



**T.C.**  
**KONYA TEKNİK ÜNİVERSİTESİ**  
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**



**EVRIŞİMSEL SİNİR AĞLARI VE**  
**METİN BENZERLİĞİ KULLANILARAK**  
**FATURA GÖRÜNTÜLERİNDE**  
**SINIFLANDIRMA**

**Ömer ARSLAN**

**YÜKSEK LİSANS TEZİ**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Mart-2021**  
**KONYA**  
**Her Hakkı Saklıdır**

## TEZ KABUL VE ONAYI

Ömer ARSLAN tarafından hazırlanan “Evrişimsel Sinir Ağları ve Metin Benzerliği Kullanılarak Fatura Görüntülerinde Sınıflandırma” adlı tez çalışması 01/03/2021 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

### Jüri Üyeleri

#### Başkan

Prof. Dr. Erkan ÜLKER

#### Danışman

Dr. Öğr. Üyesi Sait Ali UYMAZ

#### Üye

Dr. Öğr. Üyesi Burak YILMAZ

### İmza

.....

.....

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan Kesen  
Enstitü Müdürü

## **TEZ BİLDİRİMİ**

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

## **DECLARATION PAGE**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ömer ARSLAN

Tarih: 01/03/2021

## ÖZET

### YÜKSEK LİSANS TEZİ

#### EVİRİŞİMSEL SİNİR AĞLARI VE METİN BENZERLİĞİ KULLANILARAK FATURA GÖRÜNTÜLERİNDE SINIFLANDIRMA

Ömer ARSLAN

Konya Teknik Üniversitesi  
Lisansüstü Eğitim Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Sait Ali UYMAZ

2021, 96 Sayfa

Jüri

Dr. Öğr. Üyesi Sait Ali UYMAZ  
Prof. Dr. Erkan ÜLKER  
Dr. Öğr. Üyesi Burak YILMAZ

Günümüzde tüm firmalar, çeşitli sebeplerle firma adına yapılan harcamalar için ödeme yapmakta ve ödemeye ait faturalar muhasebe işlemleri sebebiyle saklanmaktadır. Firma büyüdükçe personel sayısı artmakta olup firmanın çalışmakta olduğu tedarikçi firmaların sayısı da buna paralel olarak artmaktadır. Personel ve beraber çalışılan tedarikçi firma sayısının artmasıyla firma adına yapılan harcama miktarı da artmaktadır. Artan harcama miktarına bağlı olarak harcamalara ait fatura sayısı da büyük sayılara ulaşmaktadır. Fatura görüntüleri üzerindeki bilgilere daha sonra tekrar ulaşılabilmesi için üzerindeki bilgilerin saklanması gerekmektedir. Bu bilgilerin saklanması, yasal sebeplerden dolayı da gereklidir. Fiziksel olarak muhasebe birimlerine gönderilen faturalar, muhasebe destek personelleri tarafından dijital ortama aktarılmaktadır. Çok büyük firmalar çok daha fazla faturaya sahip olduklarından dolayı, faturaların dijital ortama aktarımı sırasında çok daha fazla personel gücü sarf edilmektedir. Personellerin devasa miktarda faturaları aktaramadıkları durumlarda ise daha fazla muhasebe personeli işe alınmaktadır. Ayrıca çok fazla faturanın kısıtlı bir zaman içerisinde girildiği zamanlarda giriş sırasında yapılan hata miktarı da artmakta olup düzeltmeler için ayrıca zaman ve iş gücü harcanmaktadır.

Bu tez çalışmasının amacı, firma bünyesinde çalışan personeller tarafından gerçekleştirilen fatura giriş işlemlerini otomatik bir hale getirmektir. Bu doğrultuda, derin öğrenme yöntemlerinden biri olan evrişimsel sinir ağları tabanlı bir mimari kurularak faturalar şablonlarına göre sınıflandırılmıştır. Sınıflandırma işlemi, fatura üzerinde yer alan fatura alanlarının konumlarına ve şablon yapısına göre yapılmıştır. Fatura şablonuna göre sınıflandırma yapıldıktan sonra şablonu tespit edilen faturaya OCR uygulanarak fatura üzerindeki karakterler otomatik olarak okunmuş ve doğal dil işleme birimine aktarılmıştır. Fatura görüntüsünden alınan verilerin hangi fatura alanına ait olduğunun tespiti ve fatura üzerindeki konumunun belirlenmesi için doğal dil işleme birimi kullanılmıştır. Minimum personel gücü gerektiren bu sistem sayesinde personellerin fatura girişi sırasında yapması muhtemel olan hata sayısı minimuma ineceğinden fatura giriş süreci çok daha kolay ve güvenli bir hale gelmiş olacaktır.

**Anahtar Kelimeler:** Derin Öğrenme, Doğal Dil İşleme, Fatura, Görüntü Sınıflandırma, Optik Karakter Tanıma, ESA

## **ABSTRACT**

## **MS THESIS**

# **CLASSIFICATION OF INVOICE IMAGES BY USING CONVOLUTIONAL NEURAL NETWORKS AND TEXT SIMILARITY**

**Ömer ARSLAN**

**Konya Technical University  
Institute of Graduate Studies  
Department of Computer Engineering**

**Advisor: Asst. Prof. Dr. Sait Ali UYMAZ**

**2021, 96 Pages**

**Jury  
Asst. Prof. Dr. Sait Ali UYMAZ  
Prof. Dr. Erkan ÜLKER  
Asst. Prof. Dr. Burak YILMAZ**

Today, all companies pay for the expenses made on behalf of the company for various reasons and the invoices for the payment are kept for accounting purposes. As the company grows, the number of personnel increases and the number of supplier companies that the company works with increases in parallel. With the increase in the number of personnel and supplier companies, the amount of expenditure on behalf of the company also increases. Depending on the increasing amount of expenditure, the number of invoices for expenditures also reaches large numbers. In order to access the information on the invoice images later, the information on the image must be stored. Storage of this information is also necessary for legal reasons. Invoices that are physically sent to the accounting units are transferred to the digital environment manually by the accounting support personnel. Since very large companies will have much more invoices, much more personnel power will be spent during the transfer of invoices to the digital environment. In cases where the staff cannot transfer huge amounts of invoices, more accounting personnel are recruited. In addition, when too many invoices are entered in a limited time, the amount of error made during entry also increases and additional time and effort is consumed for corrections.

The purpose of this thesis study is to automate the manual invoice entry processes performed by the personnel working within the company. Accordingly, an architecture based on convolutional neural networks, which is one of the deep learning methods, was established and invoices were classified according to their templates. The classification process is made according to the location of the invoice fields on the invoice and the template structure. After the invoice is classified according to the template, OCR is applied to the invoice whose template has been determined, and the characters on the invoice are automatically read and transferred to the natural language processing unit. Natural language processing unit was used to determine which invoice field belongs to the data taken from the invoice image and to determine its position on the invoice. With the new method that requires minimum personnel power, the number of possible mistakes that the personnel may make during entry will be minimized, and the invoice entry process will become much easier and safer.

**Anahtar Kelimeler:** Deep Learning, Natural Language Processing, Invoice, Image Classification, Optical Character Recognition, CNN

## ÖNSÖZ

Tez çalışmam ve yüksek lisans öğrenimim boyunca desteğini, bilgisini benden esirgemeyen danışman hocam Sayın Dr. Öğr. Üyesi Sait Ali UYMAZ' a ve ders dönemim boyunca desteklerini esirgemeyen Konya Teknik Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi Bilgisayar Mühendisliği Bölümü'ndeki saygıdeğer hocalarıma teşekkürlerimi sunarım.

Bu süreçte her zaman yanımda olup benden desteklerini esirgemeyen, beni motive eden annem ve babama sevgilerimi ve teşekkürlerimi sunarım.

Ömer ARSLAN  
KONYA-2021



## İÇİNDEKİLER

|                                                                       |      |
|-----------------------------------------------------------------------|------|
| ÖZET .....                                                            | iv   |
| ABSTRACT.....                                                         | v    |
| ÖNSÖZ .....                                                           | vi   |
| İÇİNDEKİLER .....                                                     | vii  |
| ŞEKİLLER LİSTESİ .....                                                | ix   |
| ÇİZELGELER LİSTESİ .....                                              | x    |
| SİMGELER VE KISALTMALAR .....                                         | xiii |
| 1. GİRİŞ .....                                                        | 1    |
| 2. KAYNAK ARAŞTIRMASI .....                                           | 4    |
| 3. MATERYAL VE YÖNTEM.....                                            | 10   |
| 3.1. Evrişimsel Sinir Ağları (CNN) ve Ana Bileşenleri .....           | 10   |
| 3.1.1. Evrişim (Konvolüsyon) katmanı.....                             | 10   |
| 3.1.2. Havuzlama katmanı .....                                        | 11   |
| 3.1.3. Tam bağlı katman .....                                         | 13   |
| 3.1.4. Aktivasyon katmanı .....                                       | 14   |
| 3.1.5. Optimizasyon algoritması .....                                 | 15   |
| 3.1.6. Dropout (Seyreltme) katmanı .....                              | 17   |
| 3.1.7. Sınıflandırma katmanı.....                                     | 18   |
| 3.2. Evrişimsel Sinir Ağları Mimarileri .....                         | 19   |
| 3.2.1. LeNet-5 mimarisi .....                                         | 19   |
| 3.2.2. VGG-16 mimarisi .....                                          | 21   |
| 3.2.3. VGG-19 mimarisi .....                                          | 22   |
| 3.2.4. AlexNet mimarisi.....                                          | 23   |
| 3.2.5. ZFNet mimarisi.....                                            | 24   |
| 3.2.6. GoogleNet (Inception-V1) mimarisi.....                         | 25   |
| 3.2.7. DenseNet121 mimarisi .....                                     | 26   |
| 3.2.8. MobileNetV2 mimarisi .....                                     | 28   |
| 3.3. Optik Karakter Tanıma .....                                      | 29   |
| 3.3.1. Optik karakter tanıma sistemlerindeki ana aşamalar .....       | 29   |
| 3.3.2. Optik karakter tanıma sistemlerinde karşılaşılan sorunlar..... | 32   |
| 3.4. Doğal Dil İşleme .....                                           | 33   |
| 3.4.1. Adlandırılmış varlık tanıma .....                              | 34   |
| 3.4.2. Metin benzerlik algoritmaları .....                            | 35   |
| 3.5. Veri Ön İşleme Teknikleri .....                                  | 39   |
| 3.5.1. Fatura verisi içermeyen görüntülerin ayıklanması.....          | 39   |
| 3.5.2. Eğik Metin İçeren Görüntülerin Düzeltilmesi .....              | 40   |
| 3.5.3. Sıfır dolgu uygulanması.....                                   | 42   |
| 3.6. Veri Arttırma.....                                               | 43   |

|                                                                        |            |
|------------------------------------------------------------------------|------------|
| 3.7. Öznitelik Çıkarma.....                                            | 44         |
| 3.8. Veri Setleri .....                                                | 46         |
| 3.8.1. BAFGV1 veri seti .....                                          | 46         |
| 3.8.2. BAFGV2 veri seti .....                                          | 49         |
| 3.8.3. BAFGV3 veri seti .....                                          | 50         |
| 3.9. Yapılan Eğitim ve Test Çalışmaları.....                           | 51         |
| 3.10. Fatura Görüntülerini Okuma ve Anlamlandırma Uygulaması .....     | 52         |
| 3.10.1. Sistem mimarisi .....                                          | 52         |
| 3.10.2. Sistem uygulaması .....                                        | 55         |
| <b>4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....</b>                         | <b>59</b>  |
| 4.1. BAFGV1 Veri Seti Üzerinde Yapılan Deneysel Çalışma Sonuçları..... | 60         |
| 4.1.1. LeNet-5 mimarisi kullanılarak elde edilen sonuçlar .....        | 61         |
| 4.1.2. VGG-16 mimarisi kullanılarak elde edilen sonuçlar .....         | 63         |
| 4.1.3. VGG-19 mimarisi kullanılarak elde edilen sonuçlar .....         | 65         |
| 4.1.4. DenseNet121 mimarisi kullanılarak elde edilen sonuçlar .....    | 67         |
| 4.1.5. MobileNetV2 mimarisi kullanılarak elde edilen sonuçlar.....     | 69         |
| 4.2. BAFGV2 Veri Seti Üzerinde Yapılan Deneysel Çalışma Sonuçları..... | 71         |
| 4.2.1. LeNet-5 mimarisi kullanılarak elde edilen sonuçlar .....        | 72         |
| 4.2.2. VGG-16 mimarisi kullanılarak elde edilen sonuçlar .....         | 74         |
| 4.2.3. VGG-19 mimarisi kullanılarak elde edilen sonuçlar .....         | 76         |
| 4.2.4. DenseNet121 mimarisi kullanılarak elde edilen sonuçlar .....    | 78         |
| 4.2.5. MobileNetV2 mimarisi kullanılarak elde edilen sonuçlar.....     | 80         |
| 4.3. BAFGV3 Veri Seti Üzerinde Yapılan Deneysel Çalışma Sonuçları..... | 82         |
| 4.3.1. LeNet-5 mimarisi kullanılarak elde edilen sonuçlar .....        | 83         |
| 4.3.2. VGG-16 mimarisi kullanılarak elde edilen sonuçlar .....         | 85         |
| 4.3.3. VGG-19 mimarisi kullanılarak elde edilen sonuçlar .....         | 87         |
| 4.3.4. DenseNet121 mimarisi kullanılarak elde edilen sonuçlar .....    | 89         |
| 4.3.5. MobileNetV2 mimarisi kullanılarak elde edilen sonuçlar.....     | 91         |
| <b>5. SONUÇLAR VE ÖNERİLER .....</b>                                   | <b>93</b>  |
| <b>ÖZGEÇMİŞ .....</b>                                                  | <b>105</b> |

## ŞEKİLLER LİSTESİ

|                                                                                                                                                                                                                                                                                                                                                           |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Şekil 3.1. Tek boyutlu evrişim işlemine ait şema. (a) Tek kanallı girdi; (b) Çok kanallı girdi (Jiao ve ark 2020).....                                                                                                                                                                                                                                    | 11 |
| Şekil 3.2. Maksimum havuzlama ve ortalama havuzlama işlemlerinin atlama sayısı 2 seçilerek uygulanması (Sze ve ark 2017) .....                                                                                                                                                                                                                            | 13 |
| Şekil 3.3. $c_i$ ve $y_i$ değerlerinin giriş ve çıkış özniteliklerini temsil ettiği durumda doğrusal olmayan ReLU fonksiyonuna ait grafik (Jiao ve ark 2020).....                                                                                                                                                                                         | 14 |
| Şekil 3.4. 32x32 boyutunda bir görüntüye uygulanan CNN mimarisinin genel yapısı..                                                                                                                                                                                                                                                                         | 19 |
| Şekil 3.5. LeNet-5 mimarisinin genel yapısı (LeCun ve ark 1998).....                                                                                                                                                                                                                                                                                      | 20 |
| Şekil 3.6. VGG mimarileri (a) VGG-16 (b) VGG-19 (Shaha ve Pawar 2018) .....                                                                                                                                                                                                                                                                               | 23 |
| Şekil 3.7. AlexNet mimarisi (Krizhevsky ve ark 2012) .....                                                                                                                                                                                                                                                                                                | 24 |
| Şekil 3.8. ZFNet mimarisi (Zeiler ve Fergus 2014) .....                                                                                                                                                                                                                                                                                                   | 25 |
| Şekil 3.9. Parçalama, dönüşüm ve birleştirme yapısını gösteren temel inception bloğu mimarisi (Khan ve ark 2020) .....                                                                                                                                                                                                                                    | 26 |
| Şekil 3.10. Büyüme oranı olarak $k = 4$ kullanan ve 5 Dense bloğu içeren bir DenseNet mimarisi (Huang ve ark 2017) .....                                                                                                                                                                                                                                  | 27 |
| Şekil 3.11. MobileNetV2 mimarisinin genel çalışma şeması (Toğaçar ve ark 2021)....                                                                                                                                                                                                                                                                        | 28 |
| Şekil 3.12. OCR sistemlerinde uygulanan ana işlem adımları (Awel ve Abidi 2019) ...                                                                                                                                                                                                                                                                       | 30 |
| Şekil 3.13. OCR sonrası anlamlı veriler içermediği için veri setinden çıkarılan örnek görüntüler.....                                                                                                                                                                                                                                                     | 39 |
| Şekil 3.14. Doğruya ait Hough uzayının temsili (Duda ve Hart 1972) .....                                                                                                                                                                                                                                                                                  | 41 |
| Şekil 3.15. Örnek bir fatura görüntüsüne Canny Operatörünün ve Hough Dönüşümünün uygulanması aşamaları. (a) Orijinal görüntü. (b) Siyah beyaz görüntü. (c) Canny operatörü uygulanmış görüntü. (d) Hough dönüşümü kullanılarak doğrular çizilmiş görüntü. (e) Doğruların eğim ortalaması alınarak elde edilen eğim açısına göre düzeltilmiş görüntü ..... | 42 |
| Şekil 3.16. Sunulan yöntemde bulunan veri ön işleme, öznitelik çıkarma ve sınıflandırma aşamaları.....                                                                                                                                                                                                                                                    | 45 |
| Şekil 3.17. BAFGV1 veri setinde bulunan her bir sınıfa ait görüntü örnekleri. (a) Kart dolum fişi. (b) Otopark fişi. (c) Taksi fişi. (d) Yemek fişi .....                                                                                                                                                                                                 | 47 |
| Şekil 3.18. KartDolum sınıfına ait görüntülerin en boy oranlarını gösteren histogram .                                                                                                                                                                                                                                                                    | 47 |
| Şekil 3.19. Otopark sınıfına ait görüntülerin en boy oranlarını gösteren histogram .....                                                                                                                                                                                                                                                                  | 48 |
| Şekil 3.20. Taksi sınıfına ait görüntülerin en boy oranlarını gösteren histogram .....                                                                                                                                                                                                                                                                    | 48 |
| Şekil 3.21. Yemek sınıfına ait görüntülerin en boy oranlarını gösteren histogram.....                                                                                                                                                                                                                                                                     | 49 |
| Şekil 3.22. BAFGV2 veri setinde bulunan örnek fatura görüntüleri. (a) KartDolum fişi (b) Otopark fişi (c) Taksi fişi (d) Yemek fişi.....                                                                                                                                                                                                                  | 50 |
| Şekil 3.23. Parlaklık tabanlı görüntü arttırma ile elde edilen örnek görüntüler. (a) Orijinal görüntü (b) Elde edilen ilk görüntü (c) Elde edilen ikinci görüntü .....                                                                                                                                                                                    | 51 |
| Şekil 3.24. Geliştirilen sisteme ait genel çalışma grafiği .....                                                                                                                                                                                                                                                                                          | 52 |
| Şekil 3.25. Yemek sınıfına ait örnek bir fatura görüntüsü.....                                                                                                                                                                                                                                                                                            | 53 |
| Şekil 3.26. Geliştirilen sistemin CNN, OCR ve doğal dil işleme aşamaları .....                                                                                                                                                                                                                                                                            | 54 |
| Şekil 3.27. Uygulamanın giriş ekranı .....                                                                                                                                                                                                                                                                                                                | 55 |
| Şekil 3.28. Fatura sınıflandırma ekranı .....                                                                                                                                                                                                                                                                                                             | 55 |
| Şekil 3.29. Fatura görüntü işleme ekranında Hough dönüşümü uygulaması .....                                                                                                                                                                                                                                                                               | 56 |
| Şekil 3.30. Fatura görüntü işleme ekranında mavi piksel ekleme uygulaması.....                                                                                                                                                                                                                                                                            | 57 |
| Şekil 3.31. Fatura alan tespit ekranı .....                                                                                                                                                                                                                                                                                                               | 57 |
| Şekil 3.32. Fatura alan tespiti ekranında örnek alan tespiti uygulaması.....                                                                                                                                                                                                                                                                              | 58 |

## ÇİZELGELER LİSTESİ

|                                                                                                                                                                            |    |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Çizelge 3.1.</b> OCR sistemlerinde kullanılan bazı önemli ön işleme teknikleri (Hamad ve Kaya 2016).....                                                                | 31 |
| <b>Çizelge 3.2.</b> Fatura sınıfları ve sınıf başına fatura görüntüsü sayıları .....                                                                                       | 46 |
| <b>Çizelge 3.3.</b> BAFGV3 veri setinde sınıf başına fatura görüntü sayıları .....                                                                                         | 51 |
| <b>Çizelge 4.1.</b> LeNet-5 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                            | 61 |
| <b>Çizelge 4.2.</b> LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                            | 61 |
| <b>Çizelge 4.3.</b> LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....   | 61 |
| <b>Çizelge 4.4.</b> LeNet-5 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                  | 62 |
| <b>Çizelge 4.5.</b> VGG-16 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları.....                                                                              | 63 |
| <b>Çizelge 4.6.</b> VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                             | 63 |
| <b>Çizelge 4.7.</b> VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....    | 63 |
| <b>Çizelge 4.8.</b> VGG-16 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                   | 64 |
| <b>Çizelge 4.9.</b> VGG-19 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları.....                                                                              | 65 |
| <b>Çizelge 4.10.</b> VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                            | 65 |
| <b>Çizelge 4.11.</b> VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....   | 65 |
| <b>Çizelge 4.12.</b> VGG-19 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                  | 66 |
| <b>Çizelge 4.13.</b> DenseNet121 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                       | 67 |
| <b>Çizelge 4.14.</b> DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                       | 67 |
| <b>Çizelge 4.15.</b> DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .. | 67 |
| <b>Çizelge 4.16.</b> DenseNet121 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                             | 68 |
| <b>Çizelge 4.17.</b> MobileNetV2 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                       | 69 |
| <b>Çizelge 4.18.</b> MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                       | 69 |
| <b>Çizelge 4.19.</b> MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .. | 69 |
| <b>Çizelge 4.20.</b> MobileNetV2 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                             | 70 |
| <b>Çizelge 4.21.</b> LeNet-5 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları ....                                                                            | 72 |
| <b>Çizelge 4.22.</b> LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                           | 72 |
| <b>Çizelge 4.23.</b> LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....  | 72 |
| <b>Çizelge 4.24.</b> LeNet-5 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                 | 73 |
| <b>Çizelge 4.25.</b> VGG-16 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları ....                                                                             | 74 |

|                                                                                                                                                                            |    |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Çizelge 4.26.</b> VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                            | 74 |
| <b>Çizelge 4.27.</b> VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....   | 74 |
| <b>Çizelge 4.28.</b> VGG-16 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                  | 75 |
| <b>Çizelge 4.29.</b> VGG-19 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları....                                                                              | 76 |
| <b>Çizelge 4.30.</b> VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                            | 76 |
| <b>Çizelge 4.31.</b> VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....   | 76 |
| <b>Çizelge 4.32.</b> VGG-19 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                  | 77 |
| <b>Çizelge 4.33.</b> DenseNet121 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                       | 78 |
| <b>Çizelge 4.34.</b> DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                       | 78 |
| <b>Çizelge 4.35.</b> DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .. | 78 |
| <b>Çizelge 4.36.</b> DenseNet121 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                             | 79 |
| <b>Çizelge 4.37.</b> MobileNetV2 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                       | 80 |
| <b>Çizelge 4.38.</b> MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                       | 80 |
| <b>Çizelge 4.39.</b> MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .. | 80 |
| <b>Çizelge 4.40.</b> MobileNetV2 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                             | 81 |
| <b>Çizelge 4.41.</b> LeNet-5 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları ....                                                                            | 83 |
| <b>Çizelge 4.42.</b> LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                           | 83 |
| <b>Çizelge 4.43.</b> LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....  | 83 |
| <b>Çizelge 4.44.</b> LeNet-5 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                 | 84 |
| <b>Çizelge 4.45.</b> VGG-16 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları....                                                                              | 85 |
| <b>Çizelge 4.46.</b> VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                            | 85 |
| <b>Çizelge 4.47.</b> VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....   | 85 |
| <b>Çizelge 4.48.</b> VGG-16 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                  | 86 |
| <b>Çizelge 4.49.</b> VGG-19 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları....                                                                              | 87 |
| <b>Çizelge 4.50.</b> VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                            | 87 |
| <b>Çizelge 4.51.</b> VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri .....   | 87 |
| <b>Çizelge 4.52.</b> VGG-19 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                                  | 88 |

|                                                                                                                                                                           |    |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| <b>Çizelge 4.53.</b> DenseNet121 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                      | 89 |
| <b>Çizelge 4.54.</b> DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                      | 89 |
| <b>Çizelge 4.55.</b> DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri.. | 89 |
| <b>Çizelge 4.56.</b> DenseNet121 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                            | 90 |
| <b>Çizelge 4.57.</b> MobileNetV2 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları .....                                                                      | 91 |
| <b>Çizelge 4.58.</b> MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi.....                                                      | 91 |
| <b>Çizelge 4.59.</b> MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri.. | 91 |
| <b>Çizelge 4.60.</b> MobileNetV2 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri .....                            | 92 |
| <b>Çizelge 5.1.</b> BAFGV1 veri seti üzerinde yapılan çalışmalara ait başarı oranları .....                                                                               | 93 |
| <b>Çizelge 5.2.</b> BAFGV2 veri seti üzerinde yapılan çalışmalara ait başarı oranları .....                                                                               | 94 |
| <b>Çizelge 5.3.</b> BAFGV3 veri seti üzerinde yapılan çalışmalara ait başarı oranları .....                                                                               | 95 |
| <b>Çizelge 5.4.</b> CNN mimarileri tarafından üretilmiş model boyutları .....                                                                                             | 95 |
| <b>Çizelge 5.5.</b> Model eğitimleri sırasında geçen dakika sayıları .....                                                                                                | 96 |

## SİMGELER VE KISALTMALAR

### Kısaltmalar

|               |                                                                                                       |
|---------------|-------------------------------------------------------------------------------------------------------|
| <b>CNN</b>    | : Evrişimsel Sinir Ağları (Convolutional Neural Network)                                              |
| <b>NLP</b>    | : Doğal Dil İşleme (Natural Language Processing)                                                      |
| <b>NER</b>    | : Adlandırılmış Varlık Tanıma (Named Entity Recognition)                                              |
| <b>OCR</b>    | : Optik Karakter Tanıma (Optical Character Recognition)                                               |
| <b>SGD</b>    | : Olasılıksal Gradyan İnişi (Stochastic Gradient Descent)                                             |
| <b>ReLU</b>   | : Düzleştirilmiş Doğrusal Birim (Rectified Linear Units)                                              |
| <b>SVM</b>    | : Destek Vektör Makineleri (Support Vector Machine)                                                   |
| <b>VGG</b>    | : Görsel Geometri Grubu (Visual Geometry Group)                                                       |
| <b>LM</b>     | : Levenshtein Matrisi (Levenshtein Matrix)                                                            |
| <b>TP</b>     | : Doğru Pozitif (True Positive)                                                                       |
| <b>TN</b>     | : Doğru Negatif (True Negative)                                                                       |
| <b>FP</b>     | : Yanlış Pozitif (False Positive)                                                                     |
| <b>FN</b>     | : Yanlış Negatif (False Negative)                                                                     |
| <b>BAFGV1</b> | : Bankadan Alınan Fatura Görüntüleri 1. Versiyon                                                      |
| <b>BAFGV2</b> | : Bankadan Alınan Fatura Görüntüleri 2. Versiyon                                                      |
| <b>BAFGV3</b> | : Bankadan Alınan Fatura Görüntüleri 3. Versiyon                                                      |
| <b>DNN</b>    | : Dekonvolüsyonel Sinir Ağı (Deconvolutional Neural Network)                                          |
| <b>ILSVRC</b> | : ImageNet Büyük Ölçekli Görüntü Tanıma Yarışması (ImageNet Large-Scale Visual Recognition Challenge) |

## 1. GİRİŞ

Firmaların çalışmalarına bağlı olarak her gün milyonlarca finansal işlem gerçekleşmektedir. Finansal işlemler, beraberinde banka çekleri ve faturalar gibi belgeleri getirmektedir. Hem güvenlik hem de zaman tasarrufu açısından bu belgelerin dijital ortama aktarılması ve sonrasında otomatik olarak işlenmesi çok önemlidir (Tang ve ark 1995). Banka işlemleri sonrasında oluşan finansal belgeler birbirine benzer şablonlara sahiptir. Dolayısıyla oluşan bu finansal belgeleri otomatik olarak tanıyabilecek bir sistemin kurulması mümkün olup kurulan sistem sayesinde doküman işleme süreci daha kolay ve güvenli olmaktadır.

Dokümanların işlenerek üzerinde yer alan gerekli bilgilerin alınması çok zaman alabilen bir işlemdir. Örnek olarak bir firma çalışanın ortalama bir doküman üzerinde yer alan bilgileri bilgisayara girmesi yaklaşık 2 dolara mâl olmaktadır (Casey ve ark 1992). Büyük firmalar gün içerisinde binlerce finansal doküman işlemektedir. Dokümanların üzerinde yer alan bilgiler firma çalışanları tarafından tek tek dijital ortama aktarıldığından bu işlem firmalara çok yüksek personel maliyeti getirmektedir. Dolayısıyla doküman üzerinde yer alan bilgilerin otomatik olarak dijital ortama aktarılması durumunda özellikle büyük firmalar çok yüksek bir personel maliyetinden kurtulmaktadır.

Günümüzde yüksek işlemci gücüne ve yüksek depolama kapasitesine sahip bilgisayarlara erişim çok kolay bir hale gelmiştir. Dokümanların fiziksel olarak saklanması hem maliyetli hem de güvenliği az bir yöntemdir. Dokümanların fiziksel olarak saklanması yerine dijital ortamda saklanması, günümüz teknolojisiyle çok daha kolay ve güvenli bir hale gelmiştir. Dijital ortamda saklanan dokümanlara sonradan erişim de fiziksel olarak saklamaya göre çok daha kolay olmaktadır.

Normal şartlarda her dokümanın kendine has bir şablonu olsa da birden farklı şablona sahip dokümanları gruplar halinde işleyerek girişlerini yapmak hâlâ zorlu bir süreçtir. Çünkü bu işlem sırasında her bir doküman için ayrı ayrı şablon tespiti yapılmalı ve daha sonra üzerindeki gerekli bilgiler alınmalıdır. Bu konuda yapılan bir çalışmada (Taylor ve ark 1992), özellikle üzerinde barkod gibi bir tanımlayıcı olmayan yazdırılmış dokümanların yönetimi için bir sistem geliştirilmiştir. Kurulan sistem; dokümanın taranması, görüntünün zenginleştirilmesi, doküman şablonunun tespiti, dokümanda gerekli bilgi içeren bölgelerin tespiti ve son olarak yorumlama aşamalarından oluşmaktadır (Taylor ve ark 1992).

Bilgisayar teknolojilerinde çok önemli gelişmeler olmasına rağmen dokümanların otomatik olarak işlenmesi konusunda az sayıda çalışma yapılmış olup hâlâ fazla miktarda insan gücü harcanmaktadır (Tang ve ark 1995).

Önerilen çalışma kapsamında personeller tarafından gerçekleştirilen fatura giriş işlemleri otomatik hale getirilmiştir. Geliştirilen sistem sayesinde harcanan personel gücü minimuma indirilmekte olup fatura girişi sırasında yapılması muhtemel olan kullanıcı hatalarının da önüne geçilmektedir.

Bu çalışmada, finansal birer belge olan fatura görüntülerinin sınıflandırılması sağlayan CNN tabanlı bir sistem sunulmuştur. Bir banka veri tabanından alınan fatura görüntüleriyle veri seti oluşturulmuştur. Çalışmada beş farklı CNN mimarisi olan LeNet5 (LeCun ve ark 1998), VGG16 (Simonyan ve Zisserman 2014), VGG19 (Simonyan ve Zisserman 2014), DenseNet121 (Huang ve ark 2017) ve MobileNetV2 (Sandler ve ark 2018) mimarileri kullanılmıştır. Kullanılan CNN mimarileri bankadan alınan fatura veri seti üzerinde karşılaştırılmıştır. Ayrıca fatura sınıflandırma çalışmalarında görüntülerin en-boy oranlarının CNN sınıflandırma performansına olan etkisi araştırılmıştır. Çalışma kapsamında fatura görüntülerine görüntü işleme teknikleri uygulanarak üç farklı veri seti oluşturulmuştur. Oluşturulan veri setleri üzerinde CNN mimarileri ayrı ayrı karşılaştırılmıştır.

Bu çalışma üç ana aşamadan oluşmaktadır. İlk aşamada Evrimsel Sinir Ağları (CNN) mimarisi bir öznitelik çıkarıcı olarak kullanılarak fatura şablonlarına ait özniteliklerin çıkarılması sağlanmaktadır. Öznitelikler, tamamen fatura görüntüsünden çıkarılmaktadır. CNN mimarisi; faturanın şablonuna, varsa üzerinde yer alan görsellere, satır ve sütun yapısına, varsa faturadaki tablo yapısına bakarak öznitelik çıkarmaktadır. Klasik makine öğrenmesi algoritmalarında öznitelikler kullanıcı tarafından sisteme tanıtılır. Özniteliklerin çok belirgin olduğu durumlarda klasik makine öğrenmesi algoritmaları yeterli olmaktadır. Ancak özniteliklerin belirgin olmadığı durumlarda derin öğrenme algoritmaları kullanılarak otomatik öznitelik çıkarma işlemi gerçekleştirilir (Chartrand ve ark 2017). Çalışmada kullanılan fatura görüntüleri veri setinde de çok fazla türde fatura tipi olduğundan ve öznitelikler belirgin olmadığından CNN kullanılarak otomatik öznitelik çıkarma işlemi gerçekleştirilmiştir.

Çalışmanın ikinci aşamasında sınıflandırma işlemi yapılmıştır. Çalışma kapsamında sınıflandırılacak ikiden fazla fatura tipi olduğu için, öznitelikleri çıkarılan faturaların şablonlarına göre sınıflandırılması işlemi Softmax aktivasyon fonksiyonu kullanılarak gerçekleştirilmiştir. Evrimsel Sinir Ağları mimarisinin çıkışında bulunan

Softmax sınıflandırma katmanı, çıktı olarak ilgili faturanın hangi şablona sahip olduğuna karar vermektedir. Çalışmanın son aşamasında ise Doğal Dil İşleme destekli bir Optik Karakter Tanıma sistemi kullanılarak fatura görüntüsü üzerinde gerekli alanların tespit edilerek dijital ortama aktarılması sağlanmıştır.

Yapılan çalışmada kullanılan fatura görüntüleri Türkiye’de faaliyet gösteren bir bankadan alınmış olup bütün faturalar Türkçe veri içermektedir. Dolayısıyla geliştirilen fatura uygulaması sadece Türkiye’de kullanılmakta olan Türkçe faturalarda çalışmaktadır.

Geliştirilen fatura sınıflandırma çalışmasında otopark, kart dolum, yemek ve taksi olmak üzere 4 farklı sınıfta fatura görüntüsü kullanılmıştır. Fatura sınıflandırma işlemi bütün sınıflarda gerçekleşmektedir. Ancak fatura alanlarının görüntü üzerinde tespit edilmesi işlemi sadece otopark, kart dolum ve yemek faturalarında gerçekleştirilmiştir. Taksi fişlerinde bilgisayar puntosuyla yazılı fatura alanları olduğu gibi el yazısı fatura alanları da bulunmaktadır. Geliştirilen fatura uygulamasının taksi fişlerindeki el yazısı fatura alanlarını da okuyabilmesi için geliştirme yapılmayıp ileriki sürümlerinde bu geliştirme eklenecektir.

Özetle, yapılan çalışmada fatura görüntülerinin şablonlarına ve üzerindeki fatura alanlarının konumlarına göre sınıflandırılması için evrişimsel sinir ağları mimarisi kullanılmıştır. Fatura şablonlarına ve fatura alanlarının konumlarına göre öznitelik çıkarımı yapan evrişimsel sinir ağı mimarisi, çıktı olarak fatura görüntüsünün hangi şablona ait olduğunu vermiştir. Doğal dil işleme adımı ise, OCR ile fatura görüntüsünden alınan verilerin hangi fatura alanlarına ait olduğu tespit edilmiş ve fatura görüntüsü üzerindeki konumları belirlenmiştir.

## 2. KAYNAK ARAŞTIRMASI

Yapay zekâ, 1950’lerde ortaya çıkan yeni bir bilimdir. Bu bilim dalı sadece teknoloji üzerinde çalışmaz aynı zamanda ilişkili teknolojiyi ürünlere uygulayarak akıllı ürünler üretmektedir. Yapay zekâ disiplinler arası bir alan olup matematik, istatistik gibi alanlarla beraber çalışmaktadır (Zhang ve Dahu 2019). Yapay Zekâ, insan zekâsı gerektiren işlemlerin bilgisayarlar tarafından yapılabilmesi yeteneğini tanımlar. Yapay zekânın bir alt dalı olan makine öğrenmesi, bilgisayarı kullanarak sınıflar arasında ayırım yapmayı öğrenebilen bir sistemdir (Enamandram ve ark 2020). Makine öğrenmesi algoritmaları sayesinde bilgisayarlar verileri kullanarak öğrenebilmekte ve sınıflandırma yapabilmektedir. Klasik makine öğrenmesinde, bir uzman tarafından öznitelikler tanıtılır ve sistemin verileri kullanarak bu öznitelikleri öğrenmesi sağlanır. Özniteliklerin belirgin olduğu durumlarda uzmanlar tarafından özniteliklerin belirlenmesi mümkündür ancak her zaman özniteliklerin belirlenmesi kolay olmamaktadır. Klasik makine öğrenmesi algoritmalarının yetersiz kaldığı bu gibi durumlarda derin öğrenme algoritmaları devreye girmektedir. Derin öğrenme algoritmaları, sınıflandırma yaparken gerekli olan öznitelikler kümesini otomatik olarak veriden çıkaran algoritmalar (Chartrand ve ark 2017).

Derin öğrenme sistemleri, yapay sinir ağı mimarisi kullanarak gerekli özniteliklerin kodlanmasını sağlar. Bu sistemler, birbirine bağlı düğümlerden oluşmakta olup biyolojik sinir ağlarından esinlenerek kurulmuştur. Sinir ağlarının 1950’lere dayanan uzun bir tarihçesi vardır (Rosenblatt 1958). Sonrasında 1980 yılında geri yayılım isimli sistematik bir metot geliştirilmiştir. Bu çalışmada geliştirilen sistemin; ikili sistemdeki sayıları çarpması, Tic-Tac-Toe oyununu oynaması, yatay ve dikey çizgiler arasında ayırım yapabilmesi ve karakterleri tanıyabilmesi için geri yayılım algoritması üzerinde çalışılmıştır (Rumelhart ve ark 1985). Son yıllarda etiketli veri miktarındaki büyük artış, paralel hesaplama yeteneğine sahip bilgisayarların ulaşılabilir olması gibi sebeplerden dolayı büyük sinir ağlarının eğitilmesi yaygınlaşmıştır. Özellikle resim işleme çalışmalarında kullanılmak üzere Evrişimsel Sinir Ağları isimli derin sinir ağları kullanılmaya başlanmıştır (Enamandram ve ark 2020).

Evrişimsel Sinir Ağları(Convolutional Neural Networks), canlıların görme mekanizmasından esinlenerek kurulan derin öğrenme mimarileridir (Gu ve ark 2018). Derin ağlar kategorisine giren CNN mimarileri, özellikle görüntü sınıflandırma ve örüntü tanıma konusunda çok popüler olup (Liu ve ark 2015) birçok uygulaması bulunmaktadır.

Gazete sayfalarını anlamsal olarak bölütlemeyi amaçlayan bir çalışmada; gazete sayfaları üzerinde yer alan makale, reklam ve sayfa başlıkları gibi bölümlerin otomatik olarak sınıflandırılması ve etiketlenmesi sağlanmıştır. Gazete bölümlerinin sınıflandırılmasında sadece ilgili bölümlerin görsel öznitelikleri kullanılmıştır ve işlem öncesinde gazete sayfası üzerinde herhangi bir önilem yapılmamıştır. Çalışmada mask R-CNN isimli derin öğrenme bölütleme(Segmentation) metodu kullanılmıştır (Almutairi ve Almashan 2019). Mask R-CNN; görüntü içerisinde yer alan nesneleri tespit edebilen ve tespit edilen nesne üzerinde yüksek kaliteli bir maske oluşturan bir yaklaşımdır (He ve ark 2017). Mask R-CNN modelinin eğitimi kolay olup Faster R-CNN modelinin (Ren ve ark 2015) düzenlenmiş halidir. Mask R-CNN, temelde R-CNN isimli bölge tabanlı (Region Based) CNN yaklaşımını (Girshick ve ark 2014) kullanmaktadır. R-CNN yaklaşımında, resim içerisinde öncelikle aranan nesnelerin geçebileceği aday bölgeler tespit edilmekte ve etraflarına kutucuk çizilmektedir. Aday bölgeler daha sonra CNN mimarisi kullanılarak değerlendirilmektedir (He ve ark 2017).

Derin Yüz(Deep Face) isimli yüz tanıma sisteminde insan yüzüne ait derin özniteliklerin çıkarılması aşamasında SENet (Hu ve ark 2018), ResNet (He ve ark 2016), GoogleNet (Szegedy ve ark 2015), VGGNet (Simonyan ve Zisserman 2014) ve AlexNet (Krizhevsky ve ark 2012) isimli CNN mimarileri yaygın olarak kullanılmıştır. Derin Yüz Tanıma sisteminde öncelikle görüntü içerisindeki insan yüzü tespit edilmekte ve yüzün bulunduğu koordinatlara göre görüntü ayarlanmaktadır. Daha sonra tespit edilen yüzün gerçek veya sahte olduğuna karar veren yüz aldatmasını önleme (Face Anti-Spoofing) aşaması gerçekleştirilmektedir. Sistemin eğitimi sırasında farklı mimariler ve kayıp fonksiyonları kullanılmıştır (Wang ve Deng 2018).

Düşük kalitedeki trafik kameralarının çektiği resimlerdeki araçları tespit eden ve sınıflandıran bir çalışmada CNN mimarisinin performansı ölçülmüştür. Çalışmada toplam dört farklı deney yapılmıştır. İlk iki deneyde CNN mimarisindeki filtre sayısı ve boyutunun ağ performansına olan etkisi gözlemlenmiştir. Diğer iki deneyde ise ağ performansını arttırmak için sırasıyla özellik çıkarıcı katman ve sınıflandırma katmanına derin öğrenmenin dâhil edilme olasılıkları ölçülmüştür. Test edilen her senaryoya ait ortalama başarı oranları kaydedilmiştir. Filtre boyutunun ağ performansına etkisinin araştırıldığı ilk deneyde filtre boyutu 3x3'den 11x11 boyutuna kadar arttırılmasına rağmen ağ performansı sadece %94.76'dan %95'e yükselmiştir. Bu da filtre boyutunun ağ performansını belirgin olarak etkilemediğini göstermiştir. Filtre sayısının test edildiği ikinci deneyde filtre sayısı 20'den 50'ye kadar yükseltile de ağ performansı sadece

%94.72'den %94.78'e yükselmiştir. Ağ derinliğinin test edildiği üçüncü deneyde ise etap sayısı 3 iken, 4 ve 5 olmasına göre daha iyi bir başarı elde etmiştir. Gizli katman sayısının test edildiği son deneyde ise hiç gizli katman yokken %94.72 başarı elde eden sistem 3 gizli katman varken %95.82 başarı elde etmiştir. Sonuç olarak kurulan sistem ortalama %94.72 başarı elde etmiştir (Bautista ve ark 2016).

Görüntü sınıflandırma alanında yaygın olarak kullanılan CNN mimarileri, belge sınıflandırma çalışmalarında da kullanılmıştır. DeepDocClassifier isimli belge sınıflandırma çalışmasında, içerisinde milyonlarca örnek görüntünün bulunduğu ImageNet veri seti kullanılarak CNN mimarisi tabanlı bir belge sınıflandırma sistemi sunulmuştur. Sistem; 3x227x227 boyutunda doküman görüntüleri almaktadır. 4 evrişim ve 3 tam bağlı katmandan oluşan sistem toplamda 7 tane eğitilebilen katmana sahiptir ve AlexNet (Krizhevsky ve ark 2012) mimarisinden esinlenerek kurulmuştur. Sunulan sistem, daha önce sunulmuş olan CNN tabanlı bir yaklaşımla (Kang ve ark 2014) karşılaştırılmıştır. Sınıf başına sadece 20 görüntü ile yapılan sınıflandırma çalışmasında %68.25 başarı elde edilmiştir. Tobbacoo-3428 veri seti üzerinde elde edilen sonuçlara göre sunulan sistem, sınıf başına 100 görüntü ile yapılan eğitim ve validasyonda ortalama %77.6 başarı elde etmiştir. Daha önce sunulmuş CNN tabanlı yaklaşım (Kang ve ark 2014) ise ortalama %65.35 başarı elde edebilmiştir (Afzal ve ark 2015).

Doküman görüntüleri yapısal olarak benzer özelliklere sahiptir. Önceki yaklaşımlar doküman yapısını tespit etmek için el-yapımı öznitelikler kullanmakta iken, dokümana ait ham görüntüdeki piksellerden öğrenebilen CNN tabanlı bir sistem geliştirilmiştir. Doküman görüntülerinin doğasında bulunan hiyerarşik sayfa yapısı sebebiyle CNN kullanımı uygun görülmüştür. Kurulan sistemde 2 evrişim katmanına sahip bir CNN mimarisi kullanılmıştır. Doküman görüntüleri genelde 2000x2000 boyutundan yüksektir ve bu da CNN ağını eğitmek için çok büyüktür. Bu sebeple görüntüler öncelikle Çift Doğrusal Ara Değerleme(Bilinear Interpolasyon) kullanılarak 150x150 boyutuna getirilmiştir. Sistemin eğitim süresinin hızını arttırmak için Rectified Linear Units (ReLU) (Nair ve Hinton 2010) isimli aktivasyon fonksiyonu kullanılmıştır. Ağın ezberlemesinin önlenmesi için ise Dropout (Hinton ve ark 2012) kullanılmıştır. Sunulan sistem, içerisinde çok fazla alt bölüm çeşitliliği içeren dokümanlarda bile başarılı olmuştur. 10 sınıf içeren Tobacco veri setinde ortalama doğruluk değeri %65.37 olmuştur. Ağ başarısı; 100 rastgele eğitim, validasyon ve test işlemi göz önünde bulundurularak elde edilmiştir. 20 sınıflı NIST Tax-Form veri setinde yapılan çalışmada validasyon veri seti kullanılmamıştır ve eğitim 50 Epoch sürmüş olup ortalama %100 başarı elde

edilmiştir. Bu kadar yüksek başarı elde edilmesinin sebebi, NIST veri setinde aynı sınıfta yer alan Tax-Form görüntülerinin birbirine yapısal olarak çok benzemesi ve farklı sınıflar arası benzerliklerinin nispeten düşük olmasıdır (Kang ve ark 2014).

CNN mimarileri doğrudan fatura sınıflandırma çalışmalarında da kullanılmıştır. Fatura görüntülerini; el yazımı, bilgisayar çıktısı ve fiş olarak üç gruba otomatik olarak sınıflandıran bir çalışmada özellik çıkarıcı olarak AlexNet (Krizhevsky ve ark 2012) isimli Evrimsel sinir ağı mimarisi kullanılmıştır. Çalışmada öznitelik çıkarımından sonra sınıflandırma aşamasında K En Yakın Komşu (KNN) (Brown 2017), Random Forests (Breiman 2001), Naive Bayes gibi çeşitli makine öğrenmesi algoritmaları kullanılmıştır. Sistemin değerlendirilmesi aşamasında ayrıca farklı Çapraz Doğrulama(Cross-Validation) yaklaşımları da uygulanmıştır. Sistemde veri seti olarak; 220 tane el yazımı, 90 tane fiş ve 1070 tane bilgisayar çıktısı olmak üzere toplamda 1380 fatura görüntüsü kullanılmıştır. Çalışma esnasında 5-katman, 10-katman Çapraz Doğrulama ve verinin %66 / %34 eğitim/test olarak parçalanması yaklaşımları uygulanmış olup çalışmalar KNN, Random Forests ve Naive Bayes sınıflandırıcıları için ayrı ayrı gerçekleştirilmiştir. CNN tabanlı olarak kurulan bu sistemde 10-katman çapraz doğrulama yaklaşımı ile sınıflandırıcı olarak KNN kullanılması durumunda %98.4 ile en yüksek başarı elde edilmiştir. Elde edilen bu yüksek başarı, kurulan sistemin OCR sistemlerinde önileme aşamalarında kullanılmasını mümkün kılmıştır (Tarawneh ve ark 2019).

Fatura tanıma ve sınıflandırma çalışmalarında, çalışma sonrasında fatura üzerinde yer alan bilgilerin tanınarak bilgisayar ortamına aktarılması gerekmektedir. Bu durumda görüntü kalitesinin düşük olması ağ performansını olumsuz etkilemektedir. CNN tabanlı olarak kurulan bir sistemde doküman görüntülerinin çözünürlüklerini arttırmak için yeni bir metot sunulmuştur. Geleneksel interpolasyon yöntemlerini alarak her birinden en iyi öznitelikleri seçerek öğrenebilen bu sistemde, öğrenilen bu özniteliklerden sonuç olarak yüksek çözünürlüklü görüntüler elde edilmiştir. Sistemde kullanılan CNN mimarisi, çeşitli interpolasyon tekniklerinin çıktılarının ağırlıklı doğrusal-olmayan birleşimlerini kullanmakta ve düşük çözünürlüklü görüntü parçasından yüksek çözünürlüklü bir görüntü parçası öğrenmektedir. Geliştirilen bu metot ayrıca geleneksel interpolasyon yöntemlerine göre daha yüksek bir hesaplama maliyeti getirmemektedir. Çalışmada düşük kaliteli doküman kalitesinin 2 ve 4 kat artırılması sağlanmıştır. 2 kat arttırma yönteminde 16x16 boyutunda verilen görüntü 3 katmanlı bir CNN mimarisi ve çeşitli geleneksel interpolasyon yöntemleri kullanılarak 32x32

boyutuna getirilmiştir. 4 kat arttırma yönteminde de aynı sistem kullanılmış olup 16x16 boyutundaki görüntü 64x64 boyutuna getirilmiştir. Çalışma sonucunda görüntünün 2 kat büyütüldüğü durumda OCR kalitesinde %54 başarı artışı sağlanırken 4 kat büyütme durumunda ise %33 iyileşme sağlanmıştır (Pandey ve ark 2017).

Derin Siamese Evrişimsel Sinir Ağı(CNN) (Chopra ve ark 2005) kullanılan bir çalışmada CNN ve destek vektör makinesi tabanlı genel bir OCR sistemi sunulmuştur. Sunulan çalışma aslında Arapça dokümanlara odaklansa da farklı dillere de uygulanabilir özelliktedir. Sunulan çalışmanın asıl amacı minimum eğitim gerektiren genel bir OCR sistemi kurmaktır. Kurulan sistem iki parçadan oluşmaktadır. Birinci parça öznitelik çıkarma parçası olup Siamese CNN tabanlı olarak kurulmuştur. Çalışma sonucunda MNIST veri seti kullanılarak sınıflandırıcı olarak KNN kullanılması durumunda %79.08 OCR başarısı elde edilirken SVM kullanılması durumunda %82.70 OCR karakter tanıma başarısı elde edilmiştir (Sokar ve ark 2018).

Bölütleme işlemi içermeyen bir OCR sisteminin sunulduğu bir çalışmada sentetik eğitim verisi üretimi ve veri arttırma teknikleri kullanılmıştır (Namysl ve Konya 2019). Tekrarlı geri yayılım algoritması kullanılarak bilgilerin uzun süreli depolanması için öğrenilmesi işlemi çok uzun zaman almaktadır. Bu sorunun çözülmesi amacıyla Long Short-Term Memory (LSTM) (Hochreiter ve Schmidhuber 1997) isimli metot geliştirilmiştir. Kurulan OCR sisteminde CNN ve LSTM yaklaşımlarını beraber bulunduran hibrit modeller geliştirildiği gibi sadece CNN içeren modeller de geliştirilmiştir. Çalışmada metin görüntülerinden özniteliklerin çıkarılmasında 10 tane evrişim katmanı içeren bir evrişimsel sinir ağı mimarisi kullanılmıştır. Sistemin eğitim aşamasında optimizasyon yöntemi olarak Adam (Kingma ve Ba 2014) kullanılmıştır. Tamamen CNN mimarisi kullanarak çalışan modellerde her bir evrişim bloğundan sonra BatchNormalization (Ioffe ve Szegedy 2015) uygulanarak eğitim hızı arttırılmıştır. Çalışmada kullanılan hibrit modeller yaklaşık 300 Epoch ile eğitilmiş olup tamamen CNN tabanlı modeller 500 Epoch ile eğitilmiştir. Geliştirilen sistem; karmaşık/bozuk metin örneklerinde, önde gelen ticari OCR sistemlerinden olan ABBYY FineReader ve açık kaynak kodlu OCR sistemlerinden olan Tesseract (Smith 2007) sisteminin karakter tanıma başarısını geride bırakmıştır. 11 adet taranmış güncel Almanca faturalara ait görüntüler üzerinde yapılan testlerde ABBY FineReader isimli ticari OCR sistemi % 2.5 hata ile çalışırken Tesseract sistemi %6.11 hata ile çalışmıştır. Çalışmada sunulan ve tamamen CNN ile çalışan sistem %1.66 hata oranı ile çalışmasını tamamlamıştır. Hibrit

model ise %1.63 hata oranı ile fatura görüntüleri üzerindeki çalışmasını tamamlamıştır (Namysl ve Konya 2019).

InceptionV3 isimli CNN mimarisi kullanılarak geliştirilen bir sistemde OCR sisteminin eğitilmesi ve test edilmesi CNN mimarisi ile gerçekleştirilmiştir. InceptionV3 isimli CNN mimarisi 53.342 tane gürültülü karakter görüntüsü üzerinde eğitilmiş ve burada kullanılan görüntüler, fiş ve gazete görüntülerinden derlenmiştir. Çalışmada önceden eğitilmiş olan InceptionV3 mimarisi kullanılarak Öğrenme Aktarması(Transfer Learning) uygulanmış ve eğitim süresi kısaltılmıştır. Sunulan çalışmanın karakter tanıma performansı a9t9 isimli herkese açık OCR sistemi ile karşılaştırılmıştır. Geliştirilen sistem a9t9 sisteminde bulunan hata oranını %21.5 oranında azaltmış ve düşük kaliteli metin görüntülerinde ortalama %78 başarı elde etmiştir. Düşük kaliteli görüntülerde a9t9 OCR sistemi ise ortalama %56 başarı elde etmiştir. Sunulan OCR sistemi yüksek kaliteli görüntülerde ise %90.6 karakter tanıma başarıları elde etmiştir (Wei ve ark 2018).

OCR performansını arttırmak için kurulan bir sistemde, sistem tarafından otomatik olarak uygun doküman önışleme algoritmalarının seçilmesi sağlanmıştır. Çalışmada görüntü önışleme algoritmalarının bozuk görüntülü veriler üzerindeki OCR performansı üzerindeki etkileri araştırılmıştır. Sonuç olarak önışleme algoritmasının başarısının OCR sisteminin doğasına ve görüntüdeki bozuklukların tipine bağlı olduğuna karar verilmiştir. Kurulan sistem 15 katmanlı bir CNN ağı tabanlı olup son katmanda çalışmada sunulan farklı önışleme algoritmaları bulunmaktadır. Kurulan sistemin test edilmesi amacıyla 3000 farklı kullanıcıya ait toplamda 5000 süpermarket fişine ait görüntüler toplanmıştır. Mobil cihaz kameralarıyla çekilmiş bu görüntülerden daha sonra 100.000 fiş satır görüntüsü alınarak nihai veri seti elde edilmiştir. Elde edilen bu 100.000 görüntüyü içeren test veri setindeki görüntülerin %94'ünde Binarizasyon yöntemi önışlem olarak seçilmiştir. Görüntülerin %71'inde; %37 gürültü azaltma ve %34 keskinleştirme olmak üzere Binarizasyon yönteminin başka bir önışleme metoduyla kombinasyonu seçilmiştir. 6000 görüntüden oluşan %6'luk kısımda ise kurulan CNN sistemi tarafından herhangi bir önışleme çalışmasına gerek olmadığına karar verilmiştir. Geliştirilen yöntem daha sonra Tesseract sistemi öncesine entegre edilerek Tesseract OCR başarıları test edilmiştir. 100.000 görüntü içeren aynı test veri setinde Tesseract sistemi tek başına %38 başarı elde ederken çalışmada kurulan model entegre edildiğinde Tesseract başarı oranı %63'e yükselmiştir (Bui ve ark 2017).

### 3. MATERYAL VE YÖNTEM

#### 3.1. Evrişimsel Sinir Ağları (CNN) ve Ana Bileşenleri

Evrişimsel Sinir Ağları, canlıların görme merkezinden esinlenerek geliştirilmiş ve yaygın olarak kullanılmakta olan derin öğrenme yaklaşımıdır. Başlangıçta sadece nesne tanıma çalışmalarında kullanılan CNN mimarileri; günümüzde metin tespiti, nesne takibi gibi çalışmalarda da kullanılmaktadır (Aloysius ve Geetha 2017). CNN mimarileri, öğrenme yoluyla kendi kendini optimize eden nöronlardan oluşmaları bakımından geleneksel yapay sinir ağlarına benzer. CNN mimarilerinin klasik yapay sinir ağlarından en büyük farkı ise öncelikli olarak görüntülerde yer alan örüntüleri tanımada kullanılmalarıdır (O'Shea ve Nash 2015). CNN mimarileri kullanım yerine göre farklı yapılarda kurulsalar bile genel olarak parçalar halinde gruplandırılmış evrişim ve havuzlama katmanlarından oluşur. Bu katmanların devamında ise standart geri yayımlı sinir ağlarında yer alan bir veya birden fazla tam bağlı katman yer alır (Rawat ve Wang 2017).

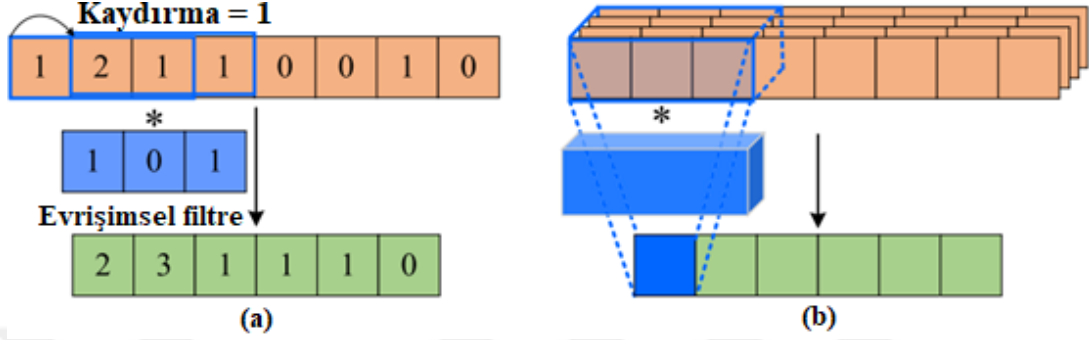
##### 3.1.1. Evrişim (Konvolüsyon) katmanı

Evrişim katmanı, CNN mimarisinde yer alan öznitelik çıkarım aşamasının merkezi bir adımıdır. Evrişim katmanının iki önemli avantajı vardır. Birinci avantajı, aralıklı bağlantı özelliğidir. Her bir evrişimsel filtre, önceki özniteliklerin yerel parçasına bağlı olduğundan ve etkili bir şekilde parametre sayısını azaltabildiğinden eğitim hızını arttırmaktadır. İkinci avantajı ise ağırlık paylaşım özelliğidir. Öznitelik haritası üzerinde gezerken, her bir filtrenin ağırlıkları değişmeden kalır ve bu da parametre sayısını daha da azaltır. Genel olarak her bir evrişim katmanında birden fazla filtre bulunur ve böylece daha kapsamlı öznitelik temsillerinin öğrenilmesi amaçlanır. Matematiksel olarak;  $\mathbf{x} \in \mathbb{R}^d$  d-boyutlu bir veri olmak üzere, j. öznitelik haritası Denklem 3.1'de gösterildiği şekilde tanımlanır:

$$\mathbf{c}^j = \mathbf{x} * \mathbf{w}^j + \mathbf{b}^j \quad (3.1)$$

Formülde geçen  $\mathbf{w}^j \in \mathbb{R}^h$  ifadesi; j. filtreyi temsil etmekte olup  $\mathbf{x}$  girdisini kodlamak için kullanılır ve  $\mathbf{c}^j$  olarak gösterilen j. öznitelik haritasını oluşturur.

Oluşturulmuş olan bu öznitelik haritası,  $c^j = [c_1^j, c_2^j, \dots, c_{d-h+1}^j]$  şeklinde ifade edilir. Formüldeki  $\mathbf{b}^j$  ifadesi ise sapma terimini ifade eder. Ek olarak, atlama sayısı ve dolgu parametreleri de evrişim yoğunluğunu ve çıktı boyutunu kontrol eder (Jiao ve ark 2020). Tek boyutlu evrişim katmanına ait sezgisel şema Şekil 3.1’de verilmiştir.



Şekil 3.1. Tek boyutlu evrişim işlemine ait şema. (a) Tek kanallı girdi; (b) Çok kanallı girdi (Jiao ve ark 2020)

### 3.1.2. Havuzlama katmanı

Havuzlama katmanı, evrişimsel katmanlar arasındaki bağlantı sayılarını azaltarak hesaplama maliyetini azalttığı için önemli bir CNN bileşenidir (Gu ve ark 2018). Havuzlama katmanının temel rolü, anlamsal olarak benzer öznitelikleri tek öznitelikte birleştirmektir (LeCun ve ark 2015). Havuzlama sırasında atlama miktarı genelde birden büyük seçilir ve bu yüzden evrişim sonrası oluşan temsillerin boyutunda azalma olur. Tipik çalışmalarda atlama miktarı havuzlama filtresinin boyutuna eşit seçilir ve bu sayede havuzlama bloklarında çakışma olmaz (Sze ve ark 2017). Sunulan fatura sınıflandırma çalışmasında; LeNet-5 mimarisinde tipik havuzlama yöntemlerinden olan ortalama-havuzlama, VGG-16 ve VGG-19 mimarilerinde ise en-büyük-havuzlama yöntemi kullanılmıştır.

#### 3.1.2.1. Maksimum havuzlama katmanı

Modern CNN mimarilerindeki maksimum havuzlama katmanı; öznitelik haritalarında üretilen en büyük değerleri alarak, görüntünün odaklanılan bölgesindeki özniteliklerin özetlenmiş bir hali şeklinde birleştirir. Buradaki amaç evrişimsel öznitelik

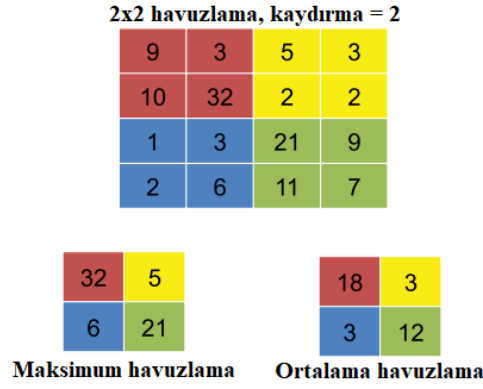
haritalarından gelen önemli bilgileri koruyarak gereksiz bilgileri silmektedir. Bu silme işlemi sayesinde CNN mimarisindeki parametre uzayı küçültülmektedir (Zheng ve ark 2019). Maksimum havuzlama işlemi ayrıca öznitelik konumlarına uzaysal olarak değişmezlik sağlamaktadır (Jaderberg ve ark 2015). En popüler havuzlama işlemi olan maksimum havuzlama işlemi (Yamashita ve ark 2018); ResNet (He ve ark 2016), GoogleNet (Szegedy ve ark 2015) ve AlexNet (Krizhevsky ve ark 2012) gibi birçok popüler CNN mimarisinde kullanılmış olup derin sinir ağlarındaki etkinliğini kanıtlamıştır (Zheng ve ark 2019). Maksimum havuzlama işlemine ait matematiksel ifade Denklem 3.2’de verilmiştir:

$$po_k^j = \max\{c_{k:k+r-1}^j\} \quad (3.2)$$

Formülde verilen  $c^j$  ifadesi girdiyi temsil eder ver r değeri havuz boyutunu ifade eder.  $po_k^j$  değeri ise karşılık gelen havuz alanındaki en büyük değeri ifade eder (Jiao ve ark 2020).

### 3.1.2.2. Ortalama havuzlama katmanı

Maksimum havuzlama tekniğine benzer bir şekilde çalışan ortalama havuzlama tekniği, maksimum havuzlama tekniğinden farklı olarak öznitelik haritalarında üretilen en büyük değerleri almak yerine bu değerlerin ortalamasını alarak çalışır (Jarrett ve ark 2009). Gri tonlamalı rakam görüntülerini sınıflandırmak için 1989 yılında sunulan ve ilk CNN yaklaşımlarından birisi olan LeNet (Le Cun ve ark 1989) CNN mimarisinde her bir evrişim katmanından sonra ortalama havuzlama tekniği kullanılmıştır. Evrişim katmanından çıkmış olan 4x4 boyutundaki bir matrise atlama miktarı 2 seçilerek, 2x2 boyutundaki filtrelerle ortalama havuzlama ve maksimum havuzlama işlemlerinin uygulanması işlemi ile elde edilen matrisler Şekil 3.2’de verilmiştir (Sze ve ark 2017).



Şekil 3.2. Maksimum havuzlama ve ortalama havuzlama işlemlerinin atlama sayısı 2 seçilerek uygulanması (Sze ve ark 2017)

### 3.1.3. Tam bağlı katman

Evrişim ve havuzlama katmanlarından oluşan blokları oluşturduktan sonra öznitelikleri daha ileri boyutta işlemek için tam bağlı katmanlar kullanılır. Tam bağlı katmanın matematiksel ifadesi Denklem 3.3’te verilmiştir.

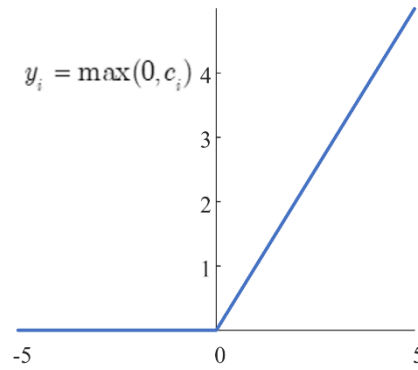
$$fc^l = \sigma(w^l \cdot fc^{l-1} + b^l) \quad (3.3)$$

Formülde geçen  $fc^l$  ifadesi,  $l$ . Sıradaki tam bağlı katmanın çıkışındaki öznitelikleri temsil eder.  $w^l$  ve  $b^l$  değerleri sırasıyla bağlantı ağırlığı ve sapma değerini ifade eder.  $\sigma(.)$  Değeri ise doğrusal olmayan aktivasyon fonksiyonunu temsil eder (Jiao ve ark 2020). Tipik bir derin sinir ağında tam bağlı katmanlar ağ parametrelerinin çoğunluğunu oluşturmaktadır. Toplamda 60 milyon parametreden oluşan AlexNet mimarisinde parametrelerin 58 milyonu tam bağlı katmanlarda bulunmaktadır. Benzer şekilde toplamda 138 milyon parametreden oluşan VGGNet ağındaki parametrelerin 123 milyon tanesi tam bağlı katmanlarda bulunmaktadır (Basha ve ark 2020). CNN ağında yer alan tam bağlı katman sayısı ve her katmandaki nöron sayısı probleme göre değişiklik gösterse de ağıın sonunda yer alan tam bağlı katmanın çıktı nöron sayısı genelde CNN ağıının sınıf sayısına eşittir (Yamashita ve ark 2018). (Basha ve ark 2020) tarafından yapılan çalışmada tam bağlı katmanların içerisinde bulunan nöron sayısının derin veya sığ sinir ağları üzerindeki etkileri araştırılmıştır. Buna göre daha yüksek performans elde etmek için derin ağlarda yer alan tam bağlı katmanlarda daha az sayıda nörona ihtiyaç duyulurken sığ ağlarda daha fazla nörona ihtiyaç duyulmaktadır. Ayrıca sığ ağlarda geniş

veri setlerinin derin veri setlerine göre daha fazla sayıda tam bağımlı katmana ihtiyaç duyduğu sonucuna varılmıştır.

### 3.1.4. Aktivasyon katmanı

Derin sinir ağlarının öğrenme başarısını arttırmanın yollarından birisi de gizli nöronlardaki aktivasyon fonksiyonunu değiştirmektir. Tarihsel olarak sigmoid fonksiyonu yaygın olarak yapay sinir ağlarında aktivasyon fonksiyonu olarak kullanılmıştır. Ancak sigmoid ve tanjant hiperbolik fonksiyonu gibi standart aktivasyon fonksiyonları kullanıldığında büyük değerlerdeki gradyanlar neredeyse sıfır olmaktadır. Bu nedenle Olasılıksal Gradyan İniş (Stochastic Gradient Descent) güncellemeleri çok küçük olur ve bu durum kaybolan gradyan sorunu olarak adlandırılır. Gradyan sorunun çözümü için daha sonra Rectified Linear Units(ReLU) aktivasyon fonksiyonu geliştirilmiştir. ReLU fonksiyonunun gradyanları pozitif değerlerde sabit olmakta ve hiç kaybolmamaktadır (Ide ve Kurita 2017). Bu sebeplerden dolayı ReLU aktivasyon fonksiyonu hem kaybolan gradyan sorununu çözmekte hem de derin sinir ağının öğrenme hızını arttırmaktadır (Jiao ve ark 2020). ReLU fonksiyonunun avantajlarından bir diğeri ise hesaplama maliyetinin düşük olmasıdır. İçerisinde üssel hesaplama ve bölme işlemleri içermediğinden hesaplama maliyeti düşüktür (Nwankpa ve ark 2018). Sunulan fatura sınıflandırma çalışmasında da evrişim ve havuzlama bloklarında kullanılan ReLU aktivasyon fonksiyonu, günümüzde derin sinir ağlarında standart aktivasyon fonksiyonu olarak kullanılmaktadır (Ide ve Kurita 2017). ReLU aktivasyon fonksiyonuna ait grafik Şekil 3.3'te verilmiştir.



Şekil 3.3.  $c_i$  ve  $y_i$  değerlerinin giriş ve çıkış özniteliklerini temsil ettiği durumda doğrusal olmayan ReLU fonksiyonuna ait grafik (Jiao ve ark 2020)

### 3.1.5. Optimizasyon algoritması

Evrişimsel sinir ağını kurma aşamasından sonra kurulan ağıın kabul edilebilir bir zamanda en iyi parametreleri elde edebilmesi yani eğitilmesi gerekmektedir. Eğitim için genellikle gradyan düşüş metotları kullanılmaktadır. Örnek olarak Toplu Gradyan İniş (Batch Gradient Descent) (BGD) ve Stochastic Gradient Descent (SGD) (Wijnhoven ve de With 2010) yaklaşımları kullanılmaktadır (Jiao ve ark 2020). BGD metodunda her güncellemedeki gradyanın alınması için bütün eğitim verileri hesaplanmaktadır ve bu yüzden dışbükey düzlemin mutlak en iyi değerine ya da dışbükey olmayan ortamın yerel en iyi değerine yaklaşılabilmektedir. BGD algoritmasında bir güncelleme için tüm veri seti güncellendiğinden bu yaklaşım gerçek dünya uygulamalarında oldukça yavaş çalışmakta olup pratik değildir. Ancak, SGD yaklaşımında her güncelleme için sadece bir örnek kullanılır ve bunun sonucu olarak SGD metodundaki her bir güncelleme süresi BGD yöntemine göre oldukça kısadır (Jiao ve ark 2020). Sunulan çalışmada da optimizasyon algoritması olarak SGD algoritması kullanılmıştır.

#### 3.1.5.1. Olasılıksal gradyan inişi ( SGD )

Danışmanlı öğrenme problemlerinde;  $P(x, y)$  olasılık dağılım fonksiyonundan alınan ve  $(x, y) \in \mathcal{X} \times \mathcal{Y}$  olarak tanımlanan eğitim seti örnekleri verilmiştir. Koşullu olasılık  $P(x, y)$  ifadesi, giriş vektörü olan  $x$  ve tahmin edilmeye çalışılan çıkış etiketi olan  $y$  arasındaki ilişkiyi temsil eder. Tahmin edilmiş etiket olan  $\hat{y}$  ve doğru etiket  $y$  arasındaki fark,  $\iota(\hat{y}, y)$  olarak gösterilen bir kayıp fonksiyonuyla temsil edilmektedir. Buradaki amaç, Denklem 3.4'te gösterilen ve beklenen riski azaltan  $f$  fonksiyonunu hesaplamaya çalışmaktır.

$$E(f) = \int \iota(f(x), y) dP(x, y) = E[\iota(f(x), y)] \quad (3.4)$$

$P(x, y)$  Bilinmeyen dağılımından  $n$  tane örnek  $(x_i, y_i), i = 1 \dots n$  şeklinde verilmiştir. Burada deneysel riski azaltan  $f_n$  fonksiyonu Denklem 3.5'te gösterildiği şekilde bulunmaktadır.

$$E_n(f_n) = \frac{1}{n} \sum_{i=1}^n \iota(f(x_i), y_i) = \mathbb{E}_n[\iota(f(x), y_i)] \quad (3.5)$$

$f$  Fonksiyonu doğrusal olarak  $\mathcal{W} \in \mathcal{R}^d$  tarafından parametre gönderilmektedir.  $d$  Değeri ise  $x$  öznitelik fonksiyonlarının boyutluluğunu ifade etmektedir. Klasik gradient düşüş yöntemlerinde, deneysel risk  $\mathcal{W}$  vektörünün gerçek gradyeni kullanılarak minimize edilmektedir. Bu işlem tipik olarak her bir eğitim örneğinden meydana gelen gradyanların toplamı olarak tahmin edilmektedir. Yapılan tahmin işlemleri Denklem 3.6 ve Denklem 3.7’de gösterildiği şekilde gerçekleştirilmektedir.

$$\mathcal{W}_{t+1} = \mathcal{W}_t - \eta \frac{\delta E_n}{\delta \mathcal{W}}(\mathcal{W}_t) \quad (3.6)$$

$$\mathcal{W}_{t+1} = \mathcal{W}_t - \eta \frac{1}{n} \sum_{i=1}^n \frac{\delta}{\delta \mathcal{W}} \iota(f_t(x_i), y_i) \quad (3.7)$$

Formülde verilen  $\eta$  değeri, güncelleme/kazanç çarpanı ya da  $t$ . Adımdaki  $\mathcal{W}_t$  çözümünü güncellemek için kullanılan adım boyutudur. Standart gradyan düşüş yöntemleri gradyanı hesaplamak için bütün eğitim seti üzerinde işlem yapar ve bu yüzden optimizasyon parametrelerini günceller. Birçok döngüde mutlak en iyiye erişme ihtiyacı olacağından standart yöntemler büyük veri setleri için pratik değildir. SGD yöntemi ise her döngüde sadece tek örnek alarak hesapladığından, zaman bağımlı bir ağırlıklandırma çarpanı kullanarak  $\mathcal{W}$  ağırlık vektörünü döngüsel olarak günceller. Yapılan güncelleme işlemi Denklem 3.8’de gösterilmiştir.

$$\mathcal{W}_{t+1} = \mathcal{W}_t - \frac{\eta}{t} \frac{\delta}{\delta \mathcal{W}} \iota(f_t(x_t), y_t) \quad (3.8)$$

Standart gradyan düşüş yöntemleriyle karşılaştırıldığında, SGD yöntemi güncelleme başına daha az zamana ihtiyaç duyar ve sonuç olarak daha hızlı yakınsama özelliğine sahiptir. Dikkat edilmesi gereken nokta, her seferinde sadece tek bir örnek göz önüne alındığı için  $\mathcal{W}_t$  optimizasyon parametrelerinin güncellenmesi gürültülü olmaktadır.

Destek Vektör Makineleri(SVM)’ de kullanıldığı gibi düzenleme kısıtı olan doğrusal bir sınıflandırıcı elde etmek için, Hinge kayıp SVM objektif fonksiyonu olan

$(\iota(z, y) = \max\{0, 1 - yz\})$  ve  $z = \mathcal{W}x + b$  kullanılmış ve Denklem 3.9'da gösterildiği şekilde fonksiyon minimize edilmiştir.

$$\frac{\lambda}{2} ||\mathcal{W}||^2 + \frac{1}{n} \sum_{i=1}^n \iota(y_i(\mathcal{W} x_i + b)) \quad (3.9)$$

Formülde geçen düzenleme terimi  $||\mathcal{W}||$ , öznitelik uzayındaki kenarların boyutunu temsil etmektedir. Bu fonksiyonun SGD yaklaşımına eklenmesi ile Denklem 3.10'da gösterilen fonksiyon elde edilir:

$$\mathcal{W}_{t+1} = \mathcal{W}_t - \eta \left( \lambda \mathcal{W}_t + \frac{\delta \iota(y_t(\mathcal{W}_t x_t + b))}{\delta \mathcal{W}_t} \right), \quad \eta_t = \frac{1}{\lambda(t + t_0)} \text{ olmak üzere} \quad (3.10)$$

Burada  $\lambda$  ve  $t_0$  parametreleri kullanıcı tarafından ayarlanmalıdır çünkü  $\eta_0$ ,  $t_0$  ve  $\eta_0$  değerleri doğrudan birbiriyle ilişkilidir (Wijnhoven ve de With 2010).

### 3.1.6. Dropout (Seyreltme) katmanı

Seyreltme katmanı, ağdaki bazı bağlantıları veya birimleri belirli bir olasılık dâhilinde rastgele bir şekilde atlayarak ağın genelleştirme yeteneğini artırır. Sinir ağlarında doğrusal olmayan bir ilişkiyi öğrenmeye çalışan çoklu bağlantılar, bazı durumlarda ağda ezberlemeye sebep olmaktadır (Hinton ve ark 2012). Bu şekilde bazı katmanların veya birimlerin rastgele atlanması sayesinde birkaç küçük ağ mimarisi üretilmektedir. Daha sonra bu mimarilerden küçük ağırlık içeren ağ belirlenerek temsil ağı olarak seçilir. Seçilen bu mimari daha sonra bütün sunulan ağların yaklaşımı olarak kabul edilir (Khan ve ark 2020).

Seyreltme işlemine uğramış nöronlar, ileri beslemeye katkıda bulunmayıp geri yayılımına katılmamaktadır. Dolayısıyla her çalışmada farklı bir mimari örneği oluşmakta olup oluşan bütün mimariler ağırlık paylaşımı yapmaktadır. Bu yöntem sayesinde nöron grupları arasındaki birbirine bağımlılık (Co-adaption) sorunu azalmaktadır. Çünkü artık bir nöron, başka nöronların varlığına dayanarak çalışmamaktadır. Bunun sonucu olarak nöronlar, daha istikrarlı öznitelikleri öğrenmeye zorlanmaktadır (Krizhevsky ve ark 2012). Sunulan fatura sınıflandırma sisteminde kullanılan bütün CNN mimarilerinde seyreltme katmanı kullanılmış olup seyreltme olasılığı 0.5 olarak kullanılmıştır.

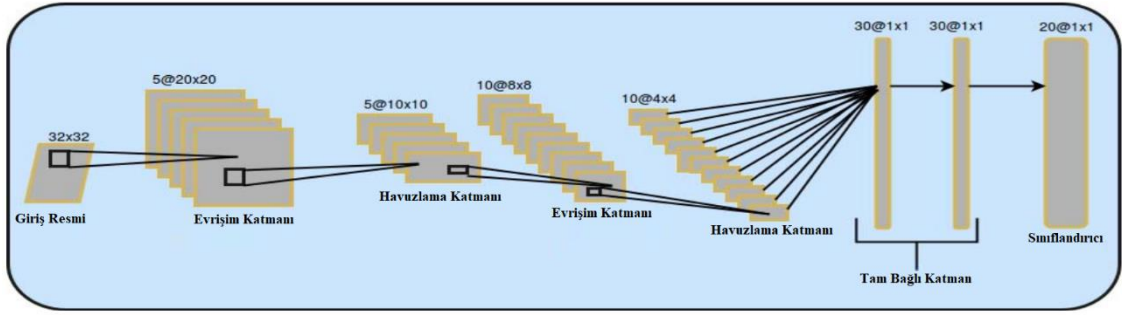
### 3.1.7. Sınıflandırma katmanı

Öznitelik çıkarma aşamasından sonra sonuçların alınması için çıktı katmanı olarak sınıflandırma katmanı kullanılır. Softmax fonksiyonu sınıflandırma katmanlarında kullanılan en popüler aktivasyon fonksiyonlarından biri olmuştur (Jiao ve ark 2020). Yıllar boyunca derin öğrenme metotları birçok çalışmada (Krizhevsky ve ark 2012, Simonyan ve Zisserman 2014) kullanılarak en yüksek performansları elde etmiştir. Yapılan bu çalışmalarda kullanılan derin sinir ağlarında sınıflandırıcı olarak Softmax aktivasyon fonksiyonu kullanılmıştır (Agarwal ve ark 2018). Softmax fonksiyonun yaygın bir şekilde sınıflandırma katmanında kullanılmasının nedeni, her bir sınıf için olasılıkları ayrı ayrı hesaplamasıdır (Aloysius ve Geetha 2017). Yaygın bir şekilde kullanılan diğer bir metot ise SVM (Cortes ve Vapnik 1995) metodudur. SVM, CNN özniteliklerine entegre edilebilir ve çeşitli sınıflandırma çalışmalarında kullanılabilir (Gu ve ark 2018). Softmax fonksiyonunun matematiksel Denklem 3.11’de verilmiştir.

$$p(z)_i = \frac{e^{z_i}}{\sum_{i=1}^{N_C} e^{z_i}} \quad (3.11)$$

Formülde geçen  $p(z)_i$  ifadesi olasılığı temsil eder.  $e^{z_i}$  ifadesi, giriş vektörü için kullanılan standart üssel bir fonksiyondur. Ayrıca formüldeki  $z$  değeri  $i$  kategorisine ait olup,  $i \in N_C$  olarak ifade edilmektedir (Jiao ve ark 2020). Yapılan çalışmada da veri setindeki görüntüleri 4 sınıfa göre sınıflandıran en son CNN katmanında Softmax aktivasyon fonksiyonu kullanılmıştır.

Bu bölümde CNN mimarisinde yer alan evrişim, havuzlama ve tam bağlı katmanlarından bahsedilmiştir. Ayrıca ağırlık güncellenmesinde kullanılan optimizasyon metodundan ve ağın son katmanında yer alan sınıflandırma katmanından bahsedilmiştir. Verilen genel bilgilere göre, 32x32 boyutunda örnek bir görüntü girdisine uygulanan CNN mimarisinin genel yapısı Şekil 3.4’te verilmiştir.



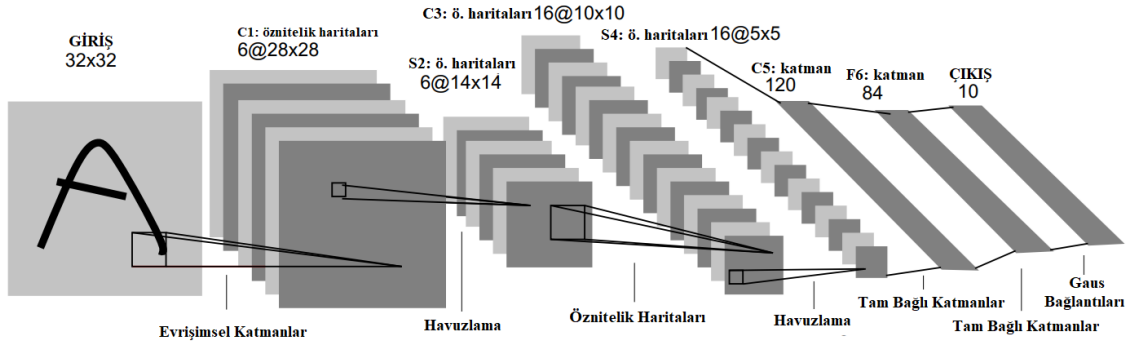
Şekil 3.4. 32x32 boyutunda bir görüntüye uygulanan CNN mimarisinin genel yapısı

(Aloysius ve Geetha 2017)

### 3.2. Evrişimsel Sinir Ağları Mimarıleri

#### 3.2.1. LeNet-5 mimarisi

Yann LeCun ve ark. tarafından sunulan ve el yazısı basamakların sınıflandırılması için geliştirilen LeNet-5 mimarisi (LeCun ve ark 1998); ham piksellerden doğrudan öğrenebilmekte ve görüntülerde çok az ön işlemeye ihtiyaç duymaktadır. Birden fazla katmana sahip LeNet-5 mimarisi geri yayılım algoritması kullanılarak eğitilebilmektedir (Gu ve ark 2018). LeNet-5 mimarisi, eğitilebilir parametrelere sahip 7 katmandan oluşan bir CNN mimarisidir. LeNet-5 mimarisi; iki adet evrişim katmanı, iki adet havuzlama katmanı, iki adet tam bağlı katman ve son olarak bir çıktı katmanı içermektedir. Geçmişte Amerika Birleşik Devletlerinde birçok bankada çeklerin üzerindeki el yazısı rakamların tanınmasında LeNet-5 kullanılmış ve mükemmel tanıma başarısı elde edilmiştir (Wang ve Gong 2019). Mimaride giriş görüntü boyutu olarak 32x32 seçilmiştir. Seçilen bu görüntü boyutu, veri tabanında bulunan ve 28x28 boyutundaki alanda ortalanmış en fazla 20x20 boyutundaki karakter boyutlarından oldukça büyüktür. Rakam görüntülerinin tanınması çalışmasını gösteren genel LeNet-5 mimarisi Şekil 3.5'te verilmiştir.



Şekil 3.5. LeNet-5 mimarisinin genel yapısı (LeCun ve ark 1998)

Şekil 3.5'te verilmiş olan LeNet-5 mimarisinde evrişim katmanları sırasıyla C1 ve C3 olarak, havuzlama katmanları sırasıyla S2 ve S4 olarak, tam bağlı katmanlar ise sırasıyla C5, F6 olarak isimlendirilmiştir. 10 basamaklı bir veri setini sınıflandırmak için kurulmuş olan LeNet-5 ağında toplam 10 sınıf olduğundan ağ çıkışında 10 nöronlu bir çıktı katmanı vardır. C1 katmanı evrişimsel bir katman olup 6 öznitelik haritasına sahiptir. Öznitelik haritasındaki her birim, giriş görüntüsündeki 5x5 boyutundaki komşuya bağlıdır. C1 katmanında toplamda 156 eğitilebilir parametre olup 122.304 tane bağlantıya sahiptir. S2 katmanı bir havuzlama katmanı olup 14x14 boyutundaki 6 öznitelik haritasına sahiptir. S2 katmanı toplamda 12 eğitilebilir parametre ve 5.880 bağlantıya sahiptir. Sonraki evrişim katmanı olan C3 katmanında 16 öznitelik haritası, 1.516 eğitilebilir parametre ve toplamda 151.600 bağlantı bulunmaktadır. S4 havuzlama katmanında 5x5 boyutundaki 16 öznitelik haritası vardır. S4 katmanında toplamda 32 eğitilebilir parametre ve 2000 bağlantı bulunmaktadır.

C5 katmanı 120 öznitelik haritasına sahip bir evrişim katmanıdır. Her bir birim, S4 katmanının 16 öznitelik haritası üzerinde 5x5 komşulara bağlıdır. Burada S4 katmanının boyutu da 5x5 olduğundan C5 katmanının öznitelik haritasının boyutu 1x1'dir. Bundan dolayı da S4 ve C5 arasında tam bağlantı oluşmaktadır. C5 katmanı bir evrişim katmanı olmasına rağmen tam bağlı katman olarak etiketlenmiştir çünkü eğer LeNet-5 girdi görüntüsü büyütülüp diğer her şey sabit tutulursa, öznitelik haritasının boyutu 1x1 boyutundan daha büyük olmaktadır. F6 tam bağlı katman C5 katmanına tam bağlı olup 10.164 eğitilebilir parametreye sahiptir (LeCun ve ark 1998).

### 3.2.2. VGG-16 mimarisi

VGG mimarisi; büyük-ölçekli görüntülerin tanınmasında, CNN mimarilerinde derinliğin arttırılmasına bağlı başarı değişimini gözlemlemek için geliştirilmiştir. VGG, her evrişim katmanında (3x3) gibi küçük filtreler kullanan ilk mimaridir. Görüntüden benzer öznitelikleri çıkarmak için (9x9 ve 11x11) gibi büyük filtreler kullanan LeNet ve AlexNet mimarilerine göre VGG mimarisi küçük filtreler kullanmaktadır. VGG mimarisinin en büyük katkısı, daha büyük maske filtrelerinin yüksek hesaplama maliyetini düşürmek için birbirini takip eden evrişim grupları kullanmasıdır (Rezende ve ark 2019).

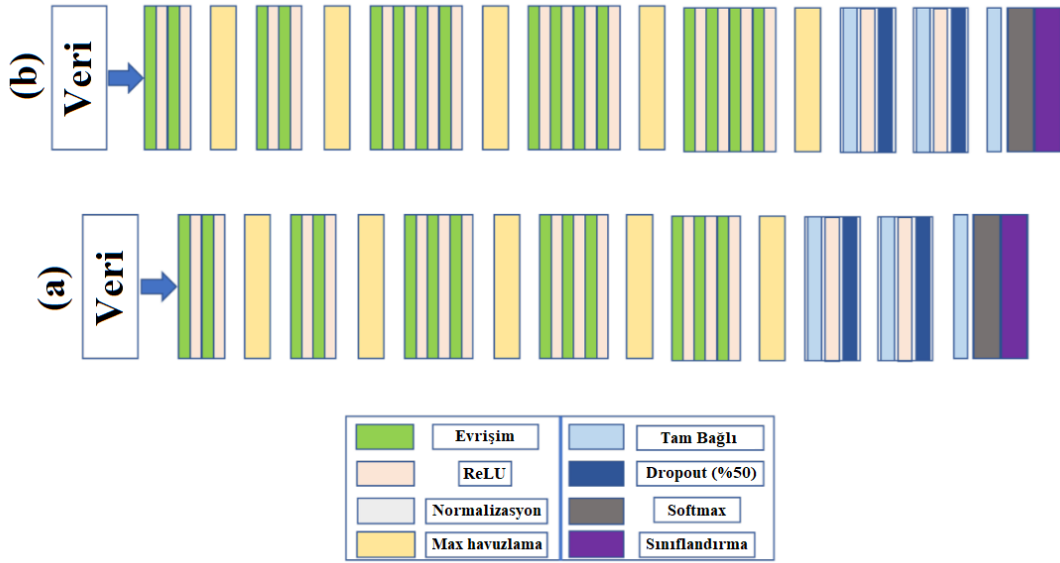
VGG ailesinin bir üyesi olan ve 16 katmandan oluşan VGG-16 mimarisi, Karen Simonyan ve Andrew Zisserman tarafından 2014 yılında sunulmuştur. VGG-16 mimarisinin eğitim aşamasında 224 x 224 boyutundaki RGB görüntüler kullanılmıştır (Simonyan ve Zisserman 2014).

VGG-16 mimarisindeki ilk iki evrişim katmanı 64 öznitelik filtresinden oluşur ve filtre boyutu 3x3' dür. Giriş görüntüsünün ilk iki evrişim katmanına gönderilmesiyle görüntü boyutu 224 x 224 x 64 olarak değiştirilmektedir. Oluşan çıkış görüntüsü daha sonra atlama sayısı 2 olarak ayarlanarak maksimum havuzlama katmanına gönderilir. 3. ve 4. evrişim katmanları 128 öznitelikten oluşur ve filtre boyutu 3x3' dür. Bu iki katmanı, atlama sayısı 2 olan bir maksimum havuzlama katmanı takip eder ve sonuçta oluşan görüntü boyutu 56x56x128'e kadar düşer. Beşinci, altıncı ve yedinci katmanlar filtre boyutu 3x3 olan evrişim katmanlarıdır. Her üç katman da 256 tane öznitelik haritası kullanır. Bu katmanları da atlama sayısı 2 olan bir maksimum havuzlama katmanı takip eder. Sekizinci ve on üçüncü katmanlar arasındaki katmanlar filtre boyutları 3x3 olan evrişimsel katman setleridir. Her üç evrişimsel katman grubunda da 512 tane filtre bulunmaktadır. Bu katmanları, atlama sayısı 1 olan maksimum havuzlama katmanı takip eder. On dördüncü ve on beşinci katmanlar, tam bağlı gizli katmanlar olup 4096 birimden oluşmaktadır. Son olarak on altıncı katman olan sınıflandırma katmanı, 1000 birimden oluşan bir Softmax katmanıdır (Tammina 2019). Gizli katmanlarda ise aktivasyon fonksiyonu olarak doğrusal olmayan bir fonksiyon olan ReLU kullanılmıştır (Masood ve ark 2020).

### 3.2.3. VGG-19 mimarisi

VGG-19 mimarisi Simonyan ve Zisserman tarafından sunulmuştur (Simonyan ve Zisserman 2014). VGG-16 ve VGG-19 mimarileri, VGG (Visual Geometry Group) takımının 2014 Imagenet yarışmasına temel katılım sebebidir. Takım, yarışmada VGG-16 ve VGG-19 mimarileriyle lokalizasyon ile sınıflandırma alanlarında sırasıyla birincilik ve ikinciliği kazanmıştır (Carvalho ve ark 2017). VGG-19 mimarisi, VGG-16 mimarisine benzer yapıda olup fazladan 3 katmanı olmasıyla VGG-16 mimarisinden ayrılmaktadır. VGG-19 mimarisinde bulunan fazladan 3 katman, daha etkili eğitim çalışmaları için görüntülerden daha detaylı örüntüler öğrenebilmektedir (Reghunath ve ark 2019). Sadelik ve kolay uygulanabilirlik gibi avantajlarından dolayı çok popüler olan VGG-19 mimarisi, görüntü sınıflandırma çalışmalarında çok yüksek performansa sahiptir. VGG-19 mimarisi de VGG-16 mimarisi gibi 224 x 224 boyutundaki görüntüleri girdi olarak almaktadır (Xia ve ark 2019).

VGG-19 mimarisi 5 evrimsel katman bloğunu takip eden 3 tam bağlı katmandan oluşmaktadır. Evrimsel katmanlar 3x3 boyutunda filtreler kullanır ve atlama adım sayıları 1'dir. Evrimsel katmanlar ayrıca her bir aktivasyon haritasının bir önceki katmanla aynı uzaysal boyutları sürdürmesini garantilemek için dolgu olarak 1 değerini kullanır. Mimaride her bir evrim katmanından sonra ReLU aktivasyon fonksiyonu kullanılmakta olup uzaysal boyutları azaltmak için genelde maksimum havuzlama işlemi yapılmaktadır. Maksimum havuzlama katmanları, 2 x 2 boyutunda filtreleri atlama sayısı 2 seçerek kullanılmaktadır. Maksimum havuzlama katmanları, önceki katmandan gelen aktivasyon haritasının her bir uzaysal boyutunun ikiye bölünmesini garanti etmek için dolgu kullanmaz. Mimarideki üç tam bağlı katmandan ilk ikisinde ReLU tarafından aktive edilen 4096 birim bulunmaktadır. Son katmanda ise Softmax aktivasyon fonksiyonunu kullanan 1000 birimli bir tam bağlı katman bulunmaktadır. VGG-19 mimarisindeki evrim blokları temelde öznetelik çıkarma katmanlarıdır. Bu evrim blokları tarafından oluşturulan aktivasyon haritaları ise darboğaz öznetelikler olarak isimlendirilmektedir (Carvalho ve ark 2017). VGG-16 ve VGG-19 mimarilerine ait şemalar Şekil 3.6' da gösterilmiştir.

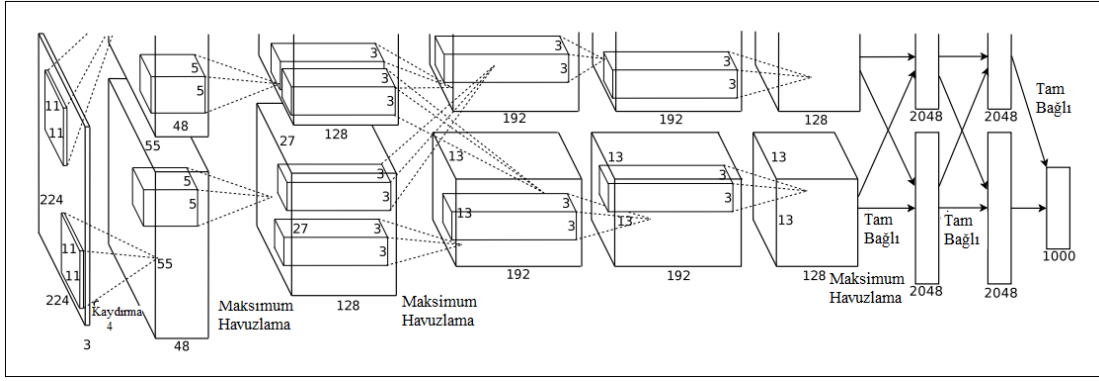


Şekil 3.6. VGG mimarileri (a) VGG-16 (b) VGG-19 (Shaha ve Pawar 2018)

### 3.2.4. AlexNet mimarisi

AlexNet mimarisi (Krizhevsky ve ark 2012), ilk derin CNN mimarisi olarak kabul edilmektedir. AlexNet mimarisi, 2012 yılında Krizhevsky ve arkadaşları tarafından sunulmuş olup derinlik artırılması ve farklı parametre optimizasyon stratejisinin kullanımı sayesinde CNN mimarisinin öğrenme kapasitesini iyileştirmiştir. AlexNet mimarisi paralel iki NVIDIA GTX 580 GPU üzerinde eğitilmiştir.(Khan ve ark 2020)

AlexNet mimarisi Şekil 3.7’de gösterilmiştir. AlexNet mimarisi, 3 evrişim ve 2 tam bağlı katmandan oluşmaktadır. İlk evrişim katmanında Yerel Tepki Normalizasyon (Local Response Normalization (LRN)) kullanılarak evrişim ve havuzlama gerçekleştirilmektedir. İlk evrişim katmanında 11x11 boyutunda 96 farklı alıcı filtre kullanılmaktadır. Maksimum havuzlama işlemleri 3x3 filtre kullanılarak ve kaydırma sayısı 2 seçilerek gerçekleştirilmektedir. İkinci katmanda aynı işlemler gerçekleştirilmiş olup filtre boyutu 5x5 olarak seçilmiştir. Üçüncü, dördüncü ve beşinci katmanlarda filtre boyutları 3x3 olup öznetelik haritası boyutları sırasıyla 384, 384 ve 296 olarak seçilmiştir. Tam bağlı katmanlarda seyreltme kullanılmış olup son katmanda Softmax aktivasyon fonksiyonu kullanılmıştır. AlexNet mimarisi ImageNet veri seti üzerinde eğitilmiş olup görüntü giriş boyutu olarak 224x224x3 boyutu seçilmiştir (Alom ve ark 2018).



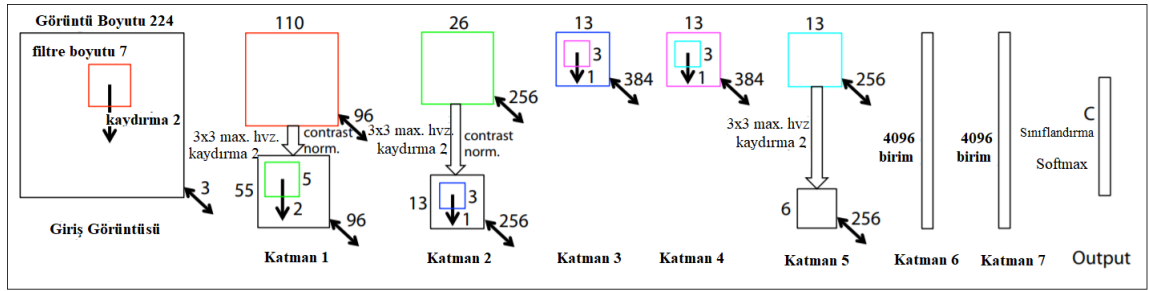
Şekil 3.7. AlexNet mimarisi (Krizhevsky ve ark 2012)

### 3.2.5. ZFNet mimarisi

2013 yılı öncesinde CNN mimarilerinin öğrenme mekanizmasında, görüntü tanıma konusunda yaşanan iyileşmenin asıl nedeni bilinmiyordu. Asıl sebebin bilinmemesinden dolayı CNN mimarilerinin karmaşık görüntülerdeki performansı sınırlıydı.

2013 yılında Mathew Zeiler ve Rob Fergus, AlexNet mimarisinin uzantısı olan bir CNN mimarisiyle ILSVRC 2013'ü kazanmıştır. Mimari, ZFNet (Zeiler ve Fergus 2014) olarak isimlendirilmiştir. ZFNet mimarisi, 11x11 filtreler yerine 7x7 boyutunda filtreler kullanarak ağırlık sayılarını büyük miktarda azaltmaktadır. Ağırlık sayısını azaltan bu mimari, genel tanıma başarısını ise arttırmaktadır (Alom ve ark 2018).

ZFNet mimarisi, temelde çok katmanlı bir Dekonvolüsyonel Sinir Ağıdır (Deconvolutional Neural Network (DNN)). ZFNet mimarisi, ağ performansını nicel bir şekilde görselleştirmek amacıyla geliştirilmiştir. Ağ aktivitesinin görselleştirilmek istenmesindeki amaç, nöron aktivitelerini yorumlayarak CNN performansını izlemektir. ZFNet ile sunulan öznetelik görselleştirme fikri AlexNet üzerinde de doğrulanmıştır. AlexNet üzerinde DNN kullanılmasıyla, sadece birkaç nöronun aktif olduğu görülmüştür. Ağı birinci ve ikinci katmanındaki diğer nöronların ise ölü yani pasif nöronlar olduğu gözlemlenmiştir. Bu bulgulara dayanarak, Zeiler ve Fergus tarafından CNN topolojisi düzenlenmiş olup parametre optimizasyonu yapılmıştır. Mimarideki ilk iki evrimsel katmanda en fazla sayıda özneteliğin tutulabilmesi için hem filtre boyutu hem de kaydırma adım sayısı azaltılarak CNN mimarisinin öğrenmesi arttırılmıştır (Khan ve ark 2020). ZFNet mimarisi şekil 3.8'de gösterilmiştir.

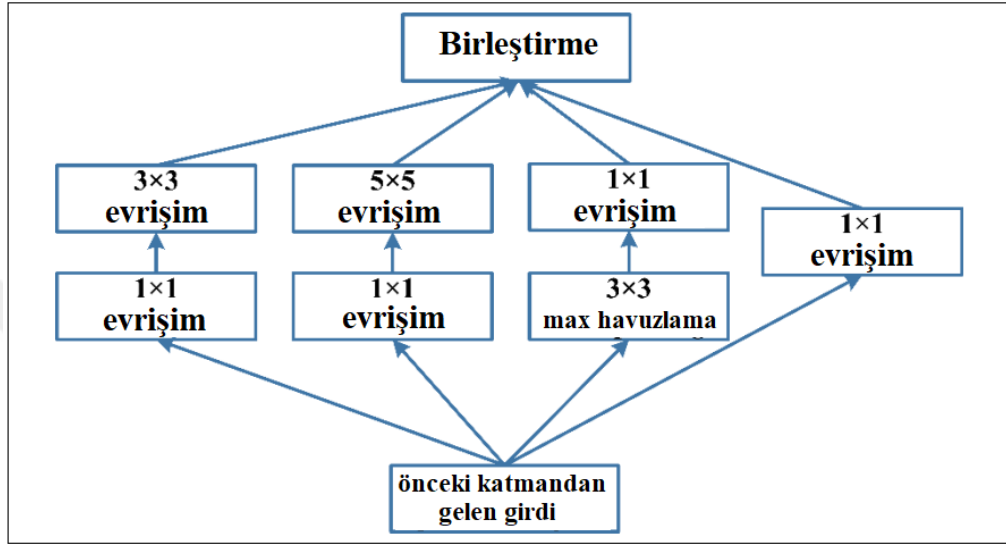


Şekil 3.8. ZFNet mimarisi (Zeiler ve Fergus 2014)

### 3.2.6. GoogleNet (Inception-V1) mimarisi

GoogleNet mimarisinin asıl amacı, daha az hesaplama maliyetiyle daha yüksek tanıma başarısı elde etmektir. Bu mimari ile CNN yapısına başlangıç (inception) blokları eklenmiştir. GoogleNet mimarisi; parçalama, dönüştürme ve birleştirme kullanarak çok ölçekli evrimsel dönüşümleri birleştirir. Inception bloğunun mimarisi şekil 3.9’da gösterilmiştir. GoogleNet mimarisinde, ağ içerisinde ağ (Network in Network) mimarisine (Lin ve ark 2013) benzer bir şekilde geleneksel evrim katmanları küçük bloklarla değiştirilmiştir. Bu bloklar, farklı ölçeklerde uzaysal bilgileri yakalamak için farklı boyutlarda filtreler (1x1, 3x3 ve 5x5) içerir. Dolayısıyla mimaride parçalama, birleştirme ve dönüşüm kullanılması amaç, farklı çözünürlükte resimlerden öğrenme yapılırken karşılaşılan sorunları çözmektir. GoogleNet mimarisi, büyük boyutta filtreleri çalıştırmadan önce 1x1 boyutundaki darboğaz (Bottleneck) evrim filtresi ekleyerek hesaplama işlemlerini düzenler. Ayrıca mimaride seyrek bağlantı yapısı kullanılmakta olup çıkıştaki her öznitelik haritası girişteki her öznitelik haritasına bağlı değildir. Bu yapıdaki amaç, gereksiz bilgi sorununu çözmek ve alakalı olmayan öznitelik haritalarını hariç tutarak maliyeti düşürmektir. Ek olarak, mimaride tam bağlı katman yerine küresel ortalama havuzlama kullanıldığı için bağlantıların yoğunluğu azaltılmıştır. (Khan ve ark 2020). GoogleNet toplamda 22 katmandan oluşmakta olup daha önceki herhangi bir ağdan daha büyüktür. Ancak katman sayısı fazla olsa da GoogleNet tarafından kullanılan parametre sayısı, AlexNet ve VGG ağlarına göre daha azdır. GoogleNet 7 milyon parametre kullanırken AlexNet ve VGG-19 ağlarında sırasıyla 60 milyon ve 138 milyon parametre kullanılmıştır (Alom ve ark 2018).

GoogleNet mimarisinde yer alan en büyük dezavantaj, heterojen bir topolojiye sahip olmasıdır. Bundan dolayı topolojinin birimden birime özelleştirilmesi gerekmektedir. Diğer bir dezavantaj ise içerisinde yer alan temsil darboğaz filtresinden dolayı sonraki katmandaki öznetelik haritası küçülmekte ve bazen önemli bilgilerin kaybolmasına sebep olmaktadır.(Khan ve ark 2020)



Şekil 3.9. Parçalama, dönüşüm ve birleştirme yapısını gösteren temel inception bloğu mimarisi (Khan ve ark 2020)

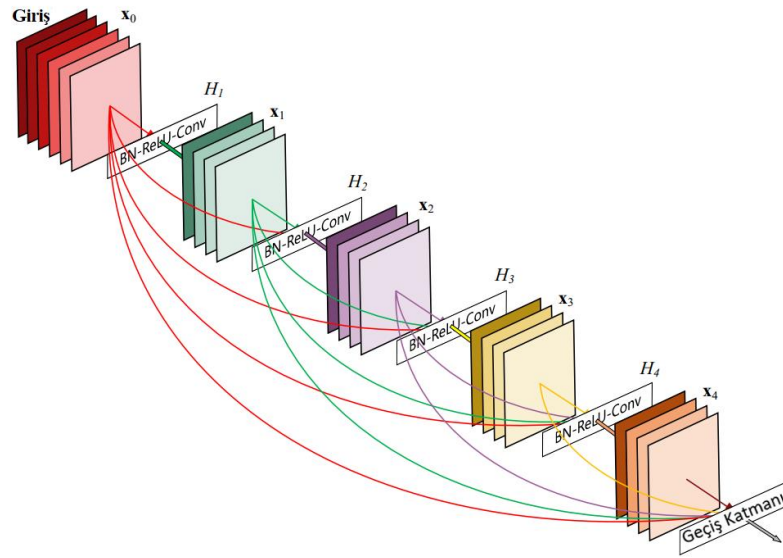
### 3.2.7. DenseNet121 mimarisi

DenseNet121 mimarisi (Huang ve ark 2017), 2017 yılında sunulan ve ağ mimarisindeki katmanlar arasındaki bilgi akışını en yüksek değere taşımayı hedefleyen bir CNN mimarisidir. Bu mimaride bütün katmanlar, eşleşen özellik haritası boyutlarıyla doğrudan birbirine bağlanarak katmanlar arası bilgi akışı arttırılmıştır (Huang ve ark 2017).

DenseNet121 mimarisi, alternatif olarak Dense bloklarının ve havuzlama katmanlarının dizilmesiyle oluşturulan basit ve verimli bir ağ mimarisidir. DenseNet121 mimarisinde, ileri besleme bağlantıları kurulurken bir katmandan diğer bütün katmanlara bağlantı oluşturulur. N tane katmana sahip olan diğer mimarilerin sonraki katmanlarının her birisi arasında N bağlantı bulunurken DenseNet121 mimarisinde N(N+1) doğrudan bağlantı bulunmaktadır. Ağın doğruluğunu ve verimliliğini arttırmak için DenseNet121 mimarisinin ilk ve son katmanları arasında daha kısa bir bağlantı kullanılır. Giriş değeri

sonraki katmanlarda ilerledikçe, her katmandaki özellik haritaları bir öncekiyle birleştirilir. Bu özellik sayesinde katmanlar tarafından diğer herhangi bir katmanın özellik haritasına erişim sağlanabilir. DenseNet121 mimarisi ayrıca gradyan kaybolma problemini azaltarak özelliğin yeniden kullanılmasını sağlar (Pranav ve ark 2020). DenseNet121 mimarisi diğer CNN mimarilerine göre daha az sayıda parametre kullanır ve gereksiz parametreleri eleyerek kullanmaz. DenseNet121 mimarisinde düşük değerli ağırlıklar olmadığından diğer CNN mimarilerine göre daha küçük bir ağ mimarisine sahiptir (Yapıcı ve ark 2019).

Büyüme oranı isminde bir hiper parametre kullanan DenseNet121 mimarisi, bu parametre sayesinde her katmanın genel ağ durumuna ne kadar katkıda bulunduğunu belirler. Ağın genel durumu bir kez yazıldıktan sonra ağın her yerinde bu duruma erişilebilir ve diğer CNN mimarileri gibi katmandan katmana kopyalanmasına gerek yoktur. Ağda kullanılan büyüme oranı parametresi  $k$  ile gösterilirse, DenseNet121 mimarisi diğer CNN mimarilerine göre  $k = 12$  gibi çok dar katmanlar içerebilir. ImageNet veri tabanı ile yapılan çalışmalarda 4 adet Dense bloğu kullanılmış ve giriş görüntü boyutu olarak  $224 \times 224$  kullanılmıştır. Yapılan çalışmalarda evrişim ve havuzlama katmanlarında atlama adım sayısı 2, evrişim boyutu olarak  $7 \times 7$  ve havuzlama katmanı olarak  $3 \times 3$  en büyük havuzlama kullanılmıştır. Büyüme oranı olarak  $k = 4$  değerini kullanan ve 5 adet Dense bloğundan oluşan bir DenseNet mimarisi şekil 3.10'da gösterilmiştir.

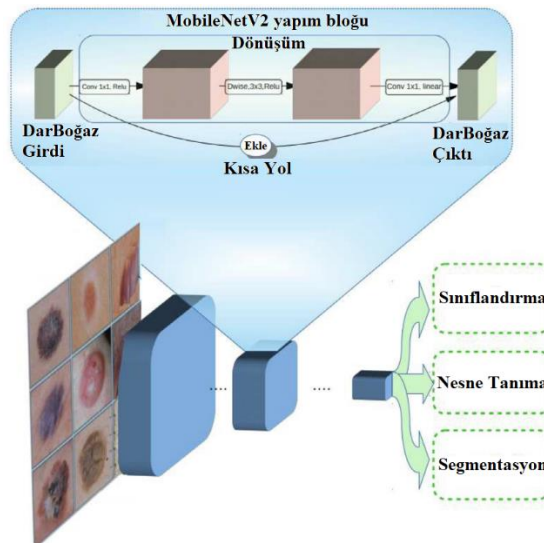


Şekil 3.10. Büyüme oranı olarak  $k = 4$  kullanan ve 5 Dense bloğu içeren bir DenseNet mimarisi (Huang ve ark 2017)

### 3.2.8. MobileNetV2 mimarisi

MobileNetV1 mimarisi; mobil cihazlar için geliştirilmiş, ağ maliyetini ve ağ boyutunu düşüren bir evrimsel sinir ağı mimarisidir. Bu mimari sayesinde mobil cihazlarda görüntü sınıflandırma işlemleri oldukça kolay yapılabilmektedir. MobileNetV2 mimarisi, 2018 yılında MobileNetV1 mimarisi üzerine geliştirilmiştir. MobileNetV2 mimarisi başlangıçta yüz özniteliklerini tanımak için tasarlanmıştır ancak daha sonra Google'nin kurum içi veri setinde eğitilmiştir (Saxen ve ark 2019). MobileNetV2 mimarisi kullanılarak segmentasyon, sınıflandırma ve nesne tanıma işlemleri gerçekleştirilebilir. MobileNetV2 mimarisi, önceki mimariye ek olarak iki yeni özellikle beraber geliştirilmiştir. Birinci özellik, katmanlar arasında doğrusal olarak darboğazların oluşabilmesidir. İkinci özellik ise darboğazlar arasında kısa yolların geliştirilmesidir. MobileNetV2 mimarisinin genel mimari şeması şekil 3.11'de gösterilmiştir (Toğaçar ve ark 2021).

Derin öğrenme algoritmalarında ağı derinliği arttıkça gradyan kaybolma sorunu artmaktadır. MobileNetV2 mimarisinde yer alan ters çevrilmiş artık blok, gradyan sorununu çözmek için BN(Toplu Normalleştirme) kullanmaktadır. Mimaride aktivasyon fonksiyonu olarak ReLU kullanıldığında, çıkış değeri sıfır ve sonsuz aralığında değişmektedir. Düşük hassasiyetli çözünürlük kullanılarak bu kadar büyük bir sayı aralığı tanımlanamadığından MobileNetV2 mimarisinde ReLU fonksiyonun düzenlenmiş bir hali olan ReLU6 fonksiyonu kullanılmaktadır (Zou ve ark 2020).



Şekil 3.11. MobileNetV2 mimarisinin genel çalışma şeması (Toğaçar ve ark 2021)

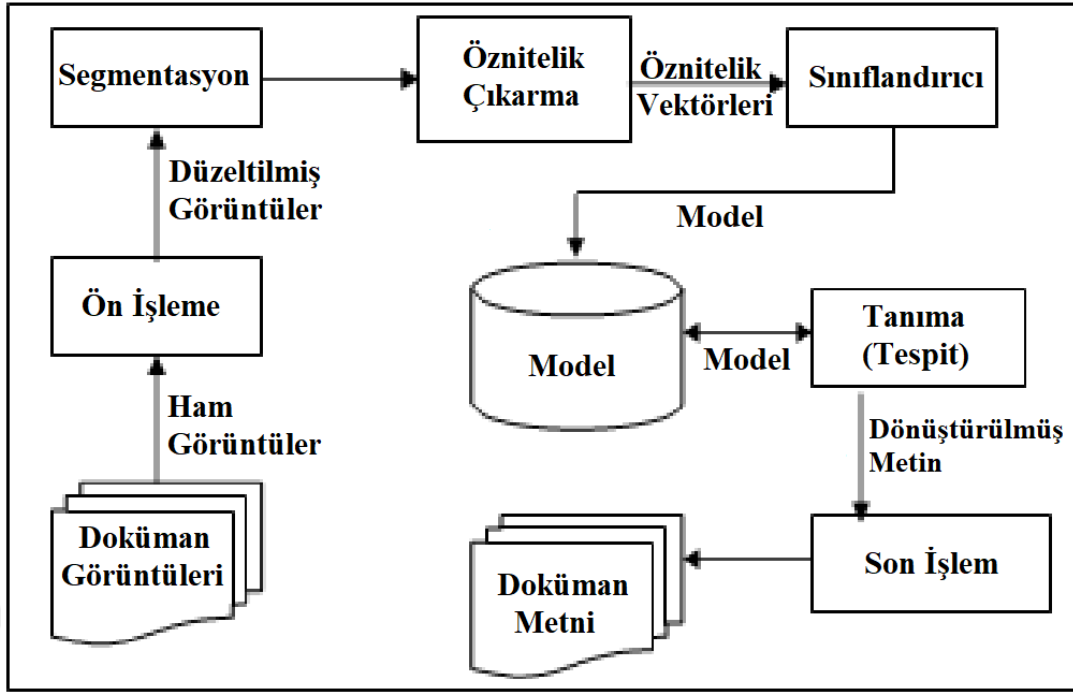
### 3.3. Optik Karakter Tanıma

Optik Karakter Tanıma (Optical Character Recognition) ya da bilinen kısaltması ile OCR, görüntülerden otomatik olarak metin çıkarıp işlemeyi amaçlayan bilgisayar sistemleri geliştirmeyi amaçlayan aktif bir araştırma alanıdır. Günümüzde el-yazımı ya da taranmış dokümanlar üzerinde yer alan bilgilerin bilgisayarda saklanmasına olan talep oldukça artmıştır. Bunu gerçekleştirmek amacıyla akla gelen en kolay yol, dokümanı tarayarak görüntüsünü bilgisayarda saklamaktır. Ancak doküman üzerinde yer alan veriler üzerinde arama yapılması ve bu veriyle çalışmanın getirdiği zorluklar sebebiyle bu yöntem pratik olmamaktadır. Bu sebepten dolayı, doküman görüntülerinde yer alan metinleri otomatik olarak dijital sisteme aktaran bir sisteme ihtiyaç doğmuştur (Hamad ve Kaya 2016).

OCR sistemlerinde yüksek başarı elde edilmesi için karşılaşılabilecek bazı sorunların çözülmesi gerekmektedir. Yazı tiplerindeki çeşitlilikler, yazı tipi boyutlarının farklılığı, doküman görüntü kalitesinin düşük olması gibi sebeplerden dolayı bazı durumlarda OCR işlemlerinde düşük başarı elde edilmektedir. OCR sistemleri insanlar kadar karakter tanıma ve işleme başarısına sahip olmayıp başarıları tamamen giriş olarak verilen görüntülerin kalitesine bağlıdır. Örnek olarak özellikle görüntülerin karanlık veya gürültülü olduğu durumlarda OCR sistemleri sayı olan “0” ve harf olan “o” arasında ayırım yapamamaktadır (Hamad ve Kaya 2016).

#### 3.3.1. Optik karakter tanıma sistemlerindeki ana aşamalar

OCR sistemi kurmak karmaşık bir işlem olup birden çok adımdan oluşmaktadır. İşlem sırasında bütün karakterlerin ve kelimelerin tek tek tanınarak dijital sisteme doğru bir şekilde aktarılması gerekmektedir. OCR sistemini baştan tasarlamak için optik tarama, ön işleme, Segmentasyon, öznitelik çıkarımı, eğitim gibi bazı adımların gerçekleştirilmesi gerekmektedir. Optik karakter tanıma sistemlerinde uygulanan ana işlem adımları Şekil 3.12’de gösterilmiştir.



Şekil 3.12. OCR sistemlerinde uygulanan ana işlem adımları (Awel ve Abidi 2019)

### 3.3.1.1. Optik tarama

Bu aşama, doküman fotoğrafının çekilmesi ya da bir tarayıcıyla taranması işlemlerini içermektedir. Ancak OCR sistemleri görüntü kalitesine bağımlı olduğundan görüntünün tarayıcı ile alınması OCR başarısı açısından daha iyi bir yöntemdir (Awel ve Abidi 2019).

### 3.3.1.2. Ön işleme

Ön işleme aşamasında görüntü kalitesini arttırmak, gürültü azaltmak için farklı ön işleme teknikleri uygulanmaktadır. Gürültüsü yok edilen görüntülerde karakterler belirgin hale geleceğinden sonraki aşama olan segmentasyon aşaması için daha uygun bir görüntü elde edilmiş olmaktadır. Ön işleme aşamasında gri tonlamaya çevirme, normalizasyon ve görüntü çevirme gibi yöntemler kullanılmaktadır (Awel ve Abidi 2019). Sunulan fatura sınıflandırma sisteminde bütün görüntülere Canny operatörü uygulanması, Hough dönüşümü uygulanması ve yamuk görüntülerin düzeltilmesi gibi ön işlemler gerçekleştirilmiştir. OCR sistemlerin kullanılan bazı ön işleme teknikleri Çizelge 3.1’de gösterilmiştir.

**Çizelge 3.1.** OCR sistemlerinde kullanılan bazı önemli ön işleme teknikleri (Hamad ve Kaya 2016)

| Ön İşleme Tekniği     | Açıklama                                                                                                          |
|-----------------------|-------------------------------------------------------------------------------------------------------------------|
| Gürültü Azaltma       | Görüntü üzerinde gürültü azaltma teknikleri kullanılarak görüntü kalitesinin artırılmasıdır.                      |
| Görüntü Eğim Düzeltme | Görüntünün alındığı cihazdan dolayı görüntüde yamukluk olacağından eğim düzeltimi yapılması gerekmektedir.        |
| Morfolojik İşlemler   | Eksik ya da fazla piksel içeren karakterlere piksel çıkarılması ya da piksel eklenmesi işlemlerinin yapılmasıdır. |
| Eşikleme              | Görüntüde yer alan önemli bilgilerin arka plandan ayrıştırılması işlemidir.                                       |

### 3.3.1.3. Segmentasyon

Gürültüden arındırılmış doküman görüntüleri, segmentasyon aşamasında çeşitli alt bileşenlere parçalanmaktadır. Segmentasyon aşamasında üç alt aşama bulunmaktadır. Birinci aşamada doküman görüntüsü üzerinde yer alan metinler satırlarına göre parçalanmaktadır. İkinci aşamada, ilk aşamada elde edilmiş olan satırlarda yer alan kelimeler alınmaktadır. Son aşamada ise kelimeler karakterlere parçalanmaktadır (Awel ve Abidi 2019).

### 3.3.1.4. Öznitelik çıkarımı

OCR sisteminde yer alan en riskli adımdır. Buradaki temel amaç, önemli örüntülerin çıkarılmasıdır. Çıkarılan özniteliklerde yer alan özniteliklerde, farklı karakterler arasında ayırım yapabilme özelliğinin olması beklenmektedir. Bu sayede sınıflandırma işlemi, bütün doküman verilerine ihtiyaç duymadan sadece elde edilen öznitelikler kullanılarak gerçekleştirilebilmektedir (Awel ve Abidi 2019).

### 3.3.1.5. Eğitim ve tanıma

OCR sistemlerindeki örüntü tanıma işlemi; şablon eşleştirme, istatistiksel teknikler, sentetik veya yapısal teknikler ve yapay sinir ağları kullanılarak gerçekleştirilebilmektedir. Sistem ayrıca eksik kelime haznesi ile ilişkili sorunların çözülebileceği şekilde eğitilmelidir (Awel ve Abidi 2019).

### 3.3.2. Optik karakter tanıma sistemlerinde karşılaşılan sorunlar

#### 3.3.2.1. Görüntü karmaşıklığı

İşlenecek görüntüde sadece metinlerin yer alması OCR için istenen bir durumdur. Ancak kamera tarafından alınan görüntüde metinlere ek olarak binalar, evler gibi diğer nesneler varsa metinlerin tanınması zorlaşmaktadır. Metin harici içeriklerin yer aldığı görüntülerde ön işleme aşaması zor olmakta olup OCR işleminin başarısı bundan etkilenmektedir (Hamad ve Kaya 2016, Awel ve Abidi 2019).

#### 3.3.2.2. Çok dilli ortamlar

Bazı diller için OCR sistemlerinin çalışması zor olmaktadır. Örnek olarak Korece, Çince ve Japonca gibi dillerde çok fazla sayıda karakter bulunmaktadır. Arapça dilinde karakterler birbirine bağlı bir şekilde yazılmaktadır. Etiyopya alfabesinde karakterler birbirine çok benzediği için karakterlerin bilgisayar tarafından birbirinden ayrılması zor olmaktadır (Awel ve Abidi 2019). Bu gibi sebeplerden dolayı OCR sırasında karakterlerin bölünmesi ve çıkarılması zor olmaktadır.

#### 3.3.2.3. Düzensiz ışıklandırma şartları

Doğal ortamlarda çekilen resimlerde düzensiz ışıklandırma ve gölgeler olabilmektedir. Bu da OCR için bir sıkıntı olup karakter tanıma, tespit etme ve segmentasyon gibi işlemlerin başarısını düşürmektedir. Işıklılandırma sorunları genellikle kullanıcılar tarafından kamera ile çekilen görüntülerde olmaktadır. Tarayıcı ile taranan görüntülerde bu sorunlar olmadığından, OCR sistemlerinde tarayıcı ile alınan görüntüler daha çok tercih edilmektedir (Hamad ve Kaya 2016).

### 3.3.2.4. Yazı tipleri

Arapça ve Hintçe 'de olduğu karakterlerin birbirine bağlı olması, yazıların eğik olması gibi sebeplerden dolayı OCR sistemleri segmentasyon sırasında sorun yaşamaktadır. Çünkü bu gibi durumlarda karakterlerin tespit edilmesi ve kelimelerin karakterlere parçalanması daha zor olmaktadır (Awel ve Abidi 2019).

### 3.3.2.5. Görüntünün eğri olması

Kamera yardımıyla çekilen görüntülerde resmin çekilme açısına bağlı olarak çekilen görüntülerde yer alan metinler eğri olabilmektedir. Bahsedilen eğrilik sorunu tarayıcı yardımıyla alınan görüntülerde yaşanmayıp genellikle kamera ile çekilen görüntülerde yaşanmaktadır. Eğrilik sorunun düzeltilmesi amacıyla RAST algoritması, Hough dönüşümü, Fourier dönüşümü gibi yöntemler kullanılmaktadır (Hamad ve Kaya 2016). Geliştirilen fatura sınıflandırma sisteminde eğrilik sorununun çözülmesi amacıyla Hough dönüşümü kullanılarak varsa eğik metin içeren görüntülerin düzeltilmesi sağlanmıştır.

### 3.3.2.6. Görüntünün bulanık olması

Bu sorun özellikle kamera ile uzaktan çekilmiş görüntülerin yakınlaştırılması sonrası yaşanmaktadır. Bu gibi görüntülerde bulanıklık olduğundan OCR sisteminin karakter tespit ve tanıma başarısı azalmaktadır. Segmentasyon ve karakter tanıma aşamalarında yüksek başarı elde edilebilmesi için bulanık görüntülerin ön işleme aşamasında keskinleştirilmesi gerekmektedir (Awel ve Abidi 2019).

## 3.4. Doğal Dil İşleme

Doğal dil işleme (Natural Language Processing) ya da bilinen kısaltmasıyla NLP, insan dili ve bilgisayarlar arasındaki etkileşimlerle ilgilenen bir bilgisayar bilimi dalıdır. Doğal diller, insanlar tarafından konuşulan dillerdir. Doğal diller aynı zamanda insanların çevreleri vasıtasıyla öğrenip diğer insanlarla iletişim kurmak için kullandığı dillerdir. İletişimin türü ne olursa olsun, bilgi ve duygu aktarımında doğal diller kullanılmaktadır. Mevcut durumda hâlâ bu dillerin işlenmemiş çoğu formu bilgisayarlar tarafından

anlaşılamamaktadır. Doğal dil işleme, bu hedefi yerine getirebilmek için kullanılan teknikler bütünüdür (Reshamwala ve ark 2013). Doğal dil işleme alanı, makine öğrenmesi ve hesaplama dayalı dilbilimi alanlarının bilgisayar bilimiyle etkileşimli çalışması ile oluşan metotlar bütünüdür. NLP, gelecekte çok büyük önem arz edecektir. NLP alanının gelecekte bu kadar önemli olmasının nedeni; bilgi yığınlarını, ses biçiminde verileri veya her ikisini içeren verileri girdi olarak alarak modellerin oluşturulup işlenmesini kolay bir hale getirmesidir (Jain ve ark 2018).

Doğal dil işleme; girişe verilen metinden anlam çıkarmak, metnin gramer yapısını çıkarmak için kullanılan teknikler bütünüdür. Eğitim sistemlerinde, yinelenen verilerin tespiti gibi konularında NLP algoritmaları kullanışlıdır (Reshamwala ve ark 2013). Metin türündeki verilen daha etkili bir şekilde işlenebilmesi için farklı algoritmalar geliştirilmiştir. Bunlardan birisi de NER kısaltmasıyla bilinen Adlandırılmış Varlık Tanıma (Named Entity Recognition) algoritmasıdır (Jain ve ark 2018).

#### **3.4.1. Adlandırılmış varlık tanıma**

NER algoritması; ilişkili isimlerin tespit edilmesi, tespit edilen isimlerin bağlı oldukları varlıklara göre sınıflandırılmasını amaçlamaktadır. NER algoritması; girdi olarak verilen metin veya ses dosyası içerisinde bulunması muhtemel olan isimleri, insanları veya diğer ilişkili bilgileri bulmayı hedeflemektedir. NER modeli iki adımda çalışmaktadır. İlk adımda sınıflandırma amacıyla metin parçalara ayrılır. Bu parçalar sonraki adımda kişi ismi, organizasyon, mekân gibi önceden tanımlanmış kategorilere sınıflandırılmaktadır (Jain ve ark 2018).

NER algoritması makine çevirisi, duygu analizi ve bilgi çıkarımı gibi farklı konularda kullanılan önemli bir NLP algoritmasıdır. Diğer birçok NLP uygulaması gibi NER algoritması da internet kullanımının artmasına ve daha fazla kullanıcı verisinin oluşmasına bağlı olarak giderek daha fazla önem kazanmaktadır. Son yıllarında Türkçe NER konusunda çeşitli çalışmalar yapılmıştır. Türkçe dili için NER algoritması ile gerçek kullanıcı verileri üzerinde çalışılırken karşılaşılan bazı önemli sorunlar vardır. Bunlardan bir tanesi, özel isimlerin farklı anlamlarda kullanılabilmesidir. Örnek olarak “Aydın” kelimesi hem özel bir kişi ismi hem de bir şehir ismi olarak kullanılmaktadır. Aynı kelime küçük harflerle yazıldığında ise aydınlatılmış anlamına gelmektedir. Bu da “Aydın” özel ismi ve aydınlatılmış anlamına gelen diğer kelime arasında ayrım yapmayı zorlaştırmaktadır. Türkçe dilinde gerçek kullanıcı verileri üzerinde yapılan NER

çalışmalarında karşılaşılan diğer bir problem ise kullanıcılar tarafından yapılan noktalama hatalarıdır. Örnek olarak “Aydın’lara” yerine “aydınlara” yazılması durumunda yine özel bir isim olan Aydın isminin metinden çıkarılması zorlaşmaktadır. (Çelikkaya ve ark 2013).

### 3.4.2. Metin benzerlik algoritmaları

Metin benzerlik algoritmaları, hızlı bir şekilde NLP tabanlı uygulamalarda anahtar bir bileşen olmaktadır. Bilgi çıkarımı, otomatik soru cevaplama, makine çevirisi ve diyalog sistemleri gibi uygulamalarda metin benzerlik algoritmaları yoğun bir şekilde kullanılmaktadır (Wang ve Dong 2020).

Geçtiğimiz yıllarda anlamsal benzerlik teknikleri konularında birçok çalışma sunulmuştur. Birçok bilim insanı; metin benzerlik yöntemlerini Wikipedia’da olduğu gibi istatistik ve bilgi gibi konulara göre sınıflandırmaktadır. Bu şekilde yapılan sınıflandırmada metin benzerliği önemsenmeyip sadece metin temsilleri göz önünde bulundurulmaktadır. Bu sırada sinir ağı temsil öğrenmesinin gelişmesiyle beraber bazı anlamsal karşılaştırma yöntemleri ve grafik yöntemlerinin göz önünde bulundurulması ihtiyacı doğmuştur. Metin benzerliği sadece metinler arasındaki benzerliği hesaba katmayıp aynı zamanda kelimeler arasında paylaşılan anlamsal özellikler arasındaki geniş bakış açısını da göz önünde bulundurmaktadır. Örnek olarak kral ve adam kelimeleri ilişki açısından birbirine yakın olsa da bu kelimeler arasında anlamsal bir benzerlik yoktur. Ancak kral ve kraliçe kelimeleri arasında anlamsal bir benzerlik söz konusudur (Wang ve Dong 2020).

Anlamsal benzerliğe ek olarak metinler arasında sözcük yapısı bakımından benzerlik olabilmektedir. Kelimeler benzer bir anlama sahipse kelimeler arasında anlamsal benzerlik vardır ancak kelimeler benzer karakter dizileri içeriyorsa kelimeler arasında sözcük yapısı bakımından benzerlik vardır. Sözcük yapısı bakımından hesaplanan benzerlik algoritmaları, metin tabanlı yöntemler olarak isimlendirilmektedir. Metin benzerlik algoritmaları, metinleri karakter dizilimi açısından analiz ederek kelimeler arasında var olan benzerlik veya uzaklık hesaplamasını yapmaktadır (Gomaa ve Fahmy 2013). Metin tabanlı yöntemler, en eski ve en basit yöntemler olmalarına rağmen hâlâ en popüler yöntemlerdir. Metin tabanlı yöntemler metin dizileri ve karakter birleşimleri üzerinde çalışarak hesaplama yapmaktadır. Temel olarak karakter tabanlı ve simge tabanlı olmak üzere iki çeşit metin benzerlik ölçüm tekniği bulunmaktadır

(Prasetya ve ark 2018). Sunulan fatura sınıflandırma sistemin son aşamasında fatura görüntüsü üzerinde arama yapılması aşamasında OCR sistemine ek olarak karakter tabanlı metin benzerlik algoritmaları kullanılmıştır. Hamming uzaklığı (Hamming 1950), Levenshtein uzaklığı (Levenshtein 1965) ve Jaro-Winkler uzaklığı (Winkler 1990) yöntemleri sunulan fatura sınıflandırma çalışmasında kullanılmıştır.

#### 3.4.2.1. Hamming uzaklığı

Hamming uzaklığı, değişken çiftleri arasındaki sapma sayısına göre iki nesne arasındaki uzaklığı ifade eden bir metriktir. Hamming uzaklığı genellikle metin ve bit tabanlı analiz için kullanılmakla beraber bazı durumlarda sayısal değişkenlerde de kullanılmaktadır. Bu uzaklık hesaplamasında belirli bir eşik değerinin altında farka sahip değişkenler eşit sayılmaktadır. Temelde eşik değeri olarak sıfır alınmaktadır. Sıfırın altındaki eşik değerleri tanımlı değildir. Hamming uzaklığının hesaplanması Denklem 3.12’de gösterilmiştir.

$$HD(m, n) = \sum_{k=0}^{n-1} [y_{m,k} \neq y_{n,k}] \quad (3.12)$$

Formülde geçen HD ifadesi  $m$  ve  $n$  nesneleri arasındaki Hamming uzaklığını temsil etmektedir. Formüldeki  $k$  değeri ise  $n$  değişkenlerinden okunan  $y$  değerlerinin indeksidir. Hamming mesafesi  $k$  ile eşleştirilmiş değişkenler arasındaki uyumsuzlukların sayısını vermektedir (Shehu ve ark 2015). Eşit uzunluktaki metinler karşılaştırılırken, metinlerde karşılıklı olarak bulunan farklı karakterlerin adedi Hamming uzaklığını vermektedir (Wang ve Dong 2020). Örnek olarak “1010” ve “1001” bit tabanlı ifadeleri arasındaki Hamming uzaklığı 2’dir. Çünkü iki ifadede karşılıklı iki karakter birbirinden farklıdır. Benzer bir şekilde “abcd” ve “abab” kelimelerinin son iki karakterleri birbirinden farklı olduğundan kelimeler arasındaki Hamming uzaklığı 2’dir (Ledru ve ark 2009).

Hamming uzaklığı farklı uzunluktaki girdilere de uygulanabilmektedir. Bunun için öncelikle en kısa uzunluktaki girdi, dolgu yöntemi kullanılarak en uzun girdiye eşitlenmektedir. Örnek olarak “ab” ve “abcde” kelimeleri arasındaki Hamming uzaklığını hesaplamak için “ab” kelimesine  $x$  karakterleriyle dolgu uygulanarak “abxxx” olarak

değiştirilmektedir. Daha sonra “abxxx” ve “abcde” kelimeleri arasındaki Hamming uzaklığı 3 olarak hesaplanmaktadır (Ledru ve ark 2009).

### 3.4.2.2. Jaro-Winkler uzaklığı

Jaro-Winkler uzaklığı, iki metin arasındaki ortak karakterlerin sırasına ve sayısına dayanan bir metin benzerlik algoritmasıdır. Jaro-Winkler uzaklığı özellikler kişi isimleri gibi kısa metinlerde kullanılmak üzere tasarlanmış olup bu metinlerde daha iyi sonuç vermektedir (Özkan 2012). Jaro-Winkler uzaklığı genellikle tekrarlı veri tespitinde kullanılmaktadır. İki metin arasında Jaro-Winkler uzaklığı arttıkça metinler arasındaki benzerlik artmaktadır. Jaro-Winkler uzaklığı için 0 değeri hiç benzerlik olmadığını, 1 değeri ise metinlerin tamamen aynı olduğunu ifade etmektedir. Jaro-Winkler uzaklığının hesaplanabilmesi için öncelikle Jaro uzaklığının hesaplanması gerekmektedir. Jaro uzaklığının hesaplanması Denklem 3.13’te gösterilmiştir.

$$d_j = \frac{1}{3} \times \left( \frac{m}{|s1|} + \frac{m}{|s2|} + \frac{m-t}{m} \right) \quad (3.13)$$

Denklemden  $s1$  ve  $s2$  değerleri sırasıyla birinci ve ikinci metnin uzunluklarını,  $t$  değeri karakter yer değişikliği sayısını,  $m$  değeri ise metinlerde eşleşen karakter sayısını ifade etmektedir (Manaf ve ark 2019). Jaro uzaklığı hesaplandıktan sonra Denklem 3.14’te gösterildiği şekilde Jaro-Winkler uzaklığı hesaplanmaktadır.

$$Jaro(x,y) + \frac{\max(4, LCP(x,y))}{10} (1 - Jaro(x,y)) \quad (3.14)$$

Denklemden geçen  $Jaro(x,y)$  ifadesi Jaro uzaklığını,  $LCP(x,y)$  ifadesi ise  $x$  ve  $y$  kelimelerinde bulunan en uzun ortak ön ek uzunluğunu ifade etmektedir (Deza ve Deza 2014).

### 3.4.2.3. Levenshtein uzaklığı

Levenshtein uzaklığı, iki metin arasındaki farklılık miktarını hesaplamak için kullanılan bir metriktir. Levenshtein uzaklığı ayrıca bir metni diğerine dönüştürmek için gereken dönüşüm maliyeti olarak da tanımlanmaktadır. Metin dönüşümü sırasında yapılan işlemler; kopyalama, ekleme, silme ve değiştirme olarak gruplandırılmaktadır.

İki metindeki  $n$ . karakterlerin aynı olması durumunda kopyalama işlemi, karakter eklenmesi durumunda ekleme işlemi, metinlerdeki  $n$ . karakterin eşit olmaması durumunda değiştirme işlemi gerçekleştirilmektedir. Metinlerden fazla karakterlerin silinmesi ihtiyacı olduğunda ise silme işlemi gerçekleştirilmektedir.

Levenshtein uzaklığı hesaplanırken Levenshtein Matrisi(LM) oluşturulmaktadır. LM, metin eşleştirmesi sırasında harcanan işlem maliyetlerini göstermektedir. Algoritmada ilk metin  $str1$  ve ikinci metin  $str2$  olmak üzere;  $str1$  metni kolon uzayı,  $str2$  metni ise satır uzayı olarak yerleştirilmektedir. Levenshtein uzaklığına ait hesaplama işlemleri Denklem 3.15, Denklem 3.16 ve Denklem 3.17’de gösterilmiştir. Formüllerde geçen  $i$  ve  $j$  ifadeleri sırasıyla matris üzerindeki mevcut kolon ve satır konumlarını temsil etmektedir (Özkan 2012).

$$LM(str1, str2, i, 0) = i \quad for \quad i = 0, 1, 2, \dots, kolonBoyutu + 1 \quad (3.15)$$

$$LM(str1, str2, 0, j) = j \quad for \quad j = 0, 1, 2, \dots, satirBoyutu + 1 \quad (3.16)$$

$$LM(str1, str2, i, j) = \min \left\{ \begin{array}{ll} LM(str1, str2, i - 1, j - 1) & (Kopyalama) \\ LM(str1, str2, i - 1, j - 1) + 1 & (Değiştirme) \\ LM(str1, str2, i, j - 1) + 1 & (Ekleme) \\ LM(str1, str2, i - 1, j) + 1 & (Silme) \end{array} \right\} \quad (3.17)$$

Levenshtein uzaklığı farklı uzunluktaki metinlere uygulanabilmektedir. Levenshtein kullanılarak yapılan uzaklık hesaplamasında kullanılan algoritma, hem zaman hem de bellek açısından  $O(m \times n)$  maliyete sahip olup  $m$  ve  $n$  değerleri metin uzunluklarını temsil etmektedir (Ledru ve ark 2009). Örnek olarak “similar” ve “silver” kelimeleri arasındaki Levenshtein uzaklığı 4 olarak hesaplanmaktadır. Çünkü ilk kelimeyi ikincisine çevirmek için 4 karakter değişikliği gerekmektedir. Bu işlemi 4 yerine 3 adımda yapmanın bir yolu bulunmamaktadır. Benzer bir şekilde “property” ve “properly” kelimeleri arasındaki Levenshtein uzaklığı 1, “congruent” ve “congruous” kelimeleri arasındaki Levenshtein uzaklığı ise 3 olarak hesaplanmaktadır (Anjali ve ark 2020).

### 3.5. Veri Önleme Teknikleri

Sistemin eğitim aşamasında fatura görüntüleri kullanılmış olup bu görüntüler bir bankadan alınmıştır. Kullanılan faturalar banka personelleri tarafından gerçekleştirilen fazla mesai, seyahat gibi işlemlere ait faturalardır. Faturalar daha sonra personeller tarafından tarayıcı kullanılarak taranmakta veya mobil cihaz kamerası kullanılarak fotoğrafı çekilmektedir. Fatura görüntüleri personeller tarafından alındığı için bazı görüntülerde ön işleme ihtiyacı olmaktadır. Bölümün devamında fatura görüntüleri üzerinde yapılan önleme teknikleri anlatılmıştır.

#### 3.5.1. Fatura verisi içermeyen görüntülerin ayıklanması

Fatura görüntülerinin personeller tarafından tarayıcı kullanılarak arkalı-önlü taranması durumunda iki görüntü elde edilmekte ve görüntülerden birisi metin içermemekte ya da genellikle faturayla alakalı olmayan veriler içermektedir. Kullanıcılar tarafından mobil cihaz kamerası kullanılarak alınan görüntüler ise bazen fatura ile alakalı olmayan görüntüler içerebilmektedir.

Bahsedilen görüntülerin veri setinden çıkarılması için öncelikle bankadan alınan bütün görüntülere OCR uygulanarak fatura görüntüleri arasında yer alan ortak fatura terimleri toplanmıştır. Daha sonra görüntülere ikinci defa OCR uygulanarak hiç metin içermeyen ya da hiç fatura terimi içermeyen görüntüler belirlenerek veri setinden çıkarılmıştır. Görüntülere OCR uygulama aşamasında Tesseract isimli açık kaynak kodlu optik karakter tanıma sistemi kullanılmıştır. Tesseract, 1990'lı yıllarda Hewlett Packard tarafından geliştirilmiştir. 2005 sonlarında açık kaynak kodlu hale getirilen Tesseract, düşük kaliteli dokümanlarda düşük performans gösterebilmektedir (Sidhwa ve ark 2018). OCR uygulaması sonrası veri setinden otomatik olarak çıkarılan örnek görüntüler Şekil 3.13'te gösterilmiştir.



Şekil 3.13. OCR sonrası anlamlı veriler içermediği için veri setinden çıkarılan örnek görüntüler

### 3.5.2. Eğik Metin İçeren Görüntülerin Düzeltilmesi

Veri seti içerisinde yer alan görüntüler farklı kullanıcılar tarafından farklı açılarda çekilmektedir. Eğik açılı görüntüler hem OCR performansını hem de CNN tarafından yapılan öznitelik çıkarma performansını etkilemektedir. Oluşabilecek bu sorunların önüne geçebilmek için görüntülerde yer alan metinlerin eğimleri hesaplanarak eğik olan görüntülerin eğiklikleri düzeltilmiştir. Eğim düzeltme çalışması sırasında öncelikle veri setindeki görüntüler siyah-beyaz formata çevrilmiştir. Daha sonra Canny isimli kenar tespit operatörü ve Hough dönüşümü kullanılarak eğim tespiti yapılmıştır.

#### 3.5.2.1. Canny kenar tespit operatörü

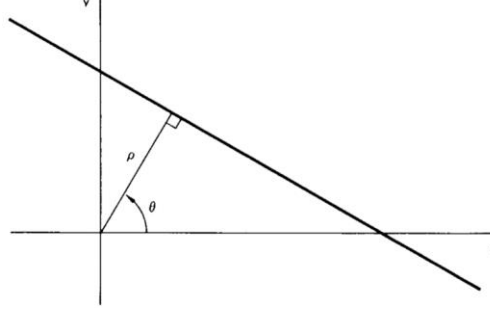
Canny operatörü, John F. Tarafından geliştirilmiş olup birden fazla adımdan oluşan algoritması sayesinde görüntülerde kenar tespiti yapmak için kullanılmıştır (Khan ve Mufti 2016). Canny algoritması temel olarak dört adımdan oluşmaktadır. Birinci adımda Gaussian filtresi kullanılarak gürültü azaltma işlemi yapılmaktadır. İkinci adımda gradyan hesaplanmakta olup üçüncü adımda kenar adayları olmayan yani maksimum olmayan piksellerin yok edilmesi gerçekleştirilmektedir. Son adımda ise piksellerin kenar pikseli olup olmadığına karar verilmektedir (Ha ve Shakeri 2016). Bankadan alınan fatura veri setinde yer alan görüntüler siyah-beyaz formata çevrildikten sonra Canny operatörü uygulanarak kenar tespit işlemi gerçekleştirilmiştir.

#### 3.5.2.2. Hough dönüşümü

Hough dönüşümü, dijital görüntü işleme çalışmalarında kullanılan bir öznitelik çıkarma tekniğidir. Normal şartlarda Hough dönüşümü görüntülerde çizgi tespiti yapmak için kullanılsa da görüntüde yer alan rastgele şekillerin konumlarının tespiti için de kullanılabilir. Hough dönüşümü çizgi tespiti yapmak üzere tasarlanmış olup Denklem 3.18'de gösterildiği gibi doğrunun parametrik temsilini kullanmaktadır.

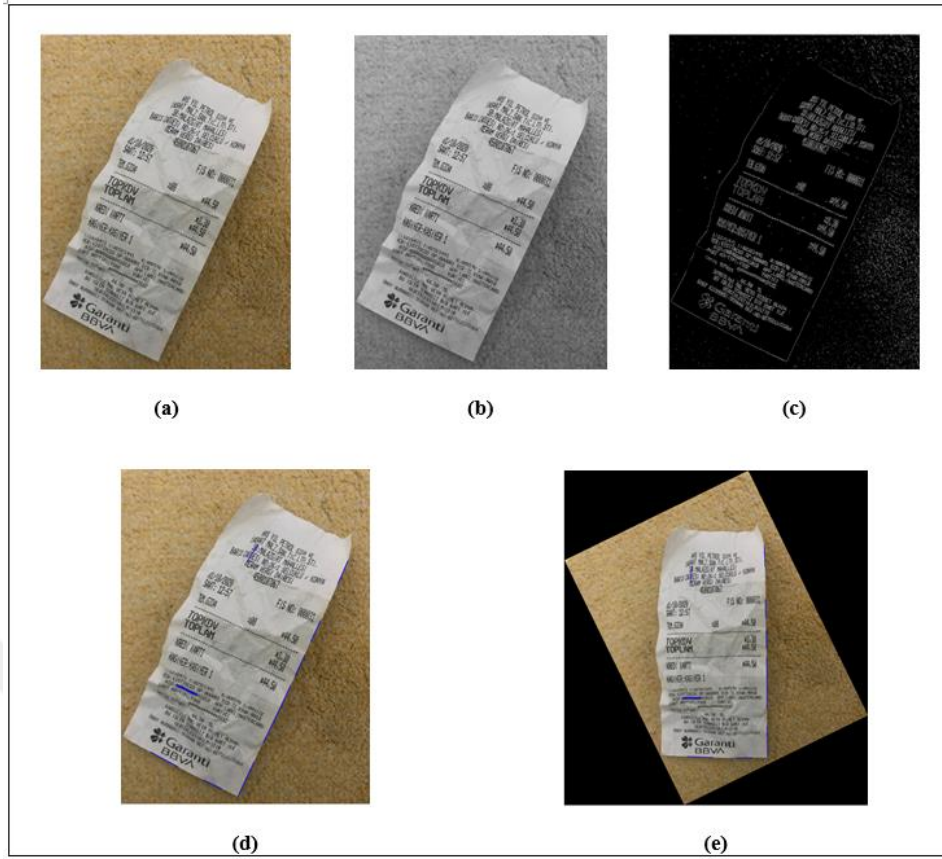
$$\rho = x \cos \theta + y \sin \theta \quad (3.18)$$

Formülde geçen  $\rho$  , doğruya dik olan vektöre olan uzaklığı ifade eder.  $\theta$  ise bu vektör ile x-ekseni arasındaki açıyı ifade eder (Chunhavittayatera ve ark 2006). Doğruya ait Hough uzayının temsili Şekil 3.14'te gösterilmiştir.



Şekil 3.14. Doğruya ait Hough uzayının temsili (Duda ve Hart 1972)

Veri setinde yer alan fatura görüntüleri öncelikle siyah-beyaz formata çevrilmiştir. Siyah-beyaz formattaki görüntülere daha sonra Canny operatörü uygulanarak görüntülerde kenar tespiti yapılmıştır. Fatura görüntülerine daha sonra Hough dönüşümü uygulanarak çizgi tespiti yapılmıştır. Fatura satırlarına ve sınırlarına Hough dönüşümü yardımıyla doğrular çizilmiştir. Sonuçta oluşan doğru listesindeki her bir doğrunun eğimi hesaplanarak toplanmıştır. Son olarak eğim toplamalarının ortalaması alınarak fatura görüntüsünün eğim açısı tespit edilmiştir. Hesaplanan eğim açısı kullanılarak ilgili görüntü, hesaplanan açı miktarınca çevrilerek düzeltilmiştir. Canny operatörünün uygulanması ve Hough dönüşümü sırasında OpenCV Kütüphanesi kullanılmıştır. OpenCV, Intel Corporation tarafından geliştirilen açık kaynak kodlu bir bilgisayarla görme kütüphanesidir. Bu kütüphane, birçok görüntü işleme algoritması sunmaktadır (Jia 2010). Veri setinde yer alan örnek bir fatura görüntüsünün sırasıyla siyah beyaz formata çevrilmesi, Canny operatörünün uygulanması ve Hough dönüşümünün uygulanması işlemleri Şekil 3.15'te gösterilmiştir.



**Şekil 3.15.** Örnek bir fatura görüntüsüne Canny Operatörünün ve Hough Dönüşümünün uygulanması aşamaları. (a) Orijinal görüntü. (b) Siyah beyaz görüntü. (c) Canny operatörü uygulanmış görüntü. (d) Hough dönüşümü kullanılarak doğrular çizilmiş görüntü. (e) Doğruların eğim ortalaması alınarak elde edilen eğim açısına göre düzeltilmiş görüntü

### 3.5.3. Sıfır dolgu uygulanması

Evrişimsel sinir ağı mimarisindeki dezavantajlardan birisi, sabit boyutlu görüntüleri girdi olarak almasıdır. Bu da girişe verilen bütün görüntülerin aynı boyutta olmasını zorunlu kılmaktadır. Dolayısıyla bütün girdilerin eşit en-boy oranına da sahip olması gerekmektedir. Ancak gerçek dünya örneklerinde bazı görüntüler yatay bazıları dikey olduğundan dolayı en-boy oranları çok farklı olabilmektedir (Tensmeyer ve Martinez 2017). Görüntü boyutları ve en-boy oranları farklı da olsa CNN ağına verilmesi için yeniden boyutlandırma yapılmalıdır. Ancak farklı en-boy oranlarına sahip görüntülerin aynı boyuta getirilmesi durumunda en-boy oranı değiştiğinden görüntü üzerindeki bazı önemli öznitelikler kaybolmaktadır. Bu da CNN mimarisinin sınıflandırma başarısını olumsuz yönde etkileyecektir. Ek olarak, çok büyük görüntülerin CNN ağı kullanılarak eğitilmesi hem zaman hem de bellek açısından maliyetli olduğundan büyük görüntülerin CNN ağına verilmeden önce küçültülmesi

gerekmektedir. Küçülme arttıkça da aynı şekilde görüntü üzerindeki öznitelikler kaybolacaktır (Hashemi 2019).

En-boy oranı sorununu çözmek için sıfır-dolgu yöntemi kullanılarak görüntüler en ve boyları eşit olacak şekilde yeniden boyutlandırılmıştır. Sıfır-dolgu yöntemine göre görüntünün kenarlarına sabit değerli ve görüntü ile alakası olmayan pikseller konarak görüntünün yeniden boyutlandırılması sağlanır. Bu yöntem ile görüntü sınırlarındaki uzaysal dağılım değiştirilmiş olur (Nguyen ve ark 2019). Sıfır-dolgu yönteminin iki önemli avantajı bulunmaktadır. Birinci avantajı; sıfır-dolgu kullanılarak boyutlandırılan görüntüler daha sonra tekrar yeniden boyutlandırılırsa en-boy oranı korunacağından öznitelik kaybı veya deformasyon yaşanmayacaktır. Bir diğer avantajı ise hesaplama işlemlerini hızlandırması ve hesaplama maliyetlerini düşürmesidir. Birbirine komşu olan sıfır-giriş değerleri, karşılık gelen evrişim katmanını aktifleştirmeyeceğinden bu bölgelerdeki ağırlıkların güncellenmesine gerek kalmayacaktır. Sıfır-piksel olan kısımlarda ağırlık güncellenmesi olmayacağından CNN ağı bu kısımlardan öğrenme yapmayıp, sadece görüntünün merkezinde yer alan ve sabit olmayan piksellerden öğrenmektedir. Bu da CNN ağının eğitim süresini kısaltmaktadır (Hashemi 2019). Sıfır-dolgu yaklaşımına göre fatura görüntüleri kare haline gelecek şekilde boyutlandırılmıştır. Burada mevcut fatura üzerinde herhangi bir ölçekleme işlemi yapılmamış olup sadece ihtiyaca göre etrafına mavi pikseller eklenmiştir.

Sonuçta elde edilen veri setindeki görüntülerin boyutları birbirine eşit olmasa da bütün görüntülerin en-boy oranı artık 1 olmuştur. Ek olarak bundan sonraki yeniden boyutlandırma işlemlerinde görüntü merkezinde yer alan fatura görüntüsünün en-boy oranı değişmeyecektir. Dolayısıyla öznitelik kaybı yaşanmayacaktır.

### 3.6. Veri Arttırma

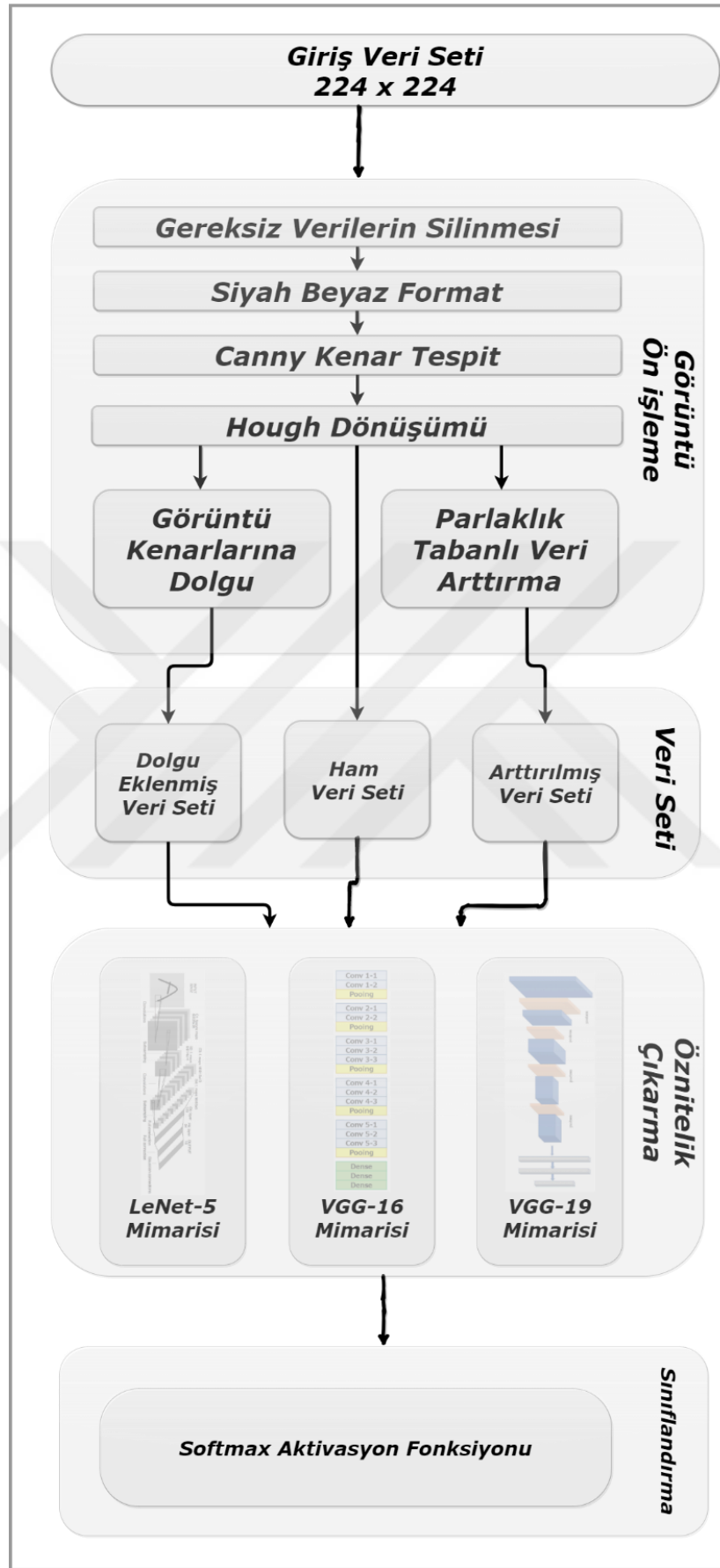
Dijital görüntü verileri; en, boy ve renk kanallarının tensörleri olarak kodlanmaktadır. Tensör, derinliği olan bir matris olarak tanımlanabilir. Renkli görüntüler, RGB kanallarından oluştuğu için her görüntü aslında derinliği 3 olan bir tensörü ifade etmektedir. Görüntülerdeki bu yapı renk uzayı olarak isimlendirilmektedir. RGB kanallarından herhangi ikisinin çıkarılıp yerine sıfır değerlerini içeren matrislerin konulmasıyla görüntünün tek kanal için bir temsili oluşturulmuş olur. Ek olarak, basit matris hesaplamaları kullanılarak RGB değerlerinin değiştirilmesiyle görüntünün parlaklık değeri arttırılabilir ya da azaltılabilir. Çok parlak veya çok karanlık görüntüleri

düzeltilmenin en kısa yolu ise, görüntü boyunca bir döngü kurarak piksel değerlerini sabit bir değer kullanarak arttırmak ya da azaltmaktır (Shorten ve Khoshgoftaar 2019).

Bankadan alınan ham fatura görüntüleri farklı kullanıcılara ait olup her birisi farklı kameralar aracılığıyla çekilmiş veya tarayıcı kullanılarak taranmıştır. Dolayısıyla her görüntünün ışık miktarı farklıdır. Geliştirilmiş olan evrimsel sinir ağı mimarisi tabanlı fatura sınıflandırma sisteminin farklı ışık değerlerine sahip faturaları tanıyabilmesi için farklı ışık miktarındaki fatura görüntüleriyle eğitilmesi gerekmektedir. Bu amaçla, sıfır-dolgu uygulayarak elde edilen veri setindeki görüntülerin parlaklık değerleri değiştirilerek yeni görüntüler elde edilmiştir. Bu yöntem kullanılarak veri setindeki her görüntüden parlaklık değerleri farklı iki görüntü daha elde edilmiştir. Dolayısıyla bu görüntü arttırma tekniğiyle, 4000 olan veri seti boyutu 12000 görüntüye çıkarılmıştır. Görüntü arttırma işlemi Keras isimli kütüphanede yer alan ImageDataGenerator sınıfı kullanılarak gerçekleştirilmiş ve parlaklık aralığı olarak [0.2, 1] aralığı kullanılmıştır. Keras, Python ile yazılmış yüksek seviye bir sinir ağı kütüphanesidir. Keras isimli kütüphane, arkada TensorFlow veya Theano ile çalışmakta olup sinir ağlarının kullanımını ve kurulumunu kolaylaştırmak için yazılmıştır (Arora ve Bhatia 2018).

### 3.7. Öznitelik Çıkarma

Öznitelik çıkarma aşamasında fatura görüntülerindeki şablona has öznitelikler çıkarılmıştır. Fatura görüntülerindeki şablona özel özniteliklerin çıkarılması amacıyla Lenet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 isimli popüler CNN mimarileri kullanılmıştır. Öznitelik çıkarımından sonra Softmax aktivasyon fonksiyonunu kullanarak şablona göre sınıflandırma işlemi gerçekleştirilmiştir. Sunulan yöntemin veri ön işleme, öznitelik çıkarma ve sınıflandırma aşamalarına ait çalışma şeması Şekil 3.16'da verilmiştir.



Şekil 3.16. Sunulan yöntemde bulunan veri ön işleme, öznitelik çıkarma ve sınıflandırma aşamaları

### 3.8. Veri Setleri

Bu bölümde, sunulan projenin eğitim aşamasında kullanılan veri setleri açıklanmıştır. Projenin eğitim aşamasında fatura görüntüleri kullanılmış olup bu görüntüler bir bankadan alınmıştır. Bankadan alınan fatura görüntüleri veri setinde dört fatura sınıfına ait toplam 4000 fatura görüntüsü bulunmaktadır. Fatura görüntülerine ait sınıf isimleri ve her bir sınıfa ait örnek fatura sayıları Çizelge 3.2’ de verilmiştir.

Çizelge 3.2. Fatura sınıfları ve sınıf başına fatura görüntüsü sayıları

| Fatura Sınıfı | Fatura Görüntüsü Sayısı |
|---------------|-------------------------|
| Kart Dolum    | 1000                    |
| Otopark       | 1000                    |
| Taksi         | 1000                    |
| Yemek         | 1000                    |
| <b>Toplam</b> | <b>4000</b>             |

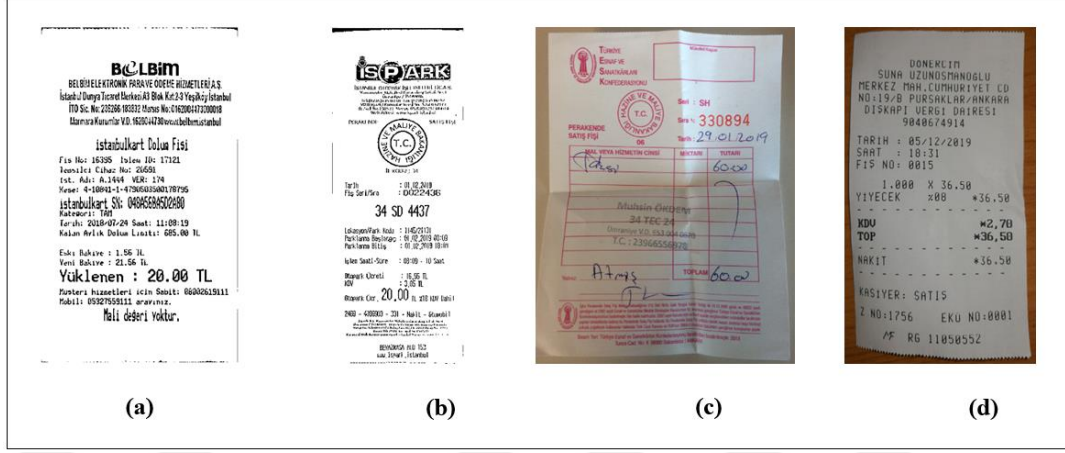
Proje kapsamında toplamda üç veri seti kullanılmıştır. Oluşturduğumuz veri setlerine birer isim verilmiş olup çalışmanın devamında bu isimler kullanılmıştır. Veri setleri şu şekildedir:

- BAFGV1: Bankadan alınan ham fatura görüntülerine Canny Operatörü ve Hough Dönüşümü uygulanarak elde edilen fatura veri seti.
- BAFGV2: BAFGV1 veri setine sıfır-dolgu uygulanarak oluşturulan veri seti.
- BAFGV3: BAFGV2 veri seti kullanılarak parlaklık değeri tabanlı veri artırma tekniğine tabi tutularak elde edilen veri seti.

#### 3.8.1. BAFGV1 veri seti

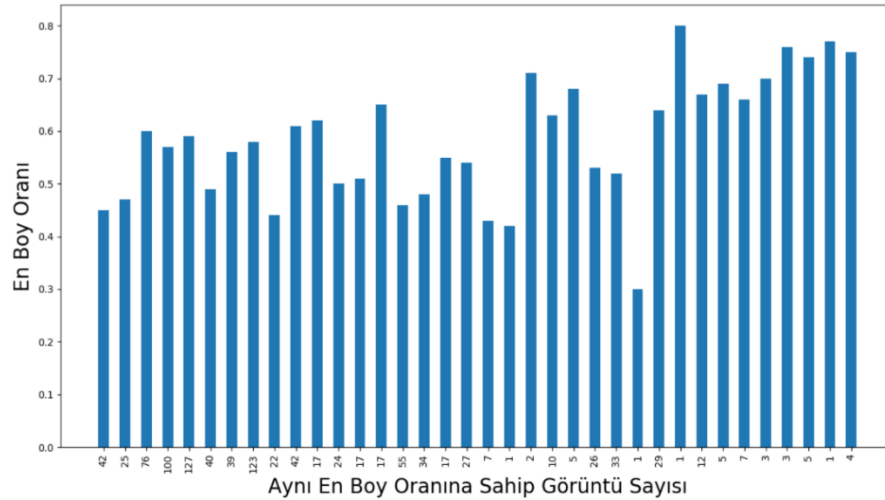
BAFGV1 veri setinde bankadan alınan ham fatura görüntülerine Canny Operatörü ve Hough Dönüşümü uygulanarak işlenmiş fatura görüntüleri bulunmaktadır. BAFGV1 veri setinde bulunan her bir sınıfa ait örnek birer fatura görüntüsü Şekil 3.17’de verilmiştir. Taksi sınıfındaki fatura görüntülerinde diğer sınıftaki görüntülerden farklı olarak el yazısı veriler de yer almaktadır. Fatura görüntülerinin üst tarafında bilgisayar puntosu yazılar olsa da faturaların genelinde el yazısı kısımlar da bulunmaktadır. Geliştirilen fatura sınıflandırma uygulaması taksi faturalarındaki sadece bilgisayar puntosu alanları tespit edebilmektedir. El yazısı fatura alanlarının tespit edilip okunması

konusunda bir geliştirme yapılmamıştır. Taksi sınıfındaki fatura görüntüleri sadece fatura sınıflandırma aşamasına dâhil edilmiştir.



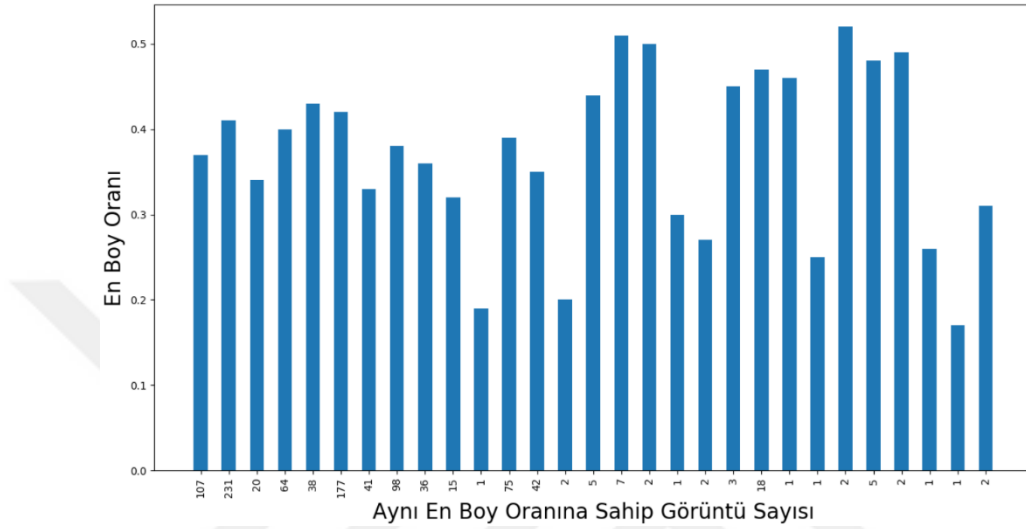
Şekil 3.17. BAFGV1 veri setinde bulunan her bir sınıfa ait görüntü örnekleri. (a) Kart dolum fişi. (b) Otopark fişi. (c) Taksi fişi. (d) Yemek fişi

Her fatura görüntüsü farklı kullanıcılar tarafından farklı cihazlar tarafından çekildiği için görüntü kaliteleri ve boyutları arasında çok büyük farklar bulunmaktadır. BAFGV1 veri setinde 300x300 boyutunda görüntüler olduğu gibi 4000x4000 boyutundan daha büyük fatura görüntüleri de bulunmaktadır. Ayrıca BAFGV1 veri setindeki fatura görüntülerinin en-boy oranları da birbirinden çok farklıdır. BAFGV1 veri setinde bulunan KartDolum sınıfına ait görüntülerin en boy oranlarını gösteren histogram Şekil 3.18’de gösterilmiştir. Şekil 3.18 incelendiğinde görüntü en boy oranlarının 0.5 ve 0.6 aralığında daha çok toplandığı görülmektedir.



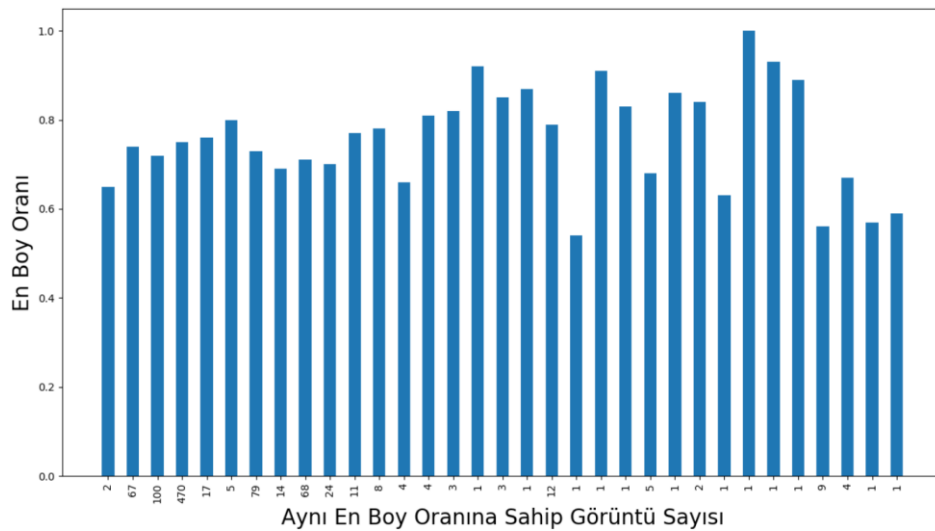
Şekil 3.18. KartDolum sınıfına ait görüntülerin en boy oranlarını gösteren histogram

Otopark sınıfına ait görüntülerin en boy oranlarını gösteren histogram Şekil 3.19’da gösterilmiştir. Histogram incelendiğinde en boy oranlarının daha çok 0.4 ve 0.5 aralığında toplandığı görülmektedir. Zira histogramın x ekseninde verilen sayılar, aynı en boy oranına sahip görüntü sayısı bilgisini içermektedir. Bu sayıların yüksek olması görüntü en boy oranlarının ilgili aralıkta daha çok toplandığını ifade etmektedir.



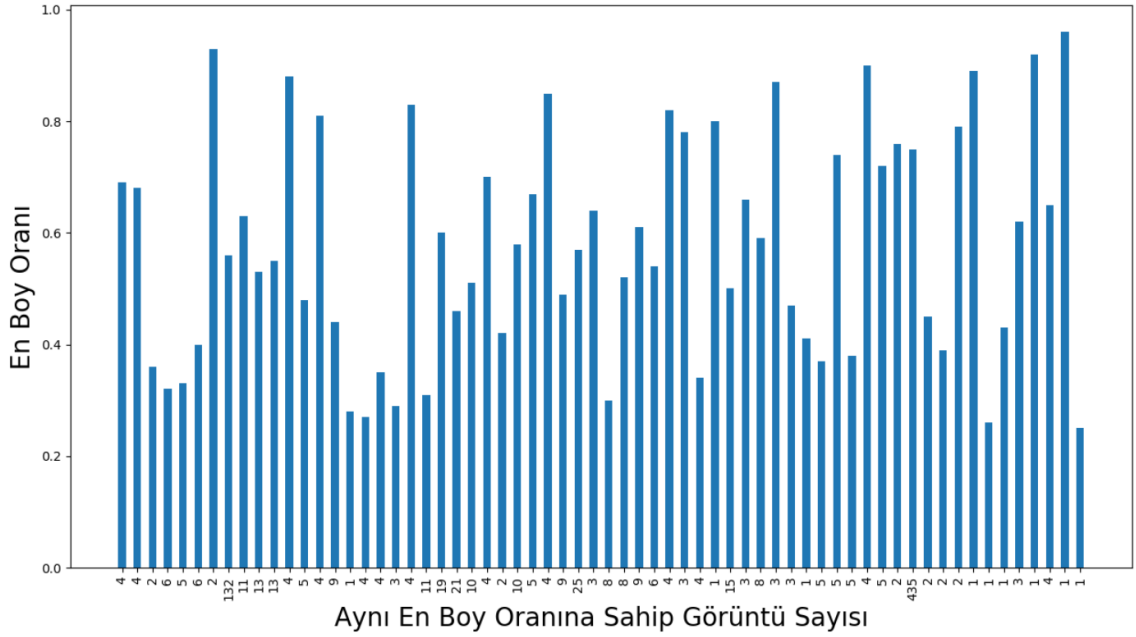
Şekil 3.19. Otopark sınıfına ait görüntülerin en boy oranlarını gösteren histogram

Taksi sınıfına ait görüntülerin en boy oranlarına ait histogram grafiği Şekil 3.20’de gösterilmiştir. X eksenine bakıldığında, 100 ve 470 gibi aynı en boy oranındaki iki fatura kümesinin 0.6 ve 0.8 aralığında toplandığı görülmektedir.



Şekil 3.20. Taksi sınıfına ait görüntülerin en boy oranlarını gösteren histogram

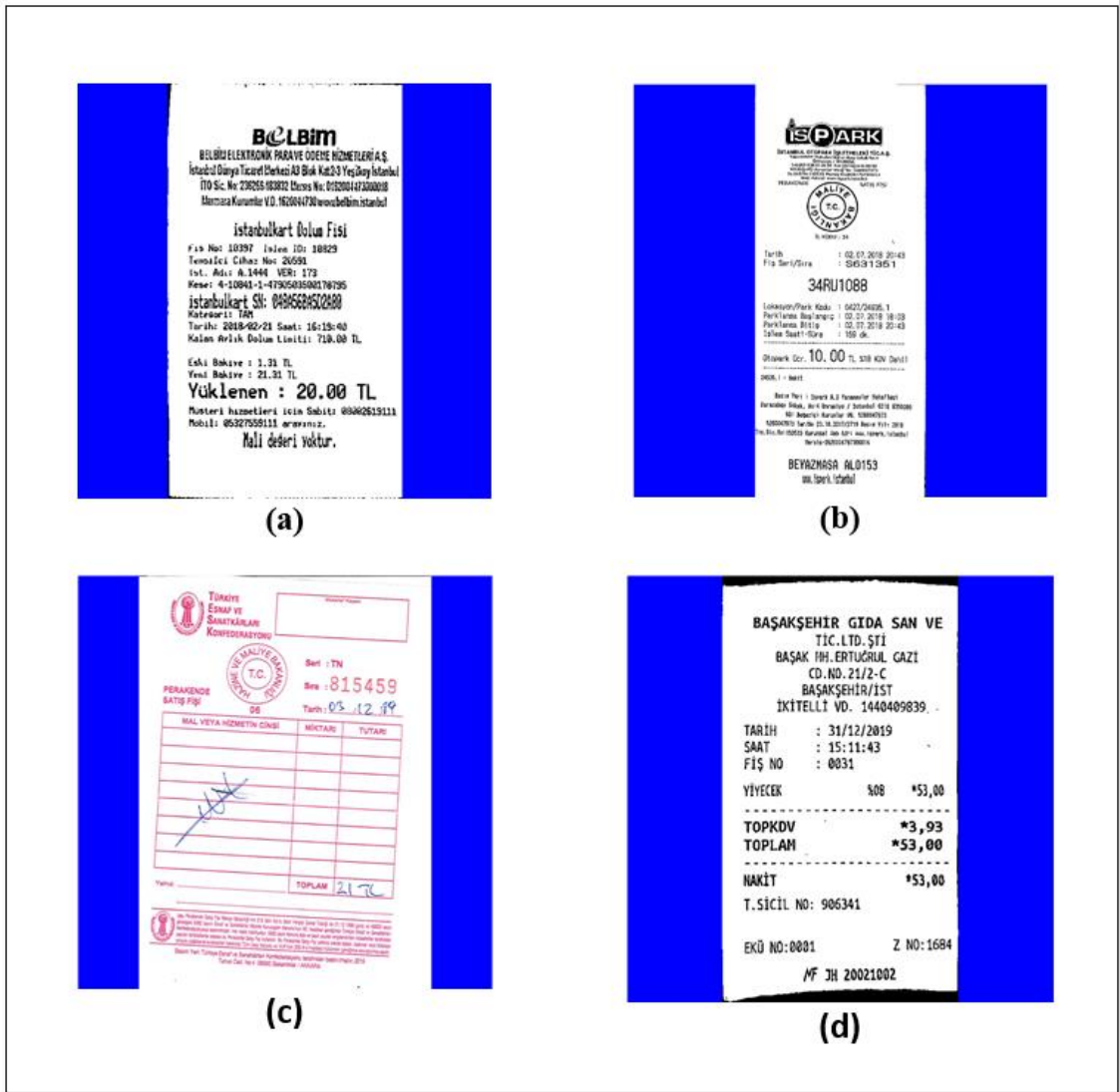
Yemek sınıfına ait görüntülerin en boy oranlarına ait histogram grafiği Şekil 3.21’de gösterilmiştir. Şekil 3.21 incelendiğinde, en boy oranlarının çok düzensiz olduğu görülmektedir. Zira farklı en boy oranına sahip görüntüler çok fazla olduğundan x eksenindeki çubuk sayısı da buna bağlı olarak çok fazla olmuştur. Dolayısıyla Evrişimsel Sınır Ağı mimarisiyle yapılacak sınıflandırma işleminde en fazla ön işlemeye ihtiyaç duyan fatura görüntüleri, Yemek sınıfına ait fatura görüntüleri olmuştur. Çünkü bu şekilde çok düzensiz en boy oranına sahip görüntüler CNN mimarisinde yeniden boyutlandırılırken detay kaybına uğramakta ve sınıflandırma başarısı düşmektedir.



Şekil 3.21. Yemek sınıfına ait görüntülerin en boy oranlarını gösteren histogram

### 3.8.2. BAFGV2 veri seti

BAFGV1 veri setinde bulunan görüntülere sıfır-dolgu işlemi uygulanarak BAFGV2 veri seti oluşturulmuştur. BAFGV1 veri setindeki görüntüler kare haline gelene kadar etraflarına mavi pikseller eklenmiş ve BAFGV2 veri seti elde edilmiştir. Dolayısıyla BAFGV2 veri setindeki her bir görüntü kare şeklinde olup hepsinin en-boy oranı birbirine eşittir. BAFGV2 veri setinde bulunan her bir sınıfa örnek birer fatura görüntüsü Şekil 3.22’de gösterilmiştir.



Şekil 3.22. BAFGV2 veri setinde bulunan örnek fatura görüntüleri. (a) KartDolum fişi (b) Otopark fişi (c) Taksit fişi (d) Yemek fişi

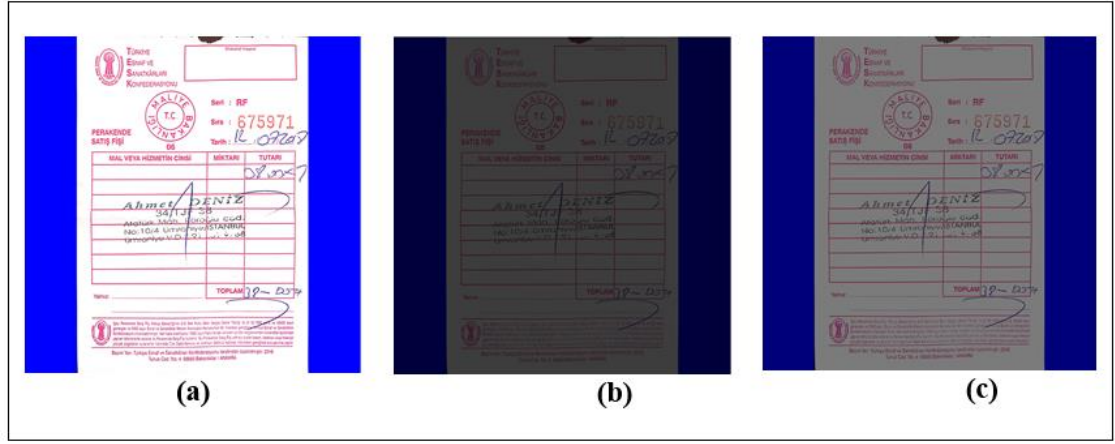
### 3.8.3. BAFGV3 veri seti

Parlaklık tabanlı veri artırma yöntemi kullanılarak BAFGV2 veri setindeki her görüntüden parlaklık değerleri farklı iki görüntü daha elde edilmiştir. Dolayısıyla bu görüntü artırma tekniğiyle 4000 olan veri seti boyutu 12000 görüntüye çıkarılmıştır. Yeni oluşturulan ve 12000 fatura görüntüsünden oluşan veri seti, BAFGV3 olarak isimlendirilmiştir. Çalışmanın devamında da bu veri setinden BAFGV3 şeklinde bahsedilecektir. BAFGV3 veri setinde sınıf başına görüntü sayıları Çizelge 3.3'te verilmiştir.

Çizelge 3.3. BAFGV3 veri setinde sınıf başına fatura görüntü sayıları

| Fatura Sınıfı | Fatura Görüntüsü Sayısı |
|---------------|-------------------------|
| Kart Dolum    | 3000                    |
| Otopark       | 3000                    |
| Taksi         | 3000                    |
| Yemek         | 3000                    |
| <b>Toplam</b> | <b>12000</b>            |

Parlaklık değeri değiştirilerek her görüntüden iki farklı görüntü daha elde edilmesine ait örnek bir işlem Şekil 3.23'te gösterilmiştir.



Şekil 3.23. Parlaklık tabanlı görüntü artırma ile elde edilen örnek görüntüler. (a) Orijinal görüntü (b) Elde edilen ilk görüntü (c) Elde edilen ikinci görüntü

### 3.9. Yapılan Eğitim ve Test Çalışmaları

Geliştirilen fatura sınıflandırma sisteminin eğitim ve test çalışması sırasında önceki bölümde bahsedilen üç farklı veri seti kullanılmıştır. Üç farklı veri seti üzerinde beş farklı CNN ağı kullanarak öznetelik çıkarma işlemi gerçekleştirilmiştir. Dolayısıyla toplam 9 eğitim çalışması yapılmıştır. Yapılan bütün eğitimler sırasında Epoch sayısı olarak 10, eğitim BatchSize olarak 32 ve öğrenme oranı olarak 0.01 değeri kullanılmıştır. Öznetelik çıkarıcı katman olarak kullanılan her beş CNN mimarisinde de giriş görüntü boyutu olarak 224 x 224 ayarlaması yapılmıştır. Dolayısıyla her üç veri setindeki görüntüler eğitim öncesinde 224 x 224 boyutuna gelecek şekilde yeniden boyutlandırılmıştır.

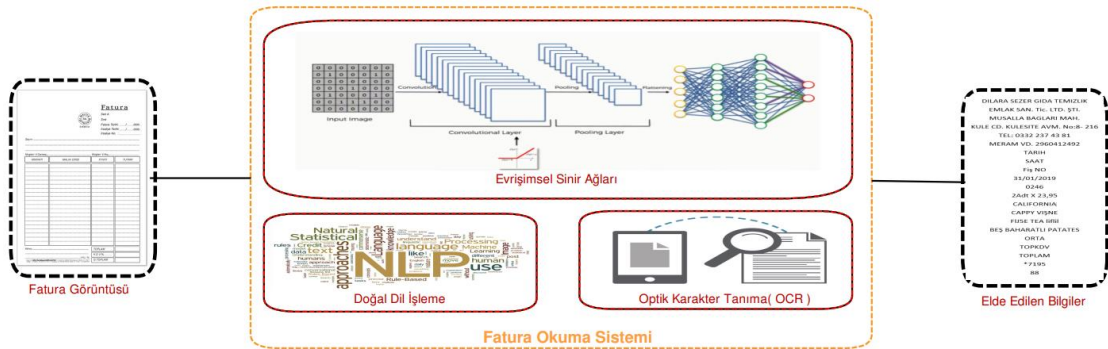
Tüm eğitim setleri ayrıca eğitim öncesinde rastgele bir şekilde eğitim, validasyon ve gerçek dünya örnekleri olarak parçalara ayrılmıştır. Parçalama sırasında eğitim, validasyon ve gerçek dünya örnekleri için sırasıyla %64, %27 ve %9 oranları kullanılmıştır.

Eğitim çalışmaları; 32 GB paylaşımlı belleğe sahip Intel HD Graphics 630 ekran kartına, 64 GB RAM belleğe sahip 4.20Ghz 8 çekirdekli Intel Core i7-7700K bir bilgisayarda gerçekleştirilmiştir. Geliştirilen sistemde öznitelik çıkarıcı olarak LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 CNN mimarileri kullanılmıştır. Her beş mimaride de ara katman aktivasyon fonksiyonu olarak ReLU ve sınıflandırma katmanı fonksiyonu olarak Softmax aktivasyon fonksiyonu kullanılmıştır. Optimizasyon fonksiyonu olarak bütün CNN mimarilerinde SGD yöntemi, kayıp fonksiyonu olarak ise Kategorik Çapraz Düzensizlik (Categorical Cross Entropy) yöntemi kullanılmıştır.

### 3.10. Fatura Görüntülerini Okuma ve Anlamlandırma Uygulaması

#### 3.10.1. Sistem mimarisi

Geliştirilen fatura sınıflandırma sistemi temelde CNN mimarisine, OCR sistemine ve Doğal Dil İşleme alanında yer alan metin benzerlik algoritmalarına dayanmaktadır. Kullanıcıdan alınan fatura görüntüsü, Fatura Okuma Sistemine gönderilmektedir. Bu sistem içerisinde CNN mimarisi kullanılarak fatura şablonuna has öznitelikler çıkarılmaktadır. Geliştirilen sisteme ait genel çalışma grafiği şekil 3.24'te gösterilmiştir.



Şekil 3.24. Geliştirilen sisteme ait genel çalışma grafiği

Projenin sonraki aşamasında sınıflandırma işlemi yapılmaktadır. Proje kapsamında sınıflandırılacak ikiden fazla fatura tipi olduğu için öznitelikleri çıkarılan faturaların şablonlarına göre sınıflandırılması işlemi Softmax aktivasyon fonksiyonu kullanılarak gerçekleştirilmektedir. CNN mimarisinin çıkışında bulunan Softmax sınıflandırma katmanı, çıktı olarak ilgili faturanın hangi şablona sahip olduğuna karar vermektedir. Daha sonra tespit edilen fatura şablonundaki veri konumları göz önünde bulundurularak fatura görüntüsü üzerinde metin arama yapılmaktadır. Metin araması sırasında doğal dil işleme destekli bir OCR uygulaması kullanılarak arama işlemi güçlendirilmektedir.

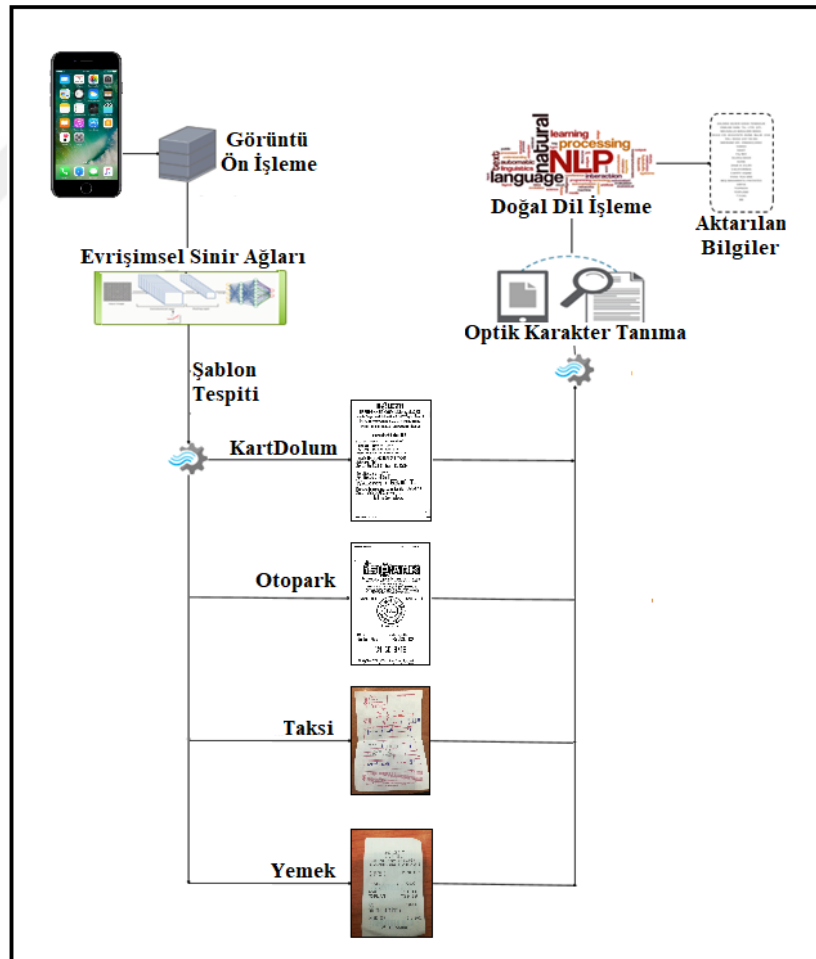
Şablon tespitinden sonraki aşamada, tespit edilen fatura şablonu hakkında genel bilgiler sistemden alınmaktadır. Örnek olarak; Yemek sınıfındaki faturalarda TOPKDV, TOPLAM gibi ifadeler vardır ve TOPKDV ifadesinin altında TOPLAM ifadesi yer almaktadır. TOPKDV ifadesinin üstünde fatura kalemleri, fatura kalemlerinin üstünde ise FİŞ NO, SAAT ve TARİH isimli alanlar bulunmaktadır. Yemek tipindeki bu faturalarda fatura alan isimleri solda yer alıp sağında o alana ait değerler yazmaktadır. Yemek sınıfına ait örnek bir fatura görüntüsü şekil 3.25'te gösterilmiştir.

|                               |     |               |
|-------------------------------|-----|---------------|
| <b>WORLD'S GIDA ÜRN. SAN.</b> |     |               |
| VE DIŞ TIC. LTD. ŞTİ.         |     |               |
| ALTINTEPE MH. BAĞDAT CD.      |     |               |
| NO:91/A MALTEPE/İST           |     |               |
| TEL: 0216 388 02 66           |     |               |
| V.D.KÜÇÜKYALI 8140499229      |     |               |
| TARİH                         | :   | 09/12/2019    |
| SAAT                          | :   | 18:24:36      |
| FİŞ NO                        | :   | 0019          |
| YİYECEK                       | %08 | *58,00        |
| <b>TOPKDV</b>                 |     | <b>*4,30</b>  |
| <b>TOPLAM</b>                 |     | <b>*58,00</b> |
| <b>NAKİT</b>                  |     | <b>*58,00</b> |
| T.SİCİL NO:                   |     |               |
| EKÜ NO:0001                   |     | Z NO:0956     |
| MF JH 20103559                |     |               |

Şekil 3.25. Yemek sınıfına ait örnek bir fatura görüntüsü

Sonraki aşamada görüntü üzerinde fatura alanları tespit edilerek referans noktası olarak belirlenmektedir. Örnek olarak Şekil 4.11'de verilen fatura görüntüsü üzerinde TOPLAM ifadesi aranır. OCR sistemi özellikle görüntü kalitesinin düşük olduğu durumlarda yanlış sonuç verebilmektedir. Burada örnek olarak TOPLAM ifadesini TOPLA şeklinde okuyabileceği için OCR sistemine ek olarak doğal dil işleme alanındaki

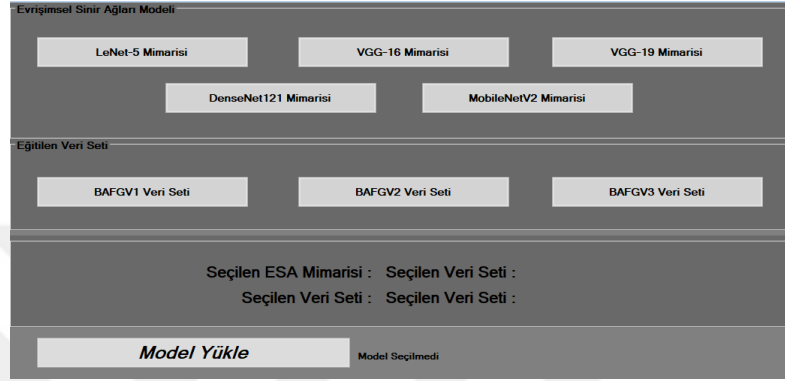
metin uzaklık fonksiyonları da kullanılmaktadır. Türkçe dokümanlar üzerinde daha önceden yapılan testlere istinaden Türkçe metinlerde Levenshtein algoritmasının daha başarılı olduğu gözlemlenmiş ve geliştirilen fatura sınıflandırma sisteminde de Levenshtein algoritması kullanılmıştır. OCR ve Levenshtein algoritması entegre bir şekilde kullanılarak daha sonra görüntü üzerinde TOPLAM ifadesine en çok benzeyen ifadeler okunarak benzerliği en fazla olan ifadenin bulunduğu kısım referans olarak alınır. Referans noktası tespit edildikten sonra, tespit edilmiş olan referans noktası kullanılarak diğer fatura alanlarının konumları tespit edilmektedir. Bütün konumlar tespit edildikten sonra ilgili konumlara OCR uygulanarak görüntü üzerindeki bilgiler dijital ortama aktarılmaktadır. Görüntünün alınması, CNN mimarisi kullanılarak şablon tespiti yapılması, OCR ve doğal dil işleme kullanılarak görüntü üzerindeki istenilen bilgilere ait konumların tespit edilmesi aşamalarını gösteren sistem çalışma grafiği şekil 3.26'da gösterilmiştir.



Şekil 3.26. Geliştirilen sistemin CNN, OCR ve doğal dil işleme aşamaları

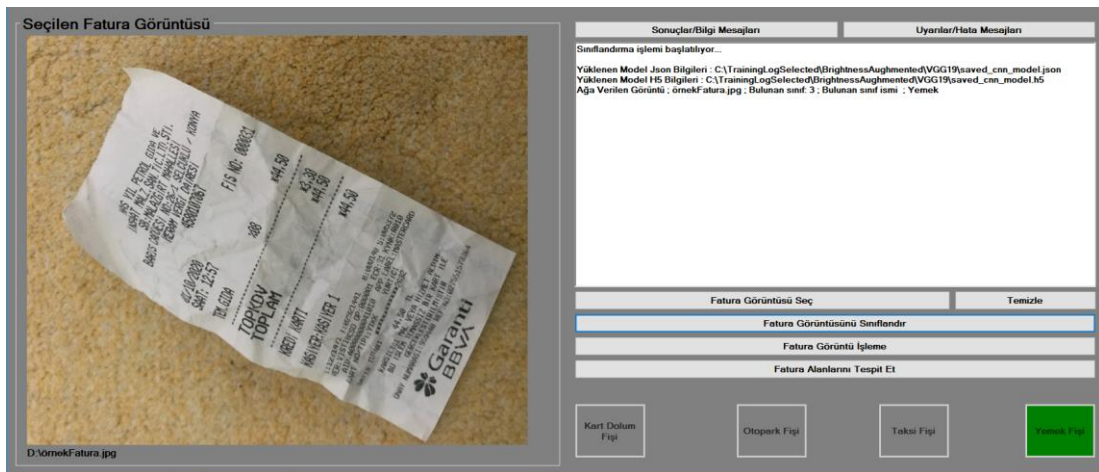
### 3.10.2. Sistem uygulaması

Python 3.6 ile yazılmış olan sistemin kullanılabilmesi için bir .Net uygulaması geliştirilmiştir. C# ile geliştirilmiş olan uygulama, kullanıcılara model seçimi ve veri seti seçimi gibi seçenekler sunmaktadır. Geliştirilen uygulamaya ait giriş ekranı Şekil 3.27’de gösterilmiştir.



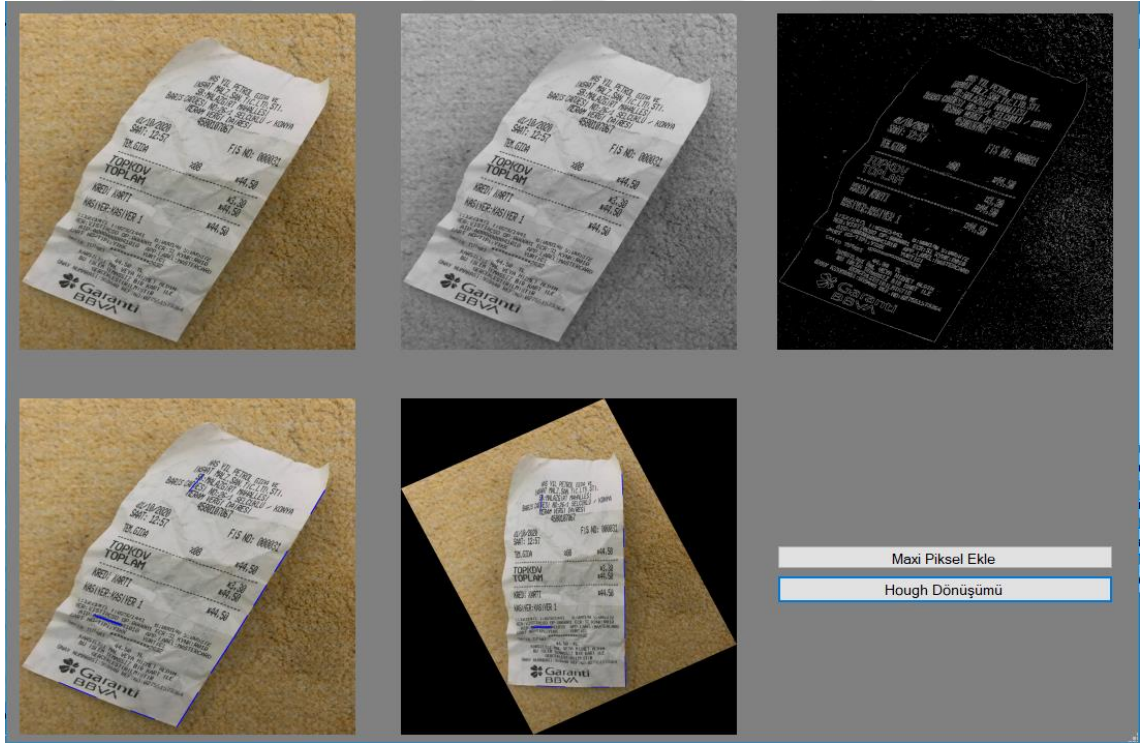
Şekil 3.27. Uygulamanın giriş ekranı

Şekil 3.27’de gösterilen giriş ekranında evrşimsel sinir ağı modeli ve modelin eğitildiği veri seti kullanıcı tarafından seçilerek modelin yüklenmesi sağlanmaktadır. Uygulamanın diğer ekranlarında, giriş ekranında seçilen modele ve veri setine göre işlem yapılmaktadır. Model yükleme işleminden sonraki aşamada kullanıcı tarafından örnek bir fatura görüntüsü seçilerek görüntünün sınıflandırılması sağlanmaktadır. Fatura sınıflandırma ekranı şekil 3.28’de gösterilmiştir.



Şekil 3.28. Fatura sınıflandırma ekranı

Şekil 3.28’de verilen fatura sınıflandırma ekranında, seçilen faturanın hangi sınıfa ait olduğu sağ alt kısımda belirtilmektedir. Ayrıca sınıflandırma ekranında, Python tarafında yazılmış olan uygulamadan dönen uyarı ve bilgi mesajları da kullanıcıya gösterilmektedir. Fatura sınıflandırma sisteminin veri önileme aşamalarında sıfır dolgu ekleme ve Hough dönüşümü teknikleri kullanılmıştır. Kullanıcı tarafından sınıflandırma ekranında seçilen görüntünün hangi aşamalardan geçtiği adım adım izlenebilmektedir. Bu doğrultuda sınıflandırma ekranında fatura görüntü işleme seçeneğiyle görüntü işleme ekranı açılmaktadır. Görüntü işleme ekranında, önceki ekranda seçilmiş fatura görüntüsüne sıfır dolgu ekleme ve Hough dönüşümü uygulama seçenekleri bulunmaktadır. Fatura görüntü işleme ekranında seçilen fatura görüntüsüne Hough dönüşümü uygulanması işlemi şekil 3.29’da gösterilmiştir.



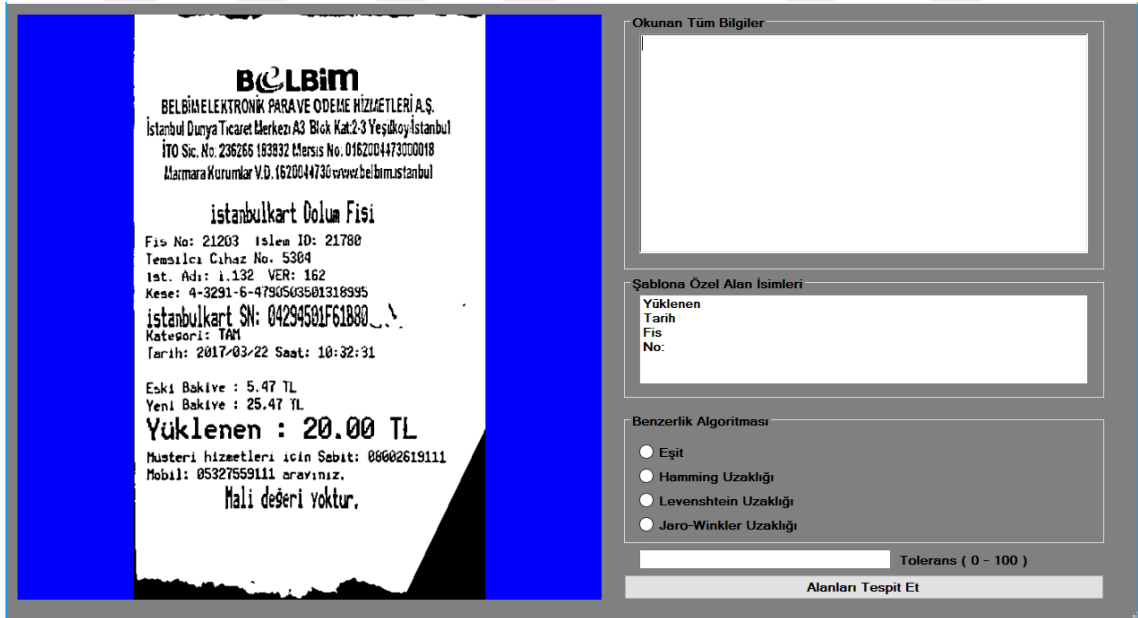
Şekil 3.29. Fatura görüntü işleme ekranında Hough dönüşümü uygulaması

Uygulamanın fatura sınıflandırma ekranında fatura görüntüsüne ayrıca sıfır dolgu uygulaması kullanılarak fatura görüntüsüne mavi pikseller eklenebilmektedir. Fatura görüntüsüne sıfır dolgu uygulaması kullanılarak mavi piksel eklenmesi şekil 3.30’da gösterilmiştir.



Şekil 3.30. Fatura görüntü işleme ekranında mavi piksel ekleme uygulaması

Şekil 3.28’de gösterilen fatura sınıflandırma ekranında ayrıca fatura alanlarını tespit etmek için bir seçenek bulunmaktadır. Bu seçenek kullanılarak fatura alan tespiti ekranı açılmaktadır. Bu ekrana gelen fatura görüntüsünün şablonu, fatura sınıflandırma ekranında tespit edilmiştir. Fatura alan tespiti ekranında şablona özel alan isimleri bölümü bulunmaktadır. Bu bölümde tespit edilmiş olan fatura görüntüsünde aranacak alanlar yer almaktadır. Bu alanlar CNN model eğitim aşamasında sisteme tanıtılmaktadır. Fatura alan tespiti ekranı şekil 3.31’de gösterilmiştir.



Şekil 3.31. Fatura alan tespiti ekranı

Şekil 3.31’de gösterilen fatura alan tespiti ekranının sol tarafında, önceki ekranda seçilmiş olan örnek bir fatura görüntüsü bulunmaktadır. Ekranda yer alan okunan tüm

bilgiler bölümünde fatura görüntüsüne OCR uygulanarak elde edilen bütün metinler yer almaktadır. Ekranda yer alan benzerlik algoritması bölümünde, görüntü üzerinde metin araması yapılırken kullanılacak metin benzerlik fonksiyonu seçimi yapılmaktadır. Bu bölümde eğer “Eşit” seçeneği seçilirse şablona özel alanlara tamamen eşit metinler fatura görüntüsü üzerinde aranır. Ekranda yer alan tolerans alanında ise görüntüde aranan metinlerin benzerlik oranı toleransı belirlenir. Örnek olarak benzerlik algoritması olarak Levenshtein ve tolerans olarak 80 seçilirse; fatura üzerinde şablona özel tüm alanlar OCR uygulanarak aranır ve her birine Levenshtein benzerlik algoritması uygulanır. Metin benzerlik oranı %80 ve üzeri olan kelimeler fatura görüntüsü üzerinde işaretlenir. Benzerlik algoritması olarak Levenshtein ve tolerans değeri olarak 80 seçilmesi durumunda fatura alan tespit ekranının çalışma sonucu şekil 3.32’da gösterilmiştir.

**Okunan Tüm Bilgiler**

**BELBİM**  
 BELBİMELEKTRONİK PARAVE ODENE HİZMETLERİ A.Ş.  
 İstanbul Dünya Ticaret Merkezi A3 Blok Kat:2-3 Yeşilköy-İstanbul  
 İTİ Sic. No: 236266 183832 Mersis No: 0162004473000018  
 Marmara Kurumlar V.D. 1620044730 vvvvvv.belbimistanbul

**İstanbulkart Dolun Fisi**  
 19 No: 21203 İşlem ID: 21780  
 Temsilci Cihaz No: 5304  
 Ist. Adı: i.132 VER: 162  
 Kasa: 4-3291-6-4790563501318995  
 İstanbulkart SN: 04294501F61880  
 Kategori: TAM  
 Tarihi: 2017-03-22 Saat: 10:32:31

Eski Bakiye : 5.47 TL  
 Yeni Bakiye : 25.47 TL  
**Yüklenen : 20.00 TL**  
 Müşteri hizmetleri için Sabit: 06002619111  
 Mobil: 05327559111 ararınız.  
 Mali değeri yoktur.

**Okunan Tüm Bilgiler**  
 BELBİM  
 BELBİMELEKTRONİK PARAVE ODENE HİZMETLERİ A.Ş.  
 İstanbul Dünya Ticaret Merkezi A3 Blok Kat:2-3 Yeşilköy-İstanbul  
 İTİ Sic. No: 236266 183832 Eersis No: 0162004473000018  
 Marmara Kurumlar V.D. 16200414730 vvvvvv.belbimistanbul  
 İstanbulkart Dolun Fisi  
 Fiz Naf 21203 İşlem 10: 21782  
 Temsilci Cihaz No: 5364  
 Ist. Adı: i.132 VER: 162

**Şablona Özel Alan İsimleri**  
 Yüklenen  
 Tarih  
 Fis  
 No:

**Benzerlik Algoritması**  
☐ Eşit  
☐ Hamming Uzaklığı  
☒ Levenshtein Uzaklığı  
☐ Jaro-Winkler Uzaklığı

80 Tolerans ( 0 - 100 )

**Alanları Tespit Et**

Şekil 3.32. Fatura alan tespiti ekranında örnek alan tespiti uygulaması

Şekil 3.30’da fatura alan tespiti ekranında yapılan örnek çalışma sonucunda şablona özel alan isimleri olan “Yüklenen”, “Tarih”, ve “Fis No” alanları fatura görüntüsü üzerinde işaretlenmiştir.

#### 4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu bölümde önerilen yöntemin deneysel çalışmalarına ait sonuçlar sunulmuştur. Çalışmalarda üç veri seti üzerinde beş farklı CNN mimarisi kullanılarak öznelilik çıkarma ve sınıflandırma yapılmıştır. Çalışmalarda veri seti olarak Türkiye’de faaliyet gösteren bir bankanın veri tabanından alınan fatura görüntüleri kullanılmıştır. Kullanılan fatura görüntüleri Türkçe veri içeren faturalardan oluşmaktadır. Dolayısıyla Türkiye’deki faturalar üzerinde çalışacak bir sistem geliştirilmiştir. Yapılan çalışmalarda LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 CNN mimarileri kullanılmıştır. Bütün eğitim çalışmaları 10 Epoch ile gerçekleştirilmiştir. Her eğitim için en son Epoch sırasında elde edilen eğitim ve validasyon doğruluk değerleri ayrı ayrı gösterilmiştir. Yapılan eğitim çalışmalarında 5-Katman çapraz doğrulama tekniği kullanılmıştır. Dolayısıyla 3 veri seti üzerinde beş farklı CNN mimarisi ile çalışma yapıldığından ve 5-Katman çapraz doğrulama tekniği de kullanıldığından toplamda 75 tane eğitim çalışması gerçekleştirilmiştir.

Sistem başarısının değerlendirilmesi için ayrıca Hata Matrisi (Confusion Matrix) metriği de kullanılmıştır. Sadece iki sınıftan oluşan sistemler için hata matrisi hesaplaması çok daha kolaydır. Çünkü hata matrisi içerisinde daha az hücre bulunmaktadır. Kanseri tespiti benzeri çalışmalarda sadece iki sınıf bulunduğu hata matrisinin; TP, TN, FP ve FN gibi parametrelerle hesaplanması gerekmektedir (Radhika ve Varadarajan 2017). TP, TN, FP ve FN sırasıyla; doğru pozitif, doğru negatif, yanlış pozitif ve yanlış negatif ifadelerinin kısaltmasıdır. Ancak sunulan fatura sınıflandırma çalışmasında dört farklı sınıfa ait sınıflandırma yapıldığından iki sınıflı problemlerde 2 x 2 olan hata matrisi boyutu, sunulan fatura sınıflandırma çalışmasında 4 x 4 olmaktadır. Hesaplanan hata matrisi kullanılarak sınıflandırma teknikleri için doğrulanması gereken genel kalite parametreleri; kesinlik, duyarlılık, özgüllük ve f1-skoru olarak sıralanabilir. Bu parametreler sırasıyla Denklem 4.1, Denklem 4.2, Denklem 4.3 ve Denklem 4.4’te gösterildiği şekilde hesaplanabilir (Radhika ve Varadarajan 2017).

$$\textbf{Kesinlik} = \frac{TP}{TP + FP} \quad (4.1)$$

$$\textbf{Duyarlılık} = \frac{TP}{TP + FN} \quad (4.2)$$

$$\text{Özgüllük} = \frac{TN}{TN + FP} \quad (4.3)$$

$$F1 - \text{Skoru} = \frac{2 \times (\text{Kesinlik} \times \text{Duyarlılık})}{\text{Kesinlik} + \text{Duyarlılık}} \quad (4.4)$$

Formüllerde geçen TP değeri, gerçekte Taksı sınıfına ait olan bir örneğin doğru bir şekilde Taksı olarak tahmin edilmesini ifade eder. FP değeri gerçekte Taksı olan bir örneğin yanlış tahmin edilmesini, FN değeri gerçekte Taksı olmayan bir örneğin yanlış tahmin edilmesini, TN değeri ise gerçekte Taksı olmayan bir örneğin doğru bir şekilde tahmin edilmesini ifade eder. Bu doğrultuda Kesinlik metriği, gerçekte Taksı sınıfına ait olan örneklerin kaçının Taksı olarak tahmin edilebildiğini ifade eder. Duyarlılık metriği, TP değerlerine karşı FN değerlerini ölçer. Özellikle hastalık tahmini yapılan çalışmalarda FN değeri de çok önemlidir. Duyarlılık, geliştirilen sistemin Taksı örneklerini doğru bir şekilde tanıma yeteneğini ölçer. Kesinlik ve Duyarlılık metrikleri, özellikle sınıfların eşit bir şekilde dağılmadığı durumlarda kullanışlıdır. F1-Skoru metriği ise sınıf dağılımlarının asimetrik olduğu, FP ve FN maliyetlerinin eşit olmadığı durumlarda Kesinlik ve Duyarlılık değerlerinin ağırlıklı ortalamasını hesaplar (Moghaddari ve ark 2020).

#### 4.1. BAFGV1 Veri Seti Üzerinde Yapılan Deneysel Çalışma Sonuçları

Bu bölümde BAFGV1 veri seti kullanarak yapılan çalışmalara ait sonuçlar sunulmuştur. BAFGV1 veri seti üzerinde öznitelik çıkarıcı olarak LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 mimarileri kullanılarak öznitelik çıkarma işlemleri gerçekleştirilmiştir. Yapılan bütün çalışmalar sırasında 5-Katman çapraz doğrulama tekniği kullanılmış olup BAFGV1 veri seti üzerinde toplamda 25 eğitim çalışması yapılmıştır. Yapılan eğitim çalışmalarına ait başarı oranları ve başarı grafikleri bölümün devamında sunulmuştur. Çizelgelerde belirtilen eğitim ve validasyon başarıları, en son eğitim döngüsü içerisinde elde edilen başarı oranlarını ifade etmektedir.

BAFGV1 veri seti üzerinde yapılan 25 eğitim çalışmasına ait eğitim, validasyon ve test başarı sonuçları sunulduğu gibi eğitim süreleri ve eğitim sonucunda oluşan model boyutları da çizelgelerde sunulmuştur. Ancak hata matrisi ve bazı metrikler sadece en başarılı katmanda alınan sonuçlar için sunulmuştur.

#### 4.1.1. LeNet-5 mimarisi kullanılarak elde edilen sonuçlar

LeNet-5 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV1 veri seti üzerinde öznitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.1’de sunulmuştur.

Çizelge 4.1. LeNet-5 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %94.22          | %91.41              | %93.500       | 12 dakika     | 22 mb        |
| Katman-2        | %82.27          | %86.09              | %71.125       | 12 dakika     | 22 mb        |
| Katman-3        | %93.24          | %93.44              | %90.625       | 12 dakika     | 22 mb        |
| Katman-4        | %87.93          | %93.12              | %85.375       | 12 dakika     | 22 mb        |
| Katman-5        | %90.90          | %86.87              | %78.875       | 12 dakika     | 22 mb        |
| <b>Ortalama</b> | %89.712         | %90.186             | %83.900       | 12 dakika     | 22 mb        |

Çizelge 4.1’de sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 1. katmanda elde edilmiştir. En yüksek başarının elde edildiği 1. katmandaki test çalışmasına ait hata matrisi Çizelge 4.2.’de sunulmuştur.

Çizelge 4.2. LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 198              | 0       | 1     | 5     |
|                         | Otopark   | 0                | 196     | 0     | 0     |
|                         | Taksi     | 0                | 0       | 165   | 6     |
|                         | Yemek     | 2                | 4       | 34    | 189   |

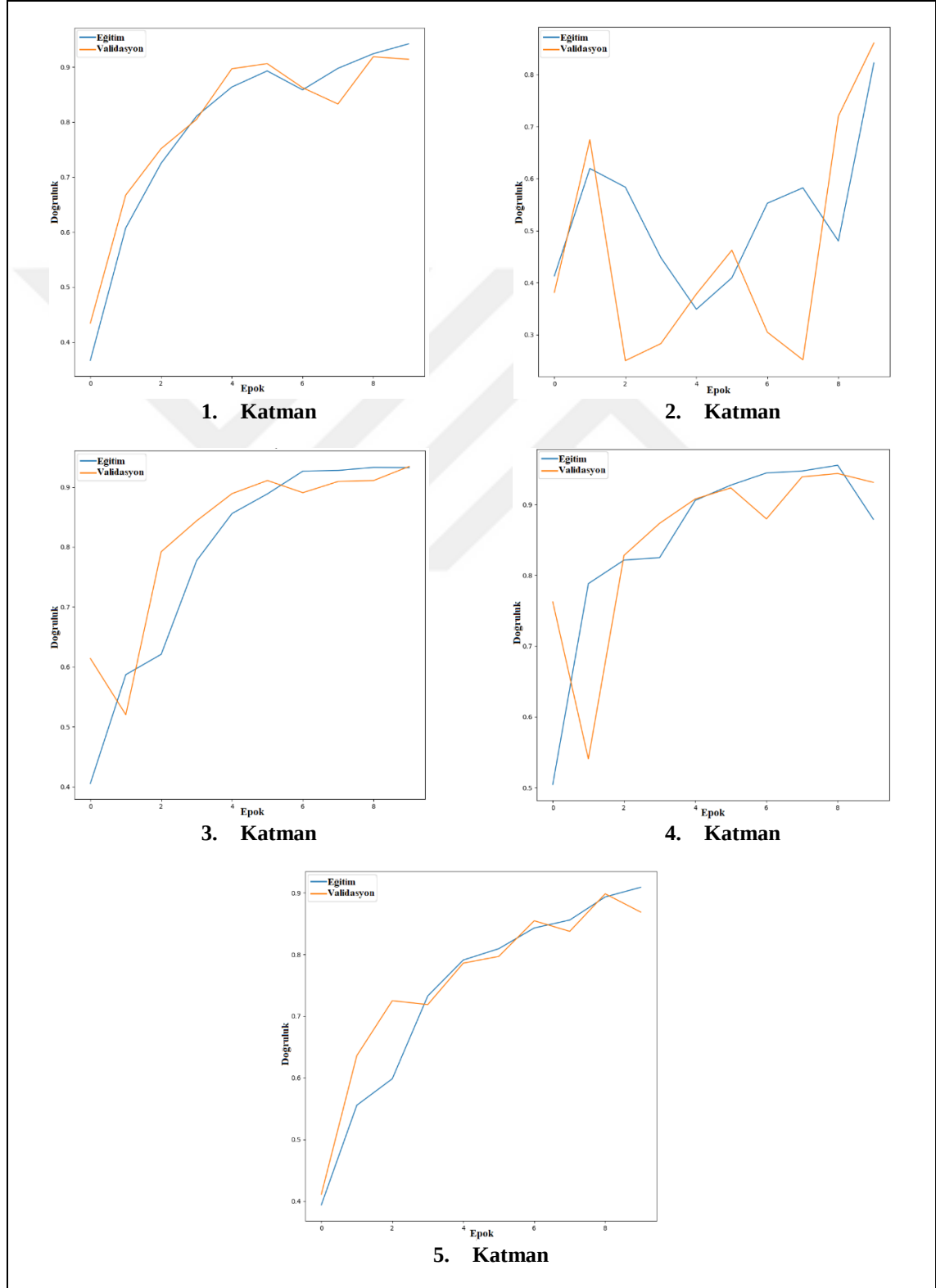
En yüksek başarının elde edildiği 1. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri Çizelge 4.3.’te sunulmuştur.

Çizelge 4.3. LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|         |           | TP  | TN  | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|---------|-----------|-----|-----|----|----|----------|----------|------------|----------|
| LeNet-5 | KartDolum | 198 | 594 | 6  | 2  | 0.99     | 0.9706   | 0.99       | 0.9802   |
|         | Otopark   | 196 | 600 | 0  | 4  | 0.995    | 1        | 0.98       | 0.9899   |
|         | Taksi     | 165 | 594 | 6  | 35 | 0.9487   | 0.9649   | 0.825      | 0.8895   |
|         | Yemek     | 189 | 560 | 40 | 11 | 0.9363   | 0.8253   | 0.945      | 0.8811   |

LeNet-5 ile BAFGV1 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.4.'te sunulmuştur.

**Çizelge 4.4.** LeNet-5 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.1.2. VGG-16 mimarisi kullanılarak elde edilen sonuçlar

VGG-16 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV1 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.5'te sunulmuştur.

Çizelge 4.5. VGG-16 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %97.7           | %87.19              | %88.00        | 155 dakika    | 524 mb       |
| Katman-2        | %63.98          | %73.59              | %68.75        | 155 dakika    | 524 mb       |
| Katman-3        | %33.44          | %25.00              | %45.75        | 155 dakika    | 524 mb       |
| Katman-4        | %53.98          | %71.41              | %49.75        | 155 dakika    | 524 mb       |
| Katman-5        | %26.25          | %25.00              | %25.00        | 155 dakika    | 524 mb       |
| <b>Ortalama</b> | %55.07          | %56.43              | %55.45        | 155 dakika    | 524 mb       |

Çizelge 4.5'te sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 1. katmanda elde edilmiştir. En yüksek başarının elde edildiği 1. katmandaki test çalışmasına ait hata matrisi Çizelge 4.6.'da sunulmuştur.

Çizelge 4.6. VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 151              | 0       | 0     | 0     |
|                         | Otopark   | 1                | 185     | 0     | 2     |
|                         | Taksi     | 46               | 15      | 199   | 29    |
|                         | Yemek     | 2                | 0       | 1     | 169   |

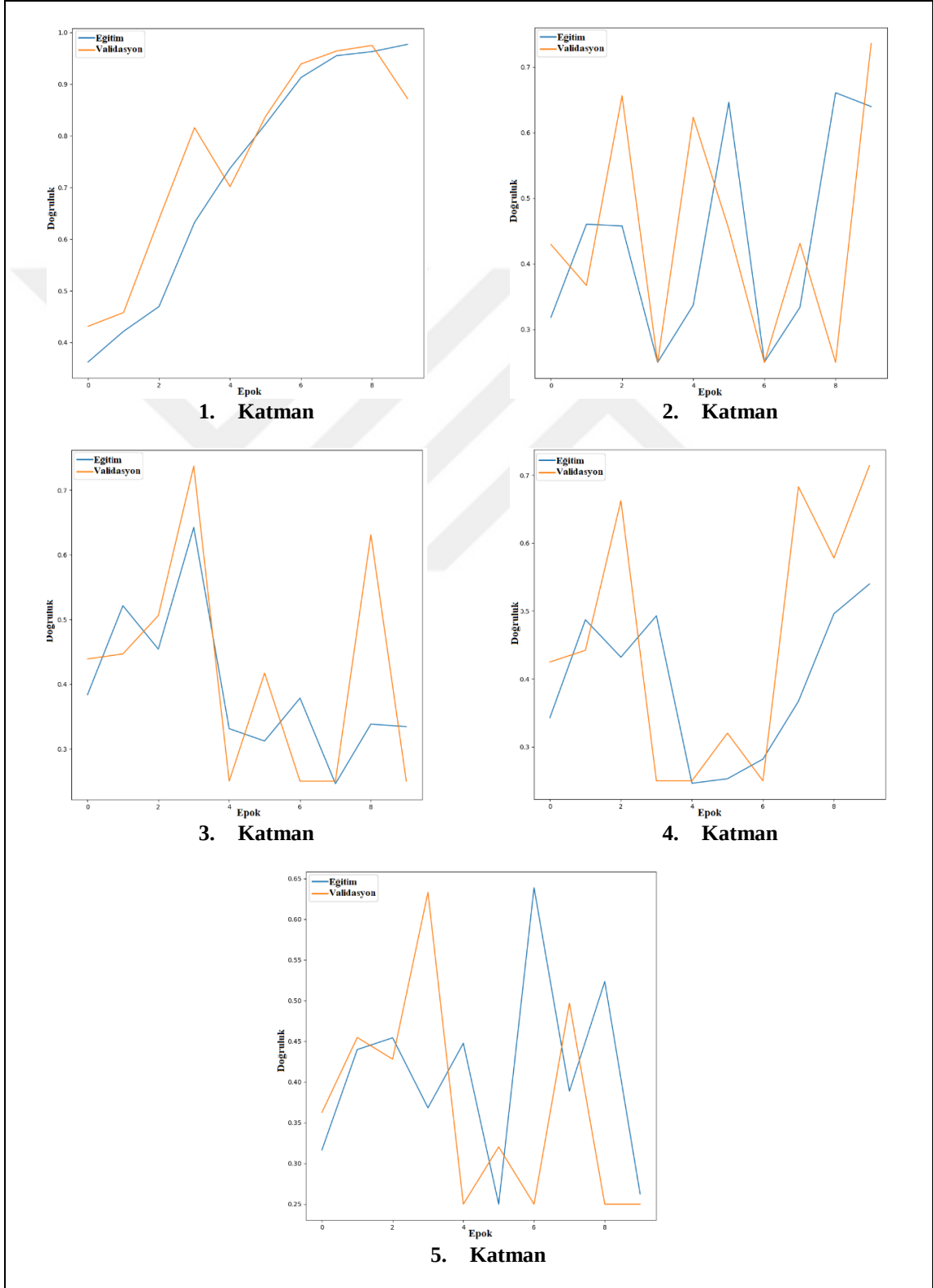
En yüksek başarının elde edildiği 1. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.7.'de sunulmuştur.

Çizelge 4.7. VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|        |           | TP  | TN  | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|--------|-----------|-----|-----|----|----|----------|----------|------------|----------|
| VGG-16 | KartDolum | 151 | 600 | 0  | 49 | 0.9387   | 1.000    | 0.7550     | 0.8604   |
|        | Otopark   | 185 | 597 | 3  | 15 | 0.9775   | 0.984    | 0.9250     | 0.9536   |
|        | Taksi     | 199 | 510 | 90 | 1  | 0.8862   | 0.6886   | 0.9950     | 0.8139   |
|        | Yemek     | 169 | 597 | 3  | 31 | 0.9575   | 0.9826   | 0.8450     | 0.9086   |

VGG-16 mimarisi ile BAFGV1 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.8.'de sunulmuştur.

**Çizelge 4.8.** VGG-16 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.1.3. VGG-19 mimarisi kullanılarak elde edilen sonuçlar

VGG-19 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV1 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.9’da sunulmuştur.

Çizelge 4.9. VGG-19 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %24.61          | %25.00              | %25.00        | 192 dakika    | 545 mb       |
| Katman-2        | %25.16          | %43.44              | %25.00        | 192 dakika    | 545 mb       |
| Katman-3        | %24.61          | %25.00              | %43.37        | 192 dakika    | 545 mb       |
| Katman-4        | %24.57          | %25.00              | %38.25        | 192 dakika    | 545 mb       |
| Katman-5        | %23.83          | %25.00              | %62.87        | 192 dakika    | 545 mb       |
| <b>Ortalama</b> | %24.55          | %28.68              | %38.90        | 192 dakika    | 545 mb       |

Çizelge 4.9’da sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 5. katmanda elde edilmiştir. En yüksek başarının elde edildiği 5. katmandaki test çalışmasına ait hata matrisi Çizelge 4.10.’da sunulmuştur.

Çizelge 4.10. VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 100              | 0       | 6     | 13    |
|                         | Otopark   | 97               | 199     | 17    | 102   |
|                         | Taksi     | 2                | 1       | 177   | 58    |
|                         | Yemek     | 1                | 0       | 0     | 27    |

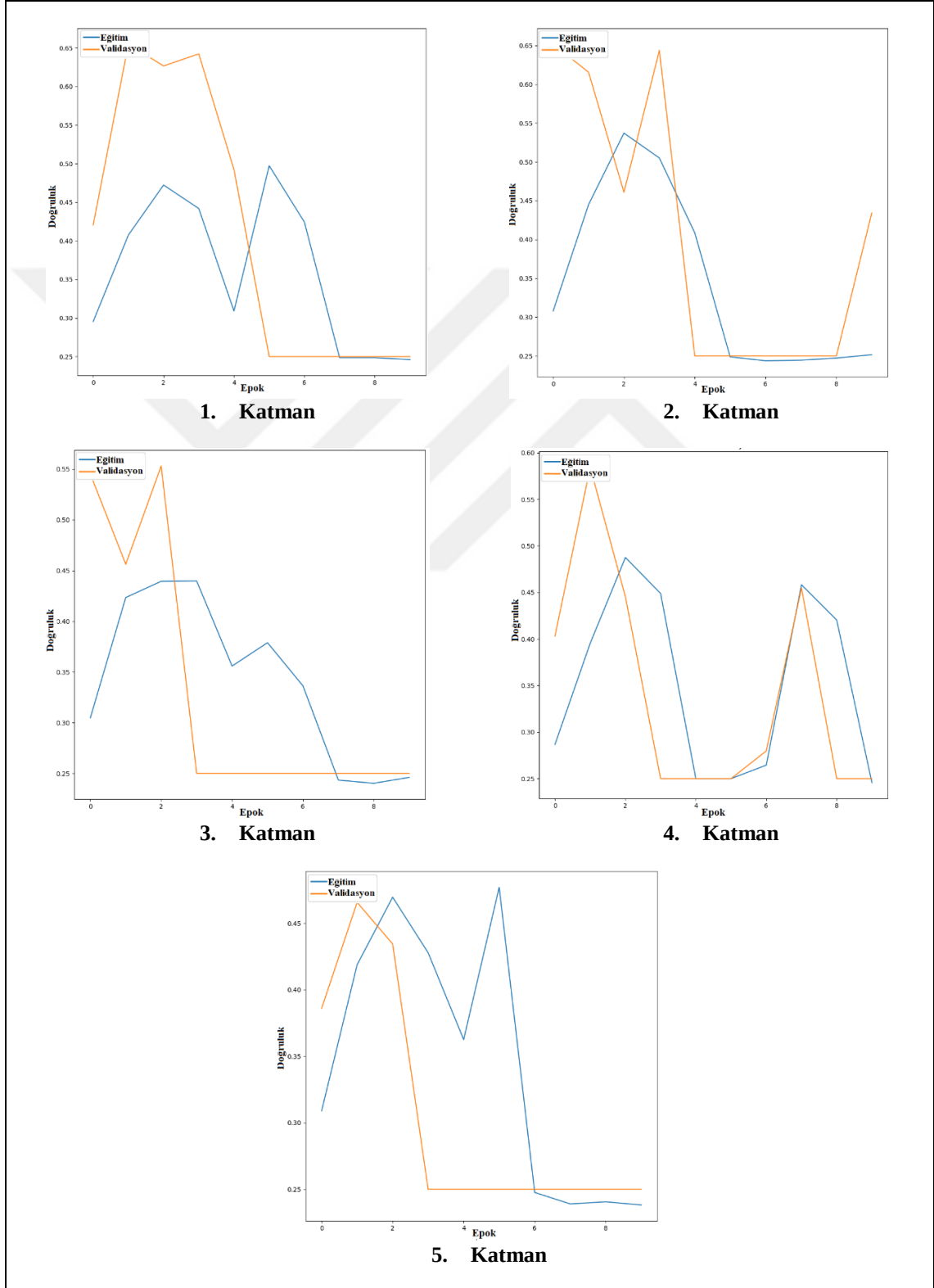
En yüksek başarının elde edildiği 5. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.11.’de sunulmuştur.

Çizelge 4.11. VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|        |           | TP  | TN  | FP  | FN  | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|--------|-----------|-----|-----|-----|-----|----------|----------|------------|----------|
| VGG-19 | KartDolum | 100 | 581 | 19  | 100 | 0.8512   | 0.8403   | 0.5000     | 0.6269   |
|        | Otopark   | 199 | 384 | 216 | 1   | 0.7288   | 0.4795   | 0.9950     | 0.6471   |
|        | Taksi     | 177 | 539 | 61  | 23  | 0.8950   | 0.7437   | 0.8850     | 0.8082   |
|        | Yemek     | 27  | 599 | 1   | 173 | 0.7825   | 0.9643   | 0.1350     | 0.2368   |

VGG-19 mimarisi ile BAFGV1 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.12.'de sunulmuştur.

**Çizelge 4.12.** VGG-19 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.1.4. DenseNet121 mimarisi kullanılarak elde edilen sonuçlar

DenseNet121 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV1 veri seti üzerinde öznitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.13'te sunulmuştur.

Çizelge 4.13. DenseNet121 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.92          | %99.84              | %25.00        | 163 dakika    | 36 mb        |
| Katman-2        | %99.88          | %99.84              | %25.00        | 163 dakika    | 36 mb        |
| Katman-3        | %99.69          | %98.91              | %25.00        | 163 dakika    | 36 mb        |
| Katman-4        | %99.92          | %99.84              | %30.75        | 163 dakika    | 36 mb        |
| Katman-5        | %100.0          | %99.69              | %25.00        | 163 dakika    | 36 mb        |
| <b>Ortalama</b> | %99.88          | %99.62              | %26.15        | 163 dakika    | 36 mb        |

Çizelge 4.13'te sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 4. katmanda elde edilmiştir. En yüksek başarının elde edildiği 4. katmandaki test çalışmasına ait hata matrisi Çizelge 4.14.'te sunulmuştur.

Çizelge 4.14. DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 123              | 34      | 77    | 132   |
|                         | Otopark   | 0                | 0       | 0     | 0     |
|                         | Taksi     | 77               | 166     | 123   | 68    |
|                         | Yemek     | 0                | 0       | 0     | 0     |

En yüksek başarının elde edildiği 4. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.15.'te sunulmuştur.

Çizelge 4.15. DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|             |           | TP  | TN  | FP  | FN  | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|-------------|-----------|-----|-----|-----|-----|----------|----------|------------|----------|
| DenseNet121 | KartDolum | 123 | 357 | 243 | 77  | 0.6000   | 0.3361   | 0.6150     | 0.4347   |
|             | Otopark   | 0   | 600 | 0   | 200 | 0.7500   | 0.0000   | 0.0000     | 0.0000   |
|             | Taksi     | 123 | 289 | 311 | 77  | 0.5150   | 0.2834   | 0.6150     | 0.3880   |
|             | Yemek     | 0   | 600 | 0   | 200 | 0.7500   | 0.0000   | 0.0000     | 0.0000   |



#### 4.1.5. MobileNetV2 mimarisi kullanılarak elde edilen sonuçlar

MobileNetV2 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV1 veri seti üzerinde öznetelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.17’de sunulmuştur.

Çizelge 4.17. MobileNetV2 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.65          | %95.54              | %41.87        | 58 Dakika     | 18 mb        |
| Katman-2        | %99.96          | %95.63              | %47.12        | 58 Dakika     | 18 mb        |
| Katman-3        | %98.96          | %99.22              | %25.25        | 58 Dakika     | 18 mb        |
| Katman-4        | %99.96          | %83.28              | %41.75        | 58 Dakika     | 18 mb        |
| Katman-5        | %98.70          | %88.45              | %53.87        | 58 Dakika     | 18 mb        |
| <b>Ortalama</b> | %99.44          | %92.42              | %41.97        | 58 Dakika     | 18 mb        |

Çizelge 4.17’de sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 5. katmanda elde edilmiştir. En yüksek başarının elde edildiği 5. katmandaki test çalışmasına ait hata matrisi Çizelge 4.18.’de sunulmuştur.

Çizelge 4.18. MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 44               | 1       | 24    | 162   |
|                         | Otopark   | 152              | 198     | 3     | 16    |
|                         | Taksi     | 3                | 1       | 173   | 6     |
|                         | Yemek     | 1                | 0       | 0     | 16    |

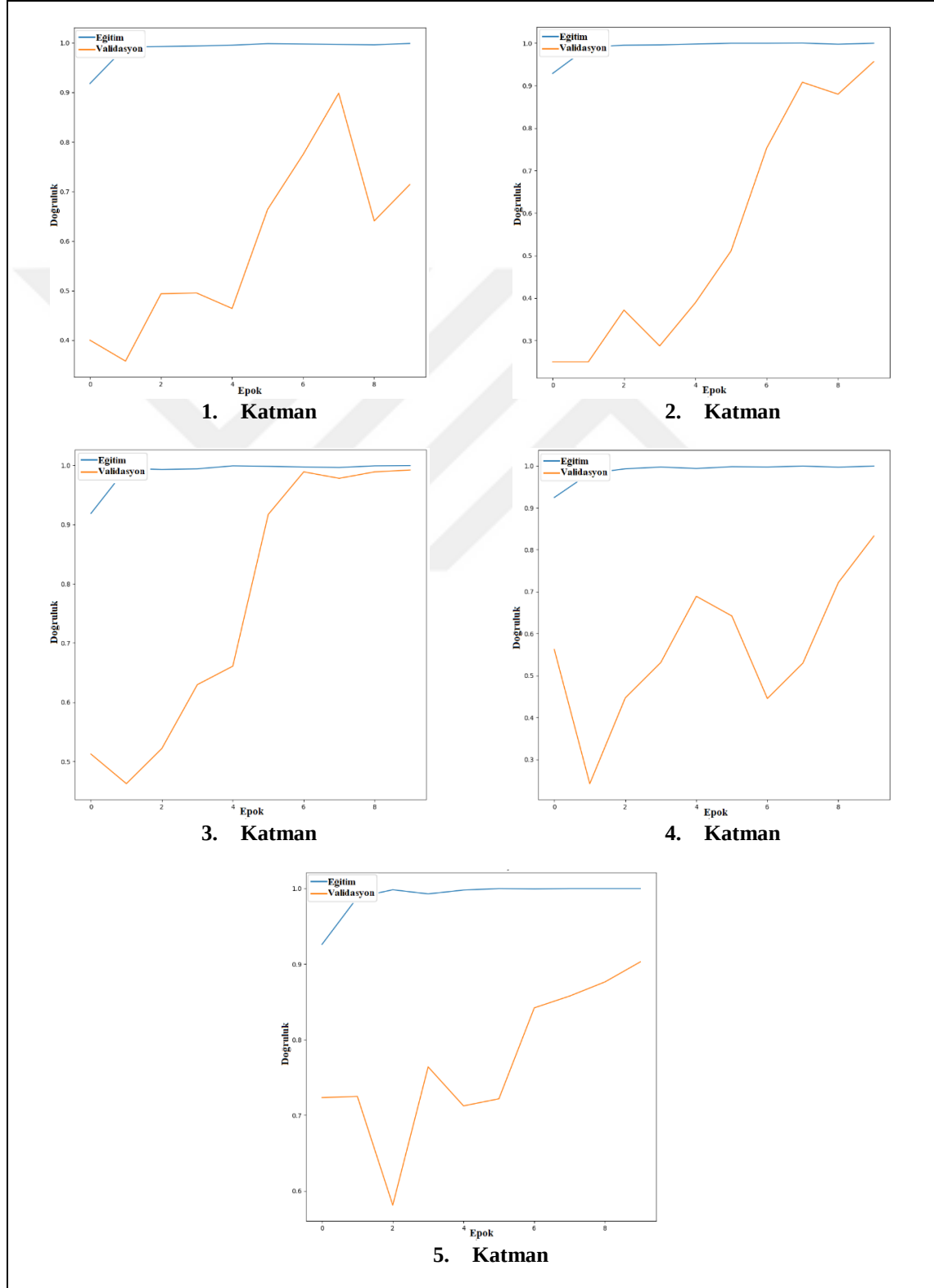
En yüksek başarının elde edildiği 5. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.19.’da sunulmuştur.

Çizelge 4.19. MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|             |           | TP  | TN  | FP  | FN  | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|-------------|-----------|-----|-----|-----|-----|----------|----------|------------|----------|
| MobileNetV2 | KartDolum | 44  | 413 | 187 | 156 | 0.5713   | 0.1905   | 0.2200     | 0.2042   |
|             | Otopark   | 198 | 429 | 171 | 2   | 0.7837   | 0.5366   | 0.9900     | 0.6960   |
|             | Taksi     | 173 | 590 | 10  | 27  | 0.9537   | 0.9454   | 0.8650     | 0.9034   |
|             | Yemek     | 16  | 599 | 1   | 184 | 0.7688   | 0.9412   | 0.0800     | 0.1475   |

MobileNetV2 mimarisi ile BAFGV1 veri seti  zerin yapılan  ğitim  alıřmalarına ait başarı grafikleri  izelge 4.20.'de sunulmuřtur.

** izelge 4.20.** MobileNetV2 mimarisinde 5-Katman  apraz doęrulama ile yapılan  ğitim  alıřmalarında elde edilen doęruluk grafikleri



#### 4.2. BAFGV2 Veri Seti Üzerinde Yapılan Deneysel Çalışma Sonuçları

Bu bölümde BAFGV2 veri seti kullanarak yapılan çalışmalara ait sonuçlar sunulmuştur. BAFGV2 veri seti üzerinde öznitelik çıkarıcı olarak LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 mimarileri kullanılarak öznitelik çıkarma işlemleri gerçekleştirilmiştir. Yapılan bütün çalışmalar sırasında 5-Katman çapraz doğrulama tekniği kullanılmış olup BAFGV2 veri seti üzerinde toplamda 25 eğitim çalışması yapılmıştır. Yapılan eğitim çalışmalarına ait başarı oranları ve başarı grafikleri bölümün devamında sunulmuştur. Çizelgelerde belirtilen eğitim ve validasyon başarıları, en son eğitim döngüsü içerisinde elde edilen başarı oranlarını ifade etmektedir.

BAFGV2 veri seti üzerinde yapılan 25 eğitim çalışmasına ait eğitim, validasyon ve test başarıları çizelgeler halinde sunulmuştur. Ayrıca her bir eğitim katmanının eğitim süresi de sunulmuştur. Eğitim süresi, kaynak tüketimi konusunda önemli bir metrik olduğundan model eğitim süreleri de önem arz etmektedir. Eğitim sonrasında oluşan modellerin, sonraki kullanımlar için depolanması gerekmektedir. Depolanacak model boyutunun çok yüksek olması, modelin taşınabilir olmasını zorlaştırmaktadır. Modellerin özellikle mobil cihazlarda kullanılabilmesi için küçük boyutlarda olması gerekmektedir. Dolayısıyla eğitilen modelin hangi ortamda kullanılacağına göre boyutu önem kazanmaktadır. Bu doğrultuda yapılan eğitim çalışmaları sonucunda oluşan modellerin boyutları da çizelgelerde sunulmuştur. Çizelgelerde yer alan model boyutları incelenerek özellikle mobil ortamda kullanıma uygun olup olmadığı belirlenebilmektedir.

Bölümün devamında ayrıca yapılan eğitim çalışmalarına ait doğruluk grafikleri CNN mimarisi ve katman tabanlı olarak tek tek sunulmuştur. Yapılan çalışmalarda 5-Katman çapraz doğrulama tekniği kullanılmış olup her CNN mimarisi için en başarılı katman seçilmiştir. En başarılı katman seçimi, test ve validasyon başarılarına göre yapılmıştır. En başarılı katman seçildikten sonra ilgili katmandaki test çalışması sırasında yapılan tahminlere ait detay bilgiler çizelgelerle sunulmuştur. Her CNN mimarisi için en başarılı katmandaki test çalışmasına ait kesinlik, duyarlılık, özgüllük ve f1-skoru gibi metrikler hesaplanıp sunulmuştur.

BAFGV2 veri seti, BAFGV1 veri setinde yer alan görüntülere mavi piksel eklenerek en-boy oranı düzenlenmiş görüntülerden oluşan veri setidir. Bu bölümde yapılan çalışmaların amacı, düzensiz en-boy oranı yerine düzenlenmiş en-boy oranına sahip görüntülerle yapılacak eğitim çalışmalarının başarı oranlarını gözlemlemektir. Sonuçlar bölümünde de ayrıca detaylı karşılaştırma sonuçları sunulmuştur.

#### 4.2.1. LeNet-5 mimarisi kullanılarak elde edilen sonuçlar

LeNet-5 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV2 veri seti üzerinde öznitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.21’de sunulmuştur.

Çizelge 4.21. LeNet-5 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %92.51          | %93.59              | %89.226       | 14 dakika     | 22 mb        |
| Katman-2        | %74.79          | %79.11              | %74.651       | 14 dakika     | 22 mb        |
| Katman-3        | %78.24          | %78.12              | %75.285       | 14 dakika     | 22 mb        |
| Katman-4        | %92.67          | %95.89              | %93.662       | 14 dakika     | 22 mb        |
| Katman-5        | %93.24          | %92.60              | %92.522       | 14 dakika     | 22 mb        |
| <b>Ortalama</b> | %86.290         | %87.862             | %85.069       | 14 dakika     | 22 mb        |

Çizelge 4.21’de sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 4. katmanda elde edilmiştir. En yüksek başarının elde edildiği 4. katmandaki test çalışmasına ait hata matrisi Çizelge 4.22.’de sunulmuştur.

Çizelge 4.22. LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 188              | 0       | 0     | 6     |
|                         | Otopark   | 0                | 198     | 0     | 3     |
|                         | Taksi     | 1                | 0       | 188   | 19    |
|                         | Yemek     | 11               | 2       | 8     | 165   |

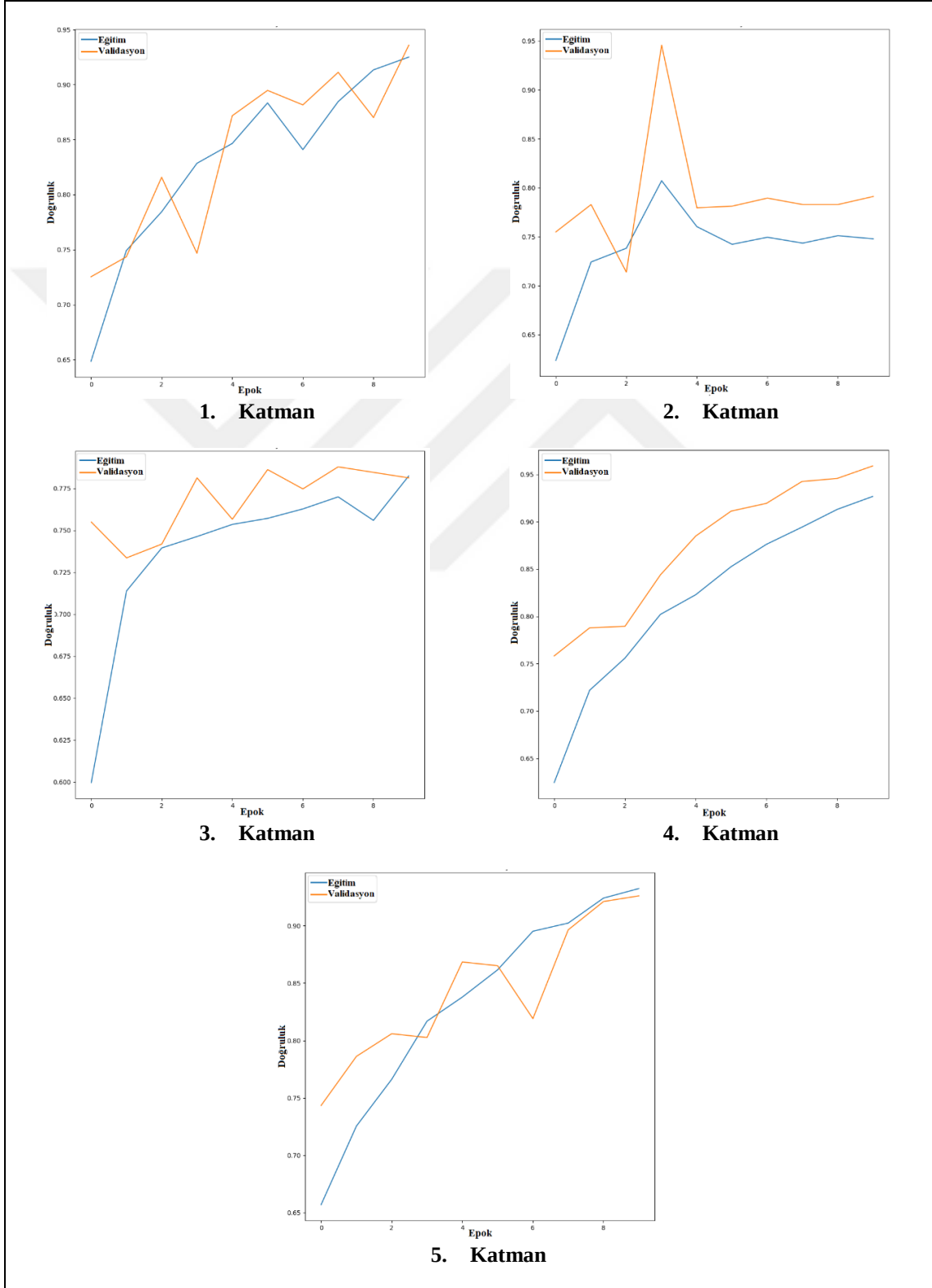
En yüksek başarının elde edildiği 4. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.23.’te sunulmuştur.

Çizelge 4.23. LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|         |           | TP  | TN  | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|---------|-----------|-----|-----|----|----|----------|----------|------------|----------|
|         |           | 188 | 583 | 6  | 12 | 0.9772   | 0.9691   | 0.9400     | 0.9543   |
| LeNet-5 | KartDolum | 188 | 583 | 6  | 12 | 0.9772   | 0.9691   | 0.9400     | 0.9543   |
|         | Otopark   | 198 | 586 | 3  | 2  | 0.9937   | 0.9851   | 0.9900     | 0.9875   |
|         | Taksi     | 188 | 573 | 20 | 8  | 0.9645   | 0.9038   | 0.9592     | 0.9307   |
|         | Yemek     | 165 | 575 | 21 | 28 | 0.9379   | 0.8871   | 0.8549     | 0.8707   |

LeNet-5 ile BAFGV2 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.24.'te sunulmuştur.

**Çizelge 4.24.** LeNet-5 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.2.2. VGG-16 mimarisi kullanılarak elde edilen sonuçlar

VGG-16 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV2 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.25'te sunulmuştur.

Çizelge 4.25. VGG-16 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %96.83          | %97.53              | %97.84        | 153 dakika    | 524 mb       |
| Katman-2        | %98.52          | %98.52              | %98.60        | 153 dakika    | 524 mb       |
| Katman-3        | %98.72          | %97.86              | %96.95        | 153 dakika    | 524 mb       |
| Katman-4        | %97.52          | %99.18              | %98.35        | 153 dakika    | 524 mb       |
| Katman-5        | %98.66          | %99.67              | %98.35        | 153 dakika    | 524 mb       |
| <b>Ortalama</b> | %96.83          | %97.53              | %97.84        | 153 dakika    | 524 mb       |

Çizelge 4.25'te sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 2. katmanda elde edilmiştir. En yüksek başarının elde edildiği 2. katmandaki test çalışmasına ait hata matrisi Çizelge 4.26.'da sunulmuştur.

Çizelge 4.26. VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 200              | 0       | 0     | 1     |
|                         | Otopark   | 0                | 199     | 0     | 1     |
|                         | Taksi     | 0                | 0       | 194   | 6     |
|                         | Yemek     | 0                | 1       | 2     | 185   |

En yüksek başarının elde edildiği 2. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.27.'de sunulmuştur.

Çizelge 4.27. VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|        |           | TP  | TN  | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|--------|-----------|-----|-----|----|----|----------|----------|------------|----------|
|        |           | 200 | 588 | 1  | 0  | 0.9987   | 0.9950   | 1.0000     | 0.9975   |
| VGG-16 | KartDolum | 199 | 588 | 1  | 1  | 0.9975   | 0.9950   | 0.9950     | 0.9950   |
|        | Otopark   | 194 | 587 | 6  | 2  | 0.9899   | 0.9700   | 0.9898     | 0.9798   |
|        | Taksi     | 185 | 593 | 3  | 8  | 0.9861   | 0.9840   | 0.9585     | 0.9711   |
|        | Yemek     |     |     |    |    |          |          |            |          |



#### 4.2.3. VGG-19 mimarisi kullanılarak elde edilen sonuçlar

VGG-19 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV2 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.29’da sunulmuştur.

Çizelge 4.29. VGG-19 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %95.51          | %96.22              | %97.71        | 187 dakika    | 545 mb       |
| Katman-2        | %98.36          | %96.55              | %96.07        | 187 dakika    | 545 mb       |
| Katman-3        | %87.33          | %98.19              | %96.83        | 187 dakika    | 545 mb       |
| Katman-4        | %75.95          | %80.10              | %74.77        | 187 dakika    | 545 mb       |
| Katman-5        | %97.23          | %98.19              | %97.71        | 187 dakika    | 545 mb       |
| <b>Ortalama</b> | <b>%90.87</b>   | <b>%93.85</b>       | <b>%92.62</b> | 187 dakika    | 545 mb       |

Çizelge 4.29’da sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 5. katmanda elde edilmiştir. En yüksek başarının elde edildiği 5. katmandaki test çalışmasına ait hata matrisi Çizelge 4.30.’da sunulmuştur.

Çizelge 4.30. VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 198              | 0       | 0     | 1     |
|                         | Otopark   | 0                | 199     | 0     | 3     |
|                         | Taksi     | 0                | 0       | 190   | 5     |
|                         | Yemek     | 2                | 1       | 6     | 184   |

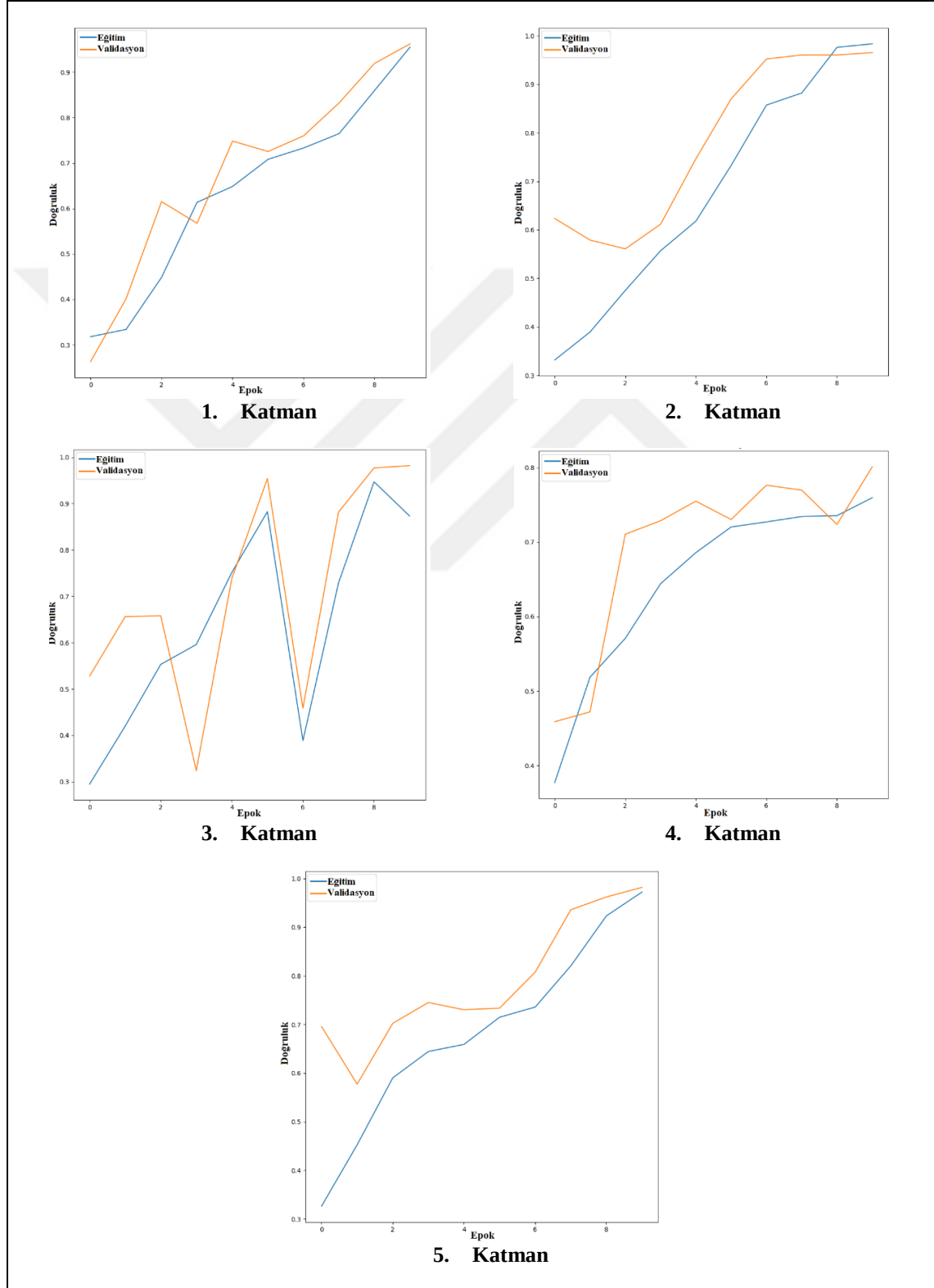
En yüksek başarının elde edildiği 5. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.31.’de sunulmuştur.

Çizelge 4.31. VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|        |           | TP  | TN  | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|--------|-----------|-----|-----|----|----|----------|----------|------------|----------|
| VGG-19 | KartDolum | 198 | 588 | 1  | 2  | 0.9962   | 0.9950   | 0.9900     | 0.9925   |
|        | Otopark   | 199 | 586 | 3  | 1  | 0.9949   | 0.9851   | 0.9950     | 0.9900   |
|        | Taksi     | 190 | 588 | 5  | 6  | 0.9861   | 0.9744   | 0.9694     | 0.9719   |
|        | Yemek     | 184 | 587 | 9  | 9  | 0.9772   | 0.9534   | 0.9534     | 0.9534   |

VGG-19 mimarisi ile BAFGV2 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.32.'de sunulmuştur.

**Çizelge 4.32.** VGG-19 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.2.4. DenseNet121 mimarisi kullanılarak elde edilen sonuçlar

DenseNet121 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV2 veri seti üzerinde öznitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.33'te sunulmuştur.

Çizelge 4.33. DenseNet121 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.88          | %99.51              | %32.95        | 165 dakika    | 36 mb        |
| Katman-2        | %100.0          | %99.67              | %41.69        | 165 dakika    | 36 mb        |
| Katman-3        | %99.96          | %100.0              | %38.78        | 165 dakika    | 36 mb        |
| Katman-4        | %99.96          | %99.51              | %40.17        | 165 dakika    | 36 mb        |
| Katman-5        | %99.96          | %99.51              | %38.14        | 165 dakika    | 36 mb        |
| <b>Ortalama</b> | %99.95          | %99.64              | %38.35        | 165 dakika    | 36 mb        |

Çizelge 4.33'te sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 2. katmanda elde edilmiştir. En yüksek başarının elde edildiği 2. katmandaki test çalışmasına ait hata matrisi Çizelge 4.34.'te sunulmuştur.

Çizelge 4.34. DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 0                | 0       | 0     | 0     |
|                         | Otopark   | 168              | 200     | 66    | 85    |
|                         | Taksi     | 32               | 0       | 129   | 108   |
|                         | Yemek     | 0                | 0       | 1     | 0     |

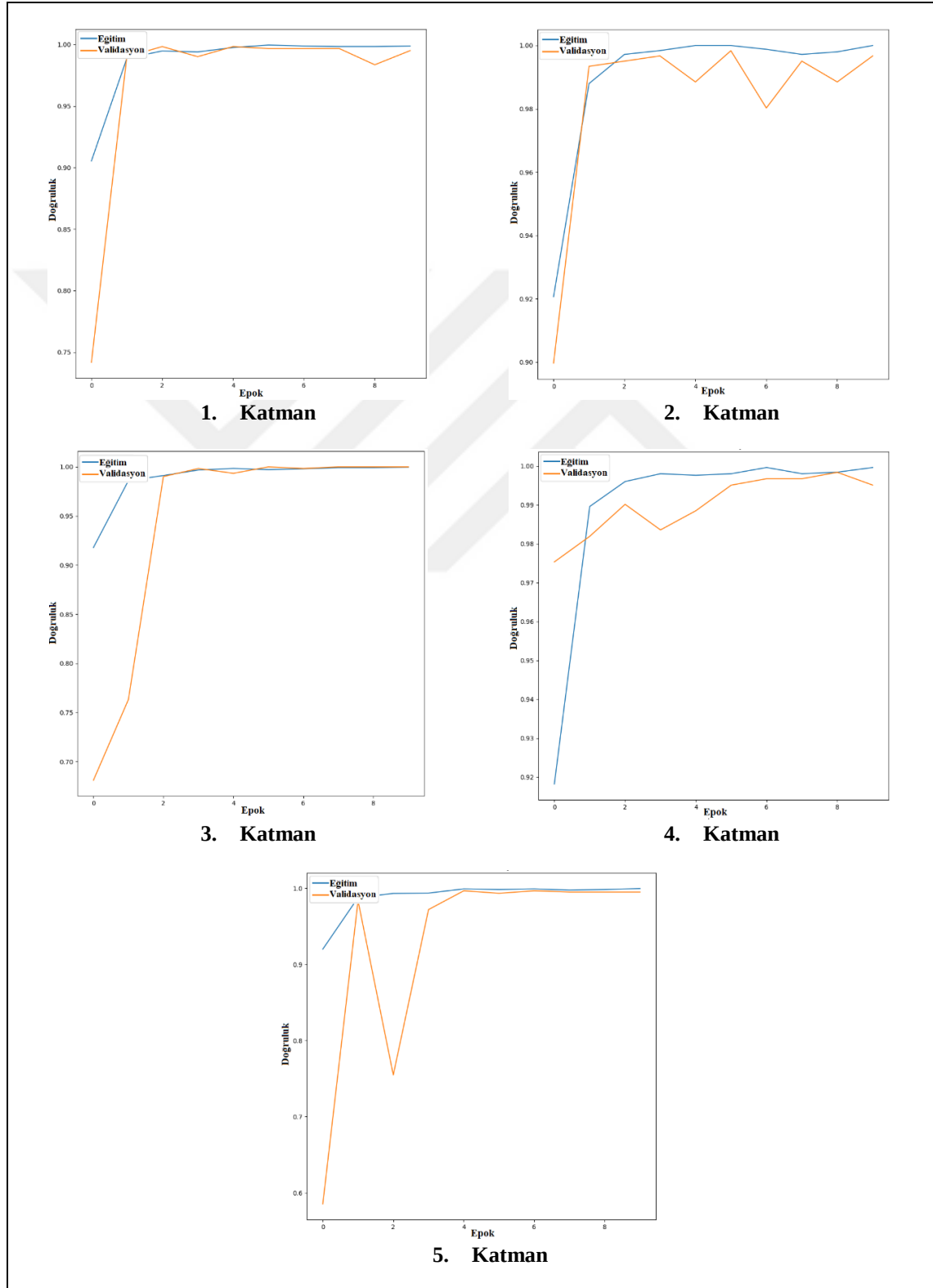
En yüksek başarının elde edildiği 2. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.35.'te sunulmuştur.

Çizelge 4.35. DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|             |           | TP  | TN  | FP  | FN  | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|-------------|-----------|-----|-----|-----|-----|----------|----------|------------|----------|
| DenseNet121 | KartDolum | 0   | 589 | 0   | 200 | 0.7465   | 0.0000   | 0.0000     | 0.0000   |
|             | Otopark   | 200 | 270 | 319 | 0   | 0.5957   | 0.3854   | 1.0000     | 0.5564   |
|             | Taksi     | 129 | 453 | 140 | 67  | 0.7376   | 0.4796   | 0.6582     | 0.5549   |
|             | Yemek     | 0   | 595 | 1   | 193 | 0.7541   | 0.0000   | 0.0000     | 0.0000   |

DenseNet121 mimarisi ile BAFGV2 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.36.'da sunulmuştur.

**Çizelge 4.36.** DenseNet121 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.2.5. MobileNetV2 mimarisi kullanılarak elde edilen sonuçlar

MobileNetV2 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV2 veri seti üzerinde öznitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.37’de sunulmuştur.

Çizelge 4.37. MobileNetV2 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.88          | %98.19              | %62.61        | 51 Dakika     | 18 mb        |
| Katman-2        | %100.0          | %99.34              | %61.21        | 51 Dakika     | 18 mb        |
| Katman-3        | %99.76          | %95.56              | %79.46        | 51 Dakika     | 18 mb        |
| Katman-4        | %99.24          | %97.11              | %67.42        | 51 Dakika     | 18 mb        |
| Katman-5        | %99.84          | %76.15              | %75.79        | 51 Dakika     | 18 mb        |
| <b>Ortalama</b> | %99.88          | %98.19              | %62.61        | 51 Dakika     | 18 mb        |

Çizelge 4.37’de sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 3. katmanda elde edilmiştir. En yüksek başarının elde edildiği 3. katmandaki test çalışmasına ait hata matrisi Çizelge 4.38.’de sunulmuştur.

Çizelge 4.38. MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 189              | 0       | 0     | 2     |
|                         | Otopark   | 0                | 183     | 0     | 0     |
|                         | Taksi     | 0                | 0       | 64    | 0     |
|                         | Yemek     | 11               | 17      | 132   | 191   |

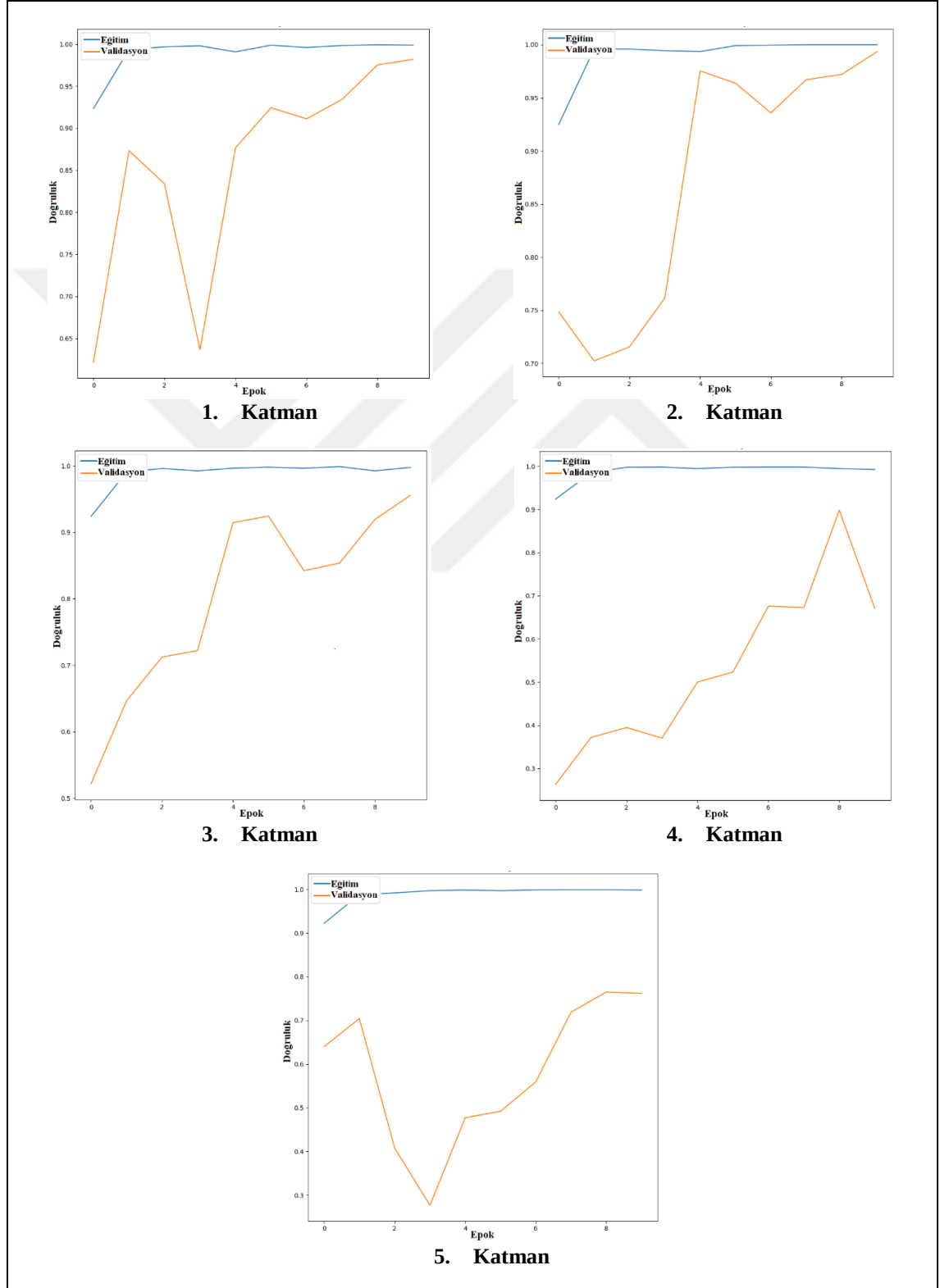
En yüksek başarının elde edildiği 3. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.39.’da sunulmuştur.

Çizelge 4.39. MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|             |           | TP  | TN  | FP  | FN  | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|-------------|-----------|-----|-----|-----|-----|----------|----------|------------|----------|
| MobileNetV2 | KartDolum | 189 | 587 | 2   | 11  | 0.9835   | 0.9895   | 0.9450     | 0.9667   |
|             | Otopark   | 183 | 589 | 0   | 17  | 0.9785   | 1.0000   | 0.9150     | 0.9556   |
|             | Taksi     | 64  | 593 | 0   | 132 | 0.8327   | 1.0000   | 0.3265     | 0.4923   |
|             | Yemek     | 191 | 436 | 160 | 2   | 0.7947   | 0.5442   | 0.9896     | 0.7022   |

MobileNetV2 mimarisi ile BAFGV2 veri seti zerin yapılan eđitim alıřmalarına ait başarı grafikleri izelge 4.40.'da sunulmuřtur.

**izelge 4.40.** MobileNetV2 mimarisinde 5-Katman apraz dođrulama ile yapılan eđitim alıřmalarında elde edilen dođruluk grafikleri



#### 4.3. BAFGV3 Veri Seti Üzerinde Yapılan Deneysel Çalışma Sonuçları

Bu bölümde BAFGV3 veri seti kullanarak yapılan çalışmalara ait sonuçlar sunulmuştur. BAFGV3 veri seti üzerinde öznitelik çıkarıcı olarak LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 mimarileri kullanılarak öznitelik çıkarma işlemleri gerçekleştirilmiştir. Yapılan bütün çalışmalar sırasında 5-Katman çapraz doğrulama tekniği kullanılmış olup BAFGV3 veri seti üzerinde toplamda 25 eğitim çalışması yapılmıştır. Yapılan eğitim çalışmalarına ait başarı oranları ve başarı grafikleri bölümün devamında sunulmuştur. Çizelgelerde belirtilen eğitim ve validasyon başarıları, en son eğitim döngüsü içerisinde elde edilen başarı oranlarını ifade etmektedir.

BAFGV3 veri seti, BAFGV2 veri setinde yer alan görüntülerin parlaklık değeri tabanlı olarak arttırılmasıyla elde edilmiş ve önceki veri setlerinin üç katı veri sayısına sahip bir veri setidir. BAFGV3 veri seti toplamda 12.000 görüntüden oluştuğu için bu bölümde yapılan bütün çalışmalar, önceki çalışmaların yaklaşık üç katı uzun sürmüştür. Yapılan çalışmalara ait eğitim süreleri de dakika cinsinden çizelgelerde sunulmuştur.

Gerçek dünya görüntü örnekleri çok farklılık göstermektedir. Çalışmalarda kullanılan fatura görüntüleri bir banka veri tabanından alınmış olup fatura görüntüleri genellikle kullanıcılar tarafından mobil cihaz kamerası ile çekilmiş görüntülerdir. Bu sebeple bütün görüntüler farklı ışık miktarı içeren ortamlarda çekilmiştir. Farklı ışık miktarında yer alan fatura görüntülerini sınıflandırmak ve bu görüntülere OCR uygulamak zorlu bir süreçtir. Bu sebeple geliştirilen fatura sınıflandırma sisteminde bu sorunun çözümü için BAFGV3 veri seti kullanılarak da bir eğitim yapılmıştır. Bu bölümde yapılan çalışmalarda, farklı ışık miktarına sahip görüntüleri içeren daha büyük bir veri setiyle yapılan eğitimin ağ başarısına olan etkisi araştırılmıştır. Yapılan çalışma sonuçları bölümün devamında çizelgeler halinde sunulmuştur. Ayrıca BAFGV3 veri setiyle yapılan çalışmalar ve önceki çalışmaların karşılaştırılması da sonuçlar bölümünde detaylı bir şekilde sunulmuştur.

#### 4.3.1. LeNet-5 mimarisi kullanılarak elde edilen sonuçlar

LeNet-5 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV3 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.41’de sunulmuştur.

Çizelge 4.41. LeNet-5 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %96.44          | %98.73              | %98.944       | 42 dakika     | 22 mb        |
| Katman-2        | %97.40          | %98.68              | %98.775       | 42 dakika     | 22 mb        |
| Katman-3        | %98.18          | %98.41              | %98.479       | 42 dakika     | 22 mb        |
| Katman-4        | %98.13          | %98.41              | %98.564       | 42 dakika     | 22 mb        |
| Katman-5        | %98.77          | %98.52              | %98.437       | 42 dakika     | 22 mb        |
| <b>Ortalama</b> | %97.78          | %98.550             | %98.640       | 42 dakika     | 22 mb        |

Çizelge 4.41’de sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 1. katmanda elde edilmiştir. En yüksek başarının elde edildiği 1. katmandaki test çalışmasına ait hata matrisi Çizelge 4.42.’de sunulmuştur.

Çizelge 4.42. LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 599              | 0       | 1     | 3     |
|                         | Otopark   | 0                | 599     | 0     | 0     |
|                         | Taksi     | 0                | 0       | 573   | 5     |
|                         | Yemek     | 1                | 1       | 14    | 572   |

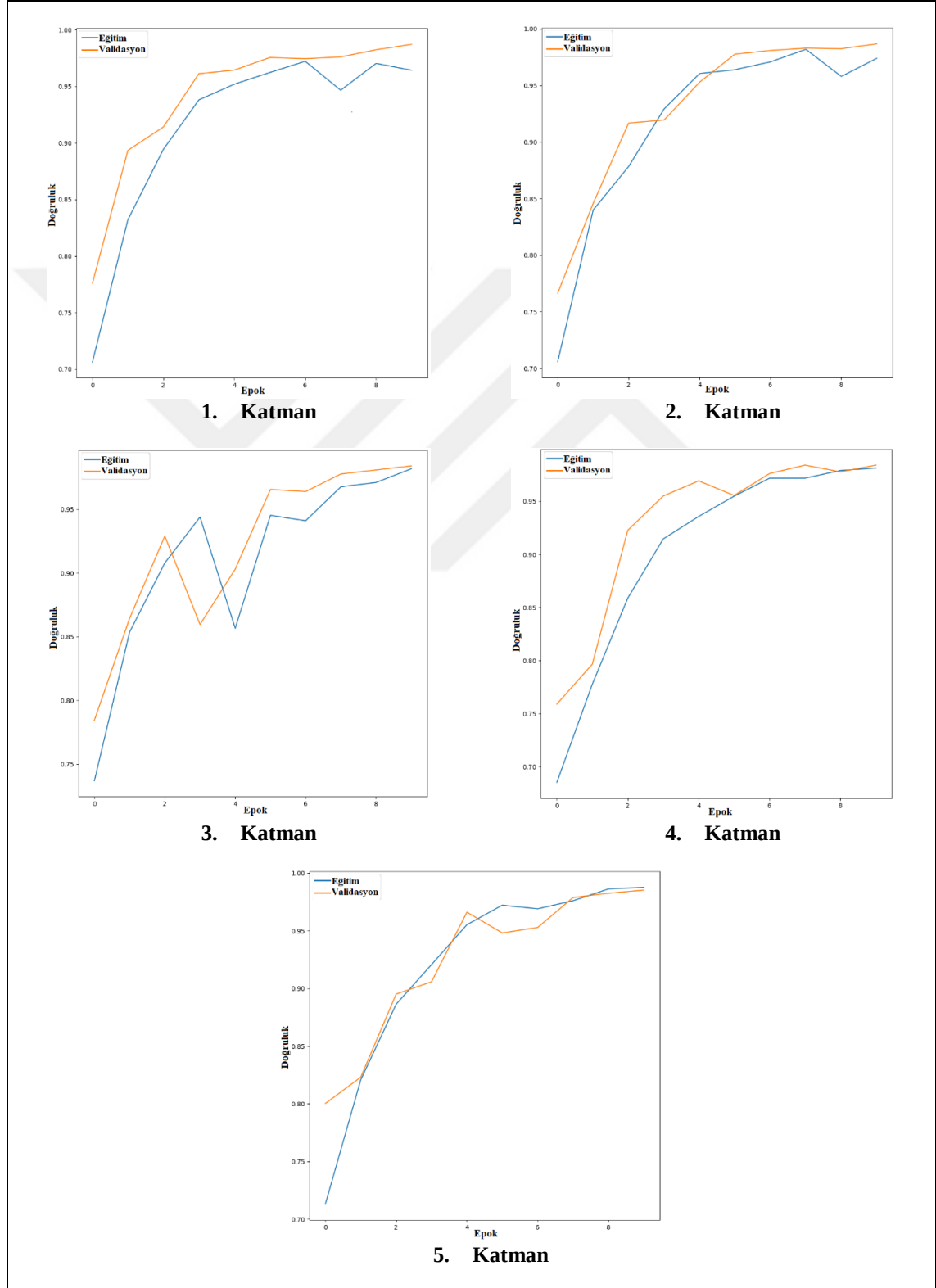
En yüksek başarının elde edildiği 1. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.43.’te sunulmuştur.

Çizelge 4.43. LeNet-5 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|         |           | TP  | TN   | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|---------|-----------|-----|------|----|----|----------|----------|------------|----------|
| LeNet-5 | KartDolum | 599 | 1764 | 4  | 1  | 0.9979   | 0.9934   | 0.9983     | 0.9958   |
|         | Otopark   | 599 | 1768 | 0  | 1  | 0.9996   | 1.0000   | 0.9983     | 0.9991   |
|         | Taksi     | 573 | 1775 | 5  | 15 | 0.9916   | 0.9913   | 0.9745     | 0.9828   |
|         | Yemek     | 572 | 1772 | 16 | 8  | 0.9899   | 0.9728   | 0.9862     | 0.9795   |

LeNet-5 ile BAFGV3 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.44.'te sunulmuştur.

**Çizelge 4.44.** LeNet-5 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.3.2. VGG-16 mimarisi kullanılarak elde edilen sonuçlar

VGG-16 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV3 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.45'te sunulmuştur.

Çizelge 4.45. VGG-16 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.82          | %99.60              | %99.70        | 465 dakika    | 524 mb       |
| Katman-2        | %99.85          | %99.63              | %99.57        | 465 dakika    | 524 mb       |
| Katman-3        | %99.80          | %99.63              | %99.36        | 465 dakika    | 524 mb       |
| Katman-4        | %99.75          | %99.68              | %99.70        | 465 dakika    | 524 mb       |
| Katman-5        | %99.88          | %96.24              | %96.91        | 465 dakika    | 524 mb       |
| <b>Ortalama</b> | %99.82          | %99.60              | %99.70        | 465 dakika    | 524 mb       |

Çizelge 4.45'te sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 4. katmanda elde edilmiştir. En yüksek başarının elde edildiği 4. katmandaki test çalışmasına ait hata matrisi Çizelge 4.46.'da sunulmuştur.

Çizelge 4.46. VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 598              | 0       | 1     | 0     |
|                         | Otopark   | 2                | 600     | 0     | 2     |
|                         | Taksi     | 0                | 0       | 585   | 1     |
|                         | Yemek     | 0                | 0       | 1     | 577   |

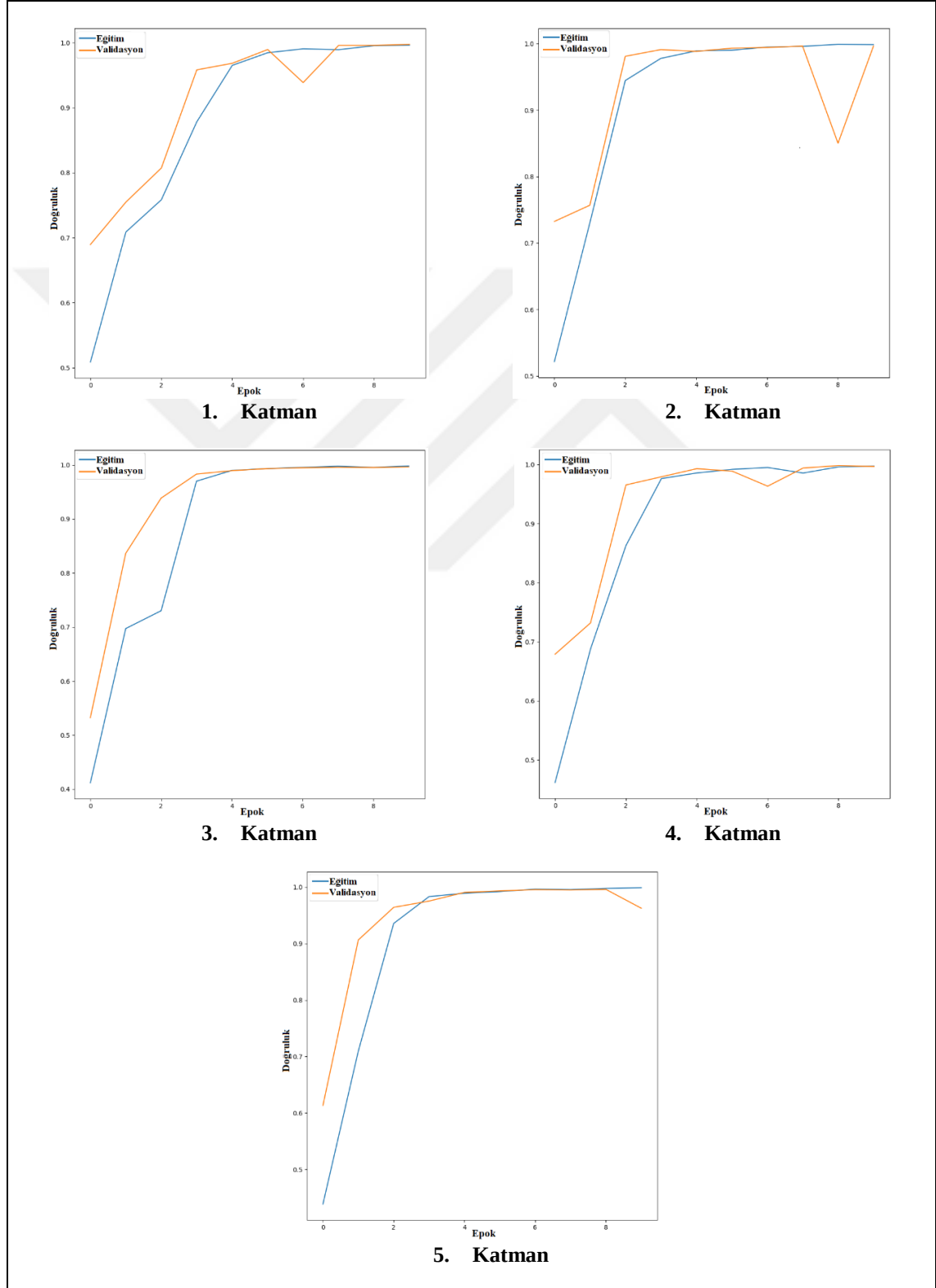
En yüksek başarının elde edildiği 4. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.47.'de sunulmuştur.

Çizelge 4.47. VGG-16 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|        |           | TP  | TN   | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|--------|-----------|-----|------|----|----|----------|----------|------------|----------|
| VGG-16 | KartDolum | 598 | 1766 | 1  | 2  | 0.9987   | 0.9983   | 0.9967     | 0.9975   |
|        | Otopark   | 600 | 1763 | 4  | 0  | 0.9983   | 0.9934   | 1.0000     | 0.9967   |
|        | Taksi     | 585 | 1779 | 1  | 2  | 0.9987   | 0.9983   | 0.9966     | 0.9974   |
|        | Yemek     | 577 | 1786 | 1  | 3  | 0.9983   | 0.9983   | 0.9948     | 0.9965   |

VGG-16 mimarisi ile BAFGV3 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.48.'de sunulmuştur.

**Çizelge 4.48.** VGG-16 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



### 4.3.3. VGG-19 mimarisi kullanılarak elde edilen sonuçlar

VGG-19 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV3 veri seti üzerinde öz nitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.49’da sunulmuştur.

Çizelge 4.49. VGG-19 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.79          | %99.63              | %99.61        | 560 dakika    | 545 mb       |
| Katman-2        | %99.81          | %96.40              | %97.00        | 560 dakika    | 545 mb       |
| Katman-3        | %99.52          | %99.47              | %99.49        | 560 dakika    | 545 mb       |
| Katman-4        | %99.77          | %99.84              | %99.83        | 560 dakika    | 545 mb       |
| Katman-5        | %99.43          | %99.52              | %99.61        | 560 dakika    | 545 mb       |
| <b>Ortalama</b> | %99.66          | %98.97              | %99.11        | 560 dakika    | 545 mb       |

Çizelge 4.49’da sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 4. katmanda elde edilmiştir. En yüksek başarının elde edildiği 4. katmandaki test çalışmasına ait hata matrisi Çizelge 4.50.’de sunulmuştur.

Çizelge 4.50. VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 600              | 0       | 1     | 0     |
|                         | Otopark   | 0                | 600     | 0     | 2     |
|                         | Taksi     | 0                | 0       | 586   | 1     |
|                         | Yemek     | 0                | 0       | 0     | 577   |

En yüksek başarının elde edildiği 4. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.51.’de sunulmuştur.

Çizelge 4.51. VGG-19 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|        |           | TP  | TN   | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|--------|-----------|-----|------|----|----|----------|----------|------------|----------|
| VGG-19 | KartDolum | 600 | 1766 | 1  | 0  | 0.9996   | 0.9983   | 1.0000     | 0.9991   |
|        | Otopark   | 600 | 1765 | 2  | 0  | 0.9992   | 0.9967   | 1.0000     | 0.9983   |
|        | Taksi     | 586 | 1779 | 1  | 1  | 0.9992   | 0.9983   | 0.9983     | 0.9983   |
|        | Yemek     | 577 | 1787 | 0  | 3  | 0.9987   | 1.0000   | 0.9948     | 0.9974   |



#### 4.3.4. DenseNet121 mimarisi kullanılarak elde edilen sonuçlar

DenseNet121 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV3 veri seti üzerinde öznitelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.53'te sunulmuştur.

Çizelge 4.53. DenseNet121 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.99          | %100.0              | %44.35        | 495 dakika    | 36 mb        |
| Katman-2        | %99.97          | %99.84              | %45.88        | 495 dakika    | 36 mb        |
| Katman-3        | %99.99          | %99.74              | %45.96        | 495 dakika    | 36 mb        |
| Katman-4        | %99.99          | %100.0              | %40.38        | 495 dakika    | 36 mb        |
| Katman-5        | %99.89          | %99.84              | %38.86        | 495 dakika    | 36 mb        |
| <b>Ortalama</b> | %99.96          | %99.88              | %43.09        | 495 dakika    | 36 mb        |

Çizelge 4.53'te sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 3. katmanda elde edilmiştir. En yüksek başarının elde edildiği 3. katmandaki test çalışmasına ait hata matrisi Çizelge 4.54.'te sunulmuştur.

Çizelge 4.54. DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 291              | 38      | 66    | 143   |
|                         | Otopark   | 166              | 506     | 230   | 383   |
|                         | Taksi     | 143              | 56      | 291   | 54    |
|                         | Yemek     | 0                | 0       | 0     | 0     |

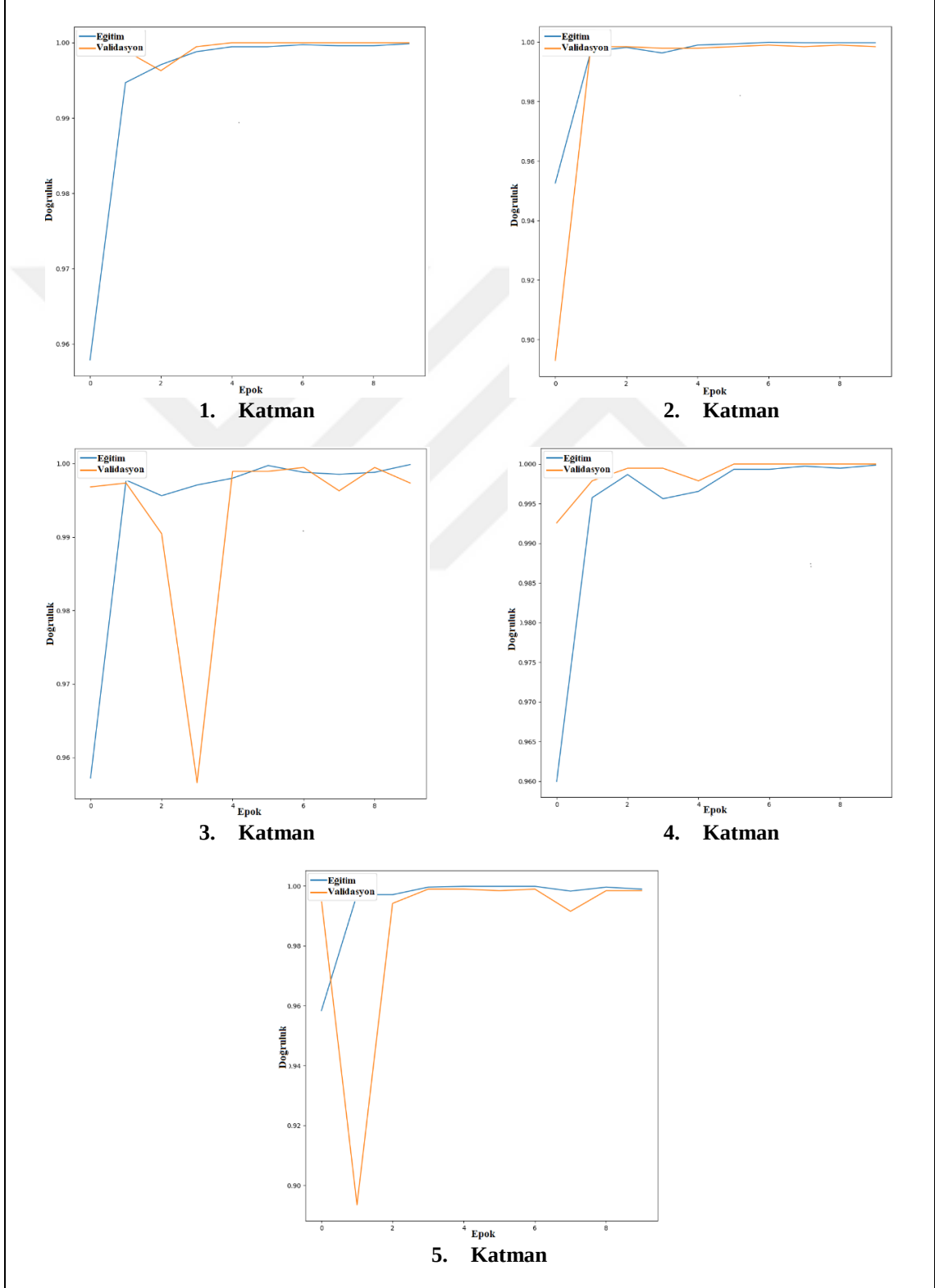
En yüksek başarının elde edildiği 3. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.55.'te sunulmuştur.

Çizelge 4.55. DenseNet121 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|             |           | TP  | TN   | FP  | FN  | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|-------------|-----------|-----|------|-----|-----|----------|----------|------------|----------|
| DenseNet121 | KartDolum | 291 | 1520 | 247 | 309 | 0.7651   | 0.5409   | 0.4850     | 0.5114   |
|             | Otopark   | 506 | 988  | 779 | 94  | 0.6312   | 0.3938   | 0.8433     | 0.5369   |
|             | Taksi     | 291 | 1527 | 253 | 296 | 0.7681   | 0.5349   | 0.4957     | 0.5146   |
|             | Yemek     | 0   | 1787 | 0   | 580 | 0.7550   | 0.0000   | 0.0000     | 0.0000   |

DenseNet121 mimarisi ile BAFGV3 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.56.'da sunulmuştur.

**Çizelge 4.56.** DenseNet121 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



#### 4.3.5. MobileNetV2 mimarisi kullanılarak elde edilen sonuçlar

MobileNetV2 mimarisi 5-Katman çapraz doğrulama tekniği kullanılarak BAFGV3 veri seti üzerinde öznetelik çıkarıcı olarak kullanılmıştır. Toplamda 5 eğitim çalışması yapılmış olup çalışmalara ait sonuçlar Çizelge 4.57’de sunulmuştur.

Çizelge 4.57. MobileNetV2 mimarisi kullanılarak yapılan çalışmalara ait başarı oranları

|                 | Eğitim Doğruluk | Validasyon Doğruluk | Test Doğruluk | Eğitim Süresi | Model Boyutu |
|-----------------|-----------------|---------------------|---------------|---------------|--------------|
| Katman-1        | %99.97          | %100.0              | %80.52        | 189 Dakika    | 18 mb        |
| Katman-2        | %99.99          | %99.79              | %82.63        | 189 Dakika    | 18 mb        |
| Katman-3        | %99.97          | %99.95              | %77.82        | 189 Dakika    | 18 mb        |
| Katman-4        | %99.96          | %100.0              | %95.98        | 189 Dakika    | 18 mb        |
| Katman-5        | %99.96          | %99.79              | %79.29        | 189 Dakika    | 18 mb        |
| <b>Ortalama</b> | %99.97          | %99.90              | %83.25        | 189 Dakika    | 18 mb        |

Çizelge 4.57’de sunulan çalışma sonuçlarına göre, test ve validasyon sonuçları dikkate alındığında en yüksek başarı 4. katmanda elde edilmiştir. En yüksek başarının elde edildiği 4. katmandaki test çalışmasına ait hata matrisi Çizelge 4.58.’de sunulmuştur.

Çizelge 4.58. MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait hata matrisi

|                         |           | Gerçek Etiketler |         |       |       |
|-------------------------|-----------|------------------|---------|-------|-------|
|                         |           | KartDolum        | Otopark | Taksi | Yemek |
| Tahmin Edilen Etiketler | KartDolum | 584              | 13      | 15    | 37    |
|                         | Otopark   | 0                | 578     | 0     | 0     |
|                         | Taksi     | 12               | 0       | 568   | 1     |
|                         | Yemek     | 4                | 9       | 4     | 542   |

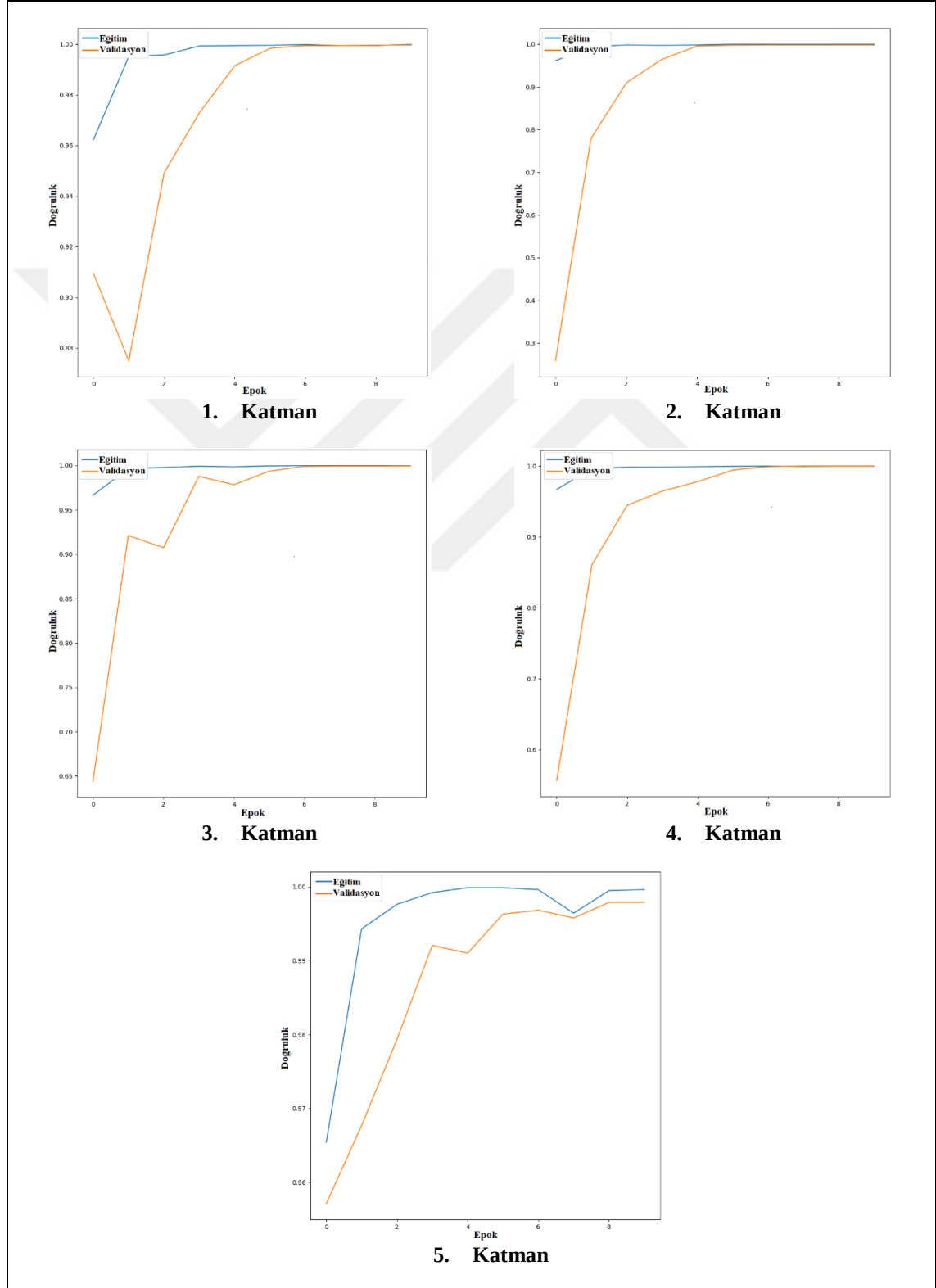
En yüksek başarının elde edildiği 4. katmandaki test çalışması sırasında yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri çizelge 4.59.’da sunulmuştur.

Çizelge 4.59. MobileNetV2 mimarisi kullanılarak en başarılı katmanda yapılan tahminlere ait TP, TN, FP, FN, Doğruluk, Kesinlik, Duyarlılık ve F1-Skoru bilgileri

|             |           | TP  | TN   | FP | FN | Doğruluk | Kesinlik | Duyarlılık | F1-Skoru |
|-------------|-----------|-----|------|----|----|----------|----------|------------|----------|
| MobileNetV2 | KartDolum | 584 | 1702 | 65 | 16 | 0.9658   | 0.8998   | 0.9733     | 0.9351   |
|             | Otopark   | 578 | 1767 | 0  | 22 | 0.9907   | 1        | 0.9633     | 0.9813   |
|             | Taksi     | 568 | 1767 | 13 | 19 | 0.9865   | 0.9776   | 0.9676     | 0.9726   |
|             | Yemek     | 542 | 1770 | 17 | 38 | 0.9768   | 0.9696   | 0.9345     | 0.9517   |

MobileNetV2 mimarisi ile BAFGV3 veri seti üzerin yapılan eğitim çalışmalarına ait başarı grafikleri Çizelge 4.60.'da sunulmuştur.

**Çizelge 4.60.** MobileNetV2 mimarisinde 5-Katman çapraz doğrulama ile yapılan eğitim çalışmalarında elde edilen doğruluk grafikleri



## 5. SONUÇLAR VE ÖNERİLER

Yapılan çalışmada Türkiye’de faaliyet gösteren bir bankanın veri tabanından alınan ve Türkçe veri içeren fatura görüntülerini şablonlarına göre sınıflandıran CNN tabanlı bir sistem geliştirilmiştir. Sistemin eğitim çalışmalarında otopark, kart dolum, yemek ve taksi olmak üzere dört farklı sınıfta fatura görüntüsü kullanılmıştır. Kullanılan fatura sınıflarından ilk üçü olan kart dolum, otopark ve yemek sınıfları için hem fatura şablon sınıflandırma işlemleri hem de görüntü üzerinde fatura alanlarının tespiti işlemleri gerçekleştirilmiştir. Ancak taksi sınıfında yer alan fatura görüntülerinde bilgisayar puntolarına ek olarak el yazısı ifadeler de yer almaktadır. Bu sebeple taksi faturalarında sadece şablon sınıflandırma yapılmış olup daha sonra varsa fatura görüntüsü üzerindeki bilgisayar puntosu ifadelerin yerleri tespit edilmiştir. Bu faturalardaki el yazısı ifadeler ile alakalı bir geliştirme yapılmamıştır.

Yapılan çalışmada LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 mimarileri üç ayrı veri seti kullanılarak eğitilmiştir. Çalışma sırasında 5-Katman çapraz doğrulama kullanıldığından toplamda 75 eğitim çalışması yapılmıştır. Daha sonra veri seti ve CNN mimarisi tabanlı olarak en başarılı katmanlar seçilmiştir. Katman başarılarını sıralarken test ve validasyon başarı oranları dikkate alınmıştır. BAFGV1 veri seti üzerinde yapılan çalışmalarda, en başarılı katmanlarda elde edilmiş başarı oranları veri seti tabanlı olarak Çizelge 5.1’de sunulmuştur.

**Çizelge 5.1.** BAFGV1 veri seti üzerinde yapılan çalışmalara ait başarı oranları

|             | <b>Eğitim Doğruluk</b> | <b>Validasyon Doğruluk</b> | <b>Test Doğruluk</b> |
|-------------|------------------------|----------------------------|----------------------|
| DenseNet121 | %99.92                 | %99.84                     | %30.75               |
| MobileNetV2 | %98.70                 | %88.45                     | %53.87               |
| VGG19       | %23.83                 | %25.00                     | %62.87               |
| VGG-16      | %97.7                  | %87.19                     | %88.00               |
| LeNet-5     | %94.22                 | %91.41                     | %93.50               |

Çizelge 5.1 incelendiğinde BAFGV1 veri seti üzerinde en başarılı test başarısını %93.50 oranıyla LeNet-5 elde etmiştir. DenseNet121 mimarisi ise %30.75 test başarısıyla en başarısız CNN mimarisi olmuştur. En düşük başarıyı elde eden mimariler olan DenseNet121 ve MobileNetV2 mimarilerinin eğitim ve validasyon başarıları çok yüksek olmasına rağmen düşük test başarısı gösterdikleri görülmektedir. Bu durumun sebebi, DenseNet121 ve MobileNetV2 mimarilerinin BAFGV1 veri seti üzerinde ezberlemesidir.

BAFGV1 veri setinden sonra BAFGV2 veri seti üzerinde eğitim çalışmaları yapılmıştır. Aynı şekilde BAFGV2 veri seti üzerinde yapılan çalışmalarda en başarılı katmanlar seçilmiştir. BAFGV2 veri seti üzerinde yapılan çalışmalarda en yüksek başarının elde edildiği katmanlar ve başarı oranları Çizelge 5.2’de sunulmuştur.

**Çizelge 5.2.** BAFGV2 veri seti üzerinde yapılan çalışmalara ait başarı oranları

|             | <b>Eğitim Doğruluk</b> | <b>Validasyon Doğruluk</b> | <b>Test Doğruluk</b> |
|-------------|------------------------|----------------------------|----------------------|
| DenseNet121 | %100.0                 | %99.67                     | %41.69               |
| MobileNetV2 | %99.76                 | %95.56                     | %79.46               |
| LeNet-5     | %92.67                 | %95.89                     | %93.66               |
| VGG19       | %97.23                 | %98.19                     | %97.71               |
| VGG-16      | %98.52                 | %98.52                     | %98.60               |

Çizelge 5.2 incelendiğinde en yüksek test başarısını elde eden CNN mimarisi VGG-16 olurken en düşük başarıyı yine DenseNet121 elde etmiştir. VGG-16 ve VGG-19 birbirine yakın ve yüksek başarılar elde etmiştir. DenseNet121 mimarisin %41.69 gibi başarısına rağmen çok yüksek eğitim ve validasyon başarısı elde etmesi, mimarinin BAFGV2 veri setinde de ezberlediğini göstermektedir. BAFGV2 veri seti, mavi piksel eklenerek görüntülerdeki en-boy oranı düzenlenmiş veri setidir. Düzenlenmiş en-boy oranının etkisi bu çalışma ile araştırılmış olup BAFGV1 veri setinde elde edilen en yüksek test başarısı %93.5 iken BAFGV2 veri setinde en yüksek başarı %98.60 olmuştur. Bu da belirgin olarak iyileşme olduğunu göstermektedir. Bu iyileşmenin sebebi, en-boy oranlarının düzenlenmesidir. Çünkü birbirinden çok farklı görüntü boyutlarıyla yapılan çalışmalarda, CNN mimarisi eğitim öncesinde görüntüleri aynı boyuta gelecek şekilde yeniden boyutlandırdığından görüntüler üzerinde detay kaybı sorunu yaşanmıştır. Detay kaybı olduğunda ise CNN mimarilerinde ezberleme yaşanmakta ve düşük test başarısı sorunu oluşmaktadır.

Son olarak BAFGV3 veri seti üzerinde eğitim çalışmaları yapılmıştır. BAFGV3 veri seti, BAFGV2 veri setindeki görüntülerin parlaklık değerleri rasgele değiştirilerek elde edilmiş ve BAFGV2 veri setinin üç katı büyüklükte olan bir veri setidir. BAFGV3 veri seti üzerinde yapılan eğitim çalışmasında, geliştirilen sistemin farklı ışık miktarı içeren ortamlarda çekilmiş görüntülere karşı daha istikrarlı çalışması amaçlanmıştır. BAFGV3 veri seti üzerinde yapılan çalışmalarda en yüksek başarının elde edildiği katmanlar ve başarı oranları Çizelge 5.3’te sunulmuştur.

**Çizelge 5.3.** BAFGV3 veri seti üzerinde yapılan çalışmalara ait başarı oranları

|             | <b>Eğitim Doğruluk</b> | <b>Validasyon Doğruluk</b> | <b>Test Doğruluk</b> |
|-------------|------------------------|----------------------------|----------------------|
| DenseNet121 | %99.99                 | %99.74                     | %45.96               |
| MobileNetV2 | %99.96                 | %100.0                     | %95.98               |
| LeNet-5     | %96.44                 | %98.73                     | %98.94               |
| VGG-16      | %99.75                 | %99.68                     | %99.70               |
| VGG19       | %99.77                 | %99.84                     | %99.83               |

Çizelge 5.3 incelendiğinde en yüksek başarıyı %99.83 ile VGG-19 elde etmiştir. Ayrıca VGG-19 tarafında elde edilen bu başarı, bütün veri setleri üzerinde yapılan çalışmalarda elde edilen en yüksek başarıdır. DenseNet121 mimarisi ise BAFGV3 veri setinde de %50 başarı oranının altında kalmıştır.

Geliştirilen sistemde beş farklı CNN mimarisi olan LeNet-5, VGG-16, VGG-19, DenseNet121 ve MobileNetV2 mimarileri kullanılmıştır. Kullanılan CNN mimarilerinden VGG-16 ve VGG-19 mimarileri, model boyutu bakımından çok fazla yer kaplayan modeller olup her ikisinde de 500 megabyte boyutunun üzerinde modeller üretilmektedir. Dolayısıyla bu mimariler tarafından üretilen modellerin mobil cihazlarda kullanımı çok zordur. Ancak geliştirilen fatura sınıflandırma sistemi mobil cihazlarda da hizmet verebilmektedir. Dolayısıyla kaynak verimli CNN mimarileri de araştırılarak eğitim çalışmalarına dâhil edilmiştir. Bu doğrultuda model boyutu çok daha düşük olan CNN mimarileri olan LeNet-5, DenseNet121 ve MobileNetV2 mimarileri de çalışmalara dâhil edilmiştir. CNN mimarileri tarafından üretilen model boyutları, kullanılan veri setinden bağımsız olarak belirlenmekte olup ağ mimarisine göre belirlenmektedir. Yapılan çalışmalar sonrasında elde edilen model boyutları, CNN mimarisi tabanlı olarak Çizelge 5.4’ te sunulmuştur.

**Çizelge 5.4.** CNN mimarileri tarafından üretilmiş model boyutları

|             | <b>Model Boyutu ( Megabyte )</b> |
|-------------|----------------------------------|
| MobileNetV2 | 18                               |
| LeNet-5     | 22                               |
| DenseNet121 | 36                               |
| VGG-16      | 524                              |
| VGG19       | 545                              |

Çizelge 5.4’te gösterilen model boyutları incelendiğinde, en düşük boyutlu ağ mimarisinin MobileNetV2 olduğu görülmektedir. MobileNetV2 mimarisi temelde mobil

cihazlarda kullanılmak üzere tasarlanan kaynak verimli bir mimari olduğundan çok küçük boyutta bir model oluşturmuştur.

Model boyutu önemli olduğu gibi model eğitimi için gerekli olan süre de önemlidir. Çünkü süre arttıkça kaynak tüketimi artmakta olup işlemci kullanımı daha fazla olmaktadır. Geliştirilen fatura sınıflandırma sisteminin eğitim çalışmaları sırasında toplam 75 adet eğitim yapılmış olup her bir eğitim için geçen süre de hesaplanmıştır. Veri seti büyüklüğüne doğru orantılı olarak eğitim süresi de arttığından, eğitim için geçen süreler veri seti tabanlı olarak listelenmiştir. Veri seti ve kullanılan CNN mimarisi tabanlı olarak eğitim sırasında geçen ortalama süreler dakika cinsinden Çizelge 5.5'te sunulmuştur. Buna göre eğitim süreleri karşılaştırıldığında en kısa süren CNN mimarisi LeNet-5 olurken en uzun süren mimari VGG-19 olmuştur. BAFGV3 veri seti büyüklüğü diğer iki veri setinin üç katı olduğu için BAFGV3 veri seti üzerindeki eğitimler diğerlerinin ortalama üç katı daha uzun sürmüştür.

Çizelge 5.5. Model eğitimleri sırasında geçen dakika sayıları

|             | <b>BAFGV1</b> | <b>BAFGV2</b> | <b>BAFGV3</b> |
|-------------|---------------|---------------|---------------|
| LeNet-5     | 12            | 14            | 42            |
| MobileNetV2 | 58            | 51            | 189           |
| DenseNet121 | 163           | 165           | 495           |
| VGG-16      | 155           | 153           | 465           |
| VGG19       | 192           | 187           | 560           |

Eğitim çalışmalarının ardından üretilen CNN modelleri kullanılarak fatura görüntülerinin sınıflandırılması gerçekleştirilmiştir. Fatura sınıflandırma işleminden sonraki aşamada fatura alanlarının görüntü üzerinde tespit edilerek sisteme otomatik aktarımı gerçekleştirilmiştir. Bunun için öncelikle fatura şablonlarına özel olarak kelime veri tabanları oluşturularak ilgili şablona özel fatura ifadeleri saklanmıştır. Fatura sınıflandırma işleminden sonra şablonu tespit edilmiş fatura görüntüleri üzerinde, şablona özel veri tabanındaki kelimeler aranmıştır. Kelimelerin fatura görüntüsü üzerinde aranması işleminde OCR sistemine ek olarak metin benzerlik algoritmaları da kullanılmıştır. OCR sisteminin kelimeyi bire bir bulamadığı durumlarda metin benzerlik algoritmaları devreye girerek belirlenen tolerans değerine göre kelimeyi görüntü üzerinde aramaktadır. Fatura sınıflandırma uygulamasında Hamming, Levenshtein ve Jaro-Winkler uzaklığı isimli metin benzerlik metrikleri kullanılmıştır. Metrik seçimi ve benzerliğe göre kelime aranırken kullanılan benzerlik tolerans değerinin belirlenmesi kullanıcıların isteğine bırakılmıştır.

## KAYNAKLAR

- Afzal MZ, Capobianco S, Malik MI, Marinai S, Breuel TM, Dengel A, Liwicki M. Deepdocclassifier: Document classification with deep convolutional neural network. 2015 13th international conference on document analysis and recognition (ICDAR), 1111-5.
- Agarwal N, Balasubramanian VN, Jawahar C, 2018. Improving multiclass classification by deep networks using DAGSVM and Triplet Loss. Pattern Recognition Letters, 112, 184-90.
- Almutairi A, Almashan M. Instance Segmentation of Newspaper Elements Using Mask R-CNN. 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), 1371-5.
- Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, Van Esesn BC, Awwal AAS, Asari VK, 2018. The history began from alexnet: A comprehensive survey on deep learning approaches. arXiv preprint arXiv:1803.01164.
- Aloysius N, Geetha M. A review on deep convolutional neural networks. 2017 International Conference on Communication and Signal Processing (ICCSP), 0588-92.
- Anjali T, Krishnaprasad T, Jayakumar P. A Novel Sentiment Classification of Product Reviews using Levenshtein Distance. 2020 International Conference on Communication and Signal Processing (ICCSP), 0507-11.
- Arora S, Bhatia MS. Handwriting recognition using Deep Learning in Keras. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 142-5.
- Awel MA, Abidi AI, 2019. Review on optical character recognition. International Research Journal of Engineering and Technology, 6, 06.
- Basha SS, Dubey SR, Pulabaigari V, Mukherjee S, 2020. Impact of fully connected layers on performance of convolutional neural networks for image classification. Neurocomputing, 378, 112-9.
- Bautista CM, Dy CA, Mañalac MI, Orbe RA, Cordel M. Convolutional neural network for vehicle detection in low resolution traffic videos. 2016 IEEE Region 10 Symposium (TENSYP), 277-81.
- Breiman L, 2001. Random forests. Machine learning, 45, 1, 5-32.
- Brown JM. Predicting math test scores using k-nearest neighbor. 2017 IEEE Integrated STEM Education Conference (ISEC), 104-6.
- Bui QA, Mollard D, Tabbone S. Selecting automatically pre-processing methods to improve OCR performances. 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), 169-74.

- Carvalho T, De Rezende ER, Alves MT, Balieiro FK, Sovat RB. Exposing computer generated images by eye's region classification via transfer learning of VGG19 CNN. 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), 866-70.
- Casey R, Ferguson D, Mohiuddin K, Walach E, 1992. Intelligent forms processing system. *Machine Vision and Applications*, 5, 3, 143-55.
- Chartrand G, Cheng PM, Vorontsov E, Drozdal M, Turcotte S, Pal CJ, Kadoury S, Tang A, 2017. Deep learning: a primer for radiologists. *Radiographics*, 37, 7, 2113-31.
- Chopra S, Hadsell R, LeCun Y. Learning a similarity metric discriminatively, with application to face verification. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 539-46.
- Chunhavittayatera S, Chitsobhuk O, Tongprasert K. Image registration using Hough transform and phase correlation. 2006 8th International Conference Advanced Communication Technology, 5 pp.-977.
- Cortes C, Vapnik V, 1995. Support-vector networks. *Machine learning*, 20, 3, 273-97.
- Çelikkaya G, Torunoğlu D, Eryiğit G. Named entity recognition on real data: a preliminary investigation for Turkish. 2013 7th International Conference on Application of Information and Communication Technologies, 1-5.
- Deza MM, Deza E, 2014. Distances on strings and permutations. In: *Encyclopedia of Distances*. Eds: Springer, p. 213-25.
- Duda RO, Hart PE, 1972. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15, 1, 11-5.
- Enamandram S, Sandhu E, Do BH, Reicher JJ, Beaulieu CF, 2020. Artificial Intelligence and Machine Learning Applications in Musculoskeletal Imaging. *Advances in Clinical Radiology*.
- Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 580-7.
- Gomaa WH, Fahmy AA, 2013. A survey of text similarity approaches. *International Journal of Computer Applications*, 68, 13, 13-8.
- Gu J, Wang Z, Kuen J, Ma L, Shahroudy A, Shuai B, Liu T, Wang X, Wang G, Cai J, 2018. Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354-77.
- Ha PS, Shakeri M. License Plate Automatic Recognition based on edge detection. 2016 *Artificial Intelligence and Robotics (IRANOPEN)*, 170-4.

- Hamad KA, Kaya M, 2016. A detailed analysis of optical character recognition technology. *International Journal of Applied Mathematics, Electronics and Computers*, 4, 1, 244-9.
- Hamming RW, 1950. Error detecting and error correcting codes. *The Bell system technical journal*, 29, 2, 147-60.
- Hashemi M, 2019. Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation. *Journal of Big Data*, 6, 1, 98.
- He K, Gkioxari G, Dollár P, Girshick R. Mask r-cnn. *Proceedings of the IEEE international conference on computer vision*, 2961-9.
- He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-8.
- Hinton GE, Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR, 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.
- Hochreiter S, Schmidhuber J, 1997. Long short-term memory. *Neural computation*, 9, 8, 1735-80.
- Hu J, Shen L, Sun G. Squeeze-and-excitation networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132-41.
- Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700-8.
- Ide H, Kurita T. Improvement of learning for CNN with ReLU activation by sparse regularization. *2017 International Joint Conference on Neural Networks (IJCNN)*, 2684-91.
- Ioffe S, Szegedy C, 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.
- Jaderberg M, Simonyan K, Zisserman A. Spatial transformer networks. *Advances in neural information processing systems*, 2017-25.
- Jain A, Kulkarni G, Shah V, 2018. Natural language processing. *Int. J. Comput. Sci. Eng*, 6, 1.
- Jarrett K, Kavukcuoglu K, Ranzato MA, LeCun Y. What is the best multi-stage architecture for object recognition? *2009 IEEE 12th international conference on computer vision*, 2146-53.
- Jia X. Fabric defect detection based on open source computer vision library OpenCV. *2010 2nd International Conference on Signal Processing Systems*, V1-342-V1-5.

- Jiao J, Zhao M, Lin J, Liang K, 2020. A comprehensive review on convolutional neural network in machine fault diagnosis. arXiv preprint arXiv:2002.07605.
- Kang L, Kumar J, Ye P, Li Y, Doermann D. Convolutional neural networks for document image classification. 2014 22nd International Conference on Pattern Recognition, 3168-72.
- Khan A, Sohail A, Zahoor U, Qureshi AS, 2020. A survey of the recent architectures of deep convolutional neural networks. Artificial Intelligence Review, 53, 8, 5455-516.
- Khan M, Mufti N. Comparison of various edge detection filters for ANPR. 2016 Sixth International Conference on Innovative Computing Technology (INTECH), 306-9.
- Kingma DP, Ba J, 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 1097-105.
- Le Cun Y, Jackel LD, Boser B, Denker JS, Graf HP, Guyon I, Henderson D, Howard RE, Hubbard W, 1989. Handwritten digit recognition: Applications of neural network chips and automatic learning. IEEE Communications Magazine, 27, 11, 41-6.
- LeCun Y, Bengio Y, Hinton G, 2015. Deep learning. nature, 521, 7553, 436-44.
- LeCun Y, Bottou L, Bengio Y, Haffner P, 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86, 11, 2278-324.
- Ledru Y, Petrenko A, Boroday S. Using string distances for test case prioritisation. 2009 IEEE/ACM International Conference on Automated Software Engineering, 510-4.
- Levenshtein V, 1965. Binary codes capable of correcting spurious insertions and deletion of ones. Problems of information Transmission, 1, 1, 8-17.
- Lin M, Chen Q, Yan S, 2013. Network in network. arXiv preprint arXiv:1312.4400.
- Liu T, Fang S, Zhao Y, Wang P, Zhang J, 2015. Implementation of training convolutional neural networks. arXiv preprint arXiv:1506.01195.
- Manaf K, Pitara S, Subaeki B, Gunawan R. Comparison of Carp Rabin Algorithm and Jaro-Winkler Distance to Determine The Equality of Sunda Languages. 2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA), 77-81.
- Masood S, Ahsan U, Munawwar F, Rizvi DR, Ahmed M, 2020. Scene Recognition from Image Using Convolutional Neural Network. Procedia Computer Science, 167, 1005-12.

- Moghaddari M, Lighvan MZ, Danishvar S, 2020. Diagnose ADHD disorder in children using convolutional neural network based on continuous mental task EEG. *Computer Methods and Programs in Biomedicine*, 105738.
- Nair V, Hinton GE. Rectified linear units improve restricted boltzmann machines. *ICML*.
- Namysl M, Konya I. Efficient, lexicon-free OCR using deep learning. 2019 International Conference on Document Analysis and Recognition (ICDAR), 295-301.
- Nguyen A-D, Choi S, Kim W, Ahn S, Kim J, Lee S. Distribution Padding in Convolutional Neural Networks. 2019 IEEE International Conference on Image Processing (ICIP), 4275-9.
- Nwankpa C, Ijomah W, Gachagan A, Marshall S, 2018. Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*.
- Özkan S. Turkish document semantic categorization using web-based encyclopedia article association. 2012 International Symposium on Innovations in Intelligent Systems and Applications, 1-5.
- Pandey RK, Maiya SR, Ramakrishnan A. A new approach for upscaling document images for improving their quality. 2017 14th IEEE India Council International Conference (INDICON), 1-6.
- Pranav JV, Anand R, Shanthi T, Manju K, Veni S, Nagarjun S, 2020. Detection and identification of COVID-19 based on chest medical image by using convolutional neural networks. *International Journal of Intelligent Networks*, 1, 112-8.
- Prasetya DD, Wibawa AP, Hirashima T, 2018. The performance of text similarity algorithms. *International Journal of Advances in Intelligent Informatics*, 4, 1, 63-9.
- Radhika K, Varadarajan S. Multi class classification of satellite images. 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 659-63.
- Reghunath A, Nair SV, Shah J. Deep learning based Customized Model for Features Extraction. 2019 International Conference on Communication and Electronics Systems (ICCES), 1406-11.
- Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 91-9.
- Reshamwala A, Mishra D, Pawar P, 2013. Review on natural language processing. *IRACST Engineering Science and Technology: An International Journal (ESTIJ)*, 3, 1, 113-6.

- Rezende VC, Costa M, Santos A, de Oliveira RC. Image Processing with Convolutional Neural Networks for Classification of Plant Diseases. 2019 8th Brazilian Conference on Intelligent Systems (BRACIS), 705-10.
- Rosenblatt F, 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65, 6, 386.
- Rumelhart DE, Hinton GE, Williams RJ, (1985). Learning internal representations by error propagation, California Univ San Diego La Jolla Inst for Cognitive Science.
- Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. Mobilenetv2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510-20.
- Saxen F, Werner P, Handrich S, Othman E, Dinges L, Al-Hamadi A. Face attribute detection with mobilenetv2 and nasnet-mobile. 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), 176-80.
- Shaha M, Pawar M. Transfer learning for image classification. 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 656-60.
- Shehu GS, Ashir AM, Eleyan A. Character recognition using correlation & hamming distance. 2015 23rd Signal Processing and Communications Applications Conference (SIU), 755-8.
- Shorten C, Khoshgoftaar TM, 2019. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 1, 60.
- Sidhwa H, Kulshrestha S, Malhotra S, Virmani S. Text extraction from bills and invoices. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 564-8.
- Simonyan K, Zisserman A, 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Smith R. An overview of the Tesseract OCR engine. Ninth international conference on document analysis and recognition (ICDAR 2007), 629-33.
- Sokar G, Hemayed EE, Rehan M. A generic ocr using deep siamese convolution neural networks. 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 1238-44.
- Sze V, Chen Y-H, Yang T-J, Emer JS, 2017. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE*, 105, 12, 2295-329.
- Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1-9.

- Tammina S, 2019. Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. International Journal of Scientific and Research Publications, 9, 10, 143-50.
- Tang YY, Suen CY, De Yan C, Cheriet M, 1995. Financial document processing based on staff line and description language. IEEE transactions on systems, man, and cybernetics, 25, 5, 738-54.
- Tarawneh AS, Hassanat AB, Chetverikov D, Lendak I, Verma C. Invoice classification using deep features and machine learning techniques. 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), 855-9.
- Taylor SL, Fritzson R, Pastor JA, 1992. Extraction of data from preprinted forms. Machine Vision and Applications, 5, 3, 211-22.
- Tensmeyer C, Martinez T. Analysis of convolutional neural networks for document image classification. 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), 388-93.
- Toğaçar M, Cömert Z, Ergen B, 2021. Intelligent skin cancer detection applying autoencoder, MobileNetV2 and spiking neural networks. Chaos, Solitons & Fractals, 144, 110714.
- Wang G, Gong J. Facial expression recognition based on improved LeNet-5 CNN. 2019 Chinese Control And Decision Conference (CCDC), 5655-60.
- Wang J, Dong Y, 2020. Measurement of Text Similarity: A Survey. Information, 11, 9, 421.
- Wang M, Deng W, 2018. Deep Face Recognition: A Survey. arXiv:1804.06655. Erişim Adresi, <https://ui.adsabs.harvard.edu/abs/2018arXiv180406655W>.
- Wei TC, Sheikh U, Ab Rahman AA-H. Improved optical character recognition with deep neural network. 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA), 245-9.
- Wijnhoven RG, de With P. Fast training of object detection using stochastic gradient descent. 2010 20th International Conference on Pattern Recognition, 424-7.
- Winkler WE, 1990. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage.
- Xia Y, Cai M, Ni C, Wang C, Shiping E, Li H. A Switch State Recognition Method based on Improved VGG19 network. 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 1658-62.
- Yamashita R, Nishio M, Do RKG, Togashi K, 2018. Convolutional neural networks: an overview and application in radiology. Insights into imaging, 9, 4, 611-29.

- Yapıcı MM, Tekerek A, Topaloğlu N. Performance comparison of convolutional neural network models on GPU. 2019 IEEE 13th International Conference on Application of Information and Communication Technologies (AICT), 1-4.
- Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. European conference on computer vision, 818-33.
- Zhang X, Dahu W, 2019. Application of artificial intelligence algorithms in image processing. Journal of Visual Communication and Image Representation, 61, 42-9.
- Zheng Y, Iwana BK, Uchida S, 2019. Mining the displacement of max-pooling for text recognition. Pattern Recognition, 93, 558-69.
- Zou Y, Zhao L, Qin S, Pan M, Li Z. Ship target detection and identification based on SSD\_MobilenetV2. 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), 1676-80.