T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜLERDE DERİN ÖĞRENME TABANLI NESNE TESPİTİ İÇİN YENİ BİR ÖNİŞLEME YÖNTEMİ GELİŞTİRİLMESİ

DOKTORA TEZİ

Muhammed TELÇEKEN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

EKİM 2024

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜLERDE DERİN ÖĞRENME TABANLI NESNE TESPİTİ İÇİN YENİ BİR ÖNİŞLEME YÖNTEMİ GELİŞTİRİLMESİ

DOKTORA TEZİ

Muhammed TELÇEKEN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

Tez Danışmanı: Prof. Dr. Devrim AKGÜN

Ortak Danışmanı: Prof. Dr. Sezgin KAÇAR

EKİM 2024

Muhammed TELÇEKEN tarafından hazırlanan "Yüksek Çözünürlüklü Görüntülerde Derin Öğrenme Tabanlı Nesne Tespiti İçin Yeni Bir Önişleme Yöntemi Geliştirilmesi" adlı tez çalışması 30.10.2024 tarihinde aşağıdaki jüri tarafından oy birliği ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Bilim Dalı'nda Doktora tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı :

Jüri Üyesi :

Jüri Üyesi :

Jüri Üyesi :

Jüri Üyesi :

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜLERDE DERİN ÖĞRENME TABANLI NESNE TESPİTİ İÇİN YENİ BİR ÖNİŞLEME YÖNTEMİ GELİŞTİRİLMESİ" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(30/10/2024).

Muhammed TELÇEKEN

Eşime ve çocuklarıma…

TEŞEKKÜR

Bu doktora tezini tamamlarken bana destek olan herkese sonsuz teşekkürlerimi sunmak istiyorum.

Öncelikle, danışmanım Prof. Dr. Devrim AKGÜN'e en derin teşekkürlerimi sunarım. Yol gösterici, bilgi dolu ve sabırlı yaklaşımınız olmadan bu çalışmayı tamamlamak mümkün olmazdı. Bilimsel düşünceyi ve araştırma yapma becerisini kazandırdığınız için minnettarım.

Ortak danışmanım Prof. Dr. Sezgin KAÇAR'a da en içten teşekkürlerimi sunarım. Her zaman motive edici, destekleyici ve yol gösterici oldunuz. Sunduğunuz değerli fikirler ve katkılarla bu çalışmanın kalitesini artırdınız.

Anne ve babama, bana verdikleri sonsuz sevgi, sabır ve destek için teşekkür ederim. Her zaman yanımda oldunuz ve beni cesaretlendirdiniz. Sizin emeğiniz ve fedakarlıklarınız olmasaydı, bugün burada olamazdım.

Son olarak, eşime en derin şükranlarımı sunarım. Bu süreç boyunca gösterdiğin anlayış, sabır ve destek olmasaydı, bu çalışma asla tamamlanamazdı. Bana olan inancın ve verdiğin moral sayesinde zorlukların üstesinden gelebildim.

Muhammed TELÇEKEN

İÇİNDEKİLER

Sayfa

KISALTMALAR

AMFD : Kümülatif Çoklu Çerçeve Farkı
AP : Ortalama Kesinlik
ASAHI : Uyarlanabilir Dilimleme Destekli Hiper Çıkarım
ASPP : Gelişmiş Uzamsal Piramit Havuzlama
BIFPN : Çift Yönlü Özellik Piramit Ağı
CA : Koordinat Dikkati
CNN : Evrişimli Sinir Ağı
CSPNet : Aşamalar Arası Kısmi Ağ
DIP : Değiştirilebilir Görüntü İşleme
DT : Karar Ağaçları
DyNN : Dinamik Sinir Ağları
ELA : Verimli Katman Birleştirme
FLF : Odak Kaybı Fonksiyonu
GAN : Üretken Sinir Ağı
GPU : Grafik İşlemci Ünitesi
HOG : Yönlendirilmiş Gradyanların Histogramı
HR : Yüksek Çözünürlük
HRS : Yüksek Çözünürlüklü Uydu
IA-YOLO : Görüntü Uyarlamalı YOLO
ISA : Görüntü Dilimleme Algoritması
İHA : İnsansız Hava Araçları
LR : Düşük Çözünürlük
LRMC : Güçlü Matris Tamamlama
MF : Çoklu Füzyon
MIM : Maskeli Görüntü Modelleme
MIOU : Manhattan-Birlik Kavşağı Mesafesi
NMS : Maksimum Olmayan Bastırma
NMS : Maksimum Olmayan Bastırma
PAN : Kişisel Alan Ağı
PHA : Arka Hiyerarşik Hizalama

POE : Aşamalı Yönelim Tahmini

PSNR : Tepe Sinyal-Gürültü Oranı

R-CNN : Bölge Tabanlı Evrişimsel Sinir Ağı

RFSR : Rastgele Orman Süper Çözünürlük

SAHI : Dilimleme Destekli Hiper Çıkarım

SAM : Her şeyi Segmente Et

Self-RegConv: Kendinden Düzenleyici Konvolüsyon

SIFT : Ölçek-Değişmez Özellik Dönüşümü

SR : Süper Çözünürlük

SRGAN : Süper Çözünürlüklü Üretken Sinir Ağı

SROD : Süper Çözünürlüklü Nesne Tespiti

SS : Seçici Arama

SSD : Tek Atışlı Çoklu Kutu Dedektörü

SSIM : Yapısal Benzerlik Endeksi Ölçümü

ST : Kaydırmalı Dönüştürücü

SVM : Destek Vektör Makinaları

TIOE-DET : Tasarım Arama ve Yönlendirme Tahmin Dedektörü

ULSAM : Ultra Hafif Alt Uzay Dikkat Modülü

VDSR : Çok Derin Süper Çözünürlük

VIT : Vizyon Transformatör

YOLO : Sadece Bir Kez Bakarsın

YOLT : Sadece İki Kere Bakarsın

TABLO LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

YÜKSEK ÇÖZÜNÜRLÜKLÜ GÖRÜNTÜLERDE DERİN ÖĞRENME TABANLI NESNE TESPİTİ İÇİN YENİ BİR ÖNİŞLEME YÖNTEMİ GELİŞTİRİLMESİ

ÖZET

Son zamanlarda yüksek çözünürlüklü uydu ve insansız hava aracı (İHA) görüntüleri, güvenlik, trafik analizi, afet yönetimi ve çevre izleme gibi alanlarda giderek popüler hale gelmektedir. Ancak, pratikte bu tür yüksek çözünürlüklü görüntülerde küçük nesnelerin tespitinde, düşük piksel yoğunluğu ve ayrıntıların kaybolması gibi problemlerden dolayı başarım düşmektedir. Bundan dolayı, küçük nesneleri daha doğru bir şekilde tespit edebilecek, yüksek doğruluk oranına sahip yeni yöntemlerin geliştirilmesine ihtiyaç duyulmaktadır.

Bu tez çalışmasında, yüksek çözünürlüklü görüntülerde nesne tespiti için yeni bir derin öğrenme tabanlı algoritma geliştirilmiştir. Tezin temel amacı, yüksek çözünürlüklü uydu ve insansız hava aracı (İHA) görüntülerinde küçük nesnelerin tespitinde karşılaşılan zorlukları aşmak ve nesne tespit performansını artırmaktır. Önerilen yöntem, yeni geliştirilmiş ISA (Image Slicing Algorithm) adı verilen ön işleme algoritması, güncel olarak kullanılan SAM (Segment Anything) ile etiket optimizasyonu, çözünürlüğü iyileştiren SRGAN (Super-Resolution Generative Adversarial Network) ve YOLO (You Only Look Once) modellerinin birleştirilmesi ile oluşturulan SROD (Super-Resolution Object Detection) Mimarisinden oluşmaktadır. Önerilen yöntem, özellikle küçük nesnelerin tespiti sırasında kaybolan piksel verilerinin korunmasını ve nesnelerin daha yüksek doğrulukla algılanmasını sağlamaktadır.

Tezde önerilen yöntem şu şekilde çalışmaktadır: ilk olarak, yüksek çözünürlüklü görüntülerdeki nesnelerin doğru bir şekilde dilimlenmesi için ISA algoritması kullanılmaktadır. Bu algoritma, dilimleme sırasında nesnelerin bütünlüğünü koruyarak, nesne tespiti için veri hazırlama sürecinde önemli bir rol oynamaktadır. Dilimleme işlemi sırasında oluşan etiketleme hatalarının kontrol edilmesi ve düzeltilmesi için SAM segmentasyon modeli kullanılmıştır. Bu sayede, dilimlenen görüntülerdeki nesnelerin etiketleme doğruluğu ve modelin performansı artırılmıştır.

Daha sonra, SRGAN modeli ile görüntülerin çözünürlüğü artırılarak, YOLO tabanlı nesne tespit modellerine çözünürlüğü iyileştirilmiş giriş verileri sağlanmıştır. SRGAN modeli, görüntülerdeki detay seviyesini artırarak, özellikle küçük nesnelerin daha iyi tespit edilmesini sağlamaktadır. Çalışmada hava görüntülerinde sıklıkla kullanılan xView ve VisDrone veri setleri kullanılmıştır. Bu veri setleri, çok sınıflı ve karmaşık görüntüler içermektedir. Bu özellikleri ile önerilen sistemin performansını test etmek için uygun bir problem sunmaktadır. Tezdeki deneysel çalışmalar YOLOv5, YOLOv7, YOLOv8 ve, YOLOv9 gibi farklı YOLO versiyonları kullanılarak yapılmıştır. Elde edilen deneysel sonuçlar, önerilen yöntemin nesne tespit doğruluğunu önemli ölçüde geliştirdiğini ve mevcut yöntemlere kıyasla daha yüksek doğruluk oranları sağladığını göstermektedir.

Tezde geliştirilen diğer önemli bir yöntem ise etiket dönüşüm algoritmasıdır. Bu algoritma, maske etiketli veri setlerinin YOLO formatına uygun hale getirilmesini sağlamaktadır. Etiket dönüşüm algoritması sayesinde, farklı veri setlerinde kullanılan etiketleme formatlarının uyumsuzluk sorunları çözülmüş ve YOLO modellerinde kolaylıkla kullanılabilir hale getirilmiştir.

Sonuç olarak, bu tez çalışması, yüksek çözünürlüklü görüntülerde nesne tespiti alanında önemli bir katkı sunmakta ve özellikle küçük nesnelerin tespitinde karşılaşılan zorlukları minimize eden yeni bir yöntem önermektedir. Bu yöntemin, güvenlik, trafik analizi, afet yönetimi ve çevre izleme gibi alanlarda kullanım potansiyeli bulunmaktadır. Geliştirilen yöntemlerin literatüre katkısı, nesne tespiti alanındaki performans iyileştirmeleri ve çoklu veri setlerinde elde edilen başarılı sonuçlar ile kanıtlanmıştır.

DEVELOPMENT OF A NEW PREPROCESSING METHOD FOR DEEP LEARNING BASED OBJECT DETECTION IN HIGH RESOLUTION IMAGES

SUMMARY

Recently, high-resolution satellite and Unmanned Aerial Vehicle (UAV) images have become increasingly popular in areas such as security, traffic analysis, disaster management and environmental monitoring. Such images enable the collection of critical information by scanning large areas in great detail, thus contributing to data-driven decision-making processes. However, in practice, the performance of detecting small objects in such high-resolution images is degraded due to problems such as low pixel density and loss of details. This poses a significant challenge, especially for security and monitoring applications where critical analyses are performed. Therefore, there is a need to develop new methods with high accuracy that can detect small objects more accurately.

In this thesis, the focus is on deep learning-based object detection in high-resolution images, with a particular emphasis on addressing the challenges encountered in detecting small objects in satellite and UAV imagery. A novel algorithm has been developed to optimize this detection process, offering new solutions to these challenges. The increasing use of high-resolution imagery, especially in areas such as urban planning, traffic monitoring, disaster management, and security, has heightened the importance of image-processing-based decision support systems. However, detecting small objects in such images poses significant challenges due to scale variations and large data sizes. The method proposed in this thesis presents a new approach to overcome these difficulties, making the object detection process both more efficient and accurate.

One of the key components of this study is a specialized slicing algorithm called the Image Slicing Algorithm (ISA). ISA works by slicing high-resolution images into more manageable sizes while minimizing data loss during object detection. Traditional methods, which resize images, often result in the loss of details for small objects, adversely affecting detection accuracy. By slicing the images, the ISA algorithm addresses these issues and preserves the integrity of objects during the process. As a result, both large and small objects can be detected more effectively. High-accuracy small object detection is crucial, especially in critical fields such as security and disaster management, and ISA was developed to achieve this level of precision.

The advantages of ISA extend beyond this. Another important function of the algorithm is its integration with the Segment Anything Model (SAM), which controls label accuracy during the slicing process. SAM checks the accuracy of object labels in sliced images using segmentation masks and automatically corrects any detected errors. By preserving label accuracy after slicing, SAM improves the quality of the training data, significantly reducing failure rates during the training of object detection models. Given the complexity of the images and the size of the objects, this automatic correction mechanism plays a crucial role in enhancing detection accuracy.

Another significant contribution of the thesis is the integration of the Super-Resolution Generative Adversarial Network (SRGAN) model, which enhances object resolution, with the You Only Look Once (YOLO) model used for object detection. The developed Super-Resolution Object Detection (SROD) ar combines SRGAN with YOLO to restore lost details in low-resolution images, enabling clearer detection of small objects. SRGAN enriches the data for object detection by enhancing the resolution of images through deep learning techniques, which is especially important for small object detection. YOLO, known for its speed and accuracy in object detection, when combined with SRGAN in the SROD algorithm, allows for more accurate and faster detection of objects in satellite and UAV imagery.

The datasets used in this thesis were carefully selected to evaluate the real-world applicability of the object detection algorithms. The xView dataset, composed of satellite images, covers different geographical regions and environmental conditions worldwide. This dataset stands out due to its wide variety of objects and presents complex images requiring multi-class detection. The VisDrone dataset consists of high-resolution images captured by UAVs under various weather conditions and environments. Both datasets provide challenging test environments for processing and evaluating high-resolution images. Experiments conducted on these datasets showed that the proposed SROD algorithm yielded significant performance improvements compared to existing methods. Specifically, experiments on the xView dataset using YOLOv5 achieved the highest accuracy rates, while YOLOv8 produced the best results on the VisDrone dataset.

Another important component of the thesis is the label conversion algorithm developed to transform label formats used in the datasets. This algorithm allows mask-based labels to be adapted to the YOLO format, facilitating the integration of various datasets with deep learning models. By ensuring compatibility between different dataset formats, the algorithm broadens the applicability of YOLO models. This conversion algorithm has great potential for use in various fields, including medical imaging and industrial applications. Developed to address the incompatibility issues of different labeling formats in datasets, this algorithm serves as a tool that extends the use of deep learning models.

The thesis does not only develop new algorithms but also tests their performance through comprehensive experimental studies. The results reveal significant advancements in the field of object detection. In particular, the resolution enhancement provided by SRGAN and the slicing process offered by ISA play a major role in improving the overall accuracy of the model. According to the results, the developed SROD algorithm achieved much higher success rates compared to existing methods. This success is not limited to satellite images but also extends to other high-resolution datasets, such as UAV imagery.

The performance evaluations of the algorithms developed in this thesis have been comprehensively analyzed through various experimental results. These experiments compare both existing methods and the proposed algorithms, clearly demonstrating the contributions of the developed approach to the field of object detection. The integrated use of the ISA, SAM, and SROD algorithms has provided a significant advantage, especially in detecting small objects. To evaluate the system performance, the complexity matrices were examined, which provide a comprehensive summary of the classification accuracy of the models. The confusion matrix is a very important tool for evaluating the performance of the classification models, as it provides a

detailed breakdown of the predicted and true class labels, highlighting the proportion of correct and incorrect predictions for each class. Therefore, the Average Precision (AP) for each class is examined, allowing the identification of classes that are more frequently misclassified or confused with others. This analysis provided valuable information about the model's strengths and areas for improvement.

Experimental studies show that the ISA algorithm substantially improves object detection during the slicing process in high-resolution images. Unlike traditional methods, where data loss occurs during resizing, the ISA algorithm preserves the integrity of objects, leading to more accurate detection results in satellite and UAV imagery. For instance, in experiments conducted on the xView dataset, the detection of small objects improved by 20% due to the slicing performed with ISA. This improvement is attributed to the preservation of details that are typically lost when high-resolution images are resized. SAM's label verification and correction processes also contributed to this performance increase, significantly reducing the rate of incorrectly labeled objects.

The SROD algorithm, developed in this thesis, achieved significant performance improvements in object detection thanks to the resolution enhancement capability of the SRGAN model. By increasing the resolution of small objects, SRGAN made their details more visible, allowing the YOLO model to detect objects more accurately. Experimental results indicate that the integration of SRGAN with YOLO produces better outcomes than using YOLO alone for object detection. In studies conducted on the VisDrone dataset, the SROD algorithm showed accuracy improvements particularly in detecting small objects such as vehicles and pedestrians. Tests with YOLOv5 and YOLOv8 models demonstrated a substantial performance increase due to SRGAN's detail-enhancing effect. In particular, the YOLOv8 model integrated with SRGAN produced higher mAP (mean Average Precision) results for small object detection on the VisDrone dataset.

The findings of this thesis have broad application potential, especially in fields such as security, urban planning, traffic monitoring, and disaster management. High-accuracy object detection is a factor that directly contributes to operational efficiency and decision-making processes. For instance, tasks such as monitoring traffic flow in a city or detecting accident or congestion points can be performed quickly and accurately with such advanced object detection systems. Similarly, post-disaster damage assessment can be made more efficient through rapid analysis of high-resolution satellite and UAV images. In this context, the methods developed in the thesis offer significant opportunities for real-world applications in various fields.

In conclusion, this thesis makes significant contributions to the field of object detection in high-resolution images and expands the use of deep learning-based methods in this area. The developed algorithms not only improve object detection performance but also offer substantial enhancements in data processing and label validation processes. The findings of this study set a new standard for object detection for high-resolution images, laying the foundation for future research and applications. Also, methods to reduce computational load without sacrificing detection accuracy, such as using model compression techniques or more efficient neural network architectures, offer potential for future work. Testing combinations of these approaches can yield a robust, resource-friendly model without compromising detection performance. The widespread application of these methods will strengthen their place in the literature and enable the development of more advanced technologies in this field.

1. GİRİŞ

Günümüzdeki teknolojik ilerlemeler, kamera ve gelişmiş bilgisayarların erişilebilirliğini artırmış, bu da her an görüntü veya video kaydetmenin olağan bir durum haline gelmesine olanak tanımıştır. Bu dijital dönüşüm, elde edilen görüntü verilerinin analizi açısından önemli bir gelişmeye yol açmıştır (Karaçeper, 2018). Görüntü işleme teknikleri, ham görüntü verilerinden anlamlı sonuçlar çıkarmak için kullanılmaktadır (Rani, 2017). Örneğin, akıllı telefon kameraları sayesinde yüz tanıma ve trafikteki araç plakalarını tanıma gibi günlük uygulamalar, görüntü işlemenin belirgin örneklerindendir (Gonzalez, 2009). Görüntü işleme uygulamaları içindeki önemli konulardan biri de nesne tespitidir. Nesne tespiti, bir görüntü dizisi veya videodaki belirli veya belirlenmemiş nesnelerin konum ve doğrultusu gibi bilgilerin elde edilmesini sağlar.

Bu alandaki yöntemler çeşitlilik göstermektedir. Basit uygulamalardan başlayarak derin öğrenme yöntemlerine kadar birçok yaklaşım bulunmaktadır. Mevcut videolarda nesnenin bulunduğu ortamın arka planının çıkartılmasıyla gerçekleştirilen nesne tespiti, günümüzde derin öğrenme yöntemleri ile gerçek zamanlı olarak başarıyla yapılabilmektedir (Mufti & Shah, 2021). Derin öğrenme, yapay sinir ağları temelli doğrusal olmayan dönüşümlerden oluşan sinir katmanlarını kullanır. Bu sebeple, üst düzey soyutlamaları modelleme esasına dayanan bir makine öğrenmesi tekniği olarak tanımlanır (Sarker, 2021). Derin öğrenme yöntemleri, artan işlem gücü ve gelişmiş grafik işlemcilerle birlikte büyük veri analizi, konuşma tanıma, görüntü sınıflandırması ve nesne tespiti gibi alanlarda başarıyla kullanılmaktadır. Derin öğrenme ile ilgili projelerde, bazı şirketler büyük miktarda veriyi günlük olarak toplamak ve analiz etmek suretiyle başarı elde etmişlerdir. Ancak eğitim süreçleri zaman alıcı olabilmektedir (Sarker, 2021).

Nesne tespiti, bilgisayarlı görmenin temel görevlerinden biri olarak öne çıkar ve belirli nesneleri algılamayı amaçlar. Bu görev, güvenlik, kalabalık analizi ve trafik analizi gibi birçok alanda başarıyla uygulanmaktadır (Dasiopoulou ve ark, 2005). Uydu teknolojisinin ilerlemesi, havadan elde edilen görüntülere yönelik nesne tespit

algoritmalarının geliştirilmesine olanak sağlamıştır (Pesaresi & Benediktsson, 2001). Uydu görüntülerinin pratik uygulamalarda kullanılması, doğal afet izleme, kentleşmenin haritalanması ve insan ile araç hareketlerinin izlenmesi gibi alanlarda önemli bir rol oynamaktadır (Pesaresi & Benediktsson, 2001). Ancak, uydu görüntülerindeki nesnelerin tespitiyle ilgili zorluklar bulunmaktadır. Özellikle nesne ölçeğindeki farklılıklar, küçük nesnelerin tespitini güçleştirebilmektedir. Uyduların farklı ölçeklerde görüntü yakalaması, nesne boyutundaki önemli farklılıklara neden olmaktadır (Mansour ve ark, 2019). Derin öğrenme algoritmalarının bu zorlukları aşma potansiyeli bulunsa da nesne ölçeğindeki farklılıklar, küçük nesnelerin tespitini zorlaştırabilmektedir. Ayrıca, uydu görüntülerindeki gürültü, nesnelerin benzersiz özelliklerinin çıkarılmasını engelleyebilir (Mansour ve ark, 2019).

Bundan dolayı, uydu görüntüleri gibi yüksek çözünürlüklü görüntülerde nesne tespitiyle ilgili zorlukların minimize edilmesi için çalışmalar devam etmektedir. Kameraların yaygın kullanımı ve uydu görüntülerinde nesne algılama konusundaki zorluklar, teknolojinin ilerlemesiyle birlikte karşılaştığımız karmaşık sorunlardan bazılarıdır. Bu alandaki ileri düzey görüntüleme sistemleri ve algoritmalarının geliştirilmesi, bilim ve teknoloji alanına önemli katkılar sağlayacaktır (Chen ve ark, 2023). Mevcut zorluklara önerilen çözümler ile, güvenlik, kentsel planlama ve doğal afet izleme gibi hayati konularda daha başarılı çalışmalar yapılabilecektir.

1.1. Tezin Hipotezi ve Amacı

Yüksek çözünürlüklü görüntülerde piksel kaybı olmaksızın dilimleme yapılarak daha yüksek doğrulukla küçük nesnelerin tespiti gerçekleştirilebilir. Bu hipoteze yönelik olarak görüntülerin dilimlenmesi için yeni bir algoritma tasarımı yapılması tezin temel amacını oluşturmaktadır.

Bu amaç doğrultusunda nesnelerin yüksek doğrulukla tespiti için YOLO (Sadece Bir Kez Bakarsın) (Redmon ve ark, 2016) algoritmaları kullanılmıştır. Tezin bir diğer hedefi, mevcut maske etiketli veri setlerinin, kullanılacak YOLO algoritmalarına uygun hale getirilmesidir. Bu amaçla, yeni bir etiket dönüşüm algoritması geliştirilmiştir.

1.2. Literatür Taraması

Günümüzde güvenlik, plaka tanıma ve yüz tanıma gibi birçok sistemde nesne tespiti çalışmaları yapılmaktadır. Donanım ve nesne tespit algoritmalarının gelişimi ile literatürde, birçok farklı çalışma yer almaktadır. Bu çalışmalar; klasik görüntülerin model girdi boyutuna göre yeniden boyutlandırılması ve dilimleme yöntemi önerilerek yapılan çalışmalar olarak ayrılmaktadır. Literatürde, yeniden boyutlandırma ve dilimleme yaklaşımı ile yapılan nesne tespit çalışmalarından bazıları Bölüm 1.2.1 ve 1.2.2'de incelenmiştir.

1.2.1. Yeniden boyutlandırma

Nesne tespiti için yapılan çalışmalarda modellerin giriş-çıkış boyutlarına göre üzerinde çalışılacak görüntüler model girişlerinde yeni boyutlandırılmaktadır. Bu yaklaşım ile yapılan çalışmalardan bazıları şunlarıdır;

Zhang ve ark. (2013) yüksek çözünürlüklü uzaktan algılama görüntülerinde karmaşık şekillere sahip nesneleri tespit etmek amacıyla, dönmeyle değişmeyen parçalara dayalı bir model önermişlerdir. Önerdikleri modelde, jeolojik uzaysal nesneleri ana parçalara ayırarak ve parçalar arasındaki yapı bilgisini kutupsal koordinatlarda tanımlayarak dönme değişmezliği elde etmeyi amaçlamışlardır. Yeni bir dönme değişmez özelliği için histogram yönelimli gradyanların genişletilmesiyle döndürülmüş parçaların ve nesnelerin özelliklerini çıkartmışlardır. Bu yöntemle, parçaların yerini belirlemek için bir kümeleme yöntemi kullanarak yeni bir tespit modeli oluşturmuş ve deneysel sonuçlarda, önerdikleri modelin sağlamlığını ve kesinliğini doğrulamışlardır.

Guo ve ark. (2018) Yüksek Çözünürlüklü Uydu (High Resolution Satellite (HRS)) görüntülerinde coğrafi nesne tespiti için birleşik çok ölçekli Evrişimli Sinir Ağı (Convolutional Neural Network (CNN)) önermişlerdir. Özellik haritaları üzerinden yüksek kaliteli nesne önerileri üreterek RSOD veri seti üzerinde inceleme yapmışlardır. (Etten, 2018) uydu görüntülerindeki küçük nesnelerin tespit performansını artırmak için rastgele boyutlardaki görüntüleri işleyen salt görünüm yaklaşımının Sadece İki Kere Bakarsın (You Only Look Twice (YOLT)) kullanılmasını önermişlerdir. Önerilen bu yöntem ile isteğe bağlı boyuttaki uydu görüntülerini hızlı bir şekilde değerlendirerek, çok sayıda sensör üzerinden az eğitim verisiyle farklı ölçeklerdeki nesneleri tespit edebildiği göstermiştir. Shermeyer ve Etten (2019) tasarladıkları Çok Derin Süper Çözünürlük (Very Deep Super Resolution

(VDSR)) ve Rastgele Orman Süper Çözünürlük (Random Forest Super-Resolution (RFSR)) çerçevelerini kullanarak süper çözünürlük tekniklerinin uydu görüntülerindeki nesne tespit algoritmaları üzerindeki etkisini incelemişlerdir. Zhu ve ark. (2020) Yüksek çözünürlüklü video karelerinde gerçek zamanlı ve yüksek doğruluklu hareketli nesne algılama konusuna odaklandıkları çalışmalarında, önceden geliştirilmiş bir çerçeveyi, yüksek çözünürlüklü görüntülerin gerçek zamanlı işlenmesini sağlamak için modifiye etmişlerdir. Yeniden boyutlandırılmış görüntüler üzerinde hareketli bölgeleri tespit etmek için bir yöntem kullanırken, derin sinir ağı olarak hafif bir omurga tercih edilmiş ve pozitif ile negatif örnekler arasındaki dengesizliği azaltmak için odak kaybı fonksiyonu (Focal Loss Function (FLF)) uygulamışlardır. Wu ve ark. (2021) yüksek çözünürlüklü uydu görüntüleri için önerdikleri, hiyerarşik nesne algılama çerçevesi, alt örneklenmiş uydu görüntülerinde havalimanları ve limanların tespiti ve tespit edilen nesnelerin orijinal yüksek çözünürlüklü görüntülere geri eşlenmesi olarak iki aşamadan oluşmaktadır. Her iki aşama için bağlamsal bilgi tabanlı derin özellik çıkarma ve özellikle küçük nesneler için eğimli bir sınırlayıcı kutu kullanımını içermektedir. Açık kaynak ve kendi kendine bir araya getirilen veri setleri üzerinde yaptıkları deneylerde, bağımsız nesne dedektörlerine göre daha iyi sonuçlar elde etmişlerdir. Yin ve ark. (2021) hareketli nesneleri tespit etmek ve izlemek için Jilin-1 uydu takımyıldızı tarafından toplanan verilere dayanan bir hareket modelleme gösterimi önermişlerdir. Tespit oranını iyileştirmek için kümülatif çoklu çerçeve farkı (Cumulative Multi-Frame Difference (AMFD)) ve yanlış sinyalleri temizlemek için güçlü matris tamamlama (Low-Rank Matrix Completion (LRMC)) kullanmışlardır. Yang ve ark. (2022b) yüksek çözünürlüklü görüntüleri veya özellik haritalarını kullanmanın getirdiği maliyetli hesaplamaları azaltmak amacıyla özellik piramidi tabanlı bir nesne dedektörü olan Basamaklı Seyrek Sorgu Tespitini (Cascaded Sparse Query Detection (QeryDet)) önermişlerdir. Önerdikleri yöntem, önce düşük çözünürlüklü özellikler üzerinde küçük nesnelerin konumlarını tahmin eetmektedir ve ardından bu konumları kullanarak seyrek olarak yönlendirilen yüksek çözünürlüklü özellikleri kullanarak doğru tespit sonuçlarını elde etmektedir. COCO veri setinde yaptıkları deneylerde, QueryDet'in algılama Ortalama Hassasiyet'ini (Mean Average Precision (mAP)) 1,0 ve mAP küçük değerini 2,0 artırdığını ve yüksek çözünürlüklü çıkarım hızını ortalama 3.0 katına çıkardığı gözlemlenmiştir. Liu ve ark. (2022) Görüntü iyileştirme ve nesne tespiti görevlerinin dengelenmesindeki zorlukları aşmak için yeni bir çerçeve olan Görüntü

Uyarlamalı YOLO (IA-YOLO) önermişlerdir. Önerdikleri bu çerçeve, her görüntünün daha iyi algılama performansı için uyarlanabilir bir şekilde geliştirilebildiği bir yapı sunar ve özellikle olumsuz hava koşullarını hesaba katmak üzere Değiştirilebilir Görüntü İşleme (Differentiable Image Processing (DIP)) modülü içermektedir. Uçtan uca ortaklaşa öğrenme sayesinde IA-YOLO, parametreleri küçük bir evrişimli sinir ağı tarafından tahmin edilen YOLO dedektörünü iyileştirmek üzere uygun bir DIP öğrenebilmektedir. IA-YOLO yöntemi ile hem sisli hem de düşük ışıklı senaryolarda daha etkili sonuçlar elde etmişlerdir. Wan ve ark. (2023) yüksek çözünürlüklü optik uzaktan algılama görüntülerinde nesne tespiti için özellik piramidi, çoklu algılama kafası stratejisi ve hibrit bir dikkat modülü kullanan YOLOv5 algoritması önermişlerdir. Ming ve ark. (2023) hava görüntülerinde yüksek kaliteli yönlendirilmiş nesne tespiti için Tasarım Arama ve Yönlendirme Tahmin Dedektörü (Task Interleaving and Orientation Estimation Detector (TIOE-Det)) önermişlerdir. TIOE-Det ile, algılama ardışık düzenini optimize etmek amacıyla Arka Hiyerarşik Hizalama (Posterior Hierarchical Alignment (PHA)) etiketini kullanarak sınıflandırma ve yerelleştirme alt görevleri arasındaki hizalamayı iyileştirmişlerdir. Ayrıca TIOE-Det ile dengeli bir hizalama kaybı geliştirilmiş ve nesnelerin yönelimini tahmin etmek için Aşamalı bir Yönelim Tahmini (Progressive Orientation Estimation (POE)) stratejisi benimsemişlerdir.

Tian ve ark. (2023) uzaktan algılama görüntülerindeki nesne tespit zorluklarını aşmak için YOLOv5 temel alınarak geliştirdikleri KCFS-YOLOv5'i önermişlerdir. KCFS-YOLOv5'te, uygun bağlantı kutularını elde etmek için K-means++ algoritmasını kullanarak başlangıçtaki kümeleme noktalarını optimize ederek ve geleneksel YOLOv5'e Koordinat Dikkati (CA) ve Çift Yönlü Özellik Piramit Ağı'nı (Bidirectional Feature Pyramid Network (BiFPN)) entegre etmişlerdir. Ayrıca, KCFS-YOLOv5'te küçük nesnelerin algılama hassasiyetini artırmak amacıyla yeni bir küçük nesne algılama kafası eklemiş ve SIoU Kaybı fonksiyonu kullanmışlardır. Akshatha ve ark. (2023) çok ölçekli özellik birleşimi ve daha büyük giriş boyutlarının kullanımıyla küçük nesnelerin tespit performansını iyileştirmişlerdir. Fang ve ark. (2023) nesne tespiti için Maskeli Görüntü Modelleme (Masked Image Modeling (MIM)) tabanlı ve önceden eğitilmiş Vizyon Transformatör'ü (ViT) uyarlamayı sunan MIMDet'i önermişlerdir. Bosquet ve ark. (2023) küçük nesnelerin tespitindeki ortalama kesinlik (mAP) değerini artırmak için yeni bir veri artırma yöntemi

önermişlerdir. Önerdikleri bu yöntemde, Üretken Çatışmacı Ağ (Generative Adversarial Network (GAN)) tabanlı nesne oluşturucu, nesne bölümleme, görüntü içi boyama ve görüntü harmanlama tekniklerini birleştiren bir işlem hattını içermektedir. İşlem hattının temel bileşeni, DS-GAN adlı yeni bir GAN tabanlı mimaridir. Önerdikleri bu yöntemin, popüler son nesne tespit algoritmalarının doğruluk değerini iyileştirdiğini, UAVDT ve iSAID veri setleri üzerinde yapmış oldukları deneylerde göstermişlerdir. Zhang J. ve ark. (2023) çok modlu verileri birleştiren ve yardımlı süper çözünürlük (SR) öğrenmeyi kullanarak Uzaktan Algılama Görüntüleri (RSI) için geliştirilmiş bir nesne algılama modeli olan SuperYOLO'yu önermişlerdir. SuperYOLO, simetrik kompakt Çoklu Füzyon (MF) kullanarak küçük nesne tespitini iyileştirir ve düşük çözünürlüklü (LR) girişle geniş arka planlardan ayırt edebilen yüksek çözünürlüklü (HR) özellik gösterimleri öğrenmek için basit ve esnek bir SR dalı içerir. VEDAI RS veri seti üzerinde yapmış oldukları deneysel çalışmalarda başarılı sonuçlar elde etmişlerdir.

1.2.2. Dilimleme yöntemleri

Nesne tespiti için yapıla çalışmalarda model giriş-çıkış boyutlarına bağlı olarak görüntülerin yeniden boyutlandırılmasından kaynaklı zorlukları aşmak için literatürde görüntülerin dilimlenmesi ile yapılan çeşitli çalışmalar mevcuttur. Bu çalışmalardan bazıları şunlardır;

Tresson ve ark. (2019) yüksek çözünürlüklü görüntülerde nesne tespiti yapmak ve gözlemleri iyileştirmek için bir işlem hattı oluşturmuşlardır. Yüksek çözünürlüklü görüntülerin yeniden boyutlanmasının önüne geçmek için görüntüleri dilimlemeyi önermişlerdir. Yang ve ark. (2021a) dilimleme ve birleştirme çerçevesi olarak adlandırılan bir görüntü tabanlı algılama ve bölümleme tekniği kullanarak zorlukları çözmeyi amaçlamışlardır. Geliştirdikleri iki farklı dedektör, Retinal-SliceNet ve iki aşamalı bir paradigma kullanarak, 3D CT taramalarındaki tehdit içermeyen çantaların analizi için başarılı bir performans elde etmişlerdir. Akyon ve ark. (2022) xViex ve Visdrone veri kümelerindeki küçük nesne algılama problemini çözmek için Dilimleme Destekli Hiper Çıkarım (Slicing Aided Hyper Inference (SAHI)) yaklaşımını önermişlerdir. Olamofe ve ark. (2022) nesnenin renk kanallarını değiştirerek, nesne görüntülerine saf gürültüler ekleyerek ve kontrast iyileştirme işlemleri uygulayarak veri artırma yaparak YOLOv3 modelinin xViex veri seti üzerindeki performansını değerlendirmişlerdir. Shen ve ark. (2022) büyük boyutlu hava görüntülerinde coğrafi

nesne algılamada doğruluk ve hız performansını iyileştirmek için, CNN ve YOLOv4 modelinin birleşimini kullanmışlardır. Doğruluk ve hızı dengeleyen çok hacimli YOLOv4 modeli ile büyük uzaktan algılama görüntülerini işlemek için dilimleme stratejisi ve Kesilmiş Maksimum Olmayan Bastırma (Non Max Suppression (NMS)) algoritması kullanan bir yöntem önermişlerdir. Önerilen yöntem ile DOTA ve DOTAv2 veri setleri üzerinde yüksek hız ve doğruluk elde etmişlerdir. Lin ve ark. (2022) küçük hedefli orman yangınlarını tespit etmek için geliştirilmiş bir model olan STPM_SAHI'yi tanıtmışlardır. Kaydırmalı Dönüştürücü (Swin Transformer (ST)) omurga ağını kullanarak, öz-dikkat mekanizması ile orman yangınlarının özelliklerini çıkartılar ve PAFPN özellik füzyon ağıyla daha geniş bilgi alanları elde etmişlerdir. SAHI yaklaşımının entegrasyonu ile, küçük hedefli orman yangınlarının tespit doğruluğunu önemli ölçüde arttırmışlardır.

Shen ve ark. (2023) uydu görüntülerinin dilenmesi, bulutlarla kaplı döşemelerin tespiti, MIOU tabanlı YOLO v7-Tiny yöntemi ile nesne tespiti ve kesik NMS yöntemi ile sonuçların filtrelenmesi adımlarını içeren bir yöntem önermişlerdir. DOTA-CD veri kümesi üzerinde başarılı bir şekilde deneysel çalışmalarını gerçekleştirmişlerdir. Oluşturdukları AIR-CD veri setini, yöntemin güvenilirliğini doğrulamak için kullanılmış ve önerdikleri yöntemin etkinliğini elde ettikleri başarılı sonuçlarla ortaya koymuşlardır. Pereira ve ark. (2023) fundus lezyonlarını tespit etmek amacıyla SAHI çerçevesini kullanarak YOLOR-CSP mimarisini birleştiren yeni bir yöntem önermişlerdir. Önerdikleri yöntem ile DDR veri seti üstünde yaptıkları çalışmada başarılı sonuçlar elde etmişlerdir. Wang ve ark. (2023) İHA hava fotoğrafçılığı nesne tespiti için geliştirilmiş bir YOLOX-X modeli olan YOLOX_w'u önerdiler. YOLOX_w, küçük nesneleri tespit etme performansını artırmak için SAHI yaklaşımı ve veri artırımı kullanılarak önceden işlenmiş eğitim verisi üzerinde eğitimlerini gerçekleştirmişlerdir. PAN ağı ve ultra hafif alt uzay dikkat modülü (ULSAM) gibi ek özelliklerle geliştirilen YOLOX_w ile VisDrone ve DIOR veri setleri üzerinde temel YOLOX-X'e göre daha yüksek tespit doğruluğu elde etmişlerdir.

Zhang ve ark. (2023) Uyarlanabilir Dilimleme Destekli Hiper Çıkarım (ASAHI) adlı yeni bir uyarlanabilir dilimleme yöntemini tanıtmışlardır. ASAHI, dilim sayısını görüntü çözünürlüğüne uyarlayarak gereksiz hesaplamayı azaltır ve Cluster-DIoU-NMS ile gelişmiş doğruluk ve çıkarım hızı sağlamaktadır. Deneysel çalışmalarında ASAHI'nin VisDrone ve xView veri setlerinde iyi bir performans sergilediğini ve

hesaplama süresinde bir azalma olduğunu gözlemlemişlerdir. Muzammul ve ark. (2024) insansız hava araçları kullanılarak yapılan görüntü analizlerini iyileştirmek amacıyla RT-DETR-X modelini, SAHI tekniği ile birleştirmişlerdir. Bu yaklaşımla, özellikle VisDrone-DET veri seti üzerinde yüksek hız ve hassasiyetle küçük nesnelerin tespit edilmesinde önemli ilerleme sağlamışlardır.

1.2.3. Tezin Literatüre Katkısı

Günümüzde nesne tespiti uygulamaları, yüksek çözünürlüklü görüntülerin kullanımıyla önemli başarılar elde etmektedir. Ancak, bu başarılar beraberinde işlem gücü ve donanım sınırlamalarıyla karşı karşıya kalmaktadır. Mevcut nesne tespiti modelleri, donanım gücüde düşünüldüğünde en fazla 1280 piksel gibi sınırlı bir girdi boyutunu kabul etmektedir. Bu durum, yüksek çözünürlüklü görüntülerin boyutlarını yeniden ölçeklendirmeyi gerektirmektedir. Yüksek çözünürlüklü görüntülerin boyutlarının yeniden ölçeklenmesi, çeşitli zorlukları beraberinde getirmektedir. Bu süreç, görüntülerin orijinal detaylarını kaybetme riski taşımakta ve nesne tespiti modelinin performansını olumsuz etkilemektedir. Özellikle karmaşık sahnelerde, küçük nesnelerin kaybolma olasılığı artar. Donanım ve işlem gücü sınırlamalarının etkisiyle, nesne tespiti modelleri daha düşük çözünürlüklü görüntülerle çalışmak zorunda kalmaktadır. Bu durum, özellikle uzaktan algılama, video gözetimi ve otonom araçlar gibi uygulamalarda önemli bir sorundur. Bu bağlamda, nesne tespiti algoritmalarının düşük çözünürlüklü girdiye uyum sağlama yeteneğini geliştirmek hem performansı artırmak hem de donanım sınırlamalarıyla daha etkili bir şekilde başa çıkmak için kritik bir öneme sahiptir. Özellik kaybını en aza indirme çözümleri, genellikle özel mimari tasarımları, transfer öğrenme yöntemleri veya veri artırma teknikleri gibi stratejileri içerir. Bu çözümler, nesnelerin özelliklerinin daha doğru bir şekilde çıkarılabilmesine ve modellerin nesne özelliklerini daha etkili bir şekilde öğrenmesine imkân tanır. Bu da nesne tespit uygulamalarının güvenilirliğini ve başarı oranını artırabilir.

Bu tezin litaratüre katkıları şunlardır:

i. Tez, genellikle görüntü ayrıntılarının kaybına yol açan ve özellikle karmaşık sahnelerdeki küçük nesneler için nesne algılama performansını olumsuz

etkileyen yüksek çözünürlüklü görüntülerin yeniden ölçeklenmesinin zorluğunu aşmak üstünedir.

ii. Yüksek çözünürlüklü görüntülerin, özellikle küçük nesneler için çözünürlüğü koruyarak ve piksel kaybını önleyerek modelin giriş boyutuna uyacak şekilde otomatik olarak dilimlendiği yeni bir ön işleme yaklaşımı önerilmektedir.

iii. Önerilen dilimleme yöntemi, dilimleme işlemi sırasında nesnelerin bütünlüğünü korumak ve görüntü kenarları boyunca nesnelerin parçalanmasını önlemek için kayan üst üste binme tekniği önerilmektedir.

iv. Parçalanmış nesne etiketlerini görüntü kenarlarında yönetmek ve dilimlenmiş görüntülerde doğru etiketlemeyi sağlamak için ön işleme adımına segmentasyona dayalı bir nesne etiketi kontrol yöntemi entegre edilmiştir.

v. Tez, nesne özelliklerinin öğrenilmesini geliştirmek ve modelin küçük ve ayrıntılı nesneleri algılama yeteneğini iyileştirmek için nesne algılama modelinin ilk katmanına SRGAN (Ledig ve ark, 2017) algoritmasını dahil etmektedir.

vi. Maske etiketli veri kümelerini otomatik olarak YOLO etiket formatına dönüştürmek için yeni bir algoritma önerilmiştir. Bu sayede önerilen ön işleme yaklaşımının mevcut maskeli veri kümeleriyle kullanılması sağlanmıştır.

vii. Etiket dönüşüm algoritmasının kolay uygulanması için bir kullanıcı arayüzü tasarlanmış, bu arayüz ile yöntemin bir uygulaması medikal alanda kullanılan bir veri seti üzerinde gösterilmiştir.

1.3. Tezin Organizasyonu

Bu tez çalışması, Giriş bölümünde problem tanımı ve literatür incelenmesi olarak başlamaktadır. 2. Bölüm, etiket yapıları, nesne tespiti ve algoritmalarının tanıtılması, deneysel çalışmaların gerçekleştirileceği veri setlerinin tanıtılması, çalışmaların gerçekleştirileceği çalışma ortamın tanıtımı ve incelenecek performans metrikleri olarak organize edilmiştir. 4. Bölüm, önerilen yöntemler sırası ile, maske etiketlerinin YOLO etiket formatına dönüştürülmesi için geliştirilen algoritma ve arayüzün tanıtımı ve önerilen dilimleme algoritması, Her Şeyi Bölütle Modelini (SAM) (Kirillov ve ark, 2023) ile etiket kontrol yapısı, SRGAN yapısı ve Süper Çözünürlüklü Nesne Tespit Mimarisinin tanıtımı yapılmaktadır. 4. Bölümde, belirlenen veri setleri üzerinde

önerilen yöntemlerin deneysel çalışmaları yer almaktadır. 5. Bölüm, sonuç ve değerlendirmelerin yapıldığı bölüm olarak organize edilmiştir.

2. MATERYAL VE METOT

2.1. Etiket Yapıları

Görüntü işleme ve bilgisayarlı görü alanlarındaki, özellikle nesne algılama ve bölütleme içeren görevlerde, "maske etiketi" terimi bir görüntüdeki bölgeleri belirlemek ve kategorilere ayırmak için kullanılan yapılandırılmış bir gösterimi ifade etmektedir. Bir maske etiketi, her bir öğenin orijinal görüntüdeki bir piksele karşılık geldiği tipik olarak ikili veya çok sınıflı bir dizidir. İkili maske etiketi, her bir öğenin 0 veya 1 olduğu bir dizidir. Genellikle ikili bölütleme görevlerinde ön plandaki nesneler ile arka plan arasında ayrım yapmak için kullanılmaktadır. Bu dizide, 1 değeri, pikselin ilgili nesnesine ait olduğunu gösterirken, 0 değeri arka planı belirtmektedir. İkili kavramını birden fazla sınıfa genişleten çok sınıflı bir maske etiketi, aynı görüntüdeki çeşitli nesnelerin veya bölgelerin bölütlenmesine olanak tanır. Bu, her pikselin önceden tanımlanmış birkaç kategoriden birine sınıflandırıldığı anlamsal bölütleme görevlerinde kullanılmaktadır (Long ve ark, 2015). Dizideki her pikselin ait olduğu sınıf etiketini temsil eden bir tam sayı değeri vardır. Örneğin, insanlar, arabalar ve ağaçlar içeren bir görüntüde, maske insanlar için 1, arabalar için 2 ve ağaçlar için 3 kullanabilir. Maske etiketi genellikle orijinal görüntüyle aynı mekânsal çözünürlüğe sahiptir ve maskedeki pikseller ile görüntü arasında birebir bir ilişki sağlar. Sınıf veya örnek sayısına bağlı olarak, maske farklı veri türleri kullanılarak temsil edilebilir. İkili maskeler genellikle Boole veya tam sayı türlerini kullanırken, çok sınıflı maskeler çok sayıda sınıf için işaretsiz tam sayılar veya daha karmaşık veri yapıları kullanabilmektedir. Maske etiketleri, manuel açıklama, yarı otomatik araçlar veya otomatik algoritmalar aracılığıyla üretilir. Gözetimli öğrenme çalışmalarında, açıklamalı maske etiketleri, bölütleme modellerini eğitmek ve değerlendirmek için temel gerçek veri görevi görür (He ve ark, 2017). Nesnelerin doğru bir şekilde konumlandırılması ve sınıflandırılmasının gerekli olduğu tıbbi görüntüleme, otonom sürüş ve artırılmış gerçeklik dahil olmak üzere çeşitli uygulamalarda kullanılmaktadır (Ronneberger ver ark, 2015). Maske etiket yapısı Şekil 2.1'de görülmektedir.

Şekil 2.1. Maske etiket yapısı.

Nesne tespitinde YOLO algoritmasına uygun etiket yapısı, bir görüntüdeki nesnelerin sınırlayıcı kutularını ve sınıf etiketlerini kodlamak için yaygın olarak kullanılan bir yöntemdir (Redmon & Farhadi, 2018). YOLO etiket yapısı, derin öğrenme modellerinde hızlı eğitimi ve çıkarımı kolaylaştıran kompakt ve etkili bir gösterim sağlamak üzere tasarlanmıştır. YOLO formatındaki etiket yapısı, her biri görüntüdeki algılanan bir nesneye karşılık gelen bir dizi açıklama içerir. Her açıklama, beş sayısal değer ve isteğe bağlı bir sınıf etiketi içeren tek bir metin satırıyla temsil edilir. Bu değerler, sınıf etiketini ve dört sınırlayıcı kutu koordinatını içerir. Sınıf etiketi, nesnenin ait olduğu sınıfı temsil eden bir tam sayıdır, genellikle önceden tanımlanmış bir sınıf listesine karşılık gelen bir dizindir. Sınırlayıcı kutu koordinatları, sınırlayıcı kutunun normalleştirilmiş koordinatlarını temsil eden dört kayan nokta değerinden oluşur. Bu koordinatlar, görüntünün boyutlarına göre belirlenmekte ve Denklem 2.1'de görüldüğü yapıda olmaktadır.

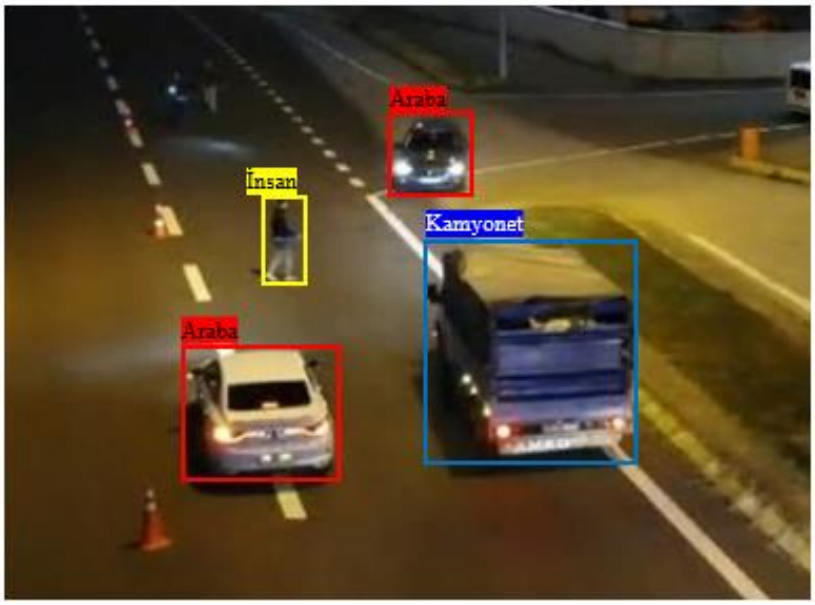
$$\mathbf{x_{merkez}}, \mathbf{y_{merkez}}, \mathbf{width}, \mathbf{height} \tag{2.1}$$

Özellikle x_merkez ve y_merkez sınırlayıcı kutunun merkezinin koordinatlarıdır ve sırasıyla görüntünün genişliği ve yüksekliğine göre normalize edilir ve değerleri 0 ile 1 arasında değişir. Sınırlayıcı kutunun genişliği ve yüksekliği de sırasıyla görüntünün genişliği ve yüksekliğine göre normalize edilir ve bu değerler de benzer şekilde 0 ile 1 arasında değişir. Örnek bir görüntüdeki nesnenin YOLO formatındaki etiketi Denklem 2.2'deki olmaktadır.

$$\mathbf{2\ 0,5\ 0,5\ 0,1\ 0,2} \tag{2.2}$$

Bu, görüntünün ortasında merkezlenmiş bir sınırlayıcı kutuya sahip sınıf 2 nesnesinin koordinatları (0,5, 0,5) olarak verilmiştir. Etiket boyutunun, görüntü boyutunun %10'u genişliğinde ve %20'si yüksekliğinde olacağı bilgisi bulunmaktadır. YOLO yapısında, sınırlayıcı kutu koordinatları görüntü boyutlarına göre normalleştirilir ve açıklamaların görüntünün gerçek boyutuna göre değişmez olması sağlanır (Redmon & Farhadi, 2018). Bu normalleştirme, özellikle farklı çözünürlüklerdeki görüntülerde model eğitimi için kullanılmaktadır. Nesne başına tek bir satır ve normalleştirilmiş koordinatlar kullanan YOLO, kompakt gösterimi, uygun depolama ve işleme hızına katkıda bulunur. YOLO etiket dosyası, birden fazla nesne sınıfını barındırabilirken, sınıf etiketlerini ve önceden tanımlanmış sınıfların listesini ayarlayarak farklı veri kümelerine kolayca genişletilebilir (Redmon & Farhadi, 2018).

YOLO formatındaki etiket yapısı, gerçek zamanlı nesne algılama ve konumlandırma gerektiren çeşitli uygulamalarda yaygın olarak kullanılmaktadır. Bu uygulamalar arasında, nesne algılamanın verimliliği ve doğruluğunun kritik olduğu otonom sürüş, gözetim sistemleri ve robotik yer almaktadır. YOLO etiket yapısı Şekil 2.2'de görülmektedir.



Şekil 2.2. YOLO etiket yapısına örnek bir görsel.

2.2. Nesne Tespiti

Nesne tespiti, bir görüntü veya video karesindeki nesneleri tanımlamayı ve yerini algılamayı amaçlayan bilgisayarlı görü alanındaki önemli bir görevdir. Bu görev otonom sürüş, gözetim, tıbbi görüntüleme ve insan-bilgisayar etkileşimi dahil olmak üzere çok çeşitli uygulamalarda kullanılır (Liu ve ark, 2016). Nesne tespiti için kullanılan metodolojiler, geleneksel bilgisayar görüşü tekniklerinden gelişmiş derin öğrenme yaklaşımlarına kadar yıllar içinde önemli ölçüde gelişmiştir (Zhao ve ark, 2019).

Nesne tespiti için geleneksel yöntemler genellikle manuel özellik çıkarımı ve basit sınıflandırıcılara dayanmaktadır. Nesneleri tespit etmek için Yönlendirilmiş Gradyanların Histogramı (HOG) (Dalal & Triggs, 2005) ve Ölçek-Değişmez Özellik Dönüşümü (SIFT) (Lowe, 1999) gibi teknikler, daha sonra Destek Vektör Makineleri (SVM) (Cortes & Vapnik, 1995) veya Karar Ağaçları (Quinlan, 1986) gibi sınıflandırıcılarla beslenerek görüntü özelliklerini çıkarmak için yaygın olarak kullanılır. Bu yöntemler belirli senaryolarda etkili olsa da genellikle değişen nesne görünümleri ve karmaşık sahnelerde nesne tespit yetenekleri sınırlı kalmaktadır.

Derin öğrenmenin ortaya çıkışı, nesne tespiti alanındaki çalışmaların hızlanmasına ve verilerden özellik çıkarımını otomatikleştiren güçlü araçlar geliştirilmesine imkân tanımıştır. CNN (Lecun ve ark, 1998), görüntülerden hiyerarşik özellikleri yakalama yetenekleri nedeniyle modern nesne algılama sistemlerinin omurgası haline gelmiştir. Nesnelerin tespiti için öncü çalışmalardan biri, önce nesne özelliklerinin (ROI) çıkarılarak tahmin edildiği ve ardından sınıflandırıldığı iki aşamalı bir yaklaşım sunan Bölge Tabanlı Evrişimsel Sinir Ağı'dır (R-CNN) (Girshick ve ark, 2014). R-CNN çerçevesi üzerine inşa edilen, hesaplama karmaşıklığını azaltmak ve algılama hızını artırmak için teknikler sunan Hızlı R-CNN (Girshick, 2015) ve Daha Hızlı R-CNN (Ren ve ark, 2017) gibi çeşitli iyileştirmeler önerilmiştir. Daha Hızlı R-CNN, nesne bölgesi tahmin etme ve sınıflandırma aşamalarını tek bir ağda birleştirerek nesne tespit hızını önemli ölçüde artırmıştır. Günümüzde gerçek zamanlı nesne tespiti için popüler algoritmalar olarak, Tek Atışlı Çoklu Kutu Dedektörü (SSD) (Liu ve ark, 2016) ve YOLO gibi görüntüleri tek seferde işleyen dedektörler öne çıkmaktadır. Bu modeller, nesne algılamayı tek bir regresyon problemi olarak çerçeveler ve ağ üzerinden tek bir ileri geçişte tam görüntülerden doğrudan sınırlayıcı kutuları ve sınıf olasılıklarını

tahmin eder. Bu yaklaşım çıkarım süresini önemli ölçüde azaltarak bu algoritmaları gerçek zamanlı uygulamalar için uygun hale getirir (Redmon ve ark, 2016).

Derin öğrenmedeki gelişmelerle beraber gelişen nesne tespit algoritmalarına rağmen, nesne algılama, özellikle birden fazla nesne, değişen görüntü çözünürlükleri ve değişen ışık koşullarının olduğu karmaşık ortamlarda zorlu bir sorun olmaya devam etmektedir. Araştırmacılar, algılama doğruluğunu ve sağlamlığını iyileştirmek için yeni mimariler ve teknikler geliştirmeye devam etmektedir.

2.3. Nesne Tespitinde Kullanılan Modeller

2.3.1. YOLO

YOLO algoritması, nesne tespitinde doğrudan görüntü pikselleri, sınırlayıcı kutu koordinatları ve sınıf olasılıklarını tek bir regresyon problemi olarak çerçeveleyen ve bu sayede yüksek doğruluk ve hıza ulaşan gerçek zamanlı bir nesne algılama yaklaşımıdır. Nesne tespitini gerçekleştirmek için sınıflandırıcıları yeniden kullanan bölge tabanlı yöntemlerin aksine, YOLO'da görüntüye tek bir sinir ağı uygulanmaktadır. Bu ağ, görüntüyü bölgelere ayırır ve her bölge için sınırlayıcı kutuları ve olasılıkları tahmin eder. Tahminler daha sonra tahmin edilen olasılıklara göre ağırlıklandırılır (Redmon ve ark, 2016).

YOLO algoritması, birden fazla sınırlayıcı kutuyu ve sınıf olasılığını aynı anda tahmin etmek için tek bir CNN kullanır. YOLO'nun mimarisi, 24 evrişimli katmandan ve ardından 2 tam bağlı katmandan oluşur (Redmon ve ark, 2016).

YOLO, giriş görüntüsünü bir S x S ızgarasına böler. Bir nesnenin merkezi hangi ızgara hücresine denk gelirse, bu hücre nesneyi algılamaktan sorumludur. Her ızgara hücresi için YOLO, B sınırlayıcı kutu ve bu kutular için güven skoru belirlemektedir. Güven skoru, sınırlayıcı kutunun doğruluğunu ve sınırlayıcı kutunun, sınıftan bağımsız olarak gerçekten bir nesne içerip içermediğini yansıtır. Güven skoru Denklem 2.3'te görüldüğü gibi hesaplanmaktadır.

$$\boldsymbol{Güven\ Skoru = P(Nesne)x\ IoU_{gerçek}^{tahmin}} \tag{2.3}$$

Burada P(Nesne) kutuda bir nesnenin bulunma olasılığıdır ve $IoU_{gerçek}^{tahmin}$ tahmin edilen kutu ile temel gerçek arasındaki birleşim üzerindeki kesişimdir. Her sınırlayıcı kutunun tahmini (x, y, w, h, güven) olmak üzere 5 bileşenden oluşmaktadır. (x, y)

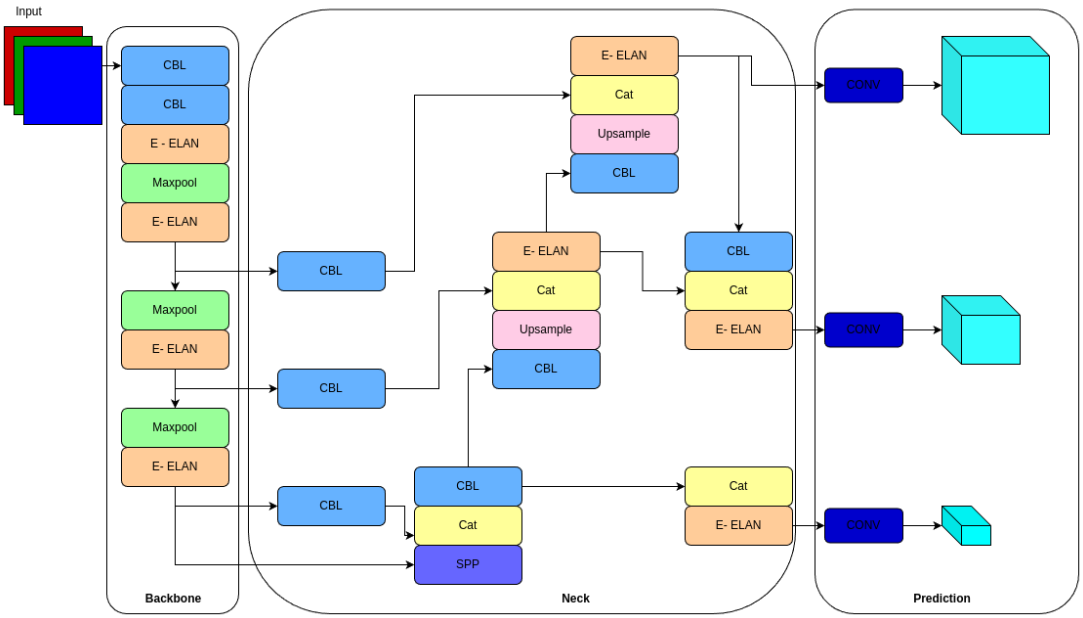
koordinatları, kutunun merkezini ızgara hücresine göre temsil eder. (w, h) sınırlayıcı kutunun tüm görüntüye göre genişliği ve yüksekliğidir. Her bir ızgara hücresi ayrıca koşullu sınıf olasılıklarını ön görmektedir. $P(Sınıf_i|Nesne)$ Sınıf olasılıkları Denklem 2.4'teki gibi hesaplanmaktadır.

$$P(Sınıf_i|Nesne) = \frac{P(Sınıf_i \cap Nesne)}{P(Nesne)} \tag{2.4}$$

Bu olasılıklar, bir nesne içeren ızgara hücresine bağlıdır. Veriler üzerindeki test aşamasında, YOLO'da koşullu sınıf olasılıkları ve bireysel kutu güven tahminleri Denklem 2.5'te görüldüğü gibi çarpılmaktadır.

$$P(Sınıf_i\ |Nesne)\ x\ IOU_{tahmin}^{gerçek} = P(Sınıf_i)\ x\ Güven\ Skoru \tag{2.5}$$

Bu çarpım, her sınırlayıcı kutu için sınıfa özgü güven skorunu vermektedir (Redmon ve ark, 2016). YOLO, tahmin ve sınıflandırmadaki hataları birlikte ele almak için çok parçalı bir yapıda kayıp miktarını optimize etmektedir. Bir nesne için birden fazla sınırlayıcı kutu yinelemesini önlemek için NMS ön görülen sınırlayıcı kutulara uygulanır. NMS, en yüksek güven puanlarına sahip sınırlayıcı kutuları tutar ve Birleşim Üzerinden Kesişimi (IoU) bir eşikten büyük olan diğer kutuları kaldırır, genellikle IoU eşit değeri 0,5 olarak ayarlanır (Bochkovskiy, 2020). YOLO algoritmasının genel ağ yapısı diyagramı Şekil 2.3'te görülmektedir.



Şekil 2.3. YOLO algoritmasının genel ağ yapısına örnek (Wang ve ark, 2023).

2.3.1.1. YOLOv5

Temel model mimarisi olarak Aşamalar Arası Kısmi Ağ (CSPNet) ve PANet içeren YOLOv5 (Jocher ve ark, 2020), hız ve doğruluğu etkili bir şekilde dengelemek üzere tasarlanmıştır. CSPNet, özellik haritasını iki parçaya bölerek ve ardından bunları aşamalar arası bir hiyerarşi aracılığıyla birleştirerek modelin hesaplama karmaşıklığını azaltmasına ve yüksek düzeyde doğruluğu korumasına olanak tanır. Bu da daha kolaylaştırılmış bir özellik çıkarma süreciyle sonuçlanır. PANet ise modelin farklı katmanlardaki özellikleri bir araya getirme becerisini geliştirerek özellikle küçük nesneler için daha sağlam nesne tespiti sağlar (Jocher ve ark, 2020). Bu mimarilerin entegrasyonu, YOLOv5'i hem hızın hem de doğruluğun çok önemli olduğu gerçek zamanlı uygulamalar için optimize edilmiş bir seçim haline getirmektedir.

2.3.1.2. YOLOv7

YOLOv7 (Wang ve ark, 2023), YOLOv5 üzerine inşa edilmiştir. Model performansı ve hızını artırmak için birkaç önemli yenilik mimariye entegre edilmiştir. YOLOv7'deki en önemli yenilik Verimli Katman Birleştirme (ELA) tekniğinin mimariye entegre eidlmedilmesidir (Wang ve ark, 2023). ELA, modelin farklı katmanlardaki özellikleri daha etkili bir şekilde bir araya getirme ve kullanma kapasitesini geliştirerek, özellikle üst üste binen veya nesnelerin yoğunlaştığı karmaşık senaryolarda daha iyi tespit performansı sağlar. YOLOv7'deki önemli yeniliklerden bir taneside daha kompakt ve uygun bir model yapısı elde etmek için eğitim sırasında

evrişim katmanlarını yeniden yapılandıran bir teknik olan Yeniden Parametrelendirilmiş Konvolüsyon (RepConv) kullanımıdır (Wang ve ark, 2023). RepConv, modelin genel hesaplama yükünü azaltırken yüksek doğruluğu korumasını sağlayarak, YOLOv7'yi özellikle verilerin sınırlı olduğu çalışmalar için uygun hale getirir.

2.3.1.3. YOLOv8

YOLOv8 (Reis ve ark, 2023), modelin giriş görüntülerinden özellikleri daha etkili bir şekilde çıkarma ve işleme yeteneğini geliştiren yeni bir omurga mimarisi sunmaktadır. YOLOv8'deki omurga, özellik temsilini iyileştirmek ve zorlu ortamlarda bile daha doğru tespit sonuçları elde edebilmek için tasarlanmıştır. Ayrıca YOLOv8, nihai nesne sınıflandırması ve lokalizasyonundan sorumlu olan optimize edilmiş bir kafa yapısı içermektedir (Reis ve ark, 2023). YOLOv8'deki yeniden tasarlanmış kafa, daha hızlı ve daha hassas tespitlere olanak tanıyarak nesnelerin tanımlanması ve konumlandırılmasında hata olasılığını azaltır. Bu da YOLOv8'i otonom araçlar, video gözetimi ve robotik gibi gerçek zamanlı işleme gerektiren uygulamalar için uygun hale getirir (Reis ve ark, 2023). YOLOv8'deki iyileştirmeler, doğruluğu koruyarak, algılama hızını artırmaya odaklanmaktadır.

2.3.1.4. YOLOv9

YOLOv9 (Wang ve ark, 2024), YOLOv7 üzerine inşa edilmiştir. Model performansını, doğruluğunu ve hızını yükseltmek için bir dizi yenilik mimariye entegre edilmiştir. YOLOv9'da modelin yapısını girdi verilerinin karmaşıklığına göre dinamik olarak ayarlayabilen Dinamik Sinir Ağları (DyNN) mimariye eklenmiştir. Bu ayarlama, YOLOv9'un hesaplama kaynaklarını daha etkili bir şekilde kullanmasını sağlayarak, tespit doğruluğunu geliştirirken işlem hızını artırır (Wang ve ark, 2024). DyNN, karışık ölçekli nesnelerin bulunduğu ortamlar veya hızla değişen dinamik sahneler gibi farklı düzeylerde nesne karmaşıklığına sahip senaryolarda etkili olabilmektedir.

YOLOv9'da, modelin giriş görüntüsünde çok ölçekli içerik bilgilerini yakalama yeteneğini geliştiren Gelişmiş Uzamsal Piramit Havuzlama (ASPP) modülü mimariye eklenmiştir (Wang ve ark, 2024). Bu modül, YOLOv9'un nesneler arasındaki mekansal ilişkileri daha iyi anlamasını ve yorumlamasını sağlayarak, dağınık veya son derece dinamik sahnelerde bile daha doğru konumlandırma ve sınıflandırma sağlar.

Ayrıca YOLOv9, aşırı uyumu önlemek ve genellemeyi iyileştirmek için eğitim sırasında otomatik olarak ayarlayarak modelin konvolüsyonel katmanlarını daha da iyileştiren Kendinden Düzenleyici Konvolüsyon (Self-RegConv) tekniğini içerir (Wang ve ark, 2024).

2.3.2. R-CNN

R-CNN, yüksek algılama doğruluğu elde etmek için bölge önerilerini CNN ile birleştiren nesne algılamada öncü bir derin öğrenme modelidir. Girshick ve ark. (2014) tarafından geliştirilen R-CNN, nesne algılamayı bölge önerisi oluşturma, özellik çıkarma ve sınıflandırma olarak üç ayrı aşamada ele almaktadır.

R-CNN, bir bölge önerileri kümesi oluşturarak nesne algılama işlemine başlar. Bu, tahmin edilen nesne bölgelerini oluşturmak için renk, doku, boyut ve şekil uyumluluğuna göre süper pikselleri birleştiren Seçici Arama (Uijlings ve ark, 2013) kullanmaktadır. Daha sonra, önerilen her bölgenin sabit boyuta ve uzunlukta örneğin (227 x 227) özellik vektörünün çıkarılması için bir CNN ağı kullanmaktadır. Çıkarılan özellikler daha sonra her bölge önerisi içindeki nesneyi sınıflandırmak ve sınırlayıcı kutu koordinatlarını iyileştirmek için kullanılır. Sınıflandırma için bir SVM eğitilir ve sınırlayıcı kutu regresyonu için doğrusal bir regresyon modeli eğitilir. R_i'nin tahmin edilen i'ninci bölgeyi temsil ettiği ve CNN ağı tarafından özellik çıkarma işleminin T(.) olarak verildiği durumda özellik vektörü, Denklem 2.6'da görüldüğü olmaktadır.

$$f_i = T(R_i) \tag{2.6}$$

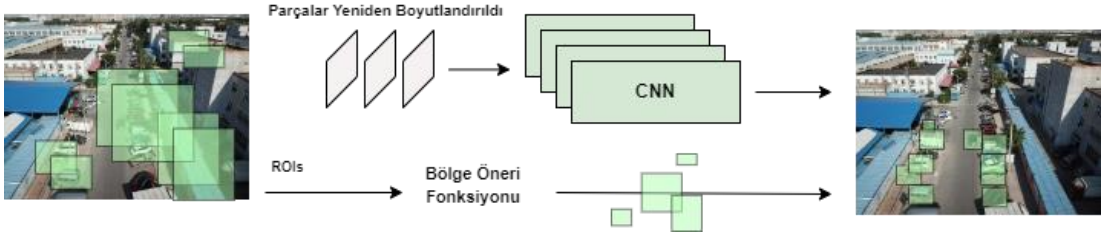
R_i tahmin bölgesi için, w ağırlık vektörünü ve b sınıflandırıcı tahmin terimini temsil ettiğinde, sınıflandırma puanı Denklem 2.7'de verildiği gibi hesaplanmaktadır.

$$s_i = w^T f_i + b \tag{2.7}$$

Sınırlayıcı kutu regresyonu için amaç, bölge önerisi koordinatlarını düzelten bir dönüşümü öğrenmektir. CNN, sınıflandırma ve sınırlayıcı kutu regresyon adımlarından gelen hataları geri yayarak nesne algılama veri kümesi üzerinde ince ayarlanabilmektedir. Sınırlayıcı kutu regresyonu, nesneye daha iyi uyması için bölge önerisi koordinatlarını (x, y, w, h) düzelten bir dönüşümü öğrenmektedir. Regresyon modeli, orijinal öneri koordinatlarını ayarlamak için kullanılan kaymaları (Δx, Δy, Δw, Δh) Denklem 2.8'deki gibi tahmin etmektedir.

$$x' = x + \Delta x \,.\, w \,, y' = y + \Delta y \,.\, h \,, w' = w \,.\, \exp(\Delta w) \,, h' = h \,.\, \exp(\Delta h) \tag{2.8}$$

R-CNN algoritmasının genel ağ yapı diyagramı Şekil 2.4'te görülmektedir.



Şekil 2.4. R-CNN algoritmasının ağ yapısı.

2.3.3. SSD

SSD algoritması, sınırlayıcı kutuların sabit boyutlu bir koleksiyonunu ve bu kutulardaki nesne sınıfı örneklerinin varlığına ilişkin puanları üreten ve ardından nihai tespitleri üretmek için maksimum olmayan bir bastırma adımı izleyen ileri beslemeli bir evrişimsel ağ etrafında tasarlanmıştır (Liu ve ark, 2016). SSD mimarisi, çeşitli boyutlardaki nesneleri işlemek için farklı çözünürlüklere sahip birden fazla özellik haritasından gelen tahminleri birleştirir. SSD'nin birincil bileşenleri arasında çok ölçekli özellik haritaları, evrişimli tahmin ediciler, varsayılan kutular, bir kayıp fonksiyonu ve NMS bulunur. SSD, farklı çözünürlüklerdeki birkaç özellik haritasında tahminler yaparak nesneleri birden fazla ölçekte algılar. Özellik haritasının l'inci katmanını x^l olarak belirlendiğinde her özellik haritası, sınıf puanlarını ve sınırlayıcı kutu kaymalarını tahmin etmek için bir dizi evrişimli filtreden geçirilir. Bir özellik haritası için x^l Denklem 2.9'daki gibi hesaplanmaktadır.

$$P^l = Conv(x^l) \tag{2.9}$$

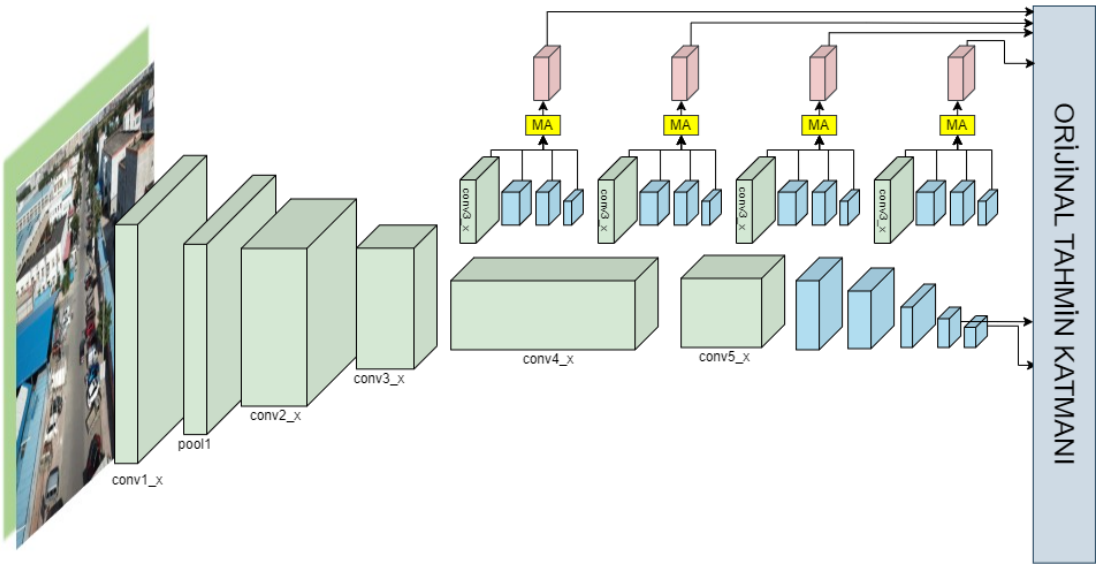
Burada P^l, l katmanındaki tahmin tensörünü temsil etmektedir.

SSD, çeşitli özellik haritalarındaki her konumda farklı en boy oranlarına sahip bir dizi varsayılan kutu kullanmaktadır. Her varsayılan kutu için, ağ varsayılan kutu koordinatlarına göre kaymaları ve her nesne kategorisi için güvenirliklerini tahmin eder. D_k 'nün konumda varsayılan kutuyu temsil ettiğini varsaydığımızda, ağ yapısının tahmin yapısı Denklem 2.10'da ve Denklem 2.11'de görüldüğü gibi olmaktadır.

$$c_k = \{c_{k,i}\}_{i=1}^{C} \tag{2.10}$$

$$b_k = (\Delta_{xk}, \Delta_{yk}, \Delta_{wk}, \Delta_{hk}) \tag{2.11}$$

Burada c_k sınıf puanları, b_k sınırlayıcı kutu kayma miktarlarıdır (Liu ve ark, 2016). Tahminler elde edildikten sonra SSD, gereksiz kutuları filtrelemek ve son tespit kümesini iyileştirmek için NMS uygular. Tahmin edilen kutular ve bunlarla ilişkili puanlar verildiğinde, NMS önemli örtüşmeye sahip kutuları bastırırken en yüksek puanlara sahip kutuları seçmektedir (Liu ve ark, 2016). SSD, çok ölçekli özellik çıkarımı için eklenen ek evrişimsel katmanlarla VGG16 (Simonyan & Zisserman, 2015) veya ResNet (He ve ark, 2016) gibi bir temel ağ kullanılarak uygulanabilir. Varsayılan kutuların, en boy oranlarının ve ölçeklerin seçimi, algılama doğruluğu ve hızını dengelemek için kritik öneme sahiptir. SSD algoritmasının tasarımı, birden fazla özellik haritasından gelen tahminleri birleştirerek ve nesneleri çeşitli ölçeklerde ele almak için evrişimli tahmincileri kullanarak verimliliği vurgular. Bu da onu gerçek zamanlı nesne algılama uygulamaları için güçlü bir araç haline getirir. SSD algoritmasının genel ağ yapısının diyagramı Şekil 2.5'te görülmektedir.



Şekil 2.5. SSD algoritmasının genel ağ yapısı.

2.4. Veri Setleri

Deneysel çalışmalar için iki farklı veri setinin incelenmesinin performans değerlendirmesi için daha uygun olduğu kararlaştırılmıştır. Veri setleri, yüksek

çözünürlüklü, çok sınıflı ve küçük nesne örneklerine göre belirlenmiştir. Literatür incelendiğinde xView (Lam ve ark, 2018) ve VisDrone (Du ve ark, 2019) veri setlerinin belirtilen özellikler için daha uygun olduğu görülmüştür. Bu tez çalışmasında deneysel çalışmalar bu iki veri seti üzerinde gerçekleştirilmiştir.

2.4.1. xView veri seti

xView veri seti, dünya çapında farklı coğrafi bölgeleri ve çevre koşullarını kapsayan uydu görüntüleri içermektedir. Bu veri çeşitliliği, algoritmaların genel performansını değerlendirmek ve iyileştirmek için oldukça uygundur. Veri seti, 60'tan fazla nesne sınıfı için 1 milyondan fazla etiket içerir (Lam ve ark, 2018). Bu nesne sınıfları arasında araçlar, binalar, altyapı unsurları, doğal yapılar ve diğer nesneler bulunur. Veri seti, askeri, şehir planlama, afet yönetimi ve çevresel izleme gibi çeşitli alanlarda kullanım için tasarlanmıştır. xView veri setinin genel yapısı Tablo 2.1'de detaylı olarak görülmektedir.

Tablo 2.1. xView veri setinin genel yapısı.

Özellikler	Açıklama
Kapsam	Dünya genelinde çeşitli coğrafi bölgeler ve çevre koşulları
Çözünürlük	Her pikselin 30 cm'yi temsil ettiği çok net görüntüler
Etiket	1 milyondan fazla etiketleme
Nesne Sınıfları	60 nesne sınıfı
Uygulama Alanları	Askeri, şehir planlama, afet yönetimi, çevresel izleme

2.4.2. VisDrone veri seti

VisDrone veri seti, İHA tarafından çekilen görüntülerde nesne algılama ve izleme gibi bilgisayarlı görüş görevlerini desteklemek için oluşturulmuştur. Çeşitli hava ve ışık koşullarında, kentsel ve kırsal gibi farklı çevre şartlarında çekilen görüntülerden oluşur (Du ve ark., 2019). Bu çeşitlilik, algoritmaların farklı senaryolarda test edilmesine ve geliştirilmesine olanak tanır. Yüksek çözünürlüklü ve ayrıntılı görüntüler içerir. Bu da küçük nesnelerin ve karmaşık sahnelerin doğru bir şekilde analiz edilmesini sağlar. VisDrone veri setinin genel yapısı Tablo 2.2'de detaylı olarak görülmektedir.

Tablo 2.2. VisDrone veri setinin genel yapısı.

Özellikler	Açıklama
Kapsam ve Çeşitlilik	Çeşitli hava koşulları, ışık koşulları ve çevre koşulları altında çekilen görüntüler
Görüntü Kalitesi ve Çözünürlük	Yüksek çözünürlüklü ve detaylı görseller
Etiket	1 milyondan fazla etiketleme
Nesne Sınıfları	10 farklı nesne sınıfı için yüz binlerce etiketli örnek
Uygulama Alanları	Güvenlik izleme, trafik yönetimi, afet müdahalesi ve şehir planlama

2.5. Çalışma Ortamı

Bu tez çalışmasında, yüksek performanslı hesaplamalar ve veri işleme görevlerini gerçekleştirmek için gelişmiş donanım ve yazılım bileşenleri kullanılmıştır. Çalışma ortamında, yüksek hızlı veri işleme yeteneği ve çoklu görev performansıyla bilinen 12. nesil Intel Core i9 işlemci, yüksek çözünürlüklü grafik işleme ve derin öğrenme modellerini eğitmek için optimize edilmiş bir NVIDIA GeForce RTX 3090 Ti grafik kartı (GPU) kullanılmıştır. Sistem, büyük veri kümelerinin hızlı bir şekilde işlenmesini sağlayarak bellek kısıtlamalarını azaltan 64 GB DDR5 RAM ile donatılmıştır. Veri analizi, derin öğrenme ve genel programlama görevleri için Python programlama dili kullanılırken, kod geliştirme ve düzenleme için bütünleşmiş geliştirme ortamı (IDE) olarak Visual Studio Code (VS Code) kullanılmıştır. VS Code, eklenti desteği ve kolay kullanım arayüzü ile kodlama sürecini kolaylaştırmıştır. Donanım ve yazılım araçlarının kapasitesi, çalışma için gerekli hesaplama gücünü ve esnekliği sağlayarak analitik süreçlerin ve model geliştirmenin rahat bir şekilde yürütülmesini sağlamıştır.

2.6. Performans Ölçütleri

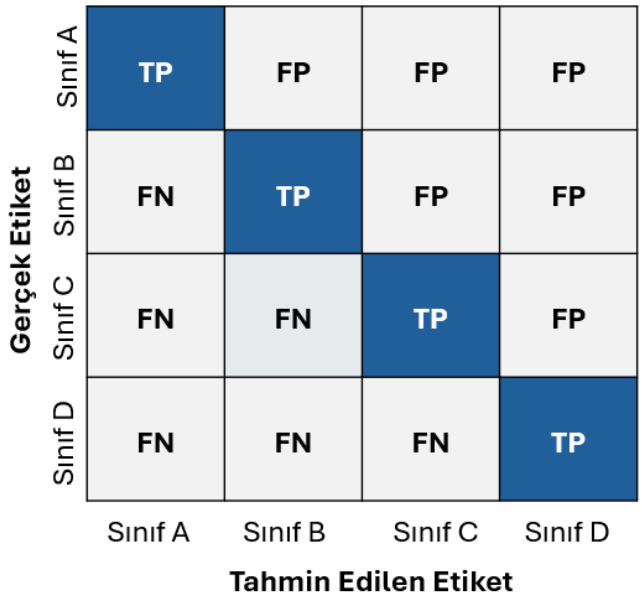
Sınıflandırma ve nesne tespiti performans sonuçlarının değerlendirilmesinde, eğitilen modeller için bir karmaşıklık matrisi kullanılır. Karmaşıklık matrisi, modelin tahminlerine ve gerçek değerlerine göre dört farklı şekilde yapılandırılır. Karmaşıklık matrisinin yapılandırması Şekil 2.6'da gösterilmiştir. Karmaşıklık matris, aşağıdaki gibi tanımlanan TP, FP, FN ve TN değerlerinden oluşur:

TP (Gerçek Pozitif): Model tarafından gerçek sınıf için doğru tahmin edilen örnekler.

FP (Yanlış Pozitif): Model tarafından gerçek sınıf için yanlış tahmin edilen örnekler.

FN (Yanlış Negatif): Modelin yanlış bir şekilde negatif tahmin ettiği örnekler.

TN (Gerçek Negatif): Modelin doğru bir şekilde negatif tahmin ettiği örnekler.



Şekil 2.6. Karmaşıklık matrisi yapısına genel bir örnek.

IoU metriği, model tarafından tahmin edilen sınırlayıcı kutu ile gerçek sınırlayıcı kutu arasındaki örtüşmenin bu kutuların birleşimine oranı olarak tanımlanır. Standart IoU değeri 0,5 olarak ayarlanır. Bu eşik değerinin üstünde bir IoU'ya sahip tahminler TP olarak kabul edilirken, altında olanlar FP olarak kabul edilir. Bir nesnenin görüntüde bir sınırlayıcı kutusu varsa ancak model bir sınırlayıcı kutu tahmin edemezse, bu FN olarak sınıflandırılır. Görüntüdeki belirlenmiş nesneler olmayan diğer tüm arka plan alanları TN'i oluşturur.

Karmaşıklık matrisi kullanılarak, Ortalama Hassasiyet (mAP) ile dört farklı performans metriği hesaplanır. Hassasiyet, her sınıf için doğru tahminlerin toplam tahmin sayısına oranı olarak Denklem 2.12'daki hesaplanmaktadır. Özgünlük, bir nesne mevcut olduğunda doğru şekilde tanımlanan örneklerin oranıdır ve Denklem 2.13'teki gibi hesaplanmaktadır. F1 puanı, hassasiyet ve özgünlüğün harmonik ortalamasıdır ve iki metrik arasında bir denge sağlamaktadır. F1 puanının hesaplaması Denklem 2.14'te gösterilmektedir.

Hassasiyet, modelin verilerdeki tüm ilgili örnekleri doğru bir şekilde tanımlama yeteneğini ölçer. Yüksek hassasiyet, modelin tespitindeki doğru pozitiflerin çokluğunu göstermektedir.

$$Hassasiyet = \frac{TP}{TP+FN} \quad (2.12)$$

Özgünlük, modelin belirli bir nesne olmadığında gerçek olumsuz örnekleri doğru bir şekilde tanımlama yeteneğini ifade eder. Tüm gerçek olumsuz durumlar arasından doğru bir şekilde tanımlanan olumsuz örneklerin oranını temsil eder.

$$Özgünlük = \frac{TN}{TN+FP} \quad (2.13)$$

F1 Puanı, hassasiyet ve özgünlüğü birleştiren dengeli bir ölçüm metriğidir. Eşit olmayan bir sınıf dağılımı olduğunda kullanışlıdır. Özellikle hem hassasiyet hem de özgünlük önemli olduğunda genel performansı değerlendirmeye yardımcı olan puanı sağlar.

$$F1\ Puanı = 2\ x\ \frac{hassasiyet\ x\ özgünlük}{hassasiyet\ + özgünlük} \quad (2.14)$$

Ortalama Hassasiyet (AP), hassasiyet-özgünlük eğrisinin altındaki alan olarak belirlenir. Eğri, sol üst ve sağ alt köşelerdeki 1 değerine ne kadar yaklaşırsa, tahmin doğruluğu o kadar yüksek olur. AP hesaplaması Denklem 2.15'te gösterilmiştir.

$$AP = \int_0^1 p(r)dr \quad (2.15)$$

Ortalama Hassasiyet (mAP), her sınıf için AP değerlerinin toplamının toplam sınıf sayısına bölünmesiyle elde edilir. Bu ölçüm metriği, modelin tüm sınıflardaki genel algılama doğruluğunu temsil eder ve modelin çok sınıflı bir ortamda nesneleri ne kadar doğru ve tutarlı bir şekilde tanımladığına dair ortalama bir ölçü sağlar. mAP hesaplaması Denklem 2.16'da gösterilmiştir.

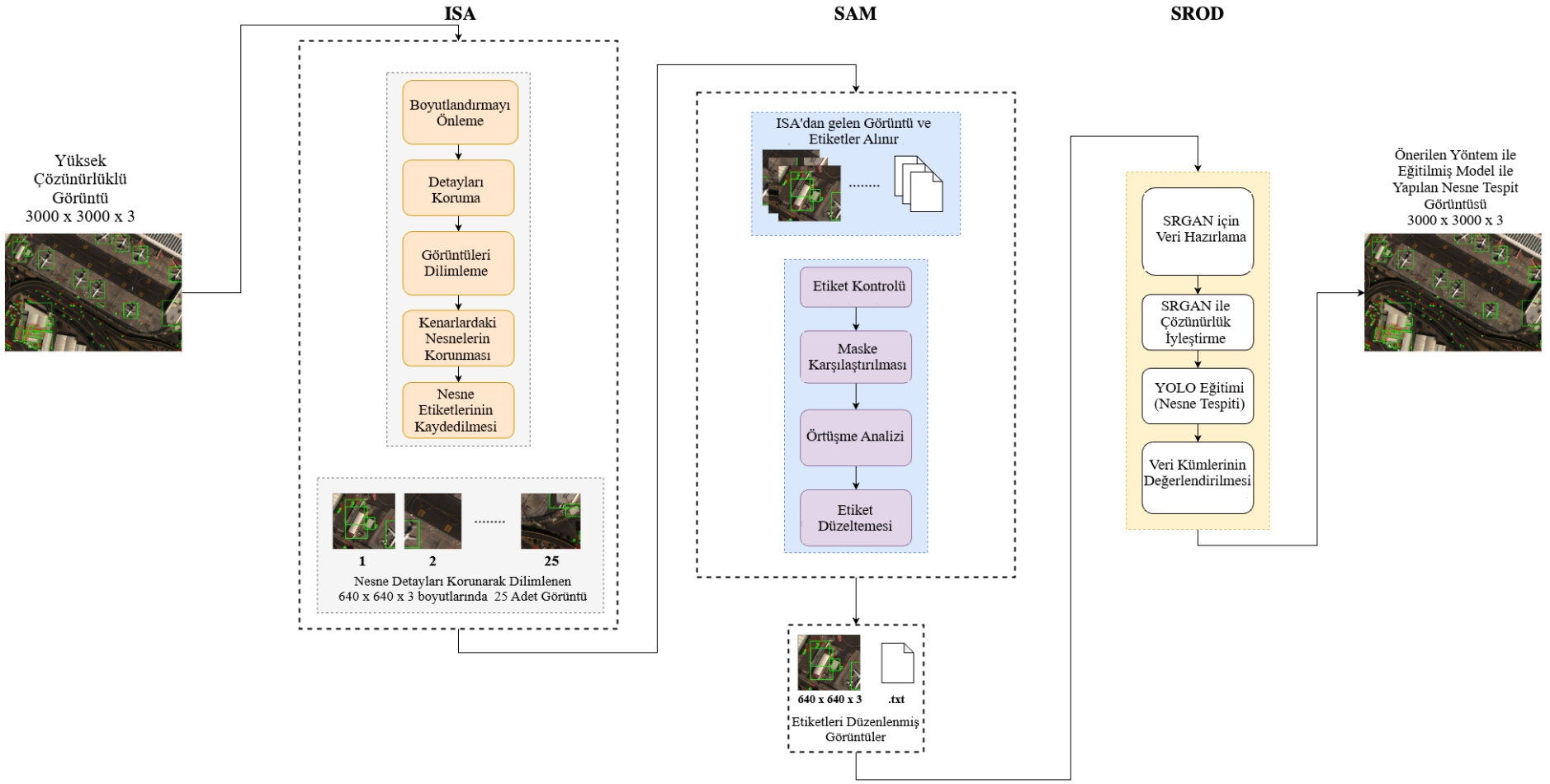
$$mAP = \frac{\sum AP_i}{N} \quad (2.16)$$

3. ÖNERİLEN YÖNTEM

Bu tez çalışması, yenilikçi bir sistem önererek, nesne tespitinde mAP değerinin artırılması amaçlamaktadır. Bu yaklaşımdaki ilk adım, nesne tespit görevlerinde sıklıkla bir darboğaz olan görüntüyü yeniden boyutlandırma zorluğunun üstesinden gelmeyi içerir. Bunu ele almak için, kritik görüntü ayrıntılarını kaybetmeden yeniden boyutlandırma sürecini yönetmek üzere özel olarak tasarlanmış Görüntü Dilimleme Algoritmasını (ISA) önerilmektedir. ISA'nın uygulanmasının ardından, nesne etiketlerinin doğruluğunu ve bütünlüğünü korumak çok önemlidir. Bu doğruluğu kontrol edip gerekli durumlarda düzenlemek için SAM (Kirillov ve ark, 2023) sisteme entegre edilmiştir.

Nesne tespitinin temel görevi için, Süper Çözünürlüklü Nesne Tespiti (SROD) mimarisi sunulmuştur. Bu algoritma, özellikle görüntü çözünürlüğünün sınırlayıcı bir faktör olduğu durumlarda tespit performansını iyileştirmek için süper çözünürlük tekniklerinin öğelerini nesne tespit yöntemleriyle birleştiren bir hibrit yapıdır.

Önerilen yöntemlerin etkinliğini doğrulamak için, çok sınıflı ve yüksek çözünürlüklü verilere sahip xView ve VisDrone veri setleri kullanılmıştır. Bu veri kümeleri, farklı ortamlar ve zorluklar arasında nesne algılama sistemlerinin sağlamlığını ve geliştirilebilirliğini test etme yetenekleri nedeniyle seçilmiştir. Şekil 3.1’de, önerilen sistemi oluşturan ana yapı taşlarının ayrıntılı bir genel yapısı görülmektedir



Şekil 3.7. Önerilen yöntemin genel yapısı.

Geliştirilen dilimleme algoritmasının mevcut maske etiketli veri kümelerine sorunsuz bir şekilde uygulanabilmesi ve YOLO algoritmasının performansını, maske etiketli veriler üzerinde değerlendirmek için veri etiketlerinin uygun formatta olması gerekmektedir. Maske etiketlerinin YOLO etiket formatına manuel olarak dönüştürülmesi, özellikle veri karmaşıklıklarının fazla olduğu biyomedikal gibi alanlarda uzman bilgisine ihtiyaç duyulmaktadır. Aynı zamanda bu etiketleme işleminin manuel olarak yapılması önemli bir ek işlem yükü getirecektir. Dilimleme algoritmasının ve YOLO performans değerlendirmesinin pratikliğini artırmak için otomatik bir etiket dönüştürme algoritması önerilmiştir. Bu algoritma, maske etiketlerini otomatik olarak YOLO etiket formatına dönüştürmek, böylece manuel iş yükünü hafifletmek ve uzman girdisine olan ihtiyacı en aza indirmek için tasarlanmıştır. Bu dönüştürme otomasyonu, özellikle yüksek çözünürlüklü görüntülerin doğru bir şekilde işlenmesini kolaylaştırmada literatüre önemli ölçüde katkı sunacaktır.

Önerilen algoritmayla birlikte, kullanımını kolaylaştırmak için bir arayüz geliştirilmiştir. Bu arayüzün amacı, araştırmacıların maske etiketlerini kolay bir şekilde YOLO etiket formatına dönüştürmelerini sağlamak ve böylece iş akış hızını artırmaktır.

3.1. Etiket Dönüşüm Algoritması

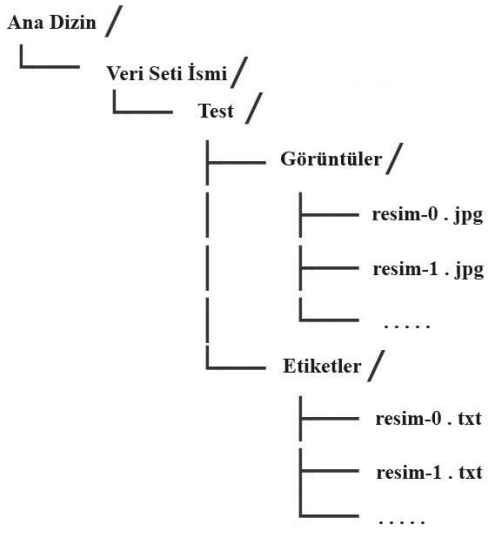
YOLO algoritması, nesne algılamayı kolaylaştırmak için etiketlerin belirli bir şekilde biçimlendirilmesini gerektirir. Özellikle, her etiket dosyası .txt biçiminde olmalı ve dosya uzantısı hariç, karşılık gelen görüntü dosyasıyla aynı isimde olmalıdır. Bu etiket dosyaları, nesne algılama ve sınıflandırma için kullanılan çokgenlerin koordinatlarını içerir ve dosyadaki her satır, sınıf indeksi aracılığıyla bir nesneyi temsil etmektedir. Koordinatları ise Denklem 3.1'de görüldüğü gibi olmaktadır.

$$sınıf\_dizin\ p1.x\ p1.y\ p2.x\ p2.y\ p3.x\ p3.y, vb. \tag{3.1}$$

Burada sınıf_dizin, sınıf listesindeki sınıf adını temsil eden 0 ila n-1 arasında bir sayıdır ve p1, p2 ve p3 ardışık noktaların poligonlarını temsil eder.

YOLO algoritması için gereken veri ve etiket dosyalarının bulunduğu dizin yapısı Şekil 3.2'de gösterilmiştir. Mevcut veri kümelerini YOLO tabanlı nesne algılama için

uyarlamak amacıyla, etiketli maskeler önce gri tonlamalı bir biçime dönüştürülür. Bu dönüşümün ardından bir kontur çıkarma işlemi gelir. Kontur çıkarma yoluyla elde edilen sınır koordinatları daha sonra YOLO biçimine eşlenir ve her biri karşılık gelen görüntü dosyasıyla aynı şekilde adlandırılan .txt dosyaları olarak kaydedilir. Bu işlem her maske için bir .txt dosyasının oluşturulmasıyla sonuçlanır.

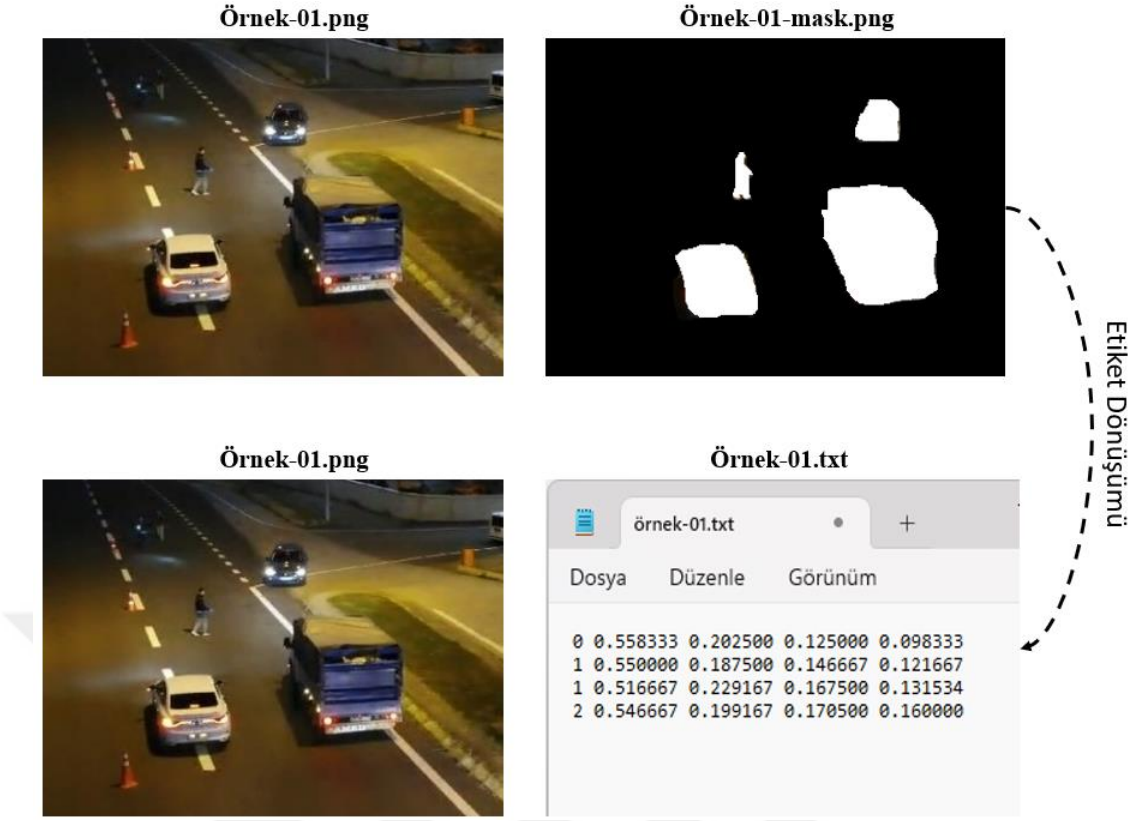


Şekil 3.8. YOLO algoritması için görüntü ve etiketlerin dosya yapısı.

Görüntü verileri ve etiketleri içeren birincil klasör, Şekil 3.2'de gösterildiği gibi, genellikle eğitim ve test için iki alt klasör içerir.

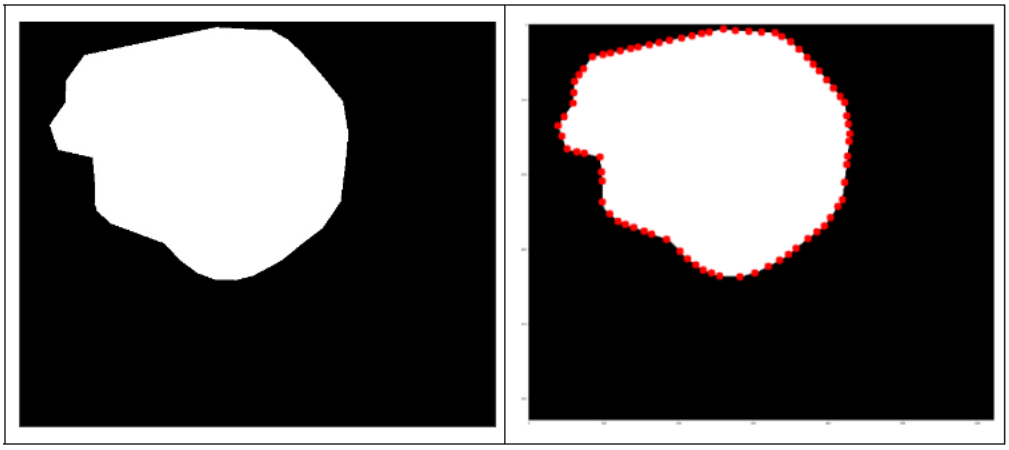
- **Görüntü dosyaları:** Model eğitimi ve testi için görüntü dosyalarını içerir.
- **Etiket dosyaları:** Her görüntü için karşılık gelen etiket dosyalarını içerir.

Örnek bir görsel için dizinde bulunan maske ve YOLO etiket dosya yapısı Şekil 3.3'te görüldüğü gibi olmaktadır.



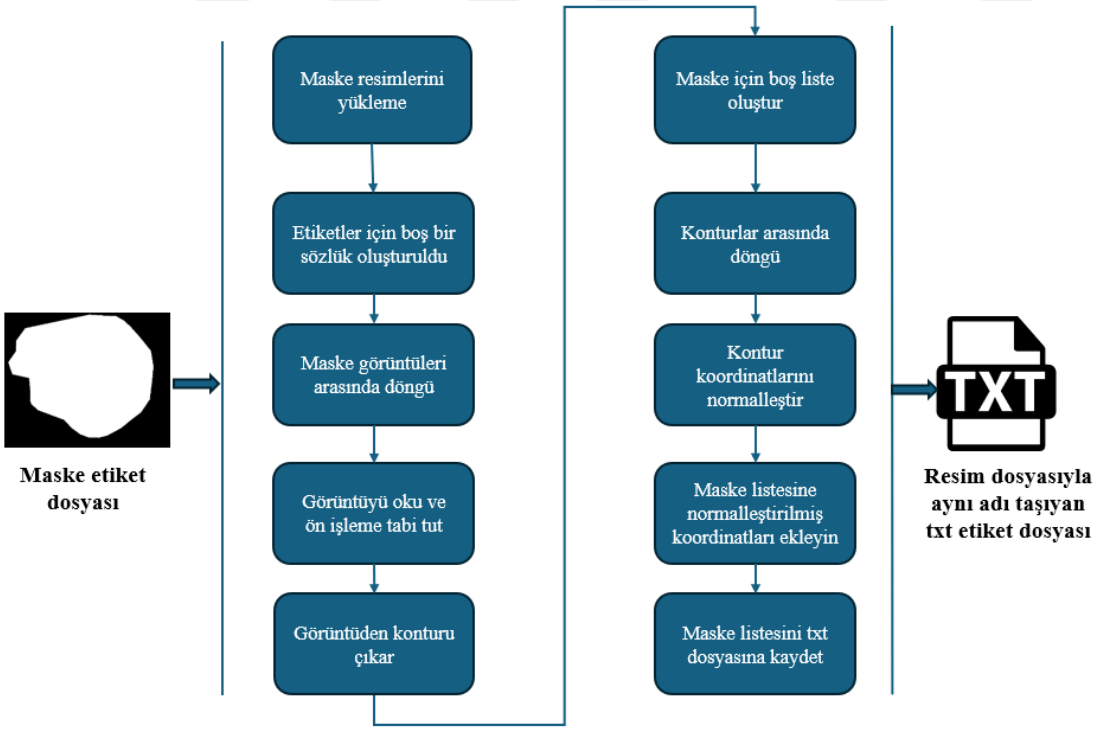
Şekil 3.9. Örnek bir görsel için Maske ve YOLO etiket formatlarındaki dosya yapısı

Görüntü klasörü genellikle JPEG, PNG veya desteklenen diğer biçimlerdeki görüntü dosyaları içermektedir. Etiket klasörü, genellikle YOLO biçimiyle uyumlu metin dosyaları olarak biçimlendirilen her görüntü dosyasına karşılık gelen etiket dosyalarını içerir. Her etiket dosyasının, .txt uzantısıyla karşılık gelen görüntü dosyasıyla aynı adı taşıması gerekir. YOLO biçiminde, her etiket dosyası her nesnenin konumunu ve türünü belirtir ve her satır tek bir nesne hakkında bilgi sağlar. Önerilen algoritma ile etiketli maskeler, YOLO formatına dönüştürmede hızlı ve doğru sonuçlar sağlamaktadır. Nesnelerin sınırlarını tanımlayan kontur çıkarma işlemi, YOLO çerçevesi içinde nesne algılamanın hassasiyetini ve doğruluğunu artırır. Kontur çıkarma yoluyla elde edilen sınır koordinatlarının görsel bir temsili Şekil 3.4'te gösterilmiştir.



Şekil 3.10. Maske etiketli verilerden kontur çıkarma örneği.

Kontur çıkarma, bir görüntüdeki nesnelerin sınırlarını, çizgilerini ve köşelerini tanımlamak için kullanılan bir yöntemdir. Bu teknik, nesnelerin çevrelerini belirler ve boyut, şekil ve konum gibi kritik bilgileri çıkarır (Gong ve ark, 2018). Kontur çıkarma, nesnelerin karakteristik özelliklerinin anlaşılmasına yardımcı olduğu için endüstriyel görüntü işleme, robotik, tıbbi görüntüleme ve biyometrik tanıma dahil olmak üzere çeşitli alanlarda yaygın olarak kullanılır. Maskeleri YOLO etiket biçimine dönüştürme sürecini gösteren akış şeması Şekil 3.5'te gösterilmiştir.



Şekil 3.11. Maskelerin YOLO uyumlu etiketlere dönüştürülmesine ilişkin akış şeması.

Etiket dönüşüm algoritmasının yapısı sözde kodu Algoritma 1 olarak Şekil 3.6'da görülmektedir.

Algoritma 1: Maske Etiketlerinin YOLO etiket formatına dönüşüm adımları

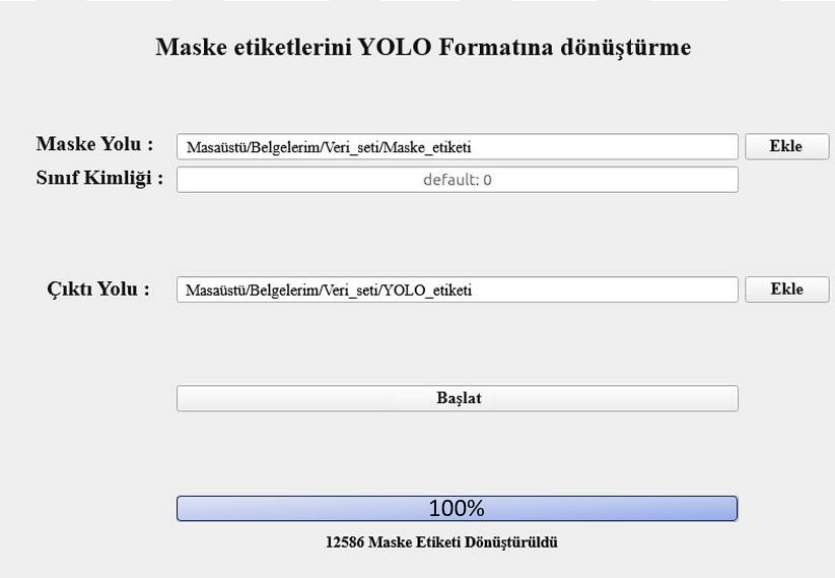
```
Mask_klasörü ← Maske etiketlerinin bulunduğu klasörü seç
Çıkış-klasörü ← Çıkışta.txt dosyalarının kaydedileceği klasörü seç
for maske_doyası in maske_klasörü: ← Her maske etiketi için işlem yapılacak
    Adım 1: Maske dosyasını gri tonlamalı olarak yükle
    Adım 2: Görüntülerdeki dış konturları tespit et
    Adım 3: Her kontur için sınır kutusu oluştur
    for contour in contours:
        x_min, y_min, width, height ← hesapla_sınırlayıcı_kutu(contour)
        x_max        x_min + width
        y_max        y_min + height
        sınırlayıcı_kutu_ekle ([x_min, y_min, x_max, y_max])
    end
    Adım 4: Yüklenen maske görüntüleri ve hesaplanan kutuların listesini döndür
end
Döndürülen kutular YOLO etiket formatına dönüştürülür
for kutu in kutular:
    x_merkez ← (x_min + x_max) / 2 / görüntü_width
    y_merkez ← (y_min + y_max) / 2 / görüntü_height
    kutu_width ← (x_max- x_min) / görüntü_width
    kutu_height ← (y_max- y_min) / görüntü_height
end
Her maske için etiketler bir. txt dosyasına yazılır
```

Şekil 3.12. Etiket dönüşüm algoritmasının sözde kodu

Bu tez çalışmasında, maske etiketlerinin YOLO algoritmasıyla uyumlu bir formata dönüştürülmesini kolaylaştırmak için bir arayüz geliştirilmiştir. Geliştirilen arayüz için PyQt5 kütüphanesi kullanılmıştır. PyQt5, Qt uygulama çerçevesi için bir Python bağlayıcısıdır ve yerel bir görünüm ile platformlar arası masaüstü uygulamaları oluşturmasına olanak tanır. PyQt5'in öne çıkan özelliklerinden biri, uygulamanın

farklı bölümleri arasında iletişimi sağlayarak olay odaklı programlamayı basitleştiren sinyal ve yuva mekanizmasıdır (Meier, 2019). PyQt5, NumPy ve Matplotlib gibi çeşitli kütüphanelerle uyumludur ve bu da onu bilimsel uygulamalar ve veri görselleştirmeleri için ideal hale getirir. Arayüzleri görsel olarak tasarlamak için Qt Designer aracını içerir ve geliştirme sürecini hızlandırır.

Bu arayüz, YOLO çerçevesini kullanarak nesne algılama görevleri için etiketli veri kümelerini hazırlama sürecini kolaylaştırmak için özel olarak tasarlanmıştır. Arayüzün birincil işlevi, mevcut maske etiketli veri kümelerinden etiketleri yüklemek ve bu etiketleri otomatik olarak YOLO formatında txt dosyaları olarak kaydetmektir. Bu otomasyon, özellikle veri etiketlemenin karmaşık ve zaman alıcı olduğu alanlardaki bu tür dönüştürmeler için gereken manuel çalışmayı ve uzman müdahalesini önemli ölçüde azaltacaktır. Tasarlanan arayüzün görsel temsili Şekil 3.7'de gösterilmiştir.



Şekil 3.13. Maske etiketlerini YOLO formatına dönüştürmek için tasarlanan arayüz.

Arayüz aşağıdaki temel bileşenleri içermektedir:

- **Maske Yolu**: Bu giriş alanı, kullanıcıların maske etiketli görüntülerin depolandığı dizin yolunu belirtmelerine olanak tanır. Kullanıcılar, giriş alanının yanındaki "Ekle" düğmesine tıklayarak yolu ekleyebilirler.
- **Sınıf Kimliği:** Bu alan varsayılan olarak 0 değerine ayarlanmıştır ancak kullanıcı tarafından maskeli etiketli görüntülerdeki nesneler için sınıf dizinini belirtmek üzere ayarlanabilir. Bu esneklik, arayüzün çeşitli nesne sınıflarına sahip veri kümeleri için kullanılabilmesini sağlar.

- **Çıkış Yolunu Kaydet:** Bu giriş alanı, kullanıcıların dönüştürülen .txt dosyalarının kaydedileceği dizin yolunu belirtmelerine olanak tanır. Maske Yolu girişine benzer şekilde, kullanıcılar giriş alanının yanındaki "Ekle" düğmesine tıklayarak yolu ekleyebilirler.
- **Başlat Düğmesi:** "Başlat" düğmesine tıklandığında, arayüz dönüştürme işlemini başlatır. Dönüştürmenin ilerlemesi, arayüzün alt kısmındaki bir ilerleme çubuğu ile görsel olarak temsil edilir ve kullanıcılara işlemin durumu hakkında gerçek zamanlı geri bildirim sağlar.

Arayüz, önce maskeli etiketli görüntüleri gri tonlamalı biçime dönüştürerek, ardından nesne sınırlarını belirlemek için kontur çıkarımı yaparak çalışır. Çıkarılan sınır koordinatları daha sonra, her nesnenin sınıf indeksini ve poligon koordinatlarını, ilgili görüntü dosyasıyla aynı adı paylaşan bir .txt dosyasına kaydetmeyi içeren YOLO formatına eşlenir.

Etiket dönüştürme sürecini otomatikleştirerek, bu arayüz yalnızca YOLO tabanlı nesne tespiti için veri kümelerini hazırlamanın hızlanmasını ve doğruluğunu artırmakla kalmaz, aynı zamanda etiketleme sürecinde insan hatası olasılığını da en aza indirir. Kullanılan kontur çıkarma yöntemi, nesne sınırlarının hassas bir şekilde tanımlanmasını sağlayarak YOLO algoritmasının tespit performansını iyileştirir.

Bu arayüzün geliştirilmesi, nesne tespiti görevleri için veri kümelerinin ön işlenmesinde önemli bir ilerlemeyi temsil eder. Arayüzün sağladığı kullanım kolaylığı ve otomasyonun, YOLO algoritmasının çeşitli araştırma ve uygulama alanlarında daha geniş bir şekilde benimsenmesini kolaylaştıracağı ve dolayısıyla literatüre önemli katkısı olacağı düşünülmektedir.

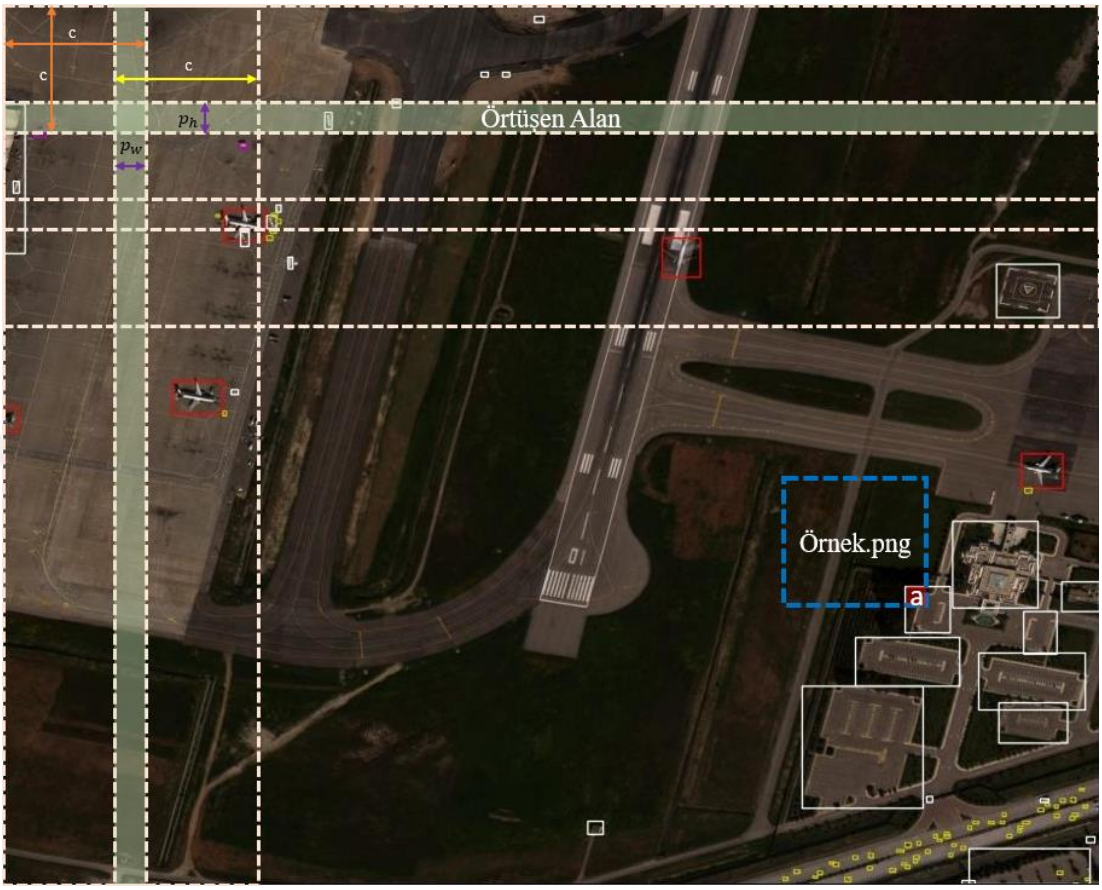
3.2. Dilimleme Algoritması (ISA)

Nesne tespit modelleri hem eğitim hem de çıkarım aşamalarında sabit bir giriş boyutuna ihtiyaç duyar. Bu gereklilik, yüksek çözünürlüklü görüntülerle çalışırken görüntülerin modelin giriş boyutuna uygun olarak yeniden boyutlandırılmasını zorunlu kılar. Ancak, bu yeniden boyutlandırma işlemi sırasında önemli piksel kayıpları meydana gelir. Piksel kayıpları, etiketli nesnelerin kritik özelliklerinin kaybolmasına neden olarak, modelin doğru ve etkili bir şekilde nesne tespiti yapma yeteneğini olumsuz yönde etkileyebilir. Bu tür piksel kayıplarını önlemek ve modelin

performansını artırmak amacıyla, veri artırma teknikleri sıklıkla kullanılmaktadır. Rastgele kırpma veya ölçekleme gibi teknikler, veri kümesinin boyutunu artırarak modelin performansını geliştirmeye çalışır. Ancak, mevcut kırpma işlemi yaklaşımlarında, görüntüdeki nesneler uygun şekilde kırpılamayabilir, bu durum modelin yanlış öğrenmesine neden olabilmektedir. Dolayısıyla, bu tür tekniklerin kullanımı, her ne kadar veri çeşitliliğini artırsa da nesne tespiti doğruluğunu olumsuz etkileyebilecek potansiyel hatalara sebebiyet verebilmektedir.

Bu tez çalışmasında, yüksek çözünürlüklü görüntüler üzerinde nesne tespiti sistemlerinin performansını optimize etmek amacıyla geliştirilmiş yeni bir veri ön işleme tekniği olan ISA önerilmektedir. ISA, geleneksel veri artırma tekniklerinin yol açabileceği sorunları aşmak ve nesne tespiti doğruluğunu artırmak için özel olarak tasarlanmıştır. ISA algoritmasının temel prensibi, dilimleme boyutunu nesne tespit modelinin gerektirdiği giriş boyutuna uygun şekilde ayarlamaktır. Bu sayede, görüntü dilimlendiğinde, modelin giriş boyutlarıyla uyumlu hale gelirken nesnelerin kritik özelliklerinin kaybı ortadan kaldırılmış olur. ISA algoritmasında, dilimleme işlemi kayan ve örtüşen bir yapı üzerine kuruludur. Dilimleme, orijinal görüntünün boyutlarına ve modelin gerektirdiği dilimleme boyutlarına göre yapılır. Dilimleme sırasında kullanılan örtüşme oranı, görüntünün genişlik ve yükseklik boyutlarına göre otomatik olarak ayarlanır. Bu örtüşme oranı, dilimleme işlemi sırasında nesnelerin kenar noktalarda parçalanmasını ve bu nedenle kaybolmasını engelleyerek, nesne bütünlüğünün maksimum düzeyde korunmasını sağlar. Bu tez çalışmasında dilimleme boyutu 640 belirlenmiş ve xView ve VisDrone veri setlerindeki görüntüler 640 x 640 olarak dilimlenmiştir. Dilimleme yapısındaki kayan örtüşen yapı sayesinde kenar noktalara denk gelen nesnelerin etiketleri ilk görsel için parçalanmış olsada bir sonraki görüntüde nesne etiketi bütünlüğü korunmuştur. Parçalanmış nesnelerin etiketlerinin doğru şekilde nesneleri temsil etme yeteneğini korumak için parçalanan nesnelerin etiketlerinin boyutları orijinal nesne etiket boyutlarıyla oranlanmıştır. Parçalanmışnesne etiketin boyutu eğer orijinal etiket boyutunun %20'sinden küçük olduğu durumlar için parçalanmış etiketler kaldırılmıştır. Bu sayede nesneyi temsil edemeyecek kadar küçük olan nesne parçaları için etiket bulunmayacaktır. Böylelikle modellerin nesneler hakkında yanlış bilgiler öğrenmesinin önlenmesi amaçlanmıştır. Özellikle ISA algoritmasının kayan örtüşen yapısı sayesinde nesne bütünlüğünün korunması ve nesneler için etiketlerin oranlarak hesaplanıp kaydedilmesi ISA

algoritmasının mevcut dilimleme yöntemlerine göre önemli farkını oluşturmaktadır. Görüntü üzerinde ISA algoritmasının uygulanmasına ait görsel Şekil 3.8'de görülmektedir.



Şekil 3.14. xView veri kümesinden alınan bir görüntüye ISA yapısının uygulanmasına dair bir örnek.

Şekil 3.8'deki (a) İşaretli bölgenin etiket boyutu orijinal görüntüdeki nesnenin etiket boyutunun %20'sinden küçük olduğu için Örnek. png görüntüsünden etiketi kaldırılacak bölge.

Görüntü dilimleme işlemi tamamlandıktan sonra, orijinal görüntüdeki sınırlayıcı kutular, dilimlenmiş yeni görüntüler için orantılı bir şekilde yeniden ayarlanır ve nesne etiketleri bu yeni görüntüler için yeniden hesaplanır. Örneğin dilimleme ile elde edilen Örnek png görüntüsü için, (a) noktasına karşılık gelen etiket kalıntısı, orijinal görüntüdeki nesne etiketiyle karşılaştırılır ve (a) noktasındaki etiket kalıntısı, orijinal etiketin %20'sinden az olduğundan, nesne etiketi olarak kaydedilmez. Görüntünün içeriğine göre, (a) noktasında etiket olarak kaydedilen nesne yoktur. Bu orantısal ayarlama, özellikle dilimleme işlemi sırasında görüntü kenarlarına denk gelen nesnelerin parçalanması sonucu oluşabilecek yanlış etiketlemeleri önlemek için kritik

öneme sahiptir. Nesnelerin dilimlenmiş bölümlerinin orijinal boyutlarını ve şekillerini doğru bir şekilde yansıtmaması durumunda ortaya çıkabilecek hatalı sınırlayıcı kutuların önüne geçilmesi, bu sürecin temel hedeflerinden biridir.

Kırpma boyutunun C, orijinal görüntünün genişliği W ile ve yüksekliği H ile temsil edildiğinde, genişlik (n_w) ve yükseklik (n_h) boyunca dilim sayıları Denklem 3.2'de görüldüğü gibi hesaplanmaktadır.

$$n_w = \frac{W}{C}, n_h = \frac{W}{C} \tag{3.2}$$

Genişlik (p_w) ve yükseklik (p_h) boyunca gerekli örtüşme miktarları Denklem 3.3'teki gibi hesaplanmaktadır.

$$p_w = \frac{(n_w x\ C)-W}{n_w},\ p_h = \frac{(n_h x\ C)-H}{n_h} \tag{3.3}$$

Doldurulan görüntü boyutları Denklem 3.4'te belirtildiği gibi hesaplanmaktadır.

$$W_p = W + p_w, H_p = H + p_h \tag{3.4}$$

ISA, görüntü özelliklerine ve modelin gerektirdiği giriş boyutlarına göre dilimleme boyutunu dinamik olarak ayarlamaktadır. Bu sayede, görüntülerin kenar noktalarına denk gelen nesnelerde bütünlük, nesne etiketlerindeki optimizasyon ile etiketlerin doğru kalmaları sağlanır. ISA algoritmasının dinamik yapısı, farklı görüntü ve model gereksinimlerine uyum sağlama yeteneği sunar ve bu da onu çeşitli uygulama senaryoları için son derece esnek ve güçlü bir araç haline getirir. Algoritmanın farklı koşullara uyum sağlayabilme yeteneği, görüntü dilimleme ve etiket optimizasyonu arasındaki ilişkiyi dikkatlice yöneterek, nesne tespitinin doğruluğunu ve hassasiyetini koruma açısından büyük önem taşımaktadır. ISA dilimleme algoritmasının adımları Algoritma 2 olarak Şekil 3.9'da görülmektedir.

Algoritma 2 : ISA algoritması adımları

```
image, boxes ← slicing.boxesFromData(imageName, labelName)
; /* veri kümesinden görüntüleri ve kutuları alın                          */

img_shape_w = img.shape[1] ;                    /* görüntü şekli genişliğini al */
img_shape_h = img.shape[0] ;                    /* görüntü şekli yüksekliğini al */
mod_w =img_shape_w % self.crop_size ;           /* kırpma için genişlik modunu al */
mod_h = img_shape_h % self.crop_size ;          /* mahsul için yükseklik modunu al */
dif_w = self.crop_size - mod_w ;                /* kırpma için genişlik farkı alın */
dif_h = self.crop_size - mod_h ;                /* mahsul için yükseklik farkı alın */
div_w = img_shape_w  / self.crop_size ;         /* kırpma için genişlik oranını al  */
div_h = img_shape_h  / self.crop_size ;         /* kırpma için yükseklik oranını elde edin */
rat_w = dif_w /div_w ;                          /* ürün için genişlik örtüşme miktarını al */
rat_h = dif_h /div_h ;                          /* ürün için yükseklik örtüşme miktarını al */
Kirpma boyutuna bagli olarak goruntu kirpma
for (0,  image_shape_w + dif_w, self.crop_size) do
    for (0,  image_shape_h + dif_h, self.crop_size) do
        if x == 0 and y == 0 then
            split_img = img[y : y + self.crop_size, x : x + self.crop_size]
            /* yeni görüntü alanını hesaplayın ve kaydedin                  */
            save_calculated_image_area()
    for (box in boxes) do
        cls, x_min, y_min, x_max, y_max = box[0], box[1], box[2], box[3], box[4] ; /* Yeni görüntü için
         etiketin gözden geçirilmesi                                        */
         box_center_x = ((new_x_min + new_x_max)/2)/split_img_shape_w
         box_center_y = ((new_y_min + new_y_max)/2)/split_img_shape_h
         box_w = (new_x_max − new_x_min)split_img_shape_w
         box_h = (new_y_max − new_y_min)split_img_shape_h
         self.save_img_txt(split_img, new_img_boxes) ;      /* yeni resmi ve etiketi kaydet */
```

Şekil 3.15. ISA dilimleme algoritmasının sözde kodu

3.3. SAM Etiket Kontrolü

Bilgisayarlı görü alanındaki çalışmalarda, nesne segmentasyonu bir görüntüdeki nesnelerin hassas bir şekilde tanımlanmasında ve konumlandırılmasında önemli bir rol oynamaktadır (Long ve ark, 2015). SAM, nesneleri sınırlarına göre doğru bir şekilde belirleyerek nesne segmentasyonunda üstün performans sağlamak için özel olarak tasarlanmış, son teknoloji derin öğrenme yaklaşımlarından biridir. Segmentasyon, bir görüntüyü çeşitli nesnelere veya ilgi alanlarına karşılık gelen belirgin bölütlere ayırmayı içermektedir. SAM, segmentasyon görevlerinde yüksek hassasiyet elde etmek için birincil mimari olarak CNN ile gelişmiş görüntü işleme teknikleri ve derin öğrenme çerçeveleri kullanır. SAM, giriş görüntüsünden hem düşük hem de yüksek seviyeli özellikleri çıkararak çalışmaktadır. Kenarlar, dokular ve renkler gibi düşük seviyeli özellikler temel bilgiler sağlarken, yüksek seviyeli özellikler nesne konumu, şekli ve anlamı gibi daha soyut nitelikleri yakalar (Ke ve ark, 2024). Bu özellikleri birleştiren model, bir görüntüdeki farklı nesneler arasında etkili bir şekilde ayrım yapabilir ve her nesnenin gerçek sınırlarıyla uyumlu doğru segmentasyona olanak tanır.

Nesne tespitinde, veri etiketlemesinin doğruluğu çok önemlidir. Etiketli nesnelerin bir görüntü içinde doğru şekilde temsil edilmesi, nesne tespit algoritmalarının performansını doğrudan etkilemektedir. Yanlış etiketler önemli hatalara yol açarak modelin doğruluğu önemli ölçüde azaltabilmektedir. Sonuç olarak, nesne etiketlerinin doğruluğunu kontrol etmek, model performansını optimize etmek için önemli bir adımdır.

Bu tez çalışmasında, nesne tespit doğruluğunu artırmak için dilimleme yöntemi kullanılmaktadır. Dilimlenen görüntülerde, sınırlayıcı kutuların içindeki nesnelerin her kutu için kontrol etmesi amacıyla sisteme SAM entegre edilmiştir. Bu işlem, nesne piksellerine 1 (veya 255) değeri atanırken, arka plan piksellerinin 0 değeriyle işaretlendiği ikili bir segmentasyon maskesi üretir. Segmentasyon maskesi, nesnenin sınırlayıcı kutu içinde doğru bir şekilde yakalanıp yakalanmadığını doğrulamak için kullanılmaktadır. Bir sınırlayıcı kutu içindeki nesnenin varlığını doğrulamak için SAM, segmentasyon maskesinin yoğunluğunu değerlendirir. Önceden belirlenmiş bir eşik değeri (t) nesnenin varlığını doğrulamak için gereken nesne piksellerinin oranını temsil eden, kabul edilebilir minimum maske yoğunluğu olarak işlev görür. Bu tez

çalışmasında etiketlerin SAM ile etiketlerinin kontrolünde t eşik değeri gerçek nesne piksel sayısının %20'si olarak belirlenmiştir.

Maskenin toplam yoğunluğu bu eşik değerin üstündeyse, nesne sınırlayıcı kutunun içinde mevcut kabul edilir. Maske yoğunluğu eşik değerin altındaysa, sınırlayıcı kutu yanlış kabul edilir ve otomatik olarak kaldırılır. Bu işlem, yalnızca iyi tanımlanmış nesneler içeren sınırlayıcı kutuların tutulmasını sağlayarak dilimlenerek elde edilmiş yeni veri kümesindeki nesne etiketlerinin doğruluğunu artırır. Ayrıca, SAM'in segmentasyon maskelerine dayalı olarak nesne sınırlarını ayarlayarak sınırlayıcı kutuları iyileştirme yeteneği, nesne tespit model performansının genel olarak iyileştirilmesine katkıda bulunmaktadır. SAM içerisinde, özellik çıkarmak için kullanılacak yapı Denklem 3.5'te görülmektedir.

$$f(x) = Conv(x) \tag{3.5}$$

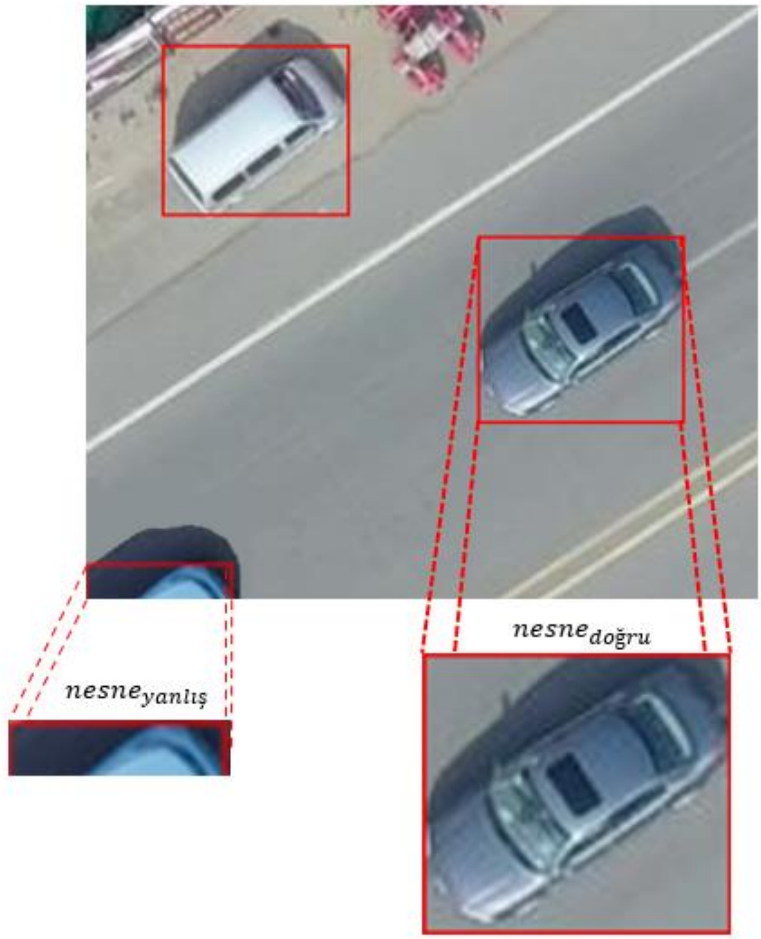
Burada Conv(x) evrişim işlemlerini, f(x) ise çıkarılan özellikleri temsil etmektedir. Segmentasyon maskeleri Denklem 3.6'te görüldüğü gibi hesaplanmaktadır.

$$maske = \sigma\,(Conv(f(x))) \tag{3.6}$$

Burada, σ sigmoid veya softmax aktivasyon fonksiyonlarını temsil etmektedir. Son olarak, nesneleri belirlenen (t) eşik değerine göre doğrulamak için Denklem 3.7 kullanılmaktadır.

$$nesne_{doğru} = \sum(maske) > t \tag{3.7}$$

Maskenin toplam yoğunluğu hesaplanır ve önceden tanımlanmış eşik değerini aşarsa, nesnenin varlığı doğrulanır. SAM'in sisteme entegre edilmesi, segmentasyon doğruluğunu önemli ölçüde artırarak nesne sınırlarının daha hassas bir şekilde belirlenmesini sağlar. Nesne varlığının bu otomatik doğrulaması, manuel müdahaleye olan ihtiyacı ortadan kaldırarak iş akışını hızlandırır. Ayrıca, bu yaklaşımın çok yönlülüğü, çeşitli görüntü türleri ve nesneler arasında etkili bir şekilde uygulanmasına olanak tanır. SAM'in sınırlayıcı kutular içindeki nesneleri doğrulamak için kullanıldığına dair örnek bir görsel Şekil 3.10'da verilmiştir.



Şekil 3.16. Dilimleme ile elde edilen görüntüde nesnelerin SAM ile kontrol edilmesi.

3.4. SRGAN Algoritması

Süper Çözünürlüklü Çekişmeli Üretici Ağlar (SRGAN), görüntüleri daha yüksek çözünürlüklere yükseltmek için güçlü bir yöntem olarak ortaya çıkmıştır ve geleneksel yöntemlere kıyasla görsel kalitede önemli iyileştirmeler sunmaktadır. SRGAN, GAN (Goodfellow ve ark, 2014) mimarilerine özgü rekabetçi bir şekilde eğitilen bir üretici ve bir ayırt edici ağ olmak üzere iki temel bileşenden oluşmaktadır (Ledig ve ark, 2017). Ayırt edici ağ, aldığı görüntülerin gerçek olma olasılığını değerlendirerek onlara 0 ile 1 arasında bir değer atar. Gerçek bir görüntü ideal olarak 1 olasılık alırken, sahte bir görüntüye 0 olasılık atanmalıdır. Ayırt edici ağ, tahmin edilen olasılıklar ile hedef değerler arasındaki fark olan kaybı en aza indirerek eğitilir. Eğitim sırasında, ayırt edici ağın parametreleri her iterasyonda güncellenerek bu hatalar 0'a yakınsamaya çalışılır.

Üretici ağ, ayırt edici ağın değerlendireceği kadar gerçekçi sahte görüntüler üretmeyi ve ideal olarak bunlara 1 olasılık vermeyi amaçlar. Ayırt edici, üretici tarafından üretilen bir görüntüye 0,6 olasılık atarsa, üreticinin hatası 1- 0,6 = 0,4 olur. Üretici ağ her iterasyonda bu hatayı en aza indirmeye çalışır. Eğitim ilerledikçe, ayırt edici ağ gerçek görüntüleri sahte olanlardan ayırt etmede daha iyi hale gelirken, üretici ağda giderek daha gerçekçi görüntüler oluşturmada gelişir. Eğitimin sonunda, üretici sistem gerçek görüntülerden neredeyse ayırt edilemeyen görüntüler üretebilir hale gelmektedir. Üretici ağ için amaç fonksiyonu iki ana bileşen içermektedir. Bu bileşenler; içerik kaybı ve karşıt kayıp olarak tanımlanır. Algısal kayıp olarak da bilinen içerik kaybı, üretilen yüksek çözünürlüklü görüntünün $G(I_{LR})$ özellik temsilleri ile her ikisi de önceden eğitilmiş bir VGG ağından çıkarılan temel gerçek yüksek çözünürlüklü görüntü I_{HR} arasındaki ortalama karesel hata olarak tanımlanır. İçerik kaybı Denklem 3.8'de görüldüğü gibi hesaplanmaktadır.

$$L_{icerik}(G) = \frac{1}{N}\sum_{i=1}^{N} ||\emptyset(I_{HR}^{i}) - \emptyset\left(G(I_{LR}^{i})\right)||_2^2 \quad (3.8)$$

Burada ∅ önceden eğitilmiş VGG ağından elde edilen bir özellik haritasını temsil eder ve N eğitim setindeki görüntü sayısıdır.

İçerik kaybına ek olarak, üretici ağ, ayırt edici ağa göre gerçek yüksek çözünürlüklü görüntülerden ayırt edilemeyen görüntüler üretmeye teşvik eden karşıt kaybını optimize eder. Karşıt kayıp Denklem 3.9'de görüldüğü gibi hesaplanmaktadır.

$$L_{karsıt}(G) = -\frac{1}{N}\sum_{i=1}^{N} logD\left(G(I_{LR}^{i})\right) \quad (3.9)$$

Burada D ayırıcı ağdır. Üretken ağ için toplam kayıp fonksiyonu, içerik kaybını ve karşıt kaybı birleştirir ve ikisini dengelemek için bir λ faktörü ile ağırlıklandırılır. λ faktörü Denklem 3.10'da gösterilmektedir.

$$L_G = L_{icerik} + \lambda_{karsıt} \quad (3.10)$$

Ayırt edici ağ D, görüntüleri gerçek (temel gerçek yüksek çözünürlüklü görüntüler) veya sahte (G tarafından oluşturulan görüntüler) olarak sınıflandıran bir evrişimli sinir ağıdır. Ayırt edici ağın amacı, gerçek ve sahte görüntüleri doğru şekilde sınıflandırma

olasılığını en üst düzeye çıkarmaktır. Ayırt edici ağ için kayıp fonksiyonu Denklem 3.11'da görüldüğü gibi hesaplanmaktadır.

$$L_D = -\frac{1}{N}\sum_{i=1}^{N}\left[logD\left(I_{HR}^{i}\right) + \log\left(1 - \mathrm{D}\left(\mathrm{G}\left(I_{LR}^{i}\right)\right)\right)\right] \quad (3.11)$$

Bu kayıp fonksiyonu, ayırt edici ağı gerçek görüntülere daha yüksek olasılıklar ve üretilen görüntülere daha düşük olasılıklar atamaya teşvik eder.

SRGAN, G üreteci ve D ayrıt edicinin dönüşümlü olarak güncellendiği çekişmeli bir şekilde eğitilir. Eğitim prosedürü tipik olarak her iki ağın da başlatılmasını içerir. Her bir düşük çözünürlüklü ve yüksek çözünürlüklü görüntü çifti grubu için ayırt edici, ilk olarak L_D kaybı hesaplanarak ve ağırlıkları ayarlanarak güncellenir. Ardından, üretici ağ hem içerik kaybını hem de karşıt kaybı içeren birleşik kayıp L_G hesaplanarak ve ağırlıkları buna göre ayarlanarak güncellenir. Bu işlem yakınsama veya belirli bir döngü sayısına ulaşılana kadar tekrarlanır. SRGAN bir derin öğrenme modeli kullanmaktadır. Modelin yapısı aşağıdaki gibidir:

- **ResidualDenseBlock_5C Sınıfı**: Bu sınıf 5 katmanlı bir yoğun artık bloğu tanımlar. Her katman bir evrişimli katmana (Conv2D) ve bir Leaky ReLU aktivasyon fonksiyonuna sahiptir. Evrişimsel katmanlar, bir filtreyi giriş görüntüsüne kaydırarak belirli özellikleri çıkarmak için kullanılır.
- **RRDB Sınıfı**: Bu sınıf, üç ResidualDenseBlock_5C bloğundan oluşan bir artık yoğun blok kümesini tanımlar.
- **RRDBNet Sınıfı**: Bu sınıf, giriş ve çıkış katmanları, RRDB blokları ve çeşitli evrişim katmanlardan oluşan eksiksiz bir süper çözünürlük modeli tanımlar.

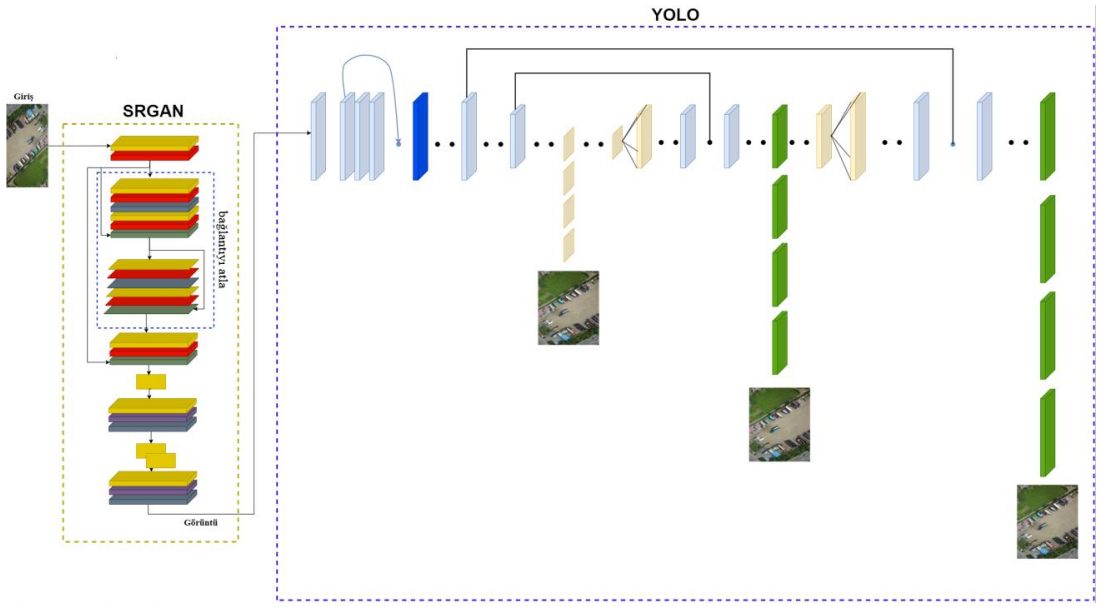
SRGAN uygulanan örnek bir görüntü Şekil 3.11'de görülmektedir.

Şekil 3.17. SRGAN katmanından geçtikten sonra görüntüdeki çözünürlük değişimine bir örnek.

İki görüntünün karşılaştırılmasıyla elde edilen Tepe Sinyal-Gürültü Oranı (Peak Signal-to-Noise Ratio (PSNR) ve Yapısal Benzerlik Endeksi Ölçümü (Structural Similarity Index Measure (SSIM) değerleri, bir görüntünün referans görüntüye göre kalitesi hakkında önemli bilgiler sağlar. PSNR, genellikle desibel (dB) cinsinden piksel bazlı hataların yoğunluğunu ölçerek iki görüntü arasındaki genel farkı niceliksel olarak belirler (Wang ve ark, 2004). Daha yüksek bir PSNR değeri, görüntülerin daha az bozulma veya gürültü ile daha benzer olduğunu gösterir. Ancak SSIM, basit piksel farklılıklarının ötesine geçer ve yapısal bilgi, parlaklık ve kontrasttaki değişiklikleri göz önünde bulundurarak görüntüler arasındaki algısal benzerliği değerlendirir. 1'e yakın bir SSIM puanı, iki görüntünün görsel yapıları açısından oldukça benzer olduğunu gösterir (Wang ve ark, 2004). Bu ölçümler birlikte hem piksel düzeyindeki hem de algısal farklılıkları niceliksel olarak belirleyerek sıkıştırma veya gürültü giderme gibi görüntü işleme algoritmalarının performansını değerlendirmek için yaygın olarak kullanılır. Şekil 3.10'da örnek olarak verilen görseldeki orijinal görüntü ile SRGAN uygulanan görüntü için PSNR ve SSIM değerleri incelendi. Görüntüler arasındaki PSNR değeri 25.1680 ve SSIM değeri 0.8694 olarak elde edildi. Bu durum SRGAN uygulanan görüntünün orijinal görüntüye göre kaliteli olduğunu göstermektedir.

3.5. SROD Mimarisi

SROD yapısı, SRGAN ve YOLO nesne algılama algoritmasının entegrasyonu, nesne algılama performansını artırmak için teknik olarak optimize edilmiştir. Mimari, SRGAN bloğu tarafından üretilen yüksek çözünürlüklü görüntülerden yararlanarak bunları YOLO ağı için girdi olarak kullanır ve bu da daha hassas ve güvenilir nesne algılama sonuçları elde edilmesini sağlar. SRGAN bloğu YOLO mimarisine bir ön işleme katmanı olarak dahil edildiği yapı Şekil 3.12'de görülmektedir. Bu, görüntünün nesne tespiti için YOLO'ya aktarılmadan önce SRGAN aracılığıyla çözünürlüğünün geliştirmesi anlamına gelmektedir. SRGAN'ın bu yapıdaki rolü kritiktir; çözünürlüğü artırarak görüntü kalitesini iyileştirir. Bu yapı, YOLO'nun üzerinde çalışabileceği daha ince ayrıntılar sağlar. Böylece, daha düşük çözünürlüklü görüntülerde atlanabilen nesnelerin tespit edilmesi sağlanır.



Şekil 3.18. SROD mimarisinin genel yapısının blok diyagramı.

YOLO algoritması, nesne tespitine yönelik kendine özgü yaklaşımı nedeniyle bu görev için özellikle uygundur. Bir görüntüyü işlemek için birden fazla aşama gerektiren geleneksel nesne algılama yöntemlerinin aksine, YOLO tüm görüntüyü tek bir geçişte işleyerek bunu basitleştirir. YOLO bunu, görüntüyü bir hücre ızgarasına bölerek ve her hücre için sınırlayıcı kutuları ve sınıf olasılıklarını doğrudan tahmin ederek başarır. Bu ızgara tabanlı yaklaşım, YOLO'nun aynı görüntüdeki birden fazla nesneyi aynı anda tespit etmesini sağlar.

YOLO'nun nesne algılama performansı, SRGAN tarafından üretilen yüksek çözünürlüklü görüntülerle birleştirildiğinde, önemli ölçüde artar. SRGAN tarafından sağlanan ayrıntılı görüntüler, YOLO'nun tam potansiyelini kullanabilmesi, normalde düşük kaliteli görüntülerde ayırt edilemeyecek nesneleri, doğru bir şekilde tespit edip sınıflandırabilmesini sağlar. Dolayısıyla bu entegrasyon yalnızca tespit doğruluğunu artırmakla kalmaz, aynı zamanda nesne tespit sürecinin genel güvenilirliğini ve sağlamlığını da geliştirir.

4. DENEYSEL ÇALIŞMALAR

Önerilen sistemin kullanıldığı deneysel çalışmalarda, mevcut yöntemlere kıyasla daha iyi sonuçlar elde edilmiştir. Karşılaştırmalı analizler, önerilen sistemin geleneksel nesne tespit algoritmalarındaki görüntülerin yeniden boyutlandırılmasından kaynaklanan piksel kayıplarını etkili bir şekilde azalttığını ve veri etiketlerini optimize ederek model performansını artırdığını göstermektedir. Sonuçların ayrıntılı analizi, piksel kayıplarının ve etiket yanlışlıklarının model performansı üzerinde doğrudan bir etkiye sahip olduğunu göstermektedir. Önerilen mimarinin etkinliğini değerlendirmek için YOLOv5, YOLOv7, YOLOv8 ve YOLOv9 olmak üzere dört farklı YOLO algoritma mimarisi kullanılarak deneyler yapılmıştır. Deneylerde kullanılacak YOLO mimarileri nesne tespitinde en yeni ve en yaygın kullanımlarına göre belirlenmiştir. Her mimarinin belirli nesne algılama görevlerinde üstün olma potansiyeli vardır. Önerilen dilimleme algoritması, SAM ile etiket kontrolü ve SRGAN-YOLO entegrasyonu ile elde edilen performans iyileştirmeleri deneysel çalışmalar ile değerlendirildi. Önerilen algoritmanın ve genel sistem yapısının performans üzerindeki etkisini açıkça göstermek için, aynı veri kümelerini üzerinde her yapı ayrı ayrı incelendi. Bu karşılaştırma, temel bir performans seviyesi oluşturmak ve önerilen sistemin parçalarının model performansları üzerindeki katkılarını daha net bir şekilde vurgulamak için oldukça önemlidir.

Bu tez çalışmasında, farklı YOLO mimarileri arasındaki karşılaştırmaların hem dengeli hem de objektif olmasını amaçlandı. Bunu başarmak için eğitim süreci boyunca tüm modellerde standart hiperparametreler kullanıldı. Hiper parametrelerde tutarlılığı koruyarak, tüm mimariler için eşit bir çalışma ortamı sağlandı. Belirlenen hiperparametrelerden bir kısmı Tablo 4.1'de gösterilmektedir. Tablo 4.1'de verilmeyen hiperparametreler YOLO algoritmasının default değerdeki hiperparametreleri olarak belirlenmiştir.

Tablo 4.1. Nesne tespiti için yapılan YOLO eğitimlerindeki hiperparametreler

Hiperparametre	Belirlenen Değer	İşlevi
lr0	0.001	Başlangıç öğrenme oranı, modelin eğitimi sırasında her adımdaki güncelleme büyüklüğünü belirler.
lrf	0.0028	Öğrenme oranı faktörü, öğrenme oranının eğitimin ilerleyen aşamalarında nasıl azalacağını belirler.
weight_decay	0.00015	Ağırlık zayıflatma katsayısı, modelin overfitting yapmasını önlemek için ağın ağırlıklarını küçültür.
translate	0.01	Veri artırma sırasında görüntülerin yatay ve dikey yönde kaydırılma miktarını belirtir.
scale	0.5	Veri artırma sırasında görüntülerin kesme dönüşümü uygulanarak bozulma miktarını belirtir.
flipud	0.1	Görüntülerin dikey olarak çevrilme olasılığıdır.
fliplr	0.1	Görüntülerin yatay olarak çevrilme olasılığıdır.
mosaic	0.0	Mosaic veri artırma tekniğinin kullanılma olasılığıdır. Bu teknik, birkaç görüntüyü birleştirerek veri artırma yapar.
mixup	0.0	Mixup veri artırma tekniğinin kullanılma olasılığıdır. Bu teknik, iki görüntüyü karıştırarak yeni bir görüntü oluşturur.
optimizer	Adam	Eğitim sırasında kayıp fonksiyonunu en aza indirmek için kullanılan yinelemeli bir optimizasyon algoritmasıdır

Bu standartlaştırılmış yaklaşım ile önerilen dilimleme algoritması, SAM ile etiket optimizasyonunun performansa etkisi ve SROD yaklaşımı ile orijinal YOLO mimarileri arasındaki karşılaştırmaların güvenilirliğini ve geçerliliği artırıldı. xView ve VisDrone veri kümeleri üzerinde yapılan deneysel değerlendirmelerimizin sonuçları sırasıyla Tablo 4.2 ve Tablo 4.3'te gösterildi. Bu tablolar, standart YOLO mimarileri ile SROD ile geliştirilmiş versiyonlar arasındaki performans farklılıklarını vurgulayarak test sonuçlarının ayrıntılı bir karşılaştırmasını sunmaktadır.

Tablo 4.2. xView veri seti deneysel test sonuçları.

Model	Hassasiyet	Özgünlük	mAP_0.5	mAP_05:0.95	F1-Puanı
ISA-YOLOv5	59.9	44.5	45.6	30.1	51.1
ISA-SAM-YOLOv5	60.5	46.3	48.1	33.1	52.5
ISA-SAM-SROD-YOLOv5	**64.3**	**52.5**	**54.4**	**46.1**	**57.8**
ISA-YOLOv7	42.4	25.2	28.3	17.4	31.6
ISA-SAM-YOLOv7	48.2	27.9	30.3	21.1	35.3
ISA-SAM-SROD-YOLOv7	52.9	32.5	35.7	23.6	40.3
ISA-YOLOv8	53.7	31.2	33.5	22.6	39.5
ISA-SAM-YOLOv8	54.0	35.3	36.2	24.1	42.7
ISA-SAM-SROD-YOLOv8	61.4	39.8	41.2	28.5	48.2
ISA-YOLOv9	45.3	27.1	27.9	17.1	33.9
ISA-SAM-YOLOv9	46.3	28.6	29.8	20.0	35.4
ISA-SAM-SROD-YOLOv9	50.5	30.9	33.6	21.9	38.3

Tablo 4.3. VisDrone veri seti deneysel test sonuçları.

Model	Hassasiyet	Özgünlük	mAP_0.5	mAP_05:0.95	F1-Puanı
ISA-YOLOv5	65.5	55.3	53.1	39.9	59.7
ISA-SAM-YOLOv5	67.3	56.4	55.8	43.2	61.7
ISA-SAM-SROD-YOLOv5	71.9	65.5	67.7	49.1	68.3
ISA-YOLOv7	68.1	64.8	67.4	44.0	66.4
ISA-SAM-YOLOv7	76.3	68.5	71.8	47.9	72.4
ISA-SAM-SROD-YOLOv7	81.2	70.6	75.6	53.5	75.9
ISA-YOLOv8	74.9	64.9	66.5	52.8	69.7
ISA-SAM-YOLOv8	80.1	61.4	69.6	55.4	70.6
ISA-SAM-SROD-YOLOv8	**80.4**	**74.4**	**77.5**	**63.8**	**77.9**
ISA-YOLOv9	59.3	45.8	47.8	35.3	51.9
ISA-SAM-YOLOv9	67.1	54.8	59.4	42.7	60.7
ISA-SAM-SROD-YOLOv9	71.6	63.2	65.1	48.7	67.2

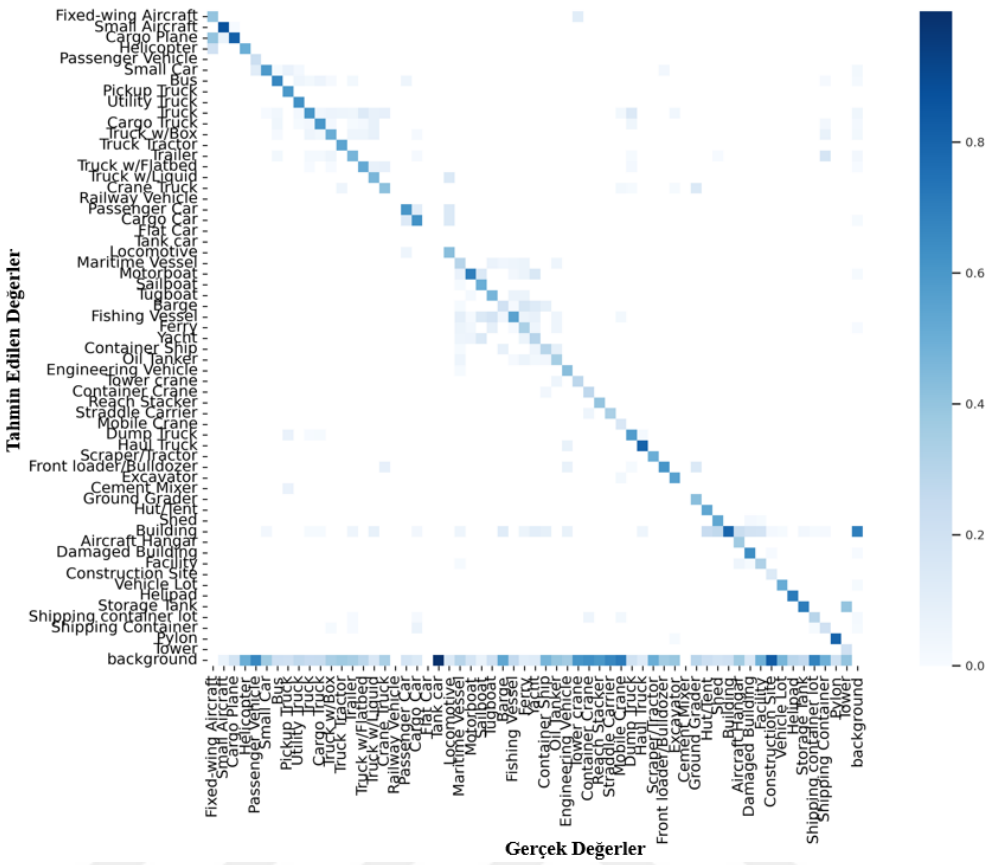
Bu tez çalışmasında elde edilen test sonuçları dikkatle incelendiğinde, önerdiğimiz ön işleme tekniklerinin ve hibrit bir model olan SROD mimarisi ile birleştirildiğinde oldukça iyi sonuçlar verdiği gözlemlendi. Özellikle, ön işleme yöntemlerinin SROD mimarisi ile entegrasyonu, modelin nesne tespit doğruluğunu önemli ölçüde arttırdı. Önerilen sistem ile özellikle kompleks yapısı ile bilinen xView ve VisDrone veri

kümeleri üzerinde elde edilen sonuçlar, sistemin nesne tespitindeki gücünü göstermektedir. Önerilen sistemin başarısı, yüksek zorluk derecesine sahip veri kümelerinin karmaşıklıklarına etkili bir şekilde uyum sağlamasına olanak tanıyan yapısal esnekliğine ve güçlü modelleme yeteneklerine dayanmaktadır.

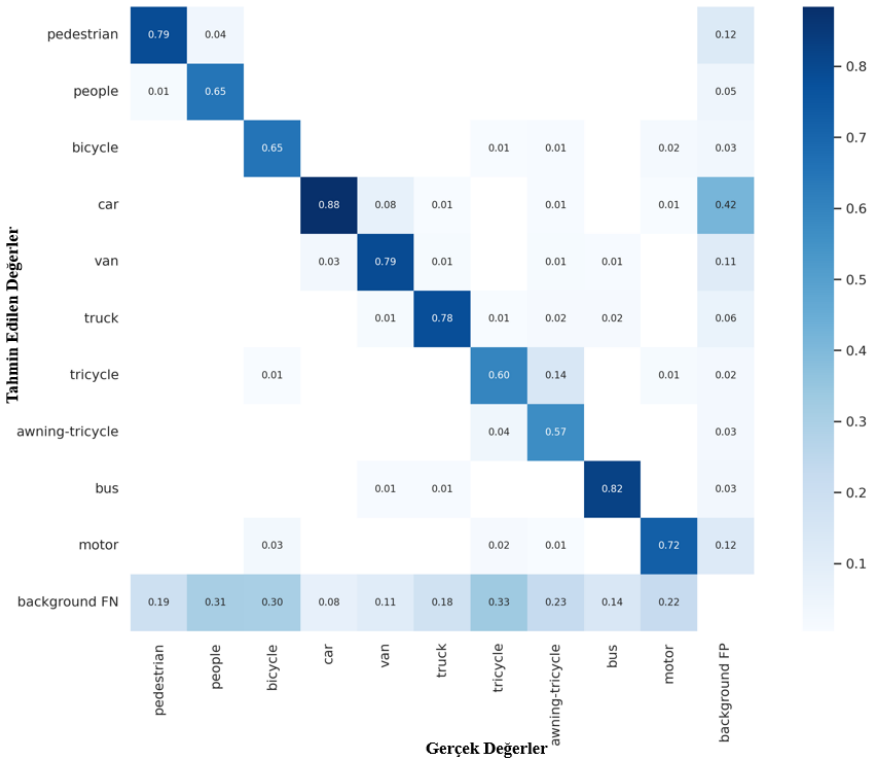
xView veri kümesi üzerinde yapılan deneysel değerlendirmelerde, önerilen sistem ile YOLOv5 algoritması beraberliğinde daha yüksek mAP sonucu elde edilmiştir. VisDrone veri kümesi üzerinde yapılan deneylerde, YOLOv8 algoritmasının eğitimin bir parçası olduğu sistemin daha yüksek mAP sonucu elde ettiği görüldü.

Sistem performansını daha anlamlı değerlendirmek için, modellerin sınıflandırma doğruluğunun kapsamlı bir özetini gösteren karmaşıklık matrisleri incelendi. Karışıklık matrisi, her sınıf için doğru ve yanlış tahminlerin oranını vurgulayarak tahmin edilen ve gerçek sınıf etiketlerinin ayrıntılı bir dökümünü sunduğundan, sınıflandırma modellerinin performansını değerlendirmede çok önemli bir araçtır. Bu nedenle, her sınıf için Ortalama Kesinlik (AP) incelenerek daha sık yanlış sınıflandırılan veya diğerleriyle karıştırılan sınıfların belirlenmesine olanak sağlanmaktadır. Bu analiz, modelin güçlü yönleri ve geliştirilmesi gereken alanlar hakkında değerli bilgiler sağladı.

xView ve VisDrone veri kümelerindeki test sonuçlarında, en iyi performans sırasıyla YOLOv5 ve YOLOv8 mollerinin entegre edildiği sistemlerde elde edildiğinden bu iki sistemin karmaşıklık matrisi sırasıyla Şekil 4.1 ve Şekil 4.2'de gösterilmiştir. Bu matrislerde, köşegen elemanlar her sınıf için doğru tahminlerin sayısını temsil eder ve modelin her bir kategoriyi sınıflandırmadaki doğruluğunu gösterir. Köşegen dışındaki elemanlar, modelin verileri yanlış sınıflandırdığı durumları yansıtan hem yanlış pozitifleri hem de yanlış negatifleri kapsayan yanlış tahminlerin sayısına karşılık gelir. Karmaşıklık matrislerinin analizi, model hiper parametrelerinin belirlenmesinde önemli bir referans noktası oluşturdu.

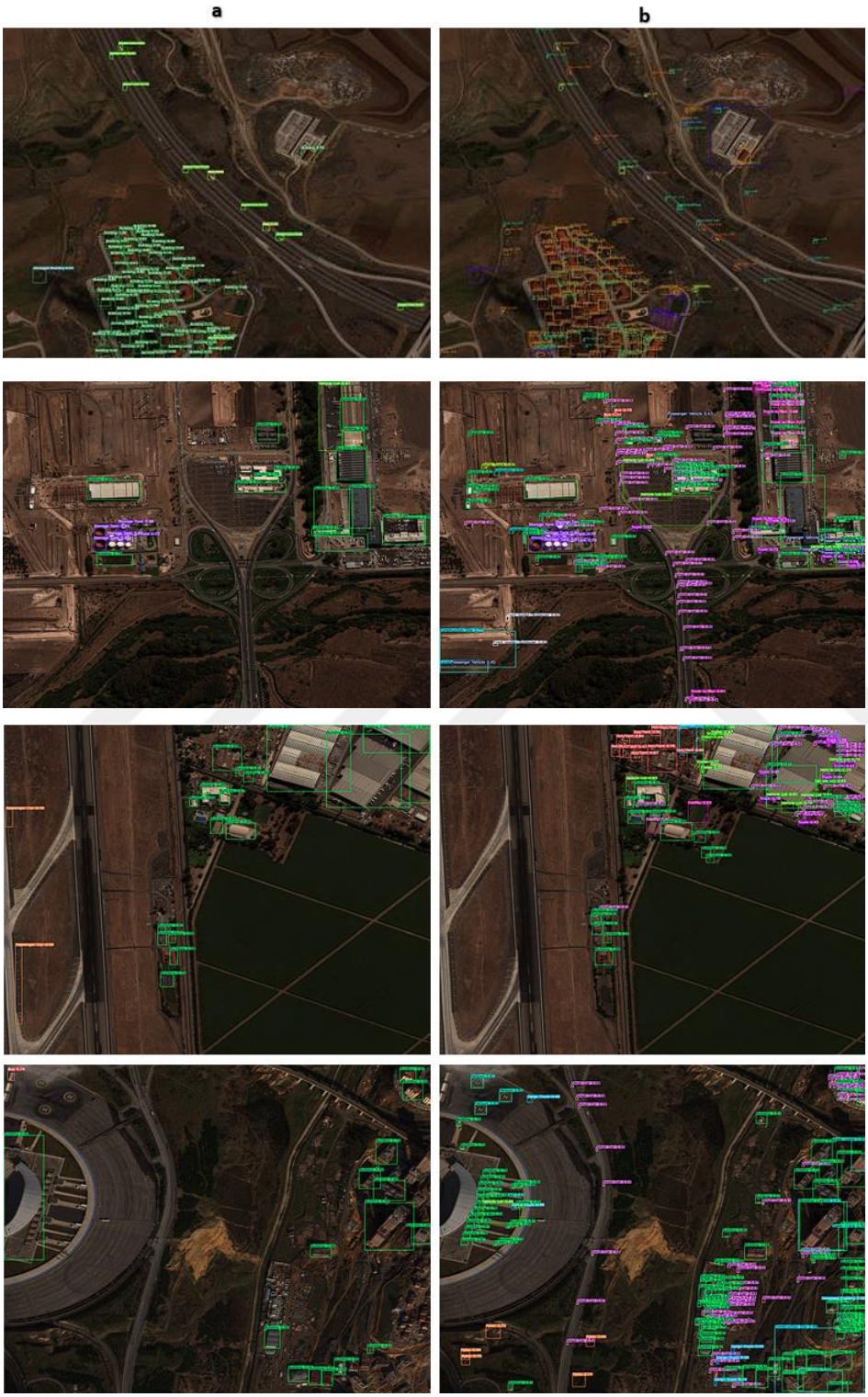


Şekil 4.1. xView veri kümesinde YOLOv5 için önerilen sistemin karmaşıklık matrisi.

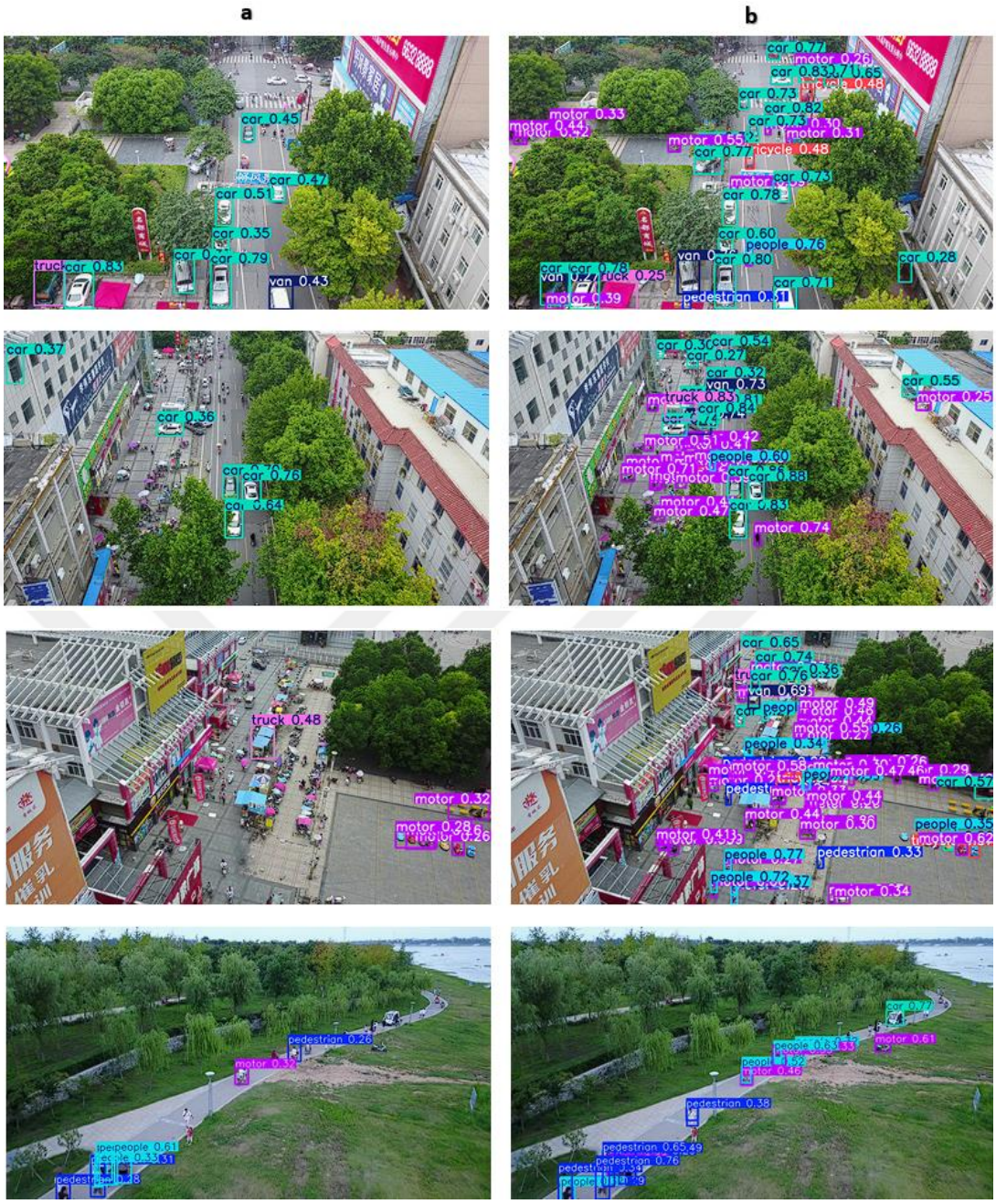


Şekil 4.2. VisDrone veri kümesinde YOLOv8 için önerilen sistemin karmaşıklık matrisi.

XView ve VisDrone veri kümeleri üzerindeki önerilen sistemin test sonuçlarından örnek görseller ile önerdiğimiz sistemin uygulanmadığı durumdaki test sonuçlarından görsellere örnekler Şekil 4.3 ve Şekil 4.4'te yer verilmiştir.



Şekil 4.3. a) xView'de sistem olmadan yapılan test sonucu b) xView'de önerilen sistemin test sonucu



Şekil 4.4. a) VisDrone'da sistem olmadan yapılan test sonucu b) VisDrone'da önerilen sistemin test sonucu

Şekilde gösterilen karşılaştırmalar, önerilen sistemin nesne tespitinde önemli performans arttırımını açıkça göstermektedir. Sistem, özellikle nesne algılamada zorlu bir görev olan küçük nesnelerin tespitinde önemli bir gelişme sağladı. Bu gelişme, önerilen sistemin etkinliğini ve sağlamlığını, özellikle de karmaşık ortamlardaki küçük nesneleri doğru bir şekilde tanımlama ve sınıflandırma becerisini vurgulamaktadır.

Bu sonuçlar, önerilen dilimleme algoritmasının SROD çerçevesiyle entegre edilmesinin avantajlarını vurgulamakta ve sistemin bu gelişmiş teknikleri içermeyen geleneksel yaklaşımlara kıyasla üstün performans sağladığını doğrulamaktadır.

5. SONUÇ VE ÖNERİLER

xView ve VisDrone veri kümeleri, çok çeşitli nesne sınıflarına sahip olmaları, karmaşık sahneleri bulundurmaları ve nesne sayılarının çok olmasından dolayı nesne algılama araştırmalarında yaygın olarak kullanılmaktadır. Ancak, bu veri kümeleri nesne algılama algoritmaları için önemli zorluklara sahiptir. Önemli zorluklardan birtanesi, modellerin genellikle daha az örnek içeren sınıfları, daha çok örneğe sahip sınıflara göre etkili bir şekilde öğrenmede başarısız olmasıdır. Derin öğrenme modelleri, çoğunluk sınıflarına aşırı uyum sağlamaya eğilimlimindedir. Bu da azınlık sınıflarına ait nesneleri algılarken optimum olmayan performansa yol açar. Bu sorunun üstesinden gelmek için eğitim sürecinde uygun veri hazırlama ve dikkatli model seçimi, nesne algılama görevlerinde başarılı sonuçlar elde etmek için oldukça önemlidir.

Bu tezde sunulan test sonuçları, önerilen ön işleme yöntemlerinin, yeni bir hibrit model olan SROD mimarisiyle birlikte etkinliğini gösterdi. SROD mimarisi veri kümelerine uygulanan ISA dilimleme algoritması ve etiketlerin SAM modeliyle kontrol edilmesi gibi ön işleme yöntemleri ile birleştirildiğinde, modelin doğruluğunu ve genel performansını önemli ölçüde geliştirdi. Önerilen sistemin başarısının en önemli göstergesi, karmaşıklıkları ve nesne algılama görevlerindeki zorluklarıyla bilinen xView ve VisDrone gibi zorlu veri kümeleri üzerinde yapılan deneysel çalışmalarda, literatürdeki çalışmalara göre daha yüksek doğruluk elde etmesidir. Bu veri kümelerinin sahip olduğu zorluklara rağmen, önerilen sistemin her adımı ayrı ayrı incelendiğinde, tutarlı bir şekilde nesne tespit doğruluğunu geliştirdiği görüldü. Buda başka çalışmalar için uygulanabilirliğini ve etkinliğini gösterdi.

Deneysel çalışmalardaki nesne tespit algoritmalarının önemli zorluklarından birisi de incelenen veri setlerinin küçük nesne boyutlarına sahip sınıfları bulundurmasıdır. Yüksek çözünürlüklü görüntülerde bulunan küçük nesnelerin tespiti, nesne tespit modelleri için oldukça zorlayıcı bir durumdur. Önerilen yöntemin öne çıkan özelliklerinden bir taneside özellikle ISA dilimleme algoritmasının uygulanması ile piksel kaybının önlenmesi ve küçük nesnelerin modeller tarafında öğrenilmesine

önemli katkı sunmuştur. Bu da önerilen yöntemin, yüksek çözünürlüklü görüntülerde küçük nesnelerin tespiti için yapılacak çalışmalarda uygulanabilirliğini gösterdi.

Bu tezde elde edilen deneysel sonuçların, xView ve VisDrone veri kümelerini kullanan önceki çalışmalarda bildirilenlerle ayrıntılı bir karşılaştırması Tablo 5.1'de verilmiştir. Akyon ve ark. (2022) xViex ve Visdrone veri kümelerindeki küçük nesne algılama problemini çözmek için SAHI yaklaşımını önerdiler. Önermiş oldukları SAHI yöntemi, test aşamasında dilimlemeli bir nesne tespit yapısına sahiptir. Önerdikleri bu yöntem ile xView veri kümesi için altmış sınıf için %23,6 VisDrone veri kümesi için 66,4 mAP elde ettiler. Akshatha ve ark. (2023) çok ölçekli özellik birleşimi ve daha büyük giriş boyutlarının kullanımıyla küçük nesnelerin tespit performansını iyleştirmek için yapmış oldukları çalışmada VisDrone veri kümesi için 58,3 mAP elde ettiler. Muzammul ve ark. (2024) insansız hava araçları kullanılarak yapılan görüntü analizlerini iyileştirmek amacıyla RT-DETR-X modelini, SAHI tekniği ile birleştirmişlerdir. Bu yaklaşımla, özellikle VisDrone veri seti 73,7 mAP elde ettiler. Olamofe ve ark. (2022) nesnenin renk kanallarını değiştirerek, nesne görüntülerine saf gürültüler ekleyerek ve kontrast iyileştirme işlemleri uygulayarak veri artırma yaparak YOLOv3 modelinin xViex veri seti üzerinde altı sınıf için yapmış oldukları çalışmada 44,7 mAP ve 65.,9 mAR elde ettiler. Bu karşılaştırma, önerilen sistemin üstün performansını göstererek, karmaşık ve çeşitli ortamlarda nesne algılamayı iyileştirmede ön işleme yöntemlerinin ve SROD mimarisinin etkinliğini gösterdi.

Tablo 5.1. Önerilen sistemin performans karşılaştırması (* ortalama özgünlük (mAR)).

Çalışma	Veri Seti	Sınıf Sayısı	mAP_0.5	mAP_0.5:0.95
Akyon ve ark. (2022)	VisDrone	10	66.4	42.2
Akshatha ve ark. (2023)	VisDrone	10	58.3	
Muzammul ve ark. (2024)	VisDrone	10	73.7	54.8
ISA-SAM-SROD-YOLOv8	**VisDrone**	**10**	**77.5**	**63.8**
Akyon ve ark. (2022)	xView	60	23.6	14.9
Olamofe ve ark. (2022)	xView	6	44.7	*(mAR) 65.9
ISA-SAM-SROD-YOLOv5	**xView**	**60**	**54.4**	**46.1**

SRGAN'ın YOLO nesne tespiti algoritması ile birleştirilmesi, genel hesaplama maliyetinde artışa neden oldu. Ancak, ayrıntılı performans değerlendirmeleri ve deneysel bulgular, hesaplama maliyetindeki bu artışın göz ardı edilebileceğini gösterdi. Hibrit yapıdaki SROD mimarisinin yoğun katman yapısına rağmen,

SRGAN'ın süper çözünürlük yetenekleri ile YOLO'nun nesne tespit doğruluğunun birleşimi, özellikle yüksek çözünürlüklü ve karmaşık görüntülerle çalışırken önemli performans iyileştirmeleri sağladı. Bu iyileştirmeler, küçük veya ayrıntılı nesneleri algılamanın doğru sonuçlar elde etmek için kritik olduğu durumlara dikkat çekmektedir.

Mevcut çalışmada, yüksek çözünürlüklü görüntülerdeki nesnelerin tespit doğruluğuna odaklanıldı. Gelecekteki çalışmalarda, hesaplama maliyetini azaltırken performansı daha da artırmak amaçlanmaktadır. Algılama doğruluğundan ödün vermeden model sıkıştırma teknikleri veya daha verimli sinir ağı mimarileri kullanmak gibi hesaplama yükünü azaltma yöntemlerinin araştırılması gerektiği düşülmektedir.

KAYNAKLAR

Akshatha K.R., A.K., K., Satish Shenoy B., K., P. P., Dhareshwar, C. V., & Johnson, D. G. (2023). Manipal-UAV person detection dataset: A step towards benchmarking dataset and algorithms for small object detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 77-89.

Akyon, F. C., Altinuc, S. O., & Temizel, A. (2022). Slicing Aided Hyper Inference and Fine-Tuning for Small Object Detection. *2022 IEEE International Conference on Image Processing (ICIP)* (s. 966-970). IEEE.

Bochkovskiy, A. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv preprint arXiv:2004.*, 10934.

Bosquet, B., Cores, D., Seidenari, L., Brea, V. M., Mucientes, M., & Bimbo, A. D. (2023). A full data augmentation pipeline for small object detection based on generative adversarial networks. *Pattern Recognition*, 108998.

Chen, W., Li, Y., Tian, Z., & Zhang, F. (2023). 2D and 3D object detection algorithms from images: A Survey. *Array*, 100305.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning* , 273-297.

Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (s. 886-893). IEEE.

Dasiopoulou, S., Mezaris, V., Kompatsiaris, I., Papastathis, V.-K., & Strintzis, M. G. (2005). Knowledge-assisted semantic video object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 1210-1224.

Du, D., Zhu, P., Wen, L., Bian, X., Lin, H., Hu, Q., & ... & Zhang, L. (2019). VisDrone-DET2019: The Vision Meets Drone Object Detection in Image Challenge Results. *In Proceedings of the IEEE/CVF international conference on computer vision workshops*, (s. 0-0).

Du, D., Zhu, P., Wen, L., Bian, X., Lin, H., Hu, Q., . . . Zhang, L. (2019). VisDrone-DET2019: The Vision Meets Drone Object Detection in Image Challenge Results. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (s. 0-0). IEEE Xplore.

Etten, A. V. (2018). You Only Look Twice: Rapid Multi-Scale Object Detection In Satellite Imagery. *arXiv preprint arXiv:1805.09512.*

Fang, Y., Yang, S., Wang, S., Ge, Y., Shan, Y., & Wang, X. (2023). Unleashing Vanilla Vision Transformer with Masked Image Modeling for Object Detection. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (s. 6244-6253). IEEE.

Girshick, R. (215). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (s. 1440-1448). IEEE Xplore.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition* (s. 580-587). IEEE Xplore.

Gong, X.-Y., Su, H., Xu, D., Zhang, Z.-T., Shen, F., & Yang, H.-B. (2018). An Overview of Contour Detection Approaches. *International Journal of Automation and Computing*, 656-672.

Gonzalez, R. C. (2009). *Digital image processing.* Pearson education india.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., . . . Bengio, Y. (2014). Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 2672-2680.

Guo, W., Yang, W., Zhang, H., & Hua, G. (2018). Geospatial object detection in high resolution satellite images based on multi-scale convolutional neural network. *Remote Sensing*, 131.

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (s. 2961-2969). IEEE.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (s. 770-778). IEEE Xplore.

Jocher, G., Stoken, A., Borovec, J., Changyu, L., Hogan, A., Diaconu, L., . . . al., e. (2020). ultralytics/yolov5: v3. 1-bug fixes and performance improvements. *Zenodo*, Available online: https://ui.adsabs.harvard.edu/abs/2020zndo...3983579J/abstract (accessed on 29 June 2024).

Karaçeper, S. (2018). Dijital teknoloji ve grafik tasarımda yenilikler. *İstanbul Aydın Üniversitesi Güzel Sanatlar Fakültesi Dergisi*, 73-83.

Ke, L., Ye, M., Danelljan, M., Liu, Y., Tai, Y.-W., Tang, C.-K., & Yu, F. (2024). Segment Anything in High Quality. *Advances in Neural Information Processing Systems*, 36.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., . . . Girshick, R. (2023). Segment Anything. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (s. 4015-4026). IEEE Xplore.

Lam, D., Kuzma, R., McGee, K., Dooley, S., Laielli, M., Klaric, M., . . . McCord, B. (2018). xView: Objects in Context in Overhead Imagery. *Computer Vision and Pattern Recognition*.

Lam, D., Kuzma, R., McGee, K., Dooley, S., Laielli, M., Klaric, M., . . . McCord, B. (2018). xView: Objects in Context in Overhead Imagery. *arXiv:1802.07856* .

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE* , 2278 - 2324.

Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., . . . Shi, W. (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (s. 4681-4690). IEEE Xplore.

Lin, J., Lin, H., & Wang, F. (2022). STPM_SAHI: A Small-Target Forest Fire Detection Model Based on Swin Transformer and Slicing Aided Hyper Inference. *Forests*, 1603.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. *14th European Conference Computer Vision–ECCV 2016* (s. 21-37). Springer International Publishing.

Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., & Zhang, L. (2022). Image-Adaptive YOLO for Object Detection in Adverse Weather Conditions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 1792-1800.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (s. 3431-3440). IEEE.

Lowe, D. (1999). Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision* (s. 1150-1157). IEEE.

Mansour, A., Hussein, W. M., & Said, E. (2019). Small Objects Detection in Satellite Images Using Deep Learning. *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)* (s. 86-91). Cairo: IEEE.

Meier, B. (2019). *Python GUI Programming Cookbook: Develop functional and responsive user interfaces with tkinter and PyQt5.* Packt Publishing Ltd.

Ming, Q., Miao, L., Zhou, Z., Song, J., Dong, Y., & Yang, X. (2023). Task interleaving and orientation estimation for high-precision oriented object detection in aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 241-255.

Mufti, N., & Shah, S. A. (2021). Automatic Number Plate Recognition:A Detailed Survey of Relevant Algorithms. *Sensors*, 3028.

Muzammul, M., Algarni, A., Ghadi, Y. Y., & Assam, M. (2024). Enhancing UAV Aerial Image Analysis: Integrating Advanced SAHI Techniques With Real-Time Detection Models on the VisDrone Dataset. *IEEE Access*, 21621 - 21633.

Olamofe, J., Dong, X., Qian, L., & Shields, E. (2022). Performance Evaluation of Data Augmentation for Object Detection in XView Dataset. *2022 International Conference on Intelligent Data Science Technologies and Applications (IDSTA)* (s. 26-33). IEEE.

Pereira, A., Santos, C., Aguiar, M., Welfer, D., Dias, M., & Ribeiro, M. (2023). Improved Detection of Fundus Lesions Using YOLOR-CSP Architecture and Slicing Aided Hyper Inference. *IEEE Latin America Transactions*, 806-813.

Pesaresi, M., & Benediktsson, J. A. (2001). A new approach for the morphological segmentation of high-resolution satellite imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 309-320.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning* , 81-106.

Rani, N. (2017). Image processing techniques: a review. *Journal on Today's Ideas-Tomorrow's Technologies 5.1*, 40-49.

Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *Computer Vision and Pattern Recognition*, arXiv preprint arXiv:1804.02767.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (s. 779-788). IEEE.

Reis, D., Kupec, J., Hong, J., & Daoudi, A. (2023). Real-Time Flying Object Detection with YOLOv8. *arXiv preprint arXiv:2305.09972*.

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1137-1149.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference* (s. 234-241). Springer International Publishing.

Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Computer Science*, 420.

Shen, Y., Liu, D., Chen, J., Wang, Z., Wang, Z., & Zhang, Q. (2023). On-Board Multi-Class Geospatial Object Detection Based on Convolutional Neural Network for High Resolution Remote Sensing Images. *Remote Sensing*, 3963.

Shen, Y., Liu, D., Zhang, F., & Zhang, Q. (2022). Fast and accurate multi-class geospatial object detection with large-size remote sensing imagery using CNN and Truncated NMS. *ISPRS Journal of Photogrammetry and Remote Sensing*, 235-249.

Shermeyer, J., & Etten, A. V. (2019). The Effects of Super-Resolution on Object Detection Performance in Satellite Imagery. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, (s. 0-0). IEEE.

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.*, 1556.

Telçeken, M., Akgun, D., Kacar, S., & Bingol, B. (2024). A New Approach for Super Resolution Object Detection Using an Image Slicing Algorithm and the Segment Anything Model. *Sensors*, 4526.

Tian, Z., Huang, J., Yang, Y., & Nie, W. (2023). KCFS-YOLOv5: A High-Precision Detection Method for Object Detection in Aerial Remote Sensing Images. *Applied Sciences*, 649.

Tresson, P., Tixier, P., Puech, W., & Carval, D. (2019). Insect interaction analysis based on object detection and CNN. *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)* (s. 1-6). Kuala Lumpur: IEEE.

Uijlings, J. R., Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective Search for Object Recognition. *International Journal of Computer Vision*, 154–171.

Wan, D., Lu, R., Wang, S., Shen, S., Xu, T., & Lang, X. (2023). YOLO-HR: Improved YOLOv5 for Object Detection in High-Resolution Optical Remote Sensing Images. *Remote Sensing* , 614.

Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2023). YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (s. 7464-7475). IEEE Xplore.

Wang, C.-Y., Yeh, I.-H., & Liao, H.-Y. M. (2024). YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv 2024, arXiv:2402.13616.*

Wang, X., He, N., Hong, C., Wang, Q., & Chen, M. (2023). Improved YOLOX-X based UAV aerial photography object detection algorithm. *Image and Vision Computing*, 104697.

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 600-612.

Wu, Z.-Z., Wang, X.-F., Zou, L., Xu, L.-X., Li, X.-L., & Weise, T. (2021). Hierarchical object detection for very high-resolution satellite images. *Applied Soft Computing*, 107885.

Yang, A., Pan, F., Saragadam, V., Dao, D., Hui, Z., Chang, J.-H. R., & Sankaranarayanan, A. C. (2021). SliceNets -- A Scalable Approach for Object Detection in 3D CT Scans. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (s. 335-344). IEEE.

Yang, C., Huang, Z., & Wang, N. (2022). QueryDet: Cascaded Sparse Query for Accelerating High-Resolution Small Object Detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (s. 13668-13677). IEEE.

Yin, Q., Hu, Q., Zhang, F., Wang, Y., Lin, Z., An, W., & Guo, Y. (2021). Detecting and Tracking Small and Dense Moving Objects in Satellite Videos: A Benchmark. *IEEE Transactions on Geoscience and Remote Sensing* , 1-18.

You only look twice: Rapid multi-scale object detection in satellite imagery. (2018). *arXiv preprint arXiv:*, 1805.09512.

Zhang, H., Hao, C., Song, W., Jiang, B., & Li, B. (2023). Adaptive Slicing-Aided Hyper Inference for Small Object Detection in High-Resolution Remote Sensing Images. *Remote Sensing*, 1249.

Zhang, J., Lei, J., Xie, W., Fang, Z., Li, Y., & Du, Q. (2023). SuperYOLO: Super Resolution Assisted Object Detection in Multimodal Remote Sensing Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 1-15.

Zhang, W., Sun, X., Fu, K., Wang, C., & Wang, H. (2014). Object Detection in High-Resolution Remote Sensing Images Using Rotation Invariant Parts Based Model. *IEEE Geoscience and Remote Sensing Letters*, 74-78.

Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2019). Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, 3212-3232.

Zhu, H., Wei, H., Li, B., Yuan, X., & Kehtarnavaz, N. (2020). Real-Time Moving Object Detection in High-Resolution Video Sensing. *Sensors*, 3591.

ÖZGEÇMİŞ

Ad-Soyad : Muhammed TELÇEKEN

ÖĞRENİM DURUMU:

- **Lisans** **:** 2020, Mersin Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
- **Yüksek lisans** **:** 2022, İskenderun Teknik Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği
- **Doktora** **:** 2024, Sakarya Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Mühendisliği

MESLEKİ DENEYİM:

- 2022- Devam ediyor Sakarya Uygulamalı Bilimler Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliğinde Araştırma Görevlisi olarak çalışıyor
- 2022- Devam ediyor Sakarya Uygulamalı Bilimler Üniversitesi Teknoloji Yarışmaları Koordinatörlüğünde çalışıyor

TEZDEN TÜRETİLEN ESERLER:

- Telçeken, M., Akgun, D., Kacar, S., & Bingol, B. (2024). A New Approach for Super Resolution Object Detection Using an Image Slicing Algorithm and the Segment Anything Model. *Sensors (Basel, Switzerland), 24*(14).
- Telceken, M., Okuyar, M., Akgun, D., Kacar, S., & Vural, M. S. (2024). A new data label conversion algorithm for YOLO segmentation of medical images. *The European Physical Journal Special Topics*, 1-10.

DİĞER ESERLER: