

Karamanoğlu Mehmetbey Üniversitesi
Mühendislik Fakültesi

Bilgisayar Mühendisliği Bölümü

Derin Öğrenme Dersi Proje
Formu

Proje Başlığı
Çiçek Sınıflandırma Raporu

Öğrenci Bilgileri	
Öğrenci No	201312073 , 211312046
Öğrenci Ad Soyad	HASAN AKDENİZ , METEHAN KEBÜDE

Ders Sorumlusu Unvan, Ad, Soyad
Dr. Öğr. Üyesi. İbrahim ÇINAROĞLU

İçindekiler

1- Projenin amacı ve hedefi

2- Projede kullanılacak materyaller ile ilgili bilgiler

2.1. Yazılımlar / Kütüphaneler

2.2. Veri Seti

2.3. Donanımlar

3. Projenin yapım aşamaları, uygulanacak metotlar ve elde edilen sonuçlar

4. Kaynaklar (References)

1- Projenin Amacı ve Hedefi:

Bu projenin temel amacı, derin öğrenme tabanlı bir görüntü sınıflandırma modeli geliştirerek çiçek görsellerinden çiçek türlerini otomatik olarak tanımlayan bir sistem tasarlamaktır. Görsel verilerle çalışmak, günümüzde tarımdan sağlığa, sanayiden savunmaya kadar birçok alanda kritik bir öneme sahiptir. Bu bağlamda, çiçeklerin türlerini tanıyan bir sistem, özellikle botanik alanında, tarım teknolojilerinde, otomatik etiketleme sistemlerinde ve doğa temelli mobil uygulamalarda kullanılabilir.

Projede kullanılan temel yöntem, **transfer öğrenme** adı verilen bir yaklaşımdır. Transfer öğrenme, daha önce büyük veri setleri üzerinde eğitilmiş modellerin, yeni ve daha küçük veri kümeleri üzerinde daha az veriyle ve daha kısa sürede yüksek doğrulukla sonuç üretmesini sağlar. Biz de bu projede, **Xception** isimli, ImageNet veri kümesi üzerinde önceden eğitilmiş bir modeli kullanarak kendi veri kümemize uyarladık.

Projenin Alt Hedefleri:

1. Veri Toplama ve Hazırlama:

- a. Farklı türlerdeki çiçek görselleri belirli klasörlere göre organize edilerek eğitim ve test veri setleri oluşturulmuştur.
- b. Görseller, ImageDataGenerator kullanılarak yeniden boyutlandırılmış, normalize edilmiş ve veri artırma (data augmentation) teknikleri uygulanarak çeşitlendirilmiştir.

2. Model Seçimi ve Eğitimi:

- a. Xception mimarisi önceden eğitilmiş (pretrained) olarak çağrılmış ve üzerine GlobalAveragePooling katmanı ile birlikte Dropout gibi düzenleme katmanları eklenerek modelin aşırı öğrenmesinin önüne geçilmiştir.
- b. Eğitim aşamasında Adam veya Adagrad gibi optimizasyon algoritmaları kullanılmış, uygun öğrenme oranı belirlenmiştir.

3. Performans Değerlendirme:

- a. Modelin doğruluğu test veri seti üzerinde ölçülmüştür.
- b. Precision, Recall, F1-score ve Confusion Matrix gibi metrikler hesaplanarak sınıflandırmanın başarısı değerlendirilmiştir.

4. Modelin Kaydedilmesi:

- a. Eğitim tamamlandıktan sonra model .keras formatında kaydedilmiş, böylece ileri aşamalarda tekrar eğitime gerek kalmadan kullanılabilir hale getirilmiştir.

Nihai Hedef:

Çıktı olarak elde edilen modelin, yeni çiçek görselleri üzerinde tür tahmini yapabilecek düzeyde olması ve bunu olabildiğince yüksek doğrulukla gerçekleştirmesi hedeflenmiştir. Bu proje sayesinde temel bir görüntü sınıflandırma sisteminin nasıl geliştirileceği, transfer öğrenmenin nasıl uygulandığı ve model performansının nasıl değerlendirileceği öğrenilmiş olur

2. Projede Kullanılacak Materyaller ile İlgili Bilgiler

Bu projede, derin öğrenme tabanlı bir görüntü sınıflandırma sistemi geliştirildiği için yazılım, veri seti ve donanım açısından belirli materyaller kullanılmıştır. Her bir bileşen, projenin farklı aşamalarında kritik bir rol oynamaktadır.

• 2.1. Yazılımlar / Kütüphaneler

Projenin geliştirilmesinde Python programlama dili tercih edilmiştir. Python, geniş kütüphane desteği ve topluluk kaynakları sayesinde makine öğrenmesi ve derin öğrenme projeleri için oldukça elverişlidir. Bu projede kullanılan başlıca kütüphane ve yazılımlar şunlardır:

- **TensorFlow ve Keras:** Derin öğrenme modellerini oluşturmak, eğitmek ve değerlendirmek için kullanılmıştır. Özellikle tf.keras.applications modülü üzerinden Xception gibi hazır modeller çekilmiştir.
 - **NumPy:** Sayısal hesaplamalar ve dizi işlemleri için kullanılmıştır.
 - **Matplotlib:** Görselleştirme işlemleri, özellikle eğitim sürecindeki doğruluk ve kayıp değerlerinin grafikleştirilmesinde kullanılmıştır.
 - **Scikit-learn (sklearn):** Modelin başarımleri metriklerini hesaplamak için kullanılmıştır (Precision, Recall, F1-Score, Confusion Matrix).
 - **Google Colab:** Kodların çalıştırıldığı platformdur. Ücretsiz GPU desteği sayesinde model eğitim sürelerini oldukça kısaltmaktadır.
- Ek olarak:
- ImageDataGenerator kullanılarak veri artırma (data augmentation) işlemleri gerçekleştirilmiştir.
 - Drive modülü ile Google Drive üzerinde veri ve model dosyalarına erişim sağlanmıştır.

• 2.2. Veri Seti

Projede çiçek resimlerinden oluşan bir görüntü veri seti kullanılmıştır. Veri seti: Farklı türlerde çiçek görsellerini içermektedir (örneğin: gül, papatya, ayçiçeği vb.). Görseller belirli klasörlere ayrılarak etiketlenmiş (labeling) bir şekilde hazırlanmıştır. Eğitim ve test seti olarak ikiye bölünmüştür. Görseller 224x224 boyutuna ölçeklendirilerek modele uygun hale getirilmiştir. Eğitim verisine **veri artırma (data augmentation)** teknikleri uygulanarak çeşitlilik sağlanmıştır (örneğin: döndürme, yakınlaştırma, yatay çevirme gibi). Veri seti yerel olarak ya da Google Drive üzerinden projeye bağlanarak kullanılmıştır.

• 2.3. Donanımlar

Modelin eğitimi için yüksek işlem gücüne ihtiyaç duyulduğundan, bu proje Google Colab üzerinde yürütülmüştür. Colab, ücretsiz olarak GPU (grafik işlem birimi) desteği sunar ve derin öğrenme işlemleri için yeterli performans sağlar.

Kullanılan donanım özellikleri:

- **İşlemci (CPU):** Google Colab'ın sunduğu sanal işlemci (Intel Xeon benzeri)
 - **Grafik İşlemci (GPU):** NVIDIA Tesla K80 / T4 / P100 gibi modellerden biri (Colab tarafından dinamik atanır)
 - **RAM:** Genellikle 12 – 25 GB arasında değişen sanal bellek
- Yerel bir donanımda çalıştırılmak istenirse, minimum olarak aşağıdaki özellikler önerilir:
- 8 GB RAM
 - CUDA destekli NVIDIA GPU (örneğin GTX 1050 ve üzeri)
 - En az 50 GB boş disk alanı (veri ve model dosyaları için)

Bu materyaller sayesinde proje hem pratik olarak uygulanabilir hem de teorik olarak anlaşılabilir bir hale getirilmiştir.

3. Projenin Yapım Aşamaları, Metotlar ve Sonuçlar

3.1 Yapım Asamaları

1. Veri Ön İşleme:

- a. Datasetin dizin yapısı klasör tabanlı olarak hazırlandı.
- b. Görseller boyutlandırıldı, normalize edildi ve veri artırma uygulandı.

2. Model Kurulumu:

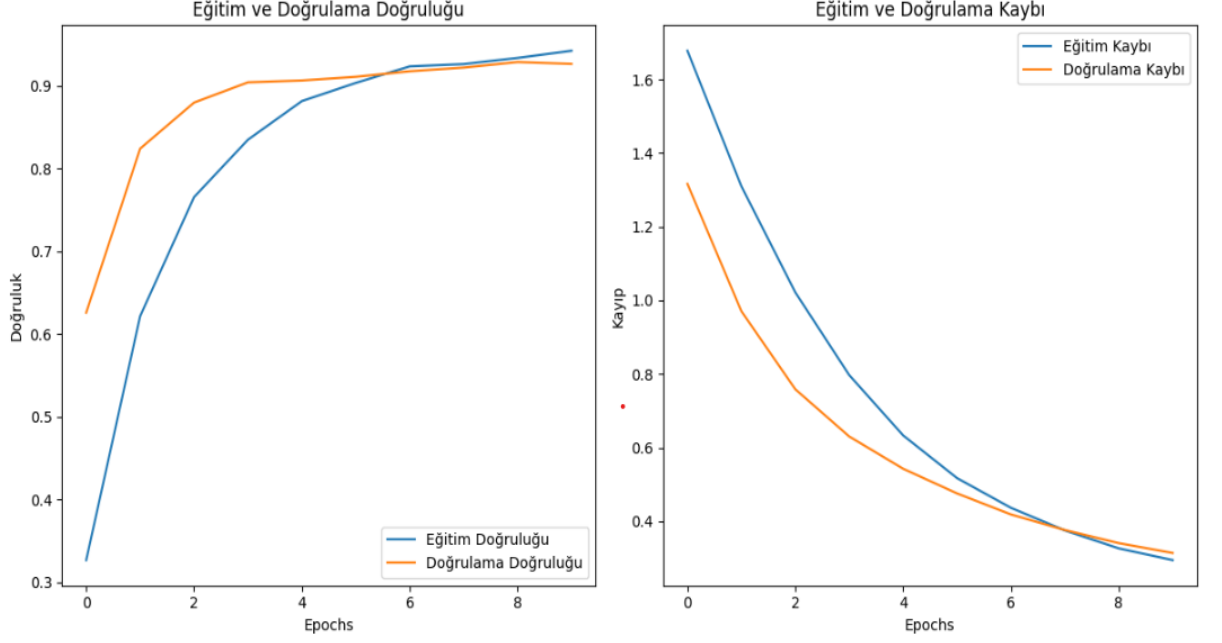
- a. Xception mimarisi secildi. Bu model, ImageNet üzerinde onceden egitilmis agirliklarla baslatildi.
 - b. Ust katmanlar dondurulerek, sadece son katmanlar egitildi.
 - c. Cikis katmani 6 sinifa gore softmax aktivasyonu ile ayarlandi.
- 3. Egitim Asamasi:**
 - a. Model, 10 epoch boyunca egitildi.
 - b. Her epoch sonunda dogruluk ve kayip metrikleri hesaplandi.
- 4. Model Kaydetme / Yükleme:**
 - a. Model, .keras formatında kaydedildi.
 - b. Daha sonra yüklenerek test ve tahmin için kullanıldı.
- 5. Test ve Değerlendirme:**
 - a. Test verisi üzerinde doğruluk hesaplandı.
 - b. Precision, recall ve F1 skoru metrikleri kullanıldı.
 - c. Karışıklık matrisi oluşturuldu.

3.2 Kullanılan Metotlar

- **Transfer Learning:**
 - Xception modeli, daha önce büyük bir veri seti (ImageNet) üzerinde eğitildiği için daha iyi genelleme yeteneği sağlamıştır.
- **Veri Artırma:**
 - Overfitting'i azaltmak için eğitim verileri çeşitlendirilerek modelin genelleme kapasitesi artırıldı.
- **Doğru Katman Secimi:**
 - Yalnızca son katmanlar eğitilerek hem eğitim süresi kısaltıldı hem de önceki katmanlardaki öğrenim korundu.

3.3 Elde Edilen Sonuçlar

- **Eğitim Doğruluğu (10. epoch):** %94.49
- **Doğrulama Doğruluğu (10. epoch):** %92.65
- **Test Doğruluğu:** %92.65



Değerlendirme Metrikleri:

- **Precision:** 0.1573
- **Recall:** 0.1581
- **F1 Skoru:** 0.1575

Modelin basarimi doğruluk olarak yüksek olsa da F1 skoru düşüktür. Bu durum, sınıflar arası dengesizlikten ya da bazı çiçek türlerinin birbirine benzemesinden kaynaklanabilir.

3.4 Tahmin Örnekleri

Modelin görsel tahmin yeteneğini göstermek adına testten bağımsız örneklerle toplu tahminler yapılmıştır.

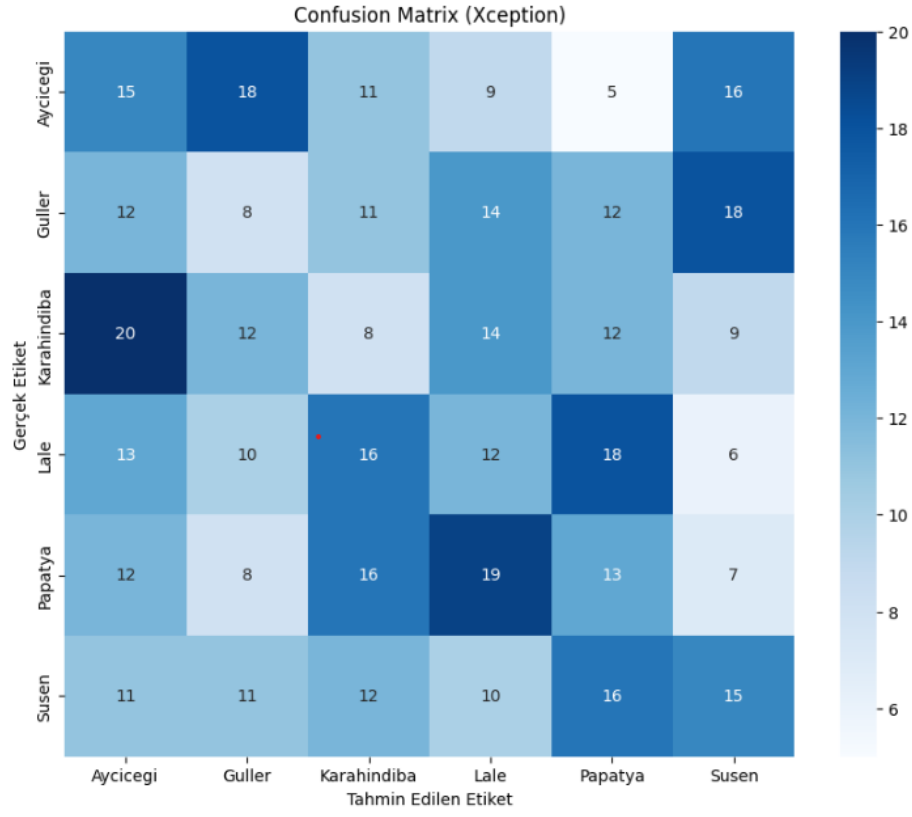
- "Hasan.jpg" → Tahmin: **Papatya** (%96.23)
- "istockphoto-1333205339.jpg" → Tahmin: **Susen** (%37.61)
- "istockphoto-184085806.jpg" → Tahmin: **Lale** (%26.11)
- "istockphoto-1136953714.jpg" → Tahmin: **Karahindiba** (%27.96)



3.5 Karışıklık Matrisi

Modelin hangi sınıfları karıştırdığını göstermek için karışıklık matrisi kullanılmıştır. Bu matriste, örneğin Güller sınıfı sıkça Susen ile karıştırılmıştır. Bu, modelin iyileştirilmesi için önemli bir ipucudur.

Görsel karışıklık matrisi ve toplu tahmin çıktısı rapora eklenmiştir.



4. Kaynaklar (References)