



Burdur Gölü Su Yüzey Alanının Coğrafi Bilgi Sistemleri ve Derin Öğrenme ile İncelenmesi

Yazılım Mühendisliği Ana

Bilim Dalı Dönem Projesi

Kübra Gül İbacık

Y220240099

Proje Danışmanı: Prof. Dr. Femin Yalçın Küçükbayrak

Burdur Gölü Su Yüzey Alanının Coğrafi Bilgi Sistemleri ve Derin Öğrenme ile İncelenmesi

ÖZ

Bu proje, Burdur Gölü'nün yüzey değişikliklerini detaylı bir şekilde inceleyerek, Coğrafi Bilgi Sistemleri (CBS) ve derin öğrenme tekniklerini entegre ederek analiz etmeyi amaçlamaktadır. Projenin temel odak noktası, 2015 ile 2023 yılları arasında elde edilen uydu görüntüleri üzerinden U-Net derin öğrenme modeli ile yüzey suyu değişikliklerini belirlemek ve haritalamaktır. Bu analiz, göl ekosistemine dair kritik bilgiler sağlamayı, su kaynaklarının yönetimine katkıda bulunmayı ve gelecekteki olası değişikliklere karşı öngörülerde bulunmayı amaçlamaktadır.

Anahtar Sözcükler: Burdur Gölü, Yüzey Değişikliği, Coğrafi Bilgi Sistemleri (CBS), Derin Öğrenme, U-Net Modeli, Uydu Görüntüleri, Haritalama

Analysis of the Surface Area of Lake Burdur with Geographic Information Systems and Deep Learning

Abstract

This project aims to extensively investigate surface changes of Lake Burdur by integrating Geographic Information Systems (GIS) and deep learning techniques. The primary focus of the project is to analyze surface water changes using U-Net deep learning models based on satellite imagery obtained between 2015 and 2023. The objective of this analysis is to provide critical insights into the lake ecosystem, contribute to water resource management, and make predictions regarding potential future changes.

Keywords: Lake Burdur, Surface Change, Geographic Information Systems (GIS), Deep Learning, U-Net Model, Satellite Imagery, Mapping

Teşekkür

Proje çalışmasına katkılarından dolayı Prof. Dr. Femin Yalçın Küçükbayrak' a teşekkür ederim.

İçindekiler

Öz.....	i
Abstract.....	ii
Teşekkür.....	iv
Şekiller Listesi.....	vii
Tablolar Listesi.....	viii
Kısaltmalar Listesi	ix
1 Giriş	1
2 Burdur Gölü ve Su Yüzeyi Değişikliği.....	4
2.1 Çalışma Alanı	4
2.2 İklim Değişikliği ve Burdur	5
3 Materyal ve Yöntem	8
3.1 Veri Toplama	8
3.1.1 Sentinel-2 Uydu Görüntüleri	8
3.1.2 Coğrafi bilgi Sistemleri ve Uydu Görüntüleri ile Burdur Gölü Değişim Analizi.....	10
4 Derin Öğrenme	16
4.1 Derin Öğrenme Modelleri ile Burdur Gölü Yüzey Değişikliği Analizi.....	16
4.1.1 U-Net Mimarisi	17
4.1.1.2 U-Net Modeli Eğitimi ve Bulgular.....	19
4.1.2 Otomatik Kodlayıcı (Autoencoder).....	20
5.Sonuçlar ve Tartışma.....	24
Kaynaklar	26
Ekler	27

Şekiller Listesi

Şekil 2.1	1979-2023 yılları aralığında Burdur'un yıllık ortalama sıcaklık değişimi	5
Şekil 2.2	1979-2023 yılları aralığında Burdur'un yıllık ortalama yağış değişimi	6
Şekil 2.3	2015-2023 yılları aralığında Burdur Gölü uydu görüntüsü	7
Şekil 3.1.1	Raster Modeli ve Vektör Modeliyle Gösterim.....	10
Şekil 3.1.2	ArcGIS programı arayüzü.....	11
Şekil 3.1.3	2015 yılı sınıflandırılmış uydu görüntüsü.....	12
Şekil 3.1.4	2023 sınıflandırılmış uydu görüntüsü	13
Şekil 3.1.5	2015-2023 yılları aralığında Burdur Gölü Yüzey Alanı Değişimi Haritası	14
Şekil 3.1.6:	2015 yılı Burdur Gölü Sentinel-2 uydu görüntüsü haritası.....	15
Şekil 3.1.7 :	2023 yılı Burdur Gölü Sentinel-2 uydu görüntüsü haritası.....	15
Şekil 4.1:	U-Net modelinin katmanlar	18
Şekil 4.2 :	U-Net modelinin sınıflandırma test sonucu görüntüsü	20
Şekil 4.3:	U-Net modelinin eğitim sonucu kayıp ve doğruluk grafikleri.....	20
Şekil 4.5:	Veri ön işleme adımı	21
Şekil 4.6:	Modelin image ve mask olarak ayırımı	22
Şekil 4.7 :	Autoencoder modeli eğitim sonrası görselleştirmesi.....	23

Tablolar Listesi

Tablo 3.1 : 2015-2023 yılları aralığında Burdur Gölü Yüzey Alanı Deęimi	13
Tablo 4: U-Net modelinin sınıflandırma raporu	91

Kısaltmalar Listesi

CBS	Coğrafi Bilgi Sistemleri
NDWI	Normalize Edilmiş Su İndeksi

Bölüm 1

1.Giriş

Burdur Gölü, Türkiye'nin Göller Bölgesi'nde yer alan ve biyoçeşitliliği ile ön plana çıkan önemli bir iç su kaynağıdır. Göl, Ramsar alanı ve Yaban Hayatı Geliştirme Sahası statüsüne sahip olmasıyla ekosistem açısından büyük bir değer taşır. Bu bölge, uluslararası öneme sahip sulak alanlar arasında kabul edilen 13 Ramsar alanından biridir. Aynı zamanda, dikkuyruk (*Oxyura leucocephala*) gibi nesli tehlike altındaki kuş türlerine kışlama alanı sağlamakta ve endemik balık türleri barındırmaktadır (Arık, 2006). Ancak, son yıllarda Burdur Gölü'nde yaşanan yüzey suyu çekilmeleri, göl ekosisteminde önemli değişikliklere neden olmuştur.

Burdur Gölü'nün yüzey suyu çekilmesi, çeşitli etkenlere bağlı olarak ortaya çıkabilir. İklim değişiklikleri, artan sulama ihtiyacı, endüstriyel faaliyetler ve su kaynaklarının kullanımı gibi faktörler, gölün su seviyesindeki düşüşe katkıda bulunabilir. Bu durum, hem göl ekosisteminin dengesini tehdit ederken hem de su kaynaklarının sürdürülebilirliği açısından endişe vericidir.

Coğrafi Bilgi Sistemleri (CBS), su kaynakları yönetiminde önemli bir araç olarak öne çıkar. CBS, göl yüzey değişikliklerini detaylı bir şekilde izleyebilme ve analiz edebilme kapasitesiyle su kaynakları yönetimine kritik katkılarda bulunur. Yüzey suyu çekilmesinin CBS üzerindeki etkisi, değişen su seviyelerini belirleme, olası tehditleri değerlendirme ve koruma stratejileri geliştirme açısından büyük önem taşır.

Derin öğrenme modelleri, su yüzeyi değişikliklerini tespit etme konusunda giderek daha etkili hale gelmiştir. U-Net modeli, özellikle görsel veri analizinde ve segmentasyon uygulamalarında başarılı sonuçlar elde etmiştir. Bu model, karmaşık görsel veri setlerinden anlamlı bilgiler çıkarmak ve su yüzeyi değişikliklerini hassas bir şekilde belirlemek için kullanılabilir. Sentinel-2 uydusundan elde edilen yüksek çözünürlüklü uydu görüntüleri, U-Net modelinin su yüzeyi değişikliklerini tespit etme yeteneğini güçlendirir. U-Net modeli, piksel düzeyinde segmentasyon kabiliyeti sayesinde göl yüzeyindeki değişiklikleri hassas bir

şekilde tahminleme için ideal bir araçtır.

Bu çalışmanın amacı, Burdur Gölü'nün 2015 ile 2023 yılları arasındaki yüzey suyu değişikliklerini CBS ve U-Net derin öğrenme modeli ile detaylı bir şekilde analiz etmek ve haritalamaktır. Analiz sonuçları, göl ekosisteminin durumu hakkında kritik bilgiler sağlamak, su kaynakları yönetimine stratejik bir perspektif sunmak ve gelecekteki su kaynaklarındaki değişiklikleri öngörmek adına değerli bir kaynak oluşturacaktır.

Bu çalışmada, Burdur Gölü'nün coğrafi özellikleri detaylı bir şekilde incelendi. Gölün bölgesel ekosistem içindeki rolü ve çevresel faktörlerle olan etkileşimleri üzerinde odaklanıldı. Bu inceleme, göl ekosisteminin genel yapısını ve dinamiklerini anlamak adına temel bir çerçeve oluşturdu.

Göletteki yüzey değişikliklerinin çeşitli etkenleri ve bu değişikliklerin göl ekosistemine olan etkileri ele alındı. Su seviyesindeki değişikliklerin nedenleri ve bu değişikliklerin göl ekosistemi üzerindeki potansiyel sonuçları, çalışmanın motivasyonunu oluşturan temel unsurlar arasında yer aldı.

CBS'nin temel prensipleri ve göl yüzey değişikliği analizindeki rolü açıklandı. CBS'nin, göl ekosisteminde meydana gelen değişiklikleri belirleme ve analiz etme süreçlerindeki önemi vurgulandı. Bu bağlamda CBS'nin sağladığı analitik yetenekler, çalışmanın metodolojisinin temelini oluşturdu.

CBS'nin kullanım avantajları ve sunduğu analitik yetenekler, çalışmanın başarılı bir şekilde gerçekleştirilmesi için kritik bir rol oynayacak olan teknolojik altyapının önemini vurguladı. CBS'nin sağladığı veri entegrasyonu ve mekânsal analiz yetenekleri, Burdur Gölü'nün yüzey değişiklikleri üzerindeki etkileşimleri daha iyi anlamamıza olanak sağlayacak. Derin öğrenmenin temel prensipleri ve uygulama alanları özetlendi. Çalışmanın odak noktası olan derin öğrenme modelinin tasarımı, prensipleri ve avantajları açıklandı. Derin öğrenme modelinin, uydu görüntüleri üzerinden yüzey suyu değişikliklerini hassas bir şekilde tespit edebilme yeteneği vurgulandı.

Bu projede, kullanılan derin öğrenme modelinin tasarımı, göl ekosistemine dair önemli bilgiler elde etmek ve gelecekteki değişikliklere karşı öngörülerde bulunmak adına kritik bir

araç olduğunu belirtti. Derin öğrenme modelinin, CBS ile birleştirilerek kullanılması, elde edilen sonuçların daha kapsamlı ve doğru olmasını sağladı.

2015 ve 2023 yıllarına ait göl yüzey değişikliği verilerinin toplanması, çalışmanın temel veri setini oluşturdu. Bu süreçte, uydu görüntülerinden elde edilen verilerin toplanması ve işlenmesi detaylı bir şekilde ele alındı.

Veri setinin hazırlanması ve işlenmesi aşamasında, CBS'nin ve derin öğrenme modelinin kullanımına uygun formata getirilmesi sağlandı. Bu adım, elde edilen verilerin analiz için uygun hale getirilmesini ve modellerle entegrasyonunu kapsayan kritik bir aşamayıdır.

Derin öğrenme modelinin eğitim süreci detaylı bir şekilde anlatıldı. Modelin performansının değerlendirilmesi, eğitim sürecinin etkinliğini belirleme adına önemli bir adımdır. Bu aşamada, modelin göl yüzey değişikliklerini ne kadar doğru ve hassas bir şekilde tespit ettiği detaylı bir analize tabi tutuldu.

CBS ile entegrasyonun sağlanması, elde edilen sonuçların daha geniş bir bağlamda değerlendirilmesini mümkün kıldı. CBS'nin sunduğu mekânsal analiz yetenekleri, derin öğrenme modelinin sağladığı detaylı sonuçlarla birleştirilerek daha kapsamlı bir veri analizi yapıldı.

Bölüm 2

2.Burdur Gölü ve Su Yüzeyi Değişikliği

2.1. Çalışma Alanı

Projenin çalışma alanı olan Burdur Gölü, Isparta ve Burdur illeri sınırlarında yer almakta olup 2012 yılı ölçümlerine göre 4965,40 hm³ su hacmine ve 122,53 km²lik bir alanı kaplamaktadır (Sabuncu, 2020). Coğrafi konumu, koordinatları 37° 36 "5.11" - 37° 57 "26.77" K ve 29° 41 "27.37" ' - 30° 25" 16.87 " D olarak belirlenen göl, 857,54 metreye kadar ulaşan maksimum su seviyesine sahiptir ve 1970 yılında bu seviyeye ulaşılmıştır. Gölün güncel su seviyesi ise 839,70 metredir. Yağışlar, göl yüzeyine ulaşan akarsular ve yer altı su kaynakları Burdur Gölü'nü besleyen temel su kaynakları arasında yer almaktadır. Göl, Burdur Kapalı Havzası'nda bulunmakta ve su kayıpları buharlaşma ve yer altı su kaçaklarından kaynaklanmaktadır. Göl seviyeleri incelendiğinde, yıllık bazda su seviyesinin aralık ayında arttığı, yaz aylarında ise azaldığı gözlemlenmektedir. 1980-1996 yılları arasında meydana gelen 7,70 metrelik azalmanın, yağış ve buharlaşma farkından 6,94 metre, göle ulaşan su miktarındaki azalmadan ise 0,83 metre kaynaklandığı tespit edilmiştir.

Burdur Gölü'nü içine alan Burdur Kapalı Havzasının genel görünümü ve gölün havza içindeki konumu, Şekil 2.1'de detaylı bir şekilde gösterilmiştir. Gölet, kuzeydoğu-güneybatı doğrultusunda geniş bir havza içerisinde konumlanmış olup, 1470 km²lik alanı kaplayarak bölgesel ekosistemde önemli bir rol oynamaktadır.

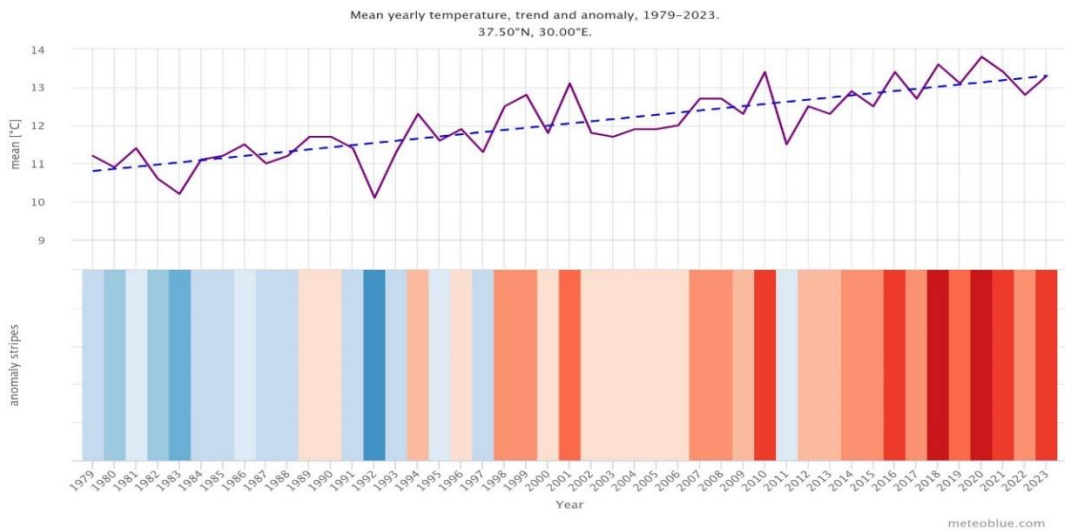
Burdur Gölü'nün coğrafi özellikleri, çeşitli etkenlere bağlı olarak yüzey değişikliklerine neden olabilir. Gölcük ve çevresindeki oluk ve vadilerin şekli, gölün yüzey değişikliklerine uyumlu bir şekilde evrim geçirmesine neden olmuştur. Bu nedenle, göldeki su seviyesindeki değişiklikler, havzanın topografik yapısıyla yakından ilişkilidir. Ayrıca, bölgedeki iklimsel faktörler de gölün su seviyesini etkileyebilir, bu da yüzey değişikliklerine katkıda bulunan diğer bir etkidir.

2.2. İklim Değişikliği ve Burdur

İklim değişikliğinin etkileri günümüzde net bir şekilde görülmektedir. Artan hava sıcaklıkları, eriyen buzullar, azalan kutup buzulları, yükselen deniz seviyeleri, çölleşme artışı ve sıcak hava dalgaları gibi olaylar, dünya genelinde iklim değişikliğinin belirgin belirtileridir. Bu etkiler, aşırı hava olaylarının daha sık ve şiddetli bir şekilde meydana gelmesine neden olmaktadır. Kuraklık, sel, fırtına gibi olaylar, iklim değişikliğinin doğrudan etkileri arasında yer almaktadır.

Özellikle belirtmek gerekir ki, iklim değişikliği küresel ölçekte homojen bir şekilde yayılmamaktadır. Bazı bölgeler, diğerlerine göre daha yoğun ve belirgin etkilenebilmektedir. Bu etkilerin yerel ve bölgesel özellikleri, iklim değişikliğiyle başa çıkmak adına alınacak önlemlerde dikkate alınması gereken önemli bir faktördür.

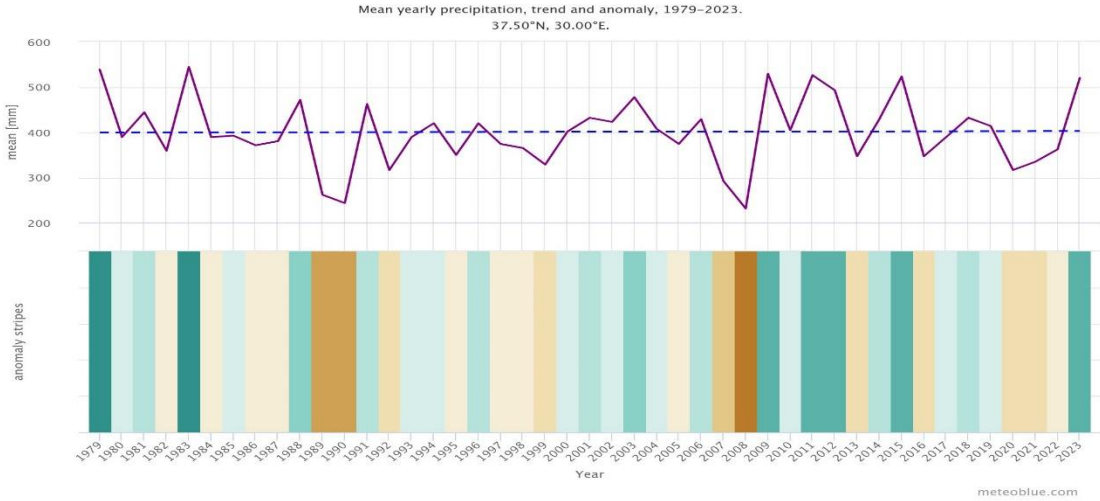
Burdur bölgesindeki iklim değişikliğinin son 40 yılda nasıl etkilendiğini gösteren aşağıdaki diyagramlar, bu bölgedeki belirli değişimleri ortaya koymaktadır. Kullanılan veri kaynağı, küresel iklimin beşinci nesil ECMWF atmosferik yeniden analizi olan ERA5'tir. Bu veri seti, 1979'dan 2023'e kadar olan zaman aralığını 30 km'lik uzaysal çözünürlükte kapsamaktadır. Bu analiz, bölgedeki sıcaklık, yağış ve diğer iklim değişkenlerindeki değişiklikleri daha ayrıntılı bir şekilde anlamamıza olanak tanır. Bu bilgiler, iklim değişikliğiyle mücadelede ve bölgesel etkilerle başa çıkmada rehberlik etmek adına önemlidir.



Şekil 2.1: 1979-2023 yılları aralığında Burdur'un yıllık ortalama sıcaklık değişimi (Meteoblue,2024)

Şekil 2.1'deki üst kısımdaki grafik, Burdur'un geniş bölgesi için ortalama yıllık sıcaklık tahminini sunmaktadır. Kesikli mavi çizgi, doğrusal iklim değişikliği eğilimini temsil etmektedir. Eğer trend çizgisi soldan sağa doğru yükseliyorsa, bu pozitif bir sıcaklık eğilimini gösterir ve Burdur'da iklim değişikliği nedeniyle hava ısınmaktadır. Eğer trend yataysa ve belirgin bir eğilim görülüyorsa veya aşağı doğru bir eğilim varsa, bu durum Burdur'da zamanla hava sıcaklığının azaldığını gösterir.

Grafik aynı zamanda alt kısımda "ısınma şeritleri" olarak adlandırılan şeritleri içermektedir. Her renkli şerit, bir yılın ortalama sıcaklığını temsil etmektedir; mavi renk daha soğuk yılları, kırmızı renk ise daha sıcak yılları gösterir. Bu şeritler, belirli yılların sıcaklık değişimlerini görselleştirmekte ve Burdur'daki ısınma veya soğuma eğilimlerini vurgulamaktadır. Bu analiz, bölgedeki iklim değişikliğinin zaman içindeki etkilerini anlamamıza yardımcı olmaktadır.



Şekil 2.2: 1979-2023 yılları aralığında Burdur'un yıllık ortalama yağış değişimi (Meteoblue,2024)

Şekil 2.2'deki üst kısımdaki grafik, Burdur'un daha geniş bölgesi için ortalama toplam yağış tahminini sunmaktadır. Kesikli mavi çizgi, doğrusal iklim değişikliği eğilimini temsil etmektedir. Eğer eğilim çizgisi soldan sağa doğru hareket ediyorsa, bu yağış eğiliminin pozitif olduğunu gösterir ve iklim değişikliği nedeniyle Burdur'da daha fazla yağış beklenebilir. Eğer eğilim yataysa ve belirgin bir eğilim görülüyorsa veya aşağı doğru bir eğilim varsa, bu durumda koşulların zaman içinde Burdur'da daha kurak hale geldiği anlamına gelir. Şekil 2.2'deki alt kısımdaki grafik, sözde yağış şeritlerini gösterir. Her renkli şerit, bir yılın toplam

yağışını temsil eder ve yeşil renk daha yağışlı yılları, kahverengi renk ise daha kuru yılları simgeler. Bu şeritler, belirli yılların yağış değişimlerini görselleştirmekte ve Burdur'da yağış koşullarındaki trendleri vurgulamaktadır.

Bu çalışmada, 2015-2023 yılları arasında elde edilen uydu görüntüleri kullanılmıştır. Şekil 2.3'te gösterilen görüntüler, söz konusu dönemde Burdur Gölü'nün su yüzeyinde önemli bir çekilme olduğunu göstermektedir. Bu çekilme, gölün su düzeyinde belirgin bir azalmayı ifade etmektedir. İklimsel verilere dayalı olarak, bu dönemde gözlemlenen su çekilmesinin, iklim değişikliği etkilerine bağlı olarak gerçekleşmiş olabileceği üzerinde durulmaktadır.

Şekil 2.3'teki görüntüler, belirli bir dönemdeki su yüzeyi değişikliklerini açıkça göstermektedir. Bu dönemdeki çekilme, göl ekosistemi üzerinde potansiyel etkiler yaratmış olabilir. Ancak, belirtilmelidir ki, iklim değişikliğinin bu çekilmeye doğrudan neden olduğunu kesin bir şekilde belirtmek zordur. Yine de, bahsedilen iklimsel verilerle uyumlu olarak göldeki su seviyesindeki değişim, iklim değişikliğinin bölgesel su kaynakları üzerindeki potansiyel etkilerini anlamak adına önemli bir gözlem sunmaktadır. Bu analiz, sadece görsel verilere dayanmakla kalmayıp aynı zamanda iklim verileriyle de desteklenerek, Burdur Gölü'nün su yüzeyindeki değişiklikleri daha kapsamlı bir bağlamda değerlendirmeyi amaçlamaktadır.



Şekil 2.3: 2015-2023 yılları aralığında Burdur Gölü uydu görüntüsü

Bölüm 3

3. Materyal ve Yöntem

3.1 Veri Toplama

Çalışma kapsamında kullanılan veriler, Copernicus Open Access Hub üzerinden elde edilen Sentinel-2 uydu görüntüleri üzerine odaklanmıştır. Veri toplama işlemi, 2015-2023 yılları arasındaki aylık dönemleri kapsayacak şekilde gerçekleştirilmiştir. Bu dönemdeki her bir ayın başından sonuna kadar olan süreçte, yüzde 10 ve daha az bulut içeren görüntüler tercih edilmiştir.

Veri seti, her biri 512 x 512 piksel çözünürlükte olan 380 adet True Color ve 380 adet NDWI (Normalized Difference Water Index) görüntüsünü içermektedir. True Color görüntüler, göl ve çevresindeki alanların doğal renklerini yansıtan ve genel arazi özelliklerini gözlemlemeye olanak tanıyan görüntülerdir. NDWI ise su varlığını tespit etmek amacıyla kullanılan bir su indeksidir.

Toplanan veri setleri, analizde kullanılacak uydu görüntülerini temsil etmektedir. Bu görüntüler, Burdur Gölü'nün su yüzeyindeki değişimleri belirlemek ve haritalamak amacıyla derin öğrenme modelleri ile analiz edilecektir. Elde edilen bu veri setleri, çalışmanın ana odak noktalarından biri olan göl ekosistemi ve su kaynakları üzerindeki değişiklikleri anlamak için temel veri kaynakları olarak kullanılmıştır.

3.1.1 Sentinel-2 Uydu Görüntüleri

Sentinel-2, Avrupa Uzay Ajansı (ESA) tarafından geliştirilen bir uzaktan algılama uydusu sistemidir ve dünyanın yüzeyini yüksek çözünürlüklü çok spektralli görüntülerle izleme kapasitesine sahiptir. Bu çalışmada, Sentinel-2 MSI uydusunun özellikle True Color (Gerçek Renk) özelliği ve NDWI (Normalized Difference Water Index) bantları üzerinde odaklanılmıştır. Sentinel-2, diğer sensörlerle karşılaştırıldığında termal bantlardan yoksun olmasına rağmen, daha iyi bulut taraması ve atmosferik etkilerin telafisi için kullanışlı olan spektral bantlara sahiptir. Bu bantlar arasında Bant 1, Bant 2, Bant 9, Bant 10 ve Bant 12 yer almaktadır. Bant 10, MODIS ve Landsat 8 OLI sensörlerinin tasarım mirasını temel alarak, sirus bulutlarının görülme sıklığını inceleme olanağı sağlar. Sentinel-2 görüntüleri, yüksek

çözünürlüklü veriler arasında yer almakta olup, VENµS misyonu gibi diğer uyduların verileri ile birlikte çeşitli gözlemleri desteklemektedir.

Bu görüntülerin kullanımı, bir bulut maskesi ile ilişkilendirilmiş olup, bu maskeler, bulutlu pikselleri tanımlayarak bunları bulutsuz olanlardan ayırır. Bu maskeler, hem yoğun bulutları hem de sirüs bulutlarını içerir ve ayrıca yoğun bulut ve sirüs piksellerinin yüzdesine ilişkin istatistiksel bilgileri içerir (Coluzzi et al., 2018).

Sentinel-2'nin True Color görüntüsü, kırmızı (Band 4), yeşil (Band 3) ve mavi (Band 2) bantların kombinasyonunu kullanır. Bu bantlar, güneş ışığında yeryüzünün doğal renklerini yansıtan ve insan gözü tarafından algılanan renklere en yakın olanları temsil eder. True Color görüntüleri, arazi özelliklerini daha gerçekçi bir şekilde görselleştirmek ve analiz etmek için yaygın olarak kullanılır (Custom Scripts Sentinel Hub, 2022).

NDWI (Normalized Difference Water Index), su varlığını belirlemek ve su bedenlerini haritalamak için geliştirilmiş bir indekstir. Bu indeks, Sentinel-2'nin 8. (Bant 8 - NIR) ve 12. (Bant 12 - SWIR) bantları arasındaki farkın, toplamlarına oranı alınarak hesaplanır. Denklem(1.1) şu şekildedir:

$$\begin{aligned} \text{Sentinel-2}_{\text{NIR}} &= \frac{(\text{Band } 8_{\text{NIR}} - \text{Band } 3_{\text{Green}})}{(\text{Band } 8_{\text{NIR}} + \text{Band } 3_{\text{Green}})} \\ \text{Sentinel-2}_{\text{SWIR}} &= \frac{(\text{Band } 11_{\text{SWIR}} - \text{Band } 3_{\text{Green}})}{(\text{Band } 11_{\text{SWIR}} + \text{Band } 3_{\text{Green}})} \end{aligned} \quad (1.1)$$

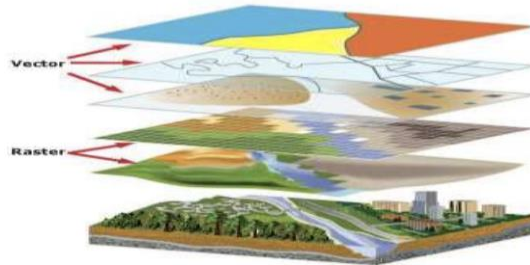
NDWI değerleri genellikle su yüzeylerinde yüksektir, bu da su varlığını belirlemek için kullanılabilir. Özellikle göller, nehirler, göletler ve diğer su kütlelerinin belirlenmesi ve izlenmesi için etkili bir araçtır. NDWI görüntüleri, Sentinel-2 için SWIR bantlarının kullanımıyla elde edilir. Bu bantlar, NIR ve yeşil bantlara kıyasla daha gelişmiş su sınıflandırması sağladığı için tercih edilir. Bu indeks, özellikle su ile ilgili çeşitli uygulamalarda yüksek çözünürlüklü ve güvenilir sonuçlar elde etmek amacıyla kullanılmaktadır (Li et al., 2013).

3.1.2 Coğrafi bilgi Sistemleri ve Uydu Görüntüleri ile Burdur Gölü Değişim Analizi

Coğrafi Bilgi Sistemleri (CBS), mekânsal veri yönetimi ve analizi konusunda kapsamlı bir bilgi sistemini ifade eder. CBS, dünya üzerindeki coğrafi verilerin toplanması, depolanması, kontrol edilmesi, analiz edilmesi ve amaca uygun şekilde kullanıcıya sunulmasını sağlayan bir bilgisayar sistemidir. Bu sistem, mekânsal ve mekânsal olmayan verileri depolayarak, analiz ederek ve görüntüleyerek bir bütünlük içinde çalışır. CBS'nin temel bileşenleri arasında konuma dayalı gözlemlerle elde edilen grafik ve grafik-olmayan bilgilerin işlenmesi yer alır.

Veri tabanı ve veri tabanı yönetim sistemi, CBS'nin temel özelliklerinden biridir. CBS, yeryüzündeki nokta, çizgi veya alan olarak gösterilen her türlü özelliği bilgisayar tabanlı bir veri tabanında depolar ve yönetir. Bilgisayar tabanlı olması, CBS veri tabanı ve yönetim sisteminin sunduğu çeşitli avantajlara erişim sağlar.

Coğrafi kodlama iki farklı metot ile gerçekleştirilmektedir. İstenilen özelliklerin türüne göre bu metotlardan biri veya her ikisi de kullanılmaktadır. Bu metotlar "vektör" ve "raster" modelleridir. Vektör modeli, coğrafi özellikleri birbirinden kesin bir şekilde ayıran durumlar için uygundur. Ancak, toprak tipleri, nüfus yoğunluğu gibi sürekli değişen coğrafi özellikler için yetersiz kalabilir. Bu durumda, "raster" modeli geliştirilmiştir. Raster modelinde gösterilecek özellikler, hücrelerin gruplanmasıyla yapılandırılır. Kullanılan ölçek, hücrelerin boyutlarını belirlerken, hücre boyutları çözünürlüğü ve dolayısıyla haritanın kalitesini belirler. Vektör modelinde ise temel yapı nokta olup, noktayı gerektiren özellikler tek bir nokta ile gösterilirken, çizgi ve alan gerektiren özellikler noktaların birbirine bağlanmasıyla ifade edilir (Kapluhan, 2014).



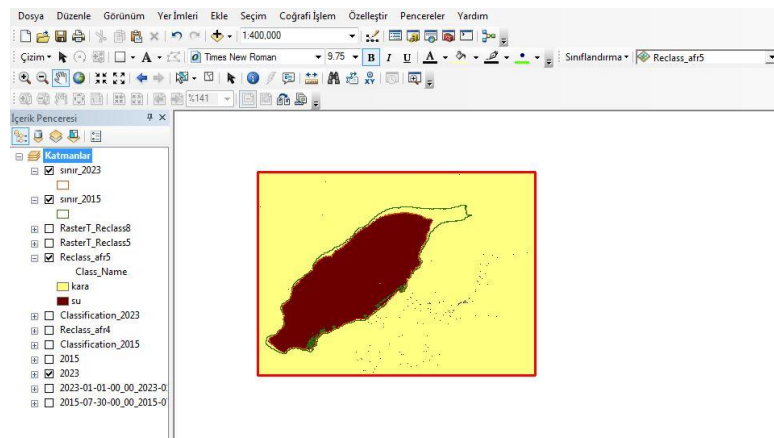
Şekil 3.1: Raster Modeli ve Vektör Modeliyle Gösterim

Uydu görüntüleri, CBS içinde önemli bir role sahiptir. CBS, uydu görüntüleri üzerinde değişiklikleri belirlemek, gözlemlemek ve analiz etmek için güçlü bir araç kümesi sunar. Uydu görüntüleri, coğrafi verilerin mekânsal özelliklerini daha iyi anlamak ve görselleştirmek için kullanılır. CBS, uydu görüntüleri üzerindeki değişiklikleri haritalandırarak, mekânsal verileri daha etkili bir şekilde yönetmeye ve analiz etmeye olanak tanır. Bu bağlamda, projenin kapsamında CBS'nin uydu görüntüleri üzerindeki yüzey değişikliklerini inceleyerek, bu değişiklikleri haritalandırma ve analiz etme süreçlerine nasıl entegre edildiği ayrıntılı bir şekilde ele alınacaktır.

Bu çalışmada, coğrafi analiz ve görüntü işleme süreçleri için ArcGIS programından yararlanılmıştır. ArcGIS, coğrafi bilgi sistemlerinin temel araçlarından biridir ve coğrafi verileri analiz etmek, düzenlemek ve görselleştirmek için oldukça güçlü bir araçtır.

Sentinel-2 uydu görüntüleri, ArcGIS programı kullanılarak işlenmiş ve değerlendirilmiştir. Özellikle, Burdur Gölü çevresindeki alanların kontrol sınıflandırması yapılarak su yüzeyi değişiklikleri belirlenmiştir. ArcGIS'in görüntü analizi araçları, yüksek çözünürlüklü uydu verilerini kullanarak doğru ve hassas sınıflandırmalar yapmayı mümkün kılmıştır. Bu süreçte, ArcGIS'in kullanıcı dostu arayüzü ve çeşitli modülleri, araştırmacılara verileri etkili bir şekilde görselleştirme, analiz etme ve sonuçları harita üzerinde görüntüleme imkânı sunmuştur. Ayrıca, programın sunduğu harita tasarım araçları, elde edilen sonuçları anlamak ve paylaşmak için etkili bir şekilde kullanılmıştır.

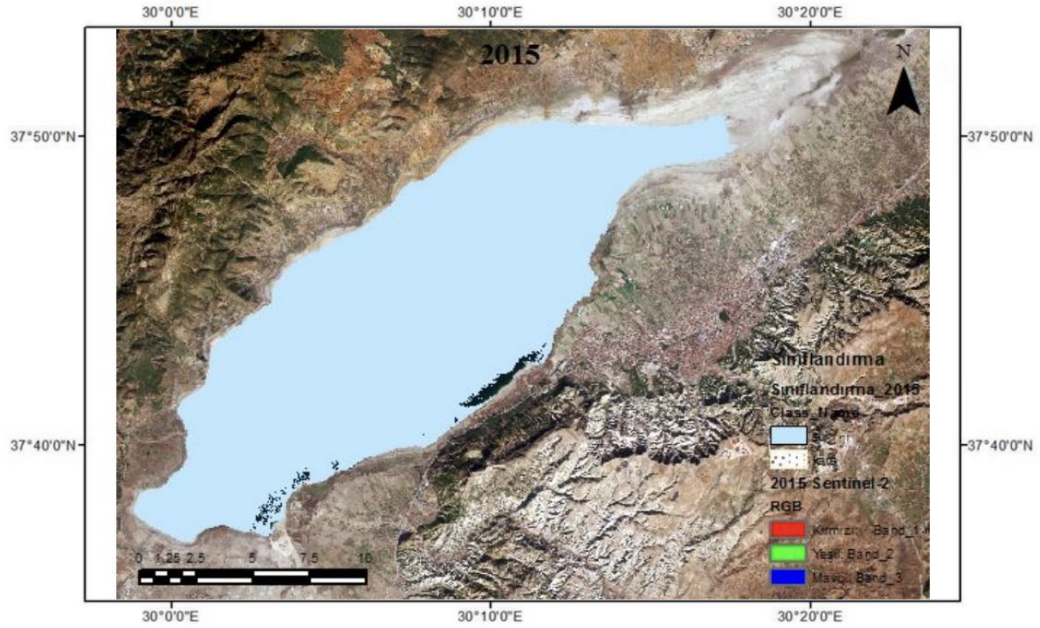
Aşağıda, ArcGIS programının ara yüzlerine dair ekran görüntüleri bulunmaktadır.



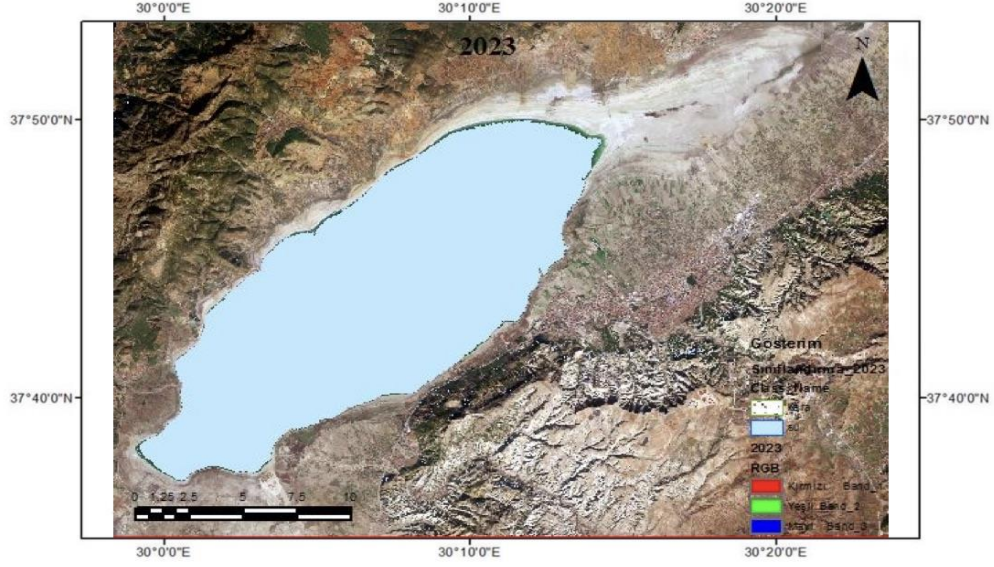
Şekil 3.2 :ArcGIS programı ara yüzü

Bu çalışma kapsamında, elde edilen uydu görüntülerinin dosyaları, ArcGIS programında kullanılmak üzere TIFF dosyası formatına entegre edilmiştir. Hem Sentinel-2 hem de diğer uydu görüntüsü, ArcGIS aracılığıyla kontrollü sınıflandırma yöntemi ile detaylı bir analiz sürecinden geçirilmiştir. Bu süreç, her iki görüntünün doğru bir şekilde sınıflandırılmasını ve coğrafi özelliklerine göre ayrıştırılmasını sağlamıştır.

Sınıflandırılmış görüntülerin sonuçları, analiz edilen bölgenin detaylı bir haritasını çıkarmak üzere kullanılmıştır. Bu haritalar, özellikle Burdur Gölü çevresindeki su yüzeyi değişikliklerini belirgin bir şekilde göstermektedir. Şekil 3.3 ve Şekil 3.4, sınıflandırma sonuçlarını açıkça ortaya koyan görsel temsiller sunmaktadır. Elde edilen sınıflandırma sonuçları, ArcGIS'in güçlü analitik araçları sayesinde detaylı coğrafi veri setlerine dönüştürülmüştür. Bu süreç, araştırmanın amacına uygun olarak coğrafi bilgileri etkili bir şekilde işlemeyi, yorumlamayı ve görselleştirmeyi mümkün kılmıştır. Sınıflandırılmış görüntüler, analiz edilen bölgedeki su yüzeyi değişikliklerini anlamak ve gelecekteki iklim değişikliği etkilerini değerlendirmek adına önemli bir kaynak oluşturmaktadır.



Şekil 3.3: 2015 yılı sınıflandırılmış uydu görüntüsü



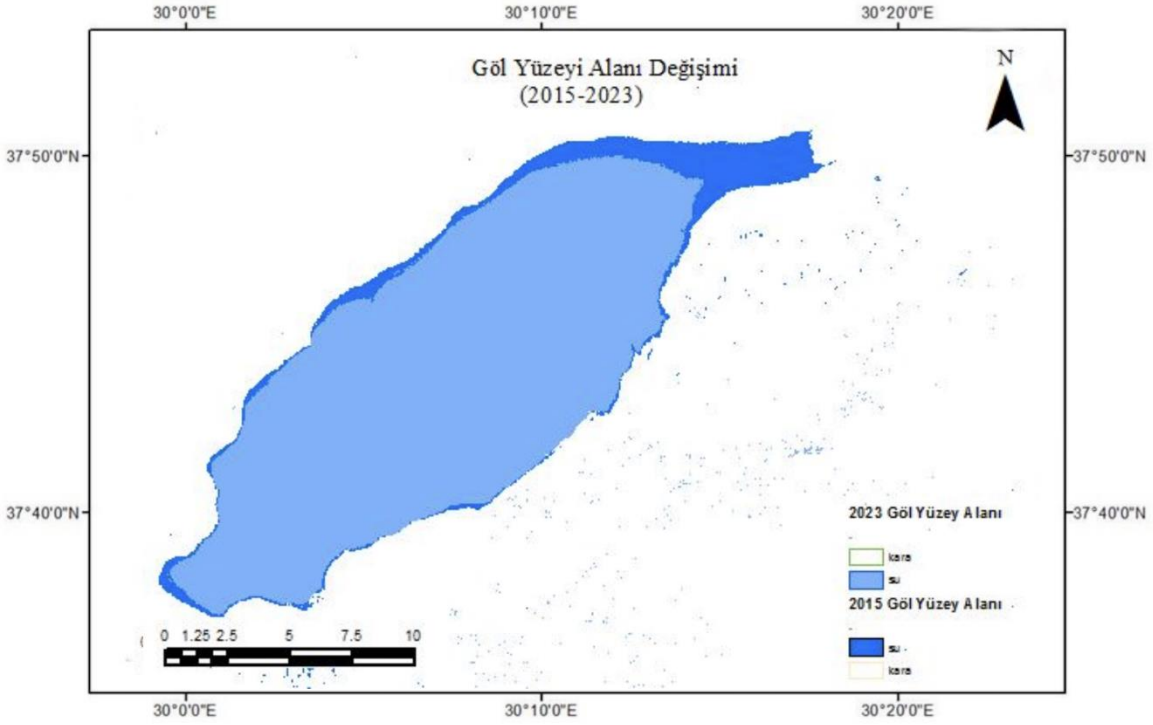
Şekil 3.4: 2023 sınıflandırılmış uydu görüntüsü

Araştırmanın sonuçlarına göre elde edilen Burdur Gölü yüzey alanı değişimine dair bulgular, detaylı bir şekilde Tablo 3.1 ve Şekil 3.5'te sunulmuştur. Tablo 3.1, analiz edilen dönem içinde Burdur Gölü'nün yüzey alanında meydana gelen değişiklikleri sayısal olarak ortaya koymaktadır. Şekil 3.5 ise bu değişimleri görsel olarak temsil etmektedir.

Tablo 3.1'te belirtilen veriler, Burdur Gölü'nün belirlenen zaman aralığında geçirdiği yüzey alanı değişimlerini ayrıntılı bir şekilde ifade etmektedir. Bu sayede, gölün su yüzeyindeki genişleme veya çekilmelerin miktarı ve zaman içindeki dağılımı değerlendirilebilmektedir. Elde edilen bu sayısal veriler, bölgedeki iklimsel ve çevresel değişimlerin göl üzerindeki etkilerini anlamak adına önemli bir kaynak sunmaktadır. Şekil 3.5, görsel bir temsil ile Burdur Gölü'nün yüzey alanındaki değişiklikleri açıkça göstermektedir. Bu grafik, belirlenen dönemdeki su seviyesindeki dalgalanmaları ve gölün genel topografik yapısındaki değişimleri anlamak için etkili bir araçtır.

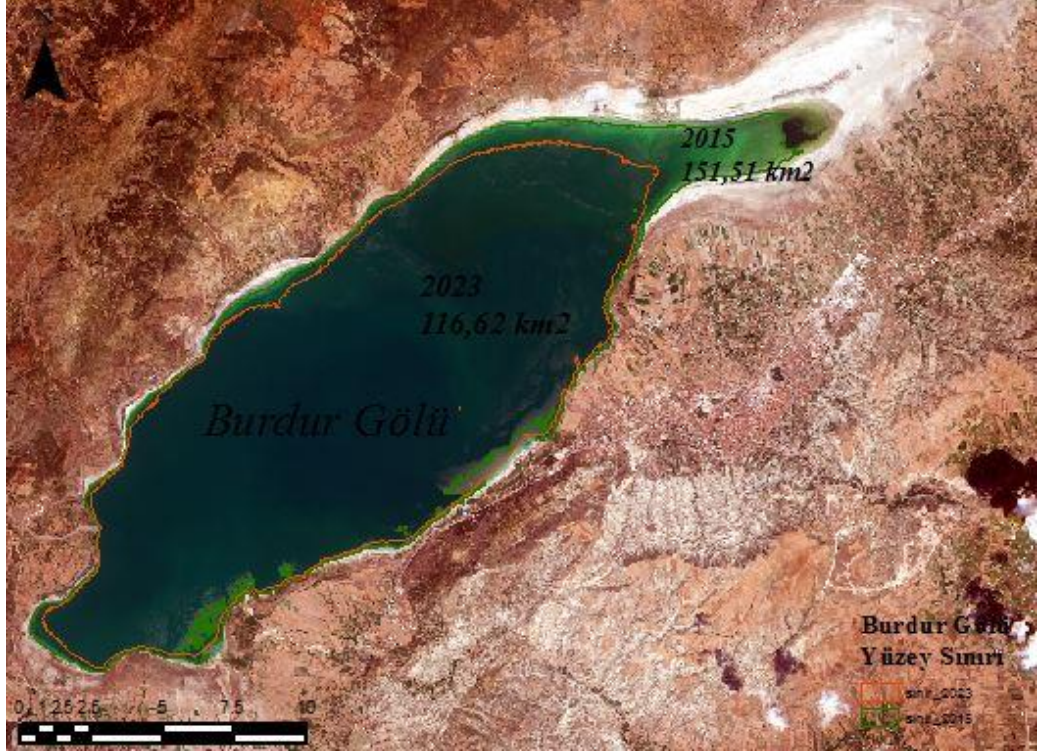
Tablo 3.1: 2015-2023 yılları aralığında Burdur Gölü Yüzey Alanı Değişimi

YILLAR	Yüzey Alanı(km ²)	Değişim (%)
2015	151,51 km ²	-30,26
2023	116,62 km ²	



Şekil 3.5: 2015-2023 yılları aralığında Burdur Gölü Yüzey Alanı Değişimi Haritası

Tablo 3.1'deki değerler, ArcGIS programında gerçekleştirilen sınıflandırma sonrasında elde edilmiş olup, bu değerler kesin sayısal ölçümler olmamakla birlikte ortalama değerlerdir. Bu ortalama değerler, 2015 yılına ait uydu görüntüsü için Şekil 3.6'da ve 2023 yılına ait uydu görüntüsü için Şekil 3.7'de görsel olarak ifade edilmiştir. Şekil 3.6 ve Şekil 3.7, ArcGIS programında elde edilen sınıflandırma sonuçlarını temsil eden grafiklerdir. Bu grafikler, belirlenen dönem içinde Burdur Gölü'nün yüzey alanındaki değişimleri detaylı bir şekilde göstermektedir. 2015 ve 2023 yıllarına ait uydu görüntülerinin sınıflandırma sonuçları, renk skalaları ve konturlar aracılığıyla sunulurken, gölün çeşitli bölgelerindeki su yüzeyi değişimlerini vurgular.



Şekil 3.6: 2015 yılı Burdur Gölü Sentinel-2 uydu görüntüsü haritası



Şekil 3.7 : 2023 yılı Burdur Gölü Sentinel-2 uydu görüntüsü haritası

BÖLÜM 4

4. Derin Öğrenme

Derin öğrenme, yapay zekâ geliştirme için en yaygın yöntemlerinden biridir, çünkü makinelerin dünyayı anlamalarını ve karmaşık sorunları çözmelerini sağlar. Bu öğrenme metodolojisi, birçok doğrusal olmayan işlem birimi katmanını kullanarak özellik çıkartma ve dönüştürme işlemlerini gerçekleştirir. Ardışık katmanlar, bir önceki katmanın çıkışı alır ve üzerinde işlemler uygular.

Derin öğrenme, geniş bir özellik yelpazesine sahip verilerle çalışır. Bu özellikler bazı durumlarda üst düzey, bazı durumlarda ise alt düzey özelliklerdir. Alt düzey özellikler, üst düzey özellikleri üretmek için kullanılır. Bu öğrenme yönteminin temelinde, verinin kendisinden en iyi şekilde öğrenme yeteneği bulunmaktadır. Derin öğrenme algoritmaları, elle çıkarılan özellikler yerine veriyi hiyerarşik bir şekilde temsil etmek için etkili algoritmalar kullanır.

Özetle, eski algoritmalar insan bağımlılığına sahipken, derin öğrenme algoritmaları ayırt edici özellikleri kendi başına öğrenebilme yeteneğine sahiptir. Derin öğrenme algoritmalarının diğer makine öğrenmesi algoritmalarından farkı, büyük veri setlerini ve bu verileri işleyebilecek yüksek performanslı donanıma ihtiyaç duymasındır. Yüksek performanslı donanımlar, derin öğrenme modellerinin etkili bir şekilde geliştirilmesine olanak tanır ve eğitim sürelerini kısaltmanın temel nedenidir (Kızrak ve Bolat, 2018).

Derin öğrenme uygulamalarında veri setinin büyüklüğü, öğrenme sürecinin önemli bir belirleyicisidir. Veri setinin büyüklüğü, öğrenmenin etkinliğini artırabilir, ancak aynı zamanda eğitim süresini uzatabilir.

4.1 Derin Öğrenme Modelleri ile Burdur Gölü Yüzey Değişikliği Analizi

Derin öğrenme, makine öğrenmesinin bir alt dalı olarak öne çıkan ve karmaşık veri setlerinde derin katmanlar içeren sinir ağlarının kullanıldığı bir yaklaşımdır. Yüzey suyu değişikliği analizinde, derin öğrenme modelleri genellikle uydu görüntüleri, özellikle de çok

spektralli görüntülerle çalışarak su kaynaklarının izlenmesi ve değişikliklerinin belirlenmesi süreçlerine katkı sağlar. Bu modeller, görsel özellikleri öğrenerek ve zaman içindeki değişiklikleri algılayarak yüzey suyu dinamiklerini anlamak için kullanılır.

Özellikle, Sentinel-2 uydu görüntüleri gibi çok spektralli veri setleri, derin öğrenme modelleri tarafından daha etkili bir şekilde işlenebilir. Uygulandıkları alanlardan biri de Burdur Gölü gibi su gövdelerinin yüzey değişikliklerinin analizidir. Bu modeller, gölün su seviyesindeki değişimleri belirlemede, göl çevresindeki arazideki değişiklikleri izlemede ve su kalitesi parametrelerini değerlendirmede kullanılabilir.

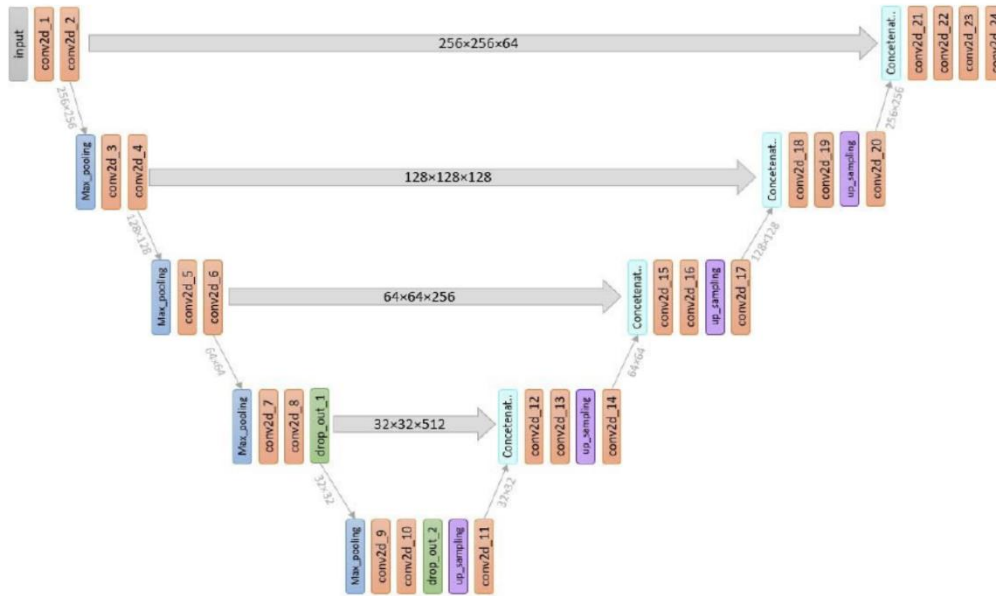
Bu araştırmada, Burdur Gölü'nün yüzey değişikliklerini analiz etmek üzere U-Net ve Autoencoder gibi derin öğrenme modelleri kullanılmıştır. U-Net, özellikle medikal görüntü analizi için geliştirilen ve yüksek performanslı segmentasyon görevlerinde kullanılan bir derin öğrenme modelidir. Diğer taraftan, Autoencoder ise veri boyutunu azaltma ve çıkarma yetenekleri ile bilinen bir unsupervised öğrenme modelidir. Bu iki modelin birleşimi, Burdur Gölü'nün yüzey değişikliklerini daha hassas bir şekilde tespit etmek ve bu değişiklikleri belirgin bir şekilde ortaya çıkarmak için güçlü bir analitik araç seti sunar. Kullanılan veri seti, toplamda 380 adet uydu görüntüsü ve bunlarla eşleşen 380 adet maskeli içermektedir. Uydu görüntüleri, Sentinel-2 tarafından sağlanan True Color uydu görüntüleridir ve yüksek çözünürlükte (512x512 piksel) seçilmiştir. Ayrıca, maskeler, aynı uydu görüntülerinin Normalized Difference Water Index (NDWI - Normalize Edilmiş Su İndeksi) kullanılarak elde edilmiştir. Bu veri seti, derin öğrenme modelinin eğitimi ve sonuçların çıkarılması için temel oluşturmaktadır. True Color uydu görüntüleri, doğal renklerle yüzey özelliklerini temsil ederken, NDWI maskeleri su varlığını belirleme amacıyla modelde kullanılmıştır. Bu sayede, model, Sentinel-2 uydu görüntüleri üzerindeki su yüzeyi değişikliklerini hassas bir şekilde tespit etme yeteneği kazanmıştır. Çözünürlük ve bulutluluk oranı seçimleri, modelin net ve güvenilir sonuçlar üretebilmesi için özenle belirlenmiştir.

4.1.1 U-Net Mimarisi

Çalışmada, U-Net mimarisi tercih edilmiştir ki bu mimari, Ronneberger ve diğerleri tarafından geliştirilen ve tekrarlayan evrişimli bloklardan oluşan bir sinir ağı mimarisidir (Atlan et al., 2020). U-Net mimarisi, özellikle segmentasyon, sınıflandırma ve görüntü üzerinde çalışma amacı güden bir görüntü segmentasyon algoritması olarak geliştirilmiştir (Yan et al., 2021). Şekil 4.1'de, bu çalışmadaki katman adımları detaylı olarak sunulmuş ve

modelin 'U' harfi şeklinde bir yapıya sahip olduğu açıkça görülmektedir. Kod oluşturucu kısımda, eğitim verilerinden özelliklerin çıkartılması sağlanır ve giriş görüntülerinin soyut bir temsilini öğrenir. Her kod oluşturucu (encoder) kısımda, ReLU aktivasyon fonksiyonu tarafından takip edilen 3x3 evrişim ağından oluşmaktadır. ReLU fonksiyonu, eğitim verilerinin geliştirilmesine yardımcı olurken aynı zamanda ağı doğrusal olmamasına olanak tanır. Bu işlemlerden sonra, özellik haritalarının boyutlarını yarıya indiren 2x2 maksimum havuzlama kısmı gelir.

Kod çözümleyici blok kısmı, 2x2 evrişim ağı ile başlar ve kodlayıcı bloğun atlamalı bağlantı özellik haritasıyla birleştirilir. Her adımda, evrişimin ReLU aktivasyon kodu tarafından takip edildiği 3x3 evrişim ağı kullanılır (Baldi, 2011). Kod çözücü kısmında ise eğitilen verilere ait maskeler oluşturulur. Bu kısım, 2x2 devrik evrişim ağı ile başlar ve sonrasında kod oluşturucu bloktaki özellik haritasıyla birleştirilir. Daha sonra, her evrişimin, ReLU aktivasyon fonksiyonu tarafından takip edildiği iki 3x3 evrişim kullanılır. Son kod çözücünün çıkışı, sigmoid aktivasyonu ile 1x1 evrişimden geçer. Sigmoid aktivasyon işlevi, piksel bazında sınıflandırmayı temsil eden segmentasyon maskesini verir. Oluşturulan model, Şekil 4.1'de görsel olarak sunulmuştur.



Şekil 4.1: U-Net modelinin katmanlar

Performans değerlendirmesi için sınıflandırma sistemi kullanılmıştır. Kullanılan sınıflandırma sistemlerinin güvenilirlik hesabı için Doğruluk (Accuracy), Kesinlik (Precision), Duyarlılık (Recall), F1-Skor formülleri kullanılmıştır. Ayrıca sınıflandırıcının tüm olası

değerler üzerindeki performansını özetlemek için ROC eğrisinden faydalanılmıştır. **Doğruluk (Accuracy):** Sınıflandırıcının ne sıklıkla doğru tahmin ettiğinin bir ölçüsüdür. Doğru tahmin edilen örneklerin toplam tahmin edilen örneklere bölünmesi ile elde edilir.

$$\text{Doğruluk} = (GP+GN) / (GP+GN+YP+YN) \quad (4.1)$$

Kesinlik (Precision): Tüm sınıflardan, doğru olarak ne kadar tahmin edildiğinin bir göstergesidir. Doğru tahmin edilen pozitif örneklerin toplam tahmin edilen pozitif örneklere bölünmesi ile elde edilir.

$$\text{Kesinlik} = GP / (GP + YP) \quad (4.2)$$

Duyarlılık (Recall): Pozitif olarak tahmin etmemiz gereken verilerin ne kadarının pozitif olduğunu tahmin ettiğimizizin göstergesidir. Doğru tahmin edilen pozitif örneklerin gerçek sınıftaki tüm örneklere bölünmesinden elde edilir.

$$\text{Duyarlılık} = GP / (GP + YN) \quad (4.3)$$

F1-Skor: Sınıflandırıcının ne kadar iyi performans gösterdiğinin bir ölçüsü olup sınıflandırıcıları karşılaştırmakta genellikle kullanılır. Bu, kesinlik ve duyarlılığın harmonik ortalamasıdır.

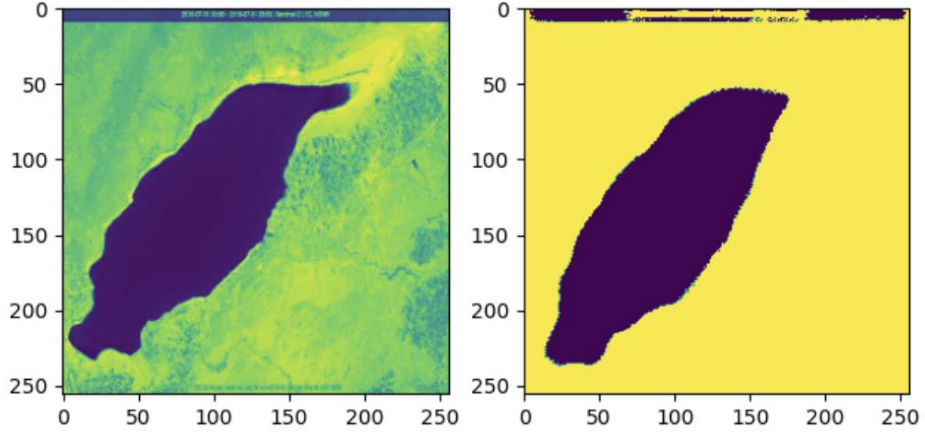
$$\text{F1-Skor} = 2 * (\text{Duyarlılık} * \text{Kesinlik}) / (\text{Duyarlılık} + \text{Kesinlik}) \quad (4.4)$$

4.1.1.2 U-Net Modeli Eğitimi ve Bulgular

Çalışmada, ilk aşamada, projede kullanılacak olan veri seti yüklenmiştir. Bu veri seti, Sentinel-2 uydu görüntüleri ve bu görüntülerle eşleşen maskelerden oluşmaktadır. Ardından, görüntüler ve maskeler uygun formata dönüştürülmüş, gri tonlamaya çevrilen ve normalize edilen veri seti, eğitim ve test verilerine ayrılmıştır.

U-Net modeli, projenin temelini oluşturmaktadır ve bu modelin yapılandırılması için encoder ve decoder blokları kullanılmıştır. Encoder blokları, görüntüyü temsil eden özellikleri çıkartmak için kullanılırken, decoder blokları, bu özellikleri orijinal boyuta getirme görevini üstlenir. Model, bu encoder ve decoder bloklarının birbirine bağlanmasıyla oluşturulmuştur.

Veri seti ve model oluřturma adımlarının ardından, model eđitilmiřtir. Eđitim s¼recinde belirlenen epoch sayısı ve erken durdurma kriteri ile modelin bařarıyla ¼đrenmesi sađlanmıřtır. Modelin eđitim s¼reci tamamlandıktan sonra, test veri seti ¼zerinde performans deđerlendirmesi yapılmıřtır. BinaryCrossentropy kaybı ve dođruluk metriđi kullanılarak elde edilen sonuřlar incelenmiřtir.

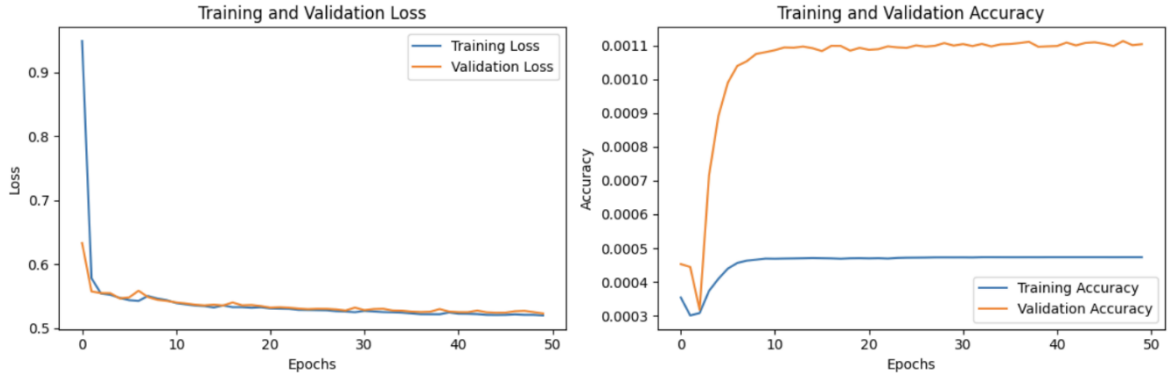


řekil 4.2 : U-Net modelinin sınıflandırma test sonucu g¼r¼nt¼s¼

Performans deđerlendirmesi, sınıflandırma raporu ve Jaccard skoru gibi metrikleri iřermektedir. Modelin sınıflandırma yeteneđi, ¼zellikle su y¼zeyi deđiřikliklerini bařarıyla tespit etme ¼zerine odaklanmıřtır. Ayrıca, eđitim sırasındaki kayıp ve dođruluk deđerlerini iřeren grafikler, modelin ¼đrenme s¼recini g¼rselleřtirmek amacıyla oluřturulmuřtur.

Class	Precision	Recall	F1
0	0.93	0.88	0.90
1	0.97	0.98	0.97
Acuraccy			0.96
Jaccard skoru			0.97

Tablo 4: U-Net modelinin sınıflandırma raporu



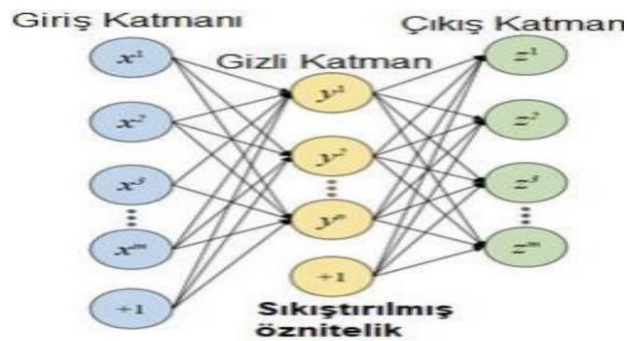
Şekil 4.3: U-Net modelinin eğitim sonucu kayıp ve doğruluk grafikleri

4.1.2 Otomatik Kodlayıcı (Autoencoder)

Makine öğrenme yöntemlerinde, genellikle veriler özniteliklerle temsil edilir. Bu, ham verilerin daha az kullanılarak daha etkili bir temsilin sağlanabilmesi açısından önemlidir. Otomatik kodlayıcılar, veriyi farklı nitelikte gösterimlerle temsil etmek amacıyla tasarlanmıştır. Bu yöntem, özellikle Hinton ve PDP grubu tarafından 1980'li yıllarda öne sürülen otomatik kodlayıcılarla başlamıştır. Derin öğrenmenin 2006 yılında gelişmesiyle birlikte, makine öğrenimi alanında önemli bir konu haline gelmiştir.

Derin Oto Kodlayıcı modeli, daha az öznitelik kullanarak daha güvenilir sonuçlar elde etmeyi amaçlayan bir derin öğrenme yöntemidir. Bu, veri setindeki öznitelik sayısını azaltarak hem doğruluk oranını hem de işleme hızını artırmayı hedefler (Baldi, 2011).

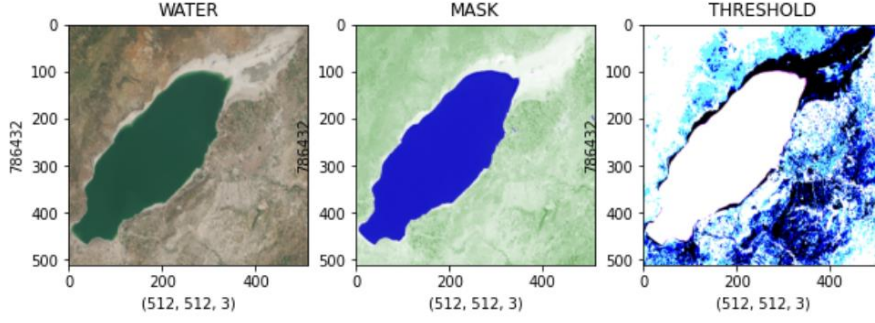
Oto Kodlayıcı, şifreleme ve şifre çözme adımlarından oluşan bir yapıya sahiptir. Şifreleme aşamasında, verilen girişi alır ve gizli katmanlarını kullanarak girişi, giriş birimlerinden daha küçük birimlerle ifade etmeye çalışır. Şifre çözme aşamasında ise, şifrelenmiş bilgiyi gizli katmanlarını kullanarak verilen girişi yeniden oluşturmaya çalışır (Bengio, 2009). Bu yöntem, verinin daha özgün ve etkili bir şekilde temsil edilmesini sağlar.



Şekil 4.4: Basit bir oto kodlayıcı ağ yapısı (Shin et al., 2013)

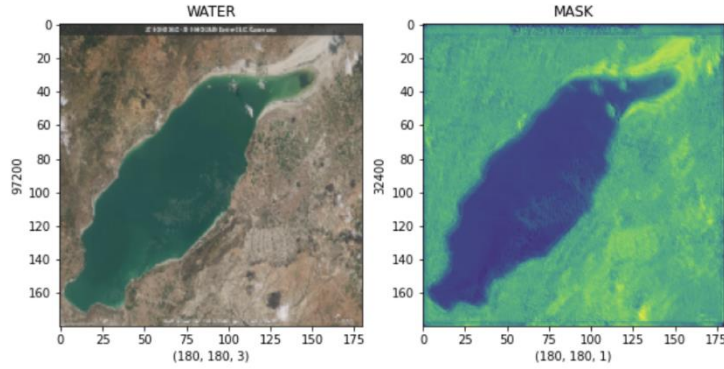
4.1.2.1 Otomatik Kodlayıcı (Autoencoder) Modeli Eğitimi ve Bulgular

Bu çalışmada, Burdur Gölü'ne ait görüntüler ve maskeler içeren bir veri kümesi üzerinde otomatik kodlayıcı (autoencoder) modeli geliştirilip eğitilmiştir. İlk aşamada, veri kümesinin özelliklerini anlamak amacıyla örnek görseller ve maskeler görselleştirilmiştir. Ayrıca, veri ön işleme adımları olarak eşikleme (thresholding) ve renk filtreleme gibi yöntemler uygulanmıştır.



Şekil 4.5: Veri ön işleme adımı

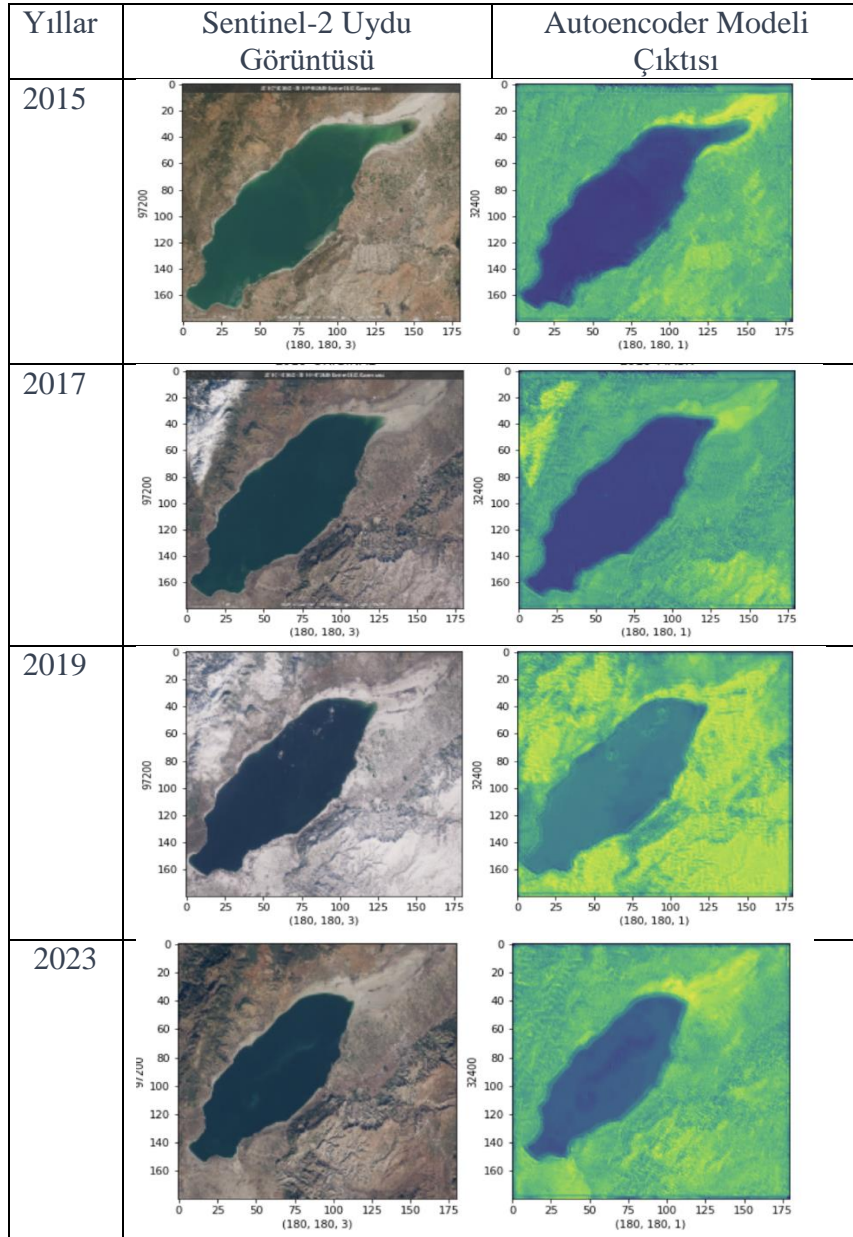
Oluşturulan otomatik kodlayıcı modeli, konvolüsyonel ve normalleştirme katmanları içeren bir kodlayıcı blok ile başlamakta ve ters konvolüsyon ve ReLU aktivasyon fonksiyonları içeren bir kod çözücü blok ile devam etmektedir. Model, su ve maske görüntülerini öğrenmek ve orijinal görselleri başarıyla yeniden oluşturmak üzere eğitilmiştir. Burada epok değeri 25 olarak alınmıştır. Eğitim sonrasında, modelin performansı örnek görsellerle değerlendirilmiş ve elde edilen sonuçlar incelenmiştir. Modelin, su ve maske ayrımını başarılı bir şekilde gerçekleştirdiği ve orijinal görselleri doğru bir şekilde rekonstrüe ettiği gözlemlenmiştir.



Şekil 4.6: Modelin image ve mask olarak ayrımı

Araştırmanın bir diğer aşamasında, model farklı yıllara ait örnek görseller üzerinde test edilmiş ve elde edilen sonuçlar görselleştirilmiştir. Modelin, farklı yıllardaki su alanlarını başarıyla tanıyabildiği ve maskeleri doğru bir şekilde oluşturabildiği gösterilmiştir.

Bu çalışma, otomatik kodlayıcıların göl suyu segmentasyonu gibi uygulamalarda etkili bir şekilde kullanılabileceğini ortaya koymaktadır. Elde edilen sonuçlar, modelin Burdur Gölü'ndeki su alanlarını başarılı bir şekilde tanıyabildiğini ve bu alanlara ait maskeleri doğru bir şekilde oluşturabildiğini göstermektedir.



Şekil 4.7 :Autoencoder modeli eğitim sonrası görselleştirmesi

BÖLÜM 5

5.Sonuçlar ve Tartışma

Bu araştırma, Burdur Gölü'nün 2015 ile 2023 yılları arasındaki yüzey suyu değişikliklerini incelemek amacıyla Coğrafi Bilgi Sistemleri (CBS) ve derin öğrenme modellerini kullanarak detaylı bir analiz gerçekleştirmiştir. Göl ekosisteminin sürdürülebilirliği ve su kaynaklarının yönetimi açısından kritik öneme sahip olan bu değişiklikler, iklim değişiklikleri, sulama ihtiyacındaki artış, endüstriyel faaliyetler ve su kaynaklarının kullanımındaki faktörlerden kaynaklanabilir.

CBS'nin su kaynakları yönetimindeki rolü vurgulanmış, göl yüzey değişikliklerini detaylı bir şekilde izleme ve analiz etme kapasitesinin önemi üzerinde durulmuştur. Bu bağlamda, Sentinel-2 uydu görüntüleri üzerinden elde edilen veriler, CBS üzerinde etkili bir analiz için temel oluşturmuştur.

İklim değişikliğinin bölgedeki etkilerini anlamak adına iklim verileri de kullanılarak, Burdur Gölü'nün su yüzeyindeki çekilmenin iklim değişikliği etkilerine bağlı olarak gerçekleşmiş olabileceği gözlemlenmiştir. İklimsel verilere dayalı analiz, göldeki su seviyesindeki değişimin iklim değişikliği ile uyumlu olduğuna işaret etmektedir.

Göl ekosistemi ve su kaynakları üzerindeki değişikliklerin anlaşılması için coğrafi özelliklerin detaylı bir şekilde incelendiği çalışmada, Burdur Gölü'nün bölgesel ekosistem içindeki rolü ve çevresel etkileşimleri odak noktası olmuştur. Su seviyesindeki değişikliklerin nedenleri ve potansiyel etkileri ele alınarak, CBS'nin analitik yetenekleri vurgulanmış ve CBS'nin sağladığı veri entegrasyonu ve mekânsal analiz yeteneklerinin önemi vurgulanmıştır. ArcGIS programı yardımı ile 2015 yılında Burdur Gölü'nün yüzey alanının 151,51 km² olduğu ve bu alanın 2023 yılında 116,62 km²'ye düşerek %30.26'lık bir azalmayı temsil ettiği belirlenmiştir. Bu önemli göl yüzeyi değişikliği, su kaynakları yönetimi ve ekosistem sürdürülebilirliği açısından dikkat çekici bir durumu ifade etmektedir.

Derin öğrenme modeli olan U-Net'in tasarımı, Sentinel-2 uydu görüntülerinden elde edilen 512x512 piksel çözünürlüklü 380 adet uydu görüntüsü ve 380 adet maske görüntüsü olmak üzere uydu görüntüsü verileriyle birleştirilerek, su yüzeyi değişikliklerini hassas bir şekilde

tespit etme yeteneđi güçlendirilmiştir. Elde edilen sonuçlar, 2015 ile 2023 yılları arasındaki göl yüzey değışikliklerini haritalandırmış ve bu değışikliklerin göl ekosistemi üzerindeki potansiyel etkilerini değerlendirmiştir.

Ayrıca, gerçekleştirilen sınıflandırma sonuçlarına göre, U-Net performansını değerlendirmek amacıyla belirli sınıflandırma metrikleri kullanılmıştır. Sınıf 0 için hassasiyet değeri %93, duyarlılık değeri %97 ve F1 skoru ise %88'dir. Sınıf 1 için ise hassasiyet değeri %98, duyarlılık değeri %90 ve F1 skoru ise %97'dir. Bu metrikler, ilgili sınıfların sınıflandırma modeli tarafından ne kadar doğru bir şekilde tahmin edildiđini göstermektedir.

Burdur Gölü'ndeki yüzey değışikliklerini ele alarak otomatik bir kodlayıcı modelin uygulanabilirliđini değerlendirmiştir. Modelin başarılı bir şekilde su alanlarını tanıma ve maskeleri oluşturma yeteneđi, göl ekosistemindeki değışiklikleri analiz etmek için önemli bir araç olabilir. Elde edilen sonuçlar, göl yüzeyindeki azalmaların ve artışların belirlenmesinde modelin etkili olduđunu göstermektedir.

Bu çalışma, su kaynakları yönetimine stratejik bir perspektif sunmak, gelecekteki su kaynaklarındaki değışiklikleri öngörmek adına değerli bir kaynak oluşturmaktadır. Derin öğrenme modelleri ve CBS'nin birleştirilerek kullanılması, elde edilen sonuçların daha kapsamlı ve doğru olmasını sağlamaktadır.

KAYNAKLAR

- [1] Arık, B.M. (2006) "Burdur Gölü", Türkiye'nin Önemli Doğa Alanları, Doğa Derneği, Ankara, 304-305
- [2] Sabuncu, A., 2020. "Burdur Gölü Kıyı Şeridindeki Değişiminin Uzaktan Algılama ile Haritalanması." Afyon Kocatepe University Journal of Sciences and Engineering. 20. Doi:10.35414/akufemubid.711653.
- [3] Meteoblue,
(https://www.meteoblue.com/tr/climate-change/burdur_t%c3%bcrtkiye_320392).
- [4] Coluzzi, R., Imbrenda, V., Lanfredi, M., Simoniello, T. (2018). "Bilinçli Yüzey Analizlerini Desteklemek İçin Sentinel-2 Seviye 1-C Bulut Maskesi Ürününün İlk Değerlendirmesi." Ortamın Uzaktan Algılanması, 217, 426-443. <https://doi.org/10.1016/j.rse.2018.08.009>.
- [5] Custom Scripts Sentinel Hub. (2022). Özel Sentinel-2 Bulut Maskesi. Erişim adresi: (https://custom-scripts.sentinel-hub.com/custom-scripts/sentinel-2/11c_optimized/)
- [6] Li, W., Du, Z., Ling, F., Zhou, D., Wang, H., Gui, Y., Sun, B., & Zhang, X. (2013). "A Comparison of Land Surface Water Mapping Using the Normalized Difference Water Index from TM, ETM+, and ALI." Remote Sensing, 5(11), 5530-5549. <https://doi.org/10.3390/rs5115530>.
- [7] Kapluhan, E. (2014). "Coğrafi Bilgi Sistemleri'nin (CBS) Coğrafya Öğretiminde Kullanımının Önemi ve Gerekliliği (The Importance and Necessity of the Use of Geographic Information System (GIS) In Geography Teaching)." Marmara Coğrafya Dergisi, Sayı: 29, 34-59. DOI: 10.14781/MCD.2014298120. İstanbul – ISSN:1303-2429, E-ISSN 2147-7825. <http://www.marmaracografya.com>.
- [8] Kızrak, M. ve Bolat, B., (2018). "Derin Öğrenme ile Kalabalık Analizi Üzerine Detaylı Bir Araştırma." Bilişim Teknolojileri Dergisi, c.11, no.11, s. 263-286.
- [9] Atlan, F., Hançer, E., & Pençe, İ. (2020). "U-Net ile Çekirdek Segmentasyonunda Hiper Parametre Optimizasyonu Etkisinin Değerlendirilmesi." Avrupa Bilim ve Teknoloji Dergisi, 60-69.
- [10] Yan, X., Tang, H., Sun, S., Ma, H., Kong, D., & Xie, X. (2021, October 25). "AFTER-UNet: Axial fusion transformer UNet for medical image segmentation." arXiv preprint arXiv:2110.10403. <https://doi.org/10.48550/arXiv.2110.10403>.

- [11] Baldi, P. (2011). "Autoencoders, unsupervised learning and deep architectures." In Proceedings of the 2011 International Conference on Unsupervised and Transfer Learning Workshop- Volume 27 (pp. 37–50).
- [12] Bengio, Y. (2009). "Learning Deep Architectures for AI." *Found. Trends Mach. Learn.*, 2(1), 1–127.
- [13] Shin, H.C.; Orton, M.R.; Collins, D.J.; Doran, S.J.; Leach, (2013). "Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data." *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 1930–43.

EKLER

EK 1:U-Net Modeli Kodları

```
[ ] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import model_selection, metrics
import tensorflow as tf

[ ] path = r"/kaggle/input/burdur/ Burdur_lake"

data = tf.keras.utils.image_dataset_from_directory(directory = path, image_size = (256, 256), batch_size = 6000, shuffle = False)

[ ] for images, masks in data:
    X = images.numpy().astype("uint8")
    y = masks.numpy().astype("uint8")

    print(X.shape, y.shape)

[ ] images = X[y == 0]
masks = X[y == 1]

print(images.shape, masks.shape)

[ ] for i in range(5):
    plt.subplot(1, 2, 1)
    plt.imshow(images[i])

    plt.subplot(1, 2, 2)
    plt.imshow(masks[i])

    plt.tight_layout()
    plt.show()

X = np.zeros(shape = images.shape[:-1] + (1, ))
y = np.zeros(shape = masks.shape[:-1] + (1, ))

for i in range(X.shape[0]):
    X[i] = tf.image.rgb_to_grayscale(images[i])
    y[i] = tf.image.rgb_to_grayscale(masks[i])

print(X.shape, y.shape)

[ ] X = X / 255.0
y = y / 255.0

[ ] X_train, X_test, y_train, y_test = model_selection.train_test_split(X, y, test_size = 0.2, random_state = 3)

print(X_train.shape, X_test.shape)

[ ] def encoder_block(inputs, num_filters):
    x = tf.keras.layers.Conv2D(num_filters, 3, padding = 'same', kernel_initializer = "he_uniform")(inputs)
    x = tf.keras.layers.Activation('elu')(x)
    x = tf.keras.layers.Conv2D(num_filters, 3, padding = 'same', kernel_initializer = "he_uniform")(x)
    x = tf.keras.layers.Activation('elu')(x)
    x = tf.keras.layers.MaxPool2D(pool_size = (2, 2), strides = 2)(x)
    return x

def decoder_block(inputs, skip_features, num_filters):
    x = tf.keras.layers.Conv2DTranspose(num_filters, (2, 2), strides = 2, padding = 'same')(inputs)
    skip_features = tf.image.resize(skip_features, size = (x.shape[1], x.shape[2]))
    x = tf.keras.layers.Concatenate()([x, skip_features])
    x = tf.keras.layers.Conv2D(num_filters, 3, padding = 'same', kernel_initializer = "he_uniform")(x)
    x = tf.keras.layers.Activation('elu')(x)
    x = tf.keras.layers.Conv2D(num_filters, 3, padding = 'same', kernel_initializer = "he_uniform")(x)
    x = tf.keras.layers.Activation('elu')(x)
    return x
```

```
[ ] def unet_model(input_shape = (128, 128, 1), num_classes = 1):
    inputs = tf.keras.layers.Input(shape = input_shape)

    s1 = encoder_block(inputs, 50)
    s2 = encoder_block(s1, 100)
    s3 = encoder_block(s2, 150)
    s4 = encoder_block(s3, 200)

    b1 = tf.keras.layers.Conv2D(filters = 100, kernel_size = (3, 3), padding = "same", kernel_initializer = "he_uniform")(s4)
    b1 = tf.keras.layers.Activation('elu')(b1)
    b1 = tf.keras.layers.Conv2D(filters = 100, kernel_size = (3, 3), padding = "same", kernel_initializer = "he_uniform")(b1)
    b1 = tf.keras.layers.Activation('elu')(b1)

    s5 = decoder_block(b1, s4, 200)
    s6 = decoder_block(s5, s3, 150)
    s7 = decoder_block(s6, s2, 100)
    s8 = decoder_block(s7, s1, 50)

    outputs = tf.keras.layers.Conv2D(filters = num_classes, kernel_size = (1, 1), padding = "same", activation = "sigmoid")(s8)

    model = tf.keras.models.Model(inputs = [inputs], outputs = [outputs], name = 'U-Net')
    return model

[ ] model = unet_model(input_shape = (256, 256, 1), num_classes = 1)

    model.summary()

[ ] model.compile(optimizer = "adam", loss = "BinaryCrossentropy", metrics = "accuracy")

    c1 = tf.keras.callbacks.EarlyStopping(patience = 5, restore_best_weights = True)

    history = model.fit(X_train, y_train, epochs = 50, validation_split = 0.2, callbacks = [c1])

[ ] model.evaluate(X_test, y_test)

[ ] y_test_pred = (model.predict(X_test) > 0.5).astype(int)

    def compare(i):
        plt.subplot(1, 2, 1)
        plt.imshow(y_test[i])

        plt.subplot(1, 2, 2)
        plt.imshow(y_test_pred[i])

        plt.tight_layout()
        plt.show()

[ ] clf_report = metrics.classification_report((y_test > 0.5).astype(int).flatten(), y_test_pred.flatten())

    print(clf_report)

[ ] model.save("model1.h5")

▶ import matplotlib.pyplot as plt

# Eğitim sırasında elde edilen metrik değerleri alın
loss_values = history.history['loss']
accuracy_values = history.history['accuracy']
val_loss_values = history.history['val_loss']
val_accuracy_values = history.history['val_accuracy']

# Loss grafiği
plt.figure(figsize=(12, 4))
plt.subplot(1, 2, 1)
plt.plot(loss_values, label='Training Loss')
plt.plot(val_loss_values, label='Validation Loss')
plt.title('Training and Validation Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
```

EK 2: Autoencoder Modeli Kodları

```
#GENERAL
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import random
#PATH PROCESS
import os
import os.path
from pathlib import Path
import glob
#IMAGE PROCESS
from PIL import Image
from keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import cv2
from keras.applications.vgg16 import preprocess_input, decode_predictions
from keras.preprocessing import image
from skimage.feature import hessian_matrix, hessian_matrix_eigvals
from scipy.ndimage.filters import convolve
from skimage import data, io, filters
import skimage
from skimage.morphology import convex_hull_image, erosion
#SCALER & TRANSFORMATION
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from keras.utils.np_utils import to_categorical
from sklearn.model_selection import train_test_split
from keras import regularizers
from sklearn.preprocessing import LabelEncoder
#ACCURACY CONTROL
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report, roc_auc_score, roc_curve
from sklearn.model_selection import GridSearchCV, cross_val_score
from sklearn.metrics import mean_squared_error, r2_score
#OPTIMIZER
from keras.optimizers import RMSprop, Adam, Optimizer, Optimizer, SGD
#MODEL LAYERS
from tensorflow.keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten, Conv2D, MaxPool2D, BatchNormalization, MaxPooling2D, BatchNormalization, \
    Permute, TimeDistributed, Bidirectional, GRU, SimpleRNN, LSTM, GlobalAveragePooling2D, SeparableConv2D, \
    ZeroPadding2D, Convolution2D, ZeroPadding2D, Conv2DTranspose, ReLU, UpSampling2D, Concatenate, Conv2DTranspose
from keras import models
from keras import layers
import tensorflow as tf
from keras.applications import VGG16, VGG19, inception_v3
from keras import backend as K
from keras.utils import plot_model
from keras.models import load_model
from keras import backend
#SKLEARN CLASSIFIER
from xgboost import XGBClassifier, XGBRegressor
from lightgbm import LGBMClassifier, LGBMRegressor
from catboost import CatBoostClassifier, CatBoostRegressor
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor
from sklearn.ensemble import GradientBoostingClassifier, GradientBoostingRegressor
from sklearn.ensemble import BaggingRegressor
from sklearn.tree import DecisionTreeClassifier, DecisionTreeRegressor
from sklearn.neural_network import MLPClassifier, MLPRegressor
from sklearn.neighbors import KNeighborsClassifier, KNeighborsRegressor
from sklearn.linear_model import LinearRegression
from sklearn.cross_decomposition import PLSRegression
from sklearn.linear_model import Ridge
from sklearn.linear_model import RidgeCV
from sklearn.linear_model import Lasso
from sklearn.linear_model import LassoCV
from sklearn.linear_model import ElasticNet
from sklearn.linear_model import ElasticNetCV
#IGNORING WARNINGS
from warnings import filterwarnings
filterwarnings("ignore", category=DeprecationWarning)
filterwarnings("ignore", category=FutureWarning)
filterwarnings("ignore", category=UserWarning)
```

```
[ ] Water_IMG_Path = Path("/kaggle/input/burdur/ Burdur_lake/Image2")
Water_Mask_Path = Path("/kaggle/input/burdur/ Burdur_lake/Mask2")

[ ] Water_JPG = list(Water_IMG_Path.glob(r"*.jpg"))
Mask_JPG = list(Water_Mask_Path.glob(r"*.jpg"))

[ ] Water_JPG_Series = pd.Series(Water_JPG,name="JPG").astype(str)
Mask_JPG_Series = pd.Series(Mask_JPG,name="MASK").astype(str)

[ ] figure,axis = plt.subplots(1,2,figsize=(10,10))

Example_Water = cv2.cvtColor(cv2.imread(Main_Water_Data["JPG"][4]),cv2.COLOR_BGR2RGB)
Example_Mask = cv2.cvtColor(cv2.imread(Main_Water_Data["MASK"][4]),cv2.COLOR_BGR2RGB)

axis[0].imshow(Example_Water)
axis[0].set_xlabel(Example_Water.shape)
axis[0].set_ylabel(Example_Water.size)
axis[0].set_title("WATER")
axis[1].imshow(Example_Mask)
axis[1].set_xlabel(Example_Mask.shape)
axis[1].set_ylabel(Example_Mask.size)
axis[1].set_title("MASK")

[ ] figure,axis = plt.subplots(1,2,figsize=(10,10))

Example_Water = cv2.cvtColor(cv2.imread(Main_Water_Data["JPG"][300]),cv2.COLOR_BGR2RGB)
Example_Mask = cv2.cvtColor(cv2.imread(Main_Water_Data["MASK"][300]),cv2.COLOR_BGR2RGB)

axis[0].imshow(Example_Water)
axis[0].set_xlabel(Example_Water.shape)
axis[0].set_ylabel(Example_Water.size)
axis[0].set_title("WATER")
axis[1].imshow(Example_Mask)
axis[1].set_xlabel(Example_Mask.shape)
axis[1].set_ylabel(Example_Mask.size)
axis[1].set_title("MASK")

[ ] figure,axis = plt.subplots(1,2,figsize=(10,10))

Example_Water = cv2.cvtColor(cv2.imread(Main_Water_Data["JPG"][237]),cv2.COLOR_BGR2RGB)
Example_Mask = cv2.cvtColor(cv2.imread(Main_Water_Data["MASK"][237]),cv2.COLOR_BGR2RGB)

axis[0].imshow(Example_Water)
axis[0].set_xlabel(Example_Water.shape)
axis[0].set_ylabel(Example_Water.size)
axis[0].set_title("WATER")
axis[1].imshow(Example_Mask)
axis[1].set_xlabel(Example_Mask.shape)
axis[1].set_ylabel(Example_Mask.size)
axis[1].set_title("MASK")

▶ figure,axis = plt.subplots(5,5,figsize=(10,10))

for indexing,operations in enumerate(axis.flat):
    Example_Water = cv2.cvtColor(cv2.imread(Main_Water_Data["JPG"][indexing]),cv2.COLOR_BGR2RGB)

    operations.set_ylabel(Example_Water.shape)
    operations.set_xlabel(round(np.mean(Example_Water)))
    operations.imshow(Example_Water)

plt.tight_layout()
plt.show()

[ ] figure,axis = plt.subplots(5,5,figsize=(10,10))

for indexing,operations in enumerate(axis.flat):
    Example_Water = cv2.cvtColor(cv2.imread(Main_Water_Data["MASK"][indexing]),cv2.COLOR_BGR2RGB)

    operations.set_ylabel(Example_Water.shape)
    operations.set_xlabel(round(np.mean(Example_Water)))
    operations.imshow(Example_Water)

plt.tight_layout()
plt.show()
```

```
[ ] figure,axis = plt.subplots(1,3,figsize=(10,10))

Example_Water = cv2.cvtColor(cv2.imread(Main_Water_Data["JPG"] [100]),cv2.COLOR_BGR2RGB)
Example_Mask = cv2.cvtColor(cv2.imread(Main_Water_Data["MASK"] [100]),cv2.COLOR_BGR2RGB)

_,Threshold_Water = cv2.threshold(Example_Water,150,255,cv2.THRESH_BINARY_INV)

axis[0].imshow(Example_Water)
axis[0].set_xlabel(Example_Water.shape)
axis[0].set_ylabel(Example_Water.size)
axis[0].set_title("WATER")
axis[1].imshow(Example_Mask)
axis[1].set_xlabel(Example_Mask.shape)
axis[1].set_ylabel(Example_Mask.size)
axis[1].set_title("MASK")
axis[2].imshow(Threshold_Water)
axis[2].set_xlabel(Threshold_Water.shape)
axis[2].set_ylabel(Threshold_Water.size)
axis[2].set_title("THRESHOLD")
```

```
[ ] Water_List = []
Mask_List = []

for water_i,mask_i in zip(Main_Water_Data.JPG,Main_Water_Data.MASK):

    Reading_IMG_Water = cv2.cvtColor(cv2.imread(water_i),cv2.COLOR_BGR2RGB)
    Reading_IMG_Water = cv2.resize(Reading_IMG_Water,(180,180))
    Reading_IMG_Water = Reading_IMG_Water / 255.

    Reading_IMG_Mask = cv2.cvtColor(cv2.imread(mask_i),cv2.COLOR_BGR2RGB)
    Reading_IMG_Mask = cv2.resize(Reading_IMG_Mask,(180,180))
    Reading_IMG_Mask = Reading_IMG_Mask / 255.

    Water_List.append(Reading_IMG_Water)
    Mask_List.append(Reading_IMG_Mask[:, :,0])
```

```
[ ] print("MASK IMG SHAPE: ",Mask_List[0].shape)
print("WATER IMG SHAPE: ",Water_List[0].shape)
```

```
[ ] print("MASK IMG TYPE: ",Mask_List[0].dtype)
print("WATER IMG TYPE: ",Water_List[0].dtype)
```

```
[ ] compile_loss = "binary_crossentropy"
compile_optimizer = "adam"
compile_metrics = ["accuracy"]
input_dim = (Water_Array.shape[1],Water_Array.shape[2],Water_Array.shape[3])
output_class = 1
```

```
[ ] Early_Stopper = tf.keras.callbacks.EarlyStopping(monitor="loss",patience=3,mode="min")
Checkpoint_Model = tf.keras.callbacks.ModelCheckpoint(monitor="val_accuracy",
                                                    save_best_only=True,
                                                    save_weights_only=True,
                                                    filepath="./modelcheck")
```

```
Encoder_G = Sequential()
Encoder_G.add(Conv2D(32,(2,2),kernel_initializer = 'he_normal'))
Encoder_G.add(BatchNormalization())
Encoder_G.add(ReLU())
#
Encoder_G.add(Conv2D(64,(2,2),kernel_initializer = 'he_normal'))
Encoder_G.add(BatchNormalization())
Encoder_G.add(ReLU())
#
Encoder_G.add(Conv2D(128,(2,2),kernel_initializer = 'he_normal'))
Encoder_G.add(BatchNormalization())
Encoder_G.add(ReLU())
```

```
[ ] Decoder_G = Sequential()
Decoder_G.add(Conv2DTranspose(64,(2,2)))
Decoder_G.add(ReLU())
#
Decoder_G.add(Conv2DTranspose(32,(2,2)))
Decoder_G.add(ReLU())
#
```



```
[ ] Auto_Encoder.compile(loss=compile_loss,optimizer=compile_optimizer,metrics=compile_metrics)
```

```
[ ] Auto_Encoder_Model = Auto_Encoder.fit(Water_Array,Mask_Array,epochs=25,callbacks=[Early_Stopper,Checkpoint_Model])
```

```
[ ] Prediction_IMG = Auto_Encoder.predict(Water_Array[:30])
```

```
[ ] figure,axis = plt.subplots(1,2,figsize=(10,10))
prediction_img_number = 1

Original_Img = Water_Array[prediction_img_number]
Predict_Mask = Prediction_IMG[prediction_img_number]

axis[0].imshow(Original_Img)
axis[0].set_xlabel(Original_Img.shape)
axis[0].set_ylabel(Original_Img.size)
axis[0].set_title("WATER")
axis[1].imshow(Predict_Mask)
axis[1].set_xlabel(Predict_Mask.shape)
axis[1].set_ylabel(Predict_Mask.size)
axis[1].set_title("MASK")
```

```
[ ] figure,axis = plt.subplots(5,6,figsize=(10,10))

for indexing,operations in enumerate(axis.flat):
    Predict_Mask = Prediction_IMG[indexing]

    operations.set_ylabel(Predict_Mask.shape)
    operations.set_xlabel(round(np.mean(Predict_Mask)))
    operations.imshow(Predict_Mask)

plt.tight_layout()
plt.show()
```

```
[ ] def prediction_process(image,model):
    backend.set_image_data_format('channels_last')

    Non_S_IMG = cv2.cvtColor(cv2.imread(image),
                             cv2.COLOR_BGR2RGB)

    Resize_IMG = cv2.resize(Non_S_IMG,(180,180))
    Resize_IMG = Resize_IMG/255.
    Resize_IMG_Prediction = Resize_IMG.reshape(-1,Resize_IMG.shape[0],Resize_IMG.shape[1],Resize_IMG.shape[2])
    Prediction_IMG_Another = model.predict(Resize_IMG_Prediction)
    Prediction_IMG_Another = Prediction_IMG_Another.reshape(Prediction_IMG_Another.shape[1],
                                                            Prediction_IMG_Another.shape[2],
                                                            Prediction_IMG_Another.shape[3])

    return Resize_IMG,Prediction_IMG_Another
```

```
[ ] Resize_Image,Prediction_N_S_Image = prediction_process("/kaggle/input/burdur/ Burdur_lake/Image2/2015_1.jpg",Auto_Encoder)

figure,axis = plt.subplots(1,2,figsize=(10,10))

axis[0].imshow(Resize_Image)
axis[0].set_xlabel(Resize_Image.shape)
axis[0].set_ylabel(Resize_Image.size)
axis[0].set_title("2015-ORIGINAL")
axis[1].imshow(Prediction_N_S_Image)
axis[1].set_xlabel(Prediction_N_S_Image.shape)
axis[1].set_ylabel(Prediction_N_S_Image.size)
axis[1].set_title("2015-MASK")

plt.savefig("2015.png")
```

```
[ ] Resize_Image,Prediction_N_S_Image = prediction_process("/kaggle/input/burdur/ Burdur_lake/Image2/2017_1.jpg",Auto_Encoder)

figure,axis = plt.subplots(1,2,figsize=(10,10))

axis[0].imshow(Resize_Image)
axis[0].set_xlabel(Resize_Image.shape)
axis[0].set_ylabel(Resize_Image.size)
axis[0].set_title("2017-ORIGINAL")
axis[1].imshow(Prediction_N_S_Image)
axis[1].set_xlabel(Prediction_N_S_Image.shape)
axis[1].set_ylabel(Prediction_N_S_Image.size)
axis[1].set_title("2017-MASK")

plt.savefig("2017.png")
```

```
[ ] Resize_Image,Prediction_N_S_Image = prediction_process("/kaggle/input/burdur/ Burdur_lake/Image2/2019_1.jpg",Auto_Encoder)

figure,axis = plt.subplots(1,2,figsize=(10,10))

axis[0].imshow(Resize_Image)
axis[0].set_xlabel(Resize_Image.shape)
axis[0].set_ylabel(Resize_Image.size)
axis[0].set_title("2019-ORIGINAL")
axis[1].imshow(Prediction_N_S_Image)
axis[1].set_xlabel(Prediction_N_S_Image.shape)
axis[1].set_ylabel(Prediction_N_S_Image.size)
axis[1].set_title("2019-MASK")

plt.savefig("2019.png")
```

```
▶ Resize_Image,Prediction_N_S_Image = prediction_process("/kaggle/input/burdur/ Burdur_lake/Image2/2023_1.jpg",Auto_Encoder)

figure,axis = plt.subplots(1,2,figsize=(10,10))

axis[0].imshow(Resize_Image)
axis[0].set_xlabel(Resize_Image.shape)
axis[0].set_ylabel(Resize_Image.size)
axis[0].set_title("2023-ORIGINAL")
axis[1].imshow(Prediction_N_S_Image)
axis[1].set_xlabel(Prediction_N_S_Image.shape)
axis[1].set_ylabel(Prediction_N_S_Image.size)
axis[1].set_title("2023-MASK")

plt.savefig("2023.png")
```