

# PROJE RAPOR ŞABLONU

## 1. GİRİŞ

**Amaç:** Bu projenin amacı, çevresel sesleri derin öğrenme tabanlı Evrişimli Sinir Ağları (CNN) kullanarak sınıflandırmaktır. Projede UrbanSound8K veri setindeki 10 farklı çevresel ses türü (köpek havlaması, siren, matkap, motor rölantisi, çocuk sesleri vb.) Mel-Spectrogram yöntemine dönüştürülerek görsel formda CNN modeline giriş olarak verilmiştir.

**Motivasyon:** Günlük hayatta akustik sinyallerin otomatik olarak algılanması; akıllı şehir uygulamaları, güvenlik sistemleri, ses bazlı olay tespiti, robotik ve gözetim sistemleri için oldukça kritik bir konudur. Ses sınıflandırma, görüntü sınıflandırmaya göre daha zorlu bir problemdir çünkü ses zaman serisi + frekans bilgisini birlikte içerir. Bu proje, bu karmaşıklığı çözmek için derin öğrenmenin güçlü özellik çıkarma kapasitesini kullanmayı amaçlamaktadır.

## 2. LİTERATÜR TARAMASI

Ses sınıflandırma konusunda literatürde farklı yaklaşımlar bulunmaktadır.

Örnek:

- Geleneksel Makine Öğrenmesi: SVM veya k-NN gibi klasik yöntemler kullanılmıştır ancak karmaşık çevresel seslerde düşük başarı gösterir.
- Mel-Spectrogram + CNN: Son yıllarda en başarılı yöntemlerden biridir. Zaman-frekans temsili görüntü olarak kullanılır ve CNN bu yapıyı çok iyi öğrenebilir.
- Derin CNN ve CRNN Modelleri: Çok daha yüksek doğruluk sunar ancak eğitim maliyeti daha yüksektir.
- UrbanSound8K Üzerine Yapılan Çalışmalar: Çoğu çalışma %80–92 arası doğruluk bildirmiştir. Bu proje bu literatür seviyesine ulaşmayı hedeflemiştir.

## 3. VERİ SETİ TANIMI

**Veri Seti Adı:** UrbanSound8K

**Kaynak:** Kaggle - <https://www.kaggle.com/datasets/chrisfilo/urbansound8k>

**Özellikler:** 8732 adet WAV formatındaki ses kaydı 10 farklı sınıftan birine aittir. Dosyalar 10 fold olarak klasörlenmiş ve etiketler metadata.csv dosyasına kaydedilmiştir.

### **Ön İşleme:**

- Her ses dosyası Librosa ile 22.050 Hz'e yeniden örneklendirilmiştir.
- Ses sinyalleri Mel-Spectrogram formatına dönüştürülmüştür.
- Tüm görüntüler  $128 \times 128 \times 1$  boyutuna normalize edilmiştir.
- Piksel değerleri  $[0,1]$  aralığına ölçeklenmiştir.
- Veri kümesinde denge sağlamak için her sınıftan 80 örnek seçilmiştir (toplam 800 veri).

## **4. METODOLOJİ**

Bu projede çevresel seslerin sınıflandırılması için Mel-Spectrogram tabanlı bir derin öğrenme yaklaşımı kullanılmıştır. İlk aşamada UrbanSound8K veri setindeki her bir WAV dosyası Librosa kütüphanesi ile okunmuş, ses sinyalleri zaman–frekans düzleminde temsil eden MelSpectrogram formatına dönüştürülmüştür. Bu dönüşümle ses dalgaları görüntü benzeri bir yapıya kavuşmuş ve CNN modelinin öğrenebileceği uygun forma getirilmiştir. Elde edilen spektrogramlar dB ölçeğine çevrilerek normalize edilmiş ve sabit boyutlu ( $128 \times 128 \times 1$ ) giriş tensörlerine dönüştürülmüştür. Veri setindeki sınıf dağılımını dengelemek amacıyla her sınıftan eşit sayıda örnek (80 adet) seçilmiş ve böylece toplam 800 görüntüden oluşan dengeli bir eğitim/test veri seti oluşturulmuştur. Bu veriler eğitim ve test olmak üzere ikiye ayrıldıktan sonra, 3 konvolüsyon bloğu içeren bir CNN modeli tasarlanmış ve Adam optimizasyon algoritması ile eğitilmiştir. Modelin eğitimi sırasında verinin %20'si doğrulama amaçlı kullanılmıştır.

### **Model Seçimi:**

Bu projede model olarak Evrişimli Sinir Ağları (Convolutional Neural Networks – CNN) tercih edilmiştir. CNN'ler, görüntü verilerinden örüntü çıkarma konusunda oldukça başarılı olduğundan, ses dosyalarının Mel-Spectrogram formatına dönüştürülmesi ile elde edilen zaman–frekans temsillerini etkili bir şekilde işleyebilmektedir. Mel-Spectrogram'lar görüntü yapısına benzediği için CNN modelleri bu verilerden yüksek seviyeli özellikleri otomatik olarak öğrenebilmektedir. Bu nedenle klasik makine öğrenmesi modelleri yerine, daha yüksek doğruluk sağlayan ve ses sınıflandırma literatüründe yaygın olarak kullanılan bir CNN mimarisi seçilmiştir. Model; performans, eğitim süresi ve hesaplama maliyeti arasında denge sağlamak amacıyla üç konvolüsyon katmanı içeren hafif ama etkili bir yapı olarak tasarlanmıştır.

### **Kullanılan Algoritmalar/Çerçeveler:**

Model TensorFlow Keras kullanılarak geliştirilmiştir. Mel-Spectrogram üretimi için Librosa, veri işleme için NumPy ve Pandas, değerlendirme metrikleri için ise scikit-learn kütüphanelerinden yararlanılmıştır. Adam optimizasyon algoritması tercih edilerek ağı hızlı ve kararlı bir şekilde öğrenmesi hedeflenmiştir.

### Hiperparametreler:

Batch Size = 32, Epoch sayısı = 20, Optimizer= Adam, Öğrenme Oranı = Default Adam Oranı  
Aktivasyon Fonksiyonu= ReLU, Çıkış Katmanı= Softmax.

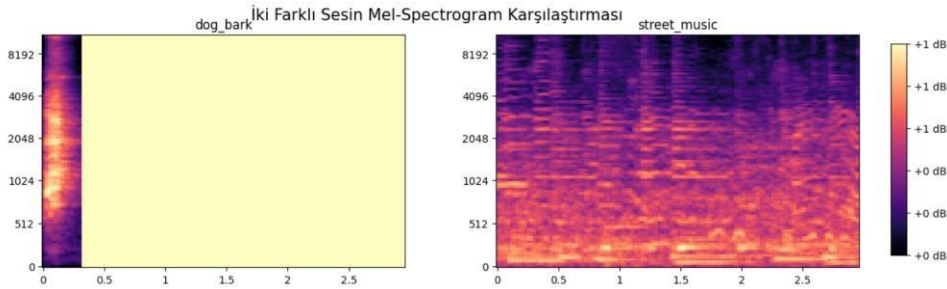
### Model Mimarisi:

Model 3 adet evrişim katmanından oluşmaktadır. Her katmanda aynı boyutta filtrelerle sahip bir Conv2D katmanı ve bir MaxPooling2D katmanı yer almaktadır. İlk katmanda 32, ikinci katmanda 64, üçüncü katmanda 128 filtre kullanılmıştır. Conv katmanlarının ardından ağ Flatten katmanı ile düzleştirilmiş ve 128 nöronlu bir tam bağlı katmana bağlanmıştır. Aşırı öğrenmeyi engellemek için %50 Dropout uygulanmış ve final katmanda 10 sınıflı bir Softmax çıkışı üretilmiştir.

## 5. KOD UYGULAMASI

### Kodun Ana Bölümleri:

- Veri yükleme ve ön işleme: UrbanSound8K veri seti okunmuş, ses dosyaları Librosa ile MelSpectrogram'a dönüştürülmüş ve eğitim için uygun boyuta getirilmiştir.



- Model tanımlama ve derleme: TensorFlow Keras kullanılarak üç konvolüsyon bloğundan oluşan bir CNN modeli oluşturulmuş ve Adam optimizier ile derlenmiştir.

```

input_shape = X_train.shape[1:]      # (128, 128, 1)
num_classes = len(np.unique(y))      # 10 sınıf

model = models.Sequential([
    layers.Conv2D(32, (3, 3), activation='relu', padding='same', input_shape=input_shape),
    layers.MaxPooling2D((2, 2)),

    layers.Conv2D(64, (3, 3), activation='relu', padding='same'),
    layers.MaxPooling2D((2, 2)),

    layers.Conv2D(128, (3, 3), activation='relu', padding='same'),
    layers.MaxPooling2D((2, 2)),

    layers.Flatten(),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.5),
    layers.Dense(num_classes, activation='softmax')
])

model.compile(
    optimizer='adam',
    loss='sparse_categorical_crossentropy',
    metrics=['accuracy']
)

model.summary()

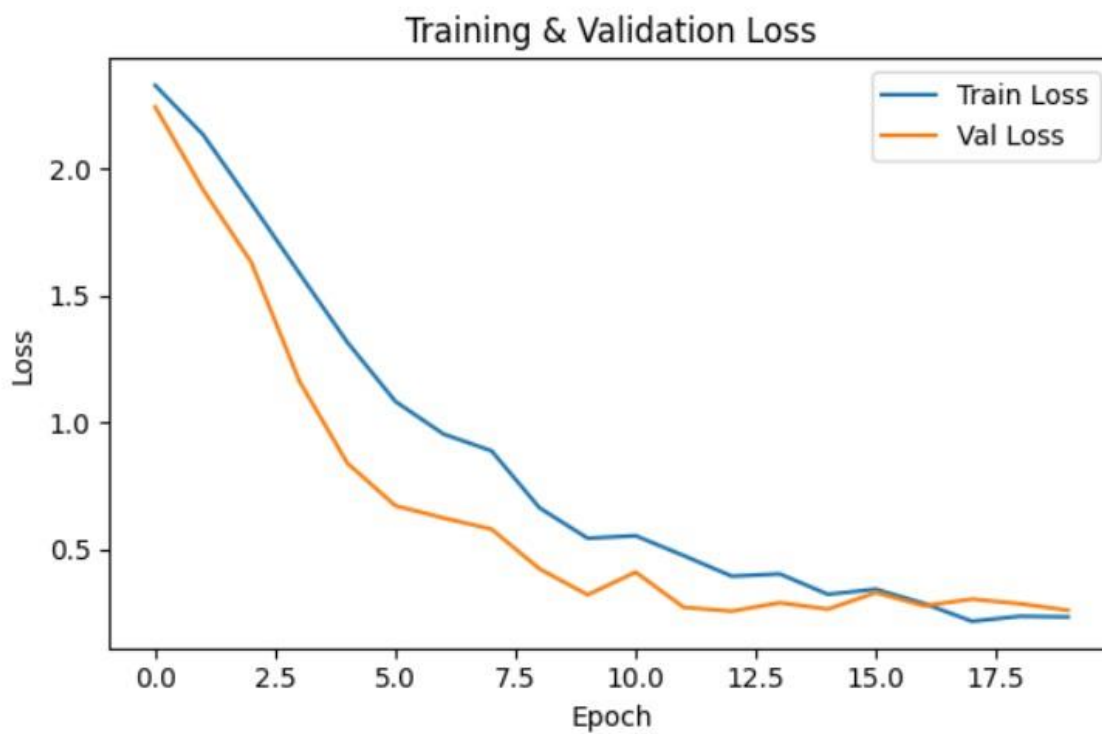
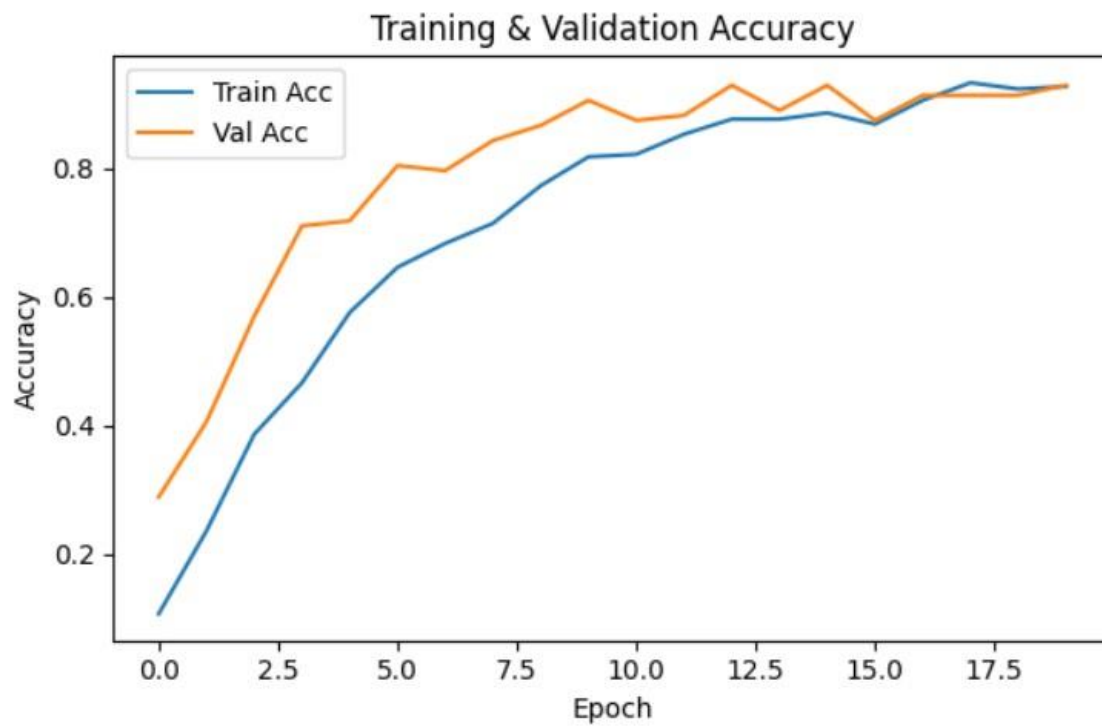
```

- Model eğitimi ve değerlendirme: Model 20 epoch boyunca eğitilmiş, ardından test doğruluğu hesaplanarak performans metrikleri elde edilmiştir.
- Sonuçların görselleştirilmesi: Eğitim/doğrulama doğruluk–kayıp grafikleri çizilmiş ve confusion matrix ile sınıf bazlı performans analiz edilmiştir.

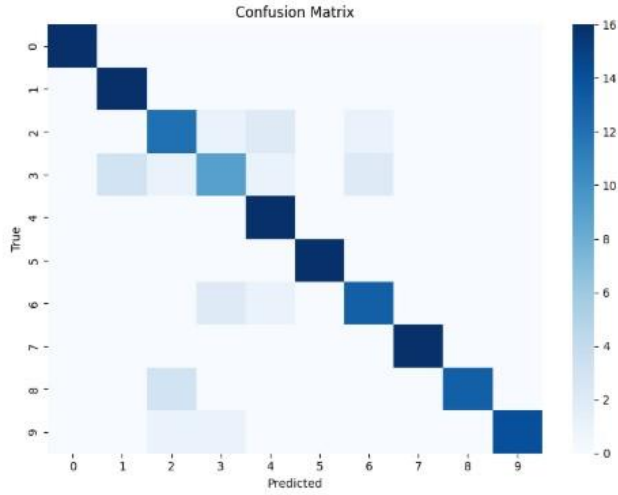
## 6. SONUÇLAR VE ANALİZ

Modelin eğitim sürecine ait doğruluk ve kayıp grafikleri, ağıın hem eğitim hem doğrulama verisi üzerinde istikrarlı bir öğrenme süreci gerçekleştirdiğini göstermektedir. Eğitim doğruluğu epoch'lar boyunca düzenli bir şekilde artmış ve validation accuracy değeri %90 seviyelerine ulaşmıştır. Test kümesinde elde edilen doğruluk oranı %88.12 olup, bu sonuç UrbanSound8K veri setinde kullanılan temel CNN modelleri için oldukça başarılı bir performanstır.

### Doğruluk ve Kayıp Grafikleri:



## Karmaşıklık Matrisi:



Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	16
1	0.94	1.00	0.91	16
2	0.71	0.75	0.73	16
3	0.69	0.56	0.62	16
4	0.88	1.00	0.89	16
5	1.00	1.00	1.00	16
6	0.81	0.81	0.81	16
7	1.00	1.00	1.00	16
8	1.00	0.81	0.90	16
9	1.00	0.88	0.93	16
accuracy			0.88	160
macro avg	0.89	0.88	0.88	160
weighted avg	0.89	0.88	0.88	160

## 7. TARTIŞMA VE SONUÇ

### Performans Değerlendirmesi:

Model literatürdeki birçok temel CNN uygulamasına denk bir başarı göstermiştir. %88 test doğruluğu UrbanSound8K gibi zor bir dataset için güçlü bir performanstır.

### Karşılaşılan Zorluklar:

Seslerin benzerliği (drilling–jackhammer), datasetin büyüklüğü ve işlem süreleri, Colab’ın runtime resetlemesi

### Bulgularınızı Özetleyin:

Bu projede UrbanSound8K veri seti kullanılarak Mel-Spectrogram tabanlı bir CNN modeli eğitilmiş ve %88 üzerinde doğruluk elde edilmiştir. Model, çevresel ses sınıflandırma problemini yüksek başarıyla çözmüştür.

### Gelecek Çalışmalar:

Bu proje gelecekte çeşitli şekillerde geliştirilebilir. Öncelikle veri artırma (noise ekleme, zaman kaydırma vb.) teknikleri kullanılarak modelin gürültüye karşı dayanıklılığı artırılabilir. CNN yerine CRNN veya Transformer tabanlı daha gelişmiş modeller denenerek doğruluk oranı yükseltilebilir. Ayrıca sistem gerçek zamanlı bir ses tanıma uygulamasına dönüştürülebilir ve daha büyük veri setleri kullanılarak modelin genelleme kabiliyeti güçlendirilebilir.

## **8. KAYNAKÇA**

Proje sırasında referans alınan tüm kaynakları, veri setlerini veya makaleleri alıntılایn. (AI araçları dahil)

[1] UrbanSound8K Dataset - <https://www.kaggle.com/datasets/chrisfilo/urbansound8k>

[2] Librosa Audio Analysis Library: <https://librosa.org/doc/latest/index.html>

[3] TensorFlow Documentation: <https://www.tensorflow.org/>

[4] AI Asistanı: ChatGPT