



**Mühendislik Fakültesi**  
**Yazılım Mühendisliği Bölümü**

**FET312 - Derin Öğrenme**

Proje İlerleme Raporu

**Trafik İşaretlerinin Evrişimli Sinir Ağları (CNN) ile**  
**Sınıflandırılması**

**Ekip Bilgileri**

Adı Soyadı	Öğrenci No	E-posta
Ramazan Bozkurt	22040301027	ramazanbozkurt@stu.topkapi.edu.tr
Ömer Utku Aktemur	22040301043	omerutkuaktemur@stu.topkapi.edu.tr

**GitHub Repo Bağlantısı:**

[https://github.com/RamazanBozkurt/DevTeam177\\_GTSRB-Traffic-Sign-Classification-CNN](https://github.com/RamazanBozkurt/DevTeam177_GTSRB-Traffic-Sign-Classification-CNN)

## İçindekiler

<b>1 Problem Tanımı ve Motivasyon .....</b>	<b>3</b>
1.1 İş/Bilimsel Soru .....	3
1.2 Görev Türü .....	3
1.3 Hedef Değişkenler .....	3
1.4 Başarı Kriterleri .....	3
<b>2 Proje Yönetimi.....</b>	<b>3</b>
2.1 Zaman Çizelgesi (Milestones) .....	3
2.2 Roller ve Sorumluluklar .....	4
<b>3 İlgili Çalışmalar (Literatür Taraması) .....</b>	<b>4</b>
<b>4 Veri Açıklaması ve Yönetimi .....</b>	<b>4</b>
4.1 Veri Kümesi (Dataset Description) .....	4
4.2 Boyut ve Şema .....	4
<b>5 Yöntemler ve Mimari .....</b>	<b>4</b>
5.1 Model A Tasarımı (Ramazan Bozkurt) .....	5
5.2 Model B Tasarımı (Ömer Utku Aktemur) .....	5
<b>6 Deney Tasarımı ve Sonuçlar .....</b>	<b>5</b>
6.1 Deney Kurulumu .....	5
6.2 Performans Karşılaştırması .....	6
<b>7 Kullanılan Araçlar .....</b>	<b>6</b>
<b>8 Kaynaklar .....</b>	<b>6</b>

## 1 Problem Tanımı ve Motivasyon

### 1.1 İş/Bilimsel Soru

Otonom sürüş sistemlerinin ve gelişmiş sürücü destek sistemlerinin (ADAS) en kritik bileşenlerinden biri, çevresel farkındalıktır. Bu proje kapsamında, araçların seyir halindeyken karşılaşılabileceği trafik işaretlerinin (hız sınırları, dur, girilmez vb.) kamera görüntüleri üzerinden otomatik olarak algılanması ve sınıflandırılması problemi ele alınmıştır. Temel bilimsel soru şudur: *"Sınırlı işlem gücüne sahip araç içi sistemlerde çalışabilecek, yüksek doğruluk oranına sahip, farklı derinliklerdeki özgün CNN mimarileri nasıl tasarlanabilir?"*

### 1.2 Görev Türü

Bu proje, bilgisayarlı görü (computer vision) alanında çok sınıflı bir **Sınıflandırma (Classification)** problemidir. Girdi olarak alınan trafik işareti görüntüleri, 43 farklı kategoriden (sınıftan) birine atanmaktadır.

### 1.3 Hedef Değişkenler

- **Girdi:** 64x64 piksel boyutuna indirgenmiş, 3 kanallı (RGB) trafik işareti görüntüleri.
- **Çıktı (Hedef):** 43 farklı trafik işareti sınıfından biri (Örn: Class 0: 20 km/s hız sınırı, Class 14: Dur işareti). Hedef değişken kategorik ve nominaldir.

### 1.4 Başarı Kriterleri

Projenin base model aşaması için belirlenen başarı kriterleri şunlardır:

- **Doğruluk (Accuracy):** Test setinde  $\geq 85\%$  doğruluk oranı.
- **Loss (Kayıp):** Eğitim ve doğrulama kaybı arasında aşırı uyum (overfitting) olmaması.
- **F1-Score:** Dengesiz veri dağılımına karşı ağırlıklı F1 skorunun tatmin edici düzeyde olması.

## 2 Proje Yönetimi

### 2.1 Zaman Çizelgesi (Milestones)

- **1. Hafta (20-27 Ekim):** GTSRB veri setinin incelenmesi, literatür taraması ve görev dağılımı.
- **2. Hafta (3-10 Kasım):** Veri ön işleme (boyutlandırma, normalizasyon) ve 'DataLoader' yapılarının kurulması.
- **3-4. Hafta:** Özgün Base Model (CNN) mimarilerinin tasarlanması ve PyTorch ile kodlanması.
  - Ramazan Bozkurt: Batch Norm ağırlıklı 3 katmanlı model tasarımı.
  - Ömer Utku Aktemur: Daha hafif (lightweight) 2 katmanlı model tasarımı.
  - Hiper parametre ayarları (Learning rate, Batch size denemeleri).
  - Modellerin karşılaştırılması, hata analizi ve raporlama.

## 2.2 Roller ve Sorumluluklar

- **Ramazan Bozkurt:** Veri setinin analizi ve 'SimpleBatchNormCNN' modelinin geliştirilmesi. Bu modelde Batch Normalization katmanlarının eğitim stabilitesine etkisi üzerine odaklanılmıştır. Ayrıca karmaşıklık matrisi görselleştirmeleri yapılmıştır.
- **Ömer Utku Aktemur:** 'SimpleCNN' modelinin geliştirilmesi. Daha az parametreye sahip, hesaplama maliyeti düşük ancak etkili bir model tasarımı (2 Konvolüsyon katmanı) üzerine çalışılmıştır. Veri düzenleme ve yükleme (DataLoader) modüllerinin optimizasyonu sağlanmıştır.

## 3 İlgili Çalışmalar (Literatür Taraması)

Trafik işareti tanıma konusunda literatürde "German Traffic Sign Recognition Benchmark (GTSRB)" veri seti üzerinde yapılan çalışmalar temel alınmıştır.

1. *Cireşan et al. (2012): "Multi-column Deep Neural Networks for Image Classification"* çalışmasında, birden fazla DNN'in birleştirilmesiyle insan üstü performans (
2. *Sermanet ve LeCun (2011): "Traffic Sign Recognition with Multi-Scale Convolutional Networks"* makalesinde çok ölçekli özelliklerin beslenmesi yöntemi kullanılmıştır.

**Farklılaşma Noktası:** Grup üyeleri olarak, hazır mimariler (ResNet, VGG) kullanmak yerine, problemin doğasına uygun, işlem gücü kısıtlı cihazlarda çalışabilecek iki farklı "Custom CNN" mimarisi tasarlayarak performanslarını kıyasladık.

## 4 Veri Açıklaması ve Yönetimi

### 4.1 Veri Kümesi (Dataset Description)

Projede **GTSRB (German Traffic Sign Recognition Benchmark)** veri seti kullanılmıştır. Veriler, Almanya'daki gerçek trafik koşullarında çekilmiş görüntülerden oluşmaktadır.

- **Kaynak:** Institut für Neuroinformatik, Ruhr-Universität Bochum.
- **Lisans:** Akademik kullanım için açıktır.

### 4.2 Boyut ve Şema

- **Sınıf Sayısı:** 43 farklı sınıf.
- **Veri Miktarı:** 39.000 Eğitim, 12.000 Test görüntüsü.
- **Ön İşleme:** Orijinalde farklı boyutlarda olan görüntüler 64x64 piksel boyutuna sabitlenmiş ve kanal bazlı normalizasyon (Mean: 0.5, Std: 0.5) uygulanmıştır.

## 5 Yöntemler ve Mimari

Her grup üyesi problemi çözmek için farklı bir Base Model yaklaşımı benimsemiştir. Aşağıda her bir modelin Evrişim (Convolution) ve Tam Bağlantılı (Fully Connected) katman yapıları detaylandırılmıştır.

### 5.1 Model A Tasarımı (Ramazan Bozkurt)

Tarafımdan geliştirilen **SimpleBatchNormCNN** modeli, kararlı bir eğitim süreci sağlamak amacıyla 3 konvolüsyon katmanına ve Batch Normalization yapısına sahiptir.

- **Evrişim Katmanları (Feature Extraction):**

1. Conv2d (32 filtre, 3x3) → BatchNorm → ReLU → MaxPool(2x2)
2. Conv2d (64 filtre, 3x3) → BatchNorm → ReLU → MaxPool(2x2)
3. Conv2d (128 filtre, 3x3) → BatchNorm → ReLU → MaxPool(2x2)

- **Tam Bağlantılı Katmanlar (Fully Connected Layers):**

1. Flatten (Düzleştirme)
2. Linear (128 nöron) → BatchNorm1d → ReLU
3. Dropout (0.5)
4. Linear (43 sınıf - Çıktı Katmanı)

- **Özellik:** Model, hem konvolüsyon hem de tam bağlantılı katmanlarda Batch Normalization kullanarak "Internal Covariate Shift" problemini azaltmayı hedefler.

### 5.2 Model B Tasarımı (Ömer Utku Aktemur)

Grup üyesi Ömer Utku tarafından geliştirilen **SimpleCNN** modeli, daha sade ve hızlı eğitilen bir yapı hedeflemiştir.

- **Evrişim Katmanları (Feature Extraction):**

1. Conv2d (16 filtre, 3x3) → ReLU → MaxPool(2x2)
2. Conv2d (32 filtre, 3x3) → ReLU → MaxPool(2x2)

- **Tam Bağlantılı Katmanlar (Fully Connected Layers):**

1. Flatten (Düzleştirme)
2. Linear (128 nöron) → ReLU
3. Linear (43 sınıf - Çıktı Katmanı)

- **Özellik:** Batch Normalization veya Dropout kullanılmamış, modelin ham özellik çıkarma yeteneğine ve hızına odaklanılmıştır.

## 6 Deney Tasarımı ve Sonuçlar

### 6.1 Deney Kurulumu

Her iki model de aynı veri seti üzerinde, benzer hiper parametrelerle (Epoch: 15, Batch: 32/64, Opt: Adam) eğitilmiştir.

## 6.2 Performans Karşılaştırması

Aşağıdaki tablo, grup üyelerinin geliştirdiği base modellerin test seti üzerindeki performanslarını özetlemektedir:

**Yorum:** Ramazan'ın geliştirdiği model (Model A), Batch Normalization ve ekstra derinliği sayesinde %98 gibi çok yüksek bir genel doğruluğa ulaşmıştır. Ömer Utku'nun modeli (Model B) ise daha basit yapısına rağmen %94 gibi oldukça başarılı bir sonuç elde etmiştir. Bu, veri setinin kalitesi ve "shallow" mimarilerin bile bu problem için etkili olabileceğini göstermektedir.

Model	Doğruluk (Acc)	Katman Sayısı	Özellikler
Ramazan (BN-CNN)	%98.0	3 Conv	Batch Norm + Dropout
Ömer Utku (SimpleCNN)	%94.0	2 Conv	Sade Mimari (Lightweight)

Tablo 1: Base Modellerin Karşılaştırılması

## 7 Kullanılan Araçlar

Python 3.8+, PyTorch, Pandas, NumPy, OpenCV, Matplotlib, Seaborn.

## 8 Kaynaklar

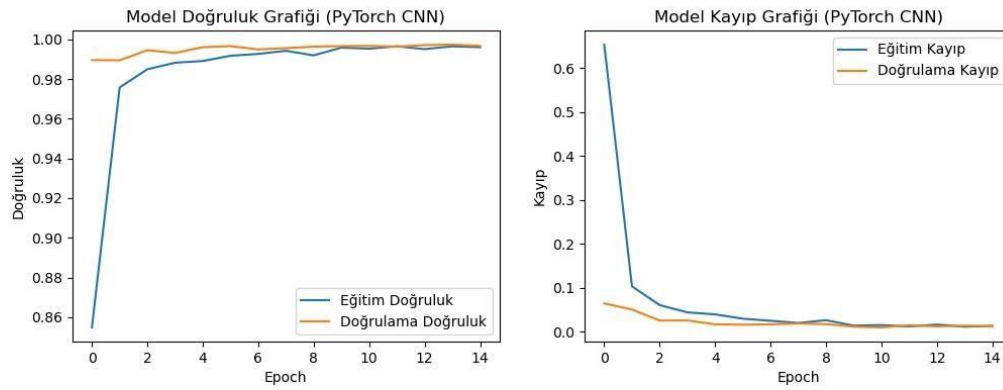
1. J. Stallkamp et al., "The German Traffic Sign Recognition Benchmark," IJCNN, 2011.
2. PyTorch Documentation: <https://pytorch.org/docs/stable/index.html>

**Ekler (Appendix)****Model A (Ramazan Bozkurt) - Performans Görselleri**

Class ID	Precision	Recall	F1-Score	Support
0	1.00	1.00	1.00	60
1	0.99	0.99	0.99	720
2	0.99	0.99	0.99	750
3	0.99	0.96	0.98	450
4	0.99	0.99	0.99	660
5	0.93	1.00	0.96	630
6	1.00	0.85	0.92	150
7	0.97	0.99	0.98	450
8	1.00	0.92	0.96	450
9	0.98	1.00	0.99	480
10	1.00	0.99	0.99	660
11	0.99	0.97	0.98	420
12	0.99	0.99	0.99	690
13	0.99	1.00	1.00	720
14	1.00	1.00	1.00	270
15	0.98	1.00	0.99	210
16	1.00	0.98	0.99	150
17	1.00	1.00	1.00	360
18	0.95	0.99	0.97	390
19	0.97	1.00	0.98	60
20	0.80	0.99	0.89	90
21	1.00	0.98	0.99	90
22	0.96	0.97	0.96	120
23	0.91	0.98	0.94	150
24	1.00	0.96	0.98	90
25	0.96	1.00	0.98	480
26	0.97	0.85	0.91	180
27	0.92	0.95	0.93	60
28	0.99	0.99	0.99	150

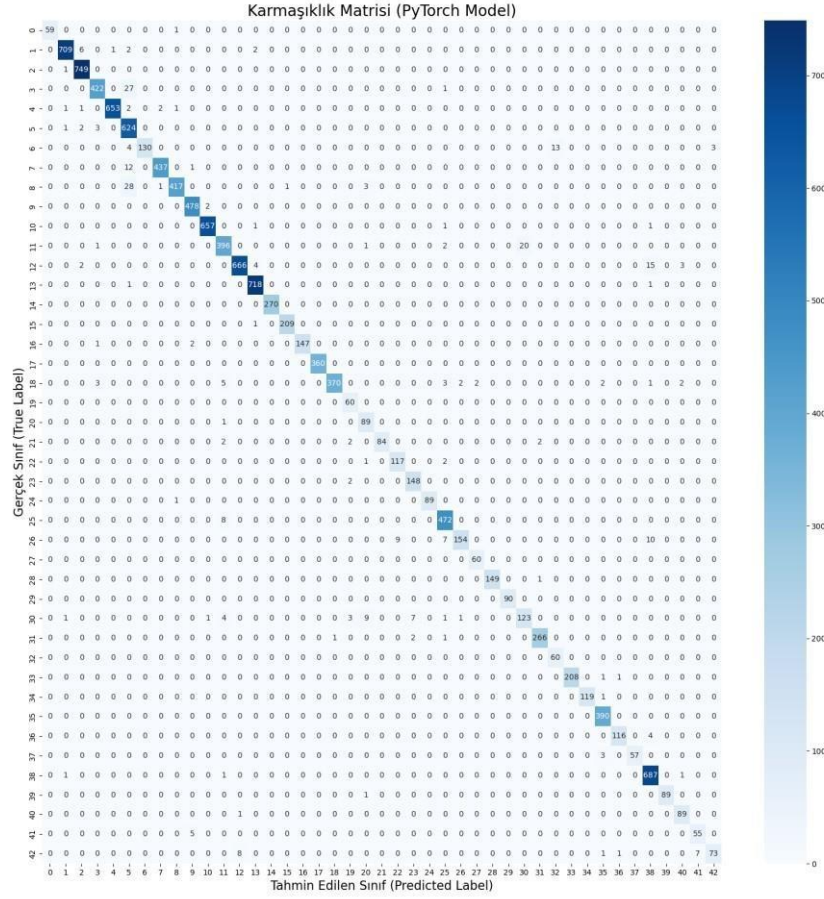
29	0.96	0.99	0.97	90
30	1.00	0.79	0.88	150
31	0.99	0.99	0.99	270
32	0.79	1.00	0.88	60
33	1.00	0.99	0.99	210
34	1.00	0.96	0.98	120
35	0.99	0.98	0.99	390
36	0.99	0.92	0.95	120
37	1.00	0.97	0.98	60
38	0.97	1.00	0.98	690
39	0.97	0.99	0.98	90
40	0.98	0.98	0.98	90
41	1.00	0.83	0.91	60
42	1.00	0.91	0.95	90
<b>Genel Doğruluk (Accuracy)</b>			<b>0.98</b>	<b>12630</b>
<b>Class ID</b>	<b>Precision</b>	<b>Recall</b>	<b>F1-Score</b>	<b>Support</b>
<b>Macro Avg</b>	<b>0.97</b>	<b>0.97</b>	<b>0.97</b>	<b>12630</b>
<b>Weighted Avg</b>	<b>0.98</b>	<b>0.98</b>	<b>0.98</b>	<b>12630</b>

Tablo 2: Model A Sınıf Bazlı Performans Metrikleri



Şekil 1: Model A (Ramazan Bozkurt) - Eğitim ve Doğrulama Başarım Grafikleri





Şekil 2: Model A (Ramazan Bozkurt) - Karmaşıklık Matrisi (Confusion Matrix)

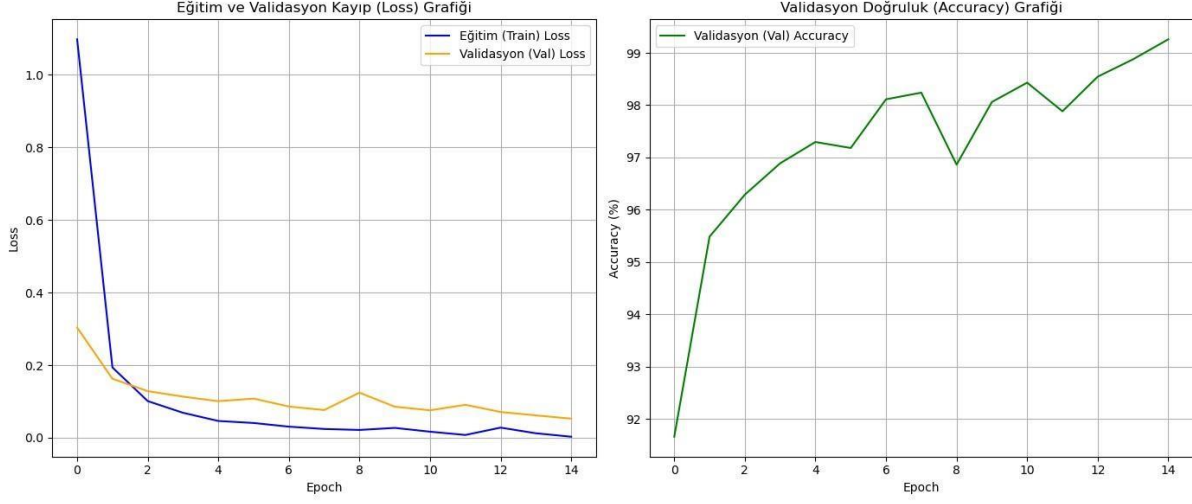
**Model B (Ömer Utku Aktemur) - Performans Görselleri**

Class ID	Precision	Recall	F1-Score	Support
0	1.00	0.57	0.72	60
1	0.93	0.96	0.94	720
2	0.90	0.98	0.94	750
3	0.93	0.93	0.93	450
4	0.99	0.94	0.96	660
5	0.89	0.91	0.90	630
6	0.93	0.85	0.89	150
7	0.96	0.88	0.92	450
8	0.89	0.96	0.92	450
9	0.96	0.95	0.95	480

10	0.97	0.98	0.98	660
11	0.93	0.93	0.93	420
12	0.99	0.97	0.98	690
13	0.98	0.99	0.99	720
14	1.00	1.00	1.00	270
15	0.96	0.98	0.97	210
16	0.95	0.97	0.96	150
17	1.00	0.99	0.99	360
18	1.00	0.82	0.90	390
19	0.58	0.97	0.72	60
20	0.77	0.91	0.83	90
21	0.94	0.70	0.80	90
22	0.96	0.98	0.97	120
23	0.75	0.69	0.72	150
24	0.82	0.86	0.84	90
25	0.92	0.94	0.93	480
26	0.88	0.82	0.85	180
27	0.77	0.90	0.83	60
28	0.90	0.93	0.92	150
29	0.86	0.83	0.85	90
30	0.74	0.65	0.69	150
31	0.85	0.94	0.89	270
32	0.75	1.00	0.86	60
33	0.97	0.99	0.98	210
34	0.99	0.99	0.99	120
35	0.99	0.98	0.99	390
36	0.93	0.95	0.94	120
37	1.00	0.97	0.98	60
38	0.98	0.96	0.97	690
39	0.97	0.93	0.95	90
40	0.87	0.92	0.90	90
41	0.92	0.73	0.81	60

42	0.93	0.96	0.95	90
<b>Genel Doğruluk (Accuracy)</b>			<b>0.94</b>	<b>12630</b>
<b>Macro Avg</b>	<b>0.91</b>	<b>0.91</b>	<b>0.91</b>	<b>12630</b>
<b>Weighted Avg</b>	<b>0.94</b>	<b>0.94</b>	<b>0.94</b>	<b>12630</b>

Tablo 3: Model B Sınıf Bazlı Performans Metrikleri



Şekil 3: Model B (Ömer Utku Aktemur) - Eğitim ve Doğrulama Başarım Grafikleri

