



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



İNSANSIZ HAVA ARAÇLARI İLE
HARİTALAMA VE YAPAY ZEKA TABANLI
NESNE TANIMA

Ahmet Furkan BÜYÜKKELEK

YÜKSEK LİSANS TEZİ

Elektrik-Elektronik Mühendisliği Anabilim Dalı

EYLÜL-2022
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Ahmet Furkan Büyükkelek tarafından hazırlanan “İnsansız Hava Araçları ile Haritalama ve Yapay Zeka Tabanlı Nesne Tanıma” adlı tez çalışması 26/09/2022 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Prof. Dr. Cemil SUNGUR

.....

Danışman

Doç. Dr. Akif DURDU

.....

Üye

Dr. Öğr. Üyesi Hakan TERZİOĞLU

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

TEZ BİLDİRİMİ

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

DECLARATION PAGE

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

Ahmet Furkan BÜYÜKKELEK

Tarih: 26/09/2022

ÖZET

YÜKSEK LİSANS TEZİ

İNSANSIZ HAVA ARAÇLARI İLE HARİTALAMA VE YAPAY ZEKA TABANLI NESNE TANIMA

Ahmet Furkan BÜYÜKKELEK

**Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalı**

Danışman: Doç. Dr. Akif DURDU

2022, 91 Sayfa

**Jüri
Prof. Dr. Cemil SUNGUR
Doç. Dr. Akif DURDU
Dr. Öğr. Üyesi Hakan TERZİOĞLU**

Gelişen teknolojiyle birlikte İnsansız Hava Araçları (İHA) ile yapılan çalışmaların sayısı her geçen gün artmaktadır. Örneğin, yangın tespiti, doğal afetler sonrası ortaya çıkan zararın tespit edilmesi ve kazazedelere hızlı ulaşılması, tarım arazilerinde; ürünlerin takip edilmesi, ilaçlanması, hastalık tespiti ve toprağın nem durumunun takip edilmesi gibi birçok alanda İHA'lar sıklıkla kullanılmaktadır. Belirtilen örneklerdeki gibi ve çeşitli uygulamalarda, hedeflenen bilgilerin çıkarılmasında GPS, IMU, LiDAR, kamera gibi sensörler hayati önem taşımakta ve çeşitli sensörlerden faydalanılmaktadır. Bu çalışma kapsamında; kamera ve GPS sensörleri kullanılarak elde edilen sıralı görüntülerden mümkün olduğunca çok bilgi çıkarılması hedeflenmiştir. Bu nedenle farklı amaçlar için kullanılabilecek üç farklı algoritmayı içeren bir yaklaşım sunulmuştur.

Yaklaşımda farklı ortamlarda çeşitli bilgilerin elde edilmesi için derin sinir ağları, coğrafi konumlandırma ve görüntü mozaikleme metodlarının birlikte kullanılması önerilmiştir. Yaklaşım ilk olarak derin öğrenme modeli ile başlamıştır. Uygulamada Faster R-CNN (Faster Region-based Convolutional Neural Network) isimli model kullanılmış ve nesne tespitindeki verimi hesaplanmıştır.

Tespit edilen nesnelerin piksel koordinatları belirlendikten sonra coğrafi konumlandırma yöntemi ile nesnelerin GPS konumları belirlenmiştir. Bu hesaplama yapılırken kameranın sürekli olarak yere doğru baktığı ve İHA'nın yüksekliğinin değişmediği varsayılmıştır.

Son olarak, belli bir sıraya göre elde edilmiş görüntüler görüntü mozaikleme yöntemi ile tek ve büyük bir görüntü haline getirilmiştir. Mozaikleme yönteminin sebep olabileceği hataların coğrafi konumlandırma ve nesne tespitinin verimini etkilememesi için önerilen yaklaşımın son aşamasında kullanılmıştır. Diğer iki yöntemin mozaikleme algoritmasının verimini etkilememesi için ise gerekli geometrik dönüşümler yapıldıktan sonra derin öğrenme ve coğrafi konumlandırma yöntemlerinin çıktısı olan görüntüler birleştirilmiştir. Böylece her bir metodun birbirini olumsuz yönde etkilemesi önlenmiştir.

Sonuç olarak bu tez çalışmasında, farklı alanlarda kullanılması için görüntülerdeki nesnelerin tespiti ve bu nesnelerin coğrafi koordinatlarının belirlenmesi ve sıralı görüntülerin birleştirilmesi yapılmıştır.

Anahtar Kelimeler: Coğrafi Konum, Faster R-CNN, Görüntü Mozaikleme, Haritalama, RANSAC, Yapay Zeka.

ABSTRACT

MS THESIS

MAPPING AND ARTIFICIAL INTELLIGENCE BASED OBJECT RECOGNITION WITH UNMANNED AERIAL VEHICLES

Ahmet Furkan BÜYÜKKELEK

**Konya Technical University
Institute of Graduate Studies
Department of Electrical and Electronics Engineering**

Advisor: Assoc. Prof. Dr. Akif DURDU

2022, 91 Pages

Jury

**Prof. Dr. Cemil SUNGUR
Assoc. Prof. Dr. Akif DURDU
Asst. Prof. Dr. Hakan TERZİOĞLU**

With the developing technology, the number of studies with Unmanned Aerial Vehicles (UAV) is increasing day by day. For example, fire detection, detection of damage after natural disasters and rapid access to victims, in agricultural lands; UAVs are frequently used in many areas such as tracking of crops, spraying, disease detection and monitoring of soil moisture. As in the examples mentioned and in various applications, sensors such as GPS, IMU, LiDAR, camera are vital in extracting the targeted information and various sensors are used. In this dissertation; It is aimed to extract as much information as possible from the sequential images obtained using camera and GPS sensors. For this reason, an approach that includes three different algorithms that can be used for different purposes is presented.

In the approach, it is proposed to use deep neural networks, geolocation and image mosaicing methods together to obtain various information in different environments. The approach first started with the deep learning model. In the application, a model named Faster R-CNN (Faster Region-based Convolutional Neural Network) was used and its efficiency in object detection was calculated.

After the pixel coordinates of the detected objects were determined, the GPS locations of the objects were determined by the geolocation method. While making this calculation, it is assumed that the camera is constantly demonstrating towards the ground and the height of the UAV does not change.

Finally, the images obtained in a certain order were turned into a single and large image with the image mosaic method. It was used in the last stage of the proposed approach so that the errors that may be caused by the mosaicking method do not affect the efficiency of geolocation and object detection. In order for the other two methods not to affect the efficiency of the mosaicing algorithm, the images that are the output of the deep learning and geolocation methods were combined after the necessary geometric transformations were made. Thus, each method has been prevented from affecting each other negatively.

In conclusion, in this dissertation study, the detection of objects in the images, the determination of the geographical coordinates of these objects and the merging of sequential images were made for use in different areas.

Keywords: Geolocation, Faster R-CNN, Image Mosaicing, Mapping, RANSAC, Artificial Intelligence.

ÖNSÖZ

İnsansız hava araçları ile haritalama ve yapay zekâ tabanlı nesne tanıma adlı tez çalışmasında kamera görüntülerinin ve GPS gibi çeşitli sensörlerin yardımıyla ortamın geniş ve birleştirilmiş bir görüntüsü, nesne tespiti ve nesnelerin coğrafi koordinatının belirlenmesi gibi mümkün olduğunca çok bilginin çıkarılması amaçlanmıştır.

İlk olarak, her zaman yanımda olan ve desteklerini benden hiçbir zaman esirgemeyen aileme teşekkür ederim.

Bu çalışma sürecinde her türlü yol gösterici olan, bilgi birikimiyle çalışmama farklı açılardan bakmamı sağlayan ve beraber çalışmaktan gurur duyduğum danışman hocam Doç. Dr. Akif DURDU'ya teşekkürlerimi sunarım.

Ayrıca bu çalışmanın gerçekleştirilmesindeki destek ve katkılarından dolayı Arş. Gör. Dr. Hasan Bilgehan MAKİNECİ'ye teşekkürü bir borç bilirim.

Çalışmanın uygulama sürecinde yardımlarını esirgemeyen arkadaşlarıma sonsuz teşekkür ederim.

Ahmet Furkan BÜYÜKKELEK
KONYA-2022

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR	ix
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMASI	3
3. MATERYAL VE YÖNTEM.....	22
3.1. Yapay Sinir Ağları	23
3.1.1. Tek katmanlı sinir ağı	24
3.1.2. Çok katmanlı sinir ağı.....	26
3.2. Derin Öğrenme	27
3.3. Evrişimli Sinir Ağı (CNN).....	28
3.3.1. Konvolüsyon katmanı	28
3.3.2. Havuzlama katmanı	29
3.3.3. Tam bağlı katman	30
3.4. Faster R-CNN	30
3.5. Coğrafi Konumlandırma	31
3.5.1. Küresel konumlandırma sistemi (GPS)	32
3.5.1.1. Uzay bölümü.....	33
3.5.1.2. Kontrol bölümü.....	33
3.5.1.3. Kullanıcı bölümü	34
3.5.2. Dünya jeodezi sistemi 1984 (WGS84)	35
3.5.3. WGS84 koordinatından ECEF koordinatına dönüşüm.....	36
3.5.4. ECEF koordinatından ENU koordinatına dönüşüm	37
3.5.5. Piksel başına düşen metre hesabı.....	38
3.5.6. Yerel Koordinatlardaki değişimin hesaplanması.....	39
3.5.7. ENU koordinatından ECEF koordinatına dönüşüm	40
3.5.8. ECEF koordinatından WGS84 koordinatına dönüşüm.....	41
3.6. Görüntü Mozaikleme	42
3.6.1. Özellik çıkarma.....	42
3.6.1.1. SIFT algoritması	42
3.6.1.1.1. Ölçek uzay aşırılık tespiti	43
3.6.1.1.2. Kilit nokta yerelleştirme	44
3.6.1.1.3. Yönlendirme ataması	45
3.6.1.1.4. Anahtar nokta tanımlayıcı.....	45
3.6.1.2. SURF algoritması	46
3.6.1.3. BRIEF algoritması	48
3.6.1.4. ORB algoritması	51

3.6.2. Özellik eşleştirme	52
3.6.3. RANSAC (RANDOM SAMPLE CONSENSUS).....	53
3.6.4. Geometrik dönüşümler	54
3.6.4.1. Perspektif dönüşüm.....	54
4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....	55
5. SONUÇLAR VE ÖNERİLER.....	70
5.1 Sonuçlar	70
5.2 Öneriler	71
KAYNAKLAR	73



SİMGELER VE KISALTMALAR

Simgeler

a	: Elipsoidin ekvator yarı çapı (m)
b	: Elipsoidin kutup yarı çapı (m)
b_{ysa}	: Bias değeri
c	: Sinyal hızı
C	: Görüntü bloğunun ağırlık merkezi
C_x	: Tespit edilen nesnenin x eksenindeki piksel koordinatı
C_y	: Tespit edilen nesnenin y eksenindeki piksel koordinatı
d	: Hedef konumun uyduya olan uzaklığı
d_x	: Kamera sensörünün yataydaki boyutu
d_y	: Kamera sensörünün dikeydeki boyutu
D	: DoG değeri
D_{xx}	: X yönünde Gauss'un ikinci dereceden türevi
D_{xy}	: XY yönünde Gauss'un ikinci dereceden türevi
D_{yy}	: Y yönünde Gauss'un ikinci dereceden türevi
$D(x)$	: İkinci dereceden Taylor serisi açılımı
$D(x, y, \sigma)$	: Gauss farkı alınan görüntü
$Det(H)$	: Hessian matrisinin determinantı
e	: Dış merkezlilik
e'	: İkinci dış merkezlilik
f	: Kameranın odak uzaklığı
f_a	: Aktivasyon fonksiyonu
f_{wgs84}	: Elipsoidin basıklığı
FOV_x	: Kameranın x eksenindeki görüş alanı
FOV_y	: Kameranın y eksenindeki görüş alanı
$G(x, y, \sigma)$	: Gauss Fonksiyonu
$G(x, y, k\sigma)$	: Farklı derecelerde Gauss fonksiyonu
h	: Hava aracının yüksekliği
H	: Hessian matris
h_{11}, h_{12}	: Görüntünün döndürülme ve ölçeklenme değerleri
h_{21}, h_{22}	: Görüntünün diğer döndürülme ve ölçeklenme değerleri
h_{13}, h_{23}	: Görüntünün taşınma değerleri
h_{31}, h_{32}	: Görüntünün projeksiyon değerleri
h_{33}	: 1 veya 0
$H(x, \sigma)$	: Hessian matris
I_x	: Özellik noktasının yatay yöndeki değişimi
I_y	: Özellik noktasının dikey yöndeki değişimi
$I(x, y)$	: Görüntü
$I_{\Sigma}(x)$	: İntegrali alınan görüntü
k	: 0,04 ile 0,06 arasında değer alan parametre
$L(x, y)$	: Gauss fitresi uygulanan görüntü
$L(x, y, k\sigma)$	: Farklı derecelerde Gauss işlemi uygulanan görüntü
M	: 2x2 matris
m_{pq}	: Görüntü bloğunun momenti ($p, q = \{0,1\}$)
m_{00}	: Görüntü bloğunun kütlesi

m_{10}	: Görüntü bloğunun yataydaki ağırlık merkezi
m_{01}	: Görüntü bloğunun dikeydeki ağırlık merkezi
MPP_x	: Kameranin x eksenindeki piksel başına düşen metre
MPP_y	: Kameranin y eksenindeki piksel başına düşen metre
$m(x, y)$	: Gradyan büyüklüğü
N	: Rastgele örnekleme sayısı
O_x	: Görüntü merkezinin x eksenindeki piksel koordinatı
O_y	: Görüntü merkezinin y eksenindeki piksel koordinatı
p	: İstenilen başarı oranı
$p(x)$	: X noktasındaki pikselin yoğunluğu (değeri)
$p(y)$	: Y noktasındaki pikselin yoğunluğu (değeri)
PD_x	: Görüntü merkezi ile nesnenin merkezi arasındaki yatay piksel mesafesi
PD_y	: Görüntü merkezi ile nesnenin merkezi arasındaki dikey piksel mesafesi
PD_h	: Görüntü merkezi ile nesnenin merkezi arasındaki direkt piksel mesafesi
r	: Maksimum özdeğer (λ_1) ile minimum özdeğerin (λ_2) oranı
r_x	: Kameranin x eksenindeki piksel sayısı
r_y	: Kameranin y eksenindeki piksel sayısı
q	: Veride yer alan aykırı veri oranı
R	: Eğrilik yarı çapı
R_h	: Harris yanıt değeri
S	: Kullanılacak veri sayısı
$\tau(p; x, y)$	: İkili test değeri
$TR(H)$	: Hessian matrisinin izi (diyagonal mesafe)
w_i	: Ağırlık parametresi
$w(x, y)$	: Görüntü penceresi
x	: Orijinden (0,0) yatay eksen boyunca olan mesafe
x	: Görüntünün merkezi x koordinatı
x'	: koordinat bilgileri değişen görüntünün yeni x koordinatı
X_{ecef}	: Ekvator ve başlangıç meridyenin kesişiminden geçen eksen
X_{enu}	: Doğu yönündeki mesafe
x_i	: Giriş değerleri
X_p	: X eksenindeki görüş alanının yarısı
X_r	: Konum hesabını yapmak için x ekseninde referans alınan nokta
X_0	: X ekseninde önceden hesaplanmış ECEF koordinatı
y_e	: Nöronun çıkış değeri
$xEast$	: Doğu batı eksenindeki mesafe değişimi
Y_{ecef}	: X ve Z eksenlerine dik olan eksen
Y_{enu}	: Kuzey yönündeki mesafe
Y_p	: Y eksenindeki görüş alanının yarısı
Y_r	: Konum hesabını yapmak için y ekseninde referans alınan nokta
Y_0	: Y ekseninde önceden hesaplanmış ECEF koordinatı
y	: Orijinden (0,0) dikey eksen boyunca olan mesafe
y	: Görüntünün merkezi y koordinatı
y'	: Koordinat bilgileri değişen görüntünün yeni y koordinatı
$yNorth$	: Kuzey güney eksenindeki mesafe değişimi
Z_{ecef}	: Kuzey kutbundan geçen eksen
Z_{enu}	: Dünya merkezinden dışarı olan yöndeki mesafe
Z_r	: Konum hesabını yapmak için z ekseninde referans alınan nokta

z_0	: Z ekseninde önceden hesaplanmış ECEF koordinatı
zUp	: Yükseklikteki değişim
Δt	: Sinyalin iletilme zamanı ile GPS tarafından alınma zamanı farkı
α	: Tespit edilen nesnenin görüntü düzleminde y eksenine göre açısı
α_0	: Tespit edilen nesnenin yaw açısı
α_x	: Kameranın x eksenindeki görüş açısının yarısı
α_y	: Kameranın y eksenindeki görüş açısının yarısı
θ	: Özellik noktasının yönü
$\theta(x, y)$	: Gradyan yönelimi
ϕ	: Enlem
λ	: Boylam
λ_1	: Maksimum özdeğer
λ_2	: Minimum özdeğer
σ	: Gauss çekirdeğinin standart sapması
ψ	: İnsansız Hava Aracının yaw açısı

Kısaltmalar

AANAP	: Adaptive As-Natural-As-Possible
ABD	: Amerika Birleşik Devletleri
APAP	: As-projective-as-possible
AT	: Aerial Triangulation
BEV	: The Birds-Eye-View
BIMOS	: Binary descriptor-based Image MOSaicing
BRIEF	: Binary Robust Independent Elementary Features
C/A	: Coarse Acquisition
CHM	: Canopy Height Model
CNN	: Convolutional Neural Network
CPB	: Convolution Pyramid Blending
CPU	: Central Processing Unit
CUDA	: Compute Unified Device Architecture
DLT	: Direct Linear Transformation
DN	: Doğru Negatif
DoG	: Difference of Gaussian
DP	: Doğru Pozitif
DSM	: Digital Surface Model
DTM	: Digital Terrain Model
ECEF	: Earth-centered, Earth-fixed (Geocentric)
ENU	: East, North, Up
EO	: Exterior Orientation
FAST	: Features from Accelerated Segment Test
FB	: Feather Blending
fps	: Frame Per Second
GNDI	: Green Difference Vegetation Index
GPS	: Global Positioning System
GPU	: Graphic Processing Unit
İHA	: İnsansız Hava Aracı
IMU	: Inertial Measurement Unit
INS	: Inertial Navigation System
KCF	: Kernelized Correlation Filter

LiDAR	: Laser Imaging Detection and Ranging
LKT	: Lucas-Kanade feature Tracker
LoG	: Laplacian of Gaussian
LTP	: Local Tangent Plane
mAP	: mean Average Precision
MBB	: Multi-band Blending
MGRAPH	: Multi-GRAPh Homography-based
MPB	: Modified Poisson Blending
MSB	: Multi-Spline Blending
MVCB	: Mean-Value Coordinates Blending
NAVSTAR	: Navigation System with Timing and Ranging
NDRE	: Normalized Difference Red Edge
NDVI	: Normalized difference vegetation index
NGA	: National Geospatial-Intelligence Agency
NIR	: Near-infrared
NNs	: Neural Networks
OpenCV	: Open Source Computer Vision Library
ORB	: Oriented FAST and Rotated BRIEF
P	: Precise
PPK	: Post Processing Kinematic
RANSAC	: RANdom SAMple Consensus
R-CNN	: Region Based Convolutional Neural Network
ReLU	: Rectified Linear Unit
REW	: Robust Elastic Warping
RGB	: Red, Green, Blue
RGFD	: Regional Gray Fitting Degree
RLS	: Recursive Least Square Estimation
RMS	: Root mean square
RNN	: Recurrent Neural Network
RPN	: Region Proposal Network
RTK-DGPS	: Real-Time Kinematic-differential global positioning system
ROS	: Robot Operating System
SAT	: Separating Axis Theorem
SFM	: Structure From Motion
SIFT	: Scale Invariant Feature Transform
SPHP	: Shape-Preserving Half-Projective
SSD	: Single Shot Multibox Detector
SURF	: Speeded Up Robust Features
SUSAN	: Small Univalue Segment Assimilating Nucleus
SVD	: Singular Value Decomposition
UMCD	: UAV Mosaicking and Change Detection
UTM	: Universal Transverse Mercator
VCP	: Virtual Control Point
WGS84	: World Geodetic System 1984
YN	: Yanlış Negatif
YP	: Yanlış Pozitif
YSA	: Yapay Sinir Ağları
YOLO	: You Only Look Once
YOLOv2	: You Only Look Once, Version 2
YOLOv3	: You Only Look Once, Version 3

YOLOv4 : You Only Look Once, Version 4
2B : İki boyutlu
3B : Üç boyutlu



1. GİRİŞ

Gelişen teknolojiyle birlikte İnsansız Hava Araçları (İHA) ile yapılan çalışmaların sayısı her geçen gün artmaktadır. Örneğin, yangın tespiti, doğal afetler sonrası ortaya çıkan zararın tespit edilmesi ve kazazedelere hızlı ulaşılması, tarım arazilerinde; ürünlerin takip edilmesi, ilaçlanması, hastalık tespiti ve toprağın nem durumunun takip edilmesi gibi birçok alanda İHA'lar sıklıkla kullanılmaktadır. Belirtilen örneklerdeki gibi ve çeşitli uygulamalarda, hedeflenen bilgilerin çıkarılmasında GPS, IMU, LiDAR, kamera gibi sensörler hayati önem taşımakta ve çeşitli sensörlerden faydalanılmaktadır. Bu çalışma kapsamında; kamera ve GPS sensörleri kullanılarak elde edilen sıralı görüntülerden mümkün olduğunca çok bilgi çıkarılması hedeflenmiştir. Ayrıca, önerilen yaklaşımın farklı ortamlara uyarlanıp kullanılabilmesi, çalışmanın diğer bir amacıdır. Bu nedenle farklı amaçlar ve ortamlar için kullanılabilecek üç farklı algoritmayı içeren bir yaklaşım sunulmuştur.

Bu tez çalışması kapsamında ilk olarak İHA'lar tarafından alınan coğrafi referanslı görüntüler üzerinde Faster R-CNN derin öğrenme modeli ile nesne tespiti ve sınıflandırılması gerçekleştirilmiştir. Daha sonra tespit edilen nesneler, İHA'da bulunan GPS (Global Positioning System) ve coğrafi konumlandırma (Geolocation) tekniği kullanılarak sınıflandırılan nesnelerin konumları belirlenmiştir. Son olarak görüntü mozaikleme tekniği kullanılarak ortamın daha büyük bir haritası çıkarılmıştır. Böylece çalışma ortamı hakkında bir ön bilgiye sahip olunmuş ve görüntülerin otomatik olarak anlamlandırılması sağlanmıştır.

Önerilen yaklaşım birçok alanda kullanılabilecektir. Örneğin, askeri alanlarda saldırı öncesi keşif, gözetleme ve istihbarat çalışmaları kaçınılmazdır. Ancak saldırının gerçekleştirilebilmesi için bölgenin coğrafi durumu, muhtemel tehditler, hedefin konumu, hedefin yapısı ve hedefin sınıfı hakkında çeşitli bilgilerin elde edilmesi gerekmektedir. Bu tür bilgilerin elde edilmesi için önerilen yaklaşımdan yararlanılabilir.

Yaklaşımın uygulanabileceği alanlardan biri ise tarımdır. Bir tarım arazisinde tarlanın haritası görüntü mozaikleme yöntemi ile çıkarıldıktan sonra temel görüntü işleme uygulamalarıyla veya derin öğrenme modelleri ile yabancı ot veya hastalık tespiti yapılabilir. Tespit edilen nesneler GPS yardımıyla konumları tespit edilebilecektir.

Böylece sorunlara hızlı çözümler üretilme imkânı sağlanacak ve ürün verimi yükseltilebilecektir.

Ayrıca, robot teknolojisi çok hızlı bir şekilde gelişmekte ve robotların bir arada uyum içinde çalışması için çok fazla araştırma yapılmaktadır. Üzerinde çalışılan uygulamalardan biri ise hava destekli kara aracıdır. Önerilen yaklaşım ile ortam hakkında bilgiler çıkarılıp kara aracının yönlendirilmesi sağlanabilecektir.

Yapılan çalışma esnek olup birçok alanda kullanılabilecek bir yapıya sahiptir. Çalışmanın uygulanması için İHA'nın kamera, GPS ve IMU'ya (Inertial Measurement Unit) sahip olması yeterlidir. Daha sonra göreve göre eğitilen derin öğrenme modeli ve görüntü mozaikleme yöntemi ile ortam hakkında hızlı bir çıkarımda bulunulabilecektir. Ayrıca hedeflerin GPS koordinatları belirlenebilecektir.

Literatürde derin öğrenmeye dayalı nesne tespiti, görüntü mozaikleme ve coğrafi konumlandırmanın birlikte kullanıldığı bir çalışmaya rastlanmamıştır. Bu nedenle, gerçekleştirilen tez çalışması literatürdeki bu boşluğu doldurmaktadır.

Bu bölümden sonra tezde 4 farklı bölüm bulunmaktadır. Bölümler; kaynak araştırması, materyal ve yöntem, araştırma sonuçları ve tartışma, sonuçlar ve öneriler olarak tezde sunulmuştur. Görüntü mozaikleme, derin öğrenme ve coğrafi konumlandırma konuları hakkında dünya çapında yapılmış çalışmalar kaynak araştırması kısmında verilmiştir. Ortam hakkında mümkün olduğunca fazla bilgi çıkarmak için kullanılan yöntemler detaylı olarak materyal ve yöntem bölümünde anlatılmıştır. Uygulanan yöntem DJI Phantom 4 Pro ile toplanan veri kümesinde test edilmiş, araştırma sonuçları ve tartışma kısmında analiz edilmiştir. Uygulanan yaklaşım ile ilgili genel değerlendirmeler sonuçlar ve öneriler kısmında verilmiştir.

2. KAYNAK ARAŞTIRMASI

Bu tez çalışmasında farklı alanlarda uygulanabilecek üç farklı algoritmanın birlikte kullanımı ile ilgili bir yaklaşımın sunumu gerçekleştirilmiştir. Bu nedenle literatür taraması kısmında öncelikle derin öğrenme, daha sonra sırasıyla coğrafi konumlandırma ve görüntü mozaikleme ile ilgili yapılmış bazı çalışmalara yer verilmiştir.

Derin öğrenme ağına dayalı yaklaşımların geçmişi 1940'lara dayanmaktadır. Ancak büyük ölçekli verilerin olmaması, sınırlama hesaplama gücüne sahip olunması ve yetersiz performans nedeniyle sinir ağlarının modası 2000'lerin başında geçmiştir (Zhao ve ark., 2019). Sonraki süreçte ImageNet gibi büyük ölçekli eğitim verilerin ortaya çıkması, paralel bilgi işleme teknolojilerinin hızlı gelişimi ve ağ yapılarındaki önemli gelişmeler sebebiyle 2010'lu yıllarda tekrardan popüler hale gelmiştir. Sonraki zaman diliminde ise derin öğrenme ağları üzerinde çok fazla çalışma yapılmıştır. Bu çalışmalardan birini Eggert ve arkadaşları 2017 yılında Faster R-CNN kullanarak gerçekleştirmiştir. Eggert ve ark. (2017) bu çalışmada, şirket logosu algılama amacıyla; bölge önerileri oluşturan ve bunların sınıflandırılmasını tek bir ardışık düzende birleştiren, nesne algılama için iyi bilinen Faster R-CNN modelini uygulamıştır. Faster R-CNN'nin küçük nesne örnekleri üzerindeki zayıf performansından yola çıkılarak, çok çeşitli nesne boyutları için hem teklif hem de sınıflandırma aşamasının ayrıntılı bir incelemesi yapılmıştır. Ek olarak, özellik haritası çözünürlüğünün, bu aşamalar üzerindeki performans etkisi incelenmiş ve küçük nesneler için daha yüksek çözünürlüklü özellik haritalarından yararlanan Faster R-CNN'de bir değişiklik önerilmiştir. Nesne örneklerinde algılama performansını iyileştiren yaklaşım Flickr veri kümesinde test edilmiştir.

Bu çalışmalardan biri ise küçük nesnelerin tespiti için yapılmıştır. Bilindiği gibi küçük nesnelerin tespiti daha zordur ve ağın verimini etkilemektedir. Bu nedenle, Tang ve ark. (2018) derin öğrenmeye dayalı çoklu görünüm (Multi-view) nesne algılama yaklaşımını önermiştir. Regresyon modellerine dayalı klasik nesne algılama yöntemleri tanıtılmış ve küçük nesneleri algılamadaki zayıf yeteneklerinin nedenleri analiz edilmiştir. Bu yöntemlerin performansını artırmak için çoklu görüntülü nesne algılama yaklaşımı önerilmiş ve bu yaklaşımın model yapısı ve çalışma prensipleri açıklanmıştır. Ek olarak, önerilen yöntem VOC 2007 veri kümesinde test edilmiştir. Deneysel sonuçlar,

Çoklu Görüntü YOLO (You Only Look Once), Çoklu Görüntü YOLOv2 (You Only Look Once, Version 2) ve Çoklu Görüntü SSD'nin nesne alma yeteneği klasik muadillerinden sırasıyla 0,177, 0,06 ve 0,169 daha yüksek AF (average F-measure) puanları elde etmiştir. Ayrıca, algılama doğruluğu açısından zor nesneler dahil edilmediğinde çoklu görüntüleme yöntemlerinin mAP (mean average precision) puanları, klasik yöntemlere göre sırasıyla %14.3, %7.4 ve %13.1 oranında daha yüksektir.

Yapılan çalışmaların çok fazla olması ve literatürdeki ilerlemeye daha geniş bir açıyla bakmak için Chauhan ve Singh 2018 yılında, geleneksel makine öğrenimi ile derin öğrenme arasındaki farkları incelemiştir. Burada SVM (Support vector machine), PCA (Principal component analysis), LDA (Linear discriminative analysis), karar ağaçları, NNs (Neural Networks) vb. gibi farklı makine öğrenimi yaklaşımları arasında karşılaştırmalı bir çalışma yapılmıştır (Chauhan ve Singh, 2018). Tüm Makine öğrenimi yaklaşımları her uygulama için iyi sonuç vermemektedir. Dolayısıyla bu yaklaşımların performansı uygulamaya göre farklılık gösterdiği ifade edilmiştir. Sonuç olarak, makine öğreniminin bir uzantısı olan derin öğrenme, bir ağı eğitmek için az sayıda verinin yeterli olduğu makine öğrenimi yöntemlerinin aksine, verimli bir ağı eğitmek için milyonlarca veri gerektiren CNN ve RNN (Recurrent Neural Network) gibi derin ağlar ve bu yaklaşımların gelişimi incelenmiştir.

Derin öğrenme tabanlı nesne algılama, karmaşık problemlerin çözümü için son yıllarda önemli bir araştırma konusu olmuştur. Bu sebeple Zhao ve ark. (2019), R-CNN'de farklı derecelerde modifikasyonlarla tıkanıklık, dağınıklık ve düşük çözünürlük gibi farklı alt sorunları ele alan derin öğrenme tabanlı nesne algılama çerçeveleri hakkında ayrıntılı bir inceleme sağlamıştır. Araştırma, diğer ilgili görevler için temel mimariler sağlayan genel nesne algılama ardışık düzenlerinde başlamıştır. Ardından, göze çarpan nesne algılama, yüz algılama ve yaya algılama gibi diğer üç ortak görev de kısaca gözden geçirilmiştir. Sonuç olarak nesne tespitinde kullanılan çeşitli ağ yapıları incelenmiş ve sunumu yapılmıştır.

Yüksek hassasiyetli gerçek zamanlı sistemlere duyulan ihtiyaç giderek daha acil hale gelmektedir. Yüksek doğruluğa ve verime sahip dedektörlere ulaşma nihai hedef olduğundan, araştırmacılar yeni mimari inşa etme, zengin özellikler çıkarma, iyi temsillerden yararlanma, işlem hızını iyileştirme, sıfırdan eğitim, çapasız yöntemler gibi

bir dizi konuda çalışma gerçekleştirmiştir. Jiao ve ark. (2019) bu çalışma konularında, nesne algılama modellerinin gelişme durumunu kapsamlı ve derinlemesine anlamak için mevcut tipik algılama modeli yöntemlerini analiz etmiş, PASCAL VOC, MS COCO, ImageNet, VisDrone2018 ve Open Images V5 gibi çeşitli veri kümeleri hakkında bilgiler vermiştir. Ayrıca, tek aşamalı ve iki aşamalı dedektörleri kapsayan çeşitli nesne algılama yöntemlerine kapsamlı genel bir bakış açısı sunulmuştur.

Nesnelerin görüntü üzerinde tam olarak nerede olduğu, nesne tespiti kadar önemli bir görevdir. Bu nedenle König ve ark. (2020), tespit edilen nesneleri sıkıca çerçevelemek amacıyla bir sınırlayıcı kutunun adım adım deformasyonunu gerçekleştirmiştir. Nesnelerin tespit edilebilmesi için Q-learning'den faydalanılmıştır. Yapılan çalışmada 2 aşamalı bir model önerilmiştir. Model 253 görüntü ile eğitilmiş ve 159 farklı görüntüde test edilmiştir. Sınırlamanın en boy oranını değiştiren eylemler ve her yakınlaştırma adımından sonra sınırlayıcı kutuları hassaslaştırma işlemleri olmadan önerilen model, %75 daha fazla nesneyi algılayabilmiştir.

Uzaktan algılama görüntülerinin sahne sınıflandırması, birkaç on yıllık geliştirme sürecinde büyük gelişmeler elde etmiştir. Uzaktan algılama görüntüleri sahne sınıflandırmasıyla ilgili makalelerin sayısı, özellikle derin öğrenmeye dayalı yöntemlerle ilgili literatür çalışmaları çok fazladır. Bu nedenle Cheng ve ark. (2020), sahne sınıflandırmasındaki hızlı ilerleme oranını hesaba katarak, ilk olarak, mevcut uzaktan algılama görüntü sahne sınıflandırma alanının karşı karşıya olduğu temel zorlukları tartışmıştır. Ardından, üç tür derin öğrenme tabanlı yöntemi ayrıntılı olarak incelenmiş ve popüler sahne sınıflandırma kıyaslamalarını tanıtmıştır. Daha sonra, derin öğrenme tabanlı yöntemlerin performansını ölçmek için yaygın olarak kullanılan üç veri kümesine göre tablo şeklinde özetlenmiş ve ayrıca sonuçların analizi sağlanmıştır.

Derin öğrenmeye dayalı yaklaşımlar otonom sistemlerin geliştirilmesinde sıklıkla tercih edilmektedir. Bu tür sistemlerin geliştirilmesi için Imad ve ark. (2021), LiDAR nokta bulutunu kullanan gerçek zamanlı bir 3B (3 boyutlu) nesne algılama yaklaşımı sunmuştur. Önerilen yöntem, hesaplama verimliliği için girdi olarak the birds-eye-view (BEV) kullanılmıştır. Yaklaşımın genel süreci, ön işleme modüllerinin LiDAR verilerini aldığı ve onu BEV bir görüntüye dönüştürdüğü uçtan uca bir yöntem olarak tanımlanmıştır. Derin öğrenme modülü, BEV'i almış ve son işleme modülünde 3B

sınırlayıcı hacimlere dönüşen bölümlere ayrılmış bir ısı haritası (heat map) çıkarılmıştır. Önerilen yaklaşım, 30 fps (frame per second) üzerinde ve etkin doğrulukla çalışmıştır. Bu sonuçlar, transfer öğrenmesi adı verilen derin öğrenme tekniğinin tanıtılmasıyla elde edilmiştir. Önde gelen yaklaşımların çoğundan farklı olarak, eğitim için büyük bir veri kümesine ihtiyaç olunmadığı vurgulanmıştır.

Bilgisayarlı görüde, derin evrişimli sinir ağlarının (CNN) hızlı gelişimi ile nesne algılamada önemli ilerlemeler kaydedilmiştir. Bu nedenle, Liu ve ark. (2021) yaptığı çalışmada, küçük nesne algılama için yakın zamanda geliştirilen derin öğrenme yöntemlerinin kapsamlı bir incelemesini sunmuştur. Ayrıca, özellik haritalarını birleştirme, bağlam bilgisi ekleme, ön plan-arka plan örneklerini dengeleme ve yeterli olumlu örnek oluşturma gibi önemli derin öğrenme teknikleri anlatılmıştır. Genel nesne algılama, yüz algılama, hava görüntülerinde nesne algılama ve segmentasyon dahil olmak üzere dört araştırma alanında geliştirilen ilgili teknikler ayrıntılı olarak tartışılmıştır. Ek olarak, bu makale, küçük nesnelerin üç büyük kıyaslama veri kümesine dayalı olarak YOLOv3, Faster R-CNN ve SSD dahil olmak üzere küçük nesne algılama için önde gelen birkaç derin öğrenme yönteminin performansları karşılaştırılmıştır. Deneysel sonuçlar, bu derin öğrenme yöntemleriyle küçük nesneler üzerindeki algılama doğruluğunun 0,4'ten daha düşük olmasına rağmen, Faster R-CNN en iyi performansı kaydetmiş, YOLOv3 ise ikinci sırada yer almıştır.

Bu kısma kadar derin öğrenmeye dayalı olarak literatürde yapılmış çalışmalardan bazıları ayrıntılı olarak verilmiştir. Literatür taramasının devamında ilk olarak coğrafi konumlandırmada yararlanılan doğrudan coğrafi referanslama yöntemi ile ilgili birkaç çalışma verilmiştir. Doğrudan coğrafi referanslama yöntemi, elde edilen görüntülerin coğrafi koordinatlarının GPS ve IMU gibi sensörler ile ifade edilmesi olarak tanımlanabilir. Chiang ve ark. (2012) bu konudaki çalışmasında gerçek zamana yakın konumsal bilgi toplamak için hızlı ve düşük maliyetli bir İHA geliştirmiştir. Konum bilgilerini toplamak için doğrudan coğrafi referanslama yöntemi kullanılmıştır. Kamera görüntüsündeki bozulmaları gidermek için kamera kalibrasyonu yapılmış ve düzeltme kolu uygulanmıştır. 300 m yükseklikte uçuş gerçekleştiren hava aracının yataydaki konumlandırma hatası 5 m, dikeydeki konumlandırma hatası ise 10 m olarak hesaplanmıştır. Roll, pitch ve yaw açılarındaki hata payı ise sırasıyla 0,02, 0,74 ve 0,31 derecedir.

Bu konu üzerindeki bir diğer çalışma Zhao ve arkadaşları tarafından 2014 yılında yapılmıştır. Çalışmada, ECEF (Earth-centered, Earth-fixed), ENU ve harita projeksiyon pencerelerinde görüntülerin doğrudan coğrafi konumlandırılması test edilmiştir. Yapılan çalışmada eğik kamera (oblique camera) kullanılmış olup toplamda 5 farklı açıda görüntüleme yapabilmektedir. Eğimli duran görüntüler ile dikey görüntüler arasında 35 derece fark bulunmaktadır. Konumlandırma için harita projeksiyon penceresinin basit olduğu ifade edilmiştir. Ancak daha doğru konumlandırma için ECEF ve ENU koordinat sistemlerinin kullanılması önerilmiştir.

Nesnelerin coğrafi koordinatlarının tespit edilmesi ve konumlandırılması önemli araştırma konularından biri olmuştur. Bu nedenle Wang ve ark. (2017), İHA elektro-optik stabilize görüntüleme sistemlerinin keşif verimliliğini artırmak için çok hedefli bir yer belirleme sistemi önermiştir. İlk olarak, bir hedef konum modeli incelenmiştir. Daha sonra, homojen koordinat dönüşümü kullanılarak çoklu hedeflerin jeodezik koordinatları hesaplanmıştır. Hedefleri yerleştirmenin doğruluğunu arttırmak için lens bozulmalarını düzeltme yöntemi ve RLS fitresi uygulanmıştır. 1140 m yükseklikten alınan görüntüler üzerinde lens bozulmalarını gideren yöntem uygulandığında dairesel hata olasılığı %7 oranında azalmıştır. RLS fitresi kullanıldığında ise dairesel hata olasılığı %25 oranında azalmıştır.

Nesnelerin konumlandırılması için yapılan çalışmalardan biri de Pai ve Naidu'ya (2017) aittir. Yapılan çalışmada, bir hedefin etrafında dönen insansız hava aracına monte edilmiş bir kameradan alınan çoklu görüntüye dayalı olarak bir hedefin coğrafi konumunu tahmin etmeye yönelik bir simülasyon çalışması sunulmuştur. Uygulamada kameradan alınan görüntü pikselleri 3 boyutlu dünya koordinatları ile ilişkilendirilmiştir. Konumlandırma hatasını azaltmak için ortalama tahmin (Average estimation), hareketli ortalama tahmin (Moving average estimation) ve RLS (Recursive Least Square Estimation) yöntemleri kullanılmıştır. RLS tekniğinin 0,8 m hata payı ile daha güvenilir olduğu tespit edilmiştir.

2018 yılında Bhattarai ve arkadaşlarının yürüttüğü çalışmada; uçuş yüksekliği, hesaplama süresi ve görüntüleme ölçeği gibi kısıtlamaların bulunduğu hava araçlarında insan tespiti amaçlanmıştır. İnsan tespiti için Haar-like özelliklerinde faydalanılmış ve insanların konumları belirlenmiştir (Bhattarai ve ark., 2018). Sınıflandırıcıların eğitimi

için INRIA veri kümesinden yararlanılmış ve DJI Matrice 100 hava aracı ile çeşitli yüksekliklerde görüntüler alınmıştır. İnsanların konumlandırılması yaklaşık 4,5 metrelik hata ile yapılmış ve Google Earth üzerinde görselleştirilmiştir. Haar sınıflandırıcıları ile insanların tespiti için optimum ölçek değerinin 1.2 olduğu belirlenmiştir. F1 skoru 0,90 olarak bulunmuştur.

Her geçen yılda daha yüksek hızda görüntü işleyebilen teknolojilerin ortaya çıkması ve daha verimli algoritmaların geliştirilmesi ile farklı algoritmaların bir arada kullanılmasına yönelik çalışmalar gerçekleştirilmiştir. Zhao ve arkadaşları 2019 yılında bu tür bir çalışma yapmıştır. Çalışmada, hava platformuna dayalı olarak hareket eden aracın tespiti, takibi ve coğrafi konumu belirlenmiştir. Önerilen yaklaşım 3 adıma ayrılmıştır. İlk adımda YOLOv3 (You Only Look Once, Version 3), Faster R-CNN (Region Based Convolutional Neural Network) ve SSD (Single Shot Multibox Detector) gibi modeller ayrı ayrı kullanılarak aralarında kıyaslanmış ve araçlar tespit edilip piksel koordinatı alınmıştır. İkinci adımda araçların coğrafi konumlarını belirlemek için ENU (East, North, Up) koordinat sisteminden yararlanılmıştır. Araçların görsel takibi için kernelized correlation filter (KCF) algoritması kullanılmıştır. Son adımda ise araçların takip edilebilmesi için aracın hareketi analiz edilerek İHA'nın uçuş kontrol parametreleri ayarlanmıştır. Algoritmanın test edilebilmesi için Jetson Tx1 entegreli DJ Matrice 100 hava platformu ile 1920×1080 piksel boyutlarında 1000'den fazla görüntü toplanmıştır. YOLOv3 araçların tespit edilmesi için %90.61 ile en iyi performansı göstermiştir. Araçların konumlandırması ise 5 metrelik bir hata payı ile yapılmıştır (Zhao ve ark., 2019).

Farklı algoritmaların bir arada kullanımını içeren bir diğer çalışma ise 2020 yılında Cai ve arkadaşları tarafından yürütülmüştür. Yapılan bu çalışmada, binalardaki hedefler için coğrafi konum algoritması önerilmiştir. Algoritmada, binaların tespit edilmesi için YOLOv4 (You Only Look Once, Version 4), kullanılmıştır. Ayrıca, bina hedeflerinin yüksek doğrulukta konumlandırılmasını sağlamak için geleneksel yöntemlerin doğrusal denklemi (collinear equation) kullandığı ifade edilmiş ve tahmin kutusunun konumuna dayalı bir bina yüksekliği tahmin modeli sunulmuştur. Uçuş testleri, bu çalışmada önerilen algoritmanın, hedef yüksekliğindeki farka bağlı olarak bina hedeflerini konumlandırma doğruluğunu yaklaşık %20 ile %50 arasındaki bir oranda iyileştirebileceğini göstermiştir (Cai ve ark., 2020).

Literatür taramasının bu kısmına kadar derin öğrenme, kamera ve GPS dayalı coğrafi konumlandırma algoritmaları ile ilgili çalışmaların incelenmesi gerçekleştirilmiştir. Literatür taramasının devamında farklı alanlarda uygulanabilen ve mümkün olduğunca fazla bilgi çıkarımını sağlayabilmek için önerilen üçlü algoritmanın son aşaması olan görüntü mozaikleme algoritması ile ilgili literatürde yapılmış çalışmalara yer verilmiştir.

Görüntü mozaikleme elde edilen görüntülerin birleştirilerek tek ve büyük bir görüntünün oluşturulmasını içermektedir. Geçmişten günümüze bu konuda çok sayıda çalışma yapılmıştır. Geçmişte yapılmış bu çalışmaların en önemlilerinde biri 2007 yılında Brown ve Lowe tarafından gerçekleştirilmiştir. Çalışmada, otomatik mozaikleme yapmak için kullanılan yöntemler ayrıntılı olarak ele alınmıştır. Ayrıca, birçok çalışma için referans olmuştur. Otomatik mozaikleme, otomatik görüntü eşleştirme problemi olarak incelenmiştir (Brown ve Lowe, 2007). Sırasız olan görüntüler arasında ilişki kurabilmek için değişmeyen yerel özelliklerden faydalanılmıştır. En iyi homografi tahminini elde etmek için RANSAC algoritması kullanılmıştır. Kamera parametrelerinin hepsini çözebilmek için bundle adjustment algoritmasından yararlanılmıştır. Görüntü hizalamadan ve ışıklandırma koşullarından kaynaklanan görüntülerdeki bozulmaları gidermek için kazanç kompanzasyonu (Gain Compensation) ve çoklu band harmanlama (Multi-Band Blending) yöntemleri sunulmuştur. Sonuç olarak otomatik görüntü dikişi için önemli bir çalışma yapılmış ve C++ yazılım dilinde uygulama geliştirilmiştir.

Görüntü mozaikleme algoritması için birçok çalışma yapılmıştır. Bu çalışmalardan biri de doğal afetler için gerçekleştirilmiştir. Bilindiği gibi Felaket zamanlarında havadan görüntüler elde etmek önemlidir. Bu nedenle, Huang ve ark. (2008), görüntü mozaiklemeye kullandıkları yöntemin sunumunu yapmıştır. Çalışma, görüntü mozaikleme yöntemini üç adımda ele almıştır. İlk adımda, SUSAN (Small Univalued Segment Assimilating Nucleus) algoritması kullanılarak özellik tespiti yapılmıştır. İkinci adımda, bulunan özellik noktaları eşleştirilmiş ve yanlış eşleştirilen özellik noktalarını elemek için RGFD (Regional Gray Fitting Degree) yöntemi önerilmiştir. Son adımda dönüşüm parametrelerinin hesabı için 2B (2 boyutlu) affine dönüşümünden yararlanılmıştır. Bilinmeyen bazı parametrelerin hesabı için en küçük kareler (The Least Squares) yöntemi kullanılmıştır. Üç görüntünün mozaikleme yapılmıştır. Görüntüler başarılı bir şekilde mozaiklenmiştir.

Görüntü mozaikleme algoritmaları kullanılan yaklaşıma göre çok zaman alabilmektedir. Bu nedenle, Zhou (2010), hızlı yanıt verme gereksinimini karşılayabilecek küçük ve düşük maliyetli bir İHA platformundan alınan görüntülerin GPS yardımıyla coğrafi referanslama yönteminin sunumunu gerçekleştirmiştir. Aynı zamanda matematiksel bir model geliştirilmeye çalışılmıştır. Piksel noktalarının coğrafi referansının daha hassas bulunabilmesi için kameranın iç parametreleri ayarlanmıştır. Kalibrasyon ayarı yapıldıktan sonra videodan elde edilen görüntülerin dikilmesiyle iki boyutlu harita elde edilmiştir. Piksel noktalarının coğrafi referansı 43 farklı noktadan alınan örnekler ile kıyaslanmış ve 1-2 metrelik bir sapma olduğu görülmüştür (Zhou, 2010).

Yer düzlemin engebeli oluşu ve kullanılan ekipmanların hassasiyeti gibi birçok etken mozaikleme algoritmasının verimini etkilemektedir. Yahyanejad ve arkadaşları ise 2011 yılında, ortorektifiye edilmiş ve artımlı görüntü mozaikleme problemini ele almıştır. Görüntü mozaiklemedeki en önemli hataların zayıf bir homografi, kamera kalibrasyonu ve projeksiyon modelinden kaynaklandığı belirtilmiştir. Geometrik bozulma düzeltilmesinde daha yüksek polinom derecelerinin kullanılmasının etkisi de analiz edilmiştir. Yöntem %60 oranında eşleşen 27 sıralı görüntü üzerinde test edilmiştir. Sonuç olarak önerilen yöntem doğru bir mozaikleme için uygun yer kontrol noktalarının seçimine yardımcı olmaktadır (Yahyanejad ve ark., 2011).

Elde edilen görüntülerin hızlı mozaiklenmesi için bir başka çalışma 2011 yılında Liu ve arkadaşları tarafından yapılmıştır. Bu çalışmada, GPS, INS (Inertial Navigation System), ve kamera sistemi ile toplanan İHA görüntülerinin hızlı işlenmesi için bir yaklaşım önerilmiştir. Yaklaşım, görüntüler arasındaki karşılık gelen noktaların belirlenmesini ve ardından görüntü birleştirmeyi gerçekleştirmek için koordinat dönüşümünün gerçekleştirilmesini içermektedir. Uygulamanın testi, 600 metre yükseklikte hem kırsal alandan hem de şehir bölgesinden alınan görüntüler ile test edilmiştir. Kırsal ortam için 36 görüntü kullanılmış, her bir görüntü için en az 2.3 piksel en fazla 9.2 piksel bulunmuştur. Şehir ortamı için 9 görüntü kullanılmış ve hata payı maksimum 7.7 piksel olarak hesaplanmıştır (Liu ve ark., 2011).

Mozaikleme algoritmaları görüntülerin birleştirilmesi için coğrafi referanslı görüntülerden de yararlanmaktadır. Bu nedenle Xiang ve Tian (2011), düşük maliyetli

IMU ve GPS sensörleri kullanılarak İHA tabanlı görüntü verisi toplama platformu için hızlı ve görüntü coğrafi referanslandırma yöntemi geliştirmeyi amaçlamıştır. Aynı zamanda coğrafi referanslı görüntüler mozaiklenmiş ve doğrulukları değerlendirilmiştir. Görüntülerin alınması için 4.5 mm ve 8 mm odak uzaklığına sahip multispektral kamera kullanılmış ve daha kısa odak uzaklığına sahip görüntülerde daha fazla bozulma tespit edilmiştir. Görüntülerdeki coğrafi referanslar RTK-DGPS (Real-Time Kinematic-differential global positioning system) ölçümleri ile kıyaslanmış, kuzey ve doğu yönü boyunca RMS ile ölçülen hata değerleri sırasıyla 31,88 cm ve 58,56 cm olarak hesaplanmıştır. Mozaiklemedeki hata payı ise ortalama 0,569 m olarak ölçülmüştür.

Park ve arkadaşları tarafından 2012 yılında yapılan çalışmada, İHA'nın otonom navigasyonu için hiyerarşik ve çok seviyeli yeni bir görüntü mozaikleme algoritması önerilmiştir. Önerilen sistem çalışırken kendini ortama göre ayarlamıştır. Sistem özellik tespiti için LKT (Lucas-Kanade feature Tracker) ve SIFT algoritmaları birlikte kullanılmıştır. Homografi tahmini ve aykırı noktaların ayıklanması için DLT (Direct Linear Transformation) ve RANSAC algoritmaları tercih edilmiştir. Yapılan testlerde işlem süresi, sağlamlık ve doğrulukta önemli bir gelişme olduğunu gösterilmiştir. Ancak SIFT algoritmasının sürekli tercih edilmesi durumunda sistemin çalışma süresi SIFT tabanlı mozaikleme yöntemine yaklaşmıştır (Park ve ark., 2012).

Bir ortam ile ilgili çekilmiş fotoğraflar üzerinden mozaikleme işlemi yapılabileceği gibi video görüntüleri üzerinden de mozaikleme gerçekleştirilebilir. 2012 yılında Eugster ve Nebiker, hafif İHA (İnsansız Hava Aracı) platformlarından yakalanan video akışının dijital bir 3B manzara modelini coğrafi olarak kaydetmeyi amaçlamıştır. GPS ve IMU verilerinin sisteme entegrasyonu dış yönlendirme parametreleri ile sağlanmıştır. Ayrıca daha doğru uçuş yükseklik verisi için birleşik coğrafi referanslama yöntemi önerilmiştir. Önerilen yaklaşımda diğer çalışmalardan farklı olarak Ayırık kalman filtresi (Discrete Kalman filter), dış yönlendirme parametreleri ile entegre edilmiştir. Yaklaşımın sonuçları doğrudan coğrafi referanslama yöntemi (direct geo-referencing) ile kıyaslanmıştır. Yükseklik bilgisi 50 ile 100 m arasındayken doğrudan coğrafi referanslamanın doğruluğu 2 ile 4 m arasındayken birleşik coğrafi referanslama (integrated geo-referencing) da 0.6 metredir (Eugster ve Nebiker, 2012).

2012 yılında görüntü mozaikleme konusunda yapılmış bir diğer çalışma ise Turner ve arkadaşları tarafından gerçekleştirilmiştir. Yapılan çalışmada, İHA fotoğrafçılığında meydana gelen geometrik bozulmalar ve mozaikleme için bir yöntem sunulmuştur. İHA'dan alınan görüntüler, başlangıçta rastgele bir model uzayında üç boyutlu nokta bulutları oluşturmak için işlenmiştir. Nokta bulutları, tahmini kamera konumlarını kullanan doğrudan bir coğrafi referans tekniği veya nokta bulutu içinde GCP tekniği kullanılarak gerçek dünya koordinat sistemine dönüştürülmüştür. Daha sonra nokta bulutu Dijital Arazi Modeli (DTM) oluşturmak için kullanılmıştır. Çalışma alanının bir mozaikini oluşturmak için ise coğrafi referanslı görüntülerden yararlanılmıştır. Sonuç olarak doğrudan coğrafi referanslama ile elde edilen mozaiklerin hata payı 65-120 cm, GCP tekniği ile yaklaşık 10-15 cm olarak hesaplanmıştır (Turner ve ark., 2012).

Buyukyazi ve arkadaşları ise 2013 yılında, İHA kullanılarak elde edilen hava görüntülerinin gerçek zamanlı stabilizasyonu ve mozaiklenmesi için yeni bir sistemi anlatmıştır. Değiştirilmiş yöntem ile SURF algoritmasının karşılaştırması yapılmıştır. Özellik çıkarmak için modifiye edilmiş SURF algoritması kullanılmış ve orijinal olan SURF algoritması ile kıyaslanmıştır. Değiştirilmiş SURF algoritması, orijinal SURF algoritmasına göre yaklaşık iki kat daha hızlı olduğu tespit edilmiştir (Buyukyazi ve ark., 2013). Ölçekleme faktörü ile bu oran 3.3 katına çıkmaktadır. Aynı zamanda orijinal SURF algoritmasına göre daha az piksel hatası yaptığı gözlenmiştir. Görüntü düzlemleri arasındaki ilişkiyi bulmak için homografi, Öklid ve benzerlik dönüşümleri test edilmiştir. Benzerlik dönüşümü tercih edilmiştir. Yapılan çalışma farklı ortam koşullarında gerçek zamanlı olarak test edilmiş ve çalışmanın uydu ile gözetleme gibi çeşitli platformlarda kolaylıkla uygulanabileceği vurgulanmıştır.

Görüntü mozaikleme yaklaşımları, elde edilen görüntüler üzerinden büyük bir topoğrafik görüntü elde etmek içinde kullanılmaktadır. Bu tip bir çalışma 2014 yılında Wang ve arkadaşları tarafından gerçekleştirilmiştir. Çalışma, İHA görüntülerini otomatik olarak mozaiklemek için yeni bir yaklaşım sunmuştur. İlk olarak temel özellik çıkarma ve eşleştirme yöntemleri uygulanmıştır. Daha sonra yerel koordinat sisteminde SFM (Structure From Motion) algoritmasına dayalı olarak kamera parametrelerinin ve özellik noktalarının 3 boyutlu hesabı yapılmıştır. Son olarak ortorektifikasyon aşamasından geçen görüntülerin mozaiklenmesi gerçekleştirilmiştir. Bu yöntemde, yer kontrol

noktalarının desteği olmadan ortamın üç boyutlu arazi modelinin çıkarılabileceği vurgulanmıştır (Wang ve ark., 2014).

Mozaikleme yönteminin kullanıldığı diğer bir alan ise tarımdır. Turner ve arkadaşlarının 2014 yılında yaptığı çalışma bunu açık bir şekilde göstermektedir. Bu çalışmada, Avustralya Houston'daki marul çiftliğinde, marul mahsullerinin sağlığını haritalamada ve izlemede, mikro İHA kullanılabilirliğini test etmek için bir çalışma yapılmıştır. Çalışmada İHA, gözetleme yapabilmek için GPS, Gimbal, IMU gibi çeşitli malzemeler ile donatılmıştır. GPS ve kamera fiziksel olarak aynı hizada kullanılmamıştır. Bu nedenle lever arm correction yöntemi uygulanmıştır. Hava aracıyla 5184×3456 piksel boyutlarında toplam 297 görüntü toplanmıştır. Bulanık görüntüleri ve aşırı örtüşen görüntüleri ortadan kaldırmak için görüntü seçim algoritmaları kullanılmış ve kullanılmak için görüntülerden 143'ü seçilmiştir. Görüntü mozaikleme için Photoscan, Pix4D web service ve Bundler method yazılımları kullanılmış ve her bir görüntü mozaikleme yazılımının çalışma süreleri sırasıyla 4,3, 11 ve 41 saat olarak hesaplanmıştır. Pix4D web tabanlı bir yazılım olduğu için resim yükleme kısmı 7 saat almıştır. Yani Photoscan ve Pix4D web service yazılımlarının işlem süresi neredeyse eşit olduğu anlaşılmıştır. Photoscan yazılımı 0,02 m standart sapma ve 0,11 m ortalama hata ile en iyi performansı göstermiştir. Bundler method ve Pix4D web service yazılımlarının standart sapması sırasıyla 0,14 m ve 0,02 m, ortalama hata ise sırasıyla 0,75 m ve 0,24 m olarak hesaplanmıştır (Turner ve ark., 2014). En kötü performansı Bundler method göstermiştir.

Kekeç ve ark. (2014), bir kameradan çekilen ardışık hava görüntülerini kullanan ve gerçek zamanlı olarak tutarlı büyük ölçekli mozaikler oluşturabilen yeni bir görüntü mozaikleme tekniği sunmuştur. Diğer yaklaşımlardan farklı olarak her yeni görüntünün bir önceki ile eşleştirmesi yerine yeni görüntü ile kesişen mevcut mozaikteki tüm görüntülerin eşleştirilmesi yapılmıştır. Görüntülerdeki kesişmelerin tespiti için SAT (Separating Axis Theorem) kullanılmıştır. Daha sonra mozağin küresel tutarlılığını geliştirmek için hizalanmış görüntü üzerinde yeni bir affine iyileştirmesi gerçekleştirilmiştir. Aynı zamanda çok bantlı harmanlama yöntemi uygulanmıştır. Algoritma Czyste, Munich Quarry ve Savona Highway veri setleri ile test edilmiş ve Gracias'ın algoritması ve bundle adjustment algoritması ile kıyaslanmıştır. Ayrıca

hafızadan tasarruf yapmak için çok bantlı harmanlamanın çevrimdışı yapılması önerilmiştir. Sonuçlar iki algoritma ile kıyaslandığında başarılı olduğu gözükmemektedir.

Afetlerin neden olduğu kayıplar ile ilgili 2015 yılında Zheng ve arkadaşları tarafından bir çalışma yapılmıştır. Yapılan bu çalışmada gerçekleşen kayıpları azaltmak amacıyla hızlı görüntü mozaikleme için kullanılan bir İHA yol planlama matematiksel modeli önerilmiştir. Çeşitli ortam koşullarında İnsansız Hava Araçlarıyla verimli bir şekilde mozaikleme yapabilmek için üç farklı yaklaşımın sunumu gerçekleştirilmiştir. Bu modeller; engebesiz araziler için ideal rota modeli, yüksek yapıların olduğu yerlerde değişken hız ve farklı yol şerit modeli son olarak da değiştirilebilir yükseklik yol planlama modelidir. Bu modeller görüntü mozaiklemede örtüşme oranını ayarlamak için farklı koşullarda hız ve yükseklik ayarının yapılmasına dayanmaktadır (Zheng ve ark., 2015).

Görüntü mozaikleme ile ilgili çok fazla çalışma yapılmış olması sebebiyle Ghosh ve Kaabouch 2016 yılında, geçmişte yapılmış olan görüntü mozaikleme algoritmalarını araştırmış ve bu algoritmalar aralarında uzamsal tabanlı yöntemler ve frekans tabanlı yöntemler olarak ikiye ayırmıştır. Sınıflandırılan algoritmalar hesaplama süresi ve doğruluk yönünden kıyaslanmış, algoritmaların avantajları ve dezavantajları tartışılmıştır (Ghosh ve Kaabouch, 2016). Çalışmada, görüntü mozaikleme konusunda yapılacak olan diğer çalışmalarda uygun bir algoritmanın seçimi için yol göstermeyi hedeflemiştir.

İnsansız Hava Araçlarından (İHA) alınan görüntülerin verimli bir şekilde birleştirilmeleri için genelde yüksek ve sabit irtifada görüntüler alınmaktadır. Yüksek irtifadaki görüntülerde ağaç gibi nesnelerden kaynaklanan derinlik değişimi önemsiz hale gelmektedir. Li ve Isler (2016)'ın yaptığı çalışmada, farklı yüksekliklerde ve yüksek çözünürlükte geniş bir harita elde edebilmek için uygulanan yöntemlerin sunumunu gerçekleştirmiştir. Görüntü mozaiklemesinden önce, ilk olarak görüntüler kullanılmak için belirli aralıklar ile seçilmiştir. Bunun ardından görüntülerdeki üç boyutlu ortam hakkında bilgi edinmek için SFM algoritmasına verilmiştir. Daha sonra yer düzlemi tahmini yapılmıştır. Değişen her bir düzlem için kamera pozisyonu ve yer düzlem açısı yeniden ayarlanmıştır. Homografi hesabı yapıp piksellerdeki bozulmaya bağlı olarak görüntü kalitesi karşılaştırıldıktan sonra mozaikleme gerçekleştirilmiştir. Geliştirilen yöntem; meyve bahçesi ortamında ve mısır tarlasında test edilmiş, Microsoft ICE software, Auto-Stitch ve Autopano yazılımları ile kıyaslanmıştır. Yapılan testlerde daha

az yanlış hizalama ve iyileştirilmiş çözünürlük ile mozaiklerin oluşturulabileceği gösterilmiştir.

Özellik eşleştirmeye dayalı görüntü mozaikleme yöntemi, kullanılan özellik çıkarma algoritmasına göre çok fazla vakit alabilmektedir. Bu nedenle, Ge ve ark. (2016) hızlı görüntü mozaikleme amacıyla bir çalışma yapmıştır. Çalışmada kullanılan yöntemin uygulanması üç adıma ayrılmıştır. Başlangıçta her bir görüntüde, 300 rastgele nokta üretilmiş ve farklı olan 50 nokta kullanılmıştır. Özellik eşleştirme için belirli alanlarda tarama yapılmış ve noktalar eşleştirilmiştir. DLT (Direct Linear Transformation) algoritması ile homografi hesabı yapılmıştır. Sonra, tahmin edilen homografiye dayalı olarak, ortorektifikasyon parametrelerinin hesaplanması için az sayıda GCP (Ground Control Point) kullanılarak bundle adjustment algoritması uygulanmıştır. Daha hızlı görüntü işleyebilmek için görüntü çözünürlüğü düşürülmüş ve sık görüntüler geldiği için her görüntüde mozaikleme uygulanmamıştır. Görüntüler 3840×2160 çözünürlük ile Google Earth üzerinden toplanmıştır ve 960x540 olarak yeniden boyutlandırılmıştır. Çalışmada önerilen yöntem özellikler eşleştirilirken ORB, SIFT ve SURF algoritmalarına göre zamansal açıdan daha iyi performans göstermiştir. Mozaiklerin ortalama hata payı 2,6 m, ortalama standart sapma ise 1,6 m olarak ölçülmüştür.

2016 yılında Bu ve arkadaşlarının yaptığı çalışmada, İnsansız Hava Aracından alınan görüntülerin kademeli olarak birleştirilip büyük bir görüntünün elde edilmesi için gerçek zamanlı bir yaklaşım sunulmuştur. Görüntü mozaikleme için yapılan diğer çalışmalardan farklı olarak, kamera konumunu tahmin etmek için monoküler bir SLAM (Simultaneous Localization and Mapping) sistemi kullanılmış ve 3B nokta bulutu haritası oluşturulmuştur. GPS sayesinde yer kontrol noktalarına gerek kalmamıştır. Çalışmanın sonuçlarını değerlendirmek için hem NPU DroneMap veri kümesi oluşturulmuş hem de Photoscan ve Pix4DMapper gibi ticari yazılımlar ile kıyaslanmıştır. Veri setinin toplanması için hexacopter ve Phantom3 İHA'ları kullanılmış ve her bir araçla alınan görüntülerdeki mozaikleme hata payı sırasıyla 3.066 m ve 5.710 m olarak hesaplanmıştır (Bu ve ark., 2016). Önerilen yöntem Photoscan ve Pix4DMapper'a göre çok daha hızlı çalışmıştır.

Mozaikleme algoritmasının kullanılabileceği alanlardan bir diğeri ise güneş enerji santralleridir. Bilindiği gibi güneş enerji santrallerinin ömrü boyunca çeşitli kusurlara ve arızalara maruz kalabileceğinden, güneş enerji santrali işletimi sırasında etkin izleme çok önemlidir ve zorunlu bir görevdir. Aghaei ve ark. (2016), bu nedenle yapılan çalışmada dijital görüntü işleme teknikleri ile güneş enerji santrallerinin haritalanmasını, güneş panellerinin izlenmesini ve sayılmasını içeren yeni bir yaklaşım önermiştir. Çalışma termograf kamerası bulunan hafif bir İnsansız Hava Sistemi ile gerçekleştirilmiştir. Görüntü mozaikleme işleminde özelliklerin çıkarılabilmesi için Harris Köşe Dedektör algoritması kullanılmıştır. Panellerin sayılabilmesi için Laplacian model üzerine sobel filtresi uygulanmıştır. Mozaiklenen görüntüler üzerinden sağlıklı ve arızalı güneş panellerinin tespiti gerçekleştirilmiştir.

Mozaikleme yönteminde sadece RGB kameralar kullanılmak zorunda değildir. Thiel ve Schmulius'un 2016 yılında yaptığı çalışma bu duruma örnek olarak gösterilebilir. Çalışmada, İHA'da kullanılan multispektral kamera ve LiDAR'dan (Laser Imaging Detection and Ranging) alınan veriler ayrı ayrı mozaiklenmiş ve mozaiklenen bu görüntüler üzerinde ağaçların sayımı yapılmıştır. Gerçekleştirilen çalışmada görüntü mozaikleme için kullanılan Photoscan yazılımından faydalanılmıştır. Ağaçların tespiti için CHM (Canopy Height Model) algoritması kullanılmıştır. Genel olarak, kamera ile elde edilen İHA verileri daha fazla ayrıntı göstermiştir. Bu durum küçük ağaçların tespitinde önemli rol oynamıştır. LiDAR verileri kullanılırken 205 ağaçtan 45'i tespit edilememiş, ancak kamera verileri kullanıldığında sadece 14 ağaç gözden kaçırılmıştır (Thiel ve Schmulius, 2016).

Liu ve ark. (2017), temel görüntü mozaikleme uygulamalarından farklı olarak birkaç farklı yenilik önermiştir. Çalışmada; kameranın parametre hatası, projeksiyon düzlemi hatası gibi hataları içeren yeni bir homografi matrisi oluşturulmuştur. Hata homografi matrisi bilinmeyen bir değişken olarak kullanılmış ve EO (Exterior Orientation) ögesini optimize etmek yerine homografi matrisi optimize edilmiştir. Kümülatif hatalardan kaçınan küresel bir optimizasyon çalışması yapılmıştır. İlk olarak alınan pozisyon bilgileri WGS84'den UTM (Universal Transverse Mercator) koordinat sistemine dönüştürülmüştür. Görüntülerdeki özelliklerin çıkarılması için SIFT algoritması kullanılmış ve daha çok özellik için Harris köşe tespit algoritması eklenmiştir. Yanlış eşleşmeleri kaldırmak için RANSAC (RANdom Sample Consensus) yaklaşımı

tercih edilmiştir. Sonuç olarak bu çalışma, doğrudan homografi dönüşümüne kıyasla dikiş hatlarındaki yanlış hizalamaları etkili bir şekilde ortadan kaldırabildiğini ortaya koymaktadır.

Tarım alanında yapılmış diğer bir çalışma Ren ve arkadaşları tarafından 2017 yılında yapılmıştır. Yapılan çalışmada, tarımsal uygulamalarda İHA ile alınan multispektral görüntüleri mozaikleme için basitleştirmiş ve etkili bir yöntem sunulmuştur. İlk olarak, GPS yardımıyla potansiyel eşleşme ilişkisi mesafeye dayalı yöntem (distance-based method) ile tespit edilmiştir. Sonra, 6 bantlı multispektral kamerada özellik eşleştirmede en uygun bandın otomatik tercihi için gerekli hesaplamalar yapılmıştır. Projeksiyon matrisini sadeleştirmek için sadece yaw açısı hesaplanmış, roll ve pitch açıları birim matris olarak ele alınmıştır. Daha sonra, özellikler eşleştirilir. Eşleştirilen özellikler kameranın pozisyonunu hesaplamak için kullanılmıştır. Kamera pozisyonunun tahmin edilebilmesi için bundle adjustment algoritmasından yararlanılmıştır. Son olarak harmanlama algoritması uygulanmıştır. Çalışma 412 ve 223 adet görüntüye sahip olan iki farklı veri setinde test edilmiş ve Pix4Dmapper ile kıyaslanmıştır. Veri boyutu arttıkça Pix4Dmapper'ın mozaikleme süresi doğrusal olmayan bir şekilde artmıştır. Uygulanan yaklaşımın çalışma süresi ise veri miktarı ile doğru orantılıdır. Ortalama piksel hatası önerilen yaklaşımda 12.84, Pix4Dmapper 38.89'dur (Ren ve ark., 2017). Sonuç olarak önerilen yaklaşım Pix4Dmapper'a göre daha iyi sonuçlar elde etmiştir.

Rojas ve ark. (2017), pirinç tarlalarının görüntü mozaiklerinin üretilmesini sağlamak için bir görüntü yakalama algoritmasının sunumunu yapmıştır. Çalışma ROS'ta (Robot Operating System) geliştirilmiş ve Gazebo simülasyon ortamında test edilmiştir. Görüntüleri birleştirmek için OpenCV'deki (Open Source Computer Vision Library) görüntü mozaikleme kütüphanesinden yararlanılmıştır. Görüntü yakalama algoritması için kameranın iç parametreleri ve görüş alanı hesabı yapılmıştır. Yöntem 2 farklı yükseklikte test edilmiş ve örtüşme oranı %30 olarak belirlenmiştir. Görüntüler başarılı bir şekilde mozaiklenmiştir.

Görüntü mozaikleme algoritmalarında gerek yanlış hizalama gerekse de görüntünün alındığı andaki ortamın parlaklık koşullarından kaynaklı olarak istenmeyen izler oluşabilmektedir. Bu izlerin yok edilebilmesi için farklı etkilere sahip görüntü

harmanlama algoritmaları kullanılmaktadır. Bu nedenle Zhu ve ark. (2018), gerçek zamanlı görüntü harmanlamada, MVCB (Mean-Value Coordinates Blending), FB (Feather Blending), CPB (Convolution Pyramid Blending), MSB (Multi-Spline Blending), MBB (Multi-band Blending) ve MPB (Modified Poisson Blending) isimli yöntemlerin zaman, hafıza ve performans açısından yedi farklı video görüntüsünde analizlerini yapmıştır. Deney, 32GB belleğe sahip Intel Xeon E5-2620 2.0GHz CPU ve 4GB belleğe sahip bir NVIDIA GTX 970 GPU'ya sahip bir bilgisayarda gerçekleştirilmiştir. Algoritmalar C++'da uygulanmış ve CUDA'dan yararlanılmıştır. FB ve MBB algoritmaları zaman ve hafıza açısından en iyi performansı (Compute Unified Device Architecture) göstermiştir. Harmanlama algoritmalarının kalitesi, harmanlama yok <MVCB <FB <CPB <MSB <MBB <MPB olarak bulunmuştur.

Başka bir çalışmada ise Ruiz ve ark. (2018) İHA sürüsünden alınan görüntülerle çalışmak üzere özel olarak tasarlanmış yeni bir mozaikleme yöntemi olan MGRAPH'in (Multi-GRAPH Homography-based) sunumunu yapmıştır. CPU'nun (Central Processing Unit) işlem yükünü hafifletmek için GPU'dan (Graphic Processing Unit) yararlanılmıştır. Algoritma, mozaiklerin üretilip birleştirilmesine dayanmaktadır. İlk olarak alınan görüntülerin ORB algoritmasıyla, GPU kullanılarak özellikleri çıkarılmıştır. Alınan anahtar noktalarının sayısı belirli miktardan az ise görüntüde işlem yapılmamıştır. Aykırı değerleri ayıklamak için kullanılan RANSAC algoritmasından önce oran mesafe filtresi uygulanmıştır. Homografi hesabı belirli bölgelerde yapılarak işlemler hızlandırılmıştır. Görüntülerin dikilmesi sırasında biriken hatayı giderebilmek için Ceres kütüphanesinden yararlanılmıştır. Algoritma; Quarry, GOLF ve Aukerman isimli üç farklı veri kümesinde test edilmiş ve BIMOS algoritması ile karşılaştırılmıştır. MGRAPH algoritmasının hatası BIMOS algoritmasına göre ortalama 1 piksel daha fazladır. Uygulanan algoritma ile görüntü dikimi başarılı bir şekilde gerçekleştirilmiştir. Sonuçların iyileştirilmesi için GPS verilerinin optimizasyon algoritmasına da dahil edilmesi önerilmiştir.

Ghazal ve ark. (2019), İnsansız Hava Araçlarında bitki örtüsünün takibi için görüntü mozaikleme ve segmentasyon yöntemlerinin birlikte kullanıldığı bir algoritmanın sunumunu gerçekleştirmişlerdir. Bitki örtüsünün takip edilebilmesi için dört farklı kamera (RGB, NIR, NDVI ve RED) ile donatılmış bir hava aracı kullanılmıştır. Hava aracından alınan görüntülerin birbirleri ile birleştirilebilmeleri için ORB (Oriented FAST and Rotated BRIEF), BRIEF (Binary Robust Independent Elementary Features),

FAST (Features from Accelerated Segment Test) ve SIFT (Scale Invariant Feature Transform) gibi özellik çıkarma yöntemleri literatürde yapılmış çalışmalarla zaman performansı açısından kıyaslanmıştır. Daha hızlı çalıştığı için ORB yöntemi tercih edilmiştir. Görüntü mozaikleme ile birleştirilen görüntülerden bitki örtüsünün tespit edilebilmesi için 2 farklı senaryo ele alınmıştır. Bitki örtüsünün ayırt edilebilmesi, canlıların farklı ışık dalga boylarında yansıtma ve soğurma özelliklerinde faydalanılmıştır. İlk senaryoda tüm kameralardan alınan görüntüler ile evrişimli sinir ağları (CNN) kullanılarak model eğitilmiştir. Diğer senaryoda ise her kameradan alınan görüntüler için ayrı ayrı model eğitimi gerçekleştirilmiştir. En yüksek doğruluk ortalaması %97.29 oranıyla ilk senaryoda elde edilmiştir.

Karmaşık problemlere çözümler bulabilmek için kullanılan yaklaşımlar ağırlaştıkça algoritmaların çalışma süresi artmaktadır. Bu sürenin azaltılması için Guo ve ark. (2019) çalışmasında yerel dalgacık dönüşümü (wavelet transform) ve görüntü mozaikleme yöntemlerinin birlikte kullanımlarının sunumunu gerçekleştirmiştir. Panoramik görüntü mozaikleri için öznitelik çıkarma aşamasında, yerel öznitelik vektörleri oluşturmak için dalgacık alt bant istatistiklerinden yararlanılmıştır. İki seviyeli bir dalgacık dönüşümü kullanılmıştır. Tasarımı gerçekleştirilen mozaik sistemi ortama 295 fps'e kadar ulaşmaktadır. Kullanılan görüntünün çözünürlüğü hakkında bilgi verilmemiştir. Tasarlanan yöntemin çalışma doğruluğu %87.4 olarak verilmiştir.

Raeva ve ark. (2019), İHA'nın tarımsal bir uygulamasını sunmuştur. Yapılan çalışmada multispektral ve termal kameralar kullanılarak iki farklı çıktısı olan görüntüler elde edilmiştir. Görüntüler mısır ve arpa arazilerinde drone eBee ile toplanmıştır. Multispektral kameralardan alınan görüntüler de NDVI (Normalized Difference Vegetation Index), GNDI (Green Difference Vegetation Index) ve NDRE (Normalized Difference Red Edge) olmak üzere 3 farklı indekste incelenmiştir. Algoritmanın başlangıcında görüntülerde özellikler çıkarılıp eşleştirilmiştir. Daha sonra Bundle block adjustment ve Automatic AT (Aerial Triangulation) yöntemleri kullanılmış ve dağınık bir nokta bulutu elde edilmiştir. Ardından nokta bulutu yoğunluğu artırılmıştır. Dijital yüzey modeli (DSM) kullanılarak algoritma tamamlanmıştır. Termal kamera ise toprak sıcaklığını ölçmek için kullanılmıştır.

Avola ve ark. (2020) görüntülerin birleştirilmesi ve görüntüler üzerindeki değişikliğin tespit edilmesi için iki temel algoritmanın kullanımını ele almıştır. Çalışmada görüntülerin birleştirilebilmesi için GPS verisinden yararlanılmıştır. Algoritmalar çeşitli parametrelerde (İHA'nın hızı, yüksekliği ve yönü), farklı ortamlardan (kent, toprak, kırsal) ve alçak irtifada alınan görüntülerden oluşan UMCD veri kümesinde test edilmiştir. UMCD veri kümesi, telemetri verisinin olduğu ve olmadığı görüntüler olarak ikiye ayrılmıştır. Bir görüntünün diğer bir görüntü ile uyumlu olacak şekilde dikilebilmesi için homografi hesabı yapılmıştır. Birleştirilen görüntüler takip edilmesi sayesinde bir sonraki çerçevenin dikilmesi için mozaığın belli kısmı kontrol edilerek işlem yükü hafifletilmiştir. Mozaikler doğru bir şekilde üretildiği ancak ortorektifikasyon aşamasının olmamasından dolayı bir miktar bronzluk distorsiyonu olduğu ifade edilmiştir. Görüntüler üzerindeki değişikliğin tespiti %81.983 doğrulukla gerçekleştirilmiştir. Doğruluk ölçütü, nesneler arasındaki bilinen mesafeler kullanılarak yapılmıştır.

Bakış açısı görüntü dikiminin kalitesini etkileyen faktörlerden bir tanesidir. Bu etkeni bazı yapıların yakınında yapılara sebep olabilmektedir. Bu sorunu çözebilmek için Xu ve ark. (2020) bölge kılavuzlu yerel projeksiyon deformasyonuna (multiregion guided local projection deformation) dayalı yeni bir dikiş yöntemi önermiştir. Önerilen yöntemde ilk olarak görüntünün SIFT algoritması ile özellikleri çıkarılmış ve RANSAC ile aykırı noktalar ayıklanmaya çalışılmıştır. Homografiyi doğru bir şekilde hesaplamak için birden fazla bölgenin farklı ağırlıklara sahip olduğu önerilen bir strateji kullanılmış ve görüntü dikimi yapılmıştır. Geliştirilen yöntem; Autostitch, SPHP (Shape-Preserving Half-Projective), APAP (As-projective-as-possible), REW (Robust Elastic Warping), AANAP (Adaptive As-Natural-As-Possible) gibi diğer yöntemler ile kıyaslanmıştır. AANAP etkili bir dikiş yöntemi olduğu ancak problemi çözmede yetersiz kaldığı söylenmiştir. Geliştirilen yöntem görüntü birleştirmeyi iyileştirmiştir. Ancak birden fazla görüntü dikilirken sonuçta bozulma gözlemlendiği ifade edilmiştir. Diğer karşılaştırılan yöntemlere göre maliyeti bir miktar daha fazladır.

Ma ve ark. (2020), nokta bulutunun yardımıyla otomatik görüş açısını yanlış hizalama kalibrasyonu ile havadaki bir LiDAR sistemine entegre olan ve dijital kamera tarafından toplanan görüntülerin doğrudan coğrafi referanslandırılması için bir yöntem sunmuştur. Yaklaşım ilk olarak bir bağlantı nokta (tie points) kümesi oluşturmak için SURF algoritmasından yararlanmıştır. Her bir bağlantı noktası çifti için, nesne noktası

uzay kesişimi ile türetilmiştir. Yükseklik bilgisi, LiDAR nokta bulutundan üretilen yükseklik bilgisi ile değiştirilmiş ve yeni bağlantı bilgisi VCP (Virtual Control Point) olarak adlandırılmıştır. Ardından yüksek kaliteli sanal kontrol noktalarını kullanmak için bir seçim prosedürü uygulanmıştır. Görüş alanının yanlış hizalanmasını gidermek için Eş doğrusallık denklemi kullanılmıştır. Sanal kontrol noktaları ile bağlantı noktaları arasındaki mesafe sabit kalana kadar yineleme süreci devam ettirilmiştir. Çalışma iki farklı veri seti ile test edilmiş ve RMS (Root mean square) ile ölçülen hata payı 0,64 m olarak bulunmuştur.

Syietawan ve ark. (2020), daha kısa sürelerde topografik harita üretmeyi amaçlamıştır. Bunun için DJI Phantom 4 RTK/PPK ile alınan görüntüler doğrudan coğrafi referanslama yöntemi ile mozaiklenmiştir. %70 ve %80 örtüşme oranına sahip 293 görüntü mozaiklemek için kullanılmıştır. Görüntülerin mozaiklenmesindeki PPK'nın (Post Processing Kinematic) etkisini ölçmek için PPK'nın kullanılıp kullanılmamasına göre mozaiklerdeki hata miktarları karşılaştırılmıştır. Görüntülerin ön işlenmesi için ASP suite yazılımından yararlanılmıştır. PPK kullanıldığında yataydaki hata payı 0.038 m, kullanılmadığında ise 5.499 m olarak ölçülmüştür.

Bu bölümde, tez çalışmasında önerilen 3 farklı algoritma ile ilgili literatürde yapılmış çalışmalar aralarında gruplandırılarak ayrıntılı bir şekilde verilmiştir. Gelecek bölümde tez çalışmasında kullanılan yöntemler anlatılmıştır.

3. MATERYAL VE YÖNTEM

Yapılan tez çalışmasında önerilen yaklaşım, Şekil 3.1’de görüldüğü gibi üç temel algoritmanın birleşiminden meydana gelmektedir.

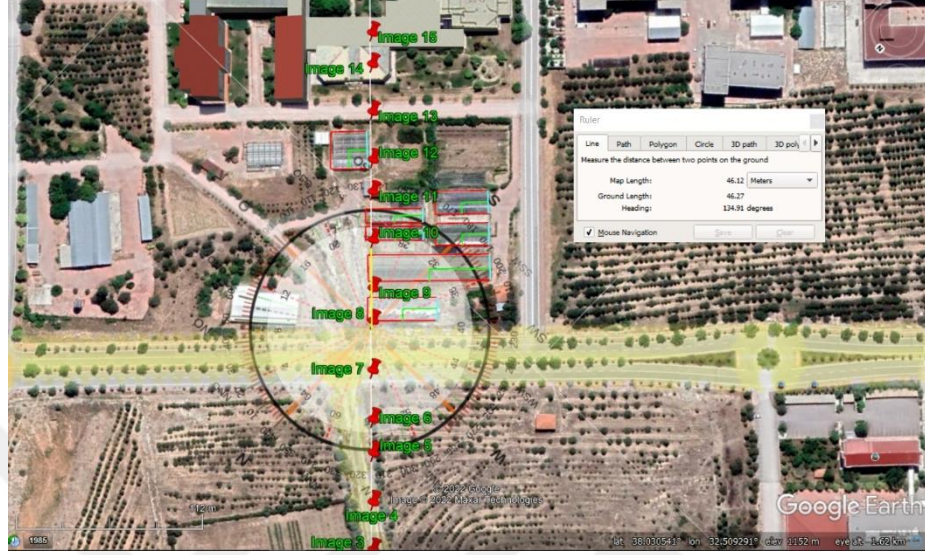


Şekil 3.1. Önerilen algoritmanın akış diyagramı

Yaklaşım, hava aracından alınan görüntülerin, Faster R-CNN modelinden yararlanılarak eğitilen derin öğrenme modelinin girişine verilmesi ile başlamaktadır. Bu bölümde, İHA’nın kullanıldığı alana göre farklı nesnelerin tespiti gerçekleştirilebilir. Ancak bu çalışmada kullanılan veri kümesi nedeniyle arabaların ve sera alanlarının tespiti yapılmıştır. Tespit edilen nesnelerin sınıfı, piksel koordinatları ve üzerinde işaretleme yapılmış görüntüler, coğrafi konumlandırma algoritmasının girişine verilmek için kullanılmıştır.

Coğrafi konumlandırmanın gerçekleştirilmesi için öncelikle yaw açısının belirlenmesi gerekmektedir çünkü kullanılan veri kümesinde hava aracının IMU bilgileri mevcut değildir. Hava aracının yönü belirlenirken Google Earth’den yararlanılmıştır. Açılı belirlenirken görüntünün çekildiği GPS koordinatlarının üzerinde “line” çizilmiştir. Google Earth’ün bu işlevi sayesinde otomatik olarak aracın yönü belirlenmiştir (Şekil 3.2). Ayrıca, elde edilen yön bilgisi Google Earth’e eklenen pusula görseli ile de doğrulanmıştır. Daha sonra kamera görüntü merkezinin GPS koordinatları alınmalı ve bu koordinatlar önce ECEF koordinat sistemine daha sonra ENU koordinat sistemine dönüştürülmelidir. Böylece yerel koordinatlardaki değişimler hesap edilecektir. Yerel koordinatlardaki değişim, kameranın merkez piksel koordinatları ile tespit edilen nesnenin merkezinin piksel değerleri arasındaki farktan yararlanarak hesap edilmiştir. Ancak birimler arasındaki uyumsuzluğu gidermek için piksel başına düşen metre hesabı yapılmalıdır. Böylece yerel koordinatlardaki değişim metre cinsinden ifade edilir. Hesap edilen ENU koordinatları ile değişimin olduğu yatay ve dikey koordinat değerleri toplandıktan sonra elde edilen yeni değerler ECEF koordinat sistemine dönüştürülür. Ardından bu koordinatlar WGS84 koordinat sistemine dönüştürülür. Böylece tespit

edilen nesnenin merkez piksel değeri GPS koordinatlarında ifade edilmiş ve konumlandırılmıştır. Son olarak elde edilen GPS koordinatları ve eğitilen derin öğrenme modelinin çıktısı olan görüntü, mozaikleme algoritmasının girişine gönderilmiştir.



Şekil 3.2. Yaw açısının nasıl belirlendiğini gösteren ve Google Earth’den alınan görsel

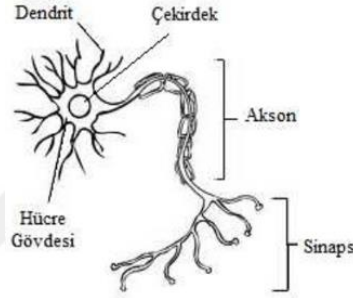
Görüntülerin mozaiklenmesi için girişe gelen görüntülerin ikiyeşerli olarak özellikleri çıkarılmıştır. Özellik çıkarmak için ayrı ayrı SIFT, SURF, BRIEF ve ORB algoritmalarından faydalanılmıştır. Daha sonra çıkarılan özelliklerin eşleştirilmesi için brute-force algoritması kullanılmıştır. Eşleştirilen özellik çiftleri arasındaki yanlış eşleştirmeleri elemek için RANSAC algoritması ve mesafe sınırlamasından yararlanılmıştır. Son aşamasında ise homografi tahmini yapılmış ve bu homografi tahminine göre görüntüler birleştirilmiştir. Sırayla mozaiklenen görüntülerden sonra derin öğrenme bilgilerinin çıktısı ve coğrafi konumlandırma algoritmalarının çıktısı, mozaikleme algoritması ile tek ve büyük bir görüntüde birleştirilmiştir.

Tezin gelecek bölümünde, kullanılan yöntemlerin ayrıntılı teorik bilgisi verilmiştir.

3.1. Yapay Sinir Ağları

Yapay bir sinir ağı (nöron ağı), sinir hücrelerinin insan beynindeki çalışma şeklini taklit eden bir hesaplama modelidir. Bu nedenle biyolojik sinir hücresi (Şekil 3.3) ile sinir hücresinin matematiksel modeli ayrıntılı olarak incelenmelidir. Biyolojik sinir sisteminin

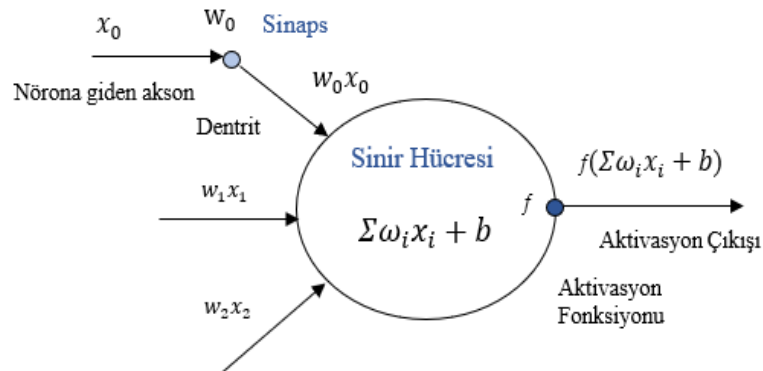
önemli bir yapı taşı olan nöronların yapısı; dendrit, akson, çekirdek ve sinaps olmak dört ana bölümden oluşmaktadır. Bağlı olduğu diğer nöronlardan veya duyu organlarından gelen sinyalleri çekirdeğe ileten dendritler ağaç kökü görünümünde bir yapıya sahiptir. Dendrit aracılığıyla gelen sinyalleri bir araya toplayan çekirdek, sinyallerin aksona iletilmesini sağlar. Toplanan bu sinyaller akson tarafından işlenerek nöronun diğer ucunda bulunan bağlantılara (sinaps) gönderilir. Bağlantılar ise yeni üretilen sinyalleri diğer nöronlara iletimini gerçekleştirir.



Şekil 3.3. Biyolojik sinir hücresinin temsili gösterimi

3.1.1. Tek katmanlı sinir ağı

Yapay sinir hücreleri (Şekil 3.4), biyolojik sinir hücrelerine benzer yapıda bulunmaktadır. Yapay olarak oluşturulan nöronlar da aralarında iletişim kurarak yapay sinir ağlarını oluşturmaktadır. Tıpkı biyolojik nöronlardaki gibi yapay nöronların da giriş sinyallerini aldıkları, bu sinyalleri toplayıp işledikleri ve çıktıları ilettikleri bölümleri bulunmaktadır. Bir yapay sinir hücresi; girdiler, ağırlıklar, toplama fonksiyonu, aktivasyon fonksiyonu ve çıktılar olmak üzere beş bölümden oluşmaktadır.



Şekil 3.4. Canlıda bulunan bir sinir hücresinin matematiksel modeli

Girdiler: Nöronlara gelen verileri göstermektedir. Girdiler yapay sinir hücresine başka bir hücreden gelebileceği gibi doğrudan dış dünyadan da gelebilmektedir. Bu girdilerden gelen bilgiler biyolojik sinir hücrelerinde olduğu gibi toplanması için nöron çekirdeğine gönderilir.

Ağırlıklar: Yapay sinir hücresine gelen bilgiler girdiler üzerinden çekirdeğe ulaşmadan hemen önce geldikleri bağlantıların ağırlık değerleriyle çarpılarak çekirdeğe iletilir. Böylece, girdilerin üretilecek çıktı üzerindeki etkisi ayarlanabilmektedir. Ağırlıkların değerleri sıfır, pozitif veya negatif olabilmektedir. Eğer ağırlık değeri sıfır olan bir girdi varsa, o kol üzerindeki girdinin çıktı üzerinde herhangi bir etkisi olmamaktadır.

Toplama Fonksiyonu (Birleştirme Fonksiyonu): Bu fonksiyon bir yapay sinir hücresine ağırlıklarla çarpılarak gelen girdileri toplayarak o hücrenin net girdisini hesaplayan bir fonksiyondur.

Aktivasyon fonksiyonu: Önceki katmandaki tüm girdilerin ağırlıklı toplamını alan ve daha sonra bir çıkış değeri (tipik olarak doğrusal olmayan) üreten ve bir sonraki katmana geçiren bir fonksiyondur. (Örneğin, ReLU (Rectified Linear Unit) veya sigmoid, doğrusal, adım, tanjant hiperbolik, sinüs).

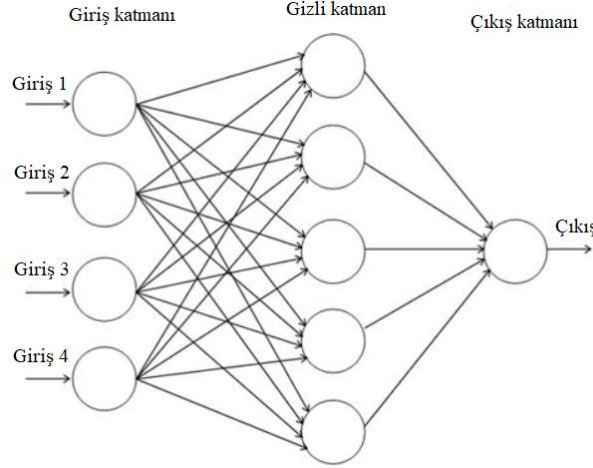
Çıktılar: Aktivasyon fonksiyonundan çıkan değer hücrenin çıktı değeri olmaktadır. Her hücrenin birden fazla girdisi olmasına rağmen bir tek çıktısı olmaktadır. Bu çıktı istenilen sayıda hücreye bağlanabilir.

En basit ağ yapısı, iki giriş ve bir çıkıştan oluşmaktadır. Bu ağ yapısının çıkışı Eşitlik 3.1'den yararlanılarak hesaplanabilir. Nöronun giriş değeri, giriş değişkenlerinin ağırlıklı toplamına ($w_i x_i$) sapma (b_{ysa}) değerinin eklenmesiyle hesaplanır. Toplama fonksiyonunda hesaplanan değer, aktivasyon fonksiyonundan (f_a) geçirildikten sonra çıkış değeri (y_e) elde edilir.

$$y_e = f_a(b_{ysa} + \sum_i^n w_i x_i) \quad (3.1)$$

3.1.2. Çok katmanlı sinir ağı

Günümüzde yaygın olarak kullanılan öğrenme modelleri genel olarak çok katmanlı sinir ağı (Şekil 3.5) yapılarından oluşmaktadır. Bu yapılar; giriş katmanı, gizli katman ve çıkış katmanı olmak üzere bünyesinde üç bölüm bulundurmaktadır.



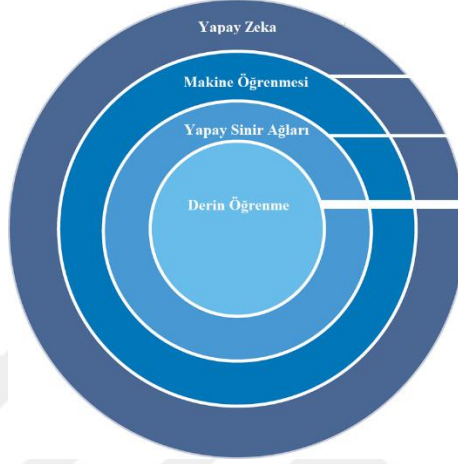
Şekil 3.5. Çok katmanlı bir sinir ağı

Girdi katmanı çok katmanlı bir sinir ağının ilk aşamasıdır ve alınan bilgilerin gizli katmana aktarıldığı kısımdır. Bu aşamada bilgilerin işlenmesi gerçekleştirilmemektedir. Gizli katmanda ise her bir eleman gizli katmandaki tüm elemanlara bağlıdır. Bu katmanda giriş elemanlarından gelen bilgilerin işlenmesi gerçekleştirilir. Gizli katmanda birden fazla katman bulunabilmektedir. Yani, problemin tipine bağlı olarak gizli katmaların sayısında değişim olabilmektedir. Son olarak çıktı katmanında, gizli katmandan gelen bilgiler işlenerek çıkışa iletilmektedir.

Şekil 3.5'e daha yakından bakılacak olursa, ilk olarak çıkarılan özelliklerine göre sınıflandırılacak olan veri girişe verilir. Veriler gizli katmana aktarılmadan önce ağırlık değerleri ile çarpılır. Elde edilen sonuçlar gizli katmanda işlendikten sonra son katmana aktarılır. Çıkış katmanında, sınıflandırılmak istenen nesnenin sayısal değeri ile hesaplanan değerler arasındaki fark hesaplanır ve ağırlık değerleri geri yayılım ile güncellenmektedir (Örnek, 2019).

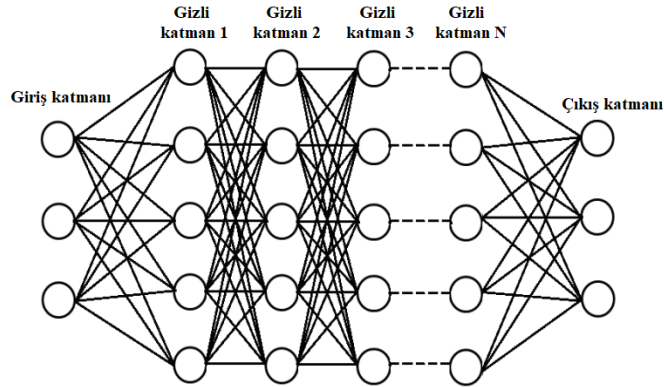
3.2. Derin Öğrenme

Derin öğrenme, makine öğreniminin bir alt alanıdır (Şekil 3.6). Derin öğrenme, bilgisayarların ham verilerden yararlı bilgileri otomatik olarak çıkarılmasını, analiz edilmesini ve anlamlandırılmasını sağlamak için kullanılan gelişmiş bir makine öğrenimi yaklaşımıdır.



Şekil 3.6. Yapa zekâ ve derin öğrenme arasındaki ilişki

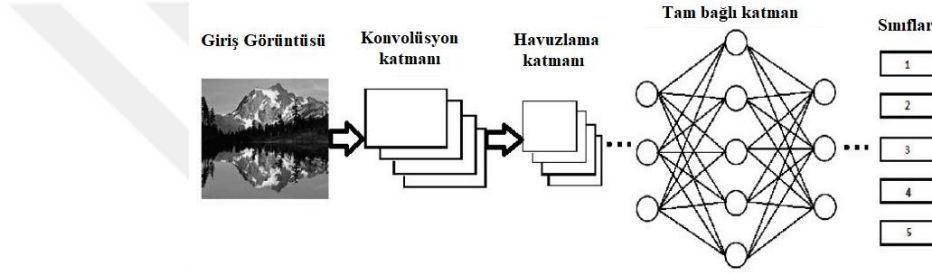
Derin öğrenmeden elde edilen sonuçlar, geleneksel makine öğrenimi yaklaşımından çok daha gelişmiştir. Derin sinir ağları, sistemin girdi ve çıktı arasındaki karmaşık ilişkiyi öğrenmesini sağlayan, doğrusal olmayan çoklu gizli katman mimarisi modelini (Şekil 3.7) kullanmaktadır. Derin öğrenmenin makine öğrenimine göre avantajı, makine öğreniminde olduğu gibi manuel olarak çıkarılmış veya el yapımı özellikler gerektirmemesidir. Yani derin öğrenme, ham verilerden özellikleri otomatik olarak çıkarır, işler ve buna göre karar verir (Chauhan ve Singh, 2018).



Şekil 3.7. Örnek bir derin ağ mimarisi

3.3. Evrişimli Sinir Ağı (CNN)

CNN, en popüler derin öğrenme yaklaşımlarından biridir. CNN'ler nesne tanıma, nesne izleme, metin tanıma ve algılama, görsel algılama, pozisyon tahmini gibi çeşitli uygulamalarda yaygın olarak kullanılmaktadır. CNN'lerin, sinir ağlarından farklı olarak nöronların gizli katmanları, yalnızca nöronların alt kümesini içeren önceki katmanla bağlantılıdır. CNN'lerdeki seyrek bağlantılar, sistemin örtük olarak özellikleri öğrenmesini sağlar. CNN katmanının her bölümü, katmanın girişi ile birlikte kıvrılan iki veya daha fazla boyutlu filtre içermektedir. CNN mimarisi, Şekil 3.8'de gösterildiği gibi konvolüsyon katmanı, havuzlama katmanı ve tam bağlı katmandan oluşmaktadır.



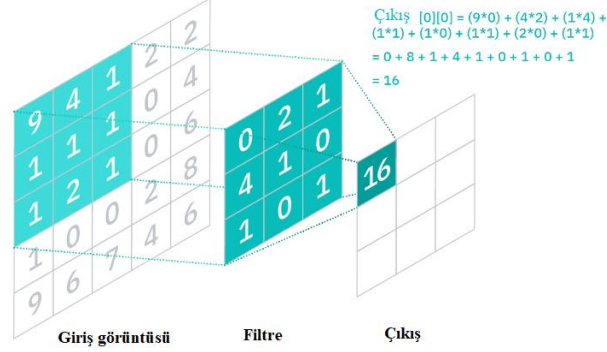
Şekil 3.8. Klasik bir evrişimli sinir ağı mimarisi (Chauhan ve Singh, 2018)

3.3.1. Konvolüsyon katmanı

Konvolüsyon katmanı, CNN'nin temel bir yapı taşıdır ve birçok hesaplamanın gerçekleştiği yerdir. Girdi verileri, filtre ve özellik haritası gibi birkaç bileşen gerektirmektedir. Giriş, 3B piksel matrisinden oluşan renkli bir görüntü olabilmektedir. Bu, girdinin bir görüntüdeki RGB görüntü uzayına karşılık gelen üç boyuta (yükseklik, genişlik ve derinlik) sahip olacağı anlamına gelmektedir. Ayrıca, görüntünün alıcı alanları boyunca hareket ederek özelliğin mevcut olup olmadığını kontrol eden, çekirdek veya filtre olarak da bilinen bir özellik algılayıcı da bulunmaktadır.

Özellik algılayıcı, görüntünün bir bölümünü temsil eden iki boyutlu bir ağırlık dizisidir. Çeşitli boyutlarda kullanılabilmesine rağmen, 3x3'lük filtre boyutu sık sık tercih edilmektedir. Boyutu belirlenen filtre daha sonra görüntünün bir alanına uygulanır. Bu işlemin sonucu daha sonra bir çıktı dizisine beslenir. Ardından, filtre bir adım öteye geçer

ve çekirdek tüm görüntü boyunca dolaşana kadar işlemi tekrarlar (Şekil 3.9). Girdi ve filtreden elde edilen matris özellik haritası olarak da bilinmektedir.

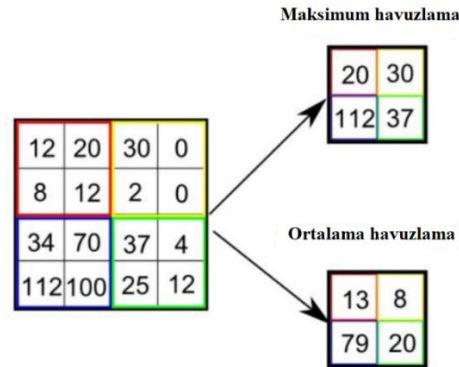


Şekil 3.9. Konvolüsyon işlemi

3.3.2. Havuzlama katmanı

Havuzlama işlemi, alt örnekleme olarak da bilinmektedir. Bu süreç, girdilerdeki parametre sayısını azaltarak matris boyut miktarını azaltmaktadır. Konvolüsyon katmanına benzer şekilde, havuzlama işlemi tüm girdi boyunca bir filtreyi süpürmektedir. Bu işlem, maksimum havuzlama ve ortalama havuzlama (Şekil 3.10) isimli iki ana havuzlama yöntemi ile gerçekleştirilebilir.

Maksimum havuzlama, filtre giriş boyunca hareket ederken, çıkış dizisine gönderilecek maksimum değere sahip pikseli seçmektedir. Ayrıca, bu yaklaşım ortalama havuzlama yöntemine kıyasla daha sık kullanıma eğilimindedir. Ortalama havuzlama filtresi ise giriş boyunca hareket ettikçe, çıkış dizisine göndermek için hesaplanan matrisin ortalama değerini hesaplamaktadır.



Şekil 3.10. Maksimum havuzlama ve ortalama havuzlama örneği

Havuzlama katmanında birçok bilgi kaybolmasına rağmen, önemli avantajları bulunmaktadır. Örneğin işlem karmaşasını azaltarak işlem sürecini hızlandırır. Bir diğer faydası ise aşırı öğrenme (overfitting) riskini azaltmasıdır.

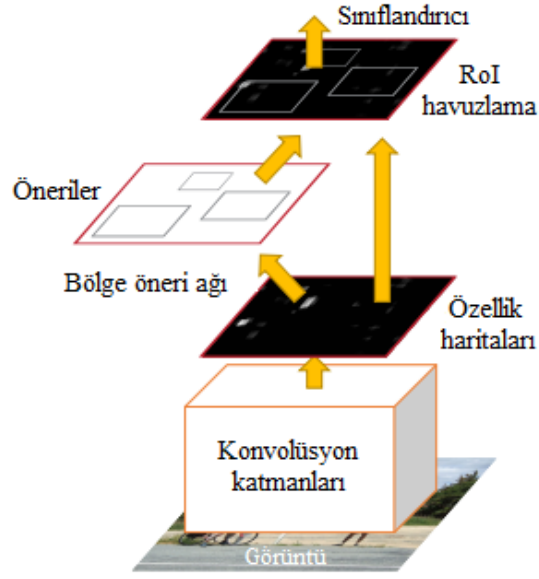
3.3.3. Tam bağlı katman

Bu katman, önceki katmanlardan ve bunların farklı filtrelerinden çıkarılan özniteliklere dayalı olarak sınıflandırma görevini gerçekleştirmektedir. Tam bağlı katmanlar genellikle CNN mimarilerinin sonuna doğru bulunur ve sınıf puanları gibi hedefleri optimize etmek için kullanılmaktadır.

3.4. Faster R-CNN

Faster R-CNN, yavaş ve algılama ağıyla aynı çalışma süresine ihtiyaç duyan ilgi alanı önermek için seçici aramayı kullanmaktadır. Faster R-CNN, çok çeşitli ölçekler ve en boy oranları ile bölge önerilerini verimli bir şekilde tahmin etmek için tam bir konvolüsyon ağına sahip yeni RPN'i (Region Proposal Network) (Şekil 3.11) önermiştir. RPN, algılama ağıyla tam görüntü konvolüsyon özelliklerini ve ortak bir konvolüsyon katmanları kümesini paylaştığı için bölge önerisi oluşturma hızını arttırmaktadır. Ayrıca, Faster R-CNN farklı boyuttaki nesnelerin tespiti için çok ölçekli çapalardan (anchor) yararlanmıştır (Jiao ve ark, 2019). Çapalar, birden fazla girdi görüntüsü veya özellik ölçeğine ihtiyaç duymadan çeşitli büyüklükteki bölge önerileri oluşturma sürecini büyük ölçüde basitleştirebilmektedir. Faster R-CNN'de tanımlanan çapa kutularının 3 farklı ölçek ve 3 en boy oranına sahiptir. Bölge önerisi parametreleri, ilgili referans bir çapa kutusuna göre belirlenmektedir. Ardından, tahmin edilen kutunun konumunu optimize etmek için tahmin edilen kutu ile ona karşılık gelen gerçek doğruluk kutusu arasındaki mesafe ölçülmektedir. Faster R-CNN'in testi PASCAL VOC 2007 test veri kümesinde yapılmış ve %69,9'luk mAP değeri elde edilmiştir.

Sonuç olarak, Faster R-CNN hem hassasiyeti hem de algılama verimliliğini büyük ölçüde iyileştirdiğini göstermiştir. Ek olarak, Fast R-CNN ve R-CNN'e göre çok daha hızlı çalışmaktadır (Jiao ve ark, 2019).



Şekil 3.11. Faster R-CNN ağının mimarisi

3.5. Coğrafi Konumlandırma

Bu aşamada tespit edilen nesnelerin koordinatlarını tespit etmek için coğrafi konumlandırma yapılmıştır. Konumlandırma için gereken adımlar aşağıdaki gibi sıralanabilir:

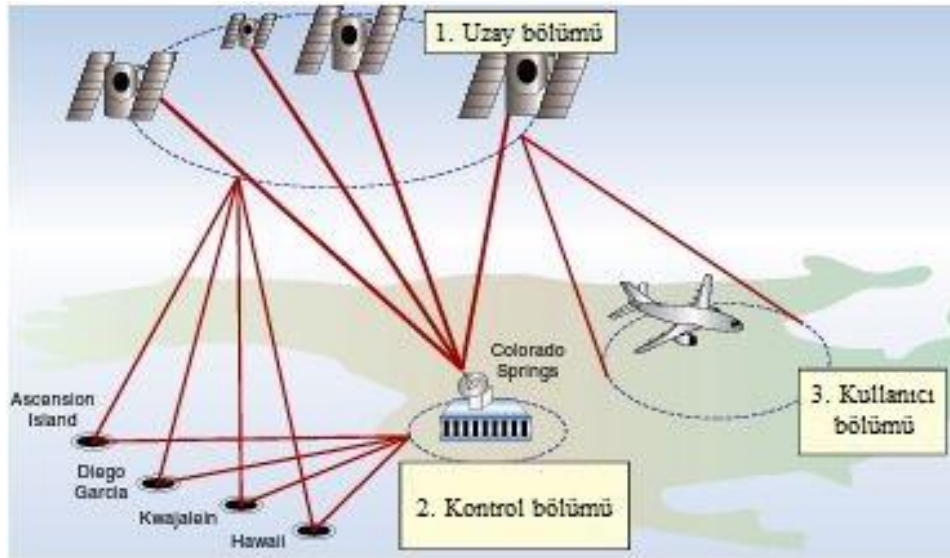
- 1) Eğitilen modelin çıkışından görüntü ve tespit edilen nesnelerin piksel koordinatları alınmalıdır.
- 2) Elde edilen görüntünün merkez noktasının konumu GPS tarafından WGS84 formatında alınmalıdır.
- 3) GPS tarafından alınan WGS84 koordinatları öncelikle ECEF daha sonra ENU koordinatlarına dönüştürülmelidir.
- 4) Daha sonra tespit edilen nesnenin merkez piksel değerleri ile kameranın merkez piksel değerlerinin farkı alınmalıdır.
- 5) Piksel başına düşen metre hesabı yapılmalıdır.
- 6) Yerel koordinatlarındaki değişim Google Earth'den alınan yaw açısı dikkate alınarak hesaplanır.
- 7) Koordinatlardaki değişim, daha önce elde edilmiş olan ENU koordinatlarına eklenmelidir.
- 8) Son olarak, elde edilen yeni değerler; önce ECEF, daha sonra WGS84 koordinatlarına dönüştürülmelidir.

Böylece kameranın herhangi bir pikseline denk gelen GPS koordinatı hesaplanabilmektedir. Ancak bu çalışma kapsamında sadece tespit edilen nesnenin karşılık geldiği piksel koordinatının konumu hesaplanmıştır.

3.5.1. Küresel konumlandırma sistemi (GPS)

Dünya üzerindeki bir noktanın koordinatını belirlemek için Amerika Birleşik Devletleri'ne (ABD) ait NAVSTAR (Navigation System with Timing and Ranging) GPS, Rusya'ya ait GLONASS (Global Navigation Satellite System) ve Avrupa Birliği'ne ait Galileo isimli sistemler bulunmaktadır. Bu çalışma kapsamında NAVSTAR GPS'den yararlanılmıştır.

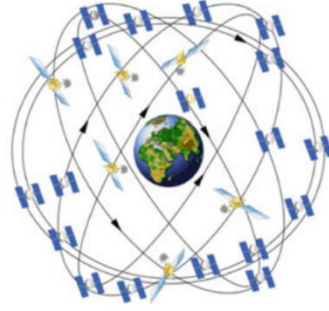
GPS projesi ilk olarak 1973'te ABD Savunma Bakanlığı tarafından başlatılmış ve ilk prototip uzay aracı 1978'de fırlatılmıştır. Ancak 24 uydudan oluşan uydu takımı 1993'de faaliyete geçmiştir (Ali, 2020). Başlangıçta GPS kullanımı Birleşik Devletler ordusunun kullanımıyla sınırlıydı. Daha sonra sivil kullanıma açılmıştır. Sistem, dünya çapındaki askeri, sivil ve ticari kullanıcılara çeşitli alanlarda kritik yetenekler sağlamaktadır. ABD tarafından korunan bu sistem, GPS alıcısı olan herkes tarafından ücretsiz olarak kullanılabilir. GPS, dünyanın her yerinde, farklı hava koşullarında ve tüm gün boyunca çalışmaktadır. GPS; uzay bölümü, kontrol bölümü ve kullanıcı bölümü olmak üzere üç temel kısımdan oluşmaktadır (Şekil 3.12).



Şekil 3.12. Küresel konumlandırma sisteminin temel bölümleri

3.5.1.1. Uzay bölümü

Bu bölüm, dünya yüzeyinden 20.180 km yükseklikte, dairesel bir yörüngede dönen 24 uydu ve yedek uydulardan oluşacak şekilde tasarlanmıştır. Ocak 2017 itibariyle, uzayda 31 operasyonel GPS uydusu vardır. Her uydunun dünya etrafındaki yörüngesi yaklaşık 11 saat 58 dakika sürmektedir. Yani, her bir uydu dünya yörüngesinde günde iki kez dönmektedir. Uzay bölümü; uydu takımı, ekvatorundan 55° eğimli 6 yörünge düzleminden oluşmaktadır ve her yörüngede 4 uydu bulunmaktadır (Şekil 3.13).



Şekil 3.13. GPS uydu takımı (Awange ve Kiema, 2019)

Uyduların kendileri güneş panelleri, dahili bileşenler (atomik saat ve radyo vericileri) ve anten gibi harici bileşenlerden oluşmaktadır. Uydunun uzaydaki yönü, güneş panelleri uyduya güç sağlamak için güneşe bakacak şekilde, antenler ise radyo sinyallerini iletmek ve almak için dünyaya bakacak şekilde konumlanmaktadır.

3.5.1.2. Kontrol bölümü

GPS kontrol bölümü; ana kontrol istasyonu, monitör ve yer istasyonlarından oluşmaktadır. Ana kontrol istasyonu, Colorado Springs’de (Schriever Hava Kuvvetleri Üssü, Colorado) bulunmaktadır ve bir yedekleme istasyonu Gaithersburg, Maryland’dedir. Kontrol istasyonları, Colorado Springs, Atlantik Okyanusundaki Ascension Adası, Hawaii, Hint Okyanusundaki Diego Garcia ve Pasifik Okyanusundaki Kwajalein Adası’nda bulunan beş istasyondan oluşmaktadır. Eylül 2005’te, NGA’nın (National Geospatial-Intelligence Agency) altı izleme istasyonu daha ağı eklenmiştir. Böylece her uydunun en az iki izleme istasyonu tarafından görülmesi sağlanmıştır (Awange ve Kiema, 2019).

Uyduları takip eden bu izleme istasyonlarından elde edilen bilgiler de uyduları kontrol etmek ve yörüngelerini tahmin etmek için kullanılmaktadır. Bu; uydudan gelen efemeris (ephemeris) ve saat parametrelerinin ana istasyon tarafından işlenmesi ve işlenen bu bilgilerin uydulara geri iletilmesi yoluyla yapılmaktadır. Bu durum, uzay bölümü ve kontrol bölümü arasındaki iletişimin çift yönlü olduğunu açık bir şekilde göstermektedir. Ayrıca, kontrol bölümü ile kullanıcı bölümü arasında doğrudan bir iletişim yoktur. Özetle yer kontrol ağı, uydu sağlığını ve uydu yörünge konfigürasyonunu koruyarak uydu takımını izlemek ve gereken düzeltmeleri yapmaktan sorumludur.

3.5.1.3. Kullanıcı bölümü

Bu bölüm GPS alıcısı kısmından oluşmaktadır. GPS alıcısı, uydudan radyo sinyalleri aracılığıyla gelen bilgileri kullanarak konum bilgisini hesaplamaktadır. Uydudan gelen mesaj; uydunun sağlığı, yörünge parametreleri (zamanın bir fonksiyonu olarak efemerislerini yayınlar), genel iyonosferik düzeltmeler, uydu saat düzeltme parametreleri, almanak bilgileri (almanac message) ve diğer uydular hakkında bazı bilgiler içermektedir.

GPS uyduları, 1575.42 MHz (L1 sinyali) ve 1227.60 MHz (L2 sinyali) frekans değerlerinde bilgileri taşımaktadır. Tüm uydular aynı frekansta yayın yapmaktadır. GPS alıcılarının algıladığı sinyallerin hangi uydudan geldiğini anlamaları için uydudan yayınlanan sinyallerin kodlanmaları gerekmektedir. Genel olarak iki farklı kodlayıcı kullanılmaktadır. Bunlar, C/A kodu (Coarse Acquisition Code) ve P kodudur (Precise Code). L1 sinyali hem C/A kodu ile hem de P kodunu taşırken, L2 sinyali sadece P kodunu taşımaktadır. C/A kodu; 1 MHz'de modüle ederken, P kodu 10 MHz'de modüle etmektedir. Ayrıca P kodu askeri uygulamalar için kullanılmaktadır.

Kullanıcı tarafında 2B coğrafi konumun hesaplanabilmesi için minimum üç uydu, 3B coğrafi konumun belirlenebilmesi için ise minimum 4 uydunun kullanılması gerekmektedir. Konumun belirlenmesi için Trilaterasyon işlemi uygulanır. Hedefin her bir uyduya göre mesafesi (d) ise sinyal hızı ($c = 3 \times 10^8 \text{ m/s}$) ve sinyalin iletilmeyle alınma zamanı arasındaki farkın (Δt) çarpımı ile elde edilir (Eşitlik 3.2). Yapılan işlemler sonucunda hesaplanan konum bilgisi WGS84 veri formatında ifade edilmektedir. Ek

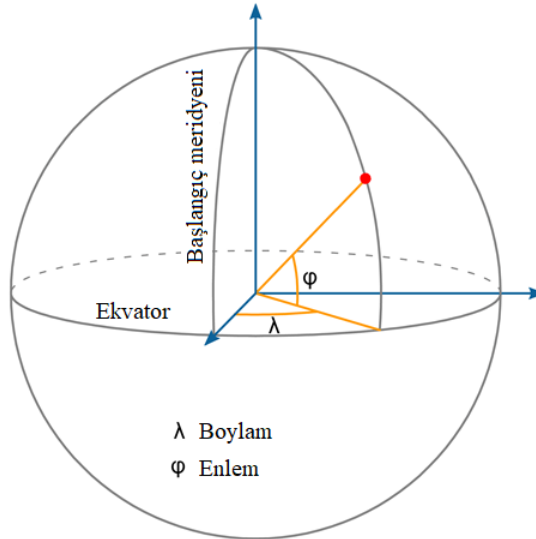
olarak, konum belirlemek için kullanılan uyduların sayısının artması konum bilgisi hassasiyetini arttırmaktadır.

$$d = c * \Delta t \quad (3.2)$$

3.5.2. Dünya jeodezi sistemi 1984 (WGS84)

Dünya Jeodezi Sistemi 1984 (WGS84); haritacılık, jeodezi ve GPS gibi birçok alanda kullanılan bir standarttır. WGS84 koordinat sistemi, dünya üzerindeki bir noktayı enlem, boylam ve elipsoid yüksekliği açısından tanımlamaktadır. WGS84 koordinat sisteminde, dünyanın kütle merkezi orijin olarak ifade edilmiştir.

Şekil 3.14’de dünyanın elipsoid modeli verilmiştir. Bu modeli ifade eden bazı denklemler Eşitlik 3.3, 3.4, 3.5 ve 3.6’da (Kalu ve ark., 2022) verilmiştir. Eşitliklerdeki sembollerde; a elipsoidin ekvator yarı çapını, f_{wgs84} elipsoidin basıklığını, b elipsoidin kutup yarı çapını ve e dış merkezliliği temsil etmektedir. Ayrıca ϕ ve λ sırasıyla enlem ve boylamı göstermektedir.



Şekil 3.14. Dünya Jeodezi modeli

$$a = 6378137.0 \quad (3.3)$$

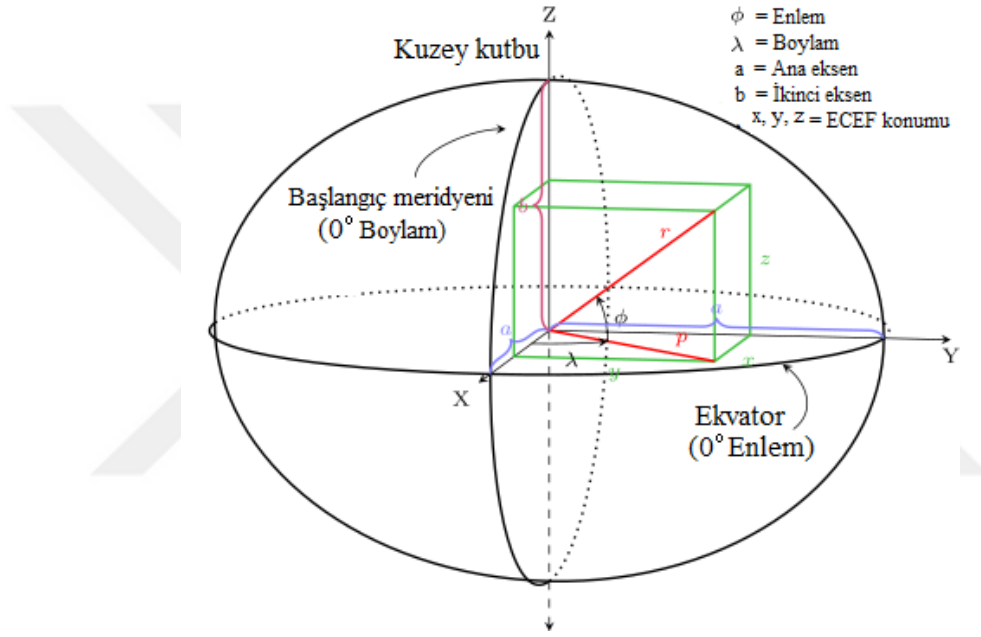
$$f_{wgs84} = 1 / 298.257223563 \quad (3.4)$$

$$b = a * (1 - f_{wgs84}) \quad (3.5)$$

$$e^2 = 2 * f_{wgs84} - f_{wgs84}^2 \quad (3.6)$$

3.5.3. WGS84 koordinatından ECEF koordinatına dönüşüm

ECEF, Jeosentrik koordinat sistemi olarak da bilinmektedir. ECEF koordinat sisteminin merkezi, dünyanın orijini. Dünyanın yüzeyinden merkezine olan herhangi bir konumu x, y, z olarak ifade etmektedir.



Şekil 3.15. ECEF koordinat sistemi

Şekil 3.15’de ECEF koordinat sisteminin temsili verilmiştir. Jeodezik koordinatların ECEF koordinatlarına dönüştürülmesi için Eşitlik 3.8, 3.9 ve 3.10’dan (Allig ve Wanielik, 2019) yararlanılmıştır. R ; eğrilik yarı çapını, h yüksekliği, X_{ecef} Y_{ecef} ve Z_{ecef} ise jeosentrik koordinatları göstermektedir.

$$R = a / \sqrt{1 - e^2 \sin^2 \phi} \quad (3.7)$$

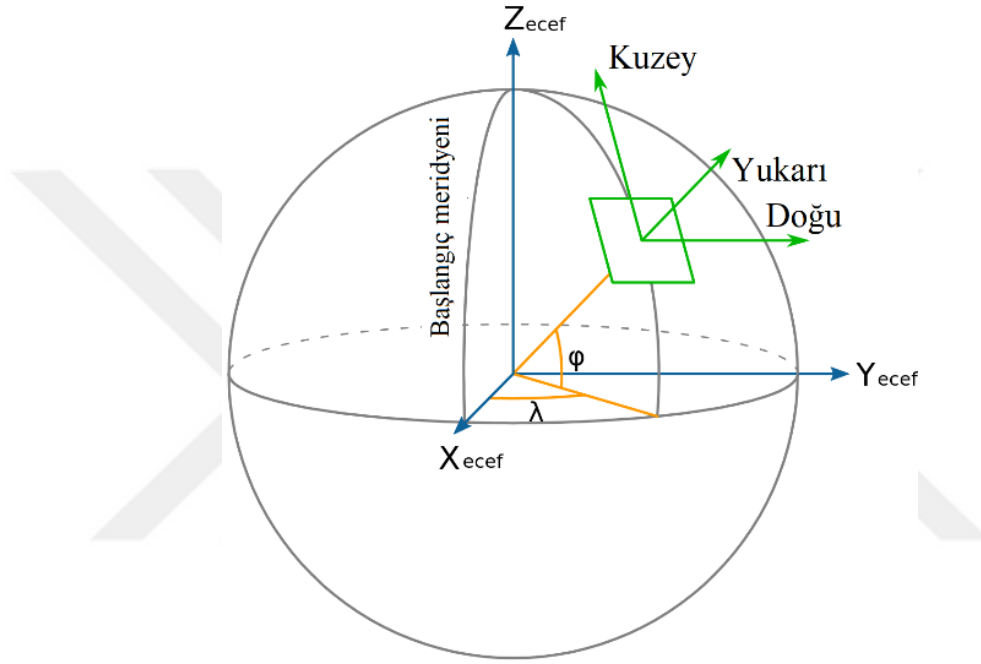
$$X_{ecef} = (R + h) * \cos \phi * \cos \lambda \quad (3.8)$$

$$Y_{ecef} = (R + h) * \cos \phi * \sin \lambda \quad (3.9)$$

$$Z_{ecef} = \left(\frac{b^2}{a^2} R + h \right) * \sin \phi \quad (3.10)$$

3.5.4. ECEF koordinatından ENU koordinatına dönüşüm

Yerel elipsoidal sistem olarak da bilinen ENU koordinat sistemi, teğet düzlemi temel alan bir uzaysal referans sistemidir. Şekil 3.16’da ENU koordinat sistemi gösterilmiştir. Yerel ENU koordinatları, belirli bir konuma sabitlenmiş ve dünya yüzeyine teğet bir düzlemden oluşur. ENU koordinat sistemi; doğu, kuzey ve yukarı olmak üzere üç farklı koordinata sahiptir.



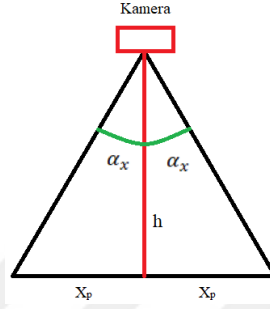
Şekil 3.16. ENU koordinat sistemi

ECEF koordinat sisteminden ENU koordinat sistemine geçiş için Eşitlik 3.11’den (Allig ve Wanielik, 2019) yararlanılmıştır. X_{enu} , Y_{enu} ve Z_{enu} ECEF koordinatlarının yerel ENU koordinatlarına dönüştürülen değerlerini ifade etmektedir. ϕ ve λ , sırasıyla enlem ve boylam açılarıdır. X_{ecef} , Y_{ecef} ve Z_{ecef} ECEF koordinat sisteminden elde edilen değerlerdir. Eşitlik 3.11’de koordinatların farkının alındığı kısım bu çalışma için (0,0,0) olacaktır çünkü GPS alıcısının merkezi koordinatları $(X_{ecef}, Y_{ecef}, Z_{ecef})$ ile hesaplama için kullanılan referans koordinatlar (X_r, Y_r, Z_r) aynı koordinatlardır.

$$\begin{bmatrix} X_{enu} \\ Y_{enu} \\ Z_{enu} \end{bmatrix} = \begin{bmatrix} -\sin \lambda & \cos \lambda & 0 \\ -\sin \phi \cos \lambda & -\sin \phi \sin \lambda & \cos \phi \\ \cos \phi \cos \lambda & \cos \phi \sin \lambda & \sin \phi \end{bmatrix} \begin{bmatrix} X_r - X_{ecef} \\ Y_r - Y_{ecef} \\ Z_r - Z_{ecef} \end{bmatrix} \quad (3.11)$$

3.5.5. Piksel başına düşen metre hesabı

Öncelikle, piksel başına düşen metre hesabı için kameranın zemine sürekli yatay olarak baktığı varsayılmıştır. Bu varsayım, Şekil 3.17 ve Şekil 3.18’de gösterilmiştir. Görüş alanını hesaplamak için üçgenlerden faydalanılmıştır. Eşitlik 3.13 ve Eşitlik 3.15 ile x eksenindeki (FOV_x) ve y eksenindeki (FOV_y) görüş alanı hesaplanabilir.

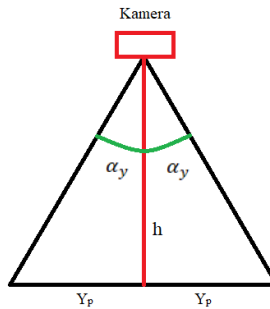


Şekil 3.17. Kameranın x eksenindeki görüş alanı

α_x ve α_y sırasıyla kameranın x eksenindeki görüş açısının yarısını ve y eksenindeki görüş açısının yarısını göstermektedir. X_p ve Y_p ; İHA’nın yüksekliğine (h) bağlı olarak hesaplanan, sırasıyla x eksenindeki ve y eksenindeki görüş alanlarının yarısını ifade etmektedir.

$$X_p = h \tan \alpha_x \quad (3.12)$$

$$FOV_x = 2h \tan \alpha_x \quad (3.13)$$



Şekil 3.18. Kameranın y eksenindeki görüş alanı

$$Y_p = h \tan \alpha_y \quad (3.14)$$

$$FOV_y = 2h \tan \alpha_y \quad (3.15)$$

Yerel koordinatlardaki değişimi bulabilmek için Eşitlik 3.16 ve Eşitlik 3.17'den yararlanılarak x ekseninde (MPP_x) ve y ekseninde (MPP_y) piksel başına düşen metre hesabı yapılmalıdır. r_x , kameranın x eksenindeki toplam piksel sayısını gösterirken r_y kameranın y eksenindeki toplam piksel sayısını ifade etmektedir.

$$MPP_x = FOV_x / r_x \quad (3.16)$$

$$MPP_y = FOV_y / r_y \quad (3.17)$$

Eğer kameranın görüş açısı verilmemiş ise Eşitlik 3.18 ve Eşitlik 3.19'dan faydalanılarak kameranın x ve y eksenlerindeki görüş açılarının yarısı (α_x ve α_y) hesaplanabilir. f , kameranın odak uzaklığıdır. d_x ve d_y kameranın sensör boyutlarıdır.

$$\alpha_x = \tan^{-1} \frac{d_x}{2f} \quad (3.18)$$

$$\alpha_y = \tan^{-1} \frac{d_y}{2f} \quad (3.19)$$

3.5.6. Yerel Koordinatlardaki değişimin hesaplanması

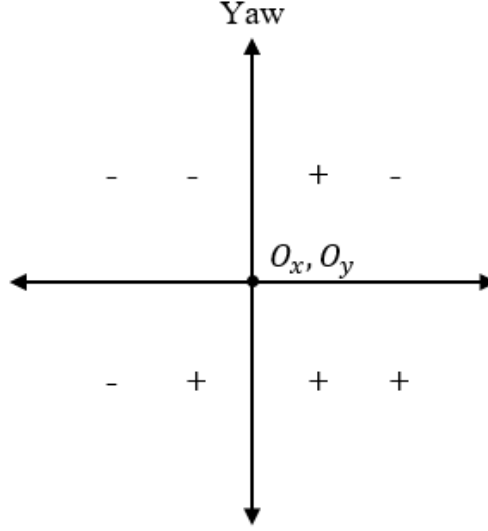
Yerel koordinatlardaki değişimin hesaplanması için tespit edilen nesnenin merkezi piksel koordinatlarından (C_x, C_y) ve görüntünün merkezi piksel koordinatından (O_x, O_y) faydalanılmaktadır. Değişimin hesaplanması için ilk olarak yataydaki (PD_x) ve dikeydeki (PD_y) piksel mesafeleri hesaplanmıştır. Daha sonra bu bilgilerden yararlanılarak nesne merkezinden görüntü merkezine olan mesafe bilgisi (PD_h) elde edilmiştir.

$$PD_x = O_x + C_x \quad (3.20)$$

$$PD_y = O_y + C_y \quad (3.21)$$

$$PD_h = \sqrt{PD_x^2 + PD_y^2} \quad (3.22)$$

Görüntü koordinat sisteminde (Şekil 3.19) tespit edilen nesnenin açısı (α) Eşitlik 3.23'den yararlanılarak bulunmuştur. Ancak bu işlem tespit edilen nesnenin gerçek dünya açısını tam anlamıyla doğru hesaplayamaz. Bu nedenle, Eşitlik 23'ten sonra Eşitlik 24 kullanılarak nesnenin dünya koordinat sistemindeki yaw açısı (α_0) bulunur.



Şekil 3.19. Görüntü koordinat sistemi

$$\alpha = \tan^{-1} \frac{PD_x}{PD_y} \quad (3.23)$$

$$\alpha_0 = \begin{cases} PD_y \geq 0, \psi + \alpha \\ PD_y < 0, \psi + \alpha + 180 \end{cases} \quad (3.24)$$

Son olarak, yerel koordinat düzleminin koordinatlarındaki değişimi hesaplamak için Eşitlik 3.25, Eşitlik 3.26 ve Eşitlik 3.27'den yararlanılmıştır. Ayrıca, bu çalışma kapsamında İHA'nın sabit yükseklikte ve yer düzlemine yatay olarak uçtuğu varsayılmıştır. Bu nedenle, z ekseninde herhangi bir ekleme veya çıkarma işlemi yapılmamıştır.

$$xEast = X_{enu} + PD_h * MPP_x * \cos(\alpha_0) \quad (3.25)$$

$$yNorth = Y_{enu} + PD_h * MPP_y * \sin(\alpha_0) \quad (3.26)$$

$$zUp = Z_{enu} \quad (3.27)$$

3.5.7. ENU koordinatından ECEF koordinatına dönüşüm

Eşitlik 3.28 (Ye ve ark., 2022), ENU koordinatlarını ECEF koordinatlarına dönüştürmeyi sağlamaktadır. X_{ecef} , Y_{ecef} ve Z_{ecef} ENU koordinat sisteminden ECEF koordinat sistemine çevrilmiş değerleri temsil etmektedir. $xEast$, $yNorth$ ve zUp yerel koordinat düzlemindeki değişimleri göstermektedir. X_0 , Y_0 ve Z_0 ise önceden

hesaplanmış ECEF koordinatlarıdır. Böylece yereldeki koordinat değişimleri hesaba katılarak yeni ECEF değerleri üretilmiştir.

$$\begin{bmatrix} X_{ecef} \\ Y_{ecef} \\ Z_{ecef} \end{bmatrix} = \begin{bmatrix} -\sin\lambda & -\sin\phi \cos\lambda & \cos\phi \cos\lambda \\ \cos\lambda & -\sin\phi \sin\lambda & \cos\phi \sin\lambda \\ 0 & \cos\phi & \sin\phi \end{bmatrix} \begin{bmatrix} xEast \\ yNorth \\ zUp \end{bmatrix} + \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} \quad (3.28)$$

3.5.8. ECEF koordinatından WGS84 koordinatına dönüşüm

ECEF koordinatlarından WGS84 koordinat sistemine dönüşüm için aşağıdaki verilmiş eşitlikler sırayla uygulanmıştır (Senapati ve ark., 2020). Eşitlik 3.42 ile yükseklik verisi (h), Eşitlik 3.43 ile enlem (ϕ) verisi, Eşitlik 3.44 ile boylam (λ) verisi elde edilir.

$$e'^2 = (a^2 - b^2)/b^2 \quad (3.29)$$

$$p_p = \sqrt{X_{ecef}^2 + Y_{ecef}^2} \quad (3.30)$$

$$F = 54b^2Z_{ecef}^2 \quad (3.31)$$

$$G = p_p^2 + (1 - e^2)Z_{ecef}^2 - e^2(a^2 - b^2) \quad (3.32)$$

$$c_c = (e^4 F p_p^2)/G^3 \quad (3.33)$$

$$s = \sqrt[3]{1 + c_c + \sqrt{c_c^2 + 2c_c}} \quad (3.34)$$

$$k_k = s + 1 + 1/s \quad (3.35)$$

$$P = F/3k_k^2 G^2 \quad (3.36)$$

$$Q = \sqrt{1 + 2e^4 P} \quad (3.37)$$

$$r_0 = -\frac{Pe^2 p_p}{1+Q} + \sqrt{\frac{1}{2}a^2 \left(1 + \frac{1}{Q}\right) - \frac{P(1-e^2)Z_{ecef}^2}{Q(1+Q)} - \frac{1}{2}Pp_p^2} \quad (3.38)$$

$$U = \sqrt{(p_p - e^2 r_0)^2 + Z_{ecef}^2} \quad (3.39)$$

$$V = \sqrt{(p_p - e^2 r_0)^2 + (1 - e^2)Z_{ecef}^2} \quad (3.40)$$

$$z_0 = b^2 Z_{ecef}/aV \quad (3.41)$$

$$h = U \left(1 - \frac{b^2}{aV}\right) \quad (3.42)$$

$$\phi = \arctan[(Z_{ecef} + e'^2 z_0)/p_p] \quad (3.43)$$

$$\lambda = \arctan2[Y_{ecef}, X_{ecef}] \quad (3.44)$$

3.6. Görüntü Mozaikleme

Görüntü mozaikleme, görüntülerin birbirleri ile benzer kısımlardan veya GPS ve IMU gibi sensörlerden yararlanarak birleştirilen görüntülerden daha geniş görüntüler elde etmek amacıyla kullanılmaktadır. Bu çalışma kapsamında görüntüleri mozaiklemek için özellik tabanlı yaklaşımlar kullanılmıştır. Önerilen yaklaşım ilk olarak özellik çıkarma ile başlamaktadır. Daha sonra çıkarılan özellikler eşleştirilmiştir. Yanlış çıkarılan özellikleri elemek için RANSAC algoritması kullanılmıştır. Aykırı özellikler elendikten sonra kalan özellik çiftlerinden homografi hesabı yapılmıştır. Son olarak görüntüler hizalanmıştır.

3.6.1. Özellik çıkarma

Özellik çıkarma, görüntü mozaiklemenin ilk aşamasıdır. Özellik çıkarmak için Harris köşe tespiti, SIFT, SURF, FAST, BRIEF ve ORB, gibi çeşitli algoritmalar kullanılmaktadır. Bu çalışma da SIFT, SURF, BRIEF ve ORB algoritmalarının incelenmesi gerçekleştirilmiştir.

3.6.1.1. SIFT algoritması

Lowe D.G. (2004) tarafından sunumu gerçekleştirilen SIFT algoritması bir tür özellik çıkarma algoritmasıdır. Bu algoritma, görüntülerin dönmesi ve ölçeklenmesi gibi durumlarda görüntüler üzerindeki değişmeyen özellikleri bulmayı amaçlamıştır. Temelde 4 ana adımdan oluşmaktadır:

- 1) Ölçek uzay tepe noktası tespiti
- 2) Anahtar noktaları yerelleştirme
- 3) Yönlendirme ataması
- 4) Anahtar nokta tanımlayıcı

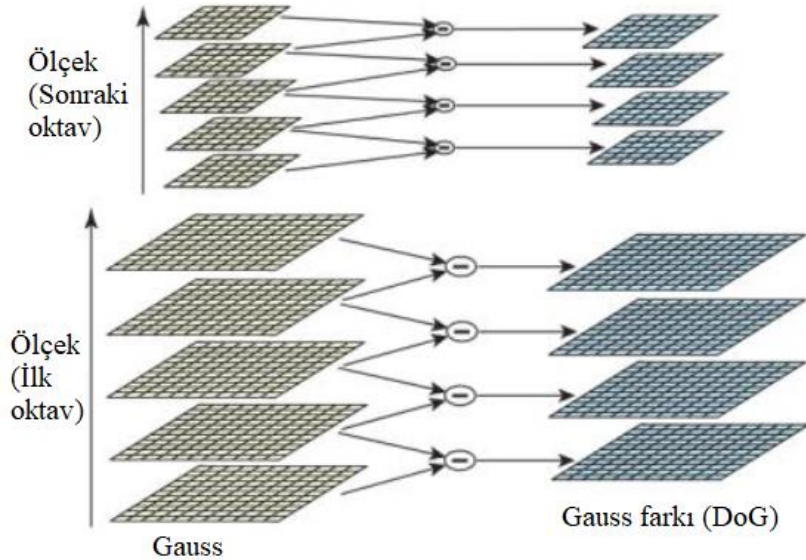
3.6.1.1.1. Ölçek uzay aşırılık tespiti

İlk aşamada, özellik tespiti için ilgi noktalarının tespiti yapılmaktadır. Farklı ölçeklerde bulunan anahtar noktaları tespit etmek için sürekli aynı boyutta matris kullanmak sonuçları olumsuz yönde etkilemektedir. Bu nedenle farklı ölçeklerdeki uzay filtresi kullanılmaktadır. Yani, orijinal görüntünün boyutları 4 kere daha yarıya indirilir ve toplamda 5 farklı çözünürlük değerinde görüntüler elde edilir. Sonra her bir çözünürlükteki görüntüye ($I(x, y)$) 5 farklı bulanıklık seviyesinde ($G(x, y, k \sigma)$) tek tek Gauss filtresi (Eşitlik 3.46) uygulanır. Aynı çözünürlüğe sahip Gauss filtresi uygulanmış görüntü ($L(x, y, k \sigma)$) grubu 1 oktav olarak nitelendirilir. Şekil 3.20’de görüldüğü gibi aynı çözünürlük değerine sahip ve Gauss filtresi uygulanmış görüntülerin farkları (DoG) ($D(x, y, \sigma)$) alınmıştır (Eşitlik 3.47). Bu işlem, görüntülerin aday kilit noktaları bulmak kullanılmaktadır.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (3.45)$$

$$L(x, y, k \sigma) = G(x, y, k \sigma) * I(x, y) \quad (3.46)$$

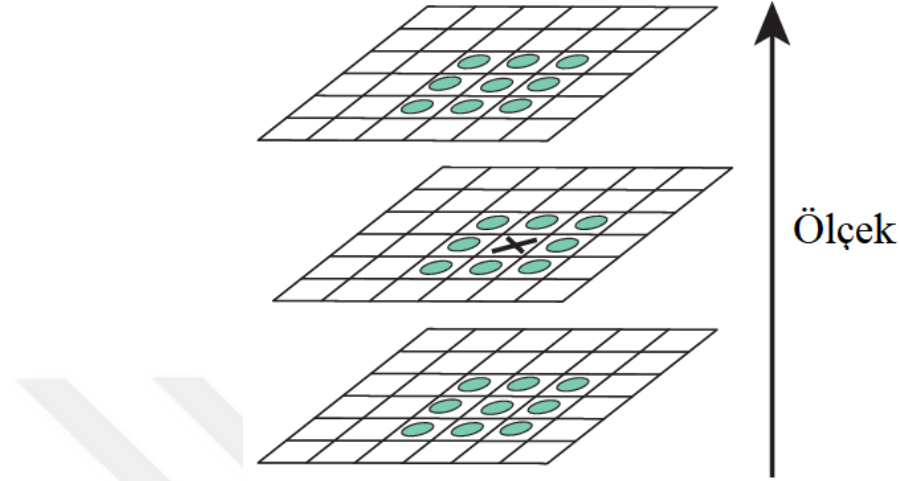
$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3.47)$$



Şekil 3.20. DoG’un (Difference of Gaussian) elde edilmesi (Lowe D.G., 2004)

Aday anahtar noktaları bulmak için iki aşamalı bir süreç uygulanır. İlk olarak DoG görüntülerinden maksimum ve minimum değerler bulunur. Aynı ölçekte bir pikseli (x)

26 piksel ile karşılaştırarak ölçek uzayındaki maksimum ve minimumları değerleri bulunur. Her nokta, mevcut görüntüdeki 8 komşusu, üstteki ve alttaki ölçeklerde (Şekil 3.21) her bir 9 komşusu ile karşılaştırılır. Böylece aday noktalar tespit edilir.



Şekil 3.21. Gauss farkı görüntülerinin maksimum ve minimumları (Lowe D.G., 2004)

3.6.1.1.2. Kilit nokta yerelleştirme

Bu aşamaya kadar aday kilit noktalar üretilmiştir. Ancak, bazıları bir kenar boyunca uzanmaktadır. Bazıları ise yeterli kontrasta sahip değildir. Her iki durumdaki noktalar özellik olarak kullanışlı değildir ve elenmesi gerekmektedir. Düşük kontrasta sahip noktaların elenmesi için 2. dereceden Taylor serisinden (Eşitlik 3.49) yararlanılmış ve DoG görüntülerinde uygulanmıştır. Eğer hesaplanan değer 0.03'ün altında ise aday kilit nokta elenmiştir.

$$x = (x, y, \sigma) \quad (3.48)$$

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \quad (3.49)$$

Ana kilit noktalar için son olarak kenarlara yanıt veren noktalar elenmelidir ve bunun için Hessian matristen (Eşitlik 3.50) yararlanılmıştır. Eğer $r > 10$ (Eşitlik 3.53) ise nokta elenir. D_{xx} , D_{xy} ve D_{yy} ilgili yönlerdeki ikinci dereceden Gauss türevlerini göstermektedir.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (3.50)$$

$$TR(H) = D_{xx} + D_{yy} = \lambda_1 + \lambda_2 \quad (3.51)$$

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \lambda_1\lambda_2 \quad (3.52)$$

$$r = \lambda_1/\lambda_2 \quad (3.53)$$

3.6.1.1.3. Yönlendirme ataması

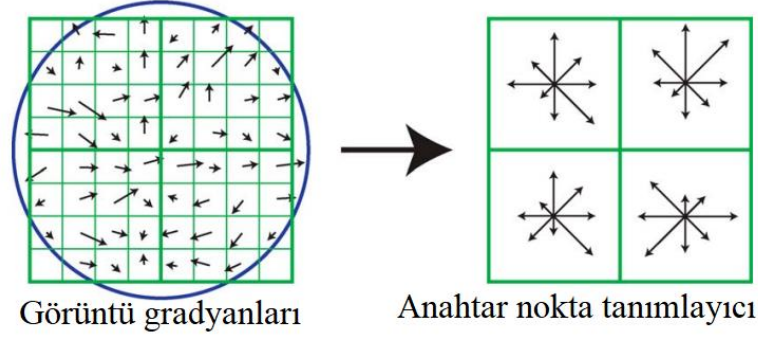
Görüntü döndürmede değişmezliği sağlamak için her bir anahtar noktaya bir yönlendirme atanır. Görüntünün ilgilenilen kısımları $L(x,y)$ olarak gösterilmekle birlikte, spesifik koordinatların çevresinde bulunan tüm pikseller arasındaki değişiklikler dikkate alınarak, gradyan büyüklüğü $m(x,y)$ ve yönelimi $\theta(x,y)$, Eşitlik 3.54 ve Eşitlik 3.55 kullanılarak hesaplanmaktadır. Her bir bölmeye 10 derece düşecek şekilde toplamda 36 dilime sahip bir histogram oluşturulur. Histogramda bulunan en uç nokta elde edilir ve bunun %80'in üzerindeki herhangi bir tepe noktası da oryantasyonu hesaplamak için kabul edilir. Aynı konum ve ölçekte, ancak farklı yönlerde anahtar noktalar edinilir.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (3.54)$$

$$\theta(x,y) = \tan^{-1} \left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \right) \quad (3.55)$$

3.6.1.1.4. Anahtar nokta tanımlayıcı

Bir anahtar nokta tanımlayıcısı Şekil 3.22'de gösterildiği gibi, anahtar nokta konumu çevresindeki bir bölgedeki her bir görüntü örnek noktasındaki gradyan büyüklüğünü ve oryantasyonunu hesaplayarak oluşturulur. Bunlar, kaplanmış daire ile gösterilen bir Gauss penceresi ile ağırlıklandırılır. Bu örnekler daha sonra, 4×4 alt bölgeler üzerindeki içerikleri özetleyen yönlendirme histogramlarında toplanır ve her bir okun uzunluğu, bölge içinde o yöne yakın gradyan büyüklüklerinin toplamına karşılık gelmektedir.



Şekil 3.22. Özellik noktalarının tanımlayıcısı (Lowe D.G., 2004)

3.6.1.2. SURF algoritması

SURF yöntemi (Speeded Up Robust Features), SIFT algoritması ile kıyaslandığında oldukça daha az zaman alan ve kaliteli bir algoritmadır. Kutu filtreler, SURF algoritması için önemli bir ilgi alanıdır çünkü hızlı çalışmasının altında bu sebep bulunmaktadır. Bu durum, nesne tespiti ve takibi gibi anlık olarak çalışan yeni algoritmaların kullanılmasına imkân vermektedir.

SURF algoritması, ilke olarak SIFT algoritmasına benzer adımlardan oluşmaktadır. Ancak bazı değişiklikler yapılmıştır. SURF algoritması özellik noktalarını tespit ederken gauss görüntülerini kullanma yerine, integrali alınan görüntüleri ($I_{\Sigma}(x)$) kullanmıştır. İntegral görüntüsünü elde etmek için (Eşitlik 3.56) orijinal görüntünün gri değerlerini toplar ve SIFT algoritmasında kullanılan Gauss filtresindeki gibi gri uzayına entegre değerleri ile değiştirir. Böylece özellik noktalarının daha kısa sürede elde edilmesini sağlamaktadır.

$$I_{\Sigma}(x) = \sum_{i=0}^{i \leq x} \sum_{j=0}^{j \leq y} I(i, j) \quad (3.56)$$

SURF algoritması, ilgi noktalarını tespit etmek için Hessian matrisinden (Eşitlik 3.58) yararlanmış ve hesaplama süresini azaltmak için DoG yerine Laplasyen Gauss (LoG) kullanmıştır. $L_{xx}(x, \sigma)$, x noktasında Gauss'un ikinci dereceden türevinin konvolüsyon sonucudur (Eşitlik 3.57). Benzer şekilde $L_{xy}(x, \sigma)$ ve $L_{yy}(x, \sigma)$ de bulunabilir. Determinant hesabı SIFT algoritmasından biraz farklı olarak Eşitlik 3.59'daki gibi hesaplanmaktadır.

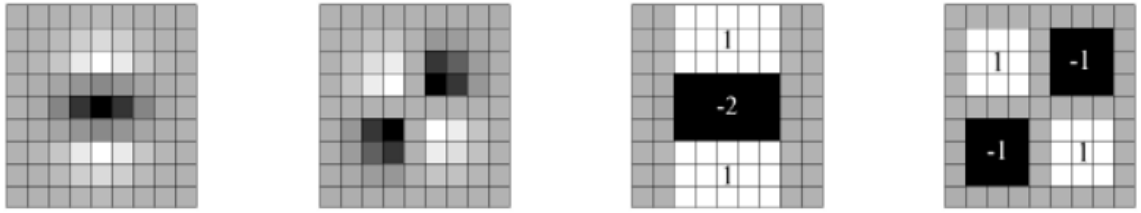
$$L_{xx}(x, \sigma) = I(x) * \frac{d^2}{dx^2} g(\sigma) \quad (3.57)$$

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (3.58)$$

$$Det(H_{yaklaşık}) = D_{xx}D_{yy} - (0.9D_{xy})^2 = \lambda_1\lambda_2 \quad (3.59)$$

İntegral görüntülerinin avantajından yararlanmak için filtreler kutu filtrelerle dönüştürülmüştür. Bu durum bir miktar performans kaybına neden olmaktadır. Ancak, işlem karmaşıklığını azalttığı için avantajı daha ağır basmaktadır çünkü İntegral görüntülerin kullanılması, hesaplama süresini filtre boyutundan bağımsız hale getirmektedir.

Şekil 3.23’de 9×9 boyutlarında kutu filtreler ve $\sigma = 1.2$ olan Gauss ikinci dereceden türevlerinin yaklaşık değerleri verilmiştir. Bu kutu filtreler uzayın en küçük seviyesini oluşturmaktadır. Kutu filtre içindeki siyah bölgeler negatifi, gri bölgeler sıfırı ve beyaz bölgeler pozitifini temsil etmektedir.

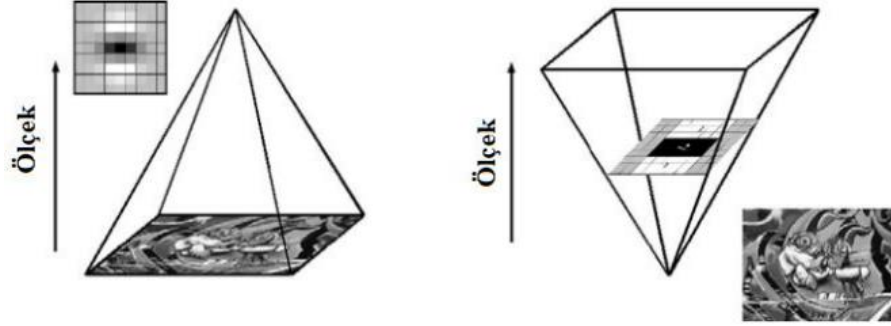


Şekil 3.23. Kutu filtreleri ve kısmi türev (Bay ve ark., 2008)

SURF algoritması, özelliklerin tespitinde aynı zamanda ölçek uzayından faydalanmaktadır. Ölçek uzayları genellikle Şekil 3.24’deki gibi farklı boyutlara sahip görüntü zinciri olarak uygulanır. Görüntüler, piramidin daha yukarı bir seviyesine ulaşmak için alt örneklenir. Ancak bundan önce görüntüler Gauss’dan faydalanılarak düzleştirilir.

Kutu filtrelerin ve entegre görüntülerin kullanılması nedeniyle SURF, daha önce filtre uygulanmış bir katmanın sonucuna farklı filtreleri uygulamasına imkân vermektedir. Böylece, çeşitli boyutlarda bulunan filtreler, benzer hızlarda direkt ham görüntü üzerinde denenebilir. Bu sebeple, ölçekleme uzayı, görüntülerdeki boyutu devamlı bir şekilde ufaltmak gibi bir yaklaşım yerine, 9×9, 15×15, 21×21 ve 27×27 boyutlarındaki gibi yukarı ölçekleyerek incelemektedir. Böylelikle oktavlar, filtre boyutlarında olan artış eş zamanlı

olarak ikiye katlanmaktadır. Belirli noktaların elde edilmesi için örnekleme mesafeleri de ikiye katlanabilir. Bu olay, filtrenin sabit bir bedelle büyütülmesine olanak tanımaktadır.



Şekil 3.24. Piramitsel ölçek uzayı

Belirlenen kilit noktalarının yöndeki değişmezliğini sağlamak için yön ataması yapılmalıdır. Bunun için x ve y yönlerinde Haar dalgacık yanıtları hesaplanmaktadır. Bunu, anahtar noktanın etrafındaki $6s$ yarıçapında dairesel bir komşulukta hesaplar. Ayrıca, örnekleme adımı ölçeğe bağlıdır ve s olarak seçilir ve dalgacık yanıtları o geçerli ölçekte (s) hesaplanmaktadır. Buna göre, yüksek ölçeklerde dalgacıkların boyutu büyüktür. Bu nedenle, hızlı filtreleme için integral görüntüleri tekrar kullanılır. Her $\pi/3$ 'lük açıda yönelim hesabı yapılır. Bulunan en büyük değer ana yönelimi temsil etmektedir.

Son olarak anahtar noktaların tanımlayıcı bileşenleri belirlenmelidir. Bu aşamada, kilit nokta etrafında ortalananmış ve zaten elde edilen oryantasyon boyunca yönlendirilmiş bir kare bölge oluşturmaktan ibarettir. Bu pencerenin boyutu $20s$ 'dir. Daha sonra bölge düzenli olarak daha küçük 4×4 kare alt bölgelere bölünür ve her bölge için hesaplanan dalgacık yanıtları Gauss ile ağırlıklandırılır. Böylece, $4 \times 4 \times 4$ boyutunda anahtar noktanın tanımlayıcı bileşenleri elde edilir.

3.6.1.3. BRIEF algoritması

Özellik noktası tanımlayıcıları; nesne tanıma, 3B yeniden oluşturma, görüntü alma ve kamera yerelleştirme gibi birçok bilgisayarlı görü teknolojisinin merkezinde yer almaktadır. Bu teknolojilerin uygulamalarının her zamankinden daha fazla veriyi işlemesi veya sınırlı hesaplama kaynaklarına sahip mobil cihazlarda çalışması gerektiğinden,

hesaplaması hızlı, eşleşmesi hızlı ve belleği verimli kullanan yerel tanımlayıcılara artan bir ihtiyaç bulunmaktadır.

BRIEF, etkili bir özellik nokta tanımlayıcısı olarak ikili dizeleri kullanan bir yapıya sahiptir. BRIEF hem oluşturma hem de eşleştirme için çok hızlıdır. Birçok durumda hız ve tanıma oranı açısından SURF ve SIFT gibi diğer hızlı tanımlayıcılardan kolayca daha iyi performans göstermektedir.

Özellik tanımlayıcıları, ilginç bilgileri bir dizi sayıya kodlamaktadır ve bir özelliği diğerinden ayırt etmek için kullanılabilecek bir tür sayısal parmak izi görevi görmektedir. Piksel (anahtar nokta) etrafında tanımlanan komşular, kare piksel genişliği ve yüksekliğine sahip bir yama (merkezi özellik noktası olan çekirdek matris) olarak bilinir.

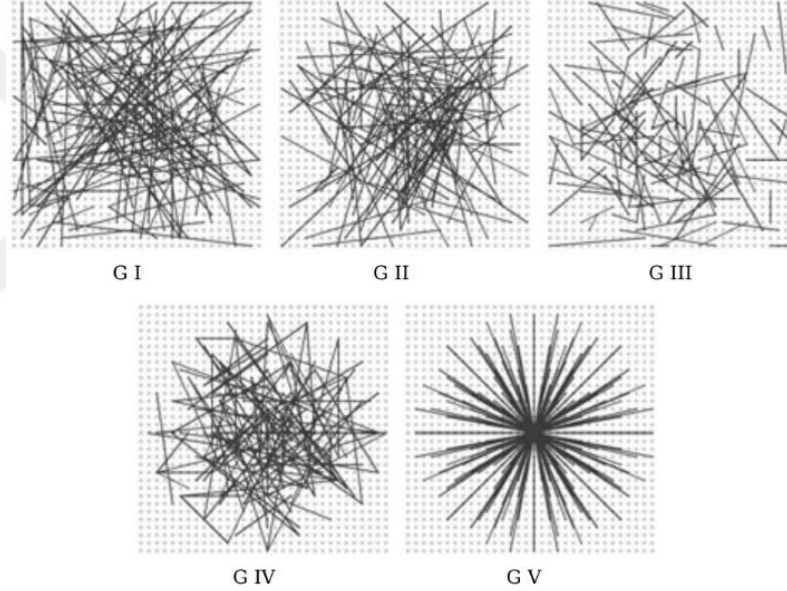
BRIEF algoritması piksel bazlı işlem yaptığı için gürültüye karşı çok duyarlıdır. Bu nedenle, ilk olarak görüntüdeki gürültüleri gidermek için yumuşatma filtresi uygulanır. Daha sonra görüntü yamalarını, nispeten az sayıda olan ikili yoğunluk karşılaştırması temelinde etkili bir şekilde sınıflandırabilmek için ikili bir özellik vektörüne dönüştürülür. İkili özellik tanımlayıcısı olarak da bilinen ikili özellik vektörü, piksel yoğunluğuna ($p(x)$ ve $p(y)$) göre yalnızca 1 ve 0 içeren bir özellik vektörüdür. Yani, Eşitlik 3.60'ın sonucunda oluşan vektördür. Her bir özellik vektörü, 128, 256 veya 512 uzunluğunda bir dizi olarak tanımlanabilir.

$$\tau(p; x, y) := \begin{cases} 1: p(x) < p(y) \\ 0: p(x) \geq p(y) \end{cases} \quad (3.60)$$

Toplam bir ikili özellik vektörü oluşturmak için rastgele çiftler seçilir ve bu çift beş yaklaşımdan biri kullanılarak seçilir (Şekil 3.25). Yama boyutunun ($S \times S$) olduğu ve kilit noktanın yamanın merkezinde olduğu varsayılırsa:

- 1) Tek düze yaklaşım (G I): Anahtar nokta ($S/2$) etrafında seçilen örnek çiftleri rastgele örneklenir. Seçilen örnek çiftleri yama sınırına yakın uzanabilmektedir.
- 2) Gauss yaklaşımı (G II): Rastgele çiftteki hem x hem de y pikselleri, anahtar nokta etrafında $0,04 * S^2$ 'lik bir Gauss dağılımından veya yayılmasından çizilir. Bu, yamanın merkezine daha yakın konumların tercih edildiği anlamına gelmektedir.

- 3) Gauss yaklaşımı 2 (G III): Rastgele çiftteki ilk piksel (x), $0,04 * S^2$ 'lik bir yayılma ile anahtar nokta etrafında ortalananmış bir Gauss dağılımından çizilir. Rastgele çiftteki ikinci piksel (y), $0,01 * S^2$ 'lik bir yayılma ile birinci piksel (x) etrafında ortalananmış bir Gauss dağılımından çizilir. Bu, testi daha yerel olmaya zorlar. Yamanın dışındaki test konumları yamanın kenarına kenetlenir.
- 4) Kaba kutupsal ızgara yaklaşımı (G IV): Rastgele çiftteki hem x hem de y pikselleri, bir uzamsal niceleme getiren kaba bir kutup ızgarasının ayrı konumlarından örneklenir.
- 5) Kaba kutup ızgarası yaklaşımı 2 (G V): Rastgele çiftteki ilk piksel (0, 0) noktasında ve rastgele çiftteki ikinci piksel (y), kaba bir kutup ızgarasının ayrı konumlarından çizilir.



Şekil 3.25. İkili özellik çiftlerinin belirlenmesi için kullanılan beş farklı yaklaşımın gösterimi (Calonder ve ark., 2010)

BRIEF algoritması özelliklerin tanımlamak için kullanılmaktadır. Bu nedenle, özellik tespiti için bir özellik tespit algoritması ile birlikte kullanılmalıdır. Bu çalışmada özellik tespiti için STAR algoritmasından yararlanılmıştır. STAR algoritması, CenSurE'den türetilen bir özellik dedektörüdür. Ancak bir daireye yaklaşmak için kareler, altıgenler ve sekizgenler gibi çokgenleri kullanan CenSurE'den farklı olarak birbiriyle örtüşen 2 kareye sahip bir daireyi taklit etmektedir (dik ve 45° döndürülmüş). Bu çokgenler iki seviyelidir ve kalın kenarlı çokgenler olarak

görülebirlirler. STAR algoritması, diğler ölçek uzay dedektörlerinden daha iyi hesaplama özelliklerine sahiptir.

3.6.1.4. ORB algoritması

Oriented FAST and Rotated BRIEF (ORB), 2011 yılında Rublee E. ve arkadaşları tarafından OpenCV laboratuvarlarında SIFT ve SURF'a etkili ve uygulanabilir bir alternatif olarak geliştirilmiştir. ORB esas olarak SIFT ve SURF patentli algoritmalar olduđu için tasarlanmıştır.

ORB, özellik algılama görevinde SIFT'e göre iki kat daha hızlı performans göstermektedir (Rublee E. Ve ark., 2011). ORB algoritmasının temelinde, iyi bilinen FAST keypoint detektörü ve BRIEF tanımlayıcısı yer almaktadır. Bu tekniklerin her ikisi de iyi performansları ve düşük maliyetleri nedeniyle çekicidir. ORB algoritmasının başlıca katkıları şunlardır:

- FAST'a hızlı ve doğru yönlendirme bileşeninin eklenmesi
- Yönlendirilmiş BRIEF özelliklerinin verimli hesaplanması
- Yönlendirilmiş BRIEF özelliklerinin varyans ve korelasyon analizi

ORB algoritması özellikleri tespit etmek için FAST yönteminden yararlanmaktadır. FAST algoritması ilk olarak bir piksel seçmektedir ve bu merkez piksel olarak adlandırılmaktadır. Daha sonra yarı çapı 3 olacak şekilde merkez pikselin çevresinde 16 piksel daha seçilir. Merkez pikselin özellik noktası olup olmadığına karar vermek için 16 piksel, gri seviyede merkez piksel ile karşılaştırılır. Eğer en az 9 (genelde 12 olarak seçilir) piksel merkez pikselden fazla veya az ise merkez nokta özellik noktası olarak belirlenmektedir.

FAST algoritması, köşe noktasının hesaplanması için yalnızca pikseller arasındaki parlaklık farkını karşılaştırmaktadır. Yani yön bileşeni ve büyüklüğü bulunmamaktadır. Bu nedenle, ORB algoritması, orijinal FAST köşe noktaları için Harris yanıt değerlerini hesaplayan ve bunları gri değere göre sıralayan ve ilk N puanını alan orijinal FAST algoritmasını geliştirmiştir. Harris yanıt değeri R_h Eşitlik 3.61 ve Eşitlik 3.62'den yararlanılarak hesaplanmaktadır. M , 2x2 matristir ve k , 0,04 ile 0,06 arasında

değişmektedir. $w(x, y)$ görüntü penceresi işlevidir, I_x özellik noktasının yatay yöndeki değişimidir ve I_y , özellik noktasının dikey yöndeki değişimidir.

$$R_h = \det(M) - k(\text{trace}(M))^2 \quad (3.61)$$

$$M = \sum w(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (3.62)$$

Özellikleri çıkarmak ve özellik noktalarına ölçek değişmezliği eklemek için görüntü ölçeği piramitleri kullanılmıştır. Çıkarılan öznitelik noktalarının dönme değişmezliğine sahip olmasını sağlamak için Yoğunluk Merkezi yöntemi (Intensity Centroid method) kullanılarak öznitelik noktalarının yönü elde edilir. İlk olarak, küçük bir görüntü bloğunun momenti (m_{pq}) Eşitlik 3.63'deki gibi hesaplanır. $I(x, y)$ karşılık gelen pikselin gri uzaydaki değeridir, x ve y piksel koordinatlarının ifade etmektedir. Ardından, görüntü bloğunun ağırlık merkezini (C) bulunmalıdır (Eşitlik 3.64). Görüntü bloğunun kütlesi m_{00} iken m_{10} ve m_{01} görüntü bloğunun ağırlık merkezidir. Son olarak, özellik noktasının yönü (θ) için Eşitlik 3.65'den yararlanılır. Bu adımlar aracılığıyla, FAST köşe noktaları, farklı görüntülerde sağlamlıklarını büyük ölçüde artıran ölçek değişmezliğine ve dönüş değişmezliğine sahiptir.

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y), \quad p, q = \{0, 1\} \quad (3.63)$$

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (3.64)$$

$$\theta = \text{atan2}(m_{01}, m_{10}) \quad (3.65)$$

Özellikler belirlendikten sonra özelliklerin tanımlayıcıları bulunmalıdır. ORB algoritması özellik tanımlayıcıları bulmak için BRIEF yönteminden faydalanmaktadır. Bu algoritmanın ayrıntıları bir önceki başlıkta verilmiştir.

3.6.2. Özellik eşleştirme

Görüntüler arasındaki ilişkiyi tespit etmek için özellik eşleştirme algoritmalarından faydalanılmaktadır. Literatürde özellikleri eşleştirmek için genelde brute-force matcher ve FLANN (Fast Library for Approximate Nearest Neighbors)

matcher algoritmaları kullanılmaktadır. Bu tez çalışmasında özellikleri eşleştirmek için brute-force algoritması kullanılmıştır.

Brute-force eşleştiricisi basit bir yapıya sahiptir ve özellikleri eşleştirmek için yapısında 2 küme bulundurmaktadır İlk kümede özelliklerin tanımları yer almaktadır. İkinci kümede ise Hamming mesafe hesabı kullanılarak diğer tüm özelliklerle eşleştirilir ve en yakın olan özellik çiftleri döndürülür. Böylece özellik çiftleri elde edilir.

3.6.3. RANSAC (RANdom SAmple Consensus)

RANSAC, geometrik dönüşüm (homografi) öncesi yanlış eşleştirilen özellik çiftlerinin elenmesi amacıyla kullanılan bir algoritma olup herhangi bir sebebe bağlı olmadan seçilen alt kümelerin kullanılmasına dayanmaktadır.

Algoritma, daha önce tespit edilmiş özellik çiftlerinin girişe verilmesi ile başlamaktadır. Sonra, rastgele bir uyarlama modeli seçilir ve seçilen bu uydurma modeli arasında kalan özellik noktaları uyarlama modeline belirlenen eşikten yakınsa oy verme hakkına sahip olur ve en uygun uyarlama modeli olduğuna dair oy verir. Tek bir örnekleme süreci bu şekilde gerçekleşmektedir. Rastgele örnekleme sayısının (N) kaç olacağına karar vermek için Eşitlik 3.67'den yararlanılır. Eşitlik 3.67, Eşitlik 3.66'da verilmiş olan denklemin logaritmasının alınmış halidir.

$$(1 - (1 - q)^S)^N = 1 - p \quad (3.66)$$

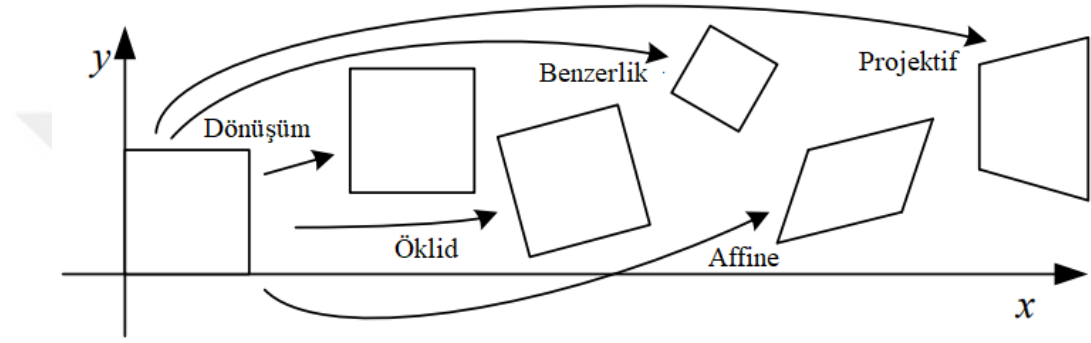
$$N = \frac{\log(1-p)}{\log(1-(1-q)^S)} \quad (3.67)$$

Veride yer alan aykırı veri oranı q 'dur. Eşitlik 3.66'da elde edilen verilerden S tane gürültüsüz veri seçme olasılığı $(1 - q)^S$ olarak ifade edilir. S tane veride en az bir tane gürültülü veri olma olasılığı $1 - (1 - q)^S$ şeklinde gösterilir. Uydurma modelinin N kez tekrarlanması durumu $((1 - (1 - q)^S)^N)$, doğru modelin seçilmeme olasılığına $(1-p)$ eşit olur.

Sonuç olarak örnekleme sayısı kadar uyarlama modelinin testi yapılır ve en çok oy alan uyarlama modeli aykırı noktaların ayıklanmasında kullanılır.

3.6.4. Geometrik dönüşümler

Görüntü mozaiklemede iki farklı görüntü düzlemi arasındaki ilişkiyi hesaplamak için 2B veya 3B geometrik dönüşümler kullanılır. Öteleme, dönme, Öklid, benzerlik dönüşümü, affine dönüşümü ve projektif dönüşüm, önemli bazı geometrik dönüşümlerdendir (Şekil 3.26). Bu çalışmada mozaikleme için projektif dönüşümden yararlanılmıştır. Geometrik dönüşüm parametrelerinin bulunması için eşleşen nokta çiftlerinden faydalanılmıştır.



Şekil 3.26. Temel geometrik dönüşümler (Szeliski, 2022)

3.6.4.1. Perspektif dönüşüm

Genel olarak 3 boyutlu dünya koordinatlarından 2 boyutlu dünya koordinatlarına geçiş için kullanılan bir dönüşüm türüdür. 8 serbestlik açısına sahiptir. Perspektif dönüşüm, en temel haliyle Eşitlik 3.68'deki gibi gösterilebilir.

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (3.68)$$

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

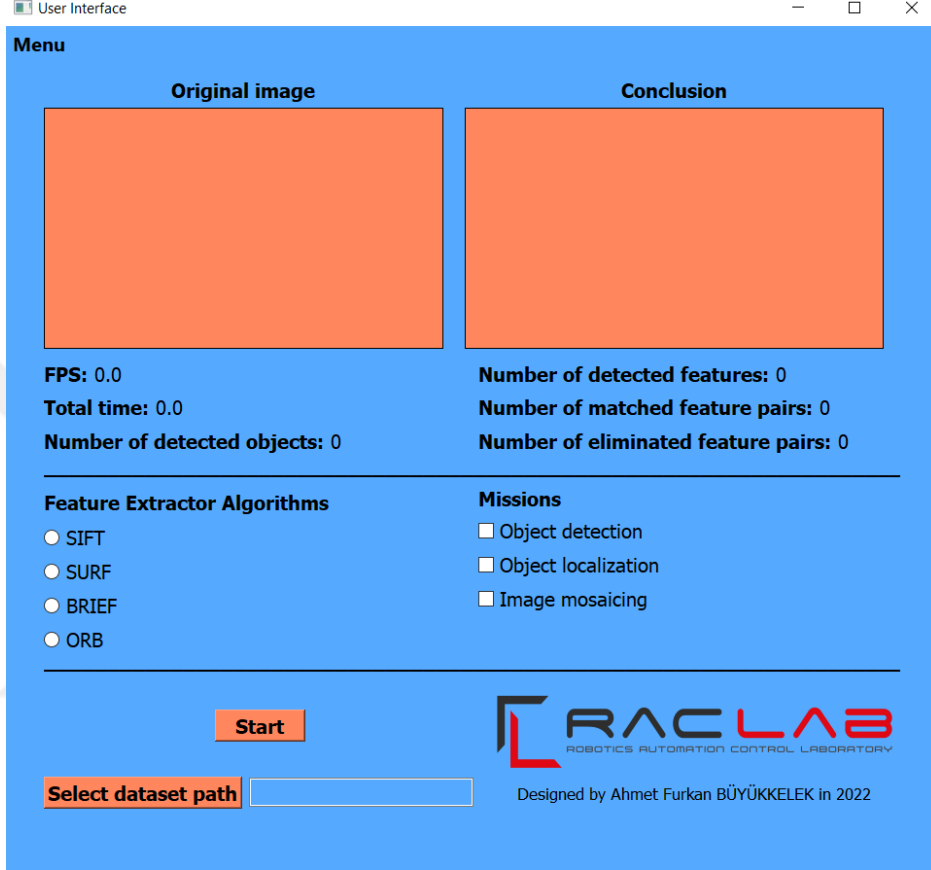
Bu tez çalışması, çalışılan ortam hakkında daha fazla bilgi edinmek için nesne tespitini, bu tespit edilen nesnelerin coğrafi koordinatlarının tespitini ve toplanan tüm bu bilgilerin tek bir görüntü üzerinde birleştirilmesini amaçlayan üç farklı algorithmadan oluşmaktadır. Bu tez kapsamında araba ve sera alanlarının tespiti yapılmıştır. Çalışma ortamına göre ve amaçlanan hedefler doğrultusunda model farklı nesneleri için eğitilip bu nesnelerin tespiti gerçekleştirilebilir.

Önerilen yaklaşımın test edilmesi için DJI Phantom 4 Pro platformuyla 488×430 m²'lik bir alanda, 197 görüntü içeren bir veri kümesi oluşturulmuştur. Veri kümesi Selçuk Üniversitesi yerleşkesinde, 75 m yükseklikte ve %60 örtüşme oranı ile toplanmıştır (Şekil 4.1). Görüntüler; sera alanı, farklı yükseklikteki binalar, arazi ve yol gibi çeşitli yapıları içermektedir. Elde edilen görüntülerin çözünürlüğü 5472×3648 'dir. Ancak işlem süresini kısaltmak için görüntülerin çözünürlüğü 1094×730 'a düşürülmüş ve tüm algoritmalar bu boyutlardaki görüntüleri işlemiştir. Görüntülerin işlenmesi Intel(R) Core (TM) i7-8750H CPU @ 2.20GHz (12 CPUs), ~2.2GHz işlemci özelliklerine sahip bir bilgisayarda gerçekleştirilmiştir.



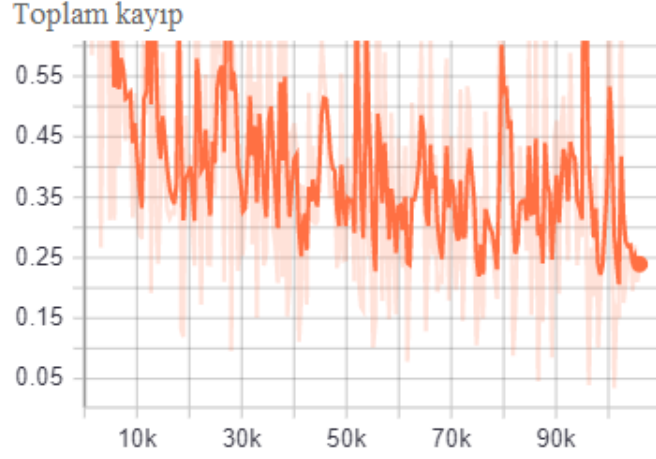
Şekil 4.1. Veri kümesinin elde edildiği ortamın Google Earth görüntüsü

Algoritmaların daha rahat kullanılabilmesi için arayüz (Şekil 4.2) tasarımı yapılmıştır. Tasarlanan arayüz ekranı sayesinde fps değeri, seçilen algoritmaların toplam çalışma süresi, tespit edilen nesnelerin sayısı, tespit edilen özellik sayısı, eşleşen özellik çifti sayısı ve elenen özellik çifti sayısı anlık olarak takip edilebilmektedir.



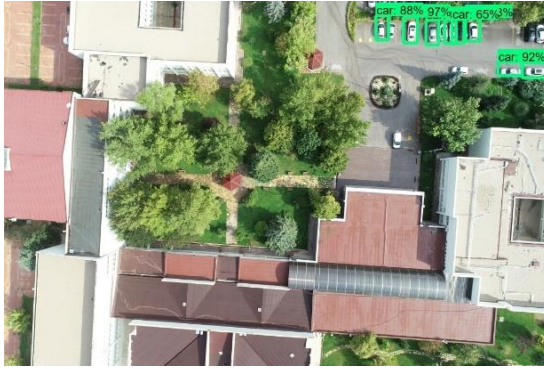
Şekil 4.2. Algoritmaların kullanılması için tasarlanan arayüz ekranı

Önerilen yaklaşımın ilk aşamasında, sera alanlarının ve arabaların tespiti için Faster R-CNN ağından yararlanılmıştır. Modelin eğitilebilmesi için 28704 görüntü kullanılmıştır. Görüntüler; DJI Phantom 4 Pro platformuyla toplanan verilerden, UAVid (Lyu ve ark., 2020), VEDAI (Razakarivony ve Jurie, 2016) ve VisDrone (Zhu ve ark., 2020) veri kümelerinin bir kısmından oluşmaktadır. Bu görüntülerin %80'i eğitim, %20'si test için 2 farklı dosyaya ayrılmıştır. Böylece, eğitim klasöründeki veriler ile modelin eğitimi gerçekleştirilirken, test klasöründeki görüntüler ile eğitilen modelin testi yapılmıştır. Model, 105934 adım eğitilmiştir ve her bir adımın eğitimi yaklaşık 0.26 s sürmüştür.



Şekil 4.3. Eğitilen modelde toplam kayıp değerinin zamanla değişimi

Modelin eğitimi çok daha fazla adımda gerçekleştirilebilirdi. Ancak Şekil 4.3’de görüldüğü gibi 75000’inci adımdan sonra toplam kayıptaki düşüş çok sınırlı olmuştur. Bu nedenle eğitim 105934’üncü adımda sonlandırılmıştır. Bu adıma gelene kadar yaklaşık her 10000 adımda bir model test edilmiştir. Ancak verim açısından gözle görülür bir değişiklik gerçekleşmemesine rağmen tespit edilen nesnelerin ortalama kesinlik değerlerinde (mAP) bir miktar yükselme olmuştur.



(a)



(b)

Şekil 4.4. Faster R-CNN ile tespit edilen (a) arabalar ve (b) sera alanları

Modelin eğitimi gerçekleştirildikten sonra testi, etiketlenmemiş olan görüntülerde gerçekleştirilmiştir. Şekil 4.4’de test edilen görüntülerden ikisi verilmiştir. Bu görseller, modelin arabaları algılamada oldukça tatmin edici olduğunu ve sera alanlarının algılanması için ise ezbere düştüğünü açık bir şekilde göstermiştir.

Model sonuçlarının daha da iyileştirilmesi için daha çok veri kullanılmalı ve kullanılan veriler veri çoğaltma yöntemleri ile çeşitlendirilmelidir. Ayrıca, araba tespiti için kullanılan veri miktarı ile sera alanı tespiti için kullanılan veri miktarı arasındaki veri dağılımı dengesizliği giderilmelidir.

Diğer adımlara girdi olarak verilecek modelin değerlendirilmesi için Şekil 4.5’de verilmiş olan karmaşıklık matrisinden yararlanılarak doğruluk (Eşitlik 4.1), kesinlik (Eşitlik 4.2), hassasiyet (Eşitlik 4.3) ve F1 skoru (Eşitlik 4.4) hesaplanmıştır.

		Gerçek Değerler	
		Pozitif (1)	Negatif (0)
Tahmin Edilen Değerler	Pozitif (1)	DP	YP
	Negatif (0)	YN	DN

Şekil 4.5. Karmaşıklık matrisinin temsili

$$\text{Doğruluk} = \frac{DP+DN}{DP+YP+DN+YN} \quad (4.1)$$

$$\text{Kesinlik} = \frac{DP}{DP+YP} \quad (4.2)$$

$$\text{Hassasiyet} = \frac{DP}{DP+YN} \quad (4.3)$$

$$\text{F1 puanı} = \frac{2}{\frac{1}{\text{Kesinlik}} + \frac{1}{\text{Hassasiyet}}} \quad (4.4)$$

Karmaşıklık matrisindeki doğru pozitif (DP), nesnelerin gerçekteki değerinin araba veya sera alanı olduğu, tahmin edilen değerinde birbirleri ile karıştırılmadan doğru bir şekilde araba veya sera alanı olduğu örnekleri gösterir. Yanlış Pozitif (YP), gerçekte nesnenin araba veya sera ortamı olmamasına rağmen yanlış çıktıyı göstermesi durumudur. Yanlış Negatif (YN), nesnenin gerçekte araba veya sera alanı olması ve modelinde araba veya sera alanı olmadığını ifade ettiği durumudur. Doğru Negatif (DN), nesnenin gerçekte araba veya bir sera alanı olmaması ve modelinde aynı düşüncede olması durumudur.

101 örnek ile hesaplanan karmaşıklık matrisinde; DP, DN, YP ve YN sırasıyla 74, 6, 1 ve 20 olarak bulunmuştur. Bu değerler referans alındığında doğruluk %79.2, kesinlik %98.66, hassaslık %78.72 ve F1 puanı %87.57 olarak hesaplanmıştır. Bu sonuçlar, Şekil 4.4 referans alınarak model üzerinde yapılan değerlendirmelerin, modelin doğruluğunun hesaplanması için kullanılan eşitliklerin sonuçlarına da yansıdığını göstermektedir. Ayrıca, eğitilen model test edilirken eğer görüntü boyutlarında bir değişim yapılmıyorsa ortalama 7-8 fps değerinde çalışmış ve maksimum 9.16 fps değerini görmüştür.

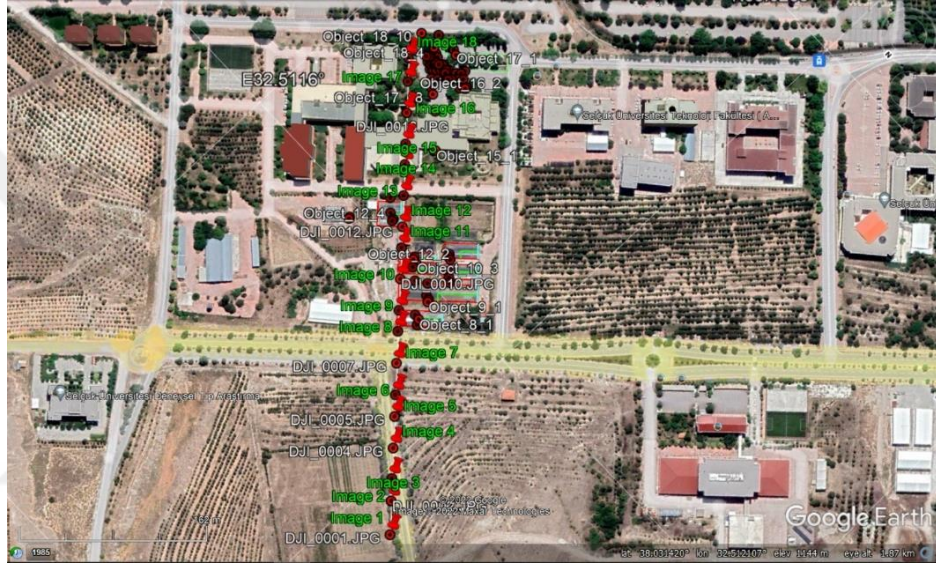
Önerilen algoritmanın ikinci aşamasında, Faster R-CNN modeli ile araba ve sera alanları tespit edildikten sonra bu nesnelerin kamera düzlemindeki coğrafi konumlarının belirlenmesi için coğrafi dönüşüm algoritmaları uygulanmıştır. Ancak, her bir görüntüdeki nesnelerin daha doğru konumlandırılabilmesi için IMU bilgisine ihtiyaç duyulmaktadır. Bu nedenle, gerekli olan yaw açısı Google Earth uygulaması aracılığıyla belirlenmiştir. Böylece, yön bilgisinin sebep olabileceği ciddi hata payları bu işlem ile önlenmiştir.

GPS yardımıyla alınan WGS84 koordinatları ilk olarak ECEF koordinatına daha sonrada ENU koordinat sistemine dönüştürülmüştür. Bu aşamada pikseller arasındaki farktan yararlanarak yereldeki değişim hesaplanmıştır. Son olarak, elde edilen yeni değerler ile önce ECEF daha sonra WGS84 koordinat sistemine dönüştürülerek nesnelerin GPS konumları hesap edilmiştir (Şekil 4.6).

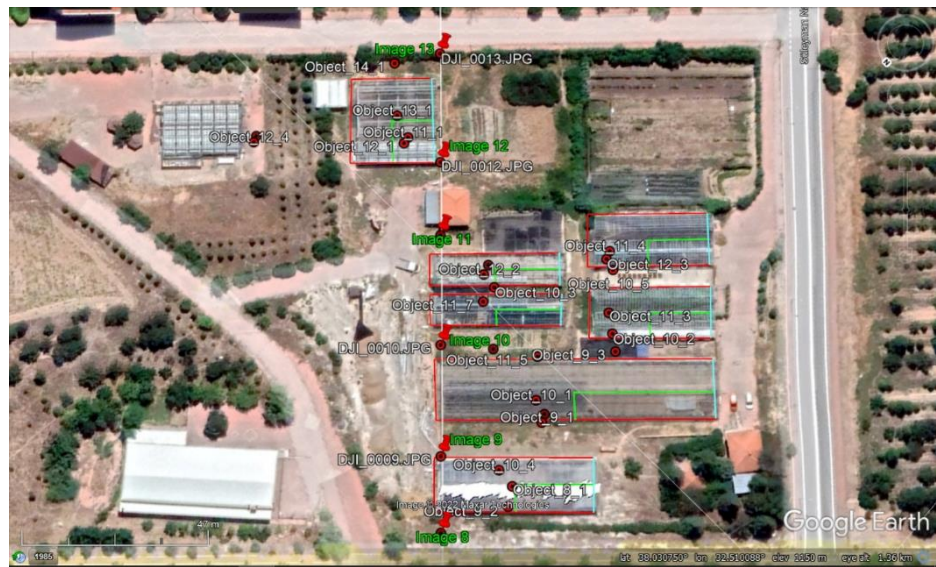


Şekil 4.6. Coğrafi koordinatı tespit edilen bazı nesneler

Konumlandırma aşamasında, Faster R-CNN modeli tarafından tespit edilebilen tüm nesnelerin coğrafi koordinatı hesaplanmıştır (Şekil 4.7). Nesnelerin tespit edilen GPS koordinatlarının değerlendirilebilmesi için 20 farklı coğrafi koordinat kullanılmıştır. Hata payının olabildiğince daha doğru hesaplanması için özellikle sabit yapılar referans noktası olarak kullanılmış (Şekil 4.8) ve Google Earth aracılığıyla görselleştirilmiştir. Konumları tespit edilen nesnelerin maksimum hatası, ortalama hatası ve minimum hatası sırasıyla 15,5 m, 7,35 m ve 0,5 m olarak hesaplanmıştır. Gerçek GPS koordinatları ile hesaplanan GPS koordinatları arasındaki hata payını ölçmek için Google Earth kullanılmıştır.

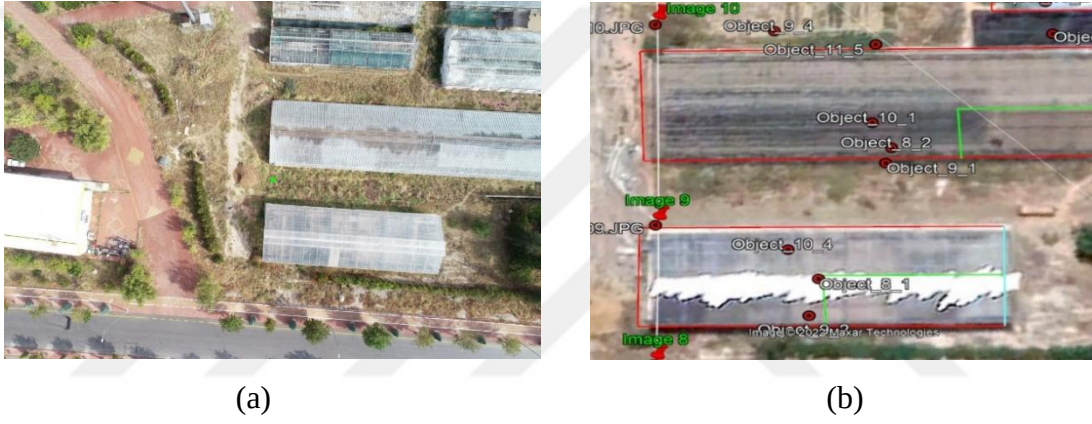


Şekil 4.7. Görüntü merkezinin koordinatları ve coğrafi konumu hesap edilen nesneler



Şekil 4.8. Nesnelerin belirlenen coğrafi koordinatları ve konumlandırma hatalarını ölçmek için kullanılan bazı referans noktalar

Konumlandırma algoritmasındaki hatalarının değişken olmasının birkaç nedeni bulunmaktadır. İlk olarak yüksekliğe dayalı olarak hesaplanan piksel başına metre değeridir. Hatanın sebeplerinden bir diğeri GPS'in veriminden kaynaklanmaktadır. Ayrıca, görüntünün çekilme zamanı ile GPS bilgisinin alınma zamanı arasındaki farkta konumlandırma algoritmasının verimini etkilemektedir. Şekil 4.9 (a)'nın merkezinde bulunan nokta görüntünün merkezini göstermektedir. Normalde bu nokta ile çekilen GPS bilgisi aynı konumda olmalıdır. Ancak Şekil 4.9 (b)'de, görüntü GPS koordinatının ("Image 9") görüntü merkez noktasından gerçekte bir miktar daha altta olduğu görülmektedir. Bu durumda konumlandırma algoritmasının verimine etki etmektedir. Son olarak, IMU sensörü verilerinin olmaması nedeniyle kullanılamamasıdır.



Şekil 4.9. (a) çekilen görüntünün merkezi, (b) görüntü merkezinin denk geldiği GPS koordinatı

Önerilen algoritmanın son bölümünde, sıralı 18 görüntünün mozaikleme işlemi gerçekleştirilmiş ve böylece ilk aşamadaki bilgilerin tek bir görüntü üzerinde birleştirilmesi sağlanmıştır. Mozaikleme algoritması ilk olarak özellik çıkarma ile başlamış ve bunun için SIFT, SURF, ORB ve BRIEF algoritmalarından yararlanılmıştır. Ayrıca bu özellik çıkartma algoritmaları, çıkarılan özellik sayısı ve özellik çıkarma süresi açısından Çizelge 4.1'de kıyaslanmıştır.

Çizelge 4.1. Homografi hesabı öncesi kullanılacak özellik çiftleri sayısındaki değişim

Özellik Çıkarma Algoritmaları	Çıkarılan Toplam Özellik Sayısı	Eşleştirilen Toplam Özellik Çifti	Elenen Toplam Özellik Çifti	Kullanılan Toplam Özellik Çifti
SIFT	576852	439904	422105	17799
SURF	532860	96603	68959	27644
BRIEF	72570	12758	9381	3377
ORB	137198	31172	21552	9620

Çizelge 4.1'e daha yakında bakıldığında en çok özellik, SIFT algoritması tarafından çıkarılmış ve çıkardığı özelliklerin %76.26 BF algoritması tarafından eşleştirilmiştir. SURF, BRIEF ve ORB özellik çıkarma algoritmalarının çıkardığı özelliklerin sırasıyla %18.12, %17.58 ve %22.72'si eşleştirilebilmiştir. Yanlış özellik eşleşmelerinin elenmesi için SIFT algoritmasına mesafe sınırlaması eklenmiş ve RANSAC algoritması uygulanmıştır. Diğer özellik çıkarma algoritmalarında özellikleri elemek için sadece RANSAC'dan yararlanılmıştır. Bu işlemler sonrası homografi hesabı için kullanılabilir özellik çiftleri sayısı düşüş meydana gelmiş ve en çok azalma, SIFT algoritmasının eşleştirdiği özellik çiftlerinde meydana gelmiştir. Buna rağmen homografi hesabı için kullanılan ikinci en çok özellik çifti sayısı SIFT tarafından sağlanmıştır.

Çizelge 4.2. Özellik çıkarma algoritmalarının süre bakımından kıyaslanması

Özellik Çıkarma Algoritmaları	İlk Görüntüdeki Özellik Çıkarma Süresi (s)	Son Görüntüdeki Özellik Çıkarma Süresi (s)	Toplam Özellik Çıkarma Süresi (s)
SIFT	1.1	2.4	40.5
SURF	0.51	1.08	21.1
BRIEF	0.07	0.3	3.98
ORB	0.1	0.19	3.35

Özellik algoritmaları süre açısından kıyaslandığında ise özellik çıkarmak için en çok zaman alan algoritmanın SIFT olduğu Çizelge 4.2'de açık bir şekilde görülmüştür. BRIEF ve ORB algoritmalarının çalışma sürelerine bakıldığında SIFT ve SURF algoritmalarına göre çok yavaş kalmaktadır. Bu çizelgede dikkat edilmesi gereken bir diğer nokta ise ilk görüntünün özellik çıkarma süresi ile son görüntünün özellik çıkarma süresi süresidir çünkü mozaiklenen görüntü büyüdükçe özellik çıkarma algoritmalarının süre maliyetinin arttığını net bir şekilde göstermiştir. Bu nedenle, sürekli sabit büyüklükteki bir alanda özellik çıkarmak süre açısından daha fazla kazanç sağlayabilir. Özellikle büyük veri kümelerinde daha faydalı olabileceği düşünülmektedir. Ayrıca, daha fazla özelliğin çıkarılması, özellik eşleştirme algoritmasının da çalışma süresini arttıracaktır.

Homografi hesabı öncesi, yanlış eşleştirilen özelliklerin elenmesi için RANSAC algoritması ve mesafe sınırlaması kullanılmıştır. RANSAC algoritmasının etkisi Şekil 4.10 ve Şekil 4.11'de daha iyi gözlemlenebilir. Şekil 4.10, mozaikleme işlemi sürecinde görüntüde tespit edilmiş olan nokta çiftlerinin bir kısmını göstermektedir. Şekil 4.11,

RANSAC algoritması sonrası görüntü üzerindeki yanlış eşleştirmeleri etkili bir şekilde ortadan kaldırdığını göstermektedir.



Şekil 4.10. Mozaikleme sürecinde görüntü üzerinde eşleştirilmiş bazı özellik çiftleri



Şekil 4.11. RANSAC sonrası görüntü üzerinde eşleştirilmiş doğru özellik çiftleri

Önerilen yaklaşımda, görüntü mozaikleme algoritmasının son aşamasında ise görüntülerin birleştirilmesi gerçekleştirilmiştir. Mozaikleme algoritmalarını kıyaslamak için ilk olarak diğer yazılımların sonuçları ele alınmıştır. Çizelge 4.3’de görüldüğü gibi mozaikleme süresi açısından OpenDroneMap algoritması, Pix4Dmapper’a göre daha

üstündür. Aynı sayıdaki görüntünün mozaikleme için aralarında yaklaşık 96 dakika bulunmaktadır. Yer kontrol noktalarının doğruluğu bakımında ise OpenDroneMap çok küçük bir farkla daha doğru sonuçlar vermiştir. Pix4Dmapper ve OpenDroneMap algoritmalarını sonuçları sırasıyla Şekil 4.12 ve Şekil 4.13’de verilmiştir. Ancak bu uygulamaların gerçek zamanlı olarak uygulanması zor gözükmetedir.

Çizelge 4.3. Mozaikleme algoritmalarının doğruluk ve zaman açısından karşılaştırılması

Mozaikleme Yöntemleri	X Eksenindeki Ortalama Hata (m)	Y Eksenindeki Ortalama Hata (m)	Mozaikleme Süresi (dk)	Görüntü Sayısı
Pix4Dmapper	-0.0221	0.0068	123	197
OpenDroneMap	0.017	-0.014	27	197



Şekil 4.12. Pix4Dmapper’ın ortomozaik sonucu

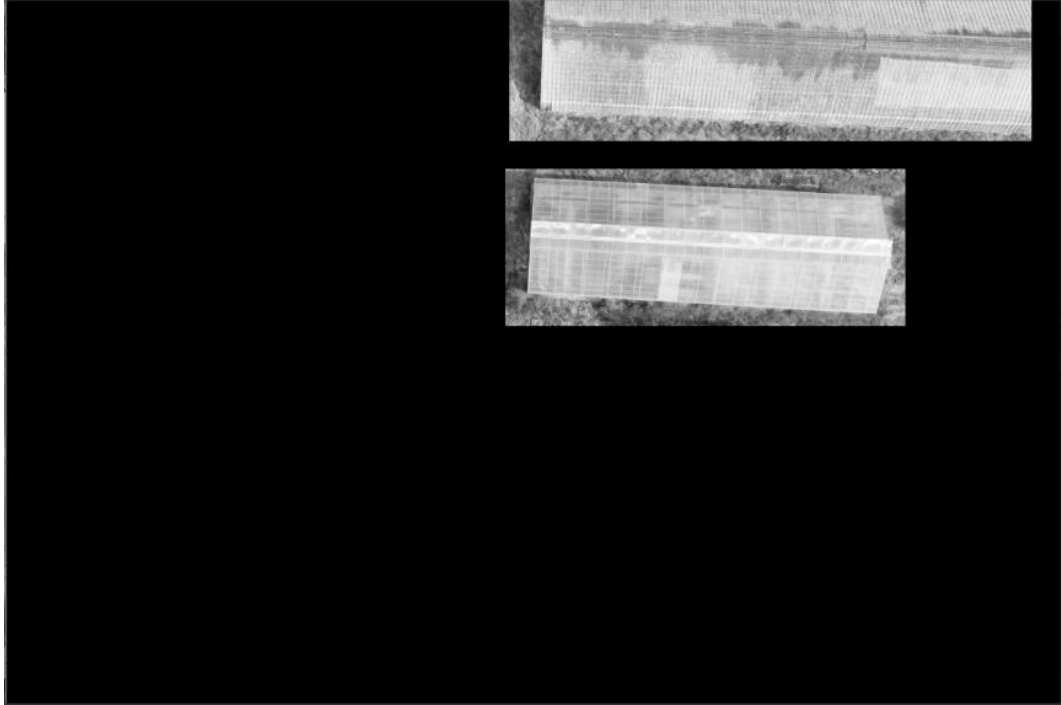


Şekil 4.13. OpenDroneMap ortomozaik sonucu

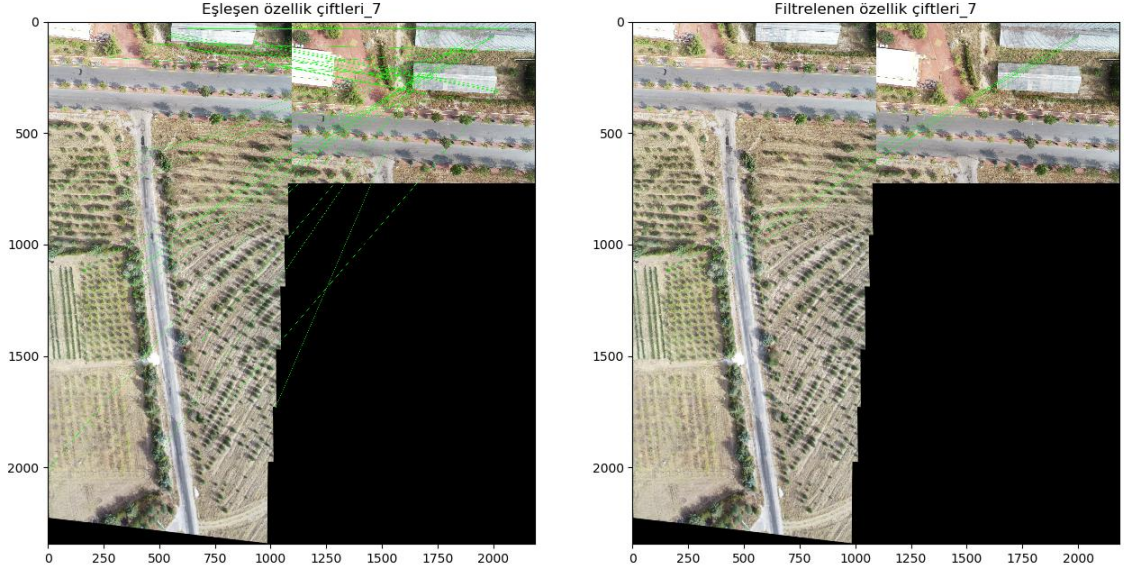
Tezde önerilen yaklaşımda; SIFT, SURF, BRIEF ve ORB algoritmalarıyla sadece görüntü mozaikleme süreleri sırasıyla 70 s, 54 s, 21.3 s ve 18.26 s'dir. Her bir özellik çıkarma yöntemi önerilen yaklaşımda ayrı ayrı test edildiğinde algoritmaların çalışma süreleri sırasıyla 85 s, 67 s, 31.1 s ve 28,2 s sürmektedir. Sonuç olarak önerilen yaklaşım SIFT özellik çıkarma algoritması ile birlikte çok yavaş çalışmaktadır. Ancak, en iyi mozaik görüntüleri SIFT algoritmasının kullanıldığı (Şekil 4.14 (a)) mozaikleme algoritmasında elde edilmiştir.

Önerilen görüntü mozaikleme algoritmasının doğruluğu alınan GPS koordinatları arasındaki fark ve piksel koordinatları arasındaki farktan yararlanılarak ölçülmüş ve hata payı ortalama 1,11 m olarak hesaplanmıştır. Önerilen bu algoritmada, SIFT özellik eşleştirme yönteminden faydalanılmış ve diğer özellik çıkarma algoritmaları için tek tek hata hesabı yapılmamıştır çünkü diğer özellik çıkarma yöntemleri ile elde edilen mozaiklerdeki hata payı daha yüksekti. Yani, bu algoritmalar için RANSAC, yanlış eşleşen özellik çiftlerini elemeye yetersiz kalmıştır. Bu nedenle, SIFT algoritmasından yararlanılarak daha az hata ile elde edilen mozaikleme sonuçları sunulmuştur (Şekil 4.14 (a)). Şekil 4.14 (b)'de ise nesne tespiti, nesnenin GPS koordinatlarının tespiti ve mozaikleme sonuçları birleştirilmiştir. Tüm algoritmaların çalışması 86 s, nesne tespit algoritmasının çalışması 16 s almıştır.

Mozaikleme algoritmasının daha hızlı bir şekilde çalışabilmesi için Faster R-CNN modelinin tespit ettiği nesnelerin bölgelerinden (Şekil 4.15) yararlanılarak mozaikleme yapmak da düşünülmüştür. Ancak tespit edilen özellik çifti sayısının az olması ve yanlış eşleşmelerin elenememesi (Şekil 4.16) nedeniyle sağlıklı bir mozaik elde edilememiştir.



Şekil 4.15. Tespit edilen nesnelerin piksel koordinatlarından yararlanılarak görüntü üzerinde işlem yapılacak alanların azaltılması



Şekil 4.16. Daha hızlı mozaiklemek için kullanılan yöntemdeki özellik çiftleri sorunu

Mozaikleme yöntemdeki verimin artırılması için kamera dış yönlendirme parametreleri kullanılmalı ve kamera iç yönlendirme parametreleri hesaplanmalıdır. Bu durum aynı zamanda mozaikleme hızını da arttıracaktır. Ek olarak özellik çıkarmak için tüm görüntünün kullanılması hem hesaplama süresini hem de algoritmanın verimini olumsuz yönde etkilemektedir. Bu nedenle özellik çıkarılacak alanın sınırlı tutulması daha faydalı olabilir.

5. SONUÇLAR VE ÖNERİLER

5.1 Sonuçlar

Bu tez çalışması kapsamında İHA'lerden alınan görüntüler üzerinde ortam hakkında ön bilgilere sahip olma ve alınan bu sonuçlara göre çeşit alanlarda farklı reaksiyonların uygulanabilmesi için bir çalışma yürütülmüştür. Yapılan bu tez çalışmasında belirtilen amacı uygulamak için üç farklı algoritma içeren bir yaklaşımın sunumu gerçekleştirilmiştir. Bu yaklaşım derin öğrenme ile nesne tespiti, tespit edilen nesnelerin coğrafi koordinatlarının tespiti ve toplanan ortam görüntüleri ile birlikte görüntü mozaikleme algoritmasıyla elde edilen tüm bilgilerin birleştirilmesini içermektedir.

Önerilen yaklaşımın ilk aşamasında, İHA'dan alınan görüntüler üzerinde sera alanlarının ve arabaların tespiti gerçekleştirilmiştir. Belirtilen nesnelerin tespiti için Faster R-CNN modelinden yararlanılmış ve yeni model eğitimi için 28704 farklı görüntü kullanılmıştır. Modelin hata değerlendirmesi için karmaşıklık matrisinden yararlanılmış ve model, etiketlenmemiş olan görüntülerde test edilmiştir. Karmaşıklık matrisinin hesaplama bilmesi için 101 farklı örnek alınmıştır. Karmaşıklık matrisinden yararlanılarak modelin doğruluğu %79.2, kesinlik %98.66, hassaslık %78.72 ve F1 puanı %87.57 olarak hesaplanmıştır. Eğitilen model saniyede ortalama 7-8 görüntüyü işleyebilmektedir. Bu nedenle gerçek zamanlı çalışmalar için uygundur.

Yaklaşımın ikinci aşamasında ise Faster R-CNN modeli tarafından çıktı olarak alınan görüntüler, coğrafi konumlandırma algoritmasının girdisi olarak kullanılmıştır. Kameradan alınan görüntülerin merkez noktasının coğrafi koordinatları zaten GPS yardımıyla bilinmektedir. Kameranın merkez coğrafi koordinatlarının diğer piksellerde hangi koordinatlara denk geldiğini hesaplamak için WGS84 koordinatları ilk olarak ECEF ve daha sonra ENU koordinat sistemine dönüştürülmüştür. Bu aşamada yereldeki mesafe değişimleri piksel başına düşen metre hesabı ile yapılmıştır. Değişim hesaplandıktan sonra bu değişimler yerel koordinatlara ilave edilmiştir. Son olarak ENU koordinatları, önce ECEF ve daha sonra WGS84 koordinatlarına dönüştürülerek her bir nesneye karşılık gelen coğrafi koordinatlar hesaplanmıştır. Konumları tespit edilen

nesnelerin maksimum hatası, ortalama hatası ve minimum hatası sırasıyla 15,5 m, 7,35 m ve 0,5 m olarak hesaplanmıştır.

Önerilen yaklaşımın son aşamasında ise hem ortamın büyük bir haritası çıkarılmış hem de bu harita üzerinde ilk iki aşamada elde edilmiş olan sonuçlar birleştirilmiştir. Mozaikleme yaklaşımında bulunan özellik çıkarma bölümünde ise SIFT, SURF, ORB ve BRIEF özellik çıkarma algoritmaları süre ve çıkarılan özellikler bakımından kıyaslanmış ve SIFT algoritmasının özellik çıkarma performansı açısından daha üstün olduğu tespit edilmiştir. Ayrıca, en az hatalı mozaik SIFT yöntemi ile çıkarılmıştır. Önerilen mozaikleme yöntemi, Pix4Dmapper ve OpenDroneMap isimli yazılımlar ile zaman ve performans açısından kıyaslanmıştır. Önerilen yaklaşımda 18 sıralı görüntü ortalama 1,11 m hata ile 70 s'de mozaiklenmiştir. Önerilen yaklaşımın tamamının çalışması 85 s almıştır. Pix4Dmapper ve OpenDroneMap algoritmaları 197 görüntünün mozaiklemesini sırasıyla 123 dk ve 27 dk süreyle gerçekleştirmiştir. Pix4Dmapper ve OpenDroneMap hata değerleri açısından birbirlerine yakındır. Ancak OpenDroneMap algoritması çalışma hızı göz önüne alındığında çok avantajlıdır.

5.2 Öneriler

Bu tez çalışması kapsamında önerilen yaklaşım 3 farklı algorithmadan oluşmaktadır. Bu nedenle her bir algoritmanın geliştirilmesi için öneriler ayrı ayrı ele alınmıştır.

Derin öğrenme ile nesne tespiti konusunda mutlaka geniş veri seti oluşturulmalıdır. Ayrıca, elde edilen veri kümesi; görüntü döndürme, çözünürlüğü değiştirme, karışıklığı artırma ve gürültü ekleme gibi adımlar uygulanarak genişletilmesi modelin daha iyi sonuçlar vermesine neden olabilir. Ayrıca, nesne tespitinin yüksek hızda yapılmasını gerektiren uygulamalarda YOLO ve SSD modellerinin kullanılması tercih edilebilir.

Nesnelerin coğrafi olarak daha iyi konumlandırılması için en basit yollardan biri daha doğru sonuçlar veren GPS ve IMU sensörlerinin kullanılması olacaktır. Diğer bir yöntem ise kameranın iç yönlendirme parametrelerinden kaynaklı olan bozulmalarının

giderilmesidir. Bunun için ya kamera kalibrasyonu yapılmalı ya da bundle adjustment algoritmasının kullanılması tavsiye edilmektedir.

Diğer bir algoritmada ise düşük özelliğe sahip ortamlar görüntü mozaikleme algoritmasının verimini önemli ölçüde etkilemektedir. Bu etkileri azaltmak için GPS ve IMU bilgisinden ve bundle adjustment algoritmasından faydalanılmalıdır. Ayrıca kullanılan sensörlerin doğruluk değerlerinin yüksek olması mozaikleme algoritmasının da verimini arttıracaktır. Ek olarak, sürekli büyüyen bir alanda özellik çıkarma ve eşleştirme zaman alıcı bir işlem olmakla birlikte yanlış eşleşmelerin artmasına da sebep olabilmektedir. Bu nedenle, çalışma yapılan alanın sınırlı ama çok küçük olmaması sonuçları iyileştirebilir. Son olarak, özellik çıkarma ve RANSAC algoritmalarının parametreleri mozaikleme verimini doğrudan etkilemektedir. Parametre değerlerinde değişiklik yaparak mozaikleme algoritmasının verimini yükseltmek mümkün olabilir.

KAYNAKLAR

- Aghaei, M., Leva, S. ve Grimaccia, F., "PV power plant inspection by image mosaicing techniques for IR real-time images," 2016 IEEE 43rd Photovoltaic Specialists Conference (PVSC), 2016, pp. 3100-3105, doi: 10.1109/PVSC.2016.7750236.
- Ali, M. E. (2020). Global Positioning System (GPS): Definition, Principles, Errors, Applications & DGPS. *no. April*.
- Allig, C. ve Wanielik, G., "Alignment of Perception Information for Cooperative Perception," 2019 IEEE Intelligent Vehicles Symposium (IV), 2019, pp. 1849-1854, doi: 10.1109/IVS.2019.8814108.
- Avola, D., Cinque, L., Foresti, G. L., Martinel, N., Pannone, D. ve Piciarelli, C., "A UAV Video Dataset for Mosaicking and Change Detection From Low-Altitude Flights," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 50, no. 6, pp. 2139-2149, June 2020, doi: 10.1109/TSMC.2018.2804766.
- Awange, J., Kiema, J. (2019). The Global Positioning System. In: Environmental Geoinformatics. Environmental Science and Engineering(). Springer, Cham. https://doi.org/10.1007/978-3-030-03017-9_5
- Bay, H., Tuytelaars, T., Van Gool, L. (2006). SURF: Speeded Up Robust Features. In: Leonardis, A., Bischof, H., Pinz, A. (eds) Computer Vision – ECCV 2006. ECCV 2006. Lecture Notes in Computer Science, vol 3951. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11744023_32
- Bhattarai, N.; Nakamura, T.; Mozumder, C. Real Time Human Detection and Localization Using Consumer Grade Camera and Commercial UAV. Preprints 2018, 2018110156 (doi: 10.20944/preprints201811.0156.v1).
- Brown, M., Lowe, D.G. *Automatic Panoramic Image Stitching using Invariant Features*. International Journal of Computer Vision 74, 59-73 (2007). <https://doi.org/10.1007/s11263-006-0002-3>
- Bu, S., Zhao, Y., Wan, G. ve Liu, Z., "Map2DFusion: Real-time incremental UAV image mosaicing based on monocular SLAM," 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, pp. 4564-4571, doi: 10.1109/IROS.2016.7759672.
- Buyukyazi, T., Bayraktar, S. ve Lazoglu, I., "Real-time image stabilization and mosaicking by using ground station CPU in UAV surveillance," 2013 6th International Conference on Recent Advances in Space Technologies (RAST), 2013, pp. 121-126, doi: 10.1109/RAST.2013.6581183.
- Cai, Y., Ding, Y., Zhang, H., Xiu, J., & Liu, Z. (2020). Geo-Location Algorithm for Building Targets in Oblique Remote Sensing Images Based on Deep Learning and Height Estimation. *Remote Sensing*, 12(15), 2427. <https://doi.org/10.3390/rs12152427>

- Calonder, M., Lepetit, V., Strecha, C., Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds) Computer Vision – ECCV 2010. ECCV 2010. Lecture Notes in Computer Science, vol 6314. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15561-1_56
- Chauhan, N. K. ve Singh, K., "A Review on Conventional Machine Learning vs Deep Learning," 2018 International Conference on Computing, Power and Communication Technologies (GUCON), 2018, pp. 347-352, doi: 10.1109/GUCON.2018.8675097.
- Cheng, G., Xie, X., Han, J., Guo, L. ve Xia, G. -S., "Remote Sensing Image Scene Classification Meets Deep Learning: Challenges, Methods, Benchmarks, and Opportunities," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 3735-3756, 2020, doi: 10.1109/JSTARS.2020.3005403.
- Chiang, K.-W., Tsai, M.-L., & Chu, C.-H. (2012). The Development of an UAV Borne Direct Georeferenced Photogrammetric Platform for Ground Control Point Free Applications. *Sensors*, 12(7), 9161–9180. <https://doi.org/10.3390/s120709161>
- Eggert, C., Brehm, S., Winschel, A., Zecha, D. ve Lienhart, R., "A closer look: Small object detection in faster R-CNN," 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017, pp. 421-426, doi: 10.1109/ICME.2017.8019550.
- Eugster, H., & Nebiker, S. (n.d.). *GEO-REGISTRATION OF VIDEO SEQUENCES CAPTURED FROM MINI UAVs-APPROACHES AND ACCURACY ASSESSMENT*.
- Ge, Y., Wen, G. ve Yang, X., "A Fast Mosaicking Method for Small UAV Image Sequence Using a Small Number of Ground Control Points," 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2016, pp. 90-94, doi: 10.1109/IHMSC.2016.110.
- Ghazal, M. A. ve ark., "Vegetation Cover Estimation Using Convolutional Neural Networks," in IEEE Access, vol. 7, pp. 132563-132576, 2019, doi: 10.1109/ACCESS.2019.2941441.
- Ghosh, D., & Kaabouch, N. (2016). A survey on image mosaicing techniques. *Journal of Visual Communication and Image Representation*, 34, 1–11. <https://doi.org/10.1016/j.jvcir.2015.10.014>
- Guo, F. ve Wang, Y. Panoramic Image Mosaics via Distributed Systems Using Color Moments and Local Wavelet-Features. *Pattern Recognit. Image Anal.* **29**, 384–388 (2019). <https://doi.org/10.1134/S1054661819030088>
- Huang, Y., Li, J. ve Fan, N., "Image Mosaicing for UAV Application," 2008 International Symposium on Knowledge Acquisition and Modeling, 2008, pp. 663-667, doi: 10.1109/KAM.2008.73.

- Imad, M., Doukhi, O., & Lee, D.-J. (2021). Transfer Learning Based Semantic Segmentation for 3D Object Detection from Point Cloud. *Sensors*, 21(12), 3964. <https://doi.org/10.3390/s21123964>
- Jiao, L. ve ark., "A Survey of Deep Learning-Based Object Detection," in *IEEE Access*, vol. 7, pp. 128837-128868, 2019, doi: 10.1109/ACCESS.2019.2939201.
- Kalu, I., Ndehedehe, C. E., Okwuashi, O., & Eyoh, A. E. (2022). A comparison of existing transformation models to improve coordinate conversion between geodetic reference frames in Nigeria. *Modeling Earth Systems and Environment*, 8(1), 611–624 (2022). <https://doi.org/10.1007/s40808-021-01090-y>
- Kekec, T., Yildirim, A., & Unel, M. (2014). A new approach to real-time mosaicing of aerial images. *Robotics and Autonomous Systems*, 62(12), 1755–1767. <https://doi.org/10.1016/j.robot.2014.07.010>
- König, J., Malberg, S., Martens, M., Niehaus, S., Krohn-Grimberghe, A., Ramaswamy, A. (2020). Multi-stage Reinforcement Learning for Object Detection. In: Arai, K., Kapoor, S. (eds) *Advances in Computer Vision. CVC 2019. Advances in Intelligent Systems and Computing*, vol 943. Springer, Cham. https://doi.org/10.1007/978-3-030-17795-9_13
- Li, Z. ve Isler, V., "Large Scale Image Mosaic Construction for Agricultural Applications," in *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 295-302, Jan. 2016, doi: 10.1109/LRA.2016.2519946.
- Liu, J., Gong, J., Guo, B. ve Zhang, W., "A Novel Adjustment Model for Mosaicking Low-Overlap Sweeping Images," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 4089-4097, July 2017, doi: 10.1109/TGRS.2017.2688385.
- Liu, Q., Liu, W., Zou, L., Wang, J., and Liu, Y.: A NEW APPROACH TO FAST MOSAIC UAV IMAGES, *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XXXVIII-1/C22, 271–276, <https://doi.org/10.5194/isprsarchives-XXXVIII-1-C22-271-2011>, 2011.
- Liu, Y., Sun, P., Wergeles, N., & Shang, Y. (2021). A survey and performance evaluation of deep learning methods for small object detection. *Expert Systems with Applications*, 172. <https://doi.org/10.1016/j.eswa.2021.114602>
- Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* **60**, 91–110 (2004). <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- Lyu, Y., Vosselman, G., Xia, G. S., Yilmaz, A., & Yang, M. Y. (2020). UAVid: A semantic segmentation dataset for UAV imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 165, 108-119.

- Ma, H., Ma, H., Liu, K., Luo, W., & Zhang, L. (2020). Direct Georeferencing for the Images in an Airborne LiDAR System by Automatic Boresight Misalignments Calibration. *Sensors*, 20(18), 5056. <https://doi.org/10.3390/s20185056>
- Örnek, A. H. (2019). *Yeni doğan bebeklerin termal görüntülerinin derin öğrenme yöntemleri ile sınıflandırılması* (Master's thesis, Konya Teknik Üniversitesi).
- Pai, P., Naidu, V.P.S. Target geo-localization based on camera vision simulation of UAV. *J Opt* **46**, 425–435 (2017). <https://doi.org/10.1007/s12596-017-0395-0>
- Park, S., Ghosh, D., Kaabouch, N., Fevig, R. A. ve Semke, W. "Hierarchical multi-level image mosaicing for autonomous navigation of UAV", Proc. SPIE 8301, Intelligent Robots and Computer Vision XXIX: Algorithms and Techniques, 830116 (23 January 2012); <https://doi.org/10.1117/12.910334>
- Raeva, P. L., Šedina, J., & Dlesk, A. (2019). Monitoring of crop fields using multispectral and thermal imagery from UAV. *European Journal of Remote Sensing*, 52(sup1), 192–201. <https://doi.org/10.1080/22797254.2018.1527661>
- Razakarivony, S., & Jurie, F. (2016). Vehicle detection in aerial imagery: A small target detection benchmark. *Journal of Visual Communication and Image Representation*, 34, 187-203.
- Ren, X., Sun, M., Zhang, X., & Liu, L. (2017). A Simplified Method for UAV Multispectral Images Mosaicking. *Remote Sensing*, 9(9), 962. <https://doi.org/10.3390/rs9090962>
- Rojas, J., Martinez, C., Mondragon, I., Colorado, J. (2017). Towards Image Mosaicking with Aerial Images for Monitoring Rice Crops. In: Chang, I., Baca, J., Moreno, H., Carrera, I., Cardona, M. (eds) *Advances in Automation and Robotics Research in Latin America. Lecture Notes in Networks and Systems*, vol 13. Springer, Cham. https://doi.org/10.1007/978-3-319-54377-2_24
- Rublee, E., Rabaud, V., Konolige, K. ve Bradski, G., "ORB: An efficient alternative to SIFT or SURF," 2011 International Conference on Computer Vision, 2011, pp. 2564-2571, doi: 10.1109/ICCV.2011.6126544.
- Ruiz, J. J., Caballero, F. ve Merino, L., "MGRAPH: A Multigraph Homography Method to Generate Incremental Mosaics in Real-Time From UAV Swarms," in *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2838-2845, Oct. 2018, doi: 10.1109/LRA.2018.2844304.
- Senapati, M., Anand, B., Barsaiyan, V. ve Rajalakshmi, P., "Geo-referencing system for locating objects globally in LiDAR point cloud," 2020 IEEE 6th World Forum on Internet of Things (WF-IoT), 2020, pp. 1-5, doi: 10.1109/WF-IoT48130.2020.9221162.
- Syetiawan, A., Gularso, H., Kusnadi, G. I., & Pramudita, G. N. (2020). Precise topographic mapping using direct georeferencing in UAV. *IOP Conference*

Series: *Earth and Environmental Science*, 500(1). <https://doi.org/10.1088/1755-1315/500/1/012029>

- Szeliski, R. (2021). *Computer Vision: Algorithms and Applications 2nd Edition*. <https://szeliski.org/Book>
- Tang, C., Ling, Y., Yang, X., Jin, W., & Zheng, C. (2018). Multi-View Object Detection Based on Deep Learning. *Applied Sciences*, 8(9), 1423. <https://doi.org/10.3390/app8091423>
- Thiel, C., & Schmullius, C. (2017). Comparison of UAV photograph-based and airborne lidar-based point clouds over forest from a forestry application perspective. *International Journal of Remote Sensing*, 38(8–10), 2411–2426. <https://doi.org/10.1080/01431161.2016.1225181>
- Turner, D., Lucieer, A. ve Wallace, L., "Direct Georeferencing of Ultrahigh-Resolution UAV Imagery," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 5, pp. 2738-2745, May 2014, doi: 10.1109/TGRS.2013.2265295.
- Turner, D., Lucieer, A., & Watson, C. (2012). An Automated Technique for Generating Georectified Mosaics from Ultra-High Resolution Unmanned Aerial Vehicle (UAV) Imagery, Based on Structure from Motion (SfM) Point Clouds. *Remote Sensing*, 4(5), 1392–1410. <https://doi.org/10.3390/rs4051392>
- Wang, H., Li, J., Wang, L., Guan, H. ve Geng, Z., "Automated mosaicking of UAV images based on SFM method," 2014 IEEE Geoscience and Remote Sensing Symposium, 2014, pp. 2633-2636, doi: 10.1109/IGARSS.2014.6947014.
- Wang, X., Liu, J., & Zhou, Q. (2016). Real-Time Multi-Target Localization from Unmanned Aerial Vehicles. *Sensors*, 17(12), 33. <https://doi.org/10.3390/s17010033>
- Xiang, H., & Tian, L. (2011). Method for automatic georeferencing aerial remote sensing (RS) images from an unmanned aerial vehicle (UAV) platform. *Biosystems Engineering*, 108(2), 104–113. <https://doi.org/10.1016/j.biosystemseng.2010.11.003>
- Xu, Q., Chen, J., Luo, L., Gong, W. ve Wang, Y., "UAV Image Mosaicking Based on Multiregion Guided Local Projection Deformation," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 3844-3855, 2020, doi: 10.1109/JSTARS.2020.3006289.
- Yahyanejad, S., Quaritsch, M. ve Rinner, B., "Incremental, orthorectified and loop-independent mosaicking of aerial images taken by micro UAVs," 2011 IEEE International Symposium on Robotic and Sensors Environments (ROSE), 2011, pp. 137-142, doi: 10.1109/ROSE.2011.6058531.
- Ye, Q., Zhang, X., Lin, Y., & Jian, J. (2022). ACCURACY ANALYSIS OF REAL-TIME OBJECT POSITIONING WITHOUT GCP FOR IMAGES FROM UAV OBLIQUE ULTRA-LONG FOCAL SMALL VIEW FIELD WHISKBROOM

CAMERA SYSTEM. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B1-2022, 43, 325–332. <https://doi.org/10.5194/isprs-archives-XLIII-B1-2022-325-2022>

- Zhao, H., Zhang, B., Wu, C., Zuo, Z., Chen, Z., & Bi, J. (2014). Direct georeferencing of oblique and vertical imagery in different coordinate systems. *ISPRS Journal of Photogrammetry and Remote Sensing*, 95, 122–133. <https://doi.org/10.1016/j.isprsjprs.2014.06.001>
- Zhao, X., Pu, F., Wang, Z., Chen, H. ve Xu, Z., "Detection, Tracking, and Geolocation of Moving Vehicle From UAV Using Monocular Camera," in IEEE Access, vol. 7, pp. 101160-101170, 2019, doi: 10.1109/ACCESS.2019.2929760.
- Zhao, Z. -Q., Zheng, P., Xu, S. -T. ve Wu, X., "Object Detection With Deep Learning: A Review," in IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 11, pp. 3212-3232, Nov. 2019, doi: 10.1109/TNNLS.2018.2876865.
- Zheng, F., Wang, F., Wu, J. ve Zheng, X., "A methodology of UAV route planning for fast image mosaicking," 2015 23rd International Conference on Geoinformatics, 2015, pp. 1-5, doi: 10.1109/GEOINFORMATICS.2015.7378663.
- Zhou, G., "Geo-Referencing of Video Flow From Small Low-Cost Civilian UAV," in IEEE Transactions on Automation Science and Engineering, vol. 7, no. 1, pp. 156-166, Jan. 2010, doi: 10.1109/TASE.2008.2010948.
- Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., & Ling, H. (2020). Detection and tracking meet drones challenge. *arXiv preprint arXiv:2001.06303*.
- Zhu, Z. ve ark., "A Comparative Study of Algorithms for Realtime Panoramic Video Blending," in IEEE Transactions on Image Processing, vol. 27, no. 6, pp. 2952-2965, June 2018, doi: 10.1109/TIP.2018.2808766.