



**T.C.
YALOVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**U-NET TABANLI EVRİŞİMLİ SİNİR AĞI İLE UZAKTAN ALGILANMIŞ
GÖRÜNTÜLERDEN OTOMATİK BİNA TESPİTİ**

**DOKTORA TEZİ
İbrahim DELİBAŞOĞLU
145305002**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Prof Dr. Müfit ÇETİN

TEMMUZ 2020



**T.C.
YALOVA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**U-NET TABANLI EVRİŞİMLİ SİNİR AĞI İLE UZAKTAN ALGILANMIŞ
GÖRÜNTÜLERDEN OTOMATİK BİNA TESPİTİ**

**DOKTORA TEZİ
İbrahim DELİBAŞOĞLU
145305002**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Prof Dr. Müfit ÇETİN

TEMMUZ 2020

YALOVA Üniversitesi Fen Bilimleri Enstitüsü'nün 145305002 numaralı Doktora Öğrencisi **İbrahim DELİBAŞOĞLU**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı **“U-NET TABANLI EVRİŞİMLİ SİNİR AĞI İLE UZAKTAN ALGILANMIŞ GÖRÜNTÜLERDEN OTOMATİK BİNA TESPİTİ”** başlıklı tezini aşağıda imzaları olan jüri önünde sunmuş ve oy birliği/oy çokluğu ile başarılı bulunmuştur.

İmza / Kanaati

(Kabul/Red)

Tez Danışmanı :

..... /.....

Jüri Üyeleri :

..... /.....

..... /.....

..... /.....

..... /.....

Teslim Tarihi : 18 Haziran 2020

Savunma Tarihi : 02 Temmuz 2020



ÖNSÖZ

Tez çalışmalarım süresince desteklerini esirgemeyen eşime, tecrübelerini paylaşan tez danışmanıma ve değerli jüri üyelerine teşekkürlerimi sunarım.

Temmuz 2020

İbrahim DELİBAŞOĞLU





İÇİNDEKİLER

	Sayfa No
KISALTMALAR	ix
ÇİZELGE LİSTESİ.....	xi
ŞEKİL LİSTESİ.....	xv
ÖZET	xxiii
ABSTRACT.....	xxv
1. GİRİŞ	1
2. HİPERSPEKTRAL GÖRÜNTÜLERDE BOYUT İNDİRGEME	5
2.1 Band Seçme Probleminde Kullanılan Veri Kümesi.....	9
2.2 Özellik Çıkarımı ile Boyut İndirme	10
2.2.1 Temel bileşen analizi.....	10
2.2.2 Bağımsız bileşen analizi.....	12
2.3 Bant Seçme ile Boyut İndirgeme	15
2.3.1 Benzerlik kriterleri	15
2.3.1.1 Spektral açı.....	15
2.3.1.2 Ortak bilgi	15
2.3.1.3 Kullback-Leibler ayrıştırma yöntemi.....	17
2.3.1.4 Yapısal benzerlik.....	17
2.3.1.5 Standardize edilmiş öklit mesafesi.....	18
2.3.2 Önerilen anahtar noktalar temelli bant seçme algoritmaları	18
2.3.2.1 Hiyerarşik kümeleme ile kümelerin oluşturulması	19
2.4 Sonuçlar	31
3. KLASİK YÖNTEMLER VE UNET İLE BİNA BÖLÜTLEMESİ.....	41



3.1 Kullanılan Görüntüler	44
3.2 Gölge tespiti	45
3.3 Derin öğrenme modeli oluşturulması.....	48
3.4 Bölütleme ve gölge doğrulama	54
4. BİNA BÖLÜTLEMESİ İÇİN ÖNERİLEN DERİN ÖĞRENME MİMARİLERİ.....	61
4.1 Görüntü işleme problemlerinde derin öğrenme yaklaşımı.....	62
4.2 Derin öğrenme ile bina bölütlemesi için literatür taraması	69
4.3 Inception Unet mimarisi.....	74
4.4 Önerilen Inception Unet-V2 mimarisi	84
4.5 Çok çıkışlı mimariler.....	90
4.6 Çapraz doğrulama ile performans testi	94
4.7 Massachusetts veri kümesi ile test sonuçları.....	95
5. HİPERSPEKTRAL VERİ SETİ İLE BAND SEÇME VE BİNA BÖLÜTLEME ...	101
5.1 Hiperspektral veri seti	101
5.2 Önerilen band seçme yöntemleri ve mimariler ile deneysel sonuçlar.....	104
6. SONUÇ VE DEĞERLENDİRME	113
KAYNAKLAR.....	115
ÖZGEÇMİŞ	123



KISALTMALAR

MS	: Çok bantlı (Multi-spectral)
HS	: Hiperspektral (Hyperspectral)
RGB	: Üç band (Kırmızı-Yeşil-Mavi)
PCA	: Temel Bileşen Analizi (Principal Component Analysis)
ICA	: Bağımsız Bileşen Analizi (Independent Component Analysis)
SA	: Spektral Açısı
MI	: Ortak Bilgi (Mutual Information)
KLD	: Kullback-Leibler ayrıştırma ölçütü
SSIM	: Yapısal benzerlik (Structural Similarity)
ROSIS-3	: Reflective Optics System Imaging Spectrometer
IP	: Anahtar noktalar (Interest points)
IPBS	: Anahtar noktalar ile bant seçme yöntemi
IPDM	: Anahtar noktalar mesafe metriği temelli bant seçme yöntemi
NIR	: Kıızıl ötesi (Near Infrared)
CNN	: Evrimsel Sinir Ağı



Çizelge 2.1 Hiyerarşik kümelemede kullanılan bazı birleştirme yöntemleri	20
Çizelge 2.2 Pavia Üniversitesi eğitim seti.....	33
Çizelge 2.3 RetigaEx (Portakal) eğitim seti.....	33
Çizelge 2.4 RetigaEx verisi için GD ve OD kıyaslamaları	38
Çizelge 2.5 Pavia verisi için GD ve OD performans kıyaslamaları.....	38
Çizelge 3.1 Kullanılan görüntülerin özet bilgisi	44
Çizelge 3.2 IKONOS 1 numaralı görüntü ile eğitim sonuçları (model-1)	51
Çizelge 3.3 Quickbird 5 ve 6 numaralı görüntüler ile eğitim sonuçları (<i>model-2</i>).....	51
Çizelge 3.4 1,2,3,4,5,6 numaralı görüntüler ile eğitim sonuçları (<i>model-3</i>).....	53
Çizelge 3.5 3 farklı eğitim seti ile oluşturulan modellerin performans kıyaslaması.....	53
Çizelge 3.6 Ek bölütleme algoritması ile elde edilen sonuçlar	58
Çizelge 3.7 Farklı yöntemlerin Piksel tabanlı F_1 skorları	59
Çizelge 4.1 Farklı katmanlar içeren VGG mimari detayları [61]	66
Çizelge 4.2 <i>Inception Unet</i> mimarisinin detayları	78
Çizelge 4.3 Test edilen blok içerikleri	79
Çizelge 4.4 Farklı bloklar ile oluşturulan mimarilerin performansları (F_1).....	80
Çizelge 4.5 Eğitim seti-2 ile elde edilen performanslar (F_1).....	84
Çizelge 4.6 Eğitim seti-1 model performansları (F_1).....	88
Çizelge 4.7 Eğitim seti-2 model performansları (F_1).....	88
Çizelge 4.8 RGB görüntülerin performans kıyaslaması (P-R- F_1).....	89
Çizelge 4.9 RGB-NIR görüntülerin performans kıyaslaması	90
Çizelge 4.10 Eğitim seti-1 çok çıkışlı model performansları (F_1)	93
Çizelge 4.11 Eğitim seti-2 çok çıkışlı model performansları (F_1)	93
Çizelge 4.12 Çapraz doğrulama ile performans kıyaslaması	95



Çizelge 4.13 Massachusetts veri kümesi performans kıyaslaması (P-R-F ₁).....	99
Çizelge 5.1 Farklı bant seçme yöntemi ve mimarilerin test doğruluk oranları.....	111





Şekil 2.1 HS görüntü küpü	5
Şekil 2.2 İki farklı HS görüntünün korelasyon matrisi	6
Şekil 2.3 (a) Pavia Üniversitesi (b) RetigaEx	10
Şekil 2.4 RetigaEx görüntülerinin temel bileşenleri. (a) 1. Temel bileşen (b) 2. Temel bileşen (c) 3. Temel bileşen.....	12
Şekil 2.5 Pavia görüntülerinin temel bileşenleri. (a) 1. Temel bileşen (b) 2. Temel bileşen (c) 3. Temel bileşen.....	12
Şekil 2.6 Portakal görüntülerinin bağımsız bileşenleri. (a) 1. bileşen (b) 2. bileşen (c) 3. bileşen.....	14
Şekil 2.7 Pavia Üniversitesi görüntülerinin bağımsız bileşenleri. (a) 1. bileşen (b) 2. bileşen (c) 3. bileşen	14
Şekil 2.8 Farklı dalga boylarındaki görüntülerin entropileri.....	16
Şekil 2.9 Hiyerarşik kümeleme ile oluşturulan ağaç yapısı	21
Şekil 2.10 Hiyerarşik kümeleme sonucu elde edilen kümelerin 2 boyutta gösterimi (a) Average (b) Ward	22
Şekil 2.11 SIFT ile anahtar noktaların bulunması [29]	23
Şekil 2.12 Pavia üniversitesi görüntüsünün ilk ve son bandındaki IP	24
Şekil 2.13 Pavia farklı iki kümede yer alan bandların IP çıkarılmış görüntüleri	24
Şekil 2.14 RetigaEx farklı iki kümede yer alan bandların IP çıkarılmış görüntüleri	25
Şekil 2.15 Pavia verisi için farklı iki kümede yer alan bantların bilgilerini ölçen ağırlık katsayıları	27
Şekil 2.16 Pavia aynı kümede yer alan bandların özellik katsayıları.....	28
Şekil 2.17 Veri kümelerindeki tüm bantların IP özellik katsayıları (a) Pavia (b) RetigaEx	29
Şekil 2.18 Beş küme için her iki veri setindeki kümelerin ağırlık katsayıları (a) Pavia (b) RetigaEx	29
Şekil 2.19 Filtrelenmiş ve morfolojik işleme tabi tutulmuş görüntüler ile fark görüntüsü	30
Şekil 2.20 Pavia Üniversitesi görüntüsü, eğitim seti ve doğru sınıf değerleri	32



Şekil 2.21 RetigaEx görüntüsü, eğitim seti ve doğru sınıf değerleri	32
Şekil 2.22 RetigaEx (Portakal) görüntüsünün önerilen IPBS yöntemi ile farklı benzerlik ölçütlerine göre k-NN sınıflandırma sonuçları	34
Şekil 2.23 Pavia görüntüsünün önerilen IPBS yöntemi ile farklı benzerlik ölçütlerine göre k-NN sınıflandırma sonuçları	35
Şekil 2.24 RetigaEx verisi için k-NN sınıflandırma sonuçları.....	36
Şekil 2.25 Pavia verisi için k-NN sınıflandırma sonuçları.....	37
Şekil 3.1 Veri setinden örnek görüntüler ve etiketler	44
Şekil 3.2 RGB ve NIR-R-G görüntüler.....	46
Şekil 3.3 Gölge ve bitki maskesi.....	46
Şekil 3.4 NDVI bitkisel alanlar	47
Şekil 3.5 Gölge maskesi.....	47
Şekil 3.6 Bina bölütlemeye kullanılan Unet mimarisi[43].....	49
Şekil 3.7 Veri setindeki örnek görüntüler için bulunan binalar (<i>model-1</i>)	52
Şekil 3.8 testpatch4 ve testpatch9 görüntüleri için bulunan binalar	54
Şekil 3.9 testpatch9 bölütleme ve bina adaylarının bulunması	55
Şekil 3.10 Gölge yönü (Gyön) ve mesafeye (dmin) göre gölge-bina eşleştirilmesi	56
Şekil 3.11 Hatalı bina adayları ve eşleştirildikleri gölge alanlar	57
Şekil 3.12 Gölge doğrulama ile hatalı bina adaylarının elenmesi.....	57
Şekil 3.13 Gölge doğrulama ile bina adayları arasından eleme yapılması ve derin öğrenme modeli ile sonuçların birleştirilmesi	58
Şekil 4.1 Nesne tanıma amaçlı tasarlanmış klasik CNN modeli [62]	62
Şekil 4.2 Bölütleme amaçlı tasarlanmış CNN modeli	63
Şekil 4.3 2x2 boyutlu pencere ve 2x2 kaydırma adımları ile ortaklama işlemi.....	64
Şekil 4.4 3x3 boyutlu pencere ile konvolüsyon işlemi	64
Şekil 4.5 SegNet mimarisi [64].....	65
Şekil 4.6 CNN ve ResNet CNN katmanları [62]	68
Şekil 4.7 Farklı katman sayılarına sahip ResNet mimarileri [62].....	68



Şekil 4.8 Çok bantlı bina bölütleme yöntemi [66]	69
Şekil 4.9 Bina bölütleme için önerilen mimarinin detayları [66].....	70
Şekil 4.10 FuseNet mimarisi [69]	72
Şekil 4.11 2 SegNet mimarilerinden oluşan bölütleme modeli [69].....	72
Şekil 4.12 İki alt model içeren bina tespit modeli [71]	73
Şekil 4.13 Inception modülleri [60]	75
Şekil 4.14 Unet modellerinde kullanılan bağlantı bloğu.....	76
Şekil 4.15 Inception blok ile genişletilmiş mimari	76
Şekil 4.16 Inception Unet mimarisi	77
Şekil 4.17 Farklı Inception blokları ile eğitim performansları.....	80
Şekil 4.18 1 ve 12 numaralı görüntülerin bölütleme sonuçları	81
Şekil 4.19 Eğitim seti-2 oluşturulmasında kullanılan görüntüler.....	82
Şekil 4.20 Unet ve Inception Unet eğitim performansları (eğitim seti-2).....	83
Şekil 4.21 Önerilen Inception Unet-V2 mimarisi	85
Şekil 4.22 Inception Unet ve Inception Unet-V2 eğitim performansları	86
Şekil 4.23 testpatch5 ve testpatch7 görüntüleri için elde edilen sonuçlar (a) RGB girdi görüntüsü (b) etiketlenmiş binalar (c) Unet (d) Inception Unet (e) Inception Unet-V2	87
Şekil 4.24 testpatch7 görüntüsünden kırılan bir alan; (a) RGB görüntü (b) etiketlenmiş binalar (c) unet (d) Inception Unet (e) Inception unet-V2	87
Şekil 4.25 Inception Unet çok çıkışlı mimari.....	91
Şekil 4.26 Inception Unet çok çıkışlı mimari ile elde edilmiş küçük ve büyük bina örnek sonuç görüntüleri	92
Şekil 4.27 Veri seti-1 için eğitim ve doğrulama grafikleri.....	94
Şekil 4.28 Massachusetts veri kümesinden örnek görüntüler	96
Şekil 4.29 Kullanılan Inception bağlantı bloğu.....	97
Şekil 4.30 Massachusetts eğitim seti-I eğitim ve doğrulama grafikleri	98
Şekil 4.31 Massachusetts eğitim seti-II eğitim ve doğrulama grafikleri.....	98
Şekil 5.1 Kullanılan HS veri setinin RGB görünümü	101



Şekil 5.2 HS veri setinden farklı bantlara ait görüntüler.....	102
Şekil 5.3 Etiketlenmiş binalar	103
Şekil 5.4 Veri setinden örnek görüntüler	104
Şekil 5.5 WaluMi yöntemi ile seçilen 3 bant ile eğitim performansları	105
Şekil 5.6 IPBS yöntemi ile seçilen 3 bant ile eğitim performansları	106
Şekil 5.7 IPDM yöntemi ile seçilen 3 bant ile eğitim performansları	106
Şekil 5.8 Farklı band seçme yöntemleri ile seçilen 7 bantlı görüntüler ile eğitilen <i>Unet</i> eğitim grafiği	107
Şekil 5.9 Farklı band seçme yöntemleri ile seçilen 7 bantlı görüntüler ile eğitilen <i>Inception-Unet</i> eğitim grafiği	108
Şekil 5.10 Farklı band seçme yöntemleri ile seçilen 7 bantlı görüntüler ile eğitilen <i>Inception-Unet-V2</i> eğitim grafiği	109
Şekil 5.11 Farklı band seçme yöntemleri ile seçilen 15 bantlı görüntüler ile eğitilen <i>Unet</i> eğitim grafiği	109
Şekil 5.12 Farklı band seçme yöntemleri ile seçilen 15 bantlı görüntüler ile eğitilen <i>Inception-Unet</i> eğitim grafiği	110
Şekil 5.13 Farklı band seçme yöntemleri ile seçilen 15 bantlı görüntüler ile eğitilen <i>Inception-Unet-V2</i> eğitim grafiği	110



U-NET TABANLI EVRİŞİMLİ SINIR AĞI İLE UZAKTAN ALGILANMIŞ GÖRÜNTÜLERDEN OTOMATİK BİNA TESPİTİ

ÖZET

Uzaydan ve havadan çekilen görüntülerdeki spektral ve uzamsal çözünürlüğün artması ile birlikte, bina ve ağaç gibi nispeten küçük nesnelerin tanınmasına olanak sağlanmıştır. Dolayısıyla uzaktan algılanmış görüntülerden nesne tanıma ve nesne bölütleme üzerine yapılan çalışmaların önemi giderek artmıştır.

Hiperspektral (HS) görüntülerin işlenmesinde bina bölütleme doğruluğunu ve işlem hızını arttırmak için yüksek spektral veri boyutunun düşürülmesine ihtiyaç duyulabilmektedir. Bu amaçla HS veri setinde spektral bant seçme işlemi yapılarak en fazla bilgiyi içeren bir spektral bant alt kümesi oluşturulmuştur. Literatürdeki çeşitli benzerlik kriterleri ve arama yöntemleri incelenerek, HS görüntülerdeki yapısal bilgiyi de göz önüne alan hiyerarşik kümelemenin kullanıldığı iki yeni bant seçme yöntemi geliştirilmiştir. Ayrıca bir HS görüntüdeki iki spektral bandın birbirlerine benzerliğini ölçen yeni bir metrik önerilmiştir. Sonuçlar uydu ve yakın mesafe HS görüntüleri olmak üzere içerik olarak farklı iki veri seti üzerinden kıyaslanmıştır.

Spektral bant seçme sürecinin ardından ise, bina bölütlemesine yönelik literatürdeki klasik ve modern görüntü işleme temelli yöntemler incelenmiş, hangi yöntemlerin kullanıldığı ve çalışmaların ne aşamada olduğu araştırılmıştır. Tez çalışması kapsamında, görüntü işlemede farklı problemlerin çözümünde son derece başarılı sonuçlar veren derin öğrenme temelli yöntemler kullanılmıştır. Bu kapsamda, U-net temelli bölütleme mimarisi geliştirilmiştir. Tez çalışmasında kullanılan çok bantlı (MS) ve HS veri setlerine uygulanan yeni mimari ile daha az eğitim adımı neticesinde daha iyi performanslar elde edildiği görülmüştür. Ayrıca, HS veri seti için önerilen bant seçme ve bina bölütleme yöntemleri bir arada kullanılarak sonuçlar analiz edilmiştir.

Anahtar Kelimeler: uzaktan algılanmış görüntüler, spektral bant seçme, derin öğrenme, bina bölütleme, bina tanıma



AUTOMATIC BUILDING DETECTION FROM REMOTELY SENSED IMAGES WITH U-NET BASED CONVOLUTIONAL NEURAL NETWORK

ABSTRACT

With the increase in spatial resolution of satellite and aerial imagery, it is allowed to recognize relatively small objects such as buildings and trees. Thus, studies on object recognition and object segmentation have started to be performed in aerial images.

It may be necessary to reduce the huge spectral dimension to increase processing speed and building segmentation accuracy in HS image processing. For this purpose, it is aimed to select a subset of the bands containing the most information by performing the band selection process in HS images. Various similarity criteria and search methods in the literature are investigated and two band selection methods using hierarchical clustering by taking structural information in images into consideration are proposed. Also a new metric is defined that measures the similarity of the two bands in the HS image. The results were compared on two different data sets: satellite and close-range images.

After the spectral band selection process, classical and modern image processing based studies are carried out for building segmentation, it is investigated that which methods are used as popular and the stage of the studies in the literature. In the thesis, deep learning based methods which became very popular in solving different problems in image processing, are investigated. Unet-based object segmentation architecture is proposed. Two different data sets consisting of MS and HS images are used for performance comparison, and it is shown that better performances have been achieved with the proposed architecture as a result of fewer training steps. Also, the results are analyzed using the proposed band selection and building segmentation methods on the labeled HS data set.

Keywords: remote sensed images, spectral band selection, deep learning, building segmentation, building detection



1. GİRİŞ

Dünya nüfusundaki artışla birlikte kentsel alanlardaki büyümenin yönetilmesi ve değişimlerin izlenmesi çevresel, sosyal ve ekonomik açıdan günümüzde önemli bir unsur haline gelmiştir. 2000'li yıllara kadar olan süreçteki ilk nesil yer gözlem uydularının düşük uzaysal çözünürlüğe sahip olmasından dolayı binalar, araçlar, ağaçlar ve yollar gibi küçük nesnelerin mekânsal olarak ayırt edilmesi neredeyse imkansızdı. Son yıllarda gelişen uzay teknolojisi ile uzamsal çözünürlüğü santimetre mertebelerine ulaşan çok yüksek çözünürlüklü (VHR) uydu görüntüleri elde edilmeye başlandı. Bunun yanı sıra havadan uçaklar vasıtasıyla da bölgesel olarak çok bantlı görüntüler elde edilebilir hale geldi. Böylece kent alanlarındaki binaların, yolların ve yeşil alanların tespiti gibi yeni çalışmaların önü de açılmış oldu. Bu kapsamda, kentsel alanlardaki binaların otomatik tespitine yönelik uzaktan algılama çalışmaları da giderek önem kazanmıştır.

Uzaydan veya havadan çekilen görüntülerde üç veya daha fazla bant içeren görüntüler elde edilebilmektedir. Bu görüntüleri Hiperspektral (HS) veya çok bantlı (MS) görüntüler olarak iki sınıfa ayırabiliriz. HS görüntüleme teknolojisi ile ışık spektrumunda çok dar bir aralığa sahip farklı dalga boylarında görüntüler elde edilebilmektedir. HS görüntü ile kastedilen şey, çok dar dalga boyu aralıklarında spektral görüntülerin elde edilmesidir. Böylece, insan gözünün görebildiği spektrum aralığından (400-700 nm) başka birçok dalga boyunda görüntüler elde edilerek, nesnelere ait daha detaylı bilgiler elde edilmektedir. Bu özelliği dolayısıyla HS görüntüler hedef tespiti, sınıflama gibi alanlarda MS görüntülere göre önemli bir avantaj sağlamaktadır. Özellikle uzaydan ve havadan çekilen HS görüntüler son yıllarda yaygın olarak kullanılmaktadır. Şehircilik, bitki örtüsü tespiti, madencilik, su kaynakları ve tarımsal ürünler, askeri alandaki uzaktan algılama uygulamaları uzaktan algılanmış görüntülerin bazı kullanım alanlarıdır. Bunların dışında ise, yakın mesafeden çekilen görüntülerin kullanıldığı mikrobiyoloji, tarım ürünlerinin ürünlerin tazeliğinin/kalitesinin tespiti, adli bilişim gibi farklı uygulama alanlarında HS

görüntüleme teknolojisi kullanılmaktadır. Dolayısıyla HS görüntüler ile insan gözünün algılayabildiği görünür bandın limitleri aşılmaktadır.

HS görüntüler band sayısının fazlalığı sebebiyle her ne kadar zengin bir spektral içerik sunsa da, bazı durumlarda problemin çözümünde kullanılan algoritmanın performansını veya işlem süresini olumsuz etkileyebilmektedir. Bu sebeple bazı uygulamalarda HS görüntüden daha fazla bilgi içeren bantların seçilmesi (bant seçme) veya yüksek sayıdaki bantların daha az sayıda bant ile ifade edilmesi (bant azaltma/indirgeme) gibi yöntemler tercih edilebilmektedir. Bu tez kapsamında HS görüntülerde bant seçme ve bant düşürme yöntemleri hakkında literatür çalışması yapılarak iki yeni yöntem geliştirilmiştir. Önerilen yöntemler literatürde kabul görmüş metotlar ile karşılaştırılmış, sonuçlar analiz edilmiştir.

Son yıllarda, uzaktan algılanmış görüntülerden nesne tanıma/bölütleme konusu literatürde üzerinde yoğun çalışılan bir başka önemli konudur. Tez kapsamında şehirlerin otomatik gözlemlenmesi, yapılaşmanın/ değişikliklerin analiz edilmesi, insan nüfusu tahmini gibi kullanım alanlarına sahip bina bölütleme konusunda gerçekleştirilen bir çok çalışma incelenmiştir. Bu kapsamda ilk olarak literatürdeki klasik görüntü işleme temelli yöntemler incelenmiş, hangi yöntemlerin kullanıldığı ve çalışmaların ne aşamada olduğu gözlemlenmiştir. Ardından görüntü işlemede son derece popüler olmaya başlayan, birçok farklı problemin çözümünde klasik yöntemlere nazaran daha iyi performans sergileyen ve bir çeşit makine öğrenmesi yöntemi olan derin öğrenme çalışmaları incelenmiştir. Literatürde bölütleme problemlerinde başarısı kanıtlanmış Unet mimarisi ile birlikte klasik yöntemlerin bir arada kullanıldığı hibrit bir yöntem uzaktan algılanmış görüntülerden bina bölütleme problemi için önerilmiştir. Bu yöntem kapsamında sadece keskin yapıda binaların tespit edilebildiği bir yardımcı bölütleme modülü derin öğrenme ile birlikte uygulanmıştır. Bu geliştirilen yöntemde, klasik yöntemlerin kullanıldığı literatür çalışmalarındaki gibi gölge bilgisinden hareket edilerek keskin yapıdaki binaların tespitine yönelik bir yöntem önerilmiştir. Geliştirilen bu yöntemde, özellikle klasik yöntemlerin karmaşık şehir görüntülerinde başarısız sonuçlar sergilediği, birden çok aşamalı işlemler gerektirdiği ve farklı tip görüntüler için özellik çıkarımında bazı zorluklar ile karşılaşıldığı tespit edilmiştir. Dolayısıyla, literatür

alışmalarındaki derin ğrenme temelli yöntemlerin nesne tanıma problemlerinde başarısı göz önüne alınarak Unet temelli bina bölütleme mimarisi geliştirilmiştir. Önerilen mimari ile hem doğruluk oranı arttırılmış hem de ağın daha hızlı eğitildiğı görölmüştür. Performans kıyaslaması için uydu ve hava MS görüntülerinden oluşan iki farklı veri seti kullanılarak eğitim kümeleri ile kıyaslamalar gerçekleştirilmiştir. Son bölümde ise bant seçme ve bina bölütleme yöntemlerinin bir arada kullanılması için HS veri seti hazırlanmıştır. İlk olarak HS görüntüdeki binalar elle etiketlenmiştir. Daha sonra, ilk bölümde önerilen bant seçme metotları kullanılarak bina bölütlemesi için geliştirilen mimariler ile birlikte test edilmiştir. Farklı bant sayılarında girdiler kabul eden derin ğrenme mimarileri, oluşturulan HS veri seti ile eğitilmiş ve sonuçlar analiz edilmiştir. Oluşturduğumuz yeni veri kümesi üzerinde derin ğrenme yöntemleriyle performans kıyaslamasının gerçekleştirildiğı son bölümde, bant seçme yöntemlerini incelediğimiz bölümde elde edilen çıktıları destekler nitelikte sonuçlar alınmıştır.

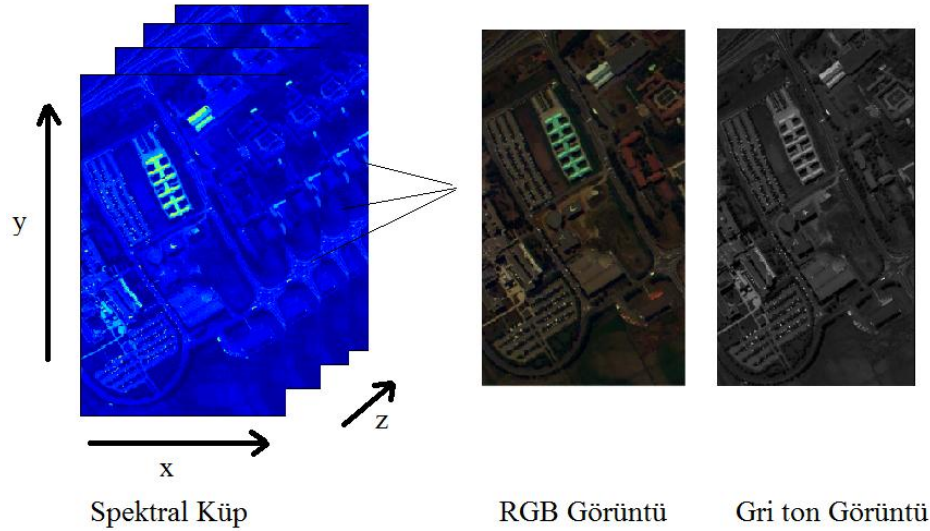




2. HİPERSPEKTRAL GÖRÜNTÜLERDE BOYUT İNDİRGEME

HS görüntüler her bir piksel için onlarca veya yüzlerce spektral banttı oluşmakta ve dalga boyları birbirlerine yakın olması sebebiyle bantlar arasında yüksek korelasyona sahip bir ilişki bulunmaktadır. Çok sayıda ve farklı dalga boylarındaki spektral görüntüler, görünür bölge dışında da zengin bir bilgiye sahiptir. Her bir görüntüye ait bilgi temsil edilen spektral bölgenin fiziksel ve kimyasal özelliklerine bağlıdır. HS görüntüleme teknolojisi; uzaktan algılama, gıda endüstrisinde kalite analizi [1], adli tıpta kan analizi [2], döküman analizi [3], minerallerin haritalandırılması [4], ilaçların analizi [5] gibi farklı disiplinlerde kullanılmaktadır.

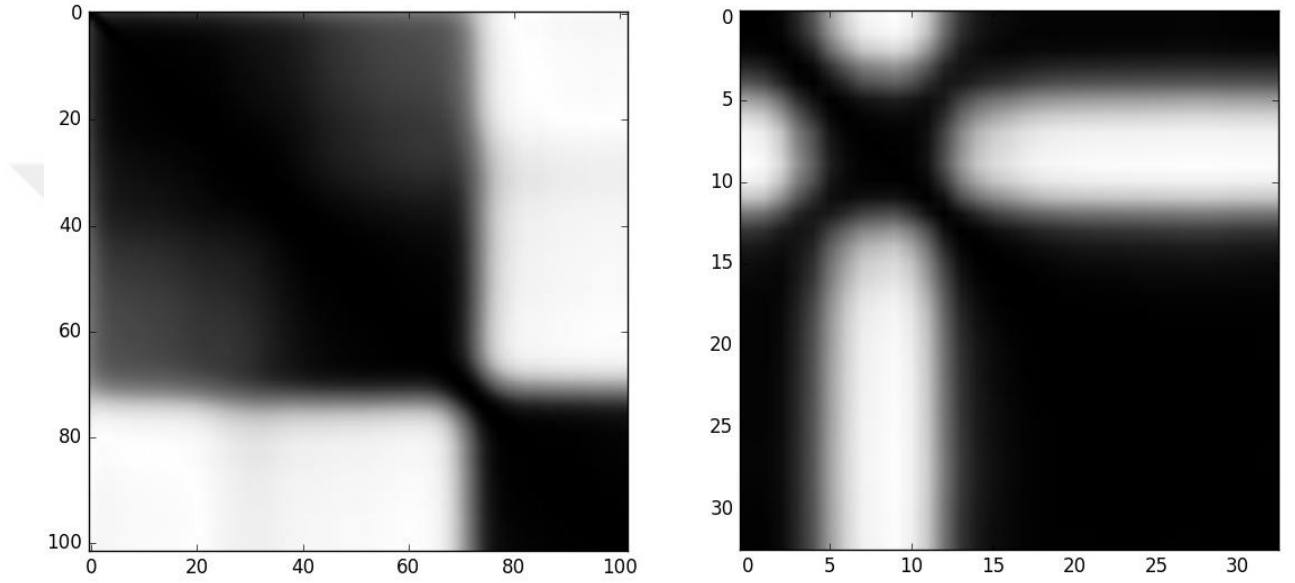
HS görüntüler, iki boyutlu uzamsal bilgiyi ve üçüncü boyut olarak spektral bilgiyi içerisinde barındırdığından, HS görüntü küpü olarak isimlendirilir. İlk iki boyut görüntünün uzamsal koordinatlarını ifade eder ve her bir (x,y) noktasına piksel adı verilir. HS verilerde her bir spektral bant aralığı bir katman olmak üzere z ekseninde yüzlerce görüntü bulunmaktadır (Şekil 2.1).



Şekil 2.1 : HS görüntü küpü

HS verilerin yüzlerce spektral banda sahip olması, görüntülerin etkin ve hızlı analizinde bir takım sorunlar ortaya çıkarmaktadır. HS görüntülerde spektral bantlar arasında genelde yüksek derecede korelasyon olduğu göz önüne alındığında, analiz yapılırken

veri boyutunun düşürülmesi ve ayırt edici özelliklerin ön plana çıkarılması görüntü işleme açısından büyük önem arz etmektedir. Şekil 2.2’de sırasıyla 103 ve 33 band içeren iki HS görüntünün bantları arasındaki ilişkiyi ifade eden korelasyon matrisi gösterilmiştir (koyu renkli bölgeler yüksek derecede ilişkili spektral bantları ifade etmektedir).



Şekil 2.2 : İki farklı HS görüntünün korelasyon matrisi

HS görüntülerde gereksiz bilginin atılması ve anlamlı bilginin çıkarılması maksadıyla uygulanan yöntemler iki temel başlık altında toplanabilir. İlki, çok boyutlu veri setinin daha az boyutlu bir başka uzaya dönüştürülmesidir. Bu yaklaşımı içeren yöntemler HS görüntülerde *özellik çıkarımı* başlığı altında toplanır. Temel bileşen analizi (PCA) ve bağımsız bileşen analizi (ICA) gibi eğiticişiz yöntemler bu amaçla en yaygın kullanılan tekniklerdir [6, 7]. Bu tip yöntemler ayırt edici nitelikte özellikler çıkarmada başarılıdır [8]. Fakat özellik çıkarımı temelli yöntemlerin fiziksel açıdan yorumlanabilirliğin güç olmasından dolayı HS görüntülerde bir diğer yaklaşım olan *bant seçme* yöntemleri daha yaygın olarak kullanılmaktadır. Zaten HS görüntülerdeki yoğun spektral bilgi, her bir farklı dalga boyunda farklı materyallerin tanınması için ihtiyaç duyulan bilgiyi içermektedir. Bunun yanında önemli olan daha yoğun bilgi içeren dalga boylarının keşfedilmesidir. Dolayısıyla, güvenilir bir bant seçme algoritması özellik çıkarımı

algoritmalarına nispeten daha sık tercih edilir. Böylelikle ilgili spektral bantların içermiş olduğu anlamlı bilgi korunmuş olacaktır [9,10].

HS görüntülerde bant seçme işlemi eğitici ya da eğitici olmayan yöntemler ile gerçekleştirilebilir. Eğitici bant seçme algoritmaları genel olarak, sınıfların ayırt edilmesinde kullanılan ölçütler gibi bazı optimizasyon kriterlerine göre arama yapılması problemi olarak düşünülür [11,12]. Fakat eğitici yöntemlerin ihtiyaç duyduğu ön bilgi genellikle bulunmayacağından dolayı eğitici olmayan bant seçme yöntemleri daha önemlidir ve tercih edilmektedir. Eğitici olmayan bant seçme yöntemlerinde herhangi bir ön bilgi ya da eğitim setine ihtiyaç duyulmaz. Bu yöntemler genellikle tüm bantlar içerisinde en fazla bilgiyi içeren bantların bulunması mantığı üzerine kuruludur [13,14]. Bu kapsamda bantların birbirlerinden ayırt edilmesini sağlayacak olan ayırt edicilik kriteri, alt bant kümesini aramak için kullanılacak arama yöntemi ve seçilecek optimum sayıdaki bant sayısı temel problemler olarak sıralanabilir. Genelde, bu alanda gerçekleştirilen çalışmalar ayırt edicilik ölçütü ve arama stratejileri üzerine yoğunlaşmıştır.

HS görüntülerde spektral bantlar saf spektral bilgi içerdiğinden dolayı, literatürde gerçekleştirilen çalışmalarda bantları birbirlerinden eğitici olmayan olarak ayırt edebilecek kriterler üzerine yoğunlaşmıştır. Bu kapsamda, spektral açı, ortak bilgi (mutual information-MI) ve Kullback-Leibler ayrıştırma (KLD) gibi benzerlik ölçütleri kullanılır. Örneğin; 2014 yılında Chang ve Liu'nun gerçekleştirdiği bir çalışmada, Spektral açı eşleme (SAM) ve KLD ölçütleri bantların birbirleri ile yakınlığını/uzaklığını belirlemek amacıyla kullanılmıştır [15.] Bantlar birbirleri ile karşılaştırılırken maksimum bilgiyi içeren bir alt kümenin bulunması amaçlanmaktadır. Bu durumda seçilebilecek tüm olası seçeneklerin araştırılması gerekmektedir. Bu ise bant sayısına bağlı olarak çok uzun zaman alabilecek ve aşırı işlem yükü gerektiren bir yöntemdir. Bunun yerine farklı arama yöntemlerine başvurulur. Bu amaçla kullanılan yöntemlerden birisi, birbirlerine benzer olan bantların aynı küme içerisinde toplanmasıdır. Bu açıdan bakıldığında bant seçme işlemi bir kümeleme problemidir. Bu amaçla benzer bantları bir arada toplayarak, alt bantlar arasındaki varyansı maksimum ve bantların kendi içerisindeki varyansı ise minimum yapan hiyerarşik kümeleme yöntemi bant seçmek için önerilmiştir [9]. Bu makalede bantlar arasında kümeleme

yapılırken KLD ve MI mesafe ölçütleri kullanılmıştır. Hiyerarşik kümeleme yapılırken ise John Ward' un birleştirme stratejisi kullanılmıştır [16]. Bu yöntemin diğer hiyerarşik kümeleme algoritmalarından daha iyi performans gösterdiği literatürde gerçekleştirilen çalışmalarda ortaya konulmuştur [17]. HS görüntülerde bantları kümelemek için kullanılan yöntemlerden bir diğeri de noktaların birbirleri arasında 'mesaj göndermeleri' suretiyle çalışan bir kümeleme algoritması olan AP (Affinity propagation) kullanılarak gerçekleştirilmiştir [18]. Hedjam ve Cheiret tarafından yapılan çalışmada ise graflar kullanılarak Markov kümeleme algoritması ile bant seçme işlemi gerçekleştirilmiştir [19].

Kümeleme temelli metotlar, kümeleme algoritmalarının direk olarak HS görüntülere uygulanması olarak da düşünülebilir. Fakat bandların kümelenmesine ihtiyaç duyulmadan gerçekleştirilen band seçme yöntemleri de mevcuttur. Bu amaçla Sun ve arkadaşları tarafından gerçekleştirilen bir çalışmada, bilgi içeren bantlar bir arama stratejisine ihtiyaç duyulmadan band önceliklendirme ile seçilmiştir [20]. Bu makalede bantlar arasındaki mesafeler yine belirli bir benzerlik kriterine göre ölçüldükten sonra, yoğunluk bölgeleri ve her bir bandın kendisine en yakın yoğun bölgeye olan mesafesi kullanılarak bantlara önceliklendirme değerleri atanmıştır. Böylelikle bantlar önceliklendirme değerlerine göre sıralandığında, en üstte yer alan bantlar daha fazla bilgi içerip diğerlerine nazaran seçilmeleri daha anlamlı olarak nitelendirilmiştir.

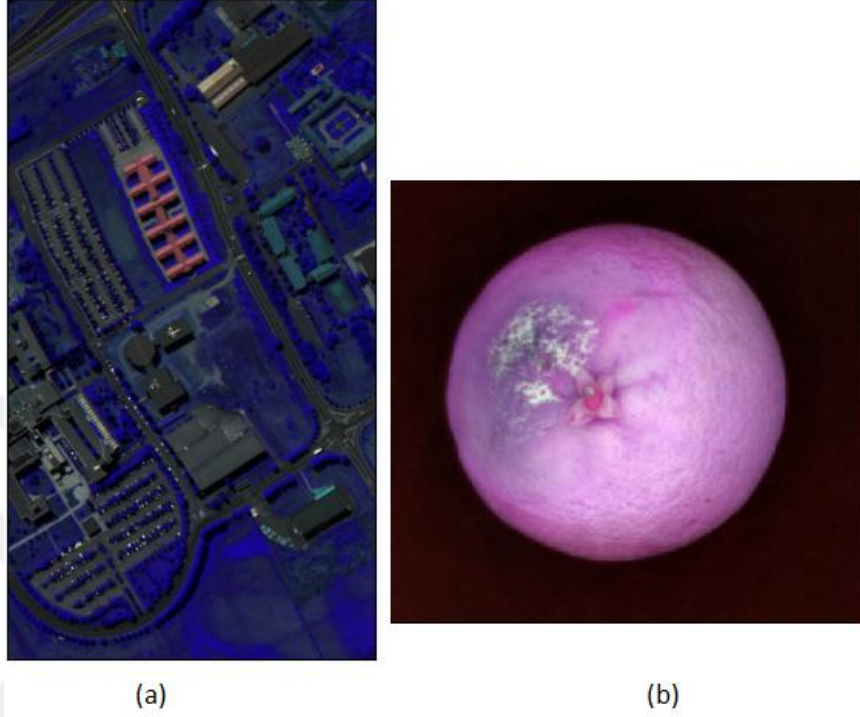
Bunların yanı sıra, bir görüntü içerisinde farklı nesneleri birbirinden ayırt etmede kenar bilgilerinin önemli olduğunu düşünüldüğünde, farklı nesnelerin kenarlarının daha iyi ayırt edilebildiği görüntülerin daha fazla bilgi içerdiği düşünülebilir. Bu bakış açısından yola çıkılarak gerçekleştirilmiş bir çalışmada [21] her bir bandın ne kadar bilgi içerdiğini belirlemek için görüntünün yatay ve dikey yönde farkları (türevleri) ile spektral bantlar arasındaki türevler kullanılarak bir özellik çıkarımı yapılmıştır. Bu çalışmada görüntünün yapısal özelliklerinin de göz önüne alınarak yapılan bir band seçme işleminin performansı arttırabildiği üzerinde durulmuş, graflar kullanılarak gerçekleştirilen bir arama işlemi ile band seçme işlemi gerçekleştirilmiştir.

Bu bölümde ilk olarak yaygın kullanılan dönüşüm bazlı temel bileşen analizi (PCA) ve bağımsız bileşen analizi (ICA) özellik çıkarma yöntemleri anlatılmıştır. Daha sonra ise,

HS görüntülerde bandların birbirleri ile benzerliğini ölçen kriterler ve gerçekleştirilen band seçme işlemi örnek veri setleri üzerinden anlatılmıştır. Hiyerarşik kümelemenin benzer bantları bir araya toplamada gösterdiği başarı göz önüne alınarak, hiyerarşik kümeleme ile birlikte yapısal bilginin de kullanıldığı bir band seçme algoritması önerilmiş ve sonuçlar iki veri seti üzerinden test edilmiştir.

2.1 Band Seçme Probleminde Kullanılan Veri kümesi

Bu bölümde tanıtılan metotların görseller üzerinden açıklanması için genel erişime açık iki HS veri seti kullanılmıştır. Bunlar; RetigaEx (Portakal) ve Pavia Üniversitesi görüntüleridir. Pavia Üniversitesi görüntüsü ROSIS-3 (Reflective Optics System Imaging Spectrometer) sensörü ile elde edilmiş İtalya'nın Pavia Üniversitesi kampüsüne ait bir görüntüdür [22]. Bu görüntü 0.43 ile 0.86 nm spektral aralığında kaydedilmiş 103 spektral banttan oluşmaktadır. Uzamsal çözünürlük 1.3 metre olup, görüntünün piksek çözünürlüğü 610 x 340'dır. Bu görüntü HS band seçme algoritmalarının test edilmesinde yaygın olarak kullanılmakta olup, etiketlenmiş 9 farklı sınıf içermektedir. Bunlar; Asfalt, çayır, çakıl, ağaç, boyalı metal, toprak, katran, tuğla ve gölgedir. Kullanılan bir diğer görüntü ise RetigaEx spektrografi ile elde edilmiş bir kısmı çürük olan bir portakal görüntüsüdür [23]. Bu görüntü 400-720 nm aralığında 10 nm spektral çözünürlükte kaydedilmiş 33 adet spektral banttan oluşmaktadır. Piksel çözünürlüğü 320 x 300'dür. Görüntü etiketlenmiş 4 farklı sınıf içermektedir. Bunlar; arka plan, portakalın sağlıklı kısmı, çürük kısım ve sap kısmıdır. Bu veri setlerine ait RGB görünümüler Şekil 2.3'te gösterilmiştir.



Şekil 2.3 : (a) Pavia Üniversitesi (b) RetigaEx

2.2 Özellik Çıkarımı ile Boyut İndirme

Özellik çıkarımında amaç çok boyutlu veriyi varyansı maksimum tutacak şekilde daha düşük boyutlu hale getirmektir. Böylece verinin genel özellikleri korunarak boyut sayısının düşürülmesi yani verinin sıkıştırılması hedeflenmektedir. Temel bileşen analizi ve bağımsız bileşen analizi bu amaçla kullanılan iki farklı yöntemdir.

2.2.1 Temel bileşen analizi

PCA, bir değişken setini bileşenler arasındaki kovaryansı maksimum yapacak şekilde yeni bir örnek uzaya dönüştürmeye yarar. Oluşturulan yeni uzayda büyük varyans değerine sahip eksenlerin seçilmesi suretiyle veri setinin içerdiği bilginin yüksek orandaki bir kısmının daha az eksen ile ifade edilebilmesine olanak sağlar. Yaygın olarak veri boyutunu düşürmek, veriyi sıkıştırmak ve daha iyi yorumlayabilmek gibi amaçlar ile kullanılır. Sütunları gözlem değerlerini içeren bir X matrisini temel bileşenlerine ayırmak için her bir gözlem değerinin (vektör) birbirleri arasındaki ilişkiyi

ifade eden korelasyon matrisi kullanılır. C matrisi X matrisinin kovaryans matrisini, v_i öz vektörleri ve e_i ise öz değerlerini temsil eder **(Denk.1)**.

$$C v_i = v_i e_i \quad (1)$$

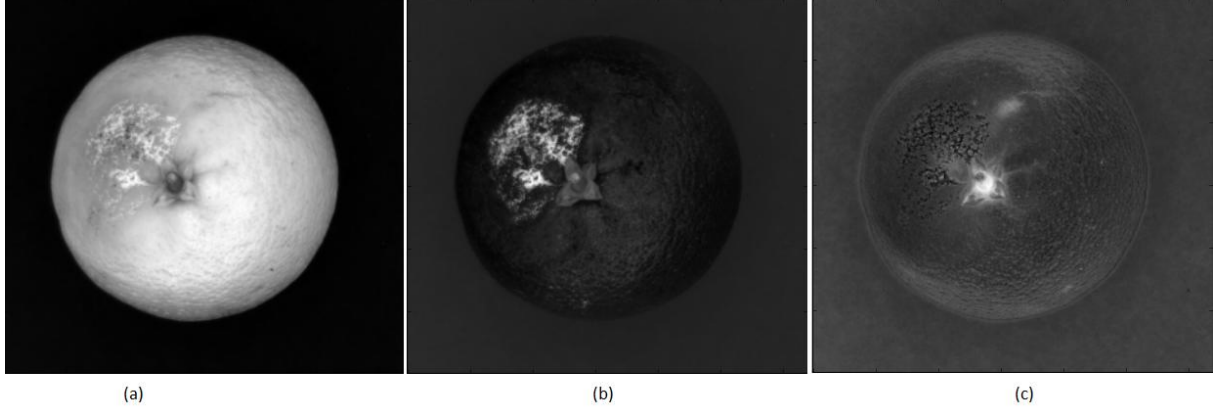
Eşitliği sağlayan ve C matrisini köşegen hale getiren dönüşüm ile C matrisinin sütunlarındaki veriler birbirleri arasında bağımsız hale getirilebilir. Kovaryans matrisi her zaman pozitif ve simetrik olduğundan köşegenleştirilebilir bir matristir. Öz vektörleri V matrisinin sütunları, öz değerleri ise E matrisinin köşegenleri olacak şekilde iki matris tanımlarsa aşağıdaki C matrisi elde edilebilir. **(Denk.2)**;

$$C = V E V^T \quad (2)$$

Dönüştürülmüş veriyi elde etmek için ise X matrisi öz vektörler matrisi olan V ile çarpılır **(Denk.3)**.

$$X_{yeni} = X V \quad (3)$$

Dönüştürülme işleminde sadece öz vektör matrisi kullanılmakta olup öz değerler kullanılmaz. Öz değerler aslında her bir öz vektörün varyansını yani veriyi hangi oranda temsil etme kabiliyetine sahip olduğunu gösterir. V öz vektörler matrisinin sütunları büyük öz değere karşılık gelen öz vektörden başlayacak şekilde düzenlendiğinde ve belirli bir eşik değerin altındaki öz değere karşılık gelen öz vektörler ihmal edildiğinde, veri boyutu çok fazla kayıp olmadan daha az sayıda vektör ile ifade edilebilir. Örneğin 33 bantlık RetigaEx görüntüsü PCA ile temel bileşenlerine ayrılırsa; ilk temel bileşen tüm verinin %98.441, 2. temel bileşen %1.20, 3. temel bileşen %0.166 'sını temsil eder (Şekil 2.4). Aynı şekilde Pavia görüntüsü temel bileşenlerine ayrıldığında; ilk temel bileşen tüm verinin %58.32, 2. temel bileşen %36.10, 3. temel bileşen %4.43 'ünü temsil eder (Şekil 2.5).



Şekil 2.4 : RetigaEx görüntülerinin temel bileşenleri. (a) 1. Temel bileşen (b) 2. Temel bileşen (c) 3. Temel bileşen



Şekil 2.5 : Pavia görüntülerinin temel bileşenleri. (a) 1. Temel bileşen (b) 2. Temel bileşen (c) 3. Temel bileşen

2.2.2 Bağımsız bileşen analizi

ICA normalde birbirine karışmış farklı sinyalleri ayırmak maksadıyla kullanılmaktadır. ICA yöntemi birden fazla sinyalden oluşan birleşik sinyali, birbirleri ile aralarında istatistiksel olarak bağımsız olan ve birden fazla bileşeni gauss dağılımı göstermeyecek

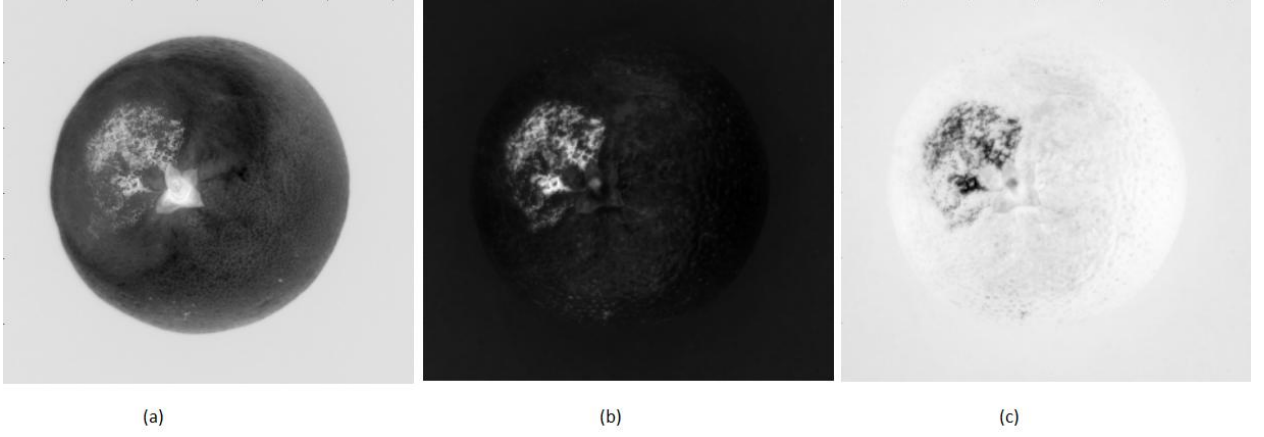
şekilde bileşenlere ayırmaya çalışır. Buna göre X gözlemlenen değerleri (HS görüntü), S ise bağımsız bileşen vektörlerinden oluşan matris olmak üzere, ICA dönüşümü şu şekilde tanımlanır **(Denk.4)**;

$$S_{m \times p} = W_{m \times n} X_{n \times p} \quad (4)$$

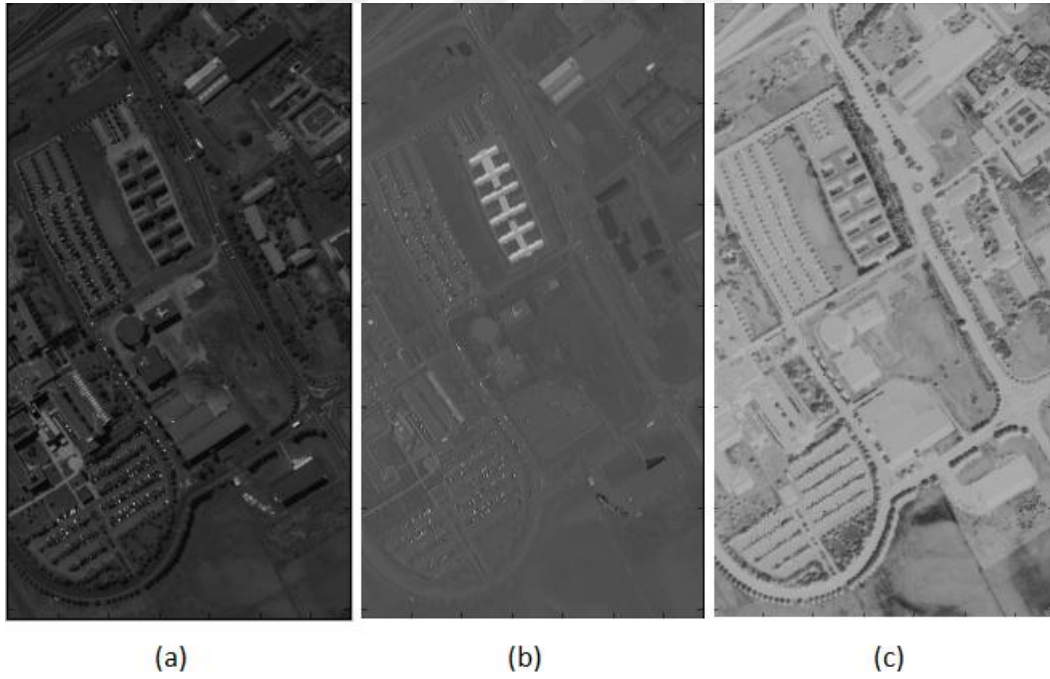
ICA dönüşümünde amaç X matrisini birbirinden bağımsız hale getirecek W dönüşüm matrisini bulmaktır. Şayet S matrisindeki bir S_i bileşeni diğer hiçbir bileşen hakkında bilgi içermiyorsa, S_i bileşeni bağımsız demektir. Buna göre S matrisini oluşturacak bağımsız bileşenlerin en az sayıda Gauss dağılımı gösteren bileşen içermesi amaçlanmaktadır. Yani ICA dönüşümde esas olan W dönüşüm matrisini bulmak ve anahtar olarak Gauss dağılımı göstermeme kriterini baz almaktır. Bunun için kullanılan ölçüm kriteri Kurtosis diye adlandırılır ve Gauss dağılımındaki basıklığı ifade eder. Gauss dağılımına sahip değişkenlerin entropisi yüksek olacağından, Kurtosis yerine negentropide W matrisini bulmak için bir kriter olarak kullanılabilir [24]. Negentropi şu şekilde tanımlanır **(Denk.5)**;

$$J(X) = H(X_{gauss}) - H(X) \quad (5)$$

X_{gauss} terimi X ile aynı kovaryans matrise sahip gauss dağılımına sahip rastgele bir değişkeni ifade etmektedir. Eğer X gauss dağılımına sahipse Negentropi değeri sıfıra eşittir. Buna göre, negentropy kriteri dikkate alınarak belirlenen amaç fonksiyonunu maksimize edecek dönüşüm matrisinin bulunması için Hyvarunen tarafından Fast-ICA algoritması geliştirilmiştir [25]. Bu algoritma HS veri küpüne uygulandığında ve görüntüler X matrisinin vektörleri olarak düşünüldüğünde, bağımsız bileşenlerine ayrılmış S matrisinin her bir vektörü birbirinden bağımsız görüntüler olacaktır. HS görüntüler için elde edilen bağımsız bileşenler şekil 2.6 ve şekil 2.7 'de gösterilmiştir.



Şekil 2.6 : Portakal görüntülerinin bağımsız bileşenleri. (a) 1. bileşen (b) 2. bileşen (c) 3. bileşen



Şekil 2.7 : Pavia Üniversitesi görüntülerinin bağımsız bileşenleri. (a) 1. bileşen (b) 2. bileşen (c) 3. bileşen

PCA ve ICA ile gerçekleştirilen dönüşümler sonucunda elde edilen görüntülerden de görüleceği gibi, dönüşüm temelli yöntemlerde çok boyutlu veri sadece 2 veya 3 boyut ile

yüksek oranda ifade edilebilmekte fakat hiperküpün sunduğu tüm spektral özellikler kaybolmaktadır. Bu sebeple, bu tip yöntemler fazla tercih edilmemektedir.

2.3 Bant Seçme ile Boyut İndirgeme

HS görüntülerde spektral bant seçme işleminde amaç tüm bantlar içerisinde en fazla bilgiyi içeren bandların bulunmasıdır. Bu kapsamda bandların (görüntülerin) birbirlerine benzerliklerini ifade eden bazı ölçütler bu başlık altında verilmiş, maksimum veri içeren alt bant kümesini aramak için kullanılan hiyerarşik kümeleme arama yöntemi ile bant seçme işlemi gerçekleştirilmiştir. Hiyerarşik kümelemeye ek olarak görüntünün yapısal özelliğinden bilgi çıkarılması için yeni bir yöntem önerilmiş, geliştirilen yöntemin performansı kullanılan veri setleri üzerinden test edilmiştir.

2.3.1 Benzerlik kriterleri

2.3.1.1 Spektral açı

İki görüntü vektör olarak ele alınıp, her ikisi arasındaki açı hesaplanır. Spektral açı (SA) $(\emptyset_{I_1, I_2})$ iki vektör $(I_1$ ve $I_2)$ arasındaki açı olarak şu şekilde tanımlanır **(Denk.6)**;

$$\emptyset_{v_1, v_2} = \arccos\left(\frac{I_1 \cdot I_2}{\|I_1\| \|I_2\|}\right) \quad (6)$$

İki vektör arasındaki açının düşük olması, bu iki vektörün birbirlerine benzer olduklarını ifade etmektedir.

2.3.1.2 Ortak bilgi

Ortak bilgi (mutual information-MI) rastgele değişkenlerin birbirleri arasındaki farklılığı ifade eden bir metriktir. HS görüntülerde, veri küpünün bantları arasındaki ilişkiyi yani bantların birbirlerinden ne kadar bağımsız olduklarını ifade etmek için kullanılır. MI değeri, görüntünün entropisi kullanılarak hesaplanır. Teorik olarak [26] $p(x)$ olasılık yoğunluk fonksiyonuna sahip X rastgele değişkeninin entropisi şu şekilde tanımlanır **(Denk.7)**;

$$H(x) = - \int p(x) \log p(x) dx \quad (7)$$

X 'in ayrık değişken olması durumunda entropi şu şekilde tanımlanır **(Denk.8)**;

$$H(x) = - \sum p(x) \log p(x) \quad (8)$$

Görüntüde entropi değerleri hesaplanırken $p(x)$ olasılık yoğunluk fonksiyonu için, görüntünün histogramı hesaplanarak elde edilen piksel olasılık değerleri kullanılır. Örneğin bir HS veri setinin farklı dalga boylarındaki görüntülerin entropileri Şekil 2.8'deki gibidir.



Şekil 2.8 : Farklı dalga boylarındaki görüntülerin entropileri

MI iki bağımsız değişkenin bağımsızlığını ya da bir değişkenden diğerinin ne kadar tahmin edilebileceğinin ölçütünü verir. $X_1, X_2, \dots, X_L$ 'nin bir HS görüntünün bantları olduğu düşünüldüğünde, herhangi iki görüntü arasındaki $MI(X_i, X_j)$ şu şekilde tanımlanır **(Denk.9)**;

$$MI(X_i, X_j) = \sum \sum p(x_i, x_j) \log \frac{p(x_i, x_j)}{p(x_i) p(x_j)} \quad (9)$$

MI daima pozitif olup iki değişken istatistiksel olarak bağımsız olduğunda 0 değerini alır. MI değerinin yüksek olması değişkenler arasındaki kuvvetli ilişkiyi ifade eder. İki değişken arasında denklem 10'daki ilişki daima geçerlidir **(Denk.10)**.

$$0 \leq MI(X_i, X_j) \leq \min\{H(X_i), H(X_j)\} \quad (10)$$

MI entropiye bağı olarak aşağıda verilen denklemdeki gibi ifade edilebilir (**Denk.11**).

$$MI(X_i, X_j) = H(X_i) + H(X_j) - H(X_i, X_j) \quad (11)$$

$H(X_i, X_j)$ iki değişkenin ortak entropisini ifade eder ve ortak olasılık dağılımı $p(x_i, x_j)$ ile hesaplanır. Görüntüde ortak olasılık dağılımının hesaplanması için iki görüntünün ortak histogramının hesaplanması gerekmektedir. MI değerinin bir karşılaştırma kriteri olarak kullanılabilmesi için normalize edilmesi daha güvenli olacaktır. Normalize edilmiş MI aşağıda verilen (**Denk.12**) eşitlik ile hesaplanır ve HS görüntülerde bantların birbirleri ile ilişkilerini tanımlamada kullanılır [9].

$$NMI(X_i, X_j) = \frac{2 * MI(X_i, X_j)}{H(X_i) + H(X_j)} \quad (12)$$

2.3.1.3 Kullback-Leibler ayrıştırma yöntemi

Bir başka benzerlik kriteri olarak Kullback-Leibler ayrıştırma (KLD) ölçütü de kullanılabilir. KLD iki farklı olasılık dağılımının birbirlerine benzerlik mesafesini verir ve X_i ve X_j görüntüleri için aşağıdaki gibi tanımlanır (**Denk.13**).

$$KLD(X_i, X_j) = \sum p(x_i) \log \frac{p(x_i)}{p(x_j)} \quad (13)$$

Denklemdeki $p(x)$ olasılık yoğunluk (kütle) fonksiyonunu ifade etmektedir. KLD simetrik olmadığından dolayı simetrik versiyonunun kullanılması tercih edilmektedir (**Denk.14**).

$$KLD(X_i, X_j) = \sum p(x_i) \log \frac{p(x_i)}{p(x_j)} + \sum p(x_j) \log \frac{p(x_j)}{p(x_i)} \quad (14)$$

2.3.1.4 Yapısal benzerlik

Yapısal benzerlik (SSIM-Structural Similarity) değeri, iki görüntünün benzerlik oranını veren bir ölçüttür [27]. SSIM ölçütü, insan gözünün algılama yapısına uygun olmayan bir benzerlik ölçütü olan ortalama karesel hata (MSE) yöntemine nazaran yapısal

benzerlikleri daha iyi veren bir metrik olup aşağıdaki gibi tanımlanır. Formülde μ piksel ortalamalarını, σ standart sapmayı, c_1 ve c_2 ise sabit sayıları ifade etmektedir (**Denk.15**).

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (15)$$

2.3.1.5 Standardize edilmiş öklit mesafesi

Öklit mesafesi iki nokta arasındaki doğrusal uzaklıktır ve n boyutlu uzay için her bir noktanın ayrı ayrı farklarının hipotenüsüne eşittir. HS görüntüde her bir bandı bir vektör olarak ele alırsak, öklit mesafesi görüntülerin birbirlerine ne kadar yakın olduğunu ölçen bir kıstas olarak kullanılabilir. Ayrıca her iki görüntü vektörü tek bir örnek kümesi gibi düşünülüp (s_i) öklit mesafesi standardize edilebilir. İki görüntü arasında benzerlik kriteri olarak kullanılacak olan standardize edilmiş öklit mesafesi (*Secludian*) şu şekilde hesaplanır (**Denk.16**);

$$Secludian(x, y) = \sqrt{\sum \frac{(x_i - y_i)^2}{s_i^2}} \quad (16)$$

Denklem 16'da x ve y ayrı görüntüleri, x_i görüntüdeki her bir pikseli, s_i ise x_i ve y_i arasındaki öklit mesafesini ifade etmektedir.

2.3.2 Önerilen anahtar noktalar temelli bant seçme algoritmaları

Tez kapsamında önerilen bant seçme algoritmasında alt bantları arama stratejisi olarak hiyerarşik kümeleme yöntemi kullanılmıştır. Bu yöntemde ilk olarak bantlar arasındaki benzerliği ölçen kıstas baz alınarak kümeleme işlemi gerçekleştirilir. Böylelikle seçilen kıstasa göre birbirlerine benzer olan görüntüler aynı grup içerisinde toplanmış olacaktır. Bu yöntem baz alınarak geliştirilen ve iyi sonuç veren bir bant seçme algoritmasında [9], alt bantlar arasındaki varyansı maksimum ve bantların kendi içerisindeki varyansı ise minimum yapan *Ward* birleştirme stratejisi kullanılmıştır. Hiyerarşik kümeleme ile alttan üste doğru birbirlerine benzer olan düğüm ve onun yaprakları bir ağaç yapısı şeklinde ifade edilir. Ward stratejisi kullanılarak gerçekleştirilen bu çalışmada mesafe/benzerlik ölçütleri olarak MI ve KLD ölçütleri kullanılarak birbirleri ile benzer olan spektral bantlar gruplanmıştır. Bantlar kümelere ayrıldıktan sonra ise her kümeden

o kümeyi en iyi temsil eden bir eleman (bant) seçilmesi suretiyle band seçme işlemi gerçekleştirilmiştir. Bu yöntem literatürde Walumi algoritması olarak adlandırılmaktadır [9]. Literatürde geliştirilen band seçme algoritmalarının performans karşılaştırılmasında Walumi algoritması yaygın olarak kullanılmaktadır. Walumi algoritmasında aynı küme içerisinde o kümeyi temsil edecek elemanı seçmek için, küme içerisindeki eleman sayısı N olmak üzere, aynı kümede (C) bulunan her bir görüntüye bir ağırlık katsayısı atanır (W_i) ve şu şekilde hesaplanır (**Denk.17**);

$$W_i = \frac{1}{N} \sum \frac{1}{\epsilon + D(i,j)^2} \quad , j \in C \text{ ve } j \neq i \quad (17)$$

$D(i,j)$, iki bant arasındaki mesafeyi ifade eder. Atanan W katsayısına göre her bir küme içerisinde en yüksek katsayı değerine sahip olan görüntü seçilir. Katsayısı düşük olan görüntü diğer görüntülere ortalama bir mesafede yani diğerleri ile arasında düşük bir korelasyon var demektir. Tersisi durumda, katsayısı büyük olan görüntü ise küme içerisindeki diğer görüntüler ile daha fazla benzer olduğu ve o kümeyi temsil etmek için daha yeterli olduğu düşünülür.

Bu tez çalışmasında önerilen algoritmada ise, bantlar alt kümelere ayrıldıktan sonra her kümeyi temsil eden bir görüntü seçmek için görüntünün yapısal özelliklerinin kullanıldığı yeni bir algoritma önerilmiş ve literatürde kabul görmüş WaluMI ve WaluDI band seçme algoritmaları ile performans kıyaslaması gerçekleştirilmiştir.

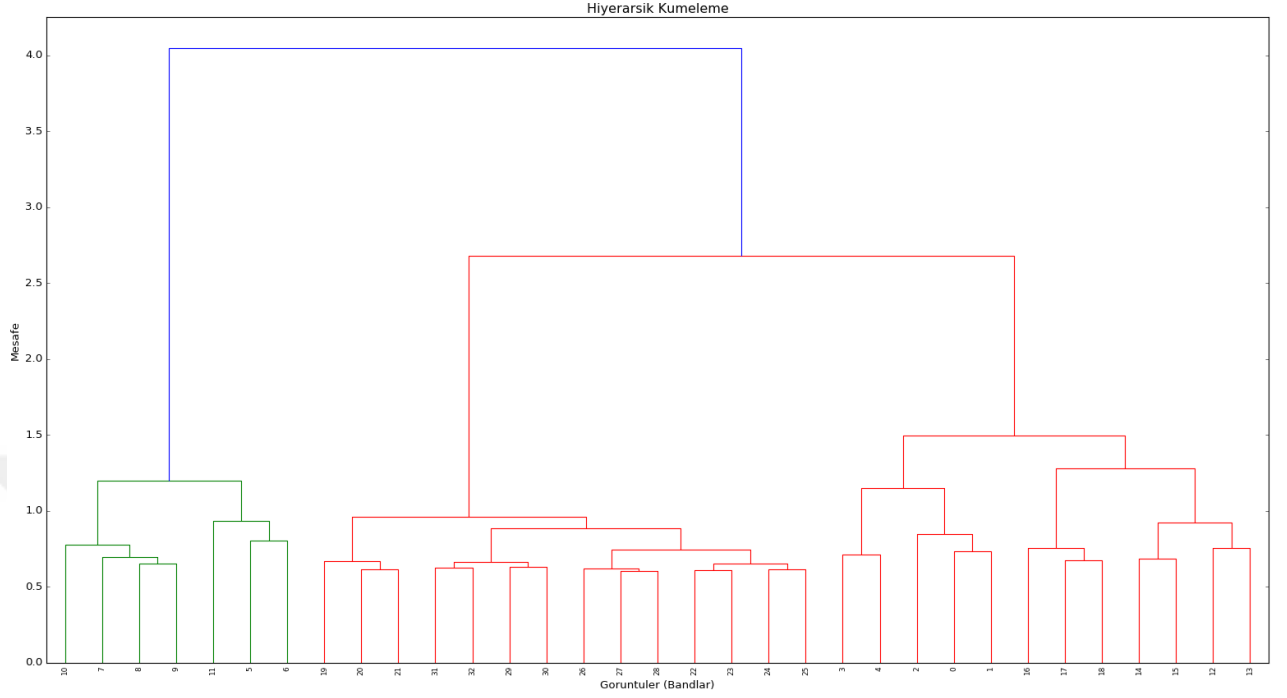
2.3.2.1 Hiyerarşik kümeleme ile kümelerin oluşturulması

Hiyerarşik kümeleme için ilk olarak her bir bant arasındaki mesafenin (benzerliğin) hesaplanması gerekmektedir. Bu kapsamda, bir önceki alt başlıkta anlatılan benzerlik kriterlerine göre ayrı ayrı değerlendirmeler yapılmıştır. Daha sonraki aşamada ise her iki bant bir kümeyi, daha üst seviyede ise birden fazla küme bir üst seviyedeki yeni bir kümeyi oluşturacak şekilde bir ağaç yapısı oluşturulur. Hiyerarşik olarak kümelerin birleştirilmesinde kullanılacak strateji son derece önemlidir ve tercih edilebilecek farklı yöntemler mevcuttur. Seçilecek birleştirme yöntemine göre oluşturulacak kümeler ile birbirinden çok farklı ağaç yapıları ortaya çıkacaktır. Hiyerarşik kümelemede kullanılan birleştirme stratejileri Çizelge 2.1’de listelenmiştir.

Çizelge 2.1 : Hiyerarşik kümelemede kullanılan bazı birleştirme yöntemleri

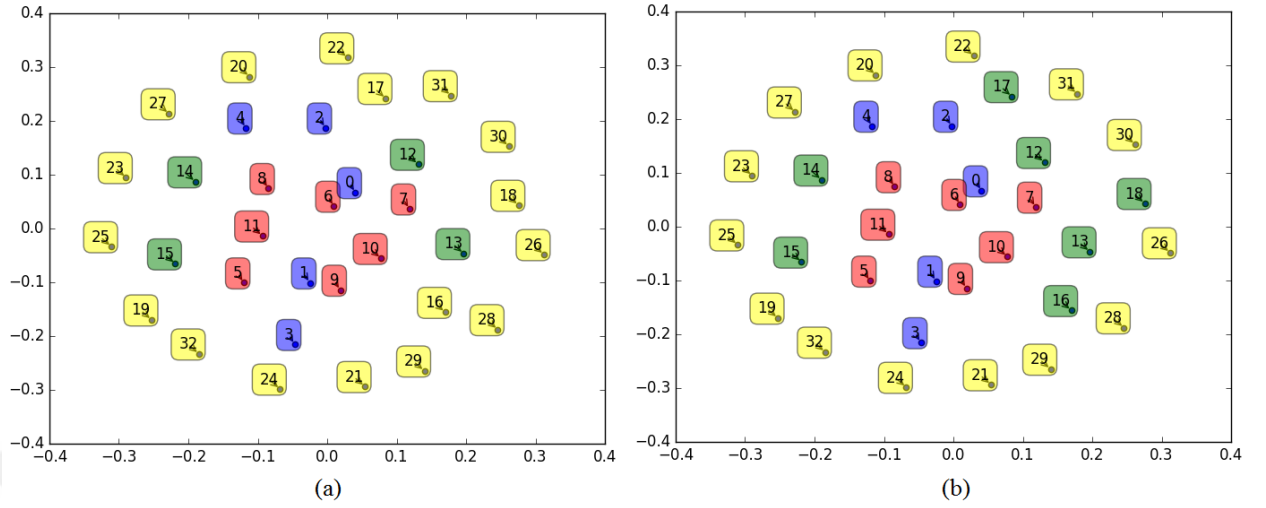
Metot	Mesafe fonksiyonu
‘Single’	$d(u, v) = \min_{i,j} (dist(u[i], v[j]))$
‘Complete’	$d(u, v) = \max_{i,j} (dist(u[i], v[j]))$
‘Average’	$d(u, v) = \frac{\sum_{ij} dist(u[i], v[j])}{(u * v)}$
‘Centroid’	$d(u, v) = \ c_u - c_v\ $
‘Ward’	$d(u, v) = \sqrt{\frac{ v + s }{T} * d(v, s)^2 + \frac{ v + t }{T} * d(v, t)^2 - \frac{ v }{T} * d(s, t)^2}$

Single stratejisi ile her iki küme içerisinde birbirlerine en yakın elemanlar arasındaki uzaklık, kümeler arasındaki uzaklık olarak hesaplanır. Bunun tam tersi olan *complete* yönteminde ise iki küme arasındaki uzaklık olarak birbirlerine maksimum uzaklıktaki elemanların uzaklığı seçilir. *Average* yönteminde, her iki küme içerisindeki tüm elemanların birbirleri ile olan uzaklıklarının ortalaması alınır. *Centroid* yönteminde her kümenin ağırlık merkezi (orta noktası) hesaplanarak, orta noktalar arasındaki mesafe kümeler arası mesafe olarak kabul edilir. Ward stratejisi [16] olarak adlandırılan yöntemde ise, s ve t elemanlarına sahip u kümesi ile v kümesi arasındaki mesafeyi veren fonksiyon çizelge 2.1’de gösterilmiştir. Ward yönteminde ise $T = |v| + |s| + |t|$ eşitliği ile hesaplanır. Çizelgede hesaplanan ward mesafesi için; u kümesinin s ve t den oluşan bir küme, v ise tek başına bir küme olduğu düşünülür. Bu yöntemin anılan diğer yöntemlerden daha iyi performans gösterdiği literatürde gerçekleştirilen çalışmada ortaya konulmuş [17] ve bant seçme problemindeki başarısı da Walumi algoritmasında gösterilmiştir [9]. Buna göre, MI benzerlik ölçütü kullanılarak Ward hiyerarşik kümeleme yöntemi ile oluşturulmuş RetigaEx görüntüsünün ağaç yapısı Şekil 2.9’da gösterilmiştir. Yatay eksen aynı küme içerisindeki görüntüleri, dikey eksen ise kümeler arasındaki mesafeyi ifade etmektedir.



Şekil 2.9 : Hiyerarşik kümeleme ile oluşturulan ağaç yapısı

Oluşturulan bu ağaç yapısından kümelerin birbirleri ile aralarındaki mesafelere bakılarak istenilen sayıda küme oluşturulabilir. İstenilen sayıda farklı küme elde edilebilmesi için birbirlerine yakın kümeler tek bir küme olarak düşünülür. Buna göre; dört küme oluşturulması istendiğinde yatay eksendeki uzaklık mesafeleri dikkate alınarak farklı birleştirme stratejileri ile seçilmiş kümeler 2 boyutlu düzlemde şekil 2.10'da gösterilmiştir. Çok boyutlu veriyi iki boyut üzerinde gösterebilmek için, bir çeşit boyut indirgeme yöntemi olan çok boyutlu ölçeklendirme yöntemi kullanılmıştır. Şekil 2.10'da verilen grafiklerdeki eksenler, çok boyutlu ölçeklendirme neticesinde elde edilen 2 boyuta ait eksenlerdir.

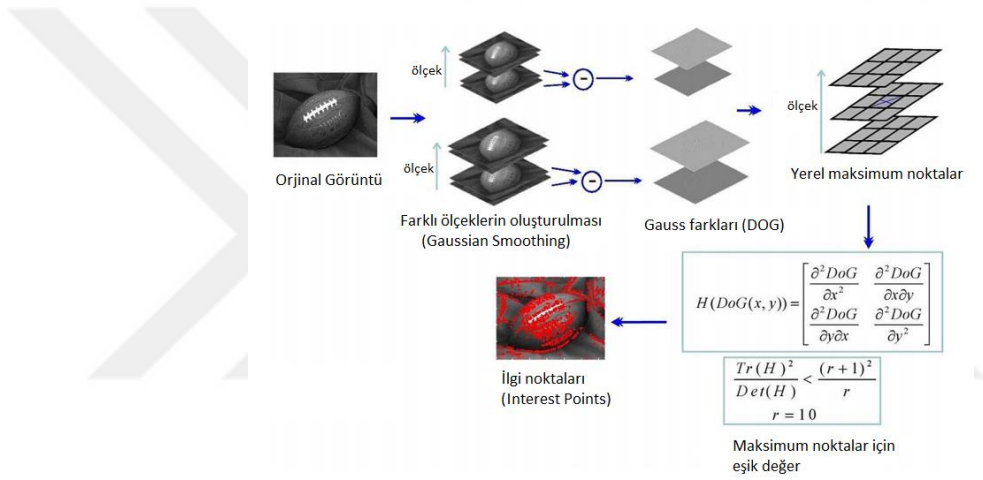


Şekil 2.10 : Hiyerarşik kümeleme sonucu elde edilen kümelerin 2 boyutta gösterimi (a) Average (b) Ward

Tez kapsamında önerilen bant seçme algoritmasında oluşturulan her bir kümeden o kümeyi temsil edecek elemanı seçmek için bir algoritma önerilmiş, böylelikle hiyerarşik kümeleme temelli yeni bir bant seçme algoritması geliştirilmiştir. Kümeleme sonucunda her kümede seçilen benzerlik kıstasına göre birbirine yakın görüntüler toplandığından dolayı, her küme içerisinde o kümeyi temsilen en fazla bilgi içeren bandın seçilmesi amaçlanmıştır. Literatürde kullanılan benzerlik kıstalarında genel olarak görüntülerin istatistiksel olarak birbirlerinden ne kadar farklı oldukları üzerine durulmuştur. Önerilen yöntemde ise görüntünün içeriği ile ilgili bilginin kullanılmasının faydalı olabileceği düşünülmüştür. Bu kapsamda ilk olarak bantlar birbirlerine benzerliklerine göre kümelere ayrılmış, ardından her bir küme içerisinde en fazla bilgi içeren bant seçilmiştir. Her bir bandın ne kadar bilgi içerdiğini ölçmek için ise *anahtar noktalar* (*interest point - IP*) temelli bir bant seçme önerilmiştir. Ayrıca bir görüntüdeki farklı bandların birbirlerine ne kadar benzediklerini ölçmek için IP temelli bir benzerlik ölçütü tanımlanmış ve bir başka bant seçme yöntemi olarak değerlendirilmiştir.

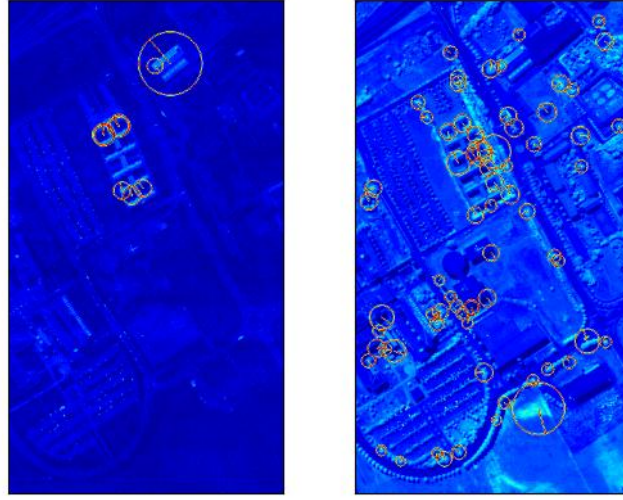
Bir görüntünün anahtar noktası olarak, görüntüde göze çarpan kenar, köşe ya da leke gibi ani renk değişimlerinin olduğu yerler düşünülür. Anahtar noktaları genelde görüntü eşleme ve 3 boyutlu görüntü oluşturmada farklı açılardan çekilmiş görüntülerdeki aynı noktaları bulmak için kullanılır. Anahtar noktaları bulmak için yaygın olarak kullanılan

algoritma Lowe tarafından önerilmiş SIFT (Scale invariant feature transform) algoritmasıdır [28]. Bu algoritmanın genel akış şeması şekil 2.11’de verilmiştir. İlk olarak görüntünün farklı ölçeklerde yerel maksimum değerleri hesaplanır. Bunun için farklı Gauss parametrelerine göre bulanıklaştırılmış görüntülerin farkları (DOG – difference of Gaussians) alınır. DOG işleminden sonra her bir ölçekte pikseller komşuları ile karşılaştırılarak yerel maksimum noktalar yani potansiyel IP ‘ler bulunur. Taylor serisi kullanılarak her bir ölçekteki potansiyel maksimum adaylarından güçlü IP ‘ler seçilir.



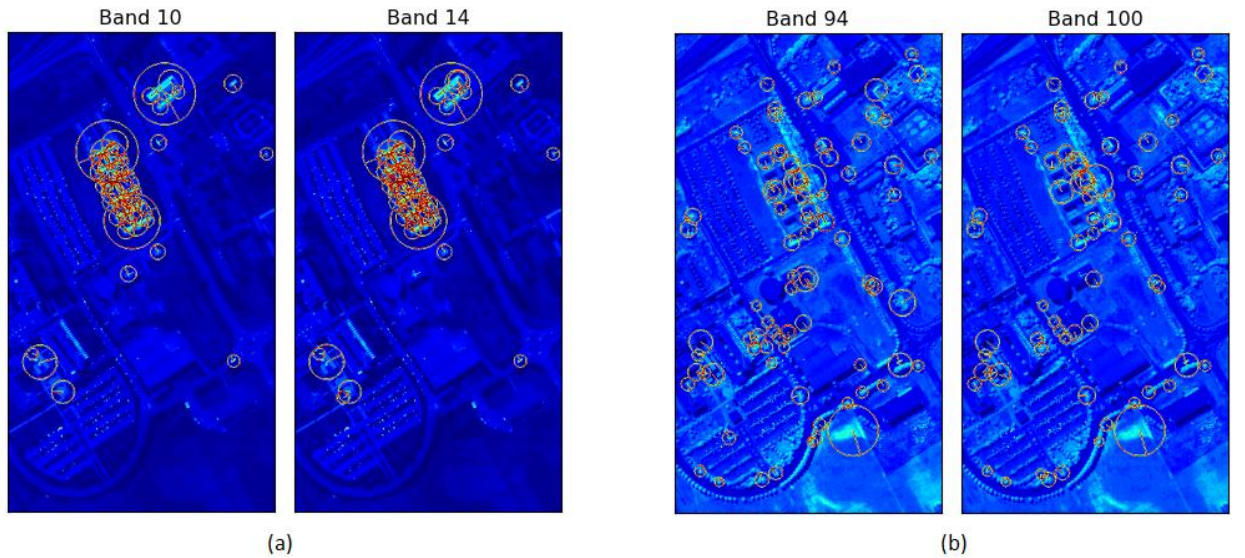
Şekil 2.11 : SIFT ile anahtar noktaların bulunması [29]

Anahtar noktalarına HS görüntü özelinde baktığımızda ise, aynı yerin farklı dalga boylarındaki çok sayıdaki görüntüsünün her birisinde farklı materyallerin/nesnelerin algılanma hassasiyeti birbirinden farklı olacağından dolayı, her bir bandaki IP birbirinden farklı sayıda ve konumlarda olacaktır. Örneğin birbirlerine uzak iki farklı dalga boyunda elde edilmiş görüntüler üzerinde bulunan IP Şekil 2.12 ‘deki gibidir.

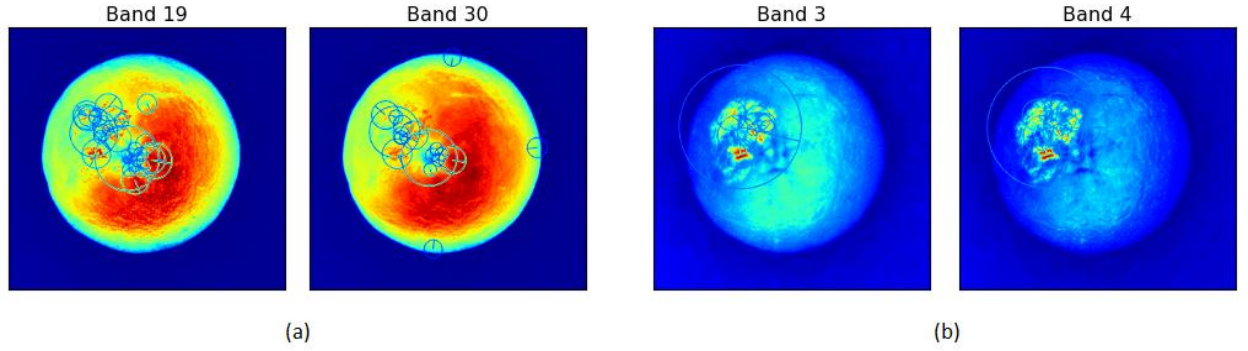


Şekil 2.12 : Pavia üniversitesi görüntüsünün ilk ve son bandındaki IP

Şekil 2.12’de görüldüğü gibi birbirlerine uzak farklı dalga boylarındaki görüntülerde bulunan IP farklılık arz etmektedir. Aynı küme içerisinde yer alan görüntüler üzerinde analiz yapıldığında ise birbirlerine benzer noktalarda IP bulunmakla birlikte yine bazı farklılıklar göze çarpmaktadır. Örneğin Şekil 2.13 ve şekil 2.14 incelendiğinde aynı küme içerisindeki görüntülere bakılırsa; IP’nin daha fazla yoğunlukta bulunduğu görüntülerin daha fazla bilgi içerdiği sonucuna varılabilir. Buna göre; her bir bandın ne kadar bilgi içerdiğini ölçmek için IP temelli bir ölçüt/özellik tanımlanmıştır.



Şekil 2.13 : Pavia farklı iki kümede yer alan bandların IP çıkarılmış görüntüleri



Şekil 2.14 : RetigaEx farklı iki kümede yer alan bandların IP çıkarılmış görüntüleri

İlgi noktaları bulunduktan sonra amaç, her bir görüntünün anahtar nokta yoğunluğunu veren bir katsayı hesaplamaktır. Buna göre; her bir görüntü $N \times N$ 'lik pencerelere bölünerek, koordinat olarak her bir pencere içerisine giren IP sayıları belirlenir. I değişkeninin görüntü matrisini, $I_{i,j}$ değişkeninin görüntünün eşit parçalara ayrılmış bölümlerini ve B matrisinin ise (x,y) koordinatlarına göre IP değerlerini temsil ettiği düşünüldüğünde, görüntünün her bir bölmesi içerisindeki anahtar nokta sayısı şu şekilde hesaplanabilir (**Denk.18**);

$$A(i,j) = \sum B(x,y), \quad x,y \in I_{i,j} \quad (18)$$

A matrisi bölmelere ayrılmış görüntünün her bir bölmesindeki anahtar nokta sayısını veren $N \times N$ 'lik bir matristir. Bu matris kullanılarak her bir görüntünün ne kadar bilgi içerdiğine dair kullanılacak ölçütler aşağıdaki gibi elde edilir (**Denk.19**).

$$\alpha_k = std(X) \quad (19)$$

$$\delta_k = \left\{ \frac{length(Y)}{N * N} \right.$$

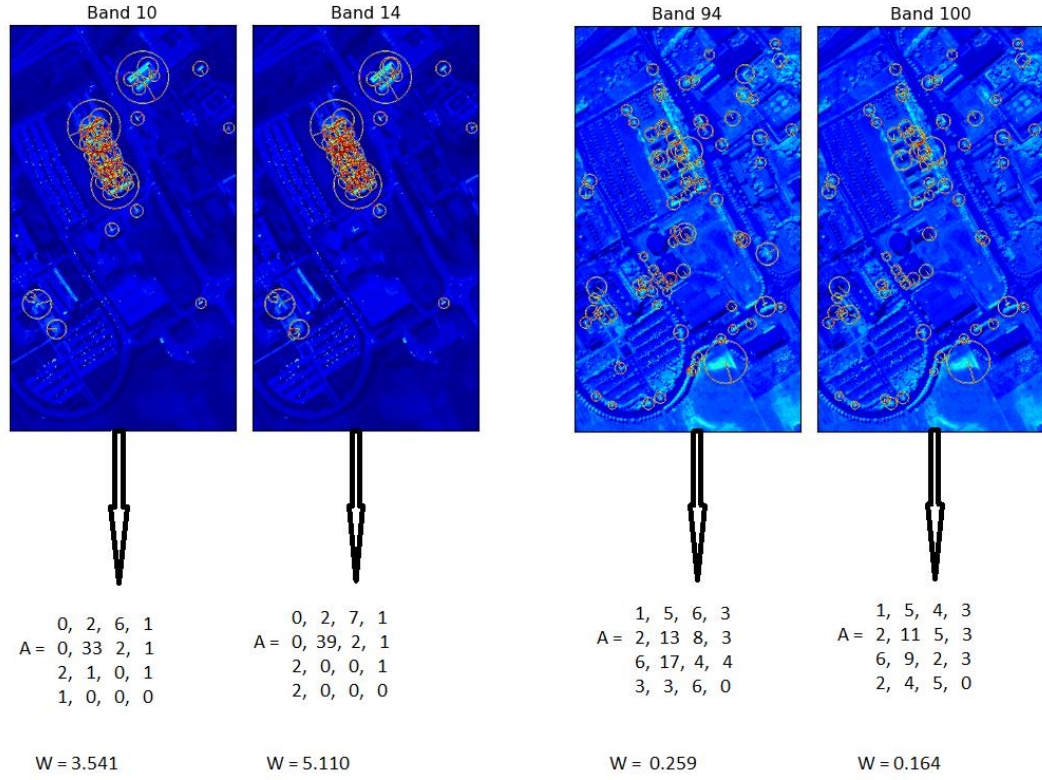
X ve Y sırasıyla A matrisindeki sıfır olmayan ve sıfır olan değerler olmak üzere; α_k görüntüdeki IP'ye göre oluşturulmuş A matrisinin sıfırdan farklı değerlerinin standart sapmasını verir. Kısacası, IP'nin görüntü üzerine nasıl dağıldığına ilişkin bilgiyi içerir. δ_k ise bölmelere ayrılmış görüntüde ne kadar boş bölme olduğunu yani IP'nin ne kadar fazla bir araya toplandığı ile ilgili bilgiyi içermektedir. Görüntünün tek bir bölmesinde

IP bulunması durumunda standart sapma yerine toplam anahtar nokta sayısı göz önüne alınmıştır. Hesaplanan bu iki ölçüt kullanılarak görüntüdeki her bir spektral bandın ne kadar bilgi içerdiğine dair katsayı (W_k) aşağıdaki gibi hesaplanır **(Denk.20)**.

$$W_k = \alpha_k * \delta_k \quad (20)$$

Hesaplanan iki ölçütün çarpımı ile IP'nın görüntü üzerinde ne kadar yoğun bulunduğu dair bir özellik çıkarılmış olur. Hesaplanan bu ağırlık katsayıları aynı kümedeki görüntüler için küçükten büyüğe doğru sıralandığında, en büyük katsayıya sahip bant (görüntü) o kümeyi temsil eden bant olarak seçilir. Bu adımlar uygulanarak IP temelli eğiticişiz bir bant seçme algoritması (IPBS) önerilmiştir.

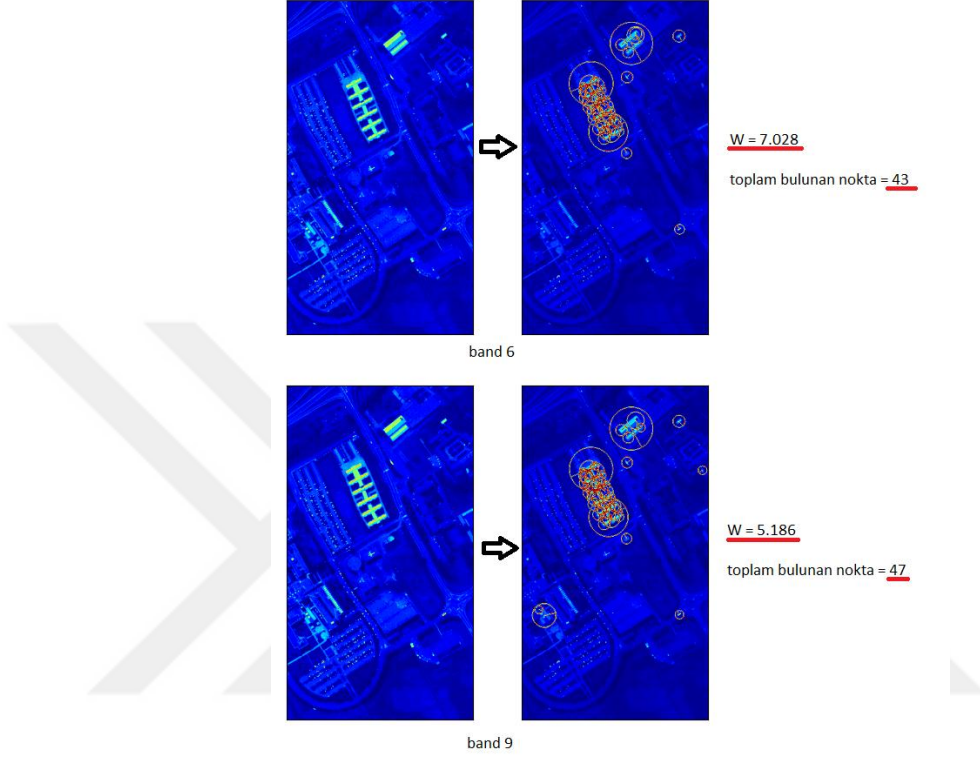
Örneğin; Pavia görüntüsünün farklı kümelerindeki görüntülerin özellik katsayıları $N=4$ için bulunursa band 14'ün band 10'a nazaran daha fazla bilgi içerdiği görülecektir. Yine hakeza band 94 ve band 100 için IP bulunup özellik değeri hesaplandığında, band 94'ün daha fazla bilgi içerdiği görülecektir (Şekil 2.15). Aynı şekilde portakal görüntüsü için özellik katsayıları hesaplanmıştır. Bu görüntü uzak mesafe görüntüsü olmayıp daha sade olduğu için bölmelere ayrılarak incelenmesine gerek yoktur.



Şekil 2.15 : Pavia verisi için farklı iki kümede yer alan bantların bilgilerini ölçen ağırlık katsayıları

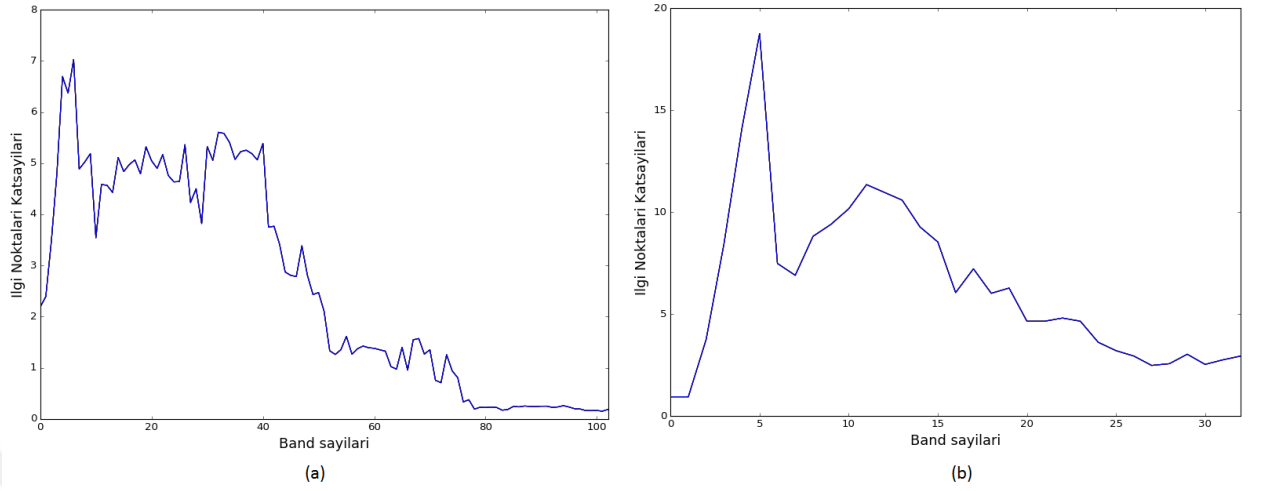
Bu sonuçlara bakıldığında genelde zaten toplam anahtar nokta sayısının fazla olduğu görüntülerin seçildiği görülmüştür. Fakat sonuçlar her zaman böyle olmayabilir. Toplam anahtar nokta sayısından ziyade IP belirli bölgelerde yoğunlaşmış görüntülerin seçilmesinin, yapılan gözlemler sonucunda daha doğru sonuç verdiği gözlemlenmiştir. Bunun sebebi görüntüler kümelere ayrılırken her bir küme içerisinde tam olarak benzer noktalara (nesnelere) daha hassas görüntülerin toplanamamasıdır. Böyle bir kümeleme işleminin muhteşem şekilde yapılabilmesi hem görüntünün içeriği hem de oluşturulacak küme sayısına göre son derece farklılıklar gösterecektir. Bu sebeple her bir kümeden IP'nin bir bölgeye olan yoğunlaşmasına bakılarak seçilecek bir eleman daha ayırt edici olacaktır. Buna göre aynı grupta yer alan 6. ve 9. bant görüntülerinin kıyaslaması şekil 2.16 'da verilmiştir. Toplam anahtar nokta sayısı daha az olmasına rağmen bant 6'nın ağırlık katsayısı daha yüksek olacaktır. Tüm kümeler üzerinde toplam IP sayısı ile

belirlenen yoğunluk katsayısına göre bir kıyaslama yapıldığında önerilen içerik bilgisi hesaplama özelliğinin daha başarılı olduğu gözlemlenmiştir.

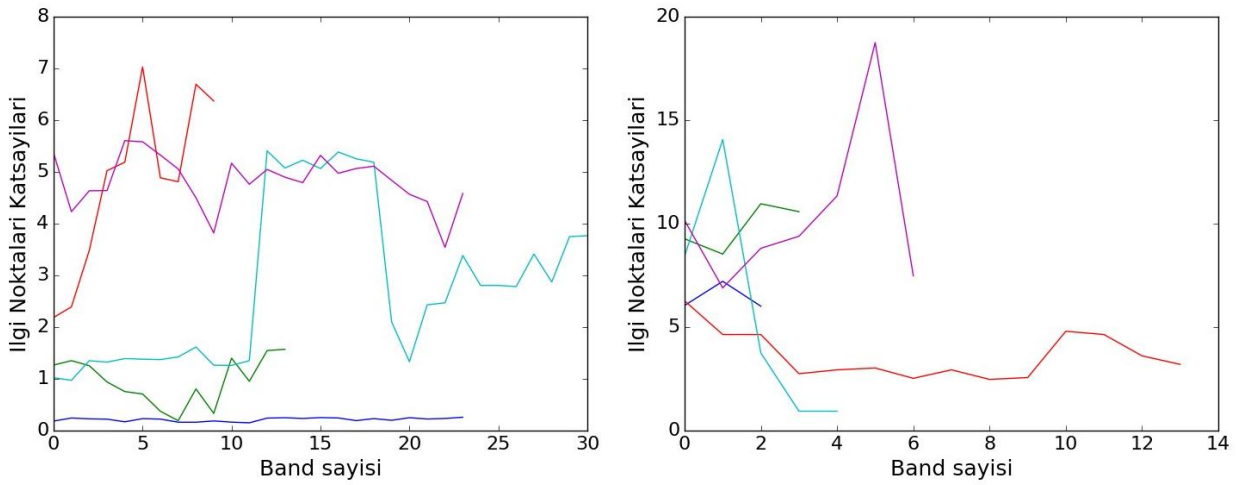


Şekil 2.16 : Pavia aynı kümede yer alan bandların özellik katsayıları

Her iki veri setinin tüm bandlarına ait hesaplanan IP özellik katsayılarını gösteren grafik şekil 2.17’de verilmiştir. Buna göre her bir küme içerisinde en yüksek ağırlık katsayısına sahip görüntü ilgili kümeyi temsil etmek üzere seçilecektir. Toplam grup sayısı 5 olacak şekilde alt kümeler seçilirse, her bir alt küme içerisindeki eleman sayısı birbirinden farklı olacaktır. Oluşturulan bu her bir küme içerisindeki bantların ağırlık katsayıları ise Şekil 2.18 ‘de gösterilmiştir.



Şekil 2.17 : Veri kümelerindeki tüm bantların IP özellik katsayıları (a) Pavia (b) RetigaEx

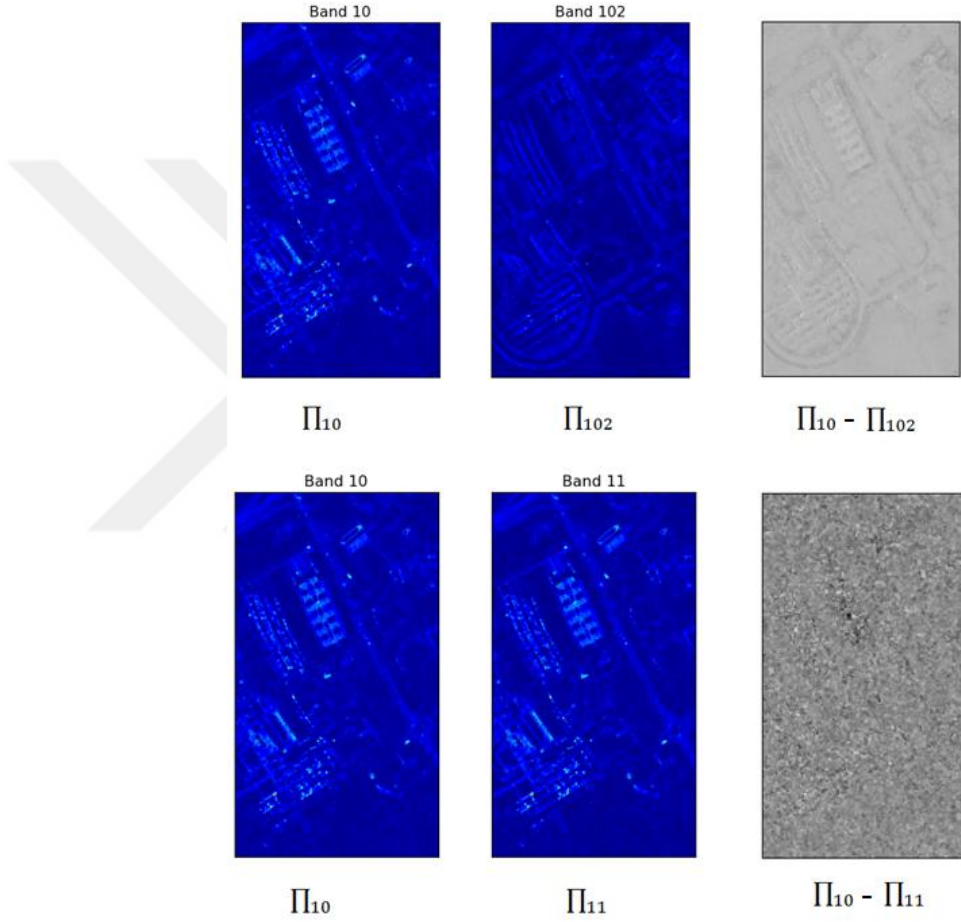


Şekil 2.18 : Beş küme için her iki veri setindeki kümelerin ağırlık katsayıları (a) Pavia (b) RetigaEx

Her bir küme içerisinde maksimum ağırlığa sahip elemanlar seçildiğinde, RetigaEx için seçilen bantların 4,5,12,19,17; Pavia veri seti için seçilen bantların ise 6,32,34,68,94 olduğu görülmektedir.

Önerilen IPBS algoritmasının dışında IP temelli yeni bir benzerlik ölçütü tanımlanmış ve yine hiyerarşik kümeleme ile birlikte bant seçmek için kullanılmıştır. Önerilen benzerlik ölçütünde iki görüntünün birbirine ne kadar benzer olduğunu ifade etmek için diğer

benzerlik ölçütlerinden farklı olarak görüntünün yapısal özellikleri dikkate alınmıştır. Bunun için ilk olarak karşılaştırılacak her iki görüntünün yatay ve dikey yönlerdeki türevleri alınarak kenarları çıkarılmış ve morfolojik kapama işlemi uygulanmıştır. Örneğin; Pavia verisinin iki farklı bandına türev ve morfolojik işlem uygulandığında elde edilen görüntüler şekil 2.19’da gösterilmiştir.



Şekil 2.19 : Filtrelenmiş ve morfolojik işleme tabi tutulmuş görüntüler ile fark görüntüsü

Görüntünün türevlerini almak için Laplacian filtresi kullanılmış ve bu işlemin ardından her iki görüntü birbirinden çıkarılarak oluşan fark görüntüsündeki IP sayısı bulunmuştur. Fark görüntüsünde bulunan IP sayısı (*IPDM – Interest point distance measure*) ile görüntülerin yapısal benzerliğini ölçen yeni bir mesafe ölçütü tanımlanmıştır. X_i ve X_j

iki farklı görüntü, IP 'yi bulan fonksiyon ve Δ_c ise morfolojik kapama işlemi olmak üzere, iki görüntünün benzerlik ölçütü $IPDM(X_i, X_j)$ şu şekilde tanımlanır **(Denk.21)**;

$$\Pi_i = \Delta_c(Laplacian(X_i))$$

$$\Pi_j = \Delta_c(Laplacian(X_j)) \quad (21)$$

$$IPDM(X_i, X_j) = IP(\Pi_i - \Pi_j)$$

Tanımlanan $IPDM$ ölçütü ile bantlar yine hiyerarşik kümelere ayrıldıktan sonra her kümeden o kümeyi en iyi temsil eden bir eleman (görüntü) seçilmesi suretiyle bant seçme işlemi gerçekleştirilmiş olur. Aynı küme içerisinde o kümeyi temsil edecek elemanı seçmek için aynı kümede (C) bulunan her bir görüntünün diğer görüntülere olan uzaklığı hesaplanarak, ilgili kümeyi temsil etmek için kümedeki diğer görüntülere en yakın olan görüntü seçilir **(Denk.22)**.

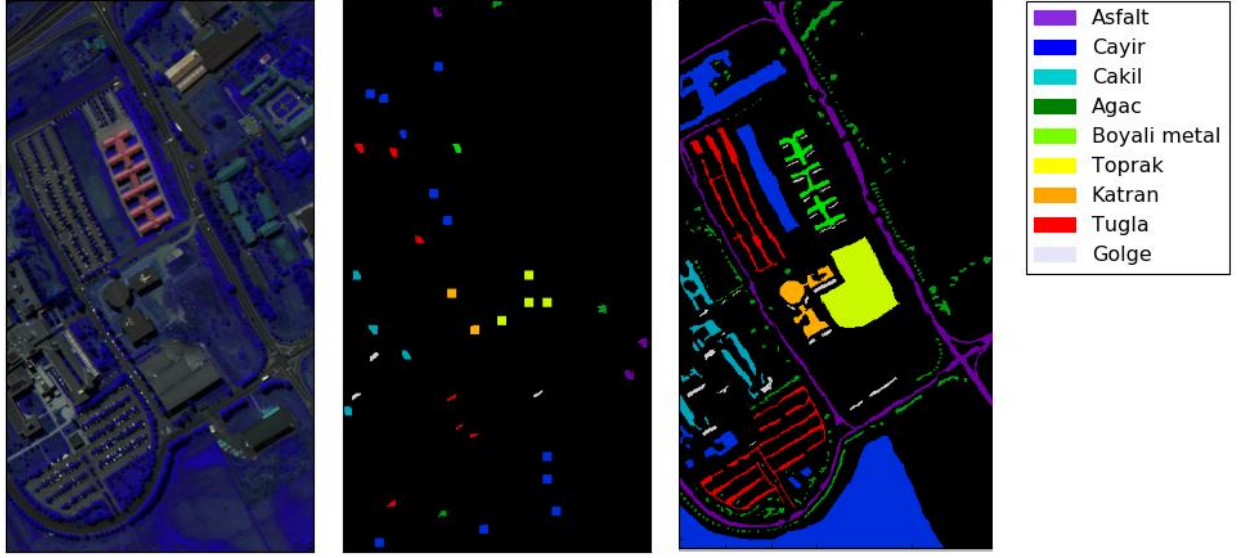
$$W_i = \frac{1}{N} \sum IPDM(i, j) \quad , j \in C \text{ ve } j \neq i \quad (22)$$

Böylelikle görüntünün yapısal özelliklerinin kullanıldığı yeni bir benzerlik ölçütü tanımlanarak ikinci bir bant seçme algoritması önerilmiştir. Önerilen bant seçme algoritmaları IPBS ve IPDM'e ait performans analizleri, sonuçlar kısmında sınıflandırma problemi üzerine farklı algoritmalar ile test edilerek karşılaştırılmıştır.

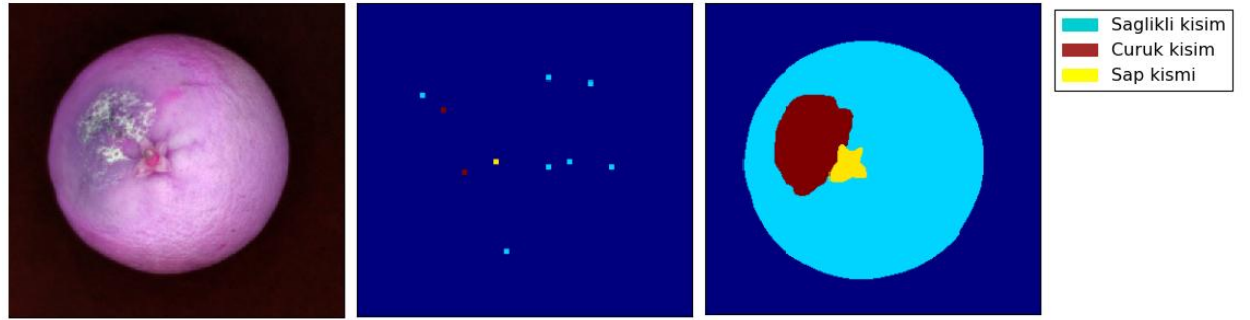
2.4 Sonuçlar

Bant seçme algoritmalarının performanslarının kıyaslanması için *k-en yakın komşu algoritması (k-NN)* kullanılmıştır. k-NN makine öğrenmesinde kullanılan en basit sınıflandırma algoritmalarından birisidir [30]. k-NN eğitici bir algoritmadır ve eğitim setine ihtiyaç duyar. Bu algoritmaya göre sınıflandırılmak istenen özellik vektörüne eğitim seti içerisinde en yakın k eleman bulunur. Bulunan k elemanın hangi sınıflara ait olduğuna bakılarak, özellik vektörünün hangi sınıfa daha çok yakın olduğuna karar verilip ilgili sınıfa ataması yapılır.

Sınıflandırma için eğitim verisi hazırlamak amacıyla kullanılacak iki veri setinden her sınıfa ait bazı bölgeler seçilmiştir. Seçilen bölgeler eğitim seti olarak sınıflandırıcıya tanıtılmış ve görüntünün geneli üzerinden performans sonuçları karşılaştırılmıştır. Kullanılan veri setleri, seçilen eğitim seti ve görüntülerin içerdiği sınıflar Şekil 2.20 (büyük harf mi küçük harf mi olacak kontrol et) ve şekil 2.21 ‘de verilmiştir.



Şekil 2.20 : Pavia Üniversitesi görüntüsü, eğitim seti ve doğru sınıf değerleri



Şekil 2.21 : RetigaEx görüntüsü, eğitim seti ve doğru sınıf değerleri

Her bir görüntüde her sınıf için seçilen eğitim setlerinin sayıları ve oranları çizelge 2.2 ve çizelge 2.3’te ayrıntılı olarak açıklanmıştır. Örneğin; Pavia görüntüsünde asfalt olarak etiketlenmiş toplam piksel sayısı 6631 olup bunların içerisinde 222 piksel eğitim seti

olarak seçilmiştir. Aynı şekilde diğer tüm sınıflar için toplam kaç adet etiketlenmiş piksel sayısı olduğu ve seçilen eğitim seti sayısı ayrıntılı olarak verilmiştir. Genel toplamda Pavia görüntüsü için toplam verinin %6.77 'si ve RetigaEx verisinin toplam %0.6 'sı eğitim seti olarak kullanılmıştır. Portakal görüntüsünün olduğu veri setinde çok daha az oranda eğitim setinin seçilmiş olmasının sebebi daha az sınıfa sahip olması ve sınıflardan birinin (sağlıklı kısmın) diğer sınıflara oranla çok daha fazla yer kaplıyor olmasıdır.

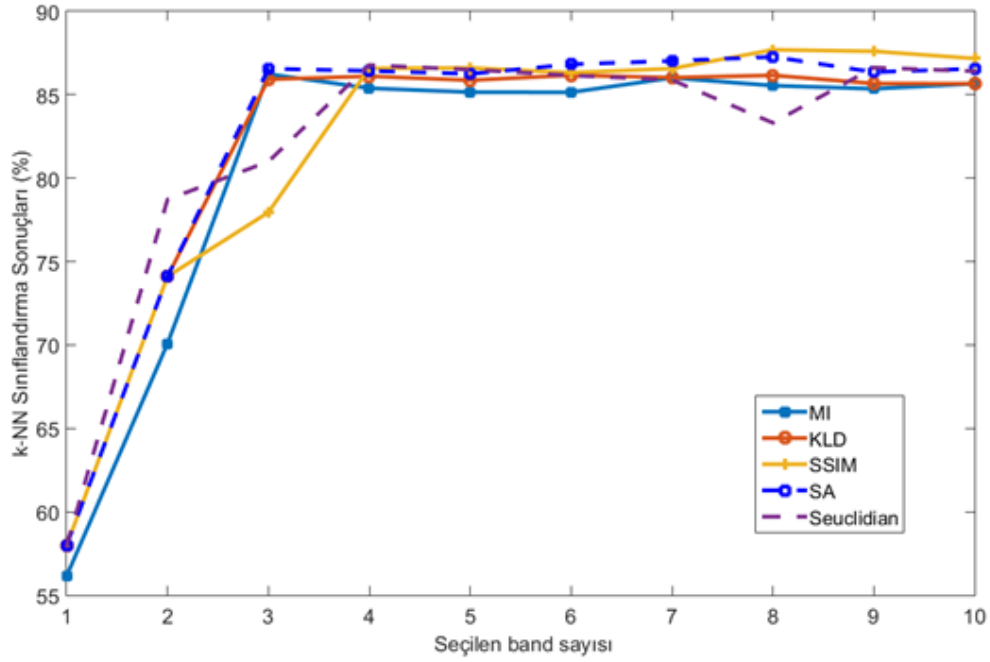
Çizelge 2.2 : Pavia Üniversitesi eğitim seti

Sınıflar	Toplam	Eğitim	Yüzde (%)
Asfalt	6631	222	3.347
Çayır	18649	1052	5.641
Çakıl	2099	322	15.34
Ağaç	3064	181	5.907
Boyalı metal	1345	67	4.981
Toprak	5029	400	7.953
Katran	1330	196	14.73
Tuğla	3682	334	9.071
Gölge	947	125	13.19

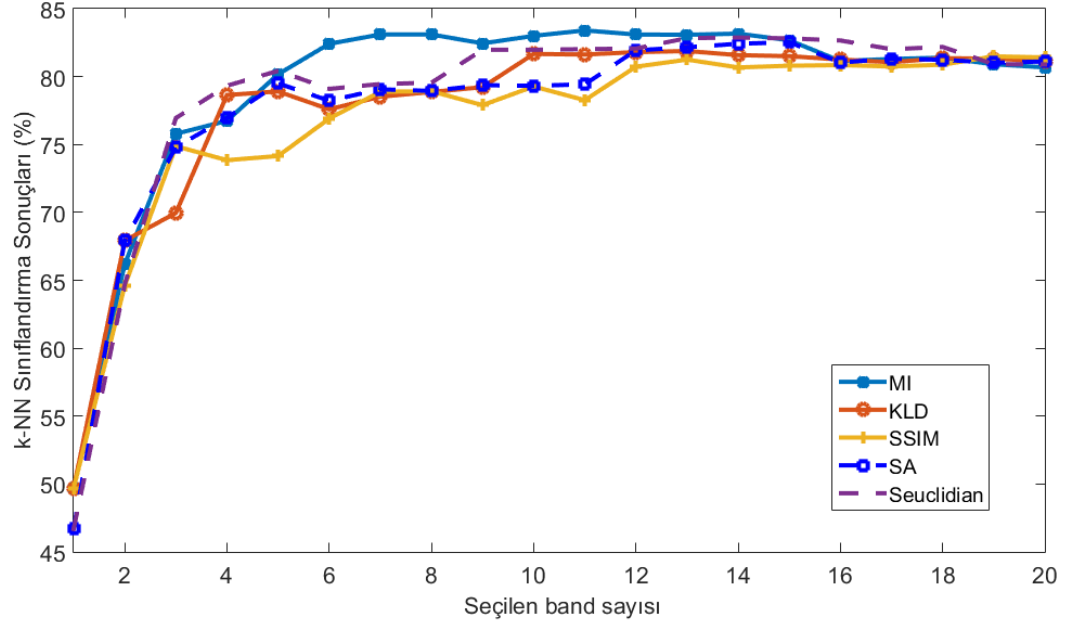
Çizelge 2.3 : RetigaEx (Portakal) eğitim seti

Sınıflar	Toplam	Eğitim	Yüzde (%)
Sağlıklı kısım	35049	175	0.499
Çürük kısım	5234	50	0.955
Sap kısmı	841	25	2.97

Önerilen IPBS bant seçme algoritmasının hiyerarşik kümelemede farklı benzerlik ölçütleri kullanılarak elde edilmiş k-NN sınıflandırma sonuçları şekil 2.22 ve şekil 2.23’de karşılaştırılmıştır. Görüntünün içeriğinin daha sade olduğu yakın mesafe portakal görüntüsünde (RetigaEx) önerilen anahtar nokta temelli özellik çıkarma yöntemi tüm benzerlik kriterlerinde kullanıldığında birbirine yakın sonuçlar elde edilmekle birlikte, SSIM ve SA benzerlik ölçütlerinde çok az da olsa daha başarılı sonuçlar alınmıştır.



Şekil 2.22 : RetigaEx (Portakal) görüntüsünün önerilen IPBS yöntemi ile farklı benzerlik ölçütlerine göre k-NN sınıflandırma sonuçları

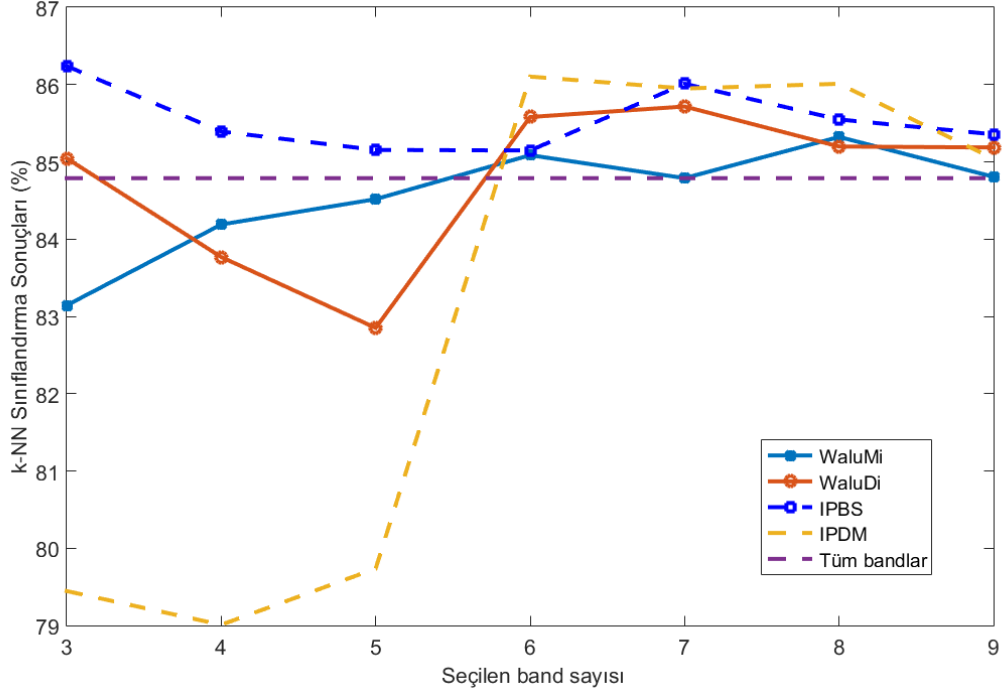


Şekil 2.23 : Pavia görüntüsünün önerilen IPBS yöntemi ile farklı benzerlik ölçütlerine göre k-NN sınıflandırma sonuçları

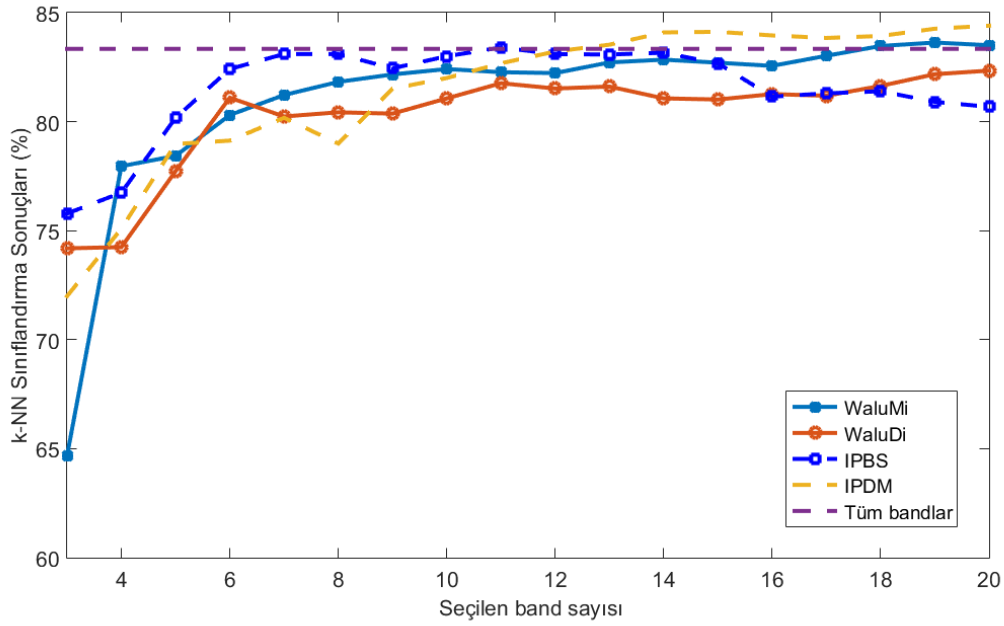
SA ve SSIM benzerlik ölçütleri diğerlerine nazaran hesaplaması daha hızlı ve basit ölçütlerdir. Fakat yakın mesafe görüntüde fazla ayrıntı olmadığından dolayı basit benzerlik kriterleri bile iyi sonuç vermiştir. Pavia görüntüsünde ise hiyerarşik kümelemede benzerlik ölçütü olarak MI kullanılması ile diğer benzerlik kriterlerine nazaran daha iyi sonuç alındığı gözlemlenmektedir (Şekil 2.23). Zaten literatürde kabul gören Walumi algoritmasında MI benzerlik ölçütünün bant seçme problemindeki başarısı bilinmektedir. Bu sebeple, önerilen IPBS yöntemi benzerlik ölçütü olarak MI ile birlikte kullanılmış ve diğer algoritmalar ile performans kıyaslaması gerçekleştirilmiştir.

Önerilen IP temelli (IPBS) yöntemde, MI benzerlik ölçütü kullanılarak diğer bant seçme yöntemleri ile karşılaştırıldığında elde edilen genel sınıflandırma doğruluk (GD) sonuçları şekil 2.24 ve şekil 2.25 'te gösterilmiştir. Genel sınıflandırma değerlendirmesinde veri setindeki tüm sınıflar bir bütün olarak kabul edilmiş ve genel performans değerlendirmesi yapılmıştır. Veri setlerindeki her bir sınıfa ait ayrı ayrı değerlendirmelerin yapıldığı ortalama doğruluk (OD) sonuçları ise çizelge 2.4 ve 2.5'te paylaşılmıştır. Önerilen algoritmaların test edilmesi için Walumi algoritmasına ek olarak Waludi algoritması da performans kıyaslamasında kullanılmıştır. Waludi

algoritmasının Walumi algoritmasından farkı MI yerine KLD benzerlik ölçütünü kullanmasıdır [9]. Ayrıca, HS verisinin tamamı kullanıldığında elde edilen sınıflandırma doğruluk sonuçları (tüm bantlar) da algoritma performanslarının kıyaslandığı grafikler üzerinde aşağıda gösterilmiştir. Performans kıyaslamasında kullanılan Walumi ve Waludi algoritmaları son dönemde gerçekleştirilen çalışmalarda performans kıyaslanmasında yaygın olarak kullanılmaktadır [21,31].



Şekil 2.24 : RetigaEx verisi için k-NN sınıflandırma sonuçları



Şekil 2.25 : Pavia verisi iin k-NN sınıflandırma sonuçları

Önerilen IPBS algoritmasını seçilen farklı bant sayıları iin k-NN sınıflandırma sonuçlarına göre test edildiğinde, literat rde kabul g ren ve performans karşılaştırılmalarında yaygın olarak kullanılan Walumi algoritmasından daha iyi doėruluk oranları elde edildiėi g r lm şt r. Buna g re; hiyerarşik k melemenin MI ile g r nt leri alt k melere ayırma bařarısına g re iyi sonuçlar alındıėı g zlemlenmektedir. Önerilen y ntem ile Walumi algoritması, her ikisi de aynı benzerlik  l t ne g re k meleme yapmaktadır, fakat k melerden ilgili k meyi temsil edecek g r nt y  seçme algoritmaları birbirlerinden farklıdır. IPBS algoritmasında  nerilen g r nt n n yapısal  zelliklerinin katkısı ile gerekleřtirilen y ntem  zellikle ok d ř k sayıda bant sayıları iin daha bařarılıdır. G r nt  ieriėinin nispeten sade olduėu RetigaEx g r nt s nde bile genel olarak Walumi algoritmasına g re bařarılı sonuçlar alınmıřtır. Ayrıca diėer benzerlik kriterleri kullanıldığında da bařarılı sonuçlar alındıėı g zlemlenmiřtir (Şekil 2.22). Yani g r nt ler birbirinden baėımsız k melere ayrıldıktan sonra yapısal  zelliklerin dikkate alınması bařarılı sonuçlar vermiřtir. Fakat k me sayısı arttıka birbirine yapısal olarak yakın g r nt ler farklı k melerde yer alabilecektir. Bu durum ise IP'nin yoėunluėunu dikkate alarak aėırlık katsayısı hesaplayan IPBS algoritmasını olumsuz etkilemektedir. IPDM algoritması ise diėer t m benzerlik kriterlerinden farklı

olarak görüntünün yapısal özelliklerini dikkate alarak kümeleme işlemi yapmakta ve her iki veri setinde de belirli bir bant sayısından sonra diğer algoritmalara nazaran daha iyi performans göstermektedir. Seçilen bant sayısı az olduğunda bu algoritmanın düşük performans vermesinin sebebi, az sayıda küme oluşturulduğunda kümelerin yeterince homojen oluşturulamamasıdır. Yani önce istatistikî yöntemlere göre kümeleme yapıp sonra yapısal özellikler dikkate alındığında çok düşük sayıda bantlar için iyi sonuç alınabilmekte fakat sadece yapısal özelliklerin kullanıldığı bant seçme algoritmasının (IPDM) düşük sayıda bantlar için nispeten kötü performans göstermesine rağmen, belirli sayıdaki bant sayısından sonra en iyi performansı gösterdiği görülmektedir.

Çizelge 2.4 : RetigaEx verisi için GD ve OD kıyaslamaları

Algoritmalar		N=3	N=4	N=7	N=8	N=9
Walumi	GD(%)	83,065	84,189	84,789	85,322	84,806
	OD(%)	64,037	65,244	66,064	66,507	65,502
Waludi	GD(%)	85,045	83,768	85,716	85,198	85,186
	OD(%)	64,599	62,981	65,734	65,994	66,848
IPBS	GD(%)	86,239	85,392	86,010	85,548	85,356
	OD(%)	62,756	63,065	64,073	64,147	64,333
IPDM	GD(%)	79,445	79,009	85,949	86,01	85,04
	OD(%)	60,527	62,098	69,574	69,737	67,548

Çizelge 2.5 : Pavia verisi için GD ve OD performans kıyaslamaları

Algoritmalar		N=3	N=5	N=8	N=10	N=15	N=20
Walumi	GD(%)	64,846	78,429	81,824	82,414	82,704	83,505
	OD(%)	69,968	79,779	83,620	84,393	84,704	85,038
Waludi	GD(%)	74,191	77,728	80,429	81,068	81,016	82,341
	OD(%)	70,882	77,950	82,328	81,888	82,931	84,280
IPBS	GD(%)	75,786	80,173	83,102	83,00	82,674	80,679
	OD(%)	74,982	79,860	84,036	83,912	83,393	80,977
IPDM	GD(%)	71,998	78,976	78,998	82,001	84,121	84,402
	OD(%)	66,668	80,961	79,480	82,744	85,540	85,877

GD ve OD sonuçlarının daha ayrıntılı kıyaslaması çizelge 2.4 ve 2.5'te yapılmıştır. Buna göre özellikle çok düşük sayıdaki bant sayılarından sonra IPDM algoritmasının iyi sonuçlar verdiği gözlemlenmiştir. 103 görüntüden oluşan Pavia hiperküpünün 10'dan az sayıda bantının seçilmesi durumunda spektral çözünürlük fazlasıyla düşürülmüş olacaktır. Fakat bant sayısı 15 ya da 20 mertebelerinde seçilerek önerilen IPDM algoritması hem GD hem de OD sonuçlarında en iyi performansı göstermektedir. Aynı şekilde 33 görüntüden oluşan RetigaEx hiper küpünün sadece 7 ya da 8 bantı seçildiğinde yine en iyi performans sonuçları IPDM ile elde edilmiştir.

Buna göre, bant seçme işleminde veri boyutlarının çok küçük mertebelere indirilmeyip daha makul seçildiği durumlarda IPDM algoritması bant seçme probleminde içerik olarak birbirlerinden bir hayli farklı her iki görüntü için de en başarılı sonuçları vermiştir. Çok düşük sayıda bant seçilmesi durumunda ise IPBS algoritması daha başarılıdır. Böylelikle bu tez kapsamında seçilebilecek farklı benzerlik ölçütlerine göre başarılı sonuçlar alınabilecek hiyerarşik kümeleme ve IP temelli yeni bir eğiticişiz bant seçme algoritması olan IPBS geliştirilmiştir. Ayrıca görüntünün yapısal özellikleri dikkate alınarak hesaplanan bir benzerlik ölçütü olan IPDM geliştirilmiş ve hiyerarşik kümeleme ile bant seçme problemindeki başarısı ortaya konulmuştur.



3. KLASİK YÖNTEMLER VE UNET İLE BİNA BÖLÜTLEMESİ

Uydu görüntülerinde uzamsal çözünürlüğün artması, metre (hatta cm) mesafelerine kadar düşmesi nispeten küçük nesnelerin de algılanabilmesine olanak sağlamıştır. Uzamsal çözünürlüğü yüksek olan bu görüntüler kullanılarak kent alanlarındaki değişikliklerin analizi, otomatik gözlemlenmesi gibi uygulamalar gerçekleştirilebilmektedir. Bunun için uzaktan algılanmış görüntülerden binaların tespit edilmesi üzerine bir çok çalışma yapılmıştır. Binaların tespit edilmesinde yararlanılan önemli bir özellik binaların oluşturduğu gölgelerdir. Bu görüntülerin gündüz çekildiği düşünüldüğünde gölgelerden hareket edilmesi suretiyle binaların tespit edilmesine yönelik çalışmalar gerçekleştirilmiş ve başarılı sonuçlar alınmıştır [32, 33]. Bazı çalışmalarda ise bina olması muhtemel pikseller tespit edildikten sonra karar aşamasında hipotezin doğrulanması ve bina yüksekliğinin tahmin edilmesi probleminde gölge bilgisi kullanılmıştır [34,35]. Liow ve Pavlidis tarafından gerçekleştirilen çalışmada ilk olarak kenar bilgileri çıkarılmış daha sonra ise kenarların oluşturduğu bölgelerin hemen yanındaki gölge alanlar kontrol edilerek binalar tespit edilmeye çalışılmıştır [36]. Bu yöntemde kenar bilgisinden tam ve doğru bir şekilde kapalı alanların bulunması büyük bir problemdir. Bunun için alan büyütme algoritması kullanılmakla birlikte binalara ait olan kenar bilgilerinin bulunmasında bir belirsizlik vardır. Gerçekleştirilen bir başka çalışmada [37] ise renk bilgisi kullanılarak gölge alanlar tespit edilmeye çalışılmış daha sonra ise güneş ışığının yönüne bakılarak binaların konumu belirlenmeye çalışılmıştır. Bu çalışmada önerilen yöntemde Canny kenar bulma algoritması ile bulunan kenarlar kullanılarak dikdörtgen alanlar çıkarılmaya başlanmıştır. Bu yöntem kenar bilgisine fazla bağımlı/hassas ve sadece dikdörtgen şeklindeki binalar için iyi sonuçlar verdiği görülmüştür.

Akçay ve Aksoy [38] tarafından geliştirilen yöntemde ilk olarak bina olmaya aday yerler gölge bilgisi ile belirlenmektedir. Bunun için bulunan tüm alanlar ile gölge alanları arasındaki ilişkiler için kısıtlamalar belirlenmekte ve bölgelere ayrılmış bu alanlar arasında kümeleme işlemi yapılmaktadır. Ali özgün Ok [39] tarafından gerçekleştirilen çalışmada ise tüm bu evrelere ek olarak CRF (Conditional random field) optimizasyon algoritması kullanılarak en son karar aşamasında piksel seviyesinde sınıflama

yapılmıştır. Bu çalışmada ilk olarak gölge alanlar üzerinden bina olması muhtemel alanlar bulunmuştur. Bunun için de renk bilgisi kullanılarak ilk olarak gölge alanları genişletilerek bulunan küçük parça halindeki gölgeler birleştirilmiştir. Daha sonra ise her bir bölge için CRF algoritması çalıştırılmış, pikseller bina ya da bina değil olarak etiketlenmiştir. Algoritmanın temelinde başlangıçtaki gölge alanların bulunması yer almaktadır. Bu aşamada bulunan bölgelerin birleştirilmesi son derece önem arz etmektedir. Alan büyültme işleminde belirlenen eşik değere göre hangi alanların birleştirilmesi gerektiği hangilerinin aynı nesnenin gölgesi olmadığı (birleştirilmemesi gerektiğine) karar verilmiştir.

Yine Ali Özgün Ok ve arkadaşları tarafından gerçekleştirilen bir başka çalışmada, gölge alanların sınırları ve şekillerinin daha iyi hale getirilmesi için kısıtlı bölge genişletme algoritması önerilmiştir [40]. Fakat, bu algortmada bazı gölge alanlarının hatalı bir şekilde birleştirildiği de belirtilmiştir. İlgili çalışmada bölge genişletme uygulanmasından önce ve sonra elde edilen gölge alanlarındaki piksel sayıları dikkate alınarak bir eşik değeri bulunmaya çalışıldığından bahsedilmiştir. Böylece eşik değeri ile bölge genişletme işlemi kontrol altına alınmaya ve hatalı birleştirmeler elimine edilmeye çalışılmıştır. Ayrıca, bir başka işlem olarak güneş ışınlarının geliş açısı ve gölge uzunlukları kullanılarak bina yükseklikleri tahmin edilmeye çalışılmıştır. Bu bilgi de farklı nesnelere ait gölge alanlarının ayırt edilebilmesi için kullanılmıştır. Gölge alanlara bu gibi düzeltme işlemleri uygulandıktan sonra, binalar ile gölge alanlar arasında bulanık ilişki yöntemi ile ilişki kurulmaya çalışılmıştır. Bina tespit sorunu ilk olarak iki sınıf değerine sahip bir sınıflama problemi gibi düşünülmüş ve sınıflama için çizge kesme algoritması kullanılmıştır. Çizge kesme algoritması aslında yarı otomatik bir ön plan/arka plan ayırt etme algoritmasıdır. Bu algortmada kullanıcı tarafından seçilen bir nesne arka plandan ayırt edilmeye çalışılmıştır [41].

Önerilen bina tespit algoritmasında, bina bölgelerinin çizge kesme algoritmasına girdi olarak verilebilmesi için ilk aşamada önerilen bulanık ilişki ve gölge alanlar kullanılmıştır. Böylece ilk seviye sınıflama işlemi tamamlanmış olur. İkinci seviye sınıflama işleminde ise ilk aşama sonucunda görüntüde bulunan binalar, yeşil alanlar ve gölge alanlara farklı sınıf değerleri atanır. Bunların dışında kalan diğer alanlara ise bir

başka sınıf değeri verilir. İkinci seviye sınıflama işlemi yapmanın amacı, ilk seviye sonucunda yanlış sınıflanan değerlerin revize edilmesidir. İkinci seviye sınıflama için, çok sınıflı çizge tabanlı optimizasyon algoritması önerilmiştir. İki seviye sınıflama işlemi neticesinde binalar etiketlenebilmektedir. İkinci seviye yani global sınıflama sonucunda da yine bazı bölgelerin hatalı olarak etiketlendiği belirtilmiştir [42]. Bu problemi en aza indirmek için, sınıflama sonucu elde edilen bina alanlarına gölge doğrulama işlemi uygulanmıştır.

Literatürde geçen birçok çalışmanın, gölge ve önışleme adımlarının üzerine kurulduğu, eşik değerlere bağımlı ve çok aşamalı sınıflama işlemlerinden sonra belirli bir başarıya ulaşılabildiği düşünüldüğünde, tüm bu izlenen yöntemler yerine görüntüden özellik çıkarımı için derin öğrenme tekniklerinin daha genel ve iyi sonuçlar verebileceği düşünülebilir. Son dönemde derin öğrenmenin son derece popüler olması, açık kaynak kütüphanelerinin bulunması ve en önemlisi görüntü işleme problemlerinin çözümünde başarılı sonuçlar vermesiyle birlikte, uzaktan algılanmış görüntülerden bina tespiti problemi için de derin öğrenme tercih edilmeye başlanmıştır. Önceki paragrafta bahsettiğimiz birçok algoritma, görüntü işleme tekniklerinin uygulanacağı alanda uzman kişiler tarafından geliştirilen kompleks görüntü işleme yöntemleri ile geliştirilmiştir. Çoğu önerilen yöntemde, kullanıcı tanımlı veya veri setine göre değişebilecek bazı parametreler ile hassas yaklaşımlar mevcuttur. Derin öğrenme ise bir çeşit makine öğrenmesi algoritması olması itibariyle klasik algoritma problem çözüm mantığına önemli bir değişiklik/revizyon getirmiştir. Böylece uzman kişilerin özellik çıkarımı için spesifik ve detaylı algoritmalar geliştirmesi yerine, çözülecek problem için örnek veri setlerinin hazırlanıp etiketlenmesi suretiyle makine öğrenmesi algoritmalarına girdi olacak veriler hazırlanır. Etiketlenmiş veriler modele verilerek, modelin eğitilmesi sağlanır. Görüntü işleme problemlerinde kullanılan derin öğrenme modellerinde ise, görüntü işleme için özel hazırlanmış katmanlar ile eğitim işlemi esnasında görüntüden özellik çıkarım işlemi de modele dahil edilmiş olur. Bu bölümde biyomedikal görüntü işleme ve bina tespitinde kullanılan *Unet* [43] ağ yapısından esinlenilerek bir derin öğrenme modeli tasarlanıp, örnek veri setleri üzerinden eğitilmiştir. Model performansı değerlendirilmiş ve gölge bilgisi ile elde edilen sonuçların iyileştirilmesi/revize edilmesi üzerine yöntemler geliştirilmiştir.

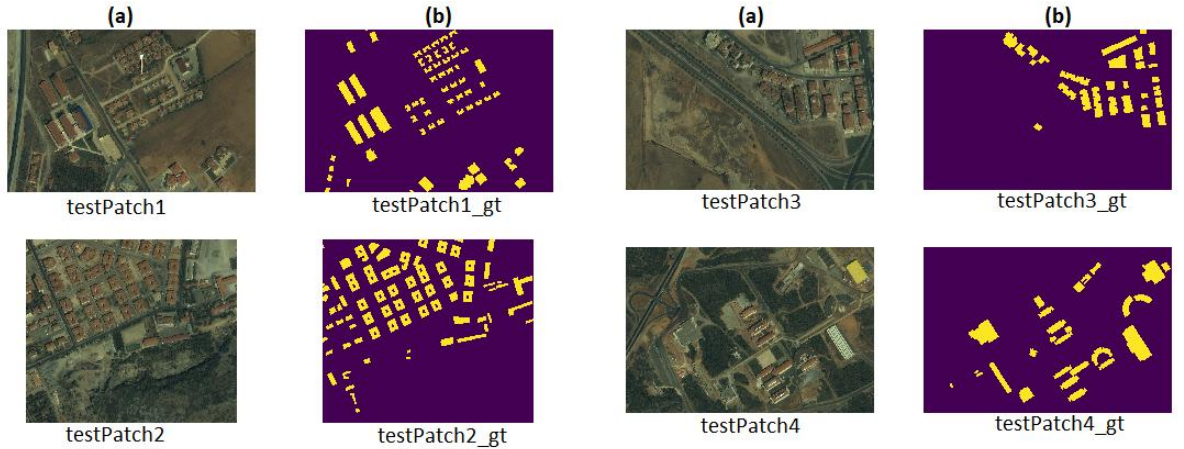
3.1 Kullanılan Görüntüler

Bu bölümde kullanılan görüntüler IKONOS (1 m) ve Quicbird (60 cm) olmak üzere iki farklı uydudan elde edilmiştir. Tüm görüntüler Mavi-Yeşi-Kırmızı ve NIR (kızıl ötesi) olmak üzere 4 band içermektedir ve 11 bit radyometrik çözünürlüğe sahiptir. Her bir görüntü ile alakalı detaylı bilgiler Çizelge 3.1’de görülebilir [42].

Çizelge 3.1 : Kullanılan görüntülerin özet bilgisi

Görüntü	Çözünürlük	Sensör
<i>testPatch1</i>	560 x 367	IKONOS (1 m)
<i>testPatch2</i>	554 x 483	
<i>testPatch3</i>	468 x 304	
<i>testPatch4</i>	896 x 600	
<i>testPatch5</i>	1213 x 958	QuickBird (0.6 m)
<i>testPatch6</i>	922 x 634	
<i>testPatch7</i>	928 x 639	
<i>testPatch8</i>	1009 x 695	
<i>testPatch9</i>	1615 x 1209	
<i>testPatch10</i>	1656 x 1240	
<i>testPatch</i>	1222 x 915	
<i>testPatch12</i>	1311 x 848	
<i>testPatch13</i>	1193 x 772	
<i>testPatch14</i>	1193 x 771	

Şekil 3.1’de veri setindeki *testPatch1*, *testPatch2*, *testPatch3*, *testPatch4* isimli görüntülerin RGB görünümünü (a) sütunlarında, her bir görüntüdeki binaların etiketlendiği görüntüleri ise (b) sütunlarında görebilirsiniz.



Şekil 3.1 : Veri setinden örnek görüntüler ve etiketler

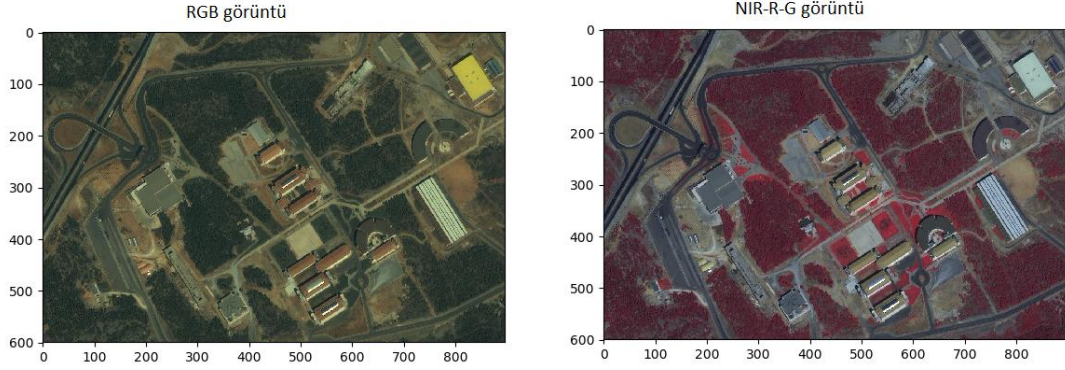
3.2 Gölge tespiti

Uydu ve hava görüntülerinde, verinin elde edilme zamanı, yüzey özelliği ve yüzeydeki nesne sayısına bağlı olarak farklı sayıda ve şekillerde gölgeler bulunur. Gölgenin nesnelerin şeklini gösteren önemli bir ipucu olduğu düşünüldüğünde, bu bilginin insan yapımı nesnelerin (bina gibi) tespit edilmesinde kullanılması önemli avantajlar sağlamaktadır. Uzaktan algılanmış görüntülerden binaların tespit edilmesi amaçlanıyorsa sadece şekil özelliklerine bakılarak otomatik bir tespit yapılması çok da kolay değildir. Fakat görüntülerin aydınlık ortamda elde edildiğinde her binanın güneşin gelme açısına göre bir gölgesi bulunmaktadır. Her binanın bir gölgesi olmakla birlikte her gölgesi olan nesne bina değildir. Uzaktan algılanmış görüntülerin çok bantlı görüntüler olduğu da düşünülürse gölge alanların bulunması nispeten daha kolay olmaktadır. Dolayısıyla, gölgeler nesne tespitinde kuvvetli bir yardımcı bilgi olarak kullanılmaktadır.

Literatürde üç bantlı RGB görüntülerin yanında çok bantlı RGB-NIR görüntüler de gölgelerin tespit edilmesinde kullanılmıştır. Örneğin, Polidorio tarafından gerçekleştirilen çalışmada doygunluk (saturation) ve yoğunluk (intensity) bileşenlerinin gölge alanlarının karakterini daha iyi yansıttığı gösterilmiştir [44]. Tsai'nin [45] çalışmasında ise beş farklı renk uzayı karşılaştırılmış ve otomatik gölge tespiti için yine HSI uzayının en optimum çözüm olduğu ortaya konulmuştur. Teke ve arkadaşları [46] tarafından önerilen yöntemde ise HSI renk uzayı kullanılmakla birlikte NIR bant bilgisinin özelliğinden de faydalanılmıştır. RGB renk uzayında renk üç ana rengin (kırmızı, yeşil mavi) karışımı ile tanımlanır. HSI renk uzayında ise renk doygunluk, parlaklık ve gri değeri ile tanımlanır. Doymunluk rengin canlılığını belirlerken, parlaklık rengin aydınlığını ifade eder. HSI uzayında siyah renk için renk ve doymunluk değeri herhangi bir değeri alabilirken parlaklık değeri sıfırdır, aynı şekilde beyaz renk için ise parlaklık değeri maksimumdur (8 bit için 255). NIR bandında gölge alanlar koyu olarak görünür ve NIR bandının bu özelliği kullanılarak gölge alanlar diğer koyu renk alanlarından ayırt edilebilir.

Teke ve arkadaşları tarafından önerilen gölge alan tespiti algoritmasına göre; gölge alanlardaki piksel değeri mavi bant için kırmızı ve kızıl ötesi gibi diğer bantlara nazaran daha yüksek değere sahiptir. Ayrıca mavi band ışık spektrumunda daha dar

yer kapladığı için daha az bilgi taşımaktadır. Bu sebeple mavi band kaldırılarak RGB uzayı yerine NIR-R-G (false color) uzayı tercih edilir. Örnek bir görüntü ve NIR-R-G (false color) uzayına dönüştürülmüş görüntü Şekil 3.2’de gösterilmiştir.

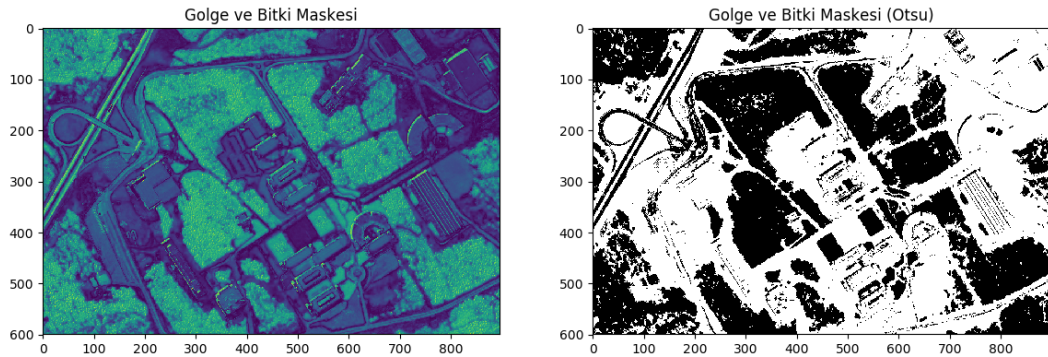


Şekil 3.2 : RGB ve NIR-R-G görüntüler

NIR-R-G görüntüsü ayarlandıktan sonra, HSI renk uzayına dönüştürülür. Dönüştürülen bu uzayda doygunluk (*saturation-S*) ve renk (*intensity-I*) bandları normalize edilir. Yeşil yani bitkisel alanlar ve koyu alanlar için normalize edilen bu (S-I) değeri benzer sonuçları verecektir. Bu alanları ayırt etmek için bitkisel ve koyu alanların haritası (K) aşağıdaki formülle çıkarılır (**Denk.23**).

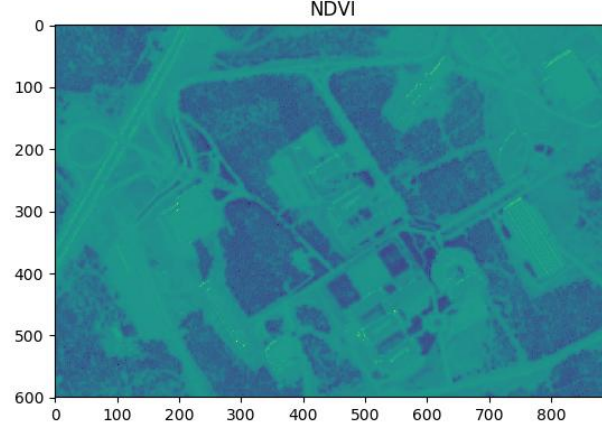
$$K = \frac{S-I}{S+I} \quad (23)$$

Doygunluk ve renk değerlerinin oranlanmasıyla elde edilen bitkisel ve koyu alan haritası Şekil 3.3’te gösterilmiştir.



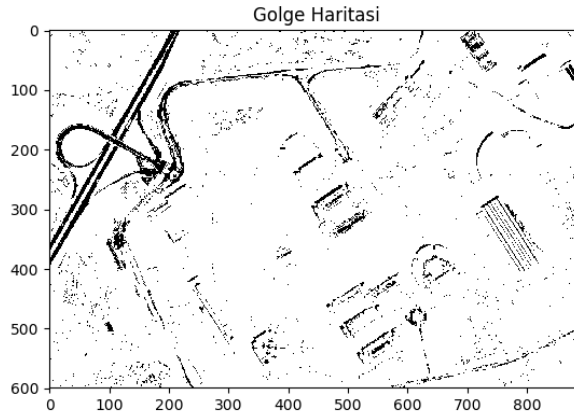
Şekil 3.3 : Gölge ve bitki maskesi

Gölge ve bitkisel alanları, görüntünün geri kalan kısımlarından ayırt etmek için ise Otsu algoritması kullanılır [47]. Otsu arka plan çıkarma algoritması ile elde edilmiş bölge ve bitkisel alan maskesi görüntüsü Şekil 3'te sağ sütunda görülmektedir. Böylece görüntüdeki gölge ve bitkisel alanları ifade eden maske görüntüsü elde edilmiş olur. Daha sonra ise kırmızı ve NIR bantlar kullanılarak bitkisel alanlar (Şekil 3.4) NDVI (*Normalized Difference Vegetation Index*) ile elde edilir [48].



Şekil 3.4 : NDVI bitkisel alanlar

Son olarak ise gölge ve bitkisel alan maskesinden, NDVI ile bulunan bitkisel alan maskesinin fark görüntüsünü hesaplanarak gölge haritası çıkarılmış olur (Şekil 3.5).



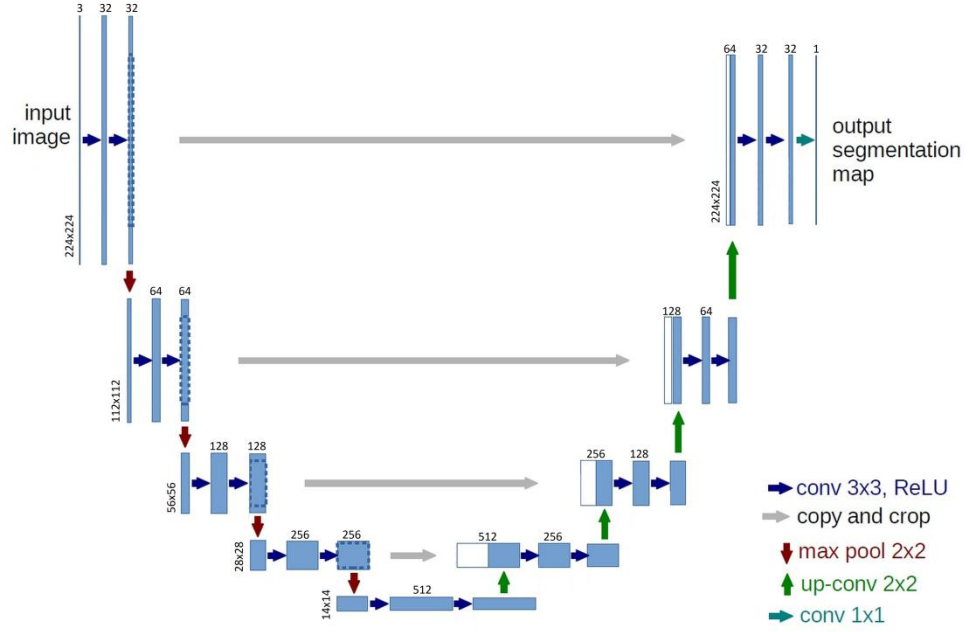
Şekil 3.5 : Gölge maskesi

3.3 Derin öğrenme modeli oluşturulması

Makine öğrenmesi algoritmalarında, modelin oluşturacağı çıktı ile gerçek değerler arasındaki benzerliğin/hatanın oranına göre model doğruluğu tespit edilip ağ eğitilir. Eğitim ile amaçlanan gerçek değerler ile tahmin edilen değerlerin arasındaki hatanın minimize edilmesidir. Görüntü işlemede bölütleme algoritmalarının performans değerlendirmesinde *dice* ölçütü tercih edilmektedir. Doğru bölütleme sonucu ile tahmin edilen sonuçların karşılaştırılmasında *dice* ölçütünün başarılı sonuçlar verdiği biyomedikal görüntüler üzerine gerçekleştirilen bir çalışmada [43] ortaya konulduğundan, benzerlik kriteri olarak **(Denk.24)** 'deki formülde ifade edilen *dice* ölçütü kullanılmıştır. (A ve B iki ayrı kümeyi ifade etmektedir)

$$\text{dice} = \frac{|A \cap B|}{|A| + |B \setminus A|} \quad (24)$$

Uzaktan algılama uygulamalarında ve HS verilerin sınıflandırılmasında derin öğrenme algoritmaları son zamanlarda tercih edilmekte ve literatürdeki diğer yöntemlere nazaran düşük hata oranları ile tatmin edici performans göstermektedir [49,50]. Geleneksel örüntü tanıma yöntemlerinde ham veriden konunun uzmanları tarafından kompleks ve ayırt edici özellikler çıkarılırken, derin öğrenme modellerinde ise probleme özgü özellik çıkarma işlemi otomatikleştirilmektedir. Böylece düşük seviyeden yüksek seviyeye kadar hiyerarşik özellikler çıkarılarak makine öğrenmesi süreci tamamlanmış olur [51,52]. Derin öğrenme algoritmaları içerisinde belki de en yaygın kullanılan *Convolutional Neural Network* (CNN) tipindeki ağ modelidir. CNN aslında ileri beslemeli gelişmiş bir yapay sinir ağı modelidir. CNN ile sinir ağının öncesine özellik çoğaltma ve özetleme katmanları eklenerek özellik çıkarma işlemi de ağ yapısının içerisine dahil edilmiş olur. Görüntü işleme problemlerinde yaygın kullanılan bu katman aslında görüntüde kenar ve köşe gibi bilgilerin bulunması için kullanılan filtrelerin işlevini gerçekleştirir. Bu bölümde *Unet* [43] modelinden esinlenilerek derin öğrenme tabanlı bina tespit algoritması önerilmiştir.



Şekil 3.6 : Bina bölütlemeye kullanılan Unet mimarisi[43]

Orijinal Unet mimarisi 572x572x3 boyutlarındaki görüntüler için tasarlanmıştır. Fakat ağır girdi boyutları arttıkça çok daha fazla işlem gücüne ihtiyaç olduğundan dolayı daha küçük boyutlu bir ağ modeli eğitilmesi tercih edilmiştir. Tez çalışmasında kullanılan 224x224x3 boyutlarında girdi görüntüsü kabul eden Unet mimarisi ve her katman neticesinde elde edilen özellik sayıları Şekil 3.6’da gösterilmiştir. Veri seti içerisinde seçilen bazı görüntülerden 224x224 boyutlarında rastgele farklı alanlar seçilerek eğitim seti oluşturulmuştur. Veri setimiz iki farklı uydudan elde edilen 14 görüntüden oluşmaktadır. Ağ eğitimi için ilk aşamada IKONOS uydusuna ait 4 farklı görüntüden sadece birisi eğitim için kullanılmış ve belirlenen görüntü üzerinde 224x224 boyutlarında 10 farklı alan seçilerek ağ eğitilmiştir (*model-1*). Eğitim sonunda eğitim seti için 0.872 doğruluk değerine ulaşılmıştır. QuickBird uydusuna ait 10 görüntüden ise 2 görüntü eğitim seti olarak seçilmiş ve bu iki görüntüden toplamda 20 farklı görüntü oluşturularak bir başka ağ eğitimi gerçekleştirilmiştir (*model-2*). Quickbird verileri ile eğitim sonucunda ise 0.8349 doğruluk değerine ulaşılmıştır.

Makine öğrenmesi model değerlendirmelerinde, gerçek değerler ile tahmin edilen değerlerin karşılaştırılmasında hata matrisleri yaygın olarak kullanılmaktadır. Hata matrisinde gerçekte doğru olan ve doğru sınıflanan değer sayısı T_p , negatif olan ve

negatif sınıflanan değerler T_n ile ifade edilir. Gerçek değer negatif ancak tahmin pozitif ise bu hata durumu F_p ile ifade edilir. Diğer yandan, gerçek değer pozitif ancak tahmin negatif ise bu hata durumu da F_n ile ifade edilmektedir. Hata matrisini oluşturan bu değerler üzerinden çeşitli performans değerlendirme kriterleri hesaplanır. Bu bölümde geliştirilen bina tespit algoritması sonuçlarının değerlendirmesinde piksel bazında kesinlik (*Precision*), hassasiyet (*Recall*) ve F_1 skoru performans kriterleri kullanılmıştır. Kesinlik (P), sonucu pozitif olan ve doğru sınıflanan sonuçların bulunan tüm doğru değerlere oranı ile hesaplanır. Hassasiyet (R) ise pozitif olan ve doğru sınıflanan sonuçların, olması gereken pozitif değerler sayısına oranıdır. F_1 skoru, kesinlik ve hassasiyet değerlerinin harmonik ortalamasıdır (**Denk.25**).

$$P = T_p / (T_p + F_p)$$

$$R = T_p / (T_p + F_n) \quad (25)$$

$$F_1 = \frac{2 P R}{P + R}$$

Kullanılan veri setindeki her bir görüntünün farklı boyutlara sahip olmasından ve oluşturulan modelin 224x224 boyutlarında girdi kabul etmesinden dolayı, ilk olarak veri setindeki görüntüler 224x224 boyutlarına indirgenerek performans değerlendirilmesi gerçekleştirilmiştir. Özellikle QuickBird uydusuna ait görüntülerin çok daha yüksek çözünürlüklü olduğu göz önüne alındığında, görüntülerin ağırlık kabul ettiği 224x224 boyutlarına indirgenmesi veri kaybına neden olmaktadır. Bunun için daha yüksek çözünürlüklü görüntüler üzerinden eğitim gerçekleştirmek bir çözüm olabilir. Fakat yüksek işlem gücü gerektiren böyle bir uygulama için güçlü donanımlara ihtiyaç vardır. Bu sebeple alternatif çözüm olarak; yüksek çözünürlüklü görüntülerin çözünürlüğünün düşürülmesinin yerine görüntüler üzerinde 224x224 boyutlarında pencere dolaştırılarak görüntünün farklı alanları için ayrı ayrı işlem yapılabilir. İlk aşamada eğitilen iki farklı modelin (*model-1* ve *model-2*), görüntü boyutları indirgenerek ve kayan pencere yöntemiyle elde edilen performans sonuçları Çizelge 3.2 ve Çizelge 3.3'de gösterilmiştir.

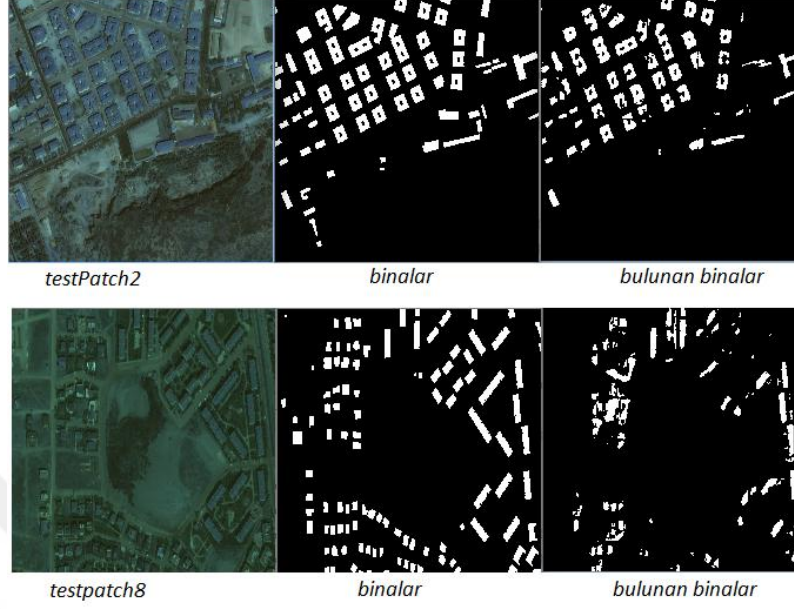
Çizelge 3.2 : IKONOS 1 numaralı görüntü ile eğitim sonuçları (*model-1*)

No	Düşük çözünürlük (kesinlik-hassasiyet- F_1)	Yüksek çözünürlük (kesinlik-hassasiyet- F_1)
1	0.65 – 0.548 – 0.594	0.870 – 0.797 – 0.832
2	0.79 – 0.691 – 0.737	0.902 – 0.754 – 0.821
3	0.763 – 0.502 – 0.606	0.896 – 0.567 – 0.694
4	0.257 – 0.334 – 0.291	0.612 – 0.302 – 0.404

Çizelge 3.3 : Quickbird 5 ve 6 numaralı görüntüler ile eğitim sonuçları (*model-2*)

No	Düşük çözünürlük (kesinlik-hassasiyet- F_1)	Yüksek çözünürlük (kesinlik-hassasiyet- F_1)
5	0.260 – 0.712 – 0.381	0.799 – 0.549 – 0.651
6	0.262 – 0.396 – 0.316	0.691 – 0.528 – 0.598
7	0.266 – 0.717 – 0.388	0.692 – 0.596 – 0.640
8	0.182 – 0.487 – 0.266	0.622 – 0.411 – 0.495
9	0.464 – 0.245 – 0.320	0.725 – 0.048 – 0.090
10	0.306 – 0.317 – 0.311	0.634 – 0.490 – 0.553
11	0.444 – 0.727 – 0.552	0.742 – 0.648 – 0.692
12	0.542 – 0.665 – 0.597	0.693 – 0.292 – 0.411
13	0.630 – 0.691 – 0.659	0.803 – 0.256 – 0.389
14	0.507 – 0.641 – 0.566	0.690 – 0.327 – 0.444

İlk aşamada eğitilen *model-1* ve *model-2* isimli derin öğrenme modellerinde sadece bir uyduya ait görüntü ya da görüntüler kullanılmıştır. Buna göre, testPatch1 görüntüsü ile eğitilen *model-1* testpatch2 ve testpatch3 görüntüleri için iyi sonuç vermiştir (Çizelge 3.2). Fakat IKONOS görüntüsü ile eğitilen model, Quickbird görüntüleri için çok düşük performans sergilemiştir. Aynı şekilde, Quickbird uydusuna ait testpatch5 ve testpatch6 görüntüleri ile eğitilen *model-2* ise IKONOS görüntüleri için aynı iyi sonuçları verememiştir. Ayrıca kayan pencere yöntemiyle yüksek çözünürlüklü görüntüler üzerinde derin öğrenme modeli çalıştırıldığında daha başarılı sonuçlar elde edildiği net bir şekilde görülmektedir. İlk aşamada ağ eğitiminde kullanılmamış testpatch2 ve testpatch8 görüntüleri için doğru bölütleme ve bulunan binaların detayları Şekil 3.7’de gösterilmiştir.



Şekil 3.7 : Veri setindeki örnek görüntüler için bulunan binalar (*model-1*)

Orijinal görüntülerin çözünürlüğünü düşürerek iyi sonuçlar alınamadığından, ilk aşamadan sonraki süreçlerde kayan pencere yöntemi ile yüksek çözünürlüklü orijinal görüntü üzerinde modeller çalıştırılmıştır. Ayrıca her iki uyduya ait görüntüler için ayrı ayrı eğitim yapıldığında, eğitim aşamasında kullanılmayan diğer uyduya ait görüntülerde başarılı sonuçlar alınamadığı tespit edilmiştir. Derin öğrenmede eğitim setindeki veri miktarı son derece önemli olduğundan hem eğitim setindeki veri sayısını artırmak, hem de daha genel bir çözüm oluşturabilmek için her iki uyduya ait görüntülerden ortak bir eğitim seti oluşturulmuştur. Buna göre, veri setindeki 1,2,5,6,7,8 numaralı görüntüler kullanılarak her bir görüntüden 244x244 boyutlarında 10 farklı alan seçilmiş ve 60 görüntüden oluşan eğitim seti hazırlanmıştır. Oluşturulan bu yeni veri seti ile gerçekleştirilen eğitim sonucunda, Tablo da görüldüğü gibi ağ performansı eğitim seti için 0.8134 doğruluk değerine ulaşmıştır (*model-3*). *model-3* için elde edilen performans sonuçları Çizelge 3.4'te gösterilmiştir. IKONOS uydusuna ait 1 numaralı görüntü ile eğitilen *model-1*, Quickbird uydusuna ait 5,6 numaralı görüntüler kullanılarak eğitilen *model-2*, her iki uyduya ait 1,2,5,6,7,8 numaralı görüntülerin kullanıldığı ortak eğitim seti ile eğitilen *model-3* için performans karşılaştırılması çizelge 3.5'te gösterilmiştir.

Çizelge 3.4 : 1,2,3,4,5,6 numaralı görüntüler ile eğitim sonuçları (*model-3*)

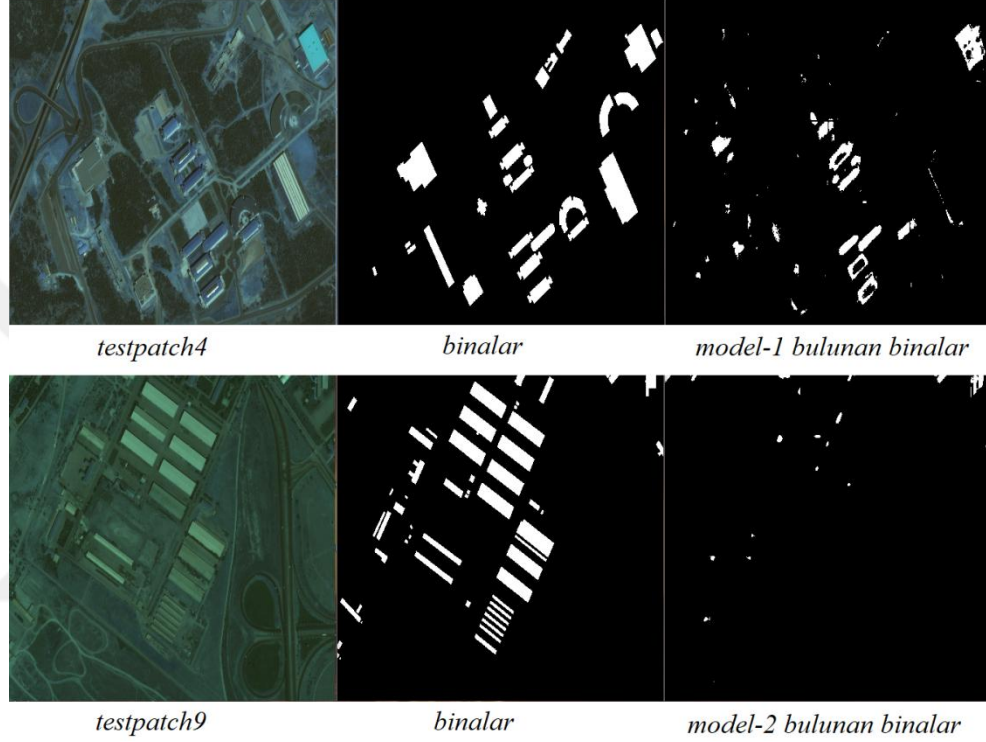
No	kesinlik-hassasiyet-F ₁
1	0.7618 – 0.7055 – 0.7326
2	0.8746 – 0.8200 – 0.8464
3	0.8314 – 0.6756 – 0.7454
4	0.4738 – 0.2854 – 0.3562
5	0.7531 – 0.5376 – 0.6131
6	0.8415 – 0.7300 – 0.7818
7	0.8291 – 0.8383 – 0.8337
8	0.7575 – 0.7695 – 0.7635
9	0.5505 – 0.0981 – 0.1666
10	0.5728 – 0.5193 – 0.5447
11	0.8614 – 0.8102 – 0.8350
12	0.8125 – 0.6930 – 0.7480
13	0.8861 – 0.4837 – 0.6258
14	0.7787 – 0.6946 – 0.7342

Çizelge 3.5 : 3 farklı eğitim seti ile oluşturulan modellerin performans kıyaslaması

No	model1 ve model2 (F ₁)	model3 (F ₁)
1	0.832	0.7326
2	0.821	0.8464
3	0.694	0.7454
4	0.404	0.3562
5	0.651	0.6131
6	0.598	0.7818
7	0.640	0.8337
8	0.495	0.7635
9	0.090	0.1666
10	0.553	0.5447
11	0.692	0.8350
12	0.411	0.7480
13	0.389	0.6258
14	0.444	0.7342

Sırasıyla *model-1* ve *model-2* ile elde edilmiş, düşük doğruluk değerine sahip 4 ve 9 numaralı görüntüler için bulunan binalar ve bina maskeleri Şekil 3.8’de gösterilmiştir. *testpatch9* görüntüsünde keskin yapıda bir şekle sahip olan binaların derin öğrenme modeli ile tespit edilemediği görülmektedir. Bunun sebebi eğitim setinin yetersiz olması

veya mevcut olumsuz sonuç alınan görüntü için benzer binaların eğitim setinde bulunmaması olabilir. Bu sebeple, derin öğrenme temelli geliştirilen bina tespit algoritmasına yardımcı olarak, görüntüde fark edilmesi nispeten kolay ve keskin yapıdaki binaları tespit edecek yardımcı bir bölütleme algoritması modülü eklenmiştir.



Şekil 3.8 : testpatch4 ve testpatch9 görüntüleri için bulunan binalar

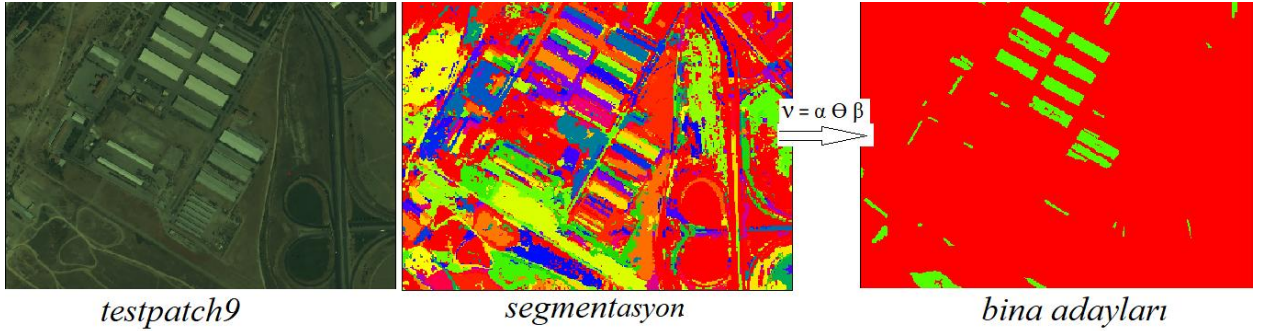
3.4 Bölütleme ve gölge doğrulama

Derin öğrenme modeline ek olarak görüntüde fark edilmesi nispeten kolay ve keskin yapıdaki binaları tespit edebilmek için ek bir bölütleme modülü bina tespit sürecine dahil edilmiştir. Yardımcı bölütleme modülü geliştirilmesinde *felzenszwalb* graf temelli bölütleme algoritması kullanılmaktadır [53]. RGB görüntüler ilk olarak graf temelli *felzenszwalb* algoritması ile analiz edilir ve segmentler çıkarılır. Bulunan her bir segmentin şekil bakımından özellikleri incelenerek keskin hatlara (insan yapımı yapılar genelde keskin hatlara sahip olduğu kabul edilerek) sahip segmentler bina adayları olarak seçilir. *Felzenszwalb* ve şekil doğrulama ile elde edilen bina adaylarına son olarak gölge

doğrulama işlemi uygulanır. Gölge doğrulama neticesinde elde edilen nihai sonuçlar, derin öğrenme modeliyle elde edilen sonuçlar ile birleştirilir.

testpatch9 görüntüsünün *felzenszwalb* algoritması ile elde edilen bölütleme sonucu şekil 3.9’da gösterilmiştir. Elde edilen her bir segment şekil bakımından incelenerek, keskin hatlara sahip olup olmamasına göre bina adayları belirlenir. Bir bölgenin bina adayı olarak seçilebilmesi için iki farklı kriter tanımlanır. Seçilen bölgeyi tamamen içerisine alacak bir konveks kapalı alan (H) tanımlanır ve seçilen bölgenin alanının (B_a) ilgili bölgeyi tamamen kapsayan konveksin alanına (H_a) oranlanmasıyla 0-1 arasında bir değer elde edilir ($\alpha = B_a / H_a$). Bir başka kriter olarak ise, hedef bölgeyi içerisine alabilecek minimum büyüklükte bir dikdörtgen hesaplanarak, bulunan dikdörtgenin uzunluğu (b_l) ve genişliği (b_w) oranlanır ($\beta = b_l / b_w$). Böylece dairesel alanlar için 1 değerini verecek olan, 0-1 arasında bir katsayı elde edilir. Bu kriter ile uzun ve ince yapıdaki düzgün şekillerin (özellikle yolların) bina adayı olarak seçilebilmesinin önüne geçilmiştir. Her iki kriter tüm segmentlere uygulandıktan sonra bina adayları çıkarılmış olur. (27) numaralı formülde Θ operatörü her iki kriterin aynı anda yerine getirilmesini, yani kesişim işlemini ifade etmektedir (**Denk.26**).

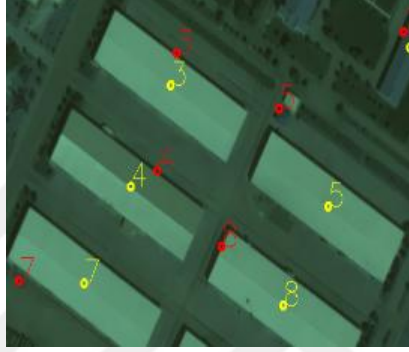
$$\alpha = B_a / H_a, \beta = b_l / b_w \quad v = \alpha \Theta \beta \quad (26)$$



Şekil 3.9 : testpatch9 bölütleme ve bina adaylarının bulunması

Bina adayları çıkarıldıktan sonra, her bir bina adayı için gölge doğrulama işlemi gerçekleştirilir. Gölge doğrulama süreci iki aşamadan oluşur. İlk olarak bina adayları kendisine en yakın gölgeler ile eşleştirilir. Bina adaylarını en yakın gölgeler ile eşleştirmek için $G_{yön}$ kriteri tanımlanır ve bu kritere uyan alanlar eşleştirilir: $E(B, G) = d_{min} \Theta G_{yön}$. $G_{yön}$ işlemi güneş açısına göre bina ve gölgelerin yön kontrolünü ifade

eder. d_{min} ise her bir bina adayına en yakın uzaklıktaki gölgelerin bulunmasını ve aradaki minimum mesafeyi ifade eder. Şekil 3.10'da kırmızı renkli alanlar tespit edilen binaların merkez noktalarını, sarı renkler ise bina adaylarına en yakın gölge alanlarının merkez noktasını göstermektedir. Eşleştirilen bina ve gölge alanları ise aynı numaralarla ifade edilmektedir.



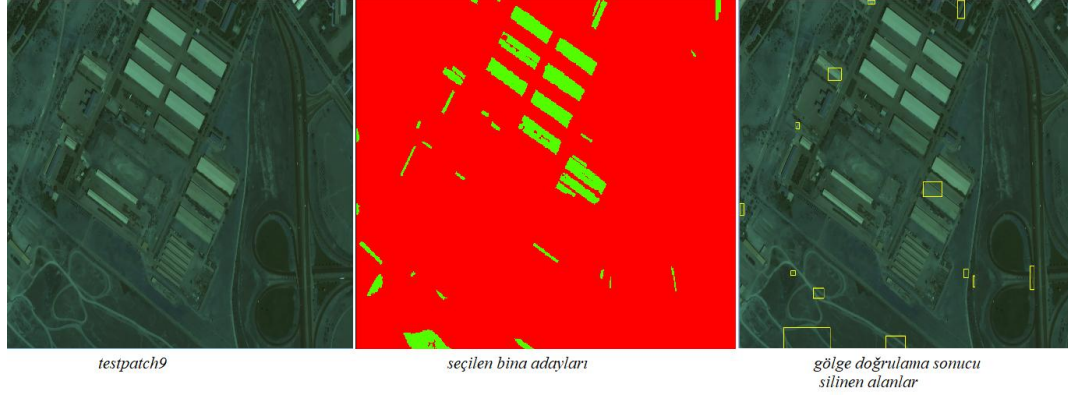
Şekil 3.10 : Gölge yönü ($G_{yön}$) ve mesafeye (d_{min}) göre gölge-bina eşleştirilmesi

$E(B,G)$ ile gölge ve bina adayları eşleştirildikten sonra, seçilen bina adayının doğrulanması için iki kriter kullanılır. Gölge alanı (G_a) ile bina adayları alanları (B_a) karşılaştırılır ve eğer gölge alanı bina alanından büyükse ilgili bina adayı elenir. İkinci kriter olarak ise gölge ile bina adayı arasındaki mesafenin belirli bir eşik değerden küçük olup olmadığı kontrol edilir. Böylece bina adayına yeterince yakın gölge bulunmaması durumunda ilgili bina adayı elenir. Burada kullanılacak eşik değer belirlenmesi için bir yöntem önerilmiştir. Buna göre bina adayını içerisine alan minimum dikdörtgenin kısa kenar uzunluğu hesaplanarak gölge ile bina bölgesi uzaklığı arasında olması gereken eşik değeri belirlenir. Şekil 3.11'de bulunan bazı hatalı bina adayları ve eşleştirildikleri gölgeler aynı sayı numaraları ile gösterilmektedir. Uygulanan gölge doğrulama kriterleri ile hatalı olarak seçilmiş 4, 5 ve 6 numaralı sarı renkli bina adayları elenmiştir.



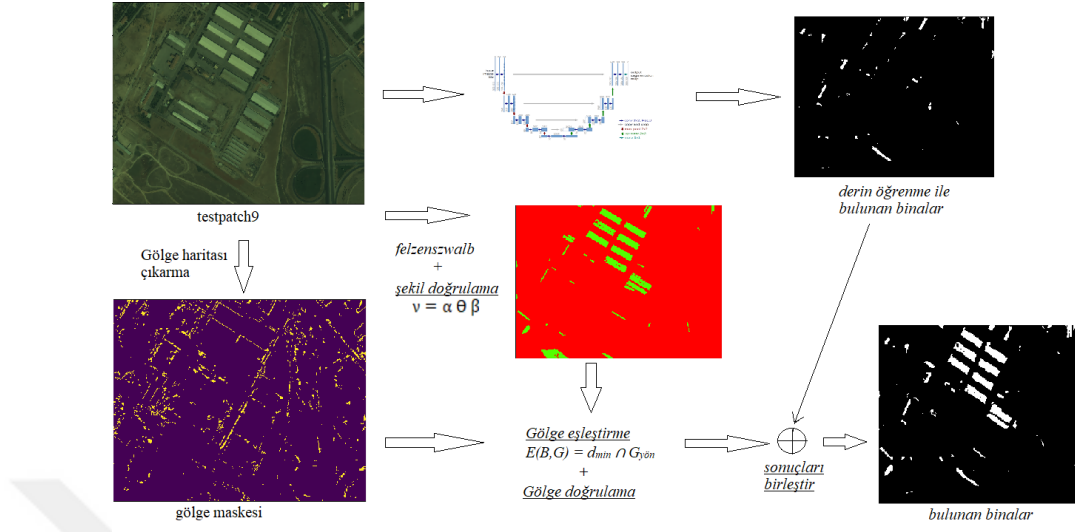
Şekil 3.11 : Hatalı bina adayları ve eşleştirildikleri gölge alanlar

testpatch9 görüntüsünün tamamına uygulanan gölge doğrulama sonucunda elenen bina adayları ise Şekil 3.12’de görülmektedir.



Şekil 3.12 : Gölge doğrulama ile hatalı bina adaylarının elenmesi

Böylece ilk bölütleme sonucunda elde edilen tüm alanlar içerisinde, şekil bazlı kontroller yapılarak ilk bina adaylarının belirlenmesi ve gölge doğrulama ile hatalı bina adaylarının elenmesi süreçlerinden oluşan bina tespit algoritması önerilmiştir. Bu yöntemle sadece şekil bakımından keskin hatlara sahip bina adayları gölge bilgisinin de yardımıyla tespit edilebilmektedir. Bu yöntem derin öğrenme ile birlikte kullanılarak önerilen bina tespit algoritmasına dahil edilmiş ve her iki yöntem ile elde edilen sonuçlar birleştirilerek algoritma performansı artırılmıştır. Şekil 3.13’te her iki yöntemin bir arada nasıl kullanıldığı ve algoritma adımlarının özet halinde gösterildiği grafikte paylaşılmıştır.



Şekil 3.13 : Gölge doğrulama ile bina adayları arasından eleme yapılması ve derin öğrenme modeli ile sonuçların birleştirilmesi

Buna göre; derin öğrenme modülü kullanılarak elde edilen sonuçlar ile sırasıyla gölge maskesi tespiti, *felzenszwalb* bölütleme, şekil doğrulama, gölge eşleştirme ve gölge doğrulama sonucu elde edilen sonuçlar birleştirilerek bina bölgeleri tespit edilmektedir. *model1-model2* ve *model-3* modelleri ile ek bölütleme ünün bir arada kullanılması suretiyle elde edilmiş bina tespit sonuçlarına dair F_1 skorları çizelge 3.6’da gösterilmiştir.

Çizelge 3.6 : Ek bölütleme algoritması ile elde edilen sonuçlar

No	model1 ve model2 (F_1)	(model1 – model2) + bölütleme	model3 (F_1)	(model3) + bölütleme
1	0.8324	0.8026	0.7326	0.6822
2	0.8218	0.8277	0.8464	0.8392
3	0.6948	0.7215	0.7454	0.7536
4	0.4047	0.4201	0.3562	0.3707
5	0.6517	0.6728	0.6131	0.6308
6	0.5989	0.5957	0.7818	0.7535
7	0.6409	0.6551	0.8337	0.8278
8	0.4958	0.5292	0.7635	0.7646
9	0.0904	0.6617	0.1666	0.6616
10	0.5533	0.5940	0.5447	0.5747
11	0.6920	0.6990	0.8350	0.8429
12	0.4111	0.4426	0.7480	0.7579
13	0.3893	0.4302	0.6258	0.6490
14	0.4441	0.4701	0.7342	0.7394

Derin öğrenme modellerinin yetersiz kaldığı 9, 12 ve 13 numaralı görüntülerde, önerilen klasik görüntü işleme yöntemleri temelli algoritma ile performans iyileşmesinin gerçekleştiği görülmektedir.

Elde edilen sonuçlar, literatürdeki farklı yöntemler ile karşılaştırılmıştır. MPP [54] algoritmasında dikdörtgen binalar renk ve gölge bilgileri kullanılarak bulunmuştur. GrabCut [39] algoritmasında ilk olarak bina olmaya aday yerler gölge bilgisi ile belirlendikten sonra bulunan tüm alanlar ile gölge alanları arasındaki ilişkiler için kısıtlamalar belirlenir. Son olarak ise tüm bu evrelere ek olarak CRF optimizasyon algoritması kullanılarak en son karar aşamasında piksel seviyesinde sınıflama yapılır. MLP [40] algoritmasında da yine gölge bilgisi, gölge genişletme, gölge-bina ilişkisi ve son olarak iki aşamalı sınıflama evrelerinden sonra bina alanları tespit edilir.

Çizelge 3.7 : Farklı yöntemlerin Piksel tabanlı F_1 skorları

No	MPP	Grabcut	MLP	model(1-2)	model-3
1	40.8	58.8	44.4	80.26	68.22
2	53.2	58.5	77.8	82.77	83.92
3	49.4	67.4	72.1	72.15	75.36
4	37.6	59.3	62.3	42.01	37.07
5	59.2	66.2	78.7	67.28	63.08
6	49.1	58.7	36.1	59.57	75.35
7	62.2	80.3	81.9	65.51	82.78
8	49.8	70.8	76.4	52.92	76.46
9	48.5	53.9	52.6	66.17	66.16
10	37.1	42.8	31.3	59.40	57.47
11	58.4	79.6	86.0	69.90	84.29
12	62.9	25.2	36.1	44.26	75.79
13	63.9	57.0	75.7	43.02	64.90
14	71.2	50.5	68.8	47.01	73.94

Çizelge 3.7’de her bir görüntü için en iyi performansa sahip algoritmanın doğruluk değerleri kalın rakamlarla ifade edilmiştir. Sonuçlar karşılaştırıldığında önerilen yöntemin mevcut yöntemlere nazaran ortalamaya yakın veya üstünde performans gösterdiği görülmektedir. Önerdiğimiz yöntemde literatürdeki yöntemlerde önışlem olarak uygulanan çok aşamalı gölge iyileştirme ve gölge doğrulama neticesinde yine çok aşamalı sınıflama süreçleri uygulanmamıştır. Özellikle derin öğrenme modelinin

karmařık yapıdaki řehir görüntülerinde özellik çıkarma konusundaki başarısı göze çarpmaktadır. Az sayıda görüntü kullanılarak eğitilen modellerin başarısı uzmanlık ve detaylı analiz gerektiren, farklı görüntüler için deęişkenlik gösterecek özellik çıkarma ve sınıflama süreçlerinin yerini alması bakımından önemlidir.



4. BİNA BÖLÜTLEMESİ İÇİN ÖNERİLEN DERİN ÖĞRENME MİMARİLERİ

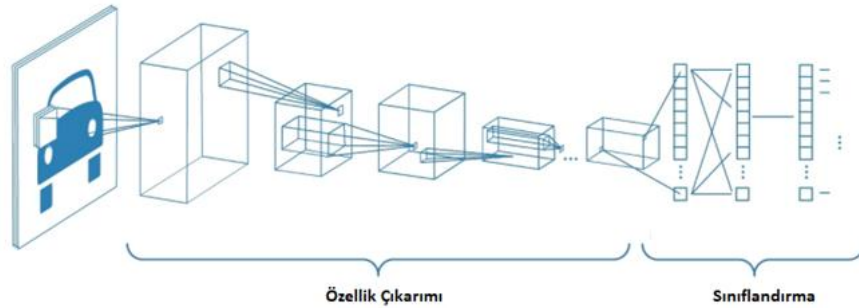
Yapay zekâ, insan zekasının karakteristiğini yani anlama, planlama, etrafını algılama ve problem çözme gibi işlevleri yerine getirebilen makinelerdir. Bu tanım ilk olarak 1956 yılınca Dartmouth konferansında John MacCarthy tarafından dile getirilmiştir. İlk yapay zekâ laboratuvarı ise 1959 yılında MIT'de kurulmuştur. Yapay zekâ en üst çerçeveden bakılacak olursa genel ve kısıtlı olarak ikiye ayrılabilir. Genel yapay zekâ, insan zekasına bir nevi eşdeğer olan yani insan zekasının tüm karakteristik özelliklerini barındıran yapay zekadır. Kısıtlı yapay zekanın ise sadece belirli alanlarda özelleşmiş yapay zekâ sistemleri olduğu düşünülebilir. Bir sistemin yapay zekâ diye adlandırılabilmesi için çeşitli girdilerden öğrenme yaparak kendini geliştirme ve öğrendikleri arasında bağ kurabilmesi gerekmektedir. Belirli girdiler karşısında belirli sonuçları veren sistemlere yapay zekadan ziyade otonom sistemler denebilir.

Yapay sinir ağları ise insan beynindeki nöronların yapısından ve birbirleri ile iletişiminden esinlenilerek tasarlanmıştır. Yapay sinir ağına ilk örnek olarak literatüre geçen perceptron modeli/algoritması 1957 yılında Frank Rosenblatt tarafından ortaya atılmıştır [55]. 1969 yılında çok katmanlı mimari ve 1986 yılında geriye yayılım (backpropagation) teorisi ile YSA modellerinin eğitilmesi çalışmaları gerçekleştirilmiştir. 1998 yılında Yann Lecun tarafından önerilen 7 katmanlı CNN modeli ile 1'den 9'a kadar rakamların sınıflandırılması problemine bir çözüm önerilmiştir. Bu çalışma sinir ağları ile başarılı sonuç alınan ilk çalışmalardan birisi olmuş ve literatüre LeNet modeli diye geçmiştir [56]. Makine öğrenmesinde kullanılan yöntemlerden birisi olan sinir ağlarının geçmişi yıllar öncesine dayanmasına rağmen, yine aslında sinir ağları temelli olan derin öğrenme yöntemleri ise son zamanlarda popülerlik kazanmıştır. Derin öğrenmenin özellikle bilgisayarlı görme alanında popülerlik kazanmasında önemli rol oynayan olaylardan birisi ImageNet yarışmalarında elde edilen sonuçlar olmuştur. ImageNet 22 bin farklı kategoriye sahip ve 15 milyon görüntü etiketlenerek oluşturulan çok büyük bir veri setidir [57]. Buradaki görüntüler üzerinde 2010 yılından itibaren nesne tanıma yarışmaları düzenlenmiş ve 2012 yılındaki yarışmada AlexNet ismiyle literatüre geçen derin öğrenme temelli yöntem son derece başarılı sonuçlar vermiştir [58]. Bu zamana kadar %72-74 seviyelerinde olan doğruluk

oranının aniden %84 değerine ulaşması ile birlikte, derin öğrenme yöntemleri üzerine ilgi artmaya başlamıştır. Yarışmanın devam eden yıllarında ise farklı derin öğrenme modelleri önerilmiş (ZfNet [59], GoogleNet [60], VGG [61], ResNet [62]) ve nesne tanıma probleminde elde edilen başarı oranları artmıştır. Derin öğrenme yöntemlerinin başarısının yanı sıra yaygınlaşmasının sebepleri olarak şunlar da sıralanabilir; eğitim işleminde gerekli olan büyük veri setlerinin oluşturulması, büyük veriyi işleyebilecek kapasitede işlemci gücünün sağlanması, model eğitime özgü çoklu işlemciye sahip donanımların üretilmesi ve derin öğrenme modeli eğitilmesine olanak sağlayan açık kaynak farklı kütüphanelerin yaygınlaşması.

4.1 Görüntü işleme problemlerinde derin öğrenme yaklaşımı

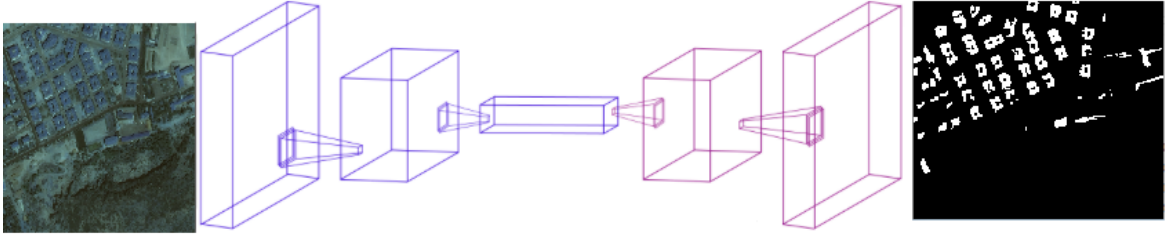
Nesne tanıma yaygın kullanılan modeller de, bir önceki bölümde kullandığımız unet modeline benzer olarak, giriş aşamasında bir dizi katmanlar ile görüntü üzerinde konvolüsyon işlemleri uygulayarak özellik çıkarımı yapmakta ve aynı zamanda görüntünün boyutunu düşürmektedirler. Özellik çıkarımının ardından ise sınıflama işlemi için klasik sinir ağı katmanları kullanılmaktadır. Sınıflama amaçlı tasarlanan klasik CNN modeli şu şekildedir (Şekil 4.1) [63]



Şekil 4.1 : Nesne tanıma amaçlı tasarlanmış klasik CNN modeli [62]

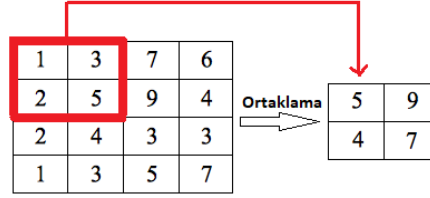
Unet mimarisinden bahsettiğimiz önceki bölümde CNN tipinde bir ağ modeli olan Unet ile bina bölütleme için model eğitimi işlemi gerçekleştirilmiş ve klasik görüntü işleme yöntemleri ile elde edilen sonuçlar arasında performans kıyaslaması yapılmıştır. Bu bölümde ise uydu görüntülerinde semantik bölütleme ve bina tespitine yönelik gerçekleştirilen farklı derin öğrenme yöntemleri daha detaylı incelenmiştir. Önceki

bölümde eğitilen Unet modelinde ilk olarak girdi görüntüsüne konvolüsyon ve ortaklama katmanları ile bir dizi işlem uygulanarak çıkışta bir vektör elde edilmektedir. Elde edilen bu vektöre, çıkışta tekrar girdi ile aynı boyuttaki bir görüntüye dönüştürmek için yine bir dizi konvolüsyon ve büyütme işlemleri uygulanır. Sonuçta görüntüde her bir piksele bir etiket değeri atanmış olur. Bölütleme işleminde sınıflamadan farklı olarak görüntüdeki her bir piksel değeri için bir karar verilmiş olur. Yani bir görüntüde aynı nesneye ait farklı bölgeler/alanlar için aynı etiket değeri atanması amaçlanmaktadır. 1 ve 3 numaralı görüntülerdeki ağ modellerinin örnek çıktılarından da anlaşılacağı gibi, bölütleme amaçlı tasarlanan sinir ağlarının ürettiği sonuçlar, sınıflama amaçlı tasarlanan modellerin aksine 3 boyutludur. Bu sebeple sınıflama problemlerinde yaygın kullanılan ve iyi sonuç veren ağ mimarilerinin son katmanları kaldırılarak tek bir sınıf değeri yerine bölütleme haritası çıktısı alınması sağlanmaktadır. Bölütleme amaçlı tasarlanan bir ağ yapısı, genelde aşağıdaki grafikte gösterilen üç kısımdan meydana gelmektedir. İlk olarak görüntünün boyutu düşürülür, ardından tam bağlı bir sinir ağına iletilir ve sonuçta girdi boyutu ile aynı boyutlarda bölütleme sonucu alınabilmesi için tekrar boyut artırılır (Şekil 4.2).



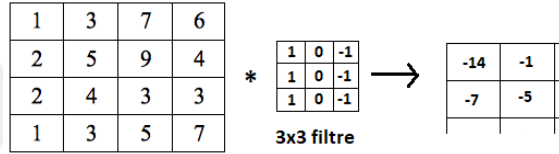
Şekil 4.2 : Bölütleme amaçlı tasarlanmış CNN modeli

Sinir ağlarında görüntünün boyutunun düşürülmesi için iki farklı yöntem tercih edilmektedir. İlk yöntemde, *ortaklama (pooling)* işlemi ile görüntüde bir pencere dolaştırılarak, dolaştırılan pencere içerisinde kalan piksellerin maksimum olanı veya ortalaması seçilebilir (Şekil 4.3).



Şekil 4.3 : 2x2 boyutlu pencere ve 2x2 kaydırma adımları ile ortaklama işlemi

Alternatif olarak diğer yöntemde ise, konvolüsyon işleminde görüntünün üzerinde gezdirilen pencerenin atlayacağı adım sayısına göre konvolüsyon sonucu elde edilen yeni görüntünün boyutu düşürülebilir (Şekil 4.4).

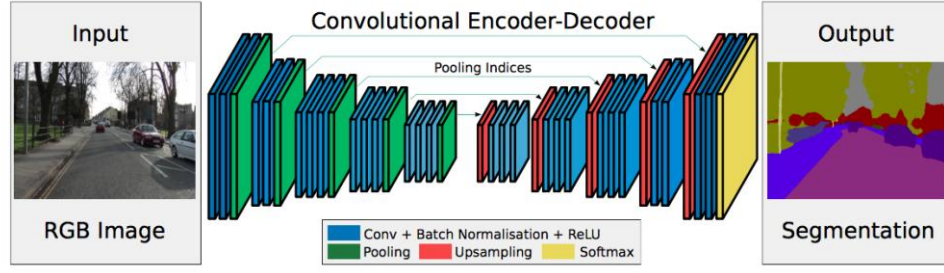


Şekil 4.4 : 3x3 boyutlu pencere ile konvolüsyon işlemi

Eğer konvolüsyon işlemi sonucunda boyutun aynı kalması isteniyorsa, uygulanan filtre boyutuna göre görüntünün kenarlarına piksel ekleme işlemi yapılmalıdır. Görüntünün kenarlarına sabit değerler veya kenar piksel değerleri ile aynı değerler eklenebilir. Filtre sonucunda boyutların sabit kalması için, piksel kaydırma adımı 1 olarak seçilmelidir. Orijinal görüntü boyutu (n), uygulanan filtre boyutu (f), piksel kaydırma adımı (s) ve kenarlara eklenen piksel sayılarına (p) göre, konvolüsyon sonucu elde edilen boyutlar şu şekilde hesaplanır (**Denk.27**);

$$\frac{n+2p-f}{s} - 1 \quad (27)$$

Bu işlemler ile uzamsal çözünürlüğün düşürülmesiyle hem işlem gücü hem de gereksinim duyulan hafıza miktarı düşürülür ve görüntüden özellik çıkarımı gerçekleştirilmiş olur. SegNet ve Unet modelleri benzer mantığa sahip, popüler bölütleme modellerindendir. SegNet modeli sadece konvolüsyon katmanlarından oluşmaktadır ve çıkış katmanında *softmax* sınıflandırıcı ile piksel bazında sınıflama işlemi gerçekleştirir (Şekil 4.5).



Şekil 4.5 : SegNet mimarisi [64]

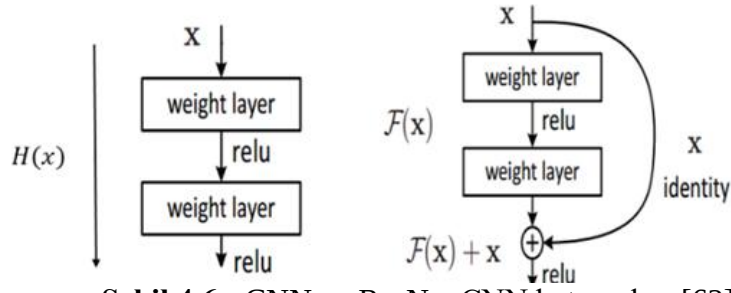
Nesne tanımda yaygın kullanılan bir başka ağ mimarisi VGG ise, 2014 yılında Simonyan ve Zisserman tarafından önerilmiş ve ImageNet veri seti üzerinde gerçekleştirilen yarışmada başarılı sonuçlar vererek literatüre geçmiştir [61]. AlexNet mimarisine benzer olarak 3x3 kernel boyutunda birçok filtre içermektedir. Konvolüsyon işlemlerinin ardından her biri 4096 düğüme sahip iki sinir ağı katmanı daha bulunmaktadır. VGG ağı 4 GPU üzerinde yaklaşık 3 haftalık süre boyunca eğitilmiştir. VGG modelinin eğitim sonucu elde edilen ağırlık katsayıları genel erişime açılmıştır. Dolayısıyla görüntülerden özellik çıkarımı için en çok tercih edilen yöntemlerden birisi haline gelmiştir. Toplamda 133 ile 144 milyon parametre içeren versiyonları mevcuttur.

Çizelge 4.1 : Farklı katmanlar içeren VGG mimari detayları [61]

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224x224 RGB images)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
Soft-max					

VGG modeli eğitimi kolaylaştırmak için ilk olarak daha az ağırlık katmanına sahip olarak tasarlanmıştır (çizelgede A ve C sütunları arası). VGG16 ve VGG19 isimleriyle literatüre geçen modeller ise, ilk aşamada tasarlanan nispeten küçük modellerin üzerine inşa edilmiş çok daha derin yapıdaki sinir ağıdır. Bu ağ yapısının en büyük dezavantajı eğitim işleminin aşırı zaman alıcı olmasıdır.

Sinir ağında derinliğin yani katman sayısının artması sonuç üzerinde son derece etkilidir. Teorik olarak katman sayısının artmasının daha iyi sonuçlar vereceği düşünülebilir, fakat bu durum ağda eğitilecek parametre sayısını arttıracığından optimum çözüm bulma işlemini daha da zorlaştıracaktır. Ayrıca ağın eğitilmesi sırasında bazı nöronların zamanla neredeyse etkisiz kaldığı ve ağın eğitime herhangi bir katkı veremediği de yapılan denemelerde görülmüştür. Bu sebeple ağın daha derin olması her zaman daha iyi performans vereceği anlamına gelmez. Bu noktada da yine optimum bir derinlik söz konusudur [65]. Her ne kadar derin sinir ağlarının eğitimi aşırı işlem gücü gerektirip zaman alıcı olsa da, görüntü işleme problemlerinin çözümünde tercih edilmektedir. Ayrıca önerilen yeni yöntemler ile bazı dezavantajlar zamanla ortadan kaldırılmaktadır. VGG modelinin eğitim süresi dezavantajını ortadan kaldıran ve aynı zamanda ImageNet yarışmasının 2015 yılının kazanan yeni modeli ResNet mimarisidir. Bu mimari art arda gelen katmanlardan oluşan VGG modelinden daha farklı bir mimariye sahiptir. ResNet mimarisinin getirdiği en büyük yenilik ağın derinliğini arttırırken işlem yükünün aşırı artması gibi yan etkileri giderebilmesidir. ResNet mimarisinin optimize edilmesi ve dolayısıyla ağın eğitimi daha kolaydır [62]. ResNet mimarisine bu özeliği getiren en önemli yenilik, geçmiş katmanlardaki bilgilerin sonraki katmanların çıkışlarına direk olarak bağlanmasıdır. Bu zamana kadar ki önerilen mimarilere nazaran getirilen bu yenilik, aslında çok küçük bir değişiklik olmakla birlikte sonuçları açısından çok önemlidir. Klasik CNN katmanları ile ResNet ile önerilen CNN katmanları Şekil 4.6'daki gibidir.



Şekil 4.6 : CNN ve ResNet CNN katmanları [62]

Klasik CNN katmanlarında, giriş art arda gelen katmanlardan geçirilip çıkışa aktarılmaktadır. *Residual* bağlantılı CNN de ise, giriş aynı zamanda 2. katmandan elde edilen çıkışa eklendikten sonra *relu* aktivasyon fonksiyonundan geçirilmektedir. Böylece geçmiş katmandaki bilgilerin sonraki katmanlara daha güçlü bir şekilde aktarılması hedeflenir. Bu tez ile gerçekleştirilen denemelerde, her iki yöntem sonucunda da aynı değerlere ulaşılsa da ağı eğitiminin 2. yöntemde daha kolay olduğu tespit edilmiştir. ResNet mimarisinin de tıpkı VGG de olduğu gibi farklı katman sayısına sahip versiyonları bulunmaktadır (Şekil 4.7).

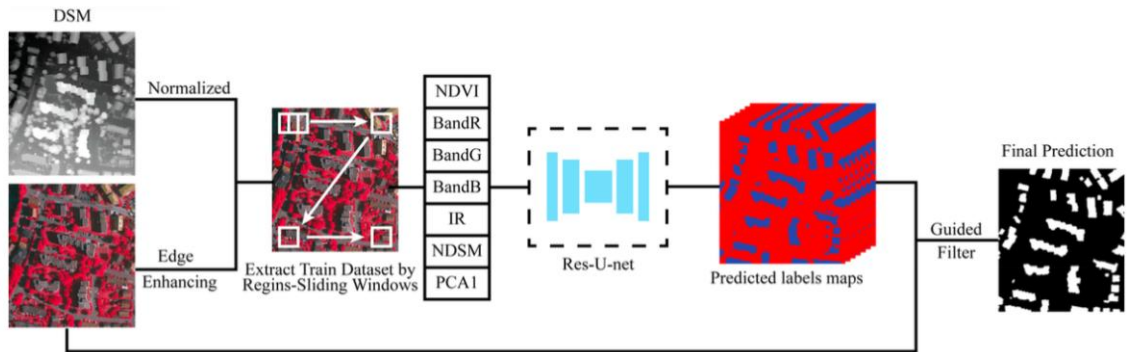
layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Şekil 4.7 : Farklı katman sayılarına sahip ResNet mimarileri [62]

ResNet mimarisinin 18 ve 34 katman içeren versiyonlarında 2 katmanlı bloklar tercih edilirken, 50, 101 ve 152 katman içeren daha derin versiyonlarında ise 3 katmanlı bloklar tercih edilmektedir. Bu kapsamda geliştirilen 152 katmanlı ResNet mimarisi, önerildiği zaman itibarıyla ImageNet veri seti üzerinde test edilen en derin ağ modelidir.

4.2 Derin öğrenme ile bina bölütlemesi için literatür taraması

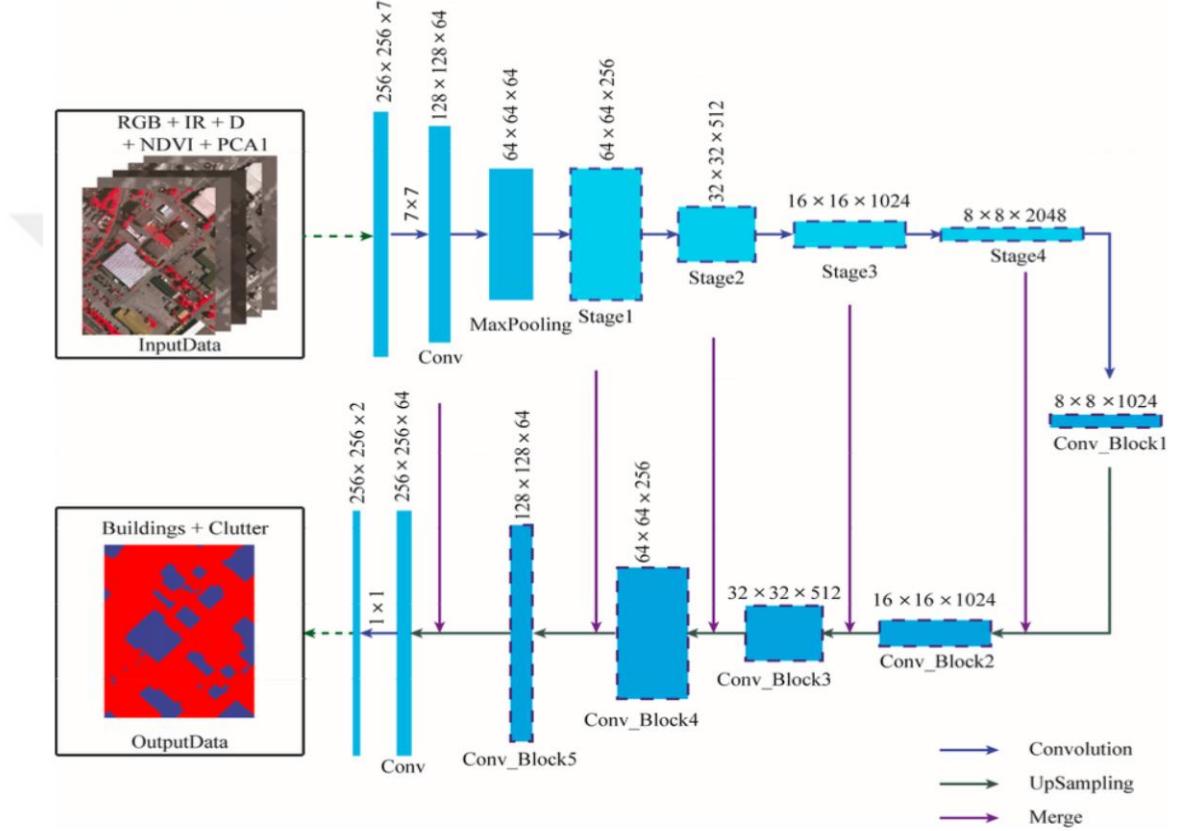
Yapay sinir ağı temelli derin öğrenme yöntemleri nesne tespitinde başarılı sonuçlar verdikten sonra, bölütleme gibi farklı görüntü işleme problemlerinde yaygın kullanılmaya başlanmıştır. Yongyang ve arkadaşları tarafından 2018 yılında yayınlanan bir makalede çok yüksek çözünürlüklü görüntülerden binaların çıkarılması için piksel tabanlı bir sınıflama yöntemi önerilmiştir [66]. İlk olarak görüntülere bir ön işlem uygulanarak kenarların daha belirginleştirilmesi sağlanmıştır. Ayrıca kullanılan çok bantlı görüntüden bitkisel alanları ifade eden NDVI, normalize edilmiş yüzey modeli (NDSM) ve temel bileşen analizi uygulanarak elde edilen en yüksek değerliğe sahip bileşen gibi bazı özellikler çıkarılmıştır. Ardından tüm bu görüntüler ResUnet diye adlandırılan, Unet modelinin modifiye edilmiş hali olan bir sinir ağına girdi olarak verilmiştir. Sinir ağının çıktısı olarak, piksel tabanlı sonuçları içeren bir bölütleme haritası elde edilmiştir. Model sonucunda elde edilen etiket değerlerine bir filtreleme işlemi uygulanmıştır. Bu şekilde sonuçların daha iyileştirilmesi amaçlanmıştır. Sinir ağının çıktısından elde edilen nesne (bina) sınırlarının tam net olmadığı, bulanık olduğunun tespiti üzerine algoritmaya böyle bir süreç eklendiği belirtilmiştir. Önerilen yöntemin genel görünümü şu şekildedir (Şekil 4.8).



Şekil 4.8 : Çok bantlı bina bölütleme yöntemi [66]

Önerilen yöntemde ResNet modelinin modifiye edilmiş ve genişletilmiş hali kullanılmış ve modelin girdi görüntüsü olarak 7 band ayarlanmıştır. Önerilen model iki aşamalı olarak düşünüldüğünde, ilk aşamada görüntüden özelliklerin çıkarılması kısmında sınıflandırma problemlerinde yaygın kullanılan ResNet modelinden esinlenilmiş girdi

sayısı 7 band için modifiye edilmiştir. Özellik çıkarımının devamındaki katmanlarda, normalizasyon işlemi ve görüntünün boyutunun düşürülmesi için 2x2 boyutunda kayan pencere ile filtreleme işlemleri uygulanmıştır. Ağ mimarisinin detayları Şekil 4.9'da gösterilmiştir.



Şekil 4.9 : Bina bölütleme için önerilen mimarinin detayları [66]

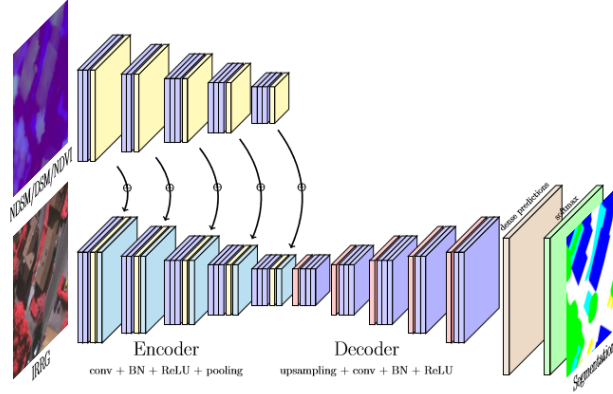
ISPRS yarışmaları kapsamında yayınlanan çok bantlı veri setleri üzerinden gerçekleştirilen bir başka çalışmada [67], yine ResNet modelinden esinlenilerek bir ağ tasarımı gerçekleştirilmiş ve aynı zamanda ResNet modeli için ImageNet nesne tanıma veri setinden elde edilen ağırlık katsayıları kullanılmıştır. Tasarlanan modelin eğitilmesi için Adam [68] optimizasyon yöntemi kullanılmış ve ağ girdisi olarak yeşil, kırmızı, mavi bantlara ek olarak derinlik bilgisi, NDVI ve kızıl ötesi bantları kullanılmıştır. Bu çalışmada girdi boyutu olarak 256x256 seçilmiş ve NVIDIA Tesla K80 GPU ile eğitim işlemi gerçekleştirilmiştir. Orijinal ResNet modeli 3 bantlı görüntüler için tasarlandığından, ResNet modelinin elde edilen ağırlık katsayılarını kullanarak sadece kızıl ötesi, kırmızı ve yeşil bantlar ile bir başka model eğitimi daha gerçekleştirilmiştir.

İlgili makalede ikinci modelin ilkinde göre daha kötü performans sergilemesinin beklendiği belirtilmektedir. Bunu sebebi ise önemli bilgiler içerdiği düşünülen derinlik ve NDVI bantlarının kullanılmamış olmasıdır. Fakat şaşırtıcı derecede bir önceki 6 bant ile eğitilen modele kıyasla %2.4 performans artışının olduğu gözlemlenmiştir. Biyomedikal görüntülerin kıyaslanmasında iyi sonuçlar veren, gölge doğrulama ile birlikte hali hazırdaki birçok klasik görüntü işleme temelli yöntemle göre başarılı sonuçlar veren Unet modeli de yine bu çalışmada kıyaslama için kullanılmıştır. Unet modeli bu çalışmada sıfırdan eğitilen modele göre daha iyi sonuçlar vermekle birlikte, ResNet modelinin eğitilmiş katsayılarının kullanıldığı modele göre performans olarak çok az da olsa geride kalmıştır.

ResNet modelinin hali hazırdaki ağırlık katsayılarını daha efektif kullanabilmek için bir başka ağ yapısı daha bu çalışmada test edilmiştir. Buna göre iki farklı ResNet tabanlı model kurulmuş ve her bir modele sırasıyla kızıl ötesi, kırmızı, yeşil ve derinlik, NDVI ve mavi bantlar girdi olarak verilmiştir. Gerçekleştirilen bu denemelerde kayda değer bir performans artışı sağlanamadığı tespit edilmiştir. Bunun sebebi olarak NDVI ve mavi bantların gereksiz olabileceği ve derinlik bilgisinde hataların olabileceği ihtimali üzerinde durulmuştur. NDVI bilgisi zaten kızıl ötesi ve kırmızı bantların bir araya gelmesinden oluştuğundan, ağın kendisinin bu ilişkiye ortaya çıkarabileceği düşünülebilir. NDVI bilgisinin etkisini tam olarak ortaya çıkarabilmek için farklı gözlemler yapılmış, bu kapsamda kırmızı, yeşil ve kızıl ötesi bantları farklı kombinasyonlarda bir araya getirilerek her birinin yanında NDVI bilgisi ile birlikte farklı ağ modelleri eğitilmiş ve sonuçlar karşılaştırılmıştır. Tüm bu gözlemler sonucunda NDVI bilgisinin girdi olarak kullanılmasının herhangi bir avantaj sağlamadığı sonucuna varılmıştır. Veri setinin elde edildiği dönemin kış dönemine geldiği belirtilmiş ve ağaçların yapraklarının bu dönemde çorak olmasının da bu sonuçlarda etkili olabileceği de belirtilmiştir.

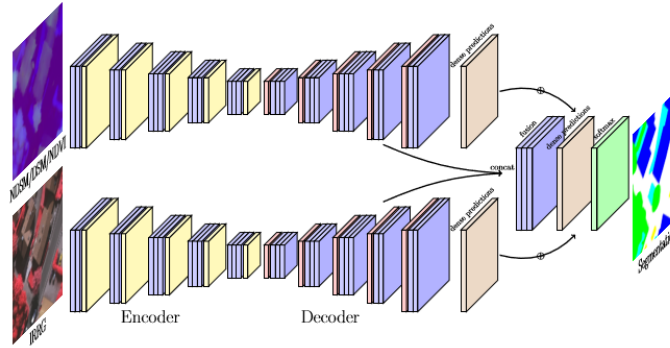
Audebert ve arkadaşları tarafından gerçekleştirilen bir çalışmada ise SegNet ve ResNet modelleri üzerine inşa edilen mimariler bina bölütleme için kullanılmıştır [69]. Önerilen mimaride 3 band görüntüleri girdi alan SegNet mimarisine görüntü birleştirme ismi verilen katmanlar eklenerek NDVI ve derinlik gibi farklı bilgileri de işleyebilmesi sağlamışlardır. Birleştirme işleminin ilk katmanlarda yapıp ardından bölütleme

katmanlarına geçildiği yönteme ek olarak, SegNet mimarisinin son katmanlarında birleştirme işleminin yapıldığı bir başka yöntem daha test edilmiştir. FuseNet diye adlandırılan bu yöntemler ayrıca ResNet mimarisinde kullanılan sonraki katmana bilgilerin aktarıldığı bağlantı yöntemi ile de test edilmiştir. Önerilen modellerin performansı Vaihingen ve Postdam veri setleri ile test edilmiştir [70].



Şekil 4.10 : FuseNet mimarisi [69]

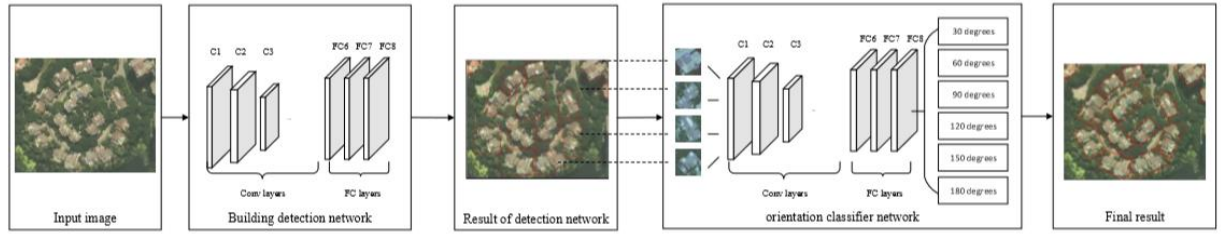
Birleştirme işleminin son katmanlarda gerçekleştirildiği mimari ise iki SegNet modelinden oluşmaktadır. ResNet mimarisindeki bağlantı yönteminin kullanıldığı bu mimarinin detayları Şekil 11’de gösterilmiştir. Aynı mimari ile SegNet yerine ResNet katmanlarının kullanılması ile bir başka model daha elde edilmiştir.



Şekil 4.11 : 2 SegNet mimarilerinden oluşan bölütleme modeli [69]

Bölütleme amaçlı tasarlanan mimarilerin aksine, nesne tanıma yapan bir ağ mimarisi olan R-CNN modeli Chen ve arkadaşları tarafından gerçekleştirilen çalışmada modifiye edilerek bina tespiti için kullanılmıştır [71]. Nesne tanıma amaçlı tasarlanan ağ mimarilerinin çıktıları, hedef nesneyi içerisine alan bir kapalı alan bulmak ve nesnenin

ne olduğunu tahmin etmektir. R-CNN temelli önerilen bu yöntemde, diğer bina bölütleme yöntemlerinin aksine binaları içerisine alan dikdörtgenler bulunur. Bu yönüyle bölütleme amaçlı tasarlanan modellerden farklıdır. Fakat binaları içerisine alan dikdörtgen bulunması, binaların sınırlarını tam olarak ifade edemeyeceğinden yetersiz kalmaktadır. Ayrıca binanın yönünün de ihmal edilmesi çok büyük bir eksikliktir. Bu sebeple, binaları çevreleyen kapalı alanlar bulunduktan sonra, binanın yönünün bulunması amacıyla ikinci bir sınıflama katmanı daha önerilmiştir. Böylece nesne tanımda başarılı sonuçlar veren R-CNN modelinden bina tanıma probleminin çözümünde de faydalanılmıştır. Önerilen bu yöntem VGG-Net R-CNN olarak isimlendirilmiştir. Model bina tanıma ve binanın yönünü tayin etme kısmı olmak üzere iki farklı aşamadan/modelden oluşmaktadır [71]. Ağ girdisi ise 224x224 boyutu ile sınırlandırılmıştır. Her iki model de 13 evrişimsel katman ve 3 tam bağlı katman içermektedir (Şekil 4.12).



Şekil 4.12 : İki alt model içeren bina tespit modeli [71]

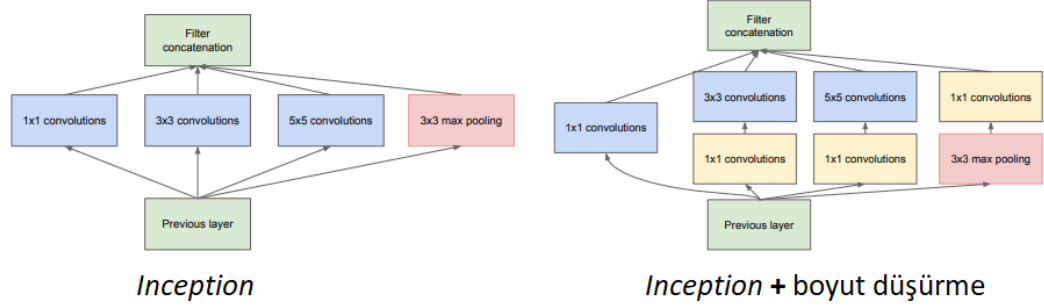
Girdi olarak bir görüntü modele verildiğinde, ilk olarak bina tespit ağı tarafından bina adayları bulunur. Ardından seçilen binalar yön tespit modeline girdi olarak verilir. Yön tespit modelinde binanın yönü için, 30-60-90-120-150-180 olmak üzere 6 farklı açı için sınıflandırma yapılır. Tahmin edilen bina açı değerine göre, bina tespiti sonucunda bulunan dikdörtgen ilgili açı değeri baz alınarak döndürülür. Böylece bina sınırlarına daha net ifade eden bir kapalı alan bulunmaya çalışılır. Bu modelde binaları içerisine alan dikdörtgenler ve açılarının bulunması, dikdörtgen şeklinde olmayan yapılarda son derece dezavantaj doğurabilmektedir. Ayrıca VGG gibi aşırı işlem yükü gerektiren iki farklı modelin eğitilmesi de, eğitim için gereken işlem yükünü son derece arttırmaktadır. İlgili makalede de [71], bu dezavantajlara değinilmiş ve iki alt modelden oluşan bu

yöntemin en azından tek bir modelde birleştirilmesi yönünde çalışmalar gerçekleştirileceği belirtilmiştir.

Nesne tanıma amaçlı tasarlanan mimarilere genişlik kazandırılması için önerilen Inception blokları, bölütleme için literatürde tasarlanan mimarilere de adapte edilmiştir. Bu kapsamda gerçekleştirilen farklı çalışmalarda, bölütlemeye yaygın kullanılan Unet mimarisi Inception blokları ile güncellenmiştir [72,73]. Inception bloklarının kullanıldığı bu çalışmalar genelde biyomedikal görüntüler üzerinde gerçekleştirilmiştir. Tez kapsamında ilk olarak bu yöntemlere benzer şekilde farklı Inception blokları ile tasarlanan mimariler test edilmiştir. Ayrıca Inception Unet mimarisinde revizyonlar gerçekleştirilerek daha güçlü bir derin öğrenme mimarisi önerilmiştir.

4.3 Inception Unet mimarisi

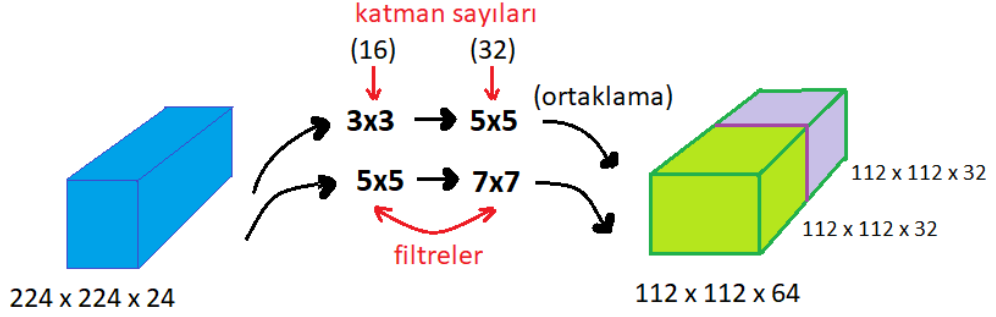
ResNet modelinde ortaya atılan yeni yaklaşımla daha derin yapıda modellerin elde edilebilmektedir. Unet mimarisinde ResNet ile ortaya atılan bağlantı şekli bölütleme problemi için de kullanılmıştır. Bölütlemeye son derece başarılı olan bu mimarinin bazı türevleri de bölütleme problemi için kullanılmıştır. Literatür kısmında açıklanan çeşitli çalışmalarda çok bantlı girdiler kabul eden veya farklı bantları paralel katmanlarda işleyen Unet versiyonlarından bahsedilmiştir. Unet görüntüyü sıkıştıran (özellik çıkarımı yapan) ve orijinal boyutlardaki maske görüntüsünü oluşturan bir mimari olduğundan, ağın derinliğinin daha fazla arttırılması ve görüntünün daha fazla sıkıştırılması performansı düşürmektedir. Bu kapsamda ilk olarak Unet mimarisindeki eğitilecek parametre sayısının arttırılması, bunu yaparken de ağın derinlemesine değil genişliğinin arttırılması üzerinde durulmuştur. Ağların derinlemesine büyümesinin yanı sıra, genişliğinin arttırılmasının hedeflendiği ağ mimarisine literatürde *Inception* diye adlandırılan bir yaklaşımla ile geçilmiştir [60]. Bu yaklaşımda modelin genişliğini arttırabilmek için *Inception* adı verilen bloklar kullanılmıştır (Şekil 4.13).



Şekil 4.13 : Inception modülleri [60]

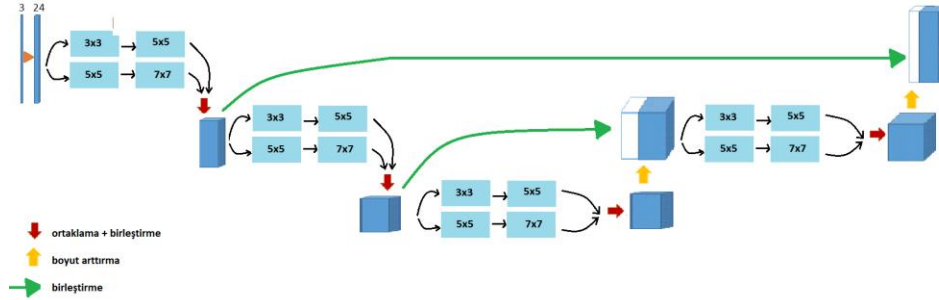
Inception modüllerinden 1x1, 3x3, 5x5 ve 3x3 maksimum ortaklama işlemleri içeren modül naif modül olarak adlandırılmaktadır. Şekil 4.13'te gösterilen paralel katmanlar ile şimdiye kadar uygulanan yöntemlerden farklı olarak, katmanlar arasında veri aktarılırken filtreleme işlemi farklı pencere boyutunda ve farklı değerlere sahip filtreler uygulanarak gerçekleştirilmektedir. Bu durum işlem yükünü artırırken, aynı verinin farklı filtrelerden geçirilerek aktarılması gibi bir ekstra durumu da beraberinde getirmektedir. Ayrıca göze çarpan bir başka dikkat çekici nokta ise 1x1 boyutunda evrişim katmanı kullanılmasıdır. Ayrıca, 1x1 boyutundaki filtre, görüntüdeki kanal sayısını (özellik sayısını) azaltmak yani boyut düşürmek için kullanılmaktadır.

Bu kapsamda ilk olarak Unet mimarisinin bölütleme problemindeki başarısını arttırmak için farklı katman sayısına ve içeriğine sahip alt modüller ile Unet mimarisinin genişliği artırılmıştır. Temelde 5 kademe sıkıştırma ve 5 kademe boyut artırma (görüntüyü orijinal boyutuna geri getirmek için) süreçlerinden oluşan Unet modelinin, katmanları arasındaki bağlantılar aşağıda gösterilen blok ile yenilenmiştir. Modellere genişlik kazandırmak için ilk olarak (3x3–5x5) ve (5x5–7x7) boyutlarındaki paralel filtrelerden oluşan blok tanımlanmıştır. Filtre sayıları sabit kalmak üzere, blok içerisindeki ilk ve ikinci katmandaki filtre sayıları değiştirilerek boyut düşürme ve yükseltme işlemleri gerçekleştirilmiştir. Ortaklama işlemi ise sadece görüntüyü sıkıştırma aşamasında uygulanmış ve art arda gelen 5 kademe sıkıştırma işlemi neticesinde boyutlar düşürülerek 7x7x1024 boyutunda özellik vektörleri elde edilmiştir.

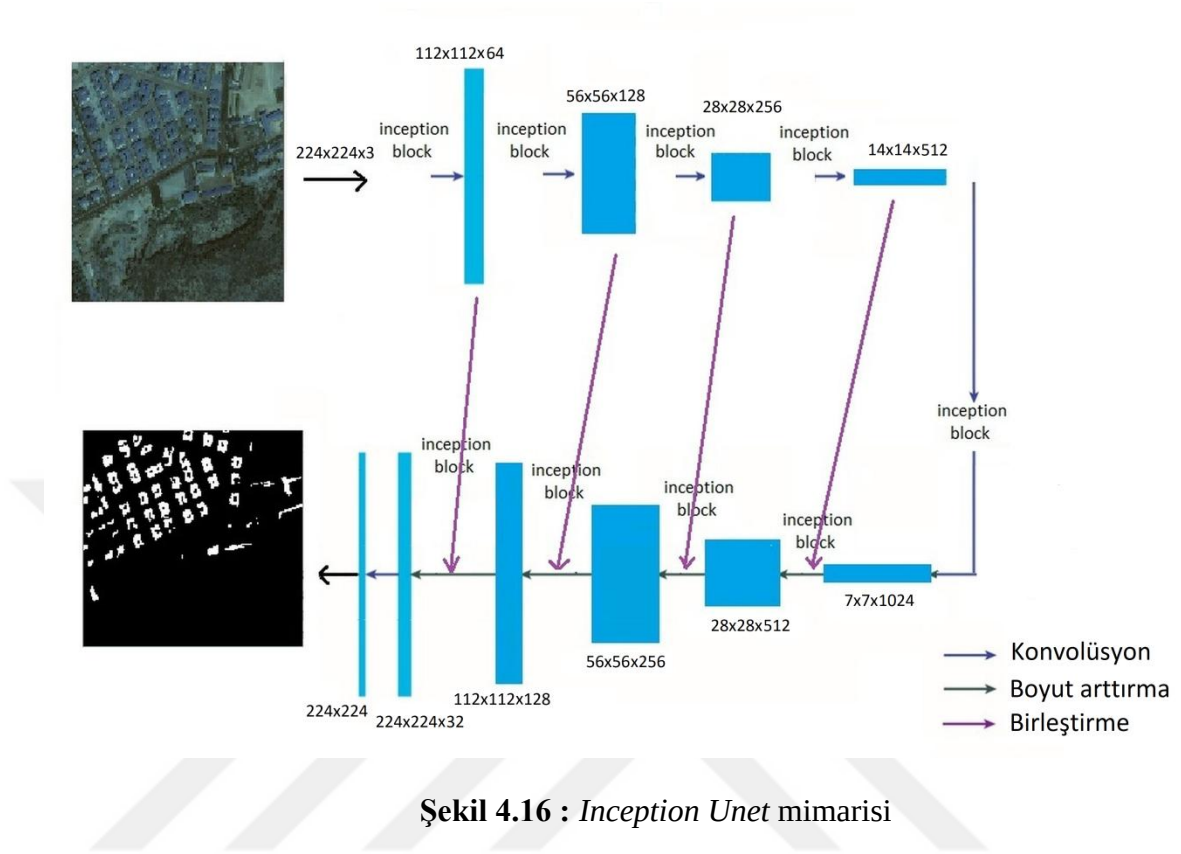


Şekil 4.14 : Unet modellerinde kullanılan bağlantı bloğu

Şekil 4.14’de grafiksel olarak gösterilen blok ile soldaki 224x224x24 boyutundaki görüntü sırasıyla 16 ve 32 düğüme sahip farklı katmanlardan geçirilerek 112x112x64 boyutunda çıkış görüntüsü elde edilmektedir. Çıkış matrisi iki farklı konvolüsyon sonucunun uç uca eklenmesi sonucu elde edilmektedir. Paralel filtreler içeren bu blok yapısı ile bina bölütlemesinde başarılı sonuçlar elde ettiğimiz Unet mimarisi güncellenmiş ve daha geniş yapıdaki yeni mimariler elde edilmiştir. Klasik Unet mimarisindeki bağlantıların yerine, yeni tanımlanan bağlantı bloğu yerleştirilerek elde edilen yeni mimarinin 2 katmanı arasındaki geçişler şekil 4.15’te gösterilmiştir. Şekil 4.15’te gösterilen bağlantı blokları kullanılarak elde edilen yeni mimari şekil 4.16’da gösterilmiştir. Tasarlanan mimarinin katmanlarının detayları ise Çizelge 4.2’de paylaşılmıştır.



Şekil 4.15 : Inception blok ile genişletilmiş mimari



Şekil 4.16 : Inception Unet mimarisi

Çizelge 4.2’de işlem tipi olarak belirtilen *blok*, önerilen *inception* bloğunu ifade etmektedir. Filtre sayısı olarak belirtilen sayılar (16, 32) ise *blok* içerisinde art arda gelen katmanlardaki filtre sayılarıdır. Çizelgede ilk sütunda katmanlara atanan numaralar gösterilmiştir. *Birleştirme* işlemi iki katman çıktısının birleştirilmesi anlamına gelmektedir. *Birleştirme* tipindeki işlemlerde parantez içerisinde kullanılan rakamlar, birleştirilen katman numaralarını ifade etmektedir. Dolayısıyla, *birleştirme*(6,5) işleminde 6 numaralı çıktı filtre sayısında belirtilen sayıda filtreden geçirilerek 5 numaralı katman çıktısı ile birleştirilir. Görüntünün boyutunun düşürüldüğü (2,3,4,5,6 numaralı) *blok* tipi katmanlarda ortaklama işlemi katman sonlarında uygulanmış, ardından gelen *blok* tipindeki katmanlarda ise ortaklama işlemi uygulanmamıştır.

Çizelge 4.2 : Inception Unet mimarisinin detayları

No	İşlem tipi	Filtre sayısı	Filtre boyutu	Çıktı boyutu
1	<i>konvolüsyon</i>	8	3x3	224 x 224 x 24
2	<i>Blok</i>	(16,32)	(3x3-5x5) - (5x5-7x7)	112 x 112 x 64
3	<i>Blok</i>	(32,64)	(3x3-5x5) - (5x5-7x7)	56 x 56 x 128
4	<i>Blok</i>	(64,128)	(3x3-5x5) - (5x5-7x7)	28 x 28 x 256
5	<i>Blok</i>	(128,256)	(3x3-5x5) - (5x5-7x7)	14 x 14 x 512
6	<i>Blok</i>	(256,512)	(3x3-5x5) - (5x5-7x7)	7 x 7 x 1024
7	<i>birleştirme (6,5)</i>	256	(2x2)	14 x 14 x 768
8	<i>Blok</i>	(512,128)	(3x3-5x5) - (5x5-7x7)	14 x 14 x 256
9	<i>birleştirme (8,4)</i>	256	(2x2)	28 x 28 x 512
10	<i>Blok</i>	(256,64)	(3x3-5x5) - (5x5-7x7)	28 x 28 x 128
11	<i>birleştirme</i>	128	(2x2)	56 x 56 x 256
12	<i>Blok</i>	(128,32)	(3x3-5x5) - (5x5-7x7)	56 x 56 x 64
13	<i>birleştirme</i>	64	(2x2)	112 x 112 x 128
14	<i>Blok</i>	(62,16)	(3x3-5x5) - (5x5-7x7)	112 x 112 x 32
15	<i>birleştirme</i>	16	(2x2)	224 x 224 x 40
16	<i>konvolüsyon</i>	32	(3x3)	224 x 224 x 32
17	<i>konvolüsyon</i>	1	(1x1)	224 x 224 x 1

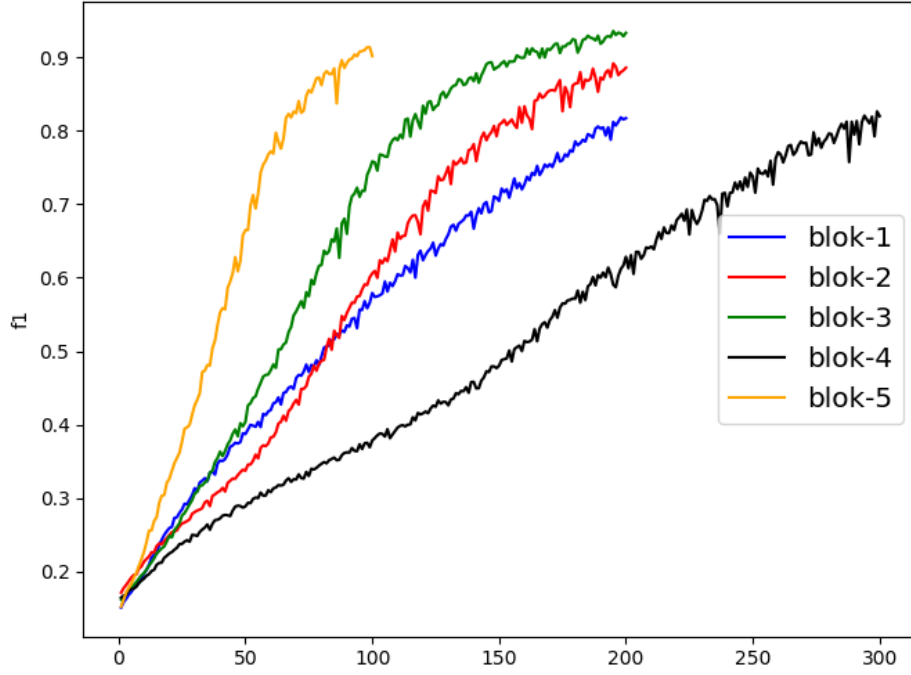
İşlem tipi olarak belirtilen *birleştirme* işleminin detayları ise şu şekildedir; Örneğin, çizelgede 9 numaralı *birleştirme* işleminde (*birleştirme (8,4)*) 8 nolu katman çıkışı 14x14x256 boyutlu matris ile 4 nolu katman çıkışı 28x28x256 boyutlu matrisler birleştirilmiştir. *Birleştirme* işleminde ilk olarak ilk girdiye 2x2 boyutundaki filtre 2x2 adım aralıkları ile devrik konvolüsyon işlemi uygulanır ve boyutu arttırılır, ardından ikinci girdi ile birleştirilerek yeni bir matris elde edilir. Yani 14x14x256 boyutlu matris önce 28x28 boyutuna çıkarılır ve 28x28x256 boyutlu matris ile birleştirilmesi sonucunda 28x28x512 boyutunda çıktı elde edilmiş olur. Böylece sıkıştırılmış görüntünün boyutu arttırılırken, birkaç katman gerideki aynı boyutlu çıktı sonuçları ile birleştirilerek geçmiş bilgiler sonraki katmanlara aktarılmış olur.

Farklı sayıda ve pencere boyutunda filtreler içeren çeşitli bloklar tanımlanarak farklı yapıda genişletilmiş mimariler oluşturulmuştur. Çizelge 4.3'te detayları belirtilen farklı filtreler içeren bloklar ile test edilen modellerin performans kıyaslamaları Çizelge 4.4'de gösterilmiştir. Unet modeli için 500 adım eğitim sonucunda elde edilen sonuçlar göz

önüne alınırken, Inception içeren mimarilerde ise eğitim işlemi sadece 200 adım çalıştırılmıştır. Sadece blok-4 için eğitim işlemi 300 adım çalıştırılmıştır. Bunun sebebi eğitim performansında tüm modeller için en azından yakın seviyelere ulaşılmasının sağlanmasıdır. Bu kapsamda, blok-4 ile oluşturulan modelin ancak 300 adım sonunda 0.80 civarı performans değerlerine ulaştığı görülmektedir. Önerilen bloklar ile elde edilen eğitim performans detayları şekil 4.17’de gösterilmiştir.

Çizelge 4.3 : Test edilen blok içerikleri

Blok Adı	Blok içeriği	Modelde eğitilecek parametre sayısı
blok -1	3x3 – 5x5 5x5 – 7x7	45.923.253
blok-2	1x1 3x3 5x5	24.200.831
blok -3	1x1 3x3 – 1x1 5x5 – 1x1	25.088.721
blok -4	3x3 – 3x3 5x5 – 5x5	35.469.057
blok-5	1x1 3x3 – 1x1 3x3 – 3x3 5x5 – 5x5	67.173.489

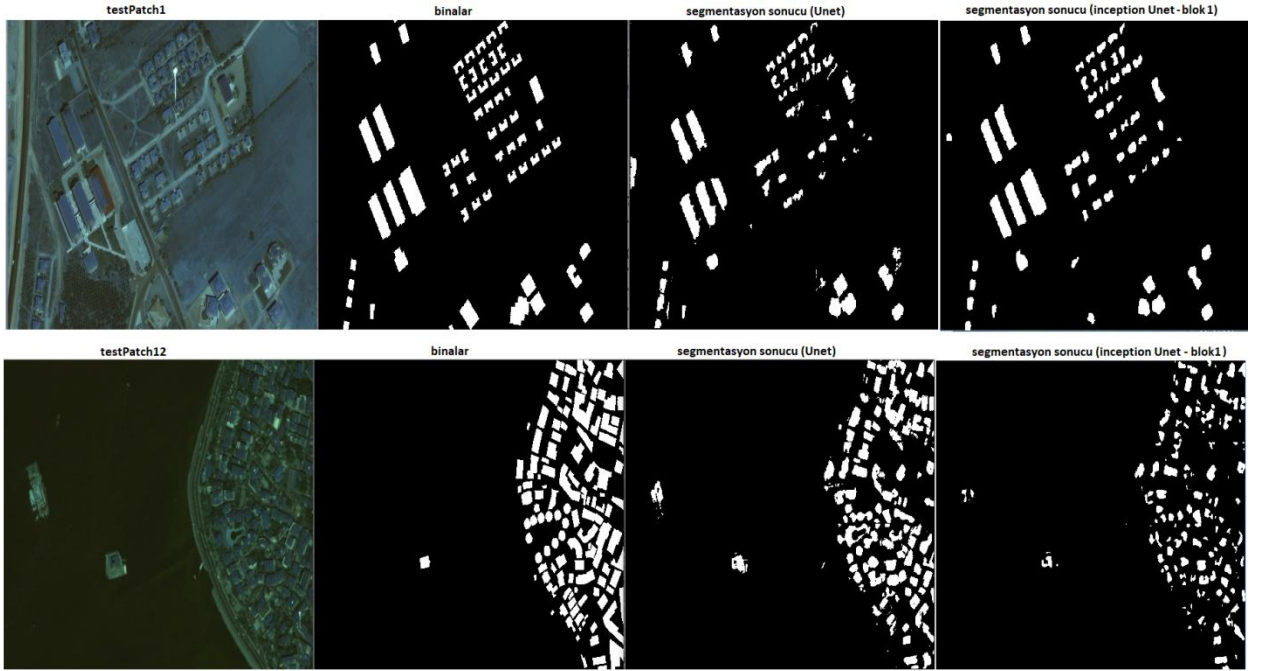


Şekil 4.17 : Farklı Inception blokları ile eğitim performansları

Çizelge 4.4 : Farklı bloklar ile oluşturulan mimarilerin performansları (F_1)

No	Unet (F_1)	Inception Unet (blok-1)	Inception Unet (blok-2)	Inception Unet (blok-3)	Inception Unet (blok-4)	Inception Unet (blok-5)
1	0.7326	0.8517	0.7797	0.8170	0.7632	0.8727
2	0.8464	0.8642	0.8197	0.8395	0.8237	0.8871
3	0.7454	0.7435	0.6949	0.7026	0.6545	0.8030
4	0.3562	0.4310	0.4371	0.4208	0.4184	0.4889
5	0.6131	0.7416	0.6683	0.6590	0.7352	0.7810
6	0.7818	0.8484	0.7870	0.7708	0.8616	0.8744
7	0.8337	0.8828	0.8365	0.8587	0.8707	0.8821
8	0.7635	0.8023	0.7583	0.7604	0.8036	0.7718
9	0.1666	0.4573	0.3974	0.4325	0.5450	0.7960
10	0.5447	0.7602	0.7390	0.7474	0.7820	0.7456
11	0.8350	0.8325	0.7427	0.7673	0.8456	0.7861
12	0.7480	0.5656	0.3934	0.4278	0.6079	0.4393
13	0.6258	0.3766	0.2283	0.2262	0.3982	0.2874
14	0.7342	0.6395	0.5142	0.5071	0.6744	0.4978
ort	0.7097	0.7420	0.6662	0.6979	0.6995	0.7377

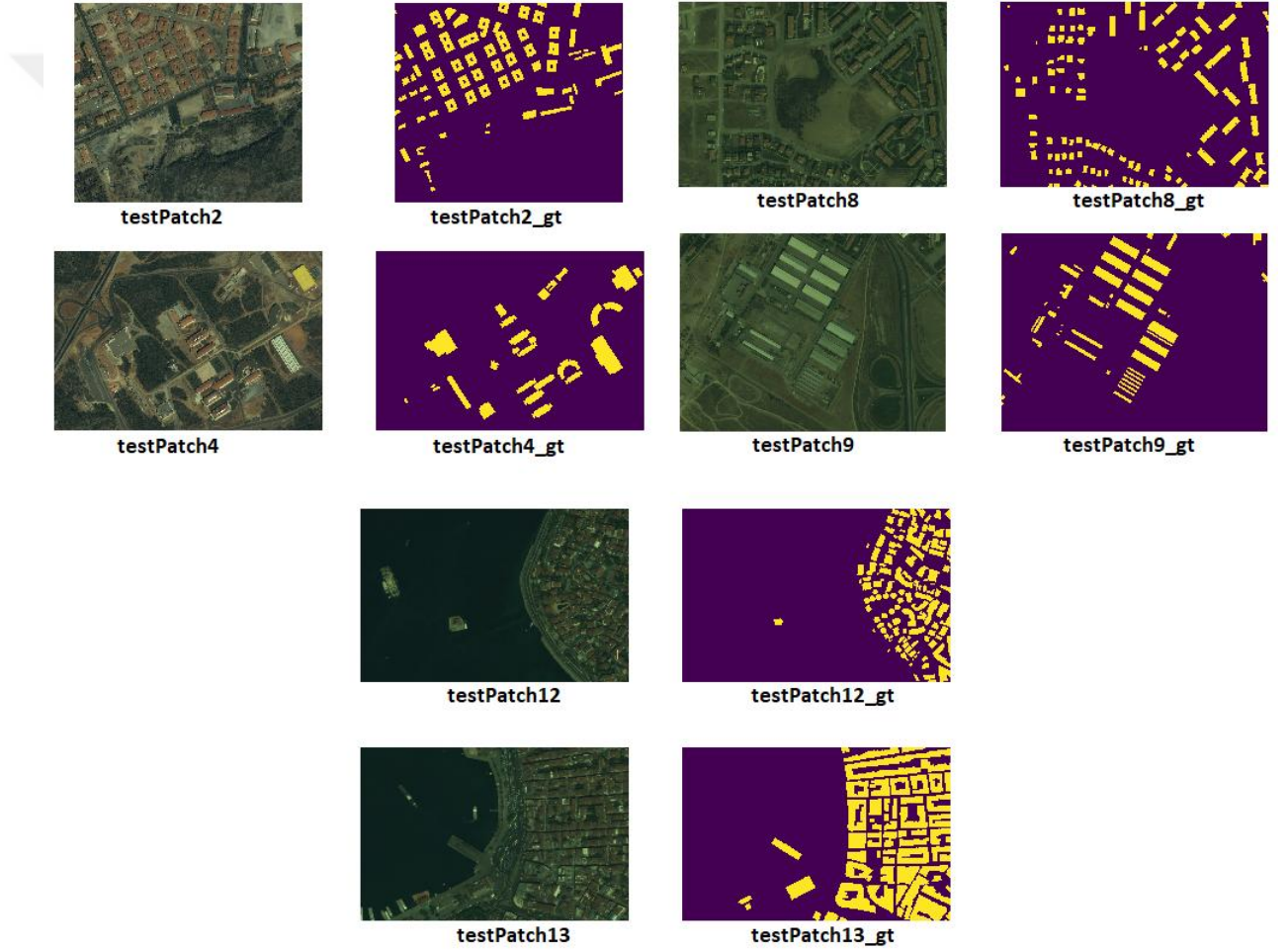
Sonuçlar hesaplanırken her bir görüntüden 224x224 alanlar seçildiğinden aslında toplamda 314 görüntü test amaçlı kullanılmaktadır. Çizelgede en alt satırda ortalama performans sonuçları paylaşılmıştır, aynı görüntü için en iyi performansı veren iki sonuç ise kalın rakamlarla gösterilmiştir. Sonuçlar incelendiğinde daha büyük boyutlu filtreler içeren blok-1 ve 3x3-5x5 ve 1x1 boyutlu farklı filtreler ile oluşturulan mimarilerin başarısı göze çarpmaktadır. Blok-5 ile hazırlanan mimarinin eğitim doğruluk oranının daha kısa sürede artmış olması da dikkat çekicidir. Blok-1 ise diğer mimarilere nazaran eğitim doğruluğu daha düşük kalmasına rağmen test görüntülerinde daha iyi sonuç vermiştir. Blok-3 ile düşük performans elde edildiğinden, Inception bloğundaki tüm katmanlarda 1x1 boyutunda filtre kullanmanın çıkarılan özellik sayısını düşürdüğü, dolayısıyla performansı olumsuz etkilediği söylenebilir. Aynı şekilde, blok-2 ile oluşturulan mimari de sadece tek bir katmandan oluştuğundan özellik çıkarımında başarısızdır. 1 ve 12 numaralı görüntüler için klasik unet modeli ile önerilen modellerin bölütleme sonuçları Şekil 4.18’de paylaşılmıştır.



Şekil 4.18 : 1 ve 12 numaralı görüntülerin bölütleme sonuçları

Şekil 4.18’deki bölütleme sonuçlarına göre, 1 numaralı görüntüde önerilen inception unet modeli iyi sonuç verirken, daha karmaşık yapıdaki 12 numaralı görüntüde unet modeli daha başarılıdır. Veri seti geneline bakıldığında ise unet modelinin sadece eğitim

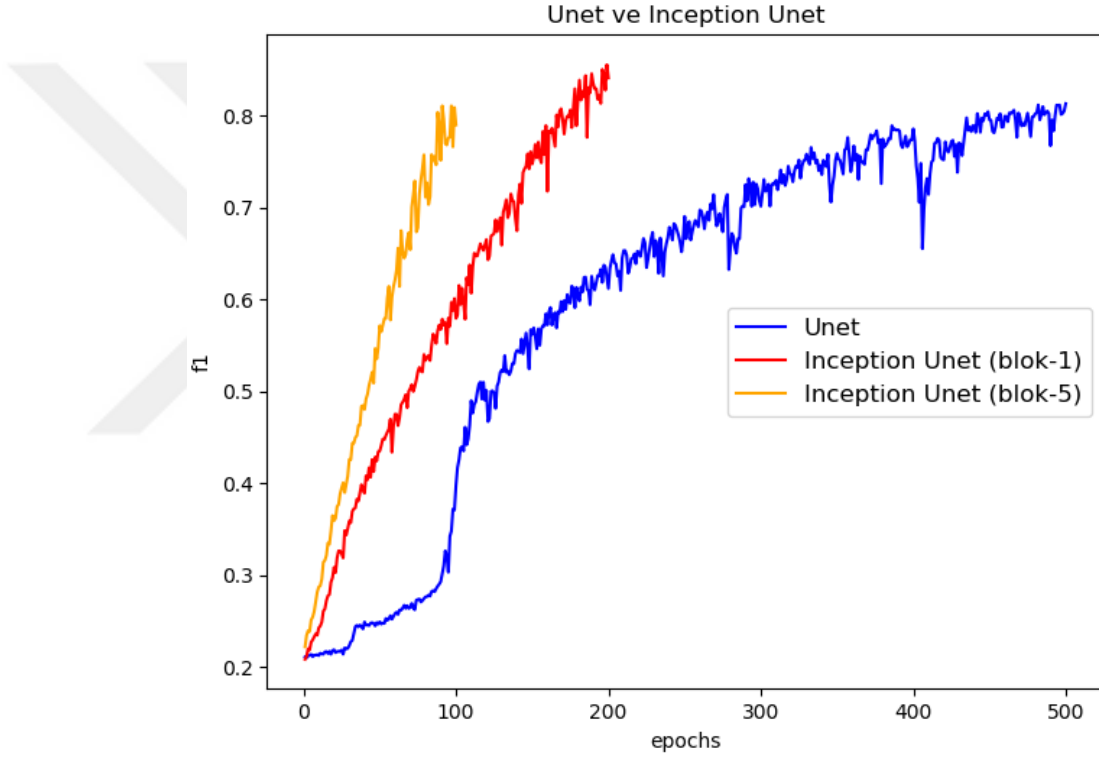
setinde bulunmayan son üç görüntüde önerilen modellere nazaran önemli derecede iyi performans sergilediği görülmektedir. Bu görüntüler birbirlerine son derece yakın binalar içeren karmaşık şehir merkezi görüntüleridir. Modeller arasında daha iyi bir kıyaslama yapılabilmesi için, farklı görüntülerden rastgele alanlar seçilerek yeni bir eğitim seti (eğitim seti-2) daha oluşturulmuştur. Oluşturulan eğitim seti-2 için 2,4,8,9,12 ve 13 numaralı görüntüler seçilmiştir (Şekil 4.19). Eğitim seti-2 için sadece blok-1 ve blok-5 kullanılarak tasarlanan Inception unet mimarileri eğitilmiştir.



Şekil 4.19 : Eğitim seti-2 oluşturulmasında kullanılan görüntüler

Modellerin başarısını test etmek için hazırlanan bu yeni veri setinde, yoğun binalar içeren görüntüler ve eğitim seti-1 ile eğitilen modellerde nispeten başarısız sonuçlar alınan görüntüler tercih edilmiştir. Unet ve *Inception* Unet modelleri bu 6 görüntüden rastgele 10 alan seçilerek elde edilmiş 224x224 boyutlu 60 görüntü ile eğitilmiştir.

Eğitim süresince elde edilen performans grafiği Şekil 4.20’de paylaşılmıştır. Unet modeli ile 500 adımlık eğitim sonucunda 0.813 doğruluk değerine ulaşılırken, *Inception* Unet (blok-1) mimarisi ile 200 adım sonunda 0.841 doğruluk değerine ulaşılmıştır. *Inception* Unet (blok-5) mimarisi ile eğitim doğruluk değeri ise sadece 100 adım neticesinde 0.798’dir. Grafikte Unet modelinde 400 adımdan sonra eğitim performansındaki artışın durduğu görülmektedir. Bu sebeple eğitim işlemi daha fazla devam ettirilmemiştir.



Şekil 4.20 : Unet ve *Inception* Unet eğitim performansları (eğitim seti-2)

Çizelge 4.5’te hazırlanan eğitim seti-2 ile eğitilmiş modellerin her bir görüntü için sergilediği bölütleme performansları kıyaslanmıştır. Buna göre Inception içeren mimariler ile çok daha iyi performans değerlerine ulaştığı görülmektedir. Her bir görüntüde önerilen mimarilerin Unet performansını geçtiği göze çarpmaktadır. Ortalamada ise Unet modeli ile 0.616 doğruluk değerine ulaşılırken, bu değer *Inception* Unet (blok-1) ile 0.766, *inception unet*(blok-2) ile 0.8080 olmaktadır. Eğitim seti-2 ile elde edilen sonuçlara bakılarak, blok-5’te 3x3, 5x5 ve 1x1 boyutlarındaki farklı filtreler

ile hazırlanan 4 paralel katman kullanılmasının özellik çıkarımını zenginleştirdiği söylenebilir.

Çizelge 4.5 : Eğitim seti-2 ile elde edilen performanslar (F_1)

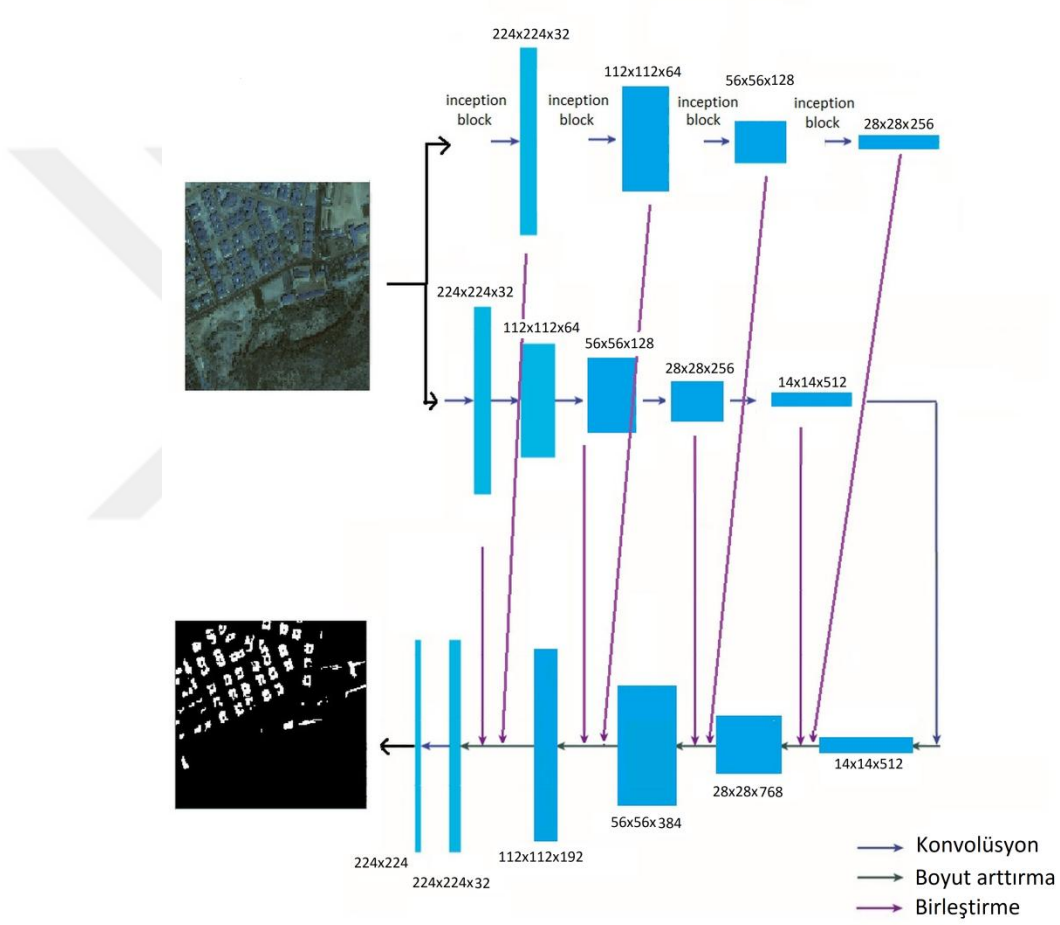
No	Unet (F_1)	Inception Unet (blok-1)	Inception Unet (blok-5)
1	0.3949	0.6361	0.7188
2	0.5995	0.7894	0.8489
3	0.4521	0.6071	0.7511
4	0.4868	0.7574	0.8310
5	0.5333	0.6154	0.7454
6	0.4179	0.6151	0.7654
7	0.5454	0.7764	0.8356
8	0.6186	0.7754	0.8419
9	0.6267	0.8460	0.8801
10	0.3450	0.7498	0.8027
11	0.6544	0.7955	0.8282
12	0.6957	0.7870	0.8191
13	0.7777	0.7808	0.8302
14	0.6043	0.7740	0.8022
ort	0.6167	0.7669	0.8080

4.4 Önerilen Inception Unet-V2 mimarisi

Inception blokları kullandığımız *Inception Unet* mimarisi ile klasik *Unet*'in performansında ciddi miktarda artışlar sağlandığını gözlemlenmektedir. Bunun sebebi gerek *Unet* mimarisinin genişletilmesi suretiyle katsayı miktarının artması, gerekse de katmanlar arasında farklı pencere boyutunda farklı değerlere sahip filtreler uygulanmasıdır. Böylece görüntüden özellik çıkarımı ve görüntünün orijinal boyutlarında yeniden oluşturulması aşamalarında farklı yapılarıdaki/boyutlardaki binaların daha iyi çıkarıldığı düşünülebilir.

Inception Unet mimarisinin eğitilecek parametre sayısı ve hafıza gereksinimini aşırı arttırmamasından dolayı yeni ve daha az sayıda parametre içeren bir mimari önerilmiştir. Önerilen mimaride *Inception* ve klasik *Unet* bağlantıları paralel katmanlarda bir arada kullanılması amaçlanmıştır. İlk olarak sadece özellik çıkarımı devamında gelen görüntünün orijinal boyutlarından elde edilmesi aşamasında paralel katmanlar

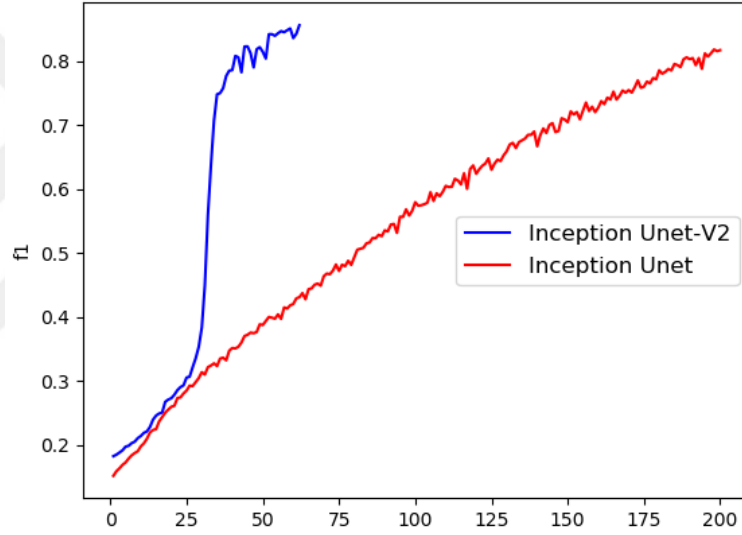
kullanılmış, fakat makul sonuçlar elde edilememiştir. Eğitim sürecinde doğruluk miktarının aşırı yavaş arttığı, dolayısıyla eğitim işleminin başarısız olduğu gözlemlenmiştir. Ardından sadece görüntüden özellik çıkarımı aşamasında *Inception* bloklarının klasik Unet katmanları ile paralel olarak kullanıldığı mimari önerilmiştir (Şekil 4.21).



Şekil 4.21 : Önerilen Inception Unet-V2 mimarisi

Önerilen bu mimarinin *Inception Unet* mimarisinden belirgin farklarından birisi sadece özellik çıkarımı aşamasında *Inception* blokların kullanılmasıdır. Ayrıca, *Inception* bloklarındaki her bir katmanda uygulanan filtre sayıları *Inception Unet*'e kıyasla yarıya indirilmiştir. Böylece eğitilecek parametre sayısının aşırı artmasının önüne geçilmiş ve klasik Unet'de kullanılan katmanlarla birlikte görüntüden özellik çıkarımı aşaması zenginleştirilmiştir. Ayrıca özellik çıkarımı katmanlarında elde edilen *Inception* ve

klasik Unet filtre sonuçlarının her ikisinin birden, görüntünün yeniden oluşturulduğu katmanlara aktarılması önemlidir. Böylece sıkıştırılan görüntüden orijinal boyutlardaki görüntü geri elde edilirken daha fazla özellik kullanılmıştır. Önerilen bu mimariye *Inception Unet-V2* ismi verilmiştir. *Inception Unet* ile elde edilen başarıdan esinlenerek önerilen bu mimaride, parametre sayısı önemli miktarda azalmasına rağmen daha iyi performans sonuçları elde edilmiştir. Aynı zamanda çok daha kısa süreli eğitim işlemine rağmen daha iyi performans değerlerine ulaşılması dikkat çekicidir. Blok-1 kullanılarak tasarlanan *Inception Unet* ve *Inception Unet-V2* mimarilerinin eğitim seti-1 için performans grafikleri Şekil 4.22’de paylaşılmıştır.

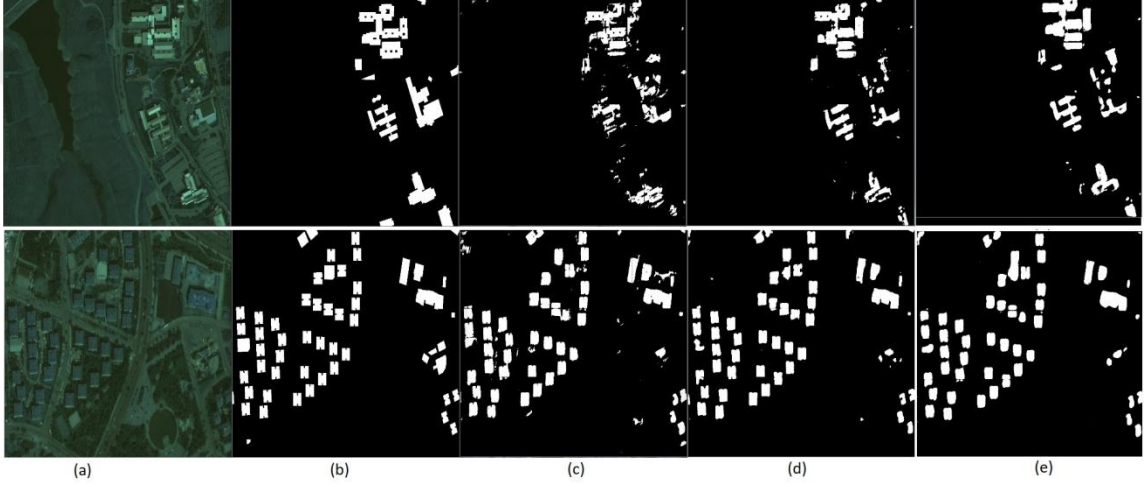


Şekil 4.22 : Inception Unet ve Inception Unet-V2 eğitim performansları

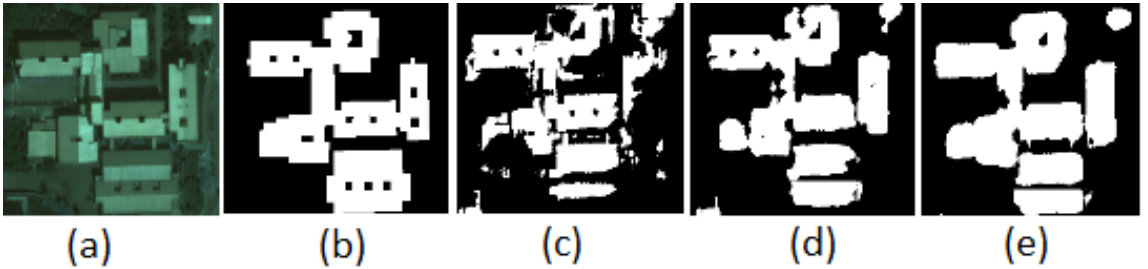
Inception-UnetV2 ile sadece 62 adım eğitim neticesinde 0.8558 performans değerine ulaşılmıştır. Önerilen mimarilerin performans kıyaslamaları nitel ve nicel değerlendirme olmak üzere iki aşamada gerçekleştirilmiştir. 5 ve 7 numaralı görüntüler için elde edilen bina bölütleme sonuçları Şekil 4.23’de paylaşılmıştır. 5 numaralı görüntü için küçük binaların, Unet modelini olumsuz yönde etkilediği görülmektedir. Unet bir dizi yanlış sınıflandırılmış piksele sahipken, önerilen yöntemlerin Inception blokları ile oldukça iyi sonuçlar verdiği görülmektedir. Sonuçlar incelendiğinde klasik Unet ile elde edilen sonuç görüntülerinde, gürültü diye adlandırabileceğimiz küçük noktaların olduğu görülmektedir. Inception-Unet ve Inception Unet-V2 sonuç görüntülerinde ise çok daha az gürültü bulunduğu görülmektedir. Tespit edilen binaların daha detaylı karşılaştırması

ise Şekil 4.24'de paylaşılmıştır. Buna göre Unet ile binaların ortalarındaki boşlukların çıkarılabildiği görülmekle birlikte, binaların bütününe bakıldığında ise önerilen mimari ile elde edilen sonuçların çok daha iyi olduğu anlaşılmaktadır.

Çizelge 4.6 ve Çizelge 4.7'de sırasıyla eğitim seti-1 ve eğitim seti-2 ile elde edilen performans sonuçları paylaşılmıştır. Inception içeren her iki mimaride de en iyi sonuçların alındığı blok-1 ve blok-5 Inception blokları kullanılmıştır.



Şekil 4.23 : testpatch5 ve testpatch7 görüntüleri için elde edilen sonuçlar (a) RGB girdi görüntüsü (b) etiketlenmiş binalar (c) Unet (d) Inception Unet (e) Inception Unet-V2



Şekil 4.24 : testpatch7 görüntüsünden kırpılan bir alan; (a) RGB görüntü (b) etiketlenmiş binalar (c) Unet (d) Inception Unet (e) Inception Unet-V2

Çizelge 4.6 : Eğitim seti-1 model performansları (F₁)

No	Unet	Inception Unet (blok-1)	Inception Unet-V2 (blok-1)	Inception Unet (blok-5)	Inception Unet-V2 (blok-5)
1	0.7326	0.8517	0.7760	0.8727	0.8042
2	0.8464	0.8642	0.8529	0.8871	0.8581
3	0.7454	0.7435	0.7435	0.8030	0.8062
4	0.3562	0.4310	0.5032	0.4889	0.4980
5	0.6131	0.7416	0.7779	0.7810	0.7614
6	0.7818	0.8484	0.8930	0.8744	0.8709
7	0.8337	0.8828	0.8822	0.8821	0.8824
8	0.7635	0.8023	0.8025	0.7718	0.8142
9	0.1666	0.4573	0.8189	0.7960	0.7740
10	0.5447	0.7602	0.7880	0.7456	0.8003
11	0.8350	0.8325	0.8332	0.7861	0.8335
12	0.7480	0.5656	0.6271	0.4393	0.6041
13	0.6258	0.3766	0.4980	0.2874	0.5327
14	0.7342	0.6395	0.6751	0.4978	0.6403
Ort	0.7097	0.7420	0.7655	0.7377	0.7736

Çizelge 4.7 : Eğitim seti-2 model performansları (F₁)

No	Unet	Inception Unet (blok-1)	Inception Unet-V2 (blok-1)	Inception Unet (blok-5)	Inception Unet-V2 (blok-5)
1	0.3949	0.6361	0.5922	0.7188	0.6692
2	0.5995	0.7894	0.7651	0.8489	0.8237
3	0.4521	0.6071	0.7019	0.7511	0.7567
4	0.4868	0.7574	0.7849	0.8310	0.8179
5	0.5333	0.6154	0.7215	0.7454	0.7436
6	0.4179	0.6151	0.7783	0.7654	0.6232
7	0.5454	0.7764	0.8150	0.8356	0.8095
8	0.6186	0.7754	0.7878	0.8419	0.7732
9	0.6267	0.8460	0.8273	0.8801	0.8587
10	0.3450	0.7498	0.6781	0.8027	0.7951
11	0.6544	0.7955	0.8294	0.8282	0.8377
12	0.6957	0.7870	0.8087	0.8191	0.8275
13	0.7777	0.7808	0.8165	0.8302	0.8135
14	0.6043	0.7740	0.7990	0.8022	0.7964
Ort	0.6167	0.7669	0.7870	0.8080	0.7967

Performans sonuçları incelendiğinde, Inception Unet-V2 mimarisi ile eğitilen parametre sayısı 67 milyondan 12 milyon seviyelerine indirilirken, genel anlamda performansın arttığı görülmektedir. Bunun sebebi olarak, mimariyi önerirken ortaya attığımız sadece görüntüden özellik çıkarımında inception blokları uygulamanın daha etkin olacağı fikrinin fayda sağlamış olduğunu söyleyebiliriz. Ortalama değerlere, kesinlik ve hassasiyet sonuçlarına bakıldığında ise elde edilen sonuçlar Çizelge 4.7'de paylaşılmıştır. Kesinlik gerçek binaların ne kadarının bulunduğunu ifade eden metriktir. Kesinlik değerinde *inception* içeren her iki mimari ile de iyi sonuçlar alındığı görülmektedir. Örneğin; Inception Unet (blok-1) ile 0.8970 kesinlik değeri elde edilirken, Inception Unet-V2 ile 0.8811 kesinlik değerine ulaşılmıştır. Hassasiyet sonuçlarının ise nispeten daha kötü olduğu görülmüştür. Ortalama kesinlik, hassasiyet ve F skor değerleri Çizelge 4.8'de paylaşılmıştır.

Çizelge 4.8 : RGB görüntülerin performans kıyaslaması (P-R-F₁)

Mimariler	Eğitim seti-1			Eğitim seti-2		
	Kesinlik(P)	Hassasiyet(R)	F ₁	Kesinlik(P)	Hassasiyet(R)	F ₁
Unet	0.7915	0.6951	0.7097	0.6141	0.7025	0.6167
InceptionUnet(blok-1)	0.8970	0.6851	0.7420	0.8552	0.7299	0.7669
InceptionUnet-v2(blok-1)	0.8493	0.7374	0.7655	0.8407	0.7736	0.7870
InceptionUnet(blok-5)	0.8887	0.6834	0.7377	0.8493	0.8068	0.8080
InceptionUnet-v2(blok-5)	0.8811	0.7294	0.7736	0.8987	0.7507	0.7967

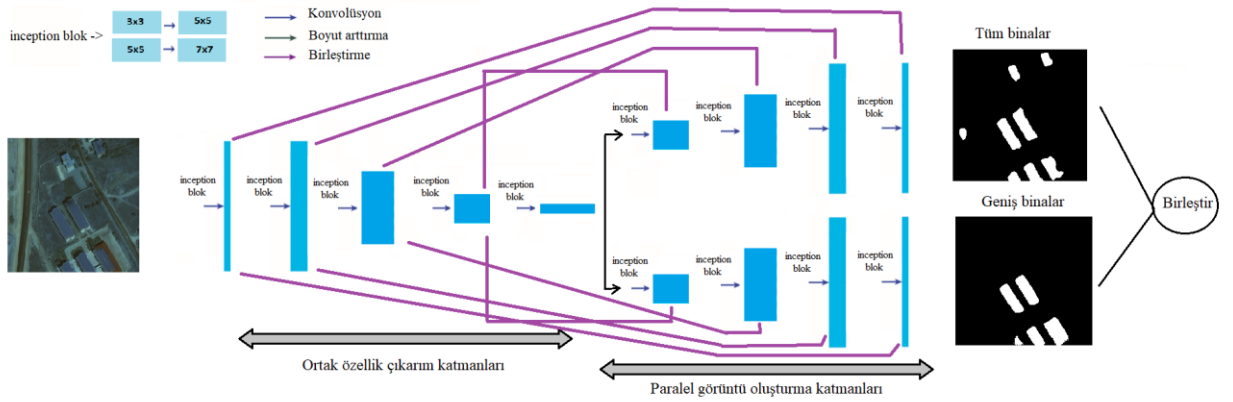
Son olarak RGB-NIR olmak üzere 4 bantlı görüntüler ile elde edilen sonuçların performans kıyaslaması çizelge 4.9'da paylaşılmıştır. Buna göre blok-5 ile oluşturulan mimarilerin daha başarılı olduğu gözlemlenmektedir. Eğitim seti-2 için en iyi performans değerleri birbirine yakın olmakla birlikte, genele bakıldığında Inception Unet-V2 mimarisinin diğer mimarilere nazaran daha iyi test sonuçlarına ulaştığı da görülmektedir. Daha iyi bir kıyaslama için, daha çok sayıda görüntü içeren bir başka veri seti ile elde edilen sonuçlar bölüm sonunda paylaşılmıştır.

Çizelge 4.9 : RGB-NIR görüntülerin performans kıyaslaması

Mimariler	Eğitim seti-1			Eğitim seti-2		
	Kesinlik(P)	Hassasiyet(R)	F ₁	Kesinlik(P)	Hassasiyet(R)	F ₁
Unet	0.7877	0.7365	0.7376	0.7241	0.7590	0.7137
Inception Unet (blok-1)	0.8413	0.7698	0.7687	0.8671	0.7502	0.7845
Inception Unet-v2 (blok-1)	0.8714	0.7661	0.7963	0.8489	0.8195	0.8183
Inception Unet (blok-5)	0.9177	0.7225	0.7724	0.8941	0.8034	0.831
Inception Unet-v2 (blok-5)	0.8920	0.7577	0.7959	0.8842	0.8114	0.8304

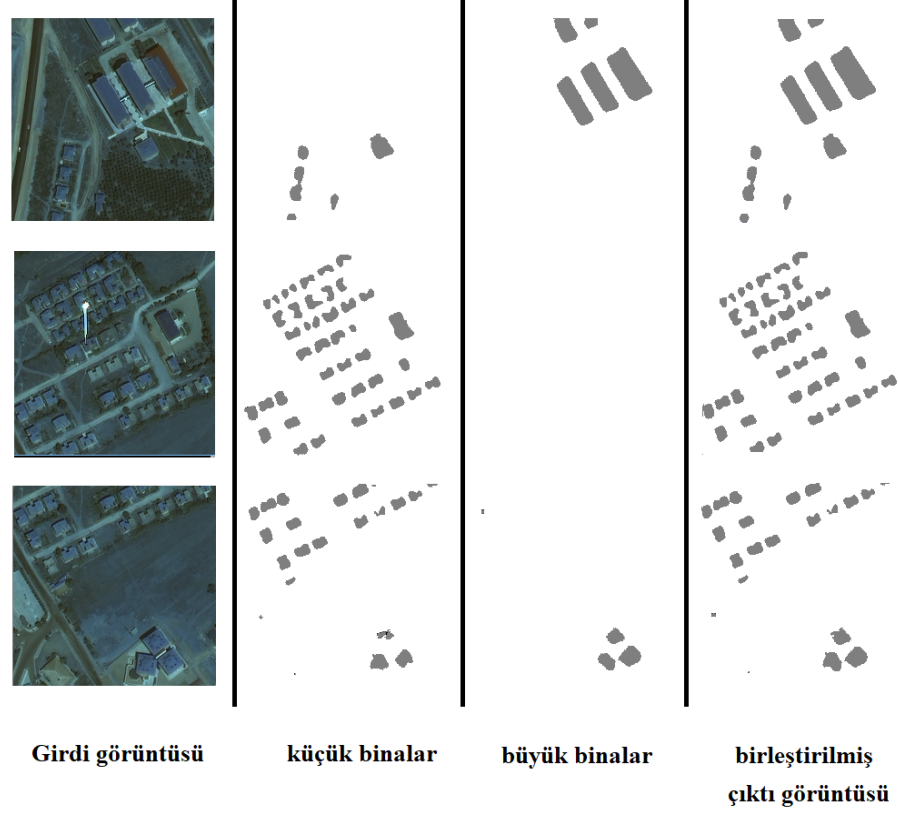
4.5 Çok çıkışlı mimariler

Unet mimarisinin revize edildiği bir çalışmada çok çıkışlı bir mimari önerilerek, küçük orta ve büyük boyutlu binalar için ayrı katmanlar önerilmiştir [74]. Literatürdeki bu çalışmadan esinlenilerek, önerdiğimiz mimarilerde bu yaklaşımın nasıl sonuç vereceği test edilmiştir. Bu kapsamda görüntüden özellik çıkarımı katmanları ortak kalmak suretiyle, görüntünün yeniden oluşturulma aşamasında paralel çoklu çıkış katmanları önerdiğimiz mimarilere eklenmiştir. Böylece hem eğitilecek parametre sayısının aşırı artmasının önüne geçilmiş, hem de bina maskelerinin oluşturulması aşamalarında farklı boyutlardaki binalar için farklı katmanların kullanılmasının avantajlarından faydalanılmak istenmiştir. Bu amaçla küçük binalar-büyük binalar ve tüm binalar-büyük binalar için önerdiğimiz çok çıkışlı mimariler test edilmiştir. *Inception Unet* mimarisinin çok çıkışlı versiyonunun genel yapısı Şekil 4.25’de görülmektedir.



Şekil 4.25 : Inception Unet çok çıkışlı mimari

Önerilen mimari ile küçük binalar ve büyük binalar için eğitim işlemi gerçekleştirildiğinde, küçük binalardaki başarımların tek çıkışlı versiyona nazaran düşük kaldığı görülmüştür. Tüm binalar ve büyük binalar kullanılarak eğitim işlemi gerçekleştirildiğinde ise, büyük binalar için elde edilen doğruluk değerinin daha yüksek seviyelere çıktığı gözlemlenmiştir. Bu sebeple, önerilen çoklu mimariler tüm binalar ve büyük binalar için çalıştırılmıştır. Şekil 4.26’da küçük ve büyük binalar kullanılarak elde edilen örnek sonuçlar paylaşılmıştır.



Şekil 4.26 : *Inception Unet* çok çıkışlı mimari ile elde edilmiş küçük ve büyük bina örnek sonuç görüntüleri

Çizelge 4.10 ve 4.11'de ise model performanslarının her bir görüntü özelinde ve ortalama olarak kıyaslamaları yapılmıştır. Sonuçlara bakıldığında çok çıkışlı mimari ile çok az miktarda performans artışının sağlandığı gözlemlenmiştir. *Inception Unet* mimarisi için ortalama f_1 skor değeri 0.7420'den 0.7500 seviyelerine ulaşmıştır. Çok daha az parametre içeren *Inception Unet-V2* mimarisinin çok çıkışlı versiyonunda ise ortalama değerin daha az değiştiği görülmektedir. Sonuçların geneli incelendiğinde, parametre sayısı daha az olan *Inception Unet-V2* 'nin diğer mimarilere nazaran daha performanslı olduğu açıktır. *Inception Unet-V2* 'nin tek çıkışlı versiyonu bile *Inception Unet* çok çıkışlı versiyonundan daha başarılıdır.

Çizelge 4.10 : Eğitim seti-1 çok çıkışlı model performansları (F_1)

No	Unet	Inception Unet	Inception Unet multi-task	Inception Unet-V2	Inception Unet-V2 multi-task
1	0.7326	0.8517	0.8507	0.7760	0.7758
2	0.8464	0.8642	0.8649	0.8529	0.8510
3	0.7454	0.7435	0.7440	0.7435	0.7435
4	0.3562	0.4310	0.4426	0.5032	0.5049
5	0.6131	0.7416	0.7654	0.7779	0.7833
6	0.7818	0.8484	0.8770	0.8930	0.8922
7	0.8337	0.8828	0.8869	0.8822	0.8821
8	0.7635	0.8023	0.8056	0.8025	0.8018
9	0.1666	0.4573	0.5442	0.8189	0.8124
10	0.5447	0.7602	0.7853	0.7880	0.7792
11	0.8350	0.8325	0.8475	0.8332	0.8450
12	0.7480	0.5656	0.5752	0.6271	0.6316
13	0.6258	0.3766	0.3828	0.4980	0.5065
14	0.7342	0.6395	0.6430	0.6751	0.6746
ort	0.7097	0.7420	0.7508	0.7655	0.7639

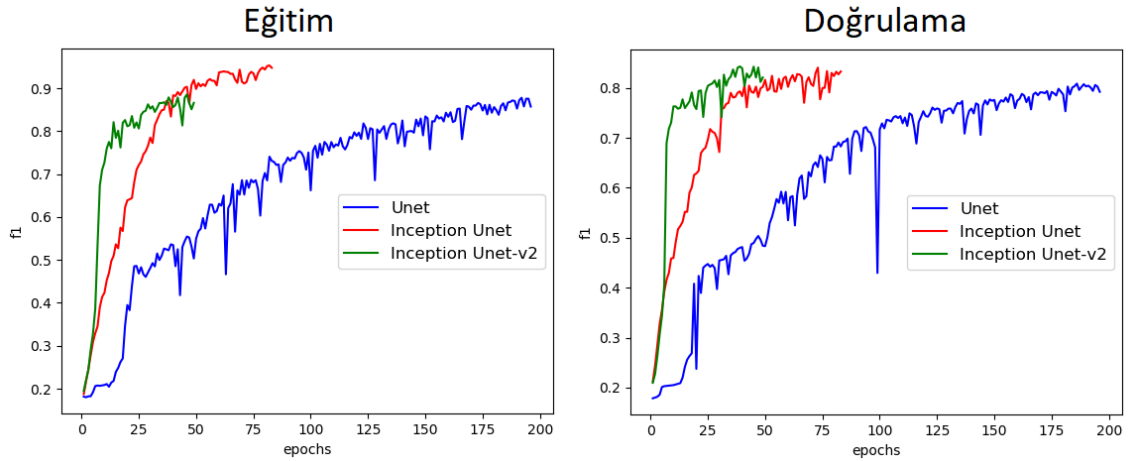
Çizelge 4.11 : Eğitim seti-2 çok çıkışlı model performansları (F_1)

No	Unet	Inception Unet	Inception Unet multi-task	Inception Unet-V2	Inception Unet-V2 multi-task
1	0.3949	0.6361	0.6434	0.5922	0.5618
2	0.5995	0.7894	0.7122	0.7651	0.7027
3	0.4521	0.6071	0.6192	0.7019	0.7018
4	0.4868	0.7574	0.7799	0.7849	0.7911
5	0.5333	0.6154	0.6883	0.7215	0.7549
6	0.4179	0.6151	0.6437	0.7783	0.7454
7	0.5454	0.7764	0.7693	0.8150	0.8266
8	0.6186	0.7754	0.7874	0.7878	0.7838
9	0.6267	0.8460	0.8544	0.8273	0.8352
10	0.3450	0.7498	0.7226	0.6781	0.7458
11	0.6544	0.7955	0.7758	0.8294	0.8340
12	0.6957	0.7870	0.7950	0.8087	0.8078
13	0.7777	0.7808	0.8240	0.8165	0.8207
14	0.6043	0.7740	0.7857	0.7990	0.7935
ort	0.6167	0.7669	0.7700	0.7870	0.7924

InceptionUnet-V2 ile sadece özellik çıkarımında paralel Inception bloklarının kullanıldığı düşünüldüğünde, çok çıkışlı (multi-task) mimari ile küçük ve büyük binaların ayrı ayrı işlenmesinin pek fazla performans artışı sağlamadığı söylenebilir.

4.6 Çapraz doğrulama ile performans testi

Derin öğrenme mimarileri önerilmesi aşamasında iki farklı eğitim kümesinden elde edilen sonuçlara göre analizler yapılmış ve performans kıyaslamaları gerçekleştirilmiştir. Mimari geliştirilmesi aşamasında 60 görüntüden oluşturulan eğitim kümeleri üzerinden yorum yapılmıştır. Bu başlıkta ise eğitimde kullanılan görüntü sayısı arttırılarak toplam veri setinin %60'ı eğitim %8'i de doğrulama için kullanılmıştır ve çapraz doğrulama yöntemi (n=3 için) ile önerilen mimarinin (blok-5 kullanılarak) Unet ve Inception Unet ile kıyaslaması daha kapsamlı olarak ele alınmıştır. Çapraz doğrulamada veriler rastgele parçalara ayrılarak değerlendirilmekte, böylece performans kıyaslamasında tutarlılık sağlanmaktadır. Ayrıca ağ eğitiminde doğrulama görüntüleri kullanılarak eğitim işleminin otomatik olarak sonlandırılması sağlanmıştır. Toplam 314 görüntüden 188'i eğitim, 26'sı doğrulama ve 100'ü test kümesi olarak kullanılmıştır. Bu şekilde rastgele 3 farklı grup oluşturularak çapraz doğrulama işlemi gerçekleştirilmiştir. Eğitim sürecinde elde edilen eğitim ve doğrulama kümesi F_1 grafiğine ait bir örnek şekil 4.27'de paylaşılmıştır.



Şekil 4.27 : Veri seti-1 için eğitim ve doğrulama grafikleri

Şekil 4.27 incelendiğinde, Inception blokları içeren her iki mimari ile de Unet'e nazaran daha erken eğitim adımlarında daha fazla doğrulama ve eğitim doğruluğu değerlerine ulaşıldığı görülmektedir. Unet mimarisi ile yaklaşık 200 adım süren eğitim süreci, diğer mimarilerde 60 ve 80 civarındadır. Rastgele seçilen üç farklı veri grubu ile elde edilen performans değerlerinin detayları ve ortalamaları Çizelge 4.12'de paylaşılmıştır.

Çizelge 4.12 : Çapraz doğrulama ile performans kıyaslaması

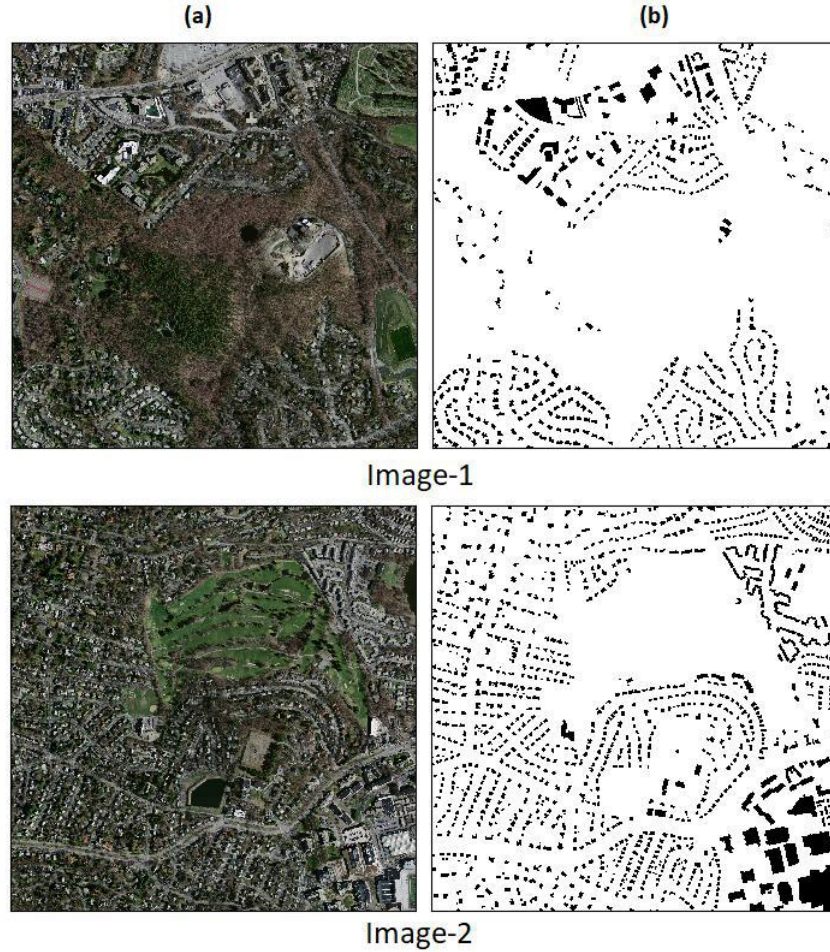
Mimariler		Unet	Inception Unet	Inception Unet-v2
Veri kümesi-1	Kesinlik (P)	0.7472	0.8624	0.8834
	Hassasiyet (R)	0.7507	0.7704	0.7932
	F ₁	0.7402	0.8014	0.8233
Veri kümesi -2	Kesinlik (P)	0.8180	0.9261	0.9108
	Hassasiyet (R)	0.8054	0.8020	0.8333
	F ₁	0.8040	0.8459	0.8530
Veri kümesi -3	Kesinlik (P)	0.7314	0.8930	0.8749
	Hassasiyet (R)	0.7287	0.8116	0.8344
	F ₁	0.7144	0.8323	0.8372
Ortalama	Kesinlik (P)	0.7655	0.8938	0.8897
	Hassasiyet (R)	0.7616	0.7947	0.8203
	F ₁	0.7529	0.8265	0.8378

Kesinlik değerleri incelendiğinde Inception Unet ile çok az bir miktar da olsa öne çıkmaktadır (Inception Unet: 0.8938, Inception Unet-V2: 0.8897). Hassasiyet ve F₁ skor değerlerinde ise Inception Unet-V2 mimarisi 0.8203 ve 0.8378 ile en iyi değerlere sahiptir. Çapraz doğrulama neticesinde elde edilen sonuçlara göre, önerilen mimariler ile kesinlik değeri Unet'e kıyasla yaklaşık %13, hassasiyet %6, F₁ ise %8 civarında artmıştır. Başlık 4.7'de çok sayıda görüntü içeren farklı bir veri seti üzerinden de performans kıyaslamaları gerçekleştirilmiştir.

4.7 Massachusetts veri kümesi ile test sonuçları

Bu aşamaya kadar test edilen mimariler nispeten kısıtlı bir veri seti üzerinde denenmiş/test edilmiş ve elde edilen sonuçlara göre yorumlar yapılmıştır. Kullanılan veri setinden 2 farklı eğitim kümesi oluşturularak ve az sayıda eğitim verisi kullanılarak elde edilen sonuçlar kıyaslanmıştır. Önerilen mimariler ile elde edilen sonuçların

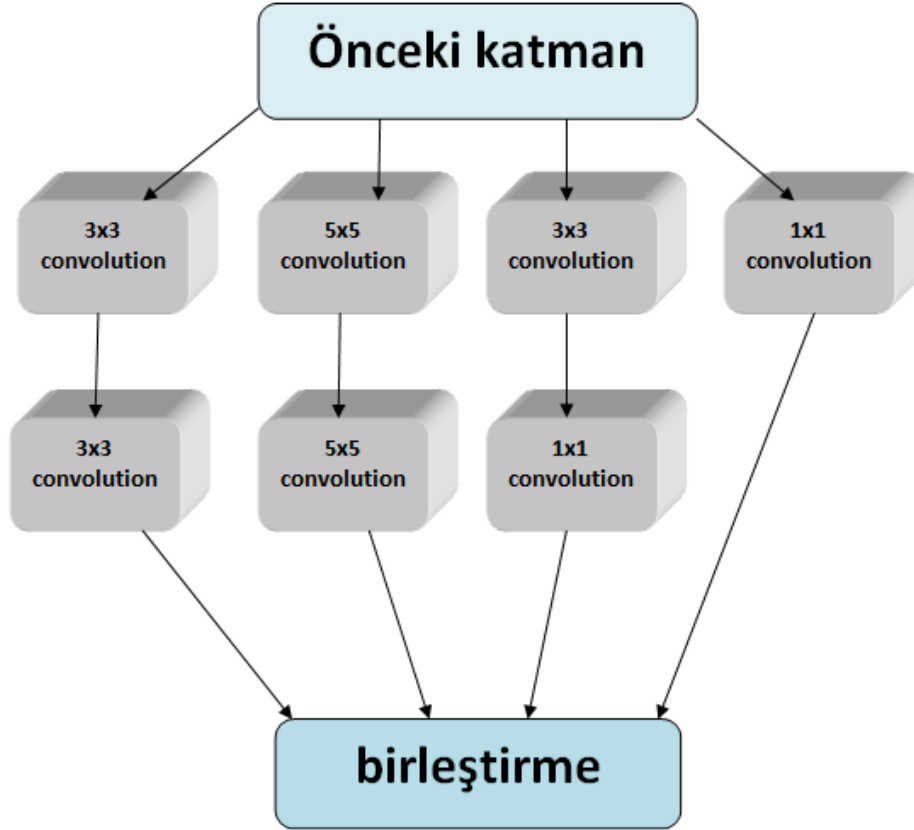
doğruluğunu daha iyi test edebilmek için, çok sayıda görüntü içeren bir başka veri seti daha kullanılmıştır. Bu kapsamda bir doktora tezi kapsamında yayımlanan ve açık olarak erişilebilen Massachusetts bina bölütleme veri kümesi kullanılmıştır [75]. Veri setinde 137 eğitim görüntüsü, 4 doğrulama görüntüsü ve 10 adet test görüntüsü bulunmaktadır. Görüntüler Boston şehrine ait şehir içi ve kırsal alana ait görüntülerdir. Veri kümesi 1 metre uzamsal çözünürlüğe sahip 1500x1500 boyutunda görüntülerden oluşmaktadır. Veri setindeki görüntüler için binaların etiketlenmiş olduğu gerçek sonuçlar da mevcuttur. Fakat eğitim setindeki 27 görüntüde görüntünün yarısına kadar varan eksik bölgeler mevcuttur. Bu sebeple eksik kısımlar içeren 24 görüntü değerlendirme dışında bırakılmıştır. Veri setine ait örnek görüntüler şekil 4.28’de paylaşılmıştır. Veri setine ait 2 örnek görüntünün RGB görünümünü (a) sütunlarında, her bir görüntüdeki binaların etiketlendiği görüntüleri ise (b) sütunlarında görülebilir.



Şekil 4.28 : *Massachusetts* veri kümesinden örnek görüntüler

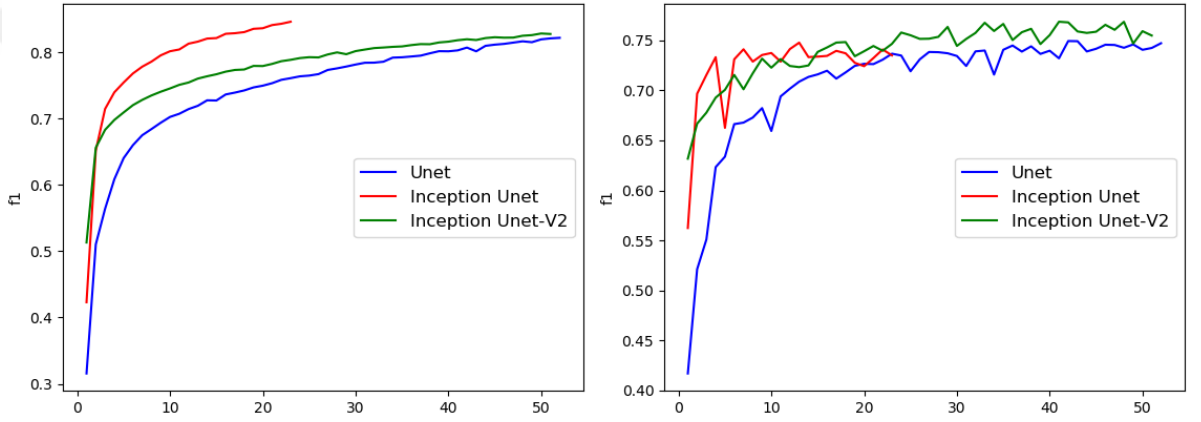
Veri setindeki toplam 124 görüntüden 224x224 boyutunda alt görüntüler oluşturulduğunda, toplamda 6076 görüntü elde edilmektedir. Massachusetts veri kümesinden 2 farklı eğitim seti oluşturularak önerilen mimarilerin performans kıyaslaması gerçekleştirilmiştir. İlk olarak veri kümesinde paylaşıldığı haliyle, 110 adet 1500x1500 boyutundaki görüntü eğitim seti olarak kullanılmış, 4 görüntü doğrulama ve 10 görüntü ise test için kullanılmıştır. Bunun yanında, 2. eğitim setinde ise, 110 eğitim görüntüsü sayısı 75'e düşürülmüş ve test sayısının artırılması amaçlanmıştır. Böylece her iki eğitim kümesinde sırasıyla, yaklaşık toplam veri kümesinin %88 ve %60'ı eğitim aşamasında kullanılmıştır. 4 adet görüntü ise doğrulama aşamasında kullanılmış, eğitimin ne zaman biteceğine doğrulama sonuçlarına bakılarak karar verilmiştir.

Massachusetts veri kümesinde yapılan testlerde blok-5 ile hazırlanan mimariler kullanılmıştır. Önerilen mimarilerde kullanılan blok-5 Inception yapısının detayı şekil 4.29'da paylaşılmıştır.

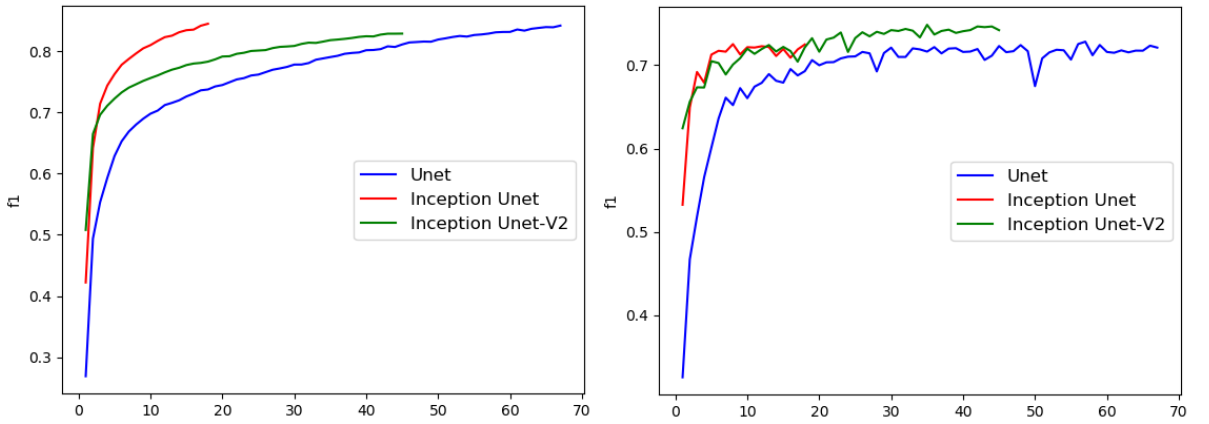


Şekil 4.29 : Kullanılan Inception bağlantı bloğu

Şekil 4.29’da detayları paylaşılan Inception blok ile tasarlanan mimarilere ait eğitim ve doğrulama doğruluk grafikleri Şekil 4.30 ve 4.31’de paylaşılmıştır. Ağ eğitiminde kullanılan doğrulama kümesi için 10 eğitim adımında iyileşme olmadığı durumda eğitim işlemi otomatik olarak durdurulmuştur. Doğrulama kümesi ile elde edilen değerlere bakıldığında 3 mimarinin de birbirine çok yakın değerlere ulaştıkları görülmektedir. Eğitim seti-1 için Unet eğitim doğruluğu (F_1) 0.8216 iken doğrulama kümesindeki doğruluk 0.7472’dir. Inception Unet için bu değerler sırasıyla 0.8458 ve 0.7351, Inception Unet-V2 için 0.8274 ve 0.7549’dur.



Şekil 4.30 : Massachusetts eğitim seti-I eğitim ve doğrulama grafikleri



Şekil 4.31 : Massachusetts eğitim seti-II eğitim ve doğrulama grafikleri

Eğitim seti-1 için Unet eğitim doğruluğu (F_1) 0.8417, doğrulama kümesindeki doğruluk 0.7212. Inception Unet için bu değerler sırasıyla 0.8448 ve 0.7250, Inception Unet-V2 için 0.8287 ve 0.7421’dir. Çizelge 4.13’de paylaşılan ortalama kesinlik, hassasiyet ve F_1

skor deęerleri incelendięinde, her iki eęitim seti iin de en yksek doęrulama deęerine sahip Inception Unet-V2 mimarisinin, en iyi performans deęerini sergiledięi grlmektedir. izelgede ilgili stn ierisinde en yksek deęere sahip veriler koyu olarak iřaretlenmiřtir.

izelge 4.13 : Massachusetts veri kmesi performans kıyaslaması (P-R-F₁)

Mimariler	Eęitim seti-I			Eęitim seti-II		
	Kesinlik(P)	Hassasiyet(R)	F ₁	Kesinlik(P)	Hassasiyet(R)	F ₁
Unet	0.7717	0.7798	0.7752	0.6605	0.7427	0.6959
Inception Unet	0.7861	0.7801	0.7822	0.6893	0.7262	0.7033
Inception Unet-V2	0.7923	0.8104	0.8009	0.6815	0.7574	0.7126



5. HİPERSPEKTRAL VERİ SETİ İLE BAND SEÇME VE BİNA BÖLÜTLEME

5.1 Hiperspektral veri seti

Bu bölümde HS veri seti üzerinden önerilen bant seçme ve sınıflama metotları test edilmiştir. Kullanılan veri seti görüntü işleme üzerine düzenlenen bir yarışmadan alınmıştır [76]. İlgili yarışmada lazer algılayıcı aracılığıyla toplanan çok bantlı görüntüler, HS görüntüler ve çok yüksek çözünürlüklü görüntüler paylaşılmıştır. Paylaşılan farklı kategorideki görüntüler için farklı yarışmalar düzenlenmiştir. Yarışma kapsamında paylaşılan HS görüntü 380 ile 1050 nm band genişliği arasında yer alan 50 band içermektedir ve 1 metre hassasiyetine sahiptir. HS görüntüden oluşturulmuş RGB görüntü ve bazı diğer bandlarına ait görüntüler şekil 5.1 ve şekil 5.2’de paylaşılmıştır.

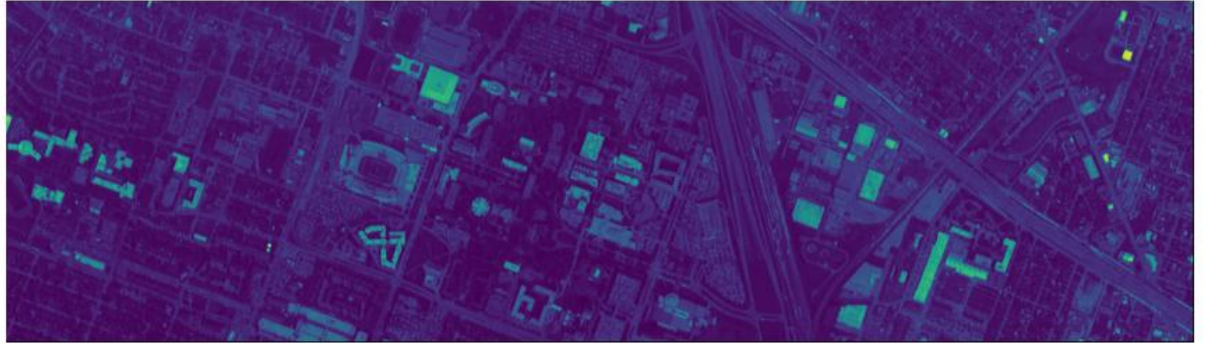


Şekil 5.1: Kullanılan HS veri setinin RGB görünümü

Farklı dalga boyunda elde edilen şekil 5.2’deki görüntülerde, her görüntünün içeriğinin birbirinden farklı olduğu görülmektedir. Bunun sebebi görüntüdeki farklı nesnelerin farklı dalga boylarındaki ışığı yansıtma/soğurma miktarlarından kaynaklanmaktadır. Bu bölümde gerçekleştirilen çalışmada, tez kapsamında önerdiğimiz bant seçme yöntemleri ile derin öğrenme mimarileri aynı problemin çözümünde bir arada kullanılmıştır.



band-1



band-5



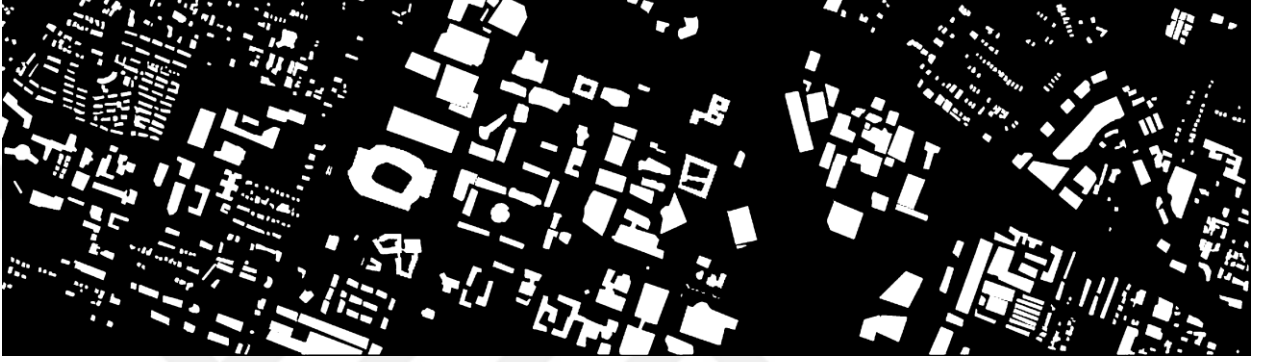
band-25



band-45

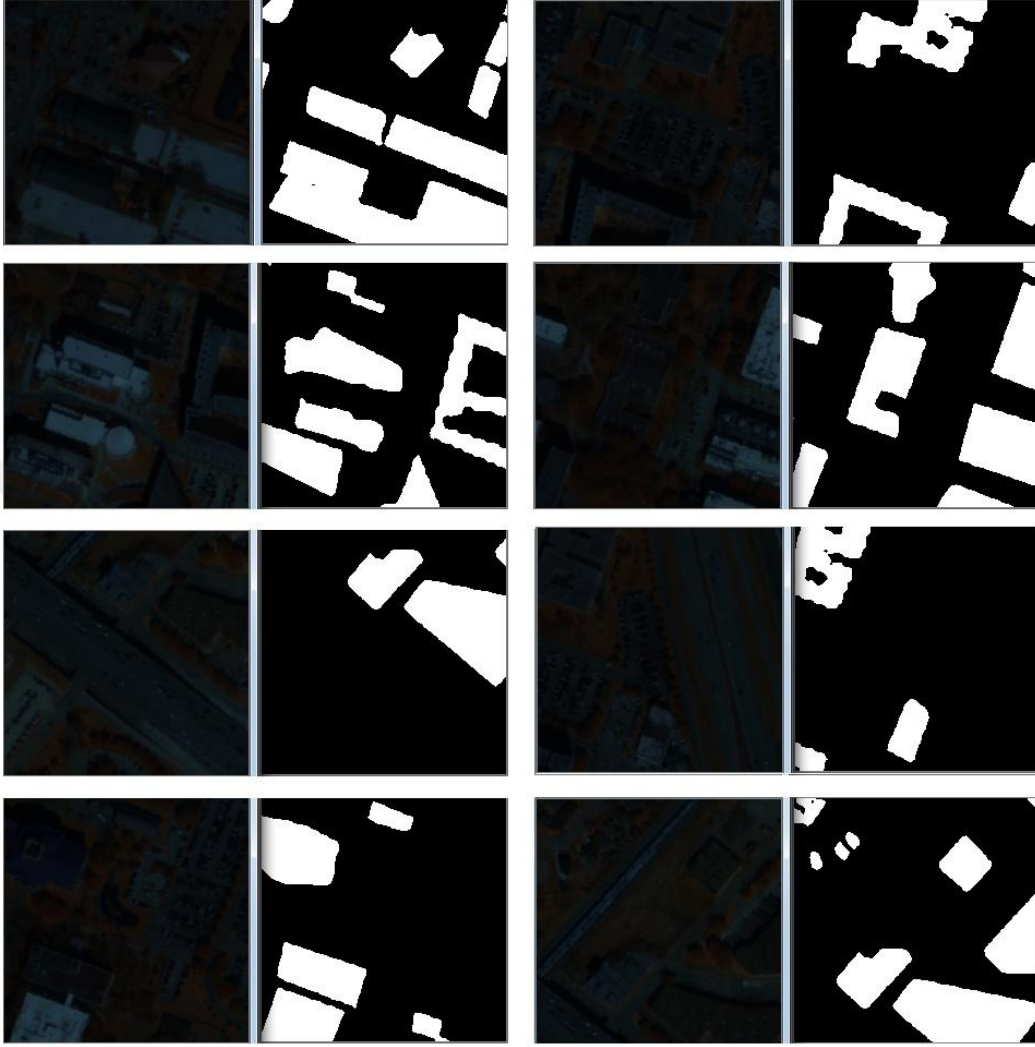
Şekil 5.2: HS veri setinden farklı bantlara ait görüntüler

Kullanılan HS görüntü 1202x4172 çözünürlüğüne sahiptir ve 224x224 boyutundaki parçalarına ayrıldığında toplam 114 farklı görüntü elde edilmektedir. Bu görüntünün tamamı için piksel bazında etiketlenmiş alanlar paylaşılmadığından, çalışmada bina sınıflama probleminde bu veri setini kullanabilmemiz için ilk önce veri seti etiketleme işlemi gerçekleştirilmiştir. Etiketlenmiş binaların görüntüsü şekil 5.3'te paylaşılmıştır.



Şekil 5.3: Etiketlenmiş binalar

Makine öğrenmesinde (dolayısıyla derin öğrenmede) gerekli olan etiketlenmiş veri hazırlandıktan sonra, ilk bölümde önerdiğimiz band seçme metotları bina bölütlemesi için önerilen mimariler ile birlikte test edilmiştir. Böylece tez kapsamında önerilen HS görüntülerden anlamlı bantları seçme ve derin öğrenme temelli bölütleme yöntemleri bir arada kullanılmıştır. Bunun için veri setinden 224x224 boyutunda rastgele seçilen 60 görüntü eğitim sürecinde kullanılmıştır. Veri setinden örnek görüntüler şekil 5.4'de paylaşılmıştır.

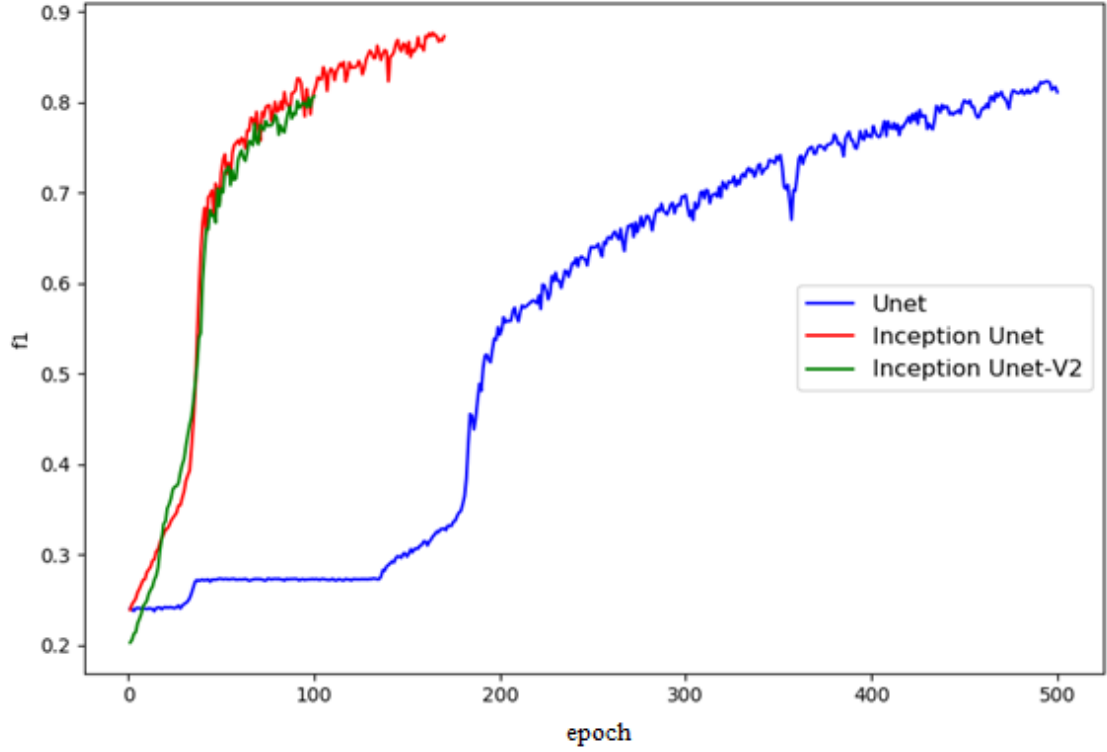


Şekil 5.4: Veri setinden örnek görüntüler

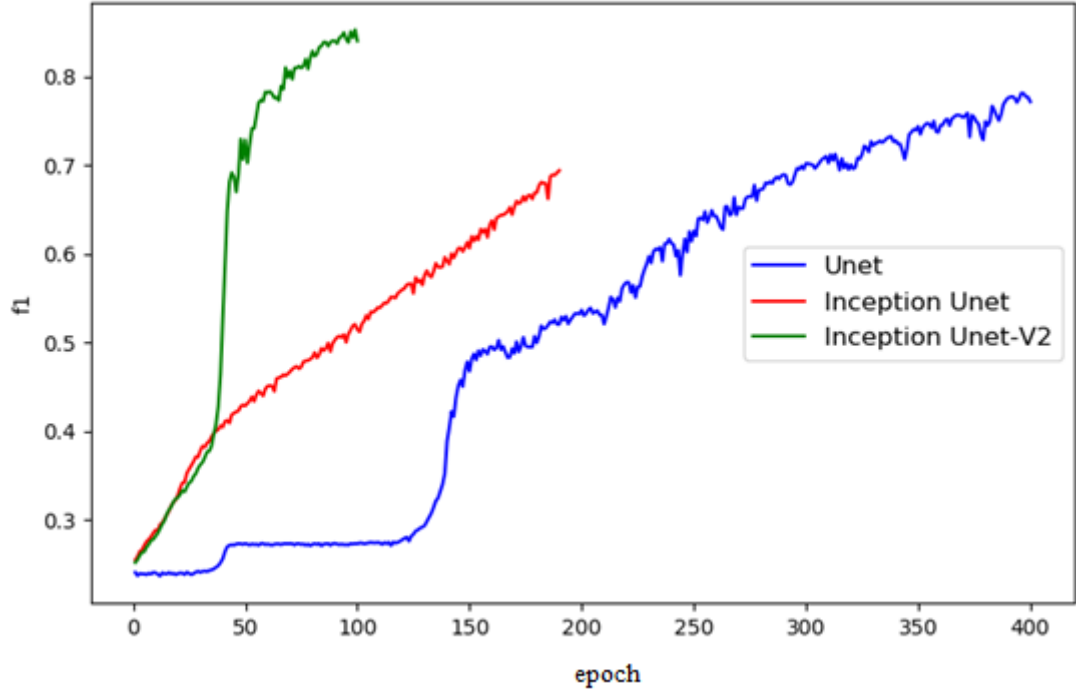
5.2 Önerilen band seçme yöntemleri ve mimariler ile deneysel sonuçlar

Tez kapsamında band seçme için problemi için IPBS ve IPDM yöntemleri, bölütleme problemi için Inception Unet-v2 mimarisi önerilmiştir. Bu bölümde hazırlanan HS veri seti ile band seçme ve bölütleme mimarileri bir arada kullanılmıştır. Bu amaçla oluşturulan eğitim seti üzerinden farklı band sayılarında girdiler kabul eden derin öğrenme modelleri hazırlanmış ve girdi olarak kullanılacak bantlar WaluMi, IPBS ve IPDM yöntemleri ile belirlenmiştir. İlk olarak HS veri setinden en anlamlı 3 band WaluMi, IPBS ve IPDM yöntemleri ile çıkarılmış ve seçilen farklı bantlara göre *Unet*, *Inception Unet* ve *Inception Unet-V2* mimarileri ile modeller eğitilmiştir. WaluMi

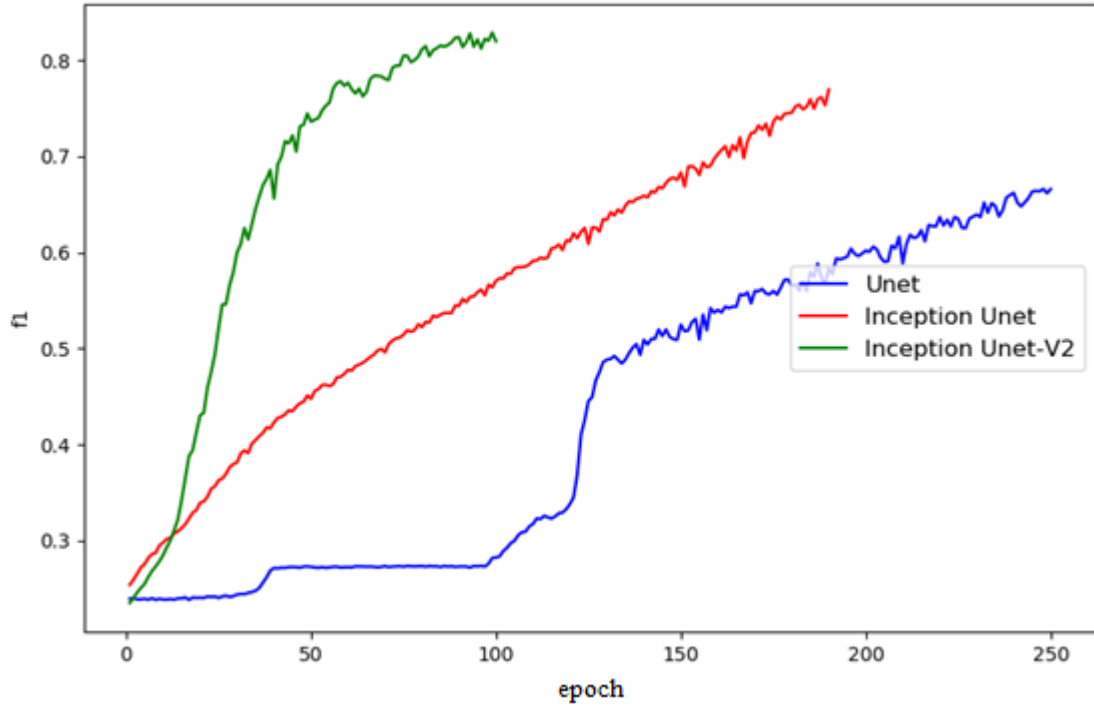
algoritması HS görüntüden [18, 38, 49] numaralı bantları seçerken, IPBS yöntemi [16, 26, 50] bantlarını, IPDM yöntemi ise [1, 6, 32] bantlarını seçmiştir. WaluMi algoritmasının seçtiği bantlar ile eğitilen modellerin eğitim grafikleri Şekil 5.5'te paylaşılmıştır. Tez kapsamında önerilen IPBS ve IPDM yöntemleri ile elde edilen bantlar kullanılarak eğitilen modellerin eğitim doğruluk grafikleri ise şekil 5.6'da ve şekil 5.7'de paylaşılmıştır.



Şekil 5.5: WaluMi yöntemi ile seçilen 3 bant ile eğitim performansları



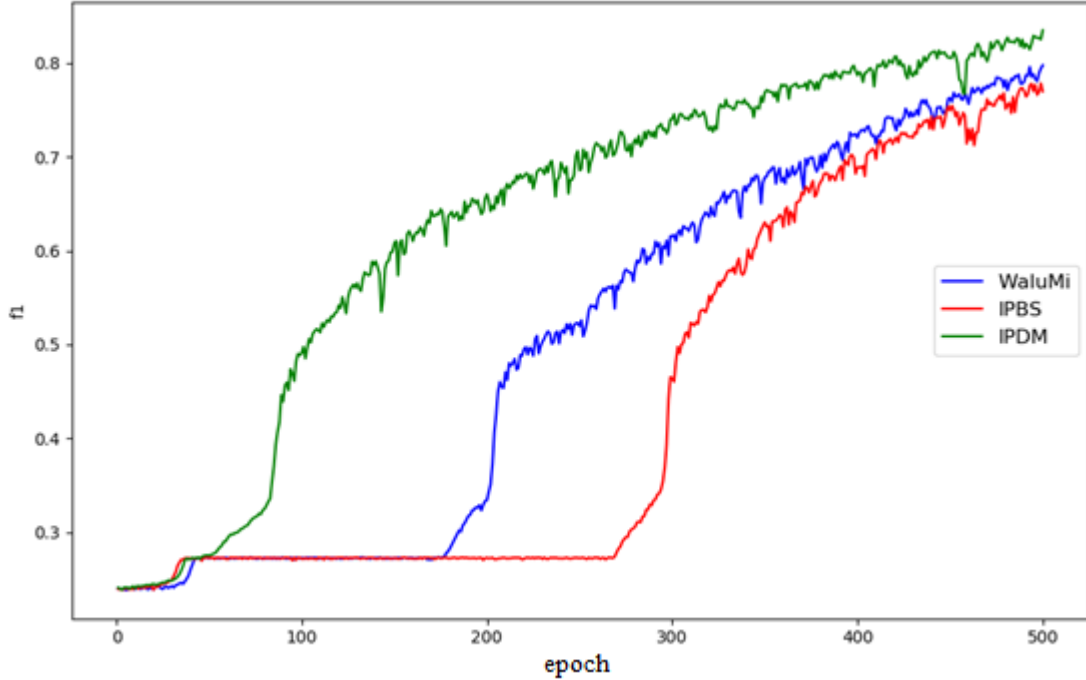
Şekil 5.6: IPBS yöntemi ile seçilen 3 bant ile eğitim performansları



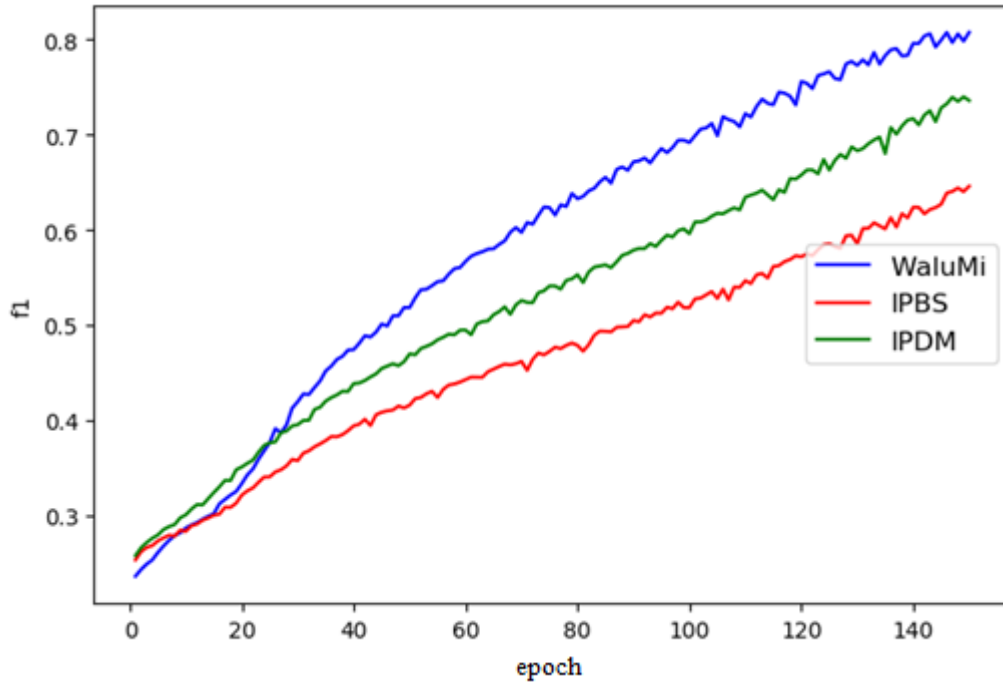
Şekil 5.7: IPDM yöntemi ile seçilen 3 bant ile eğitim performansları

Eğitim sürecinde elde edilen doğruluk grafikleri incelendiğinde benzer eğriler elde edildiği görülmektedir. Genelde *Inception Unet-V2* mimarisi ile çok daha az sayıda eğitim adımı neticesinde yüksek eğitim doğruluğu oranlarına ulaşılmaktadır. Önceki bölümde farklı eğitim seti ile eğitilmiş modeller ile de hakeza yine benzer eğitim grafikleri elde edildiği gözlemlenmiştir. Buna göre Inception modülleri eklenmiş mimarilerin daha az sayıda eğitim adımı neticesinde daha iyi doğruluk değerlerine ulaştığı görülmektedir. Önerilen derin öğrenme mimarileri 3 bant dışında farklı bant sayısında girdi görüntüleri kullanılarak da eğitilmiştir. Bunun için Walumi, IPBS ve IPDM band seçme yöntemleri ile HS görüntüden 7 ve 15 band seçilerek eğitim işlemleri gerçekleştirilmiştir. Böylece hem mimarilerin farklı band sayısındaki girdilere göre performansı, hem de band seçme yöntemlerinin kıyaslanması amaçlanmıştır.

WaluMi, IPBS ve IPDM yöntemleri ile seçilen 7 bandlı girdi görüntüleri ile eğitilen *Unet*, *Inception Unet* ve *Inception Unet-V2* eğitim grafikleri 7, 8, 9 numaralı görüntülerde paylaşılmıştır. Böylece farklı yöntemler ile seçilen bandların aynı mimari ile eğitilmesi neticesinde elde edilen eğitim grafikleri kıyaslanmıştır.

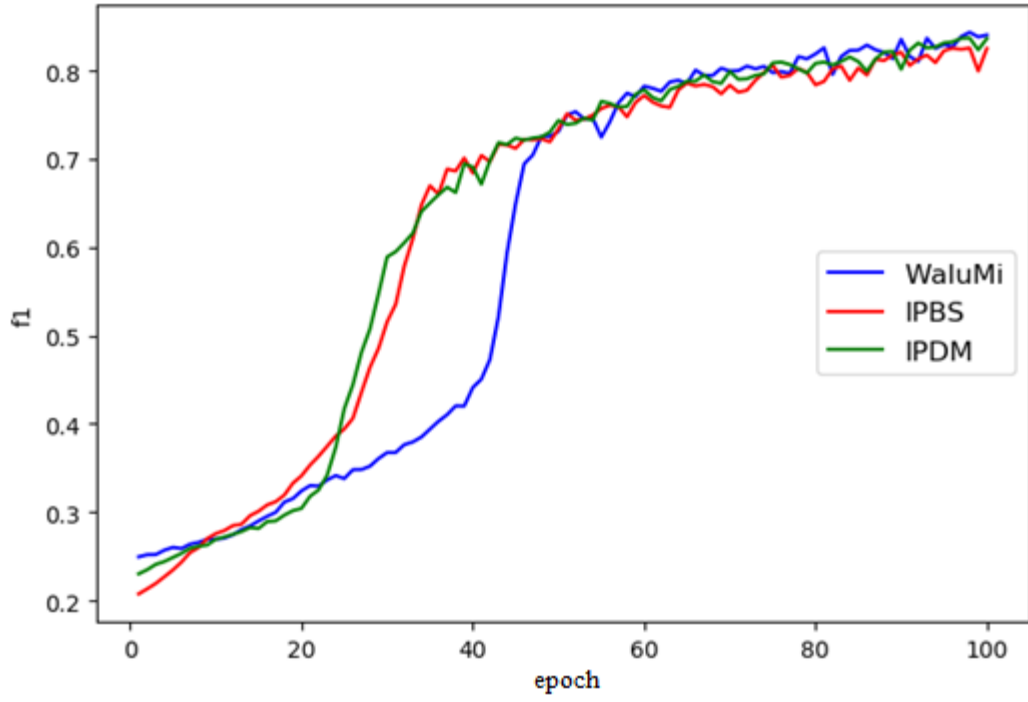


Şekil 5.8: Farklı band seçme yöntemleri ile seçilen 7 bandlı görüntüler ile eğitilen *Unet* eğitim grafiği

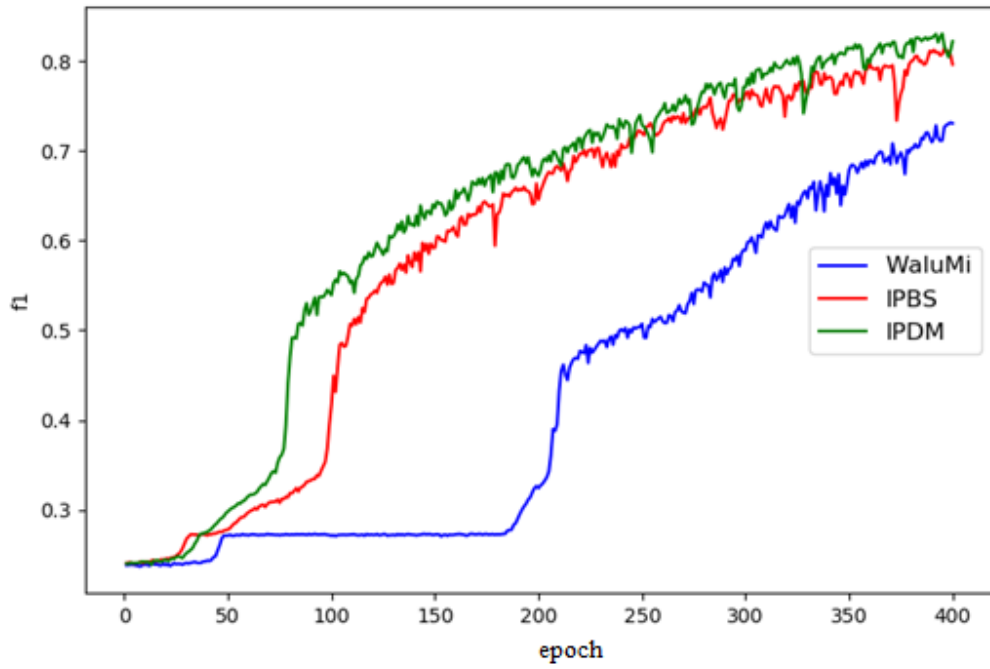


Şekil 5.9: Farklı band seçme yöntemleri ile seçilen 7 bantlı görüntüler ile eğitilen *Inception-Unet* eğitim grafiği

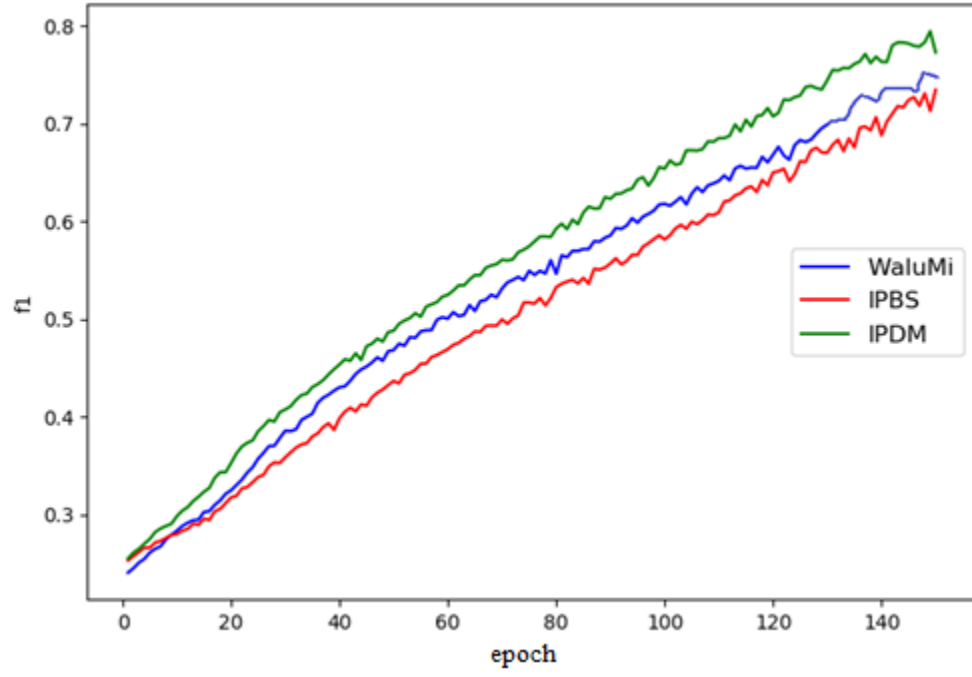
Şekil 5.8’de IPDM ile seçilen bantlar ile Unet mimarisinde en iyi eğitim doğruluğuna ulaşıldığı görülmektedir. 3 band için *Unet* mimarisi ile sırasıyla Walumi, IPBS ve IPDM ile seçilen bantlar için elde edilen test performans değerleri; 0.578, 0606 ve 0.576’tır. Buna göre en iyi performans IPBS ile seçilen bantlar ile elde edilmiştir. Şekil 5.9 incelendiğinde ise WaluMi ile seçilen bandların girdi olarak kullanıldığı *Inception Unet* mimarisinin en iyi eğitim doğruluğuna sahip olduğu görülmektedir. Fakat eğitim performansının en iyi olması test verisinde de en iyi sonucu verdiği anlamına gelmemektedir. Zira tüm veri setinde WaluMi ile seçilen bantlar ile 0.687 doğruluk değeri elde edilirken, IPBS ile 0.693, IPDM ile 0.716 doğruluk değerine ulaşılmıştır. Farklı band sayısı, farklı band seçme yöntemleri ve farklı mimariler ile elde edilen performansların detayları çizelge 1’de paylaşılmıştır. Benzer şekilde 7 band görüntüler için *Inception Unet-V2* ile elde edilen eğitim grafikleri ve 15 bantlı girdi görüntüleri ile elde edilen eğitim grafikleri aşağıda paylaşılmıştır.



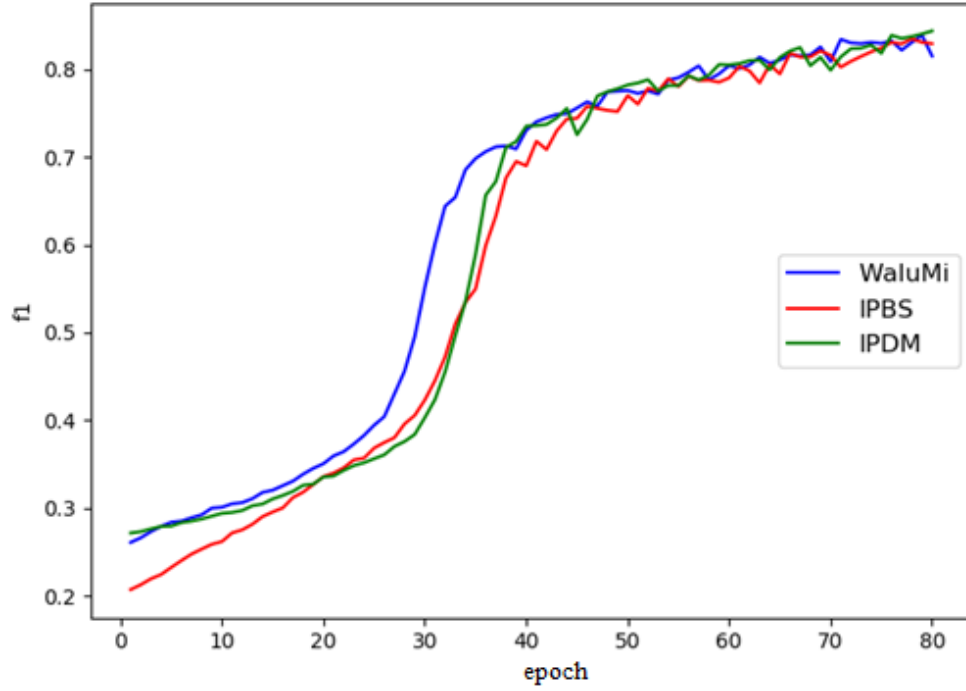
Şekil 5.10: Farklı band seçme yöntemleri ile seçilen 7 bantlı görüntüler ile eğitilen *Inception-Unet-V2* eğitim grafiği



Şekil 5.11: Farklı band seçme yöntemleri ile seçilen 15 bantlı görüntüler ile eğitilen *Unet* eğitim grafiği



Şekil 5.12: Farklı band seçme yöntemleri ile seçilen 15 bantlı görüntüler ile eğitilen *Inception-Unet* eğitim grafiği



Şekil 5.13: Farklı band seçme yöntemleri ile seçilen 15 bantlı görüntüler ile eğitilen *Inception-Unet-V2* eğitim grafiği

WaluMi, IPBS ve IPDM ile elde edilen 7 ve 15 bant girdi görüntülerinin eğitim grafikleri birbirlerine son derece benzerdir. Tüm Unet mimarilerinde eğitimin ilk adımlarda sabit gitmesinin ardından belirli bir zaman sonra aniden artmaya başladığı görülmektedir. *Inception* katmanlarının kullanıldığı mimarilerde ile daha doğrusala yakın eğitim grafikleri oluşmaktadır. WaluMi, IPBS ve IPDM yöntemleri ile seçilen 3,7 ve 15 band görüntüler için *Unet*, *Inception Unet* ve *Inception Unet-V2* mimarileri ile elde edilen f skor değerleri detayları çizelgede paylaşılmıştır. Çizelgede her bir satırda elde edilen en iyi test performans değeri koyu rakamlarla ifade edilmiştir. Ortalama satırında ise her bir sütunda (band seçme yönteminde) elde edilen ortalama f skor değeri paylaşılmıştır. Buna göre Walumi ile seçilen bantlarla eğitilen 9 modelin ortalama doğruluk değeri 0.667 iken, bu değer IPBS ile 0.672, IPDM yöntemi ile 0.679'tur.

Çizelge 5.1: Farklı bant seçme yöntemi ve mimarilerin test doğruluk oranları

Mimariler	WaluMi	IPBS	IPDM	Seçilen band sayısı
<i>Unet</i>	0.578	0.606	0.576	3 bant
<i>Inception Unet</i>	0.710	0.697	0.715	
<i>Inception UnetV2</i>	0.684	0.7256	0.719	
<i>Unet</i>	0.618	0.597	0.598	7 bant
<i>Inception Unet</i>	0.687	0.693	0.716	
<i>Inception UnetV2</i>	0.710	0.703	0.7212	
<i>Unet</i>	0.600	0.607	0.608	15 bant
<i>Inception Unet</i>	0.693	0.686	0.720	
<i>Inception UnetV2</i>	0.731	0.738	0.7455	
Ortalama	0.667889	0.672511	0.679856	

Mimari bazında kıyaslama yapıldığında ise *Inception Unet-V2* mimarisi ile en iyi sonuçların alındığı gözlemlenmiştir. Band sayısı arttıkça elde edilen doğruluk oranı da artmıştır. 15 band kullanıldığında IPDM yöntemiyle seçilen bantlar ile çok daha iyi sonuçlar alındığı görülmektedir. Band seçme yöntemlerini detaylı incelenen bölümde de IPDM'nin yüksek band sayılarında daha başarılı olduğu gözlemlenmiştir. Dolayısıyla farklı bir eğitim seti ve farklı kıyaslama yönteminin kullanıldığı bu bölümde, band

seme yntemlerini incelediėimiz blmde elde edilen ıktıları destekler nitelikte sonular alınmıřtır. rneėin 7 band ile elde edilen sonular incelendiėinde, *Inception Unet-V2* mimarisi ile IPDM yntemi kullanılarak seilen bantlar ile 0.7212 performans deėeri elde edilirken, diėer yntemler ile 0.71 ve 0.703 deėerleri elde edilmiřtir. Hakeza 15 band iin IPDM ile 0.7455 performans deėerine ulařılırken, WaluMi ve IPBS ile sırasıyla 0.731 ve 0.738 f skor deėerlerine ulařılmıřtır. Sonu itibariyle farklı bir veri seti zerinde gerekleřtirilen testler neticesinde, tez kapsamında nerdiėimiz band seme algoritmaları ve derin ėrenme mimarileri aynı problem ierisinde bir arada kullanılmıř ve nerilen yntemlerin performansları test edilmiřtir.

6. SONUÇ VE DEĞERLENDİRME

Tez kapsamında HS görüntülerde bant seçme problemi için iki yöntem ile birlikte, bölütleme problemlerinde kullanılabilecek derin öğrenme mimarisi önerilmiştir. Bant seçme problemi iki aşamalı olarak düşünülmüş, ilk aşamada hiyerarşik kümeleme ile benzer bilgiler içeren bantlar kümelere ayrılmıştır. Bu aşamada literatürdeki Walumi yöntemi ile benzer bir yaklaşım tercih edilmiştir. Oluşturulan kümelerden en fazla bilgiyi içeren bantı seçmek için ise IPBS diye adlandırılan bir yöntem geliştirilmiştir. Buna göre kümelerdeki görüntülerin yapısal özellikleri de dikkate alınarak en ayırt edici bilgiyi içeren bandın ilgili kümeyi temsilen seçilmesi hedeflenmiştir. Önerilen bir diğer bant seçme metodu olan IPBS’de ise, hiyerarşik kümelemede kullanılabilecek bir benzerlik kriteri tanımlanmıştır. Böylece bantların kümelere ayrılma sürecinde Walumi yönteminden farklı bir yöntem izlenmiştir. Bant seçme problemi için önerilen metotlar iki farklı HS görüntü üzerinde test edilmiş ve literatürde kabul gören Walumi ve Waludi yöntemleri ile performans kıyaslamaları gerçekleştirilmiştir. Performans kıyaslaması için seçilen bantlar ile k-NN algoritması kullanılarak piksel bazlı sınıflandırma işlemi gerçekleştirilmiştir. Sınıflama işlemi neticesinde genel ve ortalama doğruluk değerleri analiz edilmiş, önerilen yöntemler ile seçilen bantlar kullanıldığında yaklaşık %3 civarında daha iyi sonuçlar alınabildiği gözlemlenmiştir.

Diğer yandan, literatürdeki bina bölütlemesi konusunda gerçekleştirilen çalışmalar incelenmiş, Unet ve gölge tabanlı doğrulama içeren yöntem literatürdeki klasik görüntü işleme temelli yöntemler ile kıyaslanmıştır. Makine öğrenmesi yöntemleri ile özellik çıkarımının, uzmanlık ve detaylı analiz gerektiren klasik yöntemlere göre daha avantajlı olduğu anlaşılmıştır. Klasik yöntemlerin karşılaştığı zorluklar göz önüne alındığında, az sayıda görüntü ile eğitilen modellerin bile bina bölütlemesi probleminde çok daha başarılı olduğu görülmüştür. Bunun yanında, nesne bölütlemesi için derin öğrenme ile gerçekleştirilen çalışmalar incelenmiş ve nesne tanıma problemlerinde kullanılan bazı yaklaşımların (Inception blokları gibi) bölütleme amaçlı tasarlanan mimarilerde de kullanıldığı görülmüştür. Literatürdeki yöntemlere benzer şekilde kullandığımız Inception blokları ile genişlik kazandırılan Unet mimarisinin bölütleme probleminde başarısının arttığı gözlemlenmiştir. Bu kapsamda görüntüden özellik çıkarımı

aşamasında Inception bloklarının klasik Unet katmanları ile paralel olarak kullanıldığı özgün bir mimari geliştirilmiştir. Önerilen mimaride özellik çıkarımında yer alan Inception ve klasik Unet filtre sonuçlarının her ikisi birden görüntünün yeniden oluşturulduğu katmanlara aktarılmıştır. Böylece özellik çıkarımında paralel katmanlarda hesaplanan özellikler sonraki katmanlara aktarılmıştır. Ayrıca, önerilen Inception Unet-v2 mimarisi ile hem eğitilecek parametre sayısı azaltılmış hem de performans artışı sağlanmıştır. Bu kapsamda, iki farklı MS veri seti üzerinden performans kıyaslamaları gerçekleştirilmiştir. RGB ve NIR görüntüler içeren ilk veri setinde gerçekleştirilen performans kıyaslamalarında, önerilen mimari ile F_1 skor değerinin %1 ile %3 arasında arttığı gözlemlenmiştir. Daha fazla görüntü içeren Massachusetts veri kümesinde ise performans artışı %1 ile %2 civarındadır. Test edilen mimarilerin çok çıkışlı versiyonları da farklı büyüklüklerdeki binalar için eğitilmiş, düşük miktarda de olsa performans artışı sağlandığı gözlemlenmiştir.

Son bölümde ise HS görüntü üzerinde bina etiketlemesi gerçekleştirilerek makine öğrenmesinde kullanılabilecek örnek bir HS veri seti hazırlanmıştır. Etiketlenen bu veri seti üzerinde, bant seçme ve bina bölütleme işlemleri art arda uygulanmıştır. Böylece tez kapsamında önerilen IPBS ve IPDM bant seçme yöntemleri ile Inception Unet-v2 mimarisi bir arada kullanılmıştır. Bu kapsamda önerilen metotların performansları farklı bir veri seti üzerinden de test edilmiştir. Ayrıca bant seçme yöntemlerinin başarısı da bir anlamda derin öğrenme yöntemlerinin eğitim başarısı üzerinden test edilmiştir. Bunun için; etiketlenen HS veri kümesinden 3, 7 ve 15 bant Walumi, IPBS ve IPDM yöntemlerinin her biri ile seçilmiştir. Seçilen farklı bant numaraları için her bir mimari ile eğitim işlemleri gerçekleştirilmiş ve ortalama doğruluk değerleri hesaplanmıştır. Sonuçlar incelendiğinde, HS veri seti üzerinde de mimari bazında Inception Unet-v2 ile en iyi performans değerlerinin elde edildiği görülmüştür.

Sonuç itibarıyla, tez kapsamında uzaktan algılanmış MS ve HS görüntülerde bant seçme ve bölütleme problemleri üzerinde literatüre katkıda bulunulmuş, önerilen derin öğrenme mimarisi ile ağ büyüklüğü ve performans noktasında iyileştirmeler sağlandığı görülmüştür.

KAYNAKLAR

- [1] Sun W. D., *Hyperspectral Imaging for Food Quality Analysis*, Elsevier Science, London, 2010.
- [2] Edelman G. J., Van Leeuwen T. G., Aalders M. C. G., *Hyperspectral imaging for age estimation of blood stains at crime scene*, *Forensic Science International*, 223, 72-77, 2012.
- [3] Reed G., K. Savage, D. Edwards, N. Nic Daeid, *Hyperspectral imaging of gel pen inks: an emerging tool in document analysis*, *Sci. Justice*, 54, pp. 71–80, 2014.
- [4] Krause F. A., Boardman J. W., Hungtinton J. F., *Comparison of airborne hyperspectral data and EO-1 Hyperion for mineral mapping*, *IEEE Trans. on Geo. and Rem. Sens.*, 41, 1388-1400, 2003.
- [5] Lopes M. B., Bioucas-Dias J. M., Figueiredo M. A. T., Wolff J. C., *Spectral unmixing via minimum volume simplices: Application to near infrared spectra of counterfeit tablets*, *First Works. On Hyper. Imag. and Signal Proc. Evol. In Rem. Sens. (WHISPERS)*, Gronoble, France, 1-4 August 2009.
- [6] Zhang L., Y. Zhong, B. Huang, J. Gong, and P. Li, “Dimensionality reduction based on clonal selection for hyperspectral imagery,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4172–4186, Dec. 2007.
- [7] Farrell M. D. and R. M. Mersereau, “On the impact of PCA dimension reduction for hyperspectral detection of difficult targets,” *IEEE Geosci. Remote Sens. Lett.*, vol. 2, no. 2, pp. 192–195, Apr. 2005.
- [8] S. Jia, Z. Ji, Y. Qian, and L. Shen, “Unsupervised band selection for hyperspectral imagery classification without manual band removal,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 531–543, Apr. 2012.
- [9] A. Martinez Usó, F. Pla, J. M. Sotoca, and P. García-Sevilla, “Clustering based hyperspectral band selection using information measures,” *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4158–4171, Dec. 2007.
- [10] C. Vaiphasa, A. K. Skidmore, W. F. de Boer, and T. Vaiphasa, “A hyperspectral band selector for plant species discrimination,” *ISPRS J. Photogramm. Remote Sens.*, vol. 62, no. 3, pp. 225–235, Aug. 2007.
- [11] S. Patra, P. Modi, and L. Bruzzone, “Hyperspectral band selection based on rough set,” *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 10, pp. 5495–5503, Oct. 2015.

- [12] H. Yang, Q. Du, H. Su, and Y. Sheng, "An efficient method for supervised hyperspectral band selection," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 1, pp. 138–142, Jan. 2011.
- [13] W. Xia, B. Wang, and L. Zhang, "Band selection for hyperspectral imagery: A new approach based on complex networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 5, pp. 1229–1233, Sep. 2013.
- [14] M. Ghamary Asl, M. R. Mobasheri, and B. Mojaradi, "Unsupervised feature selection using geometrical measures in prototype space for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 7, pp. 3774–3787, Jul. 2014.
- [15] C.I. Chang and K.-H. Liu, "Progressive band selection of spectral unmixing for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 4, pp. 2002–2017, Apr. 2014.
- [16] John H. Ward. Hierarchical grouping to optimize an objective function. American Statistical Association, 58(301):236–244, 1963.
- [17] A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Prentice-Hall, 1988.
- [18] Y. Qian, F. Yao, and S. Jia, "Band selection for hyperspectral imagery using affinity propagation," *IET Comput. Vis.*, vol. 3, no. 4, pp. 213–222, Dec. 2009.
- [19] R. Hedjam and M. Cheriet, "Hyperspectral band selection based on graph clustering," in *Proc. 11th Int. Conf. Inf. Sci., Signal Process. Appl.*, pp. 813–817, 2012.
- [20] K. Sun, X. Geng, and L. Ji, "Exemplar component analysis: A fast band selection method for hyperspectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 5, pp. 998–1002, May 2015.
- [21] Zhu, G., Huang, Y., Lei, J., Bi, Z., & Xu, F., Unsupervised Hyperspectral Band Selection by Dominant Set Extraction. *Geoscience and Remote Sensing, IEEE Transactions on*, 54(1), 27-239, 2016.
- [22] Hyperspectral Remote Sensing Scenes, http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes, 2020
- [23] C. C. Chang, *Hyperspectral Data Exploitation: Theory and Applications*. Hoboken, NJ, USA: Wiley-Interscience, 2007.
- [24] Comon, Pierre. "Independent component analysis, a new concept?." *Signal processing* 36.3: 287-314, 1994.

- [25] Hyvärinen, Aapo, and Erkki Oja. "A fast fixed-point algorithm for independent component analysis." *Neural computation* 9.7: 1483-1492, 1997.
- [26] Cover, Thomas M., and Joy A. Thomas. "Information theory and statistics." *Elements of Information Theory* : 279-335, 1991.
- [27] Ou, T., Huang, Y., Chen, H., SSIM-Based perceptual rate control for video coding, *IEEE transactions on circuits and systems for video technology*, Vol. 21, NO. 5., 2011.
- [28] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
- [29] Dorado-Munoz, L. P., Velez-Reyes, M., Mukherjee, A., & Roysam, B., A vector SIFT detector for interest point detection in hyperspectral imagery. *Geoscience and Remote Sensing, IEEE Transactions on*, 50(11), 4521-4533, 2012.
- [30] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. IT-13, no. 1, pp. 21–27, Jan. 1967.
- [31] Sui, C., Tian, Y., Xu, Y., & Xie, Y., Unsupervised band selection by integrating the overall accuracy and redundancy. *Geoscience and Remote Sensing Letters, IEEE*, 12(1), 185-189, 2015.
- [32] Lin, Chungan, and Ramakant Nevatia. "Building detection and description from a single intensity image." *Computer vision and image understanding* 72.2 : 101-121, 1998.
- [33] Izadi, Mohammad, and Parvaneh Saeedi. "Three-dimensional polygonal building model estimation from single satellite images." *IEEE Transactions on Geoscience and Remote Sensing* 50.6: 2254-2272, 2012.
- [34] Irvin, R. Bruce, and David M. McKeown. "Methods for exploiting the relationship between buildings and their shadows in aerial imagery." *OE/LASE'89, 15-20 Jan., Los Angeles. CA. International Society for Optics and Photonics*, 1989.
- [35] Kim, Taejung, and Jan-Peter Muller. "Development of a graph-based approach for building detection." *Image and Vision Computing* 17.1: 3-14, 1999.
- [36] Liow, Yuh-Tay, and Theo Pavlidis. "Use of shadows for extracting buildings in aerial images." *Computer Vision, Graphics, and Image Processing* 49.2: 242-277, 1990.

- [37] Sirmacek, Beril, and Cem Unsalan. "Building detection from aerial images using invariant color features and shadow information." *Computer and Information Sciences, 2008. ISCIS'08. 23rd International Symposium on*. IEEE, 2008.
- [38] Akçay, H. Gökhan, and Selim Aksoy. "Building detection using directional spatial constraints." *Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International*. IEEE, 2010.
- [39] Ok, Ali Ozgun. Automated detection of buildings from single VHR multispectral images using shadow information and graph cuts. *ISPRS journal of photogrammetry and remote sensing*, 86: 21-40, 2013.
- [40] Ok, Ali Ozgun, Caglar Senaras, and Baris Yuksel. "Automated detection of arbitrarily shaped buildings in complex environments from monocular VHR optical satellite imagery." *IEEE Transactions on Geoscience and Remote Sensing* 51.3: 1701-1717, 2013.
- [41] Rother, C., Kolmogorov, V., Blake, A. "Grabcut: interactive foreground extraction using iterated graph cuts". *ACM Transactions on Graphics* 23 (3), 309–314, 2004.
- [42] Ok, Ali Ozgun, and Andrea Manno-Kovacs. "Detection of buildings from high resolution satellite images using urban area knowledge." *Signal Processing and Communications Applications Conference (SIU), 2015 23th*. IEEE, 2015.
- [43] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.
- [44] Polidorio, A. M., Flores, F. C., Imai, N. N., Tommaselli, A. M., & Franco, C. (2003, October). Automatic shadow segmentation in aerial color images. In *Computer Graphics and Image Processing, 2003. SIBGRAPI 2003. XVI Brazilian Symposium on* (pp. 270-277). IEEE, 2013.
- [45] Tsai, Victor JD. "A comparative study on shadow compensation of color aerial images in invariant color models." *IEEE transactions on geoscience and remote sensing* 44.6: 1661-1671, 2006.
- [46] Teke, M., Başeski, E., Ok, A. Ö., Yüksel, B., & Şenaras, Ç., Multi-spectral false color shadow detection. In *Photogrammetric Image Analysis* (pp. 109-119). Springer Berlin Heidelberg, 2011.
- [47] Otsu, Nobuyuki. "A threshold selection method from gray-level histograms." *IEEE transactions on systems, man, and cybernetics* 9.1 (1979): 62-66.

- [48] Tucker, Compton J. Red and photographic infrared linear combinations for monitoring vegetation. *Remote sensing of Environment*, 8.2: 127-150, 1979
- [49] Chen, Y., Lin, Z., Zhao, X., Wang, G., Gu, Y.: Deep learning-based classification of hyperspectral data. *Selected Topics in Applied Earth Observations and Remote Sensing*, IEEE Journal of 7, 2094–2107, 2014.
- [50] Makantasis, K., Karantza, K., Doukakis, A., Doukakis, N.: Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2015)*, 2015.
- [51] Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: *Advances in NIPS 19*, 153–160, 2007.
- [52] Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. *Science* 313, 504–507, 2006.
- [53] Felzenszwalb, Pedro F., and Daniel P. Huttenlocher. "Efficient graph-based image segmentation." *International journal of computer vision* 59.2 (2004): 167-181.
- [54] C. Benedek, X. Descombes, and J. Zerubia, "Building detection in a single remotely sensed image with a point process of rectangles," in *Proc. ICPR*, pp. 1417–1420, 2010.
- [55] Rosenblatt, Frank. "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review* 65.6 (1958): 386.
- [56] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.
- [57] ImageNet, <http://www.image-net.org/>, 2020
- [58] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.
- [59] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European conference on computer vision*. Springer, Cham, 2014.
- [60] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9), 2015.

- [61] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).
- [62] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- [63] Convolutional Neural Networks, <http://fourier.eng.hmc.edu/e176/lectures/ch10/node8.html>, 2020
- [64] Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." *arXiv preprint arXiv:1511.00561* (2015).
- [65] Microsoft sunar: Deep Residual Network, <https://medium.com/@bakiiii/microsoft-sunar-deep-residual-network-d2970003ad8b>, 2020
- [66] Xu, Yongyang, Liang Wu, Zhong Xie, and Zhanlong Chen. "Building Extraction in Very High Resolution Remote Sensing Imagery Using Deep Learning and Guided Filters." *Remote Sensing* 10, no. 1 (2018): 144.
- [67] Deep Learning for Semantic Segmentation of Aerial and Satellite Imagery, <https://www.azavea.com/blog/2017/05/30/deep-learning-on-aerial-imagery>, 2020
- [68] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).
- [69] Audebert, Nicolas, Bertrand Le Saux, and Sébastien Lefèvre. "Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks." *ISPRS Journal of Photogrammetry and Remote Sensing* 140 (2018): 20-32.
- [70] M. Cramer. The DGPF test on digital aerial camera evaluation – overview and test design. *Photogrammetrie – Fernerkundung – Geoinformation*, 2:73–82, 2010
- [71] Chen, C., Gong, W., Hu, Y., Chen, Y., & Ding, Y. Learning Oriented Region-based Convolutional Neural Networks for Building Detection in Satellite Remote Sensing Images. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 461., 2017.
- [72] N. Singh Pun and S. Agarwal, "Inception U-Net Architecture for Semantic Segmentation to Identify Nuclei in Microscopy Cell Images", *ACM Trans. Multimedia Comput. Commun.*, 16(1), 15 pages.

[73] D.E.Cahall,G. Rasool,N.C. Bouaynaya, andH.M. Fathallah-Shaykh, “Inception Modules Enhance Brain Tumor Segmentation”, *Frontiers in computational neuroscience*,13, 44 (2019).

[74] Hamaguchi, Ryuhei, and Shuhei Hikosaka. "Building detection from satellite imagery using ensemble of size-specific detectors." *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2018.

[75] Mnih, V.: Machine learning for aerial image labeling. Ph.D. thesis, University of Toronto(2013)

[76] Xu, Yonghao, Bo Du, Liangpei Zhang, Daniele Cerra, Miguel Pato, Emiliano Carmona, Saurabh Prasad, Naoto Yokoya, Ronny Hansch, and Bertrand Le Saux. "Advanced Multi-Sensor Optical Remote Sensing for Urban Land Use and Land Cover Classification: Outcome of the 2018 IEEE GRSS Data Fusion Contest." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (2019).



ÖZGEÇMİŞ



Ad Soyad: İbrahim DELİBAŞOĞLU
Doğum Yeri ve Tarihi: ERZURUM / 1987
E-Posta: ibrahimdelibasoglu@gmail.com
Lisans: Kocaeli Üniversitesi
Yüksek Lisans: Yalova Üniversitesi

Mesleki Deneyim ve Ödüller:

- Proje Araştırmacısı, Tek Bir Dc Güç Kaynağından Beslenen 8-Seviyeli Tek-Fazlı Çoklu-Kaskad Evirici Tasarımı, TÜBİTAK, 113E298, 2013–2014.
- TeknoGirişimci, Taşınabilir Yapıda Trafik Kayıt ve Analiz Cihazı Tasarımı. T.C. Bilim Sanayi ve Teknoloji Bakanlığı TeknoGirişim Sermaye Desteği Programı, 2012-2013.
- Yenilikçi Yazılımlar Yarışıyor yarışması teşvik ödülü, ODTÜ Görüntü Analizi Uygulama ve Araştırma Merkezi ve Savunma Sanayii Başkanlığı, 2019

Yayın ve Patent Listesi:

Delibasoglu, Ibrahim, and Mufit Cetin. "Hyperspectral band selection using structural information via hierarchical clustering." *Journal of Applied Remote Sensing* 13.1 (2019): 014526.

Delibaşoğlu İ., Kösesoy İ., Akbaş A., "Real-Time Traffic Data Acquisition by Using GMR Sensors; Towards a Design of Real-Time Traffic Analyze System", 19th ITS World Congress, Vienna, Austria, 22/26 October 2012

Delibaşoğlu, İ., Kösesoy, İ., İlhan, O., Çetin, Ö., "A Comparative Study of Different IHS Transformation Models with a Tradeoff Parameter", Seoul International Conference on Applied Science and Engineering (SICASE), Seoul, South Korea, 29-31 August, 2014.

Avcı C., Delibaşoğlu İ., Akbaş A., "Sleep Apnea Detection Using Wavelet Analysis of ECG Derived Respiratory Signal", The International Conference on Biomedical Engineering (ICOB 2012), Perlis-Malaysia, 27-28 February 2012.

Delibasoglu, I., Avcı, C., Akbaş, A., "ECG Based Sleep Apnea Detection Using Wavelet Analysis of Instantaneous Heart Rates", 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies, ISABEL 2011, Barcelona, Spain, 26-29 Oct 2011.

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

Dergideki Bir Makale:

- [1] Delibasoglu, Ibrahim, and Mufit Cetin. "Hyperspectral band selection using structural information via hierarchical clustering." *Journal of Applied Remote Sensing* 13.1 (2019): 014526.