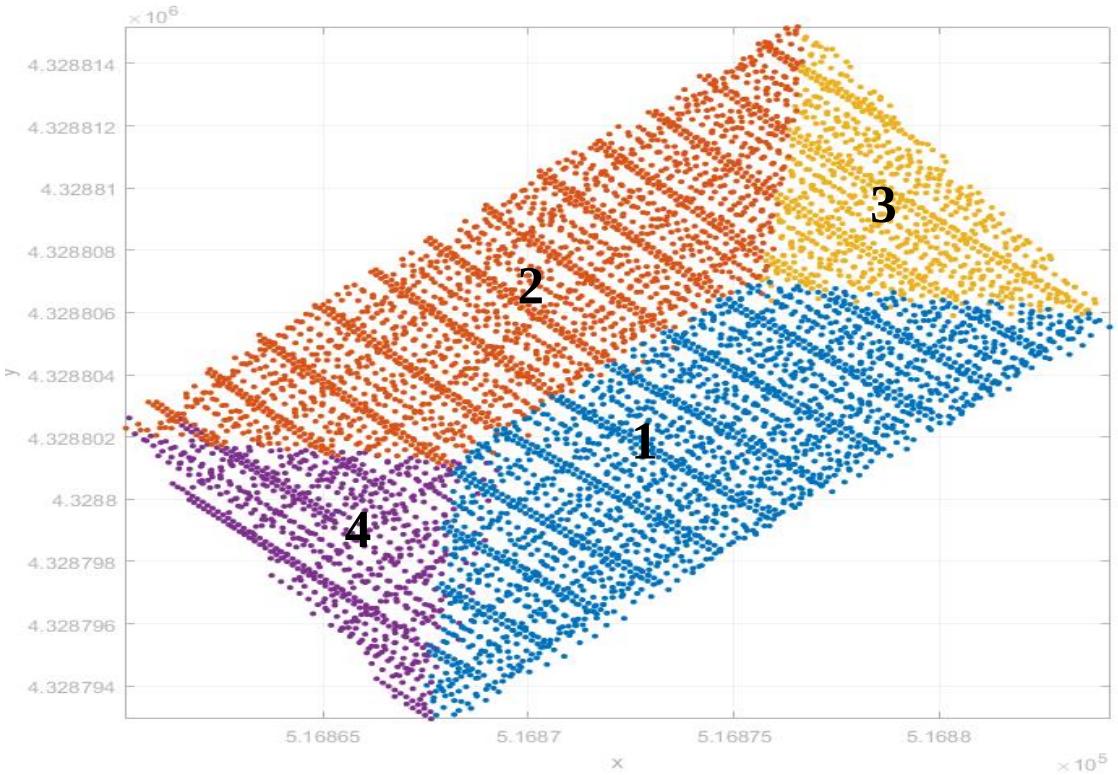
a)



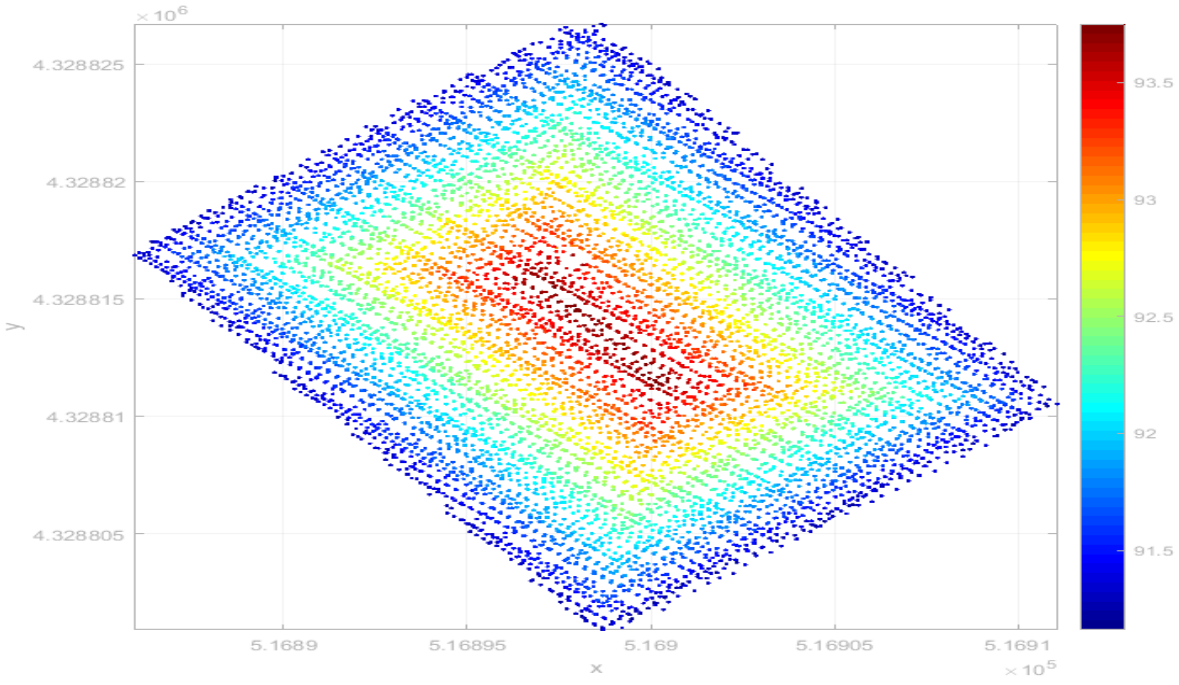
b) c)

d) e)

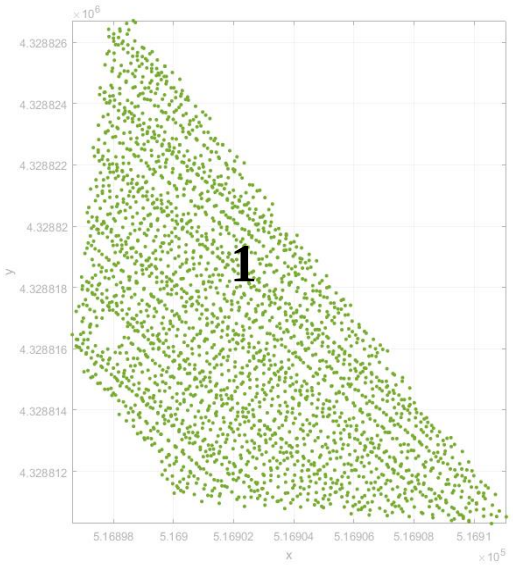
Şekil 3.14. Test alanı #3, birinci binaya ait RANSAC sonuçları a) orjinal bina nokta bulutu b) ilk düzlem noktaları c) ikinci düzlem noktaları d) üçüncü düzlem noktaları e) dördüncü düzlem noktaları.



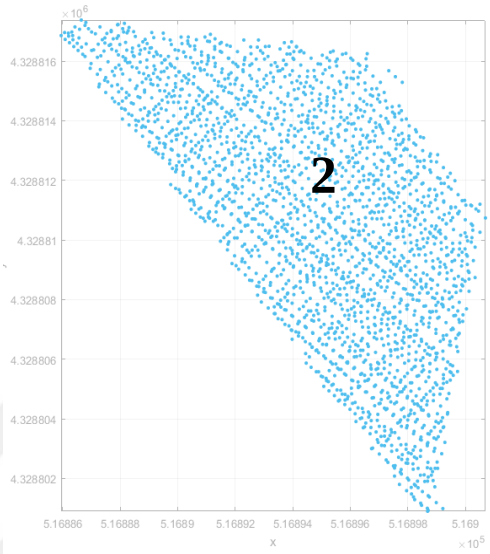
Şekil 3.15. Test alanı #3, birinci binaya ait; elde edilen dört düzleme ait noktaların bir arada gösterimi.



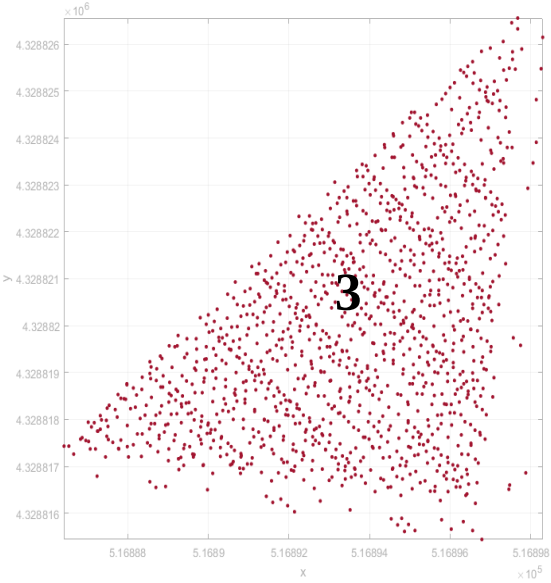
a)



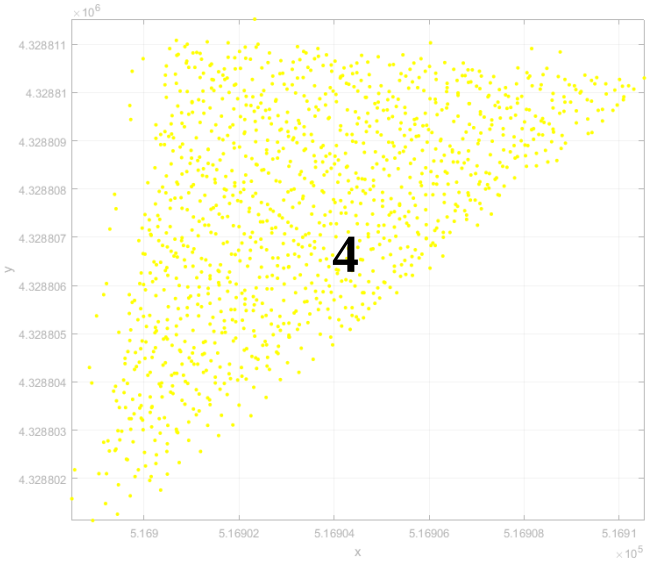
b)



c)



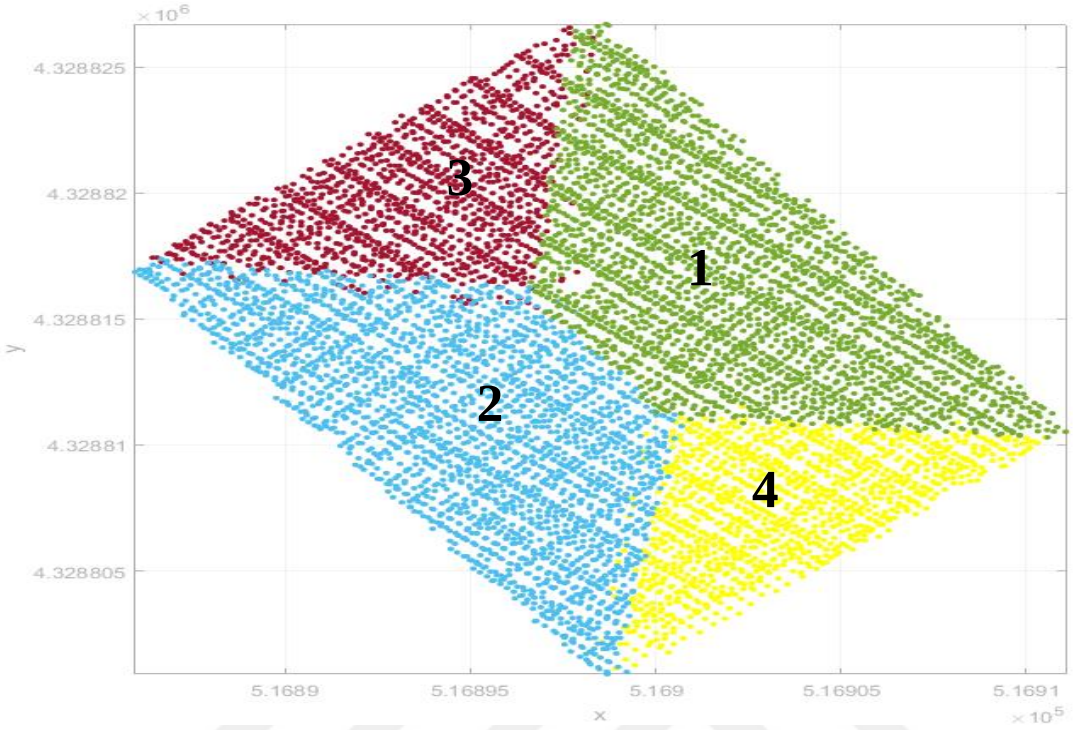
d)



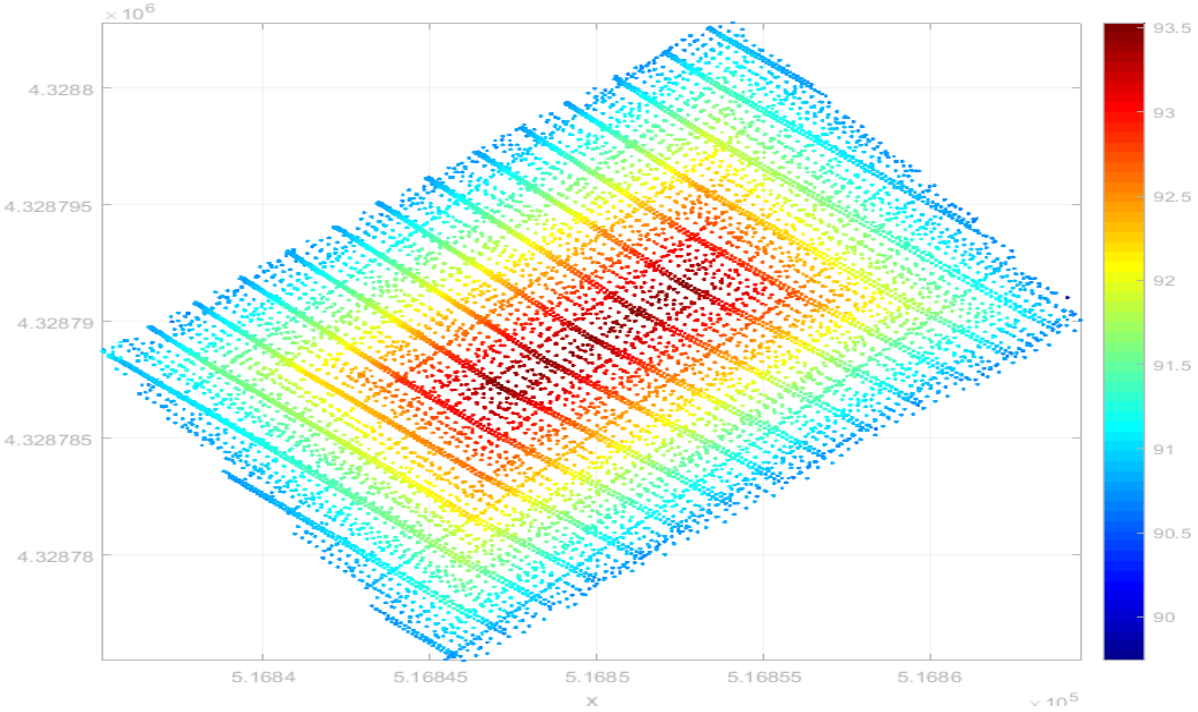
e)

Şekil 3.16. Test alanı #3, ikinci binaya ait RANSAC sonuçları a) orjinal bina nokta bulutu b) ilk düzlem noktaları c) ikinci düzlem noktaları d) üçüncü düzlem noktaları e) dördüncü düzlem noktaları.

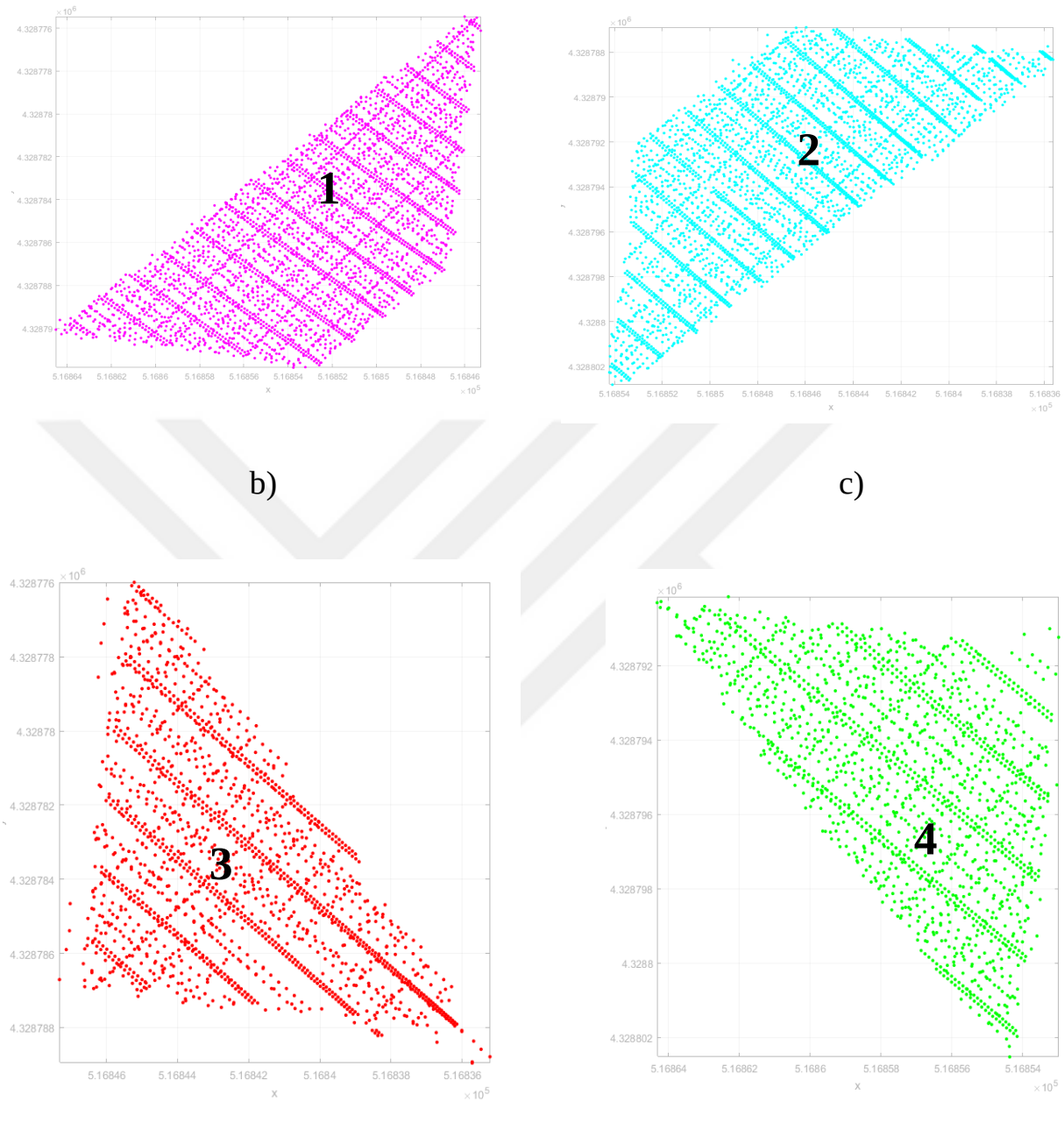
.



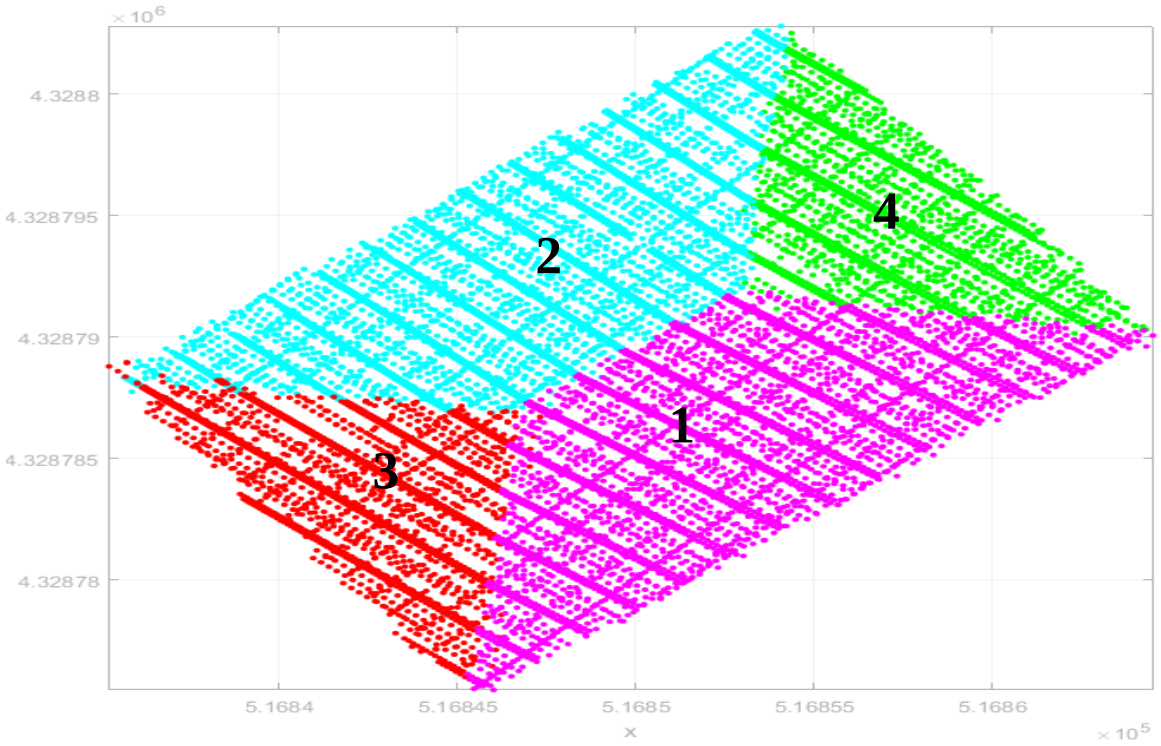
Şekil 3.17. Test alanı #3, ikinci binaya ait; elde edilen dört düzlem noktalarının; bir arada gösterimi.



a)



Şekil 3.18. Test alanı #3, üçüncü binaya ait; RANSAC sonuçları a) orjinal bina nokta bulutu b) ilk düzlem noktaları c) ikinci düzlem noktaları d) üçüncü düzlem noktaları e) dördüncü düzlem noktaları.



Şekil 3.19. Test alanı #3, üçüncü binaya ait; elde edilen dört düzlem noktalarının bir arada gösterimi.

3.5. Gürültünün Ayıklanması

Her bir LiDAR bina nokta bulutu kümesine RANSAC algoritmasının uygulanması ile düzlem çıkarımında görülmüştür ki, gözümüzün konumsal olarak diğer bir düzlemde olmasını beklediği noktalar, bazen beklenenin aksine elde edilen düzlemde bulunmaktadır. Çalışmanın bu aşamasında, gürültü olarak adlandırılan bu aykırı noktaların ayıklanması gerçekleştirilmiştir.

Gürültü noktalarının, elde edilen düzlemden kaldırılması için bölümlemeli kümeleme algoritması K-Means' den daha üstün küme oluşturma özelliğine sahip olan yoğunluk tabanlı kümele algoritmalarından DBSCAN algoritması, düzlemi temsil eden her bir nokta bulutuna uygulanmıştır. DBSCAN algoritması, nesnelerin komşuları ile olan mesafelerini hesaplayarak, belirli bir bölgede önceden belirlenmiş eşik değerden daha fazla nesne bulunan alanları gruplandırarak kümeleme işlemini gerçekleştirmektedir [42],[43]. Dolayısıyla, çalışmanın bu safhasında DBSCAN algoritması ile LiDAR düzlem noktalarının birbirleri ile olan komşulukları ortaya çıkarılmıştır.

Algoritmanın temel parametreleri şunlardır:

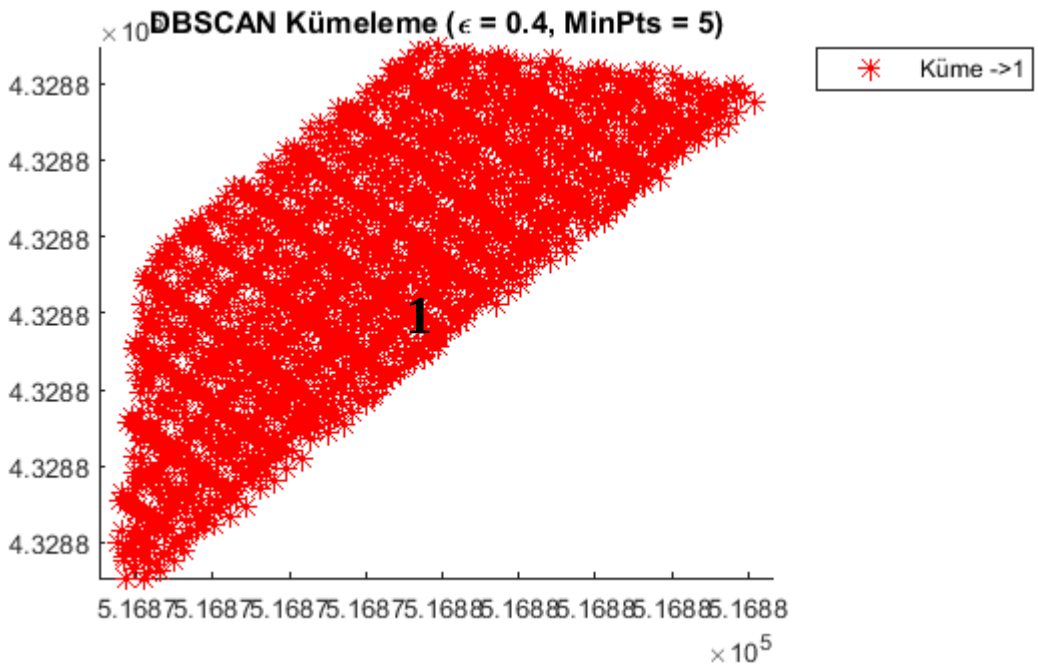
- "*Eps (ε)* " parametresi: Bir veri nesnesinin komşularını belirlemek için gerekli olan yakınlık mesafesidir.
- "*MinPts*" parametresi: Bir bölgenin yoğun olarak adlandırılabilmesi için Eps komşuluğunda bulunması gereken en az komşu sayısıdır.

Algoritma, LiDAR nokta bulutunda herhangi bir noktadan başlayarak tüm noktaları kontrol etmektedir. Önceden herhangi bir kümeye atanmamış olan herhangi bir nokta için bir bölge sorgusu yapılarak, nesnenin "*Eps*" parametresi komşuluğundaki komşu noktaları bulunmaktadır. Bulunan komşu sayısı "*MinPts*" parametresin'den fazla ise, bu nokta ve komşu noktalar yeni bir küme olarak adlandırılmaktadır. Sonrasında, önceden kümelenmemiş her bir komşu nokta için yeni bölge sorgusu yapılarak, yeni komşu noktalar bulunmaktadır. Bölge sorgusu yapılan noktaların komşu sayıları, "*MinPts*" parametresin'den fazla ise kümeye dahil edilir. Bu işlem, incelenen noktaların komşu sayısı "*MinPts*" parametresin'den az çıkana kadar sürmektedir.

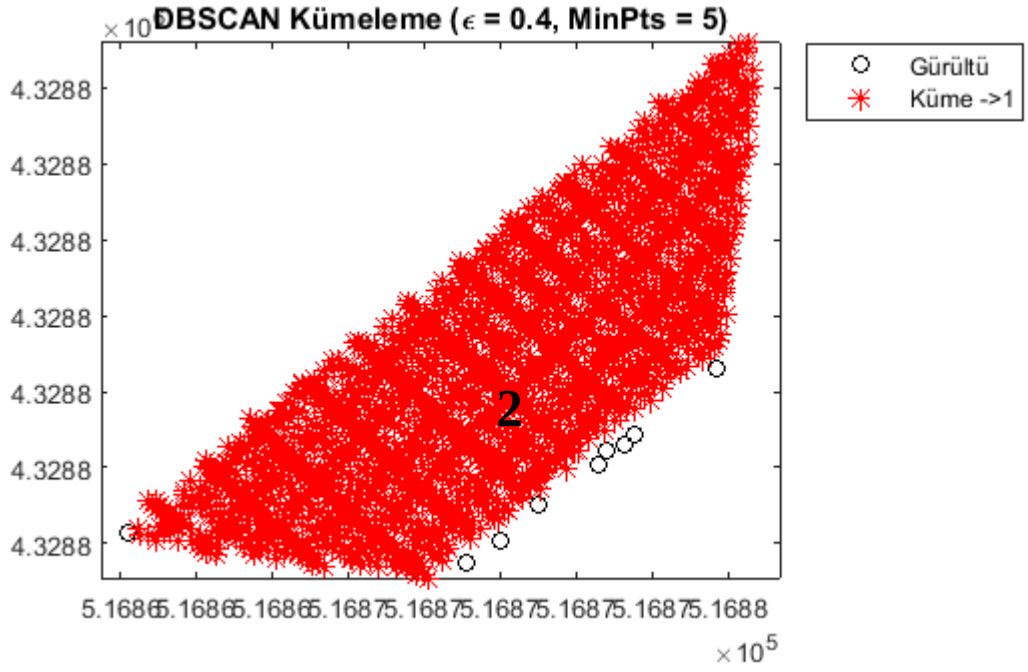
DBSCAN algoritması ile gürültüleri ayıklamak için yine denemeler yapılmıştır. Tüm test alanları için bu parametrelere deneme amaçlı verilen değerler "*MinPts*" parametresi için 3,5,8,15 ve "*Eps (ε)*" parametresi için 0.3'ten başlayarak 0.1 artışla 1.2'ye kadardır. Bu işlem sonucunda, tüm test alanları için en uygun parametre değerleri Çizelge 3.5'te görüldüğü gibi belirlemiştir. Test alanı #3'e yapılan işlem sonucunda düzlemlere ait gürültü noktaları Şekil 3.20., Şekil 3.21. ve Şekil 3.22'te gösterilmektedir. Gürültüden ayıklanmış düzlemler bir arada Şekil 3.23'te gösterilmektedir.

Çizelge 3.5. Test alanlarına uygulanan DBSCAN algoritması parametreleri.

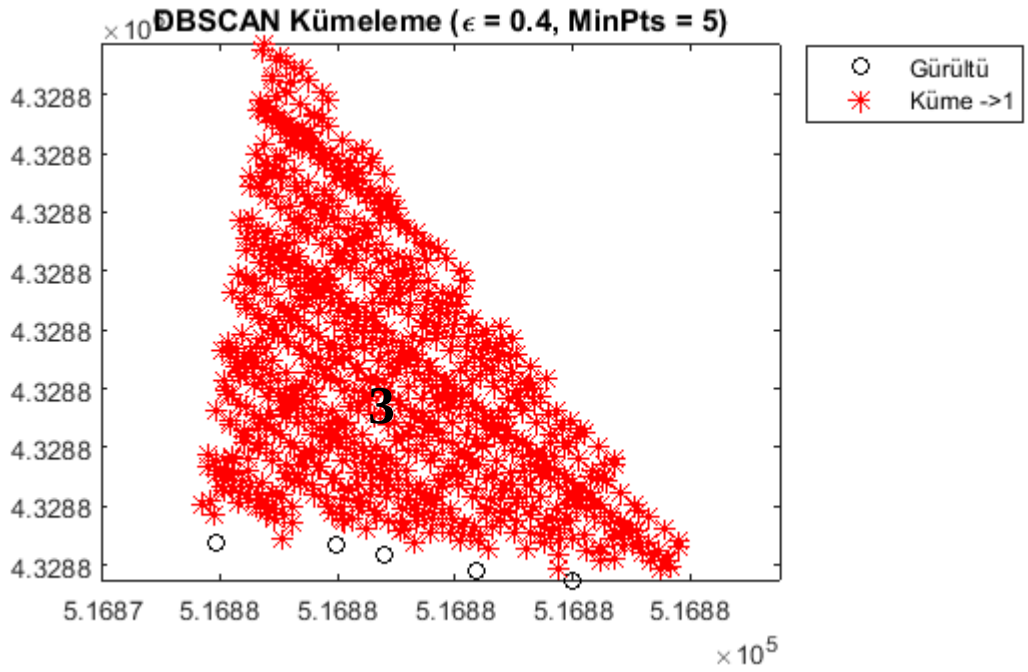
Test Alanı #1		Test Alanı #2		Test Alanı #3	
MinPts	ε	MinPts	ε	MinPts	ε
5	1.1	15	0.7	5	0.4


DBSCAN Kümeleme (ϵ = 0.4, MinPts = 5)
Küme ->1
1
×10^5

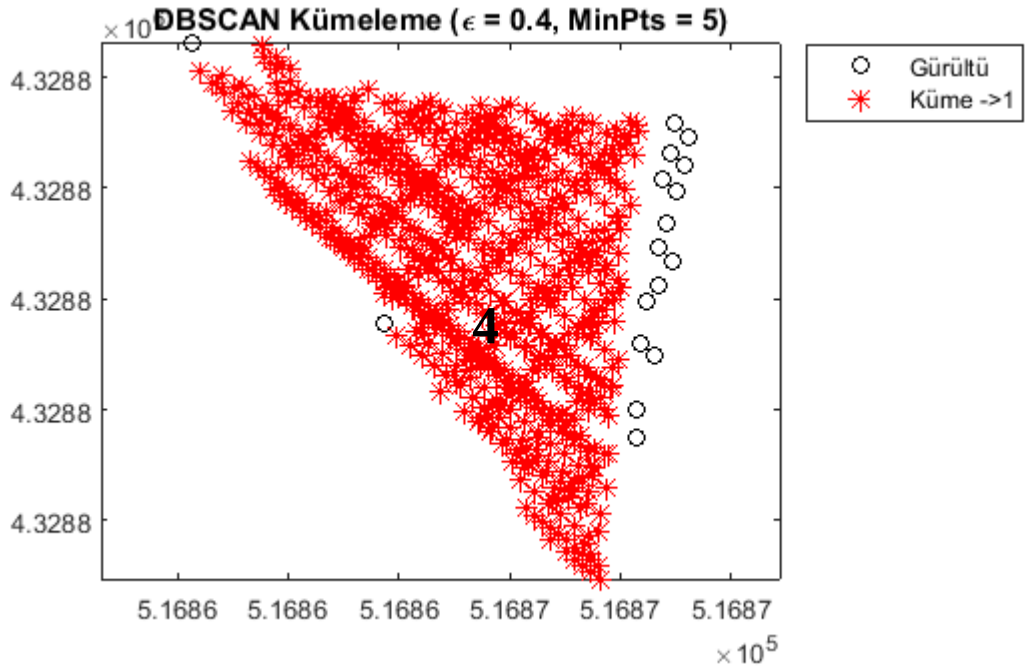
a)


DBSCAN Kümeleme (ϵ = 0.4, MinPts = 5)
Gürültü
Küme ->1
2
×10^5

b)

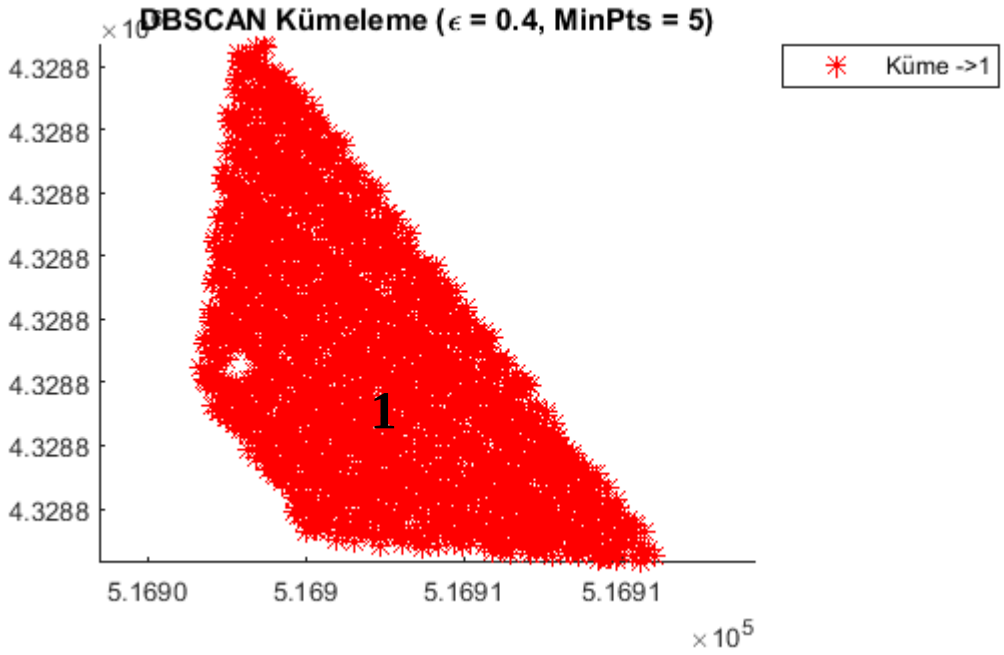


c)

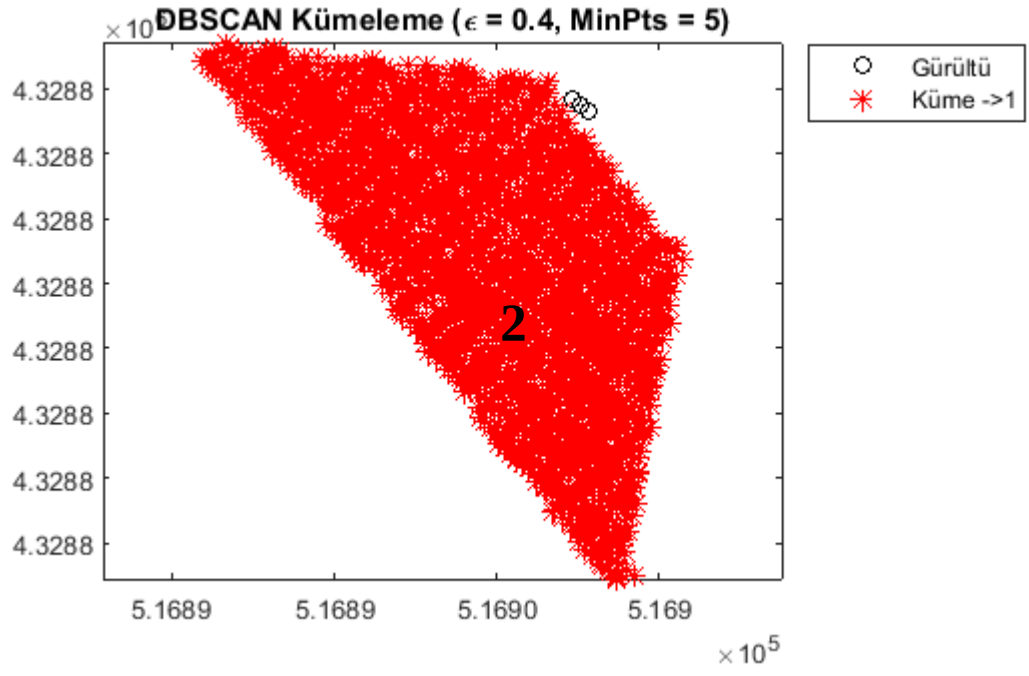


d)

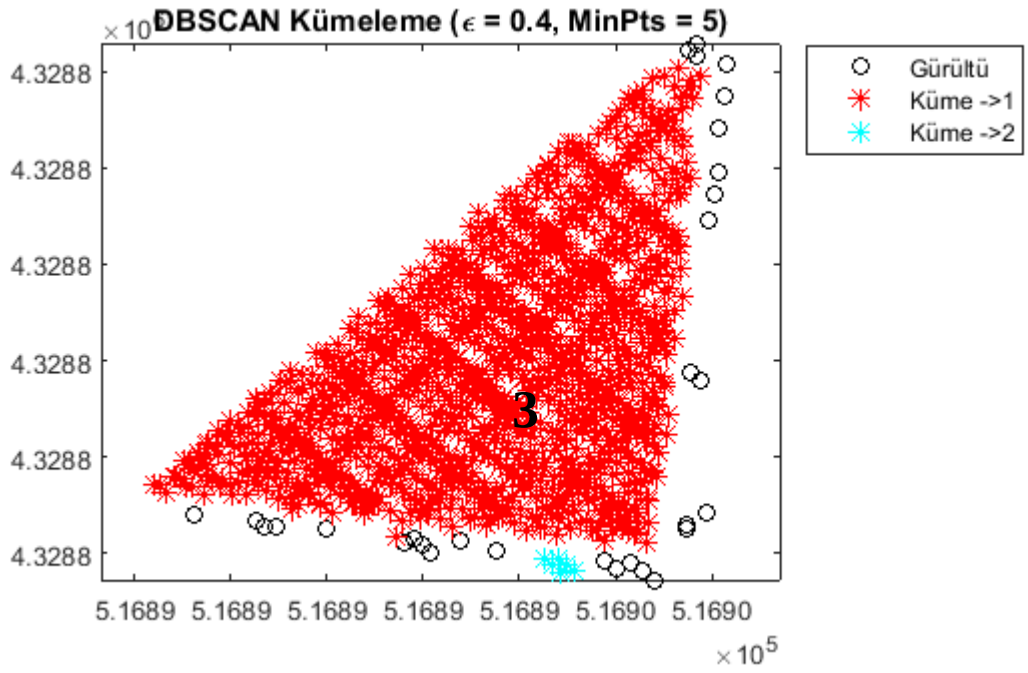
Şekil 3.20. Test alanı #3, birinci bina; a) birinci b) ikinci c) üçüncü d) dördüncü düzlem'e ait gürültü noktalarının gösterimi.


DBSCAN Kümeleme (ϵ = 0.4, MinPts = 5)
Küme ->1
1
4.3288
4.3288
4.3288
4.3288
4.3288
4.3288
4.3288
5.1690
5.169
5.1691
5.1691
×10^5

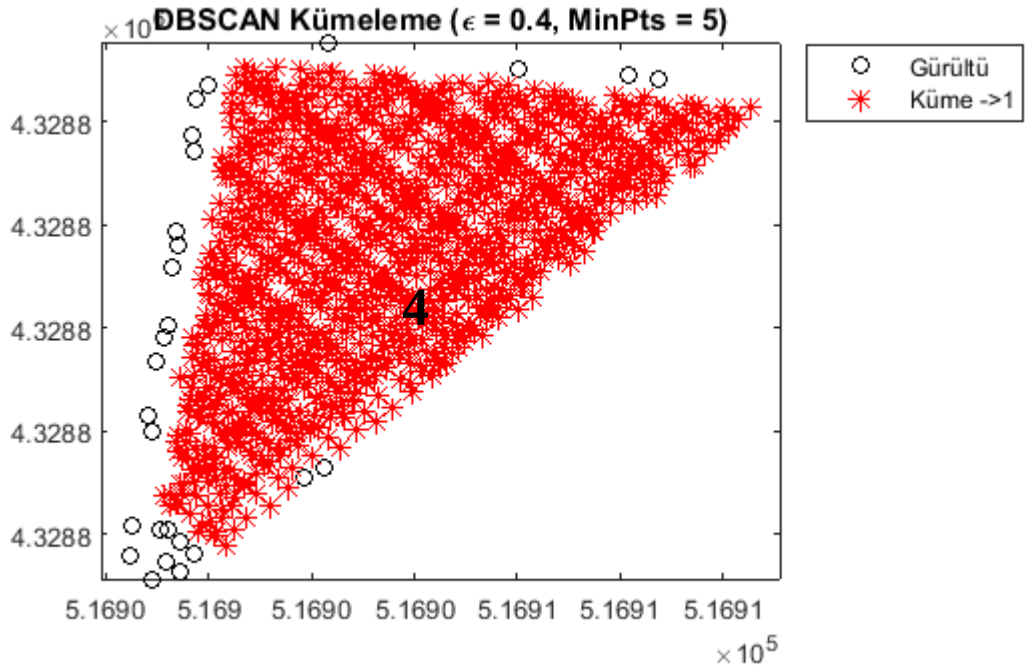
a)


DBSCAN Kümeleme (ϵ = 0.4, MinPts = 5)
Gürültü
Küme ->1
2
4.3288
4.3288
4.3288
4.3288
4.3288
4.3288
4.3288
5.1689
5.1689
5.1690
5.169
×10^5

b)

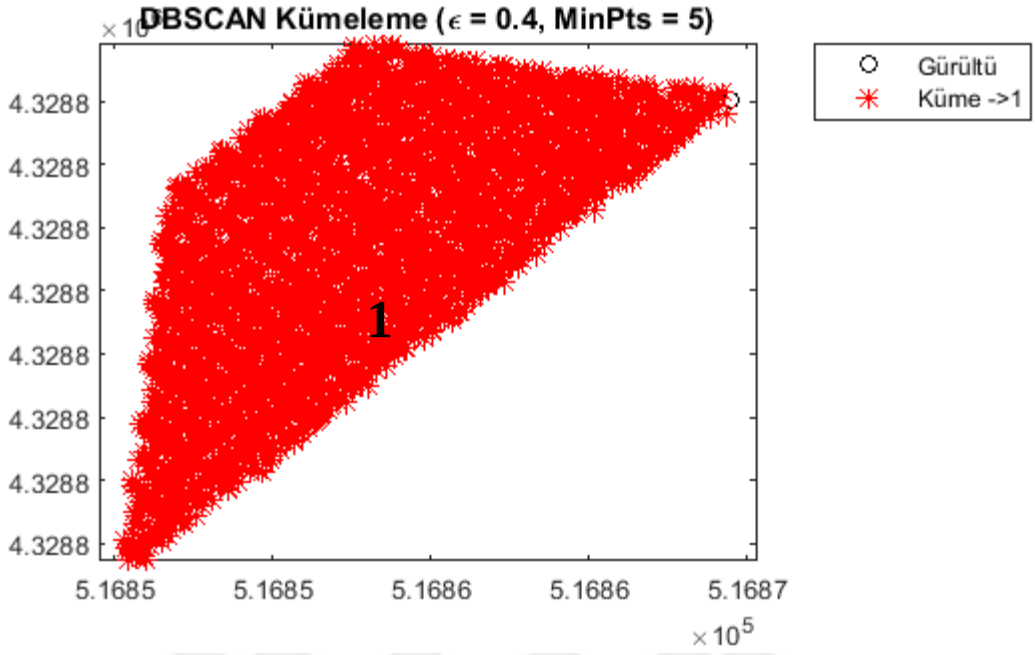


c)

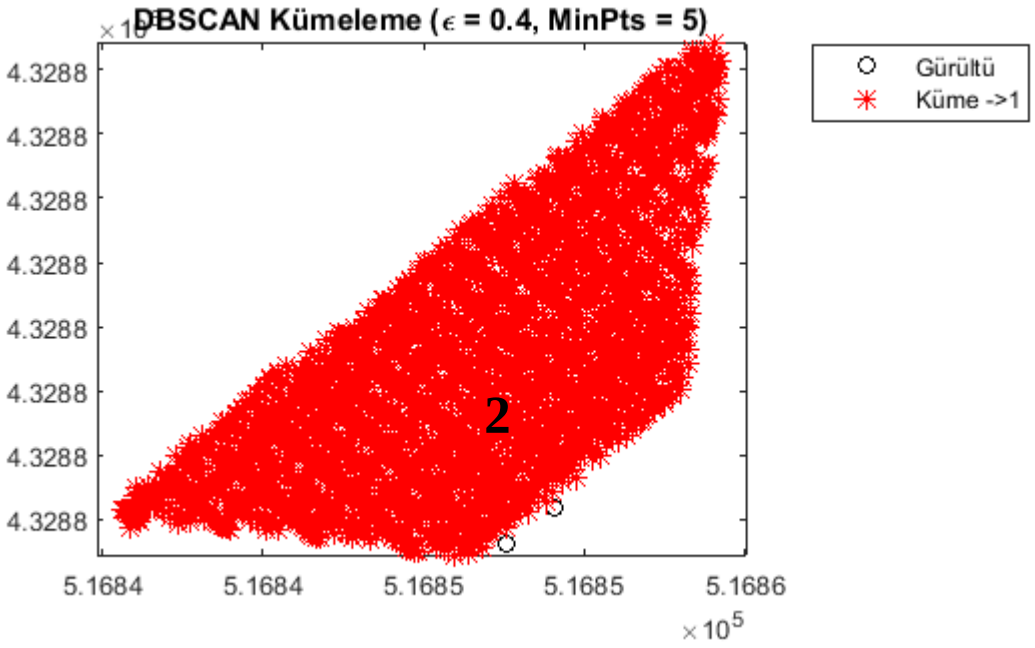


d)

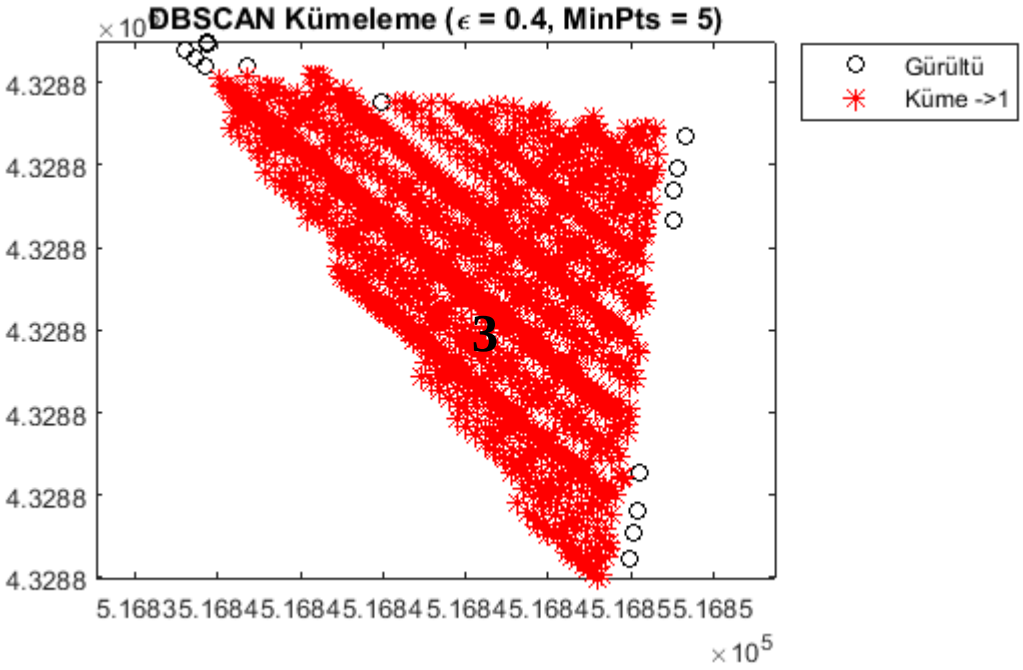
Şekil 3.21. Test alanı #3, ikinci bina; a) birinci b) ikinci c) üçüncü d) dördüncü düzlem'e ait gürültü noktalarının gösterimi.


DBSCAN Kümeleme (ϵ = 0.4, MinPts = 5)
Gürültü
Küme ->1
1
4.3288
5.1685
5.1685
5.1686
5.1686
5.1687
×10^5

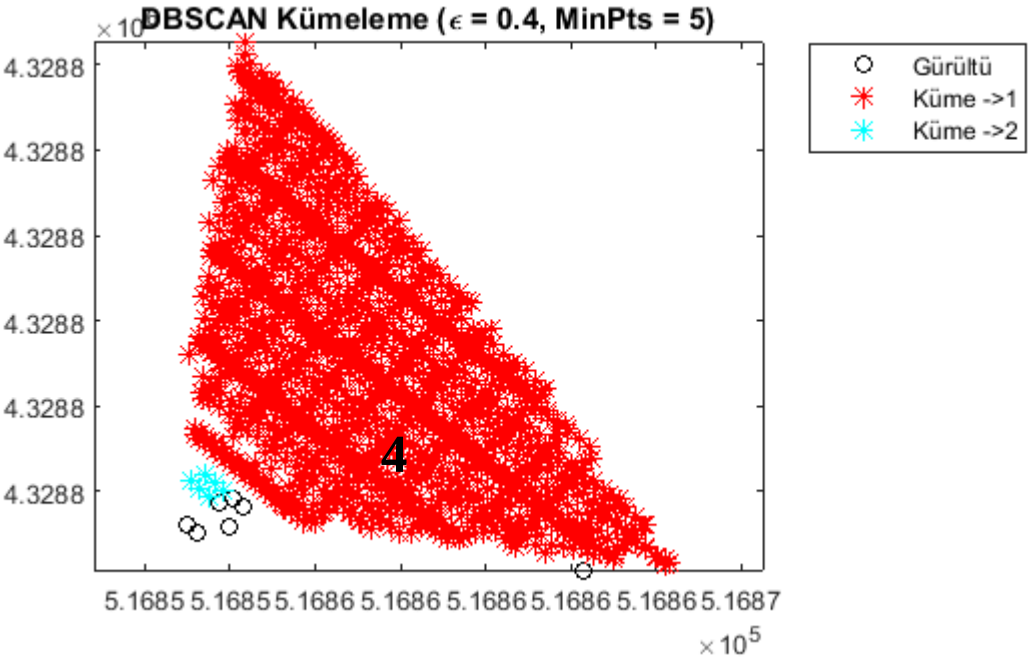
a)


DBSCAN Kümeleme (ϵ = 0.4, MinPts = 5)
Gürültü
Küme ->1
2
4.3288
5.1684
5.1684
5.1685
5.1685
5.1686
×10^5

b)

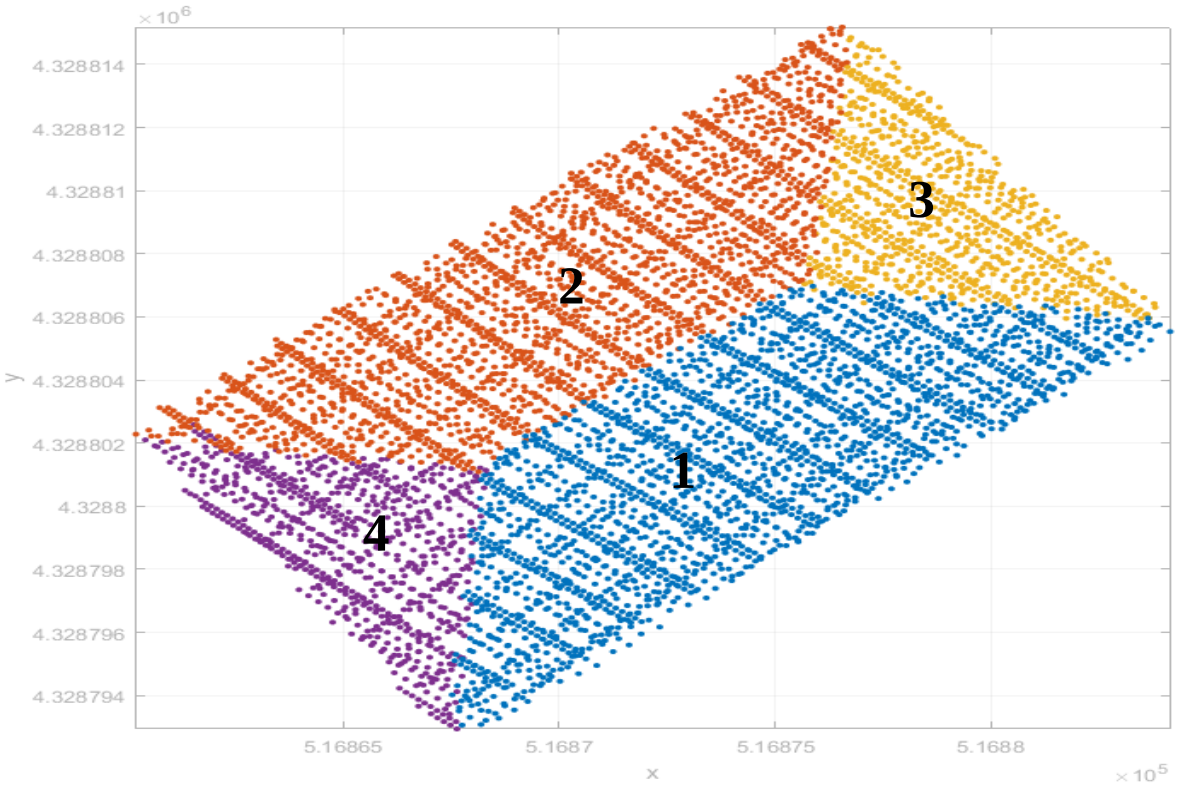


c)

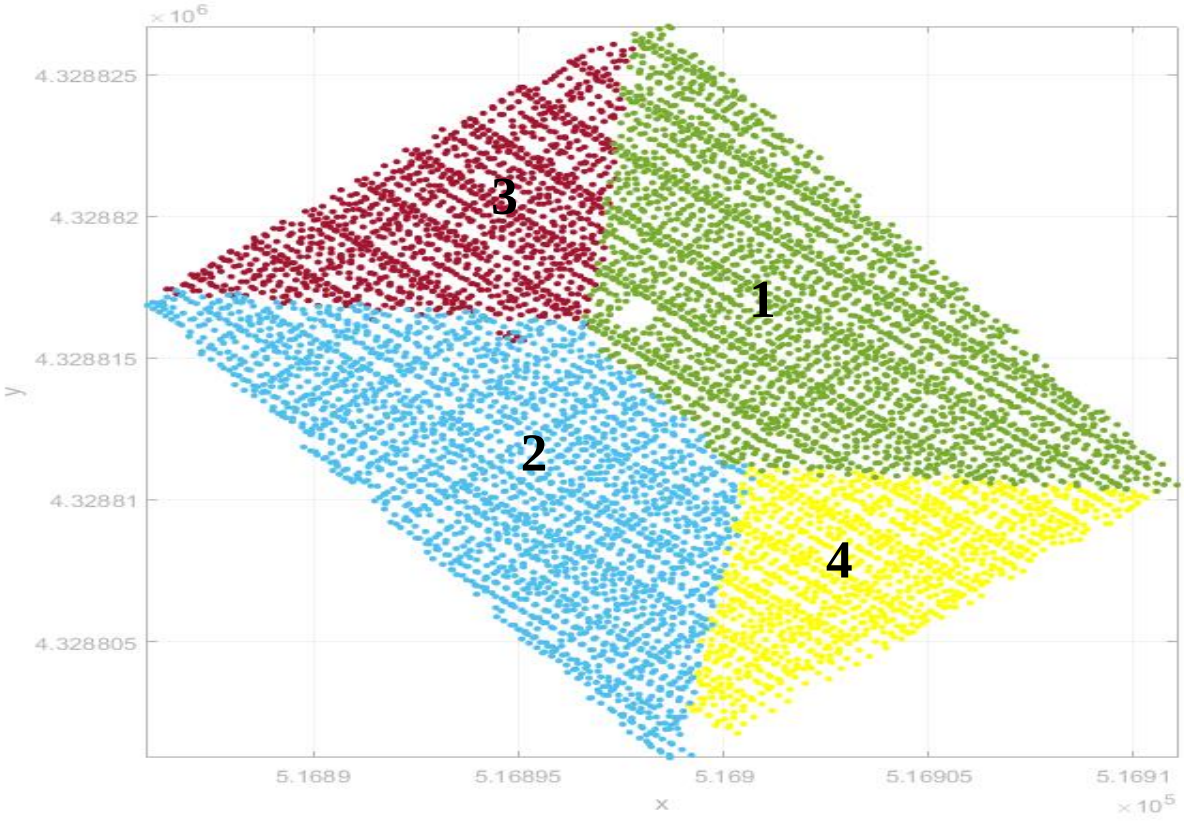


d)

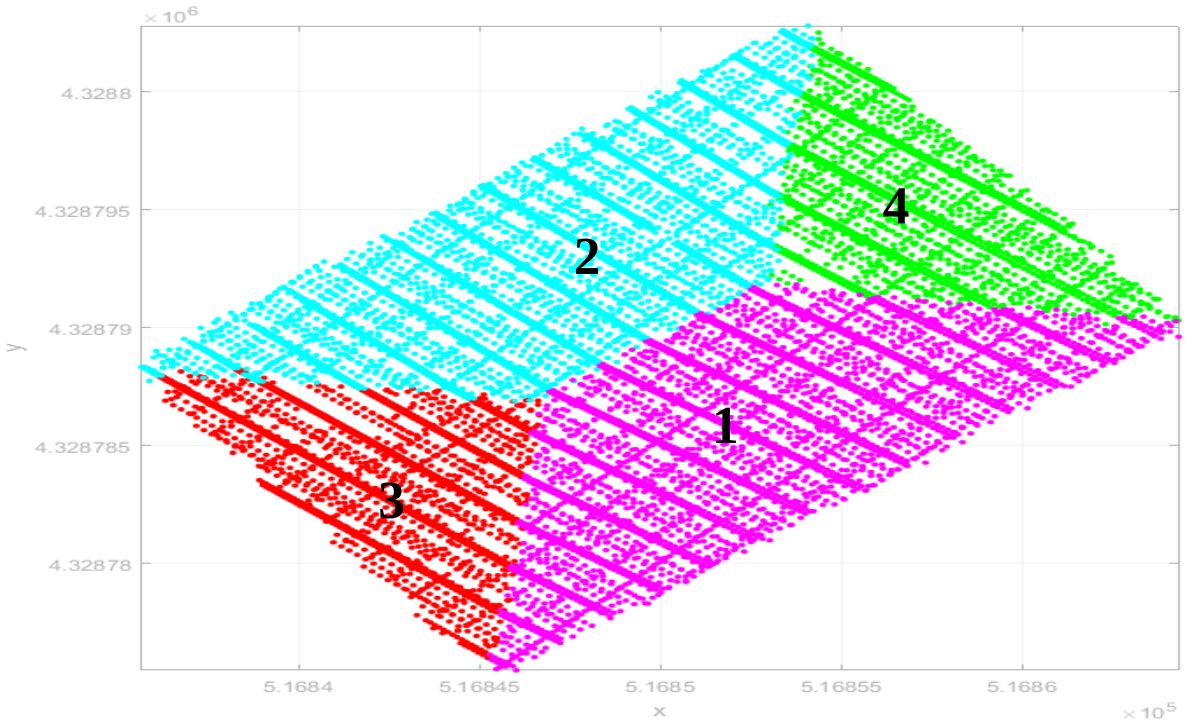
Şekil 3.22. Test alanı #3, üçüncü bina; a) birinci b) ikinci c) üçüncü d) dördüncü düzlem'e ait gürültü noktalarının gösterimi.


×10^6
4.328814
4.328812
4.32881
4.328808
4.328806
4.328804
4.328802
4.3288
4.328798
4.328796
4.328794
y
5.16865
5.1687
5.16875
5.1688
×10^5
x
1
2
3
4

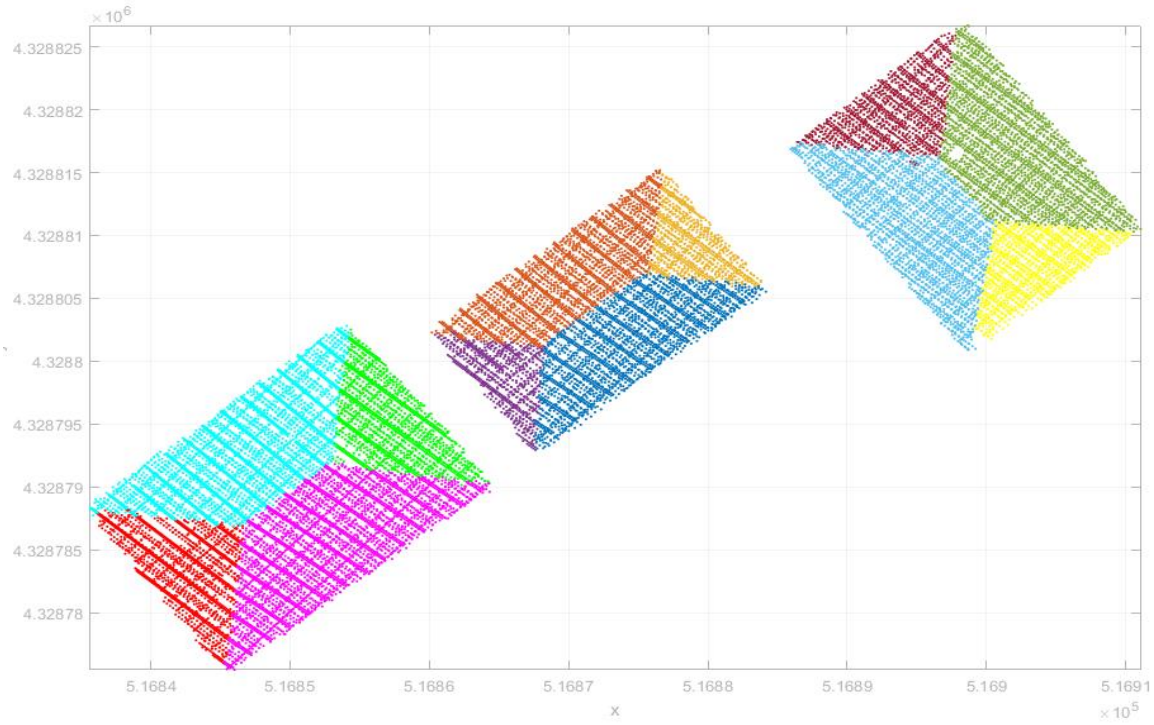
a)


×10^6
4.328825
4.32882
4.328815
4.32881
4.328805
y
5.1689
5.16895
5.169
5.16905
5.1691
×10^5
x
1
2
3
4

b)



c)

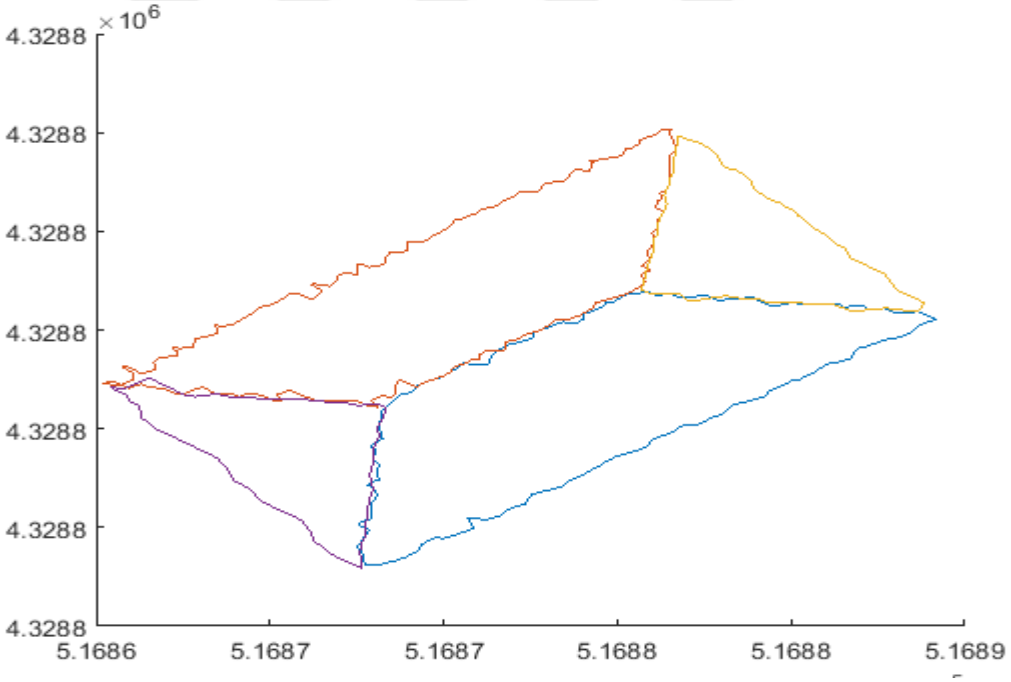


d)

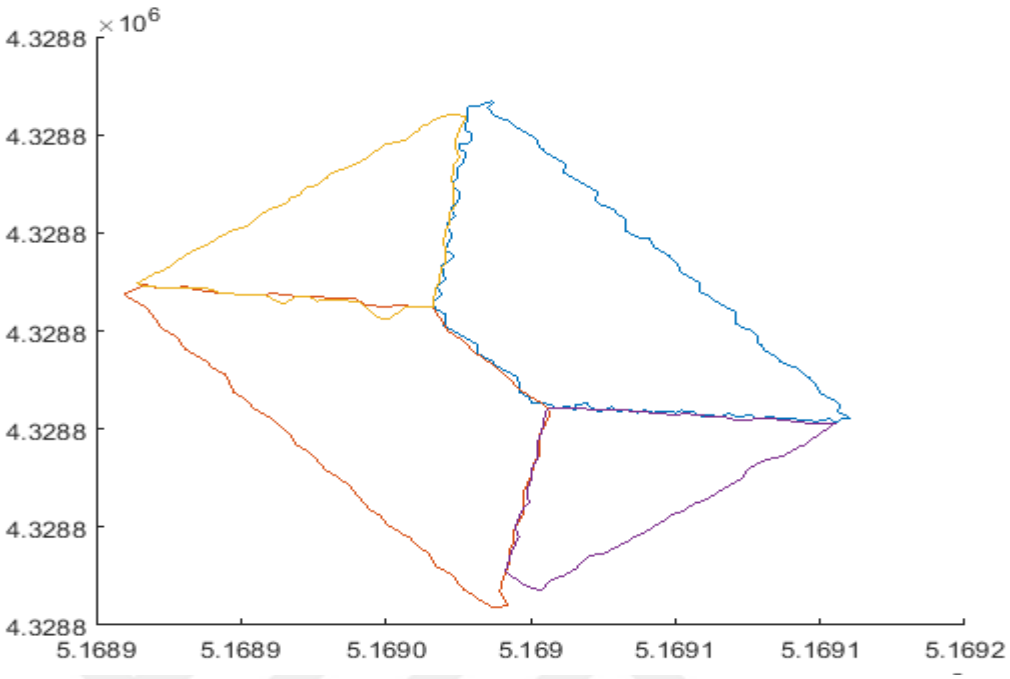
Şekil 3.23. Test alanı #3'e ait; a) birinci b) ikinici c) üçüncü ve d) tüm binaların, gürültüden ayıklanmış düzlemlerinin bir arada gösterimi.

3.6. Çatı Düzlemi Sınırlarının Çıkarılması

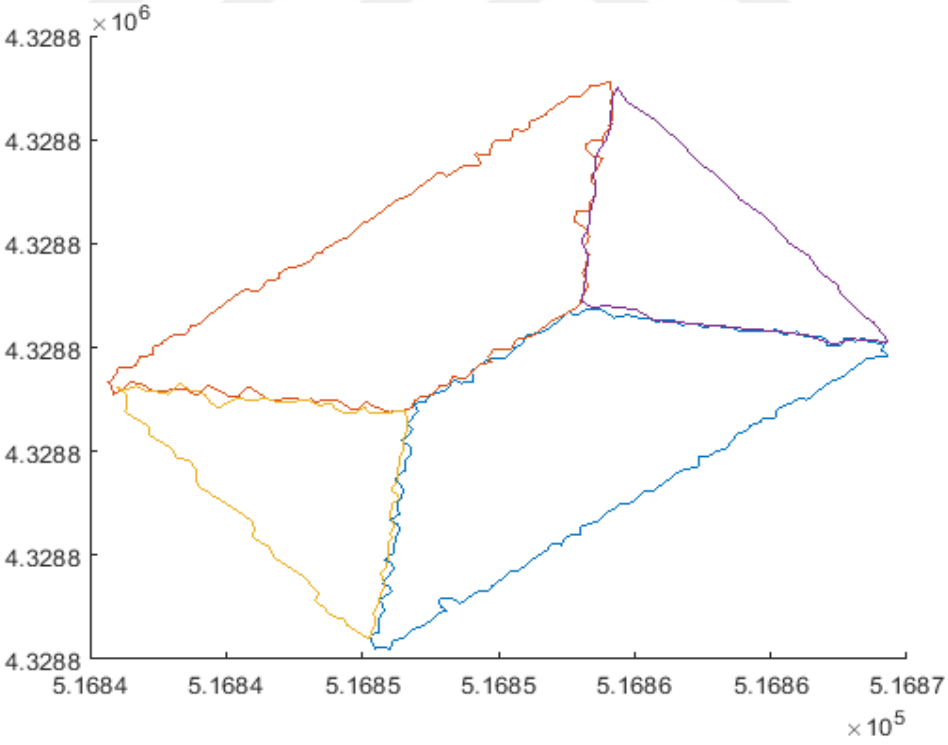
Gürültüden arındırılmış olan LiDAR düzlem nokta bulutlarına MATLAB ortamında "boundary" fonksiyonu uygulanmıştır. Bu fonksiyon, noktalar etrafında iki boyutlu sınırı temsil eden nokta indeksleri vektörünü çalıştırmaktadır. "boundary" foksiyonu ile dış bükey örtünün (convex hull) aksine, sınır noktalarını kaplamak için gövdenin iç tarafına doğru daralma olabilmektedir. Fonksiyonun girdisi noktanın sütun vektör olarak belirtilen x ve y koordinatları, çıktısı ise iki boyutlu sınırı temsil etmek için bir poligon sınırı etrafındaki noktaların dizilimini temsil eden nokta indekslerinin sütun vektörüdür. Fonksiyonda varsayılan olarak "daraltma faktörü" değeri 0.5 tir. Bu faktör kullanılarak noktalar etrafındaki sınır hesaplanmaktadır. "daraltma faktörü" değeri küçültüldüğünde noktaları kaplayan daha az sıkı bir sınır oluşmaktadır [44]. Bu işlemin sonuçları Şekil 3.24'te gösterilmiştir.



a)



b)



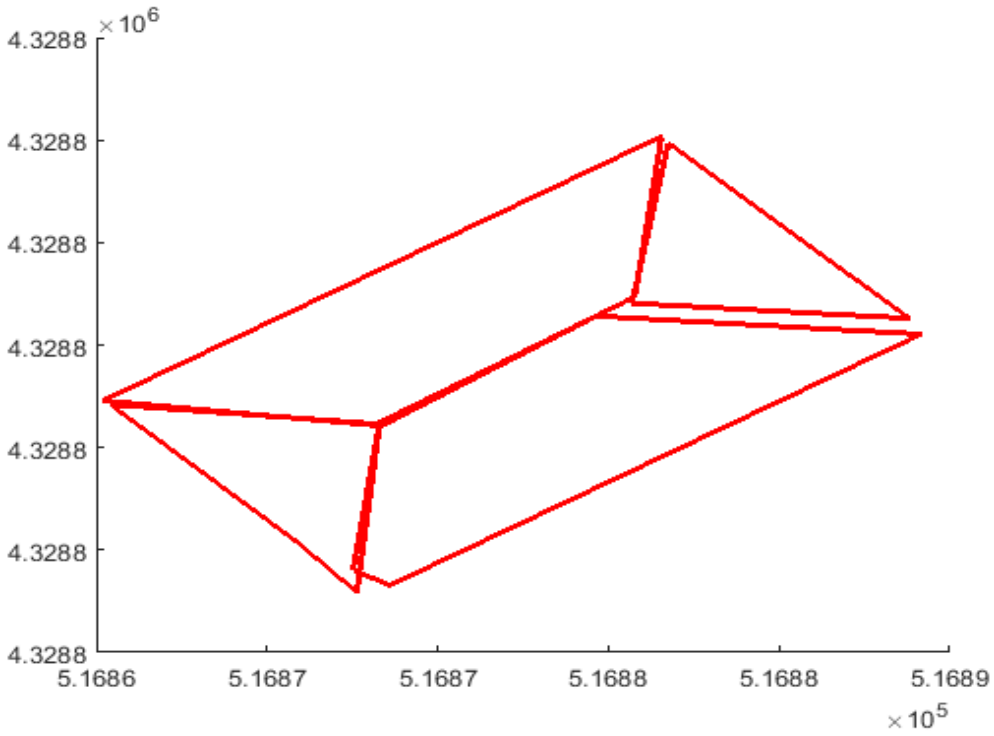
c)

Şekil 3.24. Test alanı #3'e ait; a) birinci b) ikinci ve c) üçüncü bina düzlemlerine sınır fonksiyonunun uygulanması ve bu düzlemlerin bir arada gösterilmesi sonrası oluşan sonuç görüntü.

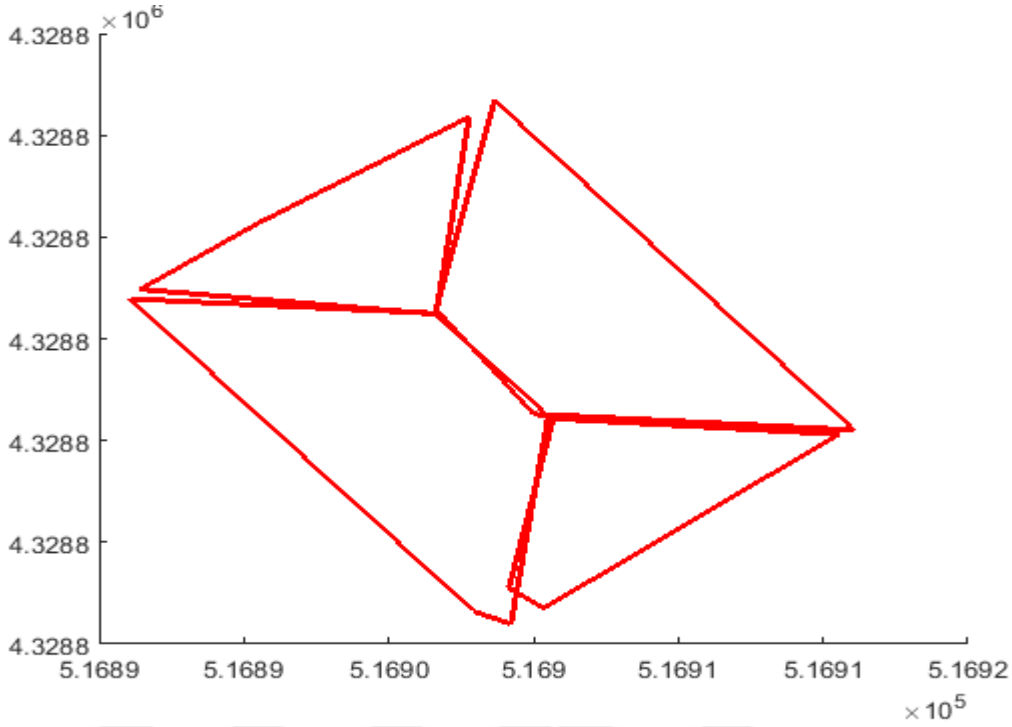
3.7.Çatı Düzlemi Sınırlarının Sadeleştirilmesi

Önceki aşamada sınırların çıkarılmasında “boundary” fonksiyonu ile elde edilen sınırı temsil eden noktalara, Douglas-Peucker algoritması uygulanmıştır.

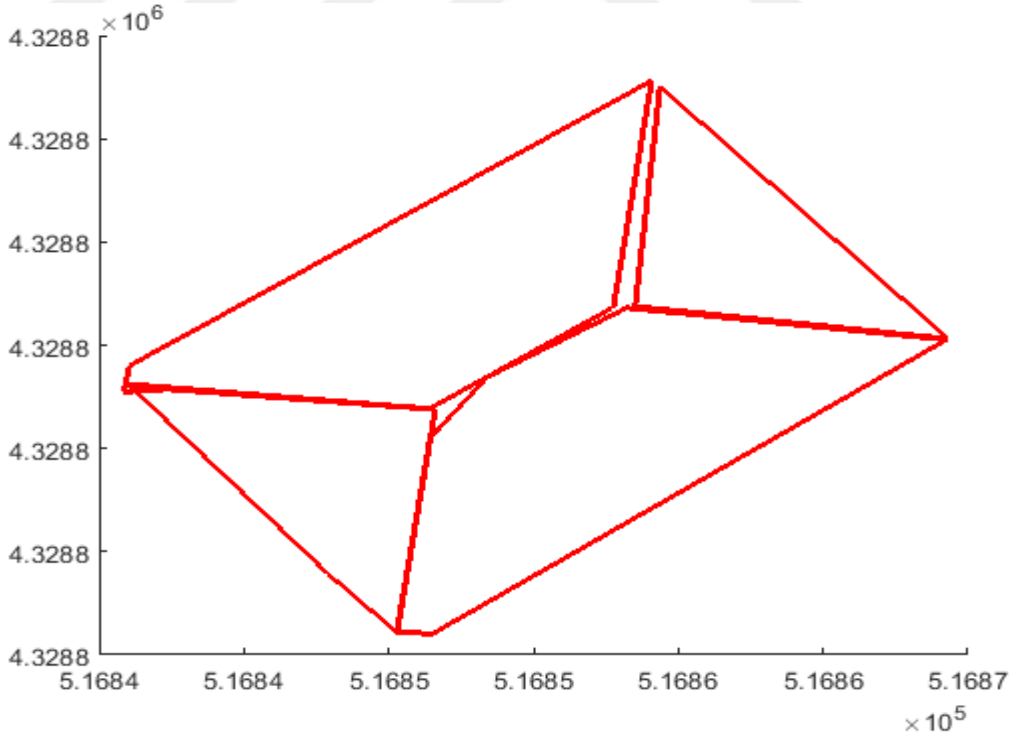
Bu algoritmada, tekrarlı bir şekilde bir poligondaki köşe sayısını azaltmak için, ilk ve son noktadan bir çizgi geçirilerek, tüm köşe noktalarından bu çizgiye olan dik mesafeler hesaplanmaktadır. Bu mesafeler belirtilen bir tolerans değerini geçmediğinde bu düz çizginin tüm satırı, basitleştirilmiş formda temsil edilmektedir. Bunun aksi durumda, düz çizgiden en uzun mesafede olan nokta, yeni nokta kabul edilerek yukarıdaki işlem, tolerans sınırı karşılanana ve sadeleştirme süreci tamamlanana kadar çizgi sadeleştirme işlemini devam ettirmektedir [45]. Yapılan işlemin sonuçları Şekil 3.25’da görülmektedir.



a)



b)



c)

Şekil 3.25. Test alanı #3'e ait; a) birinci b) ikinci ve c) üçüncü bina düzlemlerine Douglas-Peucker algoritmasının uygulanması sonrası oluşan sonuç görüntü.

4.BULGULAR

4.1. Çatı Düzlemlerinin Çıkarılması

Bir binayı temsil eden çatı düzlemlerine ait düzlem noktalarını otomatik tespit etmek için her bir binaya ait nokta bulutu kümesine RANSAC algoritması uygulanmıştır. Çalışmanın bu kısmında RANSAC algoritması uygulanarak elde edilen LiDAR düzlem noktalarının sonuçları verilecektir. RANSAC algoritmasına ait parametre değerleri Çizelge 3.4'te (Bölüm 3) gösterilmiştir. Olası her örneği denemek çoğu zaman mümkün olmamaktadır. Bu yüzden bu çalışmada, RANSAC algoritmasının parametrelerinden olan "*algoritma tarafından izin verilen en fazla yineleme sayısı (iterasyon sayısı: k)*" parametresi değeri, Eşitlik 4.1'e göre hesaplanmıştır.

$$N = \frac{log(1-p)}{log(1-w^m)} = \frac{log(1-0.99)}{log(1-0.3^3)} \approx 168 \qquad (4.1)$$

Eşitlik 4.1'de *m* düzlem modelini tanımlamak için gerekli minimum nokta sayısını, *w* modele uygun olan bir noktanın seçilme olasılığını, *p* başarılı bir işlemin çalışma olasılığını temsil etmektedir [46]. *p* 0,99'a sabit tutularak [47], w değeri deneme yanılma yöntemi ile bulunmuştur. Yine bu çalışmada, "*en fazla nokta sayısı (d)*" parametre değeri, tüm test alanlarında bulunan binaların her bir düzleme ait noktalarının çıkarımında minimum olacak şekilde belirlenmiştir. Bundan dolayı, bu parametre değeri her bir binanın her düzlemi için farklılık göstermektedir. Bu parametre değeri ile düzlemlere ait noktalar tespit edildiğinde, görsel inceleme neticesinde çatı düzlemine yakın sonuçlar veren düzlem noktaları kabul edilmiş ve düzlemsellik özelliği olmayan noktalar ise düzlem dışı noktalar olarak çalışmada göz ardı edilmiştir. Özetle, sistem temel büyük çatı düzlemlerinin tamamını doğru olarak tespit etmiştir.

Düzlem noktalarının tespitinde kullanılan "*mesafe eşik değeri*" parametresi değeri olarak 10 cm alınmıştır. Bu değer, Şekil 3.12.'de görüldüğü gibi, bu çalışmada bina çatılarından yukarıda bulunan bacalar vb. noktalar ile bina çatılarından aşağıda bulunan duvarlara ait noktaların, düzlem noktalarına dahil edilmemesi için, en uygun değer olarak belirlenmiştir.

Bir düzlem sınırsız büyüklükte olup uzayda sonu yoktur. RANSAC algoritması ile düzlem noktaları belirlenirken, başlangıçta elde edilen düzleme dahil olmaması beklenen noktalar, düzlemin sınırsız büyüklükte olmasından dolayı, elde edilen düzlemde olabilmektedir. Bu

noktalar gürültü noktaları olarak tanımlanmaktadır. Düzlem noktalarının doğru bir şekilde tespit edilememesi, diğer adımlardaki çalışma sonuçlarını da olumsuz yönde etkileyebilmektedir.

Gürültü noktalarının oluşmasına sebep olan bir diğer unsur, "*mesafe eşik değer*" parametresinin, tüm bina çatı düzlem noktalarının elde edilmesinde tek bir değer olarak alınmasıdır.

RANSAC algoritmasının sonuçları birinci test alanındaki 26 bina için incelendiğinde #1, #2 (yeşil renk düzlem noktaları), #8, ve #11 nolu binalara ait çatı düzlemlerin çok doğru çıkarıldığı görülmektedir (Şekil 4.1.). Ancak, #3 nolu bina incelendiğinde (Şekil 4.2.), bu binaya ait iki adet düzlemin çıkarılması beklenirken, RANSAC algoritması tarafından bir düzlem elde edilmiştir. Bu binada eksik düzlem elde edilmesinin sebebi bu test alanında nokta yoğunluğunun, ikinci test alanına göre, daha düşük olmasıdır. Diğer taraftan, #15 nolu bina incelendiğinde, "*en fazla nokta sayısı (d)*" parametre değeri, RANSAC algoritmasının düzlemsellik özelliği gösteren noktalara göre belirlendiği için, bu binada bir çatı düzleminin çıkarılması beklenirken iki adet çatı düzleminin (mavi ve sarı renk) çıkarıldığı görülmüştür (Şekil 4.3.). Birinci test alanında bulunan diğer binalar #2, #4-7, #9, #10, #12-15 ve #16-26 incelendiklerinde her bina için çatı düzlemlerinin tespit edildiği ancak çıkarılan binanın diğer düzlemlerinde olması beklenmeyen ve gürültü olarak adlandırılan noktaların elde edilen düzlemde bulunduğu görülmüştür. Bu durum #17 nolu bina için Şekil 4.4'te ve diğer binalar için ise Ekler kısmında gösterilmektedir.

RANSAC algoritmasının sonuçları ikinci test alanındaki 26 bina için incelendiğinde ise, #1, #2 (yeşil renk düzlem noktaları) ve #3 nolu binalarda düzlemlerinin iyi sonuç verdiği (Şekil 4.5.), diğer binarda ise (#2, #4-26) çıkarılan düzlemde olması beklenmeyen noktaların elde edilen düzlemde bulunduğu görülmüştür (Şekil 4.6. ve Ekler kısmında gösterilmektedir). Yukarıda bahsedildiği gibi bir düzlemin sınırsız büyüklükte olmasından dolayı bu tür noktaların olması çalışmada beklenmekteydi. Bu problemin giderilmesi için, bu tez çalışmasında kullanılan RANSAC algoritmasının ileride modifiye edilmesi düşünülmektedir.

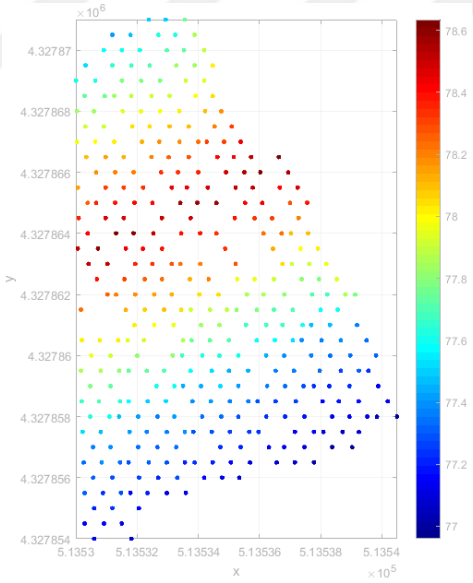
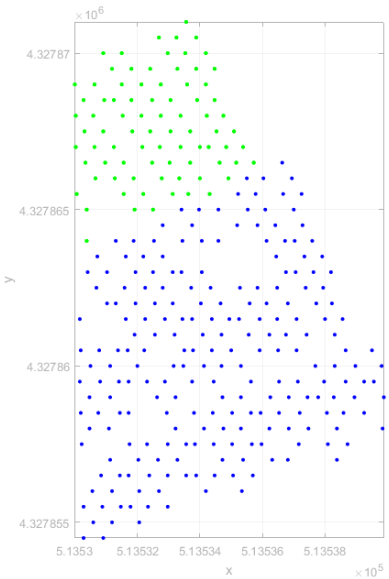
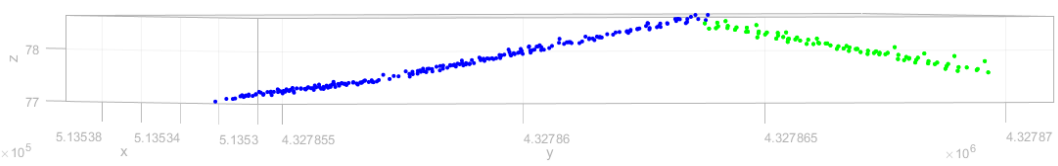
Çalışmada kullanılan iki test alanının birbirlerine göre üstünlükleri olmasına rağmen, test alanı #1 ve test alanı #2 karşılaştırıldığında, test alanı #2'de bulunan binalarda bir düzlemde birden fazla düzlem noktaları olduğu için, test alanı #1'in sonuçları genel olarak daha iyi çıkmıştır.

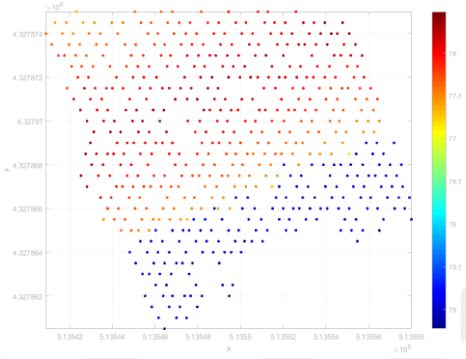
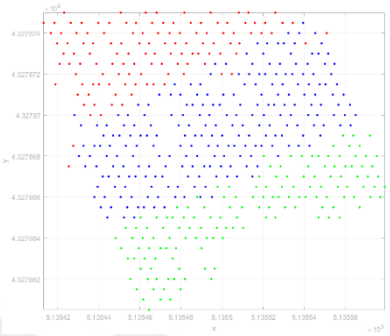
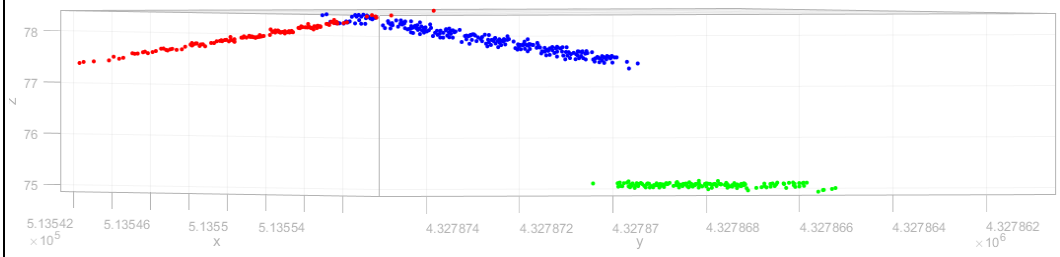
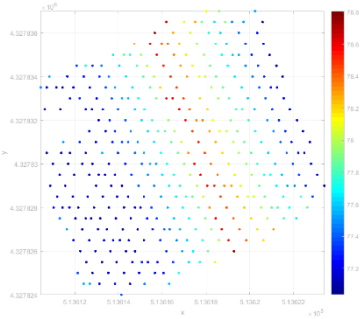
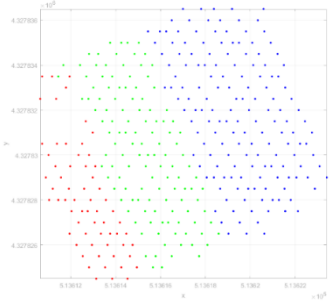
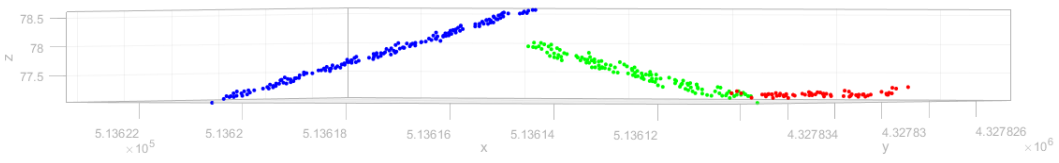
Test alanı #2'de nokta yoğunluğunun yüksek olmasından dolayı, bina bacaları üzerine düşen nokta sayısı da artmaktadır. RANSAC algoritması düzlem noktalarını tespit ettiği için bu

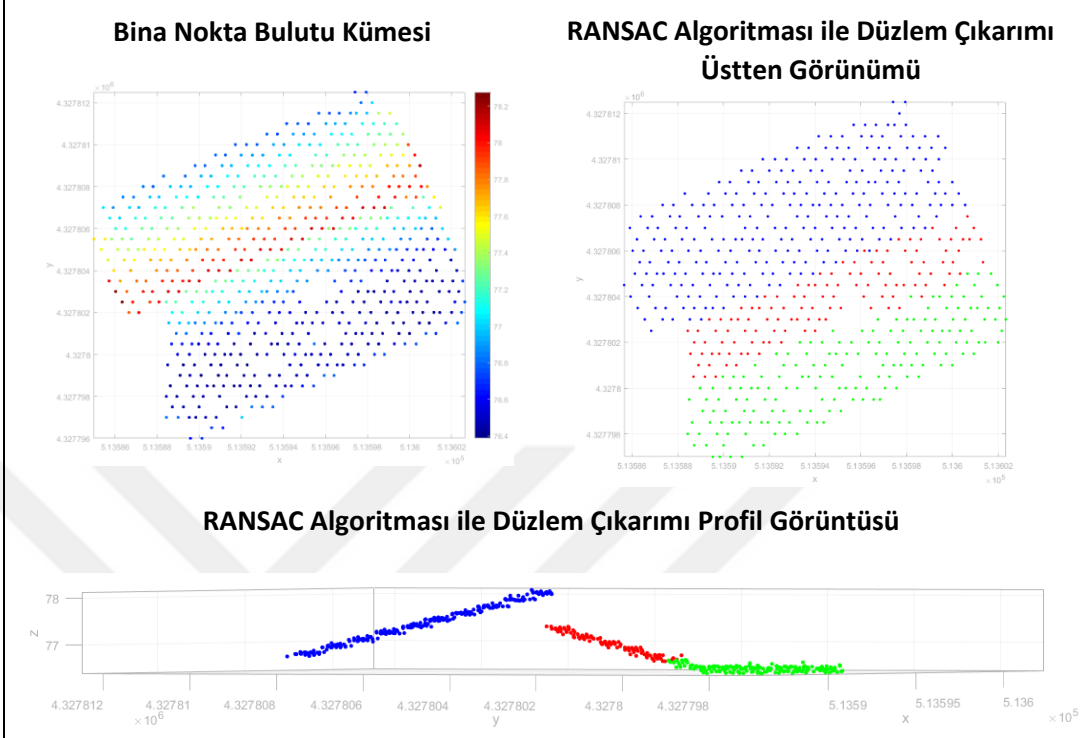
bacalara ait noktaların algoritma tarafından çıkarılan düzlem noktalarına dahil edilmediği görülmüştür. Bu da elde edilen sonuçların doğruluğunu artırmıştır.

Bu tez çalışmasında yöntem kısmında test alanı #3 incelendiğinde, nokta yoğunluğunun yüksek olması ve buna ek olarak bina çatı geometrilerinin düzgün olması bu test alanında uygulanan RANSAC algoritmasının, diğer test alanlarına göre daha başarılı sonuçlar vermesine neden olmuştur.

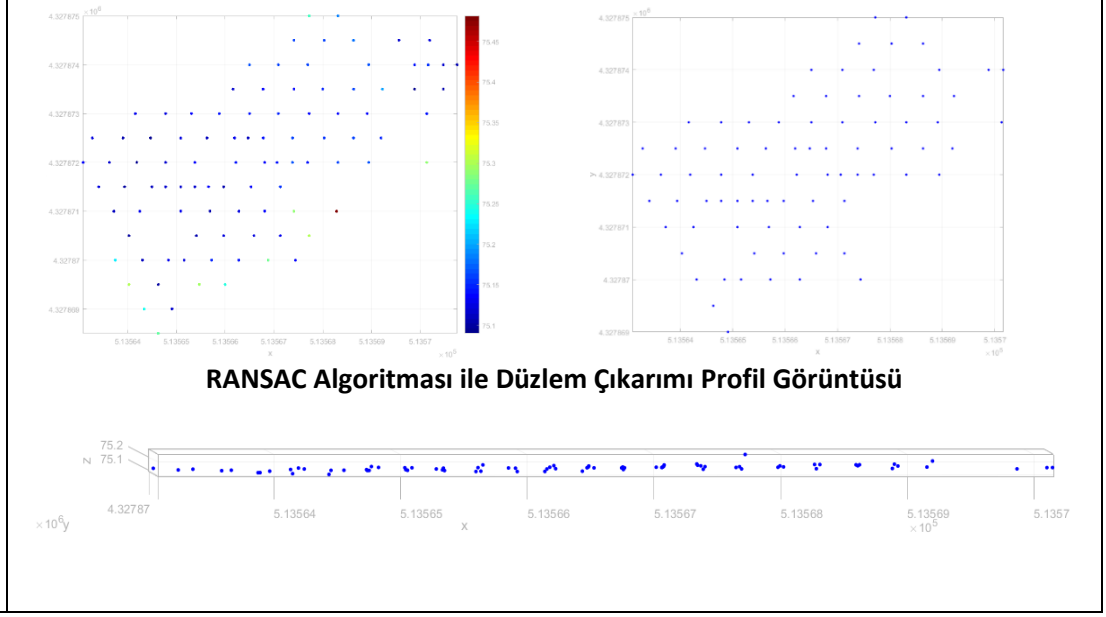
Bu çalışmada LiDAR bina nokta bulutu verisi kümesine RANSAC algoritması uygulandığında önce mavi sonra yeşil ve en son olarak da kırmızı renkli düzlemlere ait nokta grupları çıkarılmıştır. Yalnız test alanı #1'de bulunan ve dört düzleme sahip #15 nolu bina (Şekil 4.3.) için son olarak sarı renkli düzleme ait nokta bulutu çıkarılmıştır.

Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#1		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

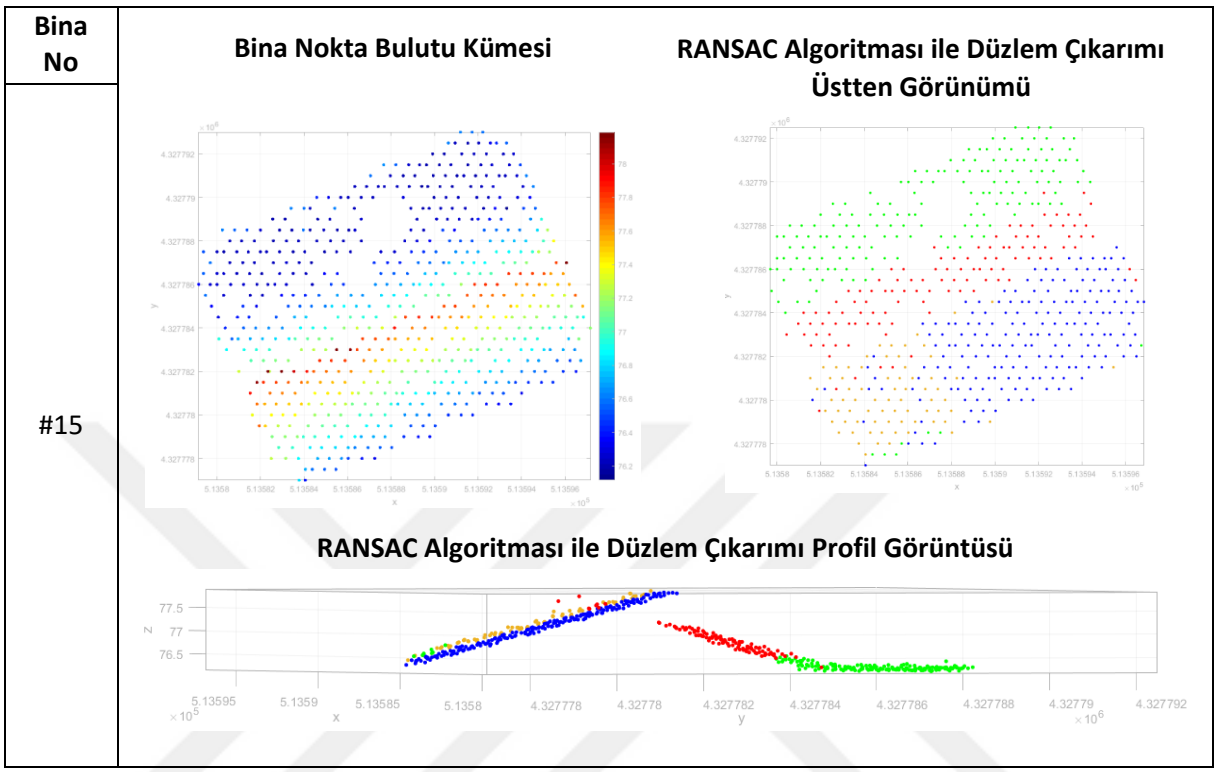
#2	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 
#8	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 

#11	

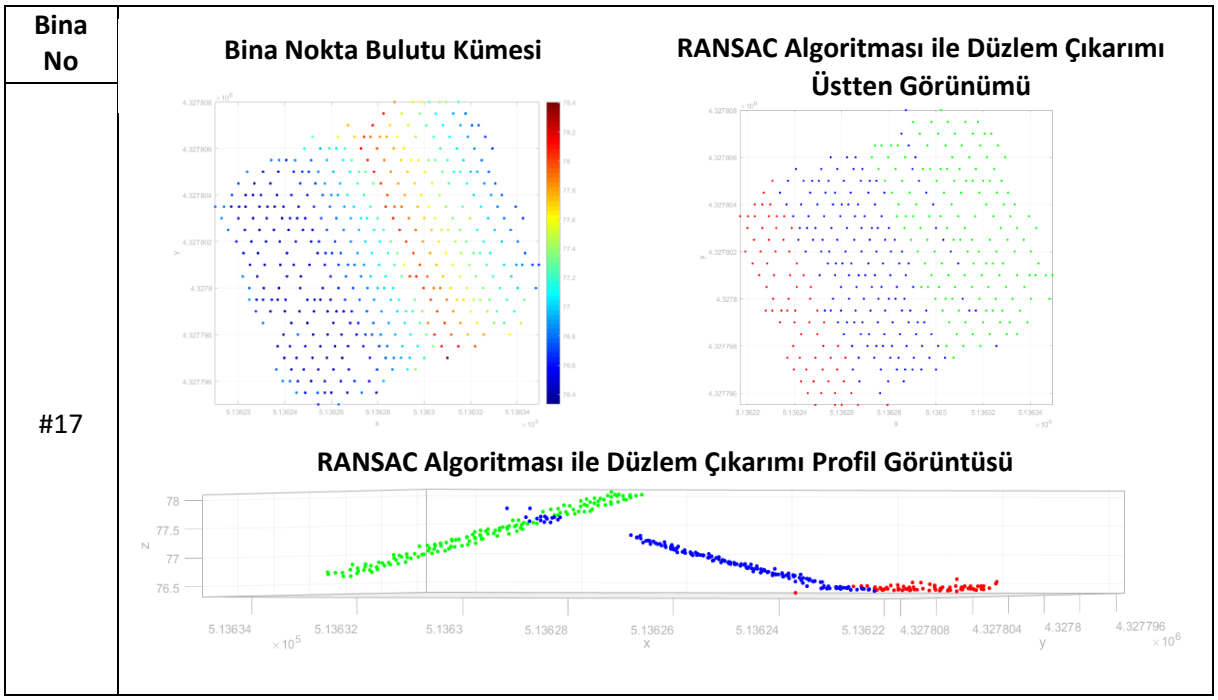
Şekil 4.1. Test alanı #1'de yer alan #1, #2, #8 ve #11 nolu binalara ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi.

Bina No	
#3	

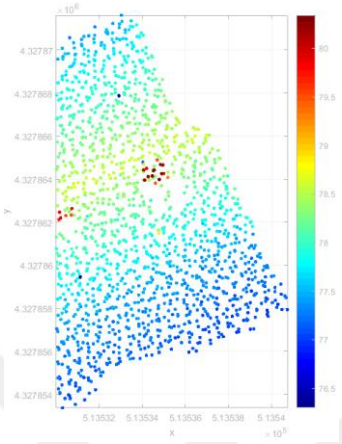
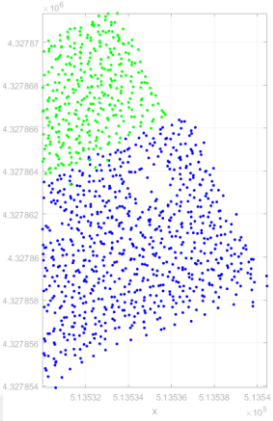
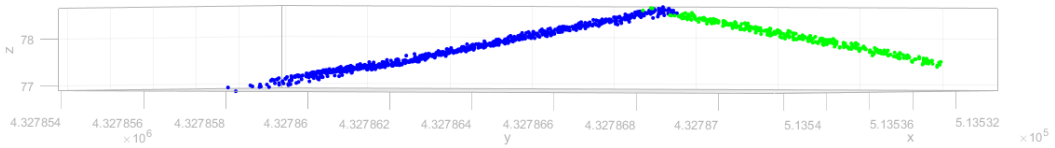
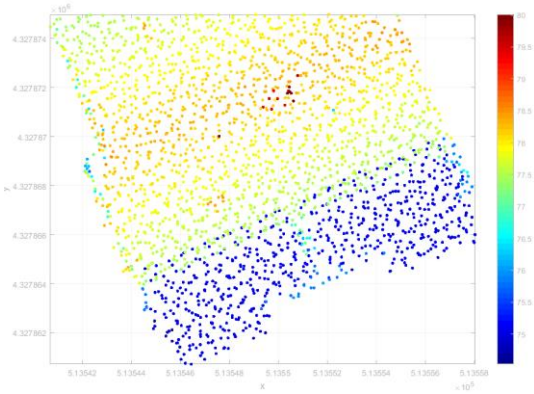
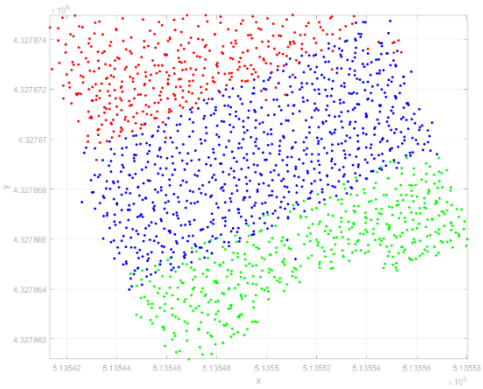
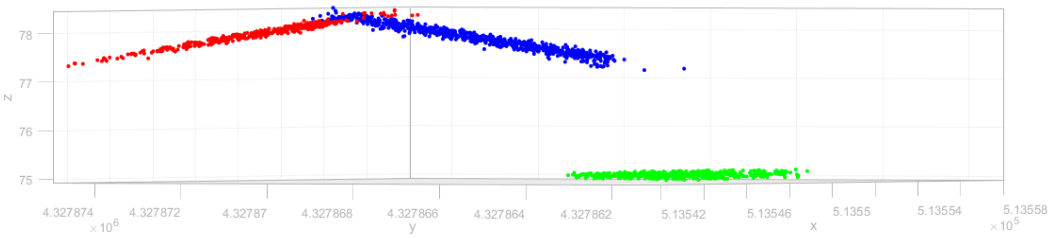
Şekil 4.2. Test alanı #1'de yer alan #3 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi.

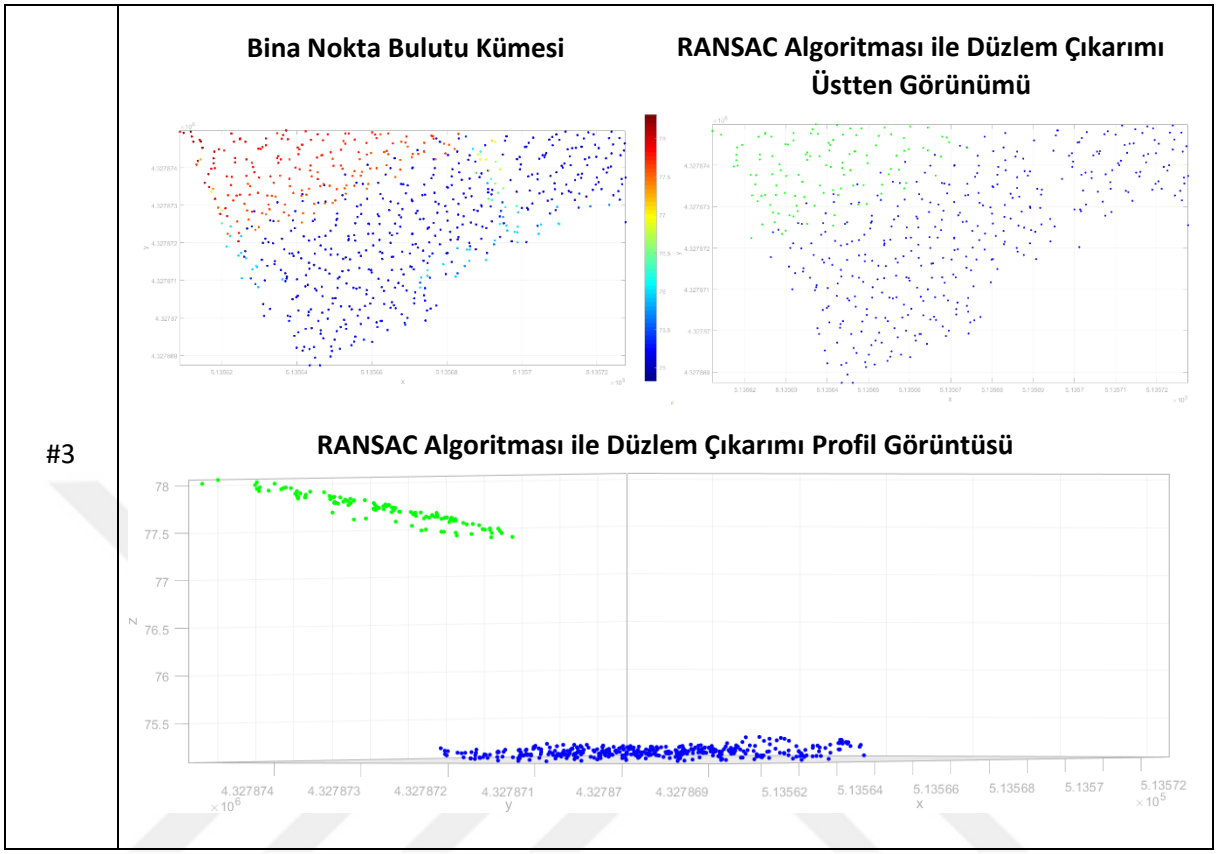


Şekil 4.3. Test alanı #1'de yer alan 15 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi.

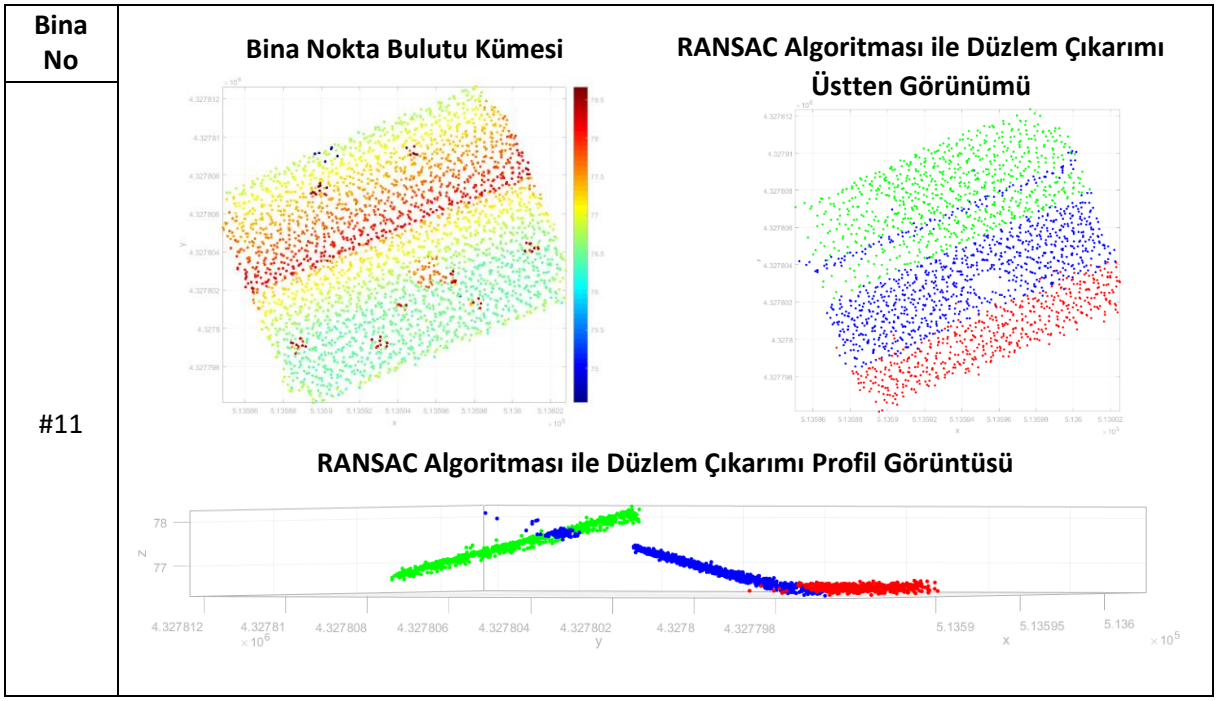


Şekil 4.4. Test alanı #1'de yer alan #17 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi.

Bina No	
#1	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 
#2	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 



Şekil 4.5. Test alanı #2'de yer alan #1, #2 ve #3 nolu binalara ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi.



Şekil 4.6. Test alanı #2'de yer alan #11 nolu binaya ait RANSAC Algoritması ile Düzlem Çıkarımı Gösterimi.

4.2. Gürültünün Ayıklanması

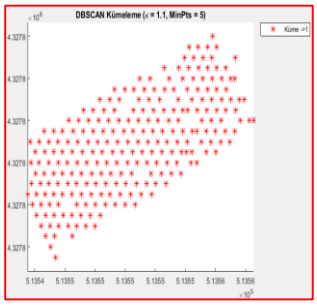
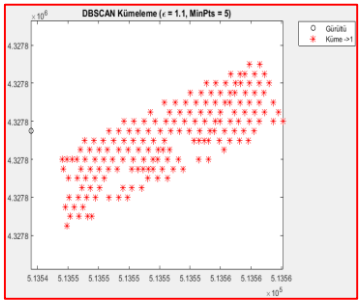
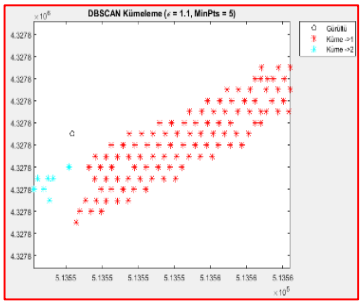
Bu çalışmada, bir düzlemde olmaması gereken noktaların ayıklanması işlemi DBSCAN algoritması ile gerçekleştirilmiştir. Test alanlarına uygulanan DBSCAN algoritması parametre değerleri Çizelge 3.5'te gösterilmiştir. Her bir test alanında, her düzlem nokta bulutu için DBSCAN algoritması parametre değerleri sabit alınmıştır.

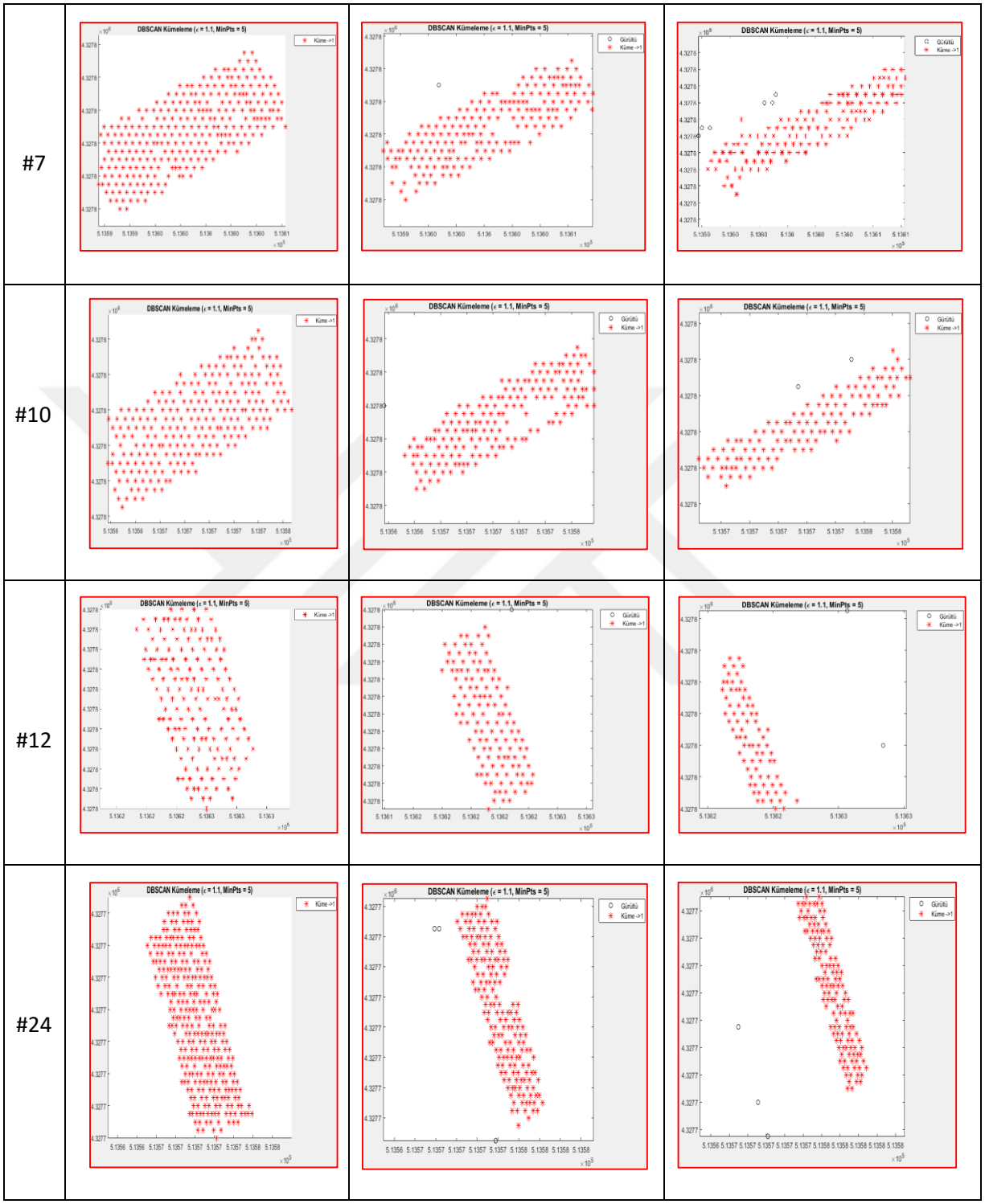
DBSCAN algoritmasının parametre değerlerinin seçiminde:

- test alanlarına ait LiDAR nokta bulutu yoğunluğunun farklılık göstermesi,
- çıkarılan düzlemde olması beklenmeyen ve gürültü olarak tanımlanan noktaların sayısının ve bu noktalar arasındaki mesafelerin farklılık göstermesi ve
- RANSAC algoritması ile çıkarılan düzlemlere ait noktalarda, gürültü noktaları ile düzlemde olması beklenen noktaları arasındaki mesafeler

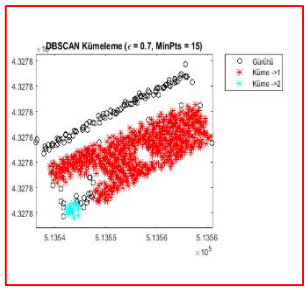
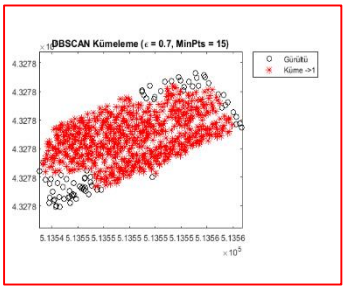
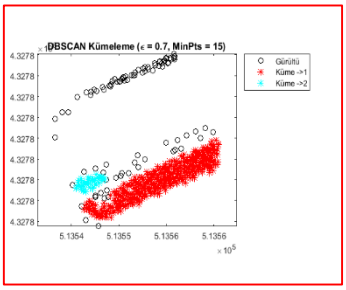
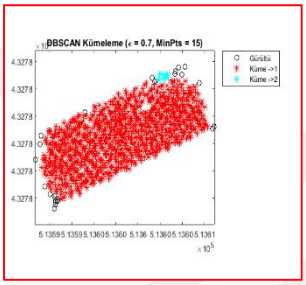
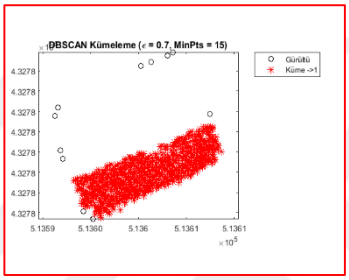
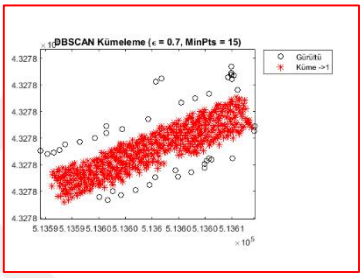
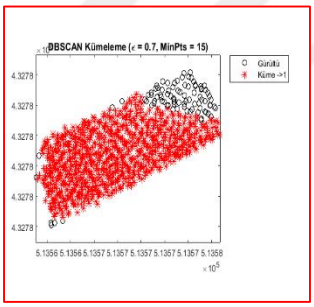
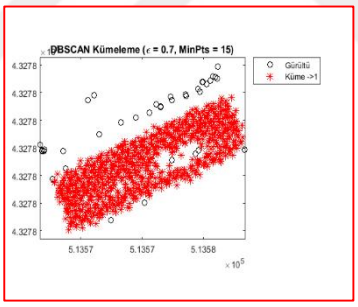
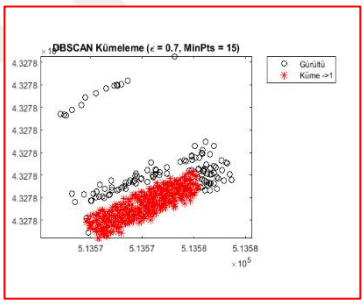
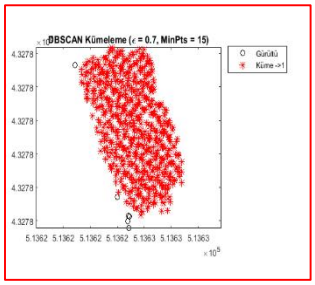
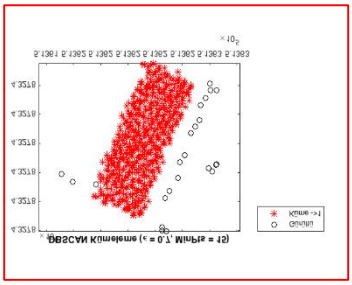
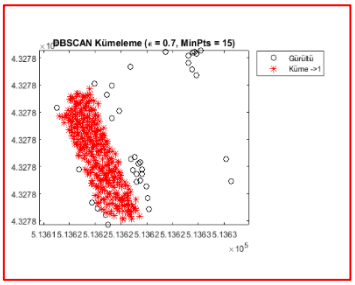
dikkate alınmıştır.

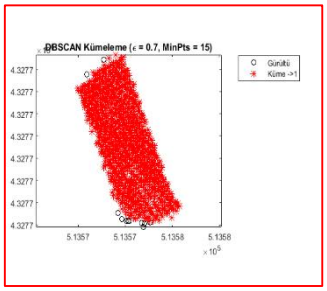
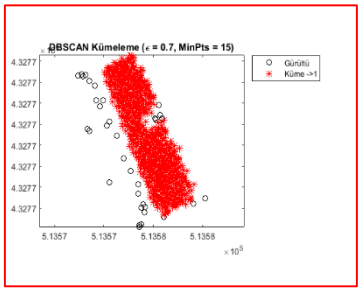
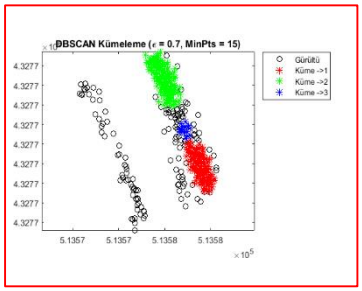
DBSCAN algoritması uygulanarak tespit edilen gürültü noktalarına örnekler olarak, test alanı #1 ve test alanı #2'den seçilen aynı numaralı binalar (#5, #7, #10, #12 ve #24) test alanı #1 için Şekil 4.7'de ve test alanı #2 için Şekil 4.8'de gösterilmiştir.

Bina No	DBSCAN Kümeleme		
#5			



Şekil 4.7. DBSCAN algoritması ile test alanı #1'de yer alan #5, #7, #10, #12 ve #24 numaralı binalara ait çatı düzlemlerinde bulunan gürültü noktalarının tespit edilmesi.

Bina No	DBSCAN Kümeleme		
#5			
#7			
#10			
#12			

#24			

Şekil 4.8. DBSCAN algoritması ile test alanı #2'de yer alan alan #5, #7, #10, #12 ve #24 numaralı binalara ait çatı düzlemlerinde yer alan gürültü noktalarının tespit edilmesi.

Her bir test alanında, o test alanındaki binaların çatılarının çıkarılması için gerekli DBSCAN algoritması parametreleri değerleri tüm binalar için aynı alınmıştır. Dolayısıyla, bu algoritma, örnek sayısı fazla ve düşük nokta yoğunluğuna sahip olan test alanı #1 için, Şekil 4.7'de de görüldüğü üzere, düzlem LiDAR nokta bulutu verilerinde otomatik gürültü noktalarının tespiti işlemini başarılı bir şekilde gerçekleştırmiştir. Ancak, örnek sayısı fazla ve yüksek nokta yoğunluğuna sahip olan test alanı #2 için, Şekil 4.8.'de de görüldüğü üzere, algoritma düzlem LiDAR nokta bulutu verilerinde otomatik gürültü noktalarının tespitini beklenen başarı ile gerçekleştirememiştir. Yöntem kısmında test alanı #3 incelendiğinde ise bu alan için tek sabit parametre değeri kullanılmıştır. Bu parametre değeri bu alanda bulunan az sayıda bina düzlemi (toplam 12 düzlem) üzerine uygulandığı için, elde edilen sonuçlar nokta yoğunluğundan etkilenmemiş ve düzlemler başarılı bir şekilde çıkarılmıştır.

Ayrıca, Şekil 4.7. ve Şekil 4.8. incelendiklerinde yüksek nokta yoğunluğuna sahip test alanı #2'de RANSAC algoritması ile çıkarılan düzlemlerde düzlem noktaları olarak algoritma tarafından çıkarılamayan, çatıda bulunan bacaların neden olduğu boşluklar ile diğer nedenlerden kaynaklı boşluklar, gürültü noktalarının tespitinde olumsuz sonuçlar çıkarmasına karşın, az yoğunluğa sahip test #1 alanında bu sorunla karşılaşılmamıştır.

Her bir test alanında bulunan binalar için aynı DBSCAN algoritması parametre değerlerinin kullanılmış olmasından dolayı, bazı binalarda gürültü olmayan noktaların gürültü noktaları, gürültü noktaların ise gürültü olmayan noktalar olarak tespit edilmesine neden olduğu görülmüştür.

DBSCAN algoritmasındaki problemlerin en aza indirgenmesi için her bir düzlem nokta bulutuna ayrı parametre değerinin uygulanması gerekmektedir. Ancak, bu hem yapılan işlemin

süresini artıracaktır hem de önerilen yöntemin otomatik olma özelliğini kaybetmesine yol açacaktır. Bu nedenle, bu çalışmada çalışma alanında yer alan tüm binalar için aynı parametre değerlerinin kullanılması tercih edilmemiştir.

Diğer binalar için çıkarılan düzlemlerde tespit edilen gürültü noktaları ekler bölümünde gösterilmiştir.

4.3. Çatı Düzlemi Sınırlarının Çıkarılması ve Sadeleştirilmesi

Her bir binaya ait gürültüden arındırılmış düzlem LiDAR nokta bulutu verisinin en dış noktalarından sınır geçirilerek o binayı temsil eden düzlem sınırları çıkarılmıştır. Bunun yanı sıra, düzlem sınır noktalarına Douglas-Peucker algoritması uygulanarak sınırların sadeleştirilmesi (düz çizgilerin oluşturulması) yapılmıştır. Özetle, bina çatı düzlem sınırlarının çıkarılması ve sadeleştirilmesi işleminde, her binaya ait tespit edilen her ayrı çatı düzleminin sahip olduğu noktalar ile çatı düzlemlerinin modellenmesi amaçlanmıştır.

Modellenmiş çatı düzlemleri birinci test alanındaki 26 bina için incelendiğinde, #1, #3, #8, #10, #11, #13, #16, #17 ve #23 nolu bina çatı modellerinin doğru bir şekilde oluşturulduğu görülmektedir (Şekil 4.9.). Yine, birinci test alanında bulunan #2, #9, #15, ve #26 nolu binalar incelendiklerinde her bir binaya ait çatı düzlemleri için modelleme yapılmış ancak beklenen doğrulukta sonuçlar elde edilememiştir (Şekil 4.10.).

Aynı şekilde, modellenmiş çatı düzlemleri sonuçları ikinci test alanındaki 26 bina için incelendiğinde #1, #3, #7, #8, #10 ve #12 nolu binalara ait çatı modellerinin doğru bir şekilde oluşturulduğu görülmektedir (Şekil 4.11.). Yine, ikinci test alanında bulunan #2, #4, #13, #14, #17 ve #18 nolu binalar incelendiklerinde her bir binaya ait çatı düzlemleri için yapılan modellemenin beklenen doğrulukta sonuçlar vermediğini göstermiştir (Şekil 4.12.).

Yöntem kısmında verilen test alanı #3 incelendiğinde, bu test alanında bulunan binalara ait tüm düzlemler için modelleme sonuçlarının oldukça tatmin edici oldukları görülmektedir.

Diğer binalar için, her bir binaya ait çatı düzlemi modelleri görüntüleri ekler bölümünde verilmiştir.

Her iki test alanında “boundary” fonksiyonu ve Douglas-Peucker algoritması uygulanarak elde edilen sonuçlar incelendiklerinde, birinci test alanında elde edilen sonuçların ikinci test alanından elde edilen sonuçlara göre daha başarılı oldukları görülmüştür.

Test alanı #3'te diğer test alanlarına daha başarılı sonuçların elde edilmesinin sebebi, DBSCAN algoritması aynı parametre değerlerinin daha az sayıda örnek üzerine uygulanmasından dolayı gürültülerin daha başarılı bir şekilde tespit edilmesi ve RANSAC algoritmasının bu test alanında daha başarılı sonuç vermesidir.

Bazı binalara ait çatı düzlemlerinde başarısız düzlem modellerinin elde edilmesinin ana nedenleri aşağıdaki gibidir:

- Bina çatı düzlemleri üzerinde boşluklar bulunması,
- Çıkarılan düzlem noktalarının bir hizada olmaması, başka bir deyişle düzem sınırlarının zig-zag'lı yapıda olması,
- Yalnız bir adet düzleme ait noktaların olması gerektiği alanda iki ayrı düzlem noktalarının bulunması (Şekil 4.3. 15 nolu bina) ve
- RANSAC, DBSCAN ve Douglas-Peucker algoritmalarında tek parametre değerlerinin kullanılmış olması.

Yukarıda belirtildiği üzere, elde edilen bazı bina düzlemlerinin beklenen doğrulukta modellenememiş olmasına neden olan en önemli etken, tespit edilen düzlemlerde noktalar arasında bulunan boşluklardır. Bu boşlukların oluşmasında, aşağıda belirtilen nedenlerden biri ya da birkaçının temel etken olduğu görülmüştür.

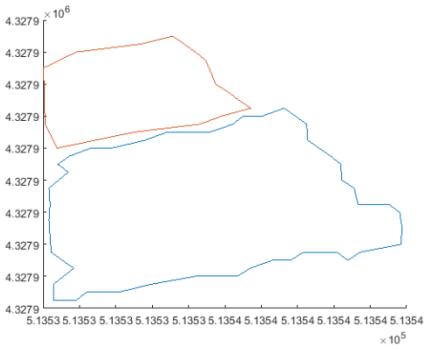
- RANSAC algoritmasında "*mesafe eşik değer*" parametresinin tüm bina çatı düzlem noktalarının elde edilmesinde tek bir değer olarak alınmasıyla, bu parametre mesafe değeri sınırları içerisinde bulunan tüm noktalardan bir düzlem belirtilirken elde edilen bazı düzlemlerde boşluklar oluşmaktadır.
- RANSAC algoritması en iyi matematiksel düzlemi tespit ettiği için, bina duvarları ve baca gibi çatı düzlemine yakın sonuçlar vermeyen noktalar, düzlem dışı noktalar olarak kabul edilmiştir. Binalarda yer alan bacalara ait noktalar düzlem noktalarına dahil edilmediği için binanın bu kısımlarında boşluklar oluşabilmektedir.
- LiDAR nokta bulutuna filtreleme uygulandıktan sonra sınıflandırma işlemi adımında yalnız bina sınıfını elde etmek için sınıflandırılmış veriden (zemin, ağaç ve bina noktaları) zemin ve ağaç (yüksek bitki örtüsü) noktaları çıkarılmıştır. LiDAR verisinden zemin ve ağaç (yüksek bitki örtüsü) sınıflarına ait noktalar çıkarıldığında, ağaçlara ait

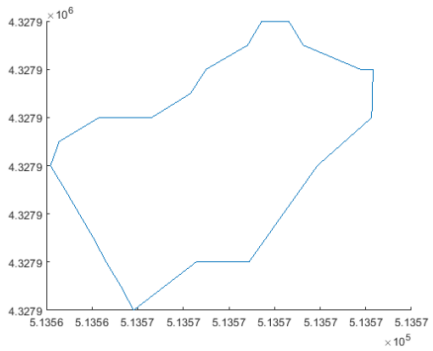
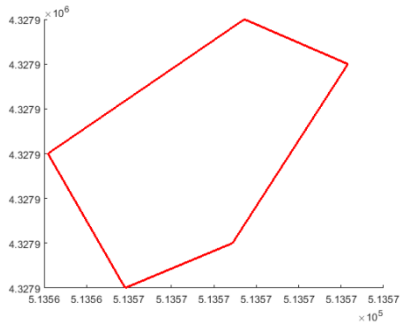
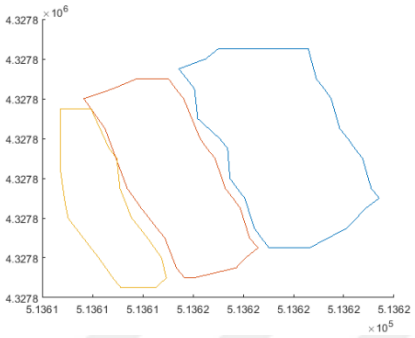
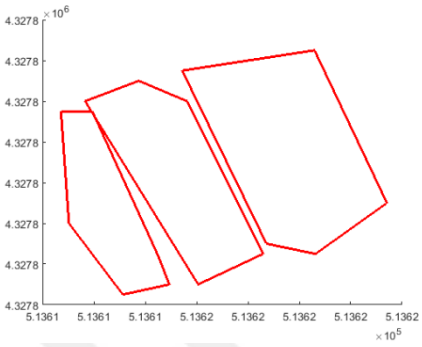
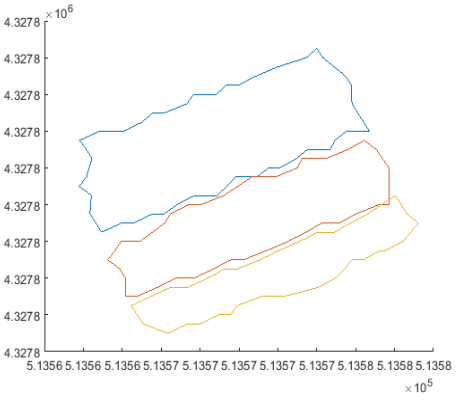
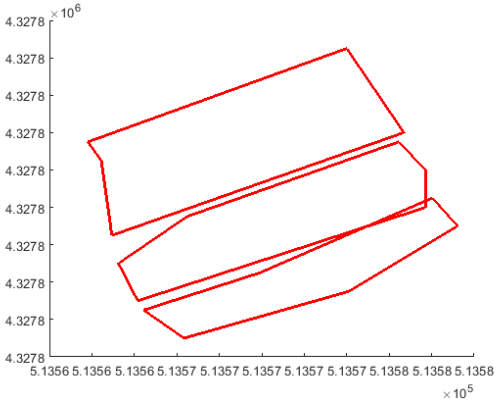
noktaların binaların üzerlerine sarkan kısımları da (Şekil 4.13.) binaların üzerlerinde boşluklar oluşturmaktadır.

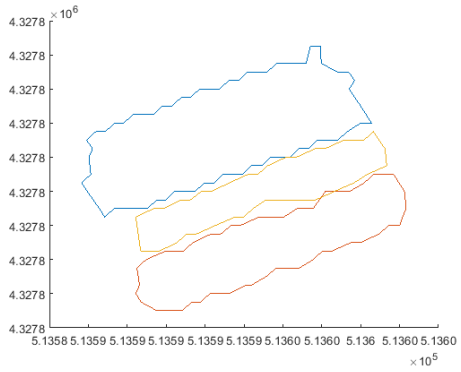
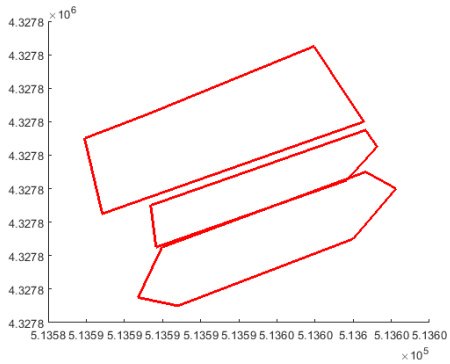
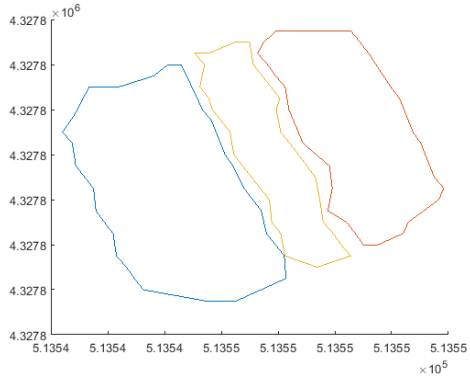
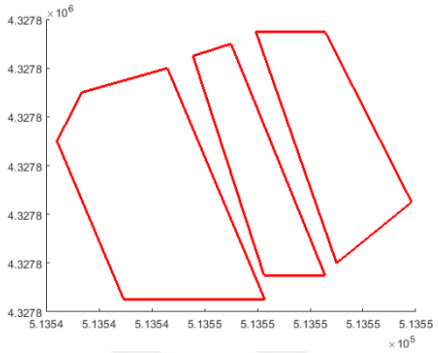
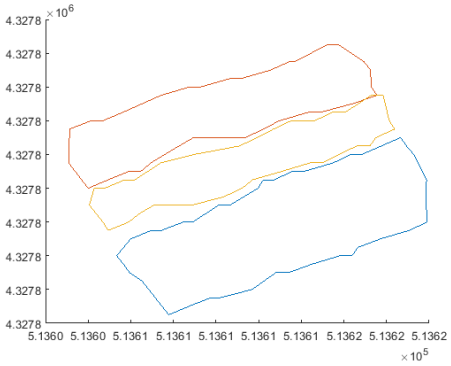
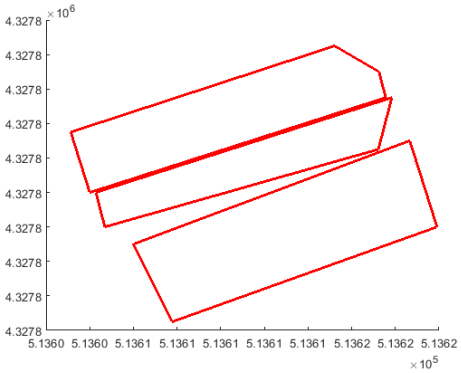
Bu çalışmada önerilen yaklaşımla çıkarılan çatı düzlemleri incelendiğinde, yukarıda bahsedilen boşlukların düzlemlerin orta kısımda yer aldığı durumlarda sınırlar başarılı bir şekilde çıkarılmıştır. Ancak, boşlukların çatı düzlemlerinin sınır kısımlarında yer almadığı durumlarda, eğer boşluklar arası mesafe az ise sınırların yine başarılı bir şekilde çıkarılmış, ancak boşluklar arası mesafenin fazla olduğu durumlarda çıkarılan sınırların düzlemlerin gerçek geometrisini temsil etmediği görülmüştür. Diğer taraftan, hem oluşan boşlukların aralarındaki mesafenin uzun olması hem de boşlukların çatı düzleminin her iki tarafında bulunması elde edilen sonuçları olumsuz yönde etkilemiştir.

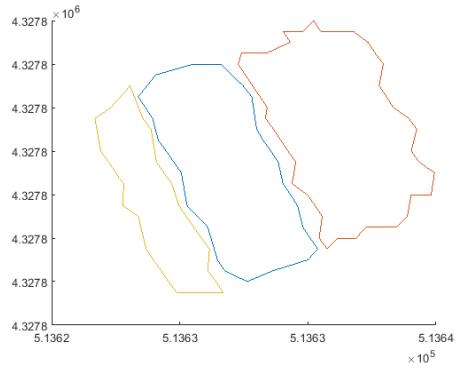
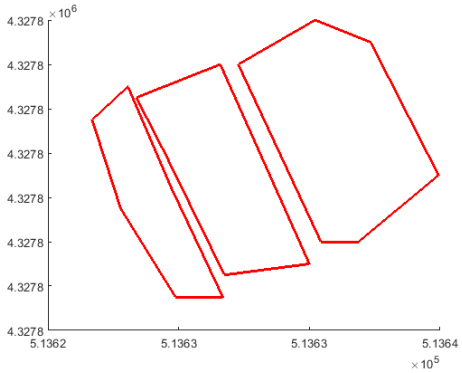
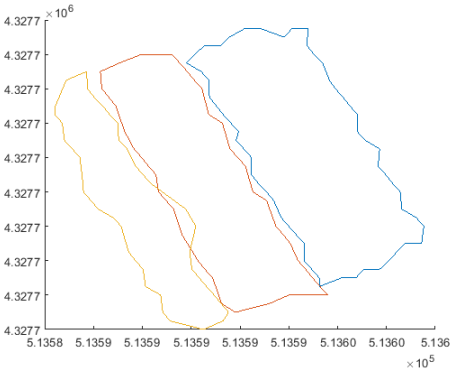
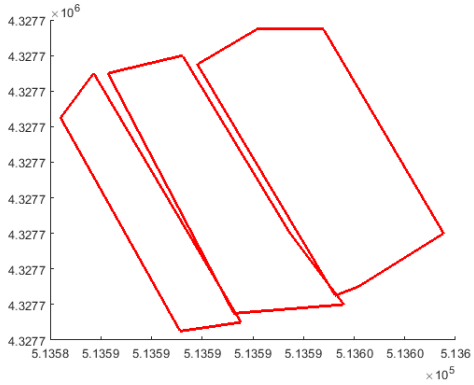
Diğer taraftan, RANSAC algoritması ile çıkarılan çatı düzlemleri düzgün bir geometriye sahip olduklarında, sınırların başarılı bir şekilde çıkarıldığı görülmüştür.

Her bir test alanında yer alan binalar için aynı DBSCAN algoritması parametre değerlerinin kullanılması durumunda bazı binalarda gürültü olmayan noktaların gürültü noktaları, gürültü noktalarının ise gürültü olmayan noktalar olarak tespit edildiği görülmüştür. Bu da sınır çıkarımı işleminde başarısızlığa sebep olmaktadır. Gürültü noktalarının gürültü olmayan noktalar olarak tespit edilip veriden atılmaması durumunda, elde edilmek istenen düz sınır çizgisi dışa doğru bir kırılma gerçekleştirerek eğri bir sınır çizgisi halini almaktadır. Yine, bina çatısı sınırına yakın konumda olup gürültü olmayan noktaların gürültü noktaları olarak tespit edilmesi durumunda düz olması beklenen sınır çizgisi içe doğru bir kırılma ile gerçek geometriden uzak bir sınır çizgisi halini almaktadır.

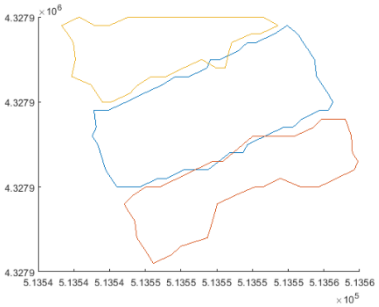
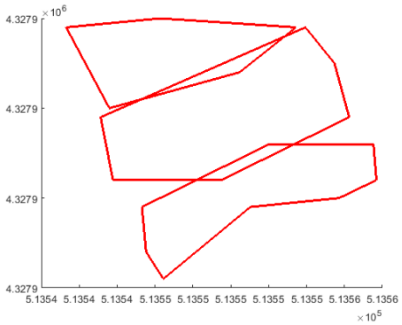
Bina No	Sınırların Çıkarılması	Sınır Sadeleştirmesi
#1		

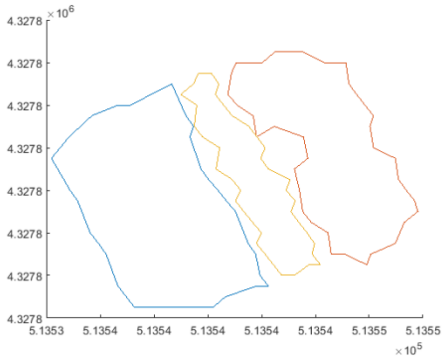
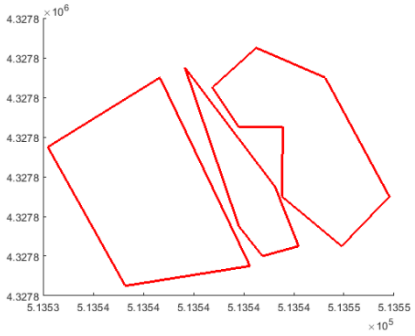
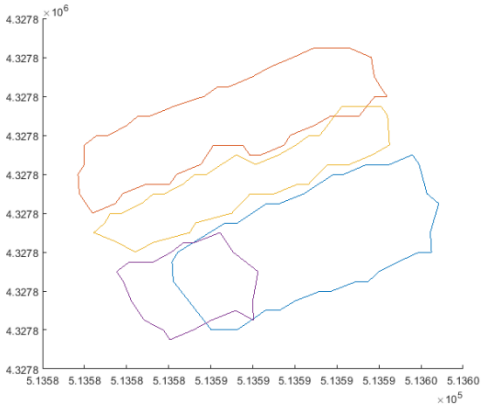
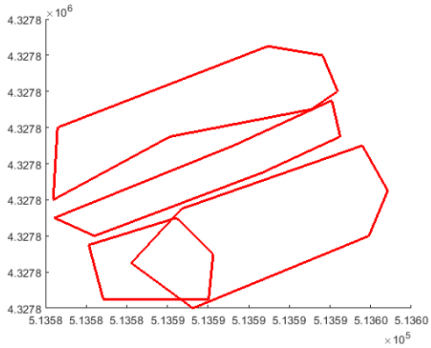
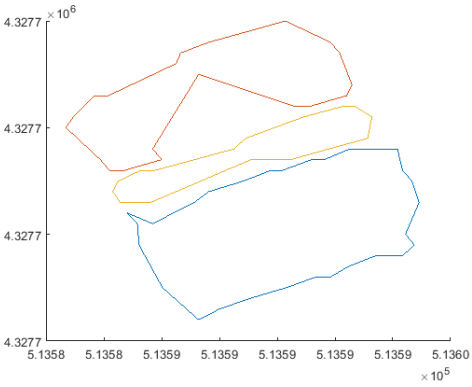
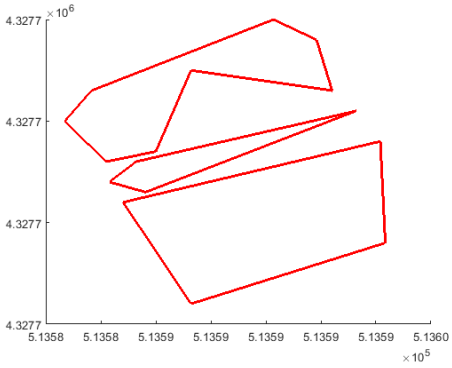
#3		
#8		
#10		

#11		
#13		
#16		

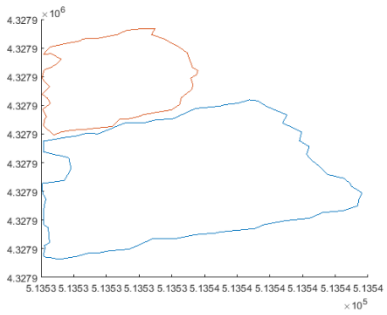
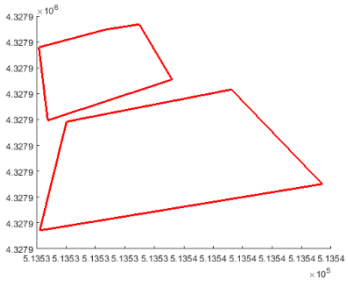
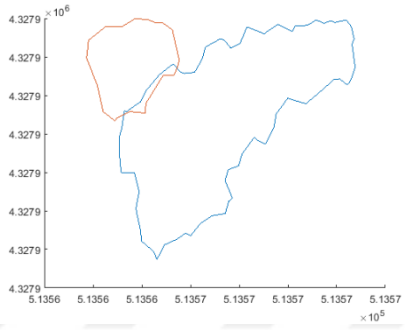
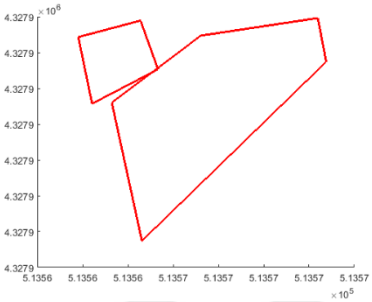
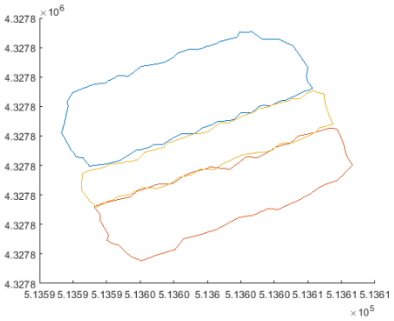
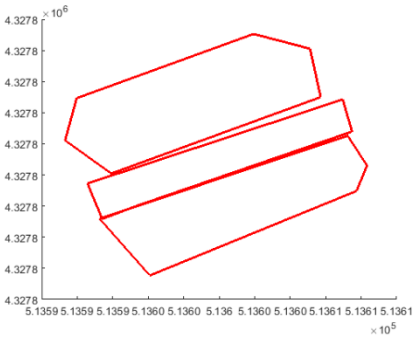
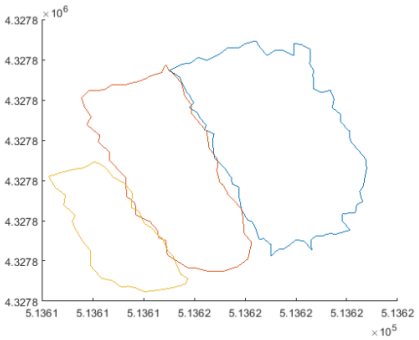
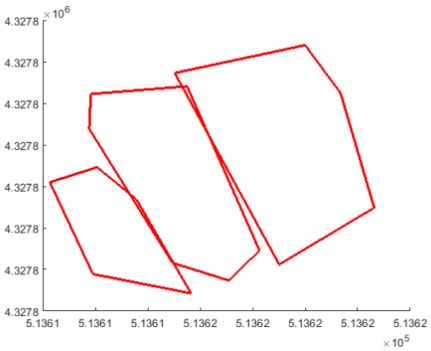
#17		
#23		

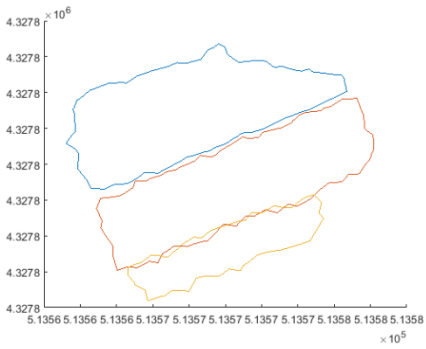
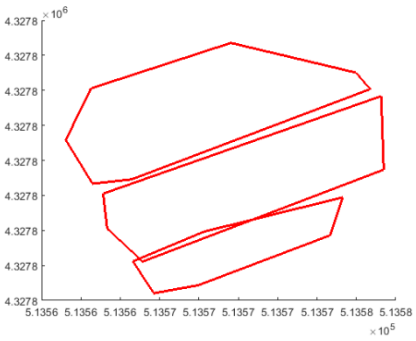
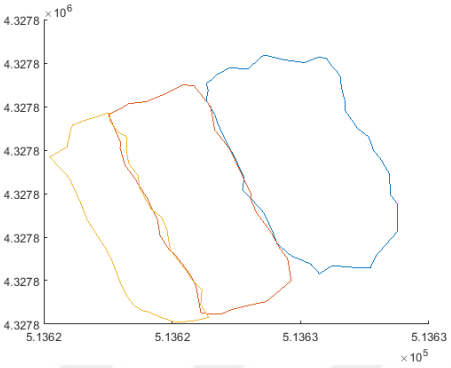
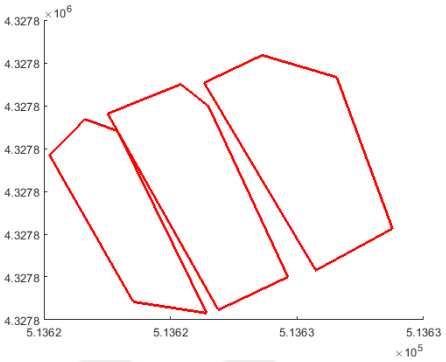
Şekil 4.9. Çatı modellemesi amacıyla test alanı #1'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarılı örnekleri.

Bina No	Sınırların Çıkarılması	Sınır Sadeleştirmesi
#2		

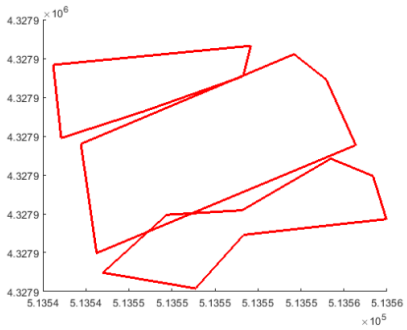
#9		
#15		
#26		

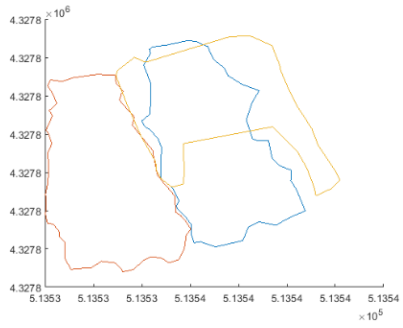
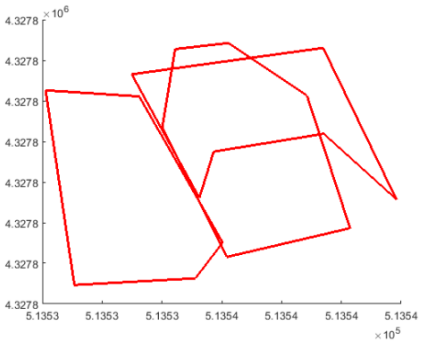
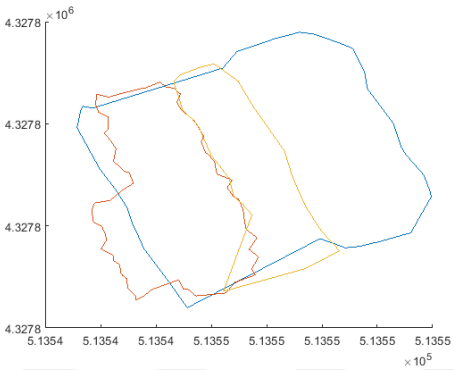
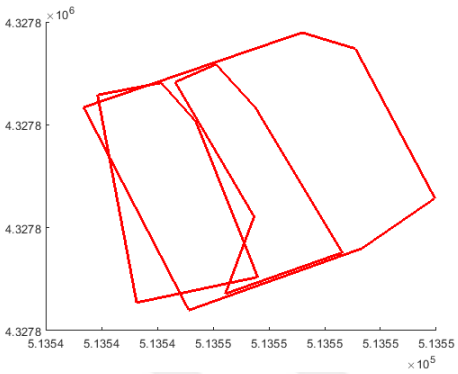
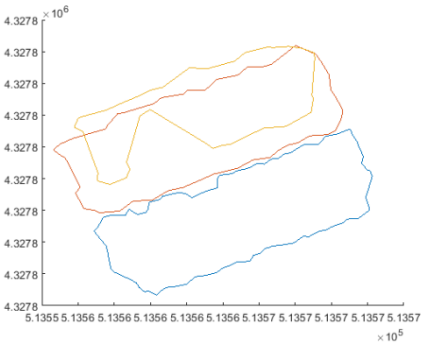
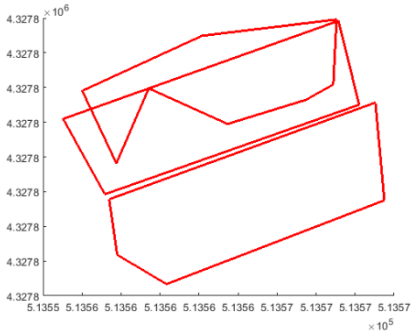
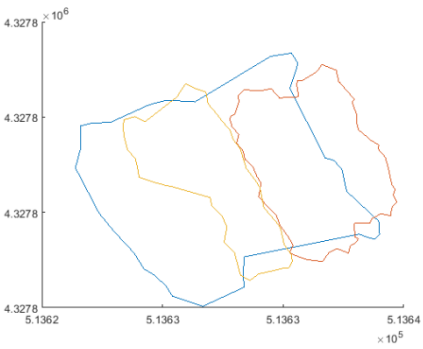
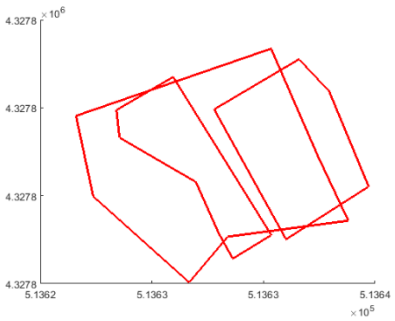
Şekil 4.10. Çatı modellemesi amacıyla test alanı #1'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarısız örnekleri.

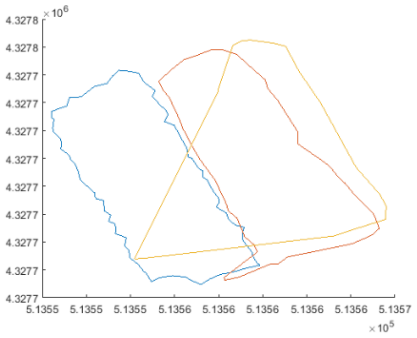
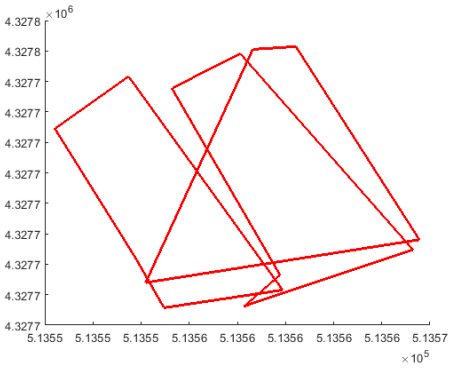
Bina No	Sınırların Çıkarılması	Sınır Sadeleştirmesi
#1		
#3		
#7		
#8		

#10		
#12		

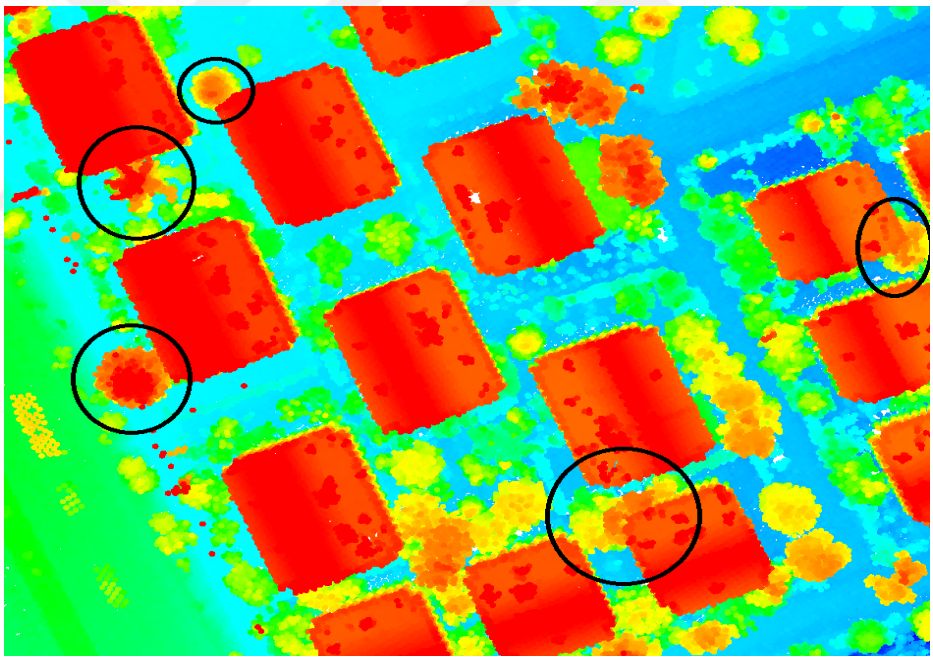
Şekil 4.11. Çatı modellemesi amacıyla test alanı #2'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarılı örnekleri.

Bina No	Sınırların Çıkarılması	Sınır Sadeleştirmesi
#2		

#4		
#13		
#14		
#17		

#18		

Şekil 4.12. Çatı modellemesi amacıyla test alanı #2'de düzlem sınırlarının çıkarılması ve bu sınırların Douglas-Peucker algoritması ile sadeleştirilmesi başarısız örnekleri.



Şekil 4.13. Binalar üzerine ağaçların sarkması.

5.SONUÇLAR

3B şehir modellerinde binalar önemli bir rol üstlenmektedir. Şehir alanlarında nüfusa dayalı olarak yeni yerleşim alanlarının belirlenmesi, kentsel dönüşüm vb. yapılaşmalardan dolayı bina nesneleri bilgisi güncelliğini yitirmektedir. Bu tez çalışmasında, 3B bina nesnelerini hızlı, doğru ve güncel olarak çıkarmak için hava LiDAR nokta bulutu verisi kullanılmıştır. Çalışmada iki farklı uçuş yüksekliğinde (1200 ve 2600 metre), üç farklı yoğunlukta (3.42, 18.77 ve 22.63 nokta/m^2), iki farklı LİDAR veri sistemi (Riegl firmasının LMS-Q1560 ve Optech firmasının Pegasus HA-500) ile elde edilen ve tek bir şeritte yer alan 3B LiDAR nokta bulutu verisi kullanılmıştır.

Bu çalışmada, farklı geometrilere sahip bina çatı düzlemleri otomatik olarak çıkarılarak modellenmiştir. Çatıların modellenmesi kapsamında, öncelikli olarak "LAStools" yazılımında seçilen üç test alanına ait LiDAR verilerinden, gürültülerin giderilmesi, çalışma alanında kaç şeridin örtüştüğünün incelenmesi, zemin ve zemin üstü nesnelerin birbirlerinden ayrılması, bina ve ağaçların sınıflandırılması ve sonrasında binaların çıkarılması işlemleri yapılmıştır. Çıkarılan tüm binaların dahil olduğu bina sınıfına "Microsoft Visual Studio" yazılımında bölge büyüme bölütleme algoritması C++ kodu uygulanarak, her bir bina nokta bulutu kümesi algoritma tarafından ayrı olarak tespit edilmiştir. Tespit edilen her bina nokta bulutu kümesine "MATLAB" ortamında RANSAC algoritması uygulanarak bina çatılarının düzlemsel yüzeyleri otomatik olarak çıkarılmıştır. Çatı düzlemlerini doğru bir şekilde modellemek için, her binaya ait nokta bulutu kümesine RANSAC algoritması uygulandıktan sonra her bireysel çatı düzleminde var olan gürültü noktaları yine "MATLAB" ortamında DBSCAN algortiması kullanılarak tespit edilmiş ve bina çatısı düzlem nokta bulutu verisinden atılmıştır. Gürültüler giderildikten sonra "MATLAB" ortamında "boundary" foksiyonu ile sınırı temsil eden düzlem noktalarından sınırlar geçirilmiş ve elde edilen bina düzlem sınırları Douglas-Peucker algoritması ile sadeleştirilmiştir.

Bu tez çalışması sonucunda elde edilen sonuçlar ve öneriler şu şekildedir:

- "LAStools" yazılımında filtreleme işlemi son dönüş noktaları üzerinden gerçekleştirilmiştir. Filtreleme işlemi temelde "*step size*" parametresine dayandığı için bu parametrenin belirlenmesinde en büyük nesne olan binaların dikkate alınması gerektiği sonucuna varılmıştır.

- "LAStools" yazılımının eğitici amaçlı sitesinde, bu tez çalışmasında kullanılan benzer çalışma alanı için "*step size*" parametresi değerinin 10 metre'ye kadar artırılması önerisi dikkate alınarak "*step size*" parametresi değerleri 5, 7 ve 10 metre olarak test alanlarına uygulanmıştır. Ancak, bu çalışmada bina büyüklüğünden daha küçük değerler olan 3 metre ve 5 metre "*step size*" parametresi değerinin yer filtrelemesinde doğru sonuç vermediği görülmüştür. Bina büyüklüğüne yakın olan 10 metre "*step size*" parametresi değeri ise tüm test alanında çok daha başarılı sonuç vermiştir.
- "LAStools" yazılımında bina ve ağaç sınıflarının tespit edilmesi için "*ground offset*" parametresi değerinin dikkatli bir şekilde belirlenmesi gerekmektedir. Bu parametre test alanlarındaki en alçak binanın yüksekliği göz önüne alınarak belirlenmiştir. "ground offset" parametresi ile belirlenen mesafeden sonra hangi noktaların çatıya dahil olduğunu belirten "*building planarity*" standart sapma parametresi, diğer parametreler sabit tutulup artırıldığında çatıya daha fazla noktanın dahil olduğu saptanmıştır. Bu yüzden, sınıflandırma işlemi yaparken ağaçların bina üzerine sarkması ve arazide yükseklik farkının olması yanlış sınıflandırmaya neden olabilmektedir.
- "LAStools" yazılımında sınıflandırma işlemi sonrasında tüm test alanlarında bulunan binalar (bina sınıfı) başarılı bir şekilde çıkarılmıştır.
- Tüm test alanlarında çıkarılan bina sınıfında bulunan her bir binaya ait nokta bulutu verisini ayırabilmek için bina sınıfı versine Microsoft Visual Studio" yazılımında Bölge Büyüme Bölütleme algoritması C++ kodu uygulanmıştır. Bu işlem sonucunda tüm test alanlarında, bina sınıfı kapsamında yer alan her bir binaya ait nokta bulutu verisinin otomatik olarak ve başarılı bir şekilde ayrılması sağlanmıştır.

- Bir binayı temsil eden çatı düzlemlerine ait düzlem noktalarını otomatik tespit etmek için her bir binaya ait nokta bulutu kümesine "MATLAB" ortamında RANSAC algoritması uygulanarak temel büyük çatı düzlemlerinin tamamı doğru olarak tespit edilmiştir. Ancak, çıkarılan bazı düzlemlerde düzleme dahil olmaması beklenen noktaların elde edilen düzlemlerde bulunduğu görülmüştür. Bu problemin giderilmesi için, bu tez çalışmasında kullanılan RANSAC algoritmasının ileride modifiye edilmesi düşünülmektedir. Çalışmada kullanılan test alanı #1 ile test alanı #2 alanları karşılaştırıldığında, test alanı #2 için elde edilen sonuçlarda geometrik olarak bir düzlemsel yüzeyi temsil eden düzlemlerde, bir düzlemde birden fazla düzlem noktaları

olduğu için, test alan #1'in sonuçları genel olarak daha iyidir. Test alanı #3 incelendiğinde, nokta yoğunluğunun yüksek olması ve buna ek olarak bina çatı geometrilerinin daha düzgün olması, bu test alanında RANSAC algoritmasının diğer test alanlarına göre daha başarılı sonuçlar vermesine neden olmuştur.

- Her bir test alanında, o test alanındaki binaların çatılarının çıkarılması için gerekli DBSCAN algoritması parametreleri değerleri tüm binalar için aynı alınmıştır. Dolayısıyla, bu algoritma, örnek sayısı fazla ve düşük nokta yoğunluğuna sahip olan test alanı #1 için, düzlem LiDAR nokta bulutu verilerinde otomatik gürültü noktalarının tespiti işlemini başarılı bir şekilde gerçekleştirmiştir. Ancak, örnek sayısı fazla ve yüksek nokta yoğunluğuna sahip olan test alanı #2 için, algoritma düzlem LiDAR nokta bulutu verilerinde otomatik gürültü noktalarının tespitini beklenen başarı ile gerçekleştirememiştir. Test alanı #3 incelendiğinde ise, bu alan için tek sabit parametre değeri kullanılmıştır. Bu parametre değeri, bu alanda bulunan az sayıda bina düzlemi üzerine uygulandığı için, elde edilen sonuçlar nokta yoğunluğundan etkilenmemiş ve düzlemler oldukça başarılı bir şekilde çıkarılmıştır.
- "MATLAB" ortamında bina çatı düzlem sınırlarının çıkarılması ve sadeleştirilmesi işleminde, her bina için, tespit edilen her ayrı çatı düzlemini oluşturan noktalar ile çatı düzlemlerinin modellenmesi amaçlanmıştır. Test alanı #1 ve test alanı #2'de "boundary" fonksiyonu ve Douglas-Peucker algoritması uygulanarak elde edilen sonuçlar incelendiklerinde, test alanı #1'de elde edilen sonuçların test alanı #2'de elde edilen sonuçlara göre daha başarılı oldukları görülmüştür. Test alanı #3'te, diğer test alanlarına göre daha başarılı sonuçların elde edilmesinin sebebi, DBSCAN algoritmasının aynı parametre değerlerinin daha az sayıda bina üzerine uygulanmasından dolayı, gürültülerin daha başarılı bir şekilde tespit edilmesi ve RANSAC algoritmasının bu test alanında daha başarılı sonuçlar vermesidir.
- Farklı yükseklikte binaların olduğu bir bölgede, bina çatı düzlemlerinin modellemesi işleminin başarılı bir şekilde sonuç vermesi için, bina sınıflarının tespit edilmesi aşamasında "*ground offset*" parametresinin dikkatli bir şekilde belirlenmesi gerekmektedir. Çünkü bu parametre, test alanındaki en alçak binanın yüksekliği göz önüne alınarak belirlenmektedir.

- Bu çalışmada, RANSAC algoritması ile temel büyük çatı düzlemlerinin tamamı doğru tespit edilmiştir. Ancak, bütün test alanlarında aynı parametre değerlerinin kullanılmasının, düzgün geometrik yapıya sahip olmayan test alanlarında bulunan binalarda başarılı sonuçlar veremeyeceği düşünülmektedir.

KAYNAKLAR

[1] C. Brenner, "Building Extraction," in *Airborne and Terrestrial Laser Scanning*., G. Vosselman and H. Maas, Eds. **2010**.

[2] G. Vosselman and S. Dijkman, "3d Building Model Reconstruction From Point Clouds And Ground Plans," *International Archives of Photogrammetry and Remote Sensing* , " Volume XXXIV-3 / W4 Annapolis , MD , 22-24 Oct . **2001**.

[3] F. Biljecki, J. Stoter, H. Ledoux, S. Zlatanova, and A. Çöltekin, "Applications of 3D City Models: State of the Art Review,"*ISPRS International Journal of Geo-Information*, vol. 4, no. 4, pp. 2842–2889, **2015**.

[4] M. Morgan and K. Tempfli, "Automatic building extraction from airborne laser scanning data," *International Archives of Photogrammetry and Remote Sensing*, Amsterdam vol. XXXIII, no. B3, pp. 616–623, **2000**.

[5] S. Wei, "Building Boundary Extraction Base on Lidar Point Clouds Data," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Beijing, Vol. XXXVII. Part B3b, pp. 157–161, **2008**

[6] N. Demir, "Lidar Verisinden Çati Düzlemlerinin Otomatik Çikarilmasi," *6. Uzaktan Algilama-Cbs Sempozyumu (Uzal-Cbs 2016)*, 5-7 Ekim, Adana, **2016**.

[7] H. Heidemann, J. Stoker, D. Brown, M. Olsen, R. Sing, K. Williams, A. Chin, A. Karlin, G. Mcclung, J. Janke, J. Shan, K. Kim, A. Sampath, S. Ural, C. Parrish, K. Waters, J. Wozencraft, C. Macon, J. Brock, C. Wright, C. Hopkinson, A. Pietroniro, I. Madin and J. Conner, "Building Extraction and Reconstruction from Lidar Point Clouds," in *Manual of Airborne Topographic Lidar*, M. Renslow, Editor, **2012**.

[8] J. Shan and A. Sampath, "Building Extraction from LiDAR Point Clouds Based on Clustering Techniques,"in *Topographic Laser Ranging and Scanning: Principles and Processing*, J. Shan and C. Toth ,Eds. **2008**.

[9] P. Dorninger and C. Nothegger, "3D Segmentation of Unstructured Point Clouds for Building Modelling," *WG I/2, III/2, III/4, III/5, IV/3 PIA07 Photogrammetric Image Anaysis*, 19-21September, Munich, Germany, vol. 36, pp. 191–196, **2007**.

[10] F. Rottensteiner and C. Briese, "Automatic Generation of High-Quality Building Models from Lidar Data," *IEEE Computer Graphics and Applications*, **2003**.

[11] F. Rottensteiner and C. Briese, "Automatıc Generatıon Of Buıldıng Models From Lidar Data And The Integratıon Of Aerıal Images," *Isprs*, Dresden, vol. XXXIV, **2003**.

[12] A. Alharthy and J. Bethel, "Detailed Building Reconstruction From Airborne Laser Data Using A Moving Surface Method," **2004**.

[13] F. Tarsha-Kurdi, T. Landes, and P. Grussenmeyer, "Hough-Transform and Extended Ransac Algorithms for Automatic Detection of 3D Building Roof Planes From Lidar Data," *ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007*, Espoo, Finland, 12-14 September, vol. XXXVI, no. 1, pp. 407–412, **2007**.

[14] K. Kim and J. Shan, "Building Roof Modeling From Airborne Laser Scanning Data Based On Level Set Approach," *ISPRS Journal of Photogrammmetry and Remote*

Sensing, vol. 66, no. 4, pp. 484–497, **2011**.

[15] F. Karsli and N. Pfeifer, "Ransac Algortiması İle Lidar Verilerinden Otomatik Detay Çıkarımı," *IV. Uzaktan Algılama ve Coğrafi Bilgi Sistemleri Sempozyumu (UZAL-CBS 2012)*, 16-19 Ekim , Zonguldak, **2012**.

[16] R. Çömert and U. Avdan, "Yersel Lazer Tarayıcı Verılerınden Basıt Geometrık Yüzeylerın Otomatık Olarak Çıkarılması," *5. Uzaktan Algılama-Cbs Sempozyumu (UZAL-CBS 2014)*, 14-17 Ekim, İstanbul, **2014**.

[17] S. Canaz, F. Karsli, A. Guneroglu, M. Dihkan, R. Sensing, and L. Detection, "Lidar Verileri Kullanılarak Göl Sınırlarının Otomatik Olarak Belirlenmesi,"*TUFUAB III. Teknik Sempozyumu*, 21-23 Mayıs, Konya, pp. 287–293, **2015**.

[18] Y. Fan, M. Wang, N. Geng, D. He, J. Chang, and J. J. Zhang, "A self-adaptive segmentation method for a point cloud," *Visual Computer*, vol. 34, no. 5, pp. 659–673, **2018**.

[19] A. Kayi, M. Erdoğan, and O. Eker, "OPTECH HA-500 ve RIEGL LMS- Q1560 ile Gerçekleştirilen LİDAR Test Sonuçları," *Harita Dergisi.*, vol. 153, pp. 42–46, **2015**.

[20] "LAS Specification Version 1.4 – R13." https://www.asprs.org/wp-content/uploads/2010/12/LAS_1_4_r13.pdf. (Haziran, **2018**)

[21] "Storing Lidar Data." http://desktop.arcgis.com/en/arcmap/10.3/manage-data/las-dataset/storing-lidar-data.htm. (Haziran, **2018**)].

[22] F. Yıldız, "Büyük Ölçekli Harita Üretiminde LiDAR ve Ortofoto." http://studylibtr.com/doc/1372830/büyük-ölçekli-harita-üretiminde-lidar-ve-ortofoto. (Haziran, **2018**)

[23] "LAStools Filtreleme Algoritması." http://lbi-archpro.org/als-filtering/lbi-project/results/lastools/filtering-algorithm-2. (Haziran, **2018**)

[24] P. Axelsson, "DEM Generation From Laser Scanner Data Using Adaptive TIN Models," *International Archives of Photogrammetry and Remote Sensing*, Amsterdam, vol. 23, no. B4, pp. 110–117, **2000**.

[25] X. Meng, N. Currit, and K. Zhao, "Ground Filtering Algorithms For Airborne Lidar Data: A Review Of Critical İssues," *Remote Sensing*, vol. 2, no. 3, pp. 833–860, **2010**.

[26] G. Sithole and G. Vosselman, *ISPRS Comparison Of Filters*, vol. 25, no. 7. **2003**.

[27] "DSM / DTM Filtering." http://home.iitk.ac.in/~blohani/LiDARSchool2008/Downloads/DTM_pfeifer.pdf. (Haziran, **2018**)

[28] "LAStools lasindex." http://www.cs.unc.edu/~isenburg/laszip/download/lasindex_README.txt. (Haziran, **2018**)

[29] "LAStools lasnoise." http://www.cs.unc.edu/~isenburg/laszip/download/lasnoise_README.txt. (Haziran, **2018**)

[30] "LAStools lasoverlap."

http://www.cs.unc.edu/~isenburg/laszip/download/lasoverlap_README.txt. (Haziran, **2018**)

[31] "LAStools: software for rapid LiDAR processing." http://www.cs.unc.edu/~isenburg/lastools/. (Haziran, **2018**)

[32] "LAStools Filtreleme Parametreleri." http://lbi-archpro.org/als-filtering/lbi-project/results/lastools/guidelines-2. (Haziran, **2018**)

[33] "LAStools lasground." https://rapidlasso.com/lastools/lasground/. (Haziran, **2018**)

[34] "LAStools lasclassify." https://rapidlasso.com/lastools/lasclassify/. (Haziran, **2018**)

[35] "LAStools las2las." https://rapidlasso.com/lastools/las2las/. (Haziran, **2018**)

[36] "LAStools lasinfo." https://rapidlasso.com/lastools/lasinfo/. (Haziran, **2018**)

[37] "LAStools las2txt." http://www.cs.unc.edu/~isenburg/laszip/download/las2txt_README.txt. (Haziran, **2018**)

[38] "Bölge Büyüme Bölütleme." http://pointclouds.org/documentation/tutorials/region_growing_segmentation.php. (Haziran, **2018**)

[39] "Yüzey Normallerini Tahmini." http://pointclouds.org/documentation/tutorials/normal_estimation.php. (Haziran, **2018**).

[40] R. B. Rusu, "*Semantic 3d Object Maps For Everyday Manipulation İn Human Living Environments*," Dissertation,Technische Universität München Institut Für Informatik, **2010**.

[41] "3 Boyutlu RANSAC Algoritması." http://www.timzaman.com/2011/03/3d-ransac-planaire-oppervlakken-uit-puntwolken/. (Haziran, **2018**)

[42] T. T. Bilgin, Y. Çamurcu. "DBSCAN, OPTICS ve K-Means Kümeleme Algoritmalarının Uygulamalı Karşılaştırılması," *Journal of Polytechnic*, vol. 8, no. 2, pp. 139–145, **2005**.

[43] H. A. R. Al-Zand, "*Bölümleyci Kümeleme Algoritmalarının Farklı Veri Yoğunluklarında Karşılaştırması*, Yüksek Lisans Tezi, Gazi Üniversitesi Bilişim Enstitüsü, Ankara " **2013**.

[44] "Sınır Fonksiyonu." https://www.mathworks.com/help/matlab/ref/boundary.html. (Haziran, **2018**)

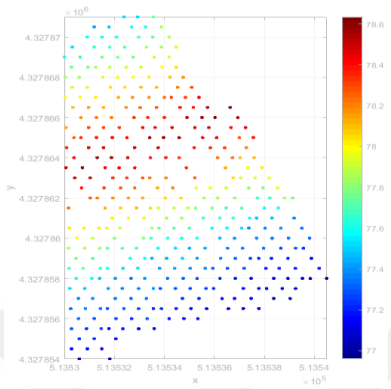
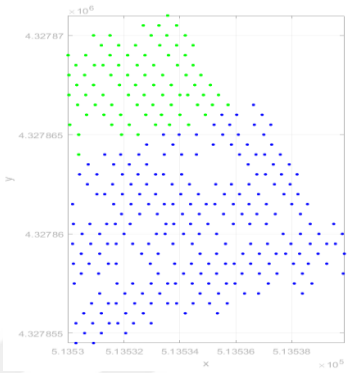
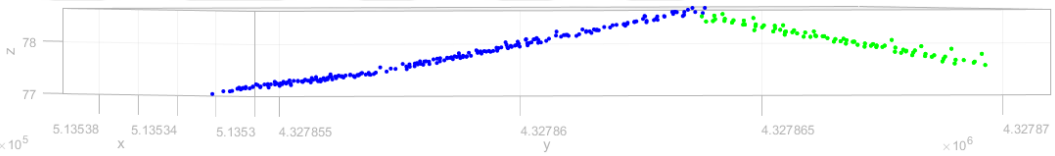
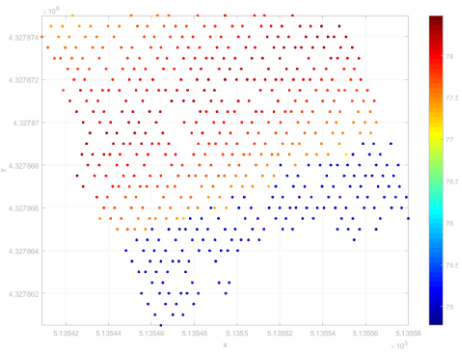
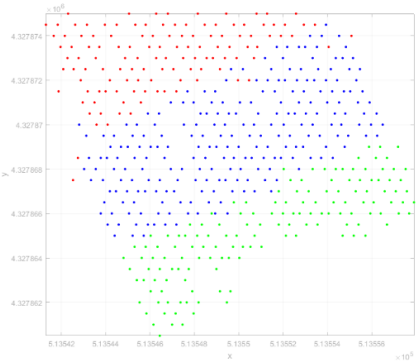
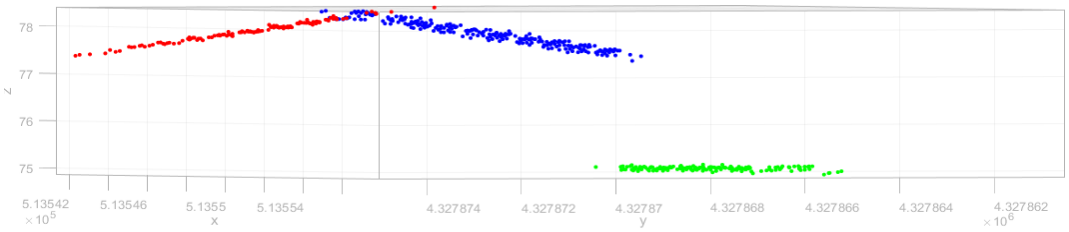
[45] M. Turker and E. H. Kok, "Field-based sub-boundary extraction from remote sensing imagery using perceptual grouping," *Isprs Journal Of Photogrammetry And Remote Sensing*, vol. 79, pp. 106–121, **2013**.

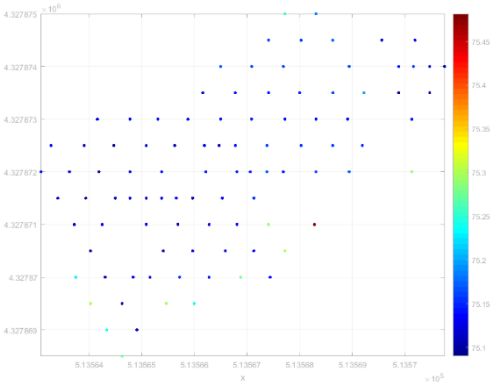
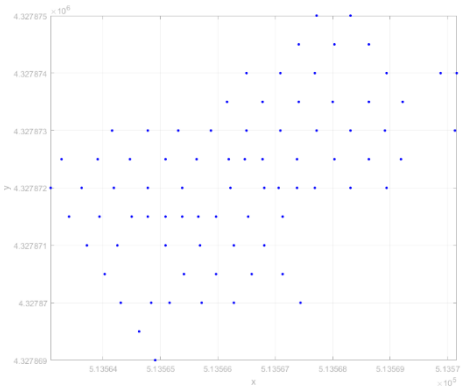
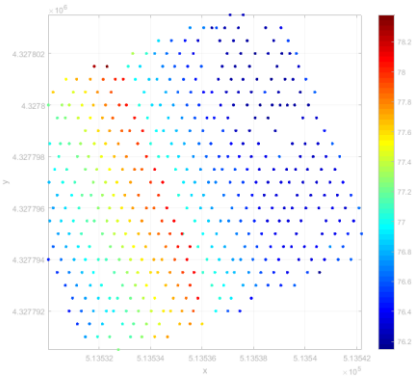
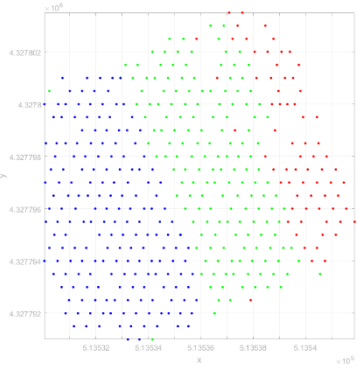
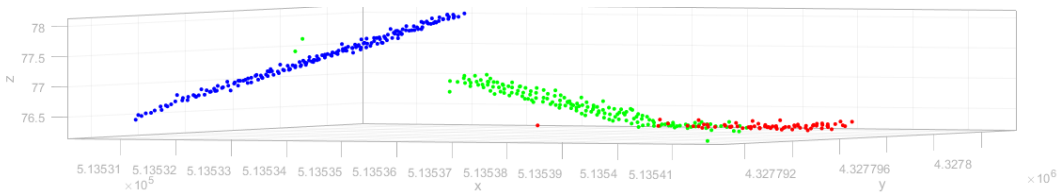
[46] "Random sample consensus (RANSAC)." http://www.math-info.univ paris5.fr/~lomn/Cours/CV/SeqVideo/Material/RANSAC-tutorial.pdf. (Haziran, **2018**)

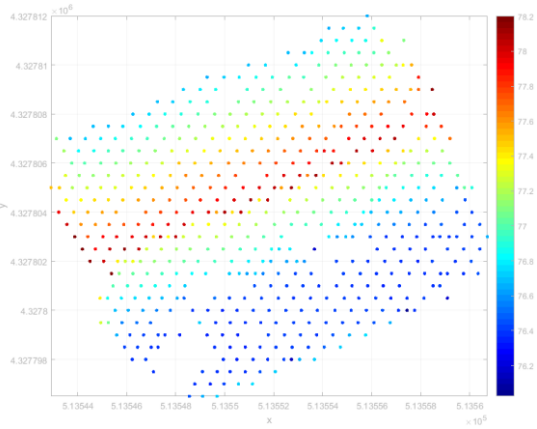
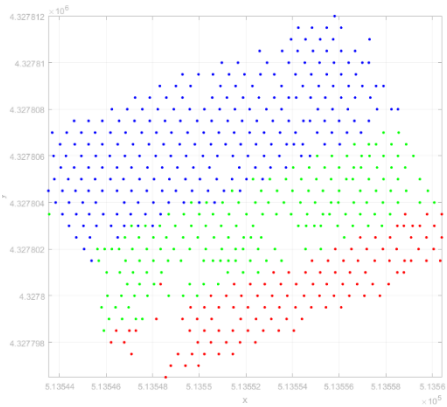
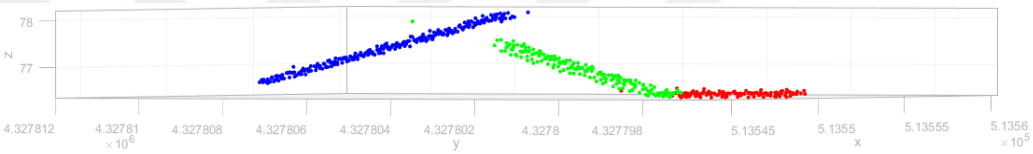
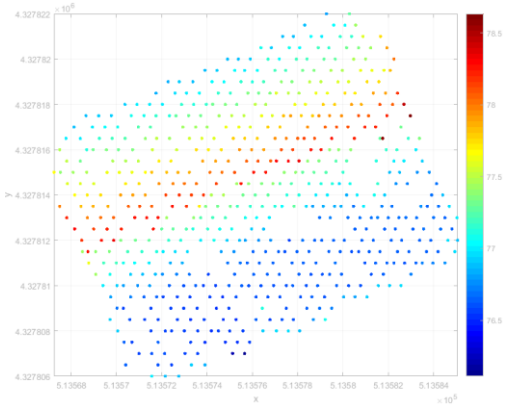
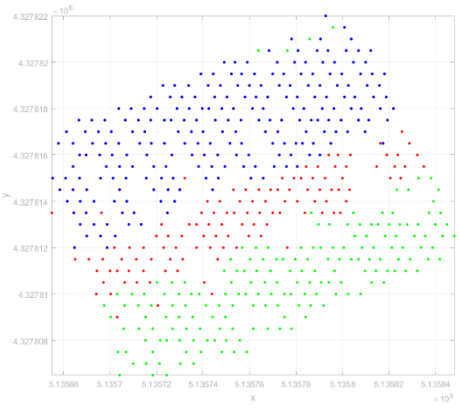
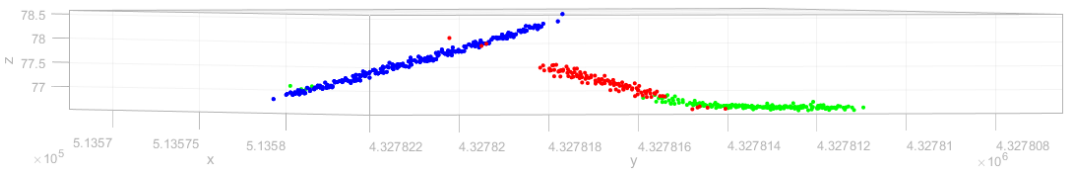
[47] F. Bretar, "Field-based sub-boundary extraction from remote sensing imagery using perceptual grouping," in *Topographıc Laser Rangıng And Scannıng Principles And Processing*, J. Shan and C. K. Toth, Eds. **2008**.

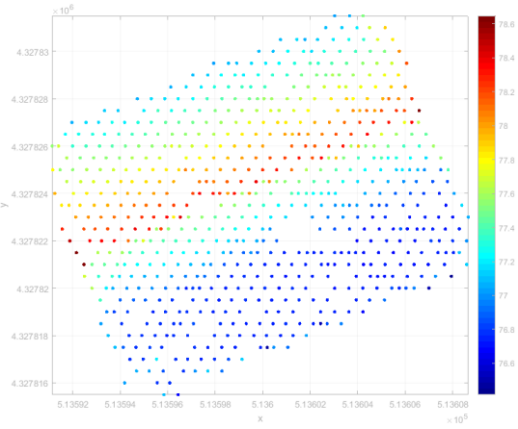
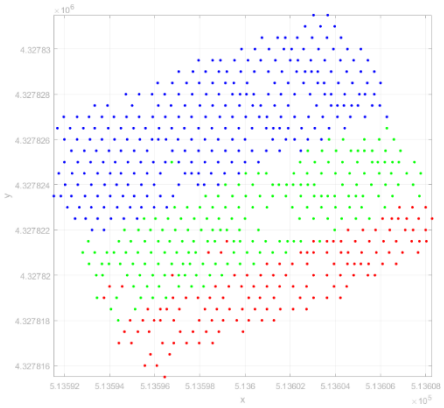
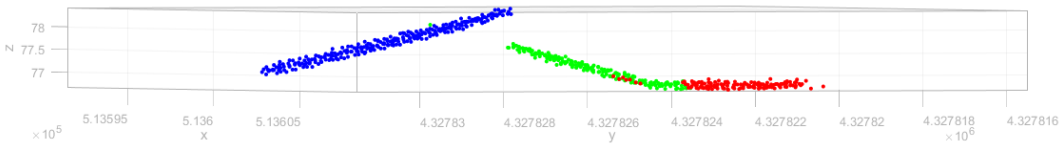
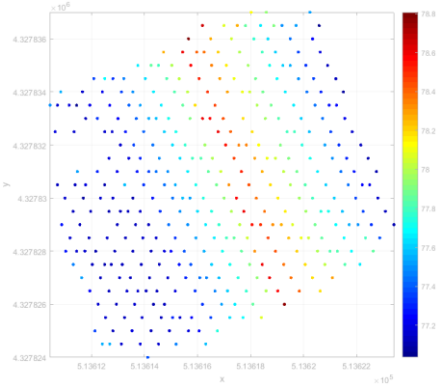
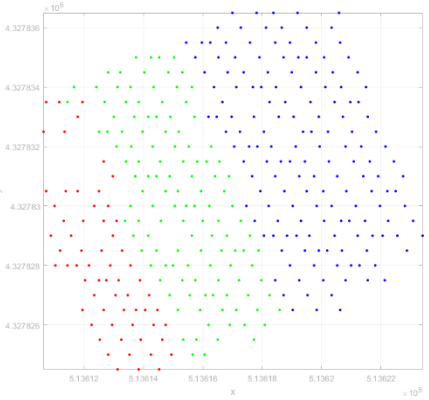
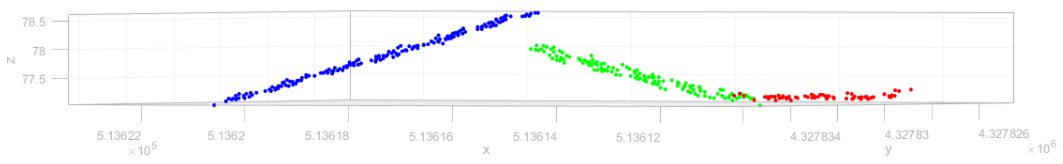
EKLER

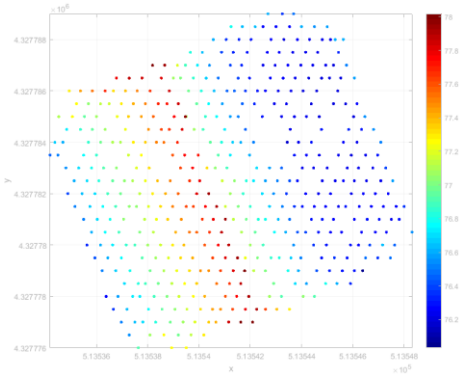
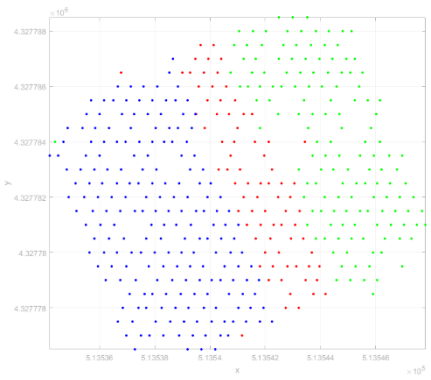
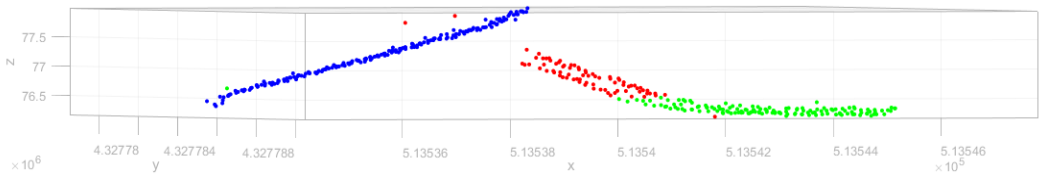
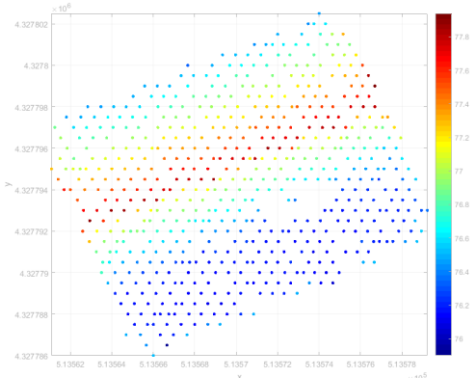
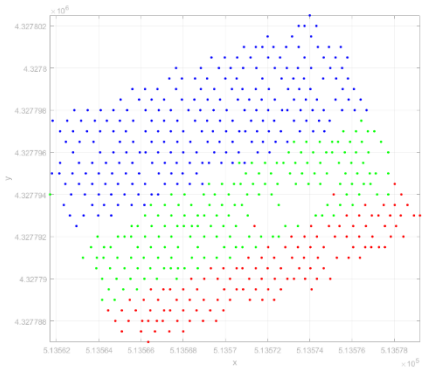
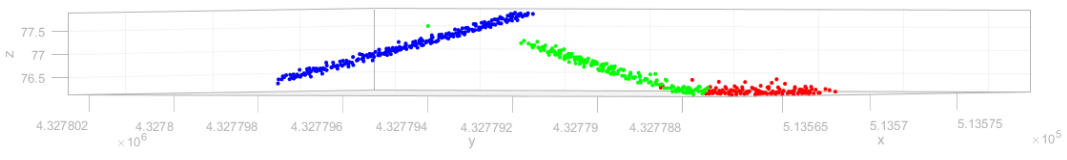
A. Test Alanı #1 Sonuçları

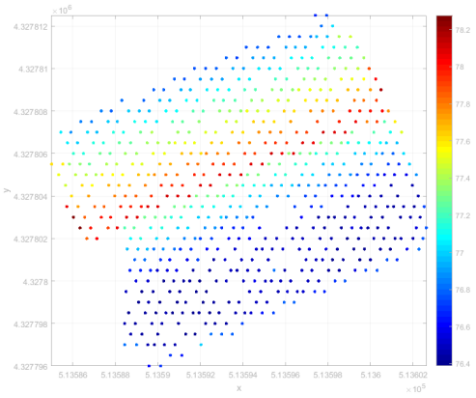
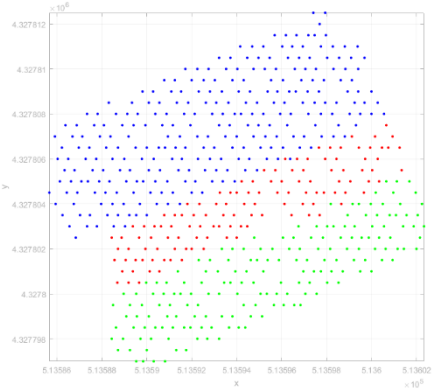
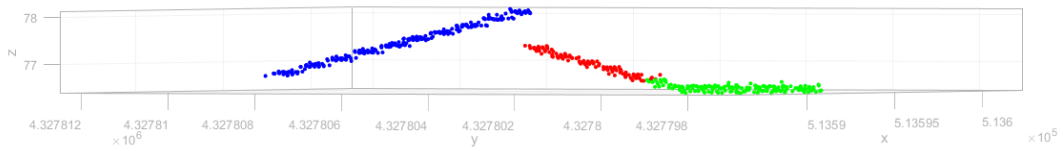
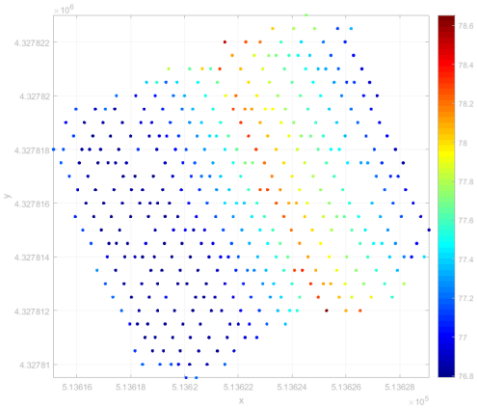
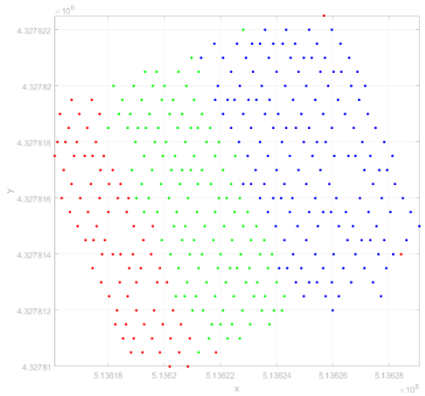
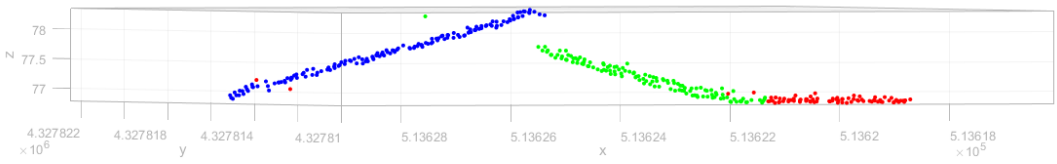
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#1		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#2	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

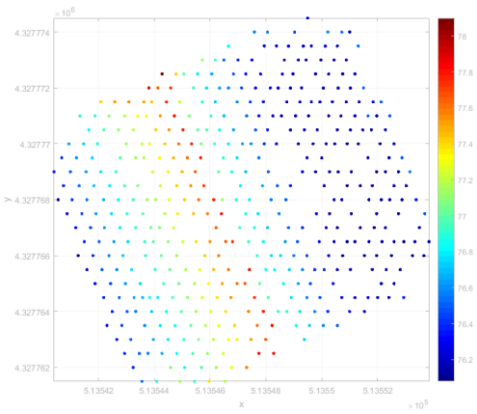
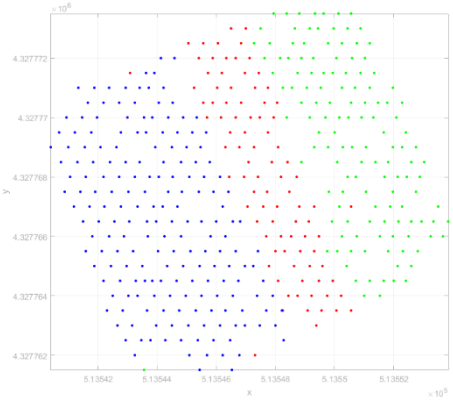
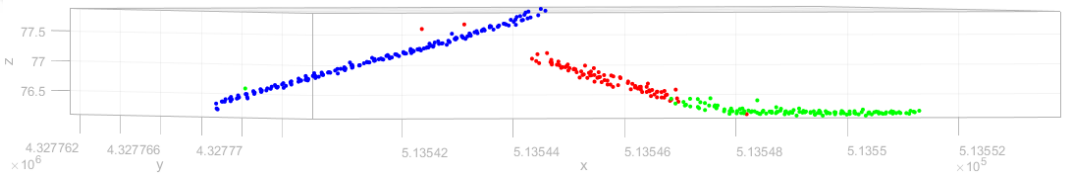
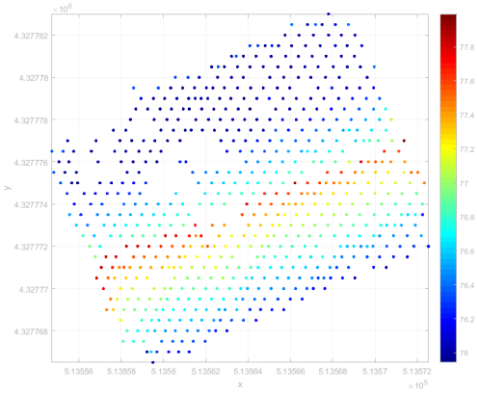
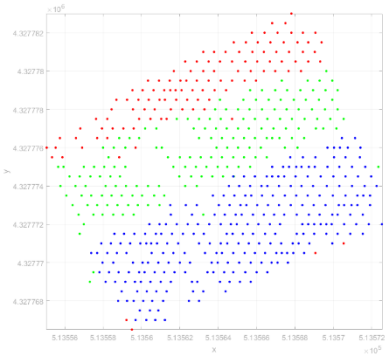
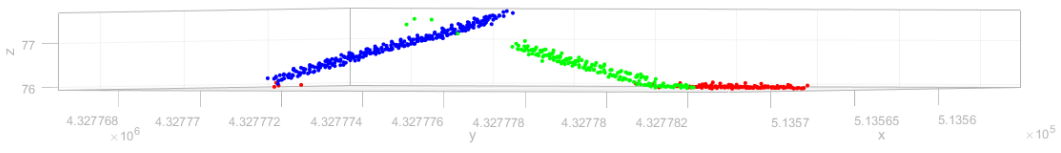
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#3		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#4	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

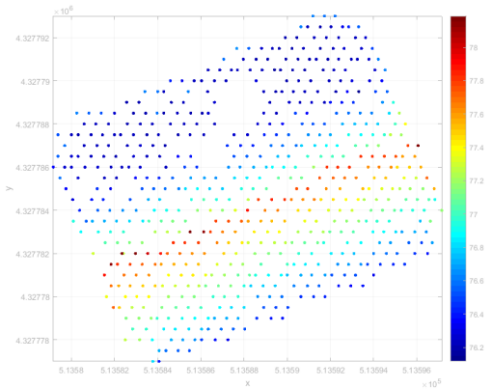
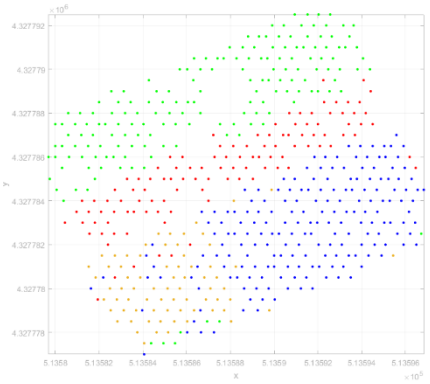
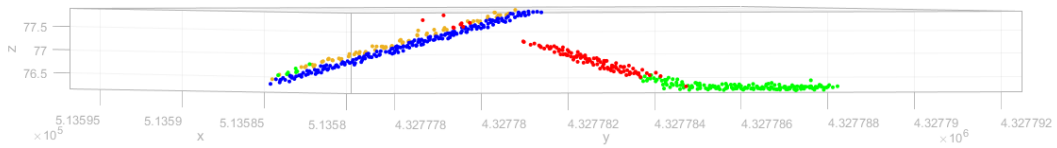
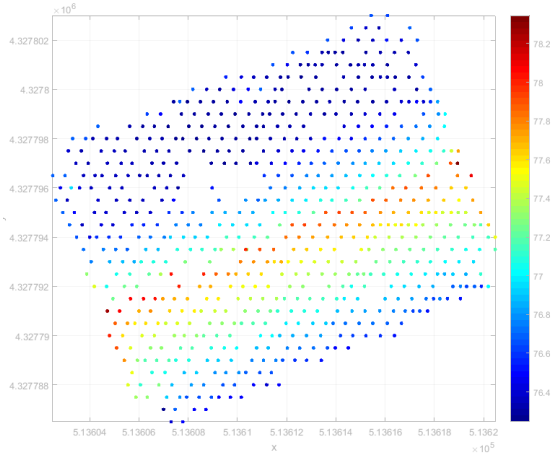
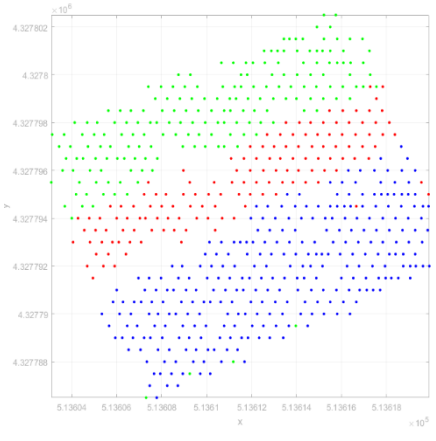
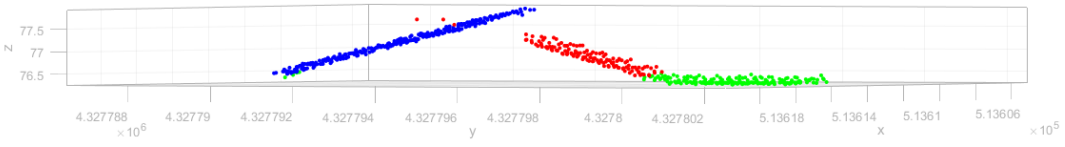
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#5		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#6	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

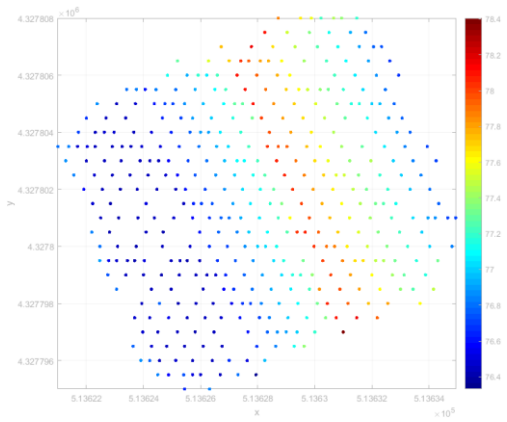
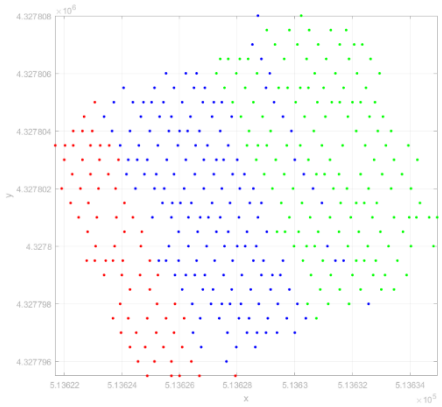
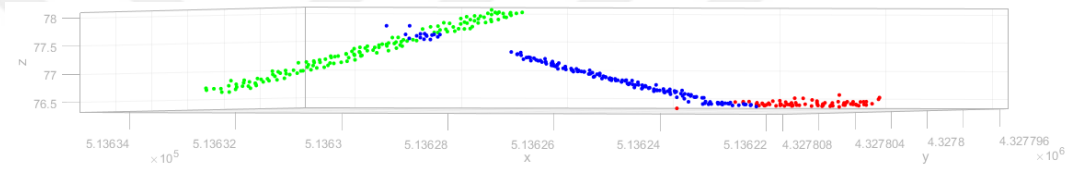
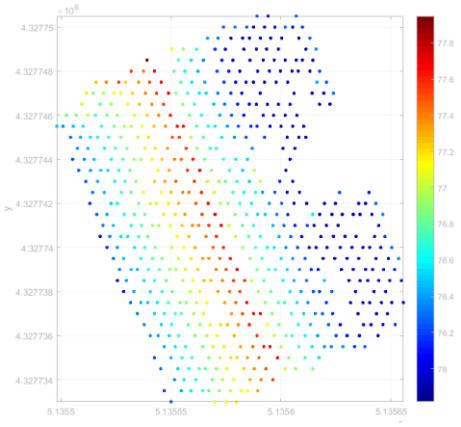
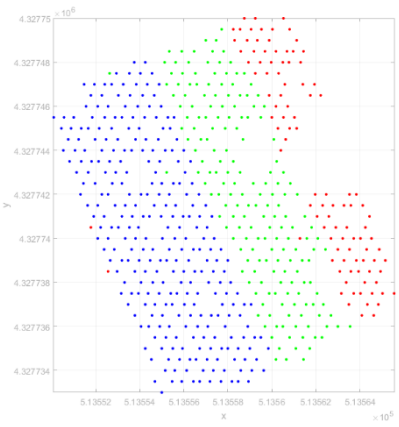
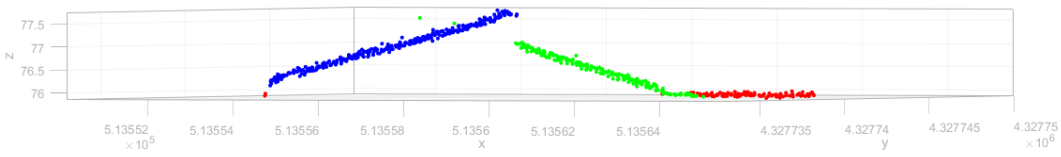
Bina No		
#7	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#8	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

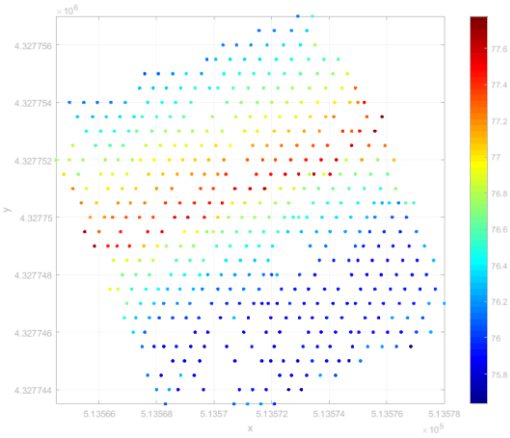
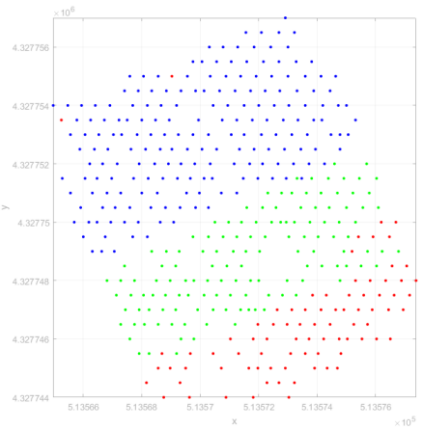
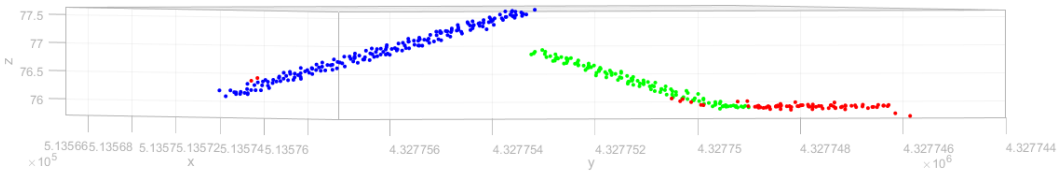
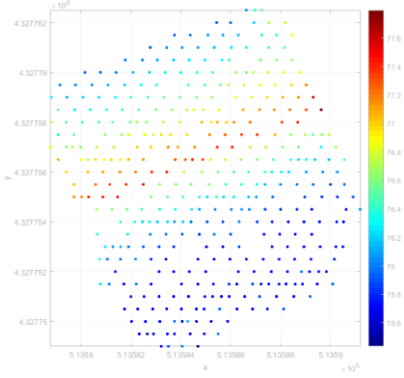
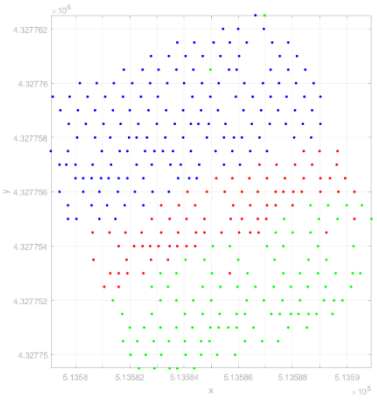
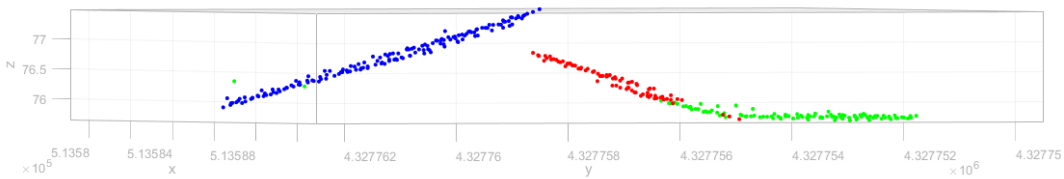
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#9		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#10	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

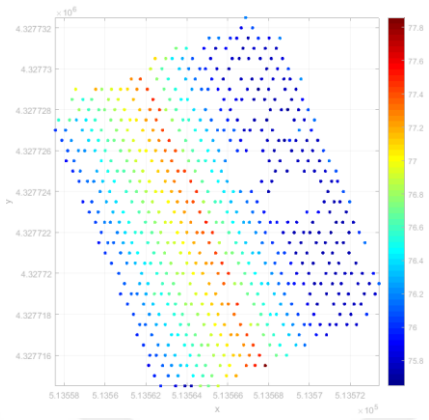
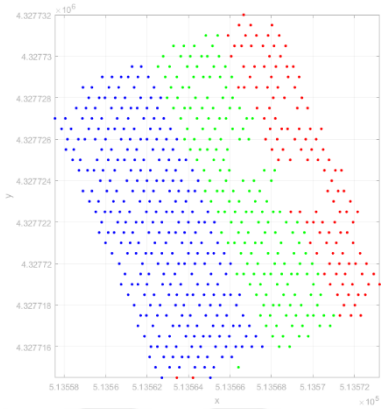
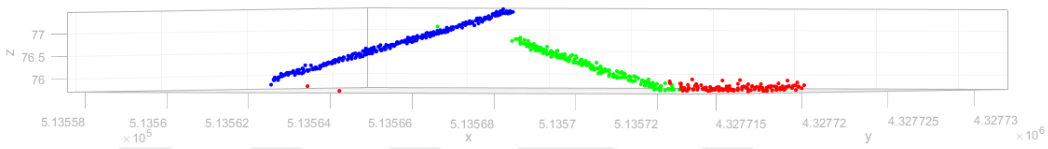
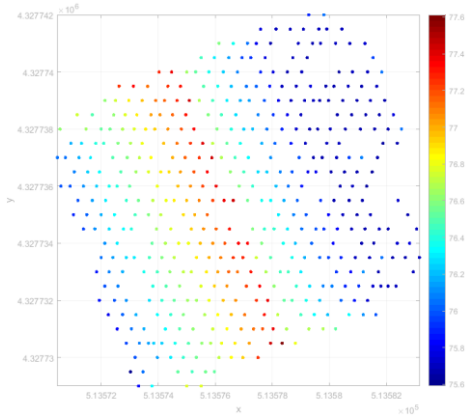
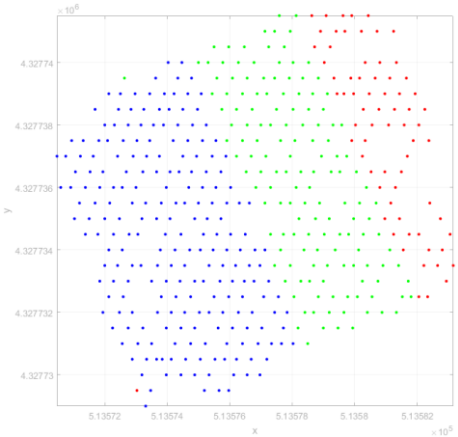
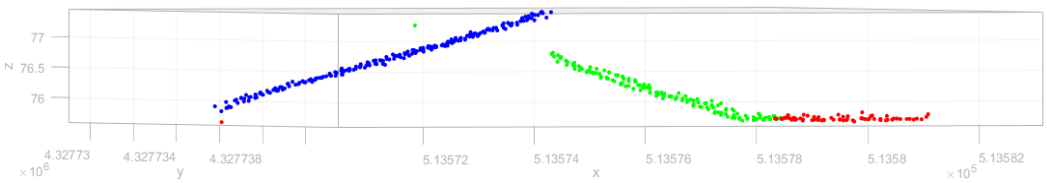
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#11		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#12	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

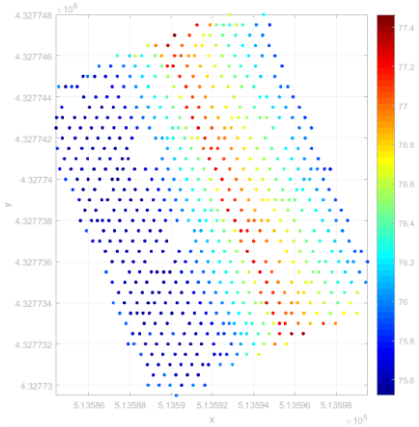
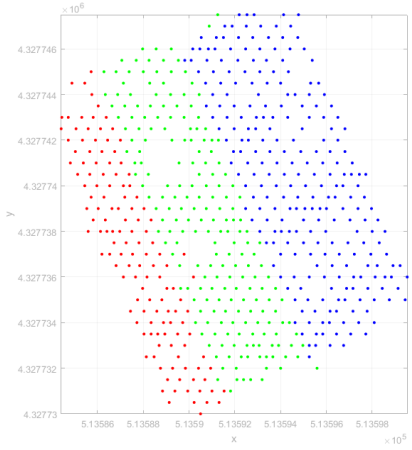
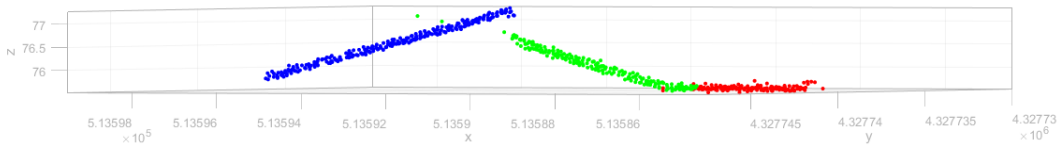
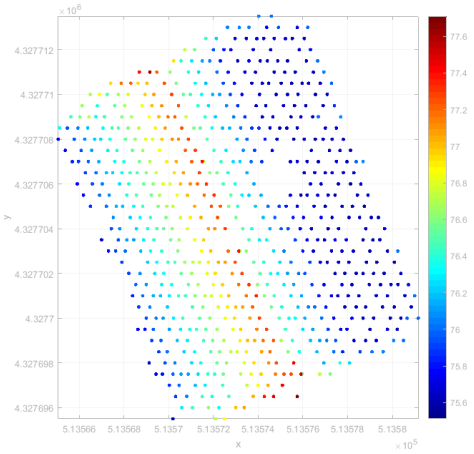
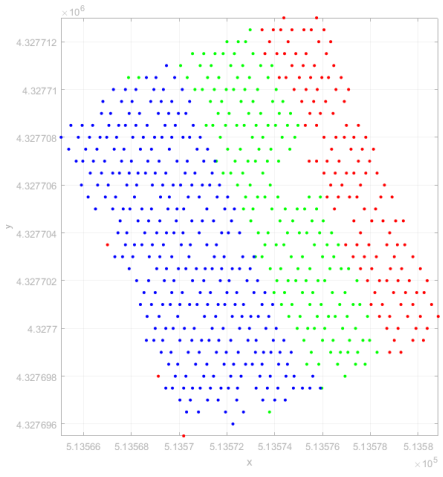
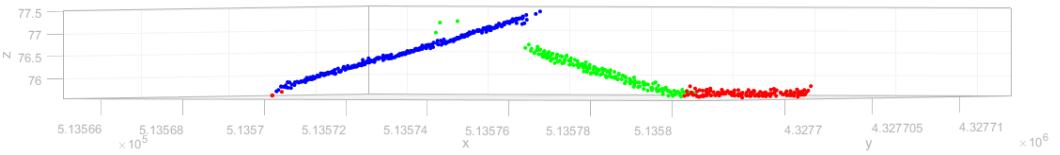
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#13		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#14	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

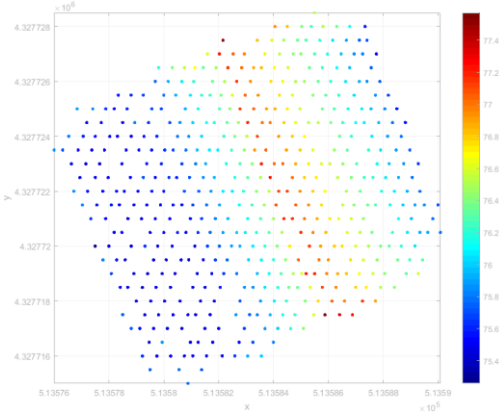
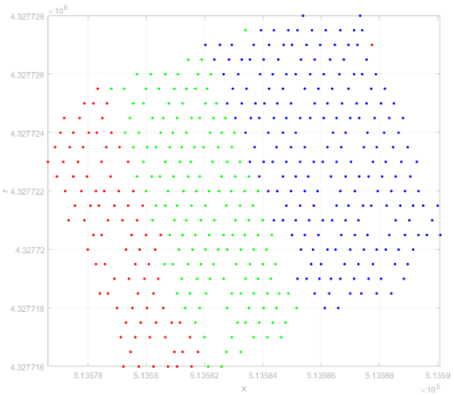
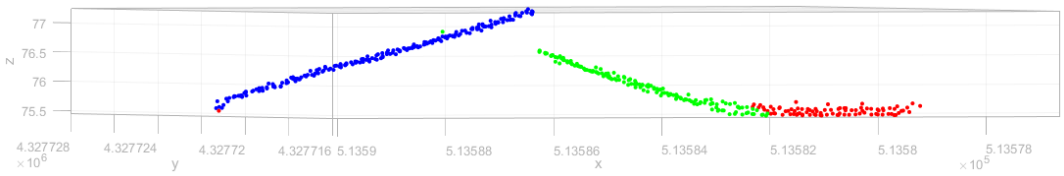
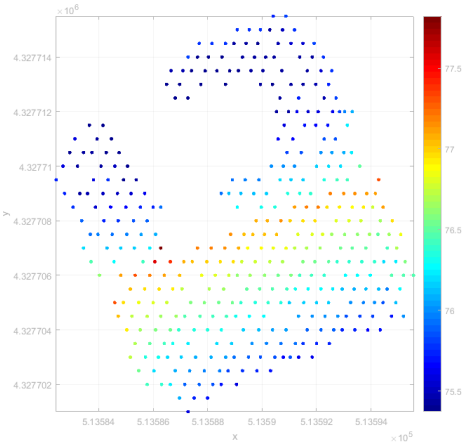
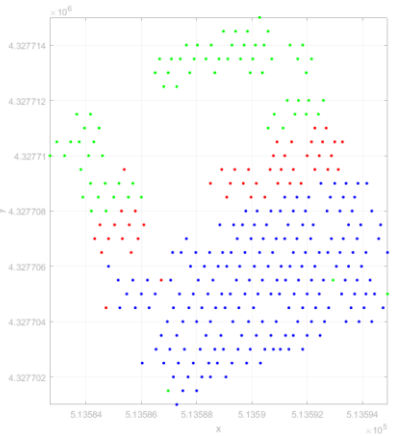
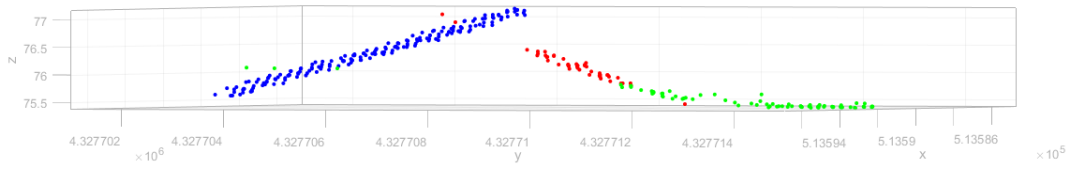
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#15		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#16	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

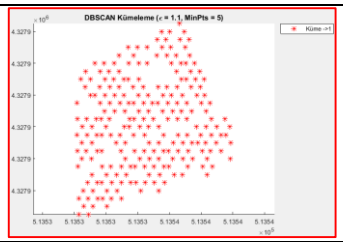
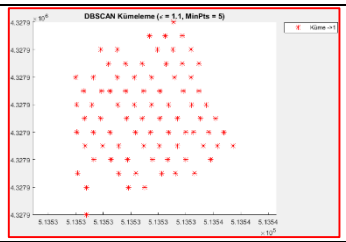
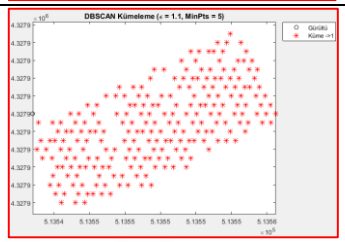
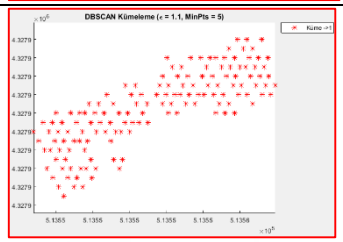
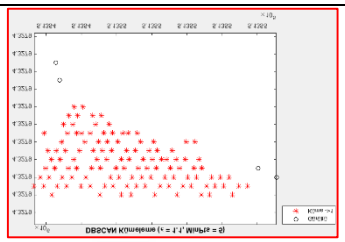
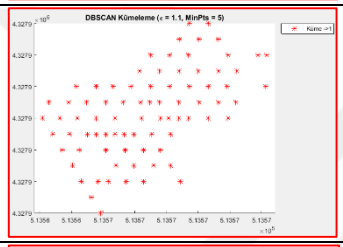
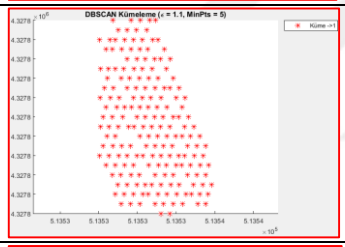
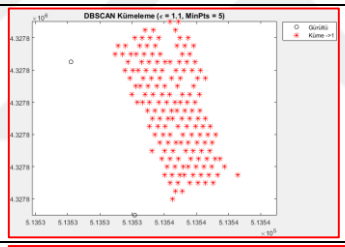
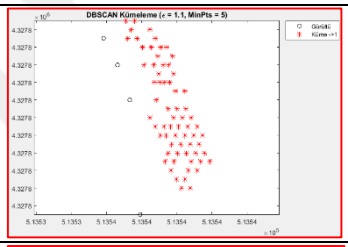
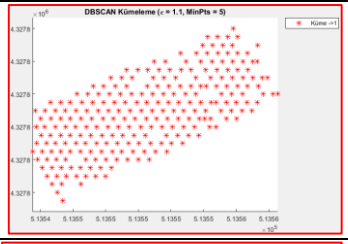
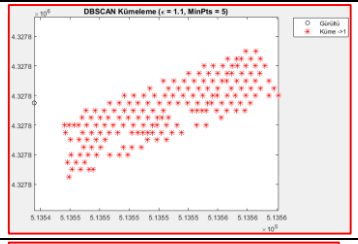
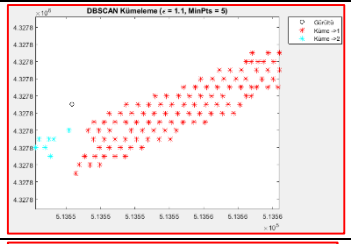
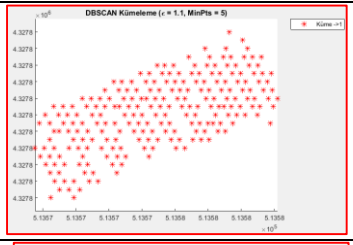
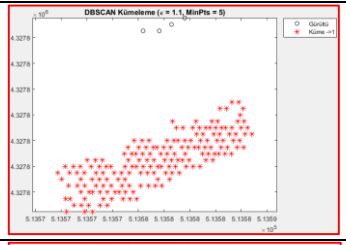
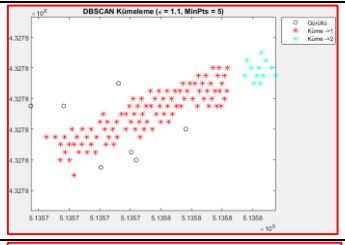
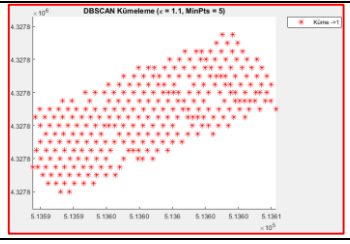
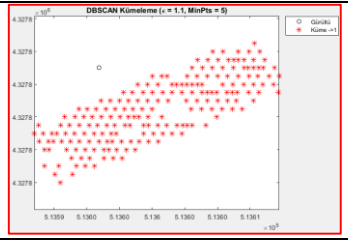
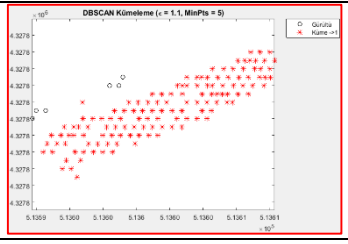
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#17		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#18	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

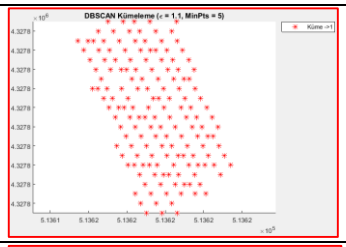
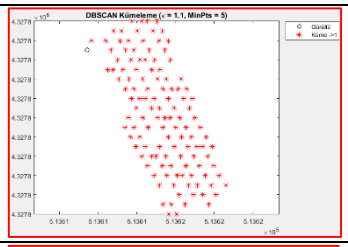
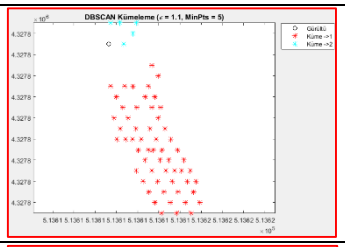
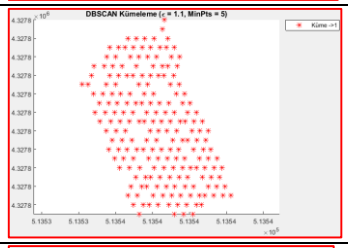
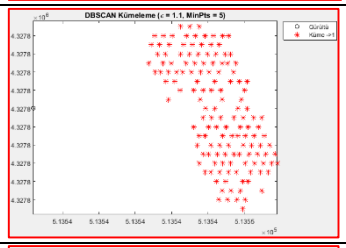
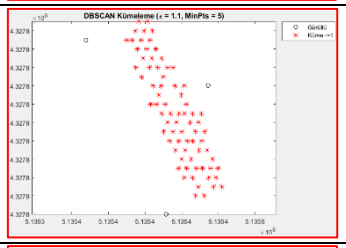
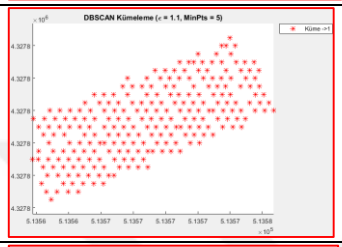
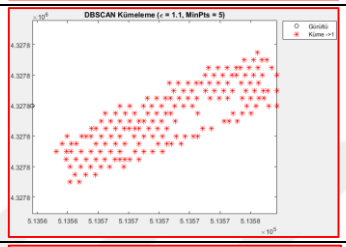
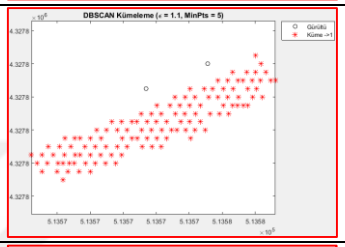
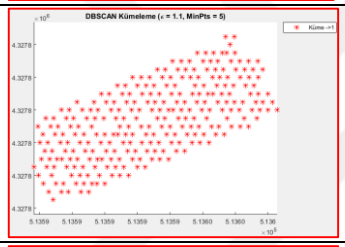
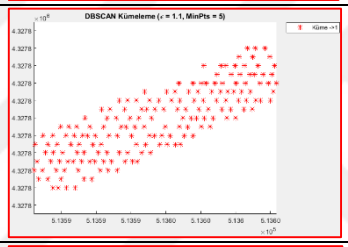
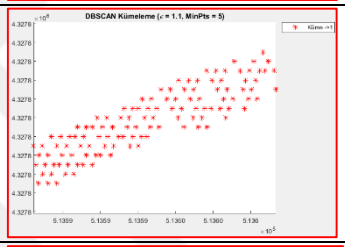
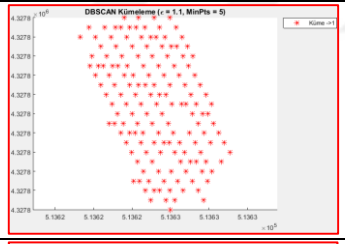
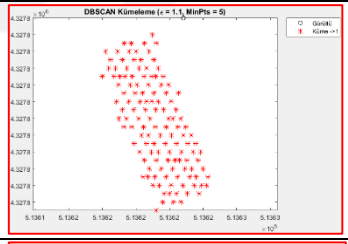
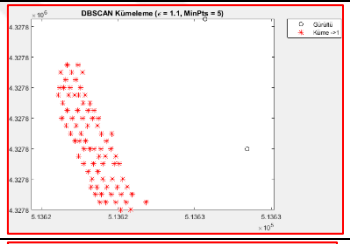
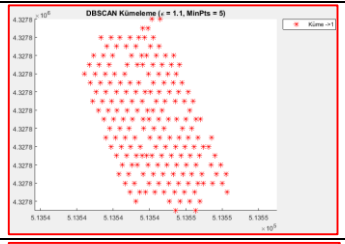
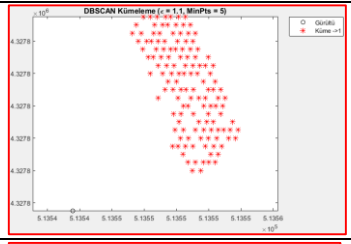
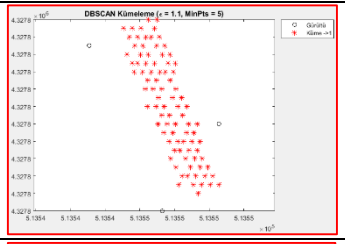
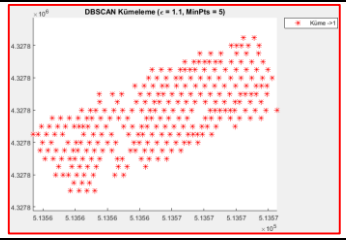
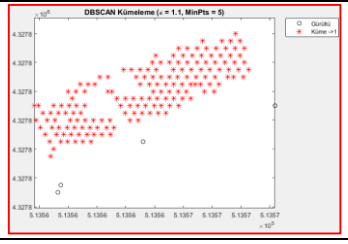
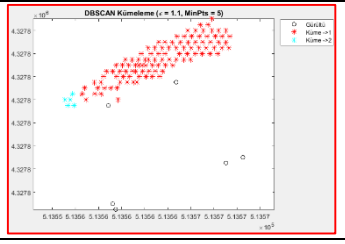
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#19		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#20	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

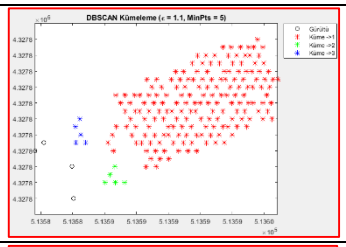
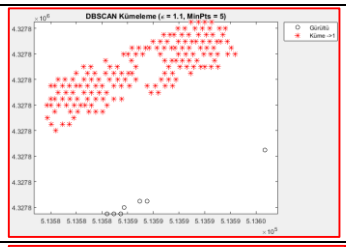
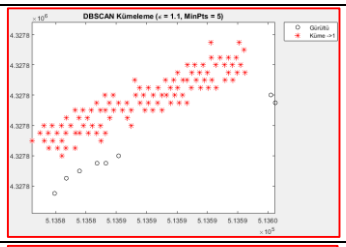
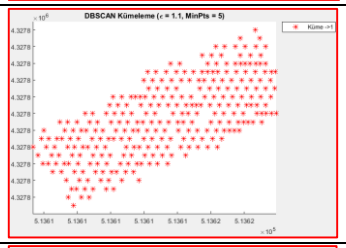
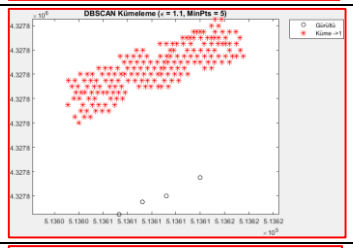
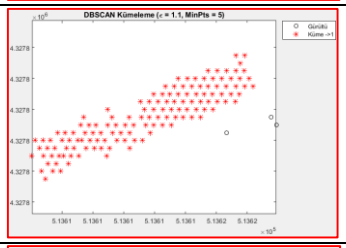
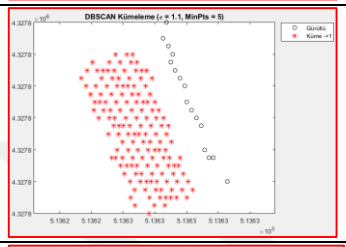
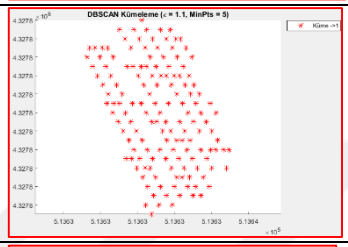
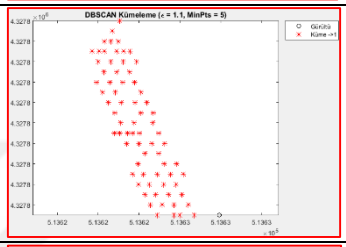
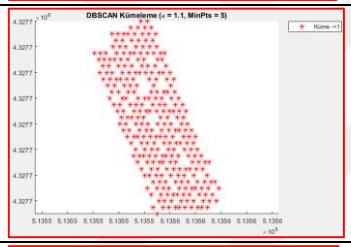
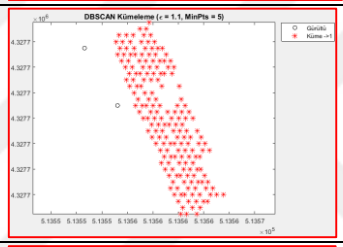
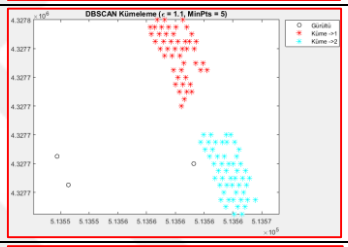
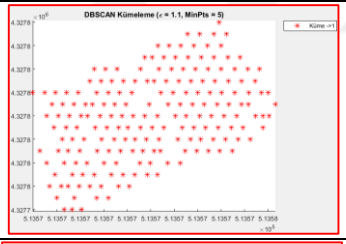
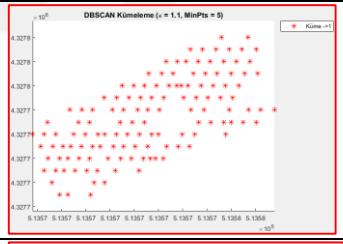
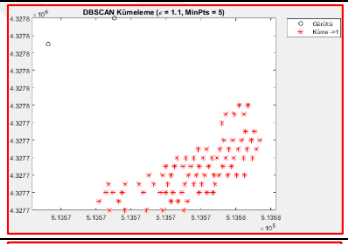
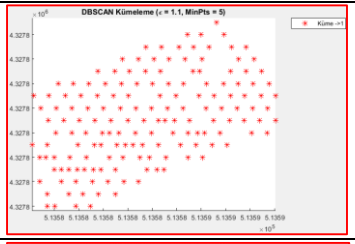
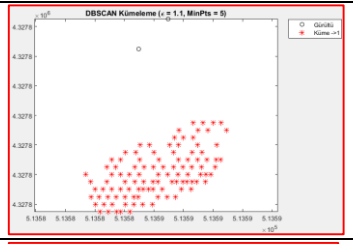
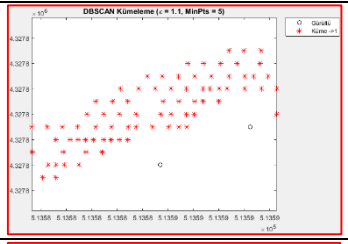
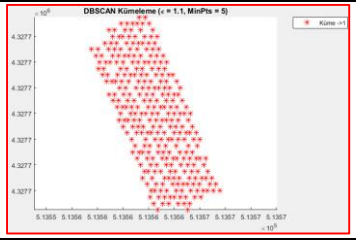
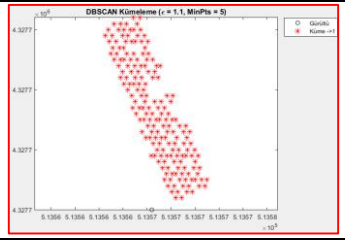
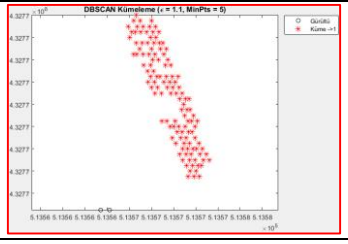
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#21		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#22	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

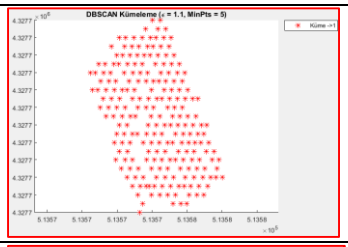
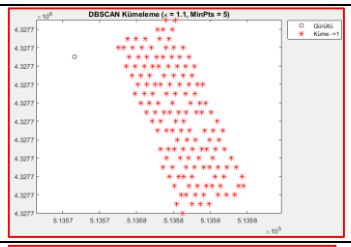
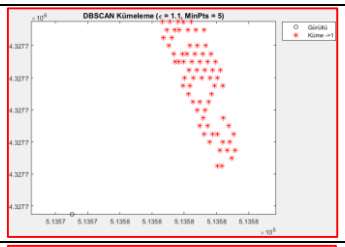
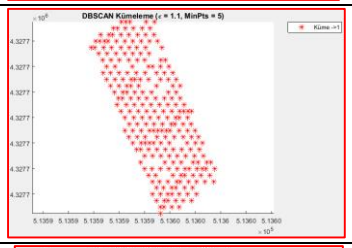
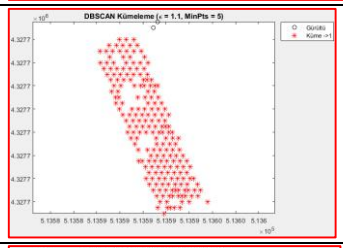
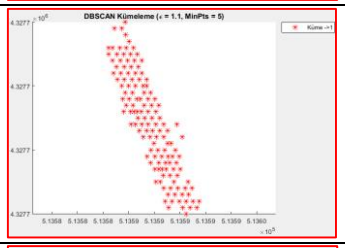
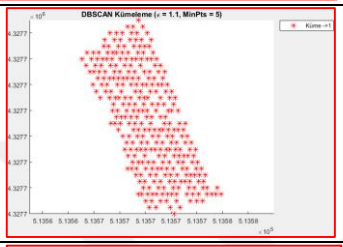
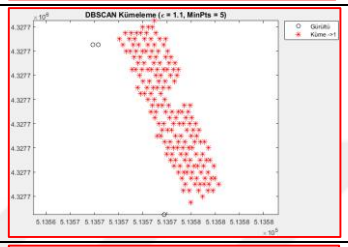
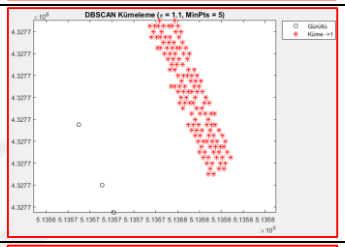
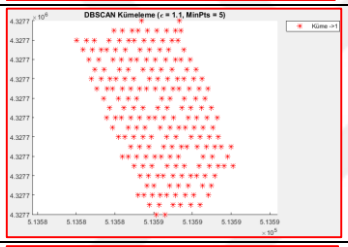
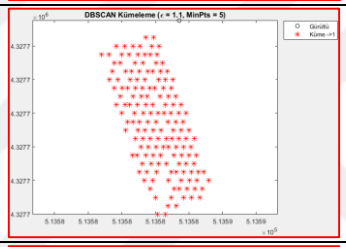
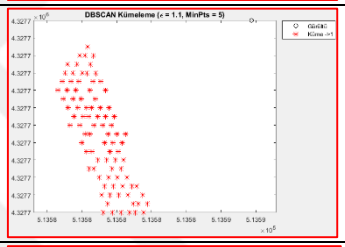
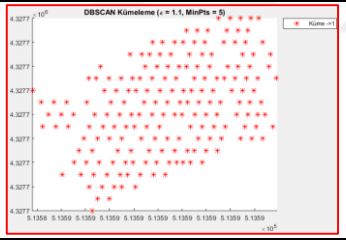
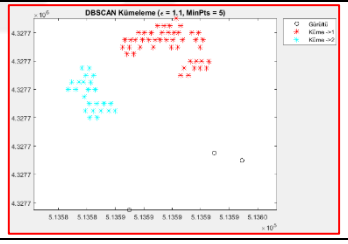
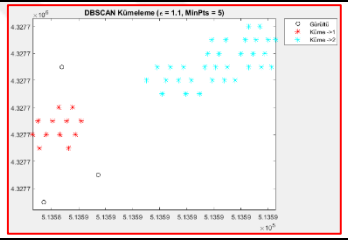
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#23		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#24	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

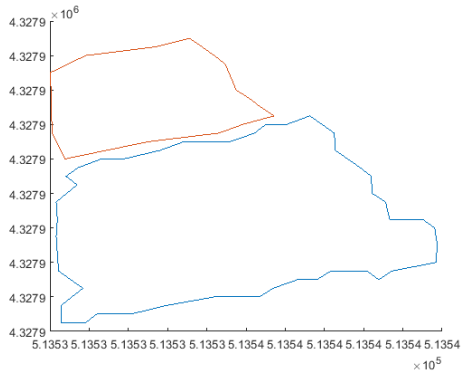
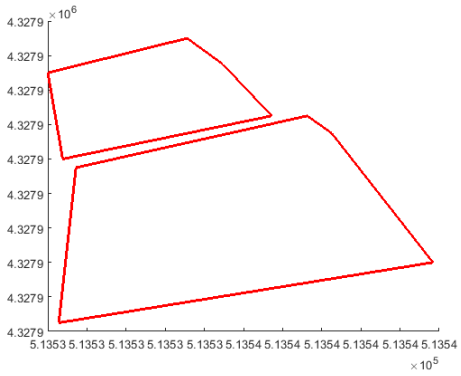
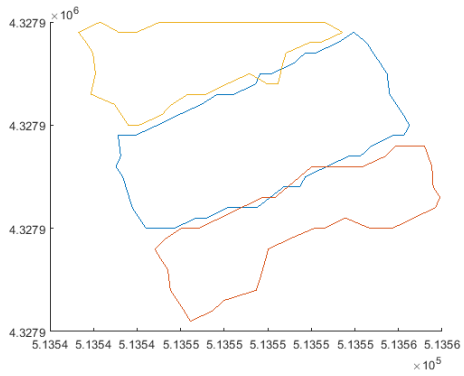
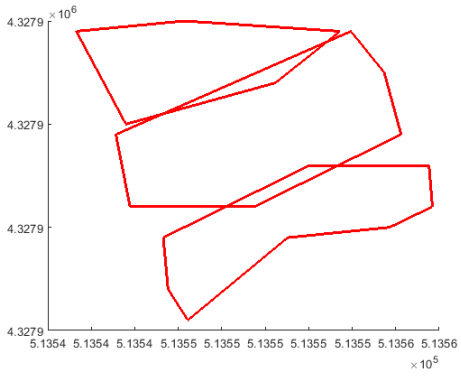
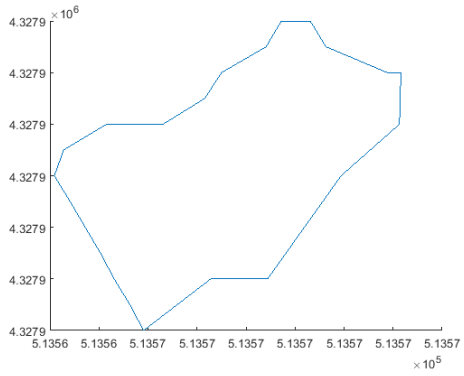
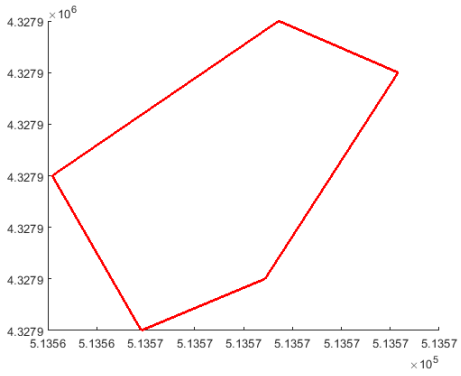
Bina No		
	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#25		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#26	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

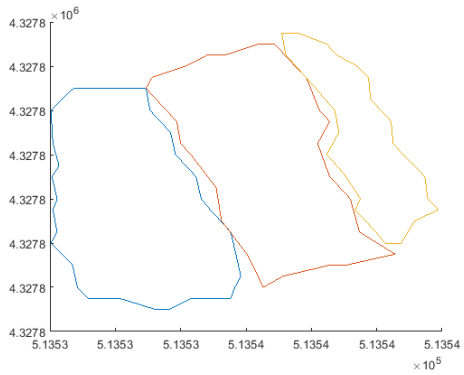
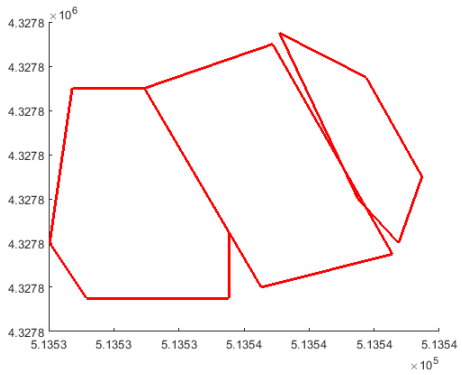
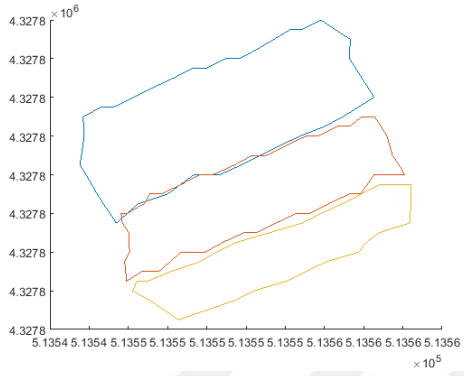
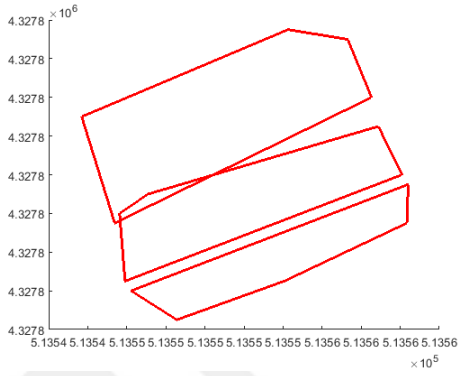
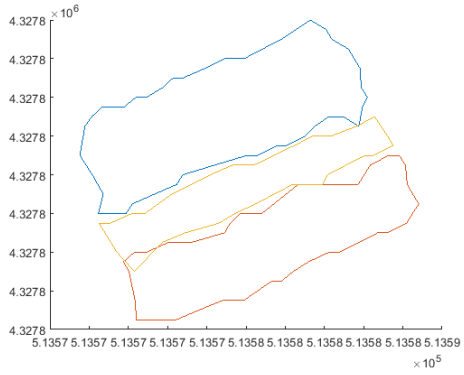
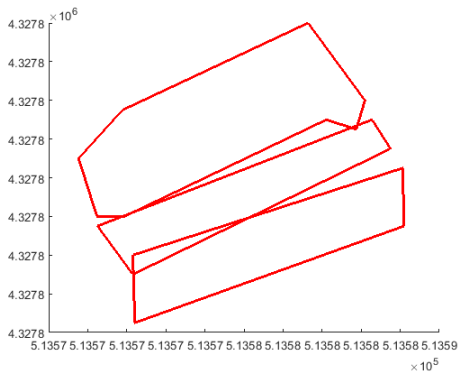
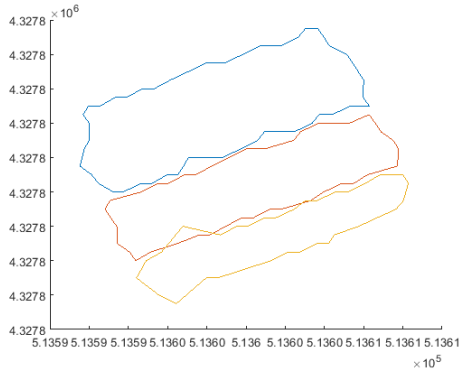
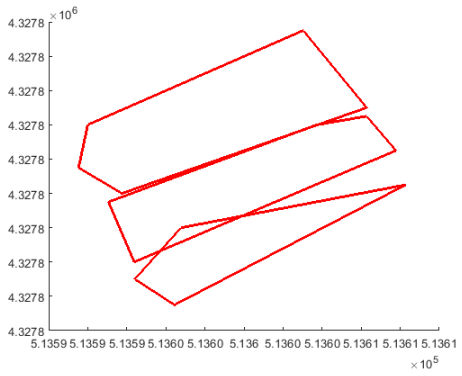
Bina No	DBSCAN Kümeleme		
#1			
#2			
#3			
#4			
#5			
#6			
#7			

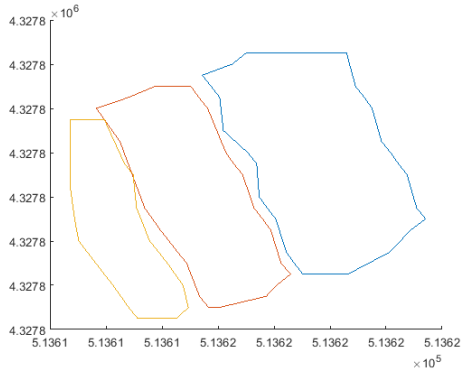
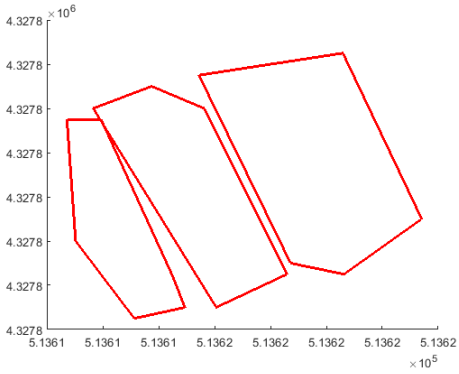
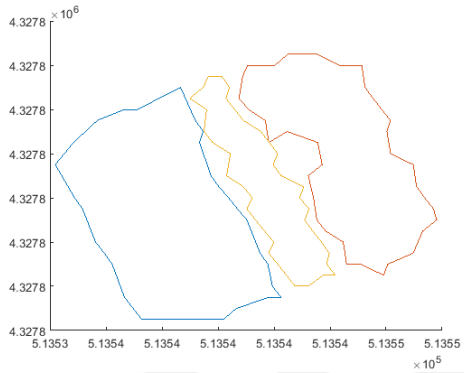
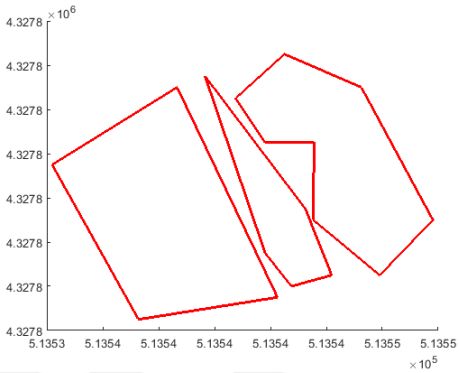
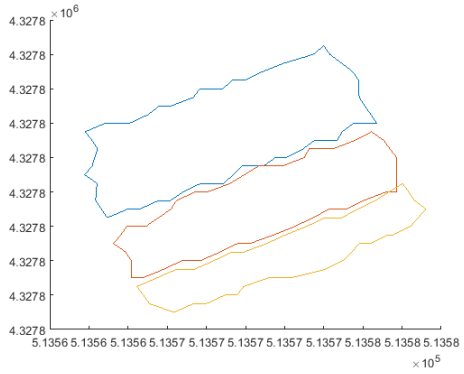
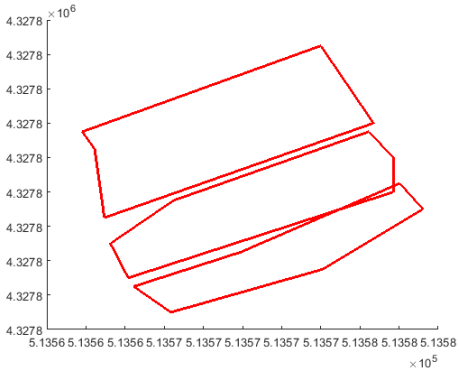
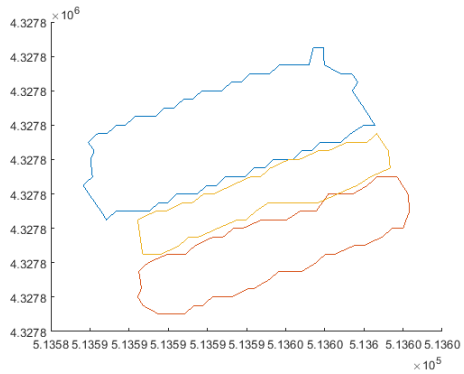
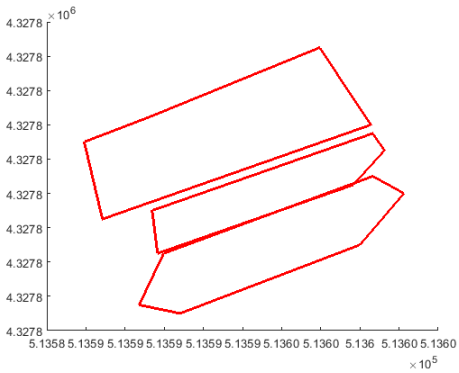
#8			
#9			
#10			
#11			
#12			
#13			
#14			

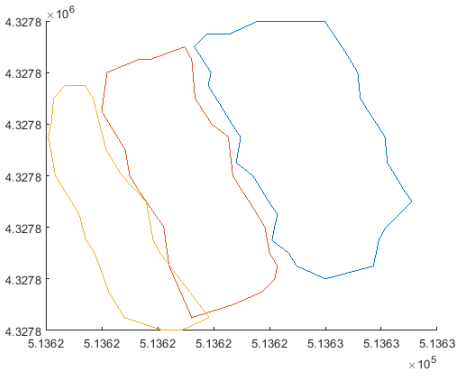
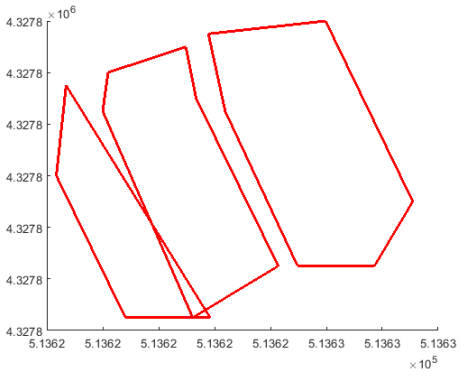
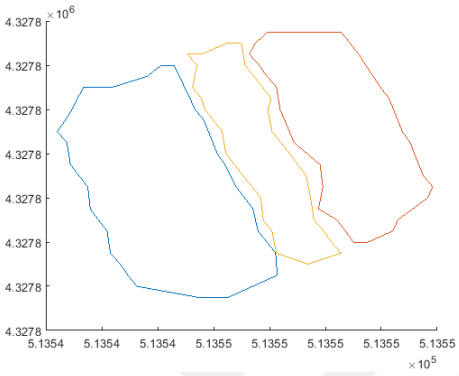
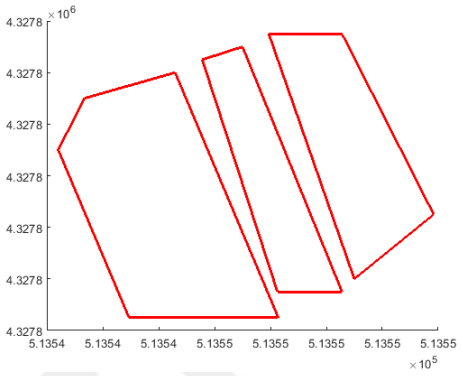
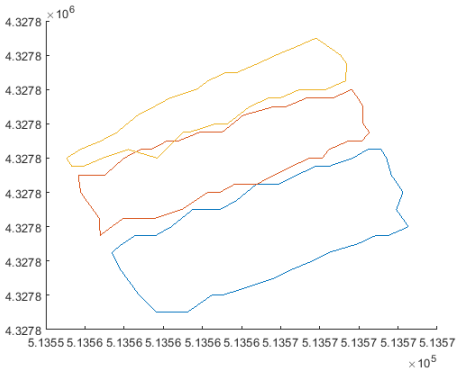
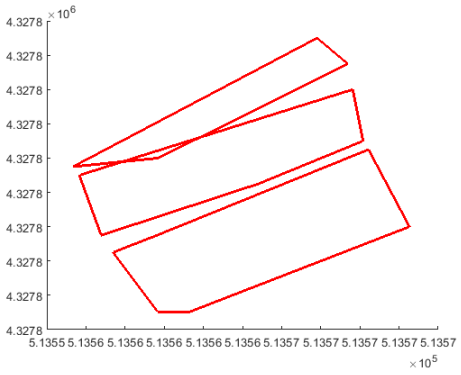
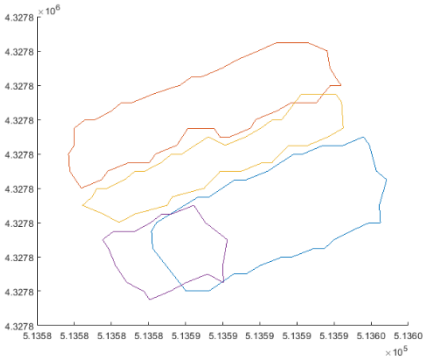
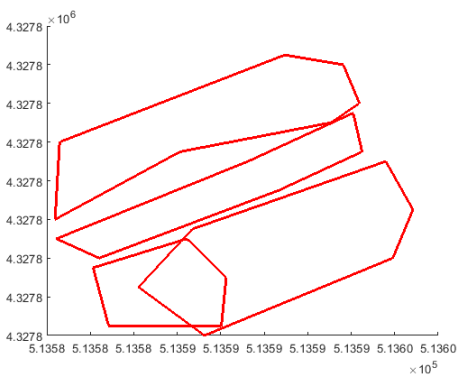
#15			
#16			
#17			
#18			
#19			
#20			
#21			

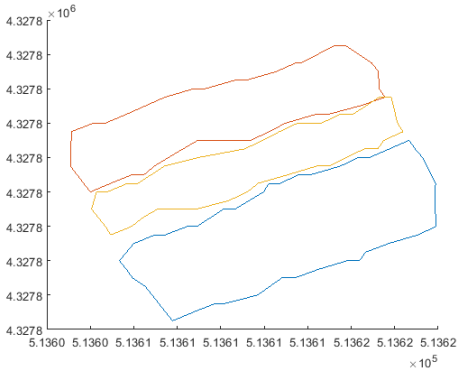
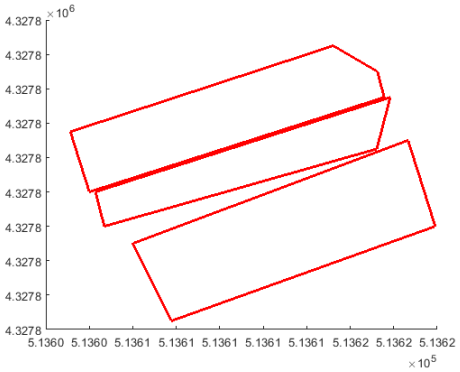
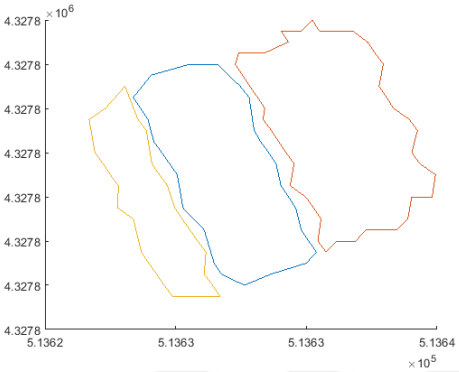
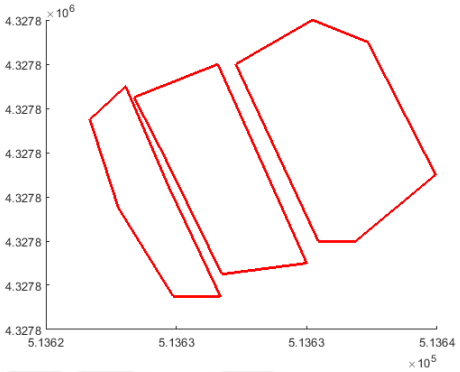
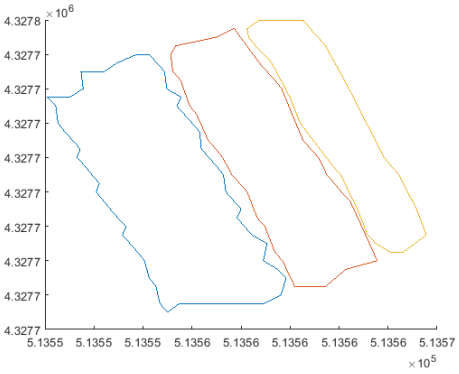
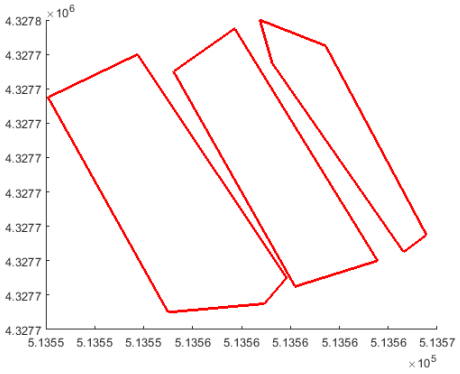
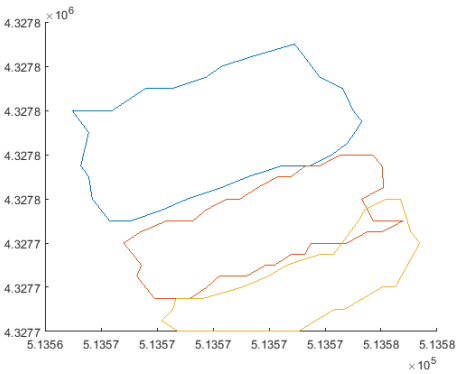
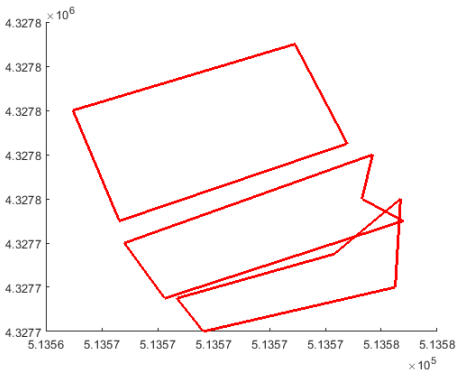
#22			
#23			
#24			
#25			
#26			

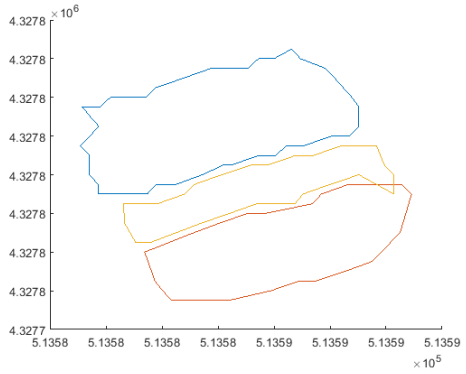
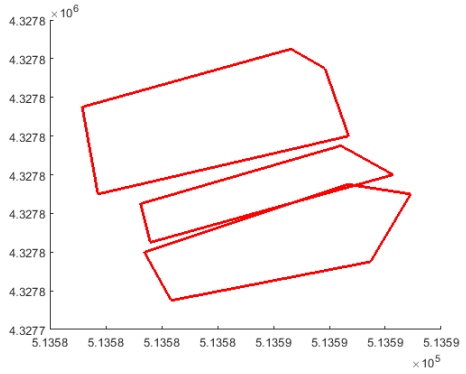
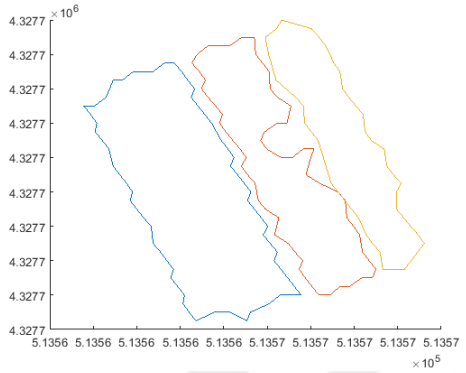
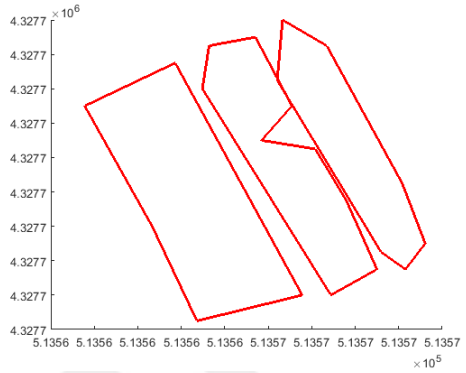
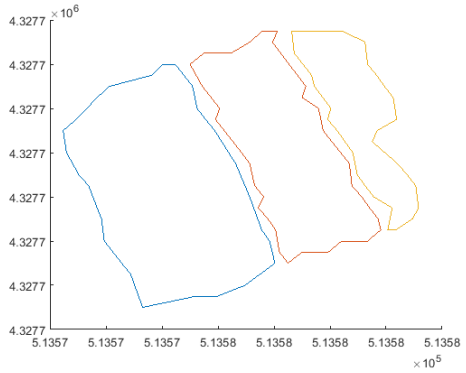
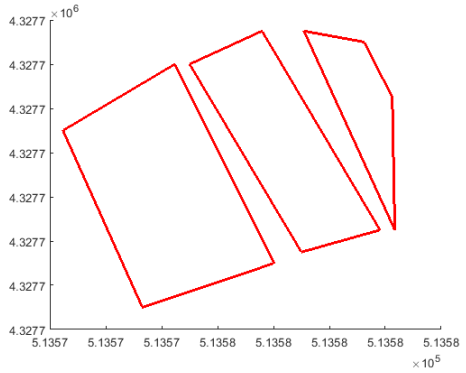
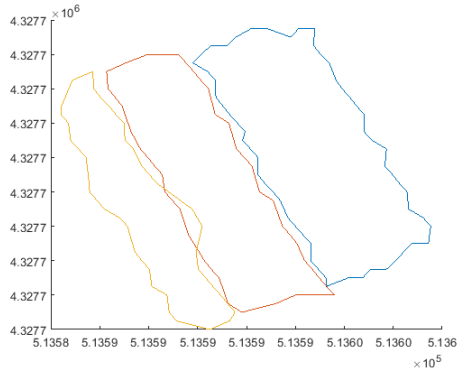
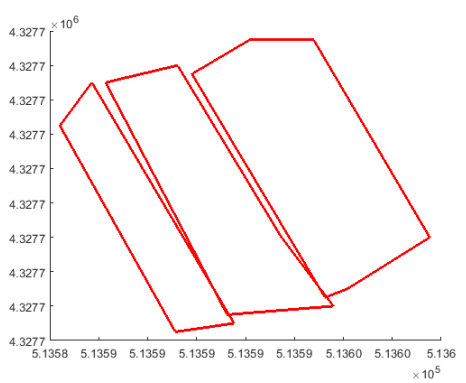
Bina No	Sınırların Çıkarılması	Sınır Sadeleştirmesi
#1		
#2		
#3		

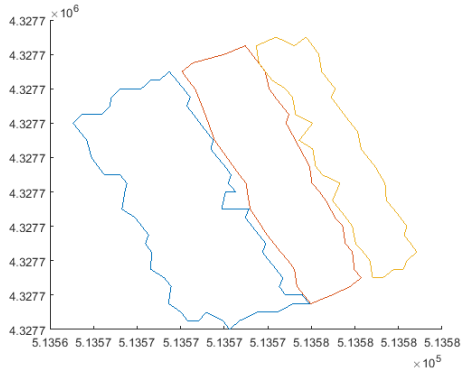
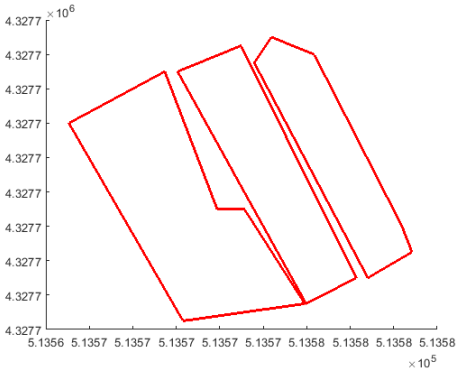
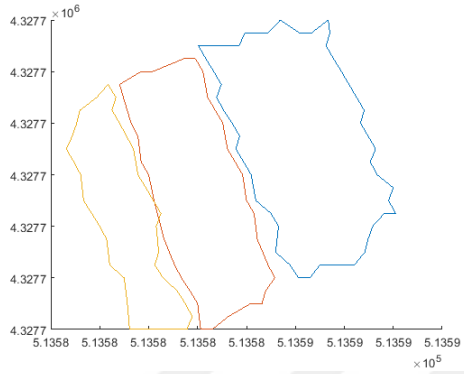
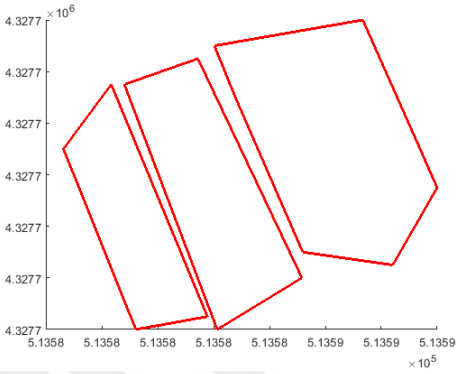
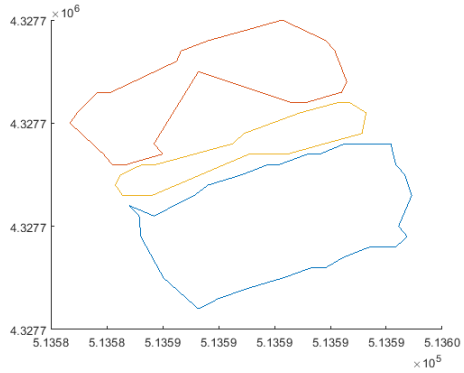
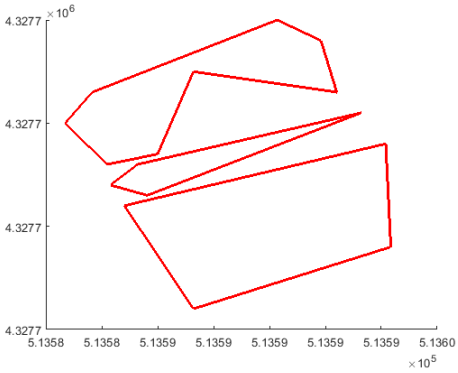
#4		
#5		
#6		
#7		

#8		
#9		
#10		
#11		

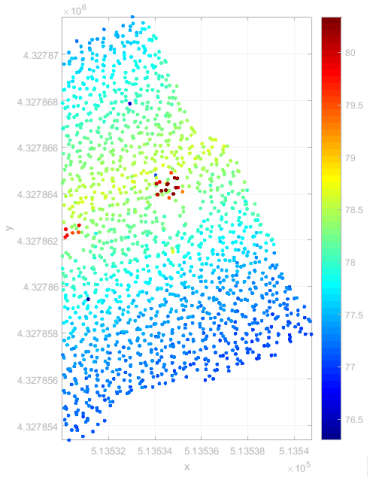
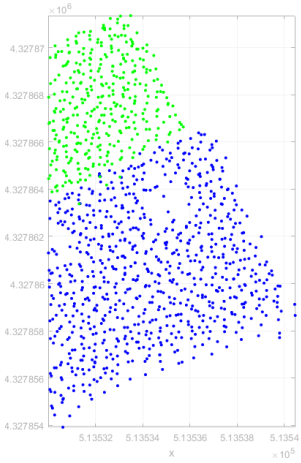
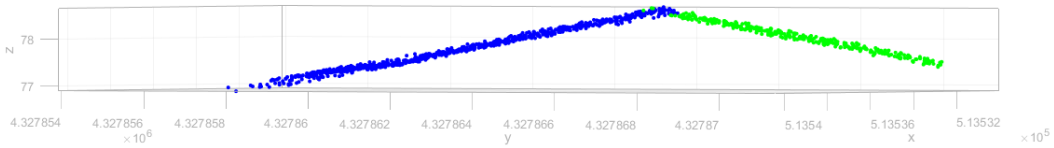
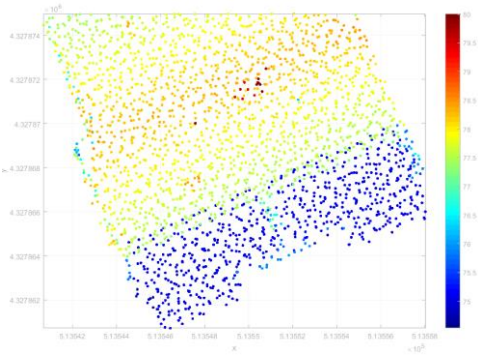
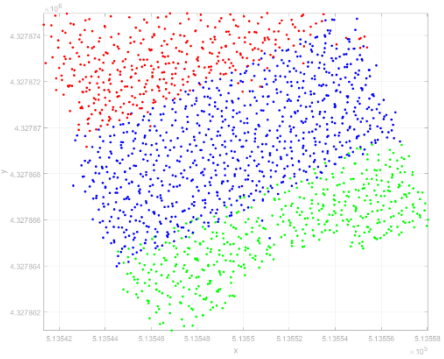
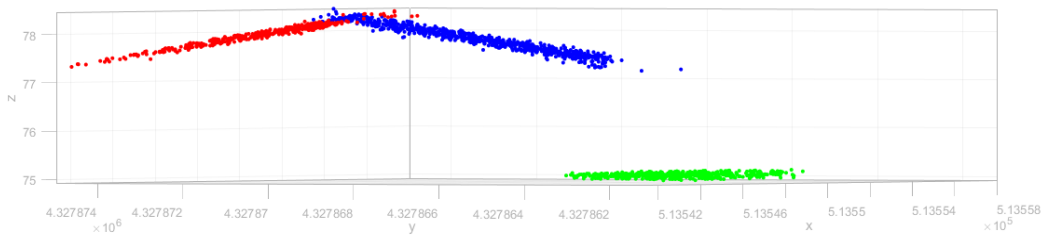
#12		
#13		
#14		
#15		

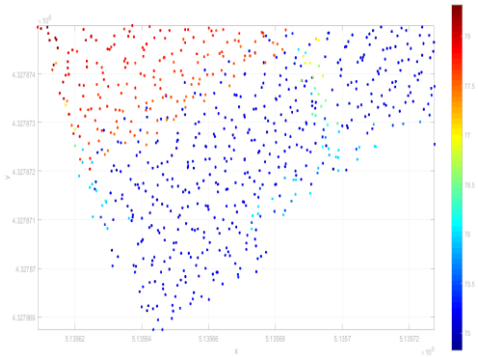
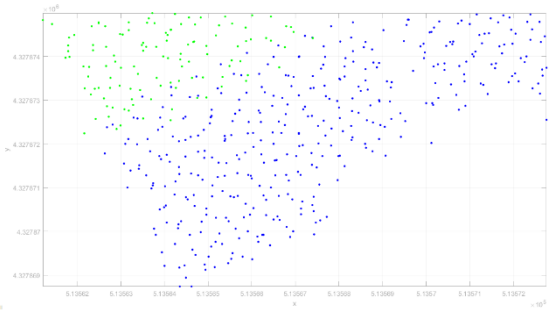
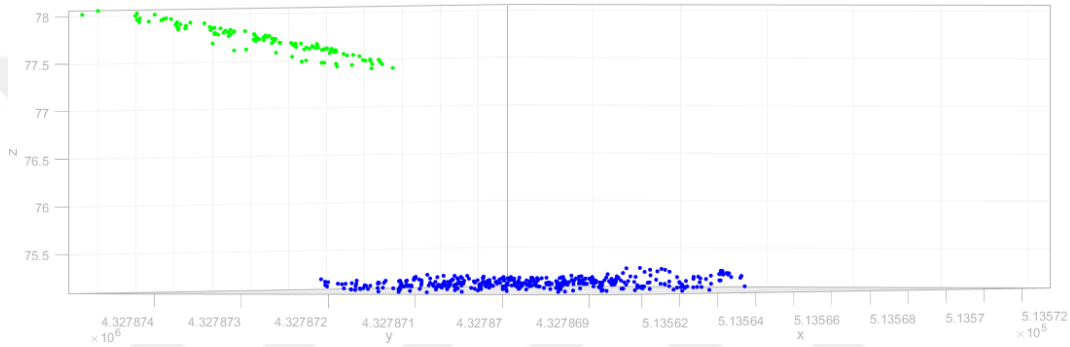
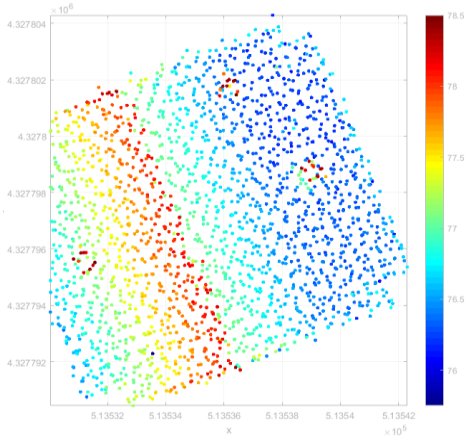
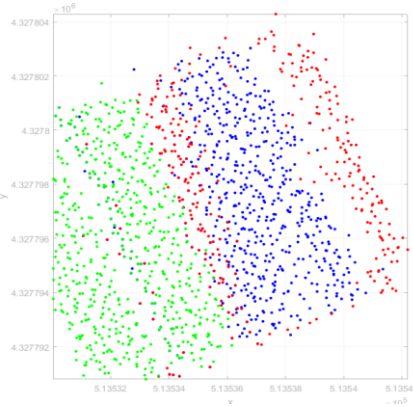
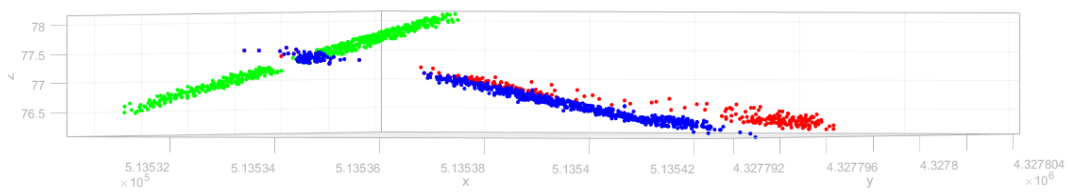
#16		
#17		
#18		
#19		

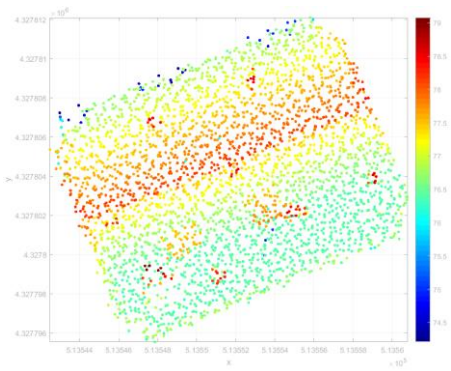
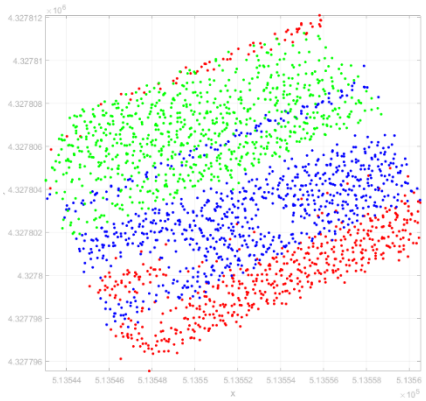
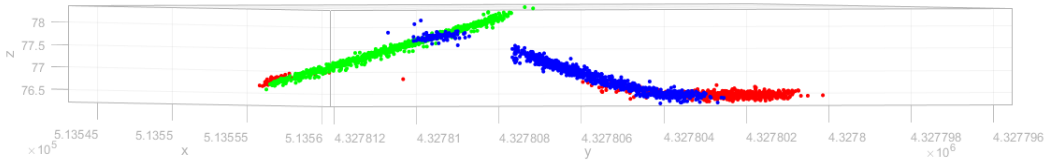
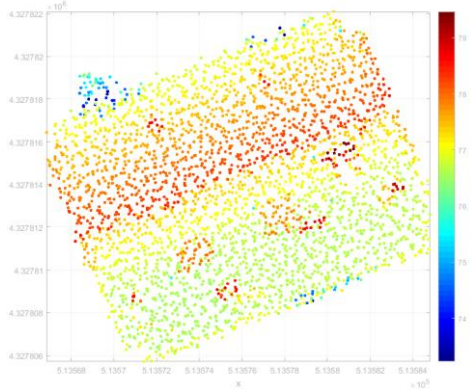
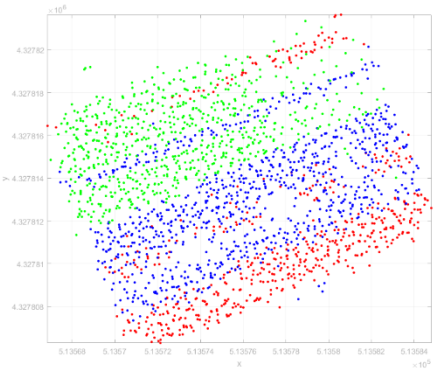
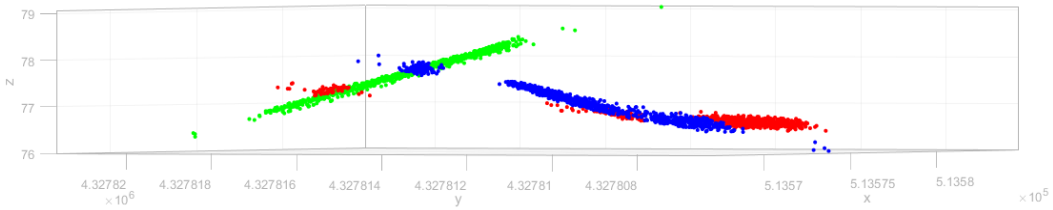
#20		
#21		
#22		
#23		

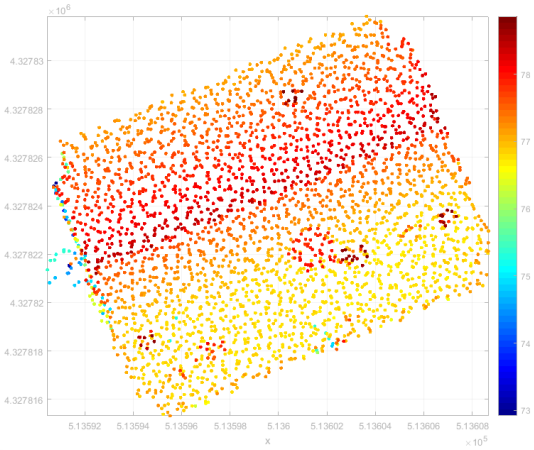
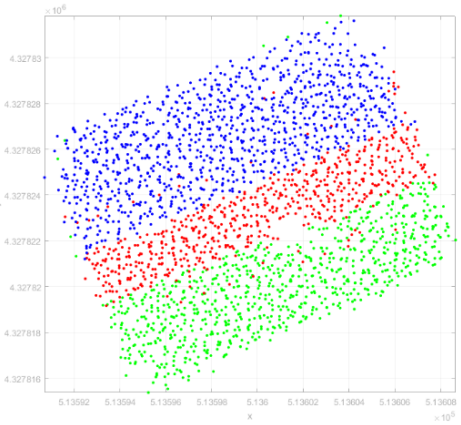
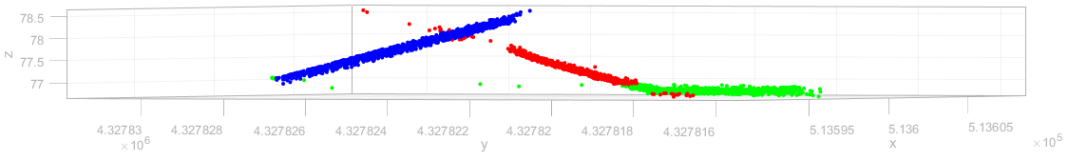
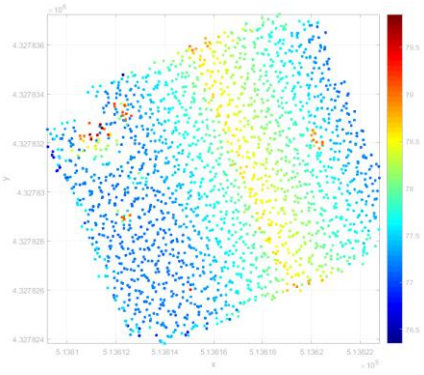
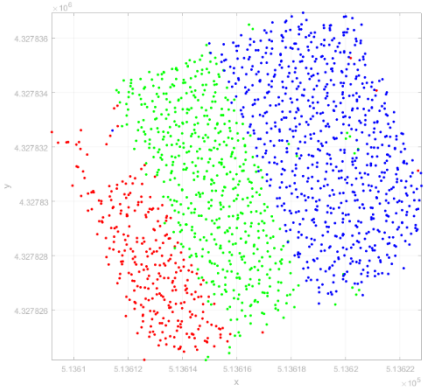
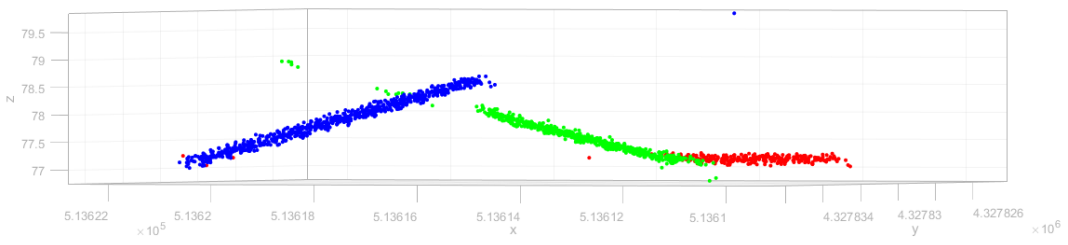
#24		
#25		
#26		

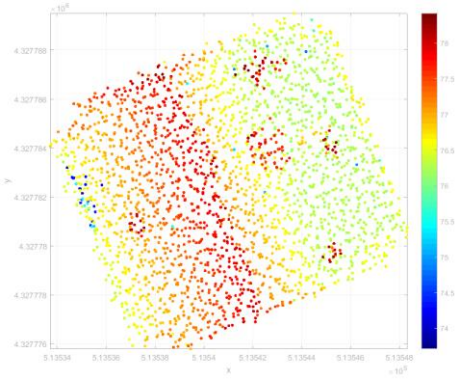
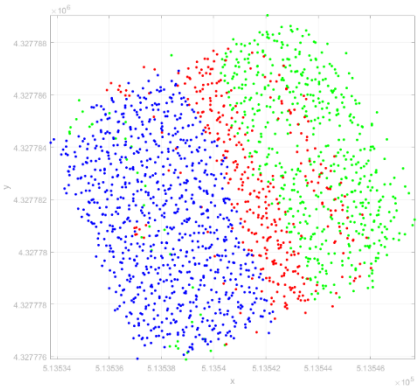
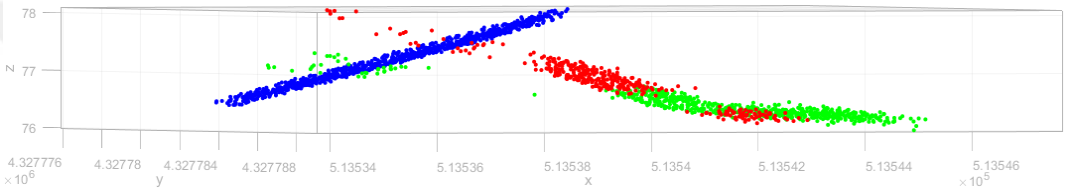
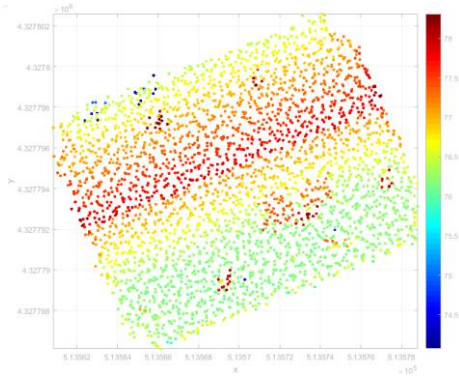
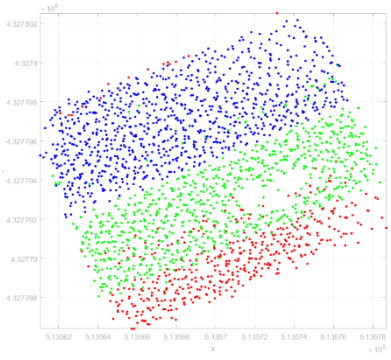
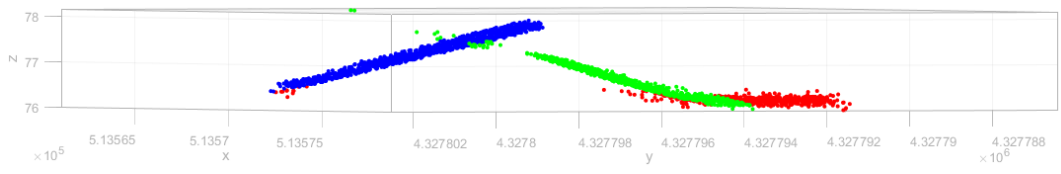
B. Test Alanı #2 Sonuçları

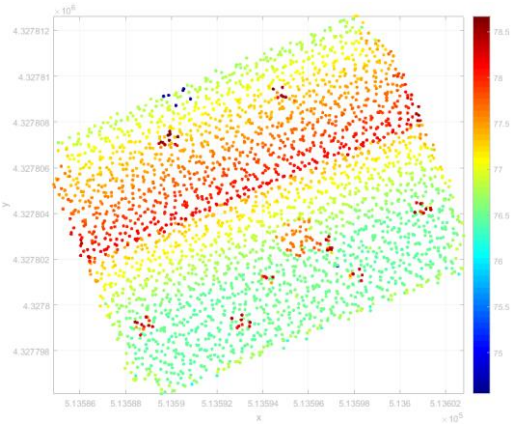
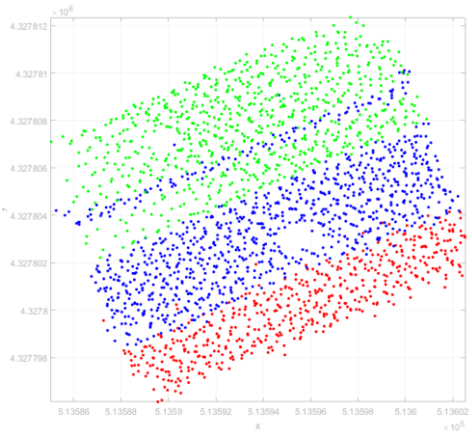
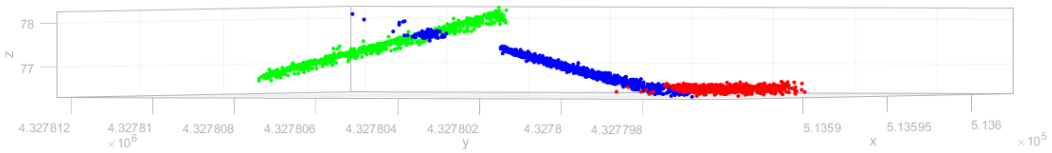
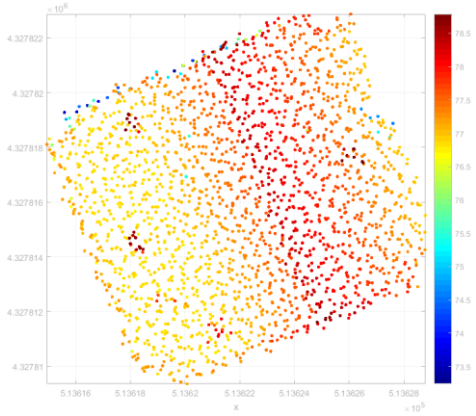
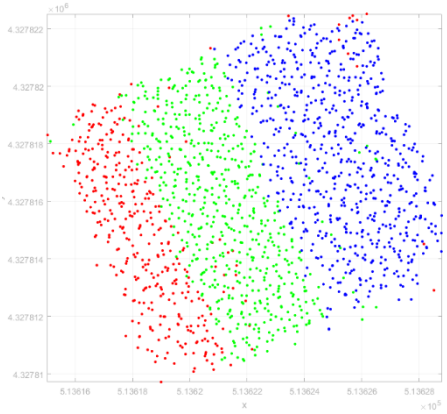
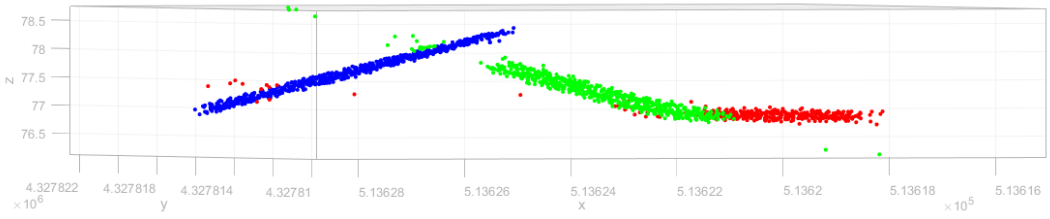
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#1		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#2	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

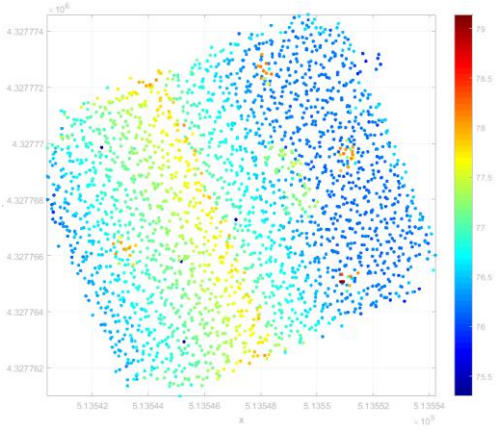
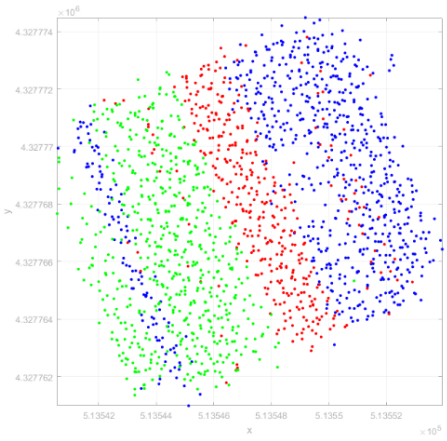
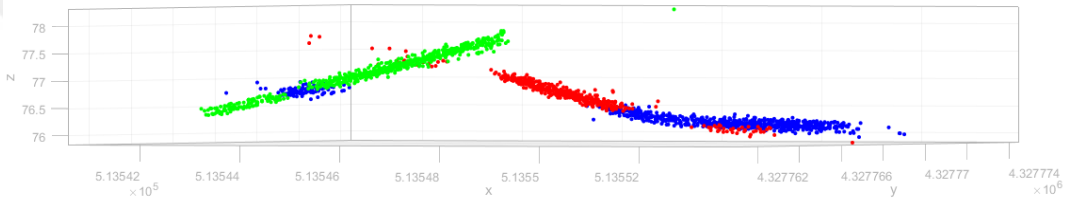
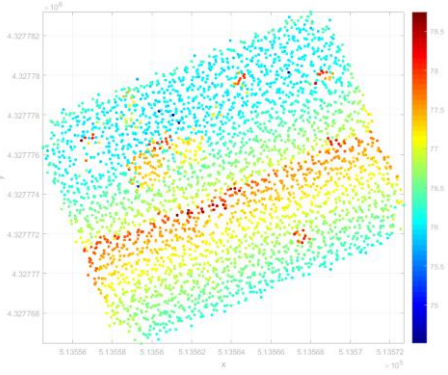
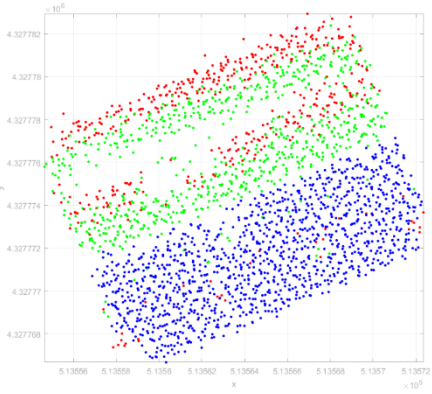
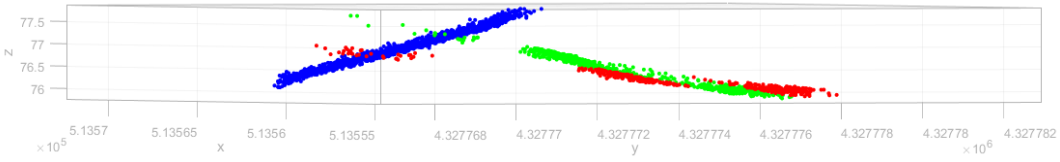
Bina No	
#3	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 
#4	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 

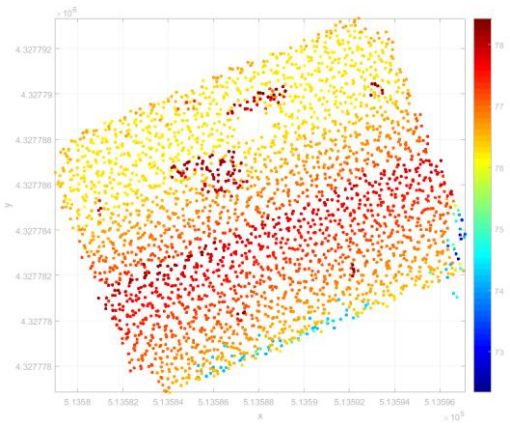
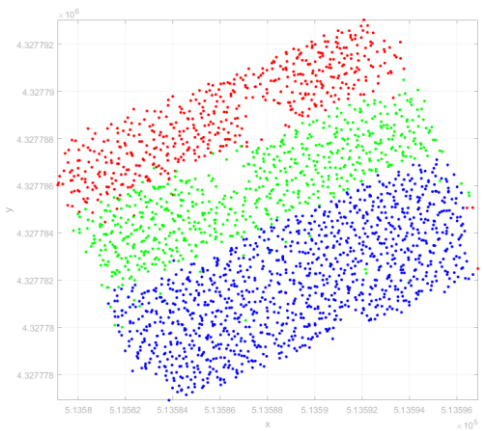
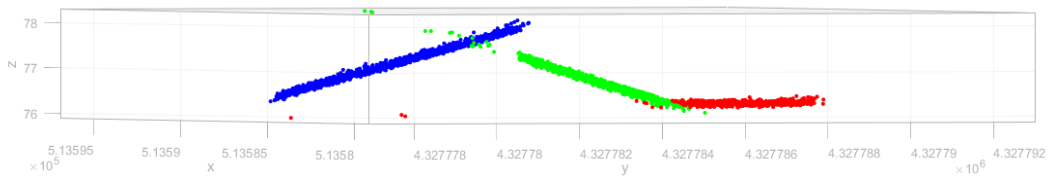
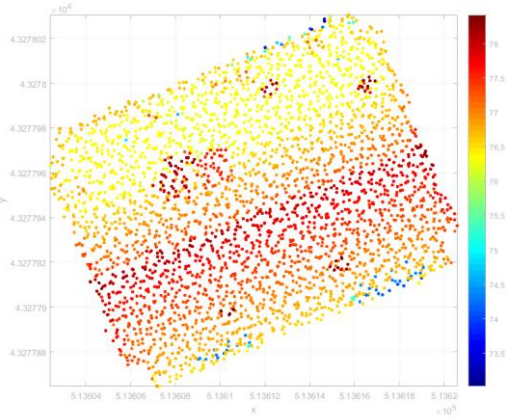
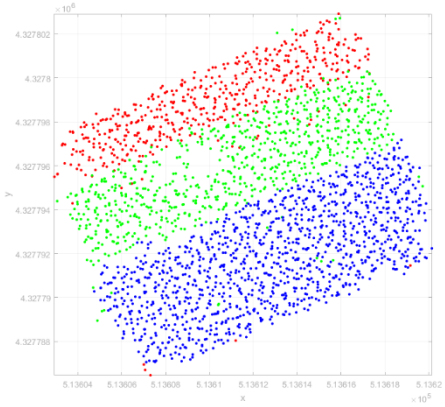
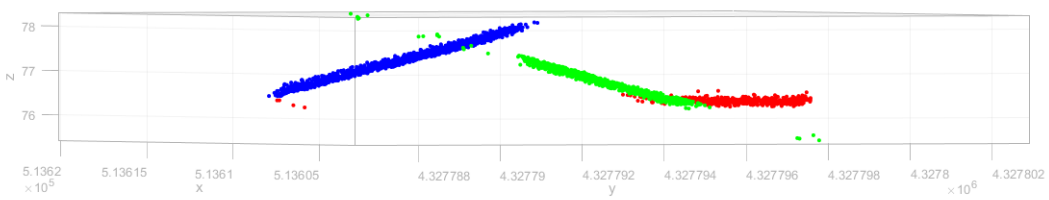
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#5		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#6	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

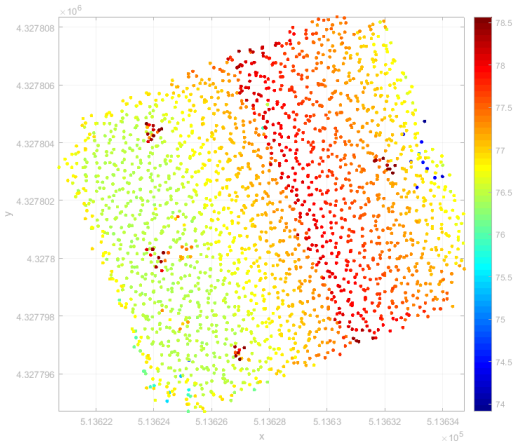
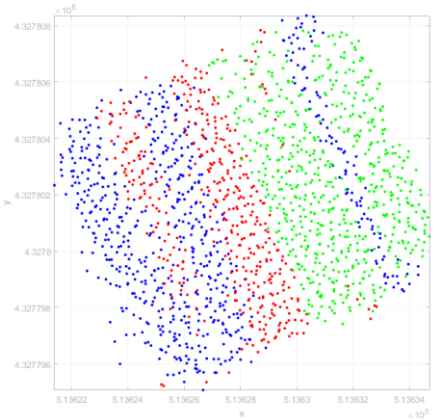
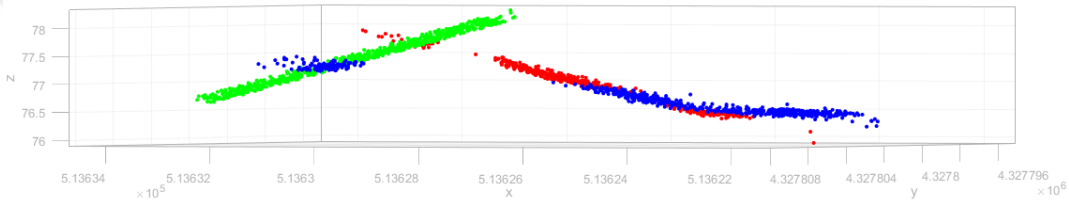
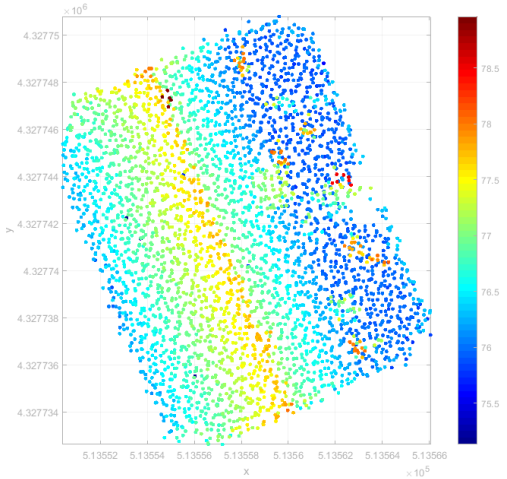
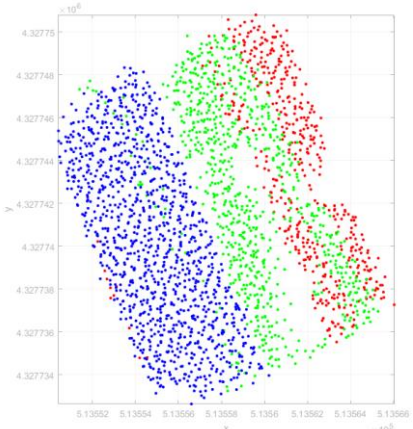
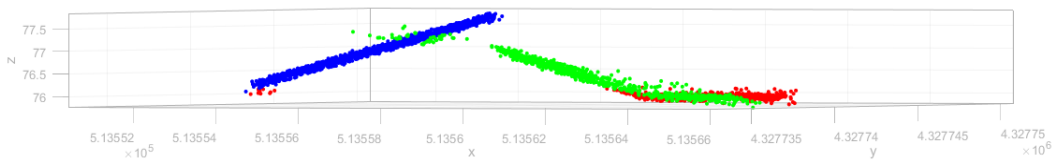
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#7		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#8	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

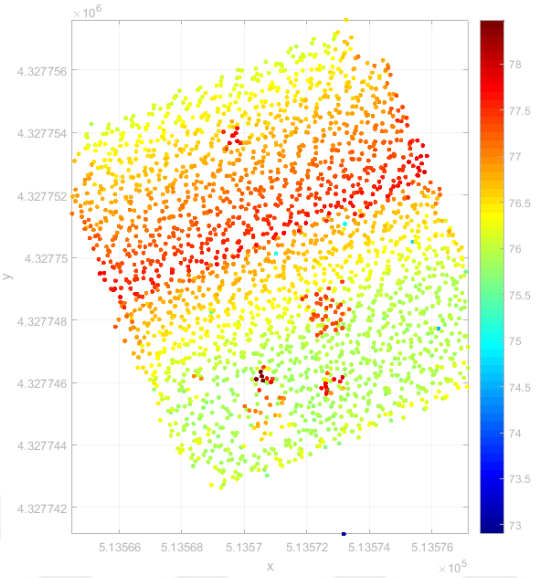
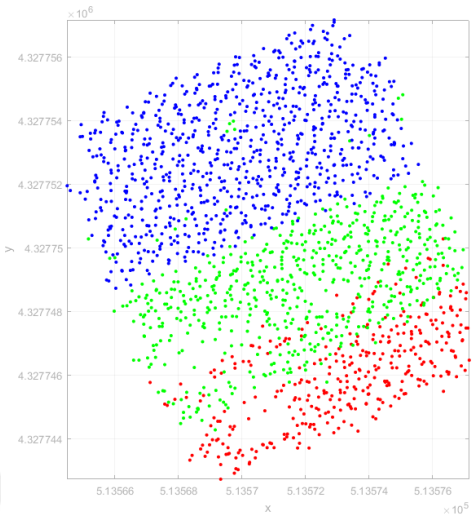
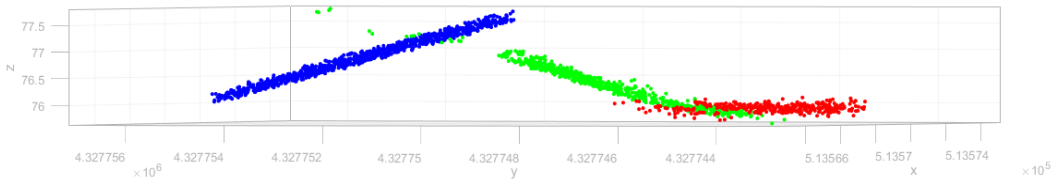
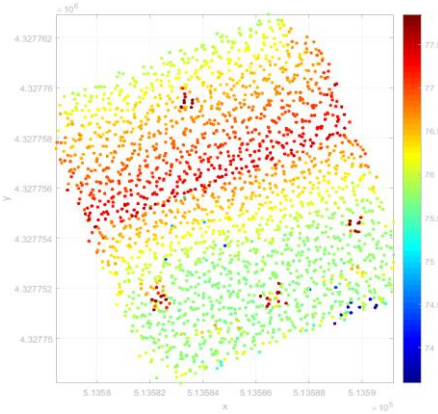
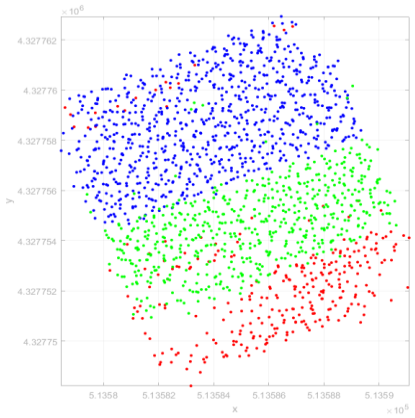
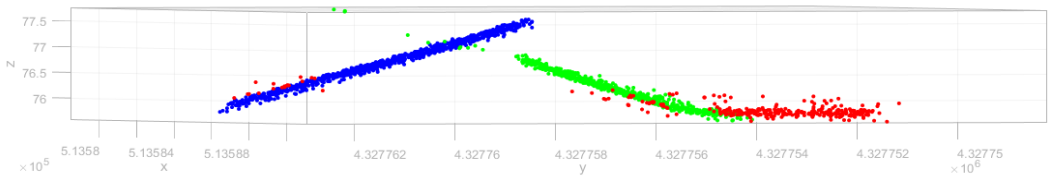
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#9		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#10	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

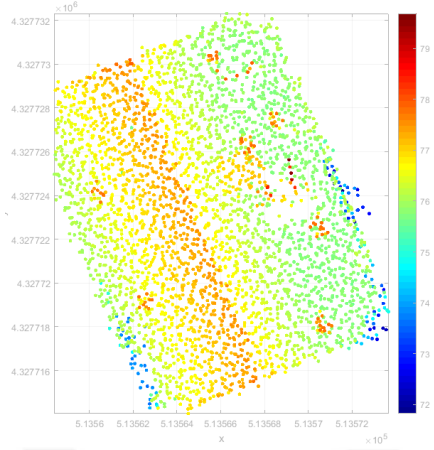
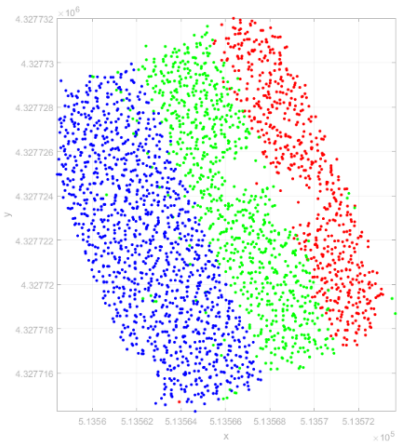
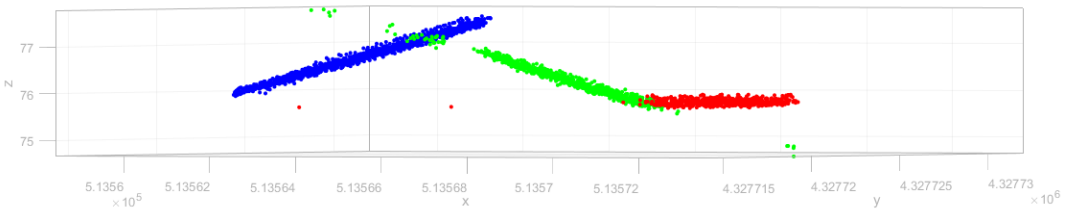
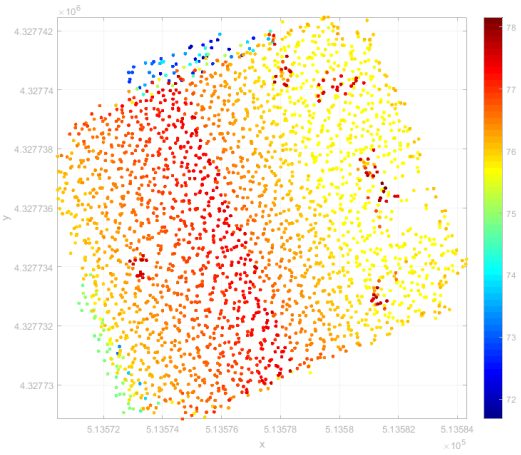
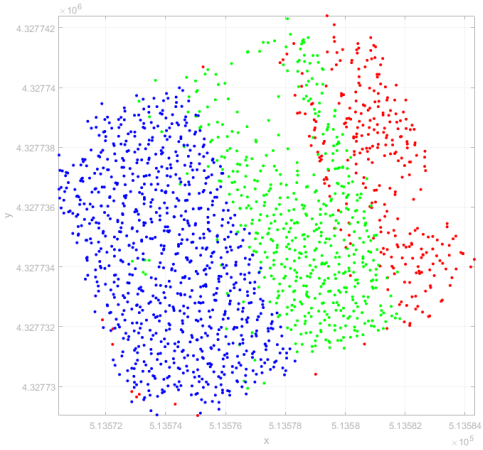
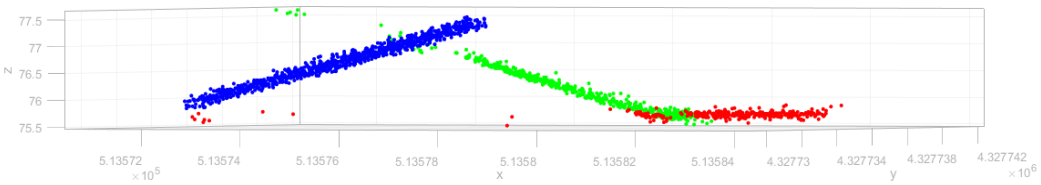
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#11		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#12	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

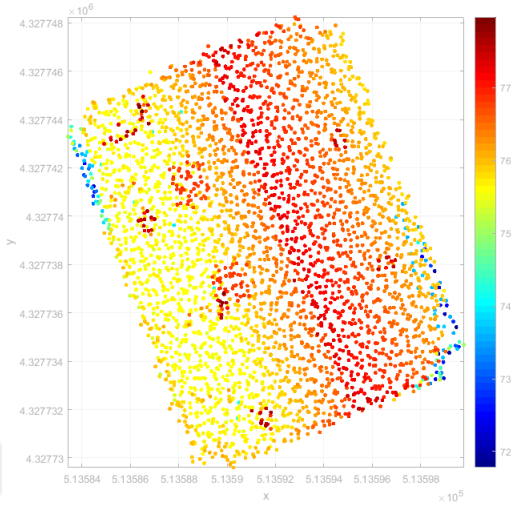
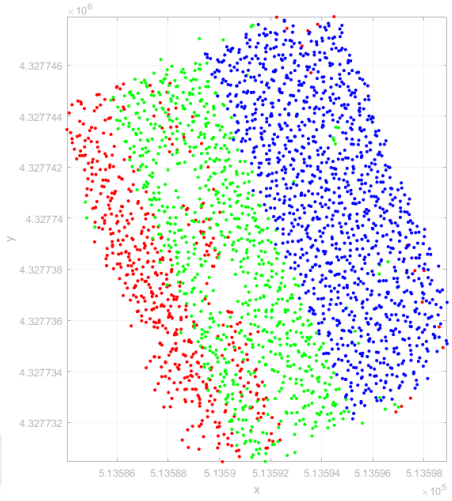
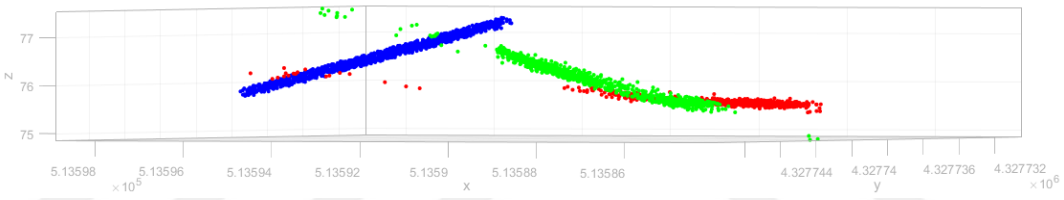
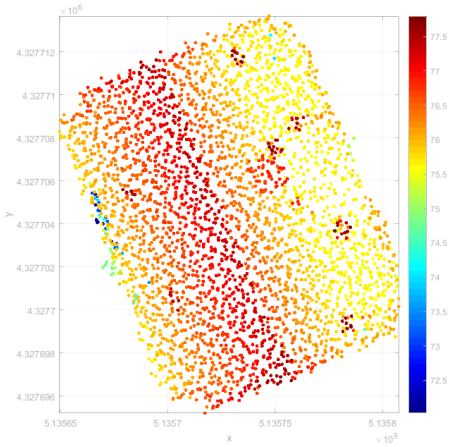
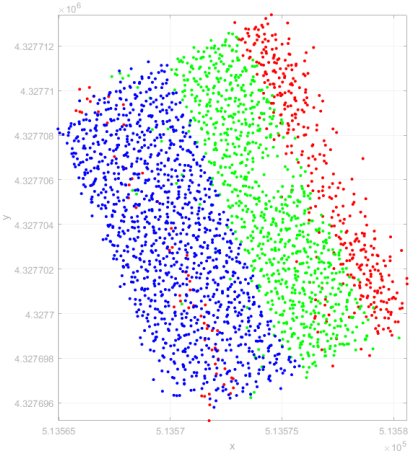
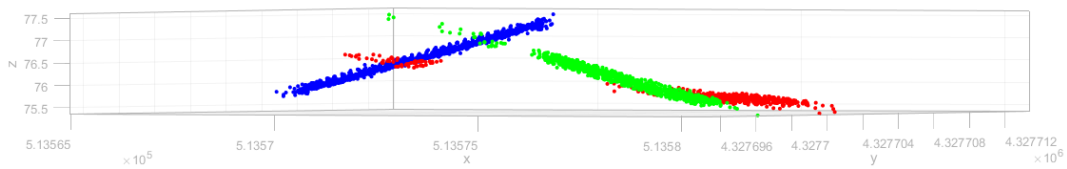
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#13		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#14	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

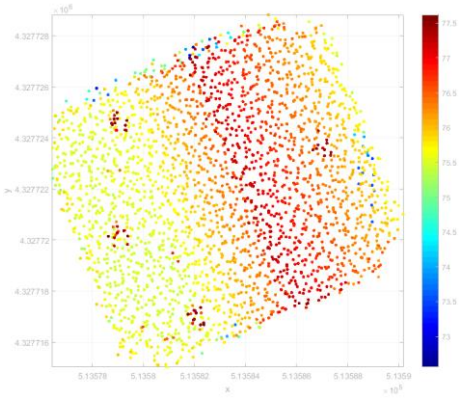
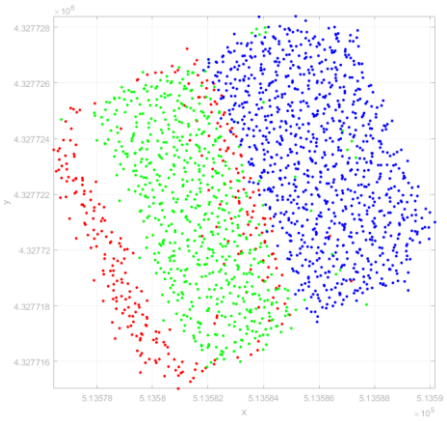
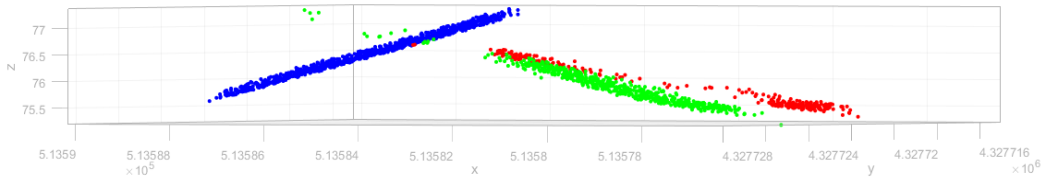
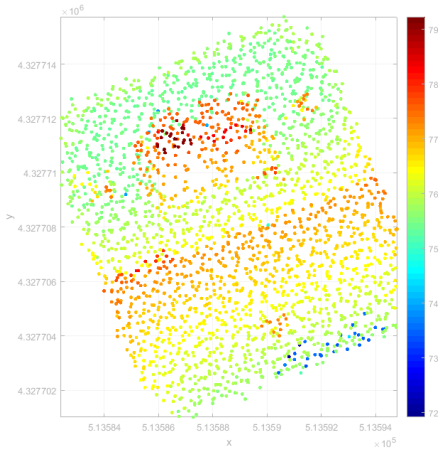
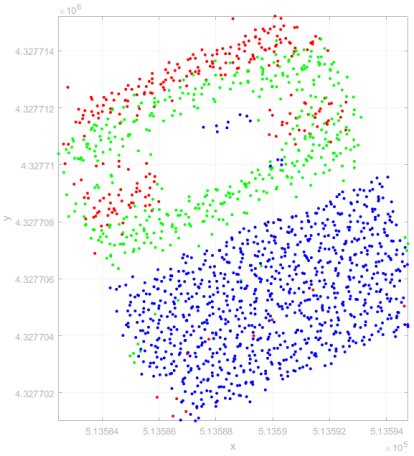
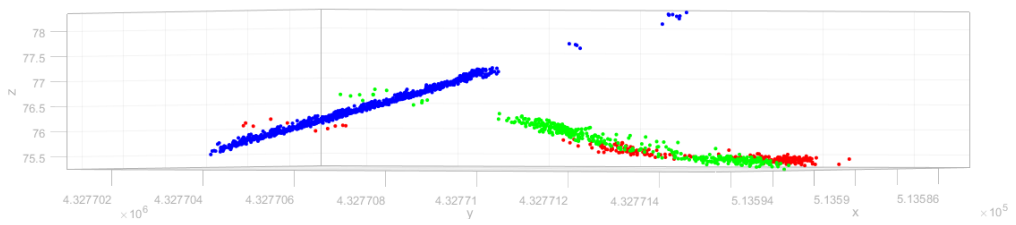
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#15		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#16	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

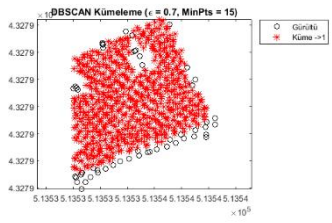
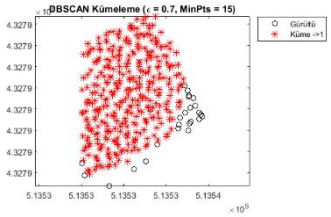
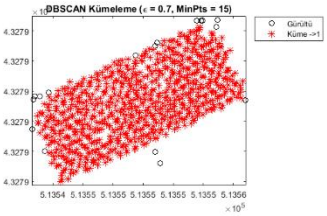
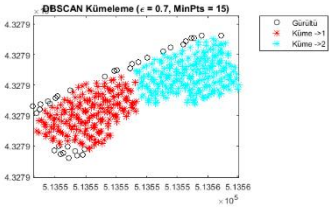
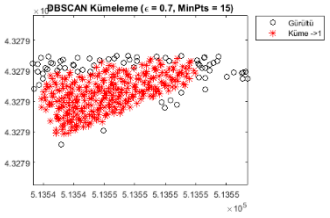
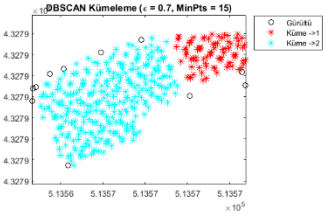
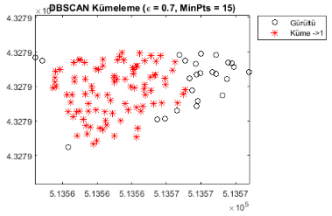
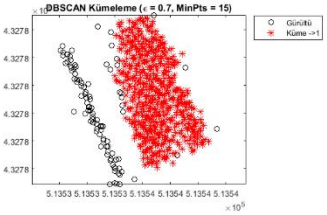
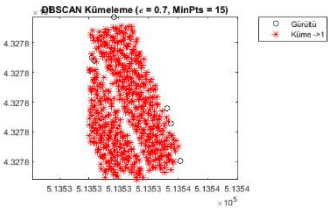
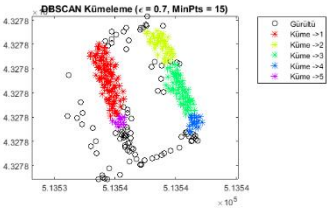
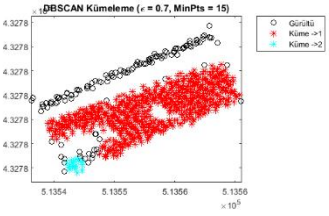
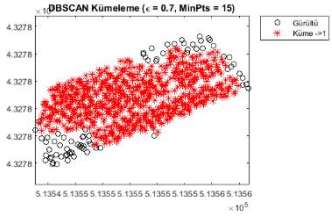
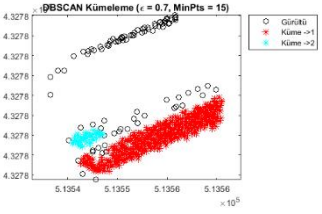
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#17		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#18	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

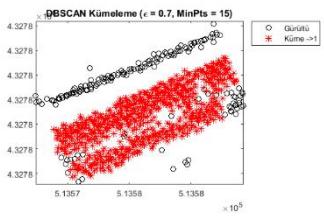
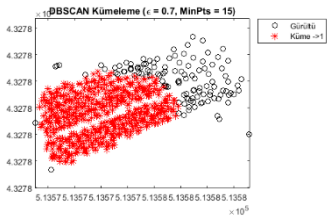
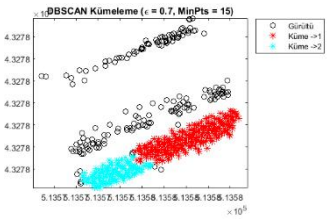
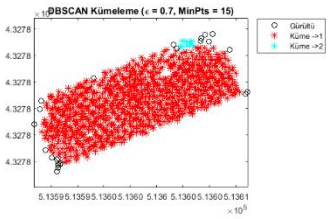
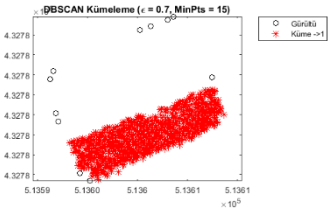
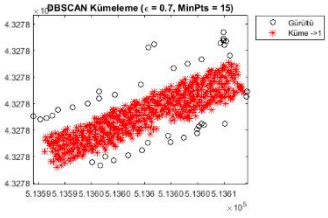
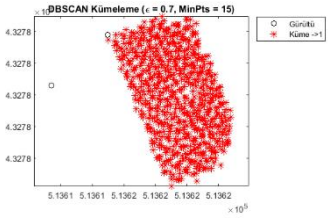
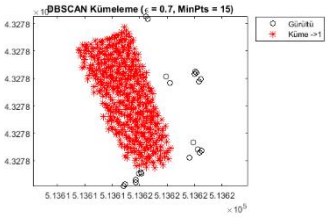
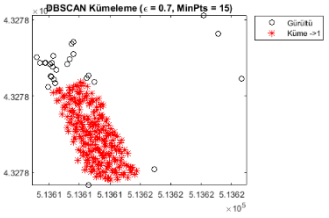
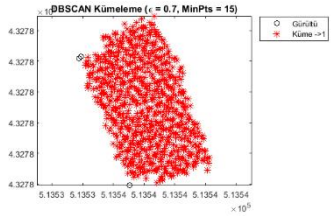
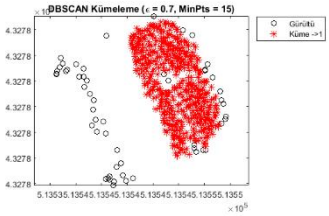
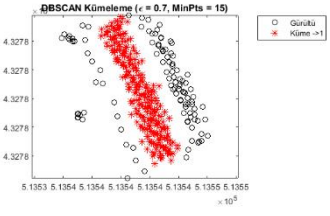
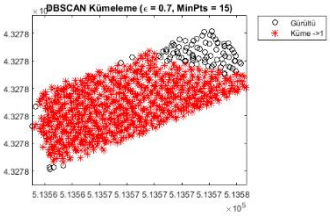
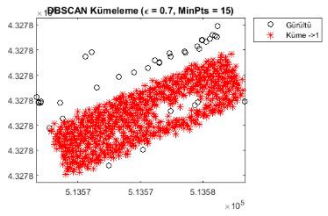
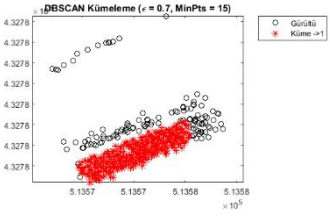
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#19		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#20	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

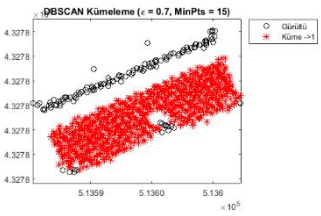
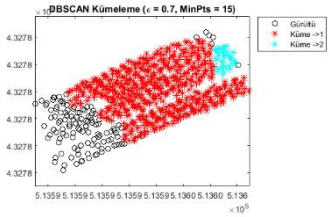
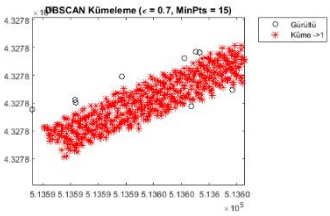
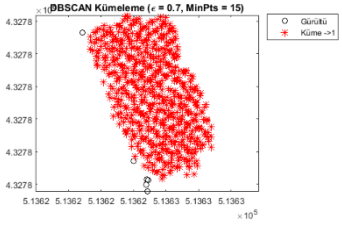
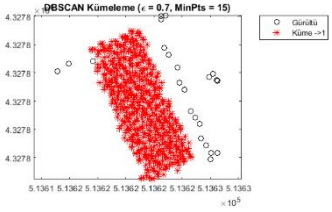
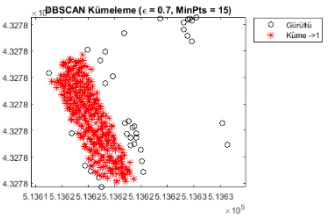
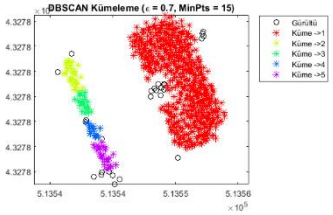
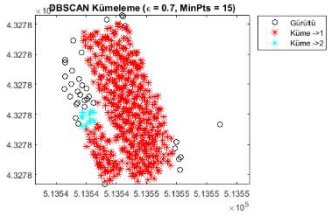
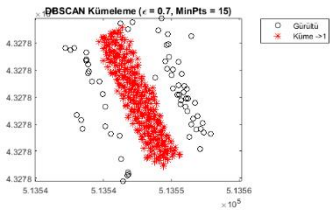
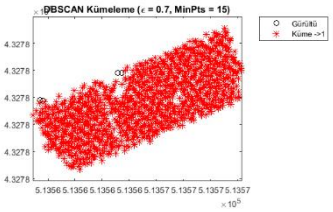
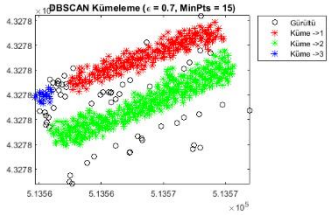
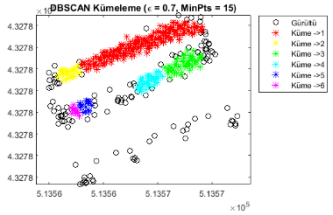
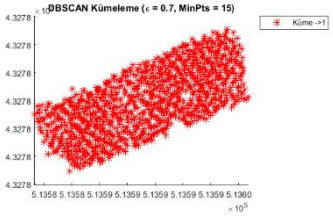
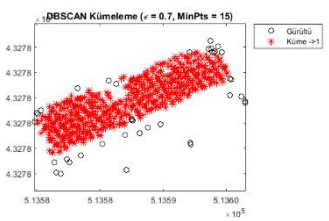
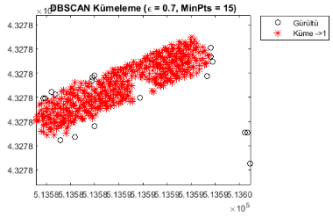
Bina No	
#21	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 
#22	**Bina Nokta Bulutu Kümesi**  **RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**  **RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü** 

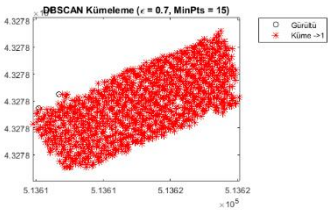
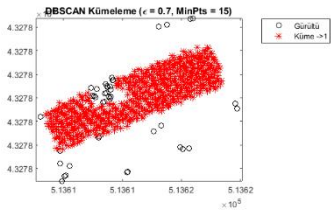
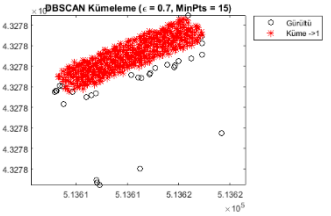
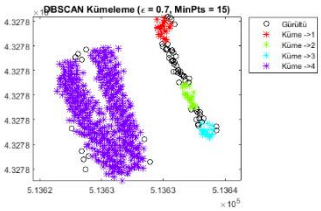
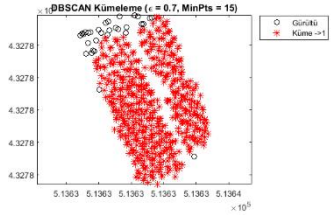
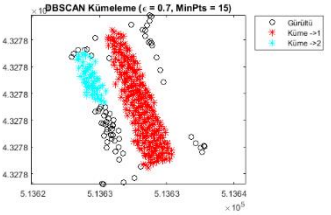
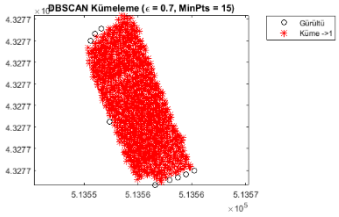
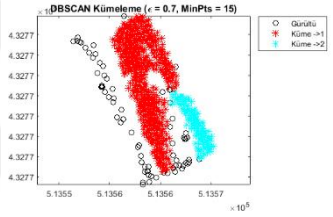
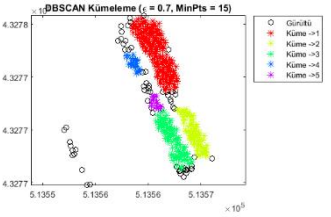
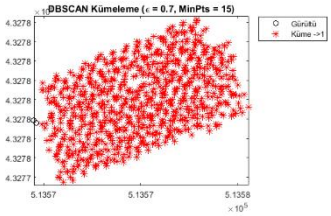
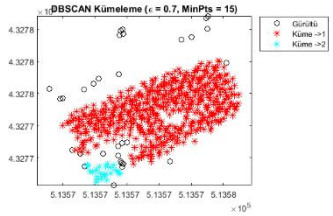
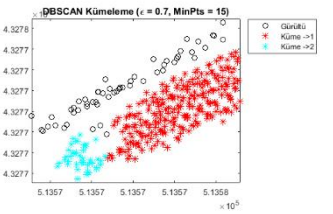
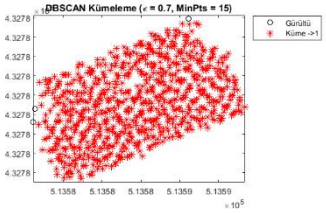
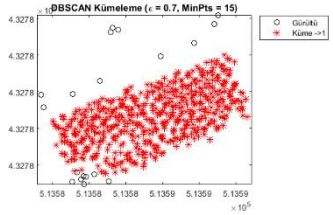
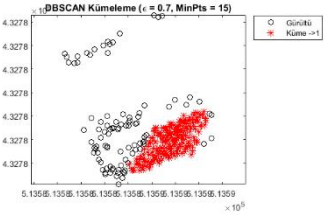
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#23		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#24	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

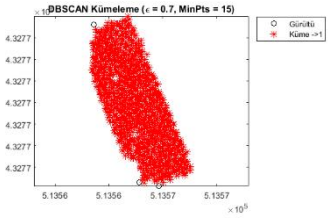
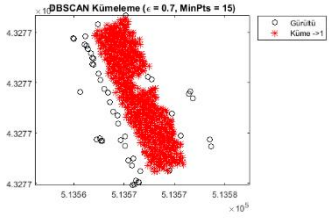
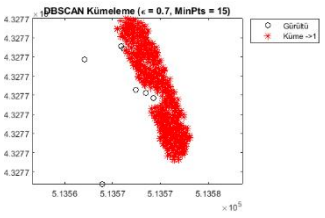
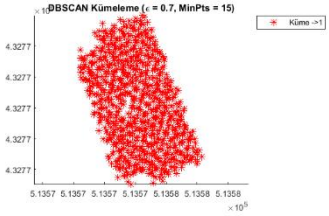
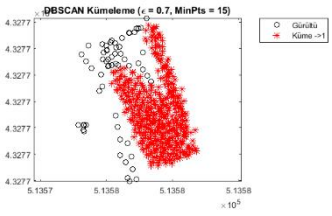
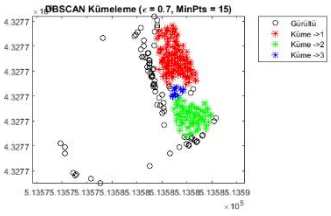
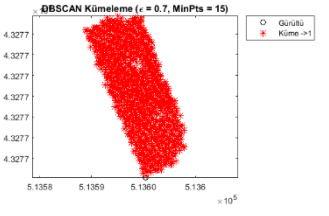
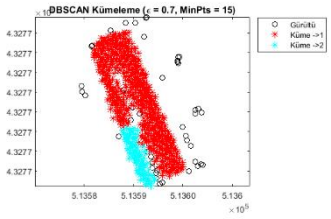
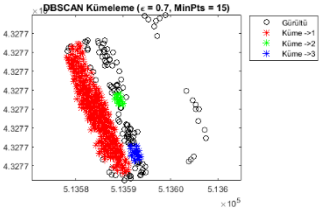
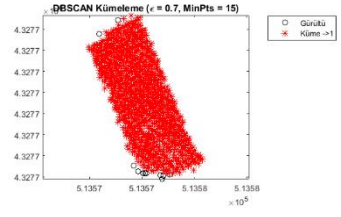
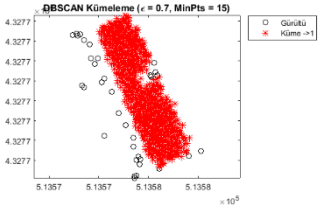
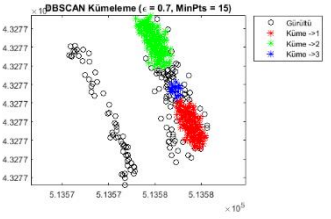
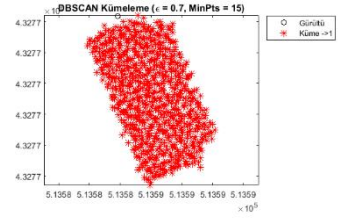
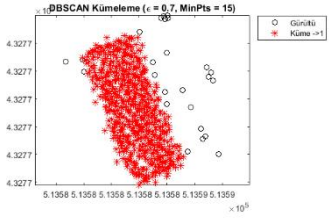
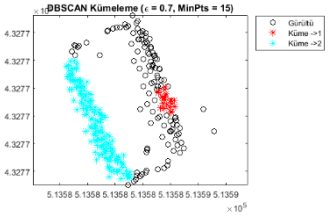
Bina No	Bina Nokta Bulutu Kümesi	RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü
#25		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		
#26	**Bina Nokta Bulutu Kümesi**	**RANSAC Algoritması ile Düzlem Çıkarımı Üstten Görünümü**
		
	RANSAC Algoritması ile Düzlem Çıkarımı Profil Görüntüsü	
		

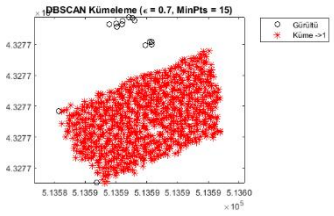
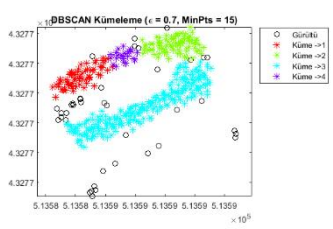
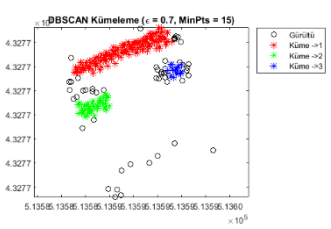
Bina No	DBSCAN Kümeleme		
#1			
#2			
#3			
#4			
#5			

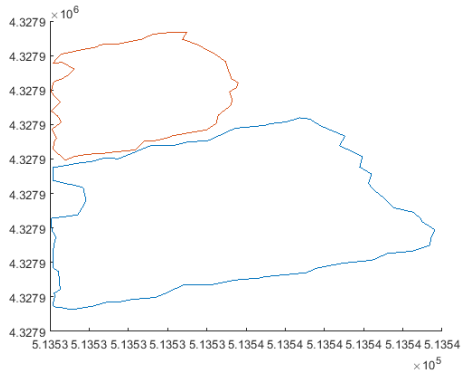
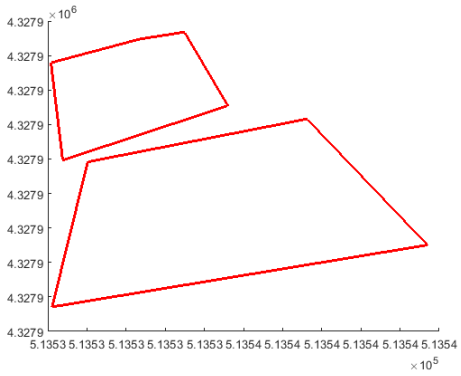
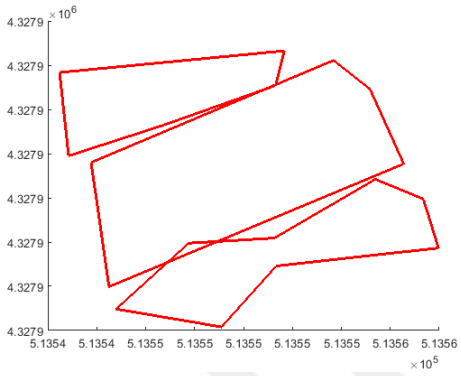
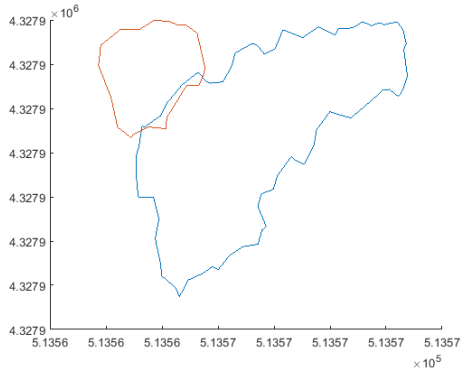
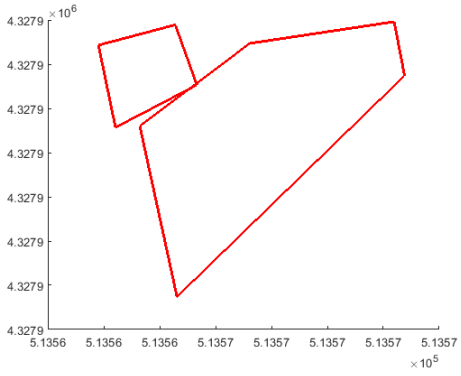
#6	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2
#7	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1
#8	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1
#9	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1
#10	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1

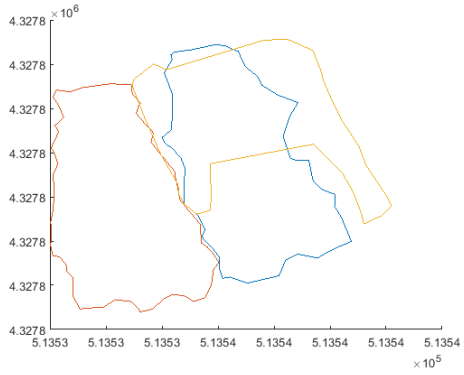
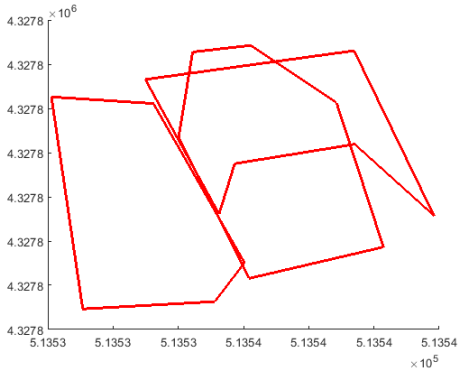
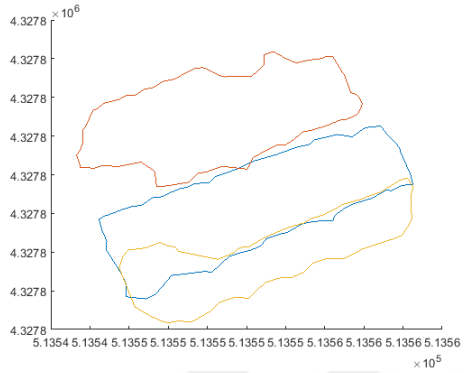
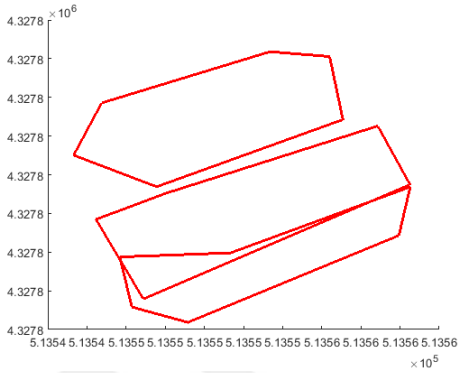
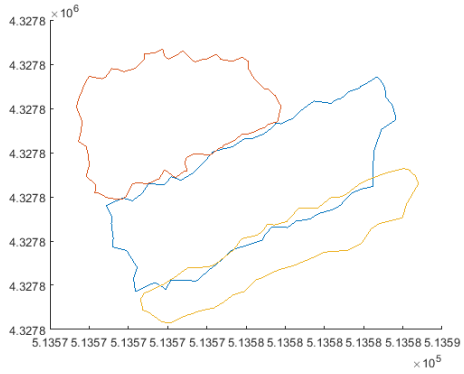
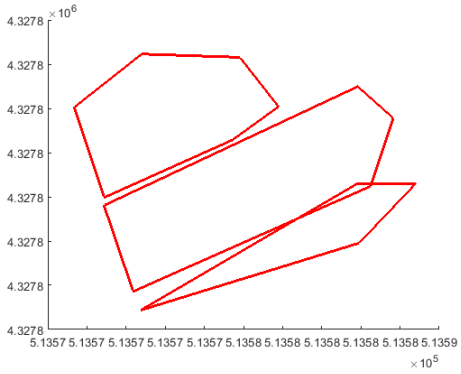
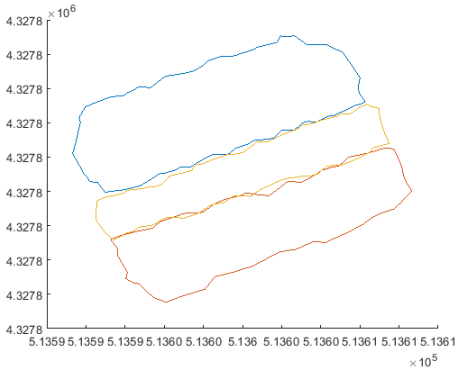
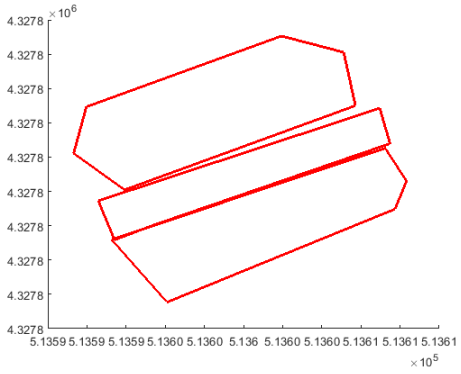
#11	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1
#12	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1
#13	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3 Küme ->4 Küme ->5	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1
#14	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3 Küme ->4 Küme ->5 Küme ->6
#15	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1

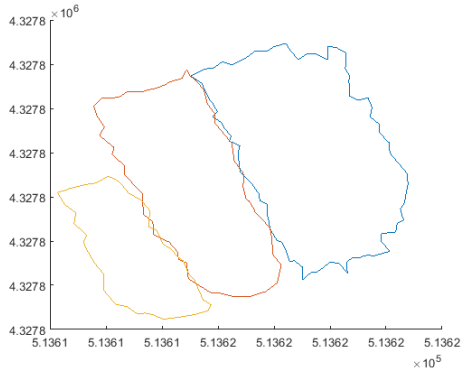
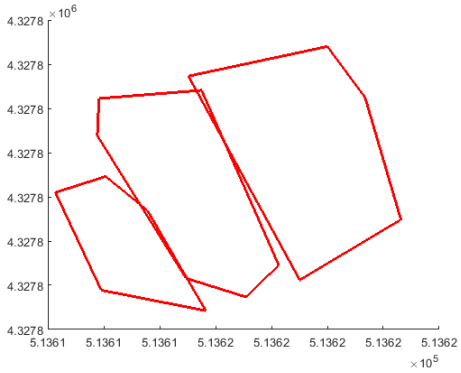
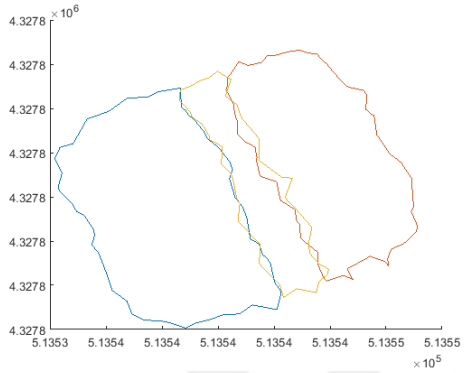
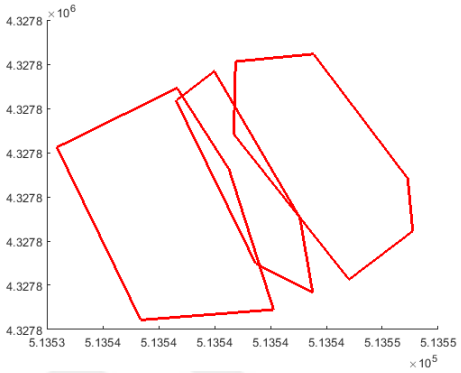
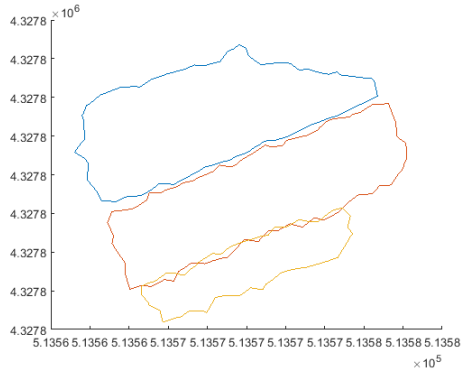
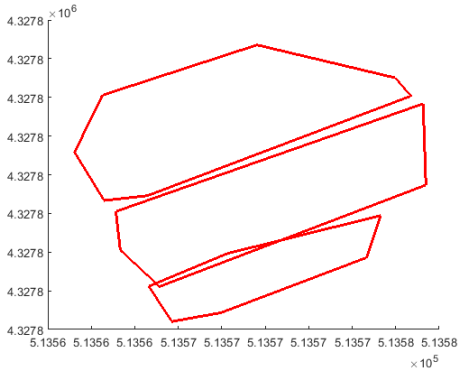
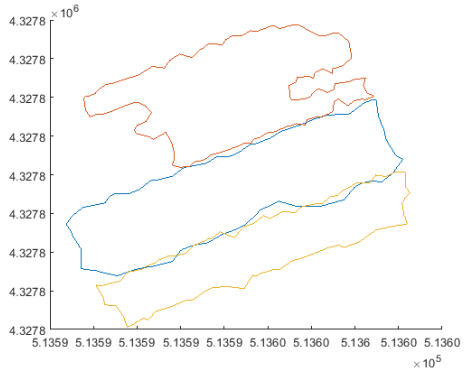
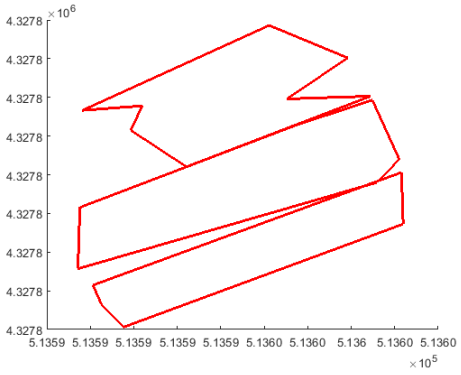
#16	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1
#17	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3 Küme ->4	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2
#18	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3 Küme ->4 Küme ->5
#19	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2
#20	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1

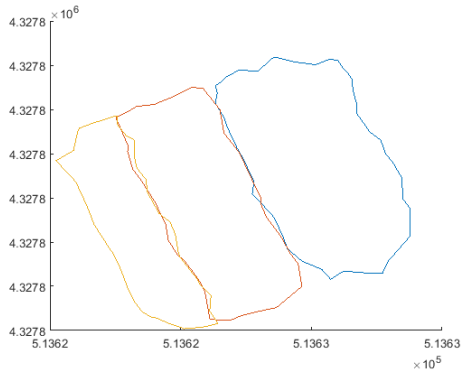
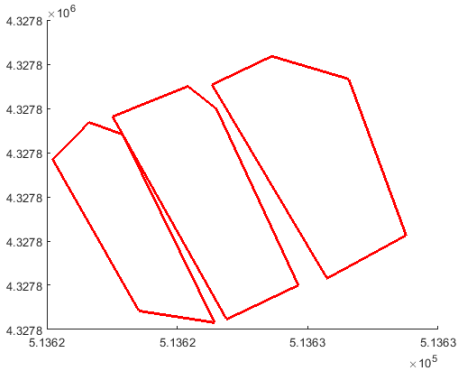
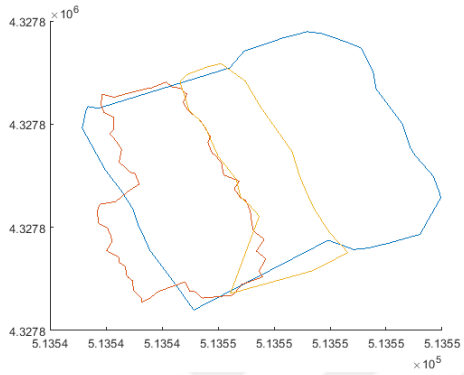
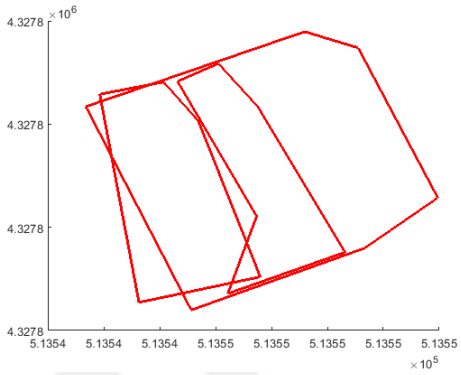
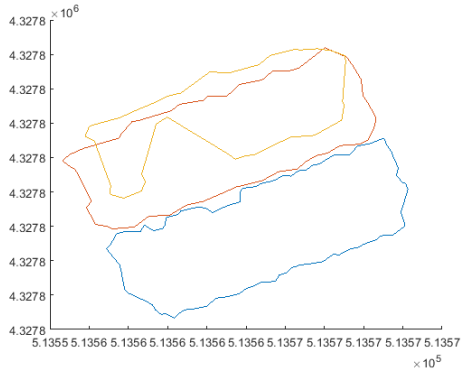
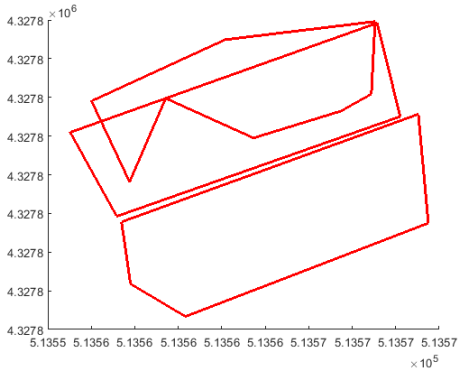
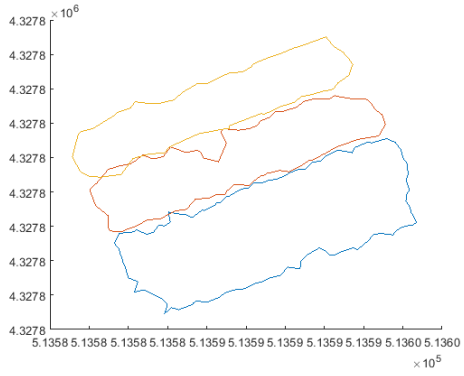
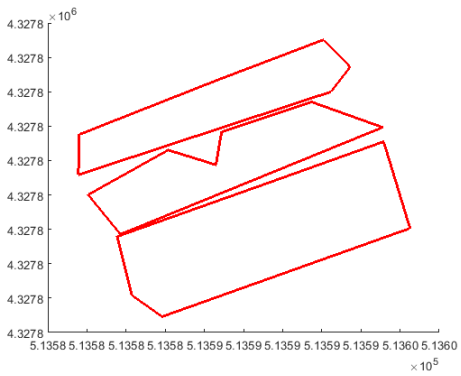
#21	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1
#22	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3
#23	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3
#24	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3
#25	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ϵ = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2

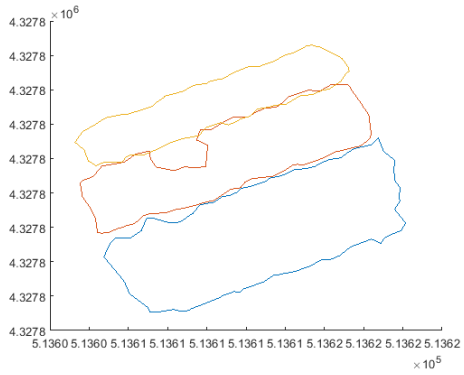
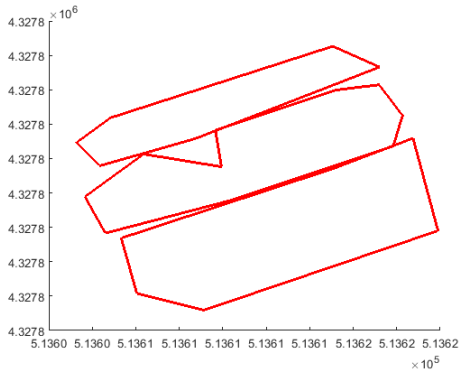
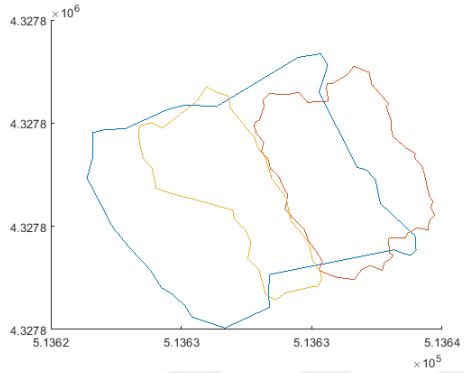
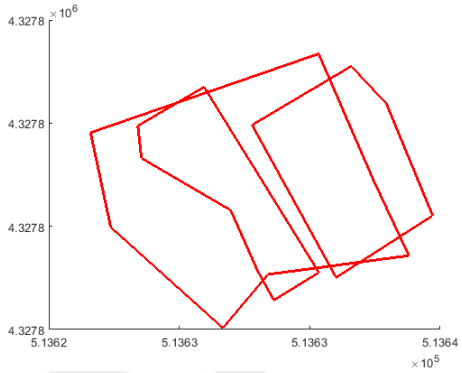
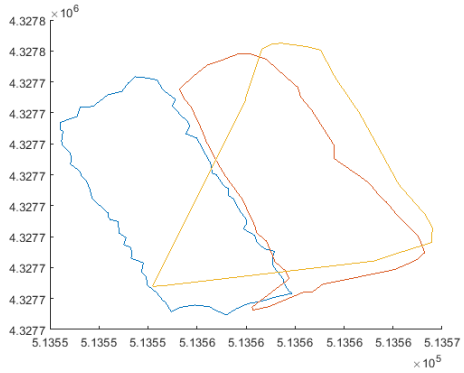
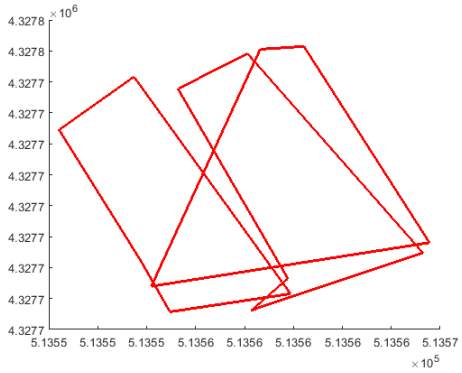
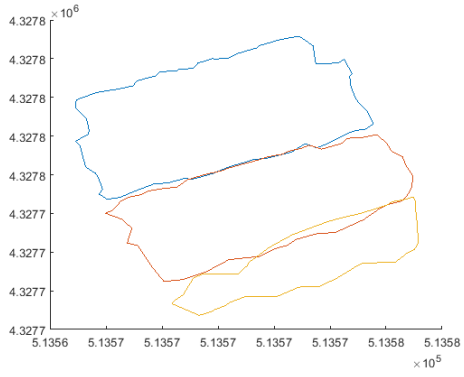
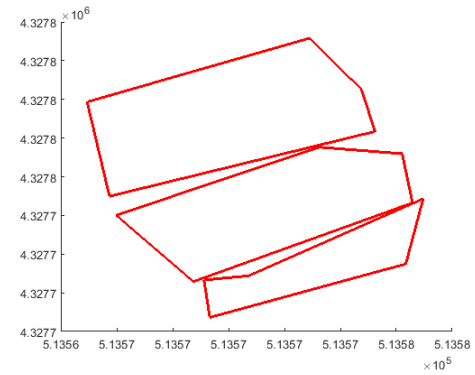
#26	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3 Küme ->4	 DBSCAN Kümeleme (ε = 0.7, MinPts = 15) Gürültü Küme ->1 Küme ->2 Küme ->3

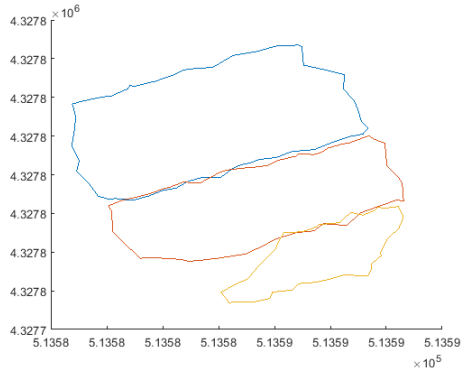
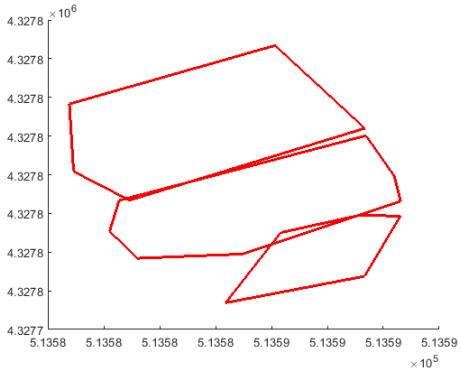
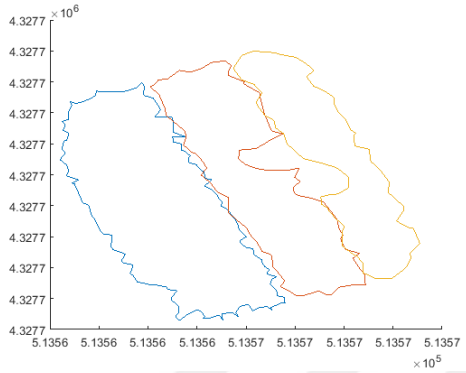
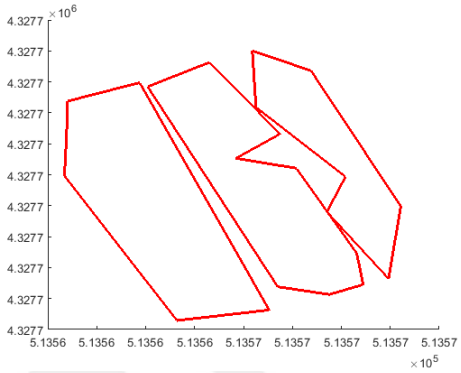
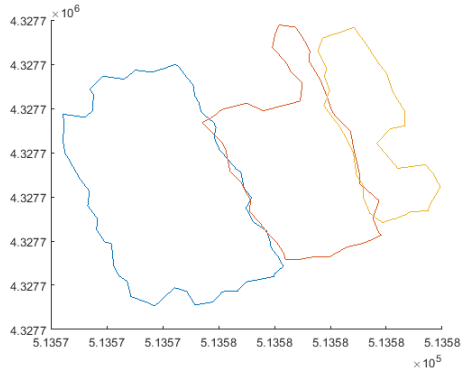
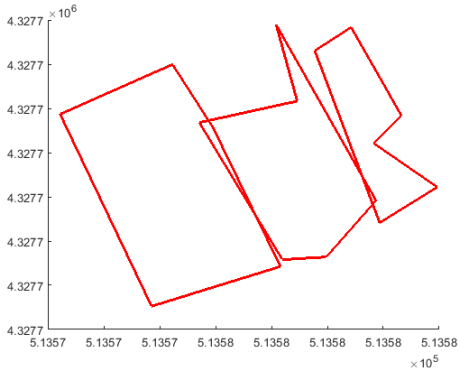
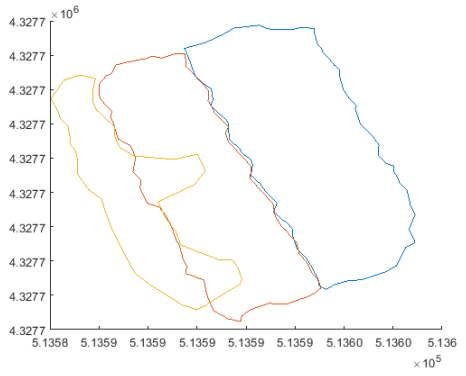
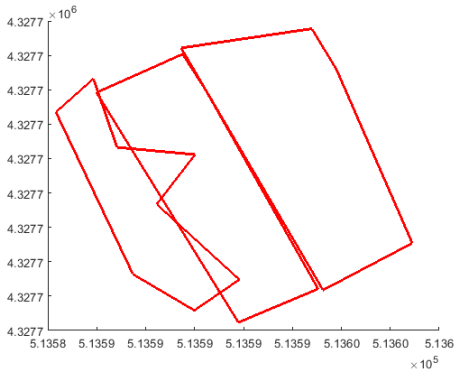
Bina No	Sınırların Çıkarılması	Sınır Sadeleştirmesi
#1		
#2		
#3		

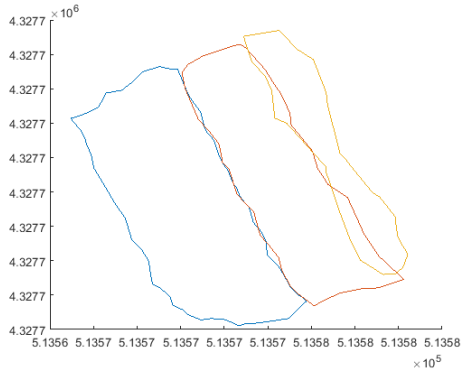
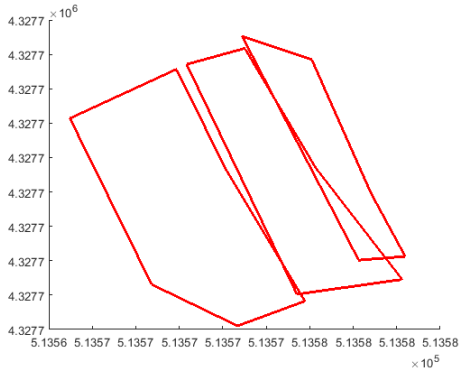
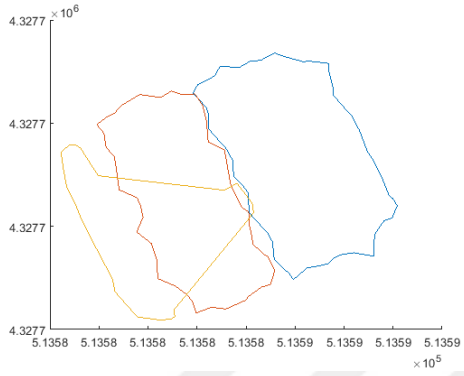
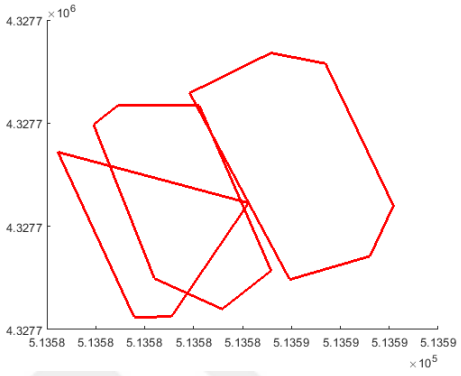
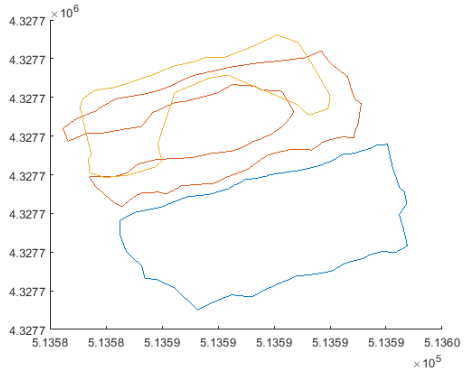
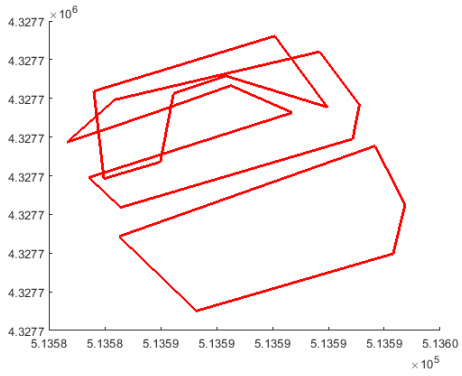
#4		
#5		
#6		
#7		

#8		
#9		
#10		
#11		

#12		
#13		
#14		
#15		

#16		
#17		
#18		
#19		

#20		
#21		
#22		
#23		

#24		
#25		
#26		

ÖZGEÇMİŞ

Kimlik Bilgileri

Adı Soyadı : Murat GÜLER

Doğum Yeri : Erciş

Medeni Hali : Evli

E-posta : muratguler@hacettepe.edu.tr

Adresi : Hacettepe Üniversitesi Beytepe Kampüsü Geomatik Mühendisliği Çankaya/Ankara

Eğitim

Lise : Atatürk Anadolu Lisesi, Van, 2009.

Lisans : Yıldız Teknik Üniversitesi, Harita Mühendisliği, İstanbul, 2014.

Yüksek Lisans : -

Doktora : -

Yabancı Dil ve Düzeyi

İngilizce (iyi)

İş Deneyimi

Araştırma Görevlisi Hacettepe Üniversitesi 2016-

Deneyim Alanları

-

Tezden Üretilmiş Projeler ve Bütçesi

-

Tezden Üretilmiş Yayınlar

-

Tezden Üretilmiş Tebliğ ve/veya Poster Sunumu ile Katıldığı Toplantılar

-

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/DOKTORA TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
GEOMATİK MÜHENDİSLİĞİ ANABİLİM DALI BAŞKANLIĞI'NA

Tarih: 07/06/2018

Tez Başlığı : LiDAR VERİSİNDEN BİNA ÇATI DÜZLEMLERİNİN OTOMATİK ÇIKARIMI VE MODELLEMESİ

Yukarıda başlığı gösterilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler ve d) Sonuç kısımlarından oluşan toplam 77 sayfalık kısmına ilişkin, 07/06/2018 tarihinde tez danışmanım tarafından *Turnitin* adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 3 'dir.

Uygulanan filtrelemeler:

1- Kaynakça hariç
2- Alıntılar hariç~~/dâhil~~
3- 5 kelimeden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Tez Çalışması Orjinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

M. Güler
07/06/2018

Adı Soyadı:	MURAT GÜLER
Öğrenci No:	N14329035
Anabilim Dalı:	GEOMATİK MÜHENDİSLİĞİ
Programı:	GEOMATİK
Statüsü:	☒ Y.Lisans ☐ Doktora ☐ Bütünleşik Dr.

DANIŞMAN ONAYI

UYGUNDUR.

M. Türker

Prof.Dr. Mustafa TÜRKER