

KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

GÖRÜNTÜDE KAMUFLE OLMUŞ NESNE TESPİTİ VE TANINMASI:
KELEBEK ÖRNEĞİ

DOKTORA TEZİ

Erkan BAYRAM

MART 2025

TRABZON



KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**GÖRÜNTÜDE KAMUFLE OLMUŞ NESNE TESPİTİ VE TANINMASI:
KELEBEK ÖRNEĞİ**

Erkan BAYRAM

**Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"DOKTOR (BİLGİSAYAR MÜHENDİSLİĞİ)"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.**

Tezin Enstitüye Verildiği Tarih : 14 / 02 / 2025

Tezin Savunma Tarihi : 17 / 03 / 2025

Tez Danışmanı : Doç. Dr. Vasif NABİYEY

Trabzon 2025

ÖNSÖZ

Kamuflaj görüntülerde yer alan kamuflle olmuş nesneler arka planla yakın doku, desen ve renk özelliklerine sahip olduklarından, tespit edilmeleri oldukça zorlu bir problemidir. Zayıf sınırlara ve arka plana benzer desenlere sahip olduklarından mevcut ikili bölütleme çözümleri kamuflle olmuş nesnenin tespit edilmesi problemiyle kolayca başa çıkamamaktadır. “Görüntüde Kamuflle Olmuş Nesne Tespiti ve Tanınması: Kelebek Örneği” isimli bu tez çalışmasında arka planla yüksek derecede benzerlik gösteren kamuflle olmuş nesnelerin tespit edilmesi amaçlanmaktadır.

Bu tez çalışmasında danışmanlığımı üstlenen, her türlü destek ve katkılarıyla çalışmamı yönlendiren çok değerli danışman hocam Sayın Doç. Dr. Vasif NABIYEV’ e teşekkürlerimi sunarım.

Bu tez çalışması süresince değerli görüş ve katkılarından yararlandığım Sayın Prof. Dr. Önder AYDEMİR’ e, Sayın Dr. Öğr. Üyesi Murat AYKUT’ a, Sayın Doç. Dr. Ferhat BOZKURT’ a ve Sayın Prof. Dr. Umut ORHAN’a teşekkür ederim.

Son olarak öğrenim hayatım boyunca benden maddi manevi desteklerini esirgemeyen sevgili anneme ve babama müteşekkir olduğumu belirtirim. Hayatıma anlam kazandıran ve bu süreçte sabırla hep bana destek olan sevgili eşim Canan BAYRAM’ a ve çocuklarım Taha BAYRAM ile Yusuf Emir BAYRAM’ a sonsuz teşekkürlerimi sunarım.

Bu tez çalışmasının bundan sonraki çalışmalara katkı sağlamasını temenni ederim.

Erkan BAYRAM

Trabzon 2025

TEZ ETİK BEYANNAMESİ

Doktora Tezi olarak sunduğum “Görüntüde Kamufle Olmuş Nesne Tespiti ve Tanınması: Kelebek Örneği” başlıklı bu çalışmayı baştan sona kadar danışmanım Doç. Dr. Vasif NABİYEV’ in sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuvarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 17/03/2025

Erkan BAYRAM

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VIII
SUMMARY	IIX
ŞEKİLLER DİZİNİ.....	X
TABLolar DİZİNİ.....	XIII
SEMBOLLER VE KISALTMALAR DİZİNİ	XIV
1. GENEL BİLGİLER	1
1.1. Giriş.....	1
1.2. Tezin Katkısı	5
1.3. Tezin Amacı ve Genel Yapısı	6
1.4. Literatür Taraması	7
1.4.1. Klasik Yöntemler (Geleneksel Görüntü İşleme Teknikleri).....	8
1.4.2. Derin Öğrenme Tabanlı Yöntemler	12
1.4.3. Hibrit ve Çoklu Modalite Yöntemler	15
1.4.4. Literatürdeki Boşluklar	16
1.4.5. Literatürde Kullanılan Önemli Kamufraj Veri Setleri	18
1.4.6. COD İçin Kullanılan Özellik Çıkarma Yöntemleri	23
1.5. COD İçin Literatürde Kullanılan Performans Ölçütleri	30
1.5.1. Birleşim Üzerindeki Kesişim (Intersection over Union, IoU)	32
1.5.2. Kesinlik (Precision).....	33
1.5.3. Duyarlılık (Recall)	34
1.5.4. F1-Score	34
1.5.5. Doğruluk (Accuracy)	35
1.5.6. F-Ölçütü (F-measure).....	36
1.5.7. S-Ölçütü(S-measure).....	37
1.5.8. Ortalama Mutlak Hata (MAE)	37
1.5.9. E-Ölçütü (E-measure)	38
1.5.10. Yapısal Benzerlik Endeks Ölçüsü (SSIM).....	39

1.5.11. Zar Benzerlik Katsayısı (DSC)	39
1.6. Yapay Zeka Kavramı.....	40
1.7. Makine Öğrenmesi	42
1.7.1. Makine Öğrenmesi Türleri.....	43
1.8. Derin Öğrenme	45
1.8.1. Yapay Sinir Ağları	45
1.8.2. Evrişimsel Sinir Ağları (CNN)	48
2. YAPILAN ÇALIŞMALAR.....	59
2.1. Kamufraj Görüntülerde Kelebek Nesnelerinin Bölütlenmesi ve Tanınması İçin Hibrit Bir Yöntem Geliştirilmesi.....	59
2.1.1. Yeni Kamufraj Veri Seti Oluşturulması.....	60
2.1.2. Önerilen Yöntem.....	69
2.2. Kamufraj Görüntülerin Sınıflandırılması İçin Transfer Öğrenme ve Özellik Birleştirme Yaklaşımı.....	93
2.2.1. Çalışmada Kullanılan Veri Seti.....	94
2.2.2. Önerilen Yöntem.....	95
2.3. Klasik Makine Öğrenmesi ve Evrişimsel Sinir Ağı Yöntemlerini Kullanarak Kamufraj Görüntülerin Sınıflandırılması İçin Karşılaştırmalı Bir Çalışma.....	98
2.3.1. Önerilen Yöntem.....	99
2.4. LBP, HOG ve Klasik Makine Öğrenmesi Yöntemleri Kullanarak Kamufraj Görüntülerin Sınıflandırılması	102
2.4.1. Önerilen Yöntem.....	103
2.5. Önceden Eğitilmiş Evrişimsel Sinir Ağı Kullanarak Kamufraj Görüntülerinin Sınıflandırılması	105
2.5.1. Önerilen Yöntem.....	105
2.6. Yerel İkilli Örüntüler (LBP) Kullanarak Kamufraj Görüntülerin Sınıflandırılması	107
2.6.1. Önerilen Yöntem.....	107
2.7. Kamufraj Görüntülerde Öklid ve Mahalanobis Uzaklık Hesabına Göre K-means Kümeleme Algoritması Kullanarak Görüntü Bölütleme	109
2.8. Yerel İkili Desenlerle Özellik Çıkarımına Dayalı Kelebek Kamufraj Görüntülerinin Sınıflandırılması	111
3. BULGULAR VE TARTIŞMA	113
3.1. “Kamufraj Görüntülerde Kelebek Nesnelerinin Bölütlenmesi ve Tanınması İçin Hibrit Bir Yöntem Geliştirilmesi” Çalışmasından Elde Edilen Bulgular.....	113

3.2. “Kamuflaj Görüntülerinin Sınıflandırılması İçin Transfer Öğrenme ve Özellik Birleştirme Yaklaşımı” Çalışmasından Elde Edilen Bulgular.....	126
3.3. “Klasik Makine Öğrenmesi ve Evrişimsel Sinir Ağı Yöntemlerini Kullanarak Kamuflaj Görüntülerinin Sınıflandırılması İçin Karşılaştırmalı Bir Çalışma” Çalışmasından Elde Edilen Bulgular.....	132
3.4. “Kamuflaj Görüntülerinin LBP, HOG ve Klasik Makine Öğrenmesi Yöntemleri Kullanılarak Sınıflandırılması” Çalışmasından Elde Edilen Bulgular	135
3.5. “Önceden Eğitilmiş Evrişimsel Sinir Ağı Kullanarak Kamuflaj Görüntülerinin Sınıflandırılması” Çalışmasından Elde Edilen Bulgular	135
3.6. “LBP Yöntemi Kullanarak Kamuflaj Görüntülerinin Sınıflandırılması” Çalışmasından Elde Edilen Bulgular.....	136
3.7. “Kamuflaj Görüntülerde Öklid, Mahalanobis ve K-ortalamlar Kümeleme Algoritması Kullanılarak Görüntü Bölütlemesi” Çalışmasından Elde Edilen Bulgular.....	137
3.8. “Yerel İkili Desenlerle Özellik Çıkarımına Dayalı Kelebek Kamuflaj Görüntülerinin Sınıflandırılması” Çalışmasından Elde Edilen Bulgular	137
4. SONUÇLAR.....	139
5. ÖNERİLER.....	141
6. KAYNAKLAR	142
ÖZGEÇMİŞ	

Doktora Tezi

ÖZET

GÖRÜNTÜDE KAMUFLE OLMUŞ NESNE TESPİTİ ve TANINMASI: KELEBEK ÖRNEĞİ

Erkan BAYRAM

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Doç. Dr. Vasif NABİYEV
2025, 158 Sayfa

Kamuflaj görüntülerde yer alan kamuflaj olmuş nesneler arka planla yakın doku, desen ve renk özelliklerine sahip olduklarından, tespit edilmeleri oldukça zordur. Zayıf sınırlara ve arka plana benzer desenlere sahip olduklarından mevcut ikili bölütleme çözümleri kamuflaj olmuş nesnenin tespit edilmesi problemiyle kolayca başa çıkamamaktadır. Kamuflajlı nesne tespiti (COD) arka planla yüksek derecede benzerlik gösteren nesnelerin tespit edilmesini amaçlamaktadır. Bu tez çalışmasında ERVA 1.0 orijinal kamuflaj veri seti oluşturularak kullanılmıştır. COD problemi için bu tez çalışmasında bölütleme ve nesne tanıma olmak üzere iki aşamalı bir çözüm sunulmaktadır. ERVA 1.0 veri seti üzerinde yer alan tüm test görüntülerinin doku özellikleri bölütleme işlemi için Gabor Filtre kullanılarak çıkarılmıştır. Çıkarılan bu özellikler K-means algoritması yardımıyla kümelenecek orijinal görüntü doku özelliklerine göre farklı bölgelere ayrılmıştır. Bu bölgelerdeki nesnelerin tespit edilmesi için LBP ve Öklid uzaklık hesabı kullanılmıştır. Daha sonra nesnenin türünün tahmin edilmesi için derin öğrenme tekniklerinden önceden eğitilmiş modeller kullanılmıştır. Çalışmada bölütleme başarısı Structural Similarity yöntemiyle %87.89, Dice Similarity Coefficient yöntemi ile %83.64 başarı oranı hesaplanmıştır. Bölütleme sonrası elde edilen nesne türünün tespit edilmesi için derin öğrenme önceden eğitilmiş modeller kullanılmıştır. Artırılmamış veri ile Deney 1 ve veri artırma yöntemi uygulanarak artırılmış veri ile Deney 2 yapılmıştır. Deney 1 için en yüksek başarı oranı InceptionResNetV2 modeli ile %92.29 ve Deney2 için en yüksek başarı oranı ise DenseNet121 modeli ile %94.81 olarak elde edilmiştir.

Anahtar Kelimeler: Bölütleme, Kamuflaj Görüntüler, Kamuflajlı Nesne Tanıma, Doku Analizi, Kamuflaj Görüntü Sınıflandırma

PhD. Thesis

SUMMARY

DETECTION and RECOGNITION CAMOUFLAGED OBJECTS IN IMAGES: THE CASE OF BUTTERFLY

Erkan BAYRAM

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering Graduate Program
Supervisor: Assoc. Prof. Vasif NABİYEYEV
2025, 158 Pages

Camouflaged objects in camouflage images are difficult to detect because they have similar texture, pattern and colour characteristics to the background. Since they have weak boundaries and patterns similar to the background, existing binary segmentation solutions cannot easily cope with the problem of camouflaged object detection. Camouflaged object detection (COD) aims to detect objects with a high degree of similarity to the background. In this thesis, ERVA 1.0 original camouflage dataset is created and used. In this thesis, a two-stage solution is presented for the COD problem: segmentation and object recognition. The texture features of all test images on the ERVA 1.0 dataset are extracted using Gabor Filter for segmentation. These extracted features are clustered with the K-means algorithm and the original image is divided into different regions according to the texture features. LBP and Euclidean distance calculation were used to detect the objects in these regions. Then, pretrained models from deep learning techniques were used to predict the type of the object. In the study, the segmentation success rate was 87.89% with the Structural Similarity method and 83.64% with the Dice Similarity Coefficient method. Deep learning pretrained models were used to determine the object type obtained after segmentation. Experiment 1 was performed with un augmented data and Experiment 2 was performed with augmented data by applying data augmentation method. The highest success rate for Experiment 1 was 92.29% with the InceptionResNetV2 model and the highest success rate for Experiment 2 was 94.81% with the DenseNet121 model.

Key Words: Segmentation, Camouflage Images, Camouflaged Object Detection, Texture analysis, Camouflage Image Classification

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1. CAMO kamuflaj veri setinden örnek görüntüler.....	19
Şekil 2. CAMO veri setinde kamufle edilmiş nesnelere ait kategoriler	19
Şekil 3. CAMO veri setindeki zorlu özniteliklere ve öznitelik dağılımına örnekler.....	20
Şekil 4. COD10K kamuflaj veri setindeki sınıflara ait örnek görüntüler	21
Şekil 5. NC4K kamuflaj veri setine ait örnek görüntüler	22
Şekil 6. Kullanılan yöntemin tespit ettiği ile olması gereken alanların doğruluğunun karşılaştırılması	32
Şekil 7. IoU için Birleşim Alanına bölünen Kesişim Alanının gösterilmesi	33
Şekil 8. Farklı IoU eşik değerlerinde nesne algılama örnekleri.....	33
Şekil 9. Yapay zekâ, makine öğrenmesi ve derin öğrenmenin kronolojisi.....	41
Şekil 10. Denetimli makine öğrenme genel akış şeması.....	43
Şekil 11. Yapay sinir ağında tek bir nöronun genel modellenmesi	46
Şekil 12. Beyin nöronu şekilsel gösterimi	46
Şekil 13. (a) Yapay bir nöronun, (b) çok katmanlı bir yapay sinir ağının mimarisi	47
Şekil 14. Evrimsel sinir ağları (CNN) temel katmanları.....	49
Şekil 15. Evrim işlemi için örnek bir uygulama.....	51
Şekil 16. Maksimum havuzlama işlemi için örnek bir uygulama.....	52
Şekil 17. Ortalama havuzlama işlemi için örnek bir uygulama	53
Şekil 18. Tam bağlantılı katman örneği.....	54
Şekil 19. Kamuflajlık derecesi düşük ve artırılmış örnek görüntüler	61
Şekil 20. ERVA 1.0 kamuflaj veri setinde yer alan türlere ait sayısal analiz gösterimi	62
Şekil 21. ERVA 1.0 kamuflaj veri setindeki kamufle olmuş nesnelerin sınırlarının belirlenmesi için örnek görüntüler	63
Şekil 22. ERVA 1.0 kamuflaj veri setindeki türlere ait örnek görüntüler	64
Şekil 23. ERVA 1.0 kamuflaj veri setindeki görüntülere ait örnek Görsel Karmaşıklık Metrik değerlerinin gösterilmesi.....	65
Şekil 24. ERVA 1.0 kamuflaj veri setindeki kamufle olmuş nesne ve arka plan görüntüsüne ait örnek Renk Histogramı oluşturma ve görüntüler arası karşılaştırma yapılması	66

Şekil 25.	ERVA 1.0 kamuflaj veri setindeki görüntülere veri artırma işleminin kamuflaj zorluk derecesine katkısının hesaplanması.....	68
Şekil 26.	Çalışmanın genel metodolojik adımlarının akış şeması	69
Şekil 27.	LBP operatörünün hesaplanması için örnek bir örnek uygulama.....	71
Şekil 28.	Gabor Filtre uygulanmış örnek kamuflaj görüntüler	75
Şekil 29.	K-means kümeleme algoritması genel akış diyagramı	76
Şekil 30.	Çalışmanın doku bölütleme aşamaları genel gösterimi	77
Şekil 31.	(a), (e), (i) Orijinal görüntü; (b), (f), (j) bölütlü bölgelerin etiketlenmesi; (c), (g), (k) bölütlü bölgelerin ayrıştırılmış görüntüleri; (d), (h), (l) tespit edilen kelebek nesne görüntüleri	78
Şekil 32.	(a), (d), (g) Orijinal görüntüler; (b), (e), (h) bölütlenmiş bölgelerin etiketlenmesi; (c), (f), (i) tespit edilen kelebek nesne görüntüleri	79
Şekil 33.	CAMO veri seti üzerinde (a), Orijinal test girdi görüntüleri; (b), bölütlenmiş bölgelerin etiketlenmesi; (c) tespit edilen kamuflaj olmuş nesne görüntüleri; (d) morfolojik işlemler sonrası tespit edilen kamuflaj olmuş nesne görüntüleri	80
Şekil 34.	(a), (d) orijinal görüntüler; (b), (e) bölütleme sonrası tespit edilen kelebek nesne görüntüleri; (c), (f) Morfolojik işlemler sonucunda elde edilen kelebek nesne görüntüleri	83
Şekil 35.	Evrişimsel sinir ağının temel mimarisi	84
Şekil 36.	VGG16 model mimarisi	86
Şekil 37.	VGG19 model mimarisi	86
Şekil 38.	Inception-V3 model mimarisi.....	87
Şekil 39.	Xception model mimarisi	88
Şekil 40.	Inception-ResNet-v2 model mimarisi.....	89
Şekil 41.	DenseNet121 model mimarisi	90
Şekil 42.	DenseNet201 model mimarisi	91
Şekil 43.	ResNet-50 model mimarisi	92
Şekil 44.	AlexNet model mimarisi.....	93
Şekil 45.	COD10K kamuflaj veri setinden seçilip çalışmada kullanılan türlere ait görüntüler.....	94
Şekil 46.	Çalışmada önerilen yeni modelin sistem tasarımının genel akış şeması	95
Şekil 47.	Çalışmada önerilen modelin genel mimari gösterimi	100
Şekil 48.	Çalışmada kullanılan sınıflara ait örnek kamuflaj görüntüler	104
Şekil 49.	Çalışmanın genel metodolojik adımlarının şeması.....	104
Şekil 50.	Çalışmada kullanılan sınıflara ait örnek kamuflaj görüntüler	106
Şekil 51.	Çalışmanın genel metodolojik adımlarının şeması	107

Şekil 52.	CAMO-V.1.0 veri setinden seçilen türlere ait örnek görüntüler	108
Şekil 53.	LBP tabanlı çalışmanın genel akış şeması	109
Şekil 54.	(a),(d),(g) Orijinal görünüş; (b),(e),(h) Etiketli görüntü; (c),(f),(i) Öklid hesaplamalı K-means çıktı görüntü.	110
Şekil 55.	(a),(d),(g) Orijinal görünüş; (b),(e),(h) Etiketli görüntü; (c),(f),(i) Mahalanobis hesaplamalı K-means çıktı görüntü.....	111
Şekil 56.	Veri集中的 türlerine ait örnek kamuflej görüntüler	112
Şekil 57.	Camo veri setinde yer alan türlere ait sayısal veriler.....	114
Şekil 58.	Çalışmada kullanılan ön eğitimli modellere göre Deney (1)'den elde edilen karmaşıklık matrisleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d) Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50.....	117
Şekil 59.	Çalışmada kullanılan ön eğitimli modeller ile Deney(1)'den elde edilmiş Eğitim / Test Doğruluk ve Eğitim / Test Kayıp Grafikleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d), Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50	118
Şekil 60.	Çalışmada kullanılan ön eğitimli modellere göre Deney (2)'den elde edilen karmaşıklık matrisleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d) Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50.....	119
Şekil 61.	Çalışmada kullanılan ön eğitimli modeller ile Deney(2)'den elde edilmiş Eğitim / Test Doğruluk ve Eğitim / Test Kayıp Grafikleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d), Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50	120
Şekil 62.	Çalışmada kullanılan ön eğitimli modellere göre elde edilen karmaşıklık matrisleri: (a) VGG19, (b) VGG16, (c) Xception, (d) EfficientNetV2, (e) InceptionV3, (f) MobileNet, (g) DenseNet169, (h) DenseNet121, (i) DenseNet201	127
Şekil 63.	Çalışmada kullanılan ön eğitimli modellerden elde edilmiş Eğitim / Test Doğruluk ve Eğitim / Test Kayıp Grafikleri: (a) VGG19, (b) VGG16, (c) Xception, (d) EfficientNetV2, (e) InceptionV3, (f) MobileNet, (g) DenseNet169, (h) DenseNet121, (i) DenseNet201.....	128
Şekil 64.	Çalışmada tüm birleştirme tabanlı önceden eğitilmiş modellerin Doğruluk grafikleri ve Kayıp grafikleri: (a)InceptionV3 + DenseNet201, (b)MobileNet + DenseNet169, (c)InceptionV3 +EfficientNetV2 + DenseNet201, (d)DenseNet201 + MobileNet	131
Şekil 65.	Çalışmadaki tüm birleştirme tabanlı önceden eğitilmiş sınıflandırma modelleri için karmaşıklık matrisleri: (a) InceptionV3 + DenseNet201, (b)MobileNet + DenseNet169, (c)InceptionV3 + EfficientNetV2 +DenseNet201, (d)DenseNet201 + MobileNet	131

TABLolar DİZİNİ

	<u>Sayfa No</u>
Tablo 1. Aktivasyon fonksiyonlarının karşılaştırılması	58
Tablo 2. Düzenli ve düzensiz örüntü örnekleri	72
Tablo 3. Bölütleme başarı oranlarının gösterilmesi	115
Tablo 4. Karmaşıklık matrislerine göre modellerden elde edilen en yüksek başarı sonucunun performans karşılaştırması.	116
Tablo 5. Deney (1) için önceden eğitilmiş modellerin performans karşılaştırması	122
Tablo 6. Deney (2) için önceden eğitilmiş modellerin performans karşılaştırması	122
Tablo 7. Kamuflej veri setleri üzerinde yapılan literatürdeki benzer son teknoloji çalışmalarının performans karşılaştırmaları	123
Tablo 8. COD10K Veri Seti Üzerinde Çalışmada Kullanılan Önceden Eğitilmiş Modellerin Performans Karşılaştırması	129
Tablo 9. COD10K Veri Setinde Önceden Eğitilmiş Modellerin Farklı Kombinasyonlarda Birleştirilmesinin Performans Karşılaştırması	130
Tablo 10. CAMO Veri Setinde Önceden Eğitilmiş Modeller ile Bu Modellerin Farklı Kombinasyonlarda Birleştirilmesinin Performans Karşılaştırması	132
Tablo 11. Deney (1) için Klasik Makine Öğrenmesi ve CNN modellerinin performans karşılaştırması.....	133
Tablo 12. Deney (2) için Klasik Makine Öğrenmesi ve CNN modellerinin performans karşılaştırması.....	134
Tablo 13. Çalışmada kullanılan CML, LBP ve HOG yöntemleri ile elde edilen sınıflandırma başarı performansları.....	135
Tablo 14. Deney (1) için kullanılan CNN modellerinin sınıflandırma performanslarının karşılaştırılması	136
Tablo 15. Deney (2) için kullanılan CNN modellerinin sınıflandırma performanslarının karşılaştırılması	136
Tablo 16. CML yöntemlerinin kullanılmasıyla elde edilen sınıflandırma başarı oranları.....	137
Tablo 17. Çalışmada kullanılan CML yöntemlerinin sınıflandırma performansları gösterimi	138

SEMBOLLER VE KISALTMALAR DİZİNİ

COD	: Kamuflejli nesne tespiti
LBP	: Yerel ikili örüntüler
ANN	: Yapay sinir ağları
KNN	: K-en yakın komşu
SVM	: Destek vektör makineleri
TP	: Doğru pozitif
TN	: Doğru negatif
S_0	: Nesnel benzerlik
S_r	: Bölgesel benzerlik
SSIM	: Yapısal Benzerlik Endeksi Ölçüsü
DSC	: Zar Benzerlik Katsayısı
YZ	: Yapay zekâ
CNN	: Evrimsel sinir ağı
CML	: Klasik makine öğrenmesi

1. GENEL BİLGİLER

1.1. Giriş

Görüntü işleme ve bilgisayarla görme alanlarında yapılan araştırmalar, son yıllarda önemli ilerlemeler kaydetmiş, çok sayıda uygulama alanında etkili çözümler geliştirilmiştir. Yapılan bu araştırmaların önemli bir kısmı, nesnelerin tespiti ve tanınması üzerine odaklanmıştır. Ancak nesnelerin içinde bulundukları çevreyle uyum içinde olduğu yani arka planıyla renk, doku ve desen özelliklerinin çok yakın olduğu durumlarda bu nesnelerin tespiti son derece zorlu bir problem haline gelmektedir. Kamuflaj, nesnelerin çevresindeki ortamla bütünleşmesini sağlayarak dışardan gözlemleyenler tarafından fark edilmelerini engelleyen bir tekniktir. Kamuflaj sadece askeri alanlarda değil, biyolojik görüntüleme, güvenlik izleme ve doğal hayatta da önemli bir yer tutmaktadır.

Askeri uygulamalarda kamuflajın etkin bir şekilde kullanımı, savaş gibi durumlarda düşman unsurlarının tespitini engelleyebilir. Bu durum görüntü işleme ve bilgisayarla görme tekniklerinin kamuflajlı nesneleri doğru bir şekilde tespit edilmesini zorunlu hale getirmektedir. Geleneksel görüntü işleme yöntemleri farklı nesnelerin, dokuların ve desenlerin kullanıldığı bir ortamda kamuflajlı nesnelerin tespiti için yetersiz kalmaktadır. Bu nedenle son yıllarda derin öğrenme yöntemleri bu zorluğu aşmak için yoğun olarak kullanılmaktadır (LeCun et al., 2015). Derin öğrenme özellikle büyük veri setleriyle çalışabilme ve karmaşık görsel desenleri öğrenme yeteneği ile kamuflajlı nesnelerin tespitinde çok etkili bir araç olarak ön plana çıkmaktadır. Kamuflajlı nesnelerin tespiti, genel olarak renk, doku, desen ve şekil gibi temel görsel öğelerin analizini gerektirir. Ancak kamuflaj işlemi bu görsel öğeleri bozarak nesnelerin içinde bulundukları arka planla uyumlu hale gelmesini sağlar. Bu durum kamufle olmuş bu nesnelerin tespit edilmesini zorlaştırmaktadır. Bununla birlikte derin öğrenme tabanlı modeller tüm bu zorlukların üstesinden gelmek için güçlü bir araç olarak kullanılmaktadır. Derin sinir ağları (Deep Neural Networks, DNN) ve Evrişimsel Sinir Ağları (Convolutional Neural Network, CNN) gibi yapılar, görüntülerdeki karmaşık desenleri ve dokusal özellikleri analiz ederek kamuflajın etkisini modelleme noktasında oldukça başarılıdır (Salari et al., 2022; Mondal, 2022; Liang et al., 2024).

Kamuflajlı görüntülerde kamufle edilmiş nesnelerin tespiti, arka planla çok yakın renk, doku ve desen özelliklerine sahip oldukları için oldukça zorlu bir problemdir. Genel olarak

kamufle olmuş nesneler, doğal kamufle olmuş nesneler ve yapay kamufle olmuş nesneler olarak iki kısma ayrılırlar (Stevens & Merilaita,2009). Doğal kamufle olmuş nesneler doğadaki kelebek, balık, kurbağa, kuşlar gibi hayvanların kendilerini avcılarından gizlemek için doğal faktörleri kullanan nesneleri ifade etmektedir. Bu nesnelerin yaptıkları bu işlem doğal kamuflajdır. Kamufle olan canlılar bazı durumlarda taklitçiliği de kullanırlar (Suzuki et al., 2019). Bunun için kendilerini başka nesneler gibi gösterirler. Örneğin ERVA 1.0 veri setinde yer alan Kallima inachus kelebek türü bir yaprağı taklit ederek kamufle olmaktadır. Yapay kamuflajlı nesneler ise askerler ya da tank, uçak, silah gibi savaş aletlerinin özel renklendirme, fiziksel veya kimyasal teknikler yoluyla düşman askerlerine karşı tespit edilebilir özellikleri insanlar tarafından azaltılan nesnelerdir. Bu işlem de yapay kamuflaj olarak adlandırılır (Salari et al., 2022). Kamuflajlı nesne tespiti (COD) belirli bir sahne çevresindeki ortamla benzer doku, şekil, renk veya desenlere sahip nesneleri tespit etmeyi ve bölümlere (segmentlere) ayırmayı amaçlamaktadır (Fan et al., 2020). COD, ön plandaki gizli nesneleri ayırt etmek için çok sayıda görsel tanıma bilgisine dayanır. Kamufle edilmiş nesneler, doğal ortamda ten rengi veya vücut deseniyle gizlenmiş hayvanlar veya modern ekipmanların yardımıyla arka planda insan kamuflajı veya sanata gömülü bazı bilgi parçaları olabilir (Ge et al, 2018).

Doğada desen, doku ve renk bakımından birbirinden farklı binlerce kelebek türü vardır. Bununla birlikte, benzer renk, desen ve doku özelliklerine sahip ancak farklı türlerde kelebek türleri de doğada mevcuttur (Stevens & Merilaita,2009; Ghazanfar et al., 2016). Birçoğu arka planda saklanarak kendilerini kamufle edebilir ve doğal kamuflajın zorlu örneklerini sunarlar. Bu kamuflajlı kelebekler arka plana karışarak mevcut algoritmalara meydan okuyan bir test ortamı sağlayabilir. Kelebekler doğal kamuflajın mükemmel bir örneğini oluştururlar. Çeşitli kelebek türleri, arka plana neredeyse tamamen entegre olan desenler ve renkler sergiler. Bu durum, tespit algoritmalarının hem doku hem de renk açısından test edilmesi için zorlu bir senaryo sunmaktadır. Bu tez çalışmamızda, kelebekleri kullanarak kamuflajlı nesneleri tespit etmek için hibrit bir yöntem geliştirmeyi ve bu yöntemlerin gerçek dünyadaki doğal kamuflaj tespitindeki performansını ölçmeyi amaçlıyoruz. Kamufle edilmiş nesnelerin tespiti, askeri stratejilerden vahşi yaşam takibine kadar geniş bir uygulama alanına sahiptir. Ancak literatürde doğal kamuflajlı nesneler üzerine yapılan çalışmalar sınırlıdır. Literatürdeki çalışmaların çoğu yapay olarak oluşturulmuş kamuflaj desenleri veya basit arka planlar üzerinde test edilmiştir. Gerçek dünyadaki doğal kamuflajın daha karmaşık doğasını modelleme konusunda sınırlı çalışma

vardır. Çalışmamız bu boşluğu doldurmayı ve kelebekleri kullanarak doğal kamuflaj tespiti için daha gerçekçi bir test ortamı sağlamayı amaçlamaktadır. Çalışmamız, doğal sahnelerdeki gerçek kamuflaj örneklerini analiz ederek bu alanda yeni bir bakış açısı sağlamayı amaçlamaktadır. Ayrıca bu araştırma, kamuflajın tespit algoritmaları üzerindeki etkilerini daha iyi anlamak ve gelecekteki çalışmalara rehberlik etmek için önemli bir temel sağlayabilir. Düşük kontrast, doku benzerliği ve arka plan karışımı nedeniyle, kamuflaj edilmiş nesneler tespit algoritmaları için önemli bir zorluk oluşturmaktadır. Çalışmamız, hem geleneksel yöntemleri (Gabor filtreleri, k-ortalamlar algoritması ve LBP) hem de derin öğrenme tabanlı modelleri birleştirerek bu zorlukların üstesinden gelmeyi amaçlamaktadır. Yöntemlerimizin performansını kelebek nesneleri kullanarak ölçmek, bu zorlukların farklı boyutlarını incelememize olanak sağlamaktadır. Mevcut literatürde doğal kamuflaj görüntüleri üzerine çalışmalar bulunsada sadece kamuflaj kelebek görüntüleri üzerine yapılmış bir çalışma bulunmamaktadır. Doğal kamuflaj örnekleri için kelebekleri seçerek, gerçek dünya uygulamalarını daha iyi temsil eden bir veri kümesi sağlamayı amaçladık. Kelebeklerin kamuflaj yeteneği evrimsel biyoloji ve ekolojide önemli bir konudur ve araştırmacıların doğal kamuflajı anlamalarına katkıda bulunabilir. Dolayısıyla tüm bu durumlar bu tez çalışmasındaki motivasyonumuzu oluşturmaktadır.

Kelebekler, doğada arka planla bütünleşme yetenekleriyle tanınan organizmalardır. Bu canlılar, kanat desenleri, renk dağılımları ve şekil varyasyonları sayesinde yırtıcılardan saklanmak için oldukça etkili bir kamuflaj mekanizmasına sahiptirler. Kelebeklerin sahip oldukları bu kamuflaj mekanizmaları sayesinde, sadece görsel estetik açısından değil, aynı zamanda görsel karmaşıklık, doku benzerliği ve arka plan uyumu gibi ölçülebilir nitelikler açısından da bilgisayarla görü sistemleri için zengin bir araştırma alanı sunmaktadırlar.

Bilgisayarla görü uygulamalarında kullanılan kamuflaj tespiti modelleri, çoğunlukla doku, renk yoğunluğu ve kenar süreksizlikleri gibi görsel ipuçlarına dayanır. Kelebeklerin doğal ortamlarda sergilediği bu varyasyonlar, özellikle Gabor filtreleri ve LBP gibi klasik dokusal özellik çıkarma yöntemleriyle etkin şekilde analiz edilebilmektedir. Aynı şekilde kelebek desenlerinin içsel yapısı, derin öğrenme tabanlı transfer öğrenme modellerinin görsel örüntüleri öğrenmesine de olanak sağlamaktadır.

Çalışmada oluşturulan ERVA 1.0 veri seti, kelebeklerin doğal kamuflaj ortamlarıyla birlikte etiketlenmiş görüntülerinden oluşmakta, böylece nesne tespiti ve tanıma problemleri için hem zorlu hem de gerçekçi bir test ortamı sunmaktadır. Kamuflaj kelebek görüntülerinin tercih edilmesindeki temel neden, kelebeklerin kamuflaj potansiyelinin görsel anlamda hem

insan algısı hem de algoritmalar açısından zorluk oluşturmaktadır. Bu zorluk, önerdiğimiz hibrit modelin farklı tekniklerle birleştirilerek test edilmesine uygun bir yapı sağlamaktadır. Kelebeklerin bu çalışmada tercih edilme nedeni yalnızca biyolojik gerekçelere değil, aynı zamanda ölçülebilir görsel zorluk düzeylerine ve analitik olarak anlamlandırılabilir desen yapılarına dayanmaktadır. Bu özellikler, kelebekleri doğal kamuflajlı nesne tespiti problemleri için güçlü bir model organizma haline getirmektedir.

Kelebeklerin kamuflaj özelliklerinin biyolojik olarak açıklanmasının yanı sıra, bu özelliklerin matematiksel veya ölçülebilir bir temele oturtulması da oldukça önemli bir problemdir. Bu problem, bilgisayarla görü ve nesne tespiti alanında, kamuflajın ölçülebilirliğini sağlayan metriklerin kullanılmasıyla çözülebilir. Öncelikle, kelebeklerin kanat desenleri ve renk dağılımlarının kamuflaj etkisi, çeşitli görsel karmaşıklık metrikleri, renk histogramları, desen entropisi ve kenar yoğunluğu ölçümleri ile analiz edilebilir. Bu tür ölçümler çalışmada kullanılarak, kamuflajın ne derece etkili olduğu nesnel olarak değerlendirilmiştir. Çalışmada kullanılan hibrit modelde Gabor filtre ve LBP gibi özellik çıkarma yöntemleri, özellikle doku ve yönelime dayalı varyasyonları matematiksel olarak temsil eder. Bu yöntemler sayesinde, kelebek desenlerinin çevre ile olan benzerliği (arka plan uyumu), ölçülebilir doku benzerliği skorlarıyla ifade edilebilir. Gabor filtreleri, farklı yönelimlerdeki frekans bileşenlerini ölçerken; LBP, mikro doku yapılarını tanımlar. Bu sayede kelebeklerin kanatlarının arka planla olan benzerlik düzeyleri sayısal olarak analiz edilmektedir. Ayrıca kamuflajın tespiti ve sınıflandırılması sürecinde kullanılan derin öğrenme temelli transfer modelleri, sınıflandırma doğrulukları, kayıp (loss) değerleri gibi istatistiksel performans metrikleriyle kelebek kamuflajlarının ayırt edilebilirliğini ölçmektedir. Bu da çalışmada kelebek seçiminin yalnızca sezgisel veya biyolojik değil, aynı zamanda ölçülebilir sınıflandırma zorluklarına dayandığını göstermektedir. Bu bağlamda, kelebeklerin seçimi yalnızca doğal kamuflaj potansiyellerine değil, aynı zamanda bu potansiyelin matematiksel olarak modellenenebilir ve sınıflandırma algoritmalarıyla ölçülebilir olmasına dayanmaktadır. Bu da çalışmanın nesnel ve tekrarlanabilir temellere dayandığını göstermektedir.

Bu tez çalışmasında, kamuflaj edilmiş kelebek nesnelerinin sınırlarını tespit etmek ve türlerini belirlemek için bölütleme (segmentasyon) ve nesne tanıma olmak üzere iki aşamalı bir uygulama geliştirilmiştir. Uygulama için doku analizi ve derin öğrenme tekniklerinden oluşan hibrit bir yöntem önerilmiştir. Bu yöntemin performansının değerlendirilmesi için ERVA 1.0 veri kümesi kullanılmıştır. Bölütleme başarı oranlarının tespit edilmesi için

Structural Similarity (SSIM) ve Dice Similarity Coefficient (DSC) performans metrikleri kullanılmıştır. Bölütleme işlemi sonrasında tespit edilen kelebek nesnesinin türünün tahmin edilmesi için derin öğrenme önceden eğitilmiş modeller kullanılmıştır.

1.2. Tezin Katkısı

Özetle, literatüre temel katkılarımız aşağıdaki gibi özetlenebilir:

- Kamuflej görüntüler için literatürde sıklıkla tercih edilen veri setleri bulunmaktadır. Fakat bu veri setlerinde kamuflej kelebeklere ait görüntülerin sayısını incelediğimizde, sayının yok denecek kadar az olduğu görülmüştür (Salari et al., 2022; Fan et al., 2020; Le et al., 2019). Oluşturduğumuz ERVA 1.0 veri seti ile spesifik olarak sadece kamuflej kelebek görüntüler üzerinde çalışma yapmak isteyen araştırmacılar için büyük bir veri setine ulaşma imkânı sağlanmaktadır.

- ERVA 1.0 veri setindeki eğitim ve test görüntülerinde bulunan kelebek nesnelerinin sınırları elle manuel olarak seçilmiştir ve ikili (binary) görüntü olarak kaydedilmiştir. Bu imkân araştırmacıların elde ettikleri bölütleme ve nesne tanıma başarı oranlarının hesaplanması için kolaylıklar sağlayacaktır.

- ERVA 1.0 veri setinde yer alan eğitim verilerine veri artırma teknikleri uygulanarak görüntü sayısı 4 katına çıkarılmıştır. Ayrıca test verilerinde kamuflejlik derecesi düşük olan görüntülerin kamuflejlik derecesi artırılmıştır. Bu şekilde, ERVA 1.0 özel bir veri seti haline gelmiştir.

- COD problemi için doku analizi ve derin öğrenme tekniklerinin hibrit olarak kullanılmasıyla literatüre yeni bir yöntem sunulmuştur. Bölütleme aşamasında, kamufle edilmiş kelebeğin sınırlarını belirlemek için doku analizi ve kümeleme yöntemleri kullanılmıştır. Sınırları belirlenen kelebek nesnesinin türünün tahmin edilmesi için de derin öğrenme önceden eğitilmiş modeller kullanılmıştır. Böylece bölütleme ve nesne tanıma için doku analizi, kümeleme yöntemleri ve derin öğrenme teknikleri birlikte kullanılmıştır.

- Bölütleme işlemi ile görüntüdeki kamufle olmuş nesnenin sınırları tespit edilmiştir. Ardından, sınırları tespit edilen nesnenin türü derin öğrenme teknikleri ile belirlenmiştir. Önerdiğimiz hibrit yöntemimiz kamufle olmuş nesnenin görüntüde yerinin tespit edilmesi ve tanınması problemi için bundan sonra yapılacak çalışmalarda ilgili araştırmacılara fikir verecek ve yardımcı olacaktır.

- Kamufajlı nesne tespitinde mevcut yöntemlerin eksik ve güçlü olan yönleri belirlenmiştir.

- Derin öğrenme önceden eğitilmiş modellerin bu alandaki performansına dair kapsamlı bir karşılaştırma yapılmıştır.

- Veri artırma ve ön işleme tekniklerinin, kamufajlı nesnelerin tespitine katkısı analiz edilmiştir.

- Kamufajlı nesnelerin tespiti için daha etkin ve verimli yöntemler geliştirilmesine yönelik öneriler sunulmuştur.

1.3. Tezin Amacı ve Genel Yapısı

Tezin temel amacı, görüntülerde kamufle olmuş nesnelerin tespit edilmesi için bölütleme ve tanıma işlemlerinin gerçekleştirilmesidir. Bu kapsamda aşağıdaki hedefler belirlenmiştir:

- Gabor Filtre, LBP, K-ortalamlar ve önceden eğitilmiş derin öğrenme tabanlı yaklaşımları araştırmak, bunların etkinliğini analiz etmek ve potansiyel iyileştirme alanlarını belirlemek,
- Bu kapsamda bu yöntemleri kullanarak kamufaj görüntülerin bölütlenmesi ve tanınmasına yönelik hibrit bir yöntem sunmak,
- Kamufajlı nesnelerin tespitine yönelik literatürdeki mevcut çalışmaların incelenmesi ve derlenmesi,
- COD probleminin çözümüne yönelik ERVA 1.0 adlı yeni orijinal bir kamufaj veri setinin oluşturulması ve araştırmacıların kullanımına sunulması,
- Kamufajlı nesne tespitinde kullanılan veri setlerinin (COD10K, CAMO, CHAMELEON vb.) özelliklerinin analiz edilmesi,
- Derin öğrenme yöntemleri arasında yer alan önceden eğitilmiş modellerin performanslarının kamufaj görüntüler üzerinde karşılaştırılması,
- Veri artırma tekniklerinin bölütleme ve tespit doğruluğuna etkisinin incelenmesi,
- Kamufajlı nesne bölütleme ve tespit süreçlerinde karşılaşılan zorlukların ve bu zorluklara yönelik çözüm önerilerinin sunulmasıdır.

Tezin birinci bölümünde, COD probleminden bahsedilmiş ve genel tanımı yapılmıştır. Bu bölümde kamufaj görüntüler üzerinde literatürde yapılmış çalışmalardan bahsedilmiş ve

bu çalışmalar hakkında kısaca özet olarak genel bilgiler verilmiştir. Bunun yanı sıra literatürdeki modeller, sıklıkla kullanılan kamuflej veri setleri açıklanmış ve önerdiğimiz yöntemin performansını ölçmek için kullanılan değerlendirme metrikleri ayrıntılı olarak açıklanmıştır.

Tezin ikinci bölümünde, önerdiğimiz hibrit yönetime yönelik çalışmalar anlatılmıştır. Oluşturduğumuz ERVA 1.0 veri setinde yer alan görüntüler üzerinde Gabor Filtre, K-ortalamlar, LBP algoritmaları kullanılarak bölütleme işleminin nasıl gerçekleştirildiğinden bahsedilmiş ve bölütleme sonrası tespit edilen kamufle olmuş nesnelerin önceden eğitilmiş derin öğrenme teknikleriyle türünün tahmin edilmesi işleminin nasıl gerçekleştirildiği ayrıntılı olarak açıklanmıştır.

Tezin üçüncü bölümünde çalışmadaki yapılan deneylerin detaylarından bahsedilmiş ve önerdiğimiz hibrit yöntem ile elde edilen sonuçların karşılaştırması yapılmıştır.

Son olarak, tezin dördüncü ve beşinci bölümlerinde bu çalışmada elde edilen deneysel sonuçlardan ve tartışmalardan bahsedilmiş ve en son bölümde ise çalışma ilgili gelecekte yapılması planlanan araştırmalardan ve ilgili önerilerden bahsedilmiştir.

1.4. Literatür Taraması

Görüntüde kamufle olmuş nesnelerin tespiti (Camouflaged Object Detection - COD), bilgisayarla görme alanında oldukça karmaşık bir problem olarak dikkat çekmektedir. COD'un temel amacı, görsel olarak çevresine uyum sağlayan, içinde bulunduğu ortamın arka planı ile çok yakın doku, desen ve renk özelliklerine sahip, dolayısıyla algılanması çok zor olan nesneleri tespit etmektir. COD probleminin çözümü için bölütleme, nesne tanıma gibi çözümler geliştirilmiştir. Kamufleli görüntülerde bölütleme, bir görüntüyü nesne ve arka plan olarak ayırma işlemidir. Kamufleli nesneler, genellikle arka planla görsel olarak çok benzer olduğu için, bölütleme işlemi oldukça zorlu bir görevdir. Bu zorluk, nesne ve arka plan arasındaki düşük kontrast, bulanık sınırlar ve karmaşık dokular nedeniyle artar. Bölütleme, özellikle tıp, askeri ve vahşi yaşam izleme gibi alanlarda kullanılır. Bu süreçte, dokusal, renk, frekans ve derin öğrenme tabanlı özellikler, kamufleli nesnelerin daha doğru şekilde ayrılmasını sağlar. Kamufleli görüntülerde nesne tanıma, bir görüntüde gizlenmiş nesneleri algılama, sınıflandırma ve tanımlama işlemidir. Bu süreç, nesnenin görsel özelliklerini analiz ederek arka planla uyum sağlayan nesnelerin fark edilmesini ve doğru şekilde sınıflandırılmasını içerir. Kamufleli nesneler genellikle düşük kontrast, bulanık

sınırlar ve karmaşık sahne yapıları nedeniyle arka plandan ayırt edilmesi zor olan nesnelerdir. Nesne tanıma işlemi, görüntü işleme, makine öğrenimi ve derin öğrenme yöntemleriyle gerçekleştirilir. Kamufajlı nesnelerde, özellikle dokusal, renk ve kenar özellikleri gibi ayırt edici özniteliklerin çıkarılması ve derin öğrenme tabanlı modellerle işlenmesi kritik öneme sahiptir.

Son yıllarda, COD problemi için geliştirilen yöntemler, derin öğrenme, frekans analizi, derinlik bilgisi ve çoklu-modalite tabanlı yaklaşımlar ile önemli gelişmeler göstermiştir. COD, farklı yaklaşımlar ve modellerle ele alınmış bir problemdir. Bu bölümde hem klasik yöntemlere hem de modern derin öğrenme tabanlı algoritmalara dayalı modeller açıklanmaktadır. Her yaklaşımın avantajları, zorlukları ve dezavantajları olabilir. Tüm bunlar aşağıdaki gibi detaylandırılmakta ve literatürdeki başlıca çalışmalar kapsamlı bir şekilde ele alınmaktadır.

1.4.1. Klasik Yöntemler (Geleneksel Görüntü İşleme Teknikleri)

COD alanında klasik yöntemler, genellikle bilgisayarla görmenin temel taşlarını oluşturan geleneksel görüntü işleme tekniklerine dayanmaktadır. Bu yöntemler, derin öğrenme öncesi dönemde sıklıkla kullanılmış olup, veriye doğrudan istatistiksel veya matematiksel işlemler uygulayarak sonuç üretirler.

Kenar Algılama Teknikleri: Kamufaj görüntülerde nesne kenarlarının belirginleştirilmesi için çeşitli kenar algılama algoritmaları kullanılmaktadır. Pan ve arkadaşları çalışmalarında gri tonlama analizi ve kenar algılama üzerine yoğunlaşarak kamufajlı nesnelerin tespit edilmesinde 3D konveks yapı özelliklerini kullanmıştır. Yaptıkları çalışmalar kenar tespitinin nesnelerin arka plan gürültüsünden ayrıştırılmasına yardımcı olmuştur. Çalışma, klasik kenar tespiti yöntemlerine kıyasla yüksek doğruluk sunmuştur fakat bulanık ve arka planla karışmış sınırların olduğu durumlarda başarısız kalmıştır (Pan vd., 2011). Zijian Liu ve arkadaşları, kenar bilgilerini daha iyi kullanmak için bir Kenar Dikkati Ağı (Edge Attention Network-EANet) geliştirmiştir. Model, gereksiz sınır bilgilerini filtreleyerek hedef nesnenin kritik kenar özelliklerini çıkarmaktadır. Çalışma, düşük hesaplama maliyetine rağmen güçlü performans göstermiştir (Liu vd., 2024). Yixuan Lyu ve arkadaşları, belirsizlik ve kenar bilgilerini birleştiren UEDG modeliyle, zor tespit edilen sınırları belirlemeyi amaçlamıştır. Model, belirsizlik tabanlı öğrenme ile kenar bilgisini birleştirerek daha doğru tespitler gerçekleştirmiştir (Lyu vd., 2024). Xingpeng

Zhang ve arkadaşları, çok seviyeli kenar özelliklerini birleştiren hafif bir model geliştirmiştir. Model, kenar bilgisi ile birlikte düşük ve yüksek seviyeli özellikleri birleştirerek düşük kaynak kullanımı ile yüksek doğruluk sağlamıştır (Zhang vd., 2024). Jiesheng Wu ve arkadaşları hem kenar hem de maske bilgilerini birleştiren bir model geliştirmiştir. Çalışma, çoklu ölçekli özellikleri birleştirerek tespit performansını artırmıştır (Wu vd., 2023).

Yujia Sun ve arkadaşları, sınır yönlendirmeli bir ağ (BGNet) geliştirerek, nesne yapılarını daha iyi tanımlamak için kenar semantiğini kullanmıştır. Çalışma, sınır doğruluğunu artırmayı hedeflemiştir (Sun vd., 2022). Dongyue Sun ve arkadaşları, kenar tespiti ve kamuflajlı nesne bölütlemesi için çapraz bir iyileştirme süreci öneren bir model geliştirmiştir. Bu model, düşük seviyeli özelliklerden yapısal detayları daha iyi çıkarmıştır (Sun vd., 2023). Hao Yuan ve arkadaşları, kenar rehberli bir modül kullanarak kamuflajlı nesnelerin daha net sınırlarla tespit edilmesini sağlamıştır. Bu model, özellikle çoklu hedefler için sınır netliğini iyileştirmiştir (Yuan vd., 2024). Qi Jia ve arkadaşları, çok aşamalı bir tespit stratejisi uygulamış ve düşük çözünürlüklü küçük nesnelerin tespiti için iyileştirmeler yapmıştır (Jia vd., 2022). Tankus ve Yeshurun, 3D konveks nesnelerin tespitinde kenar tabanlı yaklaşımların sınırlamalarını ele alan bir yöntem geliştirmiştir (Tankus & Yeshurun, 2001).

Klasik yöntemlerde nesnelerin sınırlarını belirlemek için kenar algılama teknikleri yaygın olarak kullanılmıştır. Bu yöntemlere örnek olarak Sobel, Prewitt ve Canny gibi algoritmalar verilebilir. Bu algoritmalar kamuflaj olmayan normal görüntüdeki parlaklık değişimlerini analiz ederek keskin kenarları tespit edebilir. Fakat kamuflajlı nesnelerde, arka plan ve nesne arasındaki minimal farklılıklar bazen bu tekniklerle çıkarılabilir de zayıf sınırların olduğu nesnelere ait görüntülerde başarısız olurlar. Kamuflajlı nesnelerin sınırları genellikle bulanık veya belirsiz olduğu için kenar algılayıcıların COD probleminde başarıları çok sınırlıdır. Bununla birlikte kenar algılama teknikleri, kamuflajlı nesne tespitinde sınırların belirginleştirilmesi ve genel doğruluğun artırılması için güçlü araçlar sunmaktadır. Modern derin öğrenme yöntemleriyle birleştirilen kenar bilgileri, bu alandaki ilerlemeyi daha da hızlandırmıştır.

Renk ve Doku Analizi: Kamuflajlı nesneler genellikle arka planla renk ve doku bakımından benzerlik gösterirler. Bu durum, geleneksel yöntemlerin bu farklılıkları analiz ederek nesneyi ortaya çıkarmasına olanak tanır. Renk ve doku analizi için kullanılan yöntemlere Histogram Tabanlı Renk Analizi ve Gabor Filtresi örnek olarak verilebilir. Renk

histogramları bir nesnenin renk dağılımını tanımlayarak farklılıkları belirlemeye çalışır. Doku analizi için sıklıkla kullanılan Gabor Filtresi ise farklı yönelim ve frekans bileşenlerini inceleyerek dokusal farklılıkları vurgular. Literatürde Renk ve Doku Analizi ile yapılmış çalışmalar mevcuttur:

Ren ve arkadaşları, doku farkını artırmak için derin öğrenme tabanlı bir model önerdiler. Model, kamuflajlı nesneler ile arka plan arasındaki doku farklarını belirlemek için özel bir doku-odaklı iyileştirme modülü içeriyor. Çalışma, sınır detaylarını keşfetmek için bir sınır-tutarlılık kaybı kullanarak mevcut yöntemlerden daha yüksek doğruluk sağlamıştır (Ren vd., 2021). Li ve arkadaşları, doku ve sınır bilgilerini birleştirerek kamuflajlı nesne tespiti için bir ağ geliştirdiler. FindNet modeli, nesne özelliklerini geliştirmek için doku ve sınır bilgilerini birleştirerek önemli iyileştirmeler sağlamıştır (Li vd., 2022). Zhu ve arkadaşları, doku bilgilerini iyileştiren bir rehberlik ağı geliştirmiştir. Bu model, doku farkını analiz ederek kamuflajlı nesnelerin sınırlarını ve yapılarını belirginleştirmiştir (Zhu vd., 2021). Chennamsetty ve arkadaşları, doku istatistiklerine dayalı bir yöntem geliştirmiştir. Bu yöntem, pürüzsüzleştirilmiş dokuları ve istatistiksel özellikleri kullanarak askeri ve sivil uygulamalarda kamuflajlı nesneleri tespit etmek için uygulanmıştır (Chennamsetty vd., 2020). Bhajantri ve Nagabhushan, renk-doku benzerliğine dayalı bir modelle kamuflajlı nesneleri tespit etmeye yönelik bir yöntem geliştirmiştir. Yöntem, doku farklarını belirlemek için Co-Occurrence Matris tabanlı doku analizine dayanmaktadır (Bhajantri & Nagabhushan, 2006). Cong ve arkadaşları, doku ve frekans bileşenlerini kullanan bir model geliştirmiştir. FPNet, kamuflajlı nesnelerin çevre ile olan benzerliğini azaltmak için frekans alanında özellikleri optimize etmiştir (Cong vd., 2023). Renk ve doku analizi, kamuflajlı nesne tespitinde önemli bir rol oynamaktadır. Yukarıda özetlenen çalışmalar, bu tekniklerin etkinliğini ortaya koymakta ve yeni modellerin geliştirilmesi için bir temel sağlamaktadır. Bu yöntemler, özellikle benzer renk ve dokulara sahip arka planlarda nesneleri tespit etmek için güçlü araçlar sunmaktadır. Bununla birlikte, nesne ile arka plan arasındaki çok küçük renk ve doku farklılıklarını algılamada sınırlıdır.

Hareket Tabanlı Yöntemler: Hareket tabanlı yöntemler, kamuflajlı nesnelerin tespitinde, nesnelerin hareketinden kaynaklanan ipuçlarını kullanarak arka planla nesne arasındaki farkı ortaya çıkarmayı amaçlar. Bu yöntemler, özellikle durağan sahnelerde zor tespit edilen nesnelerin hareket yoluyla belirginleştiği durumlarda etkili olmuştur. Literatürde, bu alanda önemli çalışmalar yapılmıştır.

Cheng ve arkadaşları, video tabanlı kamuflajlı nesne tespiti (VCOD) için hareket ipuçlarını ve uzun süreli zaman bilgilerini birleştiren bir yöntem geliştirdi. Çalışma, hareket tahmini ve bölütleme hatalarını minimize etmek için hareket tahmini ve nesne bölütlenmesini birleştiren bir optimizasyon çerçevesi önermiştir (Cheng vd., 2022). Hui ve arkadaşları, hareketin hem örtük hem de açık öğrenimini entegre eden bir yöntem geliştirdi. Bu yöntem, nesne sınırlarının belirginleştirilmesi için hareket ipuçlarını kullanarak zaman-mekânsal ilişkiyi optimize etmiştir (Hui vd., 2024). Zhang ve arkadaşları, hareket ipuçlarını açıkça işleyen bir iki akışlı mimari geliştirdi. Bu yöntem, optik akış temelli hareket tahmini ile kamuflajlı nesneleri belirlemek için etkileşimli bir sistem önermiştir (Zhang vd., 2024). Zhang ve Zhu, hareket eden kamuflajlı nesneleri tespit etmek için arka plan ve nesne özelliklerini ayrı ayrı modelleyen bir yaklaşım geliştirdi. Bu yöntem, hareketli nesneleri tanımlamak için hareket modelleme stratejileri kullanmıştır (Zhang & Zhu, 2015). Zhang ve arkadaşları, hareket eden kamuflajlı nesneleri tespit etmek için Bayesian bir model önerdi. Çalışma, arka plan ve nesne modellerini karşılaştırarak kamuflajlı alanları tanımladı (Zhang et al., 2017). Li, dinamik arka planlarda optik akışa dayalı bir yöntem geliştirdi. Bu yöntem, hareket düzenlerini analiz ederek hareket eden nesneleri tespit etti (Li, 2011). Lamdouar ve arkadaşları, hareket bölütlemesi kullanarak kamuflajlı nesneleri ortaya çıkaran bir model geliştirdi. Çalışma, hareketin eksik olduğu durumlarda nesne sürekliliğini koruyan bir kayıt modülü önerdi (Lamdouar vd., 2020).

Hareket tabanlı yöntemler, kamuflajlı nesnelerin hareket yoluyla ortaya çıkan özelliklerini kullanarak tespit performansını artırmayı başarmıştır. Bu yöntemler, özellikle video verileriyle çalışırken yüksek doğruluk ve zamansal tutarlılık sunmaktadır. Geliştirilen teknikler, optik akış, Bayesian modelleme ve zaman-mekânsal ilişkileri optimize eden çerçeveleri içermektedir.

Frekans Analizi: Frekans analizi, COD alanında, nesne ve arka plan arasındaki ince farklılıkları vurgulamak için güçlü bir araç olarak kullanılmaktadır. Frekans bileşenlerinin kullanımı, özellikle düşük kontrastlı veya görsel olarak uyumlu sahnelerde tespit doğruluğunu artırmayı hedefler. Literatürde, frekans analizi yöntemlerini kullanan önemli çalışmalar yer almaktadır.

Chenxi Xie ve arkadaşları, Laplace piramidi benzeri ayrıştırma kullanarak yüksek frekans bileşenlerini çıkaran bir model geliştirmiştir. Bu model, düşük frekanslı sinyalleri analiz etmek için çoklu başlı dikkat mekanizmalarını da içermektedir (Xie vd., 2023). Yijie Zhong ve arkadaşları, ayırık kosinüs dönüşümü ve frekans artırma modülleri kullanarak RGB

ve frekans alanı özelliklerini birleştiren bir yöntem geliştirmiştir. Bu yöntem, kamuflajlı nesneleri tespit etmek için öğrenilebilir frekans ipuçlarını kullanmaktadır (Zhong vd., 2022). Jiaying Lin ve arkadaşları, yüksek frekanslı doku bilgilerini bastırmak ve düşük frekanslı özellikleri vurgulamak için frekans temelli bir model (FBNet) geliştirmiştir (Lin vd., 2022). Shizhou Zhang ve arkadaşları, frekans ve uzamsal alanları birleştiren bir adaptasyon yöntemi geliştirmiştir. Model, farklı frekans bileşenlerini dinamik olarak güçlendiren veya zayıflatan bir modül içermektedir (Zhang vd., 2024). Yanguang Sun ve arkadaşları, frekans ve uzamsal alanları birleştiren karmaşık bir öğrenme yöntemi olan Frekans-Uzamsal Dolanıklık Öğrenimi (FSEL) modelini geliştirmiştir (Sun vd., 2024). Weiyun Liang ve arkadaşları, hafif bir model için yüksek ve düşük frekanslı detayları ayrı ayrı işleyen bir frekans enjeksiyon modülü geliştirmiştir (Liang vd., 2024). Frekans analizi, COD alanında, nesne ve arka plan arasındaki farklılıkları vurgulamak için güçlü bir araç olarak kullanılmaktadır. Yukarıdaki çalışmalar, frekans ipuçlarının etkin kullanımının hem düşük hem de yüksek frekanslı bileşenler için önemli olduğunu göstermektedir. Bu yöntemler, görüntülerin hem bağlamsal hem de spektral özelliklerini zenginleştirerek daha hassas tespit sonuçları sunmaktadır.

1.4.2. Derin Öğrenme Tabanlı Yöntemler

Derin öğrenme tabanlı yöntemler, COD alanında önemli ilerlemeler sağlamıştır. Özellikle nesne ile arka plan benzerliğini aşma ve karmaşık görsel ilişkileri öğrenmede büyük katkılar sunmuştur. Bu yöntemler kamuflajlı görüntülerde, nesne ve arka plan arasındaki yüksek benzerliklere rağmen nesneleri tespit etmek için karmaşık görsel ipuçlarını öğrenme yeteneği ile ön plana çıkmıştır. Literatürde COD problemi için yapılmış oldukça fazla derin öğrenme tabanlı çalışmalar mevcuttur.

Khair ve arkadaşları, derin öğrenme tabanlı VGG16 modelini kullanarak 6500'den fazla kamuflajlı görüntü üzerinde deneyler gerçekleştirmiştir. Transfer öğrenme yöntemleriyle modeli optimize ederek yüksek tespit doğruluğu elde edilmiştir (Khair vd., 2023). Wang ve arkadaşları, kamuflajlı nesneleri tespit etmek için derin özellik çıkarımı ve kamuflaj verimliliğini ölçme yeteneği olan bir model geliştirmiştir. Model, farklı sahnelerde düşük, orta ve yüksek kamuflaj seviyelerini etkili bir şekilde tespit etmiştir (Wang vd., 2020). Chen ve arkadaşları, arka yayılım olmadan çalışan ve derin sinir ağlarından elde edilen özellikleri kullanan bir model geliştirmiştir. Bu yaklaşım, daha az hesaplama gücü gerektiren bir çözüm sunmaktadır (Chen vd., 2024). Bi ve arkadaşları, derin öğrenme tabanlı

modeller ve veri setlerini kapsamlı bir şekilde incelemiştir. Çalışma, 1998-2021 yılları arasında geliştirilen otuz dokuz modeli ve dört yaygın veri setini (CHAMELEON, CAMO, COD10K, NC4K) özetlemiştir (Bi vd., 2022). Chen ve Zhang, farklı ipuçlarını uyarlayabilen bir rehberlik öğrenme ağı (AGLNet) geliştirmiştir. Bu model, RGB görüntüler ve ek bilgi kaynakları arasında derin özellik birleşimini sağlamaktadır (Chen & Zhang, 2024). Donga ve arkadaşları, büyük bir alıcı alan ve çok seviyeli özellik birleştirme ile kamufleli nesneleri daha doğru tespit eden bir derin öğrenme yöntemi geliştirmiştir (Donga vd., 2021). Zhuge ve arkadaşları, standart kodlayıcı-kod çözücü mimarisine X bağlantısı eklenerek oluşturulan CubeNet adlı yeni bir COD çerçevesi önerdiler. Önerilen model COD10K, CAMO ve CHAMELEON veri setleri üzerinde denenmiş ve sonuçlar alınmıştır (Zhuge vd.,2022). Xu ve arkadaşları, COD için yerelleştirme ve iyileştirme aşamalarından oluşan iki aşamalı bir çerçeve izleyen bir sınır yönlendirme ağı tasarlamışlardır. Yöntem ile elde edilen sonuçları hem nesne doğruluğu hem de sınır doğruluğunda gerçek zamanlı hızla mevcut algoritmalarla elde edilen sonuçlarla karşılaştırdılar (Xu vd.,2021). Le vd. kamufle edilmiş nesne bölütleme problemini ele aldılar. Kıyaslama amacıyla kamufle edilmiş nesnelerden oluşan CAMO adlı yeni bir görüntü veri seti sundular. Ayrıca hem sınıflandırma hem de bölütleme görevlerinden yararlanan Anabranh Ağı adı verilen genel bir uçtan uca ağ önerdiler (Le vd., 2019). Zhang vd. Komşu Bağlantı Modu (NCM) ve Hiyerarşik Bilgi Aktarımı (HIT) olmak üzere iki ana bileşenden oluşan yeni bir COD çerçevesi sunmuşlardır (Zhang vd.,2022). Kajiura vd. tahmin edilen kamufleli haritayı iyileştirmek için belirginliğin yanı sıra kenarları da tam olarak kullanan yeni bir çerçeve önermişlerdir. Bu çerçeve sözde kenar üretici, sözde harita üretici ve belirsizliğe duyarlı iyileştirme modülü isimli üç temel bileşenden oluşmaktadır (Kajiura vd.,2021). Sun vd. tüm nesne kapsamını etkinleştirebilecek özellik haritaları elde etmek ve segmental sonuçların gürültülü özellikler tarafından boğulmasını önlemek için Çapraz Model Detay Sorgulama Ağı (DQnet) adlı yeni bir çerçeve önermişlerdir. Önerilen DQnet yöntemi kullanılarak yaygın olarak kullanılan COD veri setleri üzerinde deneyler gerçekleştirmişlerdir (Sun vd., 2022). He vd. R2C7K adlı büyük ölçekli bir veri seti oluşturmuşlardır. Referans görüntülerden hedef nesnelerin ortak temsillerini içeren bir referans dalı ve ortak temsillerin rehberliğinde kamufle edilmiş nesneleri tanımlayan ve bölümlere ayıran bir bölümlendirme dalı ile R2CNet adlı bir çift dallı çerçeve geliştirilmiştir (He vd., 2023). Zhang vd. COD için Özellik Ayırıştırma ve Kenar Yeniden Yapılandırma (FEDER) modelini önermişlerdir (Zhang vd., 2023).

Wang vd. kamuflajlı böcekler için aşamalı iyileştirme ağı (PRNet) adlı derin öğrenmeye dayalı bir böcek görüntüsü bölütleme yöntemi geliştirmiştir. Önerdikleri PRNet'in kamuflaj edilmiş böcekleri etkili bir şekilde bölütlediğini gösterdiler. Çalışmada ortalama %3,2 mutlak hata, %89,7 piksel eşleştirme derecesi, %83,6 yapısal benzerlik ve %72 hassasiyet ve geri çağırma hatası ile üstün algılama performansı elde edilmiştir (Wang vd., 2023). Le vd. hiyerarşik piksel bazlı temel gerçeklere sahip görüntü sayısını önemli ölçüde artıran CAMO++ adlı yeni bir veri seti sunmuşlardır. Ayrıca, son teknoloji yöntemlerin performansını daha da artırmak için kamuflajlı örnek bölütleme için bir kamuflaj füzyon öğrenme (CFL) çerçevesi sunmuşlardır (Le vd., 2021). Jiang vd. kamuflaj olmuş nesneleri tespit etmek için üçlü kademeli algı tabanlı bir yöntem önermiş ve kademeli bir algı ağı (CPNet) oluşturmuşlardır. Deneyler COD10K veri kümesi üzerinde gerçekleştirilmiştir. Elde ettikleri deneysel sonuçlara göre CPNet'in 91.41'e ulaşan ortalama hassasiyet AP_{50} , 73.04'e yükselen AP_{75} ve var olan yöntemlere göre önemli ölçüde daha yüksek tespit doğruluğu elde ettiklerini belirttiler (Jiang vd., 2023). Li vd. kamuflajlı nesne tespiti için Ayrıştırıcı Bilgi Dikkati ve Çapraz Seviye Özellik Füzyon Ağı (DACF-Net) önermişlerdir. Çalışmalarında ayrıca özellik fazlalıklarını bastırmak ve daha tamamlayıcı bilgilerden yararlanmak amacıyla bir Çapraz Seviyeli Özellik Toplama Modülü (CFAM) kullanılmıştır (Li vd., 2022). Yan vd. görsel alan eşleştirme ve tanıma modülü (VFMRM) ve kademeli iyileştirme modülü (SWRM) olmak üzere iki temel modülden oluşan bir eşleştirme-tanıma-iyileştirme ağı (MRR-Net) önermektedir. Çalışmalarında VFMRM' de farklı boyut ve şekillerdeki kamuflaj edilmiş nesnelerin aday alanlarını eşleştirmek ve gerçek kamuflaj edilmiş nesnenin yaklaşık alanını uyarlamalı olarak etkinleştirmek ve tanımak için çeşitli özellik alıcı alanları kullanılmaktadır. SWRM daha sonra omurga tarafından çıkarılan özellikleri kullanarak VFMRM tarafından elde edilen kamuflaj edilmiş bölgeyi kademeli olarak iyileştirir ve böylece kamuflaj edilmiş nesnenin tamamını elde eder (Yan vd., 2023).

Huang vd. geleneksel kodlayıcı-kod çözücü yapısı yerine Geri Beslemeli Kısmi Kod Çözücüler (PFD) önerdiler. Bunun yanı sıra kademeli bir yapıdan yararlanan RGB optik sensöre dayalı bir derin öğrenme çerçevesi olan Kademeli ve Geri Beslemeli Füzyonlu Kamuflajlı Nesne Algılama (CODCEF) önerdiler (Huang vd., 2021). Liu vd. kamuflajlı nesne tespiti için yeni birçok seviyeli özellik entegrasyon ağı (MFNet) önermişlerdir. COD performansını artırmak için bir kenar yönlendirme modülü (EGM) tasarladılar. Ayrıca özellik temsili sağlayan ve tüm bağlam anlamsal bilgilerini etkili bir şekilde entegre eden çok seviyeli bir özellik entegrasyon modülü (MFIM) önerdiler. Son olarak, net tahmin

haritaları elde etmek için çapraz seviyeli özellikleri verimli bir şekilde toplamak ve iyileştirmek için bir bağlam toplama iyileştirme modülü (CARM) önerdiler (Liu vd., 2023). Yang ve Zhang, Spesifik olarak, göze çarpan nesne algılama (SOD) görevini COD modeline dahil ettiler. Çalışmalarında kamufle edilmiş nesnelerin göze çarpmayan ancak değerli ipuçlarını vurgulamak için kamuflej koşullarını yok etmek üzere COD için çok görevli bir öğrenme çerçevesi sunulmuştur (Yang & Zhang, 2023). Chou vd. COD için hedef nesne ile arka plan arasındaki mevcut yüksek içsel benzerlik probleminden bahsetmişlerdir. Bu problemin çözümü için her bir dekamuflej modülünün kamufle edilmiş nesneyi doğru bir şekilde tespit etmek için bölge geliştirme bloğu ve ters dikkat madenciliği bloğundan oluştuğu tahmin haritasını aşamalı olarak iyileştirmek için Kademeli Dekamuflej Modülünü önerdiler (Chou vd., 2022). Kamran vd. Optimize Global Refinement (OGR) adlı, COD görevleri için optimize edilmiş modüler bir çerçeve sunmuşlardır. Önerilen modelin karmaşıklığını azaltmak için optimize edilmiş yerel en iyi özellik düğüm tabanlı kural önerdiler. Deneysel çalışmalarda Kvasir-SEG, COD10K ve CAMO veri kümeleri için sırasıyla %94, %93 ve %96'lık başarı elde ettiklerini rapor ettiler (Kamran vd., 2022). Mei vd. COS için doğadaki avlanma sürecini taklit eden Konumlandırma ve Odak Ağı (PFNet) adlı biyo-esinlenmiş bir çerçeve geliştirmişlerdir. PFNet, konumlandırma modülü (PM) ve odak modülü (FM) olmak üzere iki temel modül içermektedir (Mei vd., 2021).

1.4.3. Hibrit ve Çoklu Modalite Yöntemler

Hibrit ve çoklu modalite yaklaşımlar, COD alanında farklı veri türlerini (ör. RGB, derinlik, termal görüntüler) birleştirerek tespit performansını artırmayı hedeflemektedir. Literatürde bu alanda yapılmış birçok çalışma mevcuttur.

Wang ve arkadaşları, RGB ve derinlik verilerini birleştirerek kamuflejli nesneleri daha doğru şekilde tespit eden DaCOD adlı bir model geliştirmiştir. Bu yöntem, çoklu modalite öğrenimini ve asimetrik füzyon tekniklerini kullanarak doğruluğu artırmıştır (Wang vd., 2023). Tang ve arkadaşları, multimodal büyük dil modelleri (MLLM'ler) kullanarak sıfır-atış öğrenme tabanlı bir model geliştirmiştir. Model, dilsel ve görsel ipuçlarını birleştirerek daha önce görülmemiş nesneleri tespit etmede başarılı olmuştur (Tang vd., 2024). Cheng ve arkadaşları, dil ve görsel bilgiyi birleştiren bir yaklaşım geliştirerek hedef nesnelerin ve çevresel sahnelerin derinlemesine algılanmasını sağlamıştır. Model, çok düzeyli bilgi rehberliği ile sıfır-atış yetenekleri sergilemiştir (Cheng vd., 2023). Zhao ve arkadaşları, doku

ve renk benzerliklerini analiz etmek için çoklu ölçekli ve dikkat mekanizmalarına sahip bir model geliştirmiştir. Bu model, kamuflajlı nesnelerin farklı ölçeklerdeki tespitini optimize etmiştir (Zhao vd., 2024). Zhou ve arkadaşları, farklı katmanlar arasında bilgi paylaşımını artırmak için çapraz katman füzyon modülü geliştirmiştir. Bu model, sınır odaklı öğrenim ile tespit doğruluğunu artırmıştır (Zhou vd., 2022). Xiang ve arkadaşları, mevcut derinlik tahmin yöntemlerini kullanarak RGB ve derinlik haritalarını birleştiren bir model geliştirmiştir. Model, gürültülü derinlik haritalarıyla çalışabilmektedir (Xiang vd., 2021).

1.4.4. Literatürdeki Boşluklar

COD, kamufle olmuş nesnelerin çevreyle görsel uyumunun yüksek olduğu durumlarda bölütleme ve tanıma problemini çözmeyi hedefler. Ancak, bu alandaki literatür incelendiğinde çeşitli eksiklikler ve zorlukların var olduğu görülmektedir. Kamuflajlı nesne tespiti ve bölütlenmesi alanında, veri çeşitliliği, modalite entegrasyonu, sınır tespiti, sıfır-atış öğrenme ve gerçek zamanlı çözümler gibi önemli araştırma boşlukları bulunmaktadır. Bu alanlarda yapılacak ilerlemeler, COD' un pratik uygulamalarını önemli ölçüde geliştirebilir ve daha genel modellerin geliştirilmesine olanak sağlayabilir. Aşağıdaki başlıklar, bu boşlukların mevcut literatürde nasıl ele alındığını ve ilerleyen çalışmalara yönelik ipuçlarını özetlemektedir.

Sınırlı Veri Setleri ve Anotasyon Eksiklikleri: Kamuflajlı nesneleri tespit etmek için yüksek kaliteli ve iyi anotasyon yapılmış veri setlerine ihtiyaç duyulmaktadır. Ancak literatürde mevcut veri setlerinin sayısı sınırlıdır ve çoğu yalnızca belirli sahne türlerini kapsamaktadır. COD10K gibi büyük ölçekli veri setleri geliştirilmiş olsa da bu veri setlerinde daha fazla çeşitlilik ve kapsam gereklidir (Fan vd., 2020). Ayrıca, verilerin anotasyonu yüksek uzmanlık gerektirdiğinden, anotasyon hataları veya eksiklikleri sıklıkla görülmektedir.

Görüntü Modaliteleri Arasındaki Sınırlı Birleşim: RGB görüntüler genellikle COD problemlerinde yaygın olarak kullanılmaktadır. Ancak, derinlik (RGB-D), termal ve hiperspektral görüntüler gibi farklı modalitelerin birleştirilmesi, daha iyi tespit performansı sağlayabilir. Literatürde, bu modalitelerin entegrasyonuna odaklanan çalışmaların sınırlı olduğu görülmektedir (Wang vd., 2023). Özellikle, multimodal bilgi birleştirme stratejilerinde genel bir standardizasyon eksikliği bulunmaktadır.

Zorlu Kamuflej Senaryolarında Performans Düşüşü: Mevcut modeller, yüksek benzerlik gösteren nesne ve arka plan durumlarında performans sorunları yaşamaktadır. Frekans analizi ve dokusal farkları içeren yaklaşımların bu problemleri çözme potansiyeli bulunmaktadır ancak bu alan hala yeterince araştırılmamıştır (Lin vd., 2022). Özellikle düşük ışık koşulları veya hareketli sahnelerde performans ciddi şekilde etkilenmektedir (Chen vd., 2022).

Zayıf Sınır Tespiti ve Sınır Bilgilerinin Eksik Kullanımı: Kamuflejlı nesnelerin sınırlarının net bir şekilde tespiti, bu alandaki temel zorluklardan biridir. COD modelleri, genellikle nesne sınırlarının hassas bir şekilde çıkarılmasında zorluk çekmektedir. Sınır yönlendirmeli ağlar bu sorunu kısmen çözsede literatürde bu yaklaşımın yaygın olarak uygulanmadığı görülmektedir. Sınır odaklı öğrenme yaklaşımlarının literatürde sınırlı bir şekilde ele alındığı görülmektedir (Sun vd., 2022). Sınır bilgisi ile nesne ve arka plan ayrımının iyileştirilmesi, mevcut modellerde daha fazla araştırmaya ihtiyaç duyulan bir alandır.

Sıfır-Atış ve Zayıf Denetimli Öğrenme Eksikliği: Kamuflejlı nesnelerin sınırlı veri ile tespit edilmesi durumlarında mevcut modellerin performansı düşmektedir. Kamuflejlı nesnelerin sınırlı veri ile tespiti, sıfır-atış öğrenme yaklaşımlarıyla mümkün olabilir. Sıfır-atış öğrenme yöntemleri bu sorunu çözmek için umut vaat etmektedir, ancak COD için sıfır-atış yaklaşımlarının sayısı azdır. Benzer şekilde, zayıf denetimli öğrenme yaklaşımlarında da bir araştırma boşluğu bulunmaktadır (Li vd., 2023).

Zaman-Mekânsal Bilgi Kullanımı Eksikliği: Video COD uygulamalarında zaman-mekânsal bilgilerin kullanımı, nesne hareketlerini tespit etmek için kritik öneme sahiptir. Ancak, mevcut çalışmaların çoğu yalnızca durağan görüntülere odaklanmaktadır. Hareket bilgisi ve zaman-mekânsal ilişkiler, daha doğru bölütleme ve tanıma sağlamak için kullanılabilir (Cheng vd., 2022).

Gerçek Zamanlı Uygulamalar ve Hesaplama Maliyetleri: COD için geliştirilen çoğu model, yüksek hesaplama kaynaklarına ihtiyaç duymaktadır. COD alanındaki modeller genellikle yüksek hesaplama maliyetine sahiptir. Bu durum, özellikle gerçek zamanlı uygulamalar için sınırlamalar oluşturmaktadır. Hafif ağlar ve enerji verimli çözümler konusunda daha fazla çalışma gereklidir (Chen vd., 2024).

1.4.5. Literatürde Kullanılan Önemli Kamuflej Veri Setleri

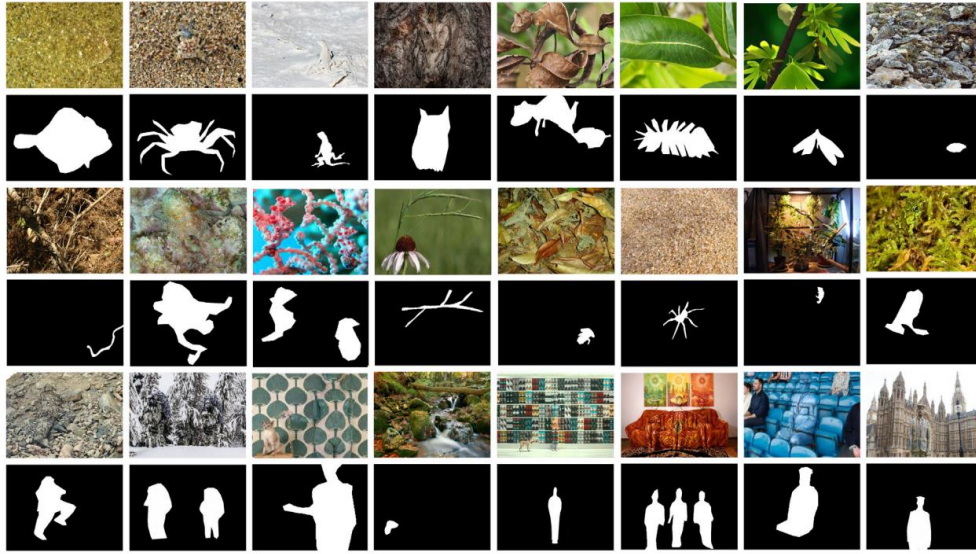
Kamufleli nesne tespiti ve bölütleme problemlerinde, doğru ve etkili veri setlerinin kullanımı kritik öneme sahiptir. Bu veri setleri, modellerin eğitilmesi, test edilmesi ve karşılaştırmalı analizler yapılması için temel teşkil eder. Literatürde kamuflej görüntülerden oluşan ve öne çıkan birçok veri seti mevcuttur. Bunlardan sık kullanılanlar CHAMELEON (Skurowski et al., 2018), CAMO (Le et al., 2019), CAMO-COCO (Le et al., 2019), COD10K (Fan et al., 2020), NC4K (Lv et al., 2021) olarak sıralanabilir.

CHAMELEON Veri Seti: Akran değerlendirmesinden geçmemiş küçük bir veri kümesidir. Mevcut en küçük COD veri kümesi olan bu veri kümesi yalnızca 76 görüntü içermektedir. Veri setindeki görseller Google arama motoru aracılığıyla “kamufle edilmiş hayvan” anahtar kelimesi kullanılarak toplanmıştır. Her görüntü, nesne düzeyinde temel gerçek etiketleriyle manuel olarak açıklanmıştır. Bu veri seti, özellikle nesnelerin arka planla neredeyse tamamen birleştiği durumları içermektedir (Skurowski vd., 2018)

CHAMELEON veri setinin oluşturulmasının literatüre katkıları aşağıdaki şekilde gibi özetlenebilir:

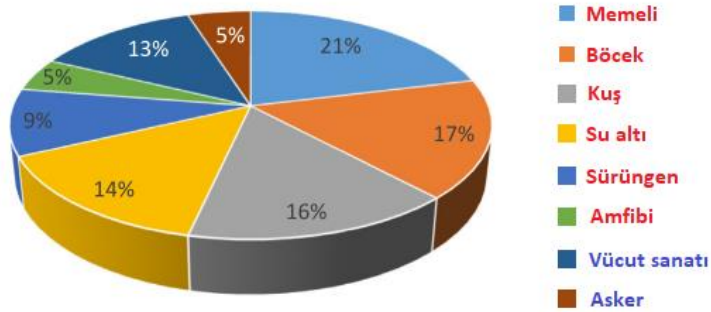
- Küçük ölçekli olmasına rağmen, yüksek kamuflej seviyeleri içerir.
- Zorlayıcı sahneler, modellerin sınırlarını test etmek için uygundur.
- Piksel seviyesinde anotasyon içerir.

CAMO Veri Seti: Le ve arkadaşları veri kümesini oluşturmak için öncelikle içinde en az bir kamufle edilmiş nesnenin bulunduğu 3000 görüntüyü "kamufle edilmiş nesneler", "kamufle edilmiş hayvanlar", "gizlenmiş nesneler", "gizli hayvanlar", "kamufle edilmiş asker", "insan vücudu boyama" vb. anahtar kelimelerle internetten topladılar. Bu aşamadan sonra, düşük çözünürlüklü veya yinelenen görüntüleri manuel olarak attılar. Sonunda 1250 görüntü elde ettiler. Daha sonra elde ettikleri bu 1250 görüntüye ayrı ayrı açıklama eklemişlerdir. Bazı örnekler Şekil 1'de karşılık gelen gerçek etiket ek açıklamalarıyla birlikte gösterilmektedir.



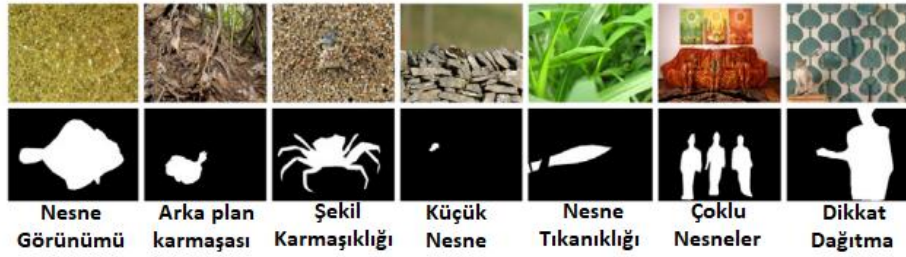
Şekil 1. CAMO kamuflaj veri setinden örnek görüntüler

Veri setini genellikle gerçek dünyada sırasıyla hayvanlara ve insanlara karşılık gelen doğal olarak kamufle edilmiş nesneler ve yapay olarak kamufle edilmiş nesneler olmak üzere iki kategoriye odaklanarak oluşturdular. Her bir kategori için yüzdelik oranlar Şekil 2' de gösterilmektedir. Doğal kamufle edilmiş nesneler kırmızı, yapay kamufle edilmiş nesneler ise mavi renkle gösterilmiştir.

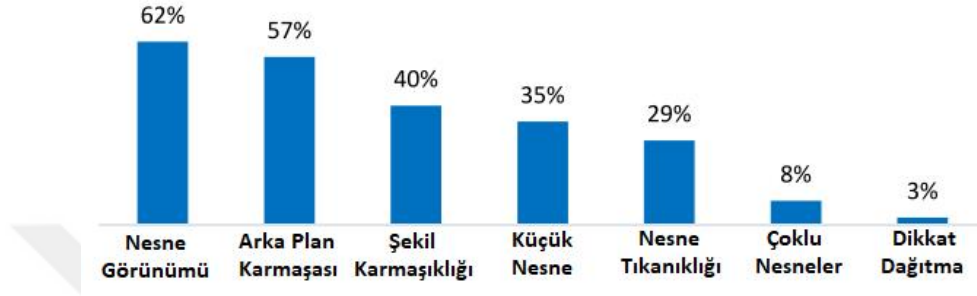


Şekil 2. CAMO veri setinde kamufle edilmiş nesnelere ait kategoriler

Şekil 3 'te görüleceği üzere CAMO veri setinde, bazı görüntülerde (%8) ayrı tek nesneler ve uzamsal olarak bağlantılı/örtüşen nesneler de dahil olmak üzere birden fazla nesnenin bulunması da dikkat çekicidir. Bu da veri setini kamuflajlı nesne bölütlenmesi için daha zorlu hale getirmektedir. CAMO veri kümesinin zorluğu, nesne görünümü, arka plan karmaşası, şekil karmaşıklığı, küçük nesne, nesne tıkanıklığı ve dikkat dağınıklığı gibi bazı özellikler nedeniyle de artmaktadır. Şekil 3'te CAMO veri setindeki bazı zorlayıcı özniteliklere örnekler ve bu özniteliklerin veri setindeki yüzdelik dağılımı gösterilmiştir.



(a) CAMO veri setindeki zorlayıcı özniteliklere örnekler



(b) CAMO veri seti üzerinde öznitelik dağılımı.

Şekil 3. CAMO veri setindeki zorlu özniteliklere ve öznitelik dağılımına örnekler

CAMO veri setinde, bazı görüntülerde (%8) ayrı tek nesneler ve uzamsal olarak bağlantılı/örtüşen nesneler de dahil olmak üzere birden fazla nesnenin bulunması da dikkat çekicidir. Bu da veri setini COD problemi için daha zorlu hale getirmektedir. CAMO veri kümesinin zorluğu, nesne görünümü, arka plan karmaşası, şekil karmaşıklığı, küçük nesne, nesne tıkanıklığı ve dikkat dağınıklığı gibi bazı özellikler nedeniyle de artmaktadır (Le vd., 2019).

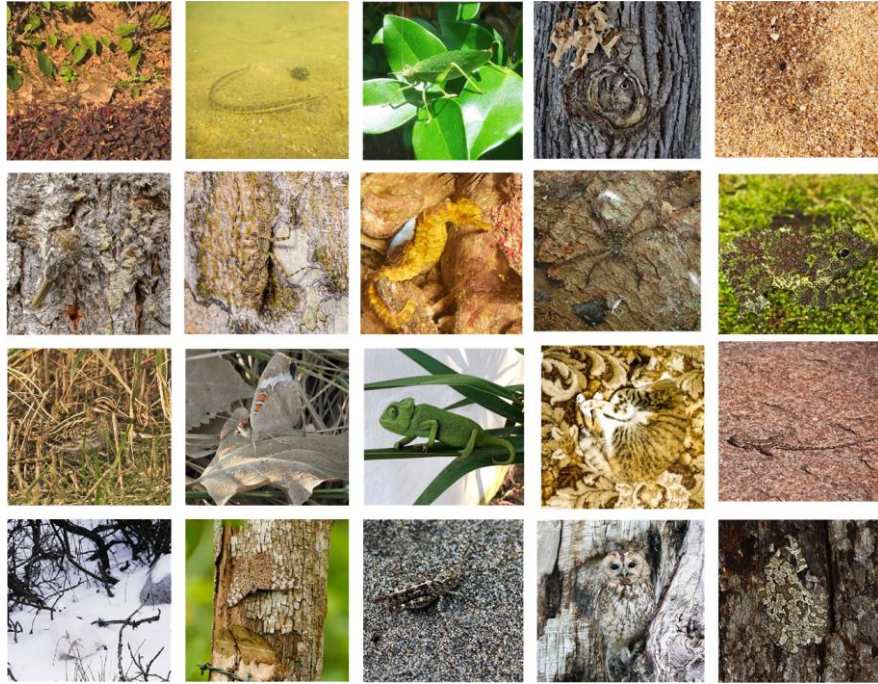
CAMO-COCO Veri Seti: CAMO veri kümesindeki herhangi bir görüntü en az bir kamufle edilmiş nesne içermektedir. Dolayısıyla kamufle edilmiş nesnelerin her zaman mevcut olduğuna dair ön bilgi vardır. Ancak gerçekçi bir senaryoda, kamufle edilmiş bir nesnenin bir görüntüde her zaman mevcut olacağının garantisi yoktur. Bu nedenle, Le ve arkadaşları kamufle edilmiş nesne görüntüleri ve kamufle edilmemiş nesne görüntülerinden oluşan CAMO-COCO adlı başka bir veri kümesini oluşturdular. Kamufle edilmiş nesne görüntüleri için CAMO veri setindeki tüm görüntüleri kullandılar ve kamufle edilmemiş nesne görüntüleri için MS-COCO veri setinden (Lin vd., 2014) ek 1250 görüntü topladılar (Le vd., 2019). CAMO-COCO veri setinin oluşturulmasının literatüre COD problemi için katkıları özetle aşağıdaki gibi özetlenebilir:

- **Çeşitlilik ve Bağlam Zenginliği:** CAMO-COCO, hem CAMO' dan gelen doğal kamufleli nesneleri hem de MS-COCO' daki çeşitli sahne ve nesne türlerini bir araya getirir.

- Gerçekçi Senaryolar: Veri setindeki görüntüler, günlük yaşamdan doğal sahnelere kadar geniş bir bağlamda, nesnelerin gerçek dünyadaki zorluklarını yansıtır.
- Karmaşıklık: Nesnelerin hem yüksek uyumlu kamuflajlı durumlarını hem de karmaşık arka planlarla olan ilişkilerini içerir.
- Bölütleme ve Tespit: Veri seti, piksel seviyesinde anotasyonlar ve nesne sınıflandırma bilgileri içerir.

CAMO-COCO veri seti, kamuflajlı ve kamuflaj olmayan nesnelerin zengin bağlamda bir araya getirilmesiyle, tespit ve bölütleme modelleri için önemli bir değerlendirme platformu sunmaktadır. Gerçekçi senaryolar ve çeşitliliği sayesinde, farklı nesne tespit algoritmalarının performansını test etmek için uygundur. Ancak, modalite eksikliği ve COCO' nun doğal kamuflaj odaklı olmaması gibi sınırlamaları bulunmaktadır. Bu veri seti, literatürdeki önemli boşlukları ele almak ve genel performansı artırmak için etkili bir kaynaktır.

COD10K Veri Seti: COD10K veri seti birden fazla fotoğrafçılık web sitesinden toplanmıştır. 10 süper sınıfı ve 78 farklı alt sınıfı kapsayan toplam 10000 görüntüden oluşmaktadır. Veri setinde 5066 gizli, 3000 arka plan ve 1934 gizli olmayan görüntü bulunmaktadır. Veri setinde günlük yaşamdan vahşi doğaya kadar geniş bir çeşitlilik bulunmaktadır. Her bir görüntü için piksel seviyesinde anotasyonlar mevcuttur (Fan vd., 2020). Şekil 4'te COD10K veri setindeki örnek kamuflaj görüntüler gösterilmiştir.

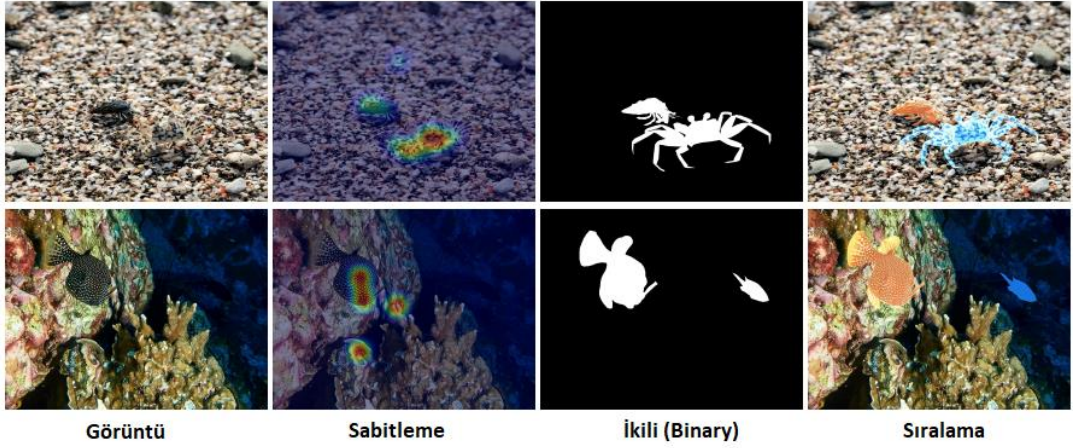


Şekil 4. COD10K kamuflaj veri setindeki sınıflara ait örnek görüntüler

COD10K veri setinin oluşturulmasının literatüre katkıları aşağıdaki şekilde özetlenebilir:

- Geniş çeşitliliği sayesinde hem eğitim hem de test için uygundur.
- Yüksek kaliteli anotasyonlar, bölütleme modellerinin hassasiyetini artırır.
- Düşük, orta ve yüksek kamuflaj seviyelerini içeren görüntüler sağlar.

NC4K Veri Seti: NC4K (Natural Camouflage 4,000) veri seti, kamuflajlı nesne tespiti ve bölütlenmesi için doğal sahnelerden derlenen 4.121 yüksek kaliteli görüntü içermektedir. Bu veri seti, doğadan toplanan gerçek dünya görüntülerini kapsar ve doğal ortamlarla yüksek uyum sağlayan nesneleri içerir. NC4K, özellikle günlük yaşam sahnelerinde ve doğal ortamlarda kamuflajlı nesnelerin tespiti için geliştirilmiştir. Şekil 5'te NC4K veri setinden örnek görüntüler verilmiştir. Geleneksel "İkili" temel gerçek yalnızca kamuflajlı nesnelerin kapsamını sağlar ve kamufle olmuş nesnelerin etiketlenmiş kısımlarını temsil etmektedir. Veri setinde bunun yanı sıra ek olarak "Sabitleme" ve "Sıralama" açıklamaları da sunulmuştur; burada ilki kamuflajlı nesneleri tespit edilebilir kılan bölgeleri keşfeder ve ikincisi kamuflaj seviyesini vurgular. "Sıralama" daki mavi renk, kamuflajın daha yüksek sırasını (daha sert) göstermektedir (Lv vd., 2021).



Şekil 5. NC4K kamuflaj veri setine ait örnek görüntüler.

NC4K veri setinin literatüre katkıları aşağıdaki şekilde özetlenebilir:

- Doğal Çeşitlilik: Veri seti, ağaçlar, bitkiler, su ortamları ve doğal araziler gibi çeşitli arka planlar içerir.

- **Piksel Seviyesinde Anotasyon:** Tüm görüntüler, nesne sınırlarını tam olarak belirlemek için piksel seviyesinde anotasyonlarla birlikte verilmiştir.
- **Kapsamlı Test Veri Seti:** NC4K, mevcut modellerin genel doğruluğunu değerlendirmek için önemli bir test veri seti olarak hizmet eder.
- **Yüksek Çözünürlük:** Görüntülerin yüksek çözünürlüğü, bölütleme ve tespit modellerinin hassasiyetini artırmak için uygundur.

NC4K, doğal ortamda kamuflajlı nesneleri tespit etmek için kullanılan önemli bir veri setidir. Yüksek çözünürlüğü, piksel seviyesinde anotasyonları ve doğal çeşitliliği ile hem tespit hem de bölütleme modellerinin performansını değerlendirmek için ideal bir kaynak sunar. NC4K'nın sınırlı kapsama alanı gelecekteki çalışmalar için genişletilme potansiyelini göstermektedir.

1.4.6. COD İçin Kullanılan Özellik Çıkarma (Feature Extraction) Yöntemleri

Özellik çıkarma, görüntü işlemede, bölütleme ve tanıma işlemlerinde bir görüntüdeki anlamlı bilgilerin temsili oluşturmak için kullanılan kritik bir tekniktir. Bu süreç, kamuflajlı nesnelerin arka planla görsel olarak uyum sağladığı durumlarda, nesne-arka plan farklılıklarını belirginleştirmeyi hedeflemektedir. Kamuflajlı görüntülerde, özellik çıkarma, nesnelerin dokusal, renk, kenar veya derinlik gibi ayırt edici özelliklerini analiz ederek tespit işlemini daha verimli hale getirir.

Kamuflajlı nesnelerin tespiti, bölütlenmesi ve tanınması için kullanılan özellik çıkarma yöntemleri, görüntüdeki anlamlı bilgileri analiz ederek işlem performansını artırmayı hedefler. COD alanında kullanılan özellik çıkarma yöntemleri, sahnenin karmaşıklığına ve veri türüne bağlı olarak çeşitlilik göstermektedir. Dokusal analiz, renk ve frekans tabanlı özellikler, kenar algılama ve derin öğrenme tabanlı özellik çıkarma yöntemleri, bu alandaki zorlukların üstesinden gelmek için kritik araçlardır. Hibrit stratejiler, bu yöntemlerin güçlü yanlarını birleştirerek daha etkili çözümler sunmaktadır. Bu yöntemler, literatürdeki mevcut boşlukları doldurmak ve modellerin doğruluğunu artırmak için kritik öneme sahiptir. Literatürde yer alan bu yöntemlerin gelecekte daha geniş veri setleri ve modalitelerle optimize edilmesi beklenmektedir. Literatürde COD problemi için kullanılan özellik çıkarma yöntemlerini aşağıdaki şekilde sıralanabilir.

Doku Tabanlı Özellik Çıkarma Yöntemleri: Dokusal özellik çıkarma, bir görüntüdeki pikseller arasındaki ilişkileri analiz ederek yüzey desenleri, yapılar ve düzenlilik gibi karakteristik bilgileri elde etmeyi amaçlar. Kamufajlı görüntülerde dokusal özellikler, nesne ve arka plan arasındaki doku farklılıklarını vurgulamak için kritik öneme sahiptir. Bu yöntem, özellikle arka planla yüksek derecede uyum sağlayan nesnelerin tespit edilmesinde etkili sonuçlar sunar. Dokusal özellikler, görüntünün yerel ve küresel düzeydeki yapısal düzenini tanımlamak için kullanılır. Dokusal özellikler kamufajlı nesnelerin tespiti sırasında:

- Nesne ve arka plan arasındaki farkların belirginleştirilmesini sağlar.
- Düşük kontrastlı alanlarda ek bilgi sunar.
- Geometrik desenlerin, simetri veya asimetri gibi yapısal detayların çıkarılmasına olanak tanır.

Literatürde dokusal özellik çıkarma yöntemlerinin önemlileri Gabor Filtreleri, Yerel İkili Örüntüler (LBP), Gri-Seviye Eş-Oluşum Matrisi(GLCM) olarak aşağıdaki şekilde özetlenmiştir:

Gabor Filtreleri: Gabor filtreleri, görüntüdeki dokusal özellikleri belirlemek için sıklık ve yönelim tabanlı analizler yapar. Filtre, belirli bir frekansta ve yönelimde dokusal bilgileri çıkarmak için kullanılmaktadır. Doku yoğunluğu, yönelim ve kontrast farklılıklarını vurgulamaktadır. Kamufajlı nesnelerin çevreye uyum sağladığı sahnelerde etkili bir araçtır. Gabor filtrelerinin avantajı yönelim duyarlılığı ve doku yoğunluğu ölçümü için uygun olmasıdır. Dezavantajı ise yüksek hesaplama maliyeti ve çoklu filtre gereksinimine ihtiyaç duyması olarak sıralanabilir. Literatürde Gabor filtreleri ile yapılmış birçok çalışma mevcuttur, bunlardan birkaçı aşağıdaki şekilde özetlenmiştir.

Jain ve arkadaşları farklı yönelim ve frekanslarda çalışan bir Gabor filtre bankası kullanarak karmaşık arka planlarda nesne tespiti gerçekleştirmiştir. Çalışmada, her piksel için lokal enerji ölçümleri hesaplanarak bir kümeleme algoritmasıyla bölütleme yapılmıştır. Gabor filtrelerinin infrared görüntülerde nesne sınırlarını belirlemede etkili olduğunu göstermişlerdir (Jain vd., 1997). Braithwaite ve Bhanu, hiyerarşik Gabor Filtreleriyle infrared görüntülerde nesne tespiti gerçekleştirmiştir. Filtreler, uzun konturları ve periyodik desenleri yakalamak için optimize edilmiştir. Çalışmalarının literatüre katkısı küçük bant genişlikli, ayrık olmayan Gabor filtreleri, nesne ve arka plan ayrımını iyileştirmesidir (Braithwaite & Bhanu, 1994). Popa ve arkadaşları, log-Gabor filtrelerini yapay sinir

ağlarıyla birleştirilerek kamuflajlı askeri hedefleri tespit etmiştir. Görüntülerin frekans içeriği otomatik olarak işlenmiş ve yüksek doğruluk oranları elde edilmiştir. Çalışmanın literatüre katkısı çok frekanslı log-Gabor filtreleriyle tespit hassasiyeti artırılmış olmasıdır (Popa vd., 2005). Le ve arkadaşları, sonar görüntülerinde mayın benzeri nesneleri tespit etmek için Gabor tabanlı derin bir sinir ağı geliştirmiştir. Çalışmada Gabor filtreleri, sonar görüntülerde farklı boyutlardaki nesneleri tespit etmeyi iyileştirmiştir (Le vd., 2020). Hu ve arkadaşları, askeri nesneleri tespit etmek için optimize edilmiş Gabor Filtreleri ve derin özellik piramit ağlarını (HU-FPN) birleştiren bir yöntem geliştirmiştir. Çalışma ile küçük nesnelerin tespitinde doğruluğun artırıldığı görülmüştür (Hu vd., 2019).

Gabor filtreleri, kamuflajlı nesnelerin tespiti için doku, frekans ve yönelim farklılıklarını analiz etmede güçlü bir araçtır. Literatürdeki bu çalışmalar, Gabor filtrelerinin geleneksel yöntemlerden derin öğrenme modellerine kadar geniş bir uygulama alanında kullanıldığını ve farklı modalitelerde etkili olduğunu göstermektedir. Özellikle sonar, infrared ve askeri görüntülerde nesne tespiti için kritik bir rol oynarlar. Bu yöntemler, gelecekte daha büyük veri setleri ve hibrit modellerle geliştirilebilir.

Yerel İkili Örüntüler (Local Binary Patterns, LBP): LBP, kamuflajlı nesnelerin tespitinde kullanılan etkili bir dokusal özellik çıkarma yöntemidir. LBP, kamuflajlı nesnelerin doku ve renk uyumu nedeniyle arka planla olan yüksek benzerliklerini analiz ederek, ayırt edici detayları tespit edebilir. LBP ile birçok çalışma yapılmıştır, bunlardan bazıları aşağıda açıklanmıştır:

Bayram ve Nabiye, kamuflajlı nesneleri sınıflandırmak için LBP tabanlı dokusal özellik çıkarma ve farklı makine öğrenmesi yöntemlerini karşılaştırmıştır. Bu çalışmada, LBP ile çıkarılan dokusal özellikler, Yapay Sinir Ağları (ANN), K-en yakın komşu (KNN) ve Destek Vektör Makineleri (SVM) kullanılarak sınıflandırılmıştır. En yüksek sınıflandırma başarı oranını LBP ve ANN yöntemlerinin birlikte kullanılmasıyla %92 olarak elde etmişlerdir (Bayram ve Nabiye, 2021). Lv ve arkadaşları, LBP'nin kamuflajlı nesne tespitinde temel dokusal özellik çıkarma yöntemlerinden biri olarak kullanılabileceğini göstermiştir. Bu çalışmada, LBP ile çıkarılan özellikler derin öğrenme modellerine entegre edilmiştir. Çalışmada kamuflajlı nesnelerin belirgin olmayan sınırları ve düşük kontrastlı dokuları başarıyla analiz edilmiştir (Lv vd., 2023). Petránek ve arkadaşları, LBP kullanarak hızlı ve düşük maliyetli bir nesne tespit yöntemi geliştirmiştir. Bu çalışma, özellikle düşük hesaplama maliyeti gerektiren sistemlerde LBP'nin etkinliğini göstermiştir. Bu çalışmada LBP'nin kontrast, parlaklık ve ölçek değişikliklerine karşı dayanıklı olduğu ve düşük

hesaplama gücüne ihtiyaç duyduğu vurgulanmıştır (Petránek vd., 2015). Nguyen ve arkadaşları, LBP'nin daha kompakt ve ayırt edici bir varyantı olan Non-Redundant Local Binary Patterns (NRLBP) yöntemini geliştirmiştir. Bu yöntem, kamuflajlı nesnelerin dokusal özelliklerini daha iyi anlamak için tasarlanmıştır. Çalışmada NRLBP, klasik LBP'ye kıyasla daha az hesaplama maliyeti ve daha yüksek tespit doğruluğu sunmuştur (Nguyen vd., 2010).

LBP, kamuflajlı nesnelerin dokusal özelliklerini analiz etmek için etkili ve düşük maliyetli bir yöntemdir. Literatürdeki bu çalışmalar, LBP'nin kamuflajlı nesnelerin düşük kontrastlı ve dokusal açıdan uyumlu ortamlarında bile başarılı sonuçlar verdiğini göstermektedir. Ayrıca, LBP tabanlı varyantların (örnek: NRLBP, LTP) geliştirilmesi, dokusal analizdeki doğruluğu ve etkinliği artırmaktadır. Bu yöntemler, gelecekteki kamuflajlı nesne tespiti çalışmalarında da önemli bir rol oynayacaktır.

Gri-Seviye Eş-oluşum Matrisi (Gray-Level Co-occurrence Matrix, GLCM): GLCM, kamuflajlı nesnelerin tespiti için yaygın olarak kullanılan bir doku analiz yöntemi olup, pikseller arasındaki mekânsal ilişkileri analiz eder. GLCM, kontrast, homojenlik, enerji ve korelasyon gibi dokusal özellikleri çıkararak nesne ve arka plan arasındaki farklılıkları anlamlandırır. Literatürde GLCM ile yapılmış birçok çalışma bulunmaktadır. Bunların önemli bir kısmı aşağıdaki şekilde açıklanmıştır:

Htoo ve Sein, GLCM tabanlı bir dokusal analiz yöntemi kullanarak hareketli hedeflerin tespiti için bir sistem geliştirmiştir. Çalışmada, doku özellikleri dört farklı yönelimde (0° , 45° , 90° , 135°) çıkarılmış ve SVM sınıflandırıcısı kullanılmıştır. Çalışmada hareketli hedeflerin yüksek doğruluk oranıyla sınıflandırılabilirdiği gösterilmiştir (Htoo & Sein, 2018). Rao ve arkadaşları, dalgacık dönüşümleri ve GLCM ile kamuflajlı hedeflerin tespiti için bir yöntem geliştirmiştir. Çalışmada, GLCM özellikleri kullanılarak hedef nesneler vurgulanmıştır. Çalışmada dalga katsayılarıyla birlikte kullanılan GLCM, düşük kontrastlı sahnelerde bile yüksek performans göstermiştir (Rao vd., 2016). Miao, paralel GLCM kullanarak dokusal özelliklerin tespiti için yeni bir yöntem önermiştir. Çalışmada, piksel haritaları ve paralel işlem yöntemleriyle hızlı analizler gerçekleştirilmiştir. Bu çalışmada paralel işlemenin, dokusal özelliklerin çıkarımında daha yüksek verimlilik sağladığı görülmüştür (Miao, 2011). Pare ve arkadaşları, çok seviyeli eşikleme tekniğiyle GLCM kullanarak renkli görüntü bölütleme gerçekleştirmiştir. Çalışmada, doğal ve uydu görüntüleri üzerinde başarılı bölütleme sonuçları elde edilmiştir. Çalışmada GLCM tabanlı eşikleme, yüksek doğruluk ve düşük hata oranları sunulmuştur (Pare vd., 2017).

GLCM, kamuflajlı nesnelerin dokusal analizinde oldukça etkili bir yöntemdir. GLCM tabanlı analizler, hareketli hedef tespitinden kamuflaj tasarımına kadar geniş bir uygulama yelpazesi sunmaktadır. Literatürdeki bu çalışmalar, GLCM' nin düşük kontrastlı ve karmaşık sahnelerde nesne tespitinde oldukça yüksek başarı oranlarına ulaştığını göstermektedir. Gelecekte, paralel işleme ve derin öğrenme ile birleştirilmiş GLCM yöntemlerinin daha yüksek performans sağlaması beklenmektedir.

Kenar Tabanlı Özellik Çıkarma Yöntemleri: Kenar özellikleri, görüntüdeki nesne sınırlarını ve geometrik farklılıkları belirlemek için kullanılır. Kamuflajlı görüntülerde, nesne ile arka plan arasındaki ayrımı güçlendirmek için kenar algılama yöntemleri kritik bir öneme sahiptir. Özellikle, kamuflajlı nesnelerin belirgin olmayan sınırlarını vurgulamak için çeşitli geleneksel ve modern kenar algılama teknikleri uygulanmaktadır. Aşağıda, bu teknikler ayrıntılı bir şekilde açıklanmış ve literatürdeki ilgili çalışmalardan bahsedilmiştir.

Canny Kenar Algılama Yöntemi: Canny Kenar Algılama (CKA), çok aşamalı bir algoritma ile görüntüdeki belirgin kenarları tespit eder (Canny, 1986). Bu algoritma, özellikle düşük kontrastlı sahnelerde nesne sınırlarını hassas bir şekilde belirlemek için tasarlanmıştır. Literatürde bu yöntem kullanılarak yapılmış birçok çalışma mevcuttur.

Ma ve arkadaşları, Canny algoritmasını iyileştirerek yüksek hızlı demiryolu ekipmanlarının kenar tespiti için kullandı. Yaptıkları çalışmada adaptif filtreleme ile tuz-biber gürültüsü temizlenerek kenar detayları korundu. Çalışmada geliştirilmiş algoritma, gürültüden kaynaklanan sahte kenarları azaltmış ve nesne kenarlarının daha hassas bir şekilde tespit edilmesini sağlamıştır (Ma vd., 2017). Song ve arkadaşları, düşük irtifada dron ve helikopter gibi hedeflerin tanınması için Canny algoritmasını iyileştirerek aktif kontur modelleriyle birleştirdi. Çalışmada geliştirilen algoritma, farklı çevresel koşullara uyum sağlayarak geleneksel yöntemlere kıyasla hedef konturlarını daha etkili bir şekilde çıkarmıştır (Song vd., 2023). Yu ve arkadaşları, morfolojik işlemler ve adaptif eşikleme yöntemlerini entegre ederek Canny algoritmasını tarımsal ürünlerin kenar tespiti için geliştirdi. Çalışmada iyileştirilmiş algoritma, ışık etkilerinden kaynaklanan sahte konturları önleyerek daha hassas kenar tespiti sağlamıştır (Yu vd., 2021).

Canny algoritması, kamuflajlı görüntülerde kenar tespiti için sıkça kullanılan bir yöntemdir. Literatürde, gürültü azaltma, adaptif eşikleme ve aktif kontur modelleri gibi iyileştirmelerle, algoritmanın performansı artırılmıştır. Bu çalışmalar, Canny algoritmasının çeşitli uygulama alanlarında etkili bir şekilde kullanılabileceğini göstermektedir.

Sobel ve Prewitt Kenar Algılama Yöntemleri: Sobel ve Prewitt kenar algılama yöntemleri, görüntülerdeki nesne sınırlarını belirginleştirmek için yaygın olarak kullanılan gradyan tabanlı algoritmalar. Kamuflejli görüntüler üzerinde bu algoritmalar, nesne ve arka plan arasındaki farklılıkları vurgulamak için etkili araçlar olarak kullanılmıştır. Literatürde Sobel ve Prewitt algoritmaları ile yapılmış birçok çalışma mevcuttur. Bu çalışmalardan bir kısmı aşağıda özetlenmiştir.

Yadav ve Bharathi, Sobel ve Prewitt algoritmalarını karşılaştırarak, Prewitt'in kenar tespitinde daha yüksek doğruluk sunduğunu bulmuştur. Prewitt yöntemi %85 doğruluk sağlarken, Sobel yöntemi %71 doğruluk elde etmiştir. Çalışmada, Prewitt algoritmasının daha hassas kenar tespiti yaptığı tespit edilmiştir (Yadav & Bharathi, 2022). Wanto ve arkadaşları, Sobel+Prewitt ve Roberts+Canny algoritmalarının kombinasyonunu karşılaştırmıştır. Passion çiçekleri üzerinde yapılan çalışmada, Roberts+Canny kombinasyonu %92,84 ile daha yüksek doğruluk oranı sunmuştur. Sobel+Prewitt kombinasyonu ile de %68,75 doğruluk oranı elde edilmiştir. Çalışmada kombine algoritmaların, tekil algoritmalara göre daha iyi performans gösterdiği belirtilmiştir (Wanto vd., 2021). Jena ve Mishra, Sobel ve Prewitt operatörlerinin kombinasyonu ile kenar tespit performansını artıran bir yöntem geliştirmiştir. Çok ölçekli analizler hem küçük hem de büyük ölçekli nesnelerin sınırlarının hassasiyetle belirlenmesini sağlamıştır. Bu çalışmada iki operatörün birleştirilmesi yaklaşımı, her iki operatörün bireysel kullanımından daha iyi sonuçlar vermiştir (Jena & Mishra, 2015). Ahmed Sobel, Prewitt ve Canny algoritmalarının performanslarını farklı veri setleri üzerinde karşılaştırmıştır. Prewitt'in parlaklık değişimlerine daha duyarlı olduğu, ancak Sobel'in yüksek gradyanlarda daha etkili sonuçlar verdiği gözlemlenmiştir. Çalışmada Prewitt, düşük kontrastlı sahnelerde üstünlük sağlarken Sobel, yüksek kontrastlı bölgelerde daha iyi olduğu tespit edilmiştir (Ahmed, 2018). Deng ve arkadaşları, Sobel operatörünü iyileştirerek Prewitt ve diğer yöntemlerle birleştirmiştir. Dalga dönüşümleri ve diğer gradyan tabanlı yöntemlerle kenar tespit performansı artırılmıştır. Çalışmada geliştirilmiş Sobel algoritması, Prewitt ile birlikte daha hassas ve net kenar tespiti sağlamıştır (Deng vd., 2011).

Sobel ve Prewitt algoritmaları, kamuflejli nesnelerin tespitinde önemli araçlardır. Literatürde, bu yöntemlerin kombinasyonu ve diğer algoritmalarla birlikte kullanımı hem hassasiyeti artırmış hem de farklı sahnelerde performansı iyileştirmiştir. Bu çalışmalar, Sobel ve Prewitt yöntemlerinin zayıf yönlerini güçlendirerek daha etkin sonuçlar elde etmeyi sağlamaktadır.

Sınır G d ml  (Boundary-Guided)  zellikler: Sınır g d ml   zellik  ıkarma, kamuflajlı nesnelerin tespitinde, nesne ve arka plan sınırlarının hassas bir  ekilde ayrılması i in kullanılan modern bir yaklaşımdır. Sınır g d ml   zellik  ıkarma ile literat rde bir ok  alı ma mevcuttur.

Sun ve arkadaşları, sınır g d ml  bir a  (Boundary-Guided Network, BGNet) geli tirerek, kenar bilgilerini temsil   renme s re lerine dahil etti. Bu y ntem, nesnelerin daha do ru sınırlarla tespit edilmesini sa ladı (Sun vd., 2022). Li ve arkadaşları, sınır g d ml   oklu d zey  zellik f zyonu a ı (BGFM-Net) geli tirerek, piksel dengesizlik problemini   zmeyi ama ladı.  apraz d zey f zyon mod l ,  zelliklerin adaptif kanal a ırlıklarıyla birle tirilmesini sa ladı.  alı mada BGFM-Net ile,    kıyaslama (benchmark) veri setinde di er COD y ntemlerinden daha ba arılı sonu lar elde edildi (Li vd., 2024). Jia ve arkadaşları, bulanık sınır bilgilerini kullanarak d   k g ven b lgelere odaklanan bir model geli tirdi.  alı mada FBG-Net, mevcut y ntemlerden daha d   k belirsizlikle sınır tespiti sa lamı tır (Jia vd., 2024). Li ve arkadaşları, y ksek d zey  zelliklerin yerelle tirme   renimini, d   k d zey  zelliklerin ise sınır   renimini y nlendirdi i BLR-Net modelini geli tirdi.  alı mada BLR-Net, d rt kıyaslama veri setinde mevcut y ntemlerden daha y ksek performans sa lamı tır (Li vd., 2024). Sınır g d ml   zellik  ıkarma y ntemleri, kamuflajlı nesnelerin zayıf kontrastlı sınırlarını daha do ru bir  ekilde belirlemek i in etkili ara lardır. Literat rde, bu y ntemlerin kullanımı nesne tespit performansını artırmı  ve  e itli veri setlerinde di er y ntemlerden  st n sonu lar sa lamı tır.

Derin   renme Tabanlı  zellikler: Derin   renme tabanlı y ntemler, kamuflajlı nesnelerin tespitinde y ksek do ruluk sa layan, karma ık yapıları ve derin  zellik  ıkarma tekniklerini birle tiren g  l  ara lardır. A a ıda, bu y ntemlerin kullanıldı ı bazı  nemli  alı malara dair ayrıntılar verilmi tir.

Ren ve arkadaşları, derin tekst r farkındalı ı mod lleri kullanarak kamuflajlı nesnelerin tespiti i in bir model geli tirmi tir. Mod l, nesne ve arka plan arasındaki ince tekst r farklılıklarını vurgulamak i in  zelle tirilmi tir.  alı ma, COD veri setlerinde mevcut y ntemlerden daha y ksek do ruluk sa lamı tır (Ren vd., 2021). Zhai ve arkadaşları, Mutual Graph Learning (MGL) adında bir model geli tirerek g r nt y  iki g rev-odaklı  zellik haritasına ayırmı tır. Hedefleri kaba konumlaması ve sınır detaylarının hassas  ıkarımıdır.  alı mada CHAMELEON, CAMO ve COD10K veri setlerinde  st n performans elde edilmi tir (Zhai vd., 2022). Wang ve arkadaşları, insan g rsel sistemini sim le ederek ikili dallı  zellik  ıkarma ve  apraz rafine birle tirme mod lleri i eren bir

model tasarlamıştır. Çalışmada üç farklı veri setinde mevcut yöntemlerden daha yüksek doğruluk ve F1 puanı sağlanmıştır (Wang vd., 2021). Cong ve arkadaşları, RGB ve frekans bilgilerini birleştirerek çok seviyeli bir frekans algılama ağı geliştirmiştir. Model, frekans yönlendirmesi ile nesne ve arka planın ayrımını iyileştirmiştir. Çalışma COD10K, CAMO ve CHAMELEON veri setlerinde üstün performans göstermiştir (Cong vd., 2023). Fan ve arkadaşları, SINet modeliyle, kamuflajlı nesnelerin yüksek doğrulukla tespiti için çoklu ölçekli birleştirme teknikleri geliştirmiştir. Model, özellikle nesne sınırlarını hassas bir şekilde tanımlamada etkilidir. Çalışmada COD veri setlerinde diğer yöntemlerden daha iyi performans elde edilmiştir (Fan vd., 2020). Lin ve arkadaşları, yüksek frekanslı bilgiyle karmaşıklık oluşturan dokuları bastıran bir frekans odaklı model geliştirmiştir. Çalışma frekans analizi, nesne sınırlarını iyileştiren bir kayıp fonksiyonu ile güçlendirilmiştir (Lin vd., 2022).

Bu çalışmalar, derin öğrenme tabanlı özellik çıkarma yöntemlerinin kamuflajlı nesne tespiti için kritik olduğunu ve özellikle sınır, frekans ve derinlik bilgilerini birleştirerek tespit doğruluğunu artırdığını göstermektedir.

1.5. COD İçin Literatürde Kullanılan Performans Ölçütleri

Kamuflajlı görüntülerde nesne bölütlemesi ve tanıma, nesne ile arka plan arasındaki görsel benzerlikten dolayı oldukça zorlu bir problemdir. Bu zorluk, düşük kontrast, bulanık sınırlar ve karmaşık sahne yapılarından kaynaklanır. Kamuflajlı görüntülerin bölütlenmesi ve tanınmasında kullanılan performans ölçütleri, modelin doğruluğunu, hassasiyetini ve genelleştirilebilirliğini değerlendirmek için kritik bir rol oynar.

Performans ölçütleri, modellerin bu zorlukları ne kadar iyi çözebildiğini değerlendirmek için kritik bir araçtır. Her ölçüt, modelin farklı bir yönünü analiz eder ve modellerin nesne tespiti doğruluğunu, sınırlayıcı kutuların veya bölütleme maskelerinin hassasiyetini ve genelleştirilebilirliğini değerlendirir. Bu ölçütler, özellikle nesne tespitinde düşük kontrastlı, bulanık ve karmaşık sahnelerde performansı değerlendirmek için gereklidir.

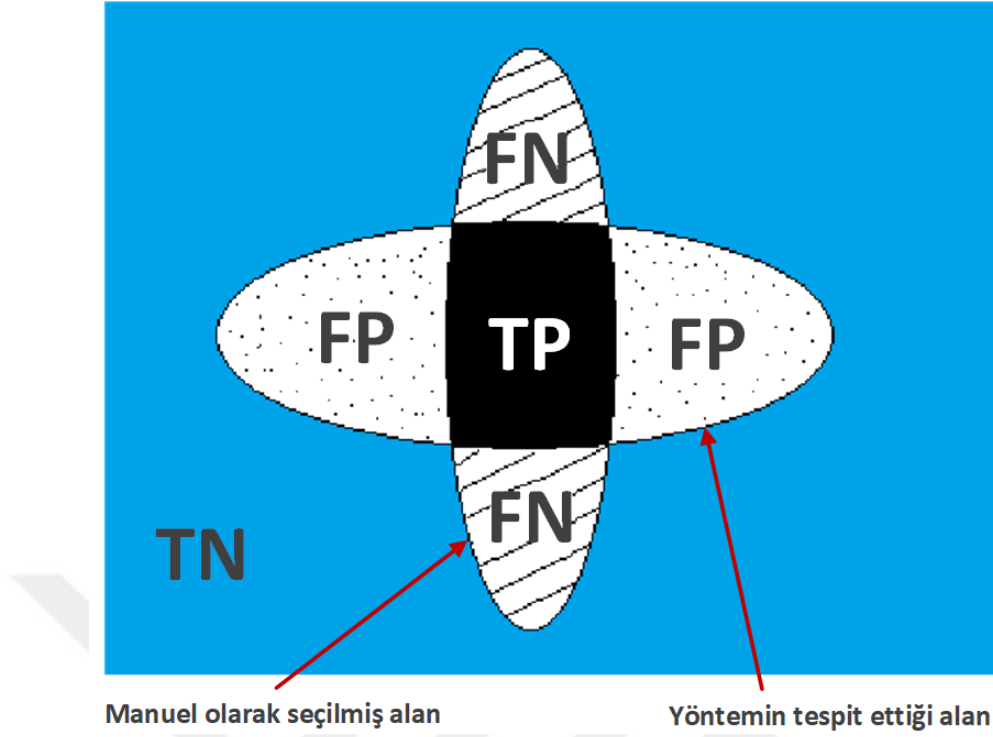
Kamuflajlı görüntülerde kullanılan performans ölçütleri, modelin çıktısının ne kadar doğru olduğunu hem matematiksel hem de görsel olarak değerlendirme imkânı sağlar. Performans ölçütleri, kamuflajlı görüntülerdeki zorlukların üstesinden gelmek için

modellerin güçlü ve zayıf yönlerini belirler. Bu metrikler aşağıdaki nedenlerden dolayı önemlidir:

- Doğru Tahminlerin Değerlendirilmesi: Gerçek nesne ile modelin tahmin ettiği nesne arasındaki örtüşmeyi ölçerek doğru tespit kapasitesini belirler.
- Yanlış Pozitif ve Yanlış Negatiflerin Azaltılması: Yanlış tespit oranlarını analiz ederek, modelin hata yapma eğilimini azaltmayı hedefler.
- Genelleştirilebilirlik: Modelin farklı veri setlerinde nasıl performans gösterdiğini anlamak için kullanılır.
- İyileştirme için Geri Bildirim: Modellerin performansını artırmak için hangi alanlarda geliştirme yapılması gerektiğine dair bilgi sağlar.

Kamuflajlı görüntülerin bölütlenmesi ve tanıma görevlerinde kullanılan performans ölçütleri, modellerin zorlu sahnelerdeki etkinliğini anlamak için hayati önem taşır. Bu ölçütler, modelin doğruluğunu, duyarlılığını ve genelleştirilebilirliğini değerlendirerek, geliştirme sürecine ışık tutmaktadır. Literatürde kullanılan birçok performans değerlendirme ölçütleri bulunmaktadır. Bunlardan önemli olan ve sık kullanılanlar aşağıdaki belirtilmiştir.

Performans ölçütlerinin kullanılması için öncelikle yöntemin elde ettiği bölütlenmiş ya da tespit edilen bölge ile daha önceden manuel seçilmiş olması gereken bölgenin karşılaştırılması için gerekli alanların Şekil 6'daki gibi tanımlanması gereklidir. Burada TP; kullanılan yöntemin tespit ettiği alan ile gerçekte olması gereken alanın eşleştiği alanı ifade eder yani doğru bir şekilde tespit edilen alanı ifade eder. FP; kullanılan yöntemin tespit ettiği fakat manuel olarak gerçekte seçilmemiş alanı ifade eder, diğer bir deyişle nesne olarak tespit edilen fakat arka plana ait olan alanı ifade eder. TN; kullanılan yöntem ile elde edilmiş alan ile manuel olarak seçilmiş alanların eşleşmediği alanları ifade eder. FN; manuel olarak seçilmiş fakat yöntemin tespit edemediği alanları ifade eder. Bu tanımlar kullanılarak aşağıdaki performans metrikleri çıkarılmaktadır.

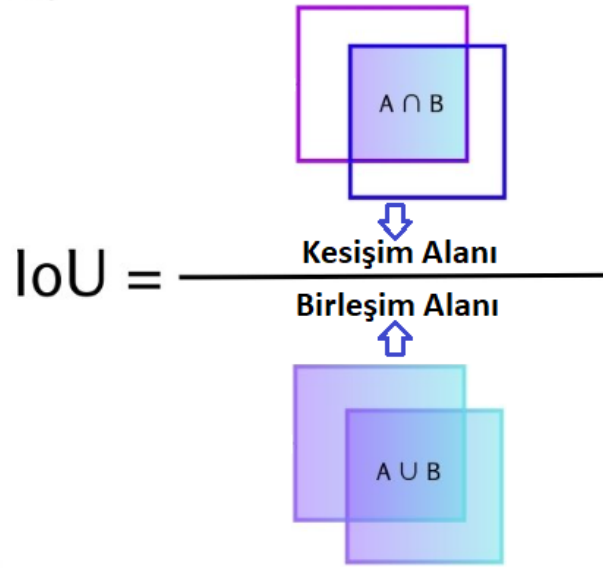


Şekil 6. Kullanılan yöntemin tespit ettiği ile olması gereken alanların doğruluğunun karşılaştırılması

1.5.1. Birleşim Üzerindeki Kesişim (Intersection over Union, IoU)

Birleşim Üzerindeki Kesişim (IoU) görüntü işleme ve nesne tespiti alanında kullanılan bir performans ölçütüdür. IoU, modelin tahmin ettiği bölgenin (örneğin, bir nesne maskesi) gerçek bölgeyle ne kadar örtüştüğünü değerlendirmek için kullanılır. Bu metrik hem bölütleme hem de nesne tanıma görevlerinde doğruluğun önemli bir göstergesidir. IoU değeri, modelin hem doğru tahmin oranını hem de yanlış pozitif ve yanlış negatif sonuçlarını dikkate almaktadır.

Şekil 7’de gösterildiği üzere, IoU tahmin edilen bölgenin gerçek bölgeyle kesişim alanının, bu iki bölgenin birleşim alanına oranıdır. Bu oran, 0 ile 1 arasında bir değer almaktadır. 1 değeri tahmin edilen bölge ile gerçek bölgenin tam olarak örtüştüğü anlamına gelir. 0 değeri ise tahmin edilen bölgenin gerçek bölgeyle hiçbir kesişimin olmadığı anlamına gelmektedir. Şekil 7’de gösterilen temsili formülde A tahmin edilen bölge, B gerçek bölge $|A \cap B|$ tahmin edilen ve gerçek bölgenin kesişim alanını, $|A \cup B|$ tahmin edilen ve gerçek bölgenin birleşim alanını ifade etmektedir.

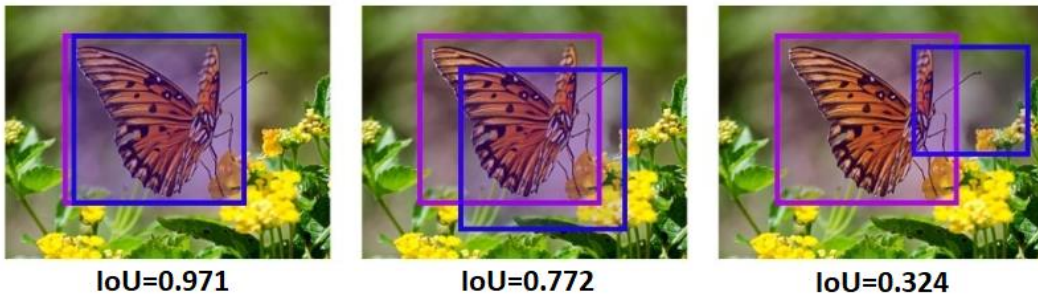


Şekil 7. IoU için Birleşim Alanına bölünen Kesişim Alanının gösterilmesi

IoU'nun denklemi (1) eşitliğinde gösterildiği gibidir.

$$IoU = \frac{TP}{TP + FP + FN} \quad (1)$$

Şekil 7'de gösterilen temsili formül 0 ile 1 arasında bir değer üretmektedir. Burada 0 hiçbir örtüşme olmadığını, 1 ise tahmin edilen kutu ile temel gerçek sınırlayıcı kutular arasında mükemmel bir eşleşme olduğunu göstermektedir. Şekil 8'de bu IoU için bir eşleşme görüntü örneği gösterilmiştir.



Şekil 8. Farklı IoU eşik değerlerinde nesne algılama örnekleri

1.5.2. Kesinlik (Precision)

Kesinlik, nesne tespiti ve bölütleme alanında, bir modelin tespit ettiği nesnelerin ne kadarının doğru olduğunu değerlendiren bir metriktir. Bu metrik, modelin yanlış pozitifleri (FP) azaltmadaki başarısını analiz etmek için kritik bir role sahiptir. Kamuflajlı görüntülerde,

Kesinlik, arka planla karışan nesnelerin doğru bir şekilde tespit edilip edilmediğini anlamak için önemli bir metriktir. Kesinlik, (2) numaralı eşitlikte gösterildiği gibi, doğru pozitif tahminlerin (TP), toplam tahmin edilen pozitif tahminlere (TP+FP) oranıdır.

$$Kesinlik = \frac{TP}{TP + FP} \quad (2)$$

Kesinlik metriğinin avantajları; modelin yanlış pozitiflere duyarlılığını ölçer ve hatalı tahminlerin azaltılmasını teşvik eder. Bunun yanı sıra özellikle düşük kontrastlı ve karmaşık sahnelerde tahmin edilen nesnelerin ya da alanların doğruluğunu analiz etmek için uygundur. Sadece Kesinlik metriğine odaklanmak, modelin kaçırdığı nesneler (FN) hakkında bilgi vermez. Bu nedenle Duyarlılık (Recall) ile birlikte değerlendirilir. Kesinlik, çok az sayıda görüntünün bulunduğu dengesiz veri setlerinde yanlış bir izlenim verebilir.

Kesinlik, COD ve bölütlemede doğru tahminlerin oranını ölçen kritik bir metriktir. Modelin doğruluk seviyesini ve yanlış pozitiflere (FP) karşı hassasiyetini anlamak için kullanılır. Precision, genellikle Duyarlılık ve F1-skoru gibi diğer metriklerle birlikte değerlendirilir ve düşük kontrastlı, karmaşık sahnelerde doğru nesne tespiti için önemli bir ölçüt sağlar.

1.5.3. Duyarlılık (Recall)

Duyarlılık, bir modelin tüm gerçek nesneleri ne kadar doğru tespit edebildiğini ölçen bir metriktir. Bölütleme ve tanıma görevlerinde Duyarlılık, doğru pozitiflerin (TP) toplam gerçek pozitiflere (TP + FN) oranını ifade eder ve şu formülle tanımlanır:

$$Duyarlılık = \frac{TP}{TP + FN} \quad (3)$$

Duyarlılık, özellikle modelin gerçek nesneleri tanıma kabiliyetini analiz etmek için kullanılır ve kamuflajlı nesnelerin düşük kontrast ve bulanık sınırları nedeniyle önemli bir değerlendirme ölçütüdür.

1.5.4. F1-Score

F1-Score, bir modelin doğruluğunu ve eksiksizliğini bir arada değerlendiren bir performans ölçütüdür. F1-Score, Kesinlik ve Duyarlılık arasında harmonik bir ortalama

olarak, özellikle dengesiz veri setlerinde modelin genel performansını ölçmek için kullanılır. Kamuflajlı nesnelerin bölütlenmesi ve tanıma görevlerinde, F1-Score, modelin hem yanlış pozitifleri (FP) azaltma hem de kaçırılan nesneleri tespit etme kapasitesini anlamak için kritik bir metriktir. F1-Score, 0 ile 1 arasında bir değer alır. 1, mükemmel bir model performansını, 0 ise başarısız bir modeli ifade eder. F1-scor'un genel tanımı aşağıdaki gibidir:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

F1-Score, kamuflajlı görüntülerde, görüntü sayısının azlığı ve arka planın yoğunluğu gibi durumlarda Kesinlik ve Duyarlılık arasında denge sağlar ve hem Kesinlik hem de Duyarlılık metriğini hesaba kattığı için modelin eksik tespit veya yanlış pozitif oranına duyarlıdır.

F1-Score, kamuflajlı nesne bölütlemesi ve tanıma problemlerinde genel model başarısını ölçmek için etkili bir metrik olarak kullanılmaktadır. Özellikle Kesinlik ve Duyarlılık arasında denge sağladığı için, dengesiz veri setlerinde veya düşük kontrastlı nesnelerde performans analizi yapmak için kritik bir araçtır.

1.5.5. Doğruluk (Accuracy)

Doğruluk (Dogan et al., 2021), bir modelin yaptığı toplam tahminlerden ne kadarının doğru olduğunu ölçen temel bir performans metriğidir. Kamuflajlı nesnelerin bölütlenmesi ve tanıma problemlerinde, Doğruluk, modelin tahmin ettiği doğru pozitif (TP) ve doğru negatif (TN) sonuçların, toplam tahmin edilen değerler içindeki oranını hesaplar. Doğruluk şu şekilde tanımlanır:

$$Doğruluk = \frac{TP + TN}{TP + FP + TN + FN} \quad (5)$$

Doğruluk, modelin genel performansını tek bir metrik ile analiz eder ve diğer metriklere göre daha basit bir hesaplama sürecine sahiptir. Doğruluk, kamuflajlı nesne tespiti ve bölütlenmesi gibi karmaşık görevlerde, modelin genel performansını değerlendirmek için önemli bir metrik olarak kullanılır. Ancak, görüntü sayısı az, dengesiz veri setleri ve düşük

kontrastlı nesnelerde yalnız başına yeterli olmayabilir; bu nedenle Kesinlik, Duyarlılık ve F1-Score gibi diğer metriklerle birlikte kullanılması önerilmektedir.

1.5.6. F-Ölçütü (F-measure)

F-Ölçütü (Chen vd.,2022; Pang vd.,2022) Kesinlik ve Duyarlılık metriklerini birleştirerek bir modelin genel doğruluğunu ve eksiksizliğini değerlendiren bir ölçüttür. Ancak bazı durumlarda Kesinlik ve Duyarlılık arasında öncelik belirlemek gerekebilir. Bu gibi durumlarda, Beta (β) değeri kullanılarak F-Ölçütü metriği ağırlıklı hale getirilebilir. F-Ölçütünün matematiksel olarak gösterimi şu şekildedir:

$$F_{\beta} = \frac{(1 + \beta^2) \times \text{Kesinlik} \times \text{Duyarlılık}}{\beta^2 \times \text{Kesinlik} + \text{Duyarlılık}} \quad (6)$$

β değeri, Kesinlik ve Duyarlılık arasında bir denge kurmak için kullanılır ve metriğin şu şekilde ağırlıklandırılmasını sağlar.

- $\beta > 1$: Duyarlılığa daha fazla önem verir. Bu, modelin eksiksizliği (nesneleri kaçırmama) üzerine odaklanmasını sağlar.
- $\beta < 1$: Kesinliğe daha fazla önem verir. Bu, modelin doğruluğu (yanlış pozitifleri azaltma) üzerine odaklanmasını sağlar.
- $\beta = 1$: Kesinlik ve Duyarlılık eşit ağırlığa sahiptir. $\beta = 1$ seçilmesi durumunda F-Ölçütü klasik F1-Score değeriyle aynıdır.

Tıbbi görüntü analizi veya güvenlik uygulamaları gibi alanlarda bir modelin kaçırdığı nesneler hayati önem taşımaktadır. Bu gibi nesnelerin tespitinin önemli olduğu durumlarda yüksek β değeri ($\beta > 1$) seçilmelidir. Örneğin çok ciddi hastalıkların teşhisinde yanlış negatifler (FN) hayati sonuçlara yol açabileceğinden Duyarlılığa daha fazla önem verilir. Spam filtreleme gibi Yanlış Pozitiflerin (FP) daha maliyetli olduğu durumlarda düşük β değeri seçilir ($\beta < 1$). Örneğin bir e-posta servisi yanlışlıkla önemli bir mesajı spam olarak işaretlemek istemeyecektir. β değeri ile ağırlıklandırılmış F-Ölçütü, model değerlendirmesinde Kesinlik ve Duyarlılık arasında bir denge kurmak için güçlü bir araçtır. Özellikle kamufleli görüntülerin bölütlenmesi ve tanıma görevlerinde, uygulamanın ihtiyaçlarına bağlı olarak β 'nın optimize edilmesi, modelin performansını artırabilir.

1.5.7. S-Ölçütü (S-measure)

S-Ölçütü (Fan vd., 2017) kamuflajlı nesne tespiti ve bölütleme görevlerinde, modelin tahmin edilen maskelerinin görsel olarak ne kadar "anamlı" olduğunu değerlendiren bir metrik olarak kullanılır. S-Ölçütü, bölgesel ve nesnel benzerlikleri birleştirerek, bölütleme maskelerinin doğruluğunu ve yapısal anlamlılığını değerlendirir. S- Ölçütü, bölgesel ve nesnel benzerlikleri ağırlıklı olarak birleştirir. S-Ölçütü matematiksel olarak şu şekilde tanımlanmaktadır:

$$S = \alpha \times S_0 + (1 - \alpha) \times S_r \quad (7)$$

Burada;

S_0 : Nesnel benzerlik

S_r : Bölgesel benzerlik

α : Nesnel ve bölgesel benzerlikler arasındaki ağırlıklandırma parametresidir. Bu parametre genellikle $\alpha=0,5$ olarak alınır.

S-Ölçütü bölütleme maskelerinin görsel anlamlılığını ölçer, bu da insan algısına uygun sonuçlar üretmektedir. Bölütleme sonuçlarının yalnızca piksel doğruluğunu değil, aynı zamanda yapısal bütünlüğünü de analiz eder. Özellikle kamuflajlı nesnelerin zorlu ve düşük kontrastlı sahnelerinde güçlü bir değerlendirme sağlar. S-Ölçütü, kamuflajlı nesne bölütlemesi ve tanıma görevlerinde, bölgesel ve nesnel benzerlikleri dikkate alarak model performansını anlamlı bir şekilde değerlendiren güçlü bir metriktir.

1.5.8. Ortalama Mutlak Hata (Mean Absolute Error, MAE)

Ortalama Mutlak Hata (MAE), model tarafından tahmin edilen değerler ile gerçek değerler arasındaki farkların mutlak ortalamasını hesaplayan bir performans ölçütüdür. Kamuflajlı nesne bölütleme ve tanıma görevlerinde, tahmin edilen bölütleme maskeleri ile gerçek maskeler arasındaki piksel tabanlı hataları değerlendirmek için yaygın olarak kullanılır. MAE'nin matematiksel formülü şu şekildedir (Liu vd., 2023):

$$MAE = \frac{1}{N} \sum_{i=1}^N |P_i - G_i| \quad (8)$$

Burada;

P_i : Tahmin edilen piksel değeri

G_i : Gerçek piksel değeri

N : Tüm piksellerin sayısı

MAE, tahmin edilen değerlerin gerçek değerlere ne kadar yakın olduğunu ölçer. Daha düşük bir MAE değeri, daha iyi bir model performansı anlamına gelir. MAE' nin avantajları; hataları mutlak değer olarak sunduğu için doğrudan anlaşılır olması ve veri içindeki aşırı uçlardan daha az etkilenmesi olarak sıralanabilir. MAE'nin sınırlamaları ise küçük hataları ve büyük hataları eşit şekilde değerlendirmesi ve görsel olarak önemli yapısal hataları hesaba katmaması olarak ifade edilebilir. MAE, kamuflajlı nesne bölütlemesi ve tanıma görevlerinde, tahmin edilen maskelerin gerçek maskelere olan yakınlığını ölçmek için etkili bir metriktir. Düşük MAE değerleri, modelin daha iyi performans sergilediğini gösterir. Ancak, MAE' nin görsel algıya uygunluk ve yapısal bütünlük gibi ek metriklerle birlikte kullanılması önerilmektedir.

1.5.9. E-Ölçütü (E-measure)

İkili bir haritanın iki özelliğini (piksel düzeyinde eşleşme ve görüntü düzeyinde istatistik) yakalamak için geliştirilmiş hizalama matrisi ϕ 'yi kullanarak, E-Ölçütü şu şekilde tanımlanmaktadır (Fan vd.,2018):

$$E = \frac{1}{W \times H} \sum_{x=1}^W \sum_{y=1}^H \phi_{FM}(x, y) \quad (9)$$

Burada:

ϕ : Bir ikili haritanın piksel düzeyinde eşleşme ve görüntü düzeyinde istatistik özelliklerini yakalamak için kullanılan hizalama matrisi

W: Görüntünün yüksekliği

H: Görüntünün genişliği

FM: Ön plan haritasıdır.

E-Ölçütü bölütleme sonuçlarının görsel ve istatistiksel değerlendirmesi için kullanılan bir metriktir. Özellikle kamuflajlı nesne bölütlemesi ve tanıma gibi zorlu görevlerde, tahmin edilen maskelerin doğruluğunu ölçmek ve insan algısıyla uyumlu sonuçlar elde etmek için

kullanılır. E-Ölçütü, görsel ve algısal anlamda daha etkili bir değerlendirme sağlayarak sadece piksel doğruluğunu değil, aynı zamanda yapısal özellikleri de dikkate alır.

1.5.10. Yapısal Benzerlik Endeksi Ölçüsü (Structural Similarity Index Measure, SSIM)

SSIM, görüntü kalitesini veya bölütleme performansını değerlendirmek için kullanılan bir ölçümdür. SSIM, insan görsel algısıyla uyumlu bir şekilde, görüntüdeki parlaklık, kontrast ve yapısal bileşenlerdeki değişiklikleri değerlendirir. Özellikle kamuflajlı nesne tespiti ve bölütleme gibi karmaşık görevlerde, modelin tahmin edilen çıktılarının gerçek görüntülerle ne kadar benzer olduğunu analiz etmek için kullanılır. SSIM iki görüntü arasındaki benzerliği şu formülle ölçer (Nasr vd., 2017):

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (10)$$

Burada:

μ_x, μ_y : Sırasıyla X ve Y görüntülerinin ortalama yoğunlukları

σ_x^2, σ_y^2 : Sırasıyla X ve Y görüntülerinin varyansları

σ_{xy} : X ve Y görüntüleri arasındaki kovaryans

C_1, C_2 : Stabilite için kullanılan sabitlerdir.

1.5.11. Zar Benzerlik Katsayısı (Dice Similarity Coefficient, DSC)

DSC, iki küme arasındaki benzerliği ölçen bir istatistiksel metriğe dayanır. Özellikle kamuflajlı nesnelerin bölütlenmesi ve tanıma problemlerinde, tahmin edilen bölütleme maskelerinin gerçek maskelerle ne kadar örtüştüğünü değerlendirmek için yaygın olarak kullanılır. DSC' nin matematiksel tanımı şu şekildedir:

$$DSC = \frac{2|A \cap B|}{|A| + |B|} \quad (11)$$

Burada:

A: Tahmin edilen piksel kümeleri

B: Gerçek piksel kümeleri

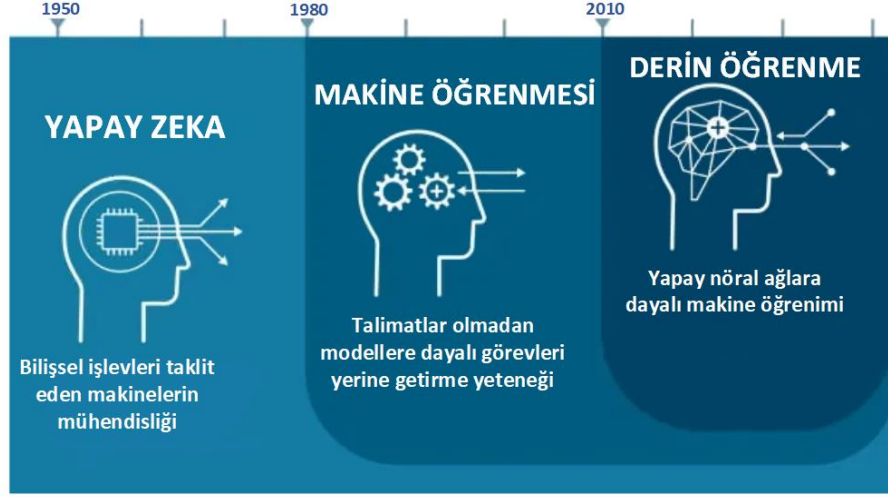
$|A \cap B|$: İki kümenin kesişimindeki ortak elemanları ifade etmektedir.

DSC, basit bir matematiksel ifadeyle bölütleme doğruluğunu ölçer. Görüntü bölütlenmesi ve diğer birçok alanda standart bir metrik olarak yaygın kullanılır. DSC, kamuflajlı nesne bölütlenmesi gibi düşük kontrastlı ve zor sahnelerde model doğruluğunu değerlendirmek için etkili bir araçtır. Bu metrik, bölütleme maskelerinin gerçek maskelere olan örtüşmesini ölçerek, modelin genel performansını analiz etmek için güçlü bir yöntem sunmaktadır (Raina vd.,2023; Liu vd., 2019)

1.6. Yapay Zekâ Kavramı

Yapay zekâ (YZ), insan benzeri zekâ sergileyerek öğrenme, mantık yürütme, problem çözme, algılama ve karar verme gibi görevleri gerçekleştirme yeteneğini ifade eder. Bu alan, bilgisayar bilimleri, matematik, sinirbilim, dilbilim ve felsefe gibi birçok disiplini birleştirerek insan zekâsına benzeyen sistemler geliştirmeyi hedefler. YZ, insan zekâsına benzer şekilde problem çözme ve öğrenme yeteneklerine sahip sistemler geliştirmeyi amaçlayan dinamik bir alandır. Hem pratik hem de teorik yönleriyle, YZ geleceğin teknolojilerini şekillendiren temel bir bilim dalı olarak öne çıkmaktadır. Ancak, bu teknolojinin etik ve toplumsal etkileri üzerine dikkatle düşünmek gerekmektedir. YZ' nin kökenleri, insan zekâsını taklit etmeyi amaçlayan erken dönem mekanik cihazlara dayanır. Alan, özellikle 1956 yılında Dartmouth Konferansı'nda resmi olarak "Yapay Zekâ" teriminin tanıtılmasıyla bir bilim dalı olarak gelişmiştir (McCarthy vd., 2006).

Nabiyev, "Yapay Zekâ" adlı eserinde, yapay zekâyı bilgisayarların insan benzeri zekâ davranışlarını sergilemesini sağlayan bir bilim dalı olarak tanımlar. Bu tanım, makinelerin öğrenme, akıl yürütme, problem çözme ve dil anlama gibi yetenekleri kazanmasını içerir. Nabiyev, yapay zekânın doğal zekâ ile karşılaştırıldığında, belirli algoritmalar ve hesaplamalar yoluyla zeki davranışların modellenmesini amaçladığını belirtir. Ayrıca, yapay zekanın evrelerine ve uygulama alanlarına da ayrıntılı olarak değinmektedir (Nabiyev, 2021). YZ alt kümelerine ayrılmaktadır. Şekil 9'da gösterildiği üzere makine öğrenmesi ve derin öğrenme günümüzde çok popüler olan YZ'nin alt kümeleridir. YZ, tarihsel olarak bilim insanlarının makinelerin insan gibi düşünebileceği fikrini benimsemesiyle ortaya çıkmıştır. Zamanla bu fikir, matematik, mühendislik ve bilişim gibi disiplinlerden gelen katkılarla şekillenmiştir.



Şekil 9. Yapay zekâ, makine öğrenmesi ve derin öğrenmenin kronolojisi.

Şekil 9’da gösterilen tarihsel süreçlerden bahsedecek olursak;

Alan Turing, bir makinenin düşünebilme yeteneğini ölçmek için "Turing Testi" ni önerdi. Turing, makinelerin insan zekâsını taklit edebileceği fikrini, bu alandaki ilk teorik temellerden biri olarak öne sürdü (Turing, 1950).

YZ terimi, John McCarthy tarafından 1956 yılında Dartmouth Konferansı’nda resmi olarak tanıtıldı. Bu konferans, YZ’nin disiplinler arası doğasını vurgulayan ve alanın başlangıç noktası olarak kabul edilen bir dönüm noktasıdır (McCarthy vd., 2006). 1970’lerin sonlarına doğru belirli bir alanda insan uzmanların bilgi ve mantık kurallarını taklit eden uzman sistemler geliştirilmeye başlandı (Van Melle vd., 1978). 1980’lerin sonunda, YZ araştırmalarına duyulan ilginin azalması ve finansman eksikliği nedeniyle bir durgunluk dönemi yaşanmıştır (Russell & Norvig, 2021). 2000’ler ve sonrasında artan işlem gücü ve büyük veri analitiği, makine öğrenimi ve derin öğrenme modellerinin geliştirilmesini sağlamıştır.

Günümüzde, YZ otonom araçlar, doğal dil işleme, sağlık hizmetleri ve daha birçok alanda uygulanmaktadır. YZ, Alan Turing’in teorilerinden günümüz derin öğrenme modellerine kadar sürekli bir evrim geçirmiştir. Tarihsel gelişimi, teknolojiyle paralel olarak ilerlemiş ve insan hayatını dönüştürmeye devam eden bir alan haline gelmiştir. YZ, günümüzde otonom araçlar, doğal dil işleme, sağlık hizmetleri ve daha sayamayacağımız birçok alanda uygulanmaktadır. Örneğin, günümüzde ChatGPT gibi büyük dil modelleri, doğal dil işleme alanında devrim gerçekleştirmiştir.

1.7. Makine Öğrenmesi

Makine öğrenmesi (ML), bilgisayarların açık bir programlamaya ihtiyaç duymadan verilerden otomatik olarak öğrenmesini sağlayan bir yapay zekâ dalıdır. Bu teknik, örüntü tanıma, istatistiksel analiz ve optimizasyon yöntemlerini kullanarak modeller oluşturur ve gelecekteki kararları daha doğru hale getirir. ML, "bilgisayar sistemlerinin deneyimlerden öğrenmesi ve performansını artırması" olarak tanımlanabilir (Mitchell, 1997). ML, insan müdahalesi olmadan örnek verilerden (eğitim verileri) öğrenerek tahmin veya sınıflandırma yapmayı amaçlamaktadır.

Bilgisayarlara teknolojinin sağladığı imkânlar dahilinde insan gibi düşünebilme yeteneğini entegre etmek makine öğrenimini sağlamıştır. Makine öğrenmesi yapay zekânın bir alt kümesi olup, deneyimlerden öğrenerek performansını iyileştirmektedir (Bini, 2018).

Makine öğrenmesinin temel bileşenleri olarak veri, model, algoritmalar, performans ölçütleri sayılabilir:

- Veri, modelin eğitilmesi ve test edilmesi için kullanılan örneklerdir.
- Model, öğrenme algoritmasının oluşturduğu matematiksel yapıdır.
- Algoritmalar, modelin verilerden örüntü öğrenmesini sağlayan karar ağaçları, sinir ağları gibi yöntemlerdir.
- Performans ölçütleri, modelin doğruluğunu ölçmek için kullanılan değerlendirme metrikleridir.

Makine öğrenmesinin avantajları:

- Büyük miktarda veriyi analiz ederek karmaşık ilişkileri öğrenebilir.
- Karar destek sistemlerinde hata oranlarını azaltır.
- Sürekli öğrenme ve kendini geliştirme yeteneğine sahiptir.

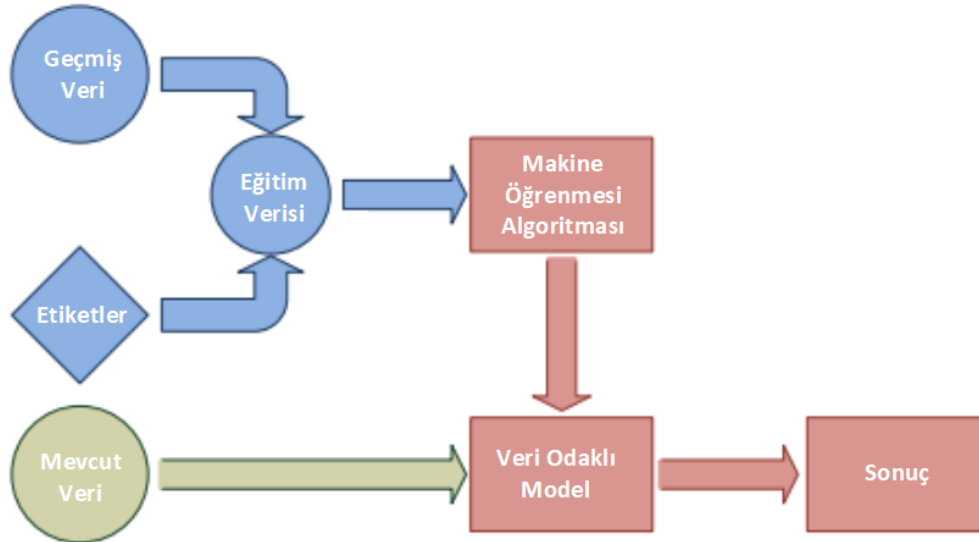
Makine öğrenmesinin zorlukları:

- Veri kalitesi ve miktarına duyarlı olması
- Aşırı öğrenme (overfitting) sorununa açık olması
- Şeffaflık ve yorumlanabilirlik eksikliği

1.7.1. Makine Öğrenmesi Türleri

ML, genel olarak denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme olarak üç alt alana ayrılmaktadır (Adam & Smith, 2008). Bu sınıflandırma, modelin eğitildiği veri türüne ve öğrenme sürecinin yapısına bağlı olarak yapılmaktadır.

Denetimli Öğrenme: Denetimli öğrenme, girdilere karşılık gelen doğru çıktılarla (etiketli veri) modelin eğitildiği bir öğrenme türüdür. Amaç, modelin eğitim verisinde öğrendiği örüntüleri yeni verilerde genelleştirebilmesini sağlamaktır. Denetimli öğrenmede model, etiketlenmiş verilerle eğitilir ve girdilerle çıktılar arasındaki ilişkiyi öğrenir (Bishop, 2006). Bu öğrenme türünde etiketli veri kümeleri kullanılarak, algoritmalar eğitilir ve verilerin sınıflandırılması ya da doğru tahminlerde bulunma amacı taşınmaktadır. Denetimli öğrenmede insan müdahalesi olmadan geçmişe dayalı verilerden algoritmalar öğrenebilmektedir. Bu öğrenmede bilinen bir girdi veri seti bulunmakta olup hem giriş hem de çıkış parametrelerine sahiptir. Girdi verileri kullanılarak bu veriler üzerinden tahminler yapılmaya başlanır ve algoritma kabul edilen performansa ulaştığında öğrenme durur. Kullanılan veri seti geçmiş veri olarak tanımlanır. Daha sonra veri setleri etiketlenerek makine öğretilmeye başlanmaktadır. Elde edilen veri setine eğitim verisi denilir. Bu veriler makine öğrenmesi tekniği ile işlenerek, giriş ve çıkış algoritmasına dayanan bir model oluşturulmaktadır (Calayır & Kabak, 2021).



Şekil 10. Denetimli makine öğrenme genel akış şeması (Jahnke, 2015).

Denetimli öğrenmenin genel akış şeması Şekil 10'da gösterilmektedir. Denetimli öğrenmede Doğrusal Regresyon, Lojistik Regresyon, Karar Ağaçları, Destek Vektör

Makineleri, Yapay Sinir Ağları gibi algoritmalar yaygın olarak kullanılmaktadır. Denetimli öğrenmenin avantajları; model performansının net bir şekilde değerlendirilebilir olması ve belirli görevler için yüksek doğruluk oranı sağlaması olarak sıralanabilir. Dezavantajları ise etiketlenmiş veri gerektirdiğinden maliyetli olması ve aşırı öğrenme (overfitting) riski taşıması olarak sıralanabilir. Denetimli öğrenme tıbbi görüntü analizi ile hastalık teşhisi, eposta spam tespiti, finansal dolandırıcılık tespiti, e-ticaret, otonom sistemler, doğal dil işleme gibi birçok alanda kullanılmaktadır.

Denetimsiz Öğrenme: Denetimsiz öğrenme, modelin etiketlenmemiş verilerdeki gizli örüntüleri ve ilişkileri keşfetmesini amaçlar. Bu yöntem, özellikle büyük miktarda etiketlenmemiş veriye sahip olduğumuzda kullanışlıdır. Denetimsiz öğrenmede kümeleme, boyut indirgeme gibi algoritmalar kullanılmaktadır. Kümeleme algoritmaları veriyi benzer özelliklere sahip gruplara ayırır. Boyut indirgeme algoritmaları ise yüksek boyutlu verilerde en önemli değişkenleri belirlemektedir. Denetimsiz öğrenmenin avantajları ve dezavantajları aşağıdaki gibi açıklanabilir.

Avantajları:

- Etiketlenmiş veri gerektirmez, dolayısıyla daha düşük maliyetlidir
- Yeni ve bilinmeyen örüntüleri keşfetmeye yardımcı olur

Dezavantajları:

- Model performansını değerlendirmek zordur
- Sonuçların yorumlanabilirliği düşüktür

Denetimsiz öğrenme pazarlama alanında müşteri bölütlemesi, siber güvenlik, genetik veri analizi gibi alanlarda kullanılmaktadır.

Takviyeli (Pekiştirmeli) Öğrenme: Takviyeli öğrenme (TÖ), bir çevreyle etkileşime girerek öğrenmedir. Bir TÖ aracı, açıkça öğretilmekten ziyade eylemlerinin sonuçlarından öğrenir ve eylemlerini geçmiş deneyimlerine ve ayrıca yeni seçimlere (keşif) dayanarak seçer, bu da esasen deneme yanılma öğrenimidir. TÖ aracısının aldığı takviye sinyali, bir eylemin sonucunun başarısını kodlayan sayısal bir ödüldür ve aracı, zamanla biriken ödülü en üst düzeye çıkaran eylemleri seçmeyi öğrenmeye çalışır (Woergoetter & Porr, 2008). TÖ, bir ajan (öğrenen sistem) ile çevre arasında etkileşim kurarak ödül ve ceza mekanizması aracılığıyla öğrenmeyi sağlayan bir öğrenme türüdür (Sutton, 2018). Bu yöntemde ajan, aldığı geri bildirimlere göre en iyi eylem dizisini öğrenmeye çalışmaktadır. Bu uygulamanın kullanım alanlarına örnek olarak satranç oynama, otonom araçlar ve robotik alanı verilebilir.

1.8. Derin Öğrenme

Derin öğrenme (DÖ), makine öğreniminin bir alt dalı olup, insan beyninin çalışma şekline esinlenerek geliştirilen çok katmanlı yapay sinir ağlarıyla verilerden otomatik olarak karmaşık örüntüler öğrenmeyi amaçlayan bir yöntemdir (Dong vd., 2021). DÖ, geleneksel makine öğrenmesi algoritmalarından farklı olarak büyük veri kümelerinden anlamlı temsil öğrenebilme yeteneğiyle öne çıkmaktadır. DÖ' de insanların öğrenme süreci taklit edilip, sinir hücreleri ve hücreler arasındaki öğrenme bağlantısı yapay olarak oluşturulup makinelerle insan beyninin öğrenme şekline benzer şekilde öğretme amaçlanmaktadır. DÖ işleminde veriler birbirlerini takip eden belirli katmanlarda işlenmektedir.

DÖ 'de kullanılan bilgisayar modeli sınıflandırma görevlerini metinden, sesten ya da görüntüden gerçekleştirmeyi öğrenir. DÖ modelleri, insan performansını aşarak daha fazla doğruluk elde edebilmektedir. DÖ' de kullanılan modeller çok sayıda katmana sahip sinir ağı mimarileri ve çok fazla sayıda etiketli veri kullanılmasıyla eğitilmektedir. DÖ ile her zamankinden daha yüksek seviyelerde tanıma doğruluğu sağlanabilmektedir (Xia vd., 2022). Son yıllarda yapay zekâ alanında büyük bir ivme kazanan DÖ, çok katmanlı yapay sinir ağları kullanarak karmaşık problemlere çözümler sunan bir makine öğrenmesi tekniği olarak adlandırılır (LeCun vd.,2015). Derin öğrenme, büyük veri setlerinden öğrenme yeteneği sayesinde görüntü işleme, doğal dil işleme, finansal analiz ve sağlık gibi çeşitli alanlarda yaygın olarak kullanılmaktadır (Goodfellow, 2016). Geleneksel makine öğrenmesi yöntemlerine kıyasla, derin öğrenme modelleri daha fazla katmana ve daha karmaşık hesaplama yeteneklerine sahiptir, bu da onların doğruluğunu ve genelleme yeteneğini artırmaktadır (Schmidhuber, 2015). DÖ teknikleri, büyük miktarda veriyi işleyerek anlamlı bilgiler çıkarabilmek için genellikle GPU'lar gibi yüksek performanslı donanımlara ihtiyaç duyar (Chen & Ran, 2019). Bu nedenle, derin öğrenmenin uygulanması hem yazılım hem de donanım gelişmelerine paralel olarak ilerlemektedir.

1.8.1. Yapay Sinir Ağları

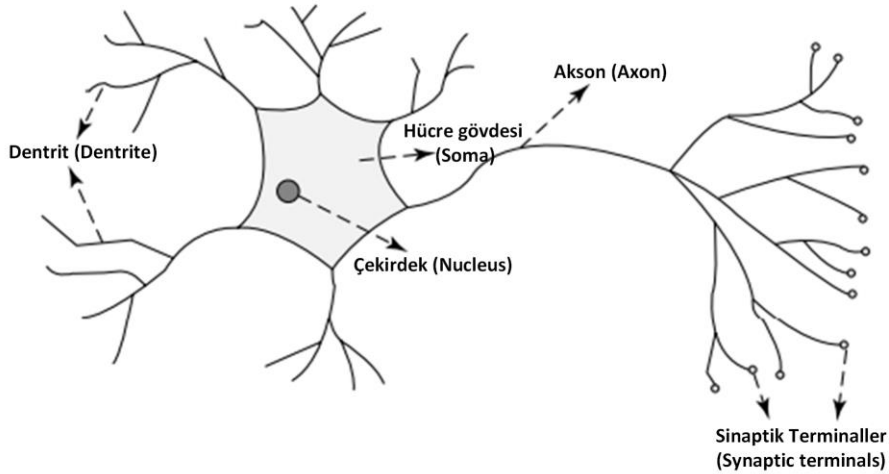
Sinir ağları, insan beynindeki nöronların çalışma mekanizmalarından ilham alınarak geliştirilmiş bilgi işleme sistemleridir. Beynin nöron yapısını taklit eden bu yapay sistemler, karmaşık problemleri çözme ve öğrenme yetenekleri ile dikkat çeker. Bu sistemler, paralel çalışan ve birbiriyle bağlantılı birçok yapay nöronlardan oluşur. Her bir yapay nöron, gelen

girdiyi işler ve sonuçlarını diğer nöronlara aktarır. Sinir ağlarının temel amacı, girdiler ve çıktılar arasındaki ilişkileri öğrenmek ve bu bilgiyi kullanarak yeni durumlar için tahminlerde bulunmaktır (Kruse vd., 2016). Yapay sinir ağlarının tek bir nöronunun genel modellenmesi Şekil 11'deki gibidir.



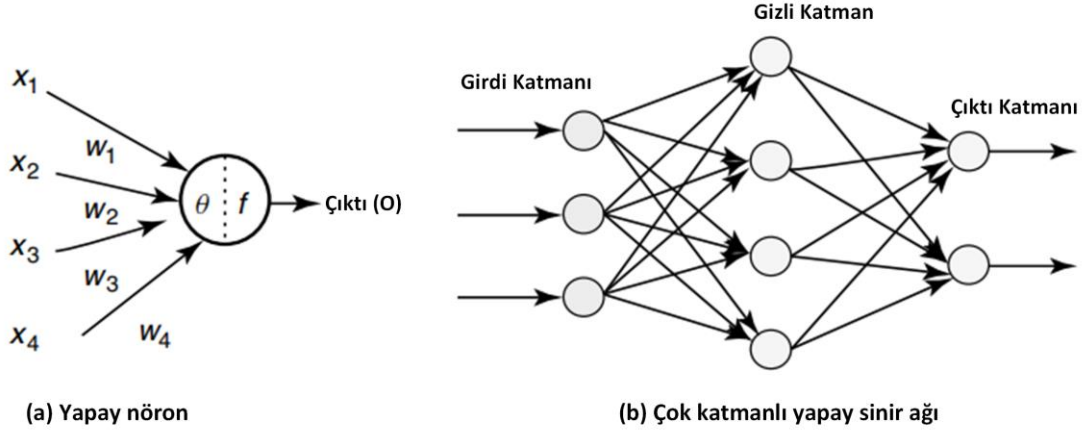
Şekil 11. Yapay sinir ağında tek bir nöronun genel modellenmesi

İnsan beyni, birbirine bağlı 10 milyardan fazla nörondan oluşan bir koleksiyondur. Her bir nöron, bilgiyi almak, işlemek ve iletmek için biyokimyasal reaksiyonları kullanan bir hücredir. Şekil 12'de insan beynine ait bir nöron şekilsel olarak gösterilmiştir. Şekil 12'de görüleceği üzere dendrit adı verilen üçlü sinir lifi ağları, hücre çekirdeğinin bulunduğu hücre gövdesine veya somaya bağlanır. Hücre gövdesinden uzanan ve akson adı verilen tek bir uzun lif, sonunda iplikçiklere ve alt iplikçiklere ayrılır ve sinaptik terminaller veya sinapslar yoluyla diğer nöronlara bağlanır. Sinapslarda sinyallerin bir nörondan diğerine iletilmesi, belirli verici maddelerin düğüm noktasının gönderen ucundan salındığı karmaşık bir kimyasal süreçtir. Bunun etkisi, alıcı hücrenin gövdesi içindeki elektrik potansiyelini yükseltmek ya da düşürmektir. Potansiyel bir eşiğe ulaşırsa, aksondan aşağıya bir sinyal gönderilir ve hücre "tetiklenir".



Şekil 12. Beyin nöronu şekilsel gösterimi (Yıldırım, 2020).

Yapay sinir ağı (YSA), biyolojik sinir sistemlerinin matematiksel modellerinin genelleştirilmesi olarak geliştirilmiştir. Sinir ağlarının temel işlem elemanlarına yapay nöronlar ya da kısaca nöronlar veya düğümler denir. Nöronun basitleştirilmiş bir matematiksel modelinde, sinapsların etkileri, ilişkili giriş sinyallerinin etkisini modüle eden bağlantı ağırlıkları ile temsil edilir ve nöronlar tarafından sergilenen doğrusal olmayan karakteristik bir transfer fonksiyonu ile temsil edilir. Nöron impulsu daha sonra transfer fonksiyonu tarafından dönüştürülen giriş sinyallerinin ağırlıklı toplamı olarak hesaplanır. Yapay bir nöronun öğrenme kabiliyeti, seçilen öğrenme algoritmasına uygun olarak ağırlıkların ayarlanmasıyla elde edilir.



Şekil 13. (a) Yapay bir nöronun, (b) çok katmanlı bir yapay sinir ağı mimarisi (Yıldırım, 2020).

Tipik bir yapay nöron ve çok katmanlı bir sinir ağı mimarisi Şekil 13'de gösterilmiştir. Şekil 13'e bakıldığında, $x_1, \dots, x_n$ girişlerinden gelen sinyal akışının, bir nöronun çıkış sinyal akışı (O) gibi oklarla gösterilen tek yönlü olduğu kabul edilir. Nöron çıkış sinyali O aşağıdaki (12) eşitlik ilişkisi ile verilir:

$$O = f(net) = f\left(\sum_{j=1}^n w_j x_j\right) \quad (12)$$

Burada w_j ağırlık vektörüdür ve $f(net)$ fonksiyonu aktivasyon (transfer) fonksiyonu olarak adlandırılır. Değişken ağ (net), (13) eşitliğinde gösterildiği gibi ağırlık ve girdi vektörlerinin skaler çarpımı olarak tanımlanır,

$$net = w^T x = w_1 x_1 + \dots + w_n x_n \quad (13)$$

Burada T bir matrisin transpozesidir ve en basit durumda O çıkış değeri şu şekilde hesaplanır:

$$O = f(net) = \begin{cases} 1 & \text{if } w^T x \geq \theta \\ 0, & d.d \end{cases} \quad (14)$$

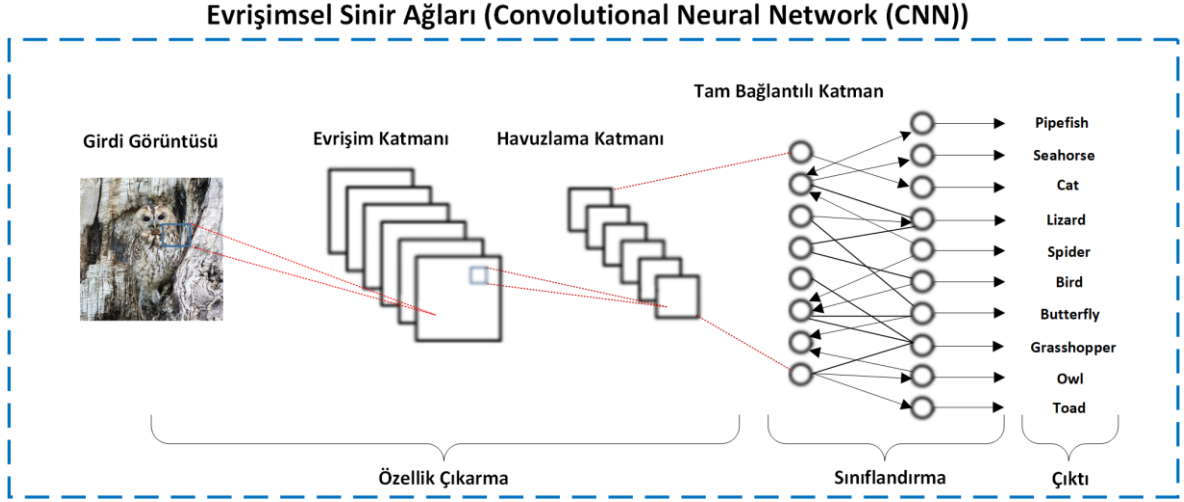
Burada θ eşik seviyesi olarak adlandırılır ve bu tip düğümlere doğrusal eşik birimi denir (Abraham, 2005).

1.8.2. Evrişimsel Sinir Ağları (CNN)

Evrişimsel Sinir Ağları (Convolutional Neural Networks - CNN), özellikle görüntü işleme ve bilgisayarla görme alanlarında kullanılan, derin öğrenmeye dayalı bir yapay sinir ağı modelidir. CNN'ler, insan görsel sisteminden ilham alarak geliştirilmiş olup, girdilerdeki (örneğin görüntülerdeki) temel özellikleri otomatik olarak çıkarma yeteneğine sahiptir. CNN'ler, görüntüleri pikseller üzerinden analiz ederek, otomatik olarak kenar, doku ve desen gibi belirli özellikleri öğrenebilen bir model yapısına sahiptir (Li vd., 2020).

CNN'in ilk başarılı uygulaması (Lecun, vd., 1998) LeNet-5 modeli adıyla optik karakter tanıma yapma amacıyla geliştirilmiştir. Günümüzde en popüler ve yaygın olarak kullanılan derin öğrenme yöntemlerinden birisi olan CNN bilgisayarlı görü, görüntü sınıflandırma, yüz tanıma, konuşma tanıma gibi birçok alanda kapsamlı şekilde kullanılmaktadır.

Şekil 14' gösterildiği gibi CNN çok katmanlı bir yapıya sahiptir. Temel olarak iki kısımdan oluşmaktadır. Birinci kısım evrişim (convolution) katmanı ve havuzlama (pooling) katmanlarından oluşur. Bu kısım derin özniteliklerin çıkarımı ile ilgilidir. İkinci kısım ise tam bağlantılı (fully connected) katman olup bu kısımda sınıflandırma işlemleri gerçekleştirilir.



Şekil 14. Evrişimsel sinir ağları (CNN) temel katmanları (Bayram & Nabyev,2024)

CNN'lerin performansı büyük ölçüde derin öğrenme teknikleri ve büyük veri setleri ile beraber kullanıldığında artar. ImageNet gibi büyük veri kümelerinde eğitilen CNN modelleri nesne tanıma, görüntü sınıflandırma gibi görevlerde insan gözlemine yakın performans sergileyebilirler. Bunun yanı sıra CNN'lerin aşırı öğrenme probleminin önlenmesi için dikkatlice tasarlanması gerekir. Dropout ve veri artırma gibi teknikler kullanılarak modelin genelleme yeteneği artırılır ve aşırı öğrenme problemiyle başa çıkılabilir.

CNN'in avantajları:

- Yüksek Doğruluk: Derin yapıları sayesinde karmaşık özellikleri öğrenebilir.
- Özellik Çıkarma Yeteneği: Başarılı bir şekilde özellik çıkarım işlemini gerçekleştirir ve manuel müdahale gerektirmez.
- Genelleme Yeteneği: Transfer öğrenme sayesinde farklı veri kümelerinde başarılı sonuçlar verir.

CNN'in zorlukları:

- Büyük Veri Gereksinimi: Yüksek doğruluk için büyük veri setleri gerekir.
- Aşırı Öğrenme (Overfitting): Küçük veri setlerinde model aşırı öğrenebilir
- Yüksek Hesaplama Maliyeti: CNN'lerin eğitimi için güçlü donanım gereklidir.

CNN, bilgisayarla görme başta olmak üzere birçok alanda yapay zekâ uygulamalarının temelini oluşturmaktadır. CNN'lerin geleceği, daha hızlı, verimli ve daha az veri gerektiren modellerin geliştirilmesi yönünde ilerlemektedir. CNN'ler görüntü ve metin verilerini

işlemek için tasarlanmış derin öğrenme modelleridir. CNN'in başarısı, katmanlar arasındaki veri akışını optimize eden özel mimarisine dayanmaktadır (Li vd., 2020). Bu katmanlar, görsel veya sayısal bilgileri çeşitli seviyelerde işler ve çıktıları daha yüksek seviyeli özelliklere dönüştürerek sınıflandırma, bölütleme veya tahmin görevlerinde kullanılmasını sağlar (Yamashita vd., 2018). CNN'in temel bileşenleri olan evrişim katmanı (convolutional layer), havuzlama katmanı (pooling layer), aktivasyon fonksiyonları (activation functions) ve tam bağlantılı katman (fully connected layer), modelin öğrenme sürecindeki kritik yapı taşlarıdır. Evrişim katmanları, girdilerden anlamlı özellikleri çıkarırken, havuzlama katmanları boyutları küçültür ve modelin genelleştirme kapasitesini artırır. Aktivasyon fonksiyonları, ağırlı doğrusal olmayan ilişkileri öğrenmesine yardımcı olurken, tam bağlantılı katmanlar ise nihai tahminleri üretmek için kullanılır. Bu katmanlar birlikte çalışarak, görüntü sınıflandırma, nesne algılama ve doğal dil işleme gibi çeşitli görevlerde yüksek doğruluk oranları sağlayan güçlü modeller oluşturur (Rawat & Wang, 2017).

Evrişim Katmanı (Convolutional Layer): Evrişim Katmanı, Evrişimli CNN'in temel yapı taşlarından biridir ve görüntü gibi girdilerdeki önemli özellikleri çıkarmak için kullanılır (Li vd., 2020). Bu katman, giriş verisi üzerine belirli çekirdekler (kernels veya filtreler) uygulayarak, kenarlar, dokular ve desenler gibi özellikleri belirgin hale getirir (Yamashita vd., 2018).

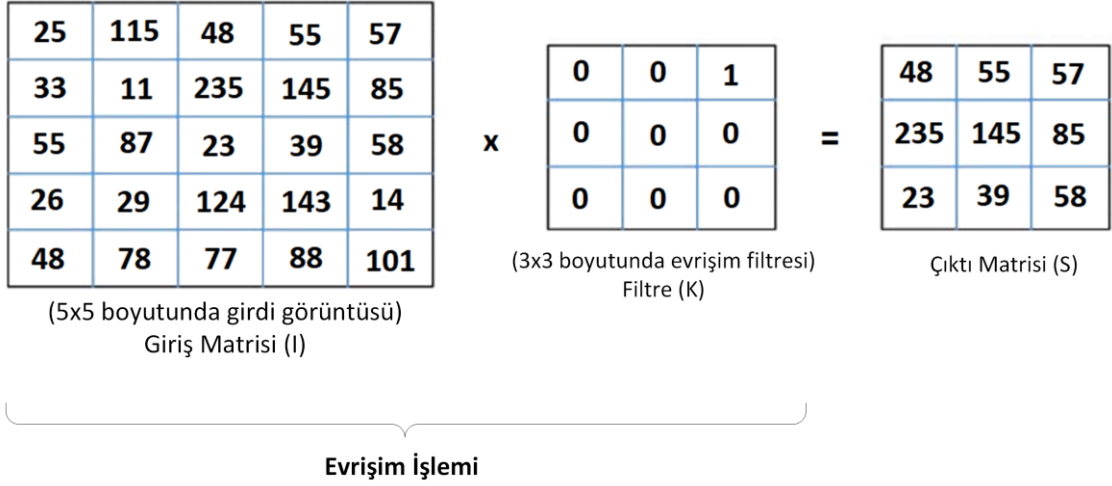
Evrişim işlemi, bir görüntü matrisine belirli bir filtre matrisinin uygulanmasıyla gerçekleştirilir. Matematiksel olarak, evrişim şu şekilde ifade edilir:

$$S(i, j) = (I \times K)(i, j) = \sum_m \sum_n I(i - m, j - n) K(m, n) \quad (15)$$

Burada:

- $I(i, j)$, giriş görüntüsünü temsil eden matristir
- $K(m, n)$, evrişim çekirdeğidir (kernel ya da filtre)
- $S(i, j)$, evrişim sonucu oluşan özellik haritasıdır (feature map).

Şekil 15'de örnekteki gösterildiği gibi evrişim işlemi sırasında, filtre matrisinin her bir değeri, giriş görüntüsündeki karşılık gelen piksel değerleri ile çarpılır ve sonuçlar toplanarak yeni bir piksel değeri oluşturulur. Örnekte görüldüğü üzere 5x5 boyutundaki giriş matrisi üzerinde 3x3 boyutundaki evrişim filtresi gezdirilmekte ve çıkış matrisinin piksel değerleri hesaplanmaktadır.



Şekil 15. Evrişim işlemi için örnek bir uygulama

Evrişim katmanı, her filtre için ayrı bir özellik haritası oluşturur. Derinleştikçe, bu haritalar daha karmaşık yapılar içerir. Örneğin, ilk katmandaki filtreler kenarları algılamak, sonraki katmanlarda nesne parçaları ve daha üst seviyeli temsiller çıkarılabilir (Yamashita vd., 2018).

Havuzlama Katmanı (Pooling Layer): Havuzlama katmanı, CNN’de kullanılan temel bileşenlerden birisidir. Bu katmanın temel amacı, özellik haritalarının boyutunu küçültmek, modelin hesaplama maliyetini azaltmak ve aşırı öğrenmeyi önlemektir (Zafar vd., 2022). Havuzlama katmanı, özellik haritalarında belirli bölgelerdeki en önemli bilgileri yoğunlaştırarak, modelin daha genelleştirilebilir hale gelmesini sağlar (Sun vd., 2017). Havuzlama katmanında kullanılan filtreler maksimum değer hesaplama veya ortalama değer hesaplama prensibine göre çalışırlar. Havuzlama işlemi genellikle 2x2, 3x3, ..., nxn boyutlarında filtreler kullanılarak yapılmaktadır.

Havuzlama işlemi, giriş özelliği haritası (F) üzerinde belirli bir filtre (P) uygulanarak gerçekleştirilir. Genel matematiksel formülü şu şekildedir:

$$S(i, j) = P(F(i: i + m, j: j + n)) \quad (16)$$

Burada:

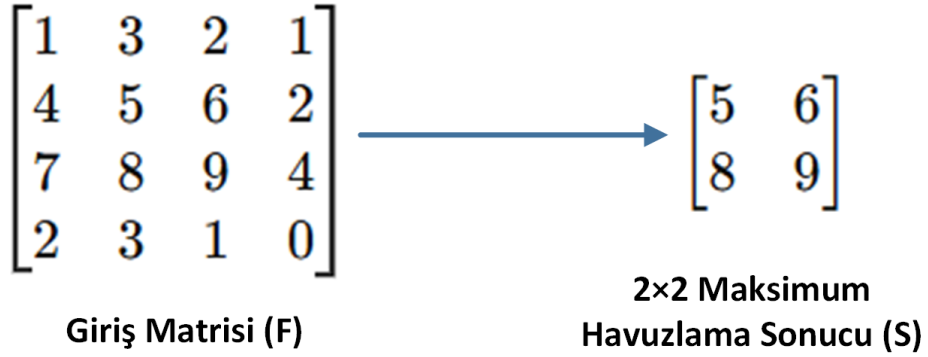
- $F(i: i + m, j: j + n)$, giriş özelliği haritasının (i, j) konumundaki küçük bir bölgesini temsil eder.
- $P()$, havuzlama işlemi (maksimum veya ortalama) anlamına gelir.
- $S(i, j)$, havuzlama işlemi sonrası oluşan çıkış değeridir.

CNN'lerde yaygın olarak kullanılan birkaç havuzlama yöntemi vardır. Bunlar maksimum havuzlama ve ortalama havuzlama olarak sıralanabilir. Havuzlama katmanı, CNN'nin hesaplama maliyetini azaltan, modelin aşırı öğrenmesini önleyen ve genelleme kapasitesini artıran önemli bir bileşendir. Maksimum ve ortalama havuzlama gibi farklı yöntemler, çeşitli kullanım senaryolarına göre optimize edilebilirler.

Maksimum Havuzlama (Max Pooling): Maksimum havuzlama belirli bölgedeki (örneğin 2x2 ya da 3x3) en büyük değeri seçerek çalışır. Bu yöntem, belirli özelliklerin daha güçlü hale getirilmesine yardımcı olur. Matematiksel olarak şu şekilde ifade edilir:

$$S(i, j) = \max(F(i:i+m, j:j+n)) \quad (17)$$

Şekil 16'da maksimum havuzlama için örnek bir uygulama gösterilmiştir. Örneğin (F) giriş matrisinde 2x2 birimlik seçilen ilk alanda yer alan sayılar 1,3,4 ve 5'dir. Bunlardan en büyük olan değer 5 olduğu için çıktı matrisine bu 4 sayının en yükseği olan 5 değeri yazılır. Giriş matrisindeki diğer 2x2'lik alanlarda da bu şekilde işlem yapılır ve çıktı matrisi oluşturulur.



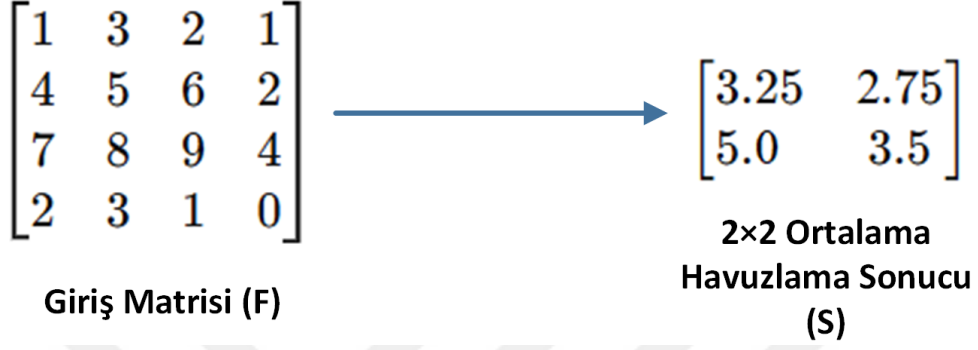
Şekil 16. Maksimum havuzlama işlemi için örnek bir uygulama

Ortalama Havuzlama (Average Pooling): Ortalama havuzlama, belirlenen bir bölgedeki tüm değerlerin ortalamasını alarak çalışır. Bu yöntem, özelliklerin daha yumuşak bir şekilde temsil edilmesini sağlar (Zafar vd., 2022).

Ortalama havuzlama matematiksel olarak şu şekilde ifade edilir:

$$S(i, j) = \frac{1}{mn} \sum_{p=0}^{m-1} \sum_{q=0}^{n-1} F(i+p, j+q) \quad (18)$$

Şekil 17’de ortalama havuzlama için örnek bir uygulama gösterilmiştir. Uygulamada (F) giriş matrisinde 2x2 birimlik seçilen ilk alanda yer alan sayılar 1,3,4 ve 5’dir. Bunların 4’ünün ortalaması olan değer 3.25 olduğu için çıktı matrisinin ilk değerine bu ortalama değeri yazılır. Giriş matrisindeki diğer 2x2’lik alanlarda da bu şekilde sırasıyla işlem yapılır ve çıktı matrisi oluşturulur.



Şekil 17. Ortalama havuzlama işlemi için örnek bir uygulama

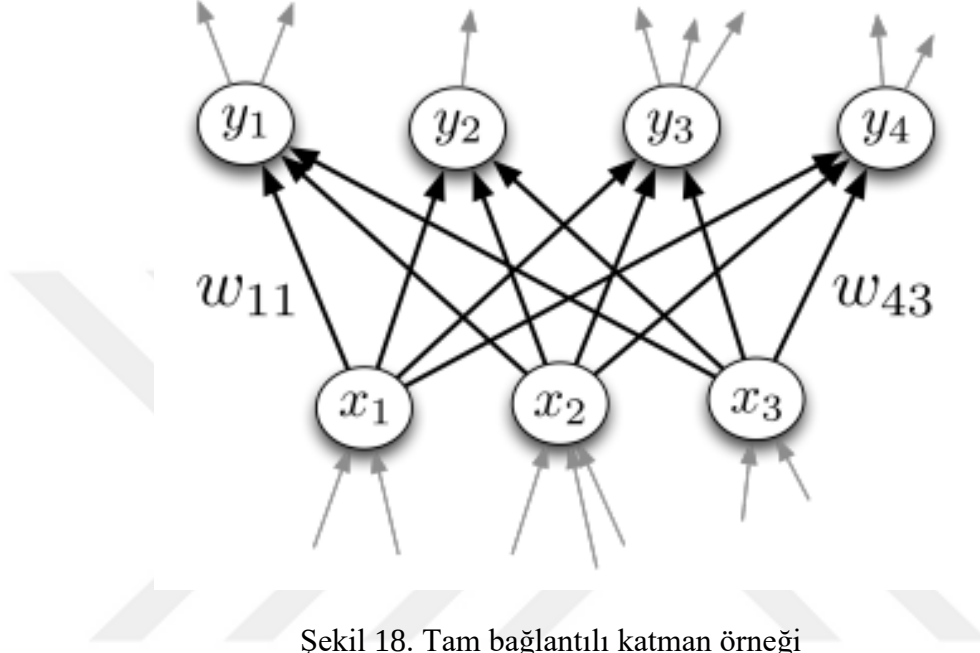
Tam Bağlantılı Katman (Fully Connected Layer): Tam Bağlantı Katmanları, mimarinin bir katmandaki bütün düğüm ve nöronlarının bir sonraki katmanlara bağlandığı bir yapay sinir ağı türüdür. CNN’in son aşamalarından biridir ve modelin tahmin yapmasını sağlayan katmandır. Bu katman, önceki katmanlardan elde edilen özellikleri kullanarak, nihai sınıflandırma veya tahmin işlemini gerçekleştirir. CNN’in son adımı bu katmanda gerçekleşir. Bu ağı türü hesaplama açısından karmaşık olmasının yanı sıra aşırı yüklenmeye meyillidir. Elde edilen verilere uygulanan Düzleştirme (Flattening) işleminden sonra sinir ağları ile öğrenme işlemi uygulanır. Basitçe ifade edersek, bu katmanda birçok kez evrişimli katman ile havuzlama katmanlarından geçen matris halindeki veriler düz bir vektör haline getirilir.

Tam Bağlantılı Katman, önceki evrişim ve havuzlama katmanlarının ürettiği özellik haritalarını düzleştirerek, tam bağlantılı nöronlara iletir. Her nöron, önceki katmandaki tüm nöronlarla bağlantılıdır.

Şekil 18’de gösterildiği gibi tam bağlı bir katmanda, tüm giriş birimleri tüm çıkış birimlerine bağlıdır. N girdiden M çıktı hesaplamamız gerekiyorsa, bunu matris çarpımı kullanarak yapabiliriz. Bu, bir $M \times N$ matrisi kullanacağımız anlamına gelir. Çıktılar, giriş birimlerinin bir fonksiyonudur. Matematiksel olarak, tam bağlantılı katmanın çalışması şu şekilde ifade edilir:

$$y = f(W_x + b) \quad (19)$$

Burada; x : giriş verisi (düzleştirilmiş özellik haritası), W : ağırlık matrisi, b : Bias terimini, $f(\cdot)$: aktivasyon fonksiyonunu (Örneğin Softmax veya ReLU), y : çıktıyı (sınıflandırma sonucu) temsil etmektedir.



Şekil 18. Tam bağlantılı katman örneği

Tam bağlantılı katman, matris çarpımı ve aktivasyon fonksiyonları ile çalışır. Eğer bir önceki katmandan gelen n boyutlu bir özellik vektörü x ve m çıkış nöronu varsa, her çıkış nöronu için işlem şu şekilde olur:

$$z_j = \sum_{i=1}^n (w_{ji}x_i + b_j) \quad (20)$$

$$y_j = f(z_j) \quad (21)$$

Burada:

- w_{ji} : i . girdiden j . çıkışa olan ağırlık
- b_j : Bias değeri
- $f(z_j)$: Aktivasyon fonksiyonu uygulanmış çıktı

Bu işlem, her nöronun tüm girdilere bağlı olduğu anlamına gelir ve modeli daha esnek hale getirir (Qian vd., 2020).

Aktivasyon Fonksiyonları (Activation Functions): Aktivasyon fonksiyonları, yapay sinir ağlarında her nöronun aldığı girdiyi işleyerek belirli bir çıktı üretmesini sağlayan matematiksel dönüşümlerdir. Aktivasyon fonksiyonları, yapay sinir ağlarında ve özellikle CNN'lerde kullanılan en önemli bileşenlerden biridir. CNN'lerde aktivasyon fonksiyonları, modelin doğrusal olmayan ilişkileri öğrenmesini sağlar ve her katmanın çıktısını belirleyerek ağırlıkların güncellenmesine yardımcı olur. Aktivasyon fonksiyonlarının temel amacı, her nöronun aldığı girdiyi işleyerek belirli bir çıktı üretmesini sağlamak ve modele doğrusal olmayan özellikler kazandırmaktır (Qian vd., 2018). Derin öğrenme modellerinde aktivasyon fonksiyonları, girdilerden elde edilen bilgileri işleyerek modelin öğrenmesini ve karar vermesini sağlar. Doğrusal bir model, yalnızca doğrusal ilişkileri öğrenebilirken, doğrusal olmayan aktivasyon fonksiyonları sayesinde sinir ağları çok daha karmaşık ilişkileri öğrenebilir (Vargas vd., 2021). Başlıca aktivasyon fonksiyonları aşağıdaki başlıklar halinde sıralanabilir.

Sigmoid Fonksiyonu: Sigmoid fonksiyonu, giriş değerlerini 0 ile 1 arasında bir değere dönüştüren doğrusal olmayan bir fonksiyondur. Çıkışı 0 ile 1 arasında sınırlandırır, bu da olasılık tahminlerinde kullanılmasını sağlar. Matematiksel olarak şu şekilde ifade edilir:

$$f(z) = \frac{1}{1 + e^{-z}} \quad (22)$$

Burada z , giriş değeri; e , Euler sabiti (≈ 2.718); $f(z)$, 0 ile 1 arasında bir çıktı üretir.

Sigmoid fonksiyonu, sinir ağlarının en eski ve temel aktivasyon fonksiyonlarından biridir. Ancak, kaybolan gradyan (vanishing gradient) problemi ve hesaplama maliyetinin yüksek olması nedeniyle derin ağlarda ReLU gibi fonksiyonlar daha yaygın hale gelmiştir (Maguolo vd., 2021). Bununla birlikte, sigmoid fonksiyonu halen olasılık tahmini, lojistik regresyon ve biyomedikal analizler gibi alanlarda yaygın olarak kullanılmaktadır (Zaheer & Shaziya, 2018).

Tanh (Hiperbolik Tanjant) Fonksiyonu: Tanh fonksiyonu, girişleri -1 ile 1 arasında dönüştüren bir fonksiyondur. Çıkışları -1 ile 1 arasında olduğu için sıfır merkezlidir, bu da öğrenmeyi hızlandırır. Matematiksel olarak şu şekilde ifade edilir:

$$f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (23)$$

Burada z , nörona gelen ağırlıklı toplam giriş değeri; e , Euler sabitidir (≈ 2.718); $f(z)$, çıktı değeridir ve -1 ile 1 arasında değişir (Maguolo vd., 2021).

ReLU (Rectified Linear Unit) Fonksiyonu: ReLU fonksiyonu, modern derin öğrenme modellerinde yaygın olarak kullanılan doğrusal olmayan bir aktivasyon fonksiyonudur. ReLU, giriş verisinin negatif değerlerini sıfıra indirirken, pozitif değerleri olduğu gibi geçirir. Bu sayede, ağdaki nöronlar yalnızca pozitif aktivasyonlar ile çalışır ve hesaplama maliyetini azaltır (Agarap, 2018). ReLU fonksiyonu matematiksel olarak şu şekilde tanımlanır:

$$f(z) = \max(0, z) \quad (24)$$

Burada z , nörona gelen ağırlıklı toplam giriş değeri; $f(z)$ ReLU fonksiyonunun çıktısıdır. Türev fonksiyonu şu şekilde hesaplanır:

$$f'(z) = \begin{cases} 1, & \text{eğer } z > 0 \\ 0 & \text{eğer } z \leq 0 \end{cases} \quad (25)$$

ReLU fonksiyonu, modern derin öğrenme modellerinde en yaygın kullanılan aktivasyon fonksiyonlarından biridir. Kaybolan gradyan problemini büyük ölçüde önlemesi ve düşük hesaplama maliyetiyle öne çıkmaktadır.

Sızdıran ReLU (Leaky ReLU) Fonksiyonu: Sızdıran ReLU, ReLU fonksiyonunun geliştirilmiş bir versiyonudur ve özellikle ölü nöron (dying neurons) problemine çözüm getirmek amacıyla geliştirilmiştir. Standart ReLU fonksiyonunda negatif girişler sıfıra sabitlenirken, Sızdıran ReLU, negatif değerler için küçük bir gradyan bırakır. Bu, ağın daha iyi öğrenmesini ve nöronların tamamen etkisiz hale gelmemesini sağlar (Mastromichalakis, 2020). Sızdıran ReLU fonksiyonu şu şekilde tanımlanır:

$$f(z) = \begin{cases} z, & \text{eğer } z > 0 \\ \alpha z, & \text{eğer } z \leq 0 \end{cases} \quad (26)$$

Burada z , giriş değeridir; α , genellikle 0.01 olan küçük bir katsayıdır ve negatif girdilere küçük bir eğim sağlar; Pozitif girişlerde ReLU ile aynıdır, ancak negatif girişlerde küçük bir değeri geçirerek ölü nöronları engeller (Xu vd., 2020). (26) numaralı fonksiyonun türevi şu şekilde hesaplanır:

$$f'(z) = \begin{cases} 1, & \text{eğer } z > 0 \\ \alpha, & \text{eğer } z \leq 0 \end{cases} \quad (27)$$

Sızdıran ReLU'nun en önemli özelliği, negatif giriş değerleri için sıfır olmayan bir türev sağlamasıdır. ReLU'da negatif girişlerde gradyan sıfır olduğu için bazı nöronlar tamamen devre dışı kalabilir. Sızdıran ReLU, negatif girişler için küçük bir eğim bırakarak bu sorunu çözer. Negatif girdileri sıfırlamamak modelin veri dağılımındaki tüm bilgiyi kullanmasını sağlar.

Softmax Fonksiyonu (Softmax Function): Softmax fonksiyonu, çok sınıflı sınıflandırma problemlerinde kullanılan en yaygın aktivasyon fonksiyonlarından biridir. Sinir ağlarının son katmanında (output layer) yer alır ve modelin her sınıfa olan güvenini olasılık dağılımı olarak normalize eder. Softmax, modelin çıktısını 0 ile 1 arasında bir olasılık değerine dönüştürerek, toplamalarının 1'e eşit olmasını sağlar (Jap vd., 2021). Softmax fonksiyonu matematiksel olarak şu şekilde tanımlanır:

$$S(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \quad (28)$$

Burada z_i , modelin i sınıfı için ürettiği ham skor; e^{z_i} girdinin üstel (exponential) dönüşümüdür. Tüm çıktılar 0 ile 1 arasında olur ve toplamaları 1'e eşittir. Softmax fonksiyonu, çok sınıflı sınıflandırma problemleri için ideal bir aktivasyon fonksiyonudur. Modelin çıktısını olasılık formuna dönüştürerek sınıflar arasında en güçlü tahmini belirler.

Aktivasyon Fonksiyonlarının Genel Karşılaştırılması: Derin öğrenme modellerinde kullanılan aktivasyon fonksiyonları, modelin öğrenme sürecini ve çıktılarının doğruluğunu doğrudan etkileyen kritik bileşenlerdir. Aktivasyon fonksiyonları, giriş verisini dönüştürerek modelin doğrusal olmayan ilişkileri öğrenmesini sağlar ve ağırlık güncellemelerinin etkili bir şekilde yapılmasına yardımcı olur.

Tablo 1'de yaygın olarak kullanılan Sigmoid, Tanh, ReLU, Leaky ReLU ve Softmax fonksiyonlarının karşılaştırması, avantajları, dezavantajları ve kullanım alanları detaylı bir şekilde ele alınmıştır.

Tablo 1. Aktivasyon fonksiyonlarının karşılaştırılması

Fonksiyon	Çıkış Aralığı	Avantajlar	Dezavantajlar	Kullanım Alanları
Sigmoid	$(0,1)$	Olasılık tahmini için uygundur, sürekli ve türevlenebilir.	Kaybolan Gradyan problemi, sıfır merkezli olmama sorunu.	Lojistik regresyon, ikili sınıflandırma, biyomedikal analizler (Zaheer & Shaziya, 2018).
Tanh	$(-1,1)$	Sıfır merkezlidir, daha güçlü gradyan yayılımı sağlar.	Büyük değerlerde kaybolan gradyan problemi görülür.	RNN ve LSTM modelleri, doğal dil işleme (NLP), görüntü işleme (Maguolo et al., 2019).
ReLU	$(0, \infty)$	Kaybolan Gradyan problemini önler, hızlı hesaplanır.	Ölü nöron problemi (negatif değerlerde sifıra kilitlenme) olabilir.	Görüntü işleme, büyük ölçekli derin öğrenme modelleri (Qiu & Cai, 2017).
Sızdıran ReLU	$(-\infty, \infty)$	Negatif değerler için küçük bir gradyan bırakarak ölü nöron problemini önler.	Optimal parametre seçimi zor olabilir.	Büyük ölçekli sinir ağları, derin evrişimli sinir ağları (Xu et al., 2020).
Softmax	$(0,1)$	Modelin tahminlerini olasılık formuna çevirir.	Aşırı güven problemi (overconfidence), yüksek hesaplama maliyeti.	Çok sınıflı sınıflandırma, doğal dil işleme, görüntü sınıflandırma (Gu et al., 2024).

2. YAPILAN ÇALIŞMALAR

Bu bölümde bu tez çalışması için oluşturulan kamuflaj veri seti ve kullanılan tüm yöntemler ve aşamaları anlatılmaktadır. Tez çalışması için yapılan tüm çalışmalar ana başlıklar halinde şu şekilde özetlenebilir:

- Kamuflaj görüntülerde kelebek nesnelerinin bölütlenmesi ve tanınması için hibrit bir yöntem geliştirilmesi (Bayram & Nabiye, 2025).
- Kamuflaj görüntülerinin sınıflandırılması için Transfer Öğrenme ve özellik birleştirme yaklaşımı
- Klasik Makine Öğrenmesi ve Evrişimsel Sinir Ağı Yöntemlerini Kullanarak Kamuflaj Görüntülerinin Sınıflandırılması İçin Karşılaştırmalı Bir Çalışma
- Kamuflaj görüntülerinin LBP, HOG ve klasik makine öğrenmesi yöntemleri kullanılarak sınıflandırılması (Bayram & Nabiye, 2024).
- Önceden eğitilmiş evrişimsel sinir ağı kullanarak kamuflaj görüntülerinin sınıflandırılması (Bayram & Nabiye, 2024).
- LBP yöntemi kullanarak kamuflaj görüntülerinin sınıflandırılması (Bayram & Nabiye, 2021).
- Kamuflaj görüntülerde Öklid, Mahalanobis ve K-Means kümeleme algoritması kullanılarak görüntü bölütlemesi (Bayram & Nabiye, 2020).
- Yerel İkili Desenlerle Özellik Çıkarımına Dayalı Kelebek Kamuflaj Görüntülerinin Sınıflandırılması (Bayram & Nabiye, 2021).

2.1. Kamuflaj Görüntülerde Kelebek Nesnelerinin Bölütlenmesi ve Tanınması İçin Hibrit Bir Yöntem Geliştirilmesi

Kamuflaj görüntülerinde kamufle edilmiş nesneleri tespit etmek, doku, desen ve renk özelliklerinin arka planla çok yakın olması nedeniyle oldukça zordur. Mevcut ikili bölütleme çözümleri, zayıf sınırlara ve arka plana çok yakın desenlere sahip oldukları için kamufle edilmiş nesneleri tespit etme sorunuyla kolayca başa çıkamaz. Bu çalışmada, arama motorlarından indirilen 10 kelebek türünün görüntülerinden oluşan ERVA 1.0 adlı orijinal bir kamuflaj kelebek veri kümesi oluşturulmuştur. Ayrıca, ham eğitim verisi veri artırma teknikleri ile artırılmıştır. COD için bu çalışma iki aşamalı bir çözüm sunmaktadır: bölütleme ve nesne tanıma. ERVA 1.0 veri setindeki tüm test görüntülerinin doku özellikleri,

bölütleme aşamasında kullanılmak üzere Gabor filtresi kullanılarak çıkarılmıştır. Daha sonra, çıkarılan bu özellikler K-ortalamalar algoritması ile kümelendir ve orijinal görüntü doku özelliklerine göre farklı bölgelere ayrılır. Bu bölgelerden hangisinin kamufle olmuş kelebek nesnesine ait olduğunu belirlemek için LBP algoritması ve Öklid uzaklık hesaplaması kullanılır. Belirlenen kelebek nesne bölgesi üzerinde morfolojik işlemler uygulandıktan sonra, kelebeğin türünü tahmin etmek için derin öğrenme tekniklerinden önceden eğitilmiş modeller kullanılır. Deneysel sonuçlara göre bölütleme başarı oranları SSIM ile %87,89 ve DSC yöntemi ile %83,64 olarak elde edilmiştir. Bölütlemeyen sonra elde edilen kelebek nesnesinin türünü belirlemek için derin öğrenme önceden eğitilmiş modeller kullanılmaktadır. Deney 1, artırılmamış ham eğitim verileriyle ve Deney 2, veri artırma teknikleri uygulanarak artırılmış verilerle gerçekleştirilmiştir. Deney 1 için en yüksek başarı oranı InceptionResNetV2 modeli ile %92,29 ve Deney 2 için en yüksek başarı oranı DenseNet121 modeli ile %94,81 olarak elde edilmiştir (Bayram&Nabiyev,2025).

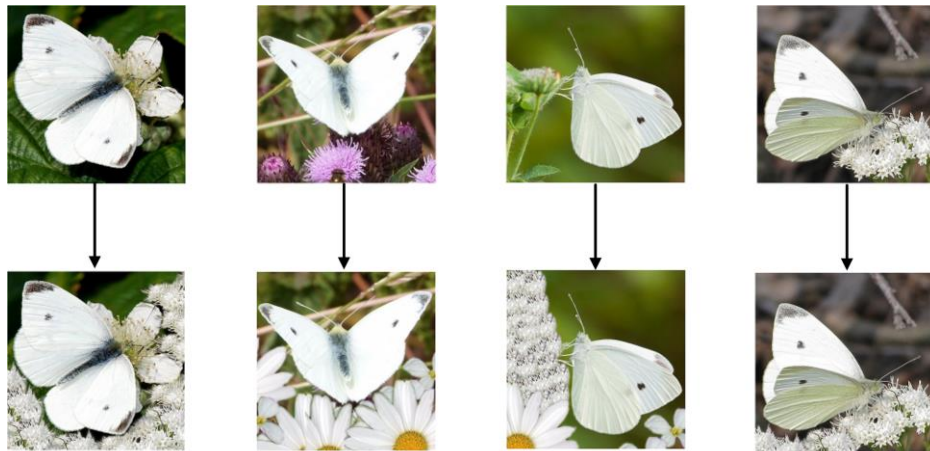
2.1.1. Yeni Kamuflej Veri Seti Oluşturma

Bu çalışmada, ERVA 1.0 adı verilen ve kamuflej kelebek görüntülerinden oluşan yeni bir veri seti oluşturulmuştur. ERVA 1.0 kamuflej kelebek veri seti eğitim ve test verileri olmak üzere iki bölümden oluşmaktadır. ERVA 1.0 oluşturulmadan önce 10 farklı kelebek türünün sınıfları belirlenmiştir. Bu 10 farklı türe ait binlerce görüntü, arama motorlarından daha önceden belirlenen bu sınıflara göre aranarak indirildi. İndirilen görüntüler üzerinde gerekli sınıflandırmalar yapılarak bu on farklı türe ait 1180 eğitim ve 597 test kelebek görüntüsü elde edilmiştir. Veri kümesindeki tüm görüntüler orijinallliğini kaybetmeden kırılmış veya 400*400 piksel boyutuna ayarlanmıştır.

ERVA 1.0 veri seti, doğal kamufleli etkili bir şekilde temsil etmek için çeşitli kelebek türlerinden ve farklı doğal arka planlardan oluşur. Veri seti 10 farklı türe ait kamuflej kelebek görüntüleri içermektedir. Görüntüler doğal ortamlarda çekilmiş ve farklı aydınlatma koşulları, bitki örtüsü, toprak yapısı ve diğer çevresel faktörler dikkate alınarak seçilmiş ve çeşitlendirilmiştir. Bu da veri setinin gerçek dünyadaki kamuflej zorluklarını temsil etmesini sağlamaktadır. Veri setindeki bazı türlerin doğal yaşam alanlarının diğerlerine göre daha çeşitli olması nedeniyle görüntü sayıları birbirine yakın seçilmeye çalışılsa da tüm türler için görüntü sayıları aynı değildir. Ancak genel olarak veri setindeki türlere ait görüntü sayıları birbirine yakındır. ERVA 1.0 kamuflej görüntü veri seti, kamufle olmuş kelebek

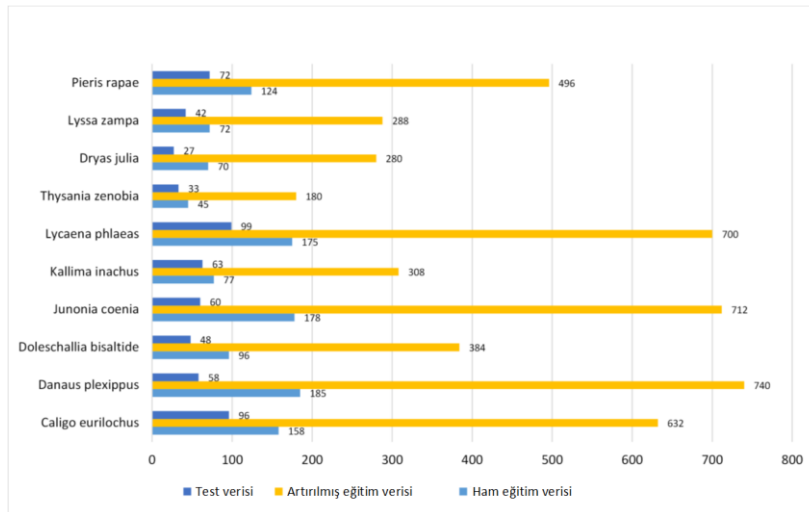
nesnelerinin tespiti için oluşturulmuştur. Bu doğrultuda, bu görüntüler farklı kamuflaj desenlerini ve bunların doğal ortamlardaki çeşitliliğini temsil edecek şekilde seçilmiştir. Veri setindeki görüntüler farklı aydınlatma koşulları, arka plan dokuları ve perspektifler içerecek şekilde çeşitlendirilmiştir. Bu, modelimizin farklı senaryolara genelleştirilebilmesini sağlar. Görüntüler arama motorlarından ve açık erişimli veri tabanlarından elde edilmiştir. Görüntüler seçilirken ağaç, çiçek ve toprak gibi arka planları olan ve farklı desen, renk ve dokulara sahip kamuflaj kelebek örneklerinin seçilmesine özen gösterilmiştir. Kamuflaj kelebek görüntüleri için yüksek çözünürlüklü ve net görüntüler seçilmiştir. Kamuflaj görüntülerindeki tüm kamuflajlı kelebek nesnelerinin sınırları MATLAB programı yardımıyla fare kullanılarak elle manuel olarak çizilmiş ve kelebeklerin sınırları etiketlenmiştir. Veri setinde çok sayıda görüntü bulunması nedeniyle bu işlem oldukça uzun sürmüştür. Kelebek türleri seçilirken iyi kamufle olan kelebek türlerinin seçilmesine özen gösterilmiş ve kamuflaj problemi için oldukça zorlu bir veri seti oluşturulmuştur.

ERVA 1.0 kamuflaj kelebek görüntülerinde kamuflajlık derecesi düşük olan görüntülerin kamuflajlık derecesi artırılmıştır. Şekil 19'da veri setinde bir türe ait örnek görüntülerde gösterildiği üzere bu işlem için başka görüntülerden alınan aynı veya benzer renklere sahip çiçek, toprak, taş gibi nesneler fotoğraf düzenleme editörleri ile görüntü arka planına eklenmiştir. Burada amaç, kamuflajlık derecesi düşük olan görüntülerin arka planına kelebek görüntülerine yakın desenler veya dokular ekleyerek görüntünün kamuflajlık derecesini artırmaktır.



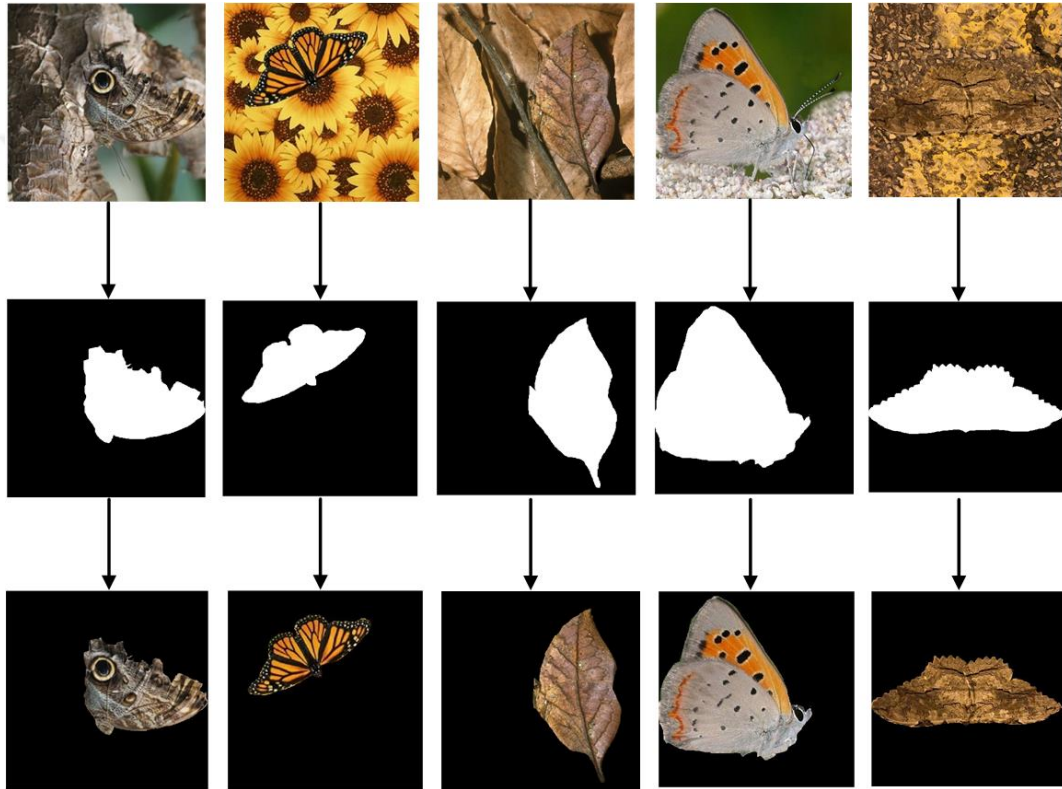
Şekil 19. Kamuflajlık derecesi düşük ve artırılmış örnek görüntüler

Bu çalışmada, kamuflaj görüntülerine bir veri artırma tekniği olarak görüntü arka planına kamufle olmuş nesneye çok yakın desen, renk ve doku eklemenin yanı sıra geometrik dönüşümler uygulanmıştır. Geometrik dönüşümler görüntünün konum, ölçek ve yönelim gibi fiziksel özelliklerini değiştirmektedir. Veri artırımı için yatay simetri, dikey simetri ve belirli açılarla sağa döndürme geometrik dönüşümleri uygulanmıştır. Yatay simetri işlemi görüntüyü yatay olarak tersine çevirir. Örneğin, bir nesne sola dönükse, yatay simetriden sonra sağa dönük görünecektir. Yatay simetrinin avantajı, görüntüdeki yapıları farklı perspektiflerden sunmasıdır. Bu yöntem, kamufle edilmiş nesnelerin bulunduğu kamuflaj görüntülerinde veri artırımı için yaygın olarak kullanılmaktadır. Dikey simetri işlemi görüntüyü dikey olarak ters çevirir. Bu, görüntünün üstünü ve altını ters çevirir. Geometrik dönüşümler uygulanarak gerçekleştirilen veri artırımı ile veri setimizdeki kamuflajlı nesnelerin farklı görünüşleri oluşturulmuş ve önerilen modelimizin genelleme kabiliyetinin artırılması amaçlanmıştır. Kamuflaj görüntülerimizdeki kamufle edilmiş nesneler, önerilen modelimizin daha iyi öğrenebilmesi için farklı yönlerden geliştirilmiştir. Bu çalışmadaki veri artırımı ile veri setimizin genişletilmesi, modelimizin genelleme kabiliyetinin artırılması ve aşırı uyum riskinin azaltılması amaçlanmıştır. Eğitim verileri için, 400*400 piksel boyutuna ayarlanmış tüm görüntülere veri artırma teknikleri uygulanmış ve bir görüntüden üç görüntü daha türetilmiştir. Böylece eğitim verisi için görüntü sayısı 1180'den 4720'ye çıkarılmıştır. Şekil 20'de ERVA 1.0 veri setindeki kelebek türlerine göre veri artırımı yapılmadan ve yapıldıktan sonraki görüntü sayıları için istatistiksel analiz gösterilmektedir.



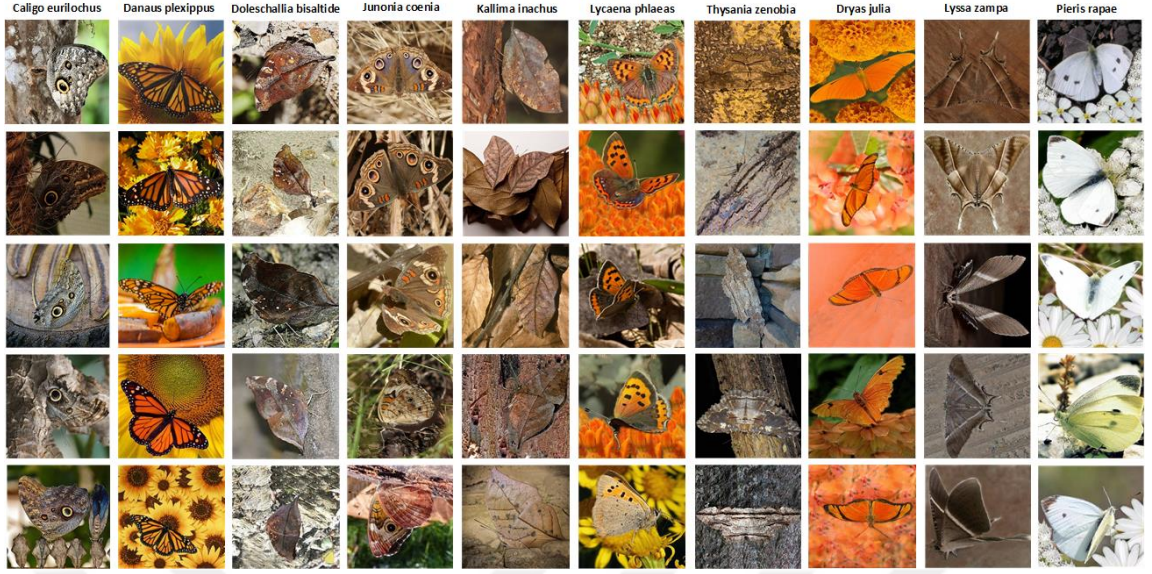
Şekil 20. ERVA 1.0 kamuflaj veri setinde yer alan türlere ait sayısal analiz gösterimi

Şekil 21’de örneklerde gösterildiği gibi ERVA 1.0 veri setindeki tüm görüntülerdeki kelebek nesnelerinin sınırları bir yazılım kullanılarak elle manuel olarak çizilmiş ve ikili (binary) görüntüler olarak ayrı bir klasöre kaydedilmiştir. Kamufle olmuş nesnelerin sınırlarının önceden belirlenmesi bölütleme sonucu elde edilen nesnenin olması gerekenle karşılaştırılıp bölütleme başarı oranlarının hesaplanması için oldukça önemlidir. Bunun yanı sıra sadece kamufle olmuş nesnenin ait olduğu bölgenin bilinmesi yapılacak çalışmalar için çok büyük kolaylıklar sağlayacaktır.



Şekil 21. ERVA 1.0 kamuflej veri setindeki kamufle olmuş nesnelerin sınırlarının belirlenmesi için örnek görüntüler

Şekil 22’de ERVA 1.0 veri setindeki tüm türlere ait kamuflej kelebek görüntüleri için örnek görüntüler gösterilmiştir.



Şekil 22. ERVA 1.0 kamuflaj veri setindeki türlere ait örnek görüntüler

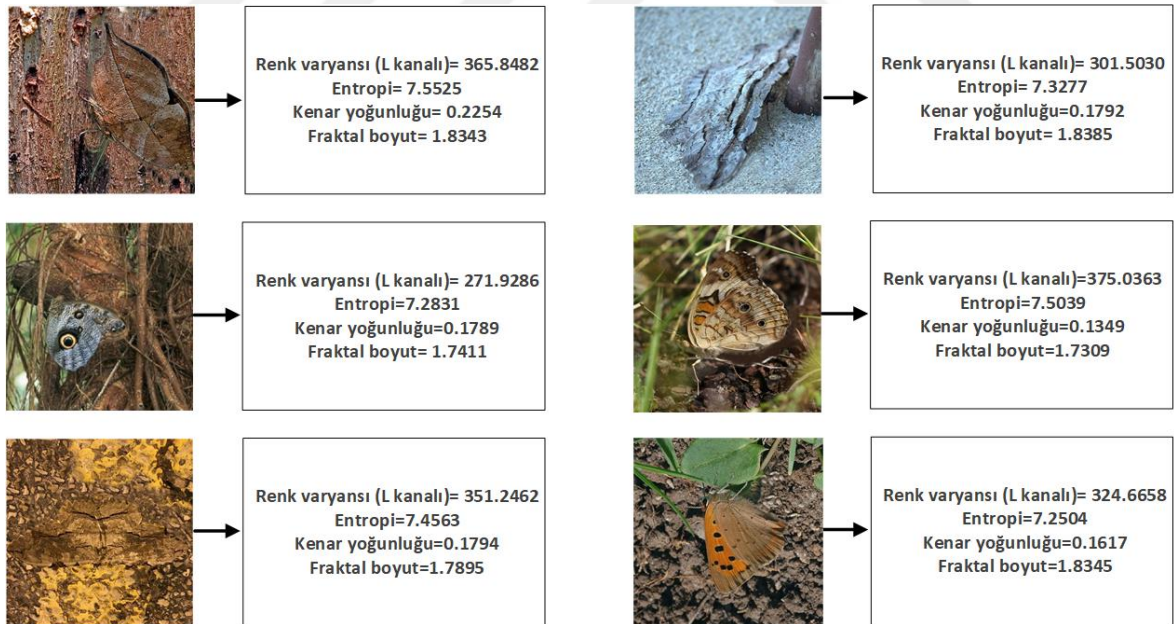
ERVA 1.0 veri seti oluşturulurken görüntülerin göz kararı ile seçilmesinin yanı sıra görsel karmaşıklık metrikleri ve renk histogramları ölçümleri kullanılmıştır. Bu metrikler, özellikle kamuflaj gibi görsel algıyı etkileyen özellikleri matematiksel olarak analiz etmek için kullanılmıştır. Bu metrikler Görsel Karmaşıklık Metrikleri ve Renk Histogramları olarak sıralanabilir.

Görsel Karmaşıklık Metrikleri: Görsel karmaşıklık, bir görüntünün ne kadar “karmaşık” görüldüğünü ölçen metriklerdir. İnsan algısına göre bazı desenler sadeyken, bazıları daha detaylı, renkli ve düzensiz olabilir. Kamuflajda, bu karmaşıklık hedef nesnenin (örneğin kelebeğin) arka plana karışmasını kolaylaştırır. Görsel karmaşıklık metrikleri olarak ERVA 1.0 veri seti oluşturulurken görüntü seçiminde kullanılan ölçütler aşağıdaki gibi sıralanabilir:

- *Renk varyansı (çeşitliliği):* LAB uzayında özellikle L (aydınlık) kanalının varyansı, görüntüdeki parlaklık değişimini yansıtır. Kamuflaj görüntülerde, kelebeğin arka plana karışması için parlaklık farkları çok yüksek olmamalıdır. Dolayısıyla, L varyansı genelde düşük-orta aralıkta olur. L varyansı, yaklaşık olarak 100–500 ölçüm birimi kullanılan görüntüye bağlı olarak normalize edilebilir.
- *Entropi:* Piksel yoğunluklarının belirsizliğini ölçer. Kamuflaj desenleri genelde karmaşık ve düzensiz olur. Bu da yüksek entropi anlamına gelir. Kamuflaj görüntülerde entropi değerinin yaklaşık 6.0-7.8 aralığında olması beklenir.

- *Kenar yoğunluğu*: Görüntüdeki yapısal kenarların yoğunluğunu gösterir. Kamuflajlı desenlerde hem kelebeğin hem arka planın kenar yapıları birbirine benzediğinden orta-yüksek kenar yoğunluğu beklenir. Ancak çok yüksek olursa yapay görünebilir. Kamuflaj görüntülerde kenar yoğunluğu için beklenen değer (görüntüdeki piksellerin kenar kabul edilen yüzdesi) aralığı yaklaşık %15 – %30 aralığıdır.
- *Fraktal boyut*: Görüntüdeki karmaşık yapının boyutsal karmaşıklığını ölçmek için kullanılır. Doğal desenler ve kamuflajlı görüntüler genellikle fraktal özellikler taşır. Dolayısıyla, fraktal boyut orta-yüksek olmalıdır. Kamuflaj görüntülerde Fraktal boyut için beklenen aralık yaklaşık 1.5 – 1.9 aralığıdır.

ERVA 1.0 veri setinde yer alan kamuflaj kelebek görüntülerinin seçilmesinde yukarıda verilen metrikler için kamuflaj görüntülerde olması gereken değer aralıklarına sahip görüntülerin seçilmesine özen gösterilmiştir. Görsel Karmaşıklık Metrikleri kullanılarak elde edilen değerler için örnek görüntüler Şekil 23’de verilmiştir.



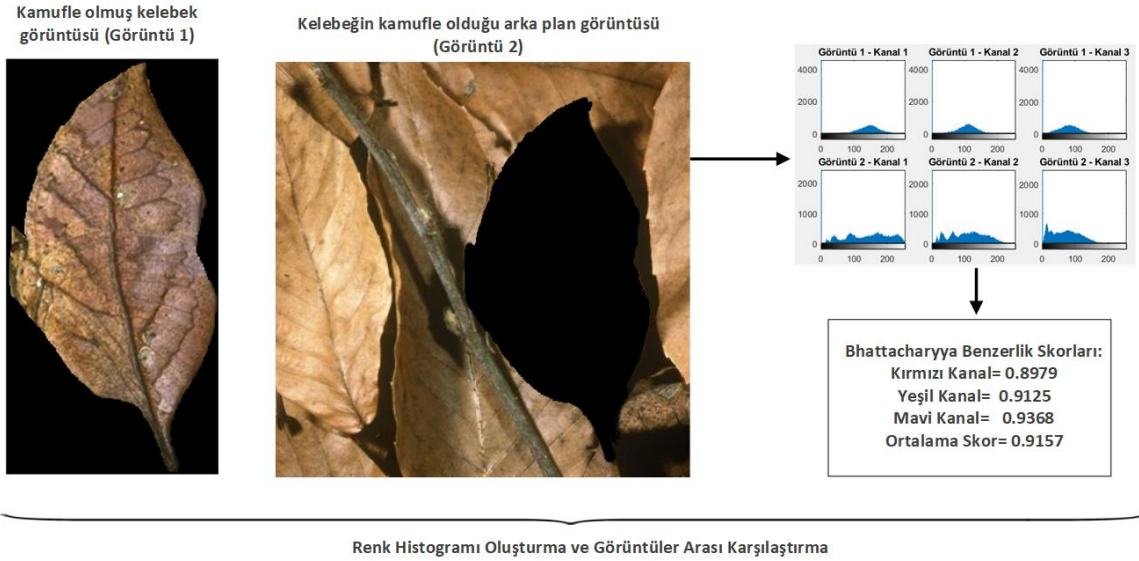
Şekil 23. ERVA 1.0 kamuflaj veri setindeki görüntülere ait örnek Görsel Karmaşıklık Metrik değerlerinin gösterilmesi

Renk Histogramları: Bir görüntünün içinde hangi renklerin, ne sıklıkta bulunduğunu gösteren grafiksel bir temsildir. Kamuflaj analizinde özellikle önemlidir çünkü nesne ile arka

palanın renk benzerliği, arka planla uyumu göstermektedir. Renk Histogram benzerliği yüksekse, görsel olarak ayırt etme zorlaşır, bu da etkili kamuflaj anlamına gelir. Renk Histogramları şu şekilde hesaplanmaktadır:

- Görüntüdeki her pikselin rengi, belirli aralıklara (örneğin 256 seviyeye) ayrılır.
- Her renk için kaç piksel bulunduğu sayılır ve histogram oluşturulur.
- İki farklı görüntünün histogramları karşılaştırılarak benzerlik dereceleri ölçülür (örneğin, kelebek nesnesi ve arka plan).
- Bhattacharyya benzerliği değeri 0–1 arasında olup, 1'e ne kadar yakınsa histogramlar o kadar benzerdir yani renk dağılımı da o kadar benzerdir.

Şekil 24'te ERVA 1.0 kamuflaj görüntülerinde yer alan kamuflle olmuş nesne ve arka plan görüntülerinden örnek bir görüntüye ait renk histogramları için hesaplanan Bhattacharyya Benzerlik Skorları hesaplanmış ve örnek olarak verilmiştir.

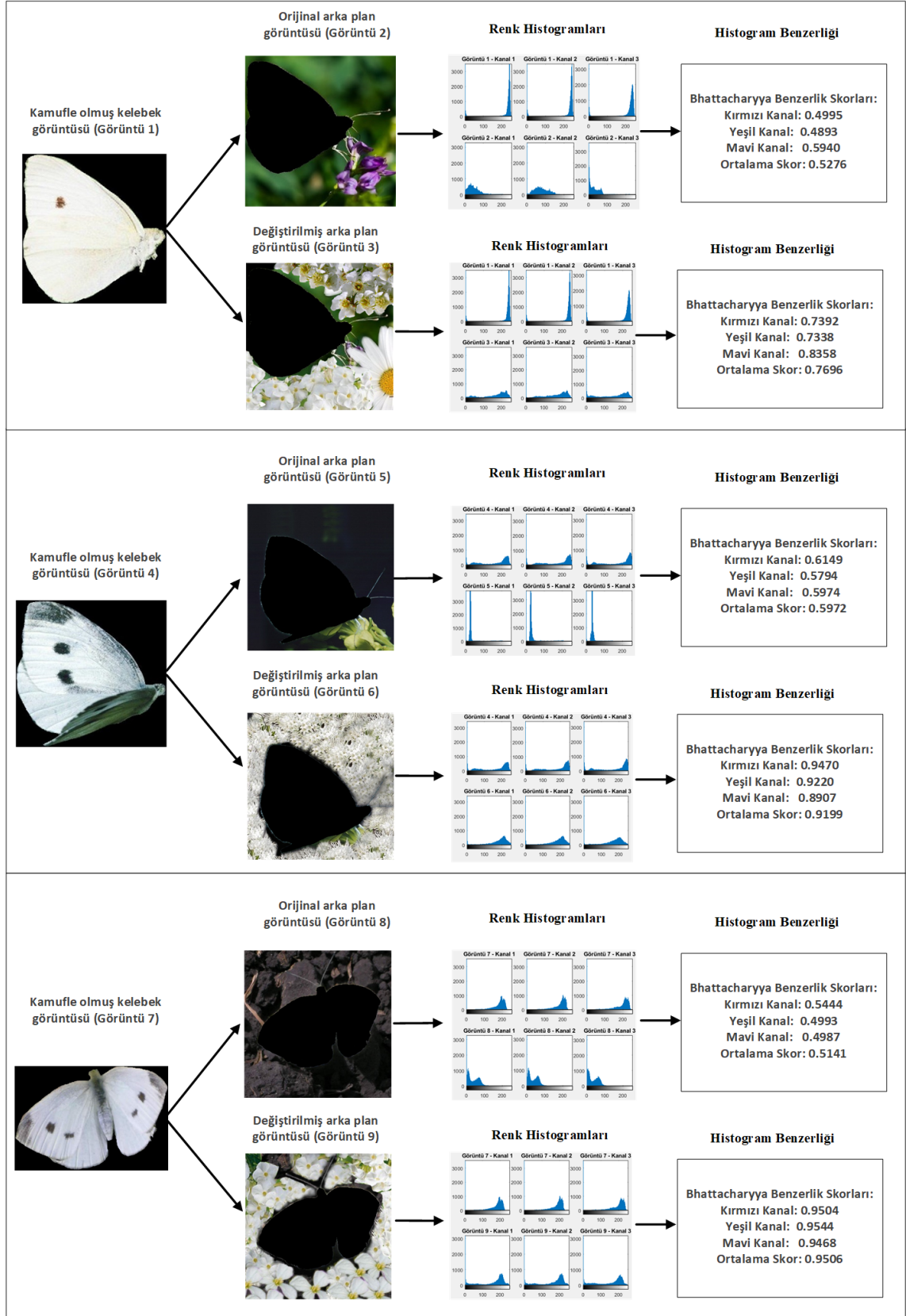


Şekil 24. ERVA 1.0 kamuflaj veri setindeki kamuflle olmuş nesne ve arka plan görüntüsüne ait örnek Renk Histogramı Oluşturma ve Görüntüler arası karşılaştırma yapılması

Veri Artırımının Kamuflaj Zorluk Derecesine Etkisinin Matematiksel Analizi:

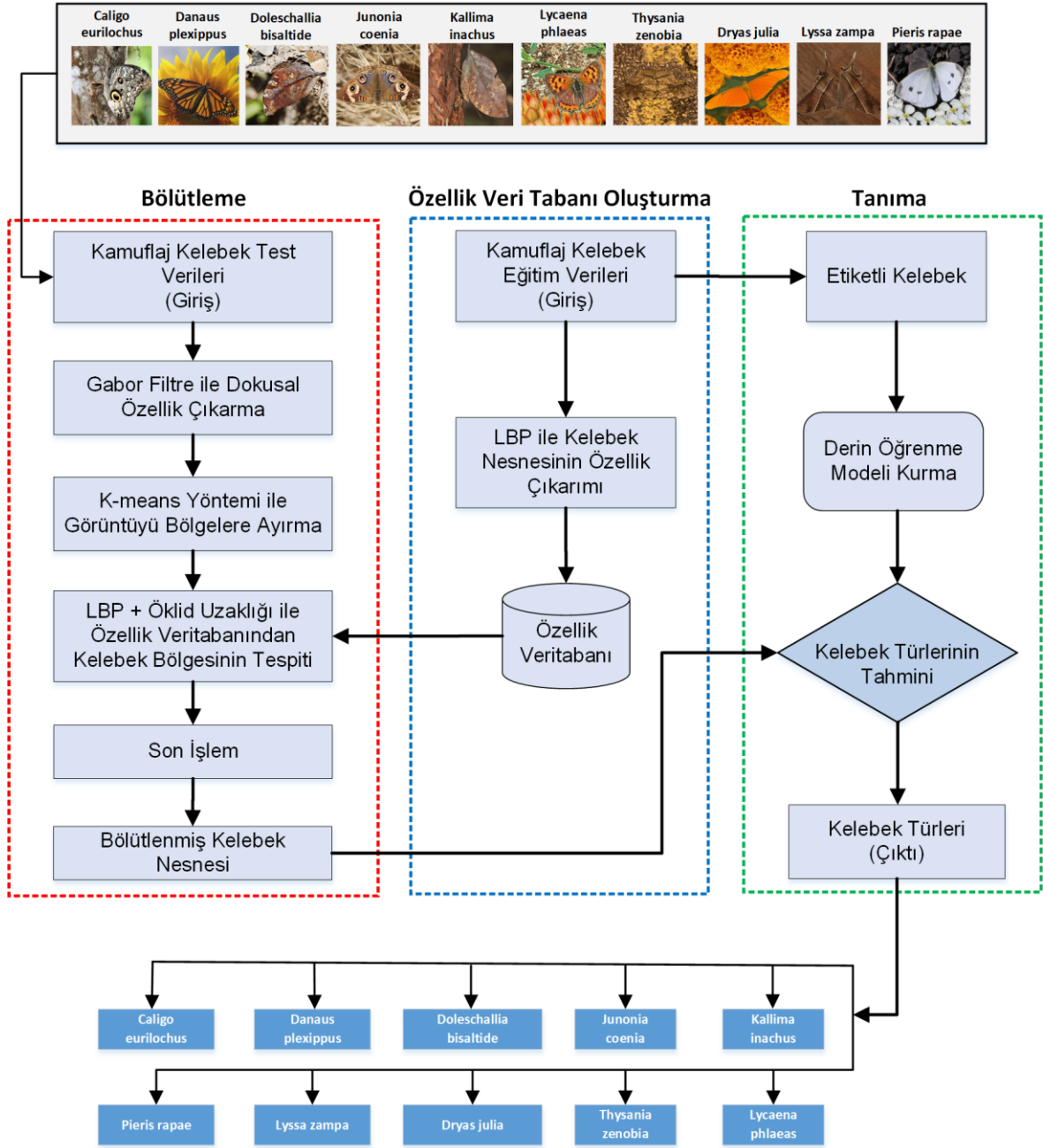
Veri artırma teknikleri, makine öğrenimi ve derin öğrenme modellerinin genelleme yeteneğini artırmak ve özellikle veri yetersizliği olan alanlarda modelin performansını iyileştirmek amacıyla sıklıkla kullanılır. Kamuflajlı nesne tespiti gibi zorluk derecesi yüksek görsel görevlerde, bu tekniklerin hem tespit doğruluğu hem de zorluk derecesine etkisi çeşitli parametrelerle değerlendirilebilir. Kamuflle olmuş nesne ile arka planı ne kadar birbiri içine

girift olur ne kadar dokusal, renk ve desen olarak birbirine yakın özellikler gösterirse tespit edilmesi o kadar zor olur ve kamuflajlık derecesi o kadar fazla olur. Kamuflaj olmuş nesneye yakın doku, renk ve desenlerle görüntü arka planına ekleme yapıldıktan sonra yeni görüntünün kamuflajlık derecesinin değişiminin matematiksel analizinin yapılması için Renk Histogramları kullanılmıştır. Şekil 25’de ERVA 1.0 veri setindeki kamuflajlık derecesinin artırılması için arka plana eklenmeden önce ve eklendikten sonraki görüntülere ait Renk Histogramları kullanılarak elde edilen örnek hesaplamalar gösterilmiş ve karşılaştırılmıştır. Şekil C incelendiğinde görüntü arka planına kamuflaj olmuş nesne benzeri nesnelerin, dokuların eklenmesiyle görüntünün kamuflajlık derecesinin artırıldığı görülmüştür. Şekil 25’ de kamuflaj olmuş nesne ile orijinal arka plan ve değiştirilmiş arka plan görüntüleri için ayrı ayrı renk histogramları çıkarılmış ve bu histogramların karşılaştırması yapılmıştır. Histogram benzerliklerinin hesaplanması için Bhattacharyya Benzerlik Skorları kullanılmıştır. Bu skorlar incelendiğinde kamuflaj olmuş nesne ile orijinal arka plan için hesaplanan skorlar genellikle 0.5 değerlerine yaklaşık düzeylerde dir. Arka plan görüntüsüne kamuflaj olmuş nesneye yakın doku, desen ve renkte nesneler eklendiğinde oluşan yeni arka plan görüntüsü ile kamuflaj olmuş nesne görüntüsü için ayrı ayrı renk histogramları çıkarıldığında histogramların birbirine çok yaklaştığı görülmektedir. Kamuflaj olmuş nesne ile arka plan değiştirilen görüntülerin renk histogramlarından elde edilen Bhattacharyya Benzerlik Skorları incelendiğinde 1 değerine çok yaklaştığı görülmektedir. Bu skorlar 1 değerine ne kadar çok yakınsa kamuflaj olmuş görüntü ile arka plan birbirine o kadar çok yakın özelliklere sahip demektir. Şekil 4’teki kamuflaj olmuş görüntüler, orijinal arka plan görüntüleri ve değiştirilmiş arka plan görüntülerinden elde edilen renk histogramları ve Bhattacharyya Benzerlik Skorları incelendiğinde görüntü arka planına kamuflaj olmuş nesneye yakın özelliklerde desen, doku ve renkteki nesnelerin eklenmesinin görüntülerin kamuflajlık zorluk derecesini iyi düzeyde artırdığı görülmektedir.



Şekil 25. ERVA 1.0 kamuflaj veri setindeki görüntülere veri artırma işleminin Kamuflaj zorluk derecesine katkısının hesaplanması

2.1.2. Önerilen Yöntem



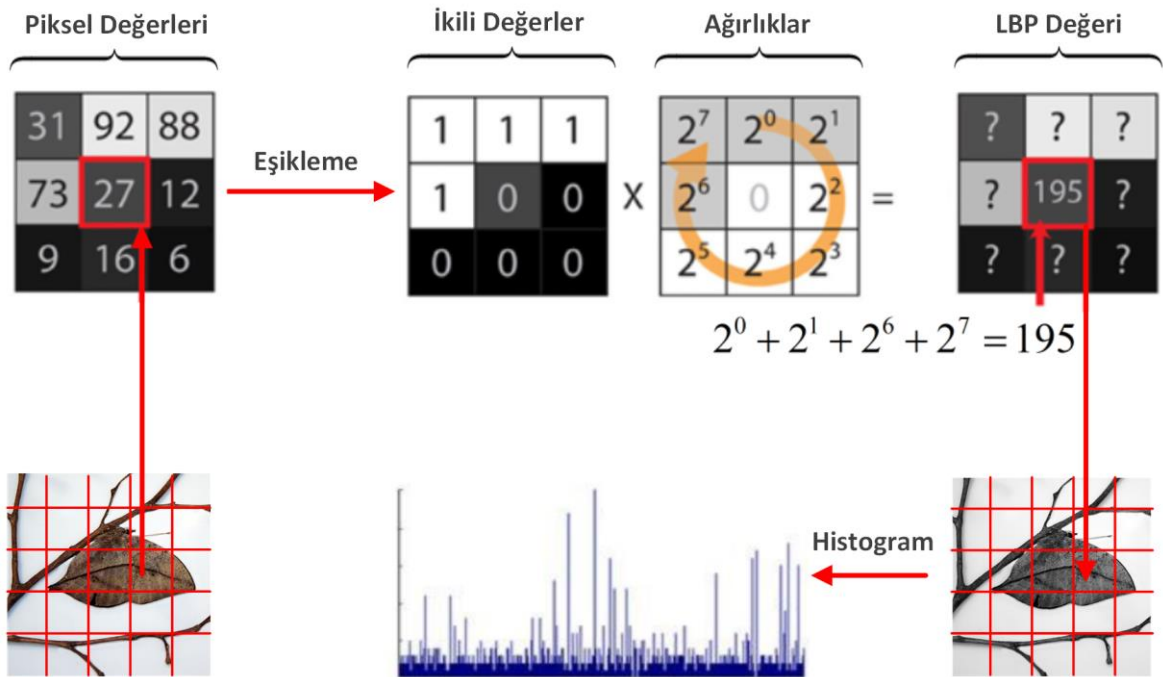
Şekil 26. Çalışmanın genel metodolojik adımlarının akış şeması

Bu çalışmada Gabor Filtre, LBP algoritması, K-means algoritması ve derin öğrenme önceden eğitilmiş modeller kullanılarak hibrit bir yöntem önerilmiştir. Yöntemin genel akış şeması Şekil 26'da gösterilmiştir. Şekil 26'da görüleceği üzere sistemi test etmek için oluşturulan ERVA 1.0 kamuflaj test görüntülerinden bir görüntü ilk olarak sisteme girdi olarak verilir. Sisteme girdi olarak verilen görüntünün doku özellik vektörleri Gabor filtresi

ile çıkarılır ve bu özellik vektörleri için K-means yöntemi ile kümeleme yapılır. K-means ile özellikler açısından birbirine en yakın olan bölgeler aynı kümeye dahil edilir. Böylece kamuflaj test görüntüsü dokusal özelliklerine göre farklı bölgelere ayrılır. Bu farklı bölgelerden hangisinin kelebek nesnesine ait olduğuna karar vermek için özellik veri tabanından yararlanılır. Özellik veri tabanı, tüm eğitim görüntülerindeki etiketli kelebek görüntülerinin LBP algoritması ile elde edilen tüm özellik vektörlerinin toplanmasıyla oluşturulur. Her farklı bölgenin özellik vektörleri LBP algoritması ile çıkarılır. Hesaplanan bu özellik vektörü daha sonra sırasıyla Öklid mesafesi kullanılarak özellik veri tabanındaki tüm özellik vektörleri ile arasındaki sayısal farkı hesaplamak için kullanılır. Hesaplanan sayısal farkın sıfıra en yakın olduğu bölge kelebek bölgesi olarak kabul edilir. Kelebek bölgesi tespit edildikten sonra, bu bölgenin sınırları içinde yer alan ancak bölgeye komşu olmayan birkaç bağlantısız pikselden oluşan parçacık görüntülerine son işlem (post-processing) uygulanır. Bu işlem sonucunda bu küçük aykırı değerler elenir. Post-processing işleminden sonra bölütleme işlemi tamamlanmıştır. Daha sonra sisteme girilen kamuflaj görüntüsü üzerinde bölütleme işlemi yapılarak kelebek nesnesi tespit edilmiştir. Bu aşamadan sonra tespit edilen kelebek nesnesinin türünün tanınmasına geçilir. Bunun için eğitim veri setindeki her bir kelebek için manuel olarak etiketlenmiş kelebek bölgesi verileri kurulan derin öğrenme modeline gönderilir. Derin öğrenme modeli, bölütleme sonucunda elde edilen kamuflaj olmuş kelebek nesnelerinin türlerini tahmin etmektedir.

Özellik Veri Tabanı Oluşturma: Bu çalışmada, aynı sınıftaki görüntülerin ayırt edici özelliklerini tespit etmek amacıyla kamuflaj görüntülerde özellik çıkarımı yapılmaktadır. Aynı görüntü sınıfında yer alan nesnelerin dokusal özellikleri birbirine benzer ve yakındır. Çalışmanın bu aşamasında ERVA 1.0 veri setindeki tüm eğitim verilerine ait 4720 görüntünün dokusal özellikleri LBP tabanlı yöntem kullanılarak çıkarılır. Çıkarılan tüm bu dokusal özellikler daha sonra özellik veri tabanına kaydedilir. Özellik veri tabanı, kamuflajlı kelebek görüntüsüne uygulanan bölütleme işlemi sırasında, kelebek nesnesinin bölge sınırlarının tespit edilebilmesi için kullanılmaktadır. LBP yöntemi çok çeşitli uygulamalarda kullanılmaktadır. Bir özellik çıkarma yöntemi olan LBP, ilk olarak 1996 yılında Ojala tarafından geliştirilmiştir (Ojala vd., 1996). LBP özellik çıkarımı, doku tanımlama (Kazak, 2016), yüz tanıma (Ahonen vd., 2006), yaş tespiti (Nabiyev & Günay, 2010), cinsiyet tespiti (Lian & Lu, 2006), yüz ifadesi tanıma (Shan vd., 2009), kamuflaj görüntü sınıflandırma (Bayram & Nabiyev, 2021) gibi uygulamalarda kullanılmaktadır.

LBP gri görüntüdeki her pikselin 3x3 komşuluklarını karşılaştırarak ikili kodlar üretmektedir. 3x3 lük görüntü dairesel olarak ele alınmakta ve merkezde yer alan pikselin değeri eşik değer olarak kabul edilmektedir. Şekil 27 LBP değerlerinin hesaplanması için örnek bir uygulama göstermektedir. Görüntüdeki her bir piksel için komşuluğunu merkez piksel değeriyle karşılaştırarak bir ikili kod elde edilir. Bu ikili kod ikili bir model olarak düşünülebilir. Eğer piksel değeri eşik değerinden büyükse veya eşitse komşu piksel 1 olur. Eğer piksel değeri eşik değerinden küçükse komşu piksele 0 değeri atanır. Daha sonra ikili modellerin frekans değerlerini belirlemek için histogram oluşturulur. Her desen görüntüde yer alan ikili desen olasılığını temsil etmektedir. Histogram bölmelerinin sayısı LBP hesaplamasındaki piksellerin sayısına bağlıdır. Örneğin LBP 8 piksel kullanıyorsa histogram bölme sayısı 256 olacaktır (Prakasa,2016).



Şekil 27. LBP operatörünün hesaplanması için örnek bir örnek uygulama (Bayram & Nabyev, 2025)

Bir pikselin $(x_i; y_i)$ konumuna bakılacak olursa LBP, merkez piksel ile onun etrafında yer alan komşu pikseller arasındaki değerleri karşılaştırır. LBP operatörü için kullanılan eşitlik denklem (29) de görüldüğü gibidir. Denklem (29)' da; $(x_i; y_i)$ merkez pikselin koordinat değerleri, g_i merkezde yer alan pikselin gri değerini, g_n de merkez

pikselin etrafındaki 3x3 komşuluğundaki piksellerin değerlerini göstermektedir. $f(k)$ Fonksiyonu ise gri seviye piksellerin 8 komşuluk değerlerini gösterir (Zhang vd., 2005).

$$LBP(x_i, y_i) = \sum_{n=0}^{n-1} f(g_n - g_i) 2^n \quad (29)$$

$$f(k) = \begin{cases} 1, & \text{eğer } k \geq 0 \\ 0 & \text{eğer } k < 0 \end{cases} \quad (30)$$

Oluşturulan LBP değerlerinin tamamı doku tanımlama işleminde kullanılmaz. Tanımlama için kullanılan düzgün örüntüler, ikili LBP kodunda 0-1 ya da 1-0 geçişi sayısı iki ya da daha az olanlardır (Nabiyev & Günay, 2010). Tablo 2’de düzgün örüntüler için örnekler gösterilmiştir.

Tablo 2. Düzenli ve düzensiz örüntü örnekleri

Örüntü Değeri	Geçiş Sayısı	Örüntü Düzgün mü?
00000000	0	Evet
11111111	0	Evet
01110000	2	Evet
11001111	2	Evet
11001000	4	Hayır
01010011	6	Hayır

Bu çalışmada kamuflej görüntüler farklı bloklara ayrılıp her bir blok için LBP operatörü ayrı ayrı hesaplanmış ve histogramlar elde edilmiştir. Görüntülerin ayırt edici özelliklerini bulmak için bölgesel histogramlar kullanılır. Bu aşamada kamuflej kelebek görüntüsü 8x8=64 dikdörtgensel bölgeye ayrılıp, her bölge için LBP kodları ayrı ayrı hesaplanıp histogramlar elde edilmiştir. Elde edilen bu histogramlar peş peşe eklenerek bir öznitelik vektörü ortaya çıkarılmaktadır. Daha sonra elde edilen bu vektör bölütleme aşamasında kelebek nesnesinin bölgesinin tespitinde kullanılmaktadır. Veri setindeki her bir görüntü için 3776 boyutunda bir öznitelik vektörü elde edilmektedir.

LBP ve Yönlendirilmiş Gradyanların Histogramı (HOG) yaygın olarak kullanılan doku ve özellik çıkarma yöntemleridir. Ancak, bu algoritmaların uygunluğu belirli sorun alanına göre değişebilir. LBP, kamufle edilmiş nesneleri benzer arka planlarından ayırt

etmek için çok önemli olan ince taneli yerel doku desenlerini yakalamak için özellikle etkilidir. Buna karşılık, HOG, net sınırları olan sahnelerde nesne algılama gibi kenar tabanlı özellikler ve yapılandırılmış geometriler içeren görevler için daha uygundur. Kamufle edilmiş kelebekler genellikle belirgin kenarlardan yoksun olduğundan ve farklılaştırma için ince doku ipuçlarına büyük ölçüde bağımlı olduğundan, LBP bölütleme ve sınıflandırma hedeflerimizle daha iyi uyumludur. LBP'nin ışık ve kontrasttaki değişikliklere karşı sağlamlığı, farklı koşullara sahip görüntü veri setimiz için uygunluğunu daha da desteklemektedir. Literatürde LBP ve HOG ile kamuflej görüntüleri üzerine yapılan çalışmalar incelendiğinde, LBP algoritması kullanıldığında çok daha başarılı sonuçların elde edildiği görülmektedir (Bayram & Nabiye, 2024). Sonuçlar, LBP'nin kamuflej desenlerinin tespitini doğrudan etkileyen doku farklılaştırmasında daha yüksek doğruluk elde ettiğini göstermiştir. Tüm bu nedenlerden dolayı, bu çalışmada HOG yerine LBP yöntemi tercih edilmiştir.

Kamuflej Görüntülerinin Bölütlenmesi: Bilgisayarlı görü için bölütleme, bir dijital görüntüyü piksel kümelerinden oluşan birden çok bölgeye bölme işlemini ifade etmektedir. Bölütlemenin amacı, bir görüntünün temsiliyi basitleştirmek, analiz edilmesi daha anlamlı ve daha kolay bir şekle dönüştürmektir (Stockman & Shapiro, 2001). Görüntü bölütleme genellikle görüntülerdeki nesneleri ve çizgi, eğri gibi sınırları bulmak için kullanılır. Görüntü bölütlemenin sonucu, toplu olarak tüm görüntüyü kapsayan bir dizi bölge veya görüntüden çıkarılan bir kontür kümesidir. Bir bölgedeki piksellerin her biri renk, yoğunluk veya doku gibi bazı karakteristik veya hesaplanmış özelliklere göre benzerdir. Bitişik bölgeler aynı karakteristikler açısından önemli ölçüde farklıdır. Görüntü bölütleme için çeşitli genel amaçlı algoritmalar ve teknikler geliştirilmiştir. Görüntü bölütleme problemine genel bir çözüm olmadığından, bu tekniklerin bir problem alanı için görüntü bölütleme problemini etkili bir şekilde çözebilmesi için genellikle alan bilgisi ile birleştirilmesi gerekir (Wang, 2008).

Görüntü bölütlenmesi genellikle gözlemci için önemli olan bir görüntüdeki bölgeleri ana hatlarıyla belirtmeyi veya sınıflandırma veya modelleme yoluyla daha kesin bilgilerin çıkarılmasını kolaylaştıran bölümler oluşturmayı amaçlar (Baatz vd., 2008). Bölütleme, bir görüntünün bileşenlerini kenar, doku ve renk gibi belirli özelliklere göre anlamlı parçalara ayırır. Bu, görüntüyü daha anlaşılır ve analiz edilmesi daha kolay hale getirir. Literatürde bölütleme için birçok yöntem kullanılır. Bu yöntemler kenar tabanlı, eşik tabanlı, bölge

tabanlı, kümeleme tabanlı, aktif kontur tabanlı ve derin öğrenme tabanlıdır (Jasim & Mohammed, 2021).

Bu çalışmada, kamuflaj görüntülerinin bölütlenmesi için Gabor Filtre ile çıkarılan öznelikler kümeleme tabanlı bir yöntem olan K-means yöntemi ile kümelendirilmiştir. K-means yöntemi bölütlemeye başarılı olmasından dolayı yaygın olarak kullanılmaktadır. K-means yöntemi ile görüntüde belirli sayıda rastgele küme ve küme merkezi belirlenmektedir. Her piksel için Öklid uzaklığı hesaplanır ve merkeze en yakın pikseller o kümeye dahil edilir. Bu işlemi tekrarlayarak yeni küme merkezleri belirlenir ve bölütleme minimum küme sayısına ulaşılan kadar devam eder (Kanungo et al., 2002).

Gabor Filtresi ile Doku Özelliği Çıkarımı: Gabor Filtre tabanlı teknikler, doku analizi ve bölütleme (Raghu & Yegnanarayana, 1996; Clark vd., 1987), kenar algılama (Namuduri vd.,1994), nesne tanıma (David vd., 1992; Jain vd.,1997), görüntülerin gösterimi (Porat & Zeevi, 1988) ve el yazısı rakamlar ve alfabe tanıma (Hamamoto vd., 1998) gibi çok çeşitli görüş tabanlı uygulamalarda yaygın olarak kullanılmaktadır. Gabor çekirdekleri bant geçiren filtrelerdir ve görsel yapısındaki çoklu kanal filtrelemesi ile temel görme bilgisi işleme anlayışı tarafından yönetilir (De Valois vd., 1982). Karmaşık bir Gabor filtresi, Gauss Kernel ile karmaşık bir sinüsoidin çarpımı şeklinde tanımlanmaktadır. 2 boyutlu bir Gabor dalgacık dönüşümü $I(x, y)$ görüntüsünün konvolüsyonuyla ifade edilir (Buciu & Gacsadi, 2009; Acar & Özerdem,2012).

$$J(x, y) = \iint I(x^t, y^t) g(x - x^t, y - y^t) dx^t dy^t \quad (31)$$

Denklem (32)'de gösterilen $g(x, y)$ fonksiyonu Gabor filtresini ifade eder:

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x_1^2 + \gamma^2 y_1^2}{2\sigma^2}\right) \exp\left(i\left(2\pi \frac{x_1}{\lambda} + \psi\right)\right) \quad (32)$$

$$x_1 = x \cos(\theta) + y \sin(\theta) \quad (33)$$

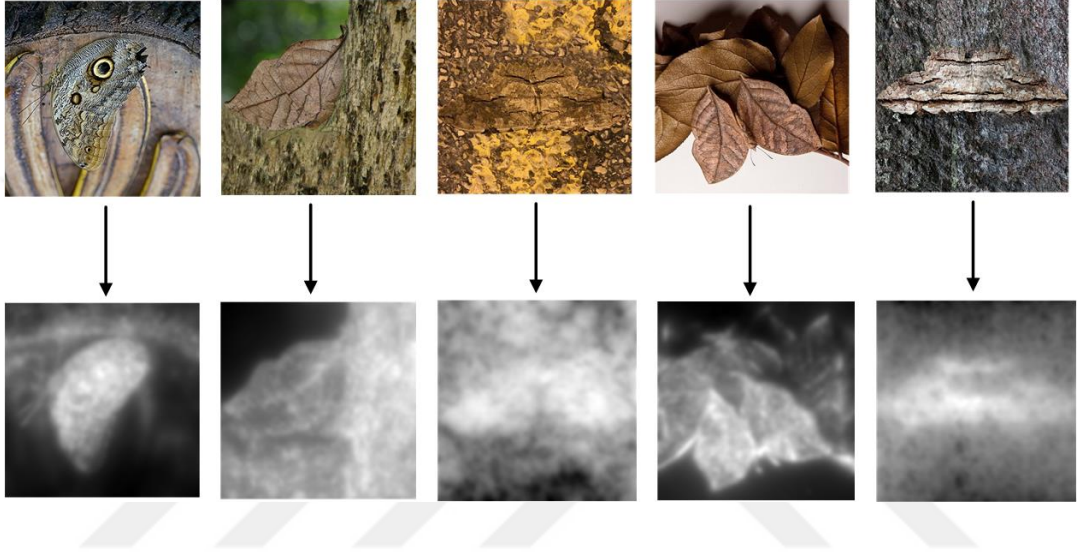
$$y_1 = -x \sin(\theta) + y \cos(\theta) \quad (34)$$

Burada:

- λ : kosinüsün dalga boyunu,
- θ : Gabor Fonksiyonu'nun yönünü,

- σ : Gauss zarfının standart sapmasını,
- ψ : fazın ofset değerini,
- γ : uzamsal görüş açısını belirtir (Acar & Özerdem,2012).

ERVA 1.0 veri setinde yer alan tüm test görüntülerine bölütleme işleminde kullanılmak üzere doku özelliklerinin tespit edilmesi için Gabor Filtre uygulanmıştır. Şekil 28’de çalışmamızda Gabor Filtre uygulanmış örnek görüntüler gösterilmiştir.



Şekil 28. Gabor Filtre uygulanmış örnek kamufraj görüntüler

K-means Yöntemiyle Görüntüyü Bölgelere Ayırma: Bu çalışmada bölütleme için kümeleme tabanlı yöntemlerden olan K-means yöntemi kullanılmıştır. K-means belirli bir veri kümesini önceden tanımlanan k tane küme sayısına sınıflandırır. Şekil 29’da gösterildiği üzere K-means yöntemi temelde dört aşamadan oluşmaktadır. Birinci aşamada başlangıçta kaç tane küme oluşturulacaksa her bir küme için küme merkezleri rastgele atanmaktadır. İkinci aşamada kamufraj görüntüdeki her bir pikselin küme merkezlerine olan uzaklığı hesaplanır. Bu hesaplama işlemi Denklem (35) eşitliğine göre yapılmaktadır. Denklem (35) eşitliğinde; k küme sayısını, n görüntüdeki piksel sayısını, c de küme merkezini temsil etmektedir. Her bir pikselin küme merkezlerine olan uzaklığını hesaplamak için Öklid uzaklık hesaplaması kullanılır. Öklid uzaklık hesaplaması için Denklem (36) eşitliği kullanılır.

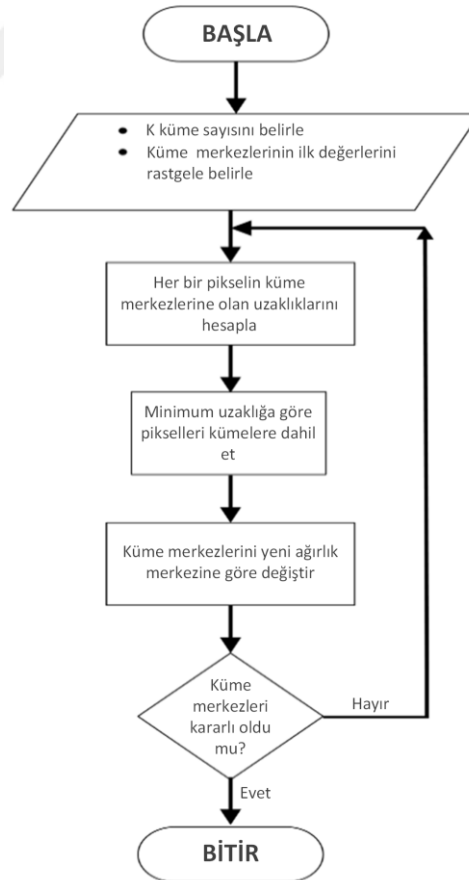
$$E = \sum_{j=1}^k \sum_{i=1}^n \|x_i^{(j)} - c_j\|^2 \quad (35)$$

$$D(x, y) = \|x - y\| = \sqrt{\sum_{i=1}^p (x_i - y_i)^2} \quad (36)$$

Tüm piksellerin tüm küme merkezlerine olan uzaklığı hesaplandıktan sonra, piksel hangi küme merkezine en yakınsa o kümeye dahil edilmektedir. Üçüncü aşamada küme merkezleri kendilerine dahil edilen piksellerin ağırlık merkezlerine kaydırılır. Tüm pikseller ait olduğu kümeye dahil edildikten sonra küme merkezlerinin yeni konumu Denklem (37)'e göre yeniden hesaplanır.

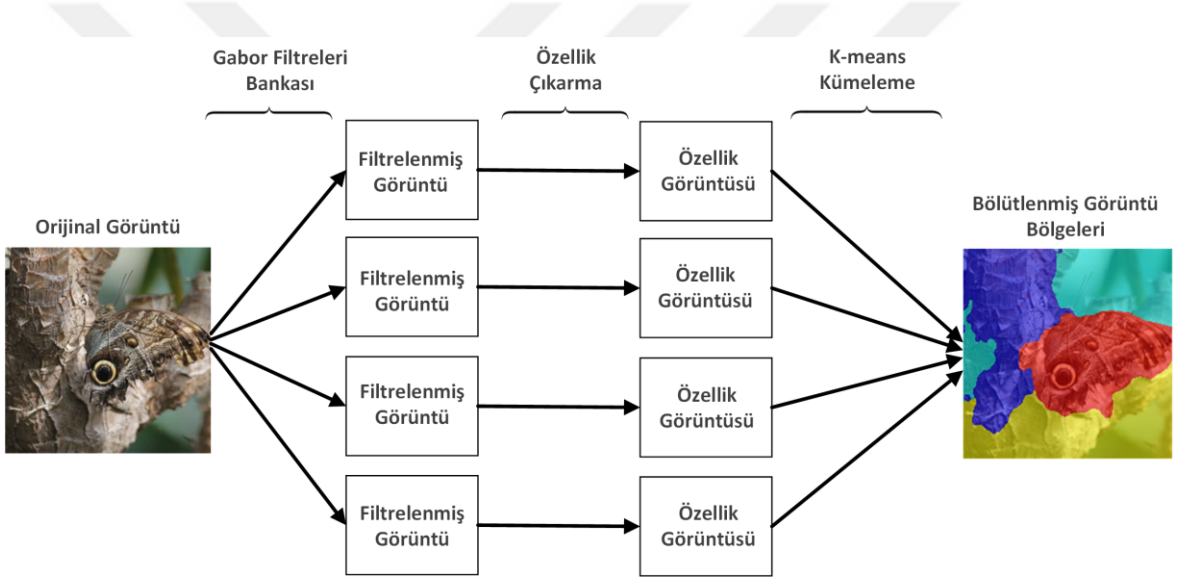
$$c_k = \frac{1}{k} \sum_{y \in C_k} \sum_{x \in C_k} p(x, y) \quad (37)$$

Son aşama olarak da küme merkezleri kararlı hale gelene kadar yani küme merkezlerinin yeri artık değişmeyene kadar ikinci ve üçüncü aşamalar tekrarlanır (Bayram & Nabiye, 2020).



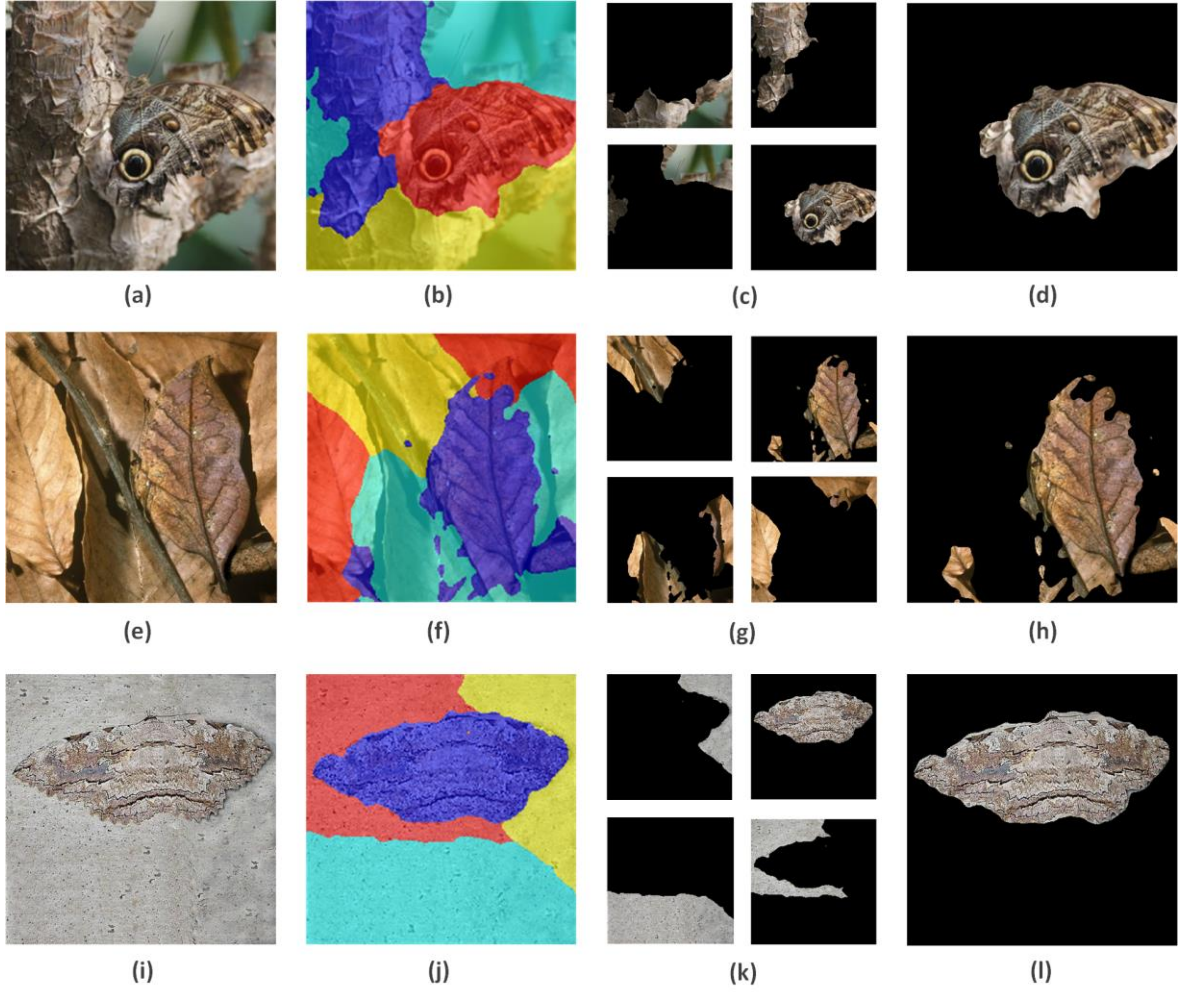
Şekil 29.K-means kümeleme algoritması genel akış diyagramı (Bayram & Nabiye, 2020).

ERVA 1.0 kamuflaj veri setinden sisteme girdi olarak verilen tüm kamuflaj test görüntülerinin her birine Gabor filtresi uygulanmaktadır. Bölütleme aşamalarının ilki Gabor filtre bankasının uygun şekilde tasarlanması ve görüntünün bir dizi filtrelenmiş görüntüye ayrıştırılmasıdır. Bu aşamadan sonra filtrelenmiş görüntülerden doku özellikleri çıkarılır. Özellik çıkarma adımının sonunda, filtre çıktılarında bir dizi özellik görüntüsü elde edilir. Aynı doku bölgesindeki pikseller aynı veya benzer doku özellikleri gösterir; bu nedenle, özellik alanında birbirine yakın bir şekilde kümelenmeleri beklenir. Çalışmanın bu aşamasında pikseller orijinal doku bölgelerini temsil eden bir dizi küme halinde kümelenmektedir. Her küme etiketlendiğinde, bölütlenmiş görüntü elde edilir. Şekil 30'da bu çalışmanın doku bölütleme süreci için genel şeması gösterilmiştir.



Şekil 30. Çalışmanın doku bölütleme aşamaları genel gösterimi.

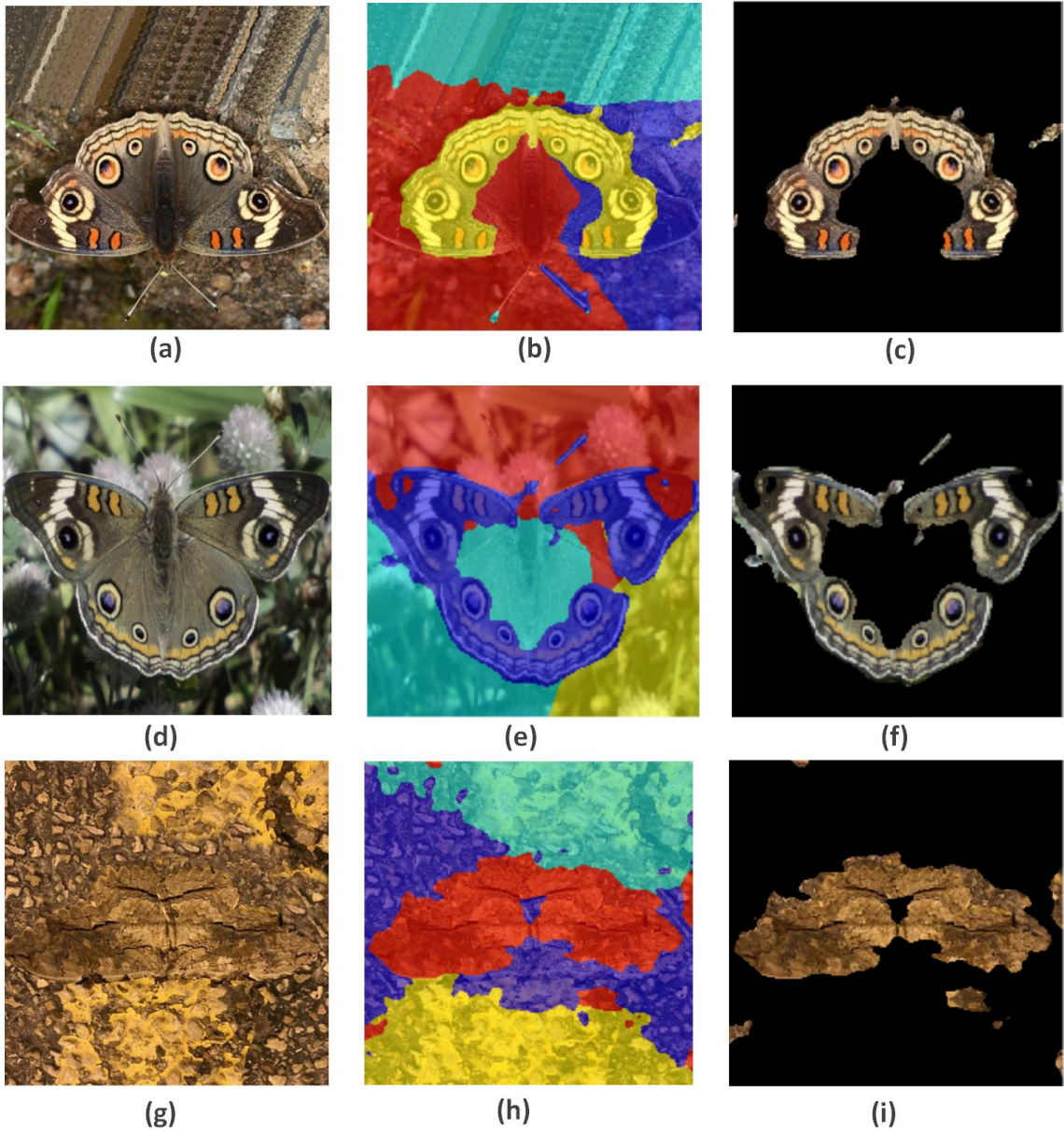
ERVA 1.0 veri setinde yer alan bazı kamuflaj test görüntüleri için orijinal görüntü ve bölütleme sonucu oluşan bölgeler için sonuçlar Şekil 31'de gösterilmiştir. Şekil 31'de (a), (e), (i) görüntüleri ERVA 1.0 kamuflaj veri setinden alınmış orijinal test görüntüleridir. Bu görüntülerin bölütleme sonucu elde edilen bölütlü bölge sınırları (b), (f), (j) görüntülerinde farklı renklendirilmiş olarak gösterilmiştir. Şekil 31'de (c), (g), (k) görüntülerinde bölütlenmiş kamuflaj görüntülerinin ayrı ayrı bölgelere ayrılmış şekli gösterilmiştir. Şekil 31'de (d), (h) ve (l) görüntüleri ise bölütleme sonucu tespit edilen kamufle olmuş kelebek nesne görüntüleridir.



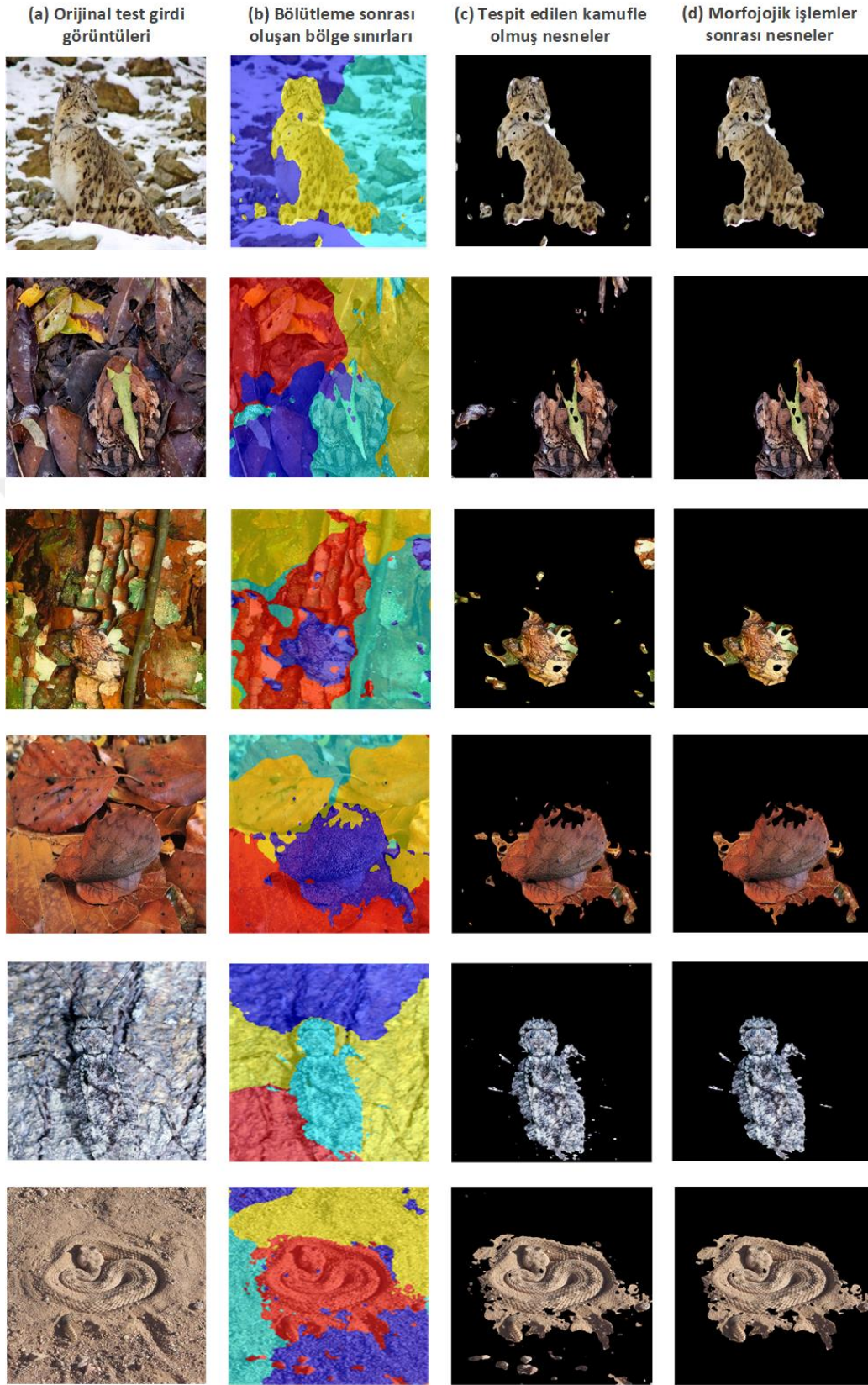
Şekil 31. (a), (e), (i) Orijinal görüntü; (b), (f), (j) bölütlü bölgelerin etiketlenmesi; (c), (g), (k) bölütlü bölgelerin ayrıştırılmış görüntüleri; (d), (h), (l) tespit edilen kelebek nesne görüntüleri

Düşük kontrast, dokusal benzerlik, karmaşık arka plan, gölgelendirme-aydınlatma koşulları, kamuflajlı nesnenin boyutu ve şekli kamuflaj görüntülerindeki bölütleme başarı oranlarını etkiler. Bu çalışmada düşük bölütleme başarı oranıyla elde edilen bölütleme sonuçlarının birkaç örneği Şekil 32'de gösterilmiştir. Şekil 32 (g)'de gösterildiği gibi, kamuflajlı nesneler arka planla benzer renk, doku veya parlaklık özelliklerine sahiptir ve bu da nesne ile arka plan arasındaki sınırların bulanıklaşmasına neden olur. Sınırlar bulanık olduğunda ve kamuflaj olmuş nesne ve arka planın renk ve doku özellikleri birbirine çok yakın olduğunda, bölütleme başarı oranı Şekil 32 (h) görüntüsündeki gibi azalmaktadır. Nesneleri arka plandan doğru bir şekilde algılamak ve ayırmak, özellikle düşük kontrastlı bölgelerde zorlu bir sorundur. Kamuflajın temel amacı nesneleri arka planla harmanlamaktır. Benzer dokular, bölütleme algoritmalarının nesne sınırlarını belirlemesini

zorlaştırır. Bu, bu çalışmada önerilen hibrit yöntemimizde kullanılan Gabor filtreleri ve LBP gibi iyi doku analiz yöntemleriyle bile yanlışlıklara neden olabilir. Kamuflajlı nesnelerin kendi içlerinde farklı desenlere, dokulara ve renklere sahip olması da bölütleme sürecinde hatalara neden olur. Şekil 32 (a), (d)'de gösterildiği gibi kamuflajlı kelebek nesnelerinin kendilerindeki doku, renk ve desen farklılıkları, (b) ve (e) görüntülerinde gösterildiği gibi, bölütlemeye hatalara neden olmuş ve başarı oranını düşürmüştür. Bu hatalar kamuflaj edilmiş nesnelerin sınırlarının doğru bir şekilde belirlenmesini etkilemekte ve çalışmamızda bölütleme başarı oranını düşürmektedir.



Şekil 32. (a), (d), (g) Orijinal görüntüler; (b), (e), (h) bölütlenmiş bölgelerin etiketlenmesi; (c), (f), (i) tespit edilen kelebek nesne görüntüleri



Şekil 33. (a), Orijinal test girdi görüntüleri; (b), bölütlenmiş bölgelerin etiketlenmesi; (c) tespit edilen kamufle olmuş nesne görüntüleri; (d) morfolojik işlemler sonrası tespit edilen kamufle olmuş nesne görüntüleri

Bu çalışmada ayrıca önerilen hibrit yöntemimiz literatürde oldukça zorlu bir veri seti olan CAMO kamuflaj veri seti üzerinde de uygulanmış ve bölütleme işlemi gerçekleştirilmiştir. CAMO veri setinde bulunan kamuflaj örnek orijinal test görüntüleri ve hibrit yöntemimizin kullanılması sonucu oluşan bölgeler için elde edilen sonuçlar Şekil 33'de gösterilmiştir. Şekil 33'de (a) sütununda yer alan görüntüleri CAMO kamuflaj veri setinden alınmış orijinal test görüntüleridir. Bu görüntülerin bölütleme sonucu elde edilen bölütlü bölge sınırları (b) sütununda yer alan görüntülerde farklı renklerle renklendirilmiş olarak gösterilmiştir. (c) sütunundaki görüntülerde bölütleme sonrası tespit edilmiş kamuflaj olmuş nesnelerin görüntüleri gösterilmiştir. (d) sütununda gösterilen görüntülerde ise bölütleme sonrası tespit edilen kamuflaj olmuş nesnelerin morfolojik işlemlere tabi tutulması sonrası elde edilen nesne görüntüleri gösterilmiştir.

LBP Algoritması ve Öklid Uzaklığı ile Özellik Veri Tabanından Kamuflaj Olmuş Nesne Bölgesinin Tespiti: Çalışmanın bu aşamasında bölütleme sonucu elde edilen bölütlü bölgelerden hangisinin kelebek nesnesine ait olduğu tespiti yapılmaktadır. Şekil 31'de (c), (g), (k) görüntülerinde görüleceği üzere orijinal kamuflaj görüntüleri K-means yöntemi ile doku özelliklerine göre farklı bölgelere ayrılmıştır. Bu bölgelerden hangisinin kelebek nesnesine ait olduğunun tespit edilmesi için denklem (29) eşitliğine göre hesaplanan LBP yöntemi ve denklem (36) eşitliğinde gösterilen Öklid uzaklık hesabı kullanılmaktadır.

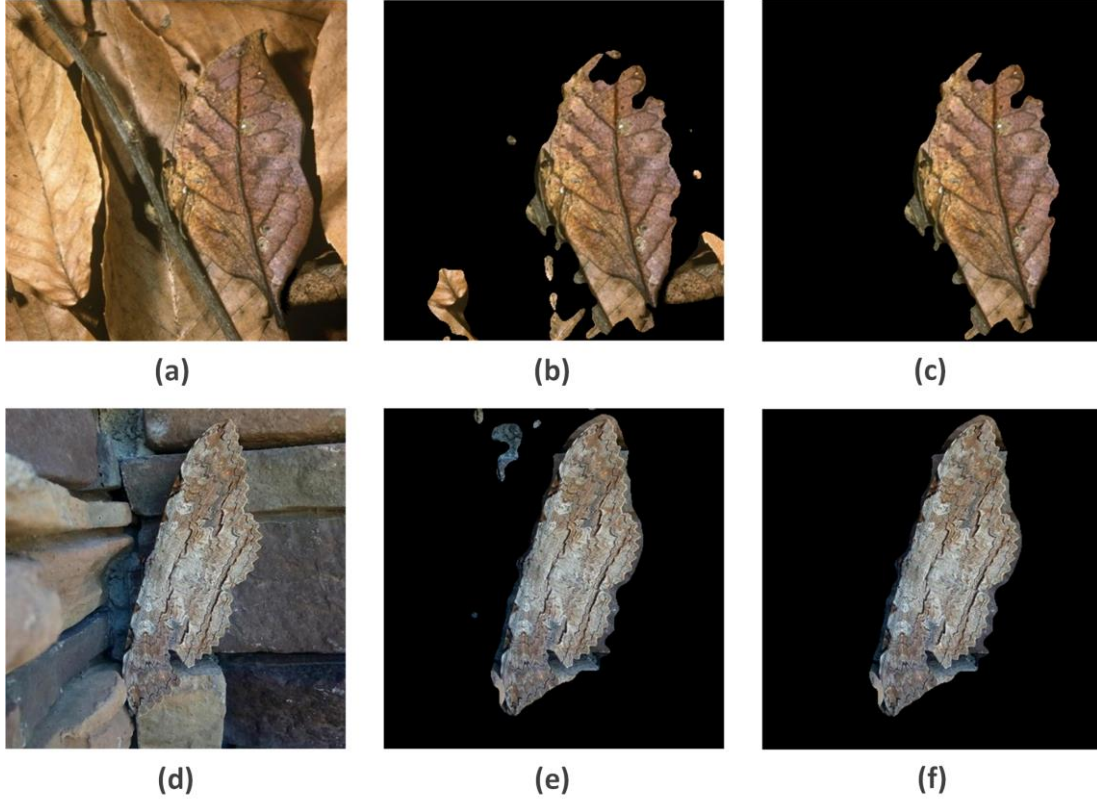
Şekil 31'de ERVA 1.0 kamuflaj test görüntülerinden alınmış (a), (e), (i) orijinal görüntülerinin Gabor filtre ile özellik çıkarımı yapılması ve sonrasında özellikli piksellerin K-means yöntemi ile kümeleme işlemi sonucunda elde edilen (c), (g), (k)'da yer alan bölütlü bölgeler ayrı ayrı gösterilmektedir. Bu bölgelerden hangisinin kelebek nesnesine ait olan bölge olduğunun tespit edilmesi gerekmektedir. Bunun için özellik veri tabanı oluşturulurken tüm eğitim görüntülerine LBP yöntemi ile özellik vektörleri hesaplandığı gibi, bu bölütlü bölgelerin de her birisi için LBP yöntemiyle özellik vektörleri hesaplanır. Her bir bölütlü bölge için hesaplanan özellik vektörleri sırasıyla özellik veri tabanında yer alan tüm özellik vektörleriyle Öklid uzaklık hesabı kullanılarak aralarındaki uzaklık bulunur. Örneğin Şekil 31'de gösterilen (a) orijinal görüntüsü için bölütleme işlemi sonucunda (c) görüntüsündeki dört farklı bölge elde edilmiştir. Bu 4 farklı bölgenin her birisi için LBP yöntemiyle özellik vektörleri hesaplanır. Hesaplanan bu dört farklı LBP özellik vektörleriyle özellik veri tabanındaki 4720 adet özellik vektörleri arasındaki uzaklık sırasıyla Öklid uzaklık hesabı ile hesaplanır. Bu hesaplama sonucunda en küçük fark hangi bölgeye ait LBP vektörü için elde edilmişse o bölge kelebek nesnesine ait bölge olarak kabul edilmektedir.

Şekil 31’de (c) görüntüsündeki bölgeler (a) orijinal görüntüsünün bölütlenmesi sonrası elde edilmiştir. (c) görüntüsündeki dört bölütlü bölge için hesaplanan Öklid uzaklık hesabının en küçük değeri (d) ’de gösterilen bölgeden elde edilmiştir. Bu sebeple (c) görüntüsünde gösterilen bölütlü bölgelerden (d)’deki bölge kelebek nesnesine ait bölge olmuştur. Her ne kadar çalışmada kamuflaj test kelebek görüntülerinin içinde bulunan kamuflaj olmuş kelebek nesnesinin bulunduğu bölge sınırlarının tespit edilme başarı oranı %100 olmasa da, bu yöntemle tüm kamuflaj test görüntülerinde yer alan kelebek nesnelerinin bulunduğu bölgelerin sınırları büyük doğruluk oranlarıyla tespit edilmiştir.

Son İşleme (Post-Processing): Bölütleme işleminin tamamlanıp kelebeğin ait olduğu bölge çıkarıldıktan sonra, bu bölge sınırları içinde yer alan fakat kelebek nesnesine ait olmayan gürültü ya da küçük piksel gruplarından oluşan parçaların bulunduğu alanlar bulunmaktadır. Bu aşamada morfolojik açma işlemi, kelebekli bölgeden gürültülerin ve kelebek nesnesi harici parçaların atılması işlemi yapılmıştır. Morfolojik açma işlemi, küçük nesneleri görüntünün ön planından kaldırıp, arka plana yerleştirmektedir. Bu aşamanın amacı, gürültüleri ve kelebek nesnesine ait olmayan parçaları ortadan kaldırmaktır.

$$I \circ H = (I \odot H) \oplus H \quad (38)$$

Açma işlemi Denklem (38)’de gösterildiği üzere, $I \circ H$ şeklinde ifade edilmektedir. Burada I görüntüyü ve H yapısal elemanı ifade eder ve bir erozyon (erosion) işlemini genişleme (dilatation) işlemi takip eder (Bozkurt et al., 2018).

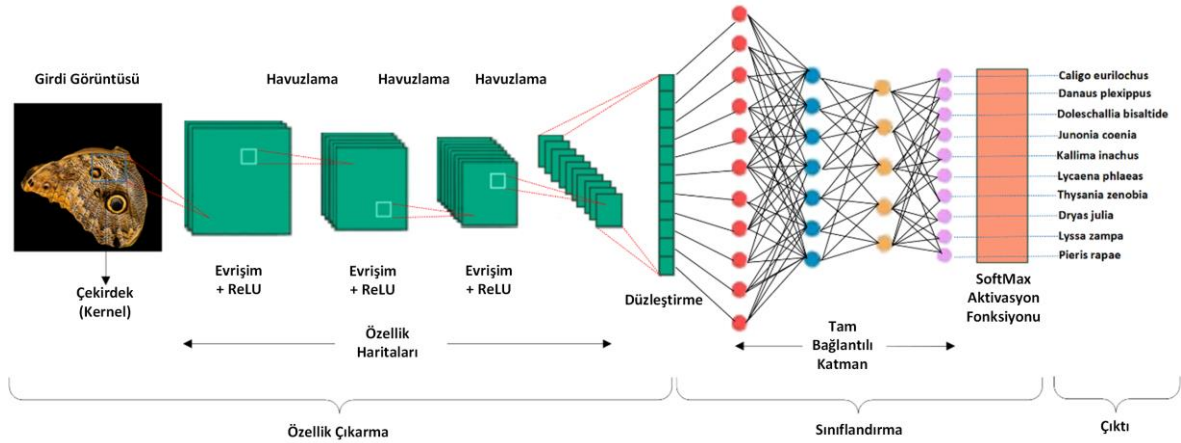


Şekil 34. (a), (d) orjinal görüntüler; (b), (e) bölütleme sonrası tespit edilen kelebek nesne görüntüleri; (c), (f) Morfolojik işlemler sonucunda elde edilen kelebek nesne görüntüleri

Bu çalışmada bölütleme işleminden sonra Şekil 34 (b)'den (c)'ye ve (e)'den (f)'ye geçişte görüldüğü gibi küçük boyutlu gürültülü yapıları ya da bölgecikleri çıkarmak için açma işlemi yapılmıştır. Elde edilen sonuçlara ait bazı örnekler Şekil 34 (c) ve (f) görüntülerinde gösterilmiştir.

Kamufle Olmuş Nesnenin Tanınması: Bu aşamada bölütleme sonrasında elde edilen kamufle olmuş kelebek nesnesinin türü belirlenmektedir. Bu işlem için derin öğrenme önceden eğitilmiş modeller kullanılmaktadır. Derin öğrenme, makine öğrenme tekniklerinin bir alt kümesi olup, günümüzde etkili bir şekilde çeşitli uygulamalarda kullanılmaktadır. Derin öğrenme çıktıyı hesaplama yöntemi sayesinde bilgisayarla görme, ses tanıma, doğal dil işleme gibi alanlarda yoğun ve başarılı bir şekilde kullanılmaktadır. Derin öğrenme geleneksel tekniklere göre, açık bir öznitelik çıkarma aşaması gerektirmemesi sebebiyle daha avantajlıdır. Derin öğrenmede ağ, ham girdiyi alıp istenen çıktıya eşlemekte ve özellikler manuel bir müdahale olmadan ağ tarafından otomatik bir şekilde öğrenilmektedir. Artan sayıda gizli katmana sahip ve geleneksel ağların bir ilerlemesi olan derin öğrenme modelleri ağların performansını büyük ölçüde artırmaktadır. Derin öğrenme teknikleri,

herhangi bir insan müdahalesi olmadan orijinal görüntülerden görünmez görüntü özelliklerini çıkarır (Singh vd., 2020). CNN, derin öğrenmede sezgisel ve güçlü bir ağ mimarisi olup desen tanıma ve görüntü sınıflandırma uygulamalarında sıklıkla kullanılmaktadır. CNN, özellik çıkarma ve öğrenmede oldukça fazla yararlı olduğu kanıtlanmış ve çok fazla sayıda çalışmada kullanılmıştır (Krizhevsky vd., 2012).



Şekil 35. Evrişimsel sınır ağının temel mimarisi (Bayram & Nabiye, 2025)

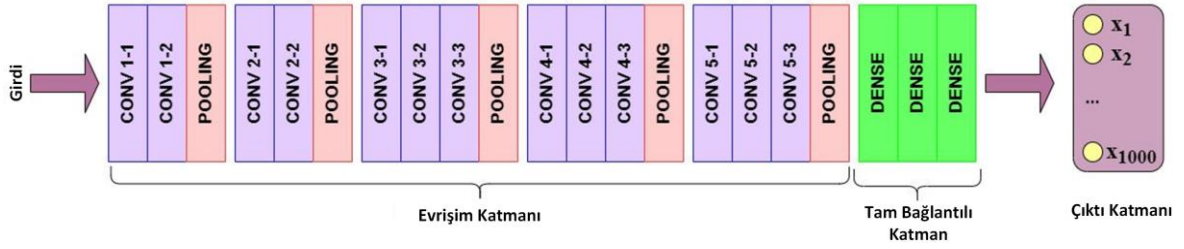
Şekil 35 CNN'nin temel mimarisi gösterilmiştir. Şekil 35'de görüldüğü üzere bu çalışmada bölütleme sonucunda tespit edilen kamufraj kelebek nesnesi görüntüleri CNN'e girdi olarak verilir. Daha sonra peş peşe evrişim ve havuzlama katmanlarıyla özellikler çıkarılmıştır. Son olarak kelebek nesnesinin türü sınıflandırma işlemi ile belirlenir. Bu çalışmada Şekil 31'de gösterildiği üzere kamufraj kelebek görüntülerinde yer alan kelebek nesnelerinin bölge sınırları, bölütleme işlemi sonucunda tespit edilmektedir. Tespit edilen kelebek nesnesinin ERVA 1.0 veri setindeki 10 türden hangisine ait olduğunun tahmin edilmesi için önceden eğitilmiş mimarilerden olan DenseNet121, InceptionResNetV2, DenseNet201, Xception, InceptionV3, VGG16, VGG19, AlexNet ve ResNet50 kullanılmıştır. Her bir yöntem eğitimde 100 iterasyon için çalıştırılmıştır.

Bu çalışmada model hiperparametreleri üzerinde deneysel bir kurulum gerçekleştirildi. Parametreler, modellerin deneysel sonuçlarına göre ayarlandı. Optimum parametreler nicel deneylerle ayarlandı ve ardından kelebek türü tahmini, önceden eğitilmiş derin öğrenme modelleri üzerinde bu parametrelere göre gerçekleştirildi. Model değerlendirmeleri, zaman içinde hesaplanan parametrelerin çalışan ortalaması kullanılarak gerçekleştirildi. Yaklaşımın doğruluğunu ayarlamaya yardımcı olan birçok hiperparametre vardır.

Bu bölümde, model hiperparametreleriyle deneysel kurulum gerçekleştirdik. Hiperparametreleri modelin deneysel sonuçlarının doğruluğuna göre ayarladık. Adam optimizatörü uyarlanabilir bir iyileştirici olarak kullandık. Adam optimizatörün derin ağlar için zaman açısından verimli ve önemli iyileştiricilerden biri olduğunu söyleyebiliriz. Ağırlıklı Kategorik Çapraz Entropi (Weighted Categorical Cross-Entropy) bir kayıp fonksiyonu olarak kullanıldı. Ayrıca, öğrenme oranı 0,00001 olarak yapılandırıldı ve modellerin eğitimi için 128'lik batch boyutu ile 100 epoch boyunca eğitim gerçekleştirilmiştir. Aşırı uyumu önlemek ve ağı genelleştirmek için Dropout (0,5) kullanılmıştır.

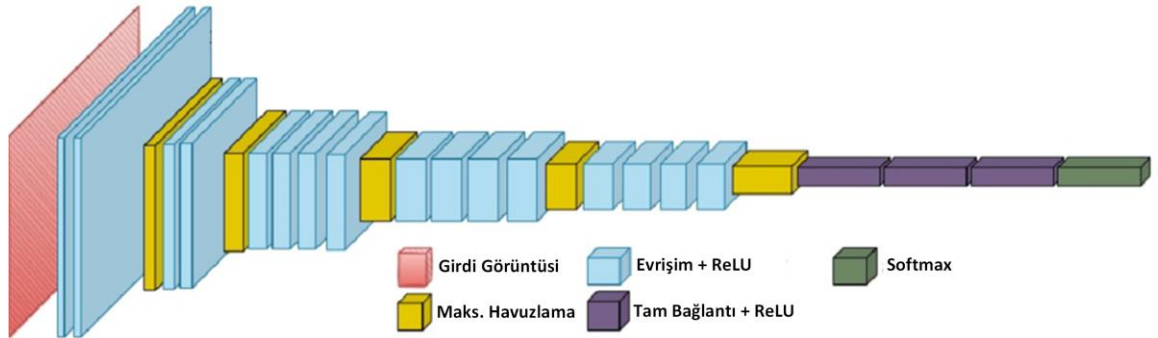
Derin öğrenme modelleri, büyük miktarda veriden öğrenerek karmaşık görevleri gerçekleştiren yapay sinir ağlarıdır. Özellikle, önceden eğitilmiş modeller, büyük veri kümeleri üzerinde önceden eğitilmiş ve transfer öğrenme ile farklı uygulamalara adapte edilebilen ağlardır. Bu bağlamda bu çalışmada bölütleme sonrası elde edilen kamufle olmuş kelebek nesnelerinin türlerinin belirlenmesi için önceden eğitilmiş derin öğrenme modelleri kullanılmıştır. Çalışmada kullanılan önceden eğitilmiş derin öğrenme modelleri aşağıdaki başlıklar halinde açıklanmıştır.

VGG16 Modeli: VGG16, 2014 yılında Zisserman ve Simonyan tarafından geliştirilen bir CNN modelidir (Simonyan & Zisserman, 2014). Bu model, özellikle görüntü sınıflandırma ve nesne tanıma gibi bilgisayarlı görü problemlerinde yaygın olarak kullanılmaktadır. VGG16, ImageNet veri kümesi üzerinde eğitilmiş olup, ILSVRC-2014 yarışmasında büyük başarı elde etmiştir. Şekil 36'de gösterildiği gibi, VGG16 modeli 13 evrimsel, 5 adet havuzlama ve 3 tam bağlantılı katmandan oluşmaktadır. Bu model her bir katmanında evrim, maksimum havuzlama ve tam bağlı katmanların olduğu 16 gizli katmandan ve 3x3'lük filtrelerden oluşmaktadır. Bu ağ oldukça büyük bir ağ olup, ReLU aktivasyon fonksiyonu kullanılmıştır ve yaklaşık 138 milyon parametreye sahiptir. VGG16 3x3 boyutlu filtreler kullanarak özellik çıkarımı yapmaktadır. Maksimum havuzlama kullanarak özellik haritalarını küçültür. Son katmanda tam bağlantılı katmanlar bulunmaktadır. Sınıflandırmada Softmax aktivasyon fonksiyonu kullanılır. 138 milyon üzerinde parametre içerdiği için yüksek bellek kullanımına ihtiyaç bu modelin dezavantajıdır.



Şekil 36. VGG16 model mimarisi (McDermott, 2021)

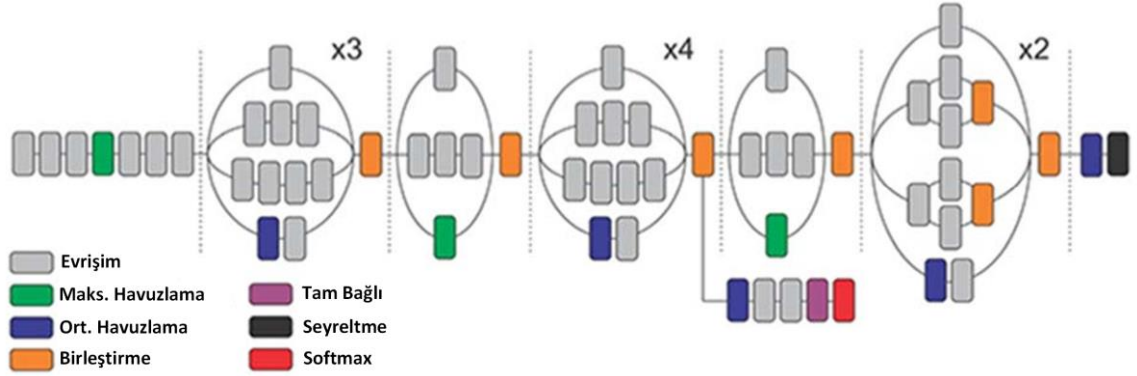
VGG19 Modeli: VGG19 (Simonyan & Zisserman, 2014), 2014 yılında geliştirilmiştir. Şekil 37’de gösterildiği gibi 16 evrişimsel, 5 adet havuzlama ve 3 tam bağlantılı katmandan oluşmaktadır. VGG19’da VGG16’ya benzer şekilde 5 havuzlama ve son katman olarak SoftMax’tan (sınıflandırıcı) oluşmaktadır. Bu model 144 milyon parametre içermektedir (Gopalakrishnan vd., 2017). Bu ağında parametre sayısını azaltmak için, tüm evrişimli katmanlarda 3x3’lük küçük filtreler kullanılmıştır. 2014’te Imagenet yarışmasında VGG-19, bir milyondan fazla görüntü üzerinde eğitilmiş olup, geniş bir görüntü yelpazesi için zengin öznitelik temsillerini öğrenmiştir (Zheng vd., 2018). VGG19, günümüz bilgisayarlı görü uygulamalarında yaygın olarak kullanılan güçlü bir modeldir. Transfer öğrenme sayesinde farklı görevler için kolayca uyarlanabilir. Bu modelin dezavantajı 144 milyon parametre içerdiğinden yüksek donanım ihtiyacı duymasındır.



Şekil 37. VGG19 model mimarisi (Nguyen et al., 2022)

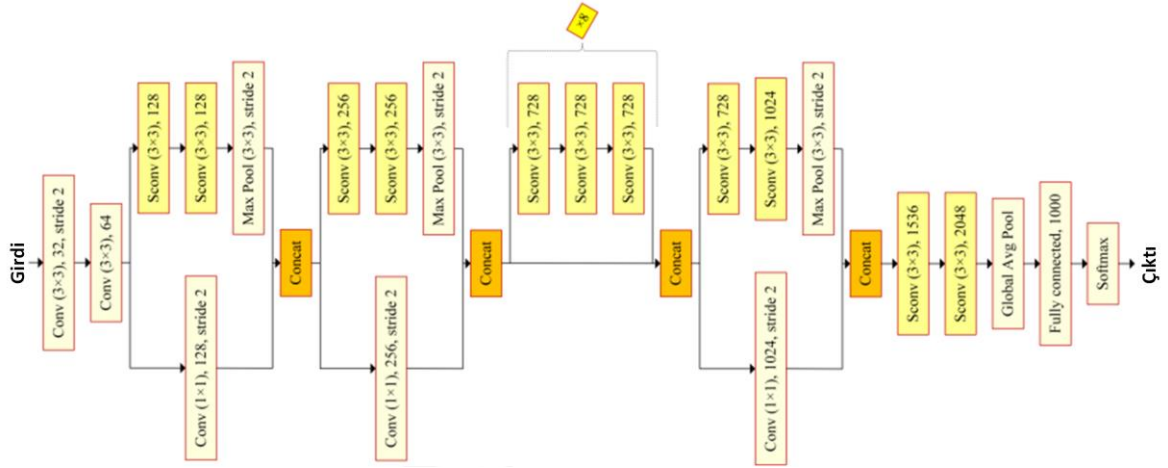
Inception-V3 Modeli: Toplamda 42 katman derinliğine sahip bir sinir ağı olan Inception-V3 modeli, Inception-V1 modeli güncellenerek oluşturulmuştur. Inception-V1 ve Inception-V2 modellerine göre daha geniş bir ağı sahip olan Inception-V3 modeli daha çok görüntü tanıma işlemlerinde kullanılmaktadır (Szegedy vd., 2015). Bu model, inception mimarisine dayalı GoogLeNet modelinin üçüncü sürümüdür. GoogLeNet’te katman sayısı

22 idi, Inception v3'te 42'ye çıkarılmıştır. Katman sayısındaki bu artışa rağmen hesaplama maliyeti yalnızca 2,5 kat daha yüksektir. Bunu sağlamak için daha fazla sayıda ve daha küçük boyutlarda evrişim işlemi yapılmaktadır (Kaya vd., 2024). Şekil 38'de Inception-V3 modelinin mimari yapısı gösterilmiştir.



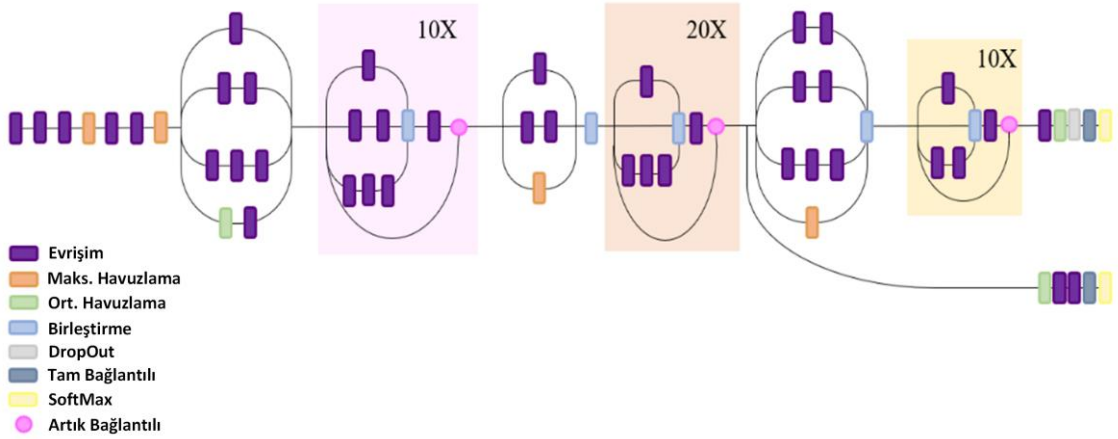
Şekil 38. Inception-V3 model mimarisi (Kaya vd., 2024).

Xception Modeli: Google tarafından Inception'ın extreme sürümü olarak sunulan Xception, 71 katman derinliğinde evrişimli bir sinir ağıdır. Xception, derinlikte ayrılabilir evrişim ve ResNet'teki gibi evrişim blokları arasındaki kısa yollar şeklinde iki ana noktaya dayanan bir mimaridir. Xception modeli ile derinlemesine ayrılabilir evrişim uygulanmıştır. Xception modeli, temelde derin ayırık evrişimler fikrine dayanır. Bu yaklaşım, klasik evrişim katmanlarına kıyasla parametre sayısını azaltarak hesaplama maliyetlerini düşürürken, aynı zamanda daha derin ve geniş özellik haritaları oluşturulmasını sağlar. Xception modeli, özellikle mobil cihazlar ve gömülü sistemler için optimize edilmiş modellerde (örneğin, MobileNet) kullanılan derin ayırık evrişimler tekniğinin öncüsüdür. Xception mimarisi; evrişim blokları, iyileştirilmiş-derinlemesine ayrılabilir evrişim blokları (Sconv), filtre birleştirme (filter-concat veya concat), ortalama ve maksimum havuzlama yaklaşımları, tam bağlantılı katman ve Softmax fonksiyonunu içermektedir (Chollet, 2017; Leonardo vd., 2018). Xception modelinin genel mimari yapısı Şekil 39'da gösterilmiştir.



Şekil 39. Xception model mimarisi (Leonardo vd., 2018)

InceptionResNetV2 Modeli: Inception-ResNet-v2, Google tarafından geliştirilen ve 2016 yılında tanıtılan bir derin öğrenme modelidir. Bu model, Inception ve ResNet mimarilerinin avantajlarını birleştirerek, görüntü sınıflandırma ve diğer bilgisayarlı görü görevlerinde yüksek doğruluk sağlamayı hedefler. Inception mimarisi enine, ResNet mimarisi ise derine gitmekle ilgilidir. İki mimarinin bir araya getirilmesiyle ortaya çıkarılan Inception-ResNet-V2 mimarisi ile hem derine hem enine gitmek için optimum sonuç elde edilebilmektedir. Inception-ResNet-v2, Inception v3 ve Inception v4 mimarilerinin modüler yapısını, ResNet'in artık bağlantıları (skip connections) ile birleştirerek daha hızlı ve etkili bir model sunar. Geleneksel Inception modüllerine artık bağlantılar ekleyerek daha iyi gradyan akışı sağlanır, bu da modelin daha derin olmasını ve verimli bir şekilde eğitilmesini mümkün kılar. Inception-ResNet-V2, Inception mimari ailesini temel alan ve artık bağlantıları (residual connection) içeren evrişimli sinir ağı algoritmasıdır. Atık bağlantılar, model eğitilirken kısa yollara izin verir ve daha iyi performansa ulaşılmasını sağlayan daha derin sinir ağlarını başarılı bir şekilde eğitilmesini sağlar. Bu aynı zamanda durum başlangıç bloklarının önemli ölçüde basitleştirilmesini de sağlamaktadır. Inception-ResNet-v2, Inception ve ResNet mimarilerinin güçlü yönlerini birleştirerek derin öğrenme modelleri içinde önemli bir yere sahiptir. Daha hızlı eğitim, yüksek doğruluk ve derin ağlarda stabilite sağlayarak birçok görüntü işleme görevinde etkili bir şekilde kullanılmaktadır (He vd., 2016; Szegedy vd., 2017). Inception-ResNet-v2 modelinin genel mimari yapısı Şekil 40'ta gösterilmiştir.



Şekil 40. Inception-ResNet-v2 model mimarisi (Özgür & Keser, 2021)

DenseNet121 Modeli: DenseNet Gao Huang ve ekibi tarafından önerilen bir derin öğrenme modelidir. Bu model, geleneksel derin CNN'e kıyasla daha iyi parametre verimliliği ve gradyan akışı sağlayarak yüksek doğruluk elde etmeyi amaçlar.

DenseNet, kaybolan gradyan problemini hafifletmesiyle, özniteliğin tekrar kullanımını artırmış ve parametre kullanımını azaltmıştır. Bunun sonucunda derin öğrenme modellerinin eğitimi yönetilebilir hale getirmiştir. DenseNet-121, bu ailenin popüler varyantlarından biridir ve 121 katmana sahiptir. DenseNet'in en belirgin özelliği, yoğun bağlantılar (dense connections) kullanmasıdır. Geleneksel CNN'lerde, her katman sadece kendisinden önceki katmandan veri alırken, DenseNet'te her katman, kendisinden önce gelen tüm katmanlardan doğrudan veri alır. Bu yapı, gradyan akışını güçlendirerek derin ağların eğitilmesini kolaylaştırır.

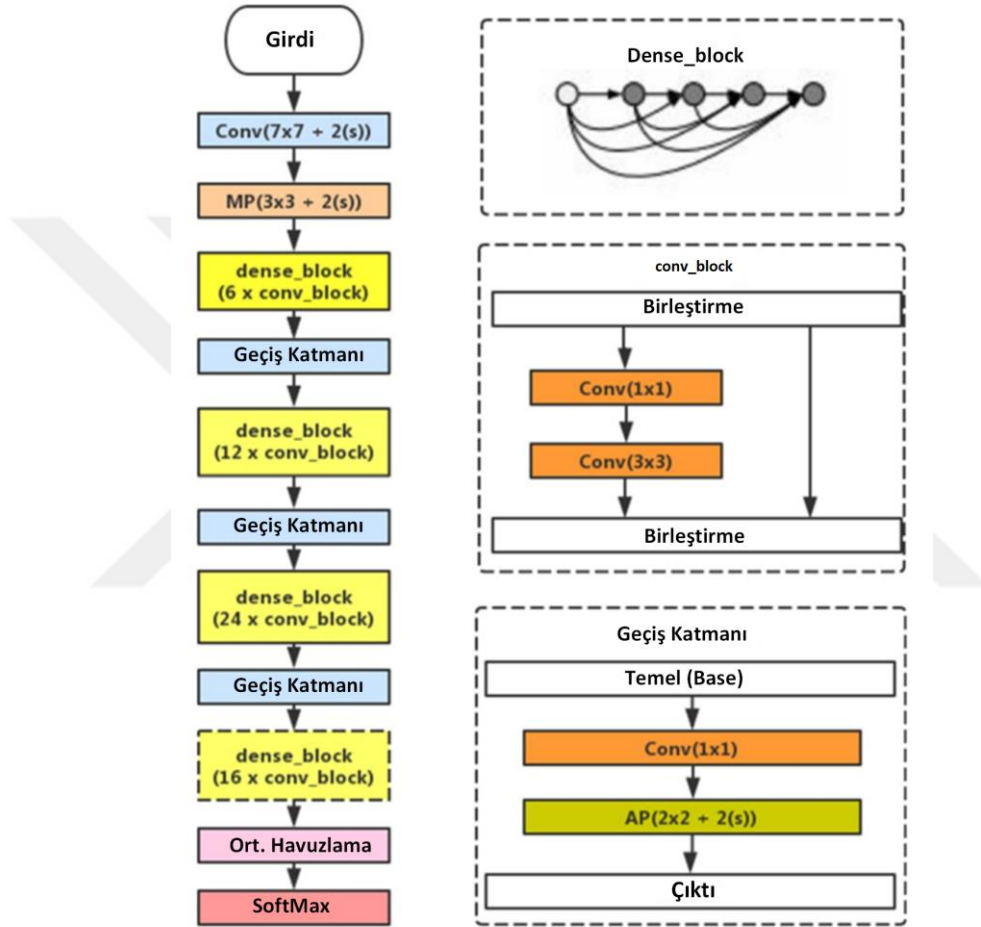
Evrişim katmanlarından oluşan dense blokları (DenseBlock) DenseNet121'in temel yapı taşlarıdır. DenseNet121, nihai çıktı katmanı olarak 1000 birimlik tam bağlı bir katmana sahip 121 adet bağlanmış evrişimli katmandan oluşmaktadır. DenseNet'te önceki tüm katmanların öznitelik haritaları birleştirilmekte ve herhangi bir belirli katman için girdi olarak kullanılmaktadır. Bunun yanı sıra, sonraki bütün katmanlar için girdi olarak kendi öznitelik haritaları kullanılmaktadır.

DenseNet-121 modelinin avantajları şu şekilde sıralanabilir:

- Yoğun bağlantılar sayesinde daha az hesaplama gücü ile yüksek doğruluk elde edilir.
- Gradyan akışını güçlendirmesiyle daha iyi öğrenme sağlar ve eğitim sürecini hızlandırır.

- Modelin farklı katmanlarında öğrenilen özellikler korunur ve her katman, tüm önceki katmanlardan gelen bilgileri kullanabilir.
- Geleneksel CNN modellerine kıyasla daha derin yapıların daha stabil ve verimli bir şekilde eğitilmesini sağlar (Huang vd., 2017; Sarker vd., 2020).

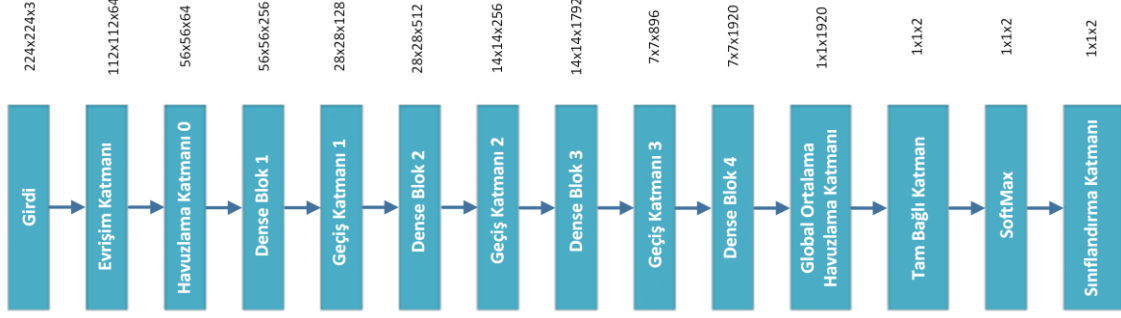
Şekil 41’de DenseNet121 modelinin genel mimari yapısı gösterilmiştir.



Şekil 41. DenseNet121 model mimarisi (Ji vd., 2019).

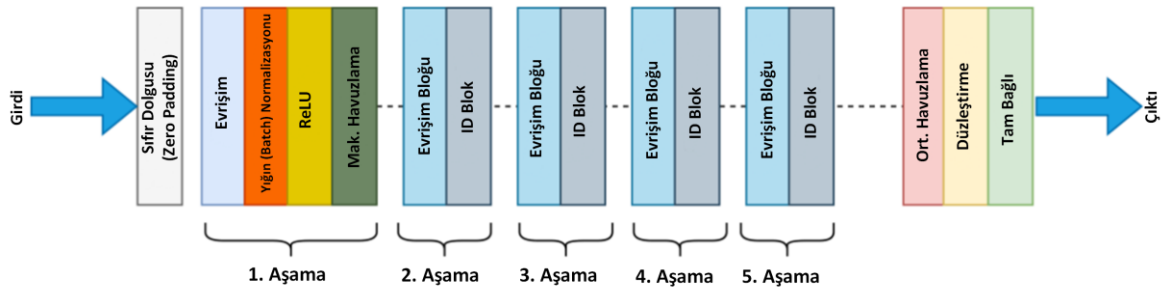
DenseNet201 Modeli: DenseNet-201, DenseNet mimarisinin daha derin bir versiyonudur. Geleneksel CNN’den farklı olarak yoğun bağlantılar kullanarak, daha az parametre ile daha verimli bir öğrenme süreci sağlar. DenseNet201, 7x7 evrişim filtresi, 3x3 MaxPool ile başlayan bir ilk katman, aralarında 3 ortalama havuz katmanı olan 196 evrişim katmanı ve 1 çıkış katmanından oluşan toplamda 201 katmana sahiptir. DenseNet201 mimarisine 224×224×3 boyutunda bir girdi verilmektedir. Bu modelde, her bir yoğun bağlantılı blok, önceki bloktaki tüm katmanların çıkışlarıyla birleştirilerek oluşturulur.

DenseNet-201, derin öğrenme modellerinde yoğun bağlantılar sayesinde verimli ve etkili bir alternatif sunmaktadır. Daha az parametre ile daha yüksek doğruluk sağlayarak, özellikle büyük ölçekli görüntü işleme ve tıbbi uygulamalarda yaygın olarak kullanılmaktadır. (Bingöl, 2022; Yu vd., 2019). Şekil 42’de DenseNet-201 modelinin genel mimarisi yapısı gösterilmiştir.



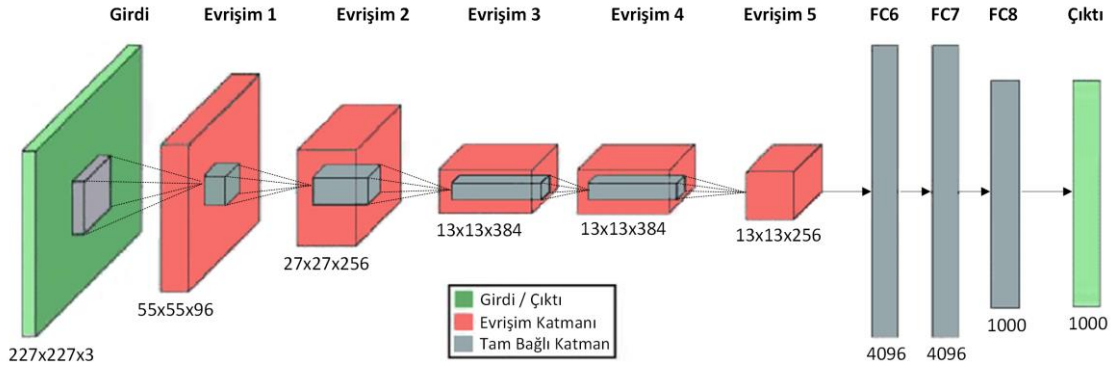
Şekil 42. DenseNet201 model mimarisi (Yu vd., 2019).

ResNet50 Modeli: ResNet50 (Residual Network 50), 2015 yılında Microsoft Research tarafından önerilen ResNet mimarisinin 50 katmanlı bir versiyonudur. Derin CNN modelinde yaşanan kaybolan gradyan problemi sorununu çözmek için geliştirilmiştir. ResNet’in en büyük yeniliği, artık bağlantılar kullanarak derin ağların daha verimli bir şekilde eğitilmesini sağlamasıdır. ResNet-50, toplamda 50 katmandan oluşur ve ağırlıklı olarak artık öğrenme yöntemi üzerine inşa edilmiştir. Geleneksel derin sinir ağlarında katman sayısı arttıkça, modelin öğrenme süreci yavaşlar ve gradyan kaybolması gibi problemler ortaya çıkar. ResNet-50, artık bağlantılar kullanarak bu sorunları minimize eder ve daha derin ağların eğitilmesine olanak tanır. Geleneksel CNN’lerde her katman, önceki katmandan gelen girdiyi doğrudan işlerken, ResNet’te bir önceki katmanın çıktısı, bir sonraki katmana doğrudan eklenir. ResNet-50, conv1, dört adet artık blok (residual block) ve tam bağlı katman içerir. Blok sayısı arttıkça modelin öğrenme kapasitesi artar. ResNet-50’de 1x1 konvolüsyonlar, parametre sayısını azaltarak modeli daha verimli hale getirir. Bu modelde Global Ortalama Havuzlama tam bağlı katmanlardan önce kullanılarak aşırı öğrenme riskini azaltır (He et al., 2016). Şekil 43’de ResNet50 modelinin genel mimari yapısı gösterilmiştir.



Şekil 43. ResNet-50 model mimarisi (Rezayi vd., 2021).

AlexNet Modeli: AlexNet, derin öğrenmenin yaygınlaşmasını sağlayan en önemli modellerden biri olarak kabul edilir. 2012 yılında Alex Krizhevsky ve arkadaşları tarafından geliştirilmiştir. AlexNet, CNN mimarisini kullanarak görüntü sınıflandırmada devrim yapmış ve günümüzdeki derin öğrenme modellerinin temelini atmıştır. Özellikle GPU kullanımıyla büyük ölçekli veri kümelerinde verimli eğitim sağlayan ilk derin model olması nedeniyle, derin öğrenmenin gelişiminde kritik bir rol oynamıştır. AlexNet, beşi evrişim katmanı, üçü tam bağlı katman olmak üzere toplam 8 katmandan oluşur. AlexNet'teki ilk evrişim katmanı, 11x11 boyutundaki filtreleri kullanarak görüntüleri işler. Daha sonraki katmanlarda 5x5 ve 3x3 boyutlarında filtreler kullanılarak özellikler çıkarılır. Bu derin yapı, modelin karmaşık görüntüleri daha iyi öğrenmesini sağlar. AlexNet, ReLU fonksiyonunu kullanarak eğitim süresini hızlandırmıştır. Bu modelde, uzamsal boyutu küçültmek ve fazla parametre kullanımını önlemek için maksimum havuzlama yöntemi kullanılmıştır. Bu sayede gereksiz bilgi elenir ve daha verimli öğrenme sağlanır. AlexNet, aşırı öğrenmeyi önlemek için dropout mekanizmasını ilk kullanan derin öğrenme modellerinden biridir. Dropout, bazı nöronları rastgele devre dışı bırakarak modelin aşırı ezberleme yapmasını engeller. Son üç katman, tam bağlı katmanlardan oluşur ve en sonunda SoftMax fonksiyonu kullanılarak sınıflandırma yapılır. Model, 1000 sınıflı ImageNet veri setinde eğitim alarak görüntüleri başarılı bir şekilde sınıflandırabilir (Krizhevsky vd., 2012). Şekil 44'de AlexNet modelinin genel mimari yapısı gösterilmiştir.



Şekil 44. AlexNet model mimarisi (Putzu vd., 2020).

2.2. Kamufraj Görüntülerinin Sınıflandırılması İçin Transfer Öğrenme ve Özellik Birleştirme Yaklaşımı

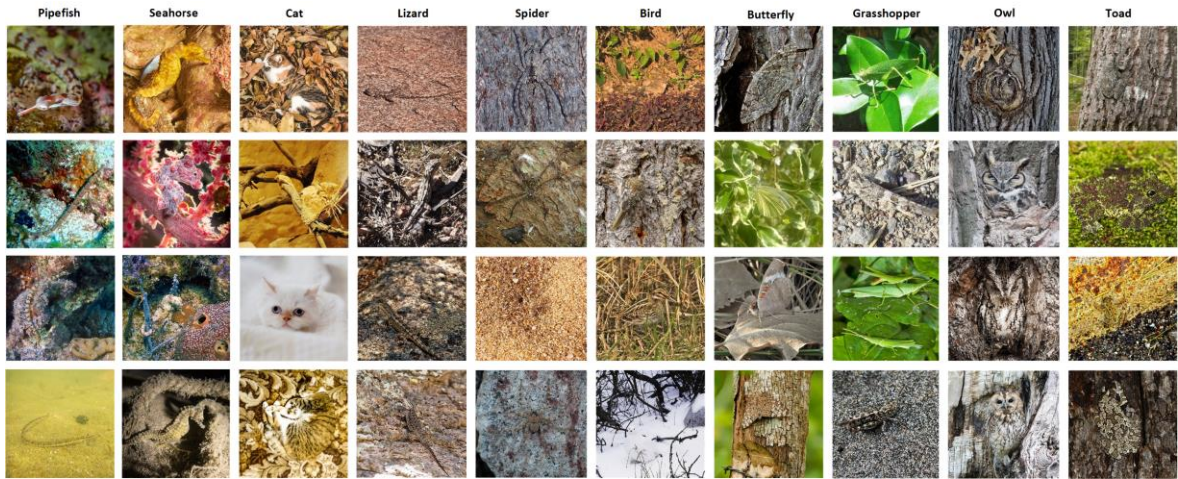
Bu çalışmada, kamufraj görüntülerinin sınıflandırılması üzerinde bir çalışma yapılmış ve derin öğrenme önceden eğitilmiş modellerden VGG19, VGG16, Xception, EfficientNetV2, InceptionV3, MobileNet, DenseNet169, DenseNet121, DenseNet201 modelleri kullanılmıştır. Çalışmada, daha kapsamlı özellik çıkarma ve genelleme yeteneklerine sahip yeni bir birleştirme tabanlı önceden eğitilmiş yaklaşım oluşturmak için farklı kombinasyonlarla önceden eğitilmiş transfer öğrenme mimarileri birleştirilmiştir. Bunun için, çalışmada kullanılan en iyi sınıflandırma performansını gerçekleştiren önceden eğitilmiş mimariler farklı kombinasyonlarda birleştirilmiştir. Bu kombinasyonlarla, birleştirme tabanlı DenseNet201 ve MobileNet modeli ile %98,41 doğruluk elde edilmiştir.

Çalışmanın literatüre katkıları aşağıdaki şekilde özetlenebilir:

- COD10K veri setinden alınan sınıflara ait eğitim görüntülerine veri artırma teknikleri uygulanarak toplam eğitim görüntü sayısı altı kat artırıldı.
- Elde edilen deneysel sonuçlar, literatürde başarılı sonuçlar gösteren çalışmalarla karşılaştırılarak sunulmuştur.
- DenseNet201, MobileNet, InceptionV3, EfficientNetV2, DenseNet169 gibi başarılı mimarilerle farklı kombinasyonların birleştirilmesiyle elde edilen sınıflandırma başarı oranları karşılaştırılmıştır.
- Kamufraj görüntülerinin sınıflandırılmasının performansını artırmak için birleştirme tabanlı önceden eğitilmiş bir yaklaşım tanıtılmıştır.

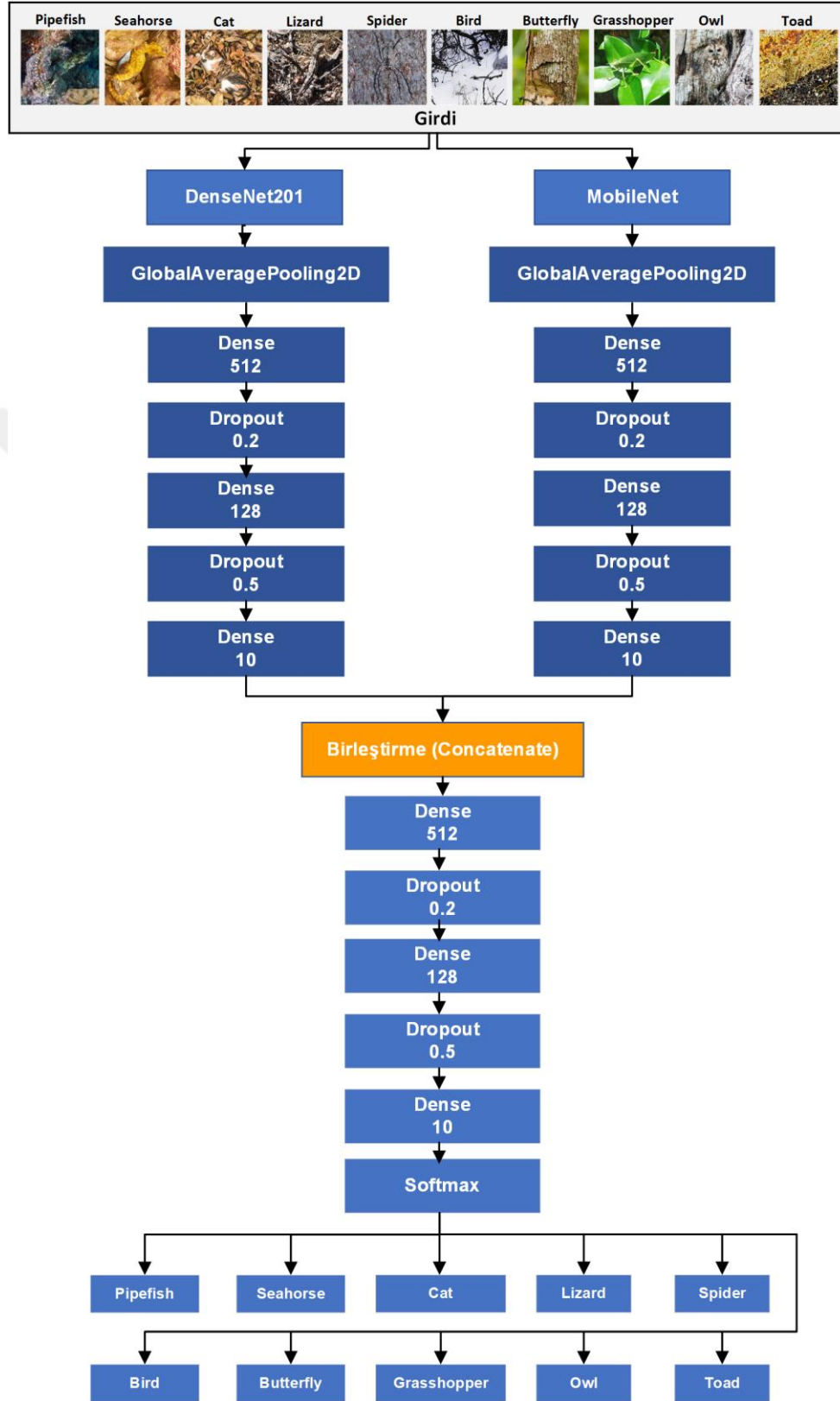
2.2.1. Çalışmada Kullanılan Veri Seti

Literatürde kamuflaj görüntülerinden oluşan birçok veri seti bulunmaktadır. En sık kullanılanlar CAMO, CAMO-COCO, COD10K, NC4K, CAM-FR'dir. Bu çalışmada en zorlu ve yüksek kaliteli COD10K kullanılmıştır. Çalışmada COD10K veri setinden seçilen 10 sınıfa ait veri setindeki eğitim görüntülerine veri artırma uygulanmıştır. Veri artırma uygulanarak görüntü çeşitliliği artırılmış ve modelin genel anlayışı iyileştirilmiştir. Veri artırma tekniği, görüntü sınıflandırma görevlerinde DL modellerinin genelleme yeteneğini artırır. Bu çalışmada, veri artırma tekniği olarak rastgele döndürme uygulanmıştır. Rastgele döndürmenin kullanılmasının nedeni, görüntülerin yönelimini değiştirebilme yeteneğidir. Bu yöntemle, kamuflaj görüntülerine rastgele açısız dönüşümler uygulanır ve veri setindeki her görüntü belirli bir aralıktan örneklenmiş bir açıyla rastgele döndürülür. Bu rastgele döndürme işlemi görüntü çeşitliliği sağlar, genelleme performansını iyileştirir ve aşırı öğrenmeyi önler. Uygulanan veri artırma yöntemiyle veri setinde yer alan eğitim görüntülerinin sayısı altı kat artırılmaktadır. Görüntü sayısı artırılan COD10K veri seti, sınıflandırma aşamasında yaklaşık olarak %80 eğitim, %10 doğrulama ve %10 test olmak üzere alt kümelerle ayrılmıştır. COD10K veri setine dahil edilen ve çalışmada kullanılan türlerin örnek kamuflaj görüntüleri Şekil 45'de gösterilmiştir.



Şekil 45. COD10K kamuflaj veri setinden seçilip çalışmada kullanılan türlere ait görüntüler

2.2.2. Önerilen Yöntem



Şekil 46. Çalışmada önerilen yeni modelin sistem tasarımının genel akış şeması

Çalışmada literatürde yaygın olarak kullanılan COD10K kamuflaj veri seti kullanılmıştır. İlk olarak, veri kümesini yeterli ve dengeli bir şekilde genişletmek için veri artırma yöntemi veri setindeki eğitim verilerine uygulanmıştır. Model seçme ve veri artırma teknikleri, görüntünün DL tabanlı sınıflandırılması gibi bilgisayarlı görme görevlerinde performansı artırmak için kritik öneme sahiptir. Çalışmada aşırı uyumu önlemek ve genelleme performansını artırmak için veri artırma tekniği olarak rastgele döndürme uygulanmıştır. Önerilen modelde, görüntüler önce sisteme girdi olarak verilir. Daha sonra özellikler evrişim ve havuzlama katmanlarıyla çıkarılır. Son olarak, sınıflandırma süreci kamuflaj görüntüsünün hangi sınıfa ait olduğunu tahmin eder. Bu süreç için önceden eğitilmiş mimariler DenseNet201, DenseNet121, DenseNet169, InceptionV3, EfficientNetV2, Xception, VGG16, VGG19 ve MobileNet kullanılmıştır. Elde edilen sonuçlar, literatürde daha önce farklı mimarilerle elde edilen sonuçlarla karşılaştırılmıştır. Sınıflandırma performansını artırmak için bireysel kullanımda en iyi sonuçları veren DenseNet201, MobileNet, InceptionV3, EfficientNetV2, DenseNet169 mimarileri çeşitli kombinasyonlarda birleştirilmiştir. Bu kombinasyonlardan en başarılı sonucu veren DenseNet201 + MobileNet mimarilerinin birleştirilmesiyle elde edilmiştir. Bu çalışmada önerilen DenseNet201 + MobileNet mimarilerinin birleştirilmesiyle elde edilen yeni model için sistem tasarımının genel akış şeması Şekil 46'da gösterilmiştir.

Transfer Öğrenme ve Önceden Eğitilmiş Derin Öğrenme Yöntemleri: Transfer öğrenmesi (TL), eğitim aşamasında öğrenilen bilgileri benzer veya farklı sorunları çözmek için kullanmayı amaçlar. Son zamanlarda çok fazla ilgi görmüştür. Temel olarak TL, büyük bir veri kümesinden daha önce öğrenilen özellikleri kullanarak ağırlıkları ağırlara aktarma süreci olarak tanımlanır (Yosinski vd., 2014). TL ile farklı bir sorunu çözerken, öğrenme süreci sıfırdan başlatılmaz, öğrenilen kalıplardan başlatılır. Bu şekilde, önceki öğrenme kullanılır. Önceden eğitilmiş bir CNN mimarisini kullanarak, araştırmacılar son katmandan çıkarılan özellikleri tam bağlı katmanlardan önce çeşitli sınıflandırıcılara uygulayabilirler. Bu mimarilerin önceden eğitilmesi, gelişmiş performans sonuçlarına yol açabilir (Kolar vd., 2018). TL, bilgisayarlı görüde çok iyi bilinen bir yöntemdir. Bunun nedeni, zamandan tasarruf sağlayan bir şekilde doğru modeller oluşturmamızı sağlamasıdır (Rawat & Wang, 2017). Bilgisayarlı görü için TL genellikle önceden eğitilmiş modellerin kullanımıyla tanımlanır. Önceden eğitilmiş bir model, büyük bir kıyaslama veri kümesinde eğitilmiş bir modeli ifade eder. Bu, bu hazır modelle çözmek istediğimize benzer bir sorunu çözme fırsatı verir. Bu çalışmada ImageNet veri kümesinde önceden eğitilmiş olan DenseNet201,

DenseNet121, DenseNet169, InceptionV3, Xception, VGG16, VGG19, EfficientNetV2 ve MobileNet TL mimarileri kullanılmıştır.

Önerilen Birleştirme Tabanlı Önceden Eğitilmiş Yaklaşım: Her ne kadar önceden eğitilmiş derin öğrenme modelleri görüntü sınıflandırma alanında önemli başarılar elde etmiş olsa da bazı durumlarda yetersiz kalabilmektedir. Özellikle kamuflaj görüntülerde arka planla aynı doku, desen, renk özelliklerine sahip kamuflaj olmuş nesnelerin sınıflandırılmasında görüntüler üzerinde ayırt edici yeni özelliklere ihtiyaç duyulmaktadır. Bunun için çeşitli özellik çıkarma yöntemlerini birleştirmek daha geniş bir özellik uzayı oluşturmaktadır. Önerdiğimiz bu yaklaşım modelin daha fazla çeşitlilik ve ayrıntı yakalamasını sağlar. Böylece modelin sınıflandırma performansı artırılır. Birbirinden farklı önceden eğitilmiş modellerin birleştirilmesi modelin birbirinden farklı daha büyük ve daha çeşitli veri kümeleri üzerinde eğitilmesine olanak tanır ve bunun sonucu olarak genelleme kabiliyetini artırır. Bu şekilde model daha iyi performans gösterir ve yeni veri örneklerine daha iyi adapte olur. Bunun yanı sıra model, çeşitli kullanıcıların ve görevlerin ihtiyaçlarına bağlı olarak özelleştirilebilir. Bu özelleştirme işlemi ile modelin farklı senaryolara ya da uygulamalara uyum sağlama esnekliği artırılır. Birleştirme tabanlı önceden eğitilmiş yaklaşımların kullanılmasıyla, görüntü sınıflandırma görevlerinde daha iyi performans elde edilebilir. Bunun yanı sıra genelleme yeteneği, transfer öğrenme ve esneklikte de elde edilebilmektedir. Zorlu bir problem olan kamuflaj görüntülerin sınıflandırılmasında önerdiğimiz bu yaklaşım gelecekte yapılacak çalışmalar için önemli bir strateji olarak düşünülebilir.

Bu çalışmada birden fazla önceden eğitilmiş modeli birleştirerek tek bir modelin zayıf yönlerinin telafi edilebileceği öne sürülmüştür. Bunu test etmek için bu çalışmada önceden eğitilmiş modellerle elde ettiğimiz en başarılı sınıflandırma başarı oranlarını veren mimariler birleştirilmiştir. DenseNet201, DenseNet169, MobileNet, InceptionV3 ve EfficientNetV2 mimarileri farklı kombinasyonlarla birleştirilerek performans ölçümleri yapılmıştır. Birleştirilen bu modellerden en başarılı sınıflandırma başarı performansı DenseNet201 ve MobileNet yöntemlerinin birleştirilmesiyle elde edilmiştir. Şekil 46'da DenseNet201 ve MobileNet mimarilerinin birleştirilmesi gösterilmiştir. En başarılı sınıflandırma performansı elde ettiğimiz bu kombinasyon, literatürdeki önceki çalışmalarla karşılaştırıldığında önerdiğimiz yaklaşım ile önemli bir başarı elde edildiği görülmüştür. Birleştirilmiş modelin daha sağlam ve güvenilir olduğu görülmüştür. Kamuflaj görüntülerin sınıflandırılması problemi için birleştirme tabanlı önceden eğitilmiş yaklaşımın genelleme yeteneği ve

etkinlik açısından umut verici gelişmeler sunduğunu düşünölmekle birlikte gelecekteki çalışmalar için daha da geliştirilebilir bir strateji olacağı düşünölmektedir.

2.3. Klasik Makine Öğrenmesi ve Evrişimsel Sinir Ağı Yöntemlerini Kullanarak Kamufraj Görüntülerin Sınıflandırılması İçin Karşılaştırılmalı Bir Çalışma

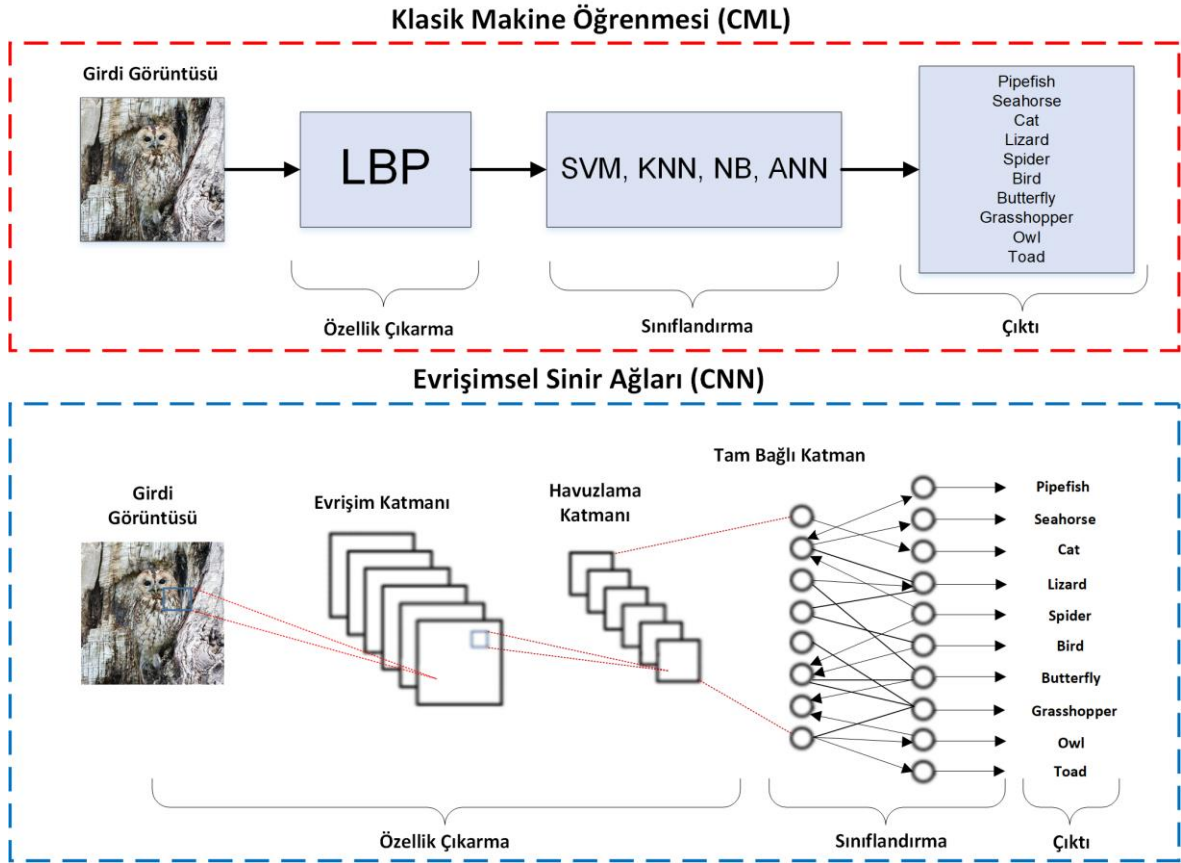
Bu çalışmada COD10K veri setinden kategorize edilerek seçilen 10 sınıftan kamufraj görüntüleri kullanılmıştır. Bu sınıflara ait eğitim görüntülerine veri artırma yöntemi uygulanmış ve görüntü sayısı artırılmıştır. Sınıflandırma başarı oranlarını ölçmek için iki farklı model kurulmuştur. Bunlardan birisi Klasik Makine Öğrenmesi (CML) modeli, diğeri ise Evrişimli Sinir Ağı (CNN) modelidir. Bu modellerle sırasıyla ham ve artırılmış görüntüler kullanılarak Deney1 ve Deney2 olmak üzere iki deney gerçekleştirilmiş ve sınıflandırma başarı oranları ölçölmüştür. CML modeli, giriş kamufraj görüntüsünden Yerel İkili Desen (LBP) algoritması ile dokusal özellik çıkarımı ve sınıflandırma aşamalarından oluşmaktadır. CNN modelinde ise giriş kamufraj görüntüleri için evrişim ve havuzlama katmanları ile özellikler çıkarılmakta ve sınıflandırma işlemi ile görüntünün sınıfı tahmin edilmektedir. Çalışmanın deneysel sonuçlarına göre en yüksek sınıflandırma başarı oranı 1. deneyde DenseNet201 modeli ile %76,73, 2. deneyde DenseNet201 modeli ile %96,63 olarak bulunmuştur. Çalışma sonucunda CML yöntemlerinin kamufraj görüntülerin sınıflandırılmasında başarısız olduğı gözlemlenmiştir. CNN yöntemlerinin kamufraj görüntülerin sınıflandırılmasında başarılı olduğı ve görüntü sayısının veri artırım yöntemi ile artırılmasının bu başarıyı çok olumlu yönde etkilediğı görölmüştür. Bu çalışmanın katkıları şu şekilde özetlenebilir:

- COD10K veri setinden seçilen 10 sınıftan alınan eğitim görüntülere veri artırma yöntemi uygulanmış ve görüntü sayısı altı katına çıkarılmıştır.
- Ham veri ve artırılmış veri kullanılarak iki farklı deney yapılmış ve veri artırmanın kamufraj görüntülerinde sınıflandırma başarısını artırdığı belirlenmiştir.
- Sınıflandırma işlemi CML ve derin öğrenme önceden eğitilmiş modeller kullanılarak gerçekleştirilmiş ve derin öğrenme modellerinin kamufraj görüntülerinde CML yöntemlerinden çok daha başarılı sonuçlar verdiği görölmüştür.
- Derin öğrenme önceden eğitilmiş modeller olan InceptionV3 ve EfficientNetV2 kullanılarak elde edilen öznitelikler birleştirilmiş ve elde edilen yeni öznitelige göre yeni bir sınıflandırma işlemi gerçekleştirilmiştir. Elde edilen sonuçlara göre iki modelin

özniteliklerinin birleştirilmesinin sınıflandırma başarı oranını önemli ölçüde artırdığı görülmüştür.

2.3.1. Önerilen Yöntem

Görüntü sınıflandırma, ön işleme, özellik çıkarma, nesne bölütlemesi, nesne tespiti ve nesne sınıflandırmasından oluşmaktadır. Kamuflaj görüntülerindeki kamuflajlı nesnelerin sınıflandırılması ve tespiti, kamuflaj olmayan görüntülere göre çok daha zorlu bir problemdir. Bunun nedeni, kamuflaj görüntülerindeki kamuflajlı nesnelerin arka planla çok benzer doku, renk ve desen özelliklerine sahip olmasıdır. Bu nedenle arka plandan ayrılmaları, tespiti veya sınıflandırılması oldukça zorlu bir problemdir. Bu çalışmada literatürde yaygın olarak kullanılan COD10K kamuflaj veri seti kullanılmıştır. Bu veri setindeki kamuflaj görüntülerinden 10 sınıf seçilerek, sınıflandırma işlemi klasik makine öğrenmesi (CML) ve derin öğrenme ile önceden eğitilmiş modeller ile yapılmış ve sisteme giren kamuflaj görüntülerinin ait olduğu sınıflar belirlenmiştir. Çalışma kapsamında gerçekleştirilen deneylerde COD10K veri setindeki kamuflaj görüntülerinin sınıflandırılmasında CML yöntemleri ile başarılı sonuçlar elde edilememiştir. Bu nedenle derin öğrenme tekniklerinden yararlanılmıştır. Derin öğrenme teknikleri, görüntü sınıflandırması (Chen vd., 2021; Lu & Weng, 2007; Wang vd., 2017) gibi birçok bilgisayarlı görü görevinde büyük ilerlemeler sağladığı için tercih edilmiştir. Şekil 47'de gösterildiği gibi, önerdiğimiz sistem CNN ve CML' den oluşan iki modelden oluşmaktadır.



Şekil 47. Çalışmada önerilen modelin genel mimari gösterimi

Klasik Makine Öğrenmesi Modeli: Şekil 47'de görüldüğü gibi, CML yöntemleriyle oluşturulan CML modeli, özellik çıkarma ve sınıflandırma aşamalarından oluşmaktadır. Sisteme girilen kamuflaj görüntüsünün LBP algoritması ile dokusal özellik vektörleri çıkarılır. Çıkarılan dokusal özellik vektörleri daha sonra Naive Bayes (NB), K-En Yakın Komşu (KNN), Yapay Sinir Ağları (ANN) ve Destek Vektör Makineleri (SVM) gibi CML yöntemleri ile sınıflandırılır. Sınıflandırma işleminin ardından sisteme girilen kamuflaj görüntüsünün COD10K veri setinden seçilen 10 sınıftan hangisine ait olduğu belirlenir.

Özellik Çıkarma: Sisteme girdi olarak verilen kamuflaj görüntüleri için aynı sınıflara ait görüntülerin ayırt edici dokusal özelliklerini belirlemek amacıyla özellik çıkarımı yapılır. Aynı görüntü sınıfındaki nesnelerin dokusal özellikleri benzer ve yakından ilişkilidir. Bu çalışmada CML modelinde özellik çıkarımı LBP tabanlı yöntem kullanılarak yapılmıştır.

Sınıflandırma: Sisteme girdi olarak verilen görüntülerin dokusal özellik vektörleri çıkarıldıktan sonra sınıflandırma işlemi gerçekleştirilmektedir. Sınıflandırma işlemi için CML yöntemleri olan SVM, KNN, NB ve ANN yöntemlerinden faydalanılmıştır. Bu yöntemler aşağıdaki şekilde açıklanmıştır.

Destek Vektör Makineleri (SVM): SVM statiksel öğrenme teoremi içerisinde yer alan yapısal risk azaltma prensibine dayanan genelleme yeteneği çok iyi olan bir sınıflandırıcıdır (Widodo & Yang, 2007). SVM’deki temel fikir pozitif ve negatif olarak adlandırılan iki sınıfın verileri arasındaki en uygun ayırıcı hiper düzlemi bulmaktır (Qi vd., 2013). SVM, iki sınıf arasındaki uzaklığın maksimum olduğu durumun bulunmasıdır. Bu verileri iki sınıfa ayıracak çok fazla sayıda ayırıcı düzlem bulunmaktadır. Optimum ayırıcı düzlem, her iki sınıfın en uç verileri arasındaki mesafenin (iki sınıfın destek vektörleri arasındaki mesafe) maksimum olduğu durumu sağlayan ayırıcı düzlemdir (Subasi & Gursoy, 2010).

K-En Yakın Komşu (KNN) algoritması: KNN algoritması denetimsiz bir algoritmadır. KNN’de veri içerisindeki noktaların kendisine en yakın noktalar ile karşılaştırılarak veri kümeleri oluşturulması hedeflenmektedir. Komşu sayısı parametresiyle, karşılaştırmanın veriye en yakın kaç komşu ile yapılacağı belirlenmektedir. KNN ile sınıflandırma işlemi nesnelerin birbirleri arasındaki yakınlık ilişkilerine göre yapılmaktadır. Geliştirme kolaylığına sahip bir algoritmadır. Dezavantajları yüksek miktarda bellek alanına gereksinim duyması, veri seti ve boyutu arttıkça işlem yükünün ve maliyetin büyük ölçüde yükselmesi, performansının k komşu sayısı gibi parametrelere bağlı olması olarak sıralanabilir. Sınıflandırma esnasında test örnekleri eğitim örnekleri ile birbirleri ile karşılaştırılır. Komşuluk mesafesi için öklid mesafesi kullanılmaktadır. Tahminler, komşu örneklerin oy çoğunluğuna dayanmaktadır (Jiang & Zhou, 2004; Liu & Zhang, 2012).

Naive Bayes (NB): NB, basit ve en etkili denetimli sınıflandırma algoritmalarından biridir. Olasılıksal bir sınıflandırıcı olan NB, bir nesnenin olasılığına dayanarak tahmin yapar. Özellik sıralama yöntemleri ve temel bileşen analizi ile birleştirildiğinde, yüksek bir tahmin doğruluğu ile iyi performans gösterir (Kumar & Vijaya, 2022).

Yapay Sinir Ağları (ANN): ANN, sınıflandırma işlemi için yaygın bir şekilde kullanılmaktadır. YSA beynin bilgi işleme sürecinden esinlenmiştir. YSA’da temel olarak üç ana katman yer almaktadır. Bu katmanlar girdi katmanı, gizli katman ve çıkış katmanıdır. Gizli katman sayısı bir veya daha fazla sayıda da olabilmektedir. Girdi ve çıkış katmanı arasında yer alan gizli katman nöron adı verilen yapılardan oluşmaktadır. Katmanda yer alan her nöron kendisinden sonra gelen katmandaki tüm nöronlar ile bağlantısı yer almakla birlikte bulunduğu katmandaki nöronlar ile bağlantısı bulunmamaktadır. Girdi katmanında çıkış katmanında sınıflandırılacak durum ile ilgili parametreleri içermektedir. Gizli katman bilgi işlemeyi gerçekleştirir ve çıkış katmanı sınıf etiketi üretir (Yegnanarayana, 2009).

Evrişimsel Sinir Ağı Modeli (CNN): CNN, görüntüleri analiz ederek veri üzerindeki örüntüyü tespit eden yapay sinir ağı modelleridir. CNN, özellik çıkarma ve öğrenmede oldukça fazla yararlı olduğu kanıtlanmış ve çok fazla sayıda çalışmada kullanılmıştır. CNN’de sınıflandırma için gereken özellikler veriler üzerinde matris filtreler uygulanarak çıkarılmaktadır. CNN mimarisi 3 katmandan oluşmaktadır. Bunlar; Evrişim katmanı, havuzlama katmanı ve tam bağlantılı katmandır. Evrişim katmanı görüntünün ilk işleme alındığı ve özellik çıkarma işleminin yapıldığı katmandır. Bu katmanın çıktısı uygulanan filtreler sonucu elde edilen özellik haritalarıdır. Havuzlama katmanı, özellik haritalarında uygulanan stokastik, maksimum, ortalama ve boyutsal piramit ortaklama gibi yöntemler yardımıyla minimum düzeyde bilgi kaybı ile boyutlar düşürülmektedir. Kısacası modelin sınıflandırma için ihtiyaç duyduğu en önemli bilgiler süzülerek hem aşırı öğrenmeye karşı direnç hem de karmaşıklıkta iyileştirmeler elde edilmektedir. Tam bağlantılı katman ise görüntü üzerindeki özellikler ile sınıflandırma etiketleri arasındaki ilişkinin kurulduğu katmandır. Bu katmanlar arasındaki nöronlar birbirlerine tamamen bağlıdır (Krizhevsky vd., 2012; Ankile vd., 2020; Aloysius & Geetha, 2017).

Şekil 47’de gösterildiği üzere kurulan CNN modelinde ilk olarak kamuflej görüntüler sisteme girdi olarak verilir. Daha sonra evrişim ve havuzlama katmanları ile öznetelik çıkarılır. Son olarak sınıflandırma işlemi ile kamuflej görüntüsünün hangi sınıfa ait olduğu tahmin edilir. Bu işlem için önceden eğitilmiş mimarilerden olan DenseNet201, DenseNet121, DenseNet169, InceptionV3, EfficientNetV2, Xception, VGG16 ve VGG19 yöntemleri kullanılmıştır. Her bir yöntem eğitimde 100 iterasyon için çalıştırılmıştır. Ayrıca bu çalışmada InceptionV3 ve EfficientNetV2 modellerinden elde edilen özellikler birleştirilerek tekrar sınıflandırma işlemi yapılmış ve sınıflandırma başarı oranlarına etkisi araştırılmıştır.

2.4. LBP, HOG ve Klasik Makine Öğrenmesi Yöntemleri Kullanarak Kamuflej Görüntülerin Sınıflandırılması

Bu çalışmada COD10K kamuflej veri setinden alınan görüntüler kullanılmıştır. Kamuflej görüntülerinin sınıflandırılması işlemi için klasik makine öğrenmesi (CML) yöntemlerinden Yapay Sinir Ağları (ANN), K-En Yakın Komşuluk Algoritması (KNN) ve Destek Vektör Makineleri (SVM) yöntemleri kullanılarak bir model oluşturulmuştur. Bu model özellik çıkarma ve sınıflandırma aşamalarından oluşmaktadır. Özellik çıkarma işlemi

için LBP ve HOG algoritmaları kullanılmıştır. Bu model ile LBP algoritması ve CML yöntemlerinin birlikte kullanılmasıyla Deney1, HOG algoritması ve CML yöntemlerinin birlikte kullanılmasıyla da Deney2 gerçekleştirilmiştir. Bu deneyler sonucunda kamuflaj görüntülerin sınıflandırma başarı oranları ölçülmüştür. Elde edilen deneysel sonuçlara göre Deney1'den en yüksek sınıflandırma başarı oranı SVM yöntemiyle %83,17, Deney2'den ise en yüksek sınıflandırma başarı oranı yine SVM yöntemiyle %69,15 olarak ölçülmüştür. Deneysel sonuçlara göre kamuflaj görüntülerde LBP algoritması ile CML yöntemlerinin birlikte kullanılması, HOG algoritmasının CML yöntemleriyle birlikte kullanılmasına göre çok daha başarılı sonuçlar vermiştir. Elde edilen sonuçlara göre kamuflaj görüntülerde sınıflandırma işleminde kullanılmak üzere özellik çıkarmada LBP algoritmasının, HOG algoritmasının kullanılmasına nazaran sınıflandırma başarı oranlarının artırılmasına çok daha olumlu etki ettiği görülmüştür. Bu çalışmanın literatüre katkıları şu şekilde özetlenebilir:

- Kamuflaj görüntüler üzerinde LBP ve HOG algoritmaları ile öznetelik çıkarılmış ve bu özneteliklere göre sınıflandırma işlemi yapılarak sonuçlar karşılaştırılmıştır. Elde edilen sonuçlara göre hangi özellik çıkarım yönteminin kamuflaj görüntülerde daha başarılı olduğu tespit edilmiştir.
- Kamuflaj görüntüler üzerinde CML yöntemleri kullanılmış ve bu yöntemler için kamuflaj görüntülerdeki sınıflandırma başarı oranları karşılaştırılarak en başarılı yöntem tespit edilmiştir

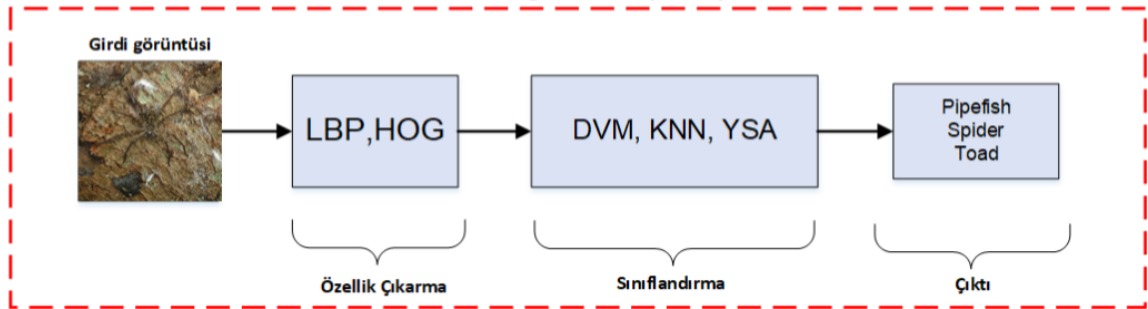
2.4.1. Önerilen Yöntem

Bu çalışmada COD10K kamuflaj veri seti kullanılmıştır. Bu veri setinde mevcut olan kamuflaj görüntülerden üç sınıf seçilmiştir. Şekil 48'de bu sınıflara ait örnek görüntüler gösterilmiştir. Bu görüntülere LBP ve HOG ile öznetelik çıkarımı yapılmıştır. Daha sonra CML yöntemlerinden ANN, SVM ve KNN yöntemleri ile sınıflandırma işlemi gerçekleştirilmiş ve görüntünün ait olduğu sınıf bilgisi belirlenmiştir.



Şekil 48. Çalışmada kullanılan sınıflara ait örnek kamuflaj görüntüler

Şekil 49’da bu çalışmada kurulan modelin genel mimari akış şeması gösterilmiştir. CML metotlarıyla kurulan bu model özellik çıkarma ve sınıflandırma aşamalarından oluşmaktadır. İlk önce kamuflaj görüntü sisteme girdi olarak verilir. Daha sonra LBP ve HOG algoritmalarıyla görüntünün dokusal özellikleri çıkarılır. Çıkarılan dokusal özellikler CML yöntemlerinden SVM, KNN ve YSA ile sınıflandırılır. Sınıflandırma işlemi sonrasında sisteme girdi olarak verilen kamuflaj görüntünün COD10K veri setinden seçilen üç sınıftan hangisine ait olduğu belirlenir.



Şekil 49. Çalışmanın genel metodolojik adımlarının şeması

Özellik Çıkarma: Kurulan sisteme girdi olarak verilen kamuflaj görüntülerinin aynı sınıfa ait olanlarının ayırt edici özelliklerinin belirlenmesi için yapılmaktadır. Aynı görüntü sınıfındaki nesnelere ait dokusal özellikler birbirlerine çok yakın olmaktadır. Bu çalışmada CML modelinde öznetelik çıkarma işlemi için LBP ve HOG yöntemleri kullanılmıştır.

Sınıflandırma: Girdi olarak sisteme verilen kamuflaj görüntülerin dokusal özellik vektörlerinin çıkarılmasından sonra sınıflandırma işlemi SVM, KNN ve ANN makine öğrenmesi yöntemleri kullanılmıştır.

2.5. Önceden Eğitilmiş Evrişimsel Sinir Ağı Kullanarak Kamuflaj Görüntülerinin Sınıflandırılması

Bu çalışmada literatürde sıklıkla kullanılan COD10K kamuflaj veri setinden seçilmiş 4 sınıfa ait kamuflaj görüntüler kullanılmıştır. Bu sınıflardaki görüntüler veri artırımı yöntemi uygulanarak görüntü sayıları artırılmıştır. Sınıflandırma başarı oranlarının hesaplanması için önceden eğitilmiş derin öğrenme yöntemlerinden DenseNet121, InceptionV3, Xception, VGG16 ve VGG19 yöntemleri kullanılmıştır. Bu modeller ile ham ve artırılmış verilerin kullanılmasıyla sırasıyla Deney 1 ve Deney 2 adlı iki deney yapılmıştır. Tüm deneylerde tüm yöntemler için kullanılan görüntülerin %10'u doğrulama, %20'si test ve %70'i de eğitim amaçlı kullanılmıştır. Yapılan deneysel çalışmalar sonucunda ham verinin kullanıldığı deney 1'den en yüksek sınıflandırma başarı oranı DenseNet121 ile %82,95 olarak elde edilmiştir. Artırılmış verinin kullanıldığı Deney 2' den ise yine DenseNet121 yöntemi ile %98,69 sınıflandırma başarı oranı elde edilmiştir. DenseNet121 modelinin kamuflaj görüntülerin sınıflandırılmasında çok başarılı olduğu tespit edilmiştir. Deneysel çalışmalarda kamuflaj görüntülerde veri artırma işleminin derin öğrenme teknikleriyle elde edilen sınıflandırma başarı oranlarına çok olumlu yönde etki ettiği ve başarı oranlarını önemli ölçüde artırdığı görülmüştür.

2.5.1. Önerilen Yöntem

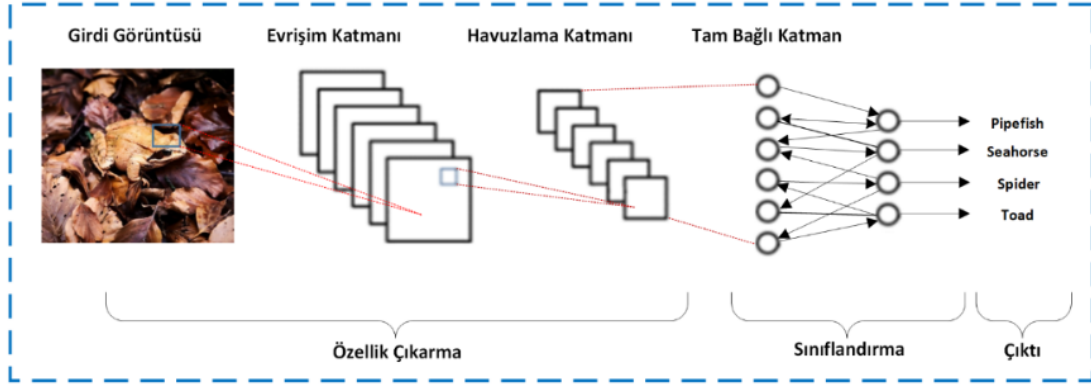
Bu çalışmada COD10K kamuflaj veri seti kullanılmıştır. Seçilen sınıflara ait örnek görüntüler Şekil 50'de gösterilmiştir. COD10K veri setinde yer alan mevcut kamuflaj görüntülerden 4 sınıf seçilerek DenseNet121, InceptionV3, Xception, VGG16 ve VGG19

derin öğrenme teknikleri ile sınıflandırma işlemi yapılmış ve sisteme girdi olarak verilen kamuflaj görüntülerin ait olduğu sınıf belirlenmiştir.



Şekil 50. Çalışmada kullanılan sınıflara ait örnek kamuflaj görüntüler

Şekil 51’de gösterildiği üzere kurulan modelde ilk olarak kamuflaj görüntüler sisteme girdi olarak verilir. Bu işlemden sonra evrişim ve havuzlama katmanları ile özellik çıkarma işlemi gerçekleştirilir. Son aşama olarak sınıflandırma işlemi ile sisteme girdi olarak verilen kamuflaj görüntüsünün 4 sınıftan hangisine ait olduğu tahmin edilir. Bu işlem için bu çalışmada önceden eğitilmiş mimarilerden olan DenseNet121, InceptionV3, EfficientNetV2, Xception, VGG16 ve VGG19 yöntemleri kullanılmıştır. Her bir yöntem eğitimde 100 iterasyon için çalıştırılmıştır



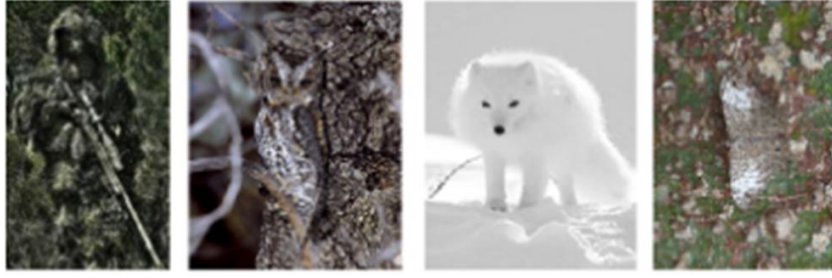
Şekil 51. Çalışmanın genel metodolojik adımlarının şeması

2.6. Yerel İkili Örüntüler (LBP) Kullanarak Kamufraj Görüntülerin Sınıflandırılması

Bu çalışmada CAMO-V.1.0 hazır veri setinden seçilerek alınan görüntüler kullanılmıştır. Tüm bu görüntülere LBP yöntemi uygulanarak dokusal öznitelikleri çıkarılmıştır. Çıkarılan bu öznitelikler kullanılarak ANN, KNN, SVM yöntemleri ile sınıflandırma işlemi gerçekleştirilmiştir. Deneysel çalışmalar sonucunda LBP ve ANN yöntemlerinin birlikte kullanılmasıyla en yüksek sınıflandırma başarı oranı %92 olarak elde edilmiştir.

2.6.1. Önerilen Yöntem

Bu çalışmada CAMO-V.1.0 veri setinden seçilerek alınmış görüntüler kullanılmıştır. Bu veri setinde kategorize edilmemiş birçok farklı türe ait görüntüler bulunmaktadır. Bu görüntülerden kamufraj asker, baykuş, beyaz kurt ve kelebek görüntüleri seçilerek dört sınıflı bir veri seti oluşturulmuştur. Oluşturulan bu veri setindeki görüntülerin her birisi 45° derece saat yönünde döndürülerek çoğaltılmıştır. Bu şekilde veri setindeki görüntülerden yeni görüntüler türetilmiş ve görüntü sayısı artırılmıştır. CAMO-V.1.0 veri setinden seçilerek alınan görüntülerin toplam sayısı 200'dür. Deneysel çalışmalarda artırılmış görüntülerin %90'ı eğitim, %10'uda test için kullanılmıştır. Şekil 52'de bu çalışma için dört farklı türden oluşturulan veri setine ait örnek görüntüler verilmiştir.



(a) Veri kümesindeki örnek kamuflaj asker, baykuş, beyaz kurt ve kelebek görüntüleri



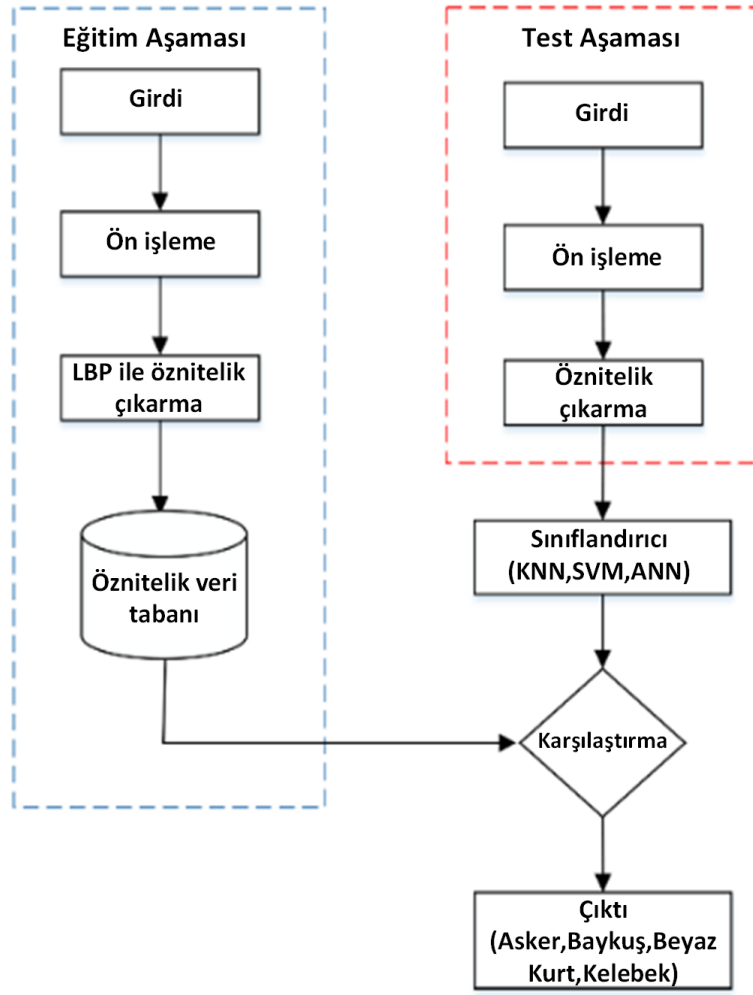
(b) Veri kümesindeki maskelenmiş örnek kamufle olmuş asker, baykuş, beyaz kurt ve kelebek görüntüleri.



(c) Veri kümesindeki örnek kamufle olmuş asker, baykuş, beyaz kurt ve kelebek ikili görüntüleri

Şekil 52. CAMO-V.1.0 veri setinden seçilen türlere ait örnek görüntüler

Bu çalışmada önerilen yöntemin genel akış şeması Şekil 53’de gösterilmiştir. Çalışmada kullanılan görüntülere LBP ile öznitelik çıkarma işlemi uygulanır ve öznitelik veri tabanı oluşturulur. Bu öznitelik veri tabanı ile danışmanlı öğrenme gerçekleştirilir ve KNN, SVM ve ANN yöntemleri kullanılarak bir sınıflandırıcı kurulur. Sonraki aşamada ise kamuflaj test görüntüleri bu sınıflandırıcılara sunularak girdi olarak verilen kamuflaj görüntüsünün asker, baykuş, beyaz kurt ya da kelebek türlerinden hangisine ait olduğu tespit edilir.



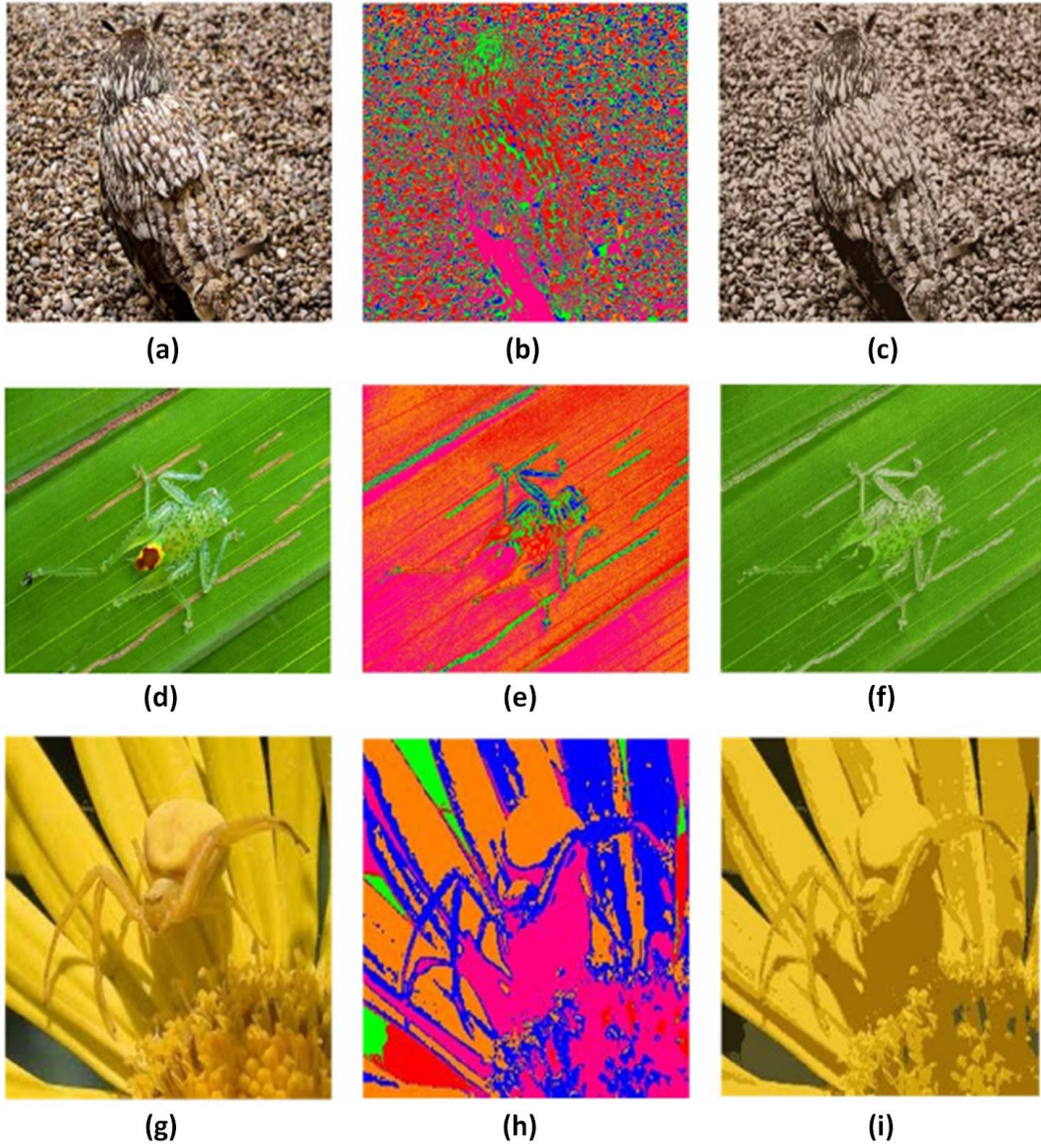
Şekil 53. LBP tabanlı çalışmanın genel akış şeması

2.7. Kamuflaj Görüntülerde Öklid ve Mahalanobis Uzaklık Hesabına Göre K-means Kümeleme Algoritması Kullanarak Görüntü Bölütleme

Bu çalışmada CAMO-V.1.0 kamuflaj veri setindeki görüntüler üzerinde Öklid ve Mahalanobis uzaklık hesapları ve K-means kümeleme algoritması kullanılarak görüntü bölütleme işlemi yapılmıştır. Çalışmada K-means yöntemi ile Öklid ve Mahalanobis uzaklıkları kullanılmış ve bu iki uzaklık ölçütünün K-means yönteminin başarısına etkisi karşılaştırılmıştır. Öklid uzaklık hesabı ile K-Means yöntemi kullanıldığında ortalama Kök Ortalama Kare Hatası (RMSE) değeri 262,47 ve ortalama Tepe-Sinyal Gürültü Oranı (PSNR) değeri 24,26 olarak elde edilmiştir. Mahalanobis uzak hesabı ile K-means yöntemi kullanıldığında ortalama RMSE değeri 799,62 ve ortalama PSNR değeri de 19,66 olarak elde edilmiştir. Bu sonuçlara göre Öklid uzaklık hesabı ile gerçekleştirilen K-means yönteminin,

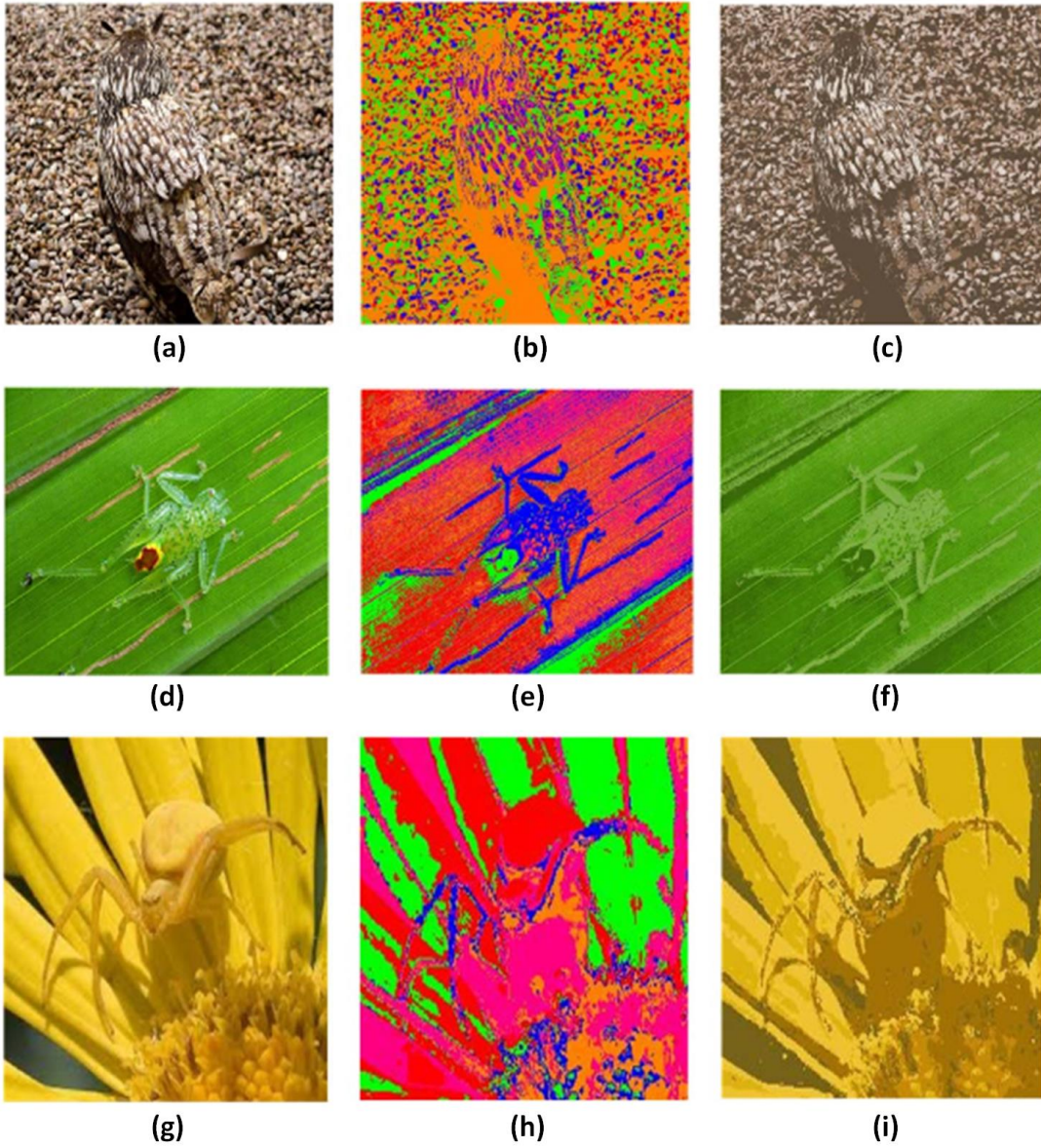
Mahalanobis uzaklık hesabıyla gerçekleştirilen K-means yöntemine göre daha başarılı olduğu tespit edilmiştir.

Şekil 54'te kamuflaj görüntüler için Öklid uzaklık hesabıyla gerçekleştirilen K-means yöntemiyle elde edilen örnek çıktı görüntüleri gösterilmiştir.



Şekil 54. (a),(d),(g) Orijinal görünüş; (b),(e),(h) Etiketli görüntü; (c),(f),(i) Öklid hesaplamalı K-means çıktı görüntü.

Şekil 55'de kamuflaj görüntüler için Mahalanobis uzaklık hesabıyla gerçekleştirilen K-means yöntemiyle elde edilen örnek çıktı görüntüleri gösterilmiştir.



Şekil 55. (a),(d),(g) Orijinal görünüş; (b),(e),(h) Etiketli görüntü; (c),(f),(i) Mahalanobis hesaplamalı K-means çıktı görüntü.

2.8. Yerel İkili Desenlerle Özellik Çıkarımına Dayalı Kelebek Kamuflaj Görüntülerinin Sınıflandırılması

Bu çalışmada internet üzerinden arama motorlarından tarama yapılarak elde edilen kamuflaj kelebek görüntülerinden dört sınıflı bir kamuflaj görüntü veri seti oluşturulmuştur. Şekil 56’da bu veri seti için örnek görüntüler gösterilmiştir. Bu veri setindeki görüntülere LBP yöntemi uygulanarak dokusal öznitelik çıkarımı yapılmıştır. Çıkarılan bu özellik vektörlerine göre sınıflandırma işlemi gerçekleştirilmiştir. Bunun için SVM, ANN ve KNN

yöntemleri kullanılmıştır. Deneysel çalışmalar sonucunda LBP + ANN yöntemi ile %90, LBP + KNN yöntemi ile %82, LBP + SVM yöntemi ile %93 sınıflandırma başarı oranları elde edilmiştir.



(a) Veri setindeki örnek kamuflaj kelebek görüntüleri.



(b) Veri setindeki örnek maskelenmiş kamuflaj kelebek görüntüleri



(c) Veri setindeki örnek kamuflaj kelebek ikili görüntüleri.

Şekil 56. Veri setindeki türlere ait örnek kamuflaj görüntüler

Bu çalışmada, öncelikle kelebek veri setindeki tüm görüntüler için yerel ikili desen (LBP) ile özellik çıkarma işlemi gerçekleştirildi. Veri seti, açık kaynaklı internetten araştırılıp indirilen kamuflaj kelebek görüntülerinden oluşmaktadır. Bu veri seti, kamuflaj görüntülerinin sınıflandırma doğruluğu başarı oranlarını hesaplamak için kullanılmıştır. Kamuflaj kelebek görüntülerinden oluşan bu veri setinden LBP ile dokusal özellikler çıkarılmıştır. Özellik çıkarma işleminden sonra, sınıflandırma işlemi için ANN, KNN ve SVM gibi sınıflandırıcılar kullanılmıştır.

3. BULGULAR VE TARTIŞMA

Bu bölümde tez çalışması için önerilen modellerimizle yapılan tüm deneyler ve elde edilen deneysel sonuçlar ayrıntılı olarak tartışılmakta ve detayları sunulmaktadır. Tez çalışması için toplamda 8 farklı çalışma yapılmıştır. Bu çalışmalar için kendi oluşturduğumuz orijinal ERVA 1.0 kamuflaj veri seti ve literatürde kullanılan COD10K ve CAMO kamuflaj veri setleri kullanılmıştır. Yapılan tüm deneysel çalışmalar Intel(R) Core(TM) i7- CPU (3.20GHz) 8 çekirdek işlemcili, NVIDIA GeForce GTX 1050 TI 4GB-128 bit ekran kartlı ve 16GB ana belleğe sahip bir bilgisayarda gerçekleştirilmiştir. Yazılımın çalıştırılması için Python 3.10.0 sürümü ve Matlab R2017b (9.3.0.713579) sürümü ve ilgili kütüphaneleri kullanılmıştır. Tez çalışması için yapılan sekiz farklı çalışmada elde edilen bulgular aşağıdaki şekilde başlıklar halinde açıklanmaktadır.

3.1. “Kamuflaj Görüntülerde Kelebek Nesnelerinin Bölütlenmesi ve Tanınması İçin Hibrit Bir Yöntem Geliştirilmesi” Çalışmasından Elde Edilen Bulgular

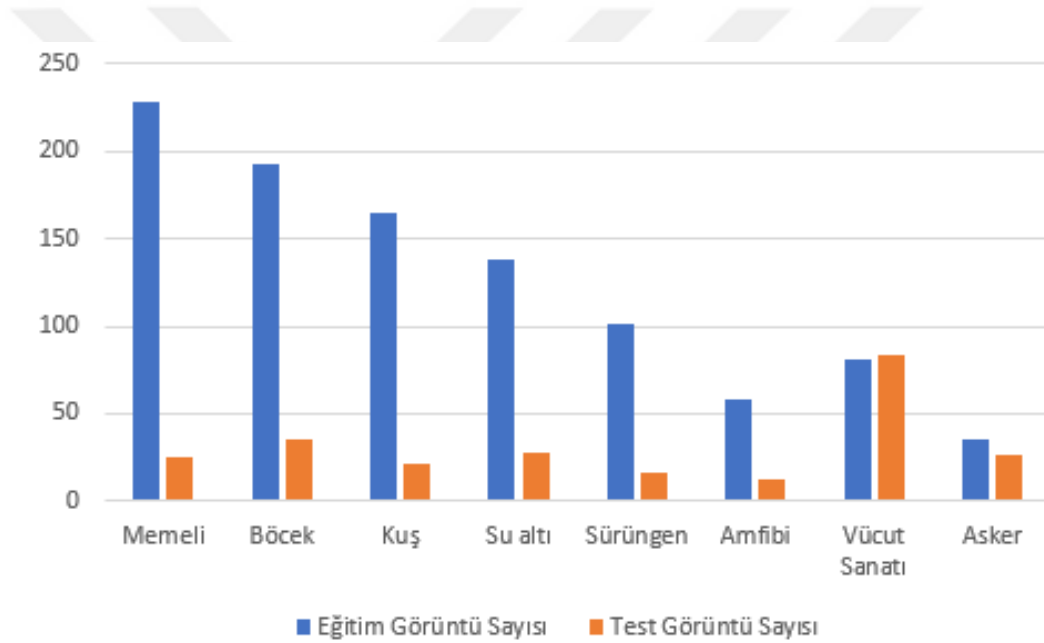
Bu çalışma bölütleme ve nesne tanıma aşamalarından oluşmaktadır. Bölütleme işlemi için Gabor Filtre, K-means algoritması ve LBP algoritması kullanılmıştır. Bölütleme başarı oranlarını tespit etmek için DSC ve SSIM metrikleri kullanılmıştır.

Bölütleme sonrası elde edilen kelebek nesnelerinin türlerinin tahmin edilmesi için önceden eğitilmiş derin öğrenme modelleri kullanılmıştır. Bu farklı derin öğrenme tekniklerinin performansı Kesinlik, Duyarlılık, F1-Score, Doğruluk, F-Ölçütü değerlendirme ölçütleri kullanılarak ölçülmüştür.

Hata matrisleri ile sınıflandırma algoritmalarının performansı değerlendirilmektedir. Bu çalışmada önerilen model için iki adet deney yapılmıştır. Bunlardan birincisinde veri artırımı uygulanmamış ham 1180 adet eğitim görüntüsü kullanılmıştır. İkinci deneyde ise eğitim görüntülerine veri artırımı uygulanması sonucu elde edilen 4720 eğitim görüntüsü kullanılmıştır.

Bu çalışmada ERVA 1.0 veri setinin yanı sıra literatürde çok zorlu bir veri seti olan CAMO veri seti üzerinde önerdiğimiz hibrit yöntem ile bölütleme işlemi gerçekleştirilmiştir. Elde edilen sonuçlar çok başarılı olmamıştır. Bunun sebebi CAMO veri setinde yeterli derecede eğitim ve test görüntüleri bulunmamaktadır. Test verisi olarak verilen görüntülerin bir kısmı için eğitim verilerinde tanımlı görüntü bulunmamaktadır. Bu sebeple eğitim

verilerinden elde edilen LBP özellik vektörlerinden oluşturulan özellik veri tabanı ile test görüntüden elde edilen LBP özellik vektörlerinin kullanılmasıyla kamufle olmuş nesne bölge tespiti başarılı bir şekilde yapılamamıştır. Bu da bölütleme başarı oranını düşürmüştür. Literatür incelendiğinde CAMO veri setiyle yapılan bölütleme çalışmalarında yüksek başarı oranlarının tespit edilemediği görülmüştür. Bunun en büyük sebebi Şekil 57’de gösterildiği üzere yeterli derecede eğitim ve test verisinin bulunmamasıdır. Örneğin CAMO veri setinde yer alan bazı türler için eğitim ve test görüntü sayıları şu şekildedir: Asker için eğitim görüntü sayısı 36, test görüntü sayısı 26; Amfibi türü için eğitim görüntü sayısı 58, test görüntü sayısı 13 tür. Bunun gibi diğer türlerde de görüntü sayıları oldukça az sayıdadır.



Şekil 57. CAMO veri setinde yer alan türlere ait sayısal veriler

Çalışmada kamuflej görüntülerinin bölütlenmesi için Gabor Filte, K-means kümeleme algoritması ve LBP algoritmasının birlikte kullanılmasıyla oluşturduğumuz hibrit yöntemden elde edilen bölütleme başarı oranları ile ERVA 1.0 ve CAMO veri setinden elde edilen sonuçlar Tablo 3’te literatürdeki benzer çalışmalarla birlikte verilmiştir. Ayrıca Tablo 3’de literatürdeki benzer çalışmalar gösterilmiştir.

Tablo 3. Bölütleme başarı oranlarının gösterilmesi

Model	Değerlendirme Ölçütü	Sonuç (%)	Veri Seti
FPN (Lin vd., 2017)	Weighted F-measure	48,30	CAMO
MaskRCNN (He vd., 2017)	Weighted F-measure	43,00	CAMO
PSPNet (Zhao vd., 2017)	Weighted F-measure	45,50	COCO
MSRCNN (Huang vd., 2019)	Weighted F-measure	45,40	COCO
ANet-SRM (Le vd., 2019)	F-measure	65,40	CAMO
SINet'20 (Fan et al., 2020)	Weighted F-measure	60,60	CAMO
MirrorNet (Yan vd., 2020)	Weighted F-measure	71,90	CAMO
Zero-Shot Segmentators (Nanni vd., 2023)	IoU	73,46	CAMO
JCnet (Jiang et al.,2023)	Weighted F-measure	80,0	CAMO
SAM2-Adapter (Chen et al.,2024)	F-measure	85,00	COD10K
Önerilen Hibrit Yöntemimiz (2025)	DSC	61,10	CAMO
Önerilen Hibrit Yöntemimiz (2025)	SSIM	70,76	CAMO
Önerilen Hibrit Yöntemimiz (2025)	DSC	83,64	ERVA 1.0
Önerilen Hibrit Yöntemimiz (2025)	SSIM	87,89	ERVA 1.0

Bu çalışmada, bölütleme ve morfolojik işlemlerden sonra elde edilen kelebek nesnesinin türünü tahmin etmek için önceden eğitilmiş derin öğrenme modelleri kullanılmıştır. Bu modeller kullanılarak iki ayrı deney gerçekleştirilmiştir.

İlk deney için, ERVA 1.0 veri setinde veri artırma olmaksızın yer alan 1180 ham eğitim görüntüsünden 1058'i eğitim için, 122'si ise doğrulama için kullanılmıştır. Ayrıca 597 kamuflaj test görüntüsü de test amaçlı kullanılmıştır.

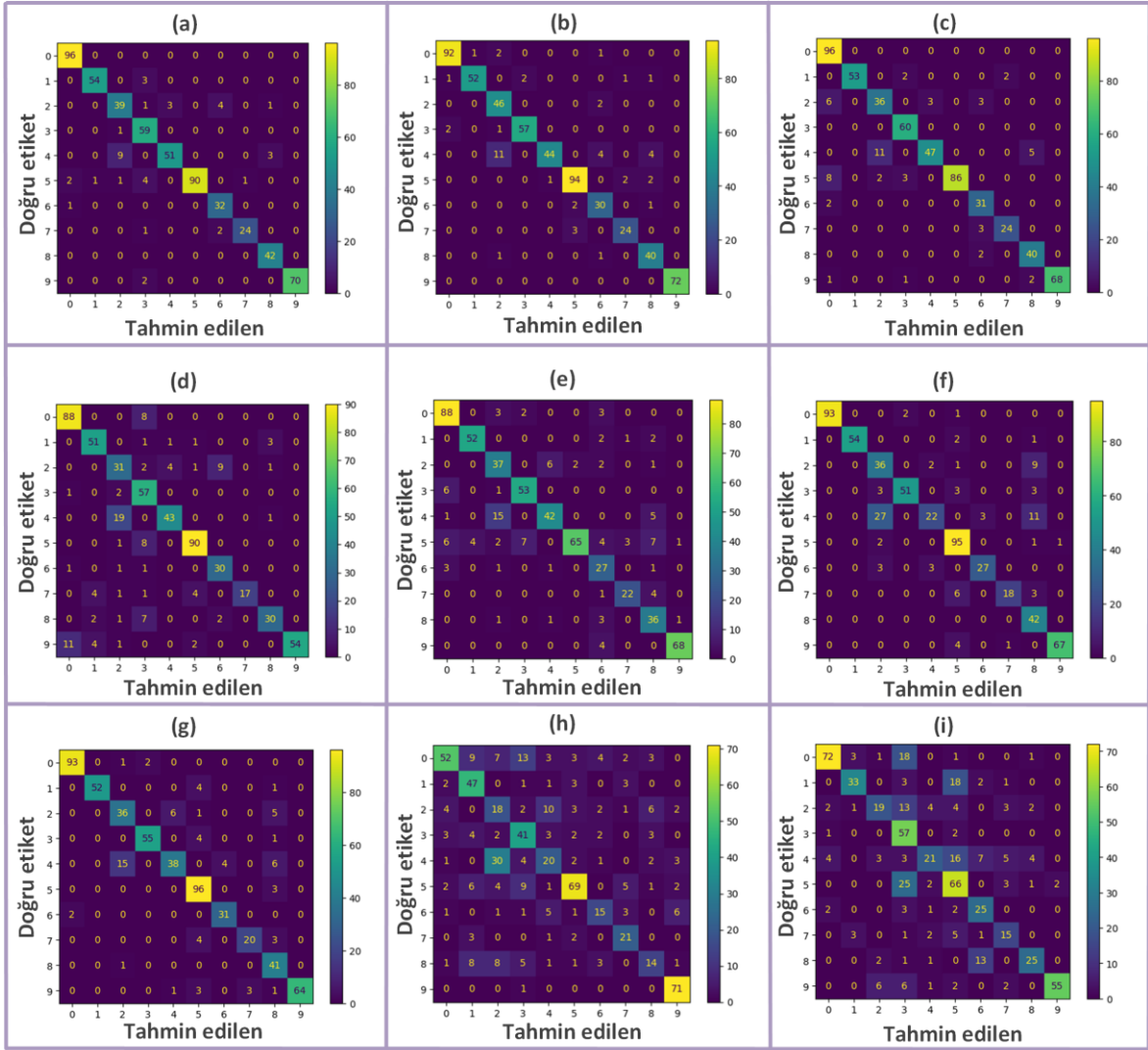
İkinci deney için, veri artırma teknikleri uygulanarak eğitim görüntülerinin sayısı %300 oranında artırılmış ve 4720 eğitim görüntüsünden 4245'i eğitim için, 475'i ise doğrulama için kullanılmıştır. İkinci deneyde, 597 kamuflaj test görüntüsü de test amaçlı kullanılmıştır.

Şekil 58'de Deney (1) ve Şekil 60'ta Deney (2) sonucunda çalışmada kullanılan tüm önceden eğitilmiş derin öğrenme modelleri için elde edilen karmaşıklık matrisleri gösterilmektedir. Matrislerde yer alan etiketler ERVA 1.0 veri setindeki sınıfları temsil etmektedir. Bunlar: 0:Pipefish, 1:Seahorse, 2:Cat, 3:Lizard, 4:Spider, 5:Bird, 6:Butterfly, 7:Grasshopper, 8:Owl, 9:Toad şeklindedir.

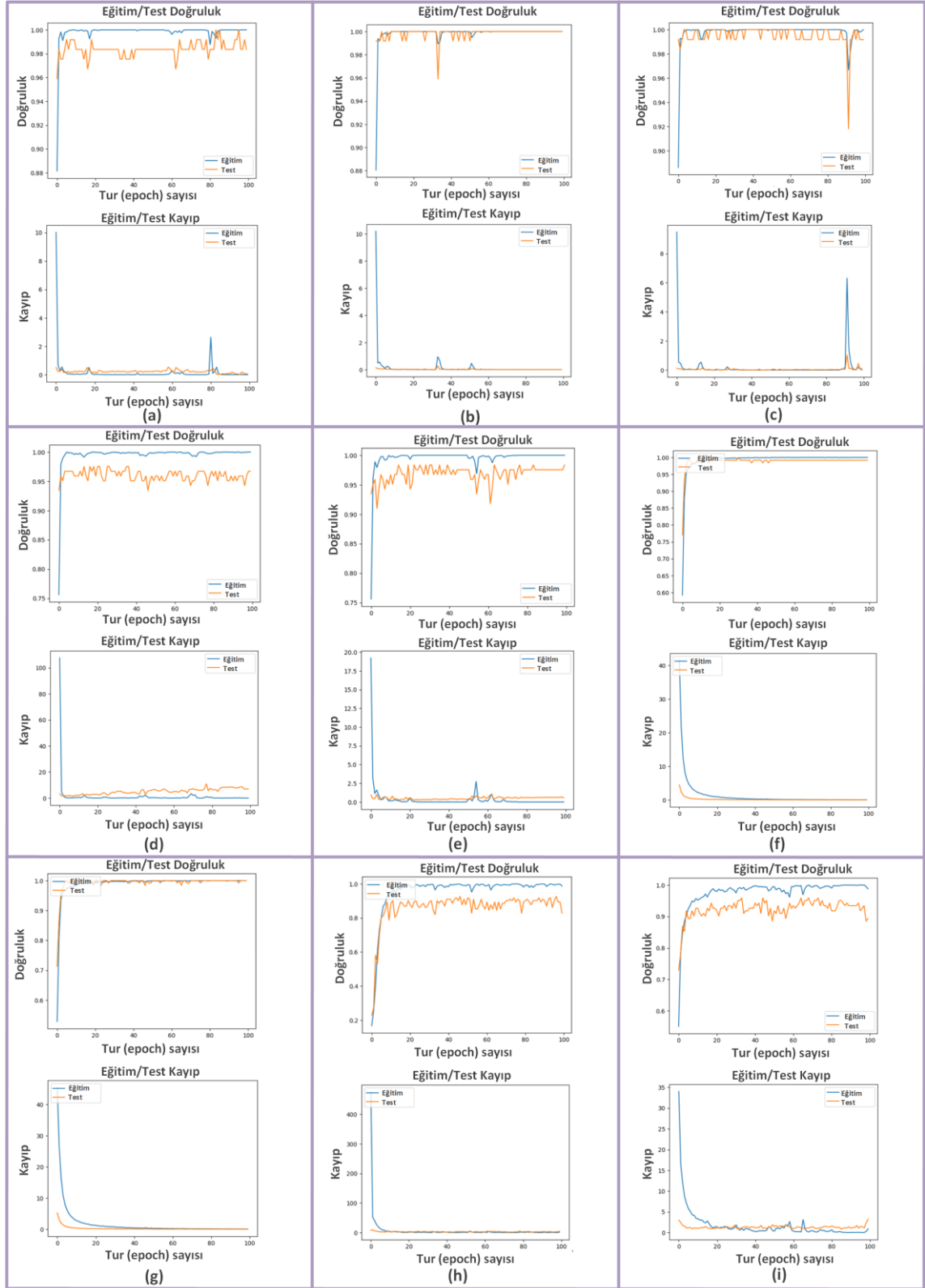
Tablo 4'te gösterildiği gibi, her deney için performans değerlendirmesi, Şekil 58 ve Şekil 60'ta gösterilen karmaşıklık matrislerinden elde edilmiştir. Deney (1)'den önceden eğitilmiş derin öğrenme modellerinin kullanılması sonucu elde edilen maksimum doğruluk oranı InceptionResNetV2 yöntemiyle %92,29 iken, ikinci deneyde en yüksek doğruluk oranı DenseNet121 modeliyle %94,81 olmuştur.

Tablo 4. Karmaşıklık matrislerine göre modellerden elde edilen en yüksek başarı sonucunun performans karşılaştırması.

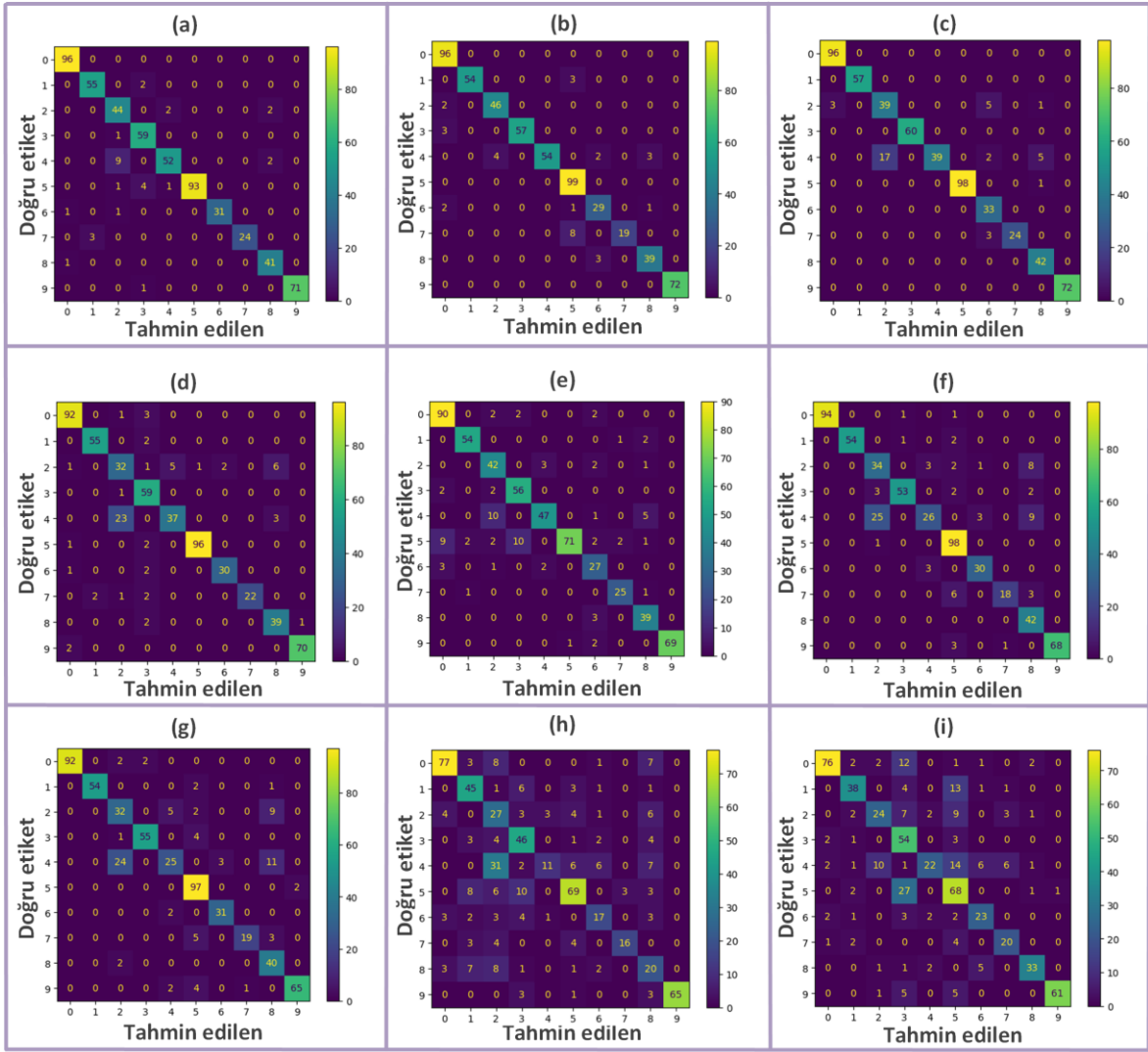
Deney Adı	Model	F- measure	Precision	Recall	F1- score	Accuracy (%)
Deney(1)	InceptionResNet V2	0.9114	0.9109	0.9177	0.9142	92.29
Deney(2)	DenseNet121	0.9460	0.9464	0.9420	0.9441	94.81



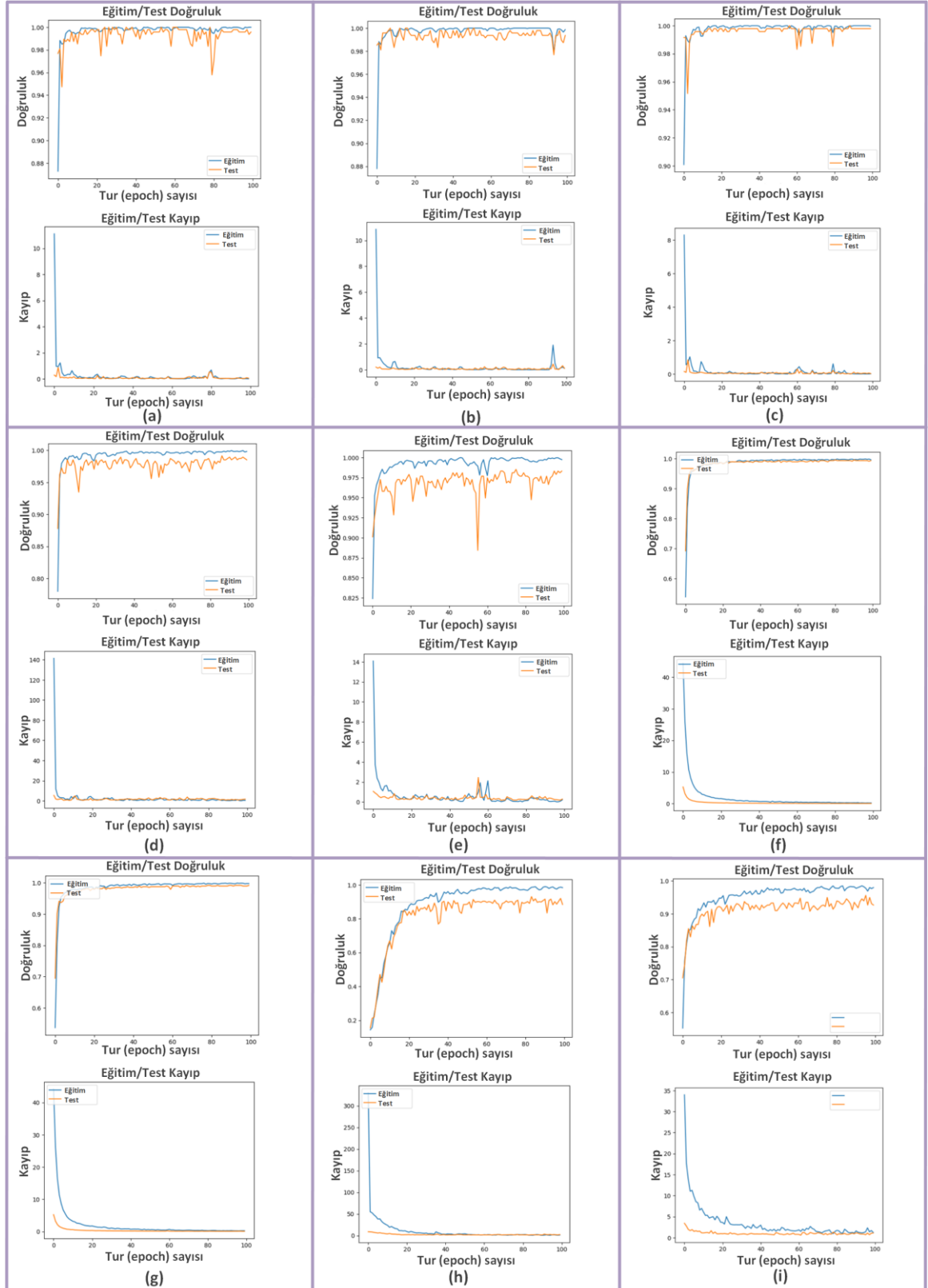
Şekil 58. Çalışmada kullanılan ön eğitilmiş modellere göre Deney (1)'den elde edilen karmaşıklık matrisleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d) Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50



Şekil 59. Çalışmada kullanılan ön eğitilmiş modeller ile Deney (1)'den elde edilmiş Eğitim / Test Doğruluk ve Eğitim / Test Kayıp Grafikleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d) Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50



Şekil 60. Çalışmada kullanılan ön eğitilmiş modellere göre Deney (2)'den elde edilen karmaşıklık matrisleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d) Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50



Şekil 61. Çalışmada kullanılan ön eğitilmiş modeller ile Deney (2)'den elde edilmiş Eğitim / Test Doğruluk ve Eğitim / Test Kayıp Grafikleri: (a) DenseNet121, (b) InceptionResNetV2, (c) DenseNet201, (d) Xception, (e) InceptionV3, (f) VGG16, (g) VGG19, (h) AlexNet, (i) ResNet50

Şekil 59 çalışmada kullanılan tüm önceden eğitilmiş modellerin kullanıldığı Deney (1)'den 100 tur (epoch) için elde edilen eğitim/test doğruluğu ve eğitim/test kaybı grafiklerini göstermektedir. Yine aynı şekilde Şekil 61 çalışmada kullanılan tüm modeller için Deney (2)'den 100 tur için elde edilen eğitim/test doğruluğu ve eğitim/test kaybı grafiklerini göstermektedir. Hem test hem de eğitim doğruluğu eğrileri, tur sayısı arttıkça artan bir eğim gösterir.

Kayıp eğrisi Şekil 59 ve Şekil 61'de gösterildiği gibi hata oranındaki azalmayı gösterir. Bu ağırlık eğitim sürecinin ve öğrenmesinin optimum bir hızda ilerlediğini gösterir. Şekil 59 ve Şekil 61'de gösterildiği gibi, kayıp değeri her turda azaldıkça, doğruluk oranı verilen eğitim setiyle artar ve öğrenme gerçekleşir.

Şekil 59 ve Şekil 61 incelendiğinde, Deney (1) ve Deney (2) için eğitim/test doğruluğu ve eğitim/test kaybı grafikleri karşılaştırılırsa, veri artırımı yapılmamış ham eğitim verilerinin kullanıldığı Deney (1)'deki bazı yinelemelerde görülen düşük doğruluk değerleri, artırılmış eğitim verilerinin kullanıldığı Deney (2)'de azaldığı görülür. Benzer şekilde Deney (1)'deki bazı yinelemelerde görülen yüksek hata oranları Deney (2)'de azalmıştır.

Bu çalışmada kullanılan tüm önceden eğitilmiş derin öğrenme modelleriyle elde edilen başarı oranlarını karşılaştırmak için bir karşılaştırma yapıldı. Deneyler (1) ve (2) için her model 100 yineleme için çalıştırıldı. Deneysel çalışmalarda, modellerin performansının geçerliliğini artırmak için on katlı çapraz doğrulama kullanıldı. Deney (1) ve Deney (2) için yöntemlerin doğruluğu on katlı çapraz doğrulama kullanılarak ayrı ayrı test edildi. Her model için on katlı çapraz doğrulama ortalama doğruluk oranlarıyla sonuçlanmıştır.

Bu çalışmada oluşturulan ERVA 1.0 veri setinde Şekil 20'de gösterildiği üzere bazı sınıflara ait görüntü sayıları diğer sınıflara göre farklılık göstermektedir. Veri setindeki sınıflara ait görüntü sayıları eşit dağılıma sahip değildir. Her ne kadar birbirlerine yakın sayıda görüntü sayısı seçilmeye çalışılsa da ERVA 1.0 oluşturulurken arama motorlarında bazı türlere ait sınırlı sayıda görüntü olduğundan tüm sınıfların görüntü sayıları birbirlerine eşit olarak ayarlanamamıştır. Bu durum çalışmamızda kısmen sınıf dengesizliğine (Class Imbalance) sebep olmuştur. Bu dengesizliğin sebep olabileceği performans düşüklüklerinin analiz edilebilmesi için modellerin sınıflandırma performans ölçütlerinde Accuracy ile birlikte Precision, Recall, F1-Score gibi denge duyarlı metrikler de kullanılmıştır. Veri setindeki sınıflar arasındaki sınıf dengesizliği problemiyle başa çıkmak için bu çalışmada Ağırlıklı Kategorik Çapraz Entropi kayıp fonksiyonu kullanılmıştır. Bu fonksiyon çalışmadaki sınıf dengesizliği problemiyle başa çıkmak için kullanılmıştır. İlerleyen

çalışmalarda ERVA 1.0 veri setinde kısmen bulunan sınıf dengesizliği problemini ortadan kaldırmak için veri setindeki tüm sınıfların görüntü sayılarının birbirine eşit ya da çok yakın sayıya ulaşması sağlanacaktır.

Bu çalışmada kullanılan tüm önceden eğitilmiş derin öğrenme modelleriyle elde edilen başarı oranları Tablo 5'teki Deney (1) ve Tablo 6'daki Deney (2)'ye göre her kriter için verilmiştir. Deney (1)'de InceptionResNetV2 %92,29 ile en başarılı model iken, Deney (2)'de DenseNet121 %94,81 ile en başarılı model olmuştur.

Tablo 5. Deney (1) için önceden eğitilmiş modellerin performans karşılaştırması

Model	F-measure	Precision	Recall	F1-score	Accuracy (%)
InceptionResNetV2	0.9114	0.9109	0.9177	0.9142	92.29
DenseNet121	0.9005	0.9002	0.9040	0.9020	90.45
DenseNet201	0.9003	0.9002	0.9020	0.9010	90.62
VGG19	0.8744	0.8752	0.8666	0.8708	88.11
VGG16	0.8537	0.8565	0.8240	0.8399	84.59
Xception	0.8303	0.8325	0.8002	0.8160	82.24
InceptionV3	0.8115	0.8104	0.8241	0.8171	82.08
AlexNet	0.6516	0.6512	0.6566	0.6538	65.18
ResNet50	0.6715	0.6751	0.6347	0.6342	64.99

Tablo 6. Deney (2) için önceden eğitilmiş modellerin performans karşılaştırması

Model	F-measure	Precision	Recall	F1-score	Accuracy (%)
DenseNet121	0.9460	0.9464	0.9420	0.9441	94.81
InceptionResNetV2	0.9480	0.9504	0.9224	0.9361	94.64
DenseNet201	0.9292	0.9291	0.9310	0.9300	93.80
Xception	0.8869	0.8880	0.8755	0.8817	89.11
InceptionV3	0.8632	0.8619	0.8787	0.8702	87.10
VGG16	0.8642	0.8661	0.8441	0.8549	86.60
VGG19	0.8518	0.8533	0.8364	0.8447	85.43
AlexNet	0.6943	0.6943	0.6943	0.6943	70.35
ResNet50	0.7281	0.7311	0.6965	0.7133	70.18

Tablo 7. Kamuflej veri setleri üzerinde yapılan literatürdeki benzer son teknoloji çalışmalarının performans karşılaştırmaları

Model / Referans	Değerlendirme Ölçütleri (%)	Veri Seti				
		CAMO	COD10K	NC4K	CAMO-COCO	ERVA 1.0
SVM-BOW (Le et al.,2019)	Accuracy= 54,0				✓	
AlexNet (Le et al.,2019)	Accuracy= 76,0				✓	
VGG-16 (Le et al.,2019)	Accuracy= 88,6				✓	
UR-SINet-v2 (Kajiura et al.,2021)	F-measure=78,70					
CubeNet (Zhuge et al.,2022)	Max F-measure= 78,8 Mean E-measure= 83,8	✓				
ERRNet (Ji et al.,2022)	Weighted F-measure= 66,0 Mean E-measure= 81,7	✓				
ERRNet (Ji et al.,2022)	Weighted F-measure= 62,9 Mean E-measure= 86,7		✓			
DQnet (Sun et al.,2022)	Weighted F-measure= 89,8 E-measure=94,4	✓				
DQnet (Sun et al.,2022)	Weighted F-measure= 80,1 E-measure=93,0		✓			
DQnet (Sun et al.,2022)	Weighted F-measure= 85,9 E-measure=94,2			✓		
VSSNet (Yin et al.,2024)	F-measure =85,50			✓		
HGINet (Yao et al., 2024)	F-measure =82,10		✓			
FGSA-Net (Zhang et al.,2024)	F-measure =84,90		✓			
Önerilen hibrit yöntemimiz (2025)	Accuracy=76,00	✓				
Önerilen hibrit yöntemimiz (2025)	Accuracy= 94,81 F-measure = 94,60					✓

Bu çalışmada ayrıca önerdiğimiz hibrit yöntem CAMO kamuflaj veri seti üzerinde uygulanmıştır. CAMO veri seti oldukça zorlu kamuflaj görüntülerden oluşmasına rağmen çok az sayıda görüntüden oluşmaktadır. Veri setindeki türlere ait eğitim görüntü sayıları oldukça az olduğundan eğitim işlemi başarılı olarak yapılamamış ve kullandığımız test verileriyle başarılı sonuçlar elde edilememiştir. CAMO veri setinin kullanılmasıyla hibrit yöntemimizden elde edilen sınıflandırma başarı oranı Tablo 7’de gösterilmiştir.

Kamuflaj görüntüleri üzerine yapılan çalışmalar literatürde analiz edilmiştir. Bu çalışmalarda çeşitli yöntem ve veri kümelerinin performansları incelenmiştir. Bu çalışmaların başarılı sonuçlar elde edenlerin bir kısmı Tablo 7’de gösterilmiştir. Tablo 7’de bölütleme sonucu elde ettiğimiz kamufle olmuş nesnelerin türlerinin belirlenmesi için yaptığımız sınıflandırılma işlemi için elde edilen sonuçlar verilmiştir. Bu çalışmamızda ERVA 1.0 veri seti ve CAMO veri seti üzerinde önerdiğimiz hibrit yöntemimiz uygulanmıştır. Literatürde kamuflaj görüntüleri üzerine yapılan çalışmalardan elde edilen sonuçlar, önerdiğimiz modelden elde edilen sonuçlarla karşılaştırılmıştır. Doğruluk açısından, önerdiğimiz hibrit yaklaşımın diğer yöntemlerden daha iyi performans gösterdiğini gözlemlemekteyiz.

Bu çalışmada, ERVA 1.0 test görüntülerindeki 10 türe ait 10 kamuflajlı kelebek arka planlarından ayrılarak türleri tanımlanmıştır. Dünyada binlerce kelebek türü bulunmaktadır. Bunların çoğu kamuflajlıdır. Farklı türlere ait kamuflajlı kelebekleri arka plandan ayırmak ve türlerini belirlemek, çok benzer doku ve desen özelliklerine sahip oldukları için zorlu bir COD problemidir. Aslanlar, kuşlar ve örümcekler gibi kamuflajlı hayvanlardan oluşan bir veri kümesi üzerinde çalışırken, bu türleri birbirinden ayırt etmek ve tanımlamak daha kolaydır çünkü çok farklı doku özelliklerine ve şekillere sahiptirler. Bu nedenle, daha zorlu bir problem olduğu için bu çalışma kelebek türleri üzerinde yapılmıştır.

Bu çalışmada karşılaşılan en zorlu problemlerden biri ERVA 1.0 veri setinin oluşturulmasıdır. Literatürde kamuflaj veri setlerinde farklı hayvan veya nesne türlerine ait görüntüler bulunmasına rağmen, bu görüntülerin tür bazında sayısı çok azdır. Bu veri setlerinde yalnızca kamuflaj kelebek görüntülerinin sayısı ihmal edilebilir düzeydedir. Bu kadar az görüntü ile derin öğrenme teknikleri kullanılarak kamuflaj kelebekleri üzerinde sağlıklı sonuçlar elde etmek mümkün değildir. Bu nedenle bir kelebek kamuflaj veri seti oluşturulmuştur.

Bu çalışmada, LBP algoritması için kamuflaj görüntüleri her biri 8x8'lik bir ızgara olan 64 bölgeye ayrılmıştır. Bu bize görüntünün yerel özelliklerini ayrıntılı olarak yakalamak için

yeterli çözünürlüğü sağlar. Her bölge için, o alanın doku yapısını temsil eden bir LBP histogramı oluşturulur. Görüntüyü 64 bölgeye bölmek bize yeterli bir çözünürlük sağladı ve hesaplama maliyetini makul bir seviyede tuttu. Çok fazla bölge hesaplama maliyetini artırabilirken, çok az bölge yerel bilginin kaybolmasına yol açabileceğinden, bölge sayısı literatürde de yaygın olarak kullanılan 64 olarak seçilmiştir. Kamufraj görüntüleri için 64 bölgeye ek olarak, çalışmada 16 ve 256 bölge de test edilmiştir. Daha az bölgeye sahip 16 bölgede daha genel özellikler çıkarılmış ve yerel detaylar kaybolmuştur. LBP algoritması için, görüntü 256 bölgeye bölündüğünde çok daha ayrıntılı bilgi elde edildi, ancak hesaplama maliyeti arttı ve bazı durumlarda aşırı ayrıntı nedeniyle genelleme yeteneği azaldı. LBP algoritması için, görüntüler 16, 64, 256 farklı bölgeye bölündü ve gözlenen sonuçlara göre, doğruluk, hesaplama verimliliği ve genelleme yeteneği açısından en iyi dengeyi sunduğu için 64 bölge tercih edildi.

K-means yöntemi, verileri özellik benzerliğine göre kümelere ayıran denetimsiz bir öğrenme algoritmasıdır. K-means, benzer desenlerin otomatik olarak gruplanmasını sağlar. Kamufraj görüntülerinde kamufrajlı nesneler ile karmaşık arka planları arasında ayrım yapmak için etkilidir. K-means kümeleme mekanizması, Gabor filtreleri ve LBP gibi doku tabanlı özellik çıkarma yöntemlerini tamamlar. Benzer doku özelliklerini gruplayarak nesne ve arka planın minimum yoğunluk farkına sahip olduğu görüntülerde bölütleme sınırlarını iyileştirmeye yardımcı olur. Çalışmada gerçekleştirdiğimiz ön deneylerde, K-means yönteminin kamufraj kelebek veri setindeki doku desenlerini etkili bir şekilde grupladığı ve bölütleme doğruluğunu artırdığı görüldü. Literatürdeki önceki çalışmalar incelendiğinde, K-means yönteminin kamufraj görüntüleri üzerindeki bölütleme sürecinde başarılı sonuçlar verdiği görülmektedir (Bayram & Nabiye, 2020). Tüm bu nedenler, K-means yöntemini hibrit yaklaşımımıza entegre etmemizi teşvik etti.

Çalışmada öncelikle kamufraj ERVA 1.0 veri setinde test kamufraj kelebek görüntüleri üzerinde bölütleme yapılmıştır. Bölütlemeyen sonra kamufrajlı kelebek bölgesi tespit edilmiştir. Bölütleme aşamasında kamufraj görüntülerinin dokusal özelliklerini çıkarmak için Gabor filtreleri kullanılmıştır. Gabor filtreleri için birçok parametre bulunmaktadır. Filtrenin sinüzoidal dalgaının açısı olan θ parametresi, filtrenin duyarlı olacağı yönelimi belirler. Bu çalışmada, Gabor Filtre kullanımında, görüntülerin farklı yönelimlerinde kenarları ve dokuları tespit etmek için birden fazla θ değeri seçilmiştir. θ değerleri 0° , 45° , 90° , 135° seçilmiş ve yönelimler eşit aralıklarla belirlenmiştir. σ parametresi hem çözünürlük hem de frekans duyarlılığını dengeleyen 2.0 olarak sabitlenmiştir. γ parametresi

0.5 olarak seçilmiştir. Bu oran kelebek desenlerinin ince ve uzun yapısına göre belirlenmiştir. ψ 0 olarak ayarlanmıştır. Böylece simetrik dokusal özellikler önceliklendirilmektedir.

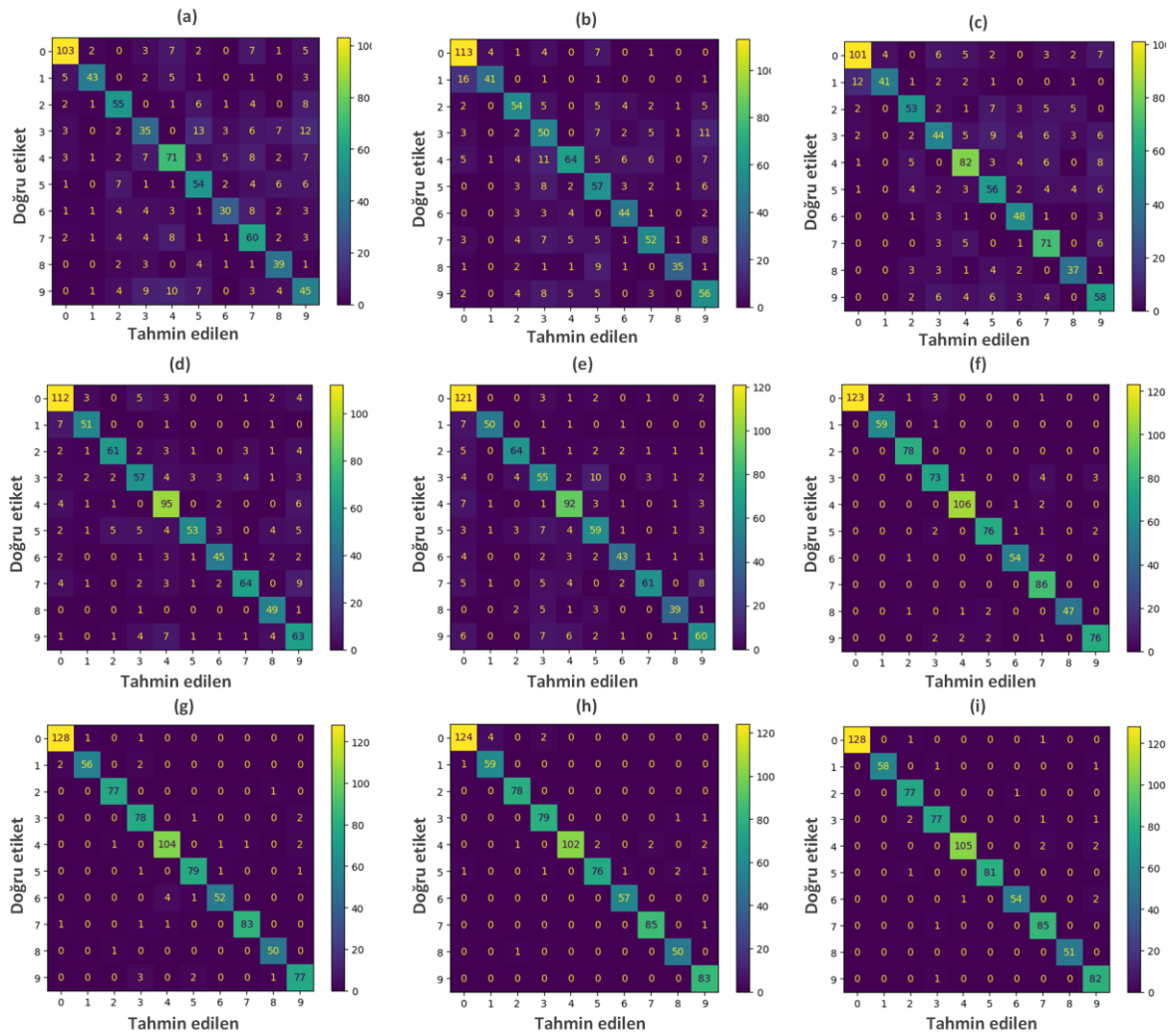
3.2. “Kamuflaj Görüntülerinin Sınıflandırılması İçin Transfer Öğrenme ve Özellik Birleştirme Yaklaşımı” Çalışmasından Elde Edilen Bulgular

Bu çalışmada VGG19, VGG16, Xception, EfficientNetV2, InceptionV3, MobileNet, DenseNet169, DenseNet121, DenseNet201 gibi farklı mimarilere sahip önceden eğitilmiş modeller kullanılmıştır. Çalışmada COD10K kamuflaj veri seti kullanılmıştır. COD10K veri setinden 10 sınıf seçilmiş ve bu çalışmada kullanılmıştır. Bu sınıflardaki eğitim görüntülerine veri artırma yöntemi uygulanarak görüntü sayısı altı kat artırılmıştır. Görüntü çeşitliliğini artırmak derin öğrenme tabanlı sistemlerde çok önemlidir. Veri artırma, aşırı uyumu önler ve görüntü çeşitliliği sağlayarak genelleme performansını artırır. Sınıflandırma, kamuflaj veri seti için dokuz farklı CNN mimarisi kullanılarak elde edilen COD10K veri seti üzerinde transfer öğrenme tarafından gerçekleştirilmiştir. Buradaki amaç, çeşitli farklı mimari tasarımlar kullanılarak optimum sınıflandırma sonuçları elde etmektir. Çok sayıda hiperparametre mimarilerde sınıflandırma doğruluğunun ayarlanmasını önemli ölçüde etkiler.

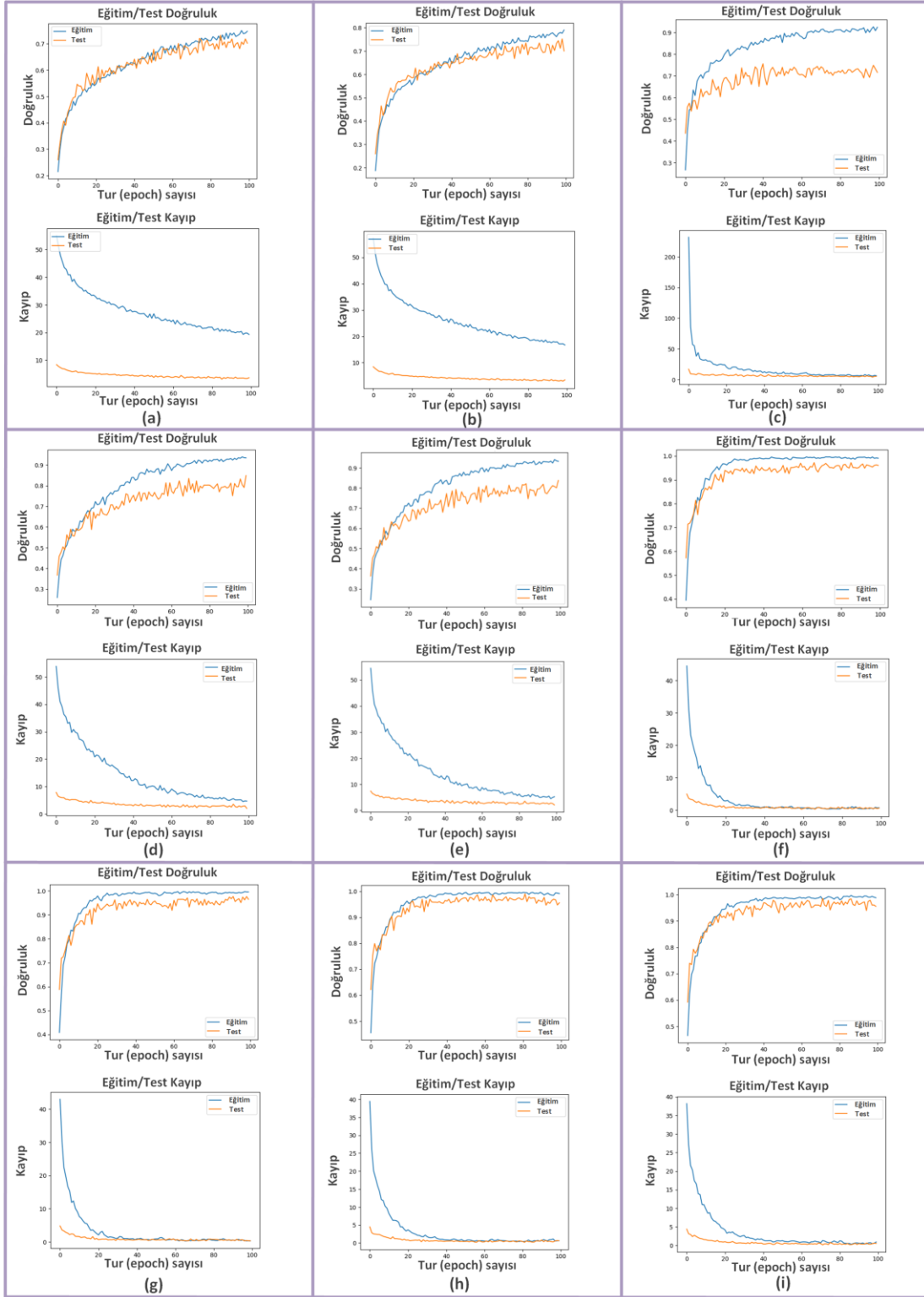
Bu çalışmada model hiperparametreleri üzerinde deneysel bir kurulum gerçekleştirildi. Parametreler modellerin deneysel sonuçlarına göre ayarlandı. Zaman verimliliği ve önemiyle bilinen Adam optimizör, derin ağlar için tercih edildi. Ağırlıklı Kategorik Çapraz Entropi bir kayıp fonksiyonu olarak kullanıldı. Ayrıca, öğrenme oranı 0,00001 olarak yapılandırıldı ve 16'lık parti boyutu kullanıldı. Değerler belirli bir noktada sabitlendiği için, çalışmada kullanılan modeller 100 yineleme için çalıştırıldı. Bu çalışmada kullanılan her mimari için, bir küresel ortalama havuzlama katmanı, ardından bir yoğun katman ve bir bırakma katmanı eklendi. Aşırı uyumu önlemek için 128 ve 512 nöronluk 2 yoğun katman kullanıldı. Bundan sonra, bırakma değerleri sırayla 0,2 ve 0,5 olarak yapılandırılarak bırakma katmanları eklendi. Ayrıca ReLu aktivasyon fonksiyonunu uygulandı ve çıktı katmanında Softmax kullanılmıştır.

Bu çalışmada COD10K veri seti test, doğrulama ve eğitim setleri olmak üzere üçe ayrılmıştır. Veri setinde görüntülerin %80'i eğitim, %10'u doğrulama ve %10'u test için ayrılmıştır. Sonuçların kararlılığını, yani test aşamasındaki rastgelelikten kaynaklanan

kararlılığı sağlamak için bu çalışmada k-fold çapraz doğrulama kullanılmıştır. K değeri 5 olarak belirlenmiştir. Veri seti 5 eşit parçaya bölünmüş ve her parça için ayrı ayrı doğrulama verisi üretilmiştir. Bu şekilde her veri noktası en az bir kez doğrulama verisi olarak kullanılmıştır. Bu şekilde modelimizin performans değerlendirmesi ve doğrulaması yapılmıştır. Çalışmada kullanılan çeşitli tekniklerin performansı F-ölçüsü, Doğruluk, Kesinlik, F1-score ve Recall performans ölçütleri kullanılarak ölçülmüştür. Bu değerler her sınıf için karışıklık matrislerinden yararlanarak hesaplanmıştır. Karışıklık matrisleri sınıflandırma algoritmalarının performansını ölçmek için kullanılmıştır.



Şekil 62. Çalışmada kullanılan ön eğitilmiş modellere göre elde edilen karmaşıklık matrisleri: (a) VGG19, (b) VGG16, (c) Xception, (d) EfficientNetV2, (e) InceptionV3, (f) MobileNet, (g) DenseNet169, (h) DenseNet121, (i) DenseNet201



Şekil 63. Çalışmada kullanılan ön eğitilmiş modellerden elde edilmiş Eğitim / Test Doğruluk ve Eğitim / Test Kayıp Grafikleri: (a) VGG19, (b) VGG16, (c) Xception, (d) EfficientNetV2, (e) InceptionV3, (f) MobileNet, (g) DenseNet169, (h) DenseNet121, (i) DenseNet201

Tablo 8. COD10K Veri Seti Üzerinde Çalışmada Kullanılan Önceden Eğitilmiş Modellerin Performans Karşılaştırması

Model	F-measure	Precision	Recall	F1-score	Accuracy (%)	Ort. bir tur için geçen süre (saniye)
VGG19	0. 6566	0. 6566	0. 6566	0. 6566	65.66	79
VGG16	0. 7132	0. 7155	0. 6882	0. 7015	68.82	77
Xception	0. 7259	0. 7258	0. 7210	0. 7236	72.34	82
EfficientNet V2	0. 8019	0. 8024	0. 7969	0. 7996	79.69	89
InceptionV3	0. 8046	0. 8071	0. 7778	0. 7921	78.82	87
MobileNet	0. 9532	0. 9534	0. 9515	0. 9524	95.23	96
DenseNet169	0. 9597	0. 9599	0. 9571	0. 9584	95.96	97
DenseNet121	0. 9685	0. 9680	0. 9744	0.9711	97.06	102
DenseNet201	0. 9771	0. 9772	0. 9764	0. 9767	97.67	105

Kamuflaj görüntüleri üzerinde sınıflandırma çalışması için her mimarinin performans sonuçları Tablo 8'de gösterilmiştir. Tablo 8'de sunulan sonuçlar analiz edilirse, DenseNet201'in %97,71 F-ölçümü, %97,72 hassasiyet, %97,64 Recall, %97,66 F1-skor ve %97,67 Doğruluk oranlarıyla en yüksek performansı verdiği görülmektedir. Grafikler, verilerin test ve eğitim kümeleri için gösterilmiştir. Bu çalışmada sınıflandırma için kullanılan her mimari için kayıp ve doğruluk grafikleri Şekil 63' de gösterilmiştir. Şekil 63'de, x eksenı turları(epoch) ve y eksenı iyileştirmeyi(düzeltilmeyi) göstermektedir. Grafiklerde gösterilen eğitim eğrisi, modellerin eğitim verilerinden yararlanılarak ne kadar etkili bir şekilde eğitildiğini göstermektedir. Şekil 63'de, kayıp grafikleri eğitim ve test için paralel bir azalma göstermektedir.

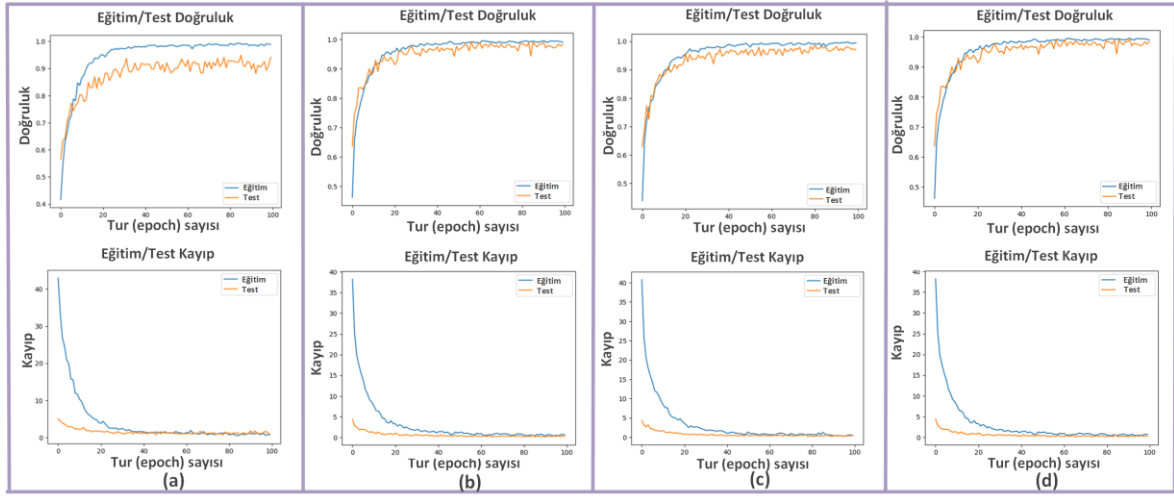
Tablo 8, bu çalışmada kullanılan önceden eğitilmiş modellerin çoğunun kamuflaj görüntülerinin sınıflandırılmasında iyi performans gösterdiğini göstermektedir. Şekil 61'de gösterildiği gibi, gerçek pozitifler tüm sınıflarda hem yanlış negatiflerden hem de yanlış pozitiflerden daha yüksektir. Karmaşıklık matrisi, her model için değerlendirme metrikleri oluşturmak için kullanılır ve bu çalışmadaki deneysel sonuçlara ilişkin performans görselleştirilmesini sağlar. Bu matris, modellerin performansının görsel bir temsilini sağlar. Şekil 62'de sunulan sınıflandırma sonuçlarından türetilen karmaşıklık matrisinde 0:Pipefish,

1:Seahorse, 2:Cat, 3:Lizard, 4:Spider, 5:Bird, 6:Butterfly, 7:Grasshopper, 8:Owl ve 9:Toad sınıflarını temsil etmektedir.

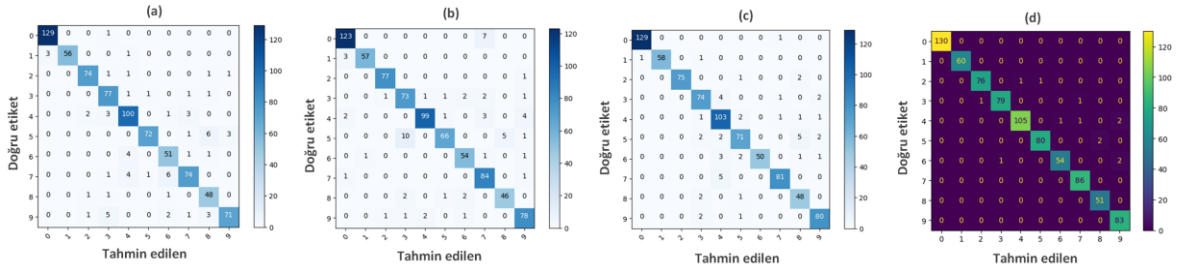
Tablo 8'de gösterildiği gibi, bu çalışmada kullanılan mimariler arasında en başarılı altı sonuç EfficientNetV2, InceptionV3, MobileNet, DenseNet169, DenseNet121, DenseNet201'den elde edilmiştir. Bu nedenle bu mimariler üzerinde hibrit bir yaklaşım kullanılmıştır ve birleştirme tabanlı önceden eğitilmiş yaklaşım önerilmiştir. Bu yaklaşımla önceden eğitilmiş modellerin çıktıları alınıp birleştirilir. Bu işlem katmanları birleştirme, vektörleri birleştirme gibi yöntemler kullanılarak yapılabilir. Daha sonra birleştirilmiş çıktılar üzerinde ek eğitim ya da ince ayarlar yapmaya ihtiyaç duyulabilir. Birleştirme tabanlı önceden eğitilmiş yaklaşım farklı özelliklerin birleştirilmesi için farklı önceden eğitilmiş modellerin birleştirilmesine olanak sağlar. Böylece modelin daha geniş bir bilgiye erişmesini sağlar. Bu da modelin daha iyi performans vermesini sağlar. Çalışmada sınıflandırma için en iyi sonucu veren DenseNet201 modeli ve ondan sonra iyi performans veren diğer modeller kendi aralarında hibrit şekilde birleştirilmiştir. DenseNet201 + MobileNet, InceptionV3+EfficientNetV2+DenseNet201, MobileNet+DenseNet169 ve InceptionV3+DenseNet201 birleştirilmiş mimari üzerinde deneysel çalışmalar yapılmıştır. Bu deneysel çalışmalarda elde edilen birleştirme tabanlı önceden eğitilmiş modellerin performans sonuçları Tablo 9'de gösterilmiştir. Tablo 9 incelendiğinde Şekil 46'da gösterilen çerçevede DenseNet201+ MobileNet yapısının en başarılı sonucu verdiği görülmüştür. DenseNet201 ve MobileNet mimarilerinin birleştirilmesiyle her iki modelin avantajlarından yararlanılmış ve daha kapsamlı özellik çıkarma ve genelleme kabiliyeti sağlanmıştır.

Tablo 9. COD10K Veri Setinde Önceden Eğitilmiş Modellerin Farklı Kombinasyonlarda Birleştirilmesinin Performans Karşılaştırması

Model	F-measure	Precision	Recall	F1-score	Accuracy (%)	Ort. bir tur için geçen süre (saniye)
InceptionV3+DenseNet201	0.9200	0.9200	0.9200	0.9200	92.00	285
MobileNet+DenseNet169	0.9300	0.9300	0.9300	0.9300	93.00	305
InceptionV3+EfficientNetV2+DenseNet201	0.9400	0.9400	0.9400	0.9400	94.00	345
DenseNet201+ MobileNet	0.9828	0.9828	0.9836	0.9831	98.41	314



Şekil 64. Çalışmada tüm birleştirme tabanlı önceden eğitilmiş modellerin Doğruluk grafikleri ve Kayıp grafikleri: (a) InceptionV3 + DenseNet201, (b) MobileNet + DenseNet169, (c) InceptionV3 + EfficientNetV2 + DenseNet201, (d) DenseNet201 + MobileNet



Şekil 65. Çalışmadaki tüm birleştirme tabanlı önceden eğitilmiş sınıflandırma modelleri için karmaşıklık matrisleri: (a) InceptionV3+DenseNet201, (b) MobileNet+DenseNet169, (c) InceptionV3+EfficientNetV2 +DenseNet201, (d) DenseNet201+ MobileNet

Bu çalışmada en iyi sonuçların alındığı modellerin farklı kombinasyonlardaki kayıp ve doğruluk grafikleri Şekil 64'de ve karmaşıklık matrisi Şekil 65'de gösterilmiştir. Çalışmada kullanılan önceden eğitilmiş modellerin mimarilerini birleştirmek için birleştirme tabanlı bir yaklaşımla çeşitli özellikleri birleştirerek daha kapsamlı bir bilgi tabanı oluşturmayı amaçladık. Bu çalışmada, kamuflej görüntülerini sınıflandırmak için dokuz farklı önceden eğitilmiş model kullanıldı ve performans değerlendirmeleri yapıldı. Tablo 8 ve Tablo 9'da verilen sonuçlar, önceden eğitilmiş modeller tek başına kullanıldığında elde edilen sonuçlar ve birleştirilerek elde edilen sonuçlar karşılaştırıldığında, birleştirme tabanlı yaklaşımla çok daha iyi sonuçlar elde edildiği görülmektedir.

Bu çalışmada ayrıca önceden eğitilmiş derin öğrenme modelleri CAMO veri seti üzerinde uygulanarak sınıflandırma işlemi yapılmış ve sınıflandırma başarı oranları Tablo 10'da gösterilmiştir. Bu çalışmada önerdiğimiz birleştirme tabanlı DenseNet201+ MobileNet modeli CAMO veri setinde de en yüksek sınıflandırma başarı oranını vermiştir.

Tablo 10. CAMO Veri Setinde Önceden Eğitilmiş Modeller ile Bu Modellerin Farklı Kombinasyonlarda Birleştirilmesinin Performans Karşılaştırması

Model	F-measure	Precision	Recall	F1-score	Accuracy (%)	Ort. bir tur için geçen süre (saniye)
MobileNet		0.6382	0.6492	0.6267	72.00	95
DenseNet169		0.6442	0.6471	0.6317	73.20	96
DenseNet121		0.6782	0.6911	0.6672	73.60	102
DenseNet201		0.7346	0.6888	0.6803	76.00	98
Önerdiğimiz Model (DenseNet201+ MobileNet)		0.7449	0.7231	0.7187	78.00	215

3.3. “Klasik Makine Öğrenmesi ve Evrimsel Sinir Ağı Yöntemlerini Kullanarak Kamufraj Görüntülerinin Sınıflandırılması İçin Karşılaştırmalı Bir Çalışma” Çalışmasından Elde Edilen Bulgular

Bu çalışmada, COD10K veri setinden seçilen ve kategorize edilen 10 sınıfa ait kamufraj görüntüleri kullanılmıştır. Bu sınıflara ait 1356 ham eğitim görüntüsüne veri artırma yöntemi uygulanmış ve eğitim görüntüsü sayısı 8136'ya çıkarılmıştır. Sınıflandırma başarı oranlarını ölçmek için iki farklı model kurulmuştur. Bunlardan biri Klasik Makine Öğrenmesi (CML) modeli, diğeri ise Evrimsel Sinir Ağı (CNN) modelidir. Bu modeller ile sırasıyla ham ve artırılmış görüntüler kullanılarak Deney 1 ve Deney 2 olmak üzere iki deney gerçekleştirilmiş ve sınıflandırma başarı oranları ölçülmüştür. CML modeli, girdi kamufraj görüntüsünden Yerel İkili Örüntü (LBP) algoritması ile dokusal özellik çıkarımı ve sınıflandırma aşamalarından oluşmaktadır. CNN modelinde ise girdi kamufraj görüntüleri için evrim ve havuzlama katmanları ile özellikler çıkarılmakta ve sınıflandırma işlemi ile görüntünün sınıfı tahmin edilmektedir.

Tablo 11. Deney (1) için Klasik Makine Öğrenmesi ve CNN modellerinin performans karşılaştırması.

Metot	Model	Sınıf Sayısı	Precision	Recall	F1-score	Accuracy (%)
DenseNet201	CNN	10	0.7885	0.7566	0.7722	76.73
ANN	CML	4	0.5852	0.8206	0.6831	73.62
DenseNet121	CNN	10	0.7433	0.7176	0.7302	73.09
SVM	CML	4	0.5657	0.8523	0.6800	72.28
DenseNet169	CNN	10	0.6957	0.6934	0.6945	70.18
KNN	CML	4	0.5407	0.8184	0.6511	68.97
NB	CML	4	0.5094	0.6464	0.5697	65.98
VGG16	CNN	10	0.5441	0.5360	0.5400	55.27
VGG19	CNN	10	0.5135	0.5118	0.5126	52.00
SVM	CML	10	0.2484	0.7547	0.3737	50.55
ANN	CML	10	0.1945	0.6835	0.3028	48.30
InceptionV3+EfficientNetV2	CNN	10	0.4310	0.4405	0.4356	47.20
Xception	CNN	10	0.5170	0.4752	0.4952	47.27
InceptionV3	CNN	10	0.4396	0.4197	0.4294	43.27
KNN	CML	10	0.1648	0.6678	0.2643	42.43
EfficientNetV2	CNN	10	0.4245	0.4197	0.4222	42.25
NB	CML	10	0.1428	0.5714	0.2284	40.22

Tablo 12. Deney (2) için Klasik Makine Öğrenmesi ve CNN modellerinin performans karşılaştırması

Metot	Model	Sınıf Sayısı	Precision	Recall	F1-score	Accuracy (%)
DenseNet201	CNN	10	0.9688	0.9641	0.9664	96.63
DenseNet121	CNN	10	0.9587	0.9569	0.9577	95.95
DenseNet169	CNN	10	0.9570	0.9499	0.9534	95.52
InceptionV3+EfficientNetV2	CNN	10	0.8900	0.9000	0.8949	90.00
InceptionV3	CNN	10	0.7844	0.7778	0.7810	77.81
ANN	CML	4	0.6204	0.8671	0.7232	77.29
SVM	CML	4	0.6161	0.8868	0.7270	76.84
EfficientNetV2	CNN	10	0.7875	0.7576	0.7722	76.74
KNN	CML	4	0.5761	0.8022	0.6706	73.16
Xception	CNN	10	0.7172	0.6953	0.7060	70.63
NB	CML	4	0.5446	0.7395	0.6272	70.51
VGG16	CNN	10	0.7038	0.6966	0.7001	70.08
VGG19	CNN	10	0.6759	0.6719	0.6738	66.95
SVM	CML	10	0.2248	0.8244	0.3532	51.08
ANN	CML	10	0.2138	0.8267	0.3397	49.84
KNN	CML	10	0.1662	0.6028	0.2605	43.49
NB	CML	10	0.1635	0.6500	0.2612	41.56

Bu çalışmada kullanılan tüm CML ve önceden eğitilmiş derin öğrenme modelleri ile elde edilen sınıflandırma başarı oranları Tablo 11'de Deney 1'e ve Tablo 12'de ise Deney 2'ye göre her metrik için verilmiştir. Deney 1'de DenseNet201 %76,73 ile en başarılı model iken, Deney 2'de DenseNet201 %96,63 ile en başarılı model olmuştur. Bu çalışma ile veri artırma teknikleriyle görüntü sayısının artırılmasının kamuflej görüntülerde sınıflandırma başarısını artırdığı tespit edilmiştir. Ayrıca kamuflej görüntülerin sınıflandırılmasında CML yöntemlerinin başarısız olduğu CNN modellerinin oldukça başarılı olduğu gözlemlenmiştir.

3.4. “Kamuflej Görüntülerinin LBP, HOG ve Klasik Makine Öğrenmesi Yöntemleri Kullanılarak Sınıflandırılması” Çalışmasından Elde Edilen Bulgular

Bu çalışmada iki çeşit deney yapılmış ve bu deneylere ait sonuçlar alınmıştır. Bunlardan birincisi COD10K veri setinden seçilmiş 3 sınıfa ait görüntülerin LBP yöntemiyle özellik çıkarımı yapıp CML yöntemleriyle sınıflandırma işleminin gerçekleştirilmesidir. İkinci deney ise yine aynı 3 sınıfa ait görüntülerin HOG yöntemiyle özellik çıkarımı yapılması ve CML yöntemleriyle sınıflandırma işleminin gerçekleştirilmesidir. Her iki deneyde de toplam verinin %70 i eğitim, %20’si test ve %10’u da doğrulama amaçlı kullanılmıştır. Birinci deneyde en yüksek sınıflandırma başarı oranı %83,17 ile SVM yöntemi ile elde edilmiştir. İkinci deneyde de en yüksek sınıflandırma başarı oranı yine SVM yöntemiyle %69,15 olarak elde edilmiştir. Bu çalışmada kullanılan tüm CML yöntemleri ile elde edilen sınıflandırma başarı oranları, her bir ölçüt için Tablo 13’ te gösterilmiştir.

Tablo 13. Çalışmada kullanılan CML, LBP ve HOG yöntemleri ile elde edilen sınıflandırma başarı performansları

Model	Deney No	Precision	Recall	F1-score	Accuracy (%)
HOG + KNN	2	0.3783	0.3255	0.3500	51.40
HOG + ANN	2	0.6035	0.7854	0.6825	68.22
HOG + SVM	2	0.5892	0.7674	0.6666	69.15
LBP + KNN	1	0.7218	0.8321	0.7730	79.90
LBP + ANN	1	0.7539	0.8907	0.8166	82.99
LBP + SVM	1	0.7410	0.8155	0.7764	83.17

3.5. “Önceden Eğitilmiş Evrimsel Sinir Ağı Kullanarak Kamuflej Görüntülerinin Sınıflandırılması” Çalışmasından Elde Edilen Bulgular

Bu çalışmada COD10K veri setinden seçilmiş dört sınıfa ait kamuflej görüntülerinin önceden eğitilmiş derin öğrenme modelleriyle sınıflandırma performansını Precision, Recall, F1-Score ve Accuracy (4) açısından değerlendirilmiştir. Çalışmada iki deney yapılmış ve bu deneylere ait sonuçlar alınmıştır. Bunlardan birincisi COD10K veri setinden seçilmiş dört sınıfa ait toplam 634 ham görüntünün kullanılmasıyla elde edilen sonuçlardır. İkinci deney ise yine aynı dört sınıfa ait görüntülere veri artırımı yöntemi uygulanması

sonucu elde edilen 3804 görüntünün kullanılmasıyla elde edilen sonuçlardır. Her iki deneyde de toplam verinin %10'u doğrulama, %20'si test ve %70' i ise eğitim amaçlı kullanılmıştır. Elde edilen deneysel sonuçlar, her bir ölçüt için Tablo 14' te Deney 1 ve Tablo 15'te Deney 2' ye göre gösterilmiştir.

Tablo 14. Deney (1) için kullanılan CNN modellerinin sınıflandırma performanslarının karşılaştırılması

Model	Precision	Recall	F1-score	Accuracy (%)
DenseNet121	0.8216	0.8344	0.8279	82.95
VGG19	0.7607	0.7272	0.7435	75.97
VGG16	0.7477	0.7236	0.7354	75.19
Xception	0.6587	0.6292	0.6436	67.44
InceptionV3	0.6400	0.5800	0.6085	62.02

Tablo 15. Deney (2) için kullanılan CNN modellerinin sınıflandırma performanslarının karşılaştırılması

Model	Precision	Recall	F1-score	Accuracy (%)
DenseNet121	0.9852	0.9864	0.9858	98.69
VGG16	0.9213	0.9002	0.9106	91.48
InceptionV3	0.8834	0.8970	0.8901	89.25
Xception	0.8540	0.8619	0.8579	85.71
VGG19	0.8437	0.8426	0.8431	62.02

3.6. “LBP Yöntemi Kullanarak Kamuflej Görüntülerinin Sınıflandırılması” Çalışmasından Elde Edilen Bulgular

Bu çalışmada sınıflandırma başarı oranlarının ölçülmesi için hassaslık (Sensitivity - SE), belirleyicilik (Specificity- SP), kesinlik (Precision - PR), duyarlılık (Recall - RE), F1-Skoru (F1 - SC) ve doğruluk (Accuracy - ACC) metrikleri kullanılmıştır. Bu değerlere yönelik bilgiler Tablo 16'da gösterilmiştir.

Tablo 16. CML yöntemlerinin kullanılmasıyla elde edilen sınıflandırma başarı oranları

Model	SE	SP	PR	RE	F1-SC	ACC (%)
ANN	0.8933	0.9411	0.7800	0.8933	0.8328	92.00
SVM	0.7714	0.9199	0.7833	0.7714	0.7773	89.00
KNN	0.8757	0.8789	0.7402	0.8757	0.8022	87.77

3.7. “Kamuflaj Görüntülerde Öklid, Mahalanobis ve K-ortalamlar Kümeleme Algoritması Kullanılarak Görüntü Bölütlemesi” Çalışmasından Elde Edilen Bulgular

Bu çalışmada CAMO-V.1.0 veri setinden seçilen kamuflaj görüntüler kullanılmıştır. K-means algoritması uygulanırken k değeri 5 olarak tanımlanmıştır. Çalışmada elde edilen sonuçlara göre çalışmada kullanılan 30 görüntü için Öklid uzaklık hesabı K-means ile birlikte kullanıldığında ortalama RMSE değeri 262,47 ve Mahalanobis uzaklık hesabı K-means ile birlikte kullanıldığında ise 799,62 olarak hesaplanmıştır. Ayrıca PSNR değerleri 30 resim için Öklid uzaklık hesabı kullanıldığında ortalama PSNR değer 24,26 ve Mahalanobis uzaklık hesabı kullanıldığında ise 19,66 olarak hesaplanmıştır. Deneysel sonuçlara göre Öklid uzaklık hesabının kullanıldığında Mahalanobis uzaklık hesabına göre daha başarılı sonuçlar elde edildiği görülmüştür

3.8. “Yerel İkili Desenlerle Özellik Çıkarımına Dayalı Kelebek Kamuflaj Görüntülerinin Sınıflandırılması” Çalışmasından Elde Edilen Bulgular

Bu çalışmada yapılan kamuflaj kelebek görüntülerinin sınıflandırma başarısını hesaplamak için CML yöntemleri ve LPB algoritması birlikte kullanılmıştır. Değerlendirme ölçütleri olarak, Hassasiyet (SE), Özgüllük (SP), Kesinlik (PR), Duyarlılık (RE), F1-Puanı (F1-SC) ve Doğruluk (ACC) metrikleri kullanılmıştır. Elde edilen sonuçlar Tablo 17’de gösterilmiştir.

Tablo 17. Çalışmada kullanılan CML yöntemlerinin sınıflandırma performansları gösterimi

Model	SE	SP	PR	RE	F1-SC	ACC (%)
SVM	0.9216	0.9383	0.9133	0.9216	0.9174	93.00
ANN	0.8100	0.9500	0.9350	0.8100	0.8680	90.00
KNN	0.7304	0.9060	0.7666	0.7304	0.7480	82.00

4. SONUÇLAR

Kamuflaj görüntülerindeki kamuflajlı nesneler arka plana yakın doku, desen ve renk özelliklerine sahiptir. Bu nedenle arka plandan ayırt edilmeleri çok zordur. Kelebekler, kurbağalar, örümcekler ve kuşlar gibi birçok hayvan doğa tarafından kamufle edilir ve bazen insan gözü için bile kamufle oldukları arka plandan ayırt edilmesi çok zordur.

Bu çalışmada, ERVA 1.0 test görüntülerindeki on türe ait kamuflajlı kelebekler arka planlarından ayrılarak türleri tanımlanmıştır. Dünyada binlerce kelebek türü bulunmaktadır. Bunların çoğu kamuflajlıdır. Farklı türlere ait kamuflajlı kelebekleri arka plandan ayırmak ve türlerini belirlemek, çok benzer doku ve desen özelliklerine sahip oldukları için zorlu bir COD problemidir. Aslanlar, kuşlar ve örümcekler gibi kamuflajlı hayvanlardan oluşan bir veri seti üzerinde çalışırken, bu türleri birbirinden ayırt etmek ve tanımlamak daha kolaydır çünkü çok farklı doku özelliklerine ve şekillere sahiptirler. Bu nedenle, daha zorlu bir problem olduğu için bu tez çalışmasında kelebek türleri üzerinde bir çalışma yapılmıştır.

Bu tez çalışmada en zorlu problemlerden biri ERVA 1.0 veri setinin oluşturulmasıdır. Literatürde kamuflaj veri setlerinde farklı hayvan veya nesne türlerinin görüntüleri olmasına rağmen, bu görüntülerin tür bazında sayısı çok azdır. Bu veri setlerinde yalnızca kamuflaj kelebek görüntülerinin sayısı yok denecek kadar azdır. Bu kadar az görüntüyle, derin öğrenme tekniklerini kullanarak kamuflaj kelebekleri üzerinde sağlıklı sonuçlar elde etmek mümkün değildir. Bu nedenle bir kelebek kamuflaj veri seti oluşturulmuştur. ERVA 1.0 kamuflaj veri setinin oluşturulmasıyla literatürde bir boşluk olan kamuflaj veri seti eksikliği problemine bir çözüm sunulmuştur. Ayrıca bu tez çalışmasında Gabor Filtre, K-means yöntemi, LBP algoritması ve derin öğrenme önceden eğitilmiş modellerle hibrit bir yöntem sunulmuştur. Geleneksel yöntemler ile derin öğrenme tekniklerinin bir araya gelmesiyle oluşan bir hibrit yöntemimiz literatüre büyük katkı sağlayacak ve araştırmacılar için fikir verici bir çalışma olacaktır. Ayrıca bu tez çalışmasında oluşturulan ERVA 1.0 veri setinin yanı sıra literatürde çok kullanılan COD10K, CAMO gibi veri setleri üzerinde farklı çalışmalar yapılmış ve kamuflajın birçok türü için sınıflandırma işlemleri yapılmıştır.

Daha sonraki çalışmalarda, ERVA 1.0 veri seti için hayvan, insan, savaş araçları gibi diğer türlere ait nesnelerin sayısını artıracğız. Bu artan veri seti ile farklı kamufle edilmiş hayvan veya nesne türleri tespit edilecek ve türleri farklı yöntemler denenerek tespit edilecektir. Gelecekteki çalışmalarda, ERVA 1.0 veri setindeki görüntü sayısı daha da artırılacak ve bu veri seti araştırmacıların kullanımına sunulacaktır. Gelecekteki

alışmalarda, Gabor Filtresi ve LBP algoritması dışındaki doku özellikleri yeni yöntemler denenerek çıkarılacak ve yeni yöntemlerden elde edilen özellik vektörleri kullanılarak bölütleme yapılacaktır. Bu alışmada kullanılan derin öğrenme tekniklerinin özelliklerinden esinlenerek, yeni bir hibrit derin öğrenme tekniğı oluşturulacak ve sonuçlar bu yeni teknikle elde edilecektir. Bu alışmada oluşturulan ERVA 1.0 veri seti, hayvanlar alemindeki kamufle edilmiş hayvanlar ve askeri alanlardaki askerlerin ve savaş araçlarının kamufle edilmiş görüntüleri eklenerek en az 20.000 görüntüden oluşan büyük bir veri setine dönüştürülecektir.



5. ÖNERİLER

Doktora tezi olarak sunulan bu çalışmada, kamufle olmuş nesnelerin bölütlenmesine ve tespit edilmesine yönelik hibrit bir yöntem sunulmuştur. Elde edilen bulgular doğrultusunda, gelecek çalışmalar için aşağıdaki öneriler sunulmaktadır:

- **Derin Öğrenme Modellerinin Çeşitlendirilmesi:** Mevcut çalışmada kullanılan derin öğrenme modellerinin yanı sıra, Transformer tabanlı ağlar veya hibrit modellerin entegrasyonu sağlanarak daha yüksek doğruluk oranları elde edilebilir.
- **Kamuflaj Veri Setlerinin Genişletilmesi ve Çeşitlendirilmesi:** Kamuflajlı nesne tespiti için kullanılan veri setleri genişletilerek farklı ortamlardan ve aydınlatma koşullarından alınmış daha çeşitli örnekler içeren veri kümeleri oluşturulabilir. Bu, modelin genelleme yeteneğini artıracaktır.
- **Gelişmiş Ön İşleme Tekniklerinin Kullanımı:** Görüntülerin bölütleme ve nesne tanıma işlemlerinde daha hassas sonuçlar elde edebilmek için kenar belirleme, kontrast artırma ve morfolojik işlemler gibi ön işleme teknikleri optimize edilebilir.
- **Gerçek Zamanlı Uygulamalar İçin Optimizasyon:** Nesne tespiti süreçlerinin gerçek zamanlı sistemlerde uygulanabilmesi için model parametreleri optimize edilmeli ve hesaplama yükü azaltılmalıdır. Mobil veya gömülü sistemlerde çalıştırılabilir hafif ağlar geliştirilebilir.
- **Zayıf Gözetimli ve Kendiliğinden Öğrenen Modellerin Kullanımı:** Tam anlamıyla etiketlenmiş veri setlerine olan bağımlılığı azaltmak adına, zayıf gözetimli veya kendiliğinden öğrenen (self-supervised learning) yöntemlerin uygulanması araştırılabilir.
- **İnsan ve Yapay Zekâ İş Birliği:** İnsan gözlemcilerin ve yapay zekâ destekli sistemlerin birlikte çalışabileceği hibrit çözümler geliştirilerek tespit süreçlerinin doğruluğu artırılabilir.

Bu önerilerin uygulanmasıyla, kamufle olmuş nesnelerin bölütlenmesi ve tespitine yönelik yöntemlerin daha verimli hale getirilmesi ve çeşitli alanlarda (askeri uygulamalar, güvenlik sistemleri, vahşi yaşam gözlemleri vb.) daha etkin kullanılması sağlanabilir.

6. KAYNAKLAR

- Abraham, A. (2005). Artificial neural networks. Handbook of measuring system design.
- Acar, E., & Özerdem, M. S. (2012, April). Image classification of Kiziltepe cropland by using Gabor Wavelet Transform. In 2012 20th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.
- ADAM, B., & SMITH, I. F. (2008). Reinforcement learning for structural control. Journal of Computing in Civil Engineering, 22(2), 133-139.
- Agarap, A. F. (2018). Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375.
- Ahmed, A. S. (2018). Comparative study among Sobel, Prewitt and Canny edge detection operators used in image processing. J. Theor. Appl. Inf. Technol, 96(19), 6517-6525.
- Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. IEEE transactions on pattern analysis and machine intelligence, 28(12), 2037-2041.
- Aloysius, N., & Geetha, M. (2017, April). A review on deep convolutional neural networks. In 2017 international conference on communication and signal processing (ICCSP) (pp. 0588-0592). IEEE.
- Ankile, L. L., Heggland, M. F., & Krange, K. (2020). Deep Convolutional Neural Networks: A survey of the foundations, selected improvements, and some current applications. arXiv preprint arXiv:2011.12960.
- Baatz, M., Hoffmann, C., & Willhauck, G. (2008). Progressing from object-based to object-oriented image analysis. Object-based image analysis: Spatial concepts for knowledge-driven remote sensing applications, 29-42.
- Bayram, E., & Nabiyeve, V. (2020, October). Image segmentation by using K-means clustering algorithm in Euclidean and Mahalanobis distance calculation in camouflage images. In 2020 28th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.
- Bayram, E., & Nabiyeve, V. (2021, April). Classification of Butterfly Camouflage Images Based on Feature Extraction with Local Binary Patterns. In 2nd International Symposium on Applied Sciences and Engineering ISASE2021 At: Ataturk University, Erzurum, TURKEY
- Bayram, E., & Nabiyeve, V. (2021, June). Classification of camouflage images using Local Binary Patterns (LBP). In 2021 29th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.
- Bayram, E., & Nabiyeve, V. V. (2025). A Hybrid Method for Segmentation and Recognition of Butterfly Objects in Camouflage Images. Arabian Journal for Science and Engineering, 1-24.

- Bayram, E., & Nabiyeve, V.V (2024). Classification of camouflage images using lbp, hog and classical machine learning methods. In: 3. Bilisel International Truva Scientific Researches and Innovation Congress, pp. 610–618.
- Bayram, E., & Nabiyeve, V. V. (2024). Önceden Eğitilmiş Evrişimsel Sinir Ağı Kullanarak Kamuflej Görüntülerinin Sınıflandırılması. In: 3rd International Conference on Engineering, Natural and Social Sciences At: Konya, TÜRKİYE
- Bhajantri, N. U., & Nagabhushan, P. (2006, December). Camouflage defect identification: a novel approach. In 9th International Conference on Information Technology (ICIT'06) (pp. 145-148). IEEE.
- Bi, H., Zhang, C., Wang, K., Tong, J., & Zheng, F. (2021). Rethinking camouflaged object detection: Models and datasets. *IEEE transactions on circuits and systems for video technology*, 32(9), 5708-5724.
- Bingol, H. (2022). Classification of OME with Eardrum Otoendoscopic Images Using Hybrid-Based Deep Models, NCA, and Gaussian Method. *Traitement du Signal*, 39(4).
- Bini, S. A. (2018). Artificial intelligence, machine learning, deep learning, and cognitive computing: what do these terms mean and how will they impact health care?. *The Journal of arthroplasty*, 33(8), 2358-2361.
- Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). New York: springer.
- Bozkurt, F., Köse, C., & Sarı, A. (2018). An inverse approach for automatic segmentation of carotid and vertebral arteries in CTA. *Expert Systems with Applications*, 93, 358-375.
- Braithwaite, & Bhanu. (1994, June). Hierarchical Gabor filters for object detection in infrared images. In 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (pp. 628-631). IEEE.
- Buciu, I., & Gacsadi, A. (2009, November). Gabor wavelet based features for medical image analysis and classification. In 2009 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (pp. 1-4). IEEE.
- Calayır, G. N., & Kabak, M. (2021). Analysis of machine learning techniques for maintenance and an application. *Journal of Turkish Operations Management (JTOM)*, 5(1).
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6), 679-698.
- Chen, J., & Ran, X. (2019). Deep learning with edge computing: A review. *Proceedings of the IEEE*, 107(8), 1655-1674.

- Chen, G., Liu, S. J., Sun, Y. J., Ji, G. P., Wu, Y. F., & Zhou, T. (2022). Camouflaged object detection via context-aware cross-level fusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(10), 6981-6993.
- Chen, H. S., Zhu, Y., You, S., Madni, A. M., & Kuo, C. C. J. (2024). GreenCOD: A Green Camouflaged Object Detection Method. *arXiv preprint arXiv:2405.16144*.
- Chen, S. B., Wei, Q. S., Wang, W. Z., Tang, J., Luo, B., & Wang, Z. Y. (2021). Remote sensing scene classification via multi-branch local attention network. *IEEE Transactions on Image Processing*, 31, 99-109.
- Chen, T., Lu, A., Zhu, L., Ding, C., Yu, C., Ji, D., ... & Zang, Y. (2024). Sam2-adapter: Evaluating & adapting segment anything 2 in downstream tasks: Camouflage, shadow, medical image segmentation, and more. *arXiv preprint arXiv:2408.04579*.
- Chen, Y. T., Shi, J., Ye, Z., Mertz, C., Ramanan, D., & Kong, S. (2022, October). Multimodal object detection via probabilistic ensembling. In *European Conference on Computer Vision* (pp. 139-158). Cham: Springer Nature Switzerland.
- Chen, Z., Zhang, X., Xiang, T. Z., & Tai, Y. (2024). Adaptive guidance learning for camouflaged object detection. *arXiv preprint arXiv:2405.02824*.
- Cheng, S., Ji, G. P., Qin, P., Fan, D. P., Zhou, B., & Xu, P. (2023). Large model based referring camouflaged object detection. *arXiv preprint arXiv:2311.17122*.
- Cheng, X., Xiong, H., Fan, D. P., Zhong, Y., Harandi, M., Drummond, T., & Ge, Z. (2022). Implicit motion handling for video camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 13864-13873).
- Chennamsetty, P. R., Avula, G., & Chunduri buchhi, R. (2020). Camouflage detection with texture statistical characterization in autonomous systems. *International Journal of Intelligent Unmanned Systems*, 9(1), 17-27.
- Chou, M. C., Chen, H. J., & Shuai, H. H. (2022, July). Finding the achilles heel: Progressive identification network for camouflaged object detection. In *2022 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1-6). IEEE.
- Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1251-1258).
- Clark, M., Bovik, A. C., & Geisler, W. S. (1987). Texture segmentation using Gabor modulation/demodulation. *Pattern Recognition Letters*, 6(4), 261-267.
- Cong, R., Sun, M., Zhang, S., Zhou, X., Zhang, W., & Zhao, Y. (2023, October). Frequency perception network for camouflaged object detection. In *Proceedings of the 31st ACM International Conference on Multimedia* (pp. 1179-1189).
- David, P. C., Smokelin, J., & Ye, A. (1992). Wavelet and Gabor transforms for detection. *Optical Engineering*, 31(9), 1893-1898.

- Deng, C., Ma, W., & Yin, Y. (2011, October). An edge detection approach of image fusion based on improved Sobel operator. In 2011 4th International Congress on Image and Signal Processing (Vol. 3, pp. 1189-1193). IEEE.
- De Valois, R. L., Albrecht, D. G., & Thorell, L. G. (1982). Spatial frequency selectivity of cells in macaque visual cortex. *Vision research*, 22(5), 545-559.
- Dogan, R. O., Dogan, H., Bayrak, C., & Kayikcioglu, T. (2021). A two-phase approach using mask R-CNN and 3D U-Net for high-accuracy automatic segmentation of pancreas in CT imaging. *Computer Methods and Programs in Biomedicine*, 207, 106141.
- Dong, S., Wang, P., & Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review*, 40, 100379.
- Donga, B., Zhugeb, M., Wangc, Y., Bid, H., & Chene, G. (2021). Towards Accurate Camouflaged Object Detection with Mixture Convolution and Interactive Fusion. *arXiv preprint arXiv:2101.05687*.
- Jahnke, P. (2015). Machine learning approaches for failure type detection and predictive maintenance. *Technische Universität Darmstadt*, 19.
- Jain, A. K., Ratha, N. K., & Lakshmanan, S. (1997). Object detection using Gabor filters. *Pattern recognition*, 30(2), 295-309.
- Fan, D. P., Cheng, M. M., Liu, Y., Li, T., & Borji, A. (2017). Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE international conference on computer vision* (pp. 4548-4557).
- Fan, D. P., Gong, C., Cao, Y., Ren, B., Cheng, M. M., & Borji, A. (2018). Enhanced-alignment measure for binary foreground map evaluation. *arXiv preprint arXiv:1805.10421*.
- Fan, D. P., Ji, G. P., Sun, G., Cheng, M. M., Shen, J., & Shao, L. (2020). Camouflaged object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2777-2787).
- Ge, S., Jin, X., Ye, Q., Luo, Z., & Li, Q. (2018). Image editing by object-aware optimal boundary searching and mixed-domain composition. *Computational Visual Media*, 4, 71-82.
- Ghazanfar, M., Malik, M. F., Hussain, M., Iqbal, R., & Younas, M. (2016). Butterflies and their contribution in ecosystem: A review. *Journal of Entomology and Zoology Studies*, 4(2), 115-118.
- Goodfellow, I. (2016). *Deep learning*.
- Gopalakrishnan, K., Khaitan, S. K., Choudhary, A., & Agrawal, A. (2017). Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection. *Construction and building materials*, 157, 322-330.

- Gu, J., Li, C., Liang, Y., Shi, Z., & Song, Z. (2024). Exploring the frontiers of softmax: Provable optimization, applications in diffusion model, and beyond. arXiv preprint arXiv:2405.03251.
- Hamamoto, Y., Uchimura, S., Watanabe, M., Yasuda, T., Mitani, Y., & Tomita, S. (1998). A Gabor filter-based method for recognizing handwritten numerals. *Pattern recognition*, 31(4), 395-400.
- He, C., Li, K., Zhang, Y., Tang, L., Zhang, Y., Guo, Z., & Li, X. (2023). Camouflaged object detection with feature decomposition and edge reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 22046-22055).
- He, C., Li, K., Zhang, Y., Tang, L., Zhang, Y., Guo, Z., & Li, X. (2023). Camouflaged object detection with feature decomposition and edge reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 22046-22055).
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961-2969).
- Htoo, C. K., & Sein, M. M. (2018). Enhanced Texture Representation for Moving Targets Classification Using Co-occurrence. In *Multi-disciplinary Trends in Artificial Intelligence: 12th International Conference, MIWAI 2018, Hanoi, Vietnam, November 18–20, 2018, Proceedings 12* (pp. 240-246). Springer International Publishing.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). MobileNets: efficient convolutional neural networks for mobile vision applications (2017). arXiv preprint arXiv:1704.04861, 126.
- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).
- Huang, K., Li, C., Zhang, J., & Wang, B. (2021). Cascade and fusion: A deep learning approach for camouflaged object sensing. *Sensors*, 21(16), 5455.
- Hu, X., Zhang, P., & Xiao, Y. (2019, July). Military object detection based on optimal gabor filtering and deep feature pyramid network. In *Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science* (pp. 524-530).
- Huang, Z., Huang, L., Gong, Y., Huang, C., & Wang, X. (2019). Mask scoring r-cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 6409-6418).
- Hui, W., Zhu, Z., Gu, G., Liu, M., & Zhao, Y. (2024). Implicit-explicit Motion Learning for Video Camouflaged Object Detection. *IEEE Transactions on Multimedia*.
- Jain, A. K., Ratha, N. K., & Lakshmanan, S. (1997). Object detection using Gabor filters. *Pattern recognition*, 30(2), 295-309.

- Jap, D., Won, Y. S., & Bhasin, S. (2021, May). Fault injection attacks on SoftMax function in deep neural networks. In *Proceedings of the 18th ACM International Conference on Computing Frontiers* (pp. 238-240).
- Jasim, W. A. N., & Mohammed, R. J. (2021). A Survey on Segmentation Techniques for Image Processing. *Iraqi Journal for Electrical & Electronic Engineering*, 17(2).
- Jena, K. K., & Mishra, S. (2015). A Programmatic Combination of SOBEL & PREWITT Operators for Edge Detection of Images. *Journal of Image Processing and Artificial Intelligence*, 1(2).
- Ji, Q., Huang, J., He, W., & Sun, Y. (2019). Optimized deep convolutional neural networks for identification of macular diseases from optical coherence tomography images. *Algorithms*, 12(3), 51.
- Ji, G. P., Zhu, L., Zhuge, M., & Fu, K. (2022). Fast camouflaged object detection via edge-based reversible re-calibration network. *Pattern Recognition*, 123, 108414.
- Jia, Q., Yao, S., Liu, Y., Fan, X., Liu, R., & Luo, Z. (2022). Segment, magnify and reiterate: Detecting camouflaged objects the hard way. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4713-4722).
- Jia, Q., Yao, S., Xu, Y., Liu, Y., Kong, D., & Latecki, L. J. (2024, July). Fuzzy Boundary-Guided Network for Camouflaged Object Detection. In *2024 IEEE International Conference on Multimedia and Expo (ICME)* (pp. 1-6). IEEE.
- Jiang, X., Cai, W., Ding, Y., Wang, X., Yang, Z., Di, X., & Gao, W. (2023). Camouflaged object detection based on ternary cascade perception. *Remote Sensing*, 15(5), 1188.
- Jiang, X., Cai, W., Ding, Y., Wang, X., Hong, D., Yang, Z., & Gao, W. (2023). Camouflaged object segmentation based on joint salient object for contrastive learning. *IEEE Transactions on Instrumentation and Measurement*.
- Jiang, Y., & Zhou, Z. H. (2004, August). Editing training data for k NN classifiers with neural network ensemble. In *International symposium on neural networks* (pp. 356-361). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Khair, K. B., Jahir, S., Ibrahim, M., & Karmaker, D. (2023, January). A Comprehensive Study of Camouflaged Object Detection Using Deep Learning. In *2023 3rd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)* (pp. 28-33). IEEE.
- Kajiura, N., Liu, H., & Satoh, S. I. (2021, December). Improving camouflaged object detection with the uncertainty of pseudo-edge labels. In *Proceedings of the 3rd ACM International Conference on Multimedia in Asia* (pp. 1-7).
- Kamran, M., Rehman, S. U., Meraj, T., Alnowibet, K. A., & Rauf, H. T. (2022). Camouflage object segmentation using an optimized deep-learning approach. *Mathematics*, 10(22), 4219.

- Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, 24(7), 881-892.
- Kaya, Z., Aslan, Z., Güneş, A., & Okatan, A. (2024). Dalgacık dönüşümü ve Inception V3 evrişimli sinir ağı modeli kullanarak beyin tümörlerinin. *Journal of the Faculty of Engineering & Architecture of Gazi University/Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 39(3).
- Kazak, N. (2016). Performance analysis of spiral neighbourhood topology based local binary patterns in texture recognition. *International Journal of Applied Mathematics Electronics and Computers*, (Special Issue-1), 338-341.
- Kolar, Z., Chen, H., & Luo, X. (2018). Transfer learning and deep convolutional neural networks for safety guardrail detection in 2D images. *Automation in Construction*, 89, 58-70.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- Kruse, R., Borgelt, C., Braune, C., Mostaghim, S., Steinbrecher, M., Kruse, R., ... & Steinbrecher, M. (2016). Introduction to neural networks. *Computational Intelligence: A Methodological Introduction*, 9-13.
- Kumar, P. R., & Vijaya, A. (2022). Naïve Bayes machine learning model for image classification to assess the level of deformation of thin components. *Materials Today: Proceedings*, 68, 2265-2274.
- Lamdouar, H., Yang, C., Xie, W., & Zisserman, A. (2020). Betrayed by motion: Camouflaged object discovery via motion segmentation. In *Proceedings of the Asian conference on computer vision*.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- Le, H. T., Phung, S. L., Chapple, P. B., Bouzerdoun, A., & Ritz, C. H. (2020). Deep gabor neural network for automatic detection of mine-like objects in sonar imagery. *IEEE Access*, 8, 94126-94139.
- Le, T. N., Nguyen, T. V., Nie, Z., Tran, M. T., & Sugimoto, A. (2019). Anabranh network for camouflaged object segmentation. *Computer vision and image understanding*, 184, 45-56.
- Le, T. N., Cao, Y., Nguyen, T. C., Le, M. Q., Nguyen, K. D., Do, T. T., ... & Nguyen, T. V. (2021). Camouflaged instance segmentation in-the-wild: Dataset, method, and benchmark suite. *IEEE Transactions on Image Processing*, 31, 287-300.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.

- Leonardo, M. M., Carvalho, T. J., Rezende, E., Zucchi, R., & Faria, F. A. (2018, October). Deep feature-based classifiers for fruit fly identification (Diptera: Tephritidae). In 2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI) (pp. 41-47). IEEE.
- Li, J. (2011). Detection of the Mobile Object with Camouflage Color Under Dynamic Background Based on Optical Flow. *Procedia Engineering*, 15, 2201-2205. <https://doi.org/10.1016/J.PROENG.2011.08.412>.
- Li, H., Feng, C. M., Xu, Y., Zhou, T., Yao, L., & Chang, X. (2023). Zero-shot camouflaged object detection. *IEEE Transactions on Image Processing*.
- Li, P., Yan, X., Zhu, H., Wei, M., Zhang, X. P., & Qin, J. (2022). Findnet: Can you find me? boundary-and-texture enhancement network for camouflaged object detection. *IEEE Transactions on Image Processing*, 31, 6396-6411.
- Li, S., Li, Z., Li, B., Li, X., & Sheng, J. (2024, June). Boundary-Guided Fusion of Multi-Level Features Network for Camouflaged Object Detection. In 2024 International Joint Conference on Neural Networks (IJCNN) (pp. 1-7). IEEE.
- Li, S., Li, X., Li, Z., Ma, H., Sheng, J., & Li, B. (2024, July). Dual Guidance Enhancing Camouflaged Object Detection via Focusing Boundary and Localization Representation. In 2024 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1-6). IEEE.
- Li, X., Li, L., Jiang, S., Yang, M., & Qi, L. (2022, July). Camouflaged object detection with discriminative information attention and cross-level feature fusion. In 2022 7th International Conference on Image, Vision and Computing (ICIVC) (pp. 248-255). IEEE.
- Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12), 6999-7019.
- Lian, H. C., & Lu, B. L. (2006, May). Multi-view gender classification using local binary patterns and support vector machines. In *International Symposium on Neural Networks* (pp. 202-209). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Liang, W., Wu, J., Wu, Y., Mu, X., & Xu, J. (2024). FINet: Frequency Injection Network for Lightweight Camouflaged Object Detection. *IEEE Signal Processing Letters*.
- Liang, Y., Qin, G., Sun, M., Wang, X., Yan, J., & Zhang, Z. (2024). A systematic review of image-level camouflaged object detection with deep learning. *Neurocomputing*, 566, 127050.
- Lin, J., Tan, X., Xu, K., Ma, L., & Lau, R. W. (2023). Frequency-aware camouflaged object detection. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(2), 1-16.

- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125).
- Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13* (pp. 740-755). Springer International Publishing.
- Lin, J., Tan, X., Xu, K., Ma, L., & Lau, R. W. (2023). Frequency-aware camouflaged object detection. *ACM Transactions on Multimedia Computing, Communications and Applications*, 19(2), 1-16.
- Liu, C., Wu, S., Wu, S., Wang, Z., & Xiao, K. (2019, July). Reliable Automatic Organ Segmentation from CT Images using Deep CNN. In *2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C)* (pp. 368-374). IEEE.
- Liu, H., & Zhang, S. (2012). Noisy data elimination using mutual k-nearest neighbor for classification mining. *Journal of Systems and Software*, 85(5), 1067-1074.
- Liu, K., Qiu, T., Yu, Y., Li, S., & Li, X. (2023). Edge-guided camouflaged object detection via multi-level feature integration. *Sensors*, 23(13), 5789.
- Liu, Z., Jiang, P., Lin, L., & Deng, X. (2024, April). Edge Attention Learning for Efficient Camouflaged Object Detection. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 5230-5234). IEEE.
- Liu, K., Qiu, T., Yu, Y., Li, S., & Li, X. (2023). Edge-guided camouflaged object detection via multi-level feature integration. *Sensors*, 23(13), 5789.
- Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International journal of Remote sensing*, 28(5), 823-870.
- Lv, Y., Zhang, J., Dai, Y., Li, A., Liu, B., Barnes, N., & Fan, D. P. (2021). Simultaneously localize, segment and rank the camouflaged objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11591-11601).
- Lv, Y., Zhang, J., Dai, Y., Li, A., Barnes, N., & Fan, D. P. (2023). Toward Deeper Understanding of Camouflaged Object Detection. *IEEE transactions on circuits and systems for video technology*, 33(7), 3462-3476.
- Ma, Q. D., Ma, Z., Ji, C., Yin, K., Zhu, T., & Bian, C. (2017, October). Artificial object edge detection based on enhanced Canny algorithm for high-speed railway apparatus identification. In *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)* (pp. 1-6). IEEE.

- Mastromichalakis, S. (2020). ALReLU: A different approach on Leaky ReLU activation function to improve Neural Networks Performance. arXiv preprint arXiv:2012.07564.
- McCarthy, J., Minsky, M. L., Rochester, N., & Shannon, C. E. (2006). A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine*, 27(4), 12-12.
- McDermott, J. (2021). Hands-on Transfer Learning with Keras and the VGG16 Model. *Learndatasci*. <https://www.learndatasci.com/tutorials/hands-on-transfer-learning-keras/> [Erişim Tarihi: 09 Şubat 2025].
- Mei, H., Ji, G. P., Wei, Z., Yang, X., Wei, X., & Fan, D. P. (2021). Camouflaged object segmentation with distraction mining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8772-8781).
- Miao, Y. E. (2011). Image texture detection based on parallel gray level grade co-occurrence matrix. *Laser Infrared*, 41, 1287-1291.
- Mitchell, T. M. (1997). *Machine learning* (Vol. 1, No. 9). New York: McGraw-hill.
- Mondal, A. (2022). Camouflage design, assessment and breaking techniques: a survey. *Multimedia Systems*, 28(1), 141-160.
- Maguolo, G., Nanni, L., & Ghidoni, S. (2021). Ensemble of convolutional neural networks trained with different activation functions. *Expert Systems with Applications*, 166, 114048.
- Nabiyev, V. (2021). *Yapay Zekâ: Makine Öğrenmesi, Stratejili Oyunlar, Örüntü Tanıma, Doğal Dil İşleme* (6. baskı). Seçkin Yayıncılık.
- Nabiyev, V. V., & Günay, A. (2010). LBP yardımıyla görüntüdeki kişinin yaşının bulunması. *Çankaya University Journal of Science and Engineering*, 8(1).
- Namuduri, K. R., Mehrotra, R., & Ranganathan, N. (1994). Efficient computation of Gabor filter based multiresolution responses. *Pattern recognition*, 27(7), 925-938.
- Nanni, L., Fusaro, D., Fantozzi, C., & Pretto, A. (2023). Improving existing segmentators performance with zero-shot segmentators. *Entropy*, 25(11), 1502.
- Nasr, M. A. S., AlRahmawy, M. F., & Tolba, A. S. (2017). Multi-scale structural similarity index for motion detection. *Journal of King Saud University-Computer and Information Sciences*, 29(3), 399-409.
- Nguyen, T. H., Nguyen, T. N., & Ngo, B. V. (2022). A VGG-19 model with transfer learning and image segmentation for classification of tomato leaf disease. *AgriEngineering*, 4(4), 871-887.
- Nguyen, D. T., Zong, Z., Ogunbona, P., & Li, W. (2010, September). Object detection using non-redundant local binary patterns. In *2010 IEEE international conference on image processing* (pp. 4609-4612). IEEE.

- Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1), 51-59.
- Özgür, S. N., & Keser, S. B. (2021). Meme kanseri tümörlerinin derin öğrenme algoritmaları ile sınıflandırılması. *Türk Doğa ve Fen Dergisi*, 10(2), 212-222.
- Pan, Y., Chen, Y., Fu, Q., Zhang, P., & Xu, X. (2011). Study on the Camouflaged Target Detection Method Based on 3D Convexity. *Mathematical Models and Methods in Applied Sciences*, 5, 152. <https://doi.org/10.5539/MAS.V5N4P152>.
- Pang, Y., Zhao, X., Xiang, T. Z., Zhang, L., & Lu, H. (2022). Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition* (pp. 2160-2170).
- Pare, S., Bhandari, A. K., Kumar, A., & Singh, G. K. (2017). An optimal color image multilevel thresholding technique using grey-level co-occurrence matrix. *Expert Systems with Applications*, 87, 335-362.
- Petránek, K., Janecka, P., & Vaněk, J. (2015). Using local binary patterns for object detection in images. *Global Journal of Computer Science*, 5(1), 07-12. <https://doi.org/10.18844/GJCS.V5I1.24>
- Popa, M. V., Bechet, P., & Demeter, S. (2005, November). Automatic detection of targets using gabor filters and neural networks. In *Proceedings of the 2005 WSEAS International Conference on Dynamical Systems and Control*, ser. CONTROL'05 (pp. 56-61).
- Porat, M., & Zeevi, Y. Y. (1988). The generalized Gabor scheme of image representation in biological and machine vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4), 452-468.
- Prakasa, E. (2016). Texture feature extraction by using local binary pattern. *INKOM Journal*, 9(2), 45-48.
- Putzu, L., Piras, L., & Giacinto, G. (2020). Convolutional neural networks for relevance feedback in content based image retrieval: A Content based image retrieval system that exploits convolutional neural networks both for feature extraction and for relevance feedback. *Multimedia Tools and Applications*, 79(37), 26995-27021.
- Qi, Z., Tian, Y., & Shi, Y. (2013). Robust twin support vector machine for pattern classification. *Pattern recognition*, 46(1), 305-316.
- Qian, S., Liu, H., Liu, C., Wu, S., & San Wong, H. (2018). Adaptive activation functions in convolutional neural networks. *Neurocomputing*, 272, 204-212.
- Qian, Z., Hayes, T. L., Kafle, K., & Kanan, C. (2020). Do we need fully connected output layers in convolutional networks?. *arXiv preprint arXiv:2004.13587*.
- Qiu, S., & Cai, B. (2017). Flexible rectified linear units for improving convolutional neural networks. *arXiv preprint arXiv:1706.08098*.

- Raghu, P. P., & Yegnanarayana, B. (1996). Segmentation of Gabor-filtered textures using deterministic relaxation. *IEEE Transactions on Image processing*, 5(12), 1625-1636.
- Raina, V., Molchanova, N., Graziani, M., Malinin, A., Muller, H., Cuadra, M. B., & Gales, M. (2023, April). Tackling bias in the dice similarity coefficient: introducing NDSC for white matter lesion segmentation. In *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)* (pp. 1-5). IEEE.
- Rao, C. P., Reddy, A. G., & Rama Rao, C. B. (2016). Target detection using multi-resolution analysis for camouflaged images. *International Journal on Cybernetics & Informatics*, 5(4), 135-147.
- Rawat, W., & Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9), 2352-2449.
- Ren, J., Hu, X., Zhu, L., Xu, X., Xu, Y., Wang, W., ... & Heng, P. A. (2021). Deep texture-aware features for camouflaged object detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(3), 1157-1167.
- Rezayi, S., Mohammadzadeh, N., Bouraghi, H., Saeedi, S., & Mohammadpour, A. (2021). Timely diagnosis of acute lymphoblastic leukemia using artificial intelligence-oriented deep learning methods. *Computational Intelligence and Neuroscience*, 2021(1), 5478157.
- Sae-Lim, W., Wettayaprasit, W., & Aiyarak, P. (2019, July). Convolutional neural networks using MobileNet for skin lesion classification. In *2019 16th international joint conference on computer science and software engineering (JCSSE)* (pp. 242-247). IEEE.
- Salari, A., Djavadifar, A., Liu, X., & Najjaran, H. (2022). Object recognition datasets and challenges: A review. *Neurocomputing*, 495, 129-152.
- Sarker, L., Islam, M. M., Hannan, T., & Ahmed, Z. (2020). COVID-DenseNet: A deep learning architecture to detect COVID-19 from chest radiology images. preprint, 2020050151(10.20944).
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
- Shan, C., Gong, S., & McOwan, P. W. (2009). Facial expression recognition based on local binary patterns: A comprehensive study. *Image and vision Computing*, 27(6), 803-816.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv 2014. arXiv preprint arXiv:1409.1556*, 1409.
- Singh, K. K., Siddhartha, M., & Singh, A. (2020). Diagnosis of coronavirus disease (COVID-19) from chest X-ray images using modified XceptionNet. *Romanian Journal of Information Science and Technology*, 23(657), 91-115.

- Skurowski, P., Abdulameer, H., Baszczyk, J., Depta, T., Kornacki, A. & Kozie, P. (2018). Animal camouflage analysis: chameleon database, Unpublished Manuscript.
- Song, Y., Luo, Y., Chang, C., Liu, J., Shi, M., & Zhai, J. (2023, November). Recognition of Low-Altitude Targets Using Improved Canny Edge Detection. In 2023 5th International Conference on Frontiers Technology of Information and Computer (ICFTIC) (pp. 319-324). IEEE.
- Stevens, M., & Merilaita, S. (2009). Animal camouflage: current issues and new perspectives. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1516), 423-427.
- Stockman, G., & Shapiro, L. G. (2001). *Computer vision*. Prentice Hall PTR.
- Subasi, A., & Gursoy, M. I. (2010). EEG signal classification using PCA, ICA, LDA and support vector machines. *Expert systems with applications*, 37(12), 8659-8666.
- Sun, D., Jiang, S., & Qi, L. (2023, July). Edge-aware mirror network for camouflaged object detection. In 2023 IEEE International Conference on Multimedia and Expo (ICME) (pp. 2465-2470). IEEE.
- Sun, W., Liu, C., Zhang, L., Li, Y., Wei, P., Liu, C., ... & Ye, Q. (2022). Dqnet: Cross-model detail querying for camouflaged object detection. *arXiv preprint arXiv:2212.08296*.
- Sun, Y., Wang, S., Chen, C., & Xiang, T. Z. (2022). Boundary-guided camouflaged object detection. *arXiv preprint arXiv:2207.00794*.
- Sun, Y., Xu, C., Yang, J., Xuan, H., & Luo, L. (2025). Frequency-spatial entanglement learning for camouflaged object detection. In *European Conference on Computer Vision* (pp. 343-360). Springer, Cham.
- Sutton, R. S. (2018). *Reinforcement learning: An introduction*. A Bradford Book.
- Suzuki, T. K., Tomita, S., & Sezutsu, H. (2019). Multicomponent structures in camouflage and mimicry in butterfly wing patterns. *Journal of Morphology*, 280(1), 149-166.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).
- Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2017, February). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 31, No. 1).
- Tang, L., Jiang, P. T., Shen, Z. H., Zhang, H., Chen, J. W., & Li, B. (2024, October). Chain of visual perception: Harnessing multimodal large language models for zero-shot camouflaged object detection. In *Proceedings of the 32nd ACM International Conference on Multimedia* (pp. 8805-8814).
- Tankus, A., & Yeshurun, Y. (2001). Convexity-based visual camouflage breaking. *Computer Vision and Image Understanding*, 82(3), 208-237.

- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433–460. <https://doi.org/10.1093/mind/LIX.236.433>
- Rawat, W., & Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9), 2352-2449.
- Russell, S., & Norvig, P. (2021). Artificial intelligence: a modern approach, 4th US ed. aima: сайт. URL: <https://aima.cs.berkeley.edu/> (дата обращения: 26.02. 2023).
- Sun, M., Song, Z., Jiang, X., Pan, J., & Pang, Y. (2017). Learning pooling for convolutional neural network. *Neurocomputing*, 224, 96-104.
- Sutton, R. S. (2018). Reinforcement learning: An introduction. A Bradford Book.
- Van Melle, W. (1978). MYCIN: a knowledge-based consultation program for infectious disease diagnosis. *International journal of man-machine studies*, 10(3), 313-322.
- Vargas, V. M., Gutiérrez, P. A., Barbero-Gómez, J., & Hervás-Martínez, C. (2021). Activation functions for convolutional neural networks: Proposals and experimental study. *IEEE Transactions on Neural Networks and Learning Systems*, 34(3), 1478-1488.
- Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., ... & Tang, X. (2017). Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3156-3164).
- Wang, J., Hong, M., Hu, X., Li, X., Huang, S., Wang, R., & Zhang, F. (2023). Camouflaged insect segmentation using a progressive refinement network. *Electronics*, 12(4), 804.
- Wang, K., Bi, H., Zhang, Y., Zhang, C., Liu, Z., & Zheng, S. (2021). D² C-Net: A Dual-Branch, Dual-Guidance and Cross-Refine Network for Camouflaged Object Detection. *IEEE Transactions on Industrial Electronics*, 69(5), 5364-5374.
- Wang, Q., Yang, J., Yu, X., Wang, F., Chen, P., & Zheng, F. (2023, October). Depth-aided camouflaged object detection. In *Proceedings of the 31st ACM International Conference on Multimedia* (pp. 3297-3306).
- Wang, Y., Li, L., Yang, X., Wang, X., & Liu, H. (2020, March). A camouflaged object detection model based on deep learning. In *2020 IEEE International Conference on Artificial Intelligence and Information Systems (ICAIS)* (pp. 150-153). IEEE.
- Wang, Z. (2008). An automatic region-based image segmentation system for remote sensing applications.
- Wanto, A., Rizki, S. D., Andini, S., Surmayanti, S., Ginantra, N. L. W. S. R., & Aspan, H. (2021, June). Combination of Sobel+ Prewitt Edge Detection Method with Roberts+ Canny on Passion Flower Image Identification. In *Journal of Physics: Conference Series* (Vol. 1933, No. 1, p. 012037). IOP Publishing.
- Widodo, A., & Yang, B. S. (2007). Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical systems and signal processing*, 21(6), 2560-2574.

- Woergoetter, F. & Porr, B. (2008). Reinforcement learning. Scholarpedia. http://www.scholarpedia.org/article/Reinforcement_learning [Erişim Tarihi: 05 Şubat 2025].
- Wu, J., Liang, W., Hao, F., & Xu, J. (2023). Mask-and-edge co-guided separable network for camouflaged object detection. *IEEE Signal Processing Letters*, 30, 748-752.
- Xia, H., Zhou, Y., & Zhang, Z. (2022). Auto insurance fraud identification based on a CNN-LSTM fusion deep learning model. *International Journal of Ad Hoc and Ubiquitous Computing*, 39(1-2), 37-45.
- Xiang, M., Zhang, J., Lv, Y., Li, A., Zhong, Y., & Dai, Y. (2021). Exploring depth contribution for camouflaged object detection. *arXiv preprint arXiv:2106.13217*.
- Xie, C., Xia, C., Yu, T., & Li, J. (2023, October). Frequency representation integration for camouflaged object detection. In *Proceedings of the 31st ACM International Conference on Multimedia* (pp. 1789-1797).
- Xu, J., Li, Z., Du, B., Zhang, M., & Liu, J. (2020, July). Reluplex made more practical: Leaky ReLU. In *2020 IEEE Symposium on Computers and communications (ISCC)* (pp. 1-7). IEEE.
- Xu, X., Zhu, M., Yu, J., Chen, S., Hu, X., & Yang, Y. (2021). Boundary guidance network for camouflage object detection. *Image and Vision Computing*, 114, 104283.
- Yadav, J. S., & Bharathi, P. S. (2022, April). Edge detection of images using Prewitt algorithm comparing with Sobel algorithm to improve accuracy. In *2022 3rd International Conference on Intelligent Engineering and Management (ICIEM)* (pp. 351-355). IEEE.
- Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, 9, 611-629.
- Yan, X., Sun, M., Han, Y., & Wang, Z. (2023). Camouflaged object segmentation based on matching–recognition–refinement network. *IEEE Transactions on Neural Networks and Learning Systems*.
- Yıldırım, E. (2020). Yapay Sinir Ağı(Artificial Neural Network) Nedir?. Veribilimiokulu. <https://www.veribilimiokulu.com/yapay-sinir-agiartificial-neural-network-nedir/> [Erişim Tarihi: 10 Mart 2025].
- Yin, C., Yang, K., Li, J., Li, X., & Wu, Y. (2024). Camouflaged object detection via complementary information-selected network based on visual and semantic separation. *IEEE Transactions on Industrial Informatics*.
- Yu, X., Zeng, N., Liu, S., & Zhang, Y. D. (2019). Utilization of DenseNet201 for diagnosis of breast abnormality. *Machine Vision and Applications*, 30, 1135-1144.

- Yan, J., Le, T. N., Nguyen, K. D., Tran, M. T., Do, T. T., & Nguyen, T. V. (2020). Mirrornet: Bio-inspired adversarial attack for camouflaged object segmentation. *arXiv preprint arXiv:2007.12881*, 3.
- Yang, Y., & Zhang, Q. (2023). Finding camouflaged objects along the camouflage mechanisms. *IEEE Transactions on Circuits and Systems for Video Technology*.
- Yao, S., Sun, H., Xiang, T. Z., Wang, X., & Cao, X. (2024). Hierarchical graph interaction transformer with dynamic token clustering for camouflaged object detection. *IEEE Transactions on Image Processing*.
- Yegnanarayana, B. (2009). Artificial neural networks. PHI Learning Pvt. Ltd..
- Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks?. *Advances in neural information processing systems*, 27.
- Yu, X., Wang, Z., Wang, Y., & Zhang, C. (2021). Edge detection of agricultural products based on morphologically improved canny algorithm. *Mathematical Problems in Engineering*, 2021(1), 6664970.
- Yu, X., Zeng, N., Liu, S., & Zhang, Y. D. (2019). Utilization of DenseNet201 for diagnosis of breast abnormality. *Machine Vision and Applications*, 30, 1135-1144.
- Yuan, H., Ge, H., He, W., & Huang, C. (2024, March). Edge optimization detection network for camouflaged targets. In *2024 6th International Conference on Natural Language Processing (ICNLP)* (pp. 728-732). IEEE.
- Zafar, A., Aamir, M., Mohd Naw, N., Arshad, A., Riaz, S., Alruban, A., ... & Almotairi, S. (2022). A comparison of pooling methods for convolutional neural networks. *Applied Sciences*, 12(17), 8643.
- Zaheer, R., & Shaziya, H. (2018, January). GPU-based empirical evaluation of activation functions in convolutional neural networks. In *2018 2nd international conference on inventive systems and control (ICISC)* (pp. 769-773). IEEE.
- Zhai, Q., Li, X., Yang, F., Jiao, Z., Luo, P., Cheng, H., & Liu, Z. (2022). Mgl: Mutual graph learning for camouflaged object detection. *IEEE Transactions on Image Processing*, 32, 1897-1910.
- Zhang, C., Wang, K., Bi, H., Liu, Z., & Yang, L. (2022). Camouflaged object detection via neighbor connection and hierarchical information transfer. *Computer Vision and Image Understanding*, 221, 103450.
- Zhang, S., Kong, D., Xing, Y., Lu, Y., Ran, L., Liang, G., ... & Zhang, Y. (2024). Frequency-Guided Spatial Adaptation for Camouflaged Object Detection. *arXiv preprint arXiv:2409.12421*.
- Zhang, W., Shan, S., Zhang, H., Gao, W., & Chen, X. (2005). Multi-resolution histograms of local variation patterns (MHLVP) for robust face recognition. In *Audio-and Video-Based Biometric Person Authentication: 5th International Conference, AVBPA 2005, Hilton Rye Town, NY, USA, July 20-22, 2005. Proceedings 5* (pp. 937-944). Springer Berlin Heidelberg.

- Zhang, X., Gao, M., Gao, G., Wang, X., & Wang, Q. (2024, June). Edge-Guided Multilevel Feature Fusion Network for Lightweight Camouflaged Object Detection. In 2024 International Joint Conference on Neural Networks (IJCNN) (pp. 1-7). IEEE.
- Zhang, X., Xiao, T., Ji, G., Wu, X., Fu, K., & Zhao, Q. (2024). Explicit Motion Handling and Interactive Prompting for Video Camouflaged Object Detection. arXiv preprint arXiv:2403.01968.
- Zhang, X., Yin, B., Lin, Z., Hou, Q., Fan, D. P., & Cheng, M. M. (2023). Referring camouflaged object detection. arXiv preprint arXiv:2306.07532.
- Zhang, X., & Zhu, C. (2015, July). Camouflage modeling for moving object detection. In 2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP) (pp. 249-253). IEEE.
- Zhang, X., Zhu, C., Wang, S., Liu, Y., & Ye, M. (2016). A Bayesian approach to camouflaged moving object detection. IEEE transactions on circuits and systems for video technology, 27(9), 2001-2013.
- Zhao, C., Zhang, Y., Su, N., & Yan, Y. (2024, January). Camouflage Object Detection based on Mixed Multi-scale and Multi-attention Network. In Proceedings of the 2024 7th International Conference on Image and Graphics Processing (pp. 32-38).
- Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2881-2890).
- Zheng, Y., Yang, C., & Merkulov, A. (2018, May). Breast cancer screening using convolutional neural network and follow-up digital mammography. In Computational Imaging III (Vol. 10669, p. 1066905). SPIE.
- Zhong, Y., Li, B., Tang, L., Kuang, S., Wu, S., & Ding, S. (2022). Detecting camouflaged object in frequency domain. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 4504-4513).
- Zhou, T., Zhou, Y., Gong, C., Yang, J., & Zhang, Y. (2022). Feature aggregation and propagation network for camouflaged object detection. IEEE Transactions on Image Processing, 31, 7036-7047.
- Zhu, J., Zhang, X., Zhang, S., & Liu, J. (2021, May). Inferring camouflaged objects by texture-aware interactive guidance network. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 4, pp. 3599-3607).
- Zhuge, M., Lu, X., Guo, Y., Cai, Z., & Chen, S. (2022). CubeNet: X-shape connection for camouflaged object detection. Pattern Recognition, 127, 108644.

ÖZGEÇMİŞ

Erkan BAYRAM, İlköğrenimini Ülkü Köy İlköğretim Okulu'nda bitirdikten sonra lise eğitimini 2001 yılında Mudanya Ahmet Rüştü (YDA) Lisesi'nde tamamladı. 2002-2003 öğretim yılında Kocaeli Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği bölümünde lisans eğitimine başladı ve 2007 yılında bu bölümden mezun oldu. 2013 yılında Atatürk Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim dalında tezli yüksek lisans programına başladı ve 2016 yılında mezun oldu. 2009 yılından itibaren Atatürk Üniversitesi bünyesinde çalışmaktadır.



Tez Kapsamında Yayınlanan Çalışmalar

- Bayram, E., & Nabiyeu, V. (2020, October). Image segmentation by using K-means clustering algorithm in Euclidean and Mahalanobis distance calculation in camouflage images. In 2020 28th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.
- Bayram, E., & Nabiyeu, V. (2021, April). Classification of Butterfly Camouflage Images Based on Feature Extraction with Local Binary Patterns. In 2nd International Symposium on Applied Sciences and Engineering ISASE2021 At: Ataturk University, Erzurum, TURKEY
- Bayram, E., & Nabiyeu, V. (2021, June). Classification of camouflage images using Local Binary Patterns (LBP). In 2021 29th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.
- Bayram, E., & Nabiyeu, V. V. (2025). A Hybrid Method for Segmentation and Recognition of Butterfly Objects in Camouflage Images. Arabian Journal for Science and Engineering, 1-24.
- Bayram, E., & Nabiyeu, V. V. (2024). Önceden Eğitilmiş Evrişimsel Sinir Ağı Kullanarak Kamuflej Görüntülerinin Sınıflandırılması. In: 3rd International Conference on Engineering, Natural and Social Sciences At: Konya, TÜRKİYE, https://drive.google.com/file/d/1847SEy_Tfx7ke1qnO6Pepq873WnlkFkV/view
- Bayram, E., & Nabiyeu, V.V (2024). Classification of camouflage images using lbp, hog and classical machine learning methods. In: 3. Bilisel International Truva Scientific Researches and Innovation Congress, pp. 610–618. https://www.researchgate.net/publication/386026396_CLASSIFICATION_OF_CAMOUFLAGE_IMAGES_USING_LBP_HOG_AND_CLASSICAL_MACHINE_LEARNING_METHODS