**Görüntü İşleme ve Derin Öğrenme Tabanlı Ağaç Türü Sınıflandırma Sistemi**

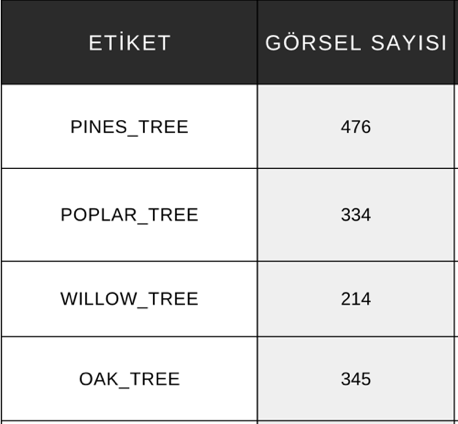
Burak Bozoğlu

210541025

1. **Giriş**

Doğal ekosistemlerin sürdürülebilirliği ve orman kaynaklarının etkin yönetimi, biyolojik çeşitliliğin izlenmesi ve korunmasıyla doğrudan ilişkilidir. Bu çerçevede, ağaç türlerinin doğru ve hızlı bir şekilde sınıflandırılması; orman envanteri çalışmaları, habitat izleme sistemleri ve çevresel karar destek mekanizmaları açısından büyük önem arz etmektedir. Bu çalışmada, çeşitli ağaç türlerine ait görseller kullanılarak eğitilen bir derin öğrenme modeli aracılığıyla, bilinmeyen bir görüntünün ait olduğu ağaç türünün otomatik olarak tahmin edilmesi hedeflenmiştir.Çalışma kapsamında, önceden eğitilmiş ve hafif yapısıyla dikkat çeken **MobileNetV2** mimarisi kullanılmış ve bu mimari **Transfer Öğrenme** yöntemiyle yeniden yapılandırılmıştır. Eğitim sürecinde sınırlı veriyle daha iyi sonuçlar elde edebilmek adına modelin alt katmanları ilk etapta dondurulmuş, ardından seçili katmanlar yeniden eğitilerek **fine-tuning** uygulanmıştır.Modelin başarısını artırmak ve genelleme kabiliyetini iyileştirmek için veri ön işleme aşamasında kapsamlı **veri artırma (data augmentation)** tekniklerinden yararlanılmıştır. Ayrıca, sınıflar arasında olası dengesizlikleri gidermek amacıyla **class\_weight** yöntemiyle ağırlıklı öğrenme stratejisi uygulanmıştır. Modelin performansı ise yalnızca doğruluk (accuracy) ile değil; aynı zamanda F1 skoru, hassasiyet (precision), duyarlılık (recall) ve karışıklık matrisi (confusion matrix) gibi gelişmiş metriklerle detaylı olarak değerlendirilmiştir.

**2. Veri Seti**

Bu çalışmada kullanılan veri seti, kullanıcı tarafından manuel olarak sınıflandırılmış görsellerden oluşmaktadır. Her bir alt klasör, farklı bir ağaç türünü temsil edecek şekilde yapılandırılmıştır. Bu yapılandırma, sınıflandırma modelinin etiketli veri ile eğitilebilmesini sağlamaktadır. Veri setinde yer alan görseller farklı formatlarda ve çözünürlüklerde bulunmakta olup, ön işleme aşamasında tüm görüntüler bir standarda oturtulmuştur.

Görsel verilerin heterojen yapısı nedeniyle, modellerin eğitimi öncesinde bir dizi ön işleme adımı uygulanmıştır. Tüm görüntüler, derin öğrenme modelleri ile uyumlu hale getirilebilmesi amacıyla 224x224 piksel boyutlarına yeniden boyutlandırılmıştır.

TABLO 1 VERİ SETİ İÇERİĞİ

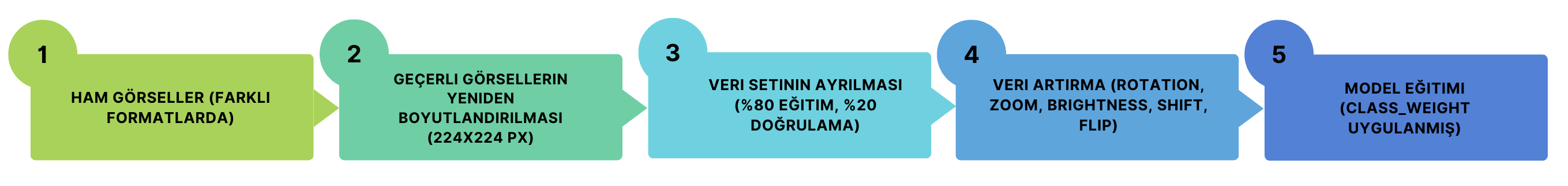
Ayrıca OpenCV kütüphanesi kullanılarak bu tür dosyalar otomatik olarak tespit edilip sistematik bir şekilde veri setinden çıkarılmıştır.

Veri seti, eğitim ve doğrulama süreçlerini destekleyecek şekilde **%80 eğitim** ve **%20 doğrulama** olmak üzere ikiye ayrılmıştır. Bu ayrım, modelin öğrenme başarısını objektif olarak değerlendirebilmek amacıyla rastgele ancak dengeli bir şekilde yapılmıştır.

Modelin genelleme yeteneğini artırmak ve aşırı öğrenmeye (overfitting) karşı dirençli hale getirmek amacıyla, hem eğitim hem de doğrulama alt kümelerine **veri artırma (data augmentation)** teknikleri uygulanmıştır. Bu süreçte kullanılan artırma parametreleri şu şekildedir:

* **Dönme (rotation):** Rastgele ±30 derece
* **Yakınlaştırma (zoom):** Maksimum %30 oranına kadar
* **Parlaklık değişimi (brightness):** [0.7, 1.3] aralığında ışık değişimi
* **Yatay ve dikey kaydırma (width & height shift):** Maksimum %20 oranında
* **Yansıtma (horizontal flip):** Rastgele yatay aynalama

Bu veri artırma stratejileri, görsel çeşitliliği zenginleştirerek modelin farklı senaryalara karşı daha dayanıklı hale gelmesini sağlamıştır. Aynı zamanda sınıf dengesizliklerinin etkisini azaltmak ve modelin her sınıfa adil şekilde öğrenebilmesini desteklemek amacıyla, eğitim sürecinde class\_weight parametresi hesaplanarak model eğitimine entegre edilmiştir.



ŞEKİL 1 Veri Seti Yapısı ve Akış Şeması

Tüm bu ön işleme ve veri artırma adımları sayesinde, modelin performans metrikleri optimize edilmiş ve hem doğruluk (accuracy) hem de F1-skora dayalı daha tutarlı sonuçlar elde edilmiştir.

## 3. Convolutional Neural Networks (CNN) ve Projedeki Önemi

Evrişimli Sinir Ağları (Convolutional Neural Networks – CNN), görsel veri analizi konusunda en etkili derin öğrenme mimarilerinden biridir. Bu çalışma kapsamında kullanılan ağaç türü sınıflandırma sistemi de, CNN mimarisine dayanan bir model (MobileNetV2) üzerinden inşa edilmiştir.

* **Evrişim Katmanı (Convolution Layer):** Görüntü üzerinde belirli filtreler (kernel) uygulanarak kenar, köşe gibi düşük seviyeli öznitelikler çıkarılır. Bu katman, modelin temel görsel özellikleri öğrenmesini sağlar.
* **Aktivasyon Fonksiyonu (ReLU):** Her evrişim katmanından sonra uygulanarak, modele doğrusal olmayanlık kazandırılır ve öğrenme kapasitesi artırılır.
* **Havuzlama Katmanı (MaxPooling):** Görüntü boyutunu küçülterek işlem yükünü azaltır, aynı zamanda özniteliklerin konumsal değişkenliğine karşı modelin esnekliğini artırır.
* **Tam Bağlantılı Katman (Dense Layer):** Elde edilen özniteliklerin daha üst düzey anlamlandırılması ve sınıflandırılması için kullanılır.
* **Çıkış Katmanı (Softmax):** Modelin her sınıfa ait tahmin olasılıklarını üretir. Bu çalışmada, ağaç türü sayısı kadar nöron içeren Softmax çıkış katmanı kullanılmıştır.

### Projede CNN Kullanımının Önemi

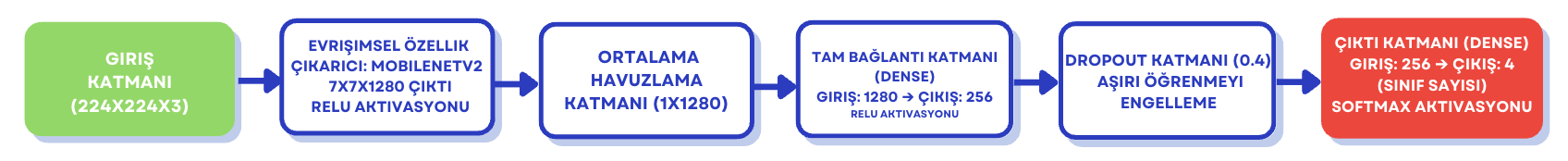
Bu çalışmada geliştirilen ağaç türü sınıflandırma sistemi, CNN mimarisinin güçlü öznitelik çıkarım yetenekleri sayesinde, farklı çözünürlüklerde ve koşullarda çekilmiş görsellerden etkili biçimde öğrenme sağlamıştır. Görsel verinin doğası gereği; renk, ışık, doku ve açısal farklılıklar gibi değişkenler modelin genelleme yetisini zorlayabilir. Ancak CNN mimarisi, bu farklılıkları soyut düzeyde özniteliklere dönüştürerek yüksek doğrulukta tahmin yapmayı mümkün kılmaktadır.

Ek olarak, bu projede kullanılan **MobileNetV2** mimarisi, CNN yapısının daha hafif ve verimli bir versiyonudur. Bu sayede hem eğitim süresi kısaltılmış, hem de model mobil sistemlerde çalışabilecek düzeye indirgenmiştir.

**4. Model Mimarisi**

Bu projede kullanılan mimari, ImageNet üzerinde önceden eğitilmiş **MobileNetV2** üzerine inşa edilmiştir. MobileNetV2, mobil ve gömülü cihazlar için optimize edilmiş hafif bir CNN mimarisidir. Özellikle sınırlı kaynaklarda yüksek doğruluk sunması açısından tercih edilmiştir.

**Mimaride Kullanılan Yapılar:**

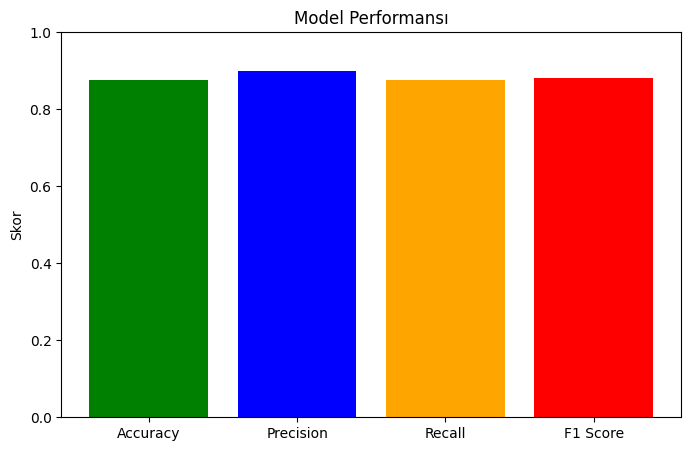
* **Base Model**: MobileNetV2 (include\_top=False, weights='imagenet')
* **GlobalAveragePooling2D**: Son evrişim haritalarından ortalama değer alınarak çıkış boyutu küçültülür.
* **ReLU (Rectified Linear Unit)**: Dense(256, activation='relu') katmanında kullanılmıştır.
* **Dropout (0.4)**: Aşırı öğrenmeyi engellemek için nöronların %40’ı rastgele devre dışı bırakılmıştır.
* **Softmax**: Dense(n\_classes, activation='softmax') katmanı ile her sınıfa ait olasılık hesaplanır.

ŞEKİL 2 SİSTEMİN ÇALIŞMA MİMARİSİ

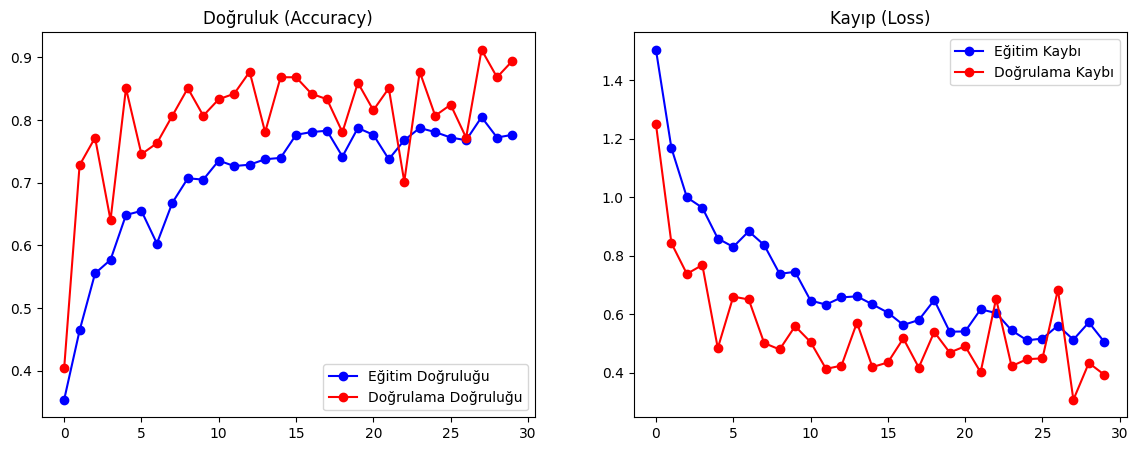
Model ilk olarak sabitlenmiş (frozen) MobileNetV2 ile eğitilmiş, ardından **son 50 katmanı yeniden eğitilecek şekilde açılarak fine-tuning** uygulanmıştır. Bu sayede hem genel hem özel öznitelikler model tarafından öğrenilmiştir.

**5. Performans Metrikleri**

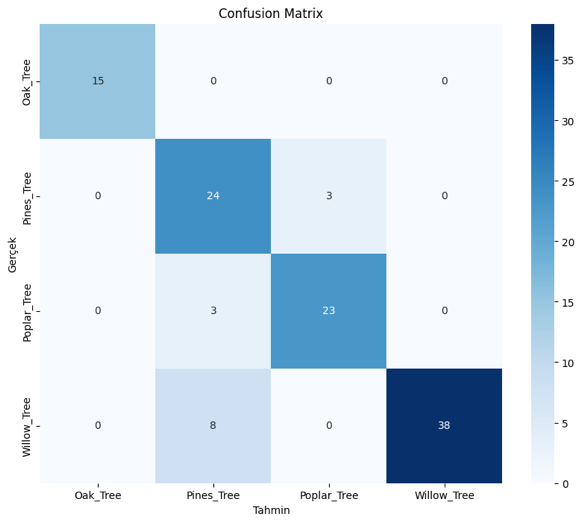
Modelin başarısını değerlendirmek için aşağıdaki metrikler kullanılmıştır:

* **Accuracy (Doğruluk):** Toplam doğru tahminlerin tüm tahminlere oranıdır.
* **Precision (Kesinlik):** Doğru pozitif tahminlerin, yapılan tüm pozitif tahminlere oranıdır.
* **Recall (Duyarlılık):** Gerçek pozitiflerin, toplam gerçek pozitiflere oranıdır.
* **F1-Score:** Precision ve Recall değerlerinin harmonik ortalamasıdır. Dengesiz veri kümelerinde daha dengeli bir performans ölçütü sağlar.
* **Confusion Matrix (Karışıklık Matrisi):** Her sınıf için yapılan doğru ve yanlış tahminleri görselleştiren bir tablodur.

Şekil3.Performans Metrikleri

**Eğitim-Doğrulama Doğruluk ve Kayıp Eğrileri**  
Aşağıdaki grafiklerde modelin eğitim sürecindeki doğruluk (accuracy) ve kayıp (loss) değerlerinin epoch’lar boyunca nasıl değiştiği gösterilmektedir.  
Soldaki grafik doğruluk değerlerini, sağdaki grafik ise kayıp değerlerini göstermektedir. Doğrulama eğrilerinin eğitim eğrilerine yakın seyretmesi modelin aşırı öğrenmeden (overfitting) kaçındığını göstermektedir.

Şekil 4. Doğruluk ve Kayıp Eğrileri

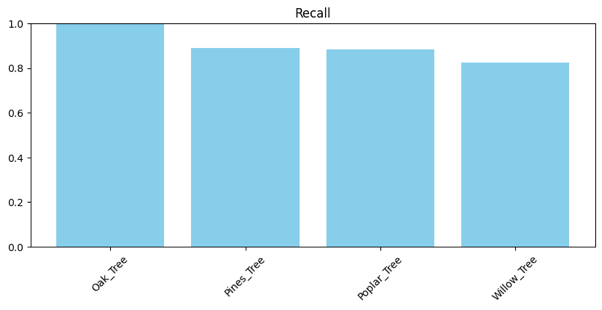
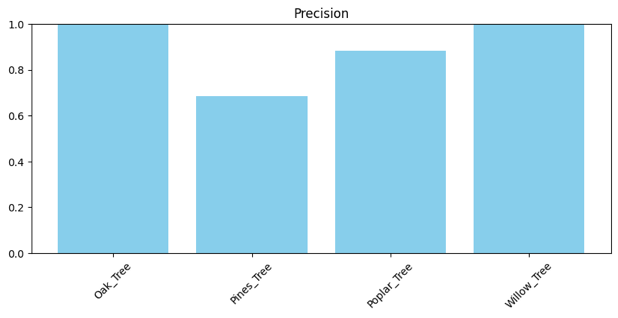
**Karışıklık Matrisi (Confusion Matrix)**  
Aşağıdaki karışıklık matrisi, modelin test verisi üzerindeki tahmin performansını göstermektedir. Diagonal üzerindeki değerler doğru tahminleri, diğer değerler ise yanlış sınıflandırmaları temsil eder.

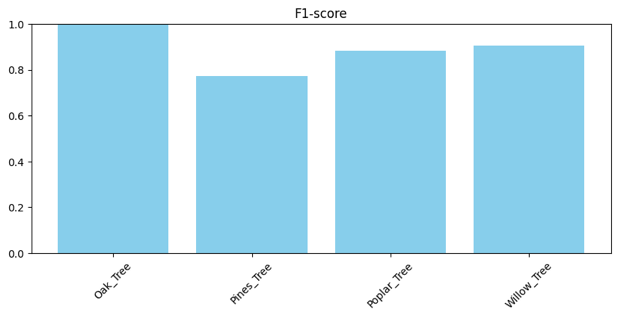
Şekil 5. Karışıklık Matrisi (Confusion Matrix)

**6. Sonuçlar ve Model Değerlendirme**

Model, test verisi üzerinde yüksek doğruluk ve dengeli sınıf performansı göstermiştir. Özellikle class\_weight ile sınıf dengesizlikleri başarılı bir şekilde telafi edilmiştir. Fine-tuning aşaması, genel doğrulukta belirgin bir artış sağlamıştır.

Ayrıca, geliştirilen sistem tek görsel tahmini özelliği ile gerçek zamanlı kullanım senaryolarına da uygundur. Görsel yüklemesi sonrası sistem tahmini yaparak sınıf ismini vermektedir. Bu modül, OpenCV ile entegre çalışmaktadır.

****

****

**7. Kaynakça**

1. Howard, A. G., et al. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*. arXiv preprint arXiv:1704.04861.
2. Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
4. Pedregosa, F., et al. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825-2830.
5. OpenCV. (2023). *Open Source Computer Vision Library*. <https://opencv.org>
6. TensorFlow Keras API Documentation. https://www.tensorflow.org/api\_docs/python/tf/keras
7. Seaborn Visualization Library. https://seaborn.pydata.org