



**DERİN ÖĞRENMEYE DAYALI BİR İMZA DOĞRULAMA YÖNTEMİNİN
GELİŞTİRİLMESİ**

Muhammed Mutlu YAPICI

**DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

EKİM 2020

Muhammed Mutlu YAPICI tarafından hazırlanan “DERİN ÖĞRENMEYE DAYALI BİR İMZA DOĞRULAMA YÖNTEMİNİN GELİŞTİRİLMESİ” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile Gazi Üniversitesi Bilgisayar Mühendisliği Ana Bilim Dalında DOKTORA TEZİ olarak kabul edilmiştir.

Danışman: Prof. Dr. Nurettin TOPALOĞLU

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

İkinci Danışman: Dr. Adem TEKEREK

Bilgi İşlem Dairesi Başkanlığı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

Başkan: Prof. Dr. Recep DEMİRCİ

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

Üye: Prof. Dr. Sabri KOÇER

Bilgisayar Yazılımı Ana Bilim Dalı, Necmettin Erbakan Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

Üye: Doç. Dr. Hüseyin POLAT

Bilgisayar Mühendisliği Ana Bilim Dalı, Gazi Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

Üye: Doç. Dr. Baha ŞEN

Bilgisayar Yazılımı Ana Bilim Dalı, Ankara Yıldırım Beyazıt Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

Üye: Doç. Dr. Salih GÖRGÜNOĞLU

Bilgisayar Donanımı Ana Bilim Dalı, Kastamonu Üniversitesi

Bu tezin, kapsam ve kalite olarak Doktora Tezi olduğunu onaylıyorum.

Tez Savunma Tarihi: 14/10/2020

Jüri tarafından kabul edilen bu çalışmanın Doktora Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

Prof. Dr. Cevriye GENCER
Fen Bilimleri Enstitüsü Müdürü

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Muhammed Mutlu YAPICI

14/10/2020

DERİN ÖĞRENMEYE DAYALI BİR İMZA DOĞRULAMA YÖNTEMİNİN GELİŞTİRİLMESİ

(Doktora Tezi)

Muhammed Mutlu YAPICI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Ekim 2020

ÖZET

El yazısı imzalar bankacılık, hukuk ve finans gibi birçok alanda yüzlerce yıldır kullanılmaktadır. Elde edilebilirliği kolay bir biyometrik olması sebebiyle gündelik hayatta sıklıkla tercih edilmektedir. Hukuki açıdan kişilere maddi manevi birçok sorumluluk yükleyen imzaların bu kadar yaygın kullanılan bir kimlik doğrulama tekniği olması, kötü niyetli kişiler tarafından taklit edilerek sıklıkla suiistimal edilmesine sebep olmaktadır. Bu sebeple imzaların gerçekliğinin veya sahteliğinin tespit edilerek doğrulanabilmesi önemli bir araştırma konusudur. İmzalar yüksek kişisel değişkenliği barındırdıklarından, aynı kişiye ait farklı zamanda atılmış imzalar arasında bile ciddi farklılıklar görülmektedir. Buda imza doğrulamayı günümüzde de çözülmesi zor bir problem yapmaktadır. Tez çalışması kapsamında çevrimdışı el yazısı imza doğrulama alanının iki önemli problemine çözüm getirilmiştir. Birinci problem imza verilerinin yetersiz olması ve ikincisi de geliştirilen sistemlerin doğrulama başarısının düşük olmasıdır. Çevrimdışı imzalar gerçek hayatta sınırlı miktarda elde edilebildikleri için yapay zekâ yöntemlerinin eğitilmesinde yetersiz kalmaktadırlar. Bu çalışma kapsamında derin öğrenme temelli yeni bir veri artırma yöntemi kullanılarak yetersiz veri sorununa çözüm getirilmiştir. Yine derin öğrenme temelli bir sınıflandırma yöntemiyle imza doğrulama alanındaki başarı artırılmış ve çevrimdışı imza doğrulama alanına katkı sağlanmıştır. Cycle-GAN temelli veri artırma yöntemi sistemin başarısını %86,6 ‘dan %96,8 gibi yüksek bir orana getirmiştir. Veri artırma yöntemiyle birleştirilerek oluşturulan CapsNet temelli sınıflandırma yöntemi çevrimdışı imza doğrulama alanında en çok tercih edilen iki veritabanı olan GPDS ve MCYT veritabanlarında test edilmiştir. Önerilen sistem MCYT-75 veri kümesi üzerinde 10 gerçek imza ile %2,58($\pm 0,43$) EER ve %98,06 F1 puanı elde ederek literatürdeki en iyi sonuca ulaşmıştır. Yine aynı veritabanı üzerinde 5 Gerçek imza ile elde edilen %8,95($\pm 0,47$) EER oranı literatürdeki en iyi ikinci başarı olmuştur.

Bilim Kodu	: 92431
Anahtar Kelimeler	: Çevrimdışı imza doğrulama, yazar bağımlı, veri artırma, yapay zekâ, derin öğrenme
Sayfa Adedi	: 111
Danışman	: Prof. Dr. Nurettin TOPALOĞLU
2. Danışman	: Dr. Adem TEKEREK

DEVELOPMENT OF A SIGNATURE VERIFICATION METHOD BASED ON DEEP LEARNING

(Ph. D. Thesis)

Muhammed Mutlu YAPICI

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

October 2020

ABSTRACT

Handwritten signatures have been used for hundreds of years in many fields such as banking, law, and finance. As it is biometric that is obtained easily, it is frequently preferred in daily life. Because of being so widely used, signatures have been an authentication technique that is often misused by malicious people, and used for fraud. Therefore, it is an important research subject to verify the genuine or forger of signatures. Since signatures have high personal variability, there are significant differences between signatures of the same person at different times. This makes signature verification a difficult problem to solve even today. Within the scope of this study, two important problems of the offline handwritten signature verification field were solved. The first problem is that the signature data are insufficient to train systems, and the second is the poor success of the developed verification systems. Since offline signatures can be obtained in limited amounts in real life, they are insufficient to train artificial intelligence methods. In this study, we proposed a new data augmentation method based on deep learning to tackle the insufficient data problem. In addition, we contribute to the signature verification field by increasing the success of signature verification with a deep learning-based classification method. The proposed Cycle-GAN based data augmentation method has increased the success of the classification system from 86,6% to 96,8%. In addition, we proposed a new hybrid classification system composed of CapsNet and the data augmentation method. The system was tested in GPDS and MCYT databases, which are the two most preferred databases in the offline signature verification field. The proposed system achieved the best results in the literature by obtaining 2,58% ($\pm 0,43$) EER and 98,06% F1 Score with 10 genuine signatures on the MCYT-75 dataset. Furthermore, the 8,95% ($\pm 0,47$) EER rate obtained with 5 Real signatures on the same database was the second-best success in the literature..

Science Code : 92431

Key Words : Offline signature verification, writer dependent, data augmentation, artificial intelligence, deep learning

Page Number : 111

Supervisor : Prof. Dr. Nurettin TOPALOĞLU

Co Supervisor : Dr. Adem TEKEREK

TEŞEKKÜR

Bu zorlu yolda benden desteklerini esirgemeyen ve değerli bilgileriyle yolumu aydınlatan danışmanlarım Prof. Dr. Nurettin TOPALOĞLU ve Dr. Adem TEKEREK, ve tez izleme komitemde bulunan Prof. Dr. Sabri KOÇER ve Doç. Dr. Hüseyin POLAT hocalarıma sonsuz şükran ve teşekkürlerimi sunarım. Çalışmalarım boyunca maddi ve manevi her türlü destek ve karşılıksız sevgilerini benden esirgemeyen biricik anneme ve babama sonsuz minnet ve teşekkürlerimi sunarım. Ayrıca, başta Uğur DEMİR olmak üzere, doktora çalışmasının yükü altında ezildiğim ve umutsuzluğa kapıldığım anlarda her türlü bilgi ve birikimleriyle bir danışman gibi bana yardımcı olan ve beni tekrar motive eden değerli iş arkadaşlarıma teşekkürlerimi sunarım.

İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	x
ŞEKİLLERİN LİSTESİ.....	xi
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. YAPILAN ÇALIŞMALAR	5
3. İMZA DOĞRULAMA VE ÖZNİTELİK ÇIKARIM YÖNTEMLERİ	15
3.1. İmza Doğrulama	19
3.1.1. Sistemlerin performans karşılaştırmasında kullanılan metrikler	20
3.2. Öznitelik Çıkarım Yöntemleri.....	23
3.2.1. İmza genişliği (Width).....	25
3.2.2. İmza yüksekliği (Height).....	25
3.2.3. En boy oranı (Aspect ratio)	25
3.2.4. Yatay iz düşüm (Horizontal projection)	26
3.2.5. Dikey iz düşüm (Vertical projection)	26
3.2.6. Yatay iz düşümünün en büyük ve en küçük değerleri (Maximum and minimum black pixels number of horizontal projection)	26
3.2.7. Dikey iz düşümünün en büyük ve en küçük değerleri (Maximum and minimum black pixels number of vertical projection)	26
3.2.8. Siyah piksel sayısı (Area of black pixels)	27

Sayfa

3.2.9. Normalize siyah piksel sayısı (Normalized area of black pixel)	27
3.2.10. Ağırlık merkezleri (Center of gravity)	27
3.2.11. Wavelet dönüşümü	27
3.2.12. Radon dönüşümü	28
3.2.13. Yönsel (Direction-based) öznitelikler	28
3.2.14. Eğrisel (Curvature-based) öznitelikler.....	29
3.2.15. Grafometrik/yazımsal (Graphometric-based) öznitelikler.....	29
3.2.16. Gölge kodu (Shadow code - based) öznitelikleri.....	30
3.2.17. Izgara tabanlı öznitelikler	30
3.2.18. Derin öğrenmeyle öznitelik çıkarımı	31
4. PROBLEM VE ÇÖZÜM ÖNERİSİ	33
5. KULLANILAN VERİTABANLARI VE ÖN İŞLEM YÖNTEMLERİ...37	
5.1. Kullanılan Veritabanları	37
5.1.1. GPDS	37
5.1.2. MCYT.....	37
5.2. Ön İşlem Yöntemleri	38
5.2.1. Dijitalleştirme (Binarization).....	38
5.2.2. Sınır/kenar belirleme (Boundary finding)	39
5.2.3. Normalleştirme (Normalization)	40
5.2.4. Yeniden boyutlandırma	40
6. EVRİŞİMSEL SİNİR AĞLARI (CNN) TABANLI İMZA DOĞRULAMA UYGULAMASI.....	43
6.1. Evrişimsel Sinir Ağları (CNN).....	43
6.2. Yöntem	47

6.3. Deney, Sonuç ve Öneriler	48
7. ÇEVİRİMİÇİ İMZA DOĞRLUMA İÇİN DERİN ÖĞRENME TABANLI VERİ ATIRMA YÖNTEMİ.....	51
7.1. Veri Artırma Yöntemleri	52
7.2. Önerilen Yöntem	54
7.3. Deney, Sonuç ve Öneriler	59
8. EVRİŞİMSEL SİNİR AĞI MODELLERİNİN GRAFİK KARTI ÜZERİNDEKİ PERFORMANSLARININ KARŞILAŞTIRILMASI	65
8.1. Materyal ve Yöntemler.....	66
8.1.1. VGGNet.....	66
8.1.2. Residual neural network (ResNet).....	68
8.1.3. DenseNet	69
8.2. Deney ve Elde Edilen Sonuçlar.....	70
8.3. Deney, Sonuç ve Öneriler	75
9. ÇEVİRİMİÇİ İMZA DOĞRLUMA İÇİN DENSENET MODELİNİN ARAŞTIRILMASI.....	77
9.1. DenseNet Katman Yapısının Araştırılması	77
9.2. Deney, Sonuç ve Öneriler	82
10. ÇEVİRİMİÇİ EL YAZISI İMZALAR İÇİN DERİN ÖĞRENME TABANLI İMZA DOĞRULAMA YÖNTEMİ	85
10.1. Capsule Network Mimarisi	86
10.2. Deney ve Sonuçlar.....	88
11. SONUÇ VE ÖNERİLER	95
KAYNAKLAR	99
ÖZGEÇMİŞ	111

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Çevrimdışı (statik) ve çevrimiçi (dinamik) imza öznitelikleri	17
Çizelge 3.2. İmza örneklerine ait karmaşıklık matrisi	21
Çizelge 5.1. Kenar sınırları bulma algoritması	39
Çizelge 6.1. Kullanılan CNN modeline ait katman ve hiper parametre özellikleri	48
Çizelge 6.2. Kullanıcı bağımlı ve kullanıcı bağımsız yaklaşımlara göre elde edilen sonuçlar.....	49
Çizelge 7.1. Önerilen çevrimdışı el yazısı imza veri çoğaltma sistemine ait algoritma	58
Çizelge 7.2. Üç senaryoda her bir CNN modeliyle elde edilen deneysel sonuçlar	61
Çizelge 8.1. Modellere ait parametre, GPU bellek kullanımı ve zaman kullanım sonuçları	71
Çizelge 8.2. Modellere ait her bir eğitim adımı ve yığın için, başarı ve toplam eğitim süresi sonuçları	72
Çizelge 9.1. DenseNet modellerinin farklı yığın oranlarında doğruluk karşılaştırması	79
Çizelge 9.2. DenseNet modellerinin ayrıntılı karşılaştırması	80
Çizelge 9.3. Rastgele sahte örnek sayılarının performansa etkisi.....	81
Çizelge 10.1. Önerilen CapsNet modeline ait katman sayıları ve parametreler	88
Çizelge 10.2. Önerilen sisteme ait 5 gerçek eğitim verisini artırma algoritması	89
Çizelge 10.3. Önerilen sistemde GPDSsyntheticSignature veritabanı üzerinde elde edilen sonuçlar ve literatür karşılaştırması	91
Çizelge 10.4. Önerilen sistemde MVYT-75 veritabanı üzerinde elde edilen sonuçlar ve literatür karşılaştırması	91

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 3.1. İnternette seçilmiş çevrimdışı imza örnekleri.....	16
Şekil 3.2. Çevrimiçi imza örneği ve imza cihazı	16
Şekil 3.3. Gerçek ve sahte imza örnekleri, ilk satır (a, b, c) gerçek, ikinci satır (d, e, f) sahte.....	18
Şekil 3.4. Equal error rate için örnek grafik.....	23
Şekil 3.6. Ayırık radon dönüşümü uygulanmış imza, (a) imzaya ait yatay (0o) ve dikey (90o) iz düşümleri (projection), (b) imzadan elde edilmiş sinogram görüntüsü	28
Şekil 3.7. Gölge kodu örneği, (a) 7 ızgaralı maske, (b) 13 ızgaralı maske, (c) maske üzerinde belirlenmiş gölge izleri.	30
Şekil 3.8. Izgara çeşitleri (a) eşit aralıklı dikdörtgen ızgara, (b) esnek dikdörtgen ızgara, (c) ve (d) kutupsal (dairese) ızgara.....	31
Şekil 3.9. CNN öznetelikleri, yukarıdan aşağıya sırasıyla genel ve yerel öznetelikler.....	32
Şekil 5.1. İmzaya ait ön işlem görüntüleri; (a) orijinal görüntü, (b) otsu algoritması sonrası elde edilen görüntü.....	38
Şekil 5.2. İmzaya ait normelleştirme işlemi görüntüleri; (a) orijinal görüntü, (b) normelleştirme işlemi sonrası elde edilen görüntü.....	40
Şekil 5.3. İmzaya ait yeniden boyutlandırma işlemi görüntüleri; (a) orijinal görüntü, (b) yeniden boyutlandırılmış (CNN için uygun boyuta getirilmiş) görüntü	41
Şekil 6.1. LeCun'ın çalışmasından alınmış örnek bir CNN mimarisi	44
Şekil 6.2. Evrişim (convolution) işleminin temsili	45
Şekil 6.3. Nöron temsili ve formül gösterimi	45
Şekil 6.4. Kullanıcı bağımsız ve kullanıcı bağımlı yaklaşımların karşılaştırılması.....	49
Şekil 7.1. Temel veri artırma yöntemleri. (a) orijinal görüntü, (b) ayna görüntü, (c) döndürme, (d) kaydırma, (e) sündürme ya da sıkıştırma, (f) rastgele kırpmaya veri artırma yöntemleri.....	53
Şekil 7.2. Cycle-GAN mimarisinin temel yapısı	56

Şekil**Sayfa**

Şekil 7.3. Cycle-GAN yöntemine göre önerilen veri artırma sistemi mimarisi. Aynı kişinin iki farklı gerçek imzası yöntemde eşleştirilir. Sistem eşleşen gerçek imzaların üst düzey özelliklerini öğrenerek haritalandırır ve bu haritadan yeni bir orijinal imza oluşturulur. Farklı iterasyonlar (Epoch) için üretilen gerçek imza örnekleri görülmektedir	57
Şekil 7.4. GPDSsyntheticSignature (ilk satır) ve MCYT75 (ikinci satır) veritabanlarındaki örnekler kullanılarak yeniden oluşturulmuş gerçek imzalar. (a) sütunu orijinal gerçek imzalıdır, (b) ve (c) sütunları önerilen artırma yöntemiyle üretilmiş gerçek imzalıdır.	58
Şekil 7.5. CNN modelleri için doğruluk (accuracy) ve kayıp (loss) grafikleri. Her bir grafik eğitim, önerilen veri çoğaltma (ODAU), temel veri çoğaltma (CDAU) ve veri çoğaltma olmadan (WODAU) gerçekleştirilen senaryolara ait doğruluk / kayıp grafiklerinden oluşur. Şekil (a) ve (b) sırasıyla VGG16 yönteminin doğruluk ve kayıp grafiklerini göstermektedir. VGG19 için sırasıyla (c) ve (d), DenseNet121 için (e) ve (f), ResNet50 için (g) ve (h) grafiklerinde doğruluk ve kayıp grafikleri gösterilmiştir.	62
Şekil 8.1. VGGNet modellerine ait görseller. (a) VGG16 ve (b) VGG19.....	67
Şekil 8.2. ResNet modelinin temel yapısı.....	68
Şekil 8.3. DenseNet modelinin yapısı.....	69
Şekil 8.4. Modellerin doğrulama başarılarının karşılaştırma grafikleri; (a) 20 adım ve 20 yığın için, (b) 20 adım ve 50 yığın için, (c) 100 adım ve 20 yığın için, (d) 100 adım ve 50 yığın için modellerin doğruluk başarı grafikleri.....	73
Şekil 9.1. DenseNet dense bloğu temel yapısı.....	78
Şekil 10.1. Üç katmanlı basit bir CapsNet mimarisi.....	87
Şekil 10.2. DigitCaps katman temsilinden bir rakamı yeniden oluşturmak için kullanılan çözümleyici yapısı.....	87
Şekil 10.3. Önerilen veri artırma ve imza doğrulama sistemine ait genel yapı	88

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklamalar
AEs	Otomatik Kodlayıcılar (Auto-Encoders)
CNN	Everişimse Sinir Ağları (Convolutional Neural Network)
DAE	Derin Oto-Kodlayıcı (Deep Auto Encoder)
DBN	Derin İnanç Ağı (Deep Belief Network)
DL	Derin Öğrenme (Deep Learning)
DMML	Deep Multitask Metric Learning
DNN	Derin Sinir Ağı (Deep Neural Network)
DSN	Derin Yığın Ağı (Deep Stacking Network)
DWT	Dinamik Zaman Kaydırma (Dynamic Time Warping)
FM	Bulanık Sistemler (Fuzzy Methods)
G	Gerçek Orijinal İmza (Genuine Signature)
GAN	Üretken Çekişmeli Ağ (Generative Adversarial Network)
HMM	Saklı Markov Modelleri (Hidden Markov Models)
LBP	Yerel İkili Desenler (Local Binary Patterns)
LCSS	En Uzun Ortak Alt Dizinler (longest common subsequences)
LSTM	Long Short Term Memory
OCN	Bir Sınıf Bir Ağ (one-class-one-network)
OC-SVM	Tek Sınıf Destek Vektör Makinesi (One-Class Support Vector Machine)
PCA	Temel Bileşen Analizi (Principal Component Analysis)
PHOG	Yönlü Gardiyanların Piramit Histogramları (Pyramid Histogram of Oriented Gradients)
PNN	Olasılıksal Sinir Ağları (Probabilistic Neural Networks)

Kısaltmalar**Açıklamalar**

PSO	Parçacık Sürüşü Optimizasyonu algoritması (Particle Swarm Optimization)
RBF	Radyal Tabanlı Fonksiyon (Radial Based Function)
RBM	Sınırlı Boltzmann Makinelerinin (Restricted Boltzmann Machines)
RNN	Tekrarlayan Sinir Ağları (Recurrent Neural Networks)
RSS	Rastgele Sahte Sayısı (Number of Random Forgers)
SAE	Yığılmış Oto Kodlayıcı (Stacked Auto Encoder)
SIFT	Dengeli Olmayan Özellik Transformasyonu (Scale-Invariant Feature Transform)
SURF	Sağlam Hızlandırılmış Özellikler (Speeded Up Robust Features)
SVM	Destek Vektör Makineleri (Support Vector Machines)
TPU	Tensor Processing Unit
YSS	Yetenekli Sahte Sayısı (Number of Skilled Forgers)
WD	Yazara Bağımlı (Writer Dependent)
WI	Yazardan Bağımsız (Writer Independent)

1. GİRİŞ

El yazısı imzalar, bankacılık, finansal işlemler ve belge doğrulamasında yüzlerce yıldır kullanılan ve uzun bir geleneğe sahip tanımlama sistemidir. İmzalar da DNA, parmak izi, retina izi ve ses gibi kişiye ait belirli ayırt edici özellikleri taşıyan biyometrik bir yapıya sahiptir. Elde edilebilirliği kolay bir biyometrik olması ve gündelik hayatta sıkça kullanılması sebebiyle imza büyük bir değer ve önem taşımaktadır. Bireylerin imzaları ile kimlik doğrulaması günlük hayatta, özellikle hukuki belge, bankacılık ve ticari işlemler gibi alanlarda yaygın olarak kabul gören bir tekniktir.

Hukuki açıdan kişilere maddi manevi birçok sorumluluk yükleyen imzaların bu kadar yaygın kullanılan bir kimlik doğrulama tekniği olması, kötü niyetli kişiler tarafından da sıklıkla suiistimal edilmesine ve sahtekarlıklarda kullanılmasına sebep olmaktadır. Basit ya da teknolojik birçok farklı yöntemle herhangi bir kişiye ait imza istenilen belge üzerine aktararak belge sahteciliği yapılabilmektedir. Kimi zaman ise, oluşturulmuş bir belge üzerine herhangi birine ait imza taklit edilmek suretiyle sahtecilik işlemi gerçekleştirilmektedir. Bu durumda imzası taklit edilen ya da kopyalanan kişi hiçbir haberi olmadığı halde, bu belgeden doğacak sorumlulukların muhatabı konumuna gelmektedir. Bu sebeple kişilere ait biyometrik özellikleri içeren imzaların gerçekliğinin veya sahteliğinin tespit edilerek doğrulanabilmesi önemli bir araştırma konusu olmaktadır.

İmza doğrulama sistemleri, sorgulanmış bir imza ile referans örneklerini karşılaştırarak, orijinal olup olmadığını veya sahte olup olmadığını belirlemeye çalışır. Günümüzde imza doğrulama işlemi çeşitli teknolojik cihazlar, yazılımlar ve bu alanda uzmanlaşmış kişiler tarafından yapılmaktadır. Özel belge inceleme laboratuvarlarında gelişmiş mikroskoplar, aydınlatma cihazları ve görüntüleme tekniklerinin yanı sıra bu alanda geliştirilmiş yazılımlar da imza doğrulaması amacıyla kullanılmaktadır.

Günümüzde teknoloji gelişmiş olsa da imza doğrulama birçok etkene bağlı ve çözülmesi zor bir problemdir. İmzaların en önemli özelliği tümüyle aynı şekilde tekrarlanabilir olmayışlarıdır. En yetenekli kişiler dahi hiçbir zaman aynı hareketi aynı şekilde yapamazlar. Buna doğal çeşitlilik veya sınıf içi çeşitlilik (intra-class variability) denilmektedir [1]. Bu nedenle de iki imzanın üst üste çakışacak biçimde birbirinin aynı

olması mümkün olmadığından, bu durum uzmanlarca taklit belirtisi sayılmaktadır. İmzanın, yazıldığı yere, kullanılan yazı vasıtalarına (kalem, kâğıt vs.), atıldığı şartlara ve hatta sahibinin ruhi ya da fiziki durumuna, kullanılan ele göre değişmesi kaçınılmazdır. Bazı insanlar ise çeşitli hukuki sorumluluklardan kaçınmak için bilinçli olarak imzalarını değiştirebilir. Aynı şekilde bazı insanların imzalarında ise hastalıklar, duygusal durumlarının etkisi veya alışkanlıklarının olmamasından dolayı büyük biçimsel farklılıklar görülmektedir. Bu nedenle de bir kişinin imzasının doğrulaması birçok özel durumun analiz edilmesini gerektiren çözülmesi oldukça zor bir problemdir.

Yüzlerce yıldır imzalar hep kâğıt üzerine kalem ile atılagelmiştir. Teknolojideki gelişmeler sayesinde günümüzde imzalar kâğıt üzerine atılmanın yanı sıra tablet gibi imzalanabilir elektronik cihazların üzerine de atılmaktadır. Bu şekilde imzalar çevrimiçi (online) / dinamik ve çevrimdışı (offline) / statik olmak üzere atıldığı yere ve belirlenen özelliklerine göre iki kategoriye ayrılmaktadırlar.

Çevrimiçi veya çevrimdışı imzalarda imza doğrulama sistemlerinde kullanılacak özniteliklerin (kişilerin imzalarında bulunan ayırt edici özellikler) belirlenmesi büyük öneme sahiptir. Dinamik modellerle birlikte bir imzaya ait onlarca farklı öznitelik alına bilinmektedir. Günümüzde, imzalar, nispeten ucuz elektronik cihazlarla (tabletler, ultrabooklar vb.) elde edilmekte ve böylece birçok dinamik özellik yakalanabilmektedir. Özniteliklerin çok olması her zaman bir avantaj olmamaktadır. İmza eşleştirmelerinin başarısı bakımından önemli olsalar da her öznitelik aynı oranda doğrulama önemine sahip değildir. Öznitelik sayısının artması imza doğrulama sistemlerinin yavaşlamasına da sebep olmaktadır. Bu nedenle, doğrulamada hangi özelliklerin tercih edileceği sorusu vardır. Başka bir deyişle, hangi özelliklerin en büyük ayırt etme faktörüne sahip olduğunu bilmemiz gerekir. Az sayıda özellik, doğrulama işlemi için gereken hesaplama süresini de azaltmaktadır. Ancak azaltılmış özellikler imzaların karakteristik özelliklerini tanımlamada yetersiz de kalabilir. Bu nedenle imza doğrulama sistemleri için kullanılacak özniteliklerin belirlenmesi ve imzayı en iyi şekilde nitelendirecek özniteliklerin doğrulama sisteminde kullanılması büyük öneme sahiptir.

İmza doğrulama sistemlerinde ön işlem aşaması olarak sistemin doğruluğunu büyük ölçüde etkileyen öznitelik çıkarımı aşaması son yıllarda yoğun olarak tercih edilen çalışma alanlarından. Literatürdeki çalışmalara bakıldığında imza doğrulama sistemleri için,

Hotelling istatistikleri gibi yöntemleri kullanarak, var olan imza özniteliklerinden birleştirilmiş yeni ve daha etkili özellikler elde edilmeye çalışıldığı görülmektedir [2, 3]. İkinci bölümde imzada kullanılan öznitelik ve öznitelik çıkartma yöntemlerinden ayrıntılı olarak bahsedilmiştir. Yine bu çalışmalarda öznitelikler arasında en etkili olanları tespit ederek öznitelik sayısını azaltma yoluna gidildiği de görülmektedir. Bu tez çalışması kapsamında da ön işlem olarak imza doğrulama sistemleri için etkili öznitelik çıkarımı çalışması da yapılmıştır.

İmza doğrulama, yukarıda bahsedildiği gibi çözülmesi gereken önemli bir problemdir. Bu problemin önemi: 2004'ten beri imza doğrulaması için yapılan çeşitli çalışmalardan anlaşılmaktadır. Bu kapsamda birçok yarışma düzenlenmiştir. Bu yarışmalarda, gerçek ve sahte imzaların tespitine yönelik çeşitli takımlar veya yarışmacılar farklı algoritmalar sunmuşlardır. Bu algoritmalar, değerli bir bilgi kaynağıdır, çünkü bu algoritmaların tanımlanma aşamaları ve kullanılan ön işleme teknikleri yayınlanmış ve erişilebilir durumdadır [4]. Son yıllarda, SigComp2011 [5], 4NSigComp2012 [6] ve SigWiComp2013 [7] gibi bir dizi yarışmalar yapılmaktadır [2]. Bu yarışmaların düzenleyicileri elde edilen sonuçların karşılaştırılmasına olanak tanıyan karşılaştırmalı veri tabanlarını da yayınlamaktadırlar. Bu probleme çözüm getirmek için her geçen gün yeni çalışmalar da yapılmaktadır. Bu çalışmaların bir kısmı imzayı doğrudan ayırt etmek için yapılırken bir kısmı da imzaları ayırt etmede kullandığımız öznitelik ve mesafe ölçüm metriğini geliştirmek için yapılmaktadır. Güvenilir imza doğrulama sistemleri oluşturmak için Saklı Markov Modelleri (HMM) [8, 9], Destek Vektör Makineleri (SVM) [10], Dinamik Zaman Kaydırma (DTW) [11, 12], Temel Bileşen Analizi (PCA) [13], bulanık sistemler (Fuzzy Methods) [14], Olasılıksal Sinir Ağı (PNN) [3] gibi bir çok farklı doğrulama yöntemleri kullanılarak çözülmeye çalışılmıştır. Aynı şekilde son zamanlarda, yüz doğrulanma problemini çözmek için, metrik öğrenme tabanlı sistemler kullanmaya yönelik bazı girişimler yapılmıştır [15]. İyi bir mesafe metriği, benzer örnekler için çıktısının sıfıra yakın olduğu ve benzer olmayanlar için tersine, büyük bir pozitif sayı olduğu bir benzerlik ölçüsüdür. İmza doğrulama literatüründe, uzaklık ölçütleri ağırlıklı olarak Euclidean ve Mahalanobis [4] olmak üzere orijinal ve sahte örnekler arasındaki benzerlikleri ve farklılıkları keşfedebilecek kadar güçlü olmayan, çeşitli uzaklık temelli sınıflandırıcılar kullanılmıştır. Bu metrik sistemlerin yanında nispeten daha güçlü olan Deep Multitask Metric Learning (DMML) [16] gibi metrik sistemleriyle de çözüm arandığı görülmektedir.

İmza doğrulamaya yönelik gerçekleştirilen tüm bu çalışmalarla birlikte henüz bu probleme tatmin edici bir çözüm getirilemediği de bir gerçektir. Günümüzde birçok alanda etkili çözümler sunan derin öğrenme yaklaşımının henüz imza doğrulama problemine tam olarak uygulanmadığını görüyoruz. Derin öğrenme, otonom araçlar, nesne tanıma, hareket tanıma, ses tanıma gibi birçok alanda ciddi gelişmeler sağlamıştır. Google, NVidia, Facebook, Microsoft gibi büyük firmalar derin öğrenme alanına milyarlarca dolar yatırım yaparak bu alana ciddi katkılar sağlamaktadırlar. Bu tez çalışması kapsamında birçok alanda başarılı bir şekilde uygulanmış olan derin öğrenme (Deep Learning (DL)) yaklaşımının imza doğrulama problemine uygulanması amaçlanmıştır.

2. YAPILAN ÇALIŞMALAR

İmza doğrulama, bir önceki bölümde de bahsedildiği gibi çözülmesi gereken önemli bir problemdir. Bu problemin önemi: 2004'ten beri imza tanıma ve doğrulama için yapılan çeşitli çalışmalardan anlaşılmaktadır. Bu kapsamda birçok yarışma da düzenlenmiştir [5–7]. Bu alandaki çalışmaları incelediğimizde, imza doğrulama ve sınıflandırma algoritmalarının geliştirilmesinin yanında doğrulama işlemi için kullanılan uzaklık metrik sistemlerinin geliştirilmesine ve doğrulamanın etkisini artıracak öznitelik çıkarımı işlemlerinin geliştirilmesine yönelik de olduğu görülmektedir.

Literatürdeki öznitelik çıkarımı için kullanılan yöntemlere baktığımızda geometrik özellik çıkarımı, grafometrik özellik çıkarımı, yönsel özellik çıkarımı, matematik dönüşümü, gölge kodu özellik çıkarımı, dokusal özellik çıkarımı, ara nokta eşleme, sahte dinamik özellik çıkarımı, kompozit özellik çıkarımı ve özellik öğrenimi gibi yöntemlerin kullanıldığını görmekteyiz. Bu bölümde kısaca değinilmiş olan bu öznitelik çıkarım yöntemleri, “İmza Doğrulama ve Öznitelik Çıkarım Yöntemleri” bölümünde detaylı bir biçimde anlatılmıştır.

Yapılan birçok çalışmada imzaların geometrik özelliklerini elde edebilmek için ızgara yaklaşımının kullanıldığı görülmektedir. Geometrik özellikler bir imzanın genel şeklini ölçmekte kullanılır. Geometrik öznitelik çıkarımı, imza yüksekliği, genişliği, kalibresi (yükseklik genişliği oranı) ve alan gibi temel tanımlayıcıları içerdiği gibi daha karmaşık tanımlayıcılar olan, bitiş noktalarının sayısı ve kapalı halkaları da içermektedir [17]. Yapılan bazı çalışmalarda imzayı bir kartezyen ızgaraya bölerek ve ızgaranın her hücresinde bulunan piksel yoğunluğu hesaplanarak yerel geometrik öznitelikler elde edilmektedir [17–19]. Bu şekilde yerel öznitelik çıkarımına farklı bir örnekte, Kai Huang ve Hong Yan [20] tarafından yapılan çalışmada değinilmiştir. Bu çalışmada, girdi olarak birden fazla imza temsilini (anahat, iskelet vb.) kullanıp, her temsili ızgaralı bir biçimde ikiye bölmüş ve imza için öznitelik olarak ızgaların her bir hücresindeki piksel yoğunluğunu kullanmışlardır [21].

Adli belge incelemecileri ve el yazısının orijinalliğin tespit edilmesi gibi çeşitli amaçlar için grafoloji ve grafikometri sıklıkla kullanılmaktadır. Oliveira ve arkadaşları [22], yaptıkları çalışmada grafoloji ve grafikometriyi otomatik imza doğrulaması için kullanmayı önermişlerdir.

Yönlü özellikler görüntüdeki çizgilerin yönünü tanımlamak için kullanılır. Sabourin [23] ve Drouhard [24], imza ana hattının gradyanından (Directional-PDF (Probability Density Function)) Yönlü Olasılık Yoğunluk Fonksiyonu çıkarmışlardır. Rivard [25] tarafından yapılan bir çalışma da ise, çoklu ölçekli ızgara sistemi kullanarak imza öznitelikleri çıkartılmış ve umut verici sonuçlar elde edilmiştir. Zhang ve arkadaşları ise çalışmalarında Yönlü gradyanların piramit histogramının (Pyramid Histogram of Oriented Gradients, PHOG) kullanımını araştırmıştır [26]. Bu öznitelik tanımlayıcıyla bir görüntünün yerel şekillerini elde etmekte ve aynı zamanda birden fazla ölçekte kenar yönelimleri histogramı oluşturmaktadırlar.

Matematik dönüşümleri öznitelik çıkarımında sıklıkla kullanılmaktadır. İmza doğrulama yaklaşımlarında da Hızlı Hadamart dönüşümü ve spektrum analizi, Contourlet dönüşümü, Radon dönüşümü ve Wavelet dönüşümü öznitelik çıkarımı için bazı çalışmalarda kullanılmıştır. Bu çalışmalara ait bilgileri imza doğrulama üzerine yapılan literatür taramalarında görülmektedir [21].

Diğer bir öznitelik çıkarım yöntemi de gölge kodudur. Sabourin ve arkadaşları [23, 27] imza doğrulaması için Genişletilmiş Gölge Kodunu (Extended Shadow Code) önermişlerdir. Yatay, dikey ve köşegen çubukları içeren imza görüntüsü üzerine, her çubuk sabit bir sayıda kutu içeren bir ızgara şeklinde yerleştirilir. İmza görüntüsünün her pikseli, daha sonra, her bir yöndeki en yakın çubuğa (diğer bir deyişle dikey / yatay / köşegen çubuklardaki "gölge") yansır ve ilgili bölmeyi etkinleştirdiği varsayılır. Tüm pikseller ilgili çubuklara yansıtıldıktan sonra, her çubuktaki aktif alanlar sayılır ve imza tanımı olarak kullanılır. Bu öznitelik çıkarma yönteminin, sırasıyla, yazardan bağımsız (writer-independent - WI) ve yazara bağımlı (writer-dependent -WD) sınıflandırmalarda da kullanıldığı Rivard [25] ve Eskander [28]'in çalışmalarında görülmektedir.

Desen öznitelikleri, özellikle Yerel İkili Desenler'in (Local Binary Patterns - LBP) çeşitli kullanım şekilleri öznitelik çıkarımı için sıklıkla kullanılmıştır. LBP operatörü Ojala ve arkadaşları [29] tarafından doku görüntüleri için ayırıcı öznitelik çıkarıcı olarak sunulmuştur. Desen öznitelik çıkarımı yöntemiyle, bir resimdeki her bir pikselin değeri, piksele komşu olan farklı 8 pikseli hesaplanmaktadır. Yeni piksel değeri tüm komşulukların bir filtrenin değerlerinin birebir çarpımlarının toplamı şeklinde hesaplanmaktadır. LBP 'nin

birçok çeşidi farklı çalışmalarda önerilmiştir, bunlar çoğunlukla dairesel komşuları tespit eden türevler olarak eşit aralıklarla yerleştirilmiştir [30].

Nesne tanıma ve 3D yeniden yapılandırma gibi bilgisayarlı görme görevleri için SIFT Ölçek Dengeli Olmayan Özellik Aktarımı (Scale-Invariant Feature Transform) ve SURF Sağlam Hızlandırılmış Özellikler (Speeded Up Robust Features)) gibi eşleştirme yöntemleriyle yerel ilgi noktaları belirlenerek öznitelik çıkarımlarında sıklıkla kullanılan diğer bir yöntemdir. Bazı çalışmalarda, bu yöntemlerin imza doğrulama sistemi için öznitelik çıkarma aşamasında da kullanıldığı görülmektedir. Ruiz-del-Solar ve arkadaşları [31] yazara bağlı bir sınıf oluşturmak için hem sorgu hem de referans örneklerinden yerel ilgi noktaları çıkarmak için SIFT kullanmışlardır. Malik ve arkadaşları [32], gerçek, sahte ve maskelenmiş imzalar arasında sınıflandırma yapmak için SURF özelliklerini kullanmıştır. Önce imza görüntülerinde ilgi noktaları çıkarmak için SURF kullanmışlar ve bu özellikleri, imzaların yerel istikrarını değerlendirmek için (diğer bir deyişle orijinal imzaların zamanla daha kararlı olan kısımları) kullanmışlardır.

Oliveira ve arkadaşları [22] grafikometrik çalışmalara dayanan bir dizi sahte dinamik özellik (pseudo-dynamic features) çıkarımı yaklaşımı sunmuşlardır. Piksellerin Dağılımı yöntemiyle, hareketlerin genliğini ölçerek hız, süreklilik ve tekdüzelik hakkında bilgi çıkarımında bulunmuşlar, diğer bir yöntem olan Eğim ve Form ölçümü ile imzaların iç bükey eğimlerinin ölçülmesini sağlamışlardır. Bertolini ve arkadaşları [33] imzanın eğriliğini göz önüne alan yeni bir öznitelik tanımlayıcı önermişlerdir.

Porwik ve arkadaşları 2016 yılında yaptıkları çalışmalarında imza doğrulama için yeni birleşik (kompozit) öznitelik çıkarım yöntemi önermiştir. Özellikler ve bağlantılı benzerlik katsayılarını kullanarak yeni bir birleşik imza özelliği oluşturulmuşlar. Daha sonra, oluşturulan bu birleşik özellikler içerisinde en çok etki edeni tespit edilerek her bir birey için en iyi ayırt ediciliğe sahip öznitelikler seçilmiştir. Bu eleme ve öznitelik seçme işlemi Hotelling yaklaşımı ile yapılmıştır [2, 3].

Son yıllarda yapılan çalışmalara bakıldığında yukarıda anlatılan bazı öznitelik çıkarım yöntemlerinde de olduğu gibi el yapımı öznitelik çıkarımlarına olan güven azalmıştır. Bu nedenle Derin öğrenme yönteminde olduğu gibi ham veriden öznitelik çıkarma yöntemlerine olan ilgi daha da artmıştır. Ne var ki yine yapılan çalışmalara baktığımızda bu tekniklerin

son yıllarda pek çok bilgisayar görme görevi için yaygın olarak kullanılmasına rağmen imza doğrulaması için yaygın olarak kullanılmadığı görülmektedir [21]. Halajzadeh [34] CNN'leri (Convolutional Neurol Network) Farsça imza doğrulaması için kullanmış ancak testlerinde yalnızca rasgele seçilmiş sahte imza örneği kullanmıştır. Sadece sahte imzaların kullanılması sınıflandırıcı performansının yanlış değerlendirilmesi riskini doğurmaktadır. Luiz G. Hafemann ve arkadaşları CNN (Convolutional Neurol Network) kullanarak yazara bağımlı imza kümelerini sınıflandırmak üzere, yazardan bağımsız bir imza öznelik çıkarım modeli geliştirmişlerdir. Ancak çalışmalarında da belirttikleri gibi eğitim verisi olarak sadece gerçek imzaları kullanmışlardır. Sadece bazı örneklerde yetenekli sahte imza örneklerini de işleme katmışlardır. Bu nedenle de eğitim verileri oldukça azdır ve öğrenmenin başarısını etkilediği düşünülmektedir [35].

Literatürdeki imza verilerinin sınıflandırması için gerekli öznelik çıkarım yöntemlerine yukarıda değinilmiştir. İmza doğrulama ve sınıflandırma yöntemleri de literatürdeki önemli çalışma alanlarından. İmza verilerinin sınıflandırılması ve sahte imza ile orijinal imzanın ayırt edilmesi amacıyla birçok çalışma yapılmıştır. Statik veya çevrimdışı (offline) imza ve dinamik veya çevrimiçi imzaların taklitlerinden ayırt edilmesi amacıyla günümüze kadar pek çok örüntü tanıma, sınıflandırma ve yapay zekâ algoritması denenmiştir. Bu algoritmaların bazıları başarılı sonuçlar vermiş olsalar da günümüzde henüz beklenen performans yakalanmış değildir. Özellikle çevrimdışı imzalarda öznelik sayısının kısıtlı olması algoritmaların performanslarını önemli ölçüde etkilemektedir. Bu nedenle de yapılan çalışmalar incelendiğinde çevrimdışı imza uygulamalarında iki farklı yöntemin izlendiği görülmektedir. Bu yöntemler yazara bağımlı ve yazardan bağımsız yöntemlerdir. Yazara bağımlı yöntemlerde yapılan öznelik çıkarımı, her bir yazarın kendi imzalarını kapsarken, yani kişiye özel ve daha spesifik öznelikler elde edilirken, yazardan bağımsız yöntemlerde yapılan öznelik çıkarımı, birden farklı yazarların gerçek imzalarını kapsamaktadır ve elde edilen öznelikler daha geneldir. Çevrimdışı imza sınıflandırmasında bu iki yöntem ve birleşimleri sıklıkla kullanılmıştır.

Eskander ve arkadaşları, yazara bağımlı sınıflandırıcıların eğitim için yeterince orijinal imza örneğine sahip olmadıkları durumda yanlış sınıflandırdıklarını belirtmişlerdir. Bu sorunun üstesinden gelebilmek için yazardan bağımsız bir sınıfın, kullanıcı için yeterli sayıda orijinal imza örneği elde edilinceye kadar sınıflandırmada kullanıldığı melez bir yazardan bağımsız-yazara bağımlı bir eğitim sistemi önermişlerdir [28]. Yılmaz, yaptığı doktor çalışmasında

hem yazardan bağımsız hem de yazara bağlı sınıflandırmaların sonuçlarının birleştirildiği karma bir yaklaşım önermektedir [36].

Porwik ve arkadaşları imza doğrulama alanında yaptıkları bir çalışmada sınıflandırma algoritmasının bağımlı iki parametresinin Parçacık Sürüsü Optimizasyonu algoritması (Particle Swarm Optimization -PSO) ile seçildiği ve sınıflandırma işleminin Olasılıksal Sinir Ağı (Probabilistic Neural Network- PNN) ile yapıldığı adaptif bir yaklaşım önermişlerdir. Bu çalışmalarında da daha önceki çalışmalarında önerdikleri birleşik öznitelik çıkarımı ve etkin öznitelik azaltmada kullanılan Hotelling algoritmasını kullanmışlardır. İki sınıflı PNN sınıflandırıcısının diğer sınıflandırıcılara kıyasla yüksek başarı gösterdiğini belirtmişlerdir [3].

Julian Fierrez ve arkadaşları Saklı Markov Modelleri (Hidden Markov Models - HMM) algoritmasına dayanan bir imza doğrulama çalışması yapmışlardır. Bu çalışmanın sonucunda model için gerekli optimum konfigürasyonun, 2 HMM ve 32 Gauss değerleriyle elde ettiklerini bildirmişlerdir. Çalışmalarını 145 denekten alınan 7000 'den fazla imza içeren MCYT bimodal biyometrik veri tabanı üzerinden gerçekleştirmişlerdir [8]. Edson J.R. Justino ve arkadaşları yaptıkları çalışmada içerisinde ön işlem, öznitelik çıkarımı ve sınıflandırma işlemlerinin bulunduğu üç aşamalı bir imza doğrulama sistemi geliştirmişlerdir. Sınıflandırma işlemini HMM modeline göre her bir yazarın imzasını temsil edebilen, aynı zamanda farklı kişiler arası ayırt ediciliği de olan bir yaklaşım sunmuşlardır [19]. A. El-Yacoubi ve arkadaşları, çevrimdışı imza doğrulama için HMMs ve Çapraz doğrulama (Cross Validation) tabanlı bir yaklaşım önermişlerdir. Çapraz doğrulama ilkesini, sadece en iyi HMM modellerini bulmak için değil aynı zamanda her yazar için en uygun kabul / ret kararı için eşik değerini elde etmek için kullanmışlardır. Doğrulama işleminde kullanılacak optimal bir karar kuralı belirlemek için gerekli olan yazara bağlı parametreleri dinamik ve otomatik olarak türetmişlerdir [18].

Christian Gruber ve arkadaşları çevrimiçi imza doğrulama için SVM algoritmasını kullanmışlardır. Çalışmalarında destek vektör makineleri (Support Vector Machine - SVM) ile imza özniteliklerinin bir çekirdek fonksiyona benzerliğini ölçen en uzun ortak alt dizinler (Longest Common Subsequences LCSS) algılama algoritmasını birleştirerek kullanmışlardır. LCSS sayesinde imzanın yerel farklılıklarının da işleme alınarak daha hassas bir karşılaştırma yaptıklarını bildirmişlerdir [10]. Yasmine Guerbai ve arkadaşları

sınırlı sayıda imza ve gerçek imzanın az olduğu durumlarda sıklıkla karşılaşılan etkili bir El Yazısı İmza Doğrulama Sistemi tasarlama sorununu SVM algoritmasıyla çözmeye çalışmışlardır. Çalışmalarında, sadece orijinal imzaları dikkate alan, yazardan bağımsız parametrelere dayanan Tek Sınıf Destek Vektör Makinesi (One-Class Support Vector Machine, OC-SVM) kullanımını önermişlerdir. İmza örneklerinin az olduğu durumlarda, yanlış sınıflandırmayı azaltmak için, OC-SVM çekirdeğinde kullanılan farklı mesafeleri birleştirerek optimum eşiği ayarlayan, OC-SVM'de kullanılan karar fonksiyonunu değiştirmişlerdir [37].

Edson J.R. Justino ve arkadaşları yaptıkları çalışmada SVM ve HMM algoritmaları ile çevrimdışı imzalar üzerinde çalışmışlardır. Bu iki algoritmanın çevrimdışı imza eğitimi ve doğrulaması üzerine başarılarını karşılaştıran araştırmacılar SVM 'nin HMM 'ye göre daha başarılı olduğunu bildirmişlerdir [38].

Son yıllarda birçok bilgisayarlı görme alanında olduğu gibi imza doğrulama yaklaşımlarında da yapay sinir ağları sıklıkla kullanılmaya başlanmıştır. Huang ve Yan yaptıkları çalışmada çevrimdışı imza doğrulama için sinir ağlarını kullanmayı önermişlerdir. Çalışmalarında öznitelik çıkarımı için geometrik öznitelik çıkarımı yöntemini kullanmışlar ve sinir ağı sınıflandırmasına dayanan bir çevrimdışı imza doğrulama yöntemi sunulmuşlardır. Bu çalışmalarında sinir ağını eğitmek için orijinal ve sahte imza örneklerinden yapay olarak elde ettikleri örnekleri kullanmışlardır [20]. Baltzakisa ve Papamarkos, çevrimdışı imza doğrulama için iki aşamalı sinir ağı sınıflandırıcısı kullanmışlardır. Önerilen sistem global, grid ve doku öznitelik çıkarımına dayanmaktadır. Bu öznitelik kümelerinden her biri için özel bir iki aşamalı Perceptron OCON (bir sınıf-bir-ağ, One-Class-One-Network) sınıflandırma yapısı uygulanmıştır. Birinci aşamada, sınıflandırıcı, sinir ağlarının karar sonuçlarını ve üç öznitelik kümesinden öklid mesafesine göre elde edilen sonuçları birleştirir. Birinci aşamada elde edilen sınıflandırıcı sonuçları, nihai kararı elde etmek için ikinci aşamada, radyal tabanlı fonksiyon (RBF-Radial Based Function) sinir ağı yapısını girdi olarak besler [17]. McCabe [39] ve arkadaşları çevrimdışı imza doğrulama için yapay sinir ağı tabanlı bir sınıflandırma mekanizması kullanmışlardır. Yazarlar çalışmalarında doğrusal bir sinir ağını (linear neural network) ve Çok Katmanlı Algılayıcı (Multi-Layer Perceptron-MLP) ağını dinamik imza öznitelikleriyle eğiterek performans karşılaştırması yapmışlardır. En başarılı modelin tek katmanlı MLP olduğunu bildirmişlerdir. Iranmanesh ve arkadaşları [40] da sınıflandırma için MLP kullanmışlardır. Yazarlar farklı olarak

sınıflandırma işlemine geçmeden önce en etkili öznitelikleri seçebilmek için temel bileşenler analizini Pprincipal Component Analysis (PCA)) kullanmışlardır. Tolosana ve arkadaşları [41] 2018 de yayınladıkları makalede bir derin öğrenme modeli olan Recurrent Neural Networks (RNNs) modelini Siames mimarisi ile tasarlayarak çevrimdışı kullanıcı bağımsız imza doğrulama için bir yöntem önermişlerdir. Elde ettikleri sonuçların literatürdeki en iyi sonuçlar olduğunu iddia etmişlerdir ve derin öğrenme yönteminin imza doğrulama alanında da umut vadettiğini göstermişlerdir.

Çevrimdışı / statik (offline) imza doğrulama sistemlerinin yanı sıra çevrimiçi / dinamik (online) imza doğrulama sistemlerinde de sinir ağlarının sıklıkla kullanıldığı görülmektedir. Maged M.M. Fahmy, dinamik imza doğrulama için yapay sinir ağı sınıflandırma algoritmasını kullanmıştır. Çalışmasında daha çok öznitelik çıkarımı üzerinde durmuştur. Araştırmasında, ayrık dalgacık dönüşümü (Discrete Wavelet Transform - DWT) yöntemini kullanarak özniteliklerin çıkarımı yapmış ve yapay sinir ağı algoritmasını kullanarak çevrimiçi imza doğrulama işlemini gerçekleştirmiştir. Öznitelik olarak çıkarttığı imzanın x ve y pozisyonlarını kullanarak üçüncü bir öznitelik olan kalem hareket açısını elde etmiştir. Kalem konumu ve kalem hareket açısı boyutlarındaki farklılıkları azaltmak için veriler normalleştirilerek yeniden örneklendirmiştir. Gerçek bir imza ile sahtesi arasındaki farkı arttırmak için imzayı, DWT alanında doğrulamıştır [11]. Bromley, LeCun ve arkadaşları 1994 yılında Siames Neural Network kullanarak çevrimiçi imza doğrulaması gerçekleştirmiştir. Araştırmacılar çalışmalarında girdi olarak iki farklı imza örneği verdikleri iki farklı yapay sinir ağını eğitmişler ve çıktı olarak iki ağdan elde ettikleri sonuçların farkını almışlardır. Böylece iki imza arasındaki farklılıkları tespit etmeye çalışmışlardır. Her iki koldaki yapay sinir ağının son katmanında öznitelik vektörleri elde edilmiş ve bu öznitelik vektörleri karşılaştırarak imzaların farklarını ortaya koymaya çalışmışlardır. Tüm ağı tabletler ile kendi topladıkları imza veri kümesinde eğitip test etmişlerdir. Dinamik imzaların farklı öznitelikleri ile ağı birden çok modelleyerek en iyi ayırt edici özellikleri seçmeye çalışmışlardır [42]. Sabourin [23] ve Drouhard [24], imza ana hattının gradyanından (Directional-PDF (Probability Density Function)) Yönlü Olasılık Yoğunluk Fonksiyonunu kullanarak çıkardıkları öznitelikleri tam bağlı Backpropagation Network (BPN) ile sınıflandırmaya çalışmışlardır. Öznitelik boyutunun aşırı fazla olması sebebiyle BPN performansını artırmak için k En Yakın Komşu (k Nearest Neighbour (kNN)) algoritmasıyla azaltmayı denemişlerdir. Son olarak ta hem BPN ve kNN sınıflandırıcılarının performanslarını kıyaslamışlardır. Pansare ve Bhatia [43] çevrimdışı imza doğrulaması için

geometrik öznitelik çıkarımı yöntemleriyle elde ettikleri öznitelikleri yapay sinir ağları ile eğiterek sınıflandırmaya çalışmışlardır. Ooi ve arkadaşları [44] çevrimdışı imza doğrulaması için olasılıksal yapay sinir ağı temelli bir yaklaşım önermişlerdir. Önerilen yaklaşımda öznitelik çıkarımı için ayırık radon dönüşümü (Discrete Radon Transform (DRT)) kullanmışlardır. Bu öznitelik çıkarım yönteminden elde edilen verileri azaltabilmek için ise temel bileşenler analizini (principal component analysis (PCA)) kullanmışlar ve son olarak olasılıksal sinir ağı ile sınıflandırma işlemini gerçekleştirmişlerdir.

Derin öğrenme yöntemlerinin özellikle de grafik kartı teknolojisindeki gelişmelerden sonra birçok alanda elde ettiği başarılar, imza doğrulama üzerine çalışan araştırmacıların da ilgisini çekmiştir. Araştırmacılar Derin öğrenme yöntemlerini imza doğrulama alanında uygulayarak bu alanda başarıyı artırmayı hedeflemişlerdir. Luiz G. Hafemann ve arkadaşları CNN (Convolutional Neurol Network) kullanarak yazara bağımlı imza kümelerini sınıflandırmak üzere, yazardan bağımsız bir imza öznitelik çıkarım modeli geliştirmişlerdir [35]. Özellikle çevrimdışı imzalarda dinamik özniteliklerin elde edilememesi ve bu nedenle son derece kısıtlı bir öznitelik ile imza karakteristiklerinin çıkarılmasının zor olması, çevrimdışı imza doğrulamanın performansını önemli ölçüde düşürmektedir. Bu nedenle yazarlar çalışmalarında yazardan bağımsız olarak CNN ile otomatik olarak elde edilmiş imza öznitelikleri ile yazara bağımlı bir çevrimdışı imza doğrulama modeli geliştirilmiştir. Ancak çalışmalarında da belirttikleri gibi, eğitim verisi olarak sadece gerçek imzaları kullanmışlardır. Sadece bazı örneklerde yetenekli sahte imza örneklerini de işleme katmışlardır. Bu nedenle de eğitim verileri oldukça azdır ve öğrenmenin başarısını etkilediğini düşünmekteyiz. Zhang ve arkadaşları [45] Deep Convolutional Generative Adversarial Networks (DCGANs) kullanarak çevrimdışı imza doğrulaması gerçekleştirmek üzere çok fazlı denetimsiz bir sınıflandırma yöntemi önermişlerdir. Önerdikleri sistemde yetersiz veri sorununu çözebilmek için sistemi hem kullanıcı bağımsız hem de kullanıcı bağımlı imzalarla eğitmişlerdir. Souza ve arkadaşları [46] yazardan bağımsız bir sınıflandırma için destek vektör makinaları (support vector machines-SVM) sistemi önermişlerdir. Önerdikleri sistemde kullanıcı bağımlı olarak öznitelik çıkarımında CNN yöntemini kullanmışlardır. 2019 yılında yayınlanan bir makalede, Sam ve ekibi [47] çevrimiçi imza doğrulama için Google inception CNN modelinin farklı versiyonlarını farklı kullanıcı sayıları ile test ederek performans karşılaştırması yapmışlardır. İmza görüntülerde 22 katmanlı inception V1 modelinin 42 katmanlı inception V3 modelinden daha iyi performansa sahip olduğunu bildirmişlerdir.

Yapay sinir ağlarının bilgisayarlı görme alanında olduğu gibi imza doğrulama sistemlerinde de kullanılmasında büyük sınırlılıklar ve dezavantajlar bulunmaktadır. Bunlardan ilki sinir ağının boyutu problemidir. Yapay sinir ağının boyutunun büyümesi sinir ağı yapısının ve sonuçlarının da iyileşmesi anlamında gelmektedir. Ancak büyüyen bir ağ karmaşıklaşır ve eğitmek için çok fazla işlem gücüne ihtiyaç duyulur. Bu nedenle büyük ağları eğitmek oldukça zordur ve karmaşık bir süreçtir [17]. Günümüzde bu gibi karmaşık ağları eğitmede sadece CPU gücünden değil GPU 'lardan ve hatta derin öğrenme gibi gelişmiş çok katmanlı ağlarda kullanılmak için Google tarafından üretilmiş TPU (Tensor Processing Unit)'lardan faydalanılması bu dezavantajı ortadan kaldırmış görünmektedir. Bu nedenle imza doğrulama sistemlerinde de derin öğrenme (Deep Learning) ilerleyen yıllarda sıklıkla tercih edilecek bir sınıflandırma yöntemi olarak ön görülmektedir. Henüz bir kaç tane olsa da derin öğrenmeyle bu alanda yapılan çalışmalar umut vadeden sonuçlarıyla bunun ispatı niteliğindedir. Diğer bir sınırlılık ise eğitim verisinin az olması ve/veya geniş bir yelpazede olmaması sorunudur. Eğitim esnasında kullanılan verilerin sayısı kadar bu verilerin test edilmesi istenilen alandaki örnek uzayını yeterince kapsıyor olması da derin öğrenme algoritmalarının başarısını önemli ölçüde etkilemektedir. İmza doğrulama alanını etkileyecek en büyük sorunlardan birisi tam olarak bahsettiğimiz bu veri yetersizliğidir. İmza doğrulaması için kullanılan veri sayısının derin öğrenme gibi gelişmiş ağları eğitmede yetersiz olduğu aşikardır. Bu verilerin sayısının artırılması geliştirilen yöntemlerin başarısı için hayati önem taşımaktadır.

Derin öğrenme tekniklerindeki bu gelişmeler birçok alanda olduğu gibi imza tanıma ve imza doğrulama alanlarında da önemli gelişmelere ön ayak olacaktır. Bu tezin omurgasını da derin öğrenme tekniklerinin imza doğrulama alanındaki başarısına katkıda bulunacak, derin öğrenme tabanlı imza doğrulama sisteminin geliştirilmesi oluşturmaktadır. Yapılan çalışma ile, özellikle yetersiz özniteliklere sahip olan çevrimdışı imzaların günümüzde başarısız sayılabilecek doğrulama performansını artırmak hedeflenmektedir. Yine bu alanda derin öğrenme yöntemleriyle yapılan çalışmalardan farklı olarak, çevrimdışı imza doğrulamanın en büyük sınırlılıklarından biri olan eğitim verisinin sınırlı sayıda olması sorununu [25, 11] da geliştirilecek veri artırma yöntemle aşmak amaçlanmaktadır.

3. İMZA DOĞRULAMA VE ÖZNİTELİK ÇIKARIM YÖNTEMLERİ

Türk Dil Kurumu (TDK) tanımına göre; İmza [48], “bir kimsenin herhangi bir belgeyi yazdığını veya onayladığını belirtmek için her zaman aynı biçimde kullandığı işaret” anlamına gelmektedir. Literatürde yapılan imza tanımları da TDK’nin bu tanımı ile paralellik göstermektedir.

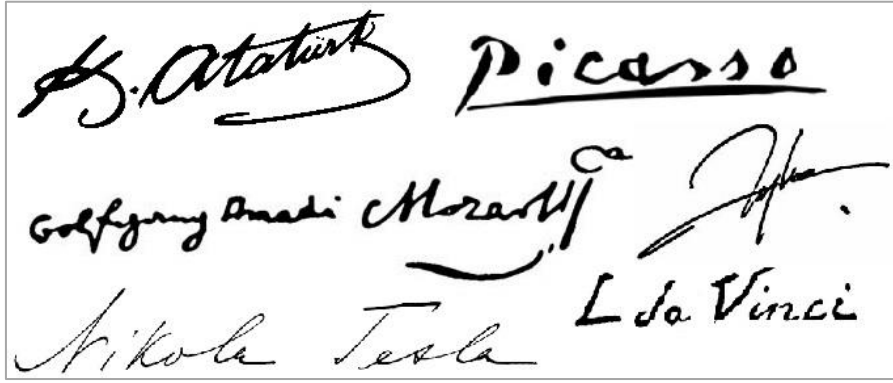
Birincioğlu [49] imzayı şöyle tanımlamaktadır; “Sahibini kuşkuya yer bırakmayacak bir biçimde belirleyen ve bir belgeyi onaylama iradesini kesin olarak belirten ayırt edici ve kişisel el yazması her tür işaret”.

İmzalar yüzyıllardan beri kişileri ayırt etmede kullanılan el ile yazılmış kişisel ad ya da işaretten oluşan bir biyometriktir ve atıldığı belgedeki her türlü şart ve yasal yükümlülüklerin imzayı atan kişi tarafından kabul edildiğini gösterir.

İmzalar da diğer biyometrik yapılar gibi karakteristik özelliklerini taşıyan çeşitli niteliklerden oluşmaktadır. Bu nedenle bir imza doğrulama sisteminin başarısı bu niteliklerin ne kadar iyi tespit edilerek diğer genel özelliklerden ayırt edebildiğine bağlıdır. Uygun olan niteliklerin seçimi biyometrik sistemlerin doğruluğunu değiştirebileceğinden özelliklerin seçimi doğrulama sürecinde önemli bir kriterdir. Bir imzanın karakteristik özellikleri imzanın türüne bağlı olarak farklılık gösterir ve statik (çevrimdışı-offline) ve dinamik (çevrimiçi, online) imzalar için ayrı ayrı değerlendirilir.

Statik (çevrimdışı, offline) imza sadece imzanın şeklini içerir. İmzaların belgelerden basit bir şekilde doğrulanması için kullanılabilir. Statik imza doğrulamada, kullanıcı kâğıda bir imza atar ve daha sonra optik bir tarayıcı resmi tarayarak görüntüyü sayısallaştırır. Yalnızca imza şeklinin tanındığı bir tekniktir. Çevrimdışı imza doğrulama sistemlerinde genellikle imza görüntüleri yüksek çözünürlüklerde (600dpi vb.) taranır. Çevrimdışı imzalar gri tonlamalı veya ikili görüntüler olarak saklanırlar. Şekil 3.1.’de çevrimdışı imzalara birkaç örnek görülmektedir. Çevrimdışı sistemlerde imzanın daha çok şekilsel özelliklerinden yararlanılarak tanıma ya da doğrulama işlemi gerçekleştirilmektedir. İmzanın şekilsel özellikleri arasında imza yüksekliği, resim alanı, tam genişlik, tam yükseklik, yatay ve düşey iz düşümü tepeleri, imzanın yatay ve düşey merkezi, genel ve yerel meyil açıları,

taban çizgisi kayması, köşe noktası sayısı, kesişme noktası sayısı olarak verilmektedir [50]. Bu imza doğrulama çeşidinde imzalara ait öznitelik sayısı az olduğu için çevrimiçi imzalara oranla daha düşük bir başarıya sahiplerdir. Öznitelik sayıları az olmasından dolayı bu tür doğrulama sistemlerinde çeşitli zorluklarla karşılaşmaktadır. Çevrimdışı imza doğrulama sistemlerinde yaşanan zorluklar, yüksek kişisel değişkenlik, sınırlı sayıda eğitim örneği, öğrenme aşamasında vasıflı sahte imzalara erişilememesi ve sahtekârların orijinal örnekleri daha kolay taklit etmeleri [25] olarak sıralanmaktadır. Bu zorlukları ele alan daha önceki çalışmalarda, daha iyi öznitelik çıkarımı için yeni yöntemler kullanmak, veri kümelerini genişletmek için sınırlı sayıdaki sınıflandırmayı iyileştirmek ve model toplulukları oluşturmak gibi çözüm önerileri getirilmiştir [21].



Şekil 3.1. İnternette seçilmiş çevrimdışı imza örnekleri

Dinamik (çevrimiçi, online) imza modelinde, kullanıcı bir tablete ya da tablet benzeri elektronik bir cihaza imza atar. Bu cihaz imza özelliklerini gerçek zamanlı olarak örnekler. Dinamik imza doğrulama çok caziptir çünkü davranışsal biyometrinin özelliklerinin kaydedilebilmesine izin verir. Bu teknik, çevrimiçi (online) teknik olarak da bilinir [51, 52]. Şekil 3.2.'de dinamik bir imza örneği ve bunun için kullanılan imza cihazı görülmektedir.



Şekil 3.2. Çevrimiçi imza örneği ve imza cihazı

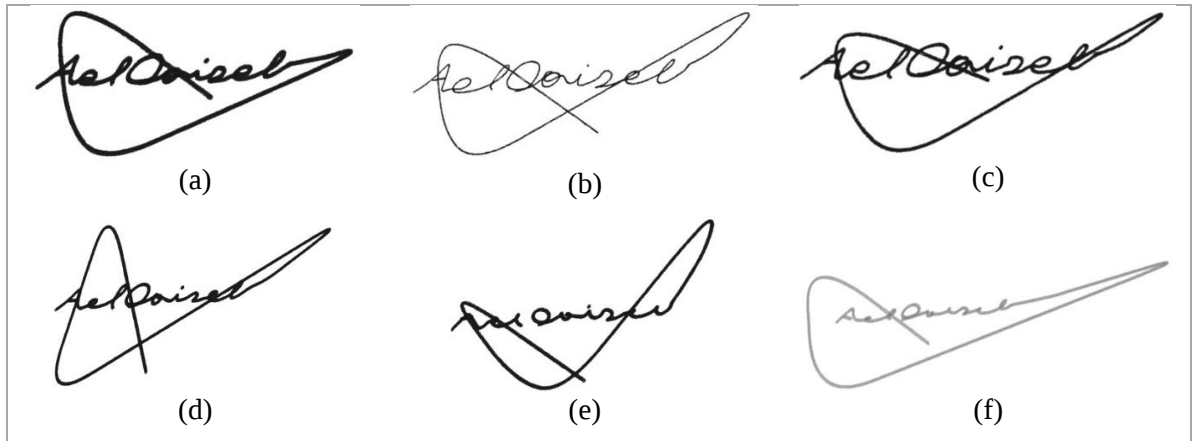
Dinamik imza statik imza niteliklerine ek olarak, imza işlemi sırasında değişen dinamik özellikler hakkında da bilgi içerir; örneğin kalem hızı, ivmesi veya basıncı gibi özellikler ek olarak kaydedilebilir. Dinamik özellikler görünmezdir ve imzayı kişiye daha benzersiz ve daha karakteristik hale getirir. Çeşitli çevrimiçi yöntemler, lokal ve global imza özelliklerine bölünebilen 40 farklı özelliğe (örn., konum, hız, ivme, kalem basıncı, kalem açısı, vb.) sahiptir [53]. Çizelge 3.1’de çevrimiçi ve çevrimdışı imzalara ait öznitelikler gösterilmiştir. "Lokal özellikler", imza izinin (örn. Nokta konumu, yerel hız, kalem basıncı, vb.) tek bir örneğindeki (nokta) özelliklere atıfta bulunurken "genel özellikler" tüm imza izini (yani, iz uzunluğu, imza alanı, vb.) tanımlar. Dijital imzaları yakalamak için özel olarak tasarlanmış bir tablet, doğrudan bir bilgisayara veri gönderebilir ve dinamik imza çalışan sınıflandırma algoritmalarının oluşturulmasına yardımcı olur.

Çizelge 3.1. Çevrimdışı (statik) ve çevrimiçi (dinamik) imza öznitelikleri

İmza Öznitelikleri			
Çevrimdışı		Çevrimiçi	
Ö1	En Boy Oranı	Ö1 - Ö13	Çevrimdışı imzanın tüm özellikleri
Ö2	Çıkış Oranı	Ö14	Yatay koordinatı $x(t)$
Ö3	Dikey ve Yatay Merkez	Ö15	Dikey koordinatı $y(t)$
Ö4	Taban çizgisi kayması	Ö16	Kalemin basıncı $p(t)$
Ö5	Moment Özellikleri	Ö17	Zaman damgası
Ö6	Dikey ve Yatay Projeksiyon Zirveleri	Ö18	Kalemin aldığı mesafe $r(t) = \sqrt{x^2(t) - y^2(t)}$
Ö7	Kenar ve Köşegen Nokta Sayısı	Ö19	Kalemin yatay hızı $v_x(t)$
Ö8	Kapalı döngü sayısı	Ö20	Kalemin dikey hızı $v_y(t)$
Ö9	Maksimum Dikey ve Yatay Projeksiyon	Ö21	Kalemin bağlı hızı, $v(t) = \sqrt{v_x^2(t) - v_y^2(t)}$
Ö10	Izgara Özellikleri	Ö22	Kalemin zamana göre hızları $v(t), v_r(t)$
Ö11	Doku Özellikleri	Ö23	Kalemin yatay ivmesi $a_x(t)$
Ö12	Yükseklik	Ö24	Kalemin dikey ivmesi $a_y(t)$
Ö13	Genişlik	Ö25	Kalemin bağlı ivmesi $a(t) = \sqrt{a_x^2(t) - a_y^2(t)}$
		Ö26	Kalemin aşağı iniş zamanı $P_d(t)$
		Ö27	Kalemin yukarı çıkış zamanı $P_u(t)$

İmzaların öznitelikleri imzaya ait karakteristiklerin ortaya çıkarılmasına yardımcı olmaktadır. Böylece imza karakteristiğine bakılarak imzanın gerçek ya da sahte imza olup olmadığı belirlenebilir. Kişilerin attığı imzalar içinde bulundukları koşullara bağlı olarak değişiklik gösterebilir ve bu nedenle de imza tespiti sırasında gerçek imza olmasına rağmen sahte imza olarak ta değerlendirilme ihtimali vardır. Yine aynı şekilde sahtekârlar tarafından atılmış bir imza çok iyi taklit edildiği için gerçek imza olarak değerlendirilebilir. Sahte ve gerçek imzaların güçlü bir şekilde sınıflandırılabilmesi için imzaya ait karakteristik özelliklerin iyi tespit edilmesi büyük önem taşımaktadır.

Sahte imzalar literatürde üç farklı kategoride değerlendirilmektedir. Bunlar rastgele sahteler (random forgery), basit sahteler (simple forgery) ve yetenekli sahteler (skilled forgery) dir [54, 55]. Şekil 3.3'de gerçek ve sahte imzalara örnekler gösterilmiştir.



Şekil 3.3. Gerçek ve sahte imza örnekleri, ilk satır (a, b, c) gerçek, ikinci satır (d, e, f) sahte

Rastgele sahte olarak adlandırılan sahte imzalar, sahtekarların imza sahibine ve imzaya ait şekil, kişi adı gibi hiçbir bilgiye sahip olmadan geliş güzel bir biçimde attıkları imzalarlardır. Bu imzaların gözle tespit işlemi kolaydır. Kişinin gerçek imzasıyla karşılaştırılması durumunda kolayca sahte olduğu tespit edilebilir.

Basit sahte imzalar, imzayı taklit edenin imza sahibine ait sadece isim bilgisine sahiptir. Çok fazla karakteristik özellik içermese bile bu tür imzalar bazı durumlarda kişinin gerçek imzasına çok benzemektedir ve sahte olduğunu belirlemek için ek analiz yöntemlerine başvurmak gerekebilir. Gerçekinden ayırt etmesi her zaman kolay olmayabilir.

Yetenekli sahte imzalar ise, imza sahibinin adı, imzanın şekli ve desen yapısı gibi birçok karakteristik özellik bilgisine sahip kişiler tarafından atılan imzalardır. Sahtekârlar bu gibi durumlarda gerçek imza örneklerine ve bu imzaları öğrenebilmek için yeterince zamana sahiptirler. Bu nedenle yetenekli sahte imzalar gerçek imzalar aşırı derecede benzer ve yakın karakteristik özellikler barındırırlar. Bu imzaların tespiti oldukça zordur ve imza doğrulama sistemleri gibi sofistike yapılara ve bu alanda uzmanlaşmış kişilere ihtiyaç vardır.

3.1. İmza Doğrulama

İmza Doğrulama bir kişinin attığı imzanın gerçek imzası olup olmadığını tespit etme işlemine verilen addır. İmza doğrulama işlemi, kişilerin daha önceden gerçek olduğu bilinen imzasının veri tabanına kaydedilmiş formları ile yeni atılan imzanın karşılaştırılması ve bu karşılaştırma neticesinde imzanın gerçek ya da sahte olduğunun belirlenmesi şeklinde gerçekleşir. İmza doğrulama yöntemlerinin temel amacı imzaları doğru bir biçimde gerçek ve sahte imzalar olmak üzere iki sınıfa ayırmaktır.

İmza yüzyıllardır kişilere büyük sorumluluklar yükleyen ve en sık kullanılan biyometrik doğrulama biçimi olmasına rağmen, parmak izi, iris, avuç içi izi ve yüz gibi diğer biyometrik doğrulama yöntemlerine göre daha az karakteristik özellik barındırmaktadır. Hiçbir kişi daha önce attığı bir imzayı aynı şekilde tekrarlayamamaktadır. İmza kişinin o anki oturma şekline, kullandığı kalemin kalitesine, imzanın atıldığı yüzeye, kişinin kalemi tutuş şekline, kişinin yaşına ve hatta kişinin o anki psikolojisine bağlı olarak farklılık göstermektedir. Buna doğal çeşitlilik veya sınıf içi çeşitlilik (intra-class variability) denilmektedir [1]. Sınıf içi çeşitliliğinin yüksek olması sebebiyle imza doğrulama yöntemlerinin başarısı diğer biyometrik doğrulama sistemlerine göre düşük çıkmaktadır.

Kullanılan imza doğrulama yöntemlerinin Yazara Bağımlı (Writer Dependent - WD) ve Yazardan Bağımsız (Writer Independent - WI) olmak üzere iki farklı şekilde tasarlandığı görülmektedir. Yazara Bağımlı sistemlerde imza doğrulaması gerçekleştirilirken veri tabanından bir kişiye ait imzalar seçilir ve sistem o kişinin imzaları ile çalıştırılır. Böylelikle kişinin attığı ve gerçek olan farklı imzaların öznelilikleri ve ortak yanları araştırılarak bir doğruluk tespiti yapılır. Sistem her kişi için sadece o kişinin imzaları ile eğitilir bu nedenle de kişiye ait spesifik özellikler sistem tarafından öğrenilebilmektedir. Yazardan Bağımsız

sistemlerde ise veri tabanında bulunan tüm yazarlara ait gerçek imzalar aynı anda kullanılarak doğrulama sistemi çalıştırılmaktadır. Buradaki amaç tüm bireylere ait gerçek imzaların ortak özelliklerinin tespitidir. İmza gibi yüksek doğal çeşitliliğe sahip biyometrik öğelerde farklı kişilerin gerçek imzaları birbirleri arasında çok az benzerliğe sahip olacağından sadece imzalara ait genel özellikler öğrenilmektedir. WD sistemler kişisel özellikleri dikkate alırken, WI sistemler genel özellikleri dikkate almaktadır. Bu nedenle yapılan çalışmalarda WD sistemlerin daha başarılı sonuçlar elde ettiği görülmektedir. WI sistemlerin avantajı ise imza verilerinin ve çeşitliliğinin fazla olmasıdır. Zira son yıllarda oldukça fazla tercih edilen sınıflandırma yöntemlerinden biri olan derin öğrenme yöntemlerinin başarısı için veri kümesinin sayısı ve çeşitliliği önem taşımaktadır. Bazı çalışmalarda bu iki yaklaşımın birleştirilerek karma bir yaklaşım elde edildiği görülmektedir.

Literatür çalışmaları incelendiğinde çeşitli imza doğrulama yöntemlerinin olduğu görülmektedir. İmza doğrulama alanında en çok kullanılan yöntemlerin HMM [8, 9, 18, 19, 38, 56, 57], SVM [10, 38, 57–59], DTW [11, 12], PCA [13], bulanık sistemler (Fuzzy Methods) [14], PNN [3], MLP [40], gibi yöntemleri olduğu görülmektedir. DMML [15, 16] gibi metrik sistemleri imza doğrulama için kullanılan diğer bir yöntemdir. Son yıllarda imza doğrulaması için en çok kullanılan yöntemler ise derin öğrenme yöntemleri olmuştur [1, 17, 23, 24, 34, 35, 39, 40, 42, 47].

3.1.1. Sistemlerin performans karşılaştırmasında kullanılan metrikler

Gerçekleştirilen farklı imza doğrulama sistemlerinin performanslarının ölçülmesinde en çok kullanılan yöntem karmaşıklık matrisleri (Confusion matrix) dir. Hata matrisi olarak ta Türkçeleştirilmektedir. Makine öğrenmesi sistemlerinin performans kıyaslaması için birçok alanda sıklıkla kullanılırlar.

Sistem 2X2'lik bir matristen oluşmaktadır. Matrisin satırları sınıflara ait tahmin edilen örnekleri temsil ederken, sütunlar ise her bir sınıfın olması gereken gerçek örneklerini temsil eder, satır ve sütunların durumları tam tersi biçimde de tasarlanabilir [60]. Çizelge 3.2'de imza doğrulama alanında kullanılan örnek bir karmaşıklık matrisi görülmektedir. Çizelgede gerçek ve sahte imzalar ve bu imzalara ait sınıflandırıcının tahminleri bulunmaktadır. Toplamda 443 gerçek ve 150 sahte imza örneğine ait sonuçları içermektedir. Karmaşıklık

matrisi üzerinde, gerçekte gerçek yani orijinal imza olduğu bilinen imzalar, sistem tarafından da gerçek olarak tahmin ediliyorsa bu imzalara “Doğru Tahmin Edilen Gerçekler (True Positives - TP)”, sistem tarafından sahte olarak tahmin yapılıyorsa (yani sistem imzaları doğru tahmin edememişse gerçeklere sahte demişse) “Yanlış Tahmin Edilen Sahteler (False Negatives - FN)” olarak gösterilir. Aynı şekilde gerçekte sahte olan imzalar sistem tarafından da sahte olarak tahmin edildiyse “Doğru Tahmin Edilen Sahteler (True Negatives - TN)”, sistem tarafından gerçek olarak tahmin edildiyse (sahte imzalara sistem gerçek dediyse yani yanlış tahmin yaptıysa) “Yanlış Tahmin Edilen Gerçekler (False Positives - FP)” olarak karmaşıklık matrisinde gösterilir. TP + FN gerçek imza sayısını verirken, TN + FP sahte imza sayısını vermektedir.

Çizelge 3.2. İmza örneklerine ait karmaşıklık matrisi

		Gerçek Sınıflar	
		Gerçek	Sahte
Tahmin Edilen Sınıflar	Gerçek	Doğru Tahmin Edilen Gerçekler (True Positives - TP) 441	Yanlış Tahmin Edilen Gerçekler (False Positives - FP) 1
	Sahte	Yanlış Tahmin Edilen Sahteler (False Negatives - FN) 2	Doğru Tahmin Edilen Sahteler (True Negatives - TN) 149

Karmaşıklık matrisindeki verileri daha doğru bir şekilde yorumlayabilmek için Doğruluk (Accuracy (ACC)), Hata Oranı (Error Rate / Misclassification Rate), Doğru Pozitif Oranı/Duyarlılık (True Positive Rate (TPR) – Sensivity – Recall), Hassasiyet (Positive Predictive Value (PPV) or precision), False Negatif Rate (FNR) False Rejected Rate (FRR), (False Positive Rate (FPR)- False Accepted Rate (FAR)), Eşit Hata Oranı (Equal Error Rate (EER)) ve F1 puanı (F1-score) oranları hesaplanmaktadır.

Doğruluk (Acuracy - ACC), doğrulama sistemimizin tüm imzalar içinden kaç tane imza örneğini doğru tahmin ettiğinin oranını verir ve “Eş.3.1” ‘de verilen denklem ile elde edilir.

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \quad (3.1)$$

Hata Oranı (Error Rate / Misclassification Rate), doğrulama sistemimizin tüm imzalar içinden kaç tane imza örneğini yanlış tahmin ettiğinin oranını verir ve “Eş.3.2” ‘de verilen denklem ile elde edilir. Ayrıca $1 - \text{ACC}$ şeklinde de bulunabilir.

$$\text{ER} = \frac{FP + FN}{TP + FP + TN + FN} \quad (3.2)$$

Doğru Pozitif Oranı/Duyarlılık (True Positive Rate (TPR) – Sensivity – Recall), doğru olarak tahmin edilen gerçek imza örneklerinin toplam gerçek imza sayısına olan oranını verir ve “Eş.3.3” ‘de verilen denklem ile elde edilir.

$$\text{TPR} = \frac{TP}{TP + FN} \quad (3.3)$$

Hassasiyet (Positive Predictive Value (PPV) or precision), doğru olarak tahmin edilen gerçek imza örneklerinin, sistem tarafından gerçek denilen imza sayısına oranını verir. Yani sistemin gerçek dediği imza örneklerinin içinde bulunan doğru tahmin edilmiş gerçek imza oranını verir ve “Eş.3.4” ‘de verilen denklem ile elde edilir.

$$\text{TPV} = \frac{TP}{TP + FP} \quad (3.4)$$

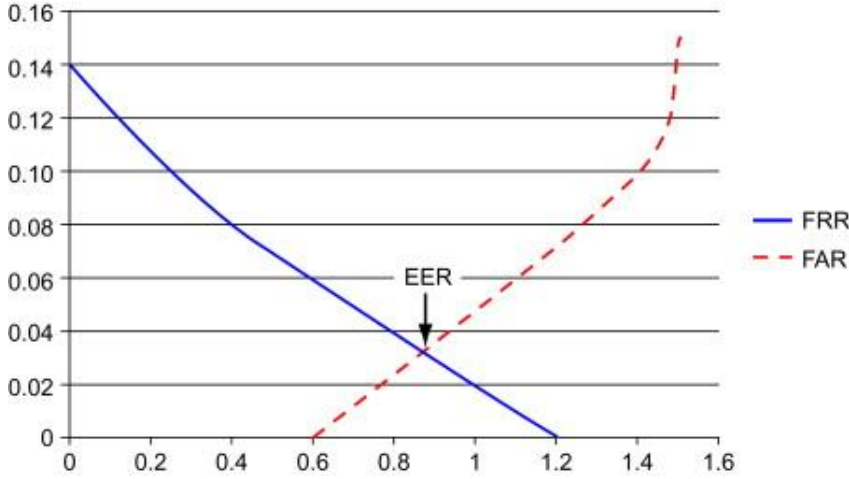
Yanlış Reddetme Oranı (False Negative Rate (FNR) - False Rejected Rate (FRR)), gerçek olmasına rağmen sistem tarafından sahte olarak sınıflandırılmış imzaların tüm gerçek imzalara oranını vermektedir ve “Eş.3.5” ‘de verilen denklem ile elde edilir.

$$\text{FNR} = \text{FRR} = \frac{FN}{TP + FN} = 1 - \text{TPR} \quad (3.5)$$

Yanlış Kabulme Oranı (False Positives Rate (FPR) - False Accepted Rate (FAR)), sahte olmasına rağmen sistem tarafından gerçek olarak sınıflandırılmış imzaların tüm sahte imzalara oranını vermektedir ve “Eş.3.6” ‘de verilen denklem ile elde edilir.

$$\text{FPR} = \text{FAR} = \frac{FP}{TN + FP} = 1 - \text{TNR} \quad (3.6)$$

Eşit Hata Oranı (Equal Error Rate (EER)), FPR ya da FAR ve FNR ya da FRR oranlarına ait grafikleri üst üste çizdirdiğimizde iki grafiğin kesiştiği nokta bize EER değerini vermektedir. Şekil 3.4.’de EER için bir örnek gösterilmektedir.



Şekil 3.4. Equal error rate için örnek grafik

Ortalama hata oranı (Average Error Rate (AER)), FAR ve FRR metriklerinin aritmetik ortalaması alınarak elde edilir. “Eş. 3.7” ‘de verilen denklem ile elde edilir.

$$AER = \frac{FAR + FRR}{2} \quad (3.7)$$

F1 puanı (F1-score), F1 ölçümü olarak ta adlandırılmaktadır. Hassasiyet (PPV = Precision) ya da yüksek geri çağırma (Recall = TPR = Sensivity) (ya da tam tersi), iki modeli karşılaştırmak zordur. Bu nedenle iki modeli karşılaştırmak için F-skorunu kullanılır. F-skoru hassasiyet ve geri çağırmayı aynı anda ölçmeye yardım eder. Aşırı değerleri daha fazla cezalandırarak için iki modelin aritmetik ortalaması yerine harmonik ortalamasını kullanır. “Eş.3.8” ‘de verilen denklem ile elde edilir.

$$F - \text{measure} = F1 \text{ Score} = \frac{2 * Recall * Precision}{Recall + Precision} \quad (3.8)$$

3.2. Öznitelik Çıkarım Yöntemleri

Öznitelik sınıflandırma problemlerinde sınıfları birbirinden ayıran en temel özelliktir. Öznitelik çıkarımı sınıflara ait karakteristik özelliklerin tespit edilmesi işlemidir. Birçok

bilgisayarlı görü alanında olduğu gibi imza doğrulama alanında da doğru özniteliklerin tespit edilmesi hayati öneme sahiptir. En iyi öznitelikler belirlenmesi istenilen sınıfları en iyi ve kesin bir biçimde tespit ederek ayırmaya yardımcı olan özniteliklerdir. Öznitelik çıkarım yöntemleri Genel (Global) ve Yerel (Local) olmak üzere temelde iki ana bölüme ayrılmaktadır. Bu tez kapsamında çevrimdışı imza doğrulama üzerinde çalışıldığı için ele alınan öznitelikler ve öznitelik çıkarım yöntemleri çevrimdışı imzalar için kullanılanlar ile sınırlandırılmıştır.

Genel öznitelik çıkarım yöntemleri, görüntüyü bir bütün olarak ele alır ve öznitelikleri görüntünün tamamına göre belirler. Genel öznitelikler imzanın tamamına ait genel özellikleri ele aldığı için yetenekli sahte imzaları (Skilled Forgeries) tespit etmede yetersiz kalırlar. Rastgele seçilmiş sahte imzaların (Random Forgeries) tespitinde iyi performansa sahip oldukları literatürde bildirilmiştir [59]. Genel özniteliklerin avantajlarından birisi görüntülerdeki gürültülerden çok az etkileniyor olmalarıdır [61]. Bu tür öznitelikler diğer özniteliklerle birleştirilerek daha iyi sonuçlar elde edilebilir. Literatürde imza doğrulama alanında en çok kullanılan genel öznitelikler imzanın Genişliği (Width), Yüksekliği (Height), En boy oranı (Aspect ratio), Yatay iz düşüm (Horizontal projection) , Dikey iz düşüm (Vertical projection), Yatay iz düşümünün en büyük ve en küçük değerleri (Maximum and minimum black pixels number of Horizontal projection) , Dikey iz düşümünün en büyük ve en küçük değerleri (Maximum and minimum black pixels number of vertical projection), Siyah piksel sayısı (Area of black pixels), Normalize edilmiş siyah piksel sayısı (Normalized area of black pixels), Ağırlık merkezleri (Center of gravity) olarak sıralanabilir. Bu öznitelik çıkarım yöntemlerine ek olarak bazı matematiksel dönüşüm yöntemleri, geometrik yöntemler (yukarıda bahsedilen öznitelikler) ve grafometrik yöntemler de çevrimdışı imzalar için kullanılmaktadır. Bunlara örnek olarak Wavelet dönüşümü, Radon dönüşümü, Yönsel (Direction-Based), Eğrisel (Curvature-Based), Grafometrik/Yazımsal (Graphometric-Based), Gölge kodu (Shadow Code -Based) vb. sayılabilir[4]. Son yıllarda derin öğrenme algoritmaları ile çıkartılan öznitelikler de imza doğrulama işleminde kullanılmaya başlanmıştır. Bu yöntemler otomatik öznitelik çıkarım yöntemleri olarak adlandırılır.

Yerel öznitelik çıkarım yöntemleri ise görüntüyü parçalar halinde işler ve her parçadan çıkarılan öznitelikler sayesinde öznitelik vektörü oluşturulmasını içerir. Öznitelikler görüntünün sınırlı bir bölümüne ya da parçalarına uygulanır. Görüntüler dikdörtgen ya da

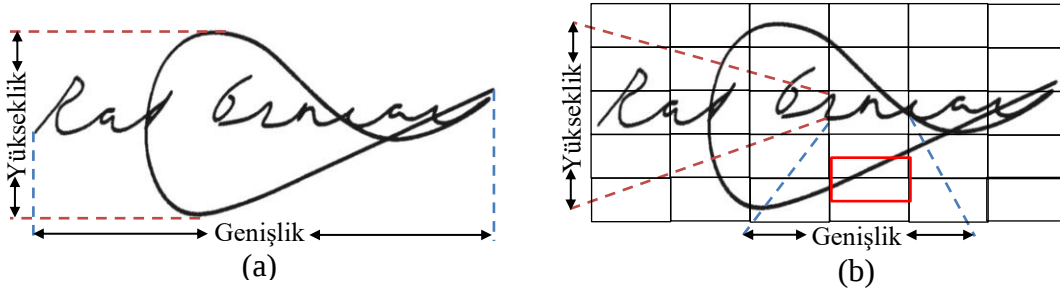
dairesel şekillerde ızgaralara bölünür ve bu ızgaraların her bir parçası ayrı ayrı değerlendirilir. Yerel özellikler sınıflandırmada daha doğru sonuçlar verirler ancak gürültülerden çok fazla etkilenirler ve her parçaya ayrı ayrı öznitelik çıkarım yöntemleri uygulanacağından hesaplama maliyetleri daha yüksektir. Genel öznitelik çıkarım yöntemlerinin hepsi yerel öznitelik çıkarımı için de kullanılabilir [61 - 63].

3.2.1. İmza genişliği (Width)

Görüntüsü elde edilmiş imzanın piksel cinsinden genişliğini ifade etmektedir. Genel öznitelik çıkarımında imzanın tamamına ait genişlik hesaplanırken yerel öznitelik çıkarımında her bir bölüm içerisinde kalan alanın genişliği hesaplanır.

3.2.2. İmza yüksekliği (Height)

Görüntüsü elde edilmiş imzanın piksel cinsinden yüksekliğini ifade etmektedir. Genel öznitelik çıkarımında imzanın tamamına ait yükseklik hesaplanırken yerel öznitelik çıkarımında her bir bölüm içerisinde kalan alanın yüksekliği hesaplanır. Şekil 3.5.'de imzaya ait örnek yükseklik ve genişlik gösterilmiştir.



Şekil 3.5. İmzanın (a) genel ve (b) yerel ölçekli genişlik ve yükseklik ölçümleri

3.2.3. En boy oranı (Aspect ratio)

İmza genişliğinin yüksekliğine olan oranı alınmak suretiyle hesaplanmaktadır. İmza genişlik ve yükseklikleri imzanın atıldığı şartlara göre değişiklik gösterebilir ancak kişinin imzayı atarken kullandığı en boy oranı aynı kalabilmektedir. Bu nedenle en boy oranı yükseklik ve genişliğe göre daha ayırt edici bir özniteliktir.

3.2.4. Yatay iz düşüm (Horizontal projection)

Yatay iz düşüm görüntüye ait yatay histogramdan oluşmaktadır. İmza görüntüsüne ait yatay eksenindeki her bir piksel sütununun sahip olduğu imza pikseli sayısından oluşur. “Eş.3.9” ‘de verilen denkleme göre hesaplanabilir [61]. n değeri görüntünün sütun sayısını ifade eder.

$$P_h[y] = \sum_{x=1}^n siyah\ piksel(x, y) \quad (3.9)$$

3.2.5. Dikey iz düşüm (Vertical projection)

Dikey iz düşüm görüntüye ait dikey histogramdan oluşmaktadır. İmza görüntüsüne ait dikey eksenindeki her bir piksel satırının sahip olduğu imza pikseli sayısından oluşur. “Eş.3.10” ‘da verilen denkleme göre hesaplanabilir [61]. n değeri görüntünün satır sayısını ifade eder.

$$P_v[x] = \sum_{y=1}^n siyah\ piksel(x, y) \quad (3.10)$$

3.2.6. Yatay iz düşümünün en büyük ve en küçük değerleri (Maximum and minimum black pixels number of horizontal projection)

Yatay iz düşüm vektöründeki en büyük ve en küçük değerlerden oluşmaktadır. “Eş.3.11” ‘da verilen denkleme göre hesaplanabilir [61].

$$MaxP_h = Max(P_h[x]), \quad MinP_h = Min(P_h[x]), \quad (3.11)$$

3.2.7. Dikey iz düşümünün en büyük ve en küçük değerleri (Maximum and minimum black pixels number of vertical projection)

Dikey iz düşüm vektöründeki en büyük ve en küçük değerlerden oluşmaktadır. İsmail ve Gad dikey iz düşümün maksimum noktasını genel dayanak noktası (Global baseline - BSL) olarak tanımlamışlardır. “Eş.3.12” ‘de verilen denkleme göre hesaplanabilir [61].

$$MaxP_v = Max(P_v[y]), \quad MinP_v = Min(P_v[y]), \quad (3.12)$$

3.2.8. Siyah piksel sayısı (Area of black pixels)

Görüntüsü elde edilmiş imzanın barındırdığı toplam piksel sayısını ifade etmektedir. Genel öznitelik çıkarımında imzanın tamamına ait pikseller sayılırken yerel öznitelik çıkarımında her bir bölüm içerisinde kalan pikseller sayılır.

3.2.9. Normalize edilmiş siyah piksel sayısı (Normalized area of black pixel)

Toplam siyah piksel sayısının görüntünün alanına (genişlik * yükseklik) bölümü sonucunda elde edilir [63].

3.2.10. Ağırlık merkezleri (Center of gravity)

Görüntünün ağırlık merkezinin yatay ve dikey eksenlerdeki koordinatlarının bulunabilmesi için, görüntüdeki tüm siyah piksellerin x ve y koordinatları ayrı ayrı toplanır. Elde edilen x toplam ve y toplam değerleri toplam siyah piksel sayısına bölünerek ağırlık merkezinin yataydaki koordinatı olan C_h ve dikeydeki koordinatı olan C_v hesaplanır. Yerel öznitelik çıkarımında ise işlem sadece istenilen bölgeye uygulanır. “Eş.3.13” ‘de verilen denkleme göre hesaplanabilir. n değeri görüntünün genişliğini, m değeri görüntünün yüksekliğini ifade eder.

$$C_h = \frac{\sum_{x=1}^n \sum_{y=1}^m x_Koordinatı(x,y)}{\text{toplam piksel sayısı}}, \quad C_v = \frac{\sum_{x=1}^n \sum_{y=1}^m y_Koordinatı(x,y)}{\text{toplam piksel sayısı}} \quad (3.13)$$

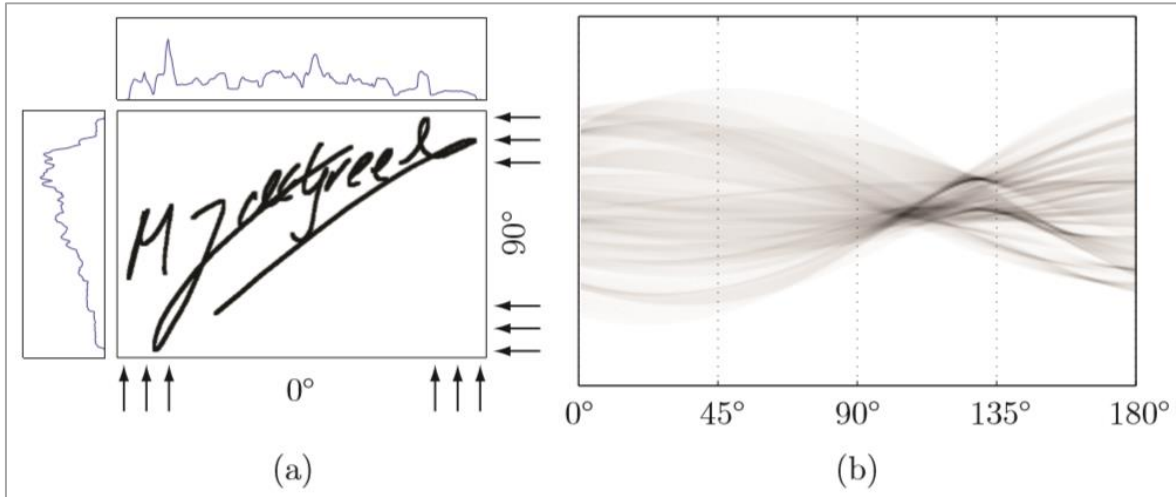
3.2.11. Wavelet dönüşümü

Dalgacık dönüşümü, zaman-frekans dönüşümlerinin popülerlerinden biridir. Ayrık dalgacık dönüşümü (Discrete Wavelet Transform - DWT), benzer hızlı Fourier dönüşümü (Fourier Transform - FT) ($O(N \log N)$ süresi) ile karşılaştırıldığında, nispeten daha az karmaşıklık ($O(N)$) zamanına sahiptir. Ayrıca Dalgacık dönüşümü, önemli yerel bilgilerin çoğunu tutarken imza bilgilerini daha kompakt bir forma dönüştürür. Farklı frekanslarda ve çözünürlüklerde sinyali analiz etmek için çoklu çözünürlük tekniği kullanır. FT'den farklı olarak DWT, yerel bilgileri hem zaman hem de frekans düzeyinde tutarak iyi bir yerel özellik belirleyicisidir.

Bu özellik, ani tepe noktaları (peak) veya duruşlar gibi önemli bilgileri kaybetmeden sinyalin daha doğru modellenmesini sağlar [36, 64].

3.2.12. Radon dönüşümü

Radon dönüşümü, bir fonksiyonun düz çizgiler üzerindeki integrallerinden oluşan bir integral dönüşümüdür. Ayrık Radon Dönüşümü (Discrete Radon Transform – DRT) ile elde edilen görüntünün her sütunu orijinal görüntünün belirli bir açıdaki iz düşümünü (projection) temsil eder. Çevrimdışı imza doğrulamasına uygulanır. DRT, gri tonlamalı bir görüntüye dönüştürüldüğünde, sinogram olarak bilinir. Tipik bir imza ve sinogramı Şekil 3.6'te gösterilmiştir [36, 65].



Şekil 3.6. Ayrık radon dönüşümü uygulanmış imza, (a) imzaya ait yatay (0°) ve dikey (90°) iz düşümleri (projection), (b) imzadan elde edilmiş sinogram görüntüsü

3.2.13. Yönsel (Direction-based) öznitelikler

Yönsel özellikler imza iskeletindeki her bir siyah pikselin komşuluklarına bakılarak yönünün yukarı, aşağı, sağ ve sol olmak üzere tespit edilmesidir. Yönsel özellikler literatürde farklı şekillerde kullanılmıştır. Bazı araştırmacılar yerel yönsel özellikleri (Local Directional Pattern - LDP) belirlemek için Kirsch maskeleri olarak bilinen 8 adet 3X3 lük maske kullanmıştır. Bu maske ile her bir pikselin 8 komşuluğuna bakılarak yönsel bir değer hesaplanır. Maskelere göre her yön eşit ağırlık değerinde değildir, fazla tepkiye sahip yönler daha yüksek puana sahiptir. Böylelikle en çok tepkiyi veren yön değerleri elde edilmeye

çalışılır [66, 67]. Diğer bir araştırmacı ise belirlediği piksellerin yönsel açısına göre özellik çıkarımı yapmıştır [55].

3.2.14. Eğrisel (Curvature-based) öznitelikler

Eğrilik (Curvature) fonksiyonu ile elde edilir. Eğrilik fonksiyonu temel olarak, bir kenarın, şekle bağlı her bir noktada ne kadar büküldüğünü açıklar ve kenarın uzunluğu boyunca eğrilikteki değişim oranı olarak tanımlanır. Sezgisel olarak eğrilik, bir eğrinin düz bir çizgiden sapma miktarını tanımlayan bir ölçüdür. Bir eğrinin eğimi, “Eş.3.14” ‘te verilen denkleme göre elde edilir [68].

$$\theta(s) = \tan^{-1} \frac{\partial y}{\partial x}, \quad K(s) = \frac{\partial \theta(s)}{\partial s} \quad (3.14)$$

$\theta(s)$ eğrideki değişimi ve s uzunluğu ifade etmektedir. $K(s)$ eğrilik fonksiyonudur. Matematiksel olarak, parametrik düzlem eğrisinin birinci ve ikinci türevi ile hesaplanır. Ayrık görüntüler veya eğriler için eğriliği hesaplamada ana yol, eğriyi parametrik bir denklem olarak yaklaşık olarak bulmak ve birinci ve ikinci türevlerini kullanmaktır, ancak çevrimdışı imzalarda geçici bilgi eksikliği sebebiyle, bir parametrik denklemin yerleştirilmesi pratik değildir [69].

3.2.15. Grafometrik/yazımsal (Graphometric-based) öznitelikler

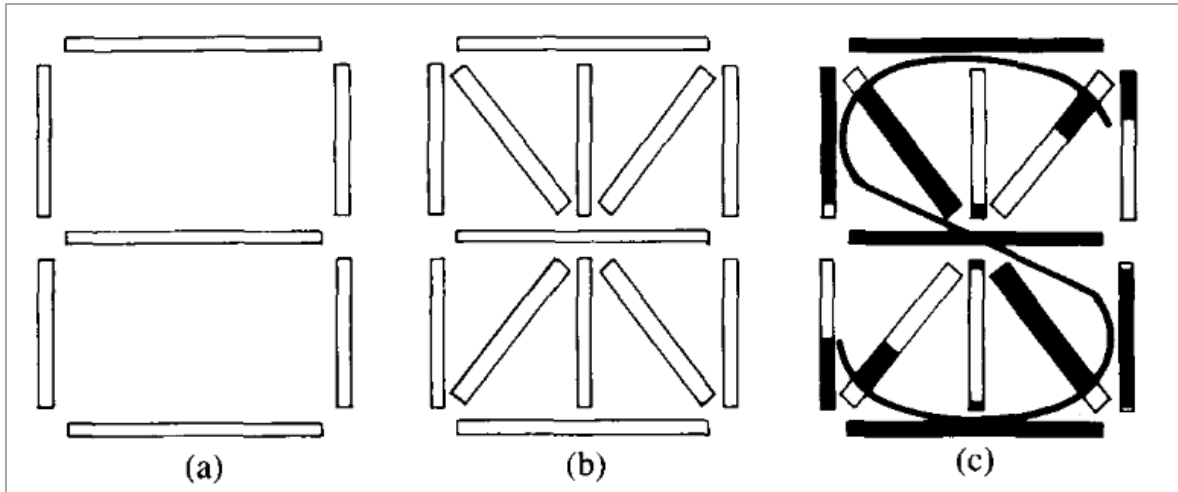
Grafoloji yazı bilimi anlamına gelmektedir. Yazının biçimiyle ilgilenir. Yazıdaki harflerin bağlantı biçimi, çizgilerin biçimi büyüklüğü, genişliği, eğimi, yoğunluğu, kalem vuruşları, başlangıç ve bitişler, ve kelimeler arası boşluklar gibi özellikler grafolojinin ilgi alanına girmektedir [70].

Grafometri yazının istatistiksel temellerini inceleyerek analizini gerçekleştirip, anlamsal bir veri elde etmeyi amaçlayan bilimsel araştırma alanıdır. Grafometride daha çok sayısal verilere dayanılarak araştırmalar incelemeler yapılmaktadır.

Graphometric özellikler, el yazısı veya imza tanıma sırasında adli tıp uzmanları tarafından kullanılabilen bireysel bir el yazısı stilinden gelen doğal özelliklerdir. Bunlar eğrilik ve diğerleri arasında basınç içerir [21, 71].

3.2.16. Gölge kodu (Shadow code - based) öznitelikleri

Gölge kodu öznitelik çıkarım yöntemi ilk olarak Burr [72] tarafından optik karakter tanıma için önerilmiştir ve daha sonra imza doğrulama alanı için gelişmiş gölge tanıma Sabourin ve ark. [73] tarafından önerilmiştir. Gölge kodunun çalışma mantığı yatay, dikey ve köşegen ızgaralar üzerine yansıtılan görüntü ile başlar. Bu görüntüye ait her bir piksel ve bu piksellerin her yöndeki (sağ, sol, yukarı ve aşağı) iz düşümleri hesaplanır. Tüm bu piksellerin üzerine düştüğü ızgara bölümleri belirlenir. Daha sonra ızgaralar üzerine düşen pikseller hesaplanır. Böylece her bir ızgara kolonundaki öznitelikler belirlenmiş olur. Örnek bir model Şekil 3.7’de [72] gösterilmiştir.

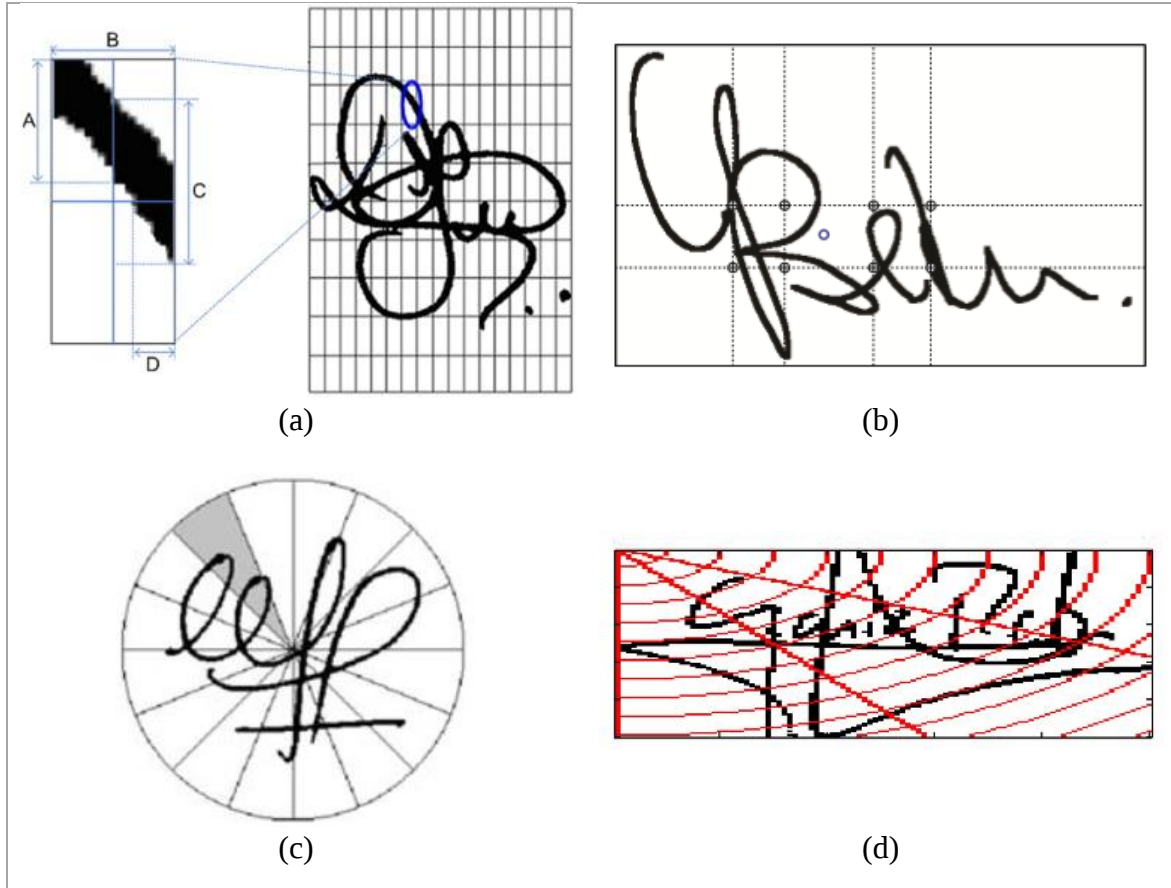


Şekil 3.7. Gölge kodu örneği, (a) 7 ızgaralı maske, (b) 13 ızgaralı maske, (c) maske üzerinde belirlenmiş gölge izleri.

3.2.17. Izgara tabanlı öznitelikler

İmza görüntülerinde yerel öznitelikleri daha iyi tespit edebilmek için ızgara sistemi kullanılmaktadır. Izgara sistemi dörtgen, esnek aralıklı dörtgen ya da kutupsal olarak tasarlanmaktadır. Yukarıda anlatılan öznitelik çıkarım yöntemleri ızgaralar ile bölümlere ayrılmış görüntünün her bir bölmesine ayrı ayrı uygulanarak yerel öznitelikler tespit edilir.

Böylece her bir görüntü için görüntünün her bölmesine ait öznitelik bilgilerini barındıran bir öznitelik vektörü elde edilmiş olur. Şekil 3.8’de ızgara çeşitlerine örnekler gösterilmektedir.



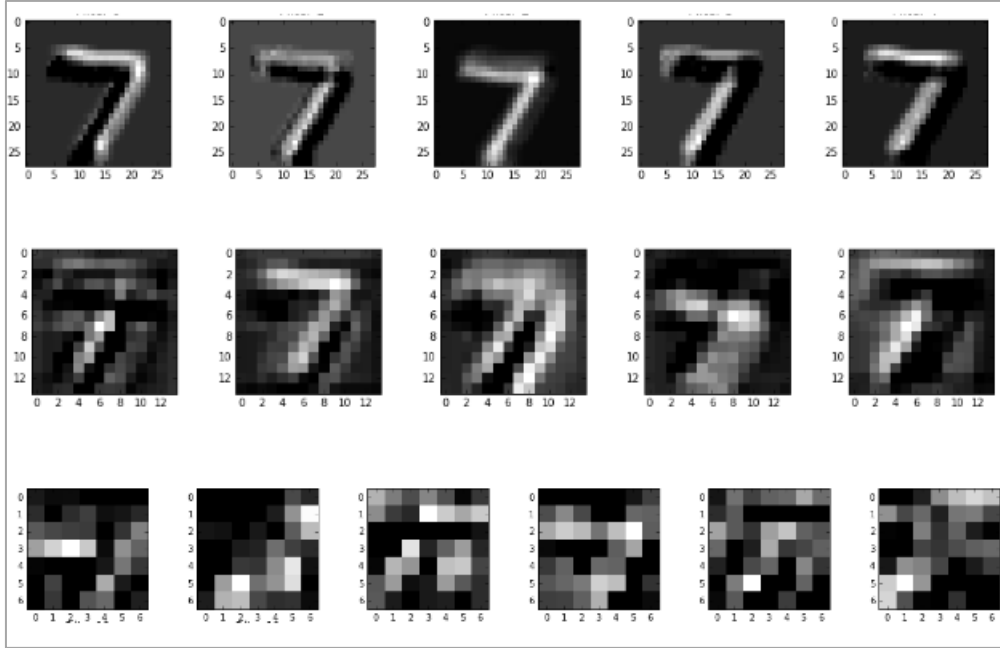
Şekil 3.8. Izgara çeşitleri (a) eşit aralıklı dikdörtgen ızgara [33], (b) esnek dikdörtgen ızgara [65], (c) ve (d) kutupsal (dairesel) ızgara [74, 75]

3.2.18. Derin öğrenmeyle öznitelik çıkarımı

Derin öğrenme yöntemleri son yıllarda birçok bilgisayarlı görü alanında başarılı bir şekilde uygulanmaya başlanmıştır. Özellikle de Evrimsel Sinir Ağları (Convolutional Neural Network - CNN) görüntü işleme alanında umut vadeden sonuçlar elde etmiştir. CNN iyi bir sınıflandırma modeli olduğu kadar aynı zamanda çok iyi bir öznitelik çıkarımı da gerçekleştirmektedir. CNN modelleri ile görüntüdeki genel ve yerel öznitelikler otomatik bir biçimde elde edilebilmektedir. Çevrimdışı imza doğrulama alanında da birçok çalışmada öznitelik çıkarım yöntemi olarak başarılı sonuçlar elde etmiştir [1, 45–47, 76–78].

CNN modelinde girdi olarak verilen görüntülere ait öznitelikler her bir katmanda bulunan maskeler sayesinde öğrenilmektedir. Geri yayılım algoritması sayesinde sürekli olarak daha

iyi öznitelikler elde edilene kadar maske değeri güncellenmektedir ve böylece otomatik olarak öznitelik çıkarımı gerçekleşmektedir. CNN ağlarında, ilk katmanlar yerel özniteliklere odaklanırken ilerleyen katmanlara inildikçe genel özniteliklere odaklanıldığı görülmektedir. Şekil 3.9'de CNN ile elde edilmiş örnek öznitelikler gösterilmiştir.



Şekil 3.9. CNN öznitelikleri, yukarıdan aşağıya sırasıyla genel ve yerel öznitelikler

4. PROBLEM VE ÇÖZÜM ÖNERİSİ

Çevrimdışı imza doğrulama elektronik bir cihaz tarafından taranmış imza görüntüsünün daha önceden kaydedilmiş ve kişinin gerçek imzası olduğu bilinen örnekler ile karşılaştırılarak gerçek ya da sahte olması durumunun tespit edilmesidir. Çevrimdışı imza doğrulaması alanında lineer algoritmalarından yapay zekâ yöntemlerine kadar birçok çalışma gerçekleştirilmiştir [36, 44, 57, 63, 69, 75]. Son yıllarda bu alandaki başarıyı en çok artıran yöntemlerse derin öğrenme yöntemleri olmuştur [1, 35, 45–47]. Ancak çevrimdışı imzanın karakteristik özellikleri ve derin öğrenme modellerinin doğası gereği bu alandaki başarının artırılmasının önünde çözülmesi gereken ciddi problemler bulunmaktadır. Bu problemler, sınıflandırma işlemi için sınırlı öznitelikçe sahip olması, derin öğrenme algoritmalarının başarılı bir şekilde eğitilebilmeleri için yetersiz veri kümesinin bulunması ve imzaların yüksek sınıf içi çeşitliliğe sahip olmasıdır.

Çevrimdışı imzalar imzanın sadece görüntüsüne ait verileri içerdiği için çok kısıtlı öznitelikler barındırmaktadır. Bu öznitelikler İMZA DOĞRULAMA VE ÖZNİTELİK ÇIKARIM YÖNTEMLERİ konusunun anlatıldığı 3. bölümünde ayrıntılı olarak verilmiştir. Bu kısıtlı sayıdaki öznitelikler kullanarak bir imzanın (özellikle de yetenekli sahtekârlar tarafından imzalanmış olanların) gerçek ya da sahte olduğunu tespit etmesi oldukça zor bir işlemdir. Çevrimdışı imza doğrulama yöntemlerinin başarısını etkileyen önemli etmenlerden bir tanesi sınırlı özniteliklerdir. Özellikle de elle hazırlanan öznitelik çıkarım yöntemleriyle elde edilen başarılar çok düşük düzeyde kalmaktadır. Buna karşılık derin öğrenme gibi öznitelik haritasını da eğitim sırasında kendi öğrenen algoritmaların başarısı daha yüksek olmaktadır [1, 35, 45–47].

Derin öğrenme algoritmalarının birçok alanda [79–87] başarılı olmasının en büyük nedenlerinden biri sınıflandırmada kullanılacak öznitelik haritalarını geri yayılım (back propagation) yöntemiyle otomatik olarak öğreniyor olmasıdır. Derin öğrenme algoritmaları bu öznitelik haritalarını öğrenirken, öğrenilmesi istenilen özelliği farklı açı ve şekillerden yansıtan birçok örneği kullanır ve değerlendirir. Eğitimde kullanılacak örnek sayısının çok olması eğitiminde başarısını doğrudan etkilemektedir. Örnek sayısı ve çeşitliliği düşük olan çalışmalarda derin öğrenme algoritmalarının yeterince öğrenemediği (underfitting) ya da aşırı öğrendiği yani ezberlediği (overfitting) gözlemlenmiştir [88, 89]. Hem gerçek hem de

yetenekli sahte imza örneklerinin sınırlı sayıda olması derin öğrenme algoritmalarının bu alanda henüz yeterince iyi bir performans elde edememesine neden olmaktadır. Bu da eğitim verisi sayısını imza doğrulama alanında en önemli problem haline getirmektedir. Bu probleme ciddi bir çözüm getirilememesi derin öğrenme algoritmalarının çevrimdışı imza doğrulama alanındaki gerçek performansını elde edemememize neden olmaktadır.

El yazısı imzalar, oldukça düşük karakteristik özelliklere sahiptir. Bunun en büyük sebebi de bir imzanın, aynı şartlar altında bile aynı şekilde tekrarlanabilir olmayışıdır. En yetenekli kişiler bile kendi imzalarını daha önce attıkları imzalar gibi bire bir taklit edememektedir. İmzaların bu şekilde yüksek sınıf içi değişkenliğe sahip olmaları onların gerçek ve sahte olarak sınıflandırılma işlemini çözülmesi zor bir problem haline getirerek sınıflandırma yöntemlerinin performansını önemli ölçüde olumsuz olarak etkilemektedir.

Bu tez çalışması kapsamında çevrimdışı el yazısı imzaların sınıflandırma başarısını artırmak için yukarıda belirtilen problemlere çözüm önerisi getirilmiştir. Literatürde de birçok alanda olduğu gibi imza doğrulama alanında da başarısıyla büyük ilgi uyandıran derin öğrenme yöntemleri üzerinde çalışılmıştır.

İlk olarak çevrimdışı imza doğrulama için yazara bağımlı (WD) ve yazardan bağımsız (WI) imza doğrulama sistemlerinin derin öğrenme yöntemleri üzerindeki başarıları araştırılmıştır. Bu araştırma sonuçlarına göre hangi yöntem üzerinden ilerlenmesi gerektiğine karar verilmiştir.

Uygun sisteme karar verildikten sonra imza doğrulamanın derin öğrenme üzerindeki en büyük sınırlılıklarından biri olan yetersiz veri problemine çözüm aranmıştır. Derin öğrenme tabanlı yeni bir veri çoğaltma (Data Augmentation) yöntemi geliştirilerek bu problemin aşılması amaçlanmıştır.

Derin öğrenme modellerinin öğrenmesini olumsuz etkileyecek yetersiz veri probleminin çözülmesinden sonra imza doğrulamasının başarısını artıracak yeni bir derin öğrenme modeli araştırılmıştır. Evrimsel sinir ağları yaklaşımına göre literatürde en çok başarı elde eden modeller incelenmiş ve imza doğrulama alanındaki başarıları araştırılmıştır. Bu modeller üzerinde iyileştirme/ince ayar (fine tuning) ve hiper parametre araştırması yapılarak en uygun model oluşturulmaya çalışılmıştır.

Son olarak tez kapsamında gerekleřtirilen tm alıřmalardan elde edilen sonular ıřıėında evrimdıřı imza doėrulama alanındaki sınıflandırma bařarısını artıracak yeni bir CNN mimarisi oluřturulması amalanmıřtır. Bu kapsamda veri artırma ve sınıflandırma yntemlerinin birleřiminden oluřan derin ėrenme tabanlı yeni bir sistem nerilerek tez alıřması sonulandırılmıřtır.

5. KULLANILAN VERİTABANLARI VE ÖN İŞLEM YÖNTEMLERİ

Bu bölümde tez çalışması kapsamında kullanılan veritabanları ve gerçekleştirilen ön işlem yöntemleri hakkında bilgi verilmektedir.

5.1. Kullanılan Veritabanları

Bu tez kapsamında yapılan tüm imza doğrulaması çalışmalarında literatürde bu alanda en çok kullanılan ve iyi bilinen iki veri seti üzerinde deneyler gerçekleştirilmiştir. Bunlar GPDS ve MCYT veri tabanlarıdır.

5.1.1. GPDS

GPDS, İspanya'daki dijital sinyal işleme grubunun (Digital Signal Processing Group) adıdır. Bu grubun araştırma alanında kullanılmak üzere oluşturduğu parmak izi, retina, el yazısı imza ve dudak izi gibi birçok veri tabanı bulunmaktadır. Bu tez çalışması kapsamında imza doğrulama işlemi için en yaygın kullanılan veri tabanlarından biri olan GPDSsyntheticSignature [90] Veritabanı kullanılmıştır. GPDS veri tabanı "Instituto Universitario para el Desarrollo Tecnológico y la Innovación en Comunicaciones (IDeTIC)" den temin edilmiştir. Bu grubun çevrimdışı imza alanında 4 farklı veri tabanı vardır: GPDS960 imzası, 4NSigComp2010 Senaryo 2, GPDS960GRAY imzası, GPDSsyntheticSignature. Çalışmada GPDSsyntheticSignature veri tabanı kullanılmıştır, çünkü bahsedilen ilk üç veri tabanı 2016/679 Genel Veri Koruma Yönergesi ("GSYİH") kapsamında verilerin korunması amacıyla artık mevcut değildir. GPDSsyntheticSignature veri kümesi 4000 farklı bireyin imzalarından oluşur. Her bireyin 24 orijinal imzası ve 30 sahte imza örneği vardır. Tüm imzalar farklı model kalemler ile oluşturulmuştur. İmza görüntüleri ".jpg" formatında ve 600 dpi'lik çözünürlüktedir.

5.1.2. MCYT

MCYT [91] biyometrik veri tabanı, Madrid'te bulunan Politeknik üniversitesinin biyometrik araştırma laboratuvarı tarafından araştırmalarda kullanılmak üzere yeterli istatistiksel veriye sahip biyometrik veri tabanı geliştirme projesi kapsamında ortaya

çıkmiştir. Veri tabanı kapsamında parmak izi ve el yazısı imza veri kümeleri bulunmaktadır. Bu tez kapsamında MCYT-75 isimli çevrimdışı imza veri kümesi kullanılmıştır. Rastgele seçilmiş 75 bireyin imzasından oluşmaktadır. Her bir bireye ait 15 gerçek ve 15 sahte (yetenekli sahte) imza örneği bulunmaktadır. Tüm imza örnekleri, BMP formatında 600 dpi çözünürlükte bir tarayıcı ile dijitalleştirilmiştir ve farklı boyutlardırlar.

5.2. Ön İşlem Yöntemleri

Çevrimdışı imza doğrulama için doğrulama sistemine girdi verisi olarak taratılmış bir imza resmi kullanılmaktadır. Taratılarak elde edilen görüntüler farklı boyutlarda, farklı renklerde ve farklı arka alanlara sahip olabilmektedirler. Bu görüntülerin doğrulama sistemlerinde işlenebilmesi için belirli standartlara kavuşması (özellikle CNN gibi DL modelleri girdi olarak aynı boyut ve yapıda görüntülere ihtiyaç duymaktadır) gerekmektedir. Bu nedenle de tüm imza görüntülerine dijitalleştirme, gürültü temizleme, sınır belirleme, normalleştirme ve yeniden boyutlandırma gibi ön işlem aşamaları uygulanmaktadır. Bu tez kapsamında uygulanan ön işlem aşamaları ve kullanılan yöntemler bu bölümde detaylı olarak anlatılmaktadır.

5.2.1. Dijitalleştirme (Binarization)

Görüntünün yüklenerek tüm resmin 0-1 renk kodlarına dönüştürülmesi aşamasıdır. Tüm çalışmalarda gri seviye görüntü işlenmiştir. İkili sayı sistemine görüntüleri dönüştürmek imzanın arka alan ile ayrılmasını ve karakteristik bilgileri içeren piksellerin belirginleşmesini sağlamaktadır. Çalışma kapsamında bu işlem için otsu [92] algoritması kullanılmıştır. Otsu algoritması ile imza görüntülerinin arka alanları temizlenmiş ve tüm imza görüntülerine ait pikseller ikili sisteme dönüştürülmüştür. Şekil 5.1'de imzaya ait orijinal ve otsu algoritmasıyla dijitalleştirilmiş halleri gösterilmektedir.



Şekil 5.1. İmzaya ait ön işlem görüntüleri; (a) orijinal görüntü, (b) otsu algoritması sonrası elde edilen görüntü

5.2.2. Sınır/kenar belirleme (Boundary finding)

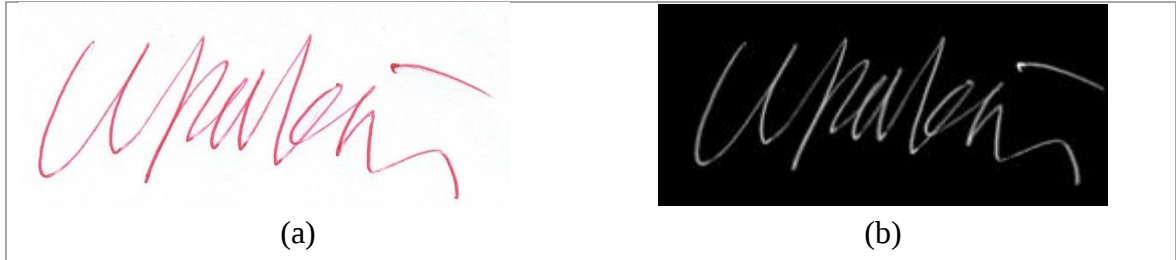
Görüntülerde bazı imzalar geniş olurken bazıları dar olabilmektedir. Aynı şekilde bazı imzaların yüksekliği fazla olurken bazıları ise ince olabilmektedir. Tüm bunlar görüntü içerisindeki imzanın konumunun farklı olmasına ve bu nedenle de görüntüdeki boş piksellerin sayısının artmasına neden olmaktadır. Bu sorunun üstesinden gelebilmek için tüm imza görüntülerinde imzaların başladığı sağ, sol, yukarı ve aşağı kenardaki sınırları tespit edilmiştir. Kenar sınırlarına göre imzalar yeniden boyutlandırılarak görüntüdeki fazla arka alan pikselleri yok edilmiştir. Kenar sınırlarını bulmak için kullanılan algoritma aşağıda verilmiştir.

Çizelge 5.1. Kenar sınırları bulma algoritması

Kenar Belirleme Algoritması	
Adım 1	Üst_Sınır = -1, Alt_Sınır = -1, Sağ_Sınır = -1, Sol_Sınır = -1
Adım 2	FOR (i 0'dan Görüntü Genişliğine)
Adım 3	FOR (j 0'dan Görüntü Yüksekliğine)
Adım 4	IF ((i, j) Noktasında Siyah Piksel Varsa)
Adım 5	Sol_Sınır = i
Adım 6	IF ((Genişlik - i, j) Noktasında Siyah Piksel Varsa)
Adım 7	Sağ_Sınır = Genişlik - i
Adım 8	IF (Sol_Sınır ve Sağ_Sınır Farklıysa -1'den)
Adım 9	Döngülerden Çık
Adım 10	FOR (j 0'dan Görüntü Yüksekliğine)
Adım 11	FOR (i 0'dan Görüntü Genişliğine)
Adım 12	IF ((i, j) Noktasında Siyah Piksel Varsa)
Adım 13	Üst_Sınır = j
Adım 14	IF ((i, Yükseklik - j) Noktasında Siyah Piksel Varsa)
Adım 15	Alt_Sınır = Yükseklik - j
Adım 16	IF (Üst_Sınır ve Alt_Sınır Farklıysa -1'den)
Adım 17	Döngülerden Çık

5.2.3. Normalleştirme (Normalization)

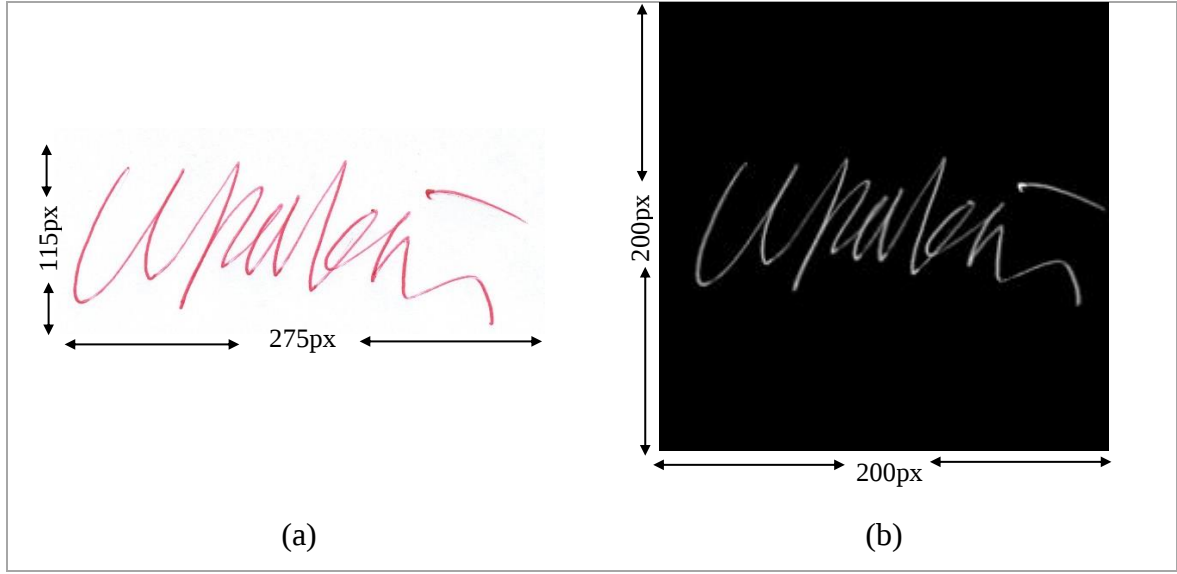
Otsu algoritmasıyla arka alanı temizlenen ve bitsel olarak ifade edilen imza görüntüsünde imzaya ait pikseller 0 (siyah) ve arka alana ait pikseller 1 (beyaz) değerine sahiptir. Arka alan, görüntünün en büyük oranına sahiptir ve asıl işlem görecektir imzaya ait piksellerse daha azdır. Daha kolay işlem yapabilmek ve performansı artırmak için işlem görecektir imza piksellerini 1 ve daha fazla olan arka alan pikselleri 0 değerine güncelledik. Bunun için görüntüdeki tüm pikselleri 1 değerinden çıkararak yeni değerlerini elde ettik. Böylece görüntü üzerinde matris işlemi hesap karmaşıklığı azalmıştır. Şekil 5.2’de imzaya ait orijinal ve normalleştirme işlemi sonrası halleri gösterilmektedir.



Şekil 5.2. İmzaya ait normalleştirme işlemi görüntüleri; (a) orijinal görüntü, (b) normalleştirme işlemi sonrası elde edilen görüntü

5.2.4. Yeniden boyutlandırma

İmza görüntüleri farklı boyutlarda olduğu için sınıflandırma işlemini gerçekleştirmek zor olmaktadır. Özellikle de DL modelleri ile sınıflandırma yapmak istendiğinde, CNN gibi DL modellerinin standart boyutta matrisel görüntüleri girdi verisi olarak kabul ettiğinden farklı boyutlardaki bu verilerle işlem gerçekleştirilemez. Tüm imza görüntüleri, imzaların en boy oranını bozmayacak şekilde yeniden boyutlandırıldı. Bu işlemi gerçekleştirmek için sınırları daha önceden belirlenen imza görüntülerinin en boy oranını çıkararak, belirlenen standart boyutlara göre yeni en ve boy oranları hesapladı. Her imza farklı en ve boy oranlarına sahiptir. Bu nedenle görüntünün istenilen standart en ve boy oranlarına, görüntü şeklini bozmadan yeniden boyutlandırılması için artan pikseller arka alan rengi (siyah) ile doldurulmuştur. Yani boyutlandırma esnasında görüntünün istenilen boyutlar içerisindeki olması gereken en büyük oranı hesaplanmış ve kalan piksellere 0 değeri atanmıştır. Şekil 5.3’de imzaya ait orijinal ve yeniden boyutlandırma işlemi sonrası oluşan yeni görüntüye ait görseller gösterilmektedir.



Şekil 5.3. İmzaya ait yeniden boyutlandırma işlemi görüntüleri; (a) orijinal görüntü, (b) yeniden boyutlandırılmış (CNN için uygun boyuta getirilmiş) görüntü

6. EVRİŞİMSEL SİNİR AĞLARI (CNN) TABANLI İMZA DOĞRULAMA UYGULAMASI

Çevrimdışı el yazısı imzalar en çok tercih edilen imza biçimi olmasıyla birlikte ayırt ediciliği de en az olan imza biçimidir. İmzanın yazıldığı yere, kullanılan yazı araçlarına (kalem, kâğıt vs.), atıldığı ortam ve şartlara ve hatta sahibinin ruhi ya da fiziki durumuna, kullanılan ele göre değişmesi kaçınılmazdır. En yetenekli kişiler dahi hiçbir zaman aynı hareketi aynı şekilde gerçekleştiremezler. Bu nedenle çevrimdışı imzalarda doğrulama işlemi çözülmesi gereken zor problemler arasında yer almaktadır. Bu bölümde çevrimdışı imza tanıma işlemi için derin öğrenme yönteminin elde edeceği başarı gözlemlenmiştir. Derin öğrenme modeli olarak Evrişimsel Sinir Ağları (Convolutional Neural Network - CNN) kullanılmıştır. Bu yöntem literatürde sıklıkla kullanılan ve Kullanılan Veritabanları bölümünde detaylı olarak anlatılan GPDSsyntheticSignature veri seti üzerine uygulanmıştır. Bu bölümde gerçekleştirilen çalışma ile CNN tabanlı modellerin çevrimdışı el yazısı imza doğrulaması problemindeki başarısının incelenmesi amaçlanmıştır. Aynı zamanda imza doğrulama işlemi için CNN ağının eğitimi bir birinden bağımsız olarak yazara bağımlı (Writer Dependent - WD) ve yazardan bağımsız (Writer Independent - WI) olmak üzere iki biçimde gerçekleştirilmiştir. Böylece bu iki yöntemin çevrimdışı imza sınıflandırma başarısına olan katkıları analiz edilmiştir. Uygulama KERAS framework üzerinde Tensorflow alt yapısı kullanılarak Python dilinde çalıştırılmıştır. Kullandığımız yöntem harici bir öznitelik çıkarım yöntemi ile desteklenmeden sonuçlar elde edilmiştir.

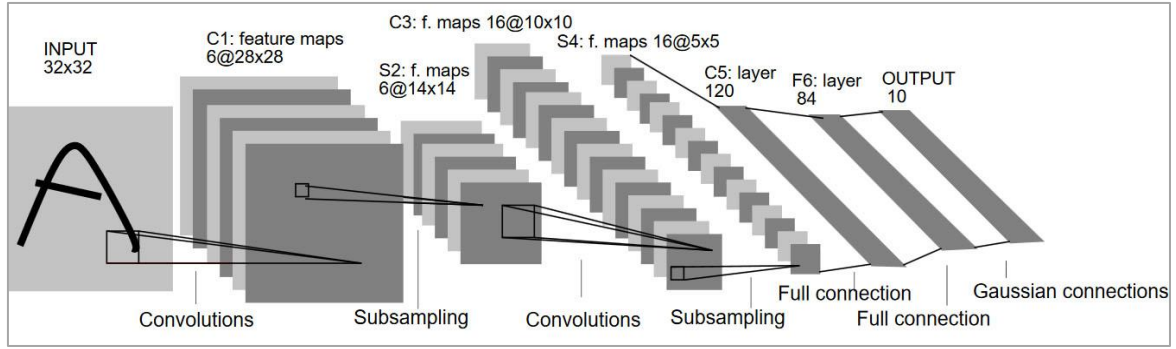
Deneylerde kullanılan veritabanı, modele ait katman ve hiper parametreler, elde edilen bulgular ve öneriler “2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)” kongresinde sunulan bildiride ayrıntılı olarak verilmiştir [93].

6.1. Evrişimsel Sinir Ağları (CNN)

Bir yapay sinir ağı modeli olan Evrişimsel Sinir Ağları (Convolutional Neural Network-CNN) ilk olarak LeCun ve ark. tarafından [94] görüntü işleme için önerilmiş ve paylaşımlı ağırlıklar (Spatial Shared Weights) ve havuzlama (Spatial Pooling) gibi iki temel özellikten oluşmuştur. Aynı ekip [95] 1998 yılında el yazısı sayı sınıflandırma başarısını artırmak için CNN algoritmalarını geliştirerek LeNet-5 adında yayınlamışlar ve ilk 7 seviyede (7-level)

başarı elde etmişlerdir. Günümüzde CNN'ler hem otomatik öznitelik çıkarımında hem de sınıflandırma işleminde yüksek başarı elde etmektedir. Otonom araçlar [82, 85, 87], tıbbi görüntü işleme [96–100], nesne tanıma [101–104], karakter tanıma [105–108], video işleme [109–113] gibi birçok alan da sıklıkla kullanılmaktadır.

LeCun ve arkadaşlarının da belirttiği gibi, bir CNN bir evrişimsel katman (convolution layer), bir havuz/alt örneklem katmanı (subsampling/pooling layer) ve bir tam bağlı katman (fully-connected layer) olmak üzere üç temel katmandan meydana gelmektedir. LeCun ve ark. tarafından tasvir edilen örnek bir CNN mimarisi Şekil 6.1'de gösterilmektedir.

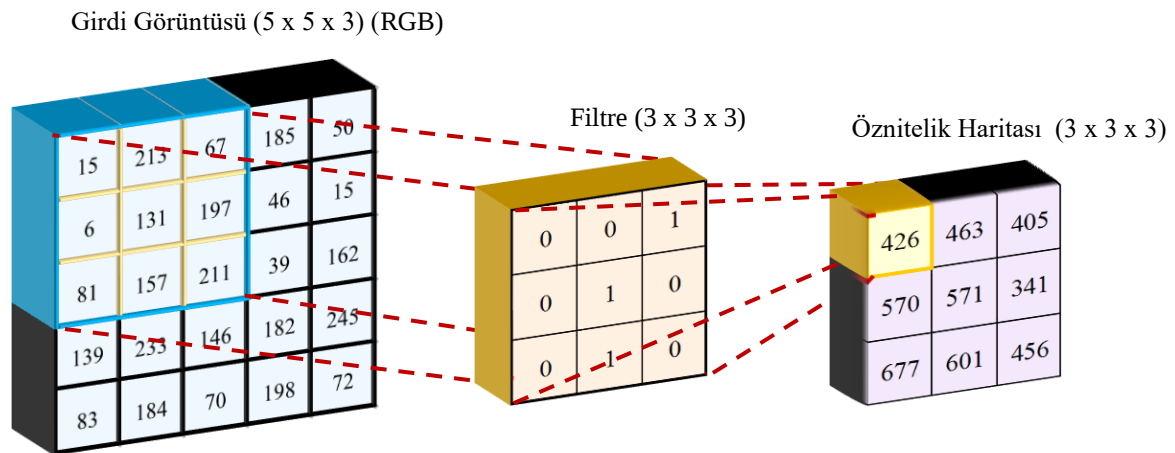


Şekil 6.1. LeCun'ın çalışmasından alınmış örnek bir CNN mimarisi [95]

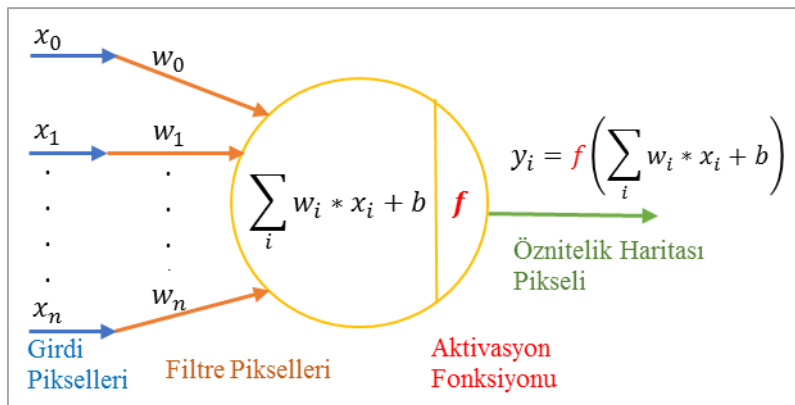
LeCun CNN'lerin çalışma yapısını şu şekilde tanımlamaktadır; CNN, evrişimsel işlemleri ve havuzlama işlemlerini kullanarak görüntülerin soyut özelliklerini öğrenmeyi amaçlamaktadır. İlk katmanlarda elde edilerek öğrenilen özellikler nesnelerin kenarları, temel ovalikleri veya renk verilerinden meydana gelirken, son katmanlara doğru ilerledikçe öğrenilen özellikler nesnelerin bütünsel yapılarından, şekillerinden ve parçalarından meydana gelmektedir [95]. Bu öğrenme biçiminin insan gözünün korteksi ile benzer biçimde olduğu tespit edilmiştir. CNN'ler de insan gibi öncelikle görüntüye ait küçük parçaları görüp ilerleyen katmanlarda bu parçaların birleşiminden oluşan büyük tabloyu görebilmektedir.

CNN modelinde girdi verisi dijitalleştirilerek matris formatına dönüştürülmüş görüntülerdir. Bu görüntüler en, boy ve renk boyutlarından meydana gelirler. Eğer bir görüntü gri seviyede ise en, boy ve bir renk olmak üzere toplam 2 boyuttan oluşur (Üç boyutlu gibi görünse de gri seviyedeki bir görüntüyü iki boyutta matrislerle ifade edebiliriz. Matrisin satır ve sütunları görüntünün en ve boy oranını temsil ederken, her bir değer de gri renk değerini temsil etmektedir. 512x512x1 gibi ve sondaki 1 gri renk tonu için). Görüntü RGB formatında

renkli bir veriden meydana gelecekse her renk için bir boyut olmak üzere (512x512x3 gibi ve sondaki 3 kırmızı, yeşil ve mavi olmak üzere renk tonlarını ifade eder) üç boyuttan meydana gelir. CNN'lerin bir görüntüdeki özellikleri öğrenmesindeki temel işlem evrişim (convolution) işlemidir. Evrişim işlemi matris biçimindeki girdi verisi üzerinde yine daha küçük bir matris olan ve özellikleri haritalandıran filtre/çekirdek matrisinin gezdirilmesiyle gerçekleşir. Görüntü matrisi üzerinde sağa ve aşağıya kaydırılarak evrişim işlemi tüm görüntü piksellerini kapsayacak şekilde tekrarlanır. Evrişim işleminde gerçekleştirilen matematiksel işlem ise, üst üste gelen görüntü pikselleri ile filtre piksellerinin karşılıklı olarak çarpımları ve bu çarpım sonuçlarının toplanarak üzerine hata değerinin (bias) eklenmesi şeklindedir. Böylelikle her işlemde elde edilen sonuç yeni görüntünün bir pikselini ifade etmektedir. Temel evrişim işlemi Şekil 6.2'de gösterilmektedir. Şekil 6.3 'de bu işlemin bir nöron düzeyinde nasıl gerçekleştiğinin temsili ve formülü görülmektedir.



Şekil 6.2. Evrişim (convolution) işleminin temsili



Şekil 6.3. Nöron temsili ve formül gösterimi

Şekil 6.3’de x girdi verisine ait her bir pikseli, w filtre/kernel matrisine ait her bir pikseli, b hata (bias) değerini, f aktivasyon fonksiyonunu ve y haritalandırılmış çıktı görüntüsüne ait her bir pikseli temsil etmektedir. Şekil 6.2’de verilen örnekte ise $(5 \times 5 \times 1)$ boyutlarında bir girdi ve $(3 \times 3 \times 1)$ boyutlarında filtre gösterilmiştir. Filtrenin girdi üzerinde her pikseli kapsayacak şekilde gezdirilmesi ile $(3 \times 3 \times 1)$ boyutlarında yeni bir öznitelik haritası oluşturulmuştur. Oluşacak yeni öznitelik haritası görüntüsünün boyutu, girdi görüntüsünün boyutu, filtre boyutu, boş kenar sayısı ve filtrenin görüntü üzerindeki kayma adım sayısına bağlı olarak değişmektedir. “Eş.6.1”de verilen formüle göre çıktı görüntüsünün boyutu hesaplanabilir.

$$O = \frac{(W - K + 2P)}{S} + 1 \quad (6.1)$$

O çıktı görüntüsünün boyutu, W girdi görüntüsü boyutu, K filtre (kernel) boyutu, P (padding) girdi görüntüsü kenarlarına eklenen boşluk sayısı ve S (stride) ise filtrenin kayma adım sayısını temsil etmektedir. Evrişim işleminin temel formülü ise “Eş.6.2” ‘deki denklemde verilmiştir. Denklemde, çıkış görüntüsünün pikselleri, giriş görüntüsünün pikselleri, filtrenin pikselleri (çekirdek) ve hata terimi sırasıyla y , x , w ve b ile temsil edilmektedir.

$$y_n = \sum_{n=1}^9 x_n * w_n + b_n \quad (6.2)$$

CNN’lerde evrişim katmanından sonra doğrusal olmayan (nonlinear) bir katman olan (rectified linear unit) ReLU katmanı kullanılmaktadır. Bu katman sayesinde doğrusal işlem (lineer regresyon) gerçekleştiren evrişim katmanına doğrusal olmayan bir açı kazandırılması amaçlanmıştır. Bu katman aslında aktivasyon katmanı olarak isimlendirilmektedir. Başlarda tanh ve sigmoid gibi fonksiyonlar aktivasyon katmanında kullanılırken, araştırmacıların ReLU’yu bulmasıyla bu fonksiyon kullanılmaya başlanmıştır. Bu fonksiyon sınıflandırma başarısını fazla değiştirmemesine rağmen işlem karmaşıklığını azaltarak performans artışı sağlamıştır. Ayrıca en çok işlem karmaşıklığına neden olan gradien yok olması (gradient vanishing) problemine de çözüm getirmektedir [114]. ReLU katmanını “Eş.6.3” ‘te verilen formül ile hesaplanmaktadır.

$$ReLU(x) = \begin{cases} 0, & \text{if } x < 0 \\ x, & \text{if } x \geq 0 \end{cases} \quad (6.3)$$

Evrişimsel sinir ağlarında kullanılan en etkili ve güçlü katmanlardan birisi havuzlama (pooling) katmanıdır. Havuzlama katmanı ile önceki katmandaki verileri alt örneklerle temsil edilebilmektedir. Bundan dolayı alt örneklem (sub-sampling) katmanı olarak da adlandırılır. Ortalama ve maksimum olmak üzere sıklıkla kullanılan iki türü mevcuttur. Ortalama havuzlama (average pooling) üzerinde gezdiği görüntüdeki piksellerin ortalamasını alarak yeni bir piksel değeri üretirken, maksimum havuzlama (max-pooling) üzerinde bulunduğu piksellerin en büyük olanını alarak bu işlemi gerçekleştirir. Havuzlama katmanının avantajı görüntüyü alt örnek boyutuna indirgeyerek parametre sayısını azalmaktadır ve böylelikle işlem sayısı azaldığından sistem performansına olumlu katkıda bulunmaktadır. Literatürde belirlenen diğer bir avantajı ise aşırı öğrenmeyi (over-fitting) önüyor olmasıdır [95]. Günümüzde havuzlama katmanının bazı olumsuz yanları tespit edilmiştir. İlerleyen bölümlerde ayrıntılı olarak anlatılan bu konu kısaca, veriler üzerinden alt örneklem oluştururken asıl verideki bazı özneliliklerin kaybolması sorunu olarak açıklanabilir.

CNN'deki son katman, geleneksel sinir ağlarının temel yapı taşları olan Tam Bağlı katmanlardır (Fully-Connected - FC). Tam bağlı katmanlar, önceki katmanların çıktı nöronlarının bir sonraki katmandaki her nörona bağlanmasıyla oluşur. Daha sonra bir Soft Max katmanı kullanılarak bu katmanda elde edilen sonuçlar sınıflandırma işlemi için bir olasılık dağılımına normalize edilir. Tam bağlı katmanlar, üst düzey filtrelenmiş görüntüleri almayı ve onları sınıfları temsil eden oranlara dönüştürmeyi amaçlamaktadır. Bu oranlar, her bir değer ile her bir kategori arasında ağırlık veya bağlantı gücü olarak ifade edilir [95, 115].

6.2. Yöntem

Evrişimsel sinir ağlarının, kullanıcı bağımsız ve kullanıcı bağımlı çevrimdışı el yazısı imza doğrulaması üzerindeki başarılarını test etmek için bir derin öğrenme modeli oluşturulmuştur. CNN tabanlı modele ait bilgiler Çizelge 6.1'de ayrıntılı olarak gösterilmektedir.

Çizelge 6.1. Kullanılan CNN modeline ait katman ve hiper parametre özellikleri

Katmanlar	Boyutları	Hiper Parametreler
Input	210x300x1	
Conv2D	3x3x32	Stride=1, Padding=0
ReLU	3x3x32	
Conv2D	3x3x64	Stride=1, Padding=0
ReLU	3x3x64	
MaxPooling2D	3x3	Stride=2, Padding=0
Conv2D	3x3x128	Stride=1, Padding=0
ReLU	3x3x128	
Conv2D	3x3x64	Stride=1, Padding=0
ReLU	3x3x64	
Conv2D	3x3x128	Stride=1, Padding=0
ReLU	3x3x64	
MaxPooling2D	3x3	Stride=2, Padding=0
Dense	256	Dropout = 0.5, Activation = ReLU
Dense	256	Dropout = 0.5, Activation = ReLU
Dense	256	Dropout = 0.5, Activation = SoftMax

Bu model üzerinde, kullanıcı bağımsız ve kullanıcı bağımlı doğrulama yöntemleri ayrı ayrı test edilmiştir. İki modelde de GPDSsyntheticSignature Veritabanı kullanılmıştır. Kullanıcı bağımlı model için rastgele seçilen bir kullanıcının gerçek ve sahte örneklerden oluşan toplam 54 imzası ile model eğitilmiş ve test edilmiştir. Modelin eğitim aşamasında 15 adet gerçek (genuine) ve 15 adet sahte (forged) örnek rastgele seçilmiş ve modelin eğitilmesi sağlanmıştır. Daha sonra test aşamasında kalan 9 adet gerçek (genuine) ve 15 adet sahte (forged) imza örneği kullanılmıştır. Kullanıcı bağımsız modelde ise yine tüm veri tabanından rastgele 10 kullanıcıya ait toplam 540 imza örneği seçilmiştir. Eğitim aşamasında tüm kullanıcıların örnekleri içerisinde 150 adet gerçek (genuine) ve 150 adet sahte (forged) örnek rastgele seçilerek kullanılmıştır. Kalan 240 imza örneği ise kullanıcı bağımsız modelin test aşamasında kullanılarak deneyler tamamlanmıştır.

6.3. Deney, Sonuç ve Öneriler

Tüm deneysel sonuçlar veritabanındaki imza örnekleri ile gerçekleştirilmiş olup imza örnekleri üzerinde herhangi bir veri çoğaltma ya da ek öznitelik çıkarım yöntemleri kullanılmamıştır. Bu bölümde gerçekleştirilen çalışmanın ana amacı CNN modelinin yalnız

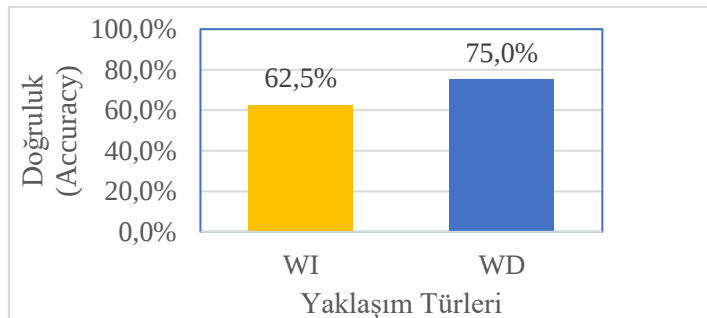
imza örnekleri üzerindeki başarısını analiz etmek ve kullanıcı bağımlı ve kullanıcı bağımsız yaklaşımların imza doğrulama üzerindeki avantajlı ve dezavantajlı yönlerini tespit ederek karşılaştırma yapmaktır. Böylece ileride gerçekleştirilecek çalışmalara temel oluşturulması hedeflenmiştir.

Elde edilen bulgular ışığında, kullanıcı bağımlı yaklaşımın, kullanıcı bağımsız yaklaşıma göre daha başarılı sonuçlar verdiği gözlemlenmiştir. Kullanıcı bağımlı yaklaşıma göre yapılan sınıflandırmada %75 doğruluk oranı elde edilirken, kullanıcı bağımsız yaklaşıma göre yapılan sınıflandırmada başarı oranı %62,5 olmuştur. İki yaklaşıma ait elde edilen deneysel sonuçlar Çizelge 6.2’de verilmiştir.

Çizelge 6.2. Kullanıcı bağımlı ve kullanıcı bağımsız yaklaşımlara göre elde edilen sonuçlar

Kullanılan Model	Yaklaşım Türü	Test Verisi	Eğitim Verisi	Başarı (Accuracy)
CNN	WI	240	300	%62,5
	WD	24	30	%75

Kullanıcı bağımsız yaklaşımların en büyük avantajı, CNN gibi derin öğrenme modellerinin eğitimi için yeterince örnek bulunduruyor olmasıdır. Buna karşın kullanıcı bağımlı yaklaşımlarda CNN modeli sadece tek bir kişinin gerçek ve sahte imza örnekleri ile eğitildiği için (bu veritabanında her birey için toplam 54 imza bulunmaktadır) örnek sayısı öznelilik haritasının öğrenilmesinde oldukça yetersiz kalabilmektedir. Ancak, elde edilen sonuçlar eğitim için yetersiz veriye sahip olmasına rağmen kullanıcı bağımlı modellerin daha başarılı olduğunu göstermiştir. İki yaklaşıma ait karşılaştırma grafiği Şekil 6.4’de görülmektedir. Literatürde yapılan çalışmalarında bu yönde olduğu görülmüştür.



Şekil 6.4. Kullanıcı bağımsız ve kullanıcı bağımlı yaklaşımların karşılaştırılması

İmza örneklerinin sınıf içi çeşitliliğinin yüksek olması aynı kullanıcıya ait örneklerle bile gerçek ve sahte örnekleri sınıflandırmayı zorlaştırırken, farklı kullanıcılara ait örneklerden yola çıkarak global gerçek ve global sahte örneklerin özelliklerini öğrenip bireysel sınıflandırma yapmak çok daha zor olmaktadır. Gerçekleştirdiğimiz deneyde elde edilen sonuçlar bizim bu görüşümüzü destekler nitelikte olmuştur. Bulgular neticesinde daha başarılı bir sınıflandırma modelinin kullanıcı bağımlı yaklaşım üzerine gerçekleştirilmesinin uygun olacağı görülmüştür. Bu nedenle tez kapsamında gelecek çalışmalar kullanıcı bağımlı yaklaşım üzerine gerçekleştirilecektir. Bu bağlamda kullanıcı bağımlı yaklaşımın en büyük sınırlılıklarından biri olan yetersiz veri sorununa çözüm getirilmesi gerekmektedir. Eğer bu sorunun üstesinden gelinirse derin öğrenme yöntemlerinin kullanıcı bağımlı yaklaşım ile çok daha başarılı sınıflandırma yapacağı düşünülmektedir. Bu kapsamda bir sonraki bölümde çevrimdışı el yazısı imza doğrulama için yeni bir veri artırma yöntemi geliştirilmesi amaçlanmıştır.

7.ÇEVİRİMİÇİ İMZA DOĞRLUMA İÇİN DERİN ÖĞRENME TABANLI VERİ ATIRMA YÖNTEMİ

Son yıllarda araştırmacılar derin öğrenme yöntemlerini kullanarak otonom araçlar [82, 85, 87], tıbbi görüntü işleme [96–100], nesne tanıma [101–104], karakter tanıma [105–108], video işleme [109–113] gibi birçok alan da etkileyici ve başarılı sonuçlar elde etmişlerdir. Ancak çok sayıda gizli katmandan oluşan ve milyonlarca eğitim parametresi bulunan derin öğrenme yöntemlerinin başarısı veri kümesinin büyüklüğü ile doğru orantılıdır. Yeterli oranda eğitim verisine sahip bu yöntemler verilerin kalitesi düşük olsa bile, tatmin edici sonuçlar elde edebilirler. Literatürde yapılan birçok çalışma bu görüşü destekler niteliktedir [116]. Örneğin, Li ve ark. [117], yaptıkları çalışmada çevrimiçi medya metin madenciliğini kullanarak ham petrol fiyat tahminine yönelik bir derin öğrenme yaklaşımı önermiştir. Araştırmacılar yöntemi eğitmek ve test etmek için altı binden fazla internet sitesindeki haberlerden elde ettikleri milyonlarca düşük kaliteli kelime verilerini kullanmışlardır. Diğer bir çalışmada, Mansour [118] duyarlılık analizi ve metin madenciliğini kullanarak kullanıcıların sosyal ağlardaki yorumlarından terörizme olan bakış açılarının analizini yapmıştır. Aynı şekilde diğer bir araştırmacı olan Mostafa [119] ise, yine kullanıcıların sosyal ağlardaki yorumlarına göre tüketici ve marka eğilimlerine yönelik bir çalışma yapmıştır. Her iki çalışmada da twitter kullanıcıları tarafından gönderilen çok yüksek miktardaki düşük kaliteli tweet verileri kullanılmıştır. Tüm bu çalışmalar, web sitelerinden toplanan düşük kaliteli ve çok sayıda hata içeren verilerle yapılmıştır ve derin öğrenme yöntemleri bu verilerle eğitilmiştir. Veriler düşük kaliteli ve hatalı olmasına rağmen eğitim için yeterli miktarda olmaları, araştırmacıların başarılı sonuçlar elde etmelerini sağlamıştır. Tüm bu araştırmalar derin öğrenme yöntemleri için, veri miktarının ne kadar önemli olduğunu kanıtlamaktadır.

Önceki bölümde anlatılan çalışmada elde edilen bulgular da net olarak göstermektedir ki çevrimdışı el yazısı imza doğrulamanın en büyük sorunlarından bir tanesi yetersiz veriye sahip olunmasıdır. Özellikle de derin öğrenme gibi eğitim aşamasında yüksek veriye ihtiyaç duyan makine öğrenmesi yöntemleri bu az miktardaki veri sebebiyle çok başarılı sonuçlar elde edememektedir.

Gerçek hayatta imzasının gerçek mi sahte mi olduğunu doğrulanmak istenen kişiye ait çok miktarda orijinal imza bulma olanağı yoktur. Çok kısıtlı miktarda gerçek imzaya ulaşılabilir

ve sahte imzaya ulaşma olasılığı ise (özellikle de yetenekli sahte imzalara) neredeyse hiç yoktur. Bu sebeplerden dolayı imza doğrulaması amacıyla geliştirilen sistemlerin bu senaryoya uygun olması gerçek hayattaki kullanılabilirliği açısından çok önemli ve gereklidir. Bu tez çalışması kapsamında eldeki az sayıda gerçek imzayı kullanarak eğitilebilecek bir sistem geliştirilmesi amaçlanmıştır. Az sayıdaki veriyi çoğaltabilmek için derin öğrenme tabanlı yeni bir veri artırma (data augmentation) yöntemi önerilmiştir. Bu sayede eldeki az sayıda orijinal imzayı çoğaltarak CNN yöntemlerini eğitmek için yeterli veriyi elde etmek hedeflenmiştir.

7.1. Veri Artırma Yöntemleri

Literatürde en çok bilinen, kullanılan ve hatta KERAS gibi bazı derin öğrenme kütüphanelerinde hazır olarak fonksiyonları bulunan veri artırma yöntemleri ayna görüntü (mirroring), döndürme (rotating), kaydırma (shifting), sündürme (flipping) ve rastgele kesme (random cropping) yöntemleridir.

Ayna görüntüsü yöntemiyle veri çoğaltmak için görüntünün x ya da y eksenine göre simetriği alınır, aynadaki görüntüsüne benzer ters bir görüntü elde edilir ve bu şekilde veri artırılır.

Döndürme yöntemiyle görüntü çoğaltma oldukça basittir. Gerçek görüntü z eksenine etrafında döndürülerek yeni görüntüler elde edilir. Rastgele bir dönme açısı verilebileceği gibi 30, 45 ve 90 derece gibi standart bir açı da verilebilir. Bu şekilde eğitim verisinin farklı açılardan görüntüsü elde edilmeye çalışılır.

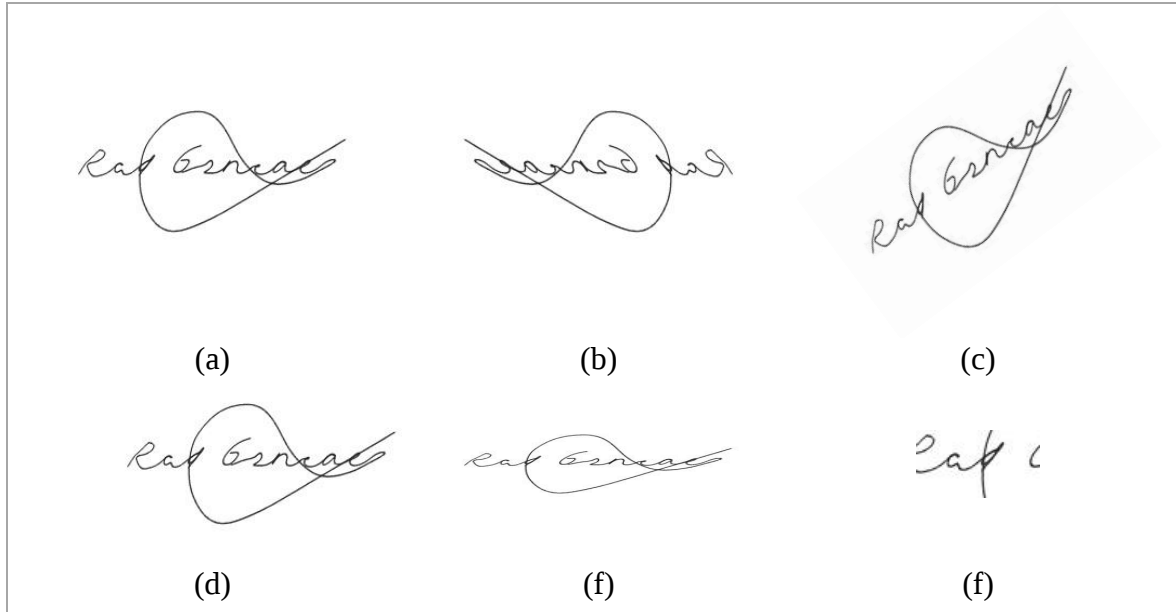
Kaydırma yöntemiyle veri artırmak için gerçek veri x ve y ekseninde ötelenir. Görüntü sağa, sola, yukarı ve aşağı kaydırılarak yeni görüntü elde edilmeye çalışılır.

Sünderme ya da sıkıştırma yönteminde görüntü en ve boy oranının bozulması hedeflenir. Görüntünün en boy oranı çekilerek ya da sıkıştırılarak değiştirilir ve böylece veri artırılmaya çalışılır.

Rastgele kesme yönteminde gerçek görüntüler üzerinden rastgele kesitler alınarak kaydedilir. Kaydedilen veriler sanki yeni bir veriymiş gibi eğitime katılır. Böylece asıl

verilerin sadece bir parçası eğitimde kullanılır. Sistemlerin verideki lokal özellikleri öğrenmesi kolaylaştırılmaya çalışılır.

Tüm bu veri çoğaltma yöntemleri en temel yöntemlerdir ve oldukça sık kullanılmaktadır. Şekil 7.1’de bu veri artırma yöntemlerine örnekler gösterilmiştir.



Şekil 7.1. Temel veri artırma yöntemleri. (a) orijinal görüntü, (b) ayna görüntü, (c) döndürme, (d) kaydırma, (e) sündürme ya da sıkıştırma, (f) rastgele kırpmaya veri artırma yöntemleri

Temel veri artırma yöntemlerinin yanında araştırmacılar tarafından geliştirilen farklı yöntemlerde literatürde bulunmaktadır. Bir çalışmada Lv ve ark. [120] yüz tanıma için beş farklı veri güçlendirme yöntemi kullanmışlardır. Araştırmacıların yöntemi simgesel bozulma (landmark perturbation) ve saç stilleri, gözlük, pozlar, aydınlatma gibi özelliklerden oluşan dört sentez yöntemi içerir. Son yıllarda derin öğrenme tabanlı veri artırma yöntemleri de sıklıkla çalışılmıştır. Tustison ve ekibi [76] "Convolutional neural networks with template-based data augmentation for functional lung image quantification" başlıklı bir makale yayınlamışlardır. Yayınladıkları çalışmada fonksiyonel akciğer Görüntü analizi için şablon tabanlı veri büyütme yöntemi ve ANTsRNet kullanmışlardır. Başka bir çalışmada, Frid-Adar ve ark. [121] bir derin öğrenme modeli olan Generatif Adversarial Networks (GAN') kullanarak sentetik tıbbi görüntüler üretmek için bir veri artırma yöntemi önermiştir.

7.2. Önerilen Yöntem

Çevrimdışı imza doğrulama alanı gibi sınırlı miktarda veriye sahip birçok alanda araştırmacılar veri artırma yöntemlerinin kullanılmasının sınıflandırma başarısını artırdığını kanıtlamışlardır. Bu çalışma kapsamında çevrimdışı el yazısı imza doğrulama alanı için derin öğrenme tabanlı bir veri artırma yöntemi önererek bu alandaki soruna çözüm getirmek hedeflenmiştir. Bilindiği kadarıyla, literatürde bu alana uygulanmış derin öğrenme tabanlı veri artırma yöntemi bulunmamaktadır. Aşağıda detaylı olarak açıklanan yöntem veri artırma yöntemi olarak çevrimdışı imza doğrulama için ilk defa bu tez çalışması kapsamında önerilmektedir.

İmza doğrulama alanında veri artırma için önerilen yöntem aslında daha önce resimden resme dönüşüm (image to image translation) işlemi için kullanılmış bir yöntemdir. Bu yöntem girdi olarak verilen bir görüntüden farklı bir görüntü oluşturmayı amaçlamaktadır. Eğer girdiyi x ile çıktıyı da y ile sembolleştirirsek işlemin formülü $G: x \rightarrow y$ şeklinde olmaktadır. Bu tür dönüşüm yöntemlerinin temel amacı sisteme girdi olarak verilen iki görüntü arasındaki öznitelik haritasını öğrenmek ve daha sonra öğrendiği bu öznitelik haritasını kullanarak ilk defa girdi olarak ve verilen görüntünün farklı formatını üretmektir. Öğrenilen öznitelik haritasına göre resimden resme dönüşüm gri görüntüden renkli görüntüye, görüntüden anlamsal etiketlere, eskiz ya da kontör görüntüden fotoğrafa, fotoğraftan desene gibi farklı biçimlerde olabilmektedir [122]. Bu çalışma kapsamında resimden resme dönüştürme yaklaşımını kullanarak, girdi olarak verilen iki imza örneğini eşleştirilerek aralarındaki özel karakteristiklerin öğrenilmesi amaçlanmıştır. Öğrenilen bu özel karakteristikleri daha sonra kullanarak yeni bir imza örneği oluşturulması hedeflenmiştir. Bu şekilde aynı yazara ait iki farklı el yazısı imza arasındaki benzerlik haritasının çıkarılması ve bu benzerlik haritasını kullanarak kişiye ait yeni bir imza örneği oluşturulması amaçlanmıştır. Böylece kişiye ait az sayıdaki imza örneği kullanılarak veri artırma işlemi gerçekleştirilmektedir.

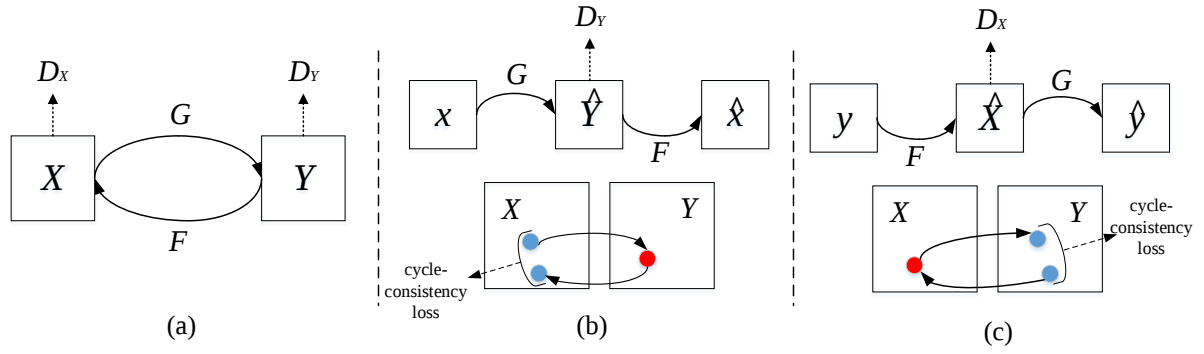
Bu tür dönüşüm yapabilen en yeni derin öğrenme yöntemlerinden biri Üretken Çekişmeli Ağ (Generatif Adversarial Networks - GAN) dır. GAN ilk olarak Goodfellow ve ark. [123] tarafından gerçekleştirilen çekişmeli ağ (adversarial network) tabanlı bir çalışmayla ortaya çıkmıştır. Bu çalışmada oluşturdukları derin öğrenme modeli, G (generatif) üretici adını verdikleri ve rastgele verilerden eğitim havuzundakilere benzer yeni bir veri oluşturmaya

çalışan bir ağ ile, D (discriminator) ayırt edici adını verdikleri ve kendisine girdi olarak gelen verinin G tarafından üretilen sahte veri mi yoksa eğitim havuzundan gelen gerçek veri mi olduğunu tespit etmeye çalışan ikinci bir ağdan meydana gelmektedir. Yani önerilen bu modeldeki G ağının amacı x verilerine göre P_x dağılımı olasılığını öğrenmektir. G ağının giriş verisi olan z rastgele bir gauss dağılımı $P_z(z)$ ile başlar, sonra z giriş değişkenlerini üretici ağ $G(z; \theta_g)$ üzerinden eğitim verisi alanındaki verilerle eşleştirilerek bu verilere benzetilmeye çalışılır. D ağı, bir görüntünün eğitim verilerinden mi yoksa G 'den mi olduğunu tanımlamak için tasarlanmış bir $D(x; \theta_d)$ sınıflandırıcısıdır [124]. Bir GAN'ın amaç fonksiyonu “Eş. 7.1”de verilen formüle göre hesaplanmaktadır.

$$L_{GAN}(G, D_Y, X, Y) = \mathbb{E}_{y \sim P_{data}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim P_{data}(x)}[\log(1 - D_Y(G(x)))] \quad (7.1)$$

Formüldeki G ağı ürettiği yeni veriyi orijinal veriye yakınlıktırarak için amaç fonksiyonunu küçültmeye çalışırken, D ağı kendisine girdi olarak gönderilen verinin sahte mi orijinal mi olduğunu tespit etmek için amaç fonksiyonunu maksimize etmeye çalışır.

İmza verilerini çoğaltmak için kullandığımız GAN modeli Cycle-GAN [125] yöntemidir. Yöntem girdi olarak verilen iki görüntü arasındaki benzerlik haritasını öğrenmek için çekişmeli ağın kayıp (loss) değerini sürekli günceller. Denetimli regresyon ve çekişmeli ağın kayıp fonksiyonunun bir kombinasyonunu kullanır. Modelde iki girdi arasındaki öznelik haritasını oluşturmak için $G: X \rightarrow Y$ ve $F: Y \rightarrow X$ şeklinde iki haritalama fonksiyonu ve bunlarla ilişkili D_Y ve D_X ayırıcıları bulunur. Model fonksiyonlarında da görüldüğü üzere, modelin çalışma prensibine göre bir giriş görüntüsü haritalandırılarak hedef görüntüsüne çevirirse ve daha sonra ters işlem yapılarak görüntüler ikinci fonksiyona göre tekrar işlenirse (çevrilmiş görüntüler giriş görüntülerine tekrar çevrilirse) başta girdi olarak verilen aynı görüntü elde edilmelidir. Bu sebeple, Cycle-GAN modelinin normal GAN ağlardan farklı olarak döngüsel bir yapı mevcuttur ve döngüler arası tutarlılık kısıtlaması vardır. Haritalama aşamasından sonra Cycle-GAN yönteminde bu amaç için bir yeniden yapılandırma aşaması oluşturulmuştur. Bu adımlar, giriş görüntüsü ve hedef görüntüler birbirine dönüştürüldükten sonra tekrarlanır. Bir görüntünün hedef görüntüye dönüştürülmesi ve daha sonra elde edilen görüntünün tekrar giriş görüntüsüne dönüştürülmesi bir çevrim (Cycle) olarak tanımlanmaktadır. Cycle-GAN yönteminin temel mimarisi Şekil 7.2'de verilmiştir.



Şekil 7.2. Cycle-GAN mimarisinin temel yapısı [125]

Şekil 7.2’de verilen yapıya göre modelin çalışma şekli şu biçimdedir: D_Y , G fonksiyonunu kendisine girdi olarak verilen X görüntüsünü Y görüntüsünden ayırt edilemeyecek şekilde dönüştürmesi için yapılandırır. Aynı şekilde D_X ‘te F fonksiyonunu bu işlemin tersini yani gidi olarak verilen Y ‘yi X görüntüsüne en çok benzeyecek şekilde dönüştürmesi için yapılandırır. Bu şekilde bir çevrimlik (cycle) döngü tamamlandıktan sonra dönüşüm işlemlerindeki başarının ve doğruluk oranının hesaplanabilmesi ve artırılması için iki dönüşüm arasındaki tutarlılık kaybı oranı (cycle consistency losses) hesaplanır. Şekil 7.2’de belirtilen (a) Döngü işlemini, (b) ileri döngü tutarlılık kaybını (loss of forward loop consistency) yani x in y ye olan tutarlılığını, (c) ise toplam döngü tutarlılık kaybını (loss of loop-consistency) yani Y nin en baştaki X e olan tutarlılığını ifade etmektedir. Döngü tutarlılık kayıpları “Eş. 7.2” göre hesaplanmaktadır.

$$L_{cyc}(G, F) = \mathbb{E}_{x \sim P_{data}(x)} [\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim P_{data}(y)} [\|G(F(y)) - y\|_1] \quad (7.2)$$

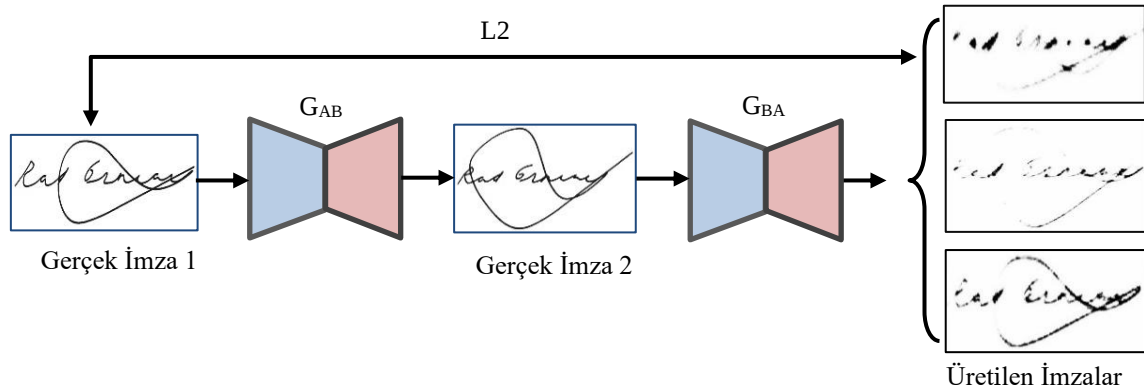
Sistemin amaç fonksiyonunun tamamı ise “Eş. 7.3” göre hesaplanmaktadır.

$$L(G, F, D_X, D_Y) = L_{GAN}(G, D_Y, X, Y) + L_{GAN}(F, D_X, Y, X) + \lambda L_{cyc}(G, F) \quad (7.3)$$

Denklemdaki λ iki amaç fonksiyonu arasındaki ağırlık değerini kontrol etmektedir.

Bu tez kapsamında önerilen Cycle-GAN temelli çevrimdışı el yazısı imza doğrulama sistemi Şekil 7.3’de gösterilmektedir. Önerilen veri artırma yönteminde, imza veri tabanından elde edilen bir kişiye ait gerçek imza örnekleri girdi verisi ve hedef verisi olarak kullanılmıştır. Aynı kişiye ait gerçek imzalar arasında rastgele seçilen iki imzanın biri giriş verisi, diğeri hedef veri olarak birbirleriyle eşleştirilmektedir. Bu şekilde, kişinin gerçek imzaları ortak

öznitelik haritalarını öğrenmesi için önerdiğimiz veri artırma yöntemini eğitir. Eğitilen sistem tarafından kişiye ait tüm gerçek imzalar rastgele eşleştirilerek gerçek imzalar arasındaki benzerlik haritaları oluşturulur, böylece kişinin tüm gerçek imzaları arasındaki üst düzey özellikler örneklenir. Önerilen sistemde Cycle-GAN ağlarının temel aşamalarından bir olan yeniden oluşturma aşamasına göre, bir çevrimlik (cycle) aşama tamamlandıktan sonra, yöntem yeniden yapılandırma aşaması için ters işlemle yeniden çalışmaktadır. Yeniden yapılanma aşamasından sonra, aynı kişinin iki farklı orijinal imzasından yeni bir gerçek imza örneği elde edilir. Bu yeniden yapılandırma aşamasında elde edilen imza önceden belirlenmiş bir eşik değeri (threshold) ile filtrelenir. Eşik değeri kullanılmasıdaki temel sebep, sistem eğitilirken ilk iterasyonlarda sistem tarafından öğrenilen öznitelik sayısı çok az olduğundan bu iterasyonlardaki yeniden oluşturulan örnekler yetersiz kalmaktadır. Çalışma kapsamında kişinin orijinal imzasındaki imza pikselleri yoğunluğunu temel alan bir eşik değeri belirlenerek bu sorun giderilmiştir. Bu eşik değerinin altında kalan yeniden oluşturulmuş örnekler yok sayılmıştır ve böylece sonraki aşamada sınıflandırma için kullanılabilecek artırılmış veriler eşik değerine göre elde edilmiştir. Bu yöntemle, özellik haritasındaki üst düzey özelliklerin kullanılması ve bu üst düzey özniteliklerle yeni imza örneklerinin oluşturulması sağlanmıştır.



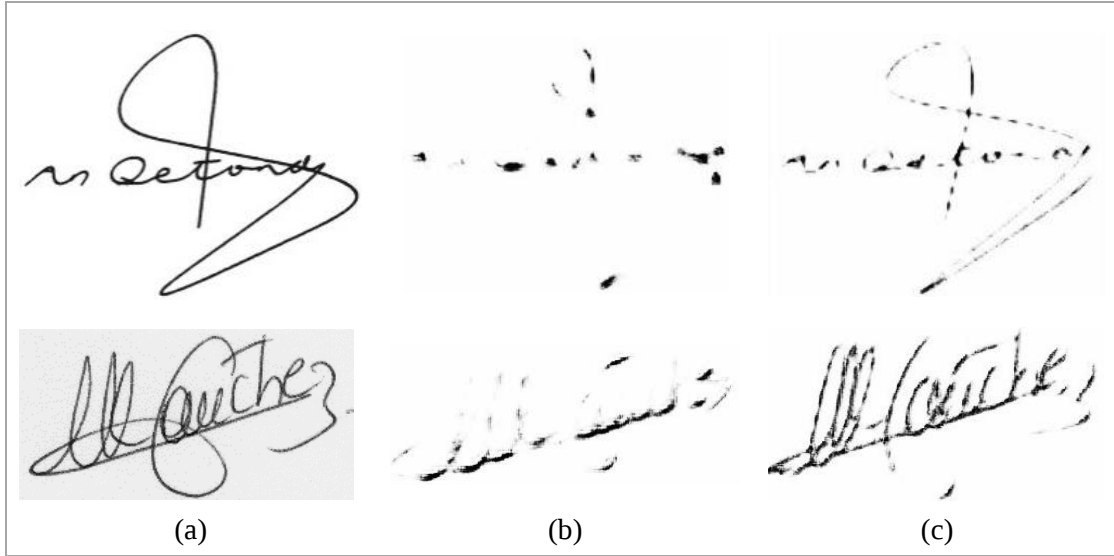
Şekil 7.3. Cycle-GAN yöntemine göre önerilen veri artırma sistemi mimarisi. Aynı kişinin iki farklı gerçek imzası yöntemde eşleştirilir. Sistem eşleşen gerçek imzaların üst düzey özelliklerini öğrenerek haritalandırır ve bu haritadan yeni bir orijinal imza oluşturulur. Farklı iterasyonlar (Epoch) için üretilen gerçek imza örnekleri görülmektedir

Önerilen veri artırma sistemine ait temel algoritma adımları Çizelge 7.1’de verilmiştir.

Çizelge 7.1. Önerilen çevrimdışı el yazısı imza veri çoğaltma sistemine ait algoritma

Algoritma Sahte Kodu	
Adım 1	: Kişiye ait tüm gerçek imza örneklerini oku
Adım 2	: Ön işlem aşamalarını uygula
Adım 3	: FOR (EPOCH _{max})
Adım 4	: FOR (BACH _{max})
Adım 5	: Okunan gerçek imza örnekleri arasından rastgele ikisini seç
Adım 6	: Önerilen veri artırma ağını eği
Adım 7	: END (İç Döngü)
Adım 8	: IF Üretilen İmza Öznitelikleri > threshold
Adım 9	: Öğrenilen öznitelik haritasına göre yeni imzayı üret ve kaydet
Adım 10	: END (IF)
Adım 11	: END (Dış Döngü)

Önerilen çevrimdışı el yazısı imza artırma yöntemiyle iki farklı veritabanından üretilen gerçek imza örnekleri, veritabanındaki orijinalleriyle birlikte Şekil 7.4’de gösterilmektedir.



Şekil 7.4. GPDSsyntheticSignature (ilk satır) ve MCYT75 (ikinci satır) veritabanlarındaki örnekler kullanılarak yeniden oluşturulmuş gerçek imzalar. (a) sütunu orijinal gerçek imzalar, (b) ve (c) sütunları önerilen artırma yöntemiyle üretilmiş gerçek imzalar.

Önerilen veri artırma algoritmasında kişiye ait gerçek imzalar eşleştirilirken kişinin iki farklı imzasının eşleştirilmesi garanti edilmiştir. Kişiye ait aynı imza örnekleri girdi ve hedef olarak eşleşemeyeceğinden sistemin örnekleri kopyalaması engellenmiştir. Bu şekilde sistem tarafından üretilen her imza farklı öznitelik haritaları ile üretileceğinden daha önceki üretilen örneklerden farklı olmaktadır. Çok küçük bir olasılıkla da olsa, daha önce eşleştirilmiş iki imza örneğinin yeniden eşleşmesi ve bu eşleştirme sonucunda daha önceden

üretilmiş aynı benzerlik haritasını yeniden oluşturma ihtimali mümkündür. Bu ihtimal çık küçük olduğundan kopya örnek oluşma olasılığı da oldukça düşüktür.

7.3. Deney, Sonuç ve Öneriler

Önerilen veri artırma yöntemi yazara bağımlı (Writer Dependent - WD) olarak GPDSsyntheticSignature veri seti ile test edilmiştir. Literatürde en çok kullanılan ve başarıları kanıtlanmış CNN modelleriyle deneysel çalışmalar gerçekleştirilmiştir. Bunlar VGGNet16, VGGNet19, ResNet50, DenseNet121 modelleridir. Modellerin tamamı 224x224 pixel gri renkte imza görüntüleriyle, 50 iterasyonda (Epoch), Stochastic Gradient Descent (SGD) optimizasyon yöntemi kullanılarak, 0.0001 öğrenme oranında (learning rate) ve binary cross entropy yöntemiyle eğitilmiştir. Tüm modeller veri artırma yöntemimizin başarısını gösterebilmek adına üç farklı senaryoya göre test edilmiş ve elde edilen bulgular karşılaştırılmıştır. Birinci senaryoda kişiye ait imza verileri üzerinde hiçbir veri artırma işlemi uygulanmamıştır ve bu senaryo “Without Data Augmentation – WODAU” olarak adlandırılmıştır. İkinci senaryoda kişiye ait tüm imza verileri literatürde en çok kullanılan ve KERAS kütüphanesinde de var olan beş temel veri artırma yöntemiyle artırılmıştır ve bu senaryo “Common Data Augmentation – CDAU” olarak adlandırılmıştır. Son senaryoda ise kişiye ait tüm imza verileri sadece çalışma kapsamında önerilen Cycle-GAN tabanlı veri artırma yöntemiyle artırılmıştır ve bu senaryo “Our Data Augmentation- ODAU” olarak adlandırılmıştır.

Deneysel işlemlere geçilmeden önce veri hazırlama işlemi gerçekleştirilmiştir. GPDSsyntheticSignature veri setinde bulunan 4000 kişi arasından rastgele 10 kullanıcı seçilmiştir. Bu kullanıcıların verileri eğitim ve test verisi olarak ayrılmıştır. Birinci senaryoya göre ayrılan veri setinde hiçbir veri artırma işlemi yapılmadığı için eğitim aşamasında sadece 54 imza verisi kullanılmıştır. İkinci senaryoya göre her kişiye ait toplam 54 imza örneği beş temel veri artırma yöntemiyle çoğaltıldığı için kişi başı toplam imza verisi $5 \times 54 = 274$ tane olmuştur. Üçüncü senaryoda önerilen veri artırma yöntemiyle imza verileri çoğaltılmıştır ve önerilen yöntemin iterasyon sayısına göre her bir kişi için ortalama 650 ile 710 arası imza verisi oluşturulmuştur. Tüm senaryolarda verilerin %60'ı eğitim (train) aşamasında, %20 si doğrulama (validation) ve kalan %20'si de test aşamasında kullanılmıştır. Tüm veriler arasından rastgele olarak seçilerek eğitim, doğrulama ve test

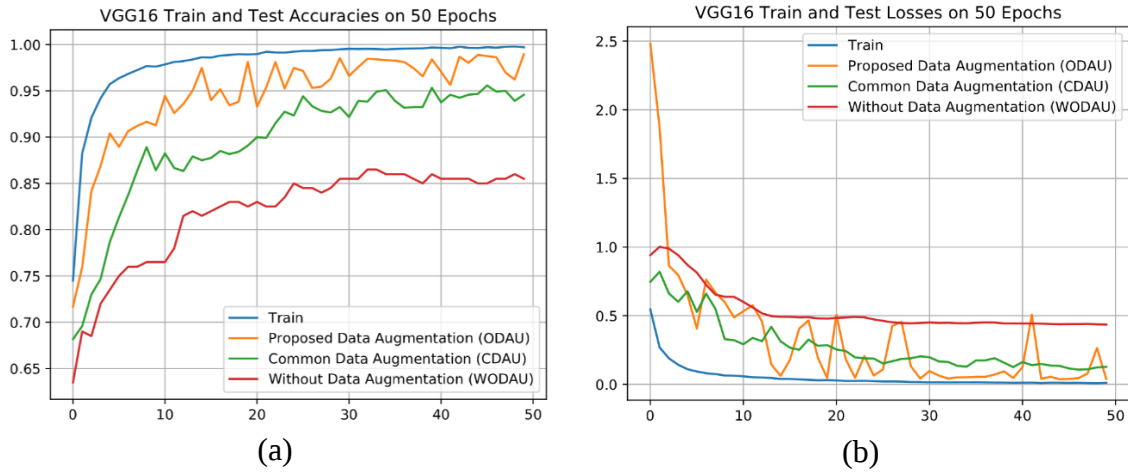
verileri oluşturulmuştur. Tüm verilere ön işlem aşamaları uygulanarak 224x224 boyutlarında normalleştirilmiş ve arka alan rengi siyah olan görüntüler elde edilmiştir.

Tüm deneyler UBUNTU işletim sisteminde PYTHON dili kullanılarak kodlanmış ve KERAS çatısında Tensorflow kütüphanesi kullanılarak grafik kartı (GPU) üzerinde gerçekleştirilmiştir. Deneylerde kullanılan sistem özellikleri 12GB DDR5 belleğe sahip NVIDIA TITAN XP grafik kartı, I7 CPU, 32GB DDR3 RAM ve 500GB SSD sabit bellekten oluşmaktadır. Deneyler yukarıda da belirtildiği gibi veri setinden rastgele seçilen 10 kişinin imzalarıyla 4 farklı CNN modelinde gerçekleştirilmiştir. Her bir modele her bir senaryoya göre ayrı ayrı on kişinin imzalarıyla eğitilmiş ve nihai sonuçlar senaryo bazlı on kişinin sonuçlarının ortalamasıyla oluşturulmuştur. Yani birinci senaryoya göre bir model on ayrı kişinin imzasıyla ayrı ayrı eğitildikten sonra bu senaryonun modele göre başarısı on kişinin sonuçlarının ortalamasıyla belirlenmiştir. Aynı senaryo aynı şekilde diğer modellerle de eğitilerek sonuçların ortalaması alınmış ve bu işlem tüm senaryo ve modellere göre tekrarlanmıştır. Daha tutarlı ve doğru sonuçların elde edilmesi ve aşırı öğrenmenin (overfitting) engellenmesi için 10 kat çapraz doğrulama (10 fold cross validation) gerçekleştirilerek her işlem 10 kez tekrarlanmış ve nihai sonuçlar bunların ortalamasıyla oluşmuştur. Veri artırma yönteminin başarısını gösterebilmek için her senaryoda 4 CNN modeli 10 kullanıcının imzalarıyla 10'ar kez eğitilmiştir ve sonuç olarak toplamda $4(\text{CNN}) \times 10(\text{kişi}) \times 10(\text{çapraz doğrulama})$ olmak üzere 400 ve tüm senaryolar için $3(\text{senaryo}) \times 400 = 1200$ deney gerçekleştirilmiştir.

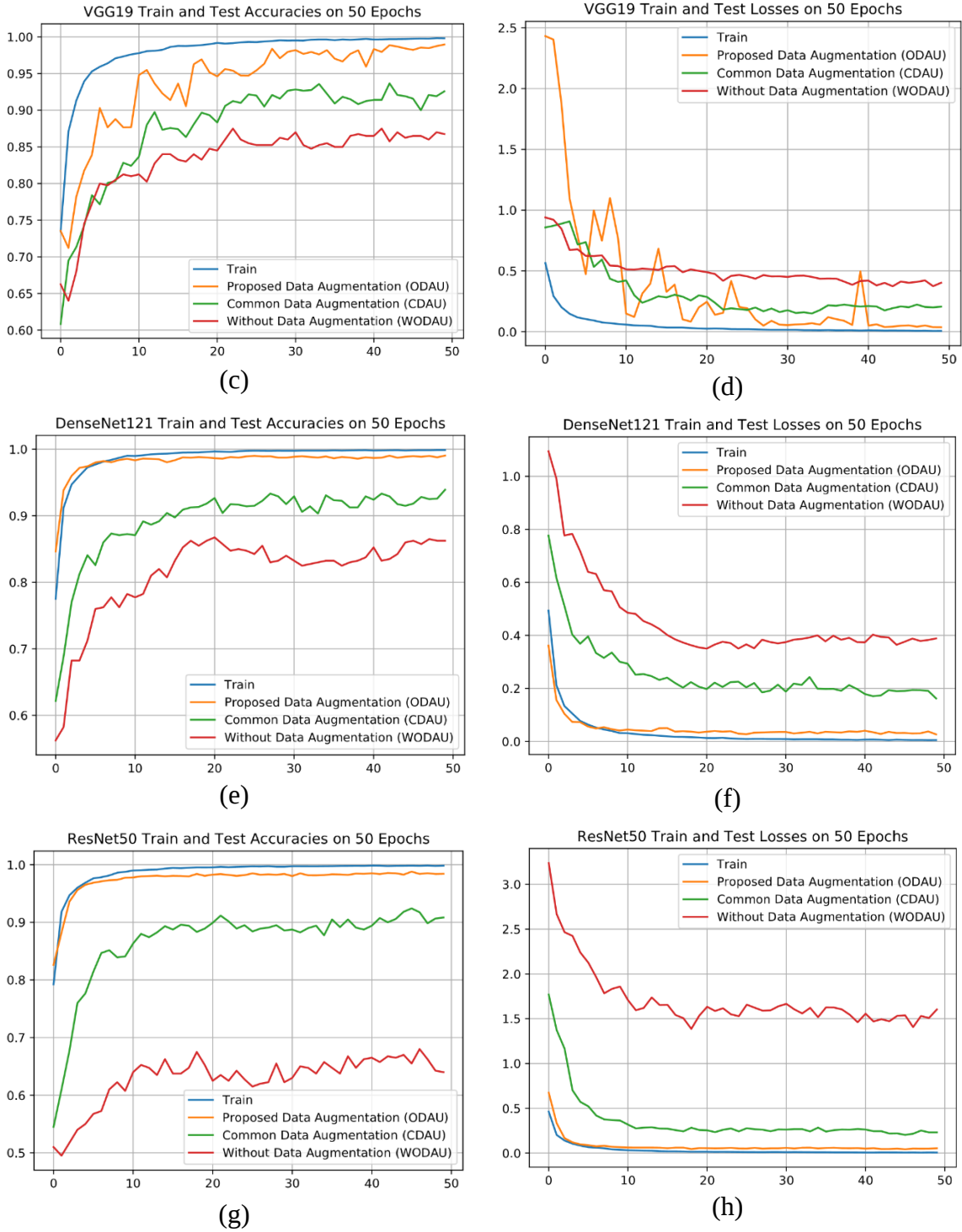
Elde edilen nihai sonuçlar her bir senaryoda kullanılan CNN modeline göre ayrı ayrı karşılaştırılmıştır. Elde edilen sonuçlara ait karşılaştırma tablosu Çizelge 7.2'de verilmiştir. Yöntemin başarısını daha rahat görebilmek ve analiz edebilmek için sonuçlar CNN modeli bazında görselleştirilmiştir. Görsel sonuçlar Şekil 7.5'te verilmiştir. Tüm karşılaştırma işlemleri test başarısı (test accuracy), test kayıpları (test loss) ve skor (score) değerlerine göre yapılmıştır.

Çizelge 7.2. Üç senaryoda her bir CNN modeliyle elde edilen deneysel sonuçlar

Senaryo	Model	Test Accuracy	Test Loss	Score
Without Data Augmentation (WODAU)	VGG16	0,864	0,435	0,866
	VGG19	0,874	0,374	0,865
	DenseNet121	0,867	0,350	0,800
	ResNet50	0,680	1,386	0,737
With Common Data Augmentation (CDAU)	VGG16	0,935	0,111	0,936
	VGG19	0,916	0,158	0,917
	DenseNet121	0,919	0,167	0,917
	ResNet50	0,904	0,213	0,905
With Proposed Data Augmentation (ODAU)	VGG16	0,969	0,044	0,967
	VGG19	0,969	0,044	0,968
	DenseNet121	0,970	0,031	0,968
	ResNet50	0,965	0,049	0,963



Şekil 7.5. CNN modelleri için doğruluk (accuracy) ve kayıp (loss) grafikleri. Her bir grafik eğitim, önerilen veri çoğaltma (ODAU), temel veri çoğaltma (CDAU) ve veri çoğaltma olmadan (WODAU) gerçekleştirilen senaryolara ait doğruluk /kayıp grafiklerinden oluşur. Şekil (a) ve (b) sırasıyla VGG16 yönteminin doğruluk ve kayıp grafiklerini göstermektedir. VGG19 için sırasıyla (c) ve (d), DenseNet121 için (e) ve (f), ResNet50 için (g) ve (h) grafiklerinde doğruluk ve kayıp grafikleri gösterilmiştir



Şekil 7.5. (devam) CNN modelleri için doğruluk (accuracy) ve kayıp (loss) grafikleri. Her bir grafik eğitim, önerilen veri çoğaltma (ODAU), temel veri çoğaltma (CDAU) ve veri çoğaltma olmadan (WODAU) gerçekleştirilen senaryolara ait doğruluk / kayıp grafiklerinden oluşur. Şekil (a) ve (b) sırasıyla VGG16 yönteminin doğruluk ve kayıp grafiklerini göstermektedir. VGG19 için sırasıyla (c) ve (d), DenseNet121 için (e) ve (f), ResNet50 için (g) ve (h) grafiklerinde doğruluk ve kayıp grafikleri gösterilmiştir.

Önerilen veri artırma yönteminin tüm CNN modellerin doğruluğunu arttırdığı açıkça

görülmektedir (Çizelge 7.2 ve Şekil 7.5). Mavi renkte olan veriler tüm modellerin eğitim aşamasında gösterdiği doğruluk ve kayıp değerleridir. Veri artırımı (WODAU) olmayan ilk senaryoya göre gerçekleştirilen deneysel sonuçlar incelendiğinde eğitim ve test verileri arasın arasında yüksek farklılıklar olduğu görülüyor. Tüm CNN modellerinin, yetersiz imza verileriyle bu eğitim aşamasında verileri öğrenemediği ve ezberlediği (overfitting) düşünülmektedir. Bu nedenle, eğitimin başarısı oldukça yüksekken, test başarısı düşüktür. Bu senaryoda elde edilen bulgular bir kez daha veri miktarının derin öğrenme için ne kadar önemli olduğunu ispatlar niteliktedir. Temel veri artırma yöntemiyle (CDAU) verilerin çoğaltıldığı ikinci senaryoda, eğitim ve test sonuçları birbirine daha yakındır. Yeterli eğitim verisi ile CNN modellerinin imza doğrulamasındaki başarısı önemli ölçüde artmaktadır. Önerilen veri artırma yönteminin (ODAU) kullanıldığı üçüncü senaryoda elde edilen eğitim ve test değerleri incelediğinde hem doğruluk hem de kayıp oranlarının eğitim değerlerine çok daha yakın olduğu ve diğer senaryolardan daha başarılı sonuçlar verdiği görülmektedir.

Deney sonuçları model temelinde incelediğinde, hiçbir veri artırma işleminin yapılmadığı birinci senaryoya göre VGGNet19 modelinin test doğruluğunun yüksek ve VGGNet16 modelinin ise skor değerinin yüksek olduğu görülmektedir. Bu senaryoda en iyi kayıp oranı ise DenseNet121 modelinde elde edilmiştir. Sadece beş temel veri artırma yönteminin kullanıldığı ikinci senaryoda en başarılı sonuçları VGGNet16 modelinin elde ettiği görülmüştür. Önerilen veri artırma yönteminin tüm CNN modellerinde yüksek başarı oranları elde ettiği ve DenseNet121 modelinde en yüksek doğruluğu sağladığı görülmüştür. VGGNet19 toplam skor sonucunda birinciliği DenseNet121 modeliyle paylaşmıştır.

Tüm deneysel sonuçlar göstermiştir ki eğitimde kullanılan veri sayısı CNN modellerinin başarısı için çok büyük öneme sahiptir. Önerilen Cycle-GAN tabanlı veri artırma yöntemi kullanılan tüm CNN modellerinde en başarılı sonuçları elde etmiştir. Elde edilen bulgulara göre her ne kadar birinci ve ikinci senaryoda en başarılı modeller VGG modelleri olsa da, VGG modellerinin öğrenmekten çok ezberlediği düşünülmektedir. VGG modellerinin elde ettiği doğruluk değerlerinin eğitimde elde ettiği değerlerden çok farklı olması bu düşüncenin temelini oluşturmaktadır. Ayrıca Şekil 7.5'te verilen grafikte de açıkça görülüyor ki VGG modellerinin hem doğruluk hem de kayıp grafiğinde her bir iterasyonda büyük dalgalanmalar olmuştur. Modellerin kararlı sonuçlar vermemesi de ezberlediği (overfitting) kanaatini güçlendirmektedir. Bu bilgiler ışığında her senaryoda daha kararlı sonuçlar vermesinden ve yüksek başarı değerlerine ulaşmasından dolayı DenseNet121 modelinin

çevrimdışı el yazısı imza doğrulama probleminde en başarılı CNN modeli olduğu sonucuna varılmıştır. Ayrıca ResNet50 modelinin de kararlı sonuçlarından dolayı en başarılı ikinci model olduğu görülmüştür.

Bu bölümde gerçekleştirilen çalışmayla imza doğrulama probleminin çözümünün önündeki en büyük engellerden biri olan eğitim aşaması için yetersiz veriye sahip olması sorununa çözüm getirilmiştir. Tez çalışması kapsamında bir sonraki hedef, bu veri artırma yöntemi ile birlikte kullanılabilir ve daha başarılı sınıflandırma sonuçları elde edilebilecek bir CNN modeli geliştirmektir. Bu bağlamda, bu bölümde elde edilen bulgular ışığında en başarılı CNN modeli olan DenseNet üzerinden ilerlenmesi planlanmaktadır. Modellerin sınıflandırma başarılarını daha iyi karşılaştırabilmek ve tez kapsamında daha sağlıklı ilerleyebilmek için bu CNN modellerinin performanslarını görerek karşılaştırıla bileceği bir çalışma daha gerçekleştirilmiştir. Bir sonraki bölümde gerçekleştirilen bu çalışmaya ait bilgiler, sonuçlar ve öneriler ayrıntılı olarak sunulmaktadır.

8. EVRİŞİMSEL SİNİR AĞI MODELLERİNİN GRAFİK KARTI ÜZERİNDEKİ PERFORMANSLARININ KARŞILAŞTIRILMASI

Son yıllarda derin öğrenme modelleri birçok alanda yüksek başarı göstermiş ve bu başarı görüntü işleme, bilgisayarla görme, otonom araçlar, karakter tanıma, ses ve video işleme gibi birçok popüler alanda çalışan araştırmacıların da ilgisini çekmiştir. Bir önceki bölümde ayrıntılı bir biçimde işlenen ve elde edilen bulgulardan da net olarak görülen derin öğrenme yöntemlerinin başarısına en büyük etkiye sahip olan etmenlerden birisi eğitim verisinin çok olmasıdır. Eğitim verisinin fazlalığı büyük bir avantaj olmakla birlikte CNN gibi derin öğrenme modellerinin eğitilmesi aşamasında kullanılan fazla miktardaki verinin çok büyük işlem gücüne ihtiyaç duyuyor olması bir dezavantaj olarak karşımıza çıkmaktadır. Günümüzdeki merkezi işlem birimleriyle (CPU) bu tür modellerin eğitilmesi günler, haftalar ve hatta aylar alabilmektedir. Araştırmacılar bu problemin üstesinden gelebilmek için grafik işlem birimleri (GPU) üzerinde çalışabilecek derin öğrenme mimarileri geliştirmiştir. Bu sayede CPU ile aylar süren işlemler GPU vasıtasıyla sadece birkaç saatte bitirilmektedir. Milyonlarca görüntü verisinden oluşan IMAGENET gibi veritabanları bile GPU'lar aracılığıyla sadece birkaç saatte işlenebilmektedir.

Bu bölümde, literatürde birçok alanda başarılı sonuçlar elde etmiş ve tez kapsamında bir önceki bölümde gerçekleştirilen çalışmada da ele alınan CNN modellerinin GPU üzerindeki başarılarına daha yakından bakılması planlanmıştır. Nihai amaç en iyi performansa sahip CNN modelini tespit ederek tez çalışması kapsamında imza doğrulama görevini yerine getirmek için, bu en başarılı model temelinde ilerlemektir. Çalışma kapsamında bu bölümde ResNet, VGGNet ve DenseNet modelleri GPDS imza veritabanı ile GPU üzerinde gösterdikleri performanslara göre karşılatırılmıştır. Karşılaştırma kriteri olarak tüm modellerin veritabanı üzerindeki başarıları, GPU üzerinde kullandıkları bellek miktarı ve eğitimde kullanılan zaman miktarları seçilmiştir.

Bu bölümde gerçekleştirilen çalışmanın detayları, elde edilen bulgular ve öneriler “2019 IEEE 13th International Conference on Application of Information and Communication Technologies (AICT), Baku, Azerbaijan” kongresinde sunulan bildiride ayrıntılı olarak verilmiştir [126].

8.1. Materyal ve Yöntemler

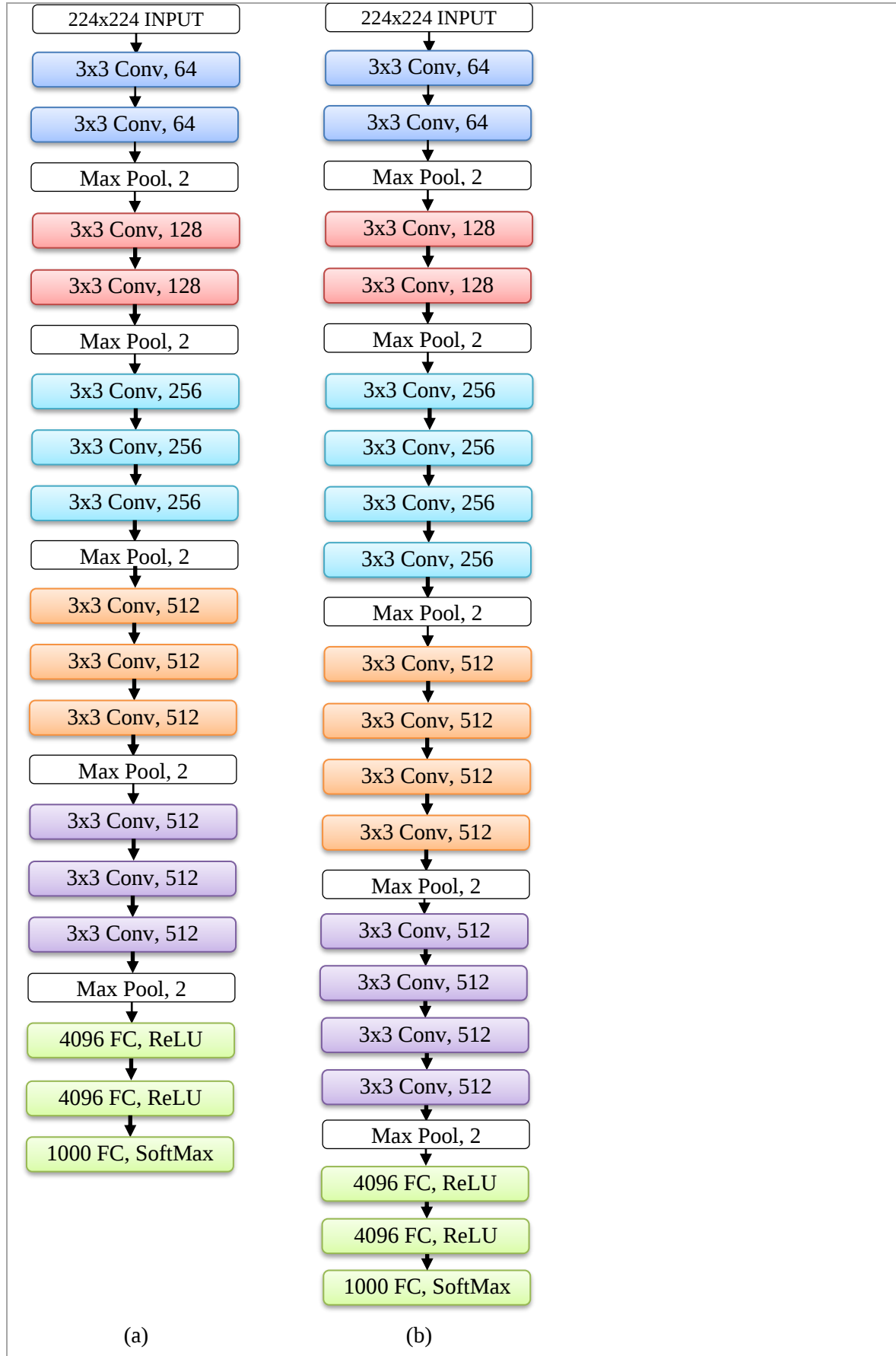
GPDSsyntheticSignature veri setinin kullanıldığı çalışmada 4000 kişi içerisinde rastgele seçilen bir kişiye ait toplam 54 gerçek ve sahte imza örneği kullanılmıştır. Bu veri setine ait detaylı bilgi bölüm 5.1.1 GPDS’de verilmiştir. Performans karşılaştırması için derin öğrenme modellerinden ilk kez LeCun ve ekibi tarafından önerilen Evrişimsel Sinir Ağları (CNN) mimarisine ait üç model kullanılmıştır. CNN mimarisi ve ayrıntıları 6.1 Evrişimsel Sinir Ağları (CNN) bölümünde verilmiştir. CNN mimarisine ait en çok kullanılan ve aşağıda ayrıntılı olarak anlatılmış olan ResNet, DenseNet ve VGGNet modelleri performans karşılaştırmasında kullanılmıştır.

Tüm modeller aynı eğitim parametreleri ile eğitilmiş ve performansları eşit şartlarda karşılaştırılmıştır. Modellere ait Hiper parametreler ve ince ayar değerleri şu şekildedir; Öğrenme oranı $Lr = 0.0001$, bozunma (decay) = $1e-6$, momentum = 0.9, nesterov = True olarak ayarlanmıştır. Stokastik gradyan iniş (SGD) optimizasyon yöntemi olarak ve İkili Çapraz Entropi (Binary Cross Entropy) kayıp fonksiyonu olarak kullanılmıştır.

Tüm deneyler UBUNTU işletim sisteminde PYTHON dili kullanılarak kodlanmış ve KERAS çatısında Tensorflow kütüphanesi kullanılarak grafik kartı (GPU) üzerinde gerçekleştirilmiştir. Deneylerde kullanılan sistem özellikleri 12GB DDR5 belleğe sahip NVIDIA TITAN XP grafik kartı, I7 CPU, 32GB DDR3 RAM ve 500GB SSD sabit bellekten oluşmaktadır.

8.1.1. VGGNet

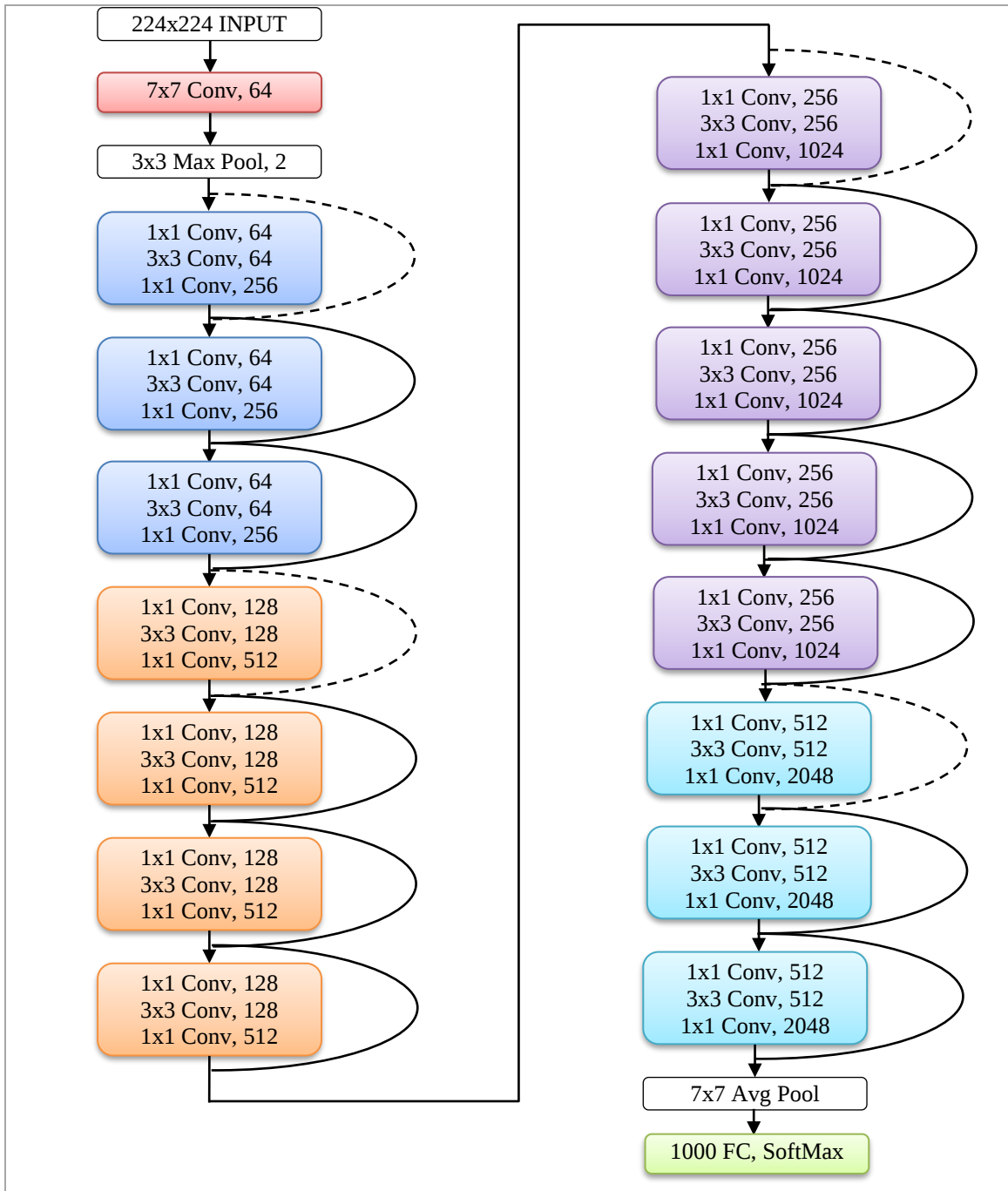
VGGNet, Oxford Üniversitesi'nde bulunan VGG (Görsel Geometri Grubu) üyeleri Simonyan ve Zisserman [127] tarafından önerilmiştir. 14 milyondan fazla verinin ve 1000 sınıfın bulunduğu ILSVRC 2014 yarışmasında TOP-5 seviyesinde önemli bir sonuç elde etmiştir. Modelin yapısı ALEXNet yapısına çok benzemektedir ve ALEXNetin 11x11 ve 5x5 boyutlarındaki filtre yapısını 3x3 boyutlarında küçültmektedir. VGGNet 16 ve 19 katman olarak iki farklı modele sahiptir. VGGNet16 modelinde 13 evrişim ve 3 tam bağlı katman bulunurken VGGNet19 modelinde 16 evrişim ve 3 tam bağlı katman yer almaktadır. Çalışma kapsamında VGGNet19 modeli kullanılmıştır ve bu model Şekil 8.1’de gösterilmektedir.



Şekil 8.1. VGGNet modellerine ait görseller. (a) VGG16 ve (b) VGG19

8.1.2. Residual neural network (ResNet)

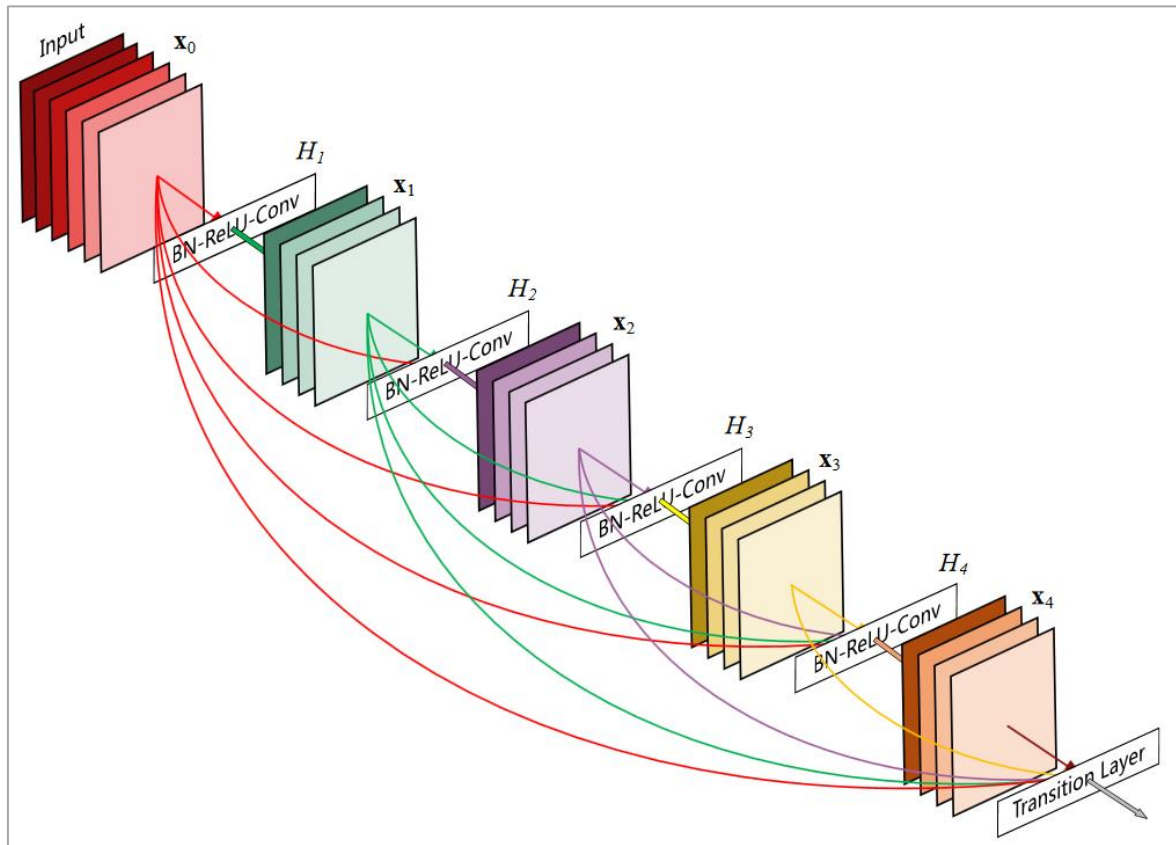
ResNet Microsoft'ta çalışan bir grup araştırmacı tarafından 2015'te geliştirilmiştir [128]. ResNet sayesinde bir önceki katmanın ağırlık değerleri bir sonraki katmana direkt olarak aktarılmıştır. Ayrıca iniş yok olması (vanishing gradient) probleminde de çözüm getirdiği iddia edilmektedir. ResNet'in 18, 34, 50, 101, 152 katmana sahip olan türleri vardır. Bu çalışma kapsamında 50 katmanlı modeli kullanılmış ve yapısı Şekil 8.2'de verilmiştir.



Şekil 8.2. ResNet modelinin temel yapısı

8.1.3. DenseNet

DenseNet, Huang ve ark. [129] tarafından CNN modellerinde ilk katmanlar ile sonraki katmanlar arasındaki kısa bağlantıların daha derin, daha doğru ve verimli olabileceği öngörüsüyle geliştirilmiştir. DenseNet klasik CNN modelindeki her katman ile sonraki katmanı arasında bir tane olmak üzere, birbirine tam bağlı katmanlardan oluşur. Normal bir N katmanlı CNN de N adet bağlantı bulunurken DenseNet modelinde her katman kendinden sonra gelen her katmana bağlı olduğundan $\frac{N(N+1)}{2}$ bağlantıdan oluşur. Her katman için, önceki tüm katmanların özellik haritaları girdi olarak kullanılır ve kendi özellik haritaları sonraki tüm katmanlara girdi olarak gönderilir. Bu sayede DenseNets, ResNet gibi iniş yok olması (vanishing gradient) problemine çözüm getirmeye çalışır, özellik yayılımını güçlendirir, özelliklerin yeniden kullanılmasını teşvik eder ve parametrelerin sayısını önemli ölçüde azaltır. DenseNet'in 121, 169, 201, 264 katmana sahip olan türleri vardır. Çalışma kapsamında 121 katmanlı (DenseNet-121) modeli kullanılmıştır. Yazarların tanıttığı DenseNet modeline ait görsel Şekil 8.3'de verilmiştir.



Şekil 8.3. DenseNet modelinin yapısı [129]

8.2. Deney ve Elde Edilen Sonuçlar

Çalışmada literatürde en çok kullanılan ve bir önceki bölümde de kullanılan CNN modellerinden VGGNet19, ResNet50 ve DenseNet121 modellerinin performans karşılaştırması gerçekleştirilmiştir.

Modellerin, imza veri kümesi kullanılarak GPU üzerindeki performans verileri deneylerde elde edilerek karşılaştırılmıştır. Karşılaştırma kriterleri modellerin toplam parametre sayısından, eğitilebilir parametre sayısından, GPU üzerindeki bellek kullanım miktarlarından, eğitim ve test başarılarından, her parti (BATCH) için eğitim sürelerinden ve toplam eğitim sürelerinden oluşmaktadır. Deneylerde, her modelin performansı 20 ve 50 partide (BATCH), 20 ve 100 eğitim aşamasıyla (EPOCH) ayrı ayrı gözlenmiştir. Karşılaştırmalar GPDSSyntheticSignature çevrimdışı el yazısı imza verilerinden rastgele seçilen bir kişinin imzaları ile Yazara Bağımlı (WD) olarak yapılan deneylerin sonuçlarına göre gerçekleştirilmiştir. Tüm eğitimlerde parametreler IMAGENET'ten aktararak (transfer learning) kullanılmıştır.

Deneylere başlamadan önce ön işlem aşamaları gerçekleştirilmiştir. İlk olarak, seçilen kişinin toplam 54 çevrimdışı el yazısı imzası, çalışma kapsamında önerilen CycleGAN tabanlı yöntemle birlikte geleneksel veri artırma yöntemleri de kullanılarak veri artırma gerçekleştirilmiş, sonuç olarak toplam 3530 el yazısı imza verisine yükseltilmiştir. Veri artırma işleminden sonra imzalara ön işlem uygulanmış ve imzaların arka alanları ve gürültüleri temizlenmiş ve boyutları 224x224 piksele ayarlanmıştır. Daha sonra eğitim, doğrulama ve test verileri tüm veriler içerisinden rastgele seçilerek ayrılmıştır. Tüm imzaların %70'i eğitim aşamasında, %20'si doğrulama aşamasında ve %10'u test aşamasında kullanılmıştır.

Üç CNN modeli içinde deneyler ayrı ayrı toplamda 4 formatta gerçekleştirilmiştir. 1. formata göre her model 20 yığınlık (batch) ölçülerde 20 eğitim adımı (epoch) eğitilerek sonuçlar elde edilmiştir. 2. formata göre her model 20 yığınlık (batch) ölçülerde 100 eğitim adımı (epoch) eğitilerek sonuçlar elde edilmiştir. 3. ve 4. formata göreyse sırasıyla her model 50 yığınlık (batch) ölçülerde 20 eğitim adımı (epoch) ve 50 yığınlık (batch) ölçülerde 100 eğitim adımı (epoch) eğitilerek sonuçlar elde edilmiştir. Tüm modellerin deneylerde elde edilen toplam ve eğitilebilir parametre sayıları, GPU belleği kullanım miktarları ve zaman

yönetimi performansları ayrıntılı olarak Çizelge 8.1’de verilmiştir.

Çizelge 8.1. Modellere ait parametre, GPU bellek kullanımı ve zaman kullanım sonuçları

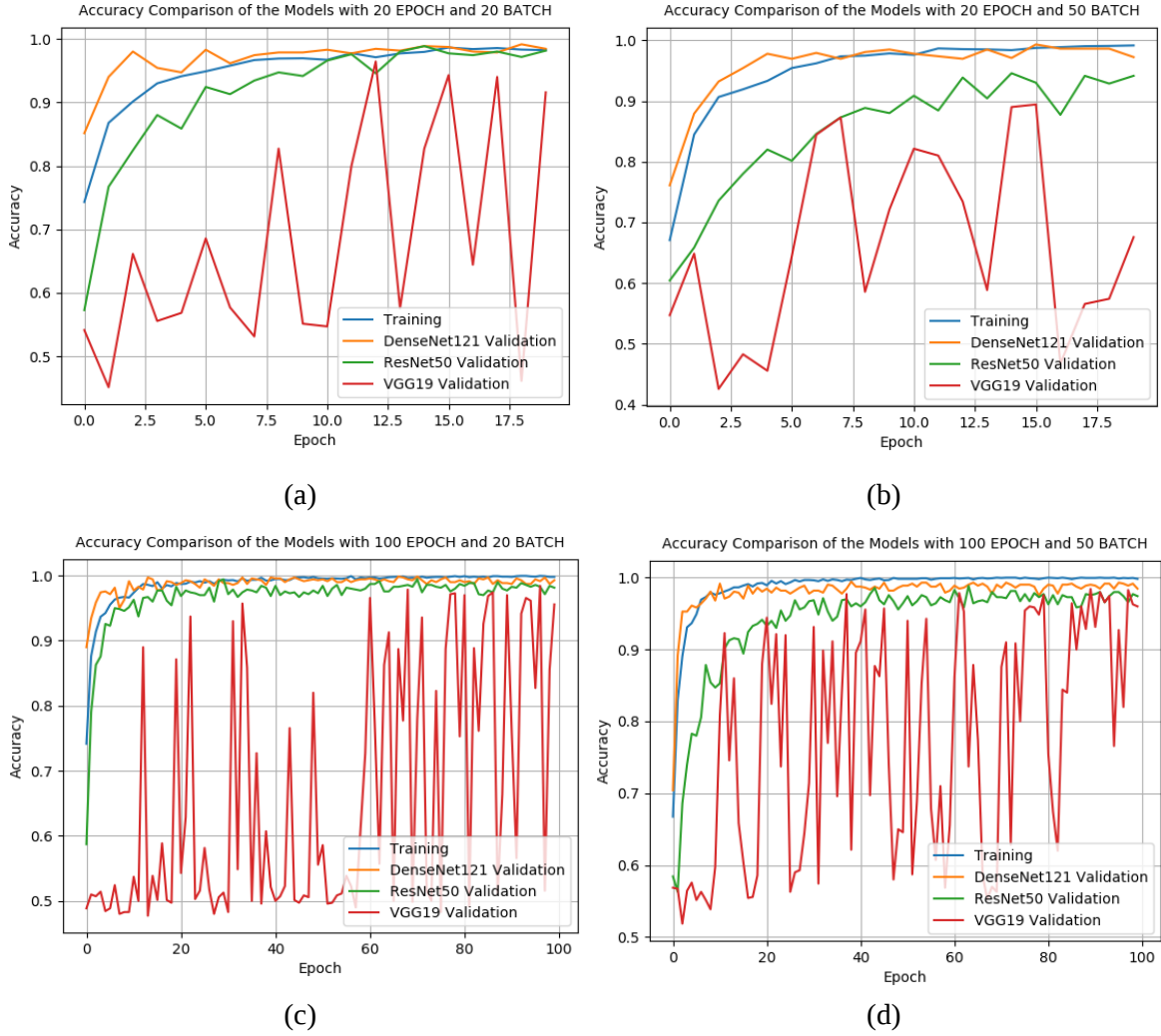
Karşılaştırma Kriterleri		VGGNet19	ResNet50	DenseNet121
Parametre Sayıları	<i>Toplam</i>	46347842	127282562	59151426
	<i>Eğitilebilir</i>	46294594	127025666	58964354
GPU Bellek Kullanımı (MB)	<i>20 Yığın</i>	8372	8414	8384
	<i>50 Yığın</i>	8380	10896	11155
Zaman Kullanımı (ms)	<i>20 Yığın</i>	0,2035	0,1649	0,1667
	<i>50 Yığın</i>	0,4434	0,3473	0,3325

Toplam parametre ve eğitilebilir parametre sayılarına göre en az parametre sayısına sahip model VGGNet19 dur. İkincisi ise DenseNet121 ve ResNet 50 en fazla parametreye sahip modeldir. Modellerin katman sayıları göz önüne alındığında 121 katmana sahip DenseNet modelinin parametre sayısının diğer modellere göre daha optimal seviyede olduğu anlaşılmaktadır. Modellerin bellek kullanım miktarları da aynı biçimde toplam katman sayısı ve parametre sayılarına göre değişmektedir ve görünüşte en az belleği hem 20 hem de 50 yığın için VGGNet19 elde etmiştir. Ancak bellek kullanım performansı en iyi olan modelin de yine DenseNet121 olduğu açıktır çünkü en fazla katmana sahip olmasına rağmen ve hatta VGGNet19 dan daha fazla parametre içermesine rağmen bellek kullanım miktarları neredeyse aynıdır. Modeller 20 ve 50 yığını eğitimde tamamlamak için harcadıkları zamana göre test edilmiştir. Zaman kullanım miktarlarına göre değerlendirildiğinde 20 yığında ResNet50, 50 yığında ise DenseNet121 en iyi zaman kullanım performansını elde etmiştir. Ancak bu iki modelinde zaman kullanım miktarları birbirlerine çok yakındır. Toplam eğitilebilir parametre sayısı göz önüne alındığında ResNet50 modelinin zaman yönetimindeki başarısı öne çıkmaktadır. En kötü zaman yönetimi ise VGGNet19 modeline aittir. Hem katman sayısı hem de eğitilebilir parametre sayısı diğer modellerden daha az olmasına rağmen 20 ve 50 yığın eğitimi için daha fazla zaman harcadığı gözlemlenmiştir. Eğitimde kullanılan veri miktarı arttıkça toplam eğitim zamanının ciddi miktarda artacağı ön görülmekte ve bu sebeple her bir yığın için geçen zamanın önemli olduğunun vurgulanmasında fayda görülmektedir.

Tüm modellerin farklı eğitim adımlarında ve farklı yığınlarda elde ettikleri eğitim ve doğrulama başarılarının ve eğitimleri için geçen toplam sürelerin karşılaştırıldığı deneylere ait sonuçlar Çizelge 8.2’de sunulmuştur. Ayrıca modellerin eğitim başarılarına göre, doğrulama başarılarını daha kolay karşılaştırmak ve anlayabilmek için her bir yığın ve eğitim adımındaki performansları Şekil 8.4’de görsel olarak sunulmuştur.

Çizelge 8.2. Modellere ait her bir eğitim adımı ve yığın için, başarı ve toplam eğitim süresi sonuçları

Karşılaştırma Kriterleri			VGGNet19	ResNet50	DenseNet121
20 Eğitim Adımı	20 Yığın	<i>Eğitim Doğruluğu</i>	0,9470	0,9002	0,9493
		<i>Doğrulama Doğruluğu</i>	0,6784	0,9151	0,9692
		<i>Eğitim Süresi (saniye)</i>	552,310	467,338	473,285
	50 Yığın	<i>Eğitim Doğruluğu</i>	0,9573	0,8504	0,9490
		<i>Doğrulama Doğruluğu</i>	0,6625	0,8543	0,9593
		<i>Eğitim Süresi (saniye)</i>	474,174	389,622	404,862
100 Eğitim Adımı	20 Yığın	<i>Eğitim Doğruluğu</i>	0,9852	0,9733	0,9874
		<i>Doğrulama Doğruluğu</i>	0,9852	0,9733	0,9874
		<i>Eğitim Süresi (saniye)</i>	2787,593	2296,787	2342,325
	50 Yığın	<i>Eğitim Doğruluğu</i>	0,9871	0,9613	0,9881
		<i>Doğrulama Doğruluğu</i>	0,7756	0,9400	0,9804
		<i>Eğitim Süresi (saniye)</i>	2377,463	1911,519	1935,826



Şekil 8.4. Modellerin doğrulama başarılarının karşılaştırma grafikleri; (a) 20 adım ve 20 yığın için, (b) 20 adım ve 50 yığın için, (c) 100 adım ve 20 yığın için, (d) 100 adım ve 50 yığın için modellerin doğruluk başarı grafikleri

Deneysel sonuçlara göre her bir eğitim adımı ve her bir yığın için modellerin başarıları şu şekilde olmuştur; 20 eğitim adımı için 20 yığında gerçekleştirilen deneylerde en iyi eğitim doğruluğu DenseNet121 modeline aittir, VGGNet19 modelinin eğitim doğruluğu ise DenseNet121'e çok yakındır ve en kötü eğitim doğruluğu ResNet50 modeline aittir. Bu formattaki doğrulama aşaması doğruluk değerlerine bakıldığında eğitimde en iyi ikinci sonucu elde eden VGGNet19 modelinin gerçekten çok kötü bir sonuç elde ettiği görülmektedir. ResNet50 ve DenseNet121 modelleri daha kararlı doğruluk sonuçları elde etmiş ve en iyi doğruluk değeri yine DenseNet121 modeline aittir.

20 eğitim adımı değeri sabit tutularak yığın 50'ye çıkarıldığında (yani daha büyük yığınlarla eğitim gerçekleştirildiğinde) en iyi eğitim doğruluğunu VGGNet19, ikincisini DenseNet121

ve sonuncusunu ResNet50 modeli elde etmiştir. Ancak yine doğruluk aşamasındaki başarı VGGNet19 modelinde çok kötüdür. Doğruluk başarısı en iyi model yine DenseNet121 ve ikinci ResNet50 olmuştur.

Deneyler 100 eğitim adımı için tekrar gerçekleştirdiğinde (yani 20 ve 50 yığında daha fazla eğitim gerçekleştirildiğinde) 20 yığın için en iyi eğitim ve doğrulama başarılarını DenseNet121 ikincisini ise VGGNet19 modelleri elde etmiş ancak bu iki modelin değerleri birbirlerine çok yakındır. ResNet50 modeli ise sonuncu olmuştur.

100 eğitim adımında ve 50 yığın ile gerçekleştirilen deneylerde en başarılı eğitim ve doğrulama sonuçları yine DenseNet121'e aittir. 50 yığında VGGNet19, ResNet50'den daha başarılı bir eğitim doğruluğuna ulaşmış olsa da yine doğrulama aşamasındaki başarısı çok kötüdür ve doğrulama aşamasında sonuncu olmuştur.

Zaman yönetimi bakımından modeller karşılaştırıldığında Çizelge 8.1'de yığın temelinde elde edilen sonuçlarla paralel sonuçlar elde edildiği görülmektedir. Yine en iyi yığın temelli zaman yönetimine sahip olan ResNet50 toplam eğitim süresinde de en iyi performansı göstermiştir. Toplam eğitim süresine göre DenseNet121 ikinci ve VGGNet19 ise üçüncü olmuştur. Eğitim zamanı yönetimine göre birinci ve ikinci olan modelleri (ResNet50 ve DenseNet121) incelediğimizde yine daha fazla eğitim parametresine sahip olmasına rağmen ResNet50 modelinin başarılı olması göze çarpmaktadır. Bunun sebebinin modellerin katman sayısı olduğu düşünülmektedir. DenseNet121 daha fazla katmana sahip olduğundan eğitim süresi daha uzun sürmüştür. Bu karşılık en az katmana ve parametreye sahip VGGNet19 modeli eğitim zamanı yönetimi bakımından en kötü performansı elde ederek sınıfta kalmıştır.

Şekil 8.4'de verilen grafikler incelendiğinde en iyi doğrulama sonuçlarını DenseNet121 modelinin, ikinci en iyi sonuçları ise ResNet50 modelinin elde ettiği açıkça görülmektedir. Ayrıca bu iki modelin eğitim sonuçlarına göre her bir eğitim adımında kararlı oldukları açıkça görülmektedir. VGGNet19 modelinin eğitim ve doğrulama başarıları arasındaki uçurum Şekil 8.4'de çok daha net olarak görülmektedir. Modelin eğitimde başarılı ancak doğrulama aşamasında (eğitilirken hiç görmediği veriler üzerinde) oldukça başarısız olmasının temel nedeni ezberlemesi (overfitting) dir. VGGNet19 modeli çevrimdışı el yazısı imza özneteliklerini öğrenemiyor ve ezberliyor. Şekil 8.4'de verilen grafiklerde VGGNet19

modelinin her bir adımında (epoch) sürekli tutarsız doğrulama sonuçları elde etmesi (grafiklerde her bir adımda büyük zikzaklar oluşması) modelin ezberlediği gerçeğini kuvvetlendirmektedir.

8.3. Deney, Sonuç ve Öneriler

Bu bölümde literatürde birçok alanda başarı elde etmiş ve bu tez çalışması kapsamında bir önceki bölümde kullanılan CNN modelleri olan VGGNet19, ResNet50 ve DenseNet121 modellerinin GPU üzerindeki performans karşılaştırması gerçekleştirilmiştir. Modellerin 100 ve 20 eğitim adımında, 20 ve 50 yığın parçalarıyla elde ettikleri performanslar ayrı ayrı gerçekleştirilen deneylerle ölçülmüştür. Modeller çevrimdışı el yazısı imza veri kümesi üzerindeki başarılarına, her bir yığındaki eğitim sürelerine, toplam eğitim sürelerine, parametre sayılarına ve GPU üzerinde kullandıkları bellek miktarlarına göre karşılaştırılmıştır.

Deneylerde elde edilen sonuçlara göre en az parametre sayısı VGGNet19 modeline aittir. İkinci DenseNet121 ve en çok parametre sayısı ResNet50 modeline aittir. En iyi bellek kullanımı ise en az parametreye ve katmana sahip olan VGGNet19 modeline aittir. Bu noktada kullanılan parametre sayıları dikkate alındığında en iyi bellek yönetiminin ResNet50 modeline ait olduğu ve ikincinin de DenseNet121 olduğu gözlemlenmiştir. Modeller zaman yönetimi bakımından karşılaştırıldığında her alanda ResNet50 modelinin en başarılı model olduğu görülmüştür. Hem 20 yığın hem 50 yığın hem de toplam eğitim süresi bakımından en hızlı modelin ResNet50 ve en kötü modelinse VGGNet19 olduğu görülmüştür. İkinci en iyi zaman yönetimi DenseNet121 modeline aittir. Ayrıca 50 yığında ResNet50 modelinden daha iyi performans sergilemiştir ancak ResNet50 ve DenseNet121 modellerinin zaman performansları birbirine çok yakındır. Modellerin veri kümesindeki başarıları karşılaştırıldığında en başarılı modelin her aşamada DenseNet121 olduğu görülmüştür. İkinci en başarılı modelse ResNet50'dir. VGGNet19 modeli eğitimlerde çok iyi performans gösterse de hiç görmediği veriler üzerinde oldukça kötü sonuçlar elde etmiş ve doğrulama başarılarının çok kötü olduğu görülmüştür. VGGNet19 modelinin eğitim ve doğrulama başarıları arasındaki bu ciddi farklılığın modelin imza veri kümesinde öznetelikleri öğrenemeyip ezberlemesinden kaynaklı olduğu anlaşılmıştır.

Bu bölümde gerçekleştirilen çalışmada elde edilen bulgular neticesinde imza veri kümesi

üzerinde en başarısız modelin VGGNet19 olduğu görülmüştür. ResNet50 modeli zaman yönetimi ve bellek yönetimi konusunda yüksek başarı elde etse de imza veri kümesindeki doğruluk başarısı DenseNet121'den kötüdür. DenseNet121 ve ResNet50 modelleri arasındaki bellek yönetimi ve zaman yönetimi sonuçlarının çok farklı olmadığı ve neredeyse aynı olduğu, ancak DenseNet121 modelinin doğrulama başarısının daha yüksek olduğu göz önüne alındığında en başarılı modelin DenseNet121 olduğu ortadadır.

Sonuç olarak çevrimdışı el yazısı imza doğrulama görevi için en iyi modelin DenseNet olduğu görülmüştür. Bu modele ait hiper parametreler araştırılır, farklı optimizasyon modelleri ve farklı katman sayılarıyla imza veri kümesi üzerinde elde edeceği sonuçlar incelenirse daha başarılı sonuçlar elde edileceği düşünülmektedir. Bu kapsamda DenseNet modelinin araştırılarak imza veri kümesi üzerinde daha başarılı sonuçlar elde edecek yeni bir DenseNet tabanlı derin öğrenme modeli geliştirilmesi düşünülmektedir. Bu hedef doğrultusunda bir sonraki bölümde DenseNet modeline ait hiper parametre, optimizasyon tekniği ve katman sayısı araştırması gerçekleştirilmiştir.

9. ÇEVİRİMİÇİ İMZA DOĞRLUMA İÇİN DENSENET MODELİNİN ARAŞTIRILMASI

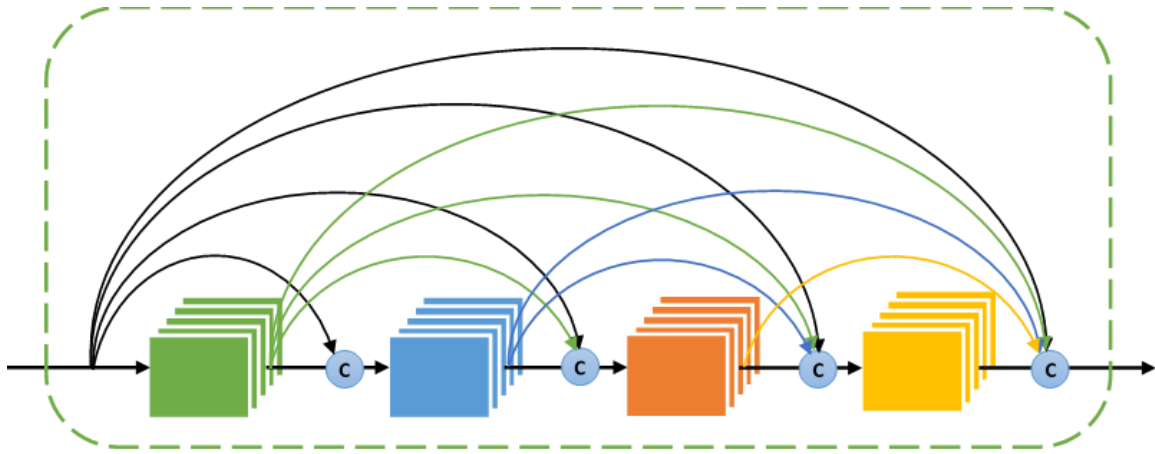
Bir önceki bölümde literatürde birçok alanda başarılı sonuçlar elde etmiş olan CNN modelleri, çevrimdışı el yazısı imza veritabanı üzerinde test edilerek modeller arasında performans karşılaştırması gerçekleştirilmiştir. Elde edilen bulgular neticesinde imza veritabanı üzerinde en başarılı model DenseNet121 olmuştur. Bu bölümde, DenseNet modeline ait katman sayısı, hiper parametre ve optimizasyon tekniği araştırması gerçekleştirilmektedir. En uygun değerleri elde ederek model üzerinde iyileştirme yapmak ve çevrimdışı el yazısı imza veritabanı üzerindeki başarısını daha çok geliştirmek amaçlanmaktadır.

9.1. DenseNet Katman Yapısının Araştırılması

Bu bölümde DenseNet modeline ait farklı katman sayılarının çevrim dışı el yazısı imza veritabanı üzerindeki etkisi araştırılmaktadır. En iyi performansa sahip katman modelinin bulunması amaçlanmaktadır.

Standart bir CNN modelinde her bir evrişim katmanının çıktısı bir sonraki katmana bağlıdır ve böylelikle bir önceki katmanda öğrenilmiş çıktılar yeni katmana girdi olarak verilmektedir. Bu şekilde son katmanlar ilk katmanlardaki temel özellikleri direkt olarak göremezler ve sadece kendinden önceki katmanın özetlediği öznitelik haritasıyla yetinirler. ResNet modelinde ise her katmandan iki önceki evrişim katmanının çıktısı da sonraki katmana ek girdi olarak gönderilir ve katmandan hemen önceki katmanın çıktı verisiyle işlemsel olarak toplanarak yeni katmana girdi olarak verilir. Böylece katmanlar kendinden iki önceki katmanın öznitelik haritası hakkında da bilgi edinmiş olur. DenseNet modelinde ise ResNet modelindeki yapı geliştirilmiş ve her katman kendinden önceki tüm katmanların öznitelik haritasını kullanır hale gelmiştir. Ayrıca önceki katmanlardan aktarılan öznitelik haritaları ResNet'te olduğu gibi yeni katmanın verileriyle toplanarak değil yeni öznitelik haritası olarak öncekilere eklenerek (Concatenate) kullanır. Böylece ilk katmanlardaki temel öznitelikler hiç değiştirilmeden birebir sonraki katmanlara aktarılmış olur.

Birbirlerine tam olarak bağılı yani önceki her katmandan sonraki her katmana bir bağlantının olduğu evrişim katmanlarına dense blokları denir. Şekil 9.1’de bir dense bloğu yapısı görülmektedir. Standart bir DenseNet modeli 4 adet dense bloğundan meydana gelmektedir. Her bir dense bloğunda farklı sayılarda evrişim katmanı bulunmaktadır. Bu evrişim katmanları arasında Batch Normalization (Normalizasyon), Convolution (evrişim) ve Pooling (Havuzlama) katmanları bulunur. DenseNet modelinin yapısı Bölüm 8.1.3’de anlatılmıştır. Bu çalışma kapsamında deneylerde kullanılan DenseNet121 modelindeki katman yapısı [6, 12, 24, 16] şeklindedir. Burada görülen yapı 4 elemanlı bir dizi biçimindedir. Her bir dizi elemanı bir dense bloğu yapısına karşılık gelirken, elemanlara ait değerler ise o dense bloğu içindeki evrişim katmanı sayısını göstermektedir.



Şekil 9.1. DenseNet dense bloğu temel yapısı

Bu aşamada çevrimdışı el yazısı imza veritabanı üzerinde en iyi sonucun alınabileceği evrişim katmanı sayısı ve sıralaması üzerine bir araştırma yapılması planlanmıştır. Farklı katman sayılarındaki DenseNet modellerini imza veritabanı ile test ederek en iyi olanın tespit edilmesi hedeflenmiştir. Gerçekleştirilen deneysel çalışmalarda [4, 6, 8, 10], [2, 4, 6, 8], [2, 4, 10, 8], [16, 4, 10, 8], [16, 6, 8, 10], [16, 32, 8, 10] ve DenseNet121 modelleri olmak üzere 7 farklı DenseNet modeli test edilmiş ve performans karşılaştırması yapılmıştır.

Deneysel çalışmalar GPDSsyntheticSignature veritabanı içinden rastgele olarak seçilen kullanıcıların imzaları üzerinde gerçekleştirilmiştir. Yazara bağımlı (WD) yaklaşıma göre gerçekleştirilen deneylerde seçilen kullanıcının gerçek ve sahte imzaları eğitim ve test aşamalarında kullanılmıştır. Kişinin yetenekli sahte (skilled forgeries) örneklerinin yanında

diğer kullanıcılardan seçilen rastgele sahte (random forgeries) örneklerde eğitim aşamasında kullanılmıştır. Tüm verilerin %70'i eğitim ve %30'u test aşamasında kullanılmıştır. Deneyler her bir seçilen kullanıcının el yazısı imzaları için veri artırma işlemi uygulamadan ve tez kapsamında önerilen veri artırma yöntemi uygulanarak olmak üzere iki farklı şekilde ayrı ayrı gerçekleştirilmiştir. Her bir model için nihai sonuçlar tüm deneylerin ortalaması alınarak elde edilmiştir ve böylece deneylerin tesadüfi etkisi azaltılarak güvenilirliğinin artırılması amaçlanmıştır.

Modellerin karşılaştırılması için öncelikle farklı yığın (batch) sayılarındaki doğruluk oranları (accuracy) test edilmiş ve karşılaştırılmıştır. Bu test aşamasında SGD optimizasyon tekniği, öğrenme oranı 0,0004 olacak şekilde, 0,9 momentum ve binnary_crossentropy yöntemi kullanılmıştır. Deneysel sonuçlar Çizelge 9.1'de verilmiştir.

Çizelge 9.1. DenseNet modellerinin farklı yığın oranlarında doğruluk karşılaştırması

Katmanlar	Eğitilebilir Parametre	Eğitilebilir Parametre	Doğruluk		
			50 Yığın	20 Yığın	15 Yığın
[4, 6, 8, 10]	2 095 330	2 119 410	95%	92%	92%
[2, 4, 6, 8]	1 299 138	1 313 826	89%	-	-
[2, 4, 10, 8]	1 734 466	1 754 274	89%	-	-
[16, 4, 8,10]	3 466 434	3 505 842	-	94%	-
[16, 6, 8,10]	3 692 578	3 734 802	-	98%	93%
[16, 32, 8,10]	9 376 802	9 485 170	-	-	97%
DenseNet121	6 949 634	7 033 282	90%	94%	96%

Modellerin performans karşılaştırması sırasında 50 yığında (batch) bazı modeller bellek hatası verdiği için yığın sayısı azaltılmıştır ve 20 yığında sonuçlar alınmıştır. Aynı şekilde 20 yığında bellek hatası verenler için de 15 yığın sayısında deneyler gerçekleştirilerek sonuçlar elde edilmiştir. Çizelge 9.1 incelendiğinde, elde edilen sonuçlara göre 50 yığında [4, 6, 8, 10] katman modeline sahip DenseNet %95 doğrulukla en iyi sonucu elde etmiştir. Bu modelin 20 ve 15 yığınlarında aynı performansı gösteremediği görülmüştür. 20 yığın ile eğitilen modellerde en iyi doğruluk oranını %98 ile [16, 6, 8,10] ve 15 yığın ile eğitilen modeller de ise %97 ile [16, 32,

8,10] elde etmiştir. [6, 12, 24, 16] katman sayısına sahip olan standart DenseNet121 modeli ise her yığın oranında çalışmasına rağmen diğer modellere göre daha kötü başarı oranına sahiptir. DenseNet121 en iyi sonucu 15 yığında %96 doğruluk oranı ile elde etmiştir.

En başarılı modelleri tespit ettikten sonra ikinci aşamada bu başarılı modeller arasında performans testleri yaparak en uygun katman sayısı araştırmasına devam edilmiştir. Bu aşamada modellerin sadece doğruluk oranlarını değil ayrıca Precision, Recall, F1, FRR, FAR oranları da incelenmiştir. Literatürde bu alanda yapılan çalışmalar incelediğinde eğitim aşamasında sadece gerçek imzaların kullanıldığı ve yetenekli sahtelerin eğitime katılmayarak sadece testlerde kullanıldığı görülmektedir. Bu sebepten bu aşamada yapılan testlerde eğitim için yetenekli sahteler kullanılmayarak sadece diğer kullanıcılardan seçilen rastgele sahte imza örnekleri kullanılmıştır. Ayrıca bu aşamada farklı sayıdaki rastgele sahte imzalarla modelleri test ederek eğitimde kullanılacak optimum rastgele sahte imza örneği sayısı da araştırılmıştır. Modeller üzerinde gerçekleştirilen bu testlerde birden fazla rastgele seçilmiş kişilere ait imzalar kullanılmıştır. Yine nihai sonuçlar her bir testin ortalaması alınarak elde edilmiştir Tüm modelleri eşit şartlarda karşılaştırmak için deneylerin tamamı 15 yığın ile gerçekleştirilmiştir. Çizelge 9.2’de ayrıntılı karşılaştırma tablosu verilmiştir.

Çizelge 9.2. DenseNet modellerinin ayrıntılı karşılaştırması

Modeller	Precision	Recall	F1	FRR	FAR
[16, 4, 8,10]	0,8942	0,8186	0,8524	0,1812	0,1333
[16, 6, 8,10]	0,8914	0,7575	0,8006	0,2478	0,1274
[16, 32, 8,10]	0,8787	0,8779	0,8547	0,1138	0,2421
DenseNet121	0,8457	0,8191	0,8519	0,1808	0,1374

Elde edilen bulgulara göre GPDSsyntheticSignature veritabanı çevrimdışı el yazısı imza örnekleri üzerinde en başarılı sonucu [16, 32, 8,10] katman yapısına sahip model elde etmiştir. Precision oranına göre %89 ile [16, 4, 8,10] ve FAR oranına göre %12 ile [16, 6, 8,10] en iyi performansı elde etmesine rağmen bu modellerin Recall, F1 ve FRR oranlarının düşük olduğu gözlemlenmiştir. Bir sonraki analiz aşamasında en uygun rastgele sahte örnek sayısının tespiti için bu testte en başarılı model olan [16, 32, 8,10] DenseNet modeli kullanılacaktır.

Literatürde çevrimdışı el yazısı imza sınıflandırması için gerçekleştirilen çalışmalar incelendiğinde eğitim aşamasında sadece gerçek verilerin kullanıldığı yetenekli sahte verilerin ise sadece test aşamasında kullanıldığı görülmektedir. Eğitim aşamasında rastgele sahte örnekler olarak adlandırılan ve diğer kullanıcılardan elde edilmiş imza örnekleri kullanılmaktadır. Bu rastgele sahte imza örneklerinin eğitimdeki sayısı sınıflandırma başarısını büyük oranda etkilemektedir. Eğer yetersiz miktarda sahte imza örneği eğitimde kullanılırsa, ağırlıklı olarak gerçek imzaları daha kolay ezberlediği ve hiç görmediği yetenekli sahteleri sınıflandırmada yetersiz kaldığı görülmektedir. Yine aynı şekilde eğer eğitimde kullanılan gerçek örnek sayısının çok üzerinde gereğinden fazla rastgele sahte örneği kullanılırsa ağırlıklı olarak gerçek imzaları göz ardı ettiği ve gerçek imzaları tespit etmede yetersiz olduğu gözlemlenmiştir. Tüm bu sorunların üstesinden gelebilmek adına deneylerde kullanılabilecek rastgele imza sayısının tespit edilmesi amaçlanmıştır. Bu amaç doğrultusunda bir önceki testte en başarılı DenseNet modeli farklı kullanıcıların imzalarıyla ve farklı rastgele sahte imza oranlarıyla ayrı ayrı test edilmiştir. Elde edilen test sonuçları Çizelge 9.3'te ayrıntılı olarak sunulmuştur.

Çizelge 9.3. Rastgele sahte örnek sayılarının performansa etkisi

		Precision	Recall	F1	FRR	FAR
Rastgele Sahte Sayısı	300	0,8815	0,8173	0,8231	0,1539	0,2332
	400	0,8726	0,8777	0,8691	0,1221	0,2110
	500	0,8421	1,000	0,9083	0	0,3333
	800	0,8604	0,9457	0,8969	0,0542	0,2666

Çizelge 9.3'ü analiz etmeye geçmeden önce önemli bir noktanın hatırlanması gerekir. Karmaşıklık matrisindeki Precision, Recall, F1, FRR, FAR gibi değerler hakkında ayrıntılı bilgi Bölüm 3.1.1'de verilmiş ve nasıl hesaplandıklarına dair formüller bu bölümde sunulmuştur. Burada Precision değerinin doğru olarak tahmin edilen gerçek imza örneklerinin sınıflandırma yaparken bu gerçektir denilen imza örneklerine oranı olduğunu, Recall değerinin ise doğru olarak tahmin edilen gerçek imza örneklerinin veri setinde bulunan toplam gerçek imza örneklerine oranı olduğunu belirtmekte fayda var.

Çizelge 9.3'ü incelediğinde Precision değerinin en yüksek olan "Rastgele Sahte Sayısı-RSS"

300 olduğu görülmektedir. 400 RSS değerinde ise en düşük FAR oranının elde edildiği görülmüştür. En iyi Recall, F1 ve FRR oranı ise 500 RSS değerinde elde edilmiştir. Rastgele imza sayısı eğitimde kullanılan gerçek imza sayısından çok az olduğunda sahte imzaları sistemin iyi öğrenemeyerek FAR oranının yükseldiğini ve aynı şekilde rastgele imza sayısının eğitimde kullanılan gerçek imza sayısından çok yüksek olduğunda da FRR oranının yükseldiği görülmektedir. Bu sebepten ötürü GPDSsyntheticSignature veritabanında WD yaklaşımına göre sınıflandırma yaparken RSS değerinin 500 olması gerekmektedir. Bu değer belirlenmesinde veri artırma yöntemimizin de etkisi büyüktür çünkü eğitim verilerinin sayısı orijinal veritabanının sunduğu sayıdan fazladır. Her çalışmada kullanılan eğitim verisi sayısına göre yeniden belirlemelidir.

Çalışmada deneyler Adam ve SGD (Stochastic Gradient Descent) optimizasyon yöntemleriyle farklı öğrenme oranlarında (η) test edilmiştir. Gerçekleştirilen testler neticesinde SGD yönteminin en iyi başarı oranına 0,0004 öğrenme oranıyla ulaştığı görülmüştür. Adam yöntemi ise en iyi performansı 0,004 öğrenme oranında elde etmiştir. SGD ‘nin Adam yönteminden daha başarılı olması sebebiyle çalışmaların genelinde 0,0004 öğrenme oranıyla SGD optimizasyon yöntemi kullanılmıştır.

9.2. Deney, Sonuç ve Öneriler

Bu bölümde, DenseNet modeline ait katman sayısı, hiper parametre ve optimizasyon tekniği araştırması gerçekleştirilmiştir. En uygun değerleri elde ederek model üzerinde iyileştirme yapmak ve çevrimdışı el yazısı imza veritabanı üzerindeki başarısını daha çok geliştirmek amaçlanmıştır.

Elde edilen deneysel sonuçlar neticesinde [16, 32, 8,10] katman yapısına sahip DenseNet modelinin çevrimdışı el yazısı imza veritabanı üzerinde en başarılı sonucu elde ettiği gözlemlenmiştir. Bu modelin üzerinde gerçekleştirilen ikinci analiz ise modelin eğitimin için kullanılan rastgele sahte imza örneği sayısının tespitine yönelik olmuştur. Eğitimde kullanılan rastgele sahte imza sayısı ile gerçek imza sayısı arasındaki oran sistemin başarısını büyük ölçüde etkilemektedir. Eğitimde kullanılan rastgele imza sayısı gerçek imza örneği sayısından çok fazla olduğunda sistemin gerçek imzaları tespit etmekte yetersiz kaldığı ve tam tersi durum gerçekleştiğinde, yani RSS’nin gerçek imza sayısından çok az kaldığı durumlardaysa sistemin sahte imza örneklerini tespit etmede yetersiz kaldığı görülmüştür.

Optimum RSS deęerini bulmak iin gerekleřtirilen deney sonular izelge 9.3’de verilmiřtir. Elde edilen sonulara gre 500 RSS deęerinde sistemin daha doęru sınıflandırma yapabildięi gzlemlenmiřtir. Bu noktada bu oranın eęitimde kullanılan gerek imza sayısına gre farklılıklar gsterebileceęini belirtmek gerekir. Yapılan son analizde sistemde verimli alıřacak optimizasyon teknięi ve bu teknięe ait ęrenme oranı tespit edilmiřtir. Adam ynteminin 0,004 ve SGD ynteminin 0,0004 ęrenme oranında en bařarılı sonuları elde ettięi gzlemlenmiřtir. SGD optimizasyon ynteminin Adam ynteminden daha iyi sonular elde etmesinden dolayı alıřmada SGD kullanılması uygun grlmüştür.

Gerekleřtirilen alıřmada elde edilen modelde 0,9083 gibi yksek bir F1 skoruna ulařıldıęı grlmüştür. Bu deęer sistem sadece gerek imzalarla eęitildięi, yetenekli sahte imzalar sistemin eęitiminde kullanılmadıęı ve sahte imza olarak eęitimde sadece rastgele sahte imza rneęi kullanıldıęı iin dikkate deęerdir. Her ne kadar yksek bir doęruluk oranına ulařsa da sistemin beklenen performansı veremedięi de gzlemlenmiřtir. Bu performans yetersizlięinin sebepleri arařtırıldıęında kullanılan CNN yntemlerinin genelinde var olan ve son yıllarda byk bir aık olarak ta gndeme gelen “Adverserial Attacks” ın bu bařarısızlıęın asıl sebebi olduęu grlmüştür. CNN yntemleri nesnelere ait zellikleri ilk katmanlarda kenar, kře, kıvrım gibi basit ve birbirinden baęımsız znitelikler olarak ęrenmekte ve bunları ilerleyen katmanlarda birbirlerinden baęımsız řekilde birleřtirerek nesnelere ait daha belirgin yapıları nitelemektedirler. Tm bunlar “Adverserial Attacks” denilen saldırıların yapılmasını kolaylařtırmakta ve sistemin kolayca kandırılmasına imkn saęlamaktadır. Her kıvrım, kře ve hatta noktasıyla bir btn olarak deęerlendirilmesi gereken imza gibi kritik neme sahip yapıların sınıflandırmasında bu aık byk bir risk oluřtırmakta ve sistem bařarısını nemli lde etkilemektedir.

Bir sonraki blmde imza rneklelerini bir btn olarak ęrenerek sınıflandırabilecek ve bylece “Adverserial Attacks” aıęının stesinden gelebilecek bir CNN yntemi arařtırılarak evrimdiři el yazısı imza tanıma iin nerilmiřtir. Bylece sistemin evrimdiři el yazısı imza tanıma bařarısının daha da geliřtirilmesi amalanmıřtır.

10. ÇEVİRİMİÇİ EL YAZISI İMZALAR İÇİN DERİN ÖĞRENME TABANLI İMZA DOĞRULAMA YÖNTEMİ

CNN modelleri son yıllarda birçok alanda olduğu gibi imza doğrulama alanında da önemli gelişmelerin ana kaynağı olmuştur. Çalışmada çevrimdışı imza doğrulama için yeni bir CNN tabanlı yöntem araştırması yapılmaktadır. Bu bağlamda gerçekleştirilen araştırmalar gösterdi ki birçok alanda ciddi başarılar gösteren geleneksel CNN yöntemlerinin aynı zamanda ciddi bir dezavantajı da var. Girdi nesnesini bir bütün olarak değerlendirmeyip parçalar halinde alt katmanlara doğru birleştiren geleneksel CNN yöntemleri kolayca kandırılabilir. Bu bölümde, “Adversarial Attacks” açığının üstesinden gelebilecek bir CNN yöntemi araştırılarak çevrimdışı imza doğrulama alanına uygulanması sağlanmaktadır.

Başarılarına rağmen, CNN'lerin de bazı sınırlamaları ve dezavantajları vardır. Her katmanda, CNN'ler sırasıyla kenarlar, şekiller ve son olarak gerçek nesneler gibi farklı özellikleri öğrenir. Bununla birlikte, bu özellikler arasındaki mekânsal ilişkileri (perspektif, boyut, parçalar arası ilişki) öğrenmez ve dikkate almaz. Örneğin, insan yüzü ile eğitilmiş bir CNN düşünün. Bu ağa giriş verileri olarak gözlerin, burnun ve ağzın yerini değiştirerek yeni bir resim verirsek, bunu bir insan yüzü olarak tanıyacaktır. Aynı şekilde öğrenilmiş haritalardan alınmış gürültü tarzı verileri bir görüntüye ekleyerek ya da direkt olarak ağa vererek CNN ağı kolayca kandırılabilir. Bu CNN'lerin en büyük dezavantajlarından biridir. Çünkü geleneksel CNN yöntemleri nesneyi öğrenirken parçaların konumları ve birbirleri arasındaki ilişkileri göz ardı ederler. Böylece, yanlış mekânsal özelliklere sahip görüntüler tarafından kolayca kandırılabilirler. Buna çekişmeli saldırılar (Adversarial Attacks) denir. Sinir ağları için, bu saldırılar ilk olarak Goodfellow ve ark. [130] tarafından 2014 yılında ortaya konulmuştur. Geleneksel CNN yöntemleri giriş katmanından aldıkları girdileri ilk katmanlarda daha basit formlarda öğrenip daha sonra öğrendiği bu bilgileri alt katmanlara özetleyerek daha karmaşık olan şekilleri öğrenirler. Bu işlemler sırasında birbirine tam bağlı katmanlar sebebiyle milyonlarca hatta milyarlarca parametrenin tekrar tekrar güncellenmesi ve öğrenilmesi gerekmektedir. Bu işlemler sistemi yavaşlatmakta ve öğrenme sürecinin uzamasına neden olmaktadır. Bu zaman ve işlem karmaşıklığı sorununun üstesinden gelmek için 1998 yılında LeCun ve ark. [95] tarafından önerilen LeNet-5 bir havuzlama (pooling) katmanı da içermektedir. Havuzlama katmanı önceki katmandan aldığı girdileri örnek azaltma (down sampling) yöntemiyle daha küçük boyutlara indirgemektedir. Böylece veri boyutunu azaltarak performans artışını ve zaman kazancını sağlamaktadır. Ancak veri

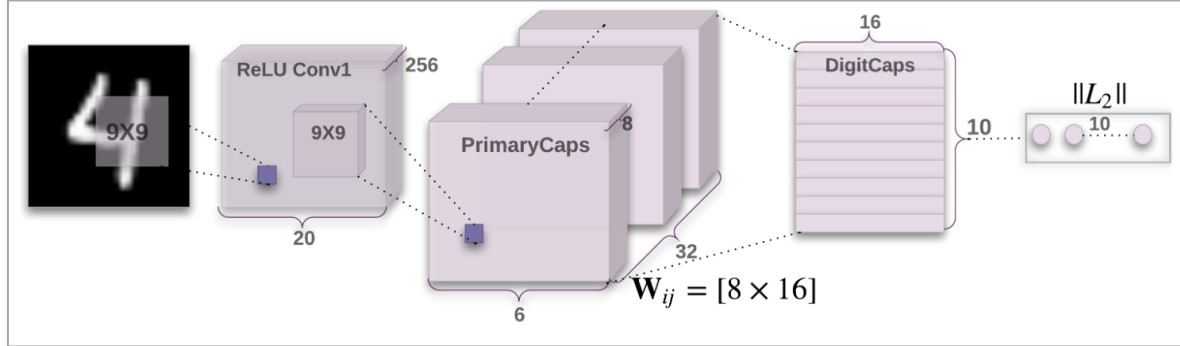
boyutu azaltılırken bazı öz nitelikler de kaybolmaktadır. Özellikle bu öz nitelikler arasındaki mekansal ilişki yok olmaktadır. Bu nedenle de çekişmeli saldırılara karşı korumasız kalmaktadırlar. Yıllarca havuzlama yöntemlerinin CNN yöntemlerine olan başarılı katkısı sebebiyle getirdikleri dezavantajlar görülememiştir. Çekişmeli saldırılara karşı etkili bir yöntem sunan Geoffrey Hinton'ın havuzlama hakkındaki *“The pooling operation used in convolutional neural networks is a big mistake and the fact that it works so well is a disaster”* (Evrışimli sinir ağlarında kullanılan havuzlama işlemi büyük bir hatadır ve bu kadar iyi çalışması bir felakettir) cümlesi her şeyi açıklar niteliktedir.

10.1. Capsule Network Mimarisi

Sinir ağlarına yönelik çekişmeli saldırılarla mücadele etmek için birçok çalışma yapılmıştır. Bu çalışmalardan biri de Frosst ve ark. [131] tarafından önerilen Capsule Network (CapsNet) 'tür. Araştırmacılar CapsNets'in çekişmeli saldırılara karşı diğer mimarilere göre daha güçlü olduğunu göstermişlerdir. Bu çalışmada kullanılan CapsNet modeli Sabour ve ark. [132] tarafından önerilen “Dynamic Routing Between Capsules” adlı makaleyi temel almaktadır. CapsNet modeli sadece temel özellikleri (çizgiler, eğriler ve harfler vb.) değil, aynı zamanda bu özellikler arasındaki mekânsal ilişkileri de dikkate alır. Modelin en büyük avantajı budur. Modelin bir diğer önemli avantajıysa sınıf başına düşen az miktardaki örnekle bile daha hızlı ve doğru bir şekilde öğrenebilmesidir. Bu avantajlar, az veri içeren imza doğrulama sistemleri için hayati önem taşırlar ve bu çalışmada sınıflandırma yöntemi olarak seçilmesinin temelinin oluşturmaktadırlar.

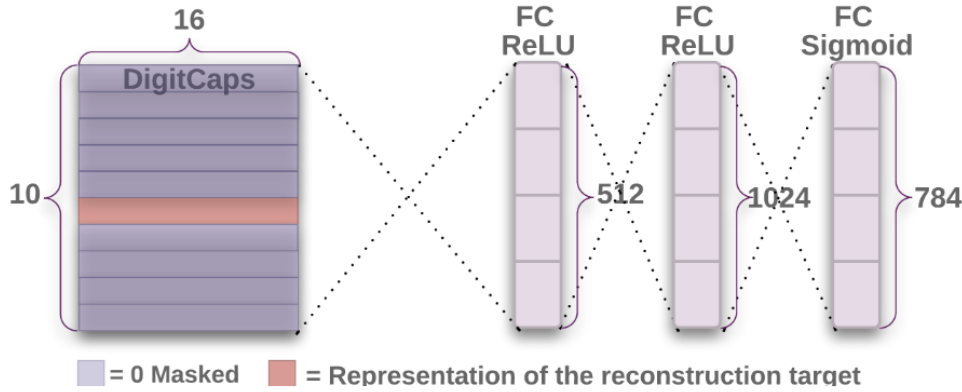
CapsNet modelinde nöronlar, kapsül olarak adlandırılan vektörler halinde gruplandırılır. Kapsüller, bu nöronların çeşitli poz parametrelerini temsil eden aktivite vektörleridir. Varlıklar, kapsül tarafından temsil edilir. Bir kapsülün çıkış vektörünün uzunluğu da o varlığın, mevcut girdide bulunma olasılığını temsil etmektedir. Böylece, mekânsal ilişkilerin daha etkin bir şekilde modellenmesi amaçlanmaktadır. Sabour ve ark. [132] tarafından önerilen makaleden alınmış temel CapsNet mimarisi Şekil 10.1'de görülmektedir. Ağ eğitimi sırasında, doğru rakam kapsülünün aktivite vektörü hariç diğer rakam kapsüllerinin tümü maskelenmektedir. Sonra kalan bu aktivite vektörü girdi görüntüsünü yeniden yapılandırmak için kullanılmaktadır. Sabour ve arkadaşlarının gerçekleştirdiği çalışmada el yazısı rakamların bulunduğu MNIST verilerinin sınıflandırılması üzerinde çalışıldığından model mimarisinde DigitCaps olarak gösterilmiştir. Rakam kapsülünün çıktısı, piksel

yoğunluklarını modelleyen tam bağlı (fully connected) 3 katmandan oluşan bir kod çözücüye beslenmektedir.



Şekil 10.1. Üç katmanlı basit bir CapsNet mimarisi.

Bu şekilde veriler çözümlenerek (decode) yeniden oluşturulmaya çalışılır. Bu yapıya ait mimari Şekil 10.2’de görülmektedir. Yeniden yapılandırma (Reconstructed) aşamasında kullanılan amaç fonksiyonu görüntü ile sigmoid tabakasının çıktısı arasındaki Öklid mesafesini eğitim sırasında en aza indirmeye çalışmaktadır.

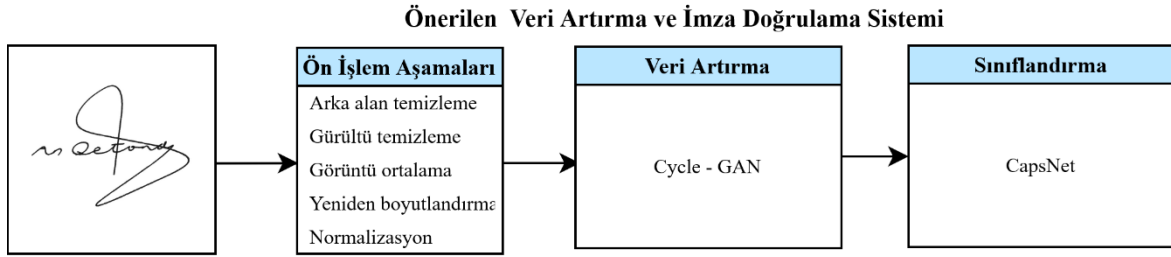


Şekil 10.2. DigitCaps katman temsilinden bir rakamı yeniden oluşturmak için kullanılan çözümleyici yapısı.

Bu tez çalışması kapsamında mimari girdisi olarak çevrimdışı el yazı imza örnekleri kullanılmıştır. CapsNet mimari yapısı bu imza örneklerine göre oluşturulmuştur. Kullanılan CapsNet mimarisine ait katman yapısı ve parametre verileri Çizelge 10.1’de verilmiştir. Çalışmada CapsNet için yönlendirme (Routing) parametresi 3 olarak ayarlanmıştır. Önerilen sistemde eğitim verisi olarak Bölüm 7’da önerilen veri artırma yönteminden elde edilen veriler kullanılmıştır. Bu bağlamda çevrimdışı el yazısı imza doğrulama sistemine ait genel yapı Şekil 10.3’de verilmiştir.

Çizelge 10.1. Önerilen CapsNet modeline ait katman sayıları ve parametreler

Katmanlar	Boyutları	Diğer Parametreler
Input	224 x 224 x 1	
Convolution	222 x 222 x 32	Stride = 1, pad = 0
BatchNormalization	222 x 222 x 32	
AveragePooling	111 x 111 x 32	Stride = 2, pad = 0
Convolution	107 x 107 x 16	Stride = 1, pad = 0
Primarycaps	50 x 50 x 336	Stride = 2, pad = 0
ReLu	50 x 50 x 336	
BatchNormalization	50 x 50 x 336	
Dropout	50 x 50 x 336	P = 0.01
Primarycaps reshape	52500 x 16	
Primarycaps Lambda	52500 x 16	
Digitcaps	2 x 16	Routing = 3
Decoder	224 x 224 x 1	



Şekil 10.3. Önerilen veri artırma ve imza doğrulama sistemine ait genel yapı

Veritabanlarından elde edilen çevrimdışı imza örnekleri öncelikle Bölüm 5’de anlatılan ön işlem aşamalarından geçirilmektedir. Ön işlem aşamasında dijitalleştirilen veriler doğrulama testinin gerçekleştirileceği senaryoya göre eğitimde kullanılmak üzere orijinal gerçek (Genuine) ve yetenekli sahte (Skilled Forgeries) örnek gruplarına ayrılmaktadır. Daha sonra önerilen veri artırma işlemine göre çoğaltılıp, önerilen imza doğrulama yöntemiyle sınıflandırılmaktadır.

10.2. Deney ve Sonuçlar

Bu bölümde imza yapısını bir bütün halinde değerlendirebilecek bir derin öğrenme tabanlı çevrimdışı imza doğrulama sistemine ait deneysel sonuçlar verilmiştir. Araştırmalar sonucunda CapsNet modelinin yapısının hem girdi verilerini kapsülleyerek girdiyi bir bütün

halinde değerlendirmesi hem de az sayıda eğitim verisiyle bile sınıflandırma yapmaya uygun olması sebebiyle kullanılması uygun görülmüştür. Tez kapsamında önerilen Cycle-GAN tabanlı veri artırma yöntemiyle sınıflandırma yöntemi birleştirilerek Şekil 10.3'te verilen yapıya uygun bir “Çevrimdışı El Yazısı İmza Doğrulama için Derin Öğrenme Tabanlı Veri Büyütme Yöntemi ve İmza Doğrulama Sistemi” önerilmiştir.

Önerilen sisteme göre ve gerçek hayatta uygulanabilirliğe uygun şekilde deneysel çalışmalar gerçekleştirilmiştir. Gerçek hayatta sahte imza (özellikle de yetenekli sahte – Skilled Forgeries) örneklerini bulmak çok zor hatta mümkün olmadığından, eğitim aşamaları sadece gerçek imzalarla gerçekleştirilmiştir. Ayrıca yine gerçek hayatta eğitimde kullanılacak gerçek imza örneklerine ulaşmak sınırlı olduğu için olabildiğince az sayıdaki gerçek imza örnekleriyle eğitim işlemi gerçekleştirilmiştir. Bu bağlamda önerilen veri artırma ve doğrulama yönteminin performansı, çevrimdışı el yazısı imza doğrulama alanında en çok tercih edilen MCYT75 ve GPDSsyntheticSignature veritabanları kullanılarak test edilmiştir. Veritabanlarıyla ilgili ayrıntılı bilgi Kullanılan Veritabanları bölümünde verilmiştir. Veritabanından rastgele seçilmiş sadece 5 adet orijinal gerçek (Genuine) imzayla sistem eğitilmiştir. Ayrıca sistemin performansı yine rastgele seçilmiş sadece 10 adet orijinal gerçek (Genuine) imza ile de eğitilmiş ve test edilmiştir. Eğitimlerde sadece gerçek imzalar kullanılmasına karşın test aşamalarında hem orijinal hem de sahte (yetenekli ve rastgele sahteler – Skilled and Random Forgeries) imzalar kullanılmıştır. Sistemin sınıflandırma performans testleri başlamadan önce Şekil 10.3'te verilen yapıya uygun olarak sırasıyla ön işlem ve veri artırma aşamaları gerçekleştirilmiştir. Doğrulama sistemini yalnızca 5 orijinal gerçek imza ile test etmek için, Çizelge 10.2 'de verilen algoritma adımları uygulanmıştır.

Çizelge 10.2. Önerilen sisteme ait 5 gerçek eğitim verisini artırma algoritması

Sistem Veri Artırma Algoritması Sahte Kodu	
Adım 1	: Kişiye ait eğitimde kullanılacak (senaryoya göre 5 adet) gerçek imza örneklerini oku
Adım 2	: Ön işlem aşamalarını uygula
Adım 3	: FOR (EPOCH _{max})
Adım 4	: FOR (BACH _{max})
Adım 5	: Okunan gerçek imza örnekleri arasından rastgele ikisini seç
Adım 6	: Önerilen veri artırma ağını eğit
Adım 7	: END (İç Döngü)
Adım 8	: IF Üretilen İmza Öznitelikleri > threshold
Adım 9	: Öğrenilen öznitelik haritasına göre yeni imzayı üret ve kaydet
Adım 10	: END (IF)
Adım 11	: END (Dış Döngü)

Senaryoya uygun olarak eğitimde kullanılacak gerçek veriler artırıldıktan sonra kalan gerçek veriler de test aşamasında kullanılmak üzere eğitim verilerinden ayrı bir şekilde artırılmıştır. Beş gerçek imza eğitim senaryosuna göre geriye kalan gerçek veriler MCYT-75 veritabanı için 10 ve GPDSsyntheticSignature veritabanı için 19 adettir. Sistemin 10 adet gerçek imza örneğiyle veritabanları üzerinde gösterdiği performansın değerlendirilmesi için tüm işlem adımları tekrar uygulanmıştır. Tüm veri büyütme işlemleri tamamlandıktan sonra, önerilen imza doğrulama sistemi sırasıyla 5 ve 10 gerçek artırılmış imza örneği ile eğitilmiştir ve böylece, sistemin imza doğrulama performansı tüm örnekler için ayrı ayrı elde edilmiştir.

Deneyler sonucunda önerilen imza doğrulama sisteminden elde edilen sonuçlar literatürdeki diğer çalışmalarla karşılaştırılmıştır. Araştırmacıların literatürde değerlendirme için iki klasik hata tipini ölçüt olarak kullandıkları belirlenmiştir. Birincisi, Tip I hatası veya FRR, ve ikincisi Tip II hatası veya FAR 'dır. Bu nedenle, literatürdeki diğer çalışmalarla karşılaştırmayı mümkün kılmak için bu tez çalışması kapsamında da bu metrikler kullanılmıştır. Bu metriklere ek olarak EER metriği, AER metriği ve F1 puanı da çalışmada sunulmuştur. Bu metriklere ait ayrıntılı bilgi Bölüm 3.1.1'de verilmiştir.

Tez kapsamındaki diğer çalışmalarda yapılan deneylerde olduğu gibi yine deneylerde öğrenme oranı $Lr = 0.0001$, bozunma (decay) = $1e-6$, momentum = 0.9, nesterov = True olarak ayarlanmıştır. Stokastik gradyan iniş (SGD) optimizasyon yöntemi olarak ve İkili Çapraz Entropi (Binary Cross Entropy) kayıp fonksiyonu olarak kullanılmıştır. Tüm deneyler UBUNTU işletim sisteminde PYTHON dili kullanılarak kodlanmış ve KERAS çatısında Tensorflow kütüphanesi kullanılarak grafik kartı (GPU) üzerinde gerçekleştirilmiştir. Deneylerde kullanılan sistem özellikleri 12GB DDR5 belleğe sahip NVIDIA TITAN XP grafik kartı, I7 CPU, 32GB DDR3 RAM ve 500GB SSD sabit bellekten oluşmaktadır.

Deneyler sonucunda elde edilen sonuçlar her iki veritabanı için ayrı ayrı literatürdeki çalışmalarla karşılaştırılmıştır. Ne yazık ki, literatürdeki çalışmalar birkaç farklı değerlendirme metriği kullandıkları için bazı çalışmalarla karşılaştırma yapmak çok zordur. Bazı çalışmalar FAR, FRR, AER metrikleriyle sonuçları rapor ederken bazı çalışmaların ise FAR, FRR, EER metrikleriyle sonuçları rapor ettiği gözlemlenmiştir. Bazı çalışmalarda ise

FAR ve FRR sonuçları bile bulunmamakta sadece EER sonucu bulunmaktadır. Bu nedenle, tüm çalışmalarla tam bir karşılaştırma yapmak zor olmaktadır. GPDSsyntheticSignature veritabanıyla 5 ve 10 adet gerçek imzayla yapılan deneylerde elde edilen sonuçlar ve literatür karşılaştırmaları Çizelge 10.3'te verilmiştir. Ayrıca MCYT-75 veritabanıyla 5 ve 10 adet gerçek imza örnekleriyle yapılan deneylerde elde edilen sonuçlar ve literatür karşılaştırmaları da Çizelge 10.4'te verilmiştir.

Çizelge 10.3. Önerilen sistemde GPDSsyntheticSignature veritabanı üzerinde elde edilen sonuçlar ve literatür karşılaştırması

Eğitim İmzaları	Yöntemler	FRR (%)	FAR (%)	EER (%)	AER (%)	F1 Skoru (%)
10G	DDML [16]	6,10	24,76	15,64	15,43	-
10G	DMML [16] *	6,51	18,23	12,80	12,37	-
10G	GA+SVM [133]	12,5	3,33	-	7,92	-
10G	SVMs+USMG-SVM [134]	-	-	6,13 (0,25)	-	-
10G	Önerilen Yöntem	10,41	8,66	12,34($\pm 0,2$)	9,53	88,97
5G	GA+SVM [133]	8,33	10	-	9,16	-
5G	Önerilen Yöntem	28,56	7,66	22,93($\pm 0,2$)	18,11	78,23

* Bu yöntem deneylerde hem WI hem de WD yaklaşımını kullanırken diğer çalışmalar sadece WD yaklaşımını kullanmıştır

Çizelge 10.4. Önerilen sistemde MVYT-75 veritabanı üzerinde elde edilen sonuçlar ve literatür karşılaştırması

Eğitim İmzaları	Yöntemler	FRR (%)	FAR (%)	EER (%)	AER (%)	F1 Skoru (%)
10G	Local and Global [135]	-	-	9,28	-	-
10G	[136]	22,93	22,04	20,0	22,48	-
10G	DDML [16]	21,93	4,18	11,58	13,05	-
10G	DMML(WD+WI) [16]*	6,13	12,71	9,86	9,42	-
10G	DRT+PCA+PNN [44]	-	-	9,87	-	-

Çizelge 12.4. (devamı) Önerilen sistemde MVYT-75 veritabanı üzerinde elde edilen sonuçlar ve literatür karşılaştırması

10G	SigNet (SVM) [35]	-	-	2,87($\pm$ 0,42)	-	-
10G	GA+SVM [133]	6,25	5,67	-	5,96	-
10G	HOCCNN [137]	4,92	5,99	-	5,46	
10G	SVMs+USMG-SVM [134]	-	-	2,93 (0,40)	-	-
10G	CNN [138]	8,00	16,80	6,84	12,40	-
10G	MCS [138]	4,00	17,24	3,91	10,62	-
10G	Inkball [139]	5,60	12,89	3,02	9,24	
10G	Önerilen Yöntem	1,33	2,66	2,58($\pm$ 0,43)	1,99	98,06
5G	Local and Global [135]	-	-	11,00	-	-
5G	[136]	32,4	26,84	22,4	29,62	-
5G	DDML [16]	25,15	4,89	14,01	13,02	-
5G	DMML(WD+WI) [16]*	14,80	12,44	13,44	13,62	-
5G	DRT+PCA+PNN [44]	-	-	13,86	-	-
5G	SigNet (SVM) [35]	-	-	3,58($\pm$ 0,54)	-	-
5G	GA+SVM [133]	6,67	6,67	-	6,67	-
5G	HOCCNN [137]	5,83	6,36	-	6,10	
5G	Önerilen Yöntem	7,32	5,33	8,95($\pm$ 0,47)	6,32	93,64

* Bu yöntem deneylerde hem WI hem de WD yaklaşımını kullanırken diğer çalışmalar sadece WD yaklaşımını kullanmıştır

Önerilen imza doğrulama yöntemi MCYT-75 veritabanında GPDSsyntheticSignature veritabanına göre daha iyi performans göstermiştir. GPDSsyntheticSignature veritabanındaki örnekler yapay olarak üretildiği ve MCYT-75 veritabanındaki örnekler ise orijinal olarak insanlar tarafından üretildiği için yöntemin MCYT-75 üzerindeki başarısı önemlidir. Önerilen yöntem, GPDSsyntheticSignature veritabanında 10 orijinal gerçek (Genuine – (G)) örnek ile eğitildiği deneylerde literatürdeki en gelişmiş sonuçlara ulaşamasa da %12,34 ($\pm$ 0,2) EER ve %88,97 F1 Skoru ile en iyi ikinci sonucu elde etmiştir. Ayrıca

GPDSsyntheticSignature veritabanında sadece WD yaklaşımıyla elde edilen bu sonucun WI ve WD yaklaşımlarının birleşiminden oluşan hibrit bir yöntem olan DMML'den de daha iyi olduğu gözlemlenmiştir. Literatürde GPDSsyntheticSignature veritabanında 5 orijinal gerçek (Genuine – (G)) örnek ile gerçekleştirilmiş sadece bir çalışma olduğu gözlemlenmiştir. GA + SVM [133] yöntemini kullanan bu çalışmada sonuç olarak sadece FRR ve FAR sonuçlarını yayınlamış ve EER yayınlamamıştır. Bu nedenle, bu çalışmayla elde edilen sonuçların ayrıntılı biçimde karşılaştırılması oldukça zordur. GA + SVM yönteminin FRR değerinin daha iyi olduğu ancak FAR değerinde ise bu tez kapsamında önerilen yöntemin daha iyi olduğu görülmüştür.

Çizelge 10.4 'te verilen sonuçlara göre, MCYT-75 veritabanında, tez kapsamında önerilen yöntem 10G yani 10 gerçek imza örneğiyle en iyi literatür sonucunu elde etmiştir. Önerilen yöntem MCYT-75'te 10 gerçek imza örneği için %2,58 ($\pm 0,43$) EER ile en iyi EER değerini elde etmiştir. Elde edilen bu değer yine hem WI hem de WD yaklaşımlarıyla birlikte eğitilen hibrit yaklaşımdan daha iyidir. Önerilen yöntemin 5 gerçek imza örneğiyle ise literatürdeki diğer çalışmalarla rekabet halinde olduğu gözlemlenmiştir.

Bu bölümde gerçekleştirilen çalışmanın detayları, elde edilen bulgular ve öneriler “*Deep Learning-Based Data Augmentation Method and Signature Verification System for Offline Handwritten Signature*” isimli makalede ayrıntılı olarak verilmiştir [140].

11. SONUÇ VE ÖNERİLER

Günümüz yüksek teknolojisine rağmen el yazısı imzalar geçmişten günümüze halen en çok kullanılan biyometrik doğrulama yöntemi olmaya devam etmektedir. Sağlık, hukuk, vergi ve tapu gibi birçok alanda yaygın şekilde kullanılan imzalar kişilere ciddi yükümlülükler getirmektedir. Bu nedenle de imzaların güvenliğinin sağlanması, sahte ya da gerçek olup olmadıklarının belirlenmesi önemlilik arz etmektedir. Bu tez çalışması kapsamında çevrimdışı el yazısı imzaların tespitine yönelik derin öğrenme temelli bir sınıflandırma yöntemi ve sınıflandırma yönteminin başarısını artıracak bir veri artırma yöntemi geliştirilmiştir.

Çevrimdışı el yazısı imzalara ait en önemli iki problem olan yetersiz eğitim verisi sorununa ve sınıflandırma sorununa derin öğrenme tabanlı yaklaşımlarla çözüm getirilmiştir. Kişiye ait imza örnekleri çoğaltılarak sınıflandırma başarısı artırılmıştır. Çalışma kapsamında veri artırma için Cycle-GAN tabanlı derin öğrenme yöntemi kullanırken, sınıflandırma sürecinde CapsNet tabanlı yaklaşım kullanılmıştır. Çalışmada öncelikle, geliştirilen veri artırma yönteminin başarısı literatürde birçok alanda başarıları kanıtlanmış dört CNN yöntemi üzerinde test edilmiştir. Bu test sırasında GPDSsyntheticSignature veri kümesi kullanılmıştır. Daha sonra geliştirilen veri artırma yöntemiyle birlikte sınıflandırma yönteminin başarısı imza doğrulama alanında yaygın olarak kullanılan GPDSsyntheticSignature ve MCYT-75 veri kümeleri üzerinde ayrı ayrı test edilerek, elde edilen sonuçlar literatürle karşılaştırılmıştır.

Önerilen veri artırma yönteminin test sürecinde GPDSsyntheticSignature veri kümesinden rastgele seçilen kişilere ait imzalar, kullanıcı bağımsız yaklaşıma göre ayrı ayrı test edilmiş ve nihai sonuç ortalama alınarak elde edilmiştir. Veri kümesinde her kullanıcıya ait 24 gerçek ve 30 sahte imza örneği bulunmaktadır. Sahte ve gerçek örnekler önerilen veri artırma yöntemiyle ayrı ayrı çoğaltılmıştır. Deneyler her bir kişiye ait örnek için VGGNet16, VGGNet19, DenseNet121 ve ResNet50 modelleri üzerinde gerçekleştirilmiştir. Her bir model öncelikle hiçbir veri artışı yapılmadan veri kümesindeki verilerle yani toplam 54 veriyle test edilmiştir. Daha sonra kişiye ait veriler literatürde yaygın olarak kullanılan ve KERAS çerçevesi içerisinde de bulunan genel veri artırma yöntemleriyle 5 kat oranında artırılarak deneyler gerçekleştirilmiştir. Son olarak ta tez kapsamında önerilen derin

öğrenme tabanlı veri artırma yöntemiyle kişiye ait veriler artırılmış ve her bir model için deneyler tekrarlanarak sonuçlar elde edilmiştir. Elde edilen sonuçlara ait karşılaştırmalar tablosu Çizelge 7.2’de verilmiştir. Ayrıca sonuçlar görselleştirilerek Şekil 7’te verilmiştir. Tüm karşılaştırma işlemleri test başarıları (test accuracy), test kayıpları (test loss) ve skor (score) değerlerine göre yapılmıştır.

Elde edilen sonuçlar incelendiğinde önerilen veri artırma yönteminin tüm modellerin sınıflandırma başarısını artırdığı görülmektedir. Önerilen yöntem en başarılı sonucu, Skor değerini %80,00 ‘den %96,80 ‘e yükselttiği görülen DenseNet121 modelinde elde etmiştir. Hiçbir veri artırma yapılmayan ve sadece geleneksel veri artırma yöntemleriyle veri artırma işlemi yapılan birinci ve ikinci deneylere ait sonuçlar incelendiğinde en başarılı test doğruluk değerine sırasıyla VGGNet19 ve VGGNet16 modellerinin ulaştığı görülmektedir. Ancak Şekil 7’te verilen eğitim ve test doğruluk oranlarının modeller bazında karşılaştırıldığı grafiklerde açıkça görülüyor ki VGG modellerinin eğitim ve test doğrulukları arasında büyük farklar bulunmaktadır. Eğitimde daha başarılı olan bu modeller hiç görmedikleri veriler üzerinde çok daha düşük sonuç elde etmişlerdir. Bu da modellerin öğrenmekten çok ezberlediğini göstermektedir. DenseNet121 ve ResNet50 modellerinin daha iyi öğrendiği görülmüştür.

CapsNet tabanlı sınıflandırma sistemi Şekil 10.3’ te gösterilen yapıya uygun olarak önerilen veri artırma yöntemiyle birlikte iki farklı imza veri kümesi üzerinde test edilmiştir. Eğitim süreci her veri kümesi için ayrı ayrı sadece 5 gerçek ve 10 gerçek imzalarla gerçekleştirilmiştir. Eğitimde yetenekli sahteler kullanılmamış ve sadece rastgele sahte imzalar kullanılmıştır. Test sürecindeyse sahte imzalar (rastgele sahte + yetenekli sahte) ve gerçek imzaların tamamı kullanılmıştır. Deneylerde elde edilen sonuçlar, literatürdeki diğer çalışmalarla karşılaştırmanın mümkün olabilmesi için FRR, FAR, EER, AER ve F1 Skor değerlerine göre verilmiştir. Elde edilen sonuçlar ve literatür karşılaştırmaları GPDSsyntheticSignature verit kümesi için Çizelge 10.3’te, MCYT-75 veri kümesi için Çizelge 10.4’te verilmiştir.

Önerilen imza doğrulama sistemi gerçek imza örneklerinden oluşan MCYT-75 veri kümesinde sentetik örneklerden oluşan GPDSsyntheticSignature veri kümesine göre daha iyi performans göstermiştir. Önerilen yöntemin, GPDSsyntheticSignature veri kümesinde

10G ‘de literatürdeki en gelişmiş sonuca ulaşamasa da %12,34 ($\pm 0,2$) EER ve %88,97 F1 Skoru ile en iyi ikinci sonucu elde ettiği görülmüştür. Literatürde GPDSsyntheticSignature veri kümesinde 5G ile gerçekleştirilmiş sadece bir çalışma olduğu gözlemlenmiştir. Bulunan bu çalışmada sonuç olarak sadece FRR ve FAR değerleri yayınlamış ve EER yayınlamamıştır. Çalışmada elde edilen FRR değerinin bu çalışmada önerilen sistemden daha iyi olduğu ancak FAR değerinin daha kötü olduğu görülmüştür.

MCYT-75 veri kümesinde, tez kapsamında önerilen yöntem 10G ‘de %2,58 ($\pm 0,43$) EER ile en iyi literatür sonucunu elde etmiştir. Elde edilen bu değer literatürdeki hem WI hem de WD yaklaşımlarını birlikte kullanarak eğitilen hibrit yaklaşımdan da daha iyi olduğu görülmüştür. Önerilen yöntemin 5 gerçek imza örneğiyle ise literatürdeki diğer çalışmalarla rekabet halinde olduğu gözlemlenmiştir.

Sonuç olarak tez çalışması kapsamında çevrimdışı el yazısı imza doğrulama alanındaki imza verilerinin yetersiz olması problemine karşı etkili bir veri artırma yöntemiyle çözüm getirilmiştir. Ayrıca imza örneklerini kapsülleyerek bir bütün halinde değerlendiren ve WD temelli bir yaklaşımla sınıflandırma yapan yeni bir imza doğrulama yöntemi önerilmiştir. Önerilen bu doğrulama yöntemi literatürdeki en iyi sonucu elde ederek katkı sağlamıştır.

Gelecek çalışmalarda önerilen sistemin başarısının artırılması için derin öğrenme yaklaşımının katmanlarından elde edilen özniteliklerin kullanılması amaçlanmaktadır. Elde edilen veriler üzerinde veri indirgeme uygulanarak sınıflandırma başarısına en çok katkı yapan öznitelikler tespit edilebilir. Elde edilen bu öznitelikler kullanılarak yeni bir sınıflandırma yönteminin eğitilmesiyle daha başarılı sonuçların elde edileceği düşünülmektedir.

KAYNAKLAR

1. Alvarez, G., Sheffer, B. and Bryant, M. (2016). Offline signature verification with convolutional neural networks. *Technical report, Stanford University*.
2. Doroz, R., Porwik, P. and Orczyk, T. (2016). Dynamic signature verification method based on association of features with similarity measures. *Neurocomputing*, 171, 921–931
3. Porwik, P., Doroz, R. and Orczyk, T. (2016). Signatures verification based on PNN classifier optimised by PSO algorithm. *Pattern Recognition*, 60, 998–1014.
4. Impedovo, D. and Pirolo, G. (2008). Automatic signature verification: The state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(5), 609–635..
5. Liwicki, M., Malik, M. I., Heuvel, C. E., Chen, X., Berger, C., Stoel, R., Found, B. (2011). Signature Verification Competition for Online and Offline Skilled Forgeries (SigComp2011). In *2011 International Conference on Document Analysis and Recognition*.
6. Liwicki, M., Malik, M. I., Alewijnse, L., Heuvel, E. and Found, B. (2012). ICFHR 2012 Competition on Automatic Forensic Signature Verification (4NsigComp 2012). In *2012 International Conference on Frontiers in Handwriting Recognition*.
7. Malik, M I, Liwicki, M., Alewijnse, L., Ohyama, W., Blumenstein, M. and Found, B. (2013). ICDAR 2013 Competitions on Signature Verification and Writer Identification for On- and Offline Skilled Forgeries (SigWiComp 2013). In *2013 12th International Conference on Document Analysis and Recognition*.
8. Fierrez, J., Ortega-Garcia, J., Ramos, D. and Gonzalez-Rodriguez, J. (2007). HMM-based on-line signature verification: Feature extraction and signature modeling. *Pattern Recognition Letters*, 28(16), 2325–2334.
9. Muramatsu, D., Kondo, M., Sasaki, M., Tachibana, S. and Matsumoto, T. (2006). A Markov chain Monte Carlo algorithm for Bayesian dynamic signature verification. *IEEE Transactions on Information Forensics and Security*, 1(1), 22–34.
10. Gruber, C., Gruber, T., Krinninger, S. and Sick, B. (2010). Online signature verification with support vector machines based on LCSS kernel functions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40(4), 1088–1100.
11. Fahmy, M. M. M. (2010). Online handwritten signature verification system based on DWT features extraction and neural network classification. *Ain Shams Engineering Journal*, 1(1), 59–70.
12. Pirolo, G., Cuccovillo, V., Impedovo, D. and Mignone, P. (2014). On-line signature verification by multi-domain classification. In *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on Frontiers in Handwriting Recognition*

13. Li, B., Wang, K. and Zhang, D. (2004). On-line signature verification based on pca (principal component analysis) and mca (minor component analysis). In *Biometric Authentication*. Springer.
14. Kudłacik, P. and Porwik, P. (2014). A new approach to signature recognition using the fuzzy method. *Pattern Analysis and Applications*, 17(3), 451–463.
15. Hu, J., Lu, J. and Tan, Y.-P. (2014). Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
16. Soleimani, A., Araabi, B. N. and Fouladi, K. (2016). Deep Multitask Metric Learning for Offline Signature Verification. *Pattern Recognition Letters*, 80, 84–90.
17. Baltzakis, H. and Papamarkos, N. (2001). A new signature verification technique based on a two-stage neural network classifier. *Engineering Applications of Artificial Intelligence*, 14(1), 95–103.
18. El-Yacoubi, A., Justino, E. J. R., Sabourin, R. and Bortolozzi, F. (2000). Off-line signature verification using HMMs and cross-validation. In *Neural Networks for Signal Processing X. Proceedings of the 2000 IEEE Signal Processing Society Workshop (Cat. No. 00TH8501)*, 2, 859–868 .
19. Justino, E. J. R., El Yacoubi, A., Bortolozzi, F. and Sabourin, R. (2000). An off-line signature verification system using HMM and graphometric features. In *Proc. of the 4th international workshop on document analysis systems*.
20. Huang, K. and Yan, H. (1997). Off-line signature verification based on geometric feature extraction and neural network classification. *Pattern Recognition*, 30(1), 9–17.
21. Hafemann, L. G., Sabourin, R. and Oliveira, L. S. (2017). Offline handwritten signature verification; Literature review. In *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*.
22. Oliveira, L. S., Justino, E., Freitas, C. and Sabourin, R. (2005). The graphology applied to signature verification. In *12th Conference of the International Graphonomics Society*.
23. Sabourin, R. and Drouhard, J. (1992). Off-Line Signature Verification Using Directional PDF and Neural Networks. *Proceedings of the 11th ICPR, The Hague*.
24. Drouhard, J.-P., Sabourin, R. and Godbout, M. (1996). A neural network approach to off-line signature verification using directional PDF. *Pattern Recognition*, 29(3), 415–424.
25. Rivard, D., Granger, E. and Sabourin, R. (2013). Multi-feature extraction and selection in writer-independent off-line signature verification. *International Journal on Document Analysis and Recognition (IJDAR)*, 16(1), 83–103.
26. Zhang, B. (2010). Off-line signature verification and identification by pyramid histogram of oriented gradients. *International Journal of Intelligent Computing and Cybernetics*.

27. Sabourin, R. and Genest, G. (1995). An extended-shadow-code based approach for off-line signature verification. ii. evaluation of several multi-classifier combination strategies. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*.
28. Eskander, G. S., Sabourin, R. and Granger, E. (2013). Hybrid writer-independent--writer-dependent offline signature verification system. *IET Biometrics*, 2(4), 169–181
29. Ojala, T., Pietikäinen, M. and Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1), 51–59.
30. Maenpaa, T. I. (2004). The local binary pattern approach to texture analysis: Extensions and applications.
31. Ruiz-Del-Solar, J., Devia, C., Loncomilla, P. and Concha, F. (2008). Offline signature verification using local interest points and descriptors. In *Iberoamerican Congress on Pattern Recognition*.
32. Malik, Muhammad Imran, Liwicki, M., Dengel, A., Uchida, S. and Frinken, V. (2014). Automatic signature stability analysis and verification using local features. In *2014 14th International Conference on Frontiers in Handwriting Recognition*.
33. Bertolini, D., Oliveira, L. S., Justino, E. and Sabourin, R. (2010). Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers. *Pattern Recognition*, 43(1), 387–396.
34. Khalajzadeh, H., Mansouri, M. and Teshnehlal, M. (2012). Persian signature verification using convolutional neural networks. *International Journal of Engineering Research and Technology*, 1(2), 7–12.
35. Hafemann, Luiz G, Sabourin, R. and Oliveira, L. S. (2017). Learning features for offline handwritten signature verification using deep convolutional neural networks. *Pattern Recognition*, 70, 163–176.
36. Yilmaz, M. B. (2015). *Offline signature verification with user-based and global classifiers of local features*.
37. Guerbai, Y., Chibani, Y. and Hadjadji, B. (2015). The effective use of the one-class SVM classifier for handwritten signature verification based on writer-independent parameters. *Pattern Recognition*, 48(1), 103–113.
38. Justino, E. J. R., Bortolozzi, F. and Sabourin, R. (2005). A comparison of SVM and HMM classifiers in the off-line signature verification. *Pattern Recognition Letters*, 26(9), 1377–1385.
39. McCabe, A., Trevathan, J. and Read, W. (2008). Neural network-based handwritten signature verification. *Journal of Computers*, 3, 9–22.
40. Iranmanesh, V., Ahmad, S. M. S., Adnan, W. A. W., Yussof, S., Arigbabu, O. A. and Malallah, F. L. (2014). Online handwritten signature verification using neural network classifier based on principal component analysis. *The Scientific World Journal*.

41. Tolosana, R., Vera-Rodriguez, R., Fierrez, J. and Ortega-Garcia, J. (2018). Exploring Recurrent Neural Networks for On-Line Handwritten Signature Biometrics. *IEEE Access*, 6, 5128–5138.
42. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E. and Shah, R. (1994). Signature verification using a " siamese" time delay neural network. In *Advances in neural information processing systems*.
43. Pansare, A. and Bhatia, S. (2012). Handwritten signature verification using neural network. *International Journal of Applied Information Systems*, 1(2), 44–49.
44. Ooi, S. Y., Teoh, A. B. J., Pang, Y. H. and Hiew, B. Y. (2016). Image-based handwritten signature verification using hybrid methods of discrete Radon transform, principal component analysis and probabilistic neural network. *Applied Soft Computing*, 40, 274–282.
45. Zhang, Z., Liu, X. and Cui, Y. (2016). Multi-phase Offline Signature Verification System Using Deep Convolutional Generative Adversarial Networks. In *2016 9th International Symposium on Computational Intelligence and Design (ISCID)*.
46. Souza, V. L. F., Oliveira, A. L. I. and Sabourin, R. (2018). A writer-independent approach for offline signature verification using deep convolutional neural networks features. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*
47. Sam, S. M., Kamardin, K., Sjarif, N. N. A., Mohamed, N. and others. (2019). Offline Signature Verification using Deep Learning Convolutional Neural Network (CNN) Architectures GoogLeNet Inception-v1 and Inception-v3. *Procedia Computer Science*, 161, 475–483.
48. TDK. İmzanın Tanımı. Retrieved from <https://sozluk.gov.tr/?kelime=imza>
49. Birincioğlu, İ. and Özkara, E. Adli belge incelemelerinde bilinmeyenler, örneklerle yazı ve imza analizi ile ıslak imza kavramı.
50. Sinan Kaymaz. *Çevrimdışı imza tanıma*. Marmara Üniversitesi Fen Bilimleri Enstitüsü.
51. Pascual-Gaspar, J. M., Faundez-Zanuy, M. and Vivascho, C. (2011). Fast on-line signature recognition based on VQ with time modeling. *Engineering Applications of Artificial Intelligence*, 24(2), 368–377.
52. Jain, A. K., Griess, F. D. and Connell, S. D. (2002). On-line signature verification. *Pattern Recognition*, 35(12), 2963–2972.
53. Porwik, P., Doroz, R. and Wrobel, K. (2009). A new signature similarity measure. In *Nature and Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on Nature & Biologically Inspired Computing (NaBIC)*.
54. Tomar, M. and Singh, P. (2011). A directional feature with energy based offline signature verification network. *ArXiv Preprint ArXiv:1103.1205*.

55. Tiwari, D. and Sharma, B. (2012). Development of intelligent network for offline signature verification using pixel density, directional method and both method together. *International Journal of Computer Trends and Technology*, 31(3), 403–411.
56. Justino, E. J. R., Bortolozzi, F. and Sabourin, R. (2001). Off-line signature verification using HMM for random, simple and skilled forgeries. In *Proceedings of Sixth International Conference on Document Analysis and Recognition*.
57. Ferrer, M. A., Alonso, J. B. and Travieso, C. M. (2005). Offline geometric parameters for automatic signature verification using fixed-point arithmetic. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6), 993–997.
58. Frias-Martinez, E., Sanchez, A. and Velez, J. (2006). Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition. *Engineering Applications of Artificial Intelligence*, 19(6), 693–704.
59. Özgündüz, E., Şentürk, T. and Karslıgil, M. E. (2005). Off-line signature verification and recognition by support vector machine. In *Signal Processing Conference, 2005 13th European*.
60. Powers, D. M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation.
61. Ismail, M. A. and Gad, S. (2000). Off-line Arabic signature recognition and verification. *Pattern Recognition*, 33(10), 1727–1740.
62. Al-Omari, Y. M., Abdullah, S. N. H. S. and Omar, K. (2011). State-of-the-art in offline signature verification system. In *2011 International Conference on Pattern Analysis and Intelligence Robotics*.
63. Kisku, D. R., Gupta, P. and Sing, J. K. (2010). Offline signature identification by fusion of multiple classifiers using statistical learning theory. *ArXiv Preprint ArXiv:1003.5865*
64. Khoh, W. H., Ong, T. S., Pang, Y. H. and Teoh, A. B. J. (2014). Score level fusion approach in dynamic signature verification based on hybrid wavelet-Fourier transform. *Security and Communication Networks*, 7(7), 1067–1078.
65. Panton, M. S. (2011). *Off-line signature verification using ensembles of local Radon transform-based HMMs*. Stellenbosch: Stellenbosch University.
66. Jabid, T., Kabir, M. H. and Chae, O. (2010). Local directional pattern (LDP) for face recognition. In *2010 digest of technical papers international conference on consumer electronics (ICCE)*.
67. Ferrer, M. A., Vargas, F., Travieso, C. M. and Alonso, J. B. (2010). Signature verification using local directional pattern (LDP). In *44th Annual 2010 IEEE International Carnahan Conference on Security Technology*.
68. Arica, N. and Vural, F. T. Y. (2003). BAS: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters*, 24(9–10), 1627–1639.

69. Soleimani, A., Fouladi, K. and Araabi, B. N. (2016). Persian offline signature verification based on curvature and gradient histograms. In *2016 6th International Conference on Computer and Knowledge Engineering (ICCCKE)*.
70. Gürbüz, F. and others. (2015). Graphology and Forensic Graphology. *Global Journal on Technology*.
71. Li, S. Z. and Jain, A. (Eds.). (2009). Graphometric Features. In *Encyclopedia of Biometrics*. Boston, MA: Springer US.
72. Burr, D. J. (1988). Experiments on neural net recognition of spoken and written text. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(7), 1162–1168.
73. Sabourin, R., Cheriet, M. and Genest, G. (1993). An extended-shadow-code based approach for off-line signature verification. In *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR'93)*.
74. Parodi, M., Gomez, J. C. and Belad, A. (2011). A circular grid-based rotation invariant feature extraction approach for off-line signature verification. In *2011 International Conference on Document Analysis and Recognition*.
75. Yilmaz, M. B., Yanikoglu, B., Tirkaz, C. and Kholmatov, A. (2011). Offline signature verification using classifier combination of HOG and LBP features. In *2011 international joint conference on Biometrics (IJCB)*.
76. Tustison, N. J., Avants, B. B., Lin, Z., Feng, X., Cullen, N., Mata, J. F., Qing, K. (2018). Convolutional Neural Networks with Template-Based Data Augmentation for Functional Lung Image Quantification. *Academic Radiology*.
77. Hafemann, L. G., Sabourin, R. and Oliveira, L. S. (2016a). Analyzing features learned for Offline Signature Verification using Deep CNNs. In *2016 23rd International Conference on Pattern Recognition (ICPR)*.
78. Hafemann, L. G., Sabourin, R. and Oliveira, L. S. (2016b). Writer-independent feature learning for Offline Signature Verification using Deep Convolutional Neural Networks. In *2016 International Joint Conference on Neural Networks (IJCNN)*.
79. Xiong, P., Wang, H., Liu, M., Zhou, S., Hou, Z. and Liu, X. (2016). ECG signal enhancement based on improved denoising auto-encoder. *Engineering Applications of Artificial Intelligence*, 52, 194–202.
80. Nasr-Esfahani, E., Karimi, N., Jafari, M. H., Soroushmehr, S. M. R., Samavi, S., Nallamothu, B. K. and Najarian, K. (2018). Segmentation of vessels in angiograms using convolutional neural networks. *Biomedical Signal Processing and Control*, 40, 240–251.
81. Gao, T., Du, J., Dai, L.-R. and Lee, C.-H. (2017). A unified DNN approach to speaker-dependent simultaneous speech enhancement and speech separation in low SNR environments. *Speech Communication*, 95, 28–39.

82. Deepika, N. and Variyar, V. V. S. (2017). Obstacle classification and detection for vision based navigation for autonomous driving. In *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*.
83. Polson, N. G. and Sokolov, V. O. (2017). Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, 79, 1–17.
84. Jan, B., Farman, H., Khan, M., Imran, M., Islam, I. U., Ahmad, A., Jeon, G. (2017). Deep learning in big data Analytics: A comparative study. *Computers and Electrical Engineering*.
85. Al-Qizwini, M., Barjasteh, I., Al-Qassab, H. and Radha, H. (2017). Deep learning algorithm for autonomous driving using GoogLeNet. In *2017 IEEE Intelligent Vehicles Symposium (IV)*.
86. Akbulut, Y., Şengür, A., Budak, Ü. and Ekici, S. (2017). Deep learning based face liveness detection in videos. In *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*.
87. Du, X., Ang, M. H. and Rus, D. (2017). Car detection for autonomous vehicle: LIDAR and vision fusion approach through deep learning framework. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
88. Gavrilov, A. D., Jordache, A., Vasdani, M. and Deng, J. (2018). Preventing model overfitting and underfitting in convolutional neural networks. *International Journal of Software Science and Computational Intelligence (IJSSCI)* , 10(4), 19–28.
89. Gavrilov, A., Jordache, A., Vasdani, M. and Deng, J. (2018). Convolutional Neural Networks: Estimating Relations in the Ising Model on Overfitting. In *2018 IEEE 17th International Conference on Cognitive Informatics and Cognitive Computing (ICCI* CC)*.
90. Ferrer, M. A., Diaz-Cabrera, M. and Morales, A. (2013). Synthetic off-line signature image generation. In *Biometrics (ICB), 2013 International Conference on*.
91. Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., Gonzalez, J., Faundez-Zanuy, M., Espinosa, V., ...others. (2003). MCYT baseline corpus: a bimodal biometric database. *IEE Proceedings-Vision, Image and Signal Processing*, 150(6), 395–401.
92. Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66.
93. Yapıcı, M. M., Tekerek, A. and Topaloglu, N. (2018). Convolutional neural network based offline signature verification application. In *2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*.
94. LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E. and Jackel, L. D. (1990). Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*.
95. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.

96. Moccia, S., Momi, E. De, Hadji, S. El and Mattos, L. S. (2018). Blood vessel segmentation algorithms — Review of methods, datasets and evaluation metrics. *Computer Methods and Programs in Biomedicine*, 158, 71–91.
97. Yapıcı, M. M., Tekerek, A. and Topaloğlu, N. (2019). Literature Review of Deep Learning Research Areas. *Gazi Mühendislik Bilimleri Dergisi (GMBD)*.
98. Kooi, T., Litjens, G., van Ginneken, B., Gubern-Mérida, A., Sánchez, C. I., Mann, R., Karssemeijer, N. (2017). Large scale deep learning for computer aided detection of mammographic lesions. *Medical Image Analysis*, 35, 303–312.
99. Pereira, S., Pinto, A., Alves, V. and Silva, C. A. (2016). Brain tumor segmentation using convolutional neural networks in MRI images. *IEEE Transactions on Medical Imaging*, 35(5), 1240–1251.
100. Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88.
101. Nguyen, V. D., Nguyen, H. Van, Tran, D. T., Lee, S. J. and Jeon, J. W. (2017). Learning Framework for Robust Obstacle Detection, Recognition, and Tracking. *IEEE Transactions on Intelligent Transportation Systems*.
102. Uçar, A., Demir, Y. and Güzeliş, C. (2016). Moving towards in object recognition with deep learning for autonomous driving applications. In *2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*.
103. Sun, X., Shi, J., Liu, L., Dong, J., Plant, C., Wang, X. and Zhou, H. (2018). Transferring deep knowledge for object recognition in Low-quality underwater videos. *Neurocomputing*.
104. de San Roman, P. P., Benois-Pineau, J., Domenger, J.-P., Paclet, F., Cataert, D. and de Ruy, A. (2017). Saliency Driven Object recognition in egocentric videos with deep CNN: toward application in assistance to Neuroprostheses. *Computer Vision and Image Understanding*, 164, 82–91.
105. Ciregan, D., Meier, U. and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*.
106. Chen, L., Wang, S., Fan, W., Sun, J. and Naoi, S. (2015). Beyond human recognition: A CNN-based framework for handwritten character recognition. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*.
107. Al-Jawfi, R. (2009). Handwriting Arabic character recognition LeNet using neural network. *Int. Arab J. Inf. Technol*, 6(3), 304–309.
108. Zheng, S., Zeng, X., Lin, G., Zhao, C., Feng, Y., Tao, J., ... Xiong, L. (2016). Sunspot drawings handwritten character recognition method based on deep learning. *New Astronomy*.

109. Li, D. and Wang, Z. (2017). Video Superresolution via Motion Compensation and Deep Residual Learning. *IEEE Transactions on Computational Imaging*, 45, 54–59.
110. Rao, Y., Lu, J. and Zhou, J. (2017). Attention-Aware Deep Reinforcement Learning for Video Face Recognition. *IEEE International Conference on Computer Vision*.
111. Ma, M., Marturi, N., Li, Y., Leonardis, A. and Stolkin, R. (2018). Region-sequence based six-stream CNN features for general and fine-grained human action recognition in videos. *Pattern Recognition*, 76, 506–521.
112. Alghyaline, S., Hsieh, J. W. and Chuang, C. H. (2017). Video action classification using symmelets and deep learning. In *2017 IEEE International Conference on Systems, Man and Cybernetics (SMC)*.
113. Ijjina, E. P. and Chalavadi, K. M. (2017). Human action recognition in RGB-D videos using motion sequence information and deep learning. *Pattern Recognition*, 72, 504–516.
114. Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*.
115. O’Shea, K. and Nash, R. (2015). An Introduction to Convolutional Neural Networks. *ArXiv E-Prints*.
116. Perez, L. and Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. *ArXiv Preprint ArXiv:1712.04621..*
117. Li, X., Shang, W. and Wang, S. (2018). Text-based crude oil price forecasting: A deep learning approach. *International Journal of Forecasting*.
118. Mansour, S. (2018). Social Media Analysis of User’s Responses to Terrorism Using Sentiment Analysis and Text Mining. *Procedia Computer Science*, 140, 95–103.
119. Mostafa, M. M. (2013). More than words: Social networks’ text mining for consumer brand sentiments. *Expert Systems with Applications*, 40(10), 4241–4251.
120. Lv, J.-J., Shao, X.-H., Huang, J.-S., Zhou, X.-D. and Zhou, X. (2017). Data augmentation for face recognition. *Neurocomputing*.
121. Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J. and Greenspan, H. (2018). GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing*, 321, 321–331.
122. Isola, P., Zhu, J.-Y., Zhou, T. and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. *ArXiv Preprint*.
123. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*.

124. Mao, X., Li, Q., Xie, H., Lau, R. Y. K., Wang, Z. and Smolley, S. P. (2017). Least squares generative adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*.
125. Zhu, J.-Y., Park, T., Isola, P. and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. *ArXiv Preprint*.
126. Yapıcı, M. M., Tekerek, A. and Topaloğlu, N. (2019). Performance Comparison of Convolutional Neural Network Models on GPU. In *2019 IEEE 13th International Conference on Application of Information and Communication Technologies (AICT)*.
127. Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *ArXiv Preprint ArXiv:1409.1556*..
128. He, K., Zhang, X., Ren, S. and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.
129. Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *CVPR*.
130. Goodfellow, I. J., Shlens, J. and Szegedy, C. (2014). Explaining and harnessing adversarial examples. *ArXiv Preprint ArXiv:1412.6572*.
131. Frosst, N., Sabour, S. and Hinton, G. (2018). DARCCC: Detecting adversaries by reconstruction from class conditional capsules. *ArXiv Preprint ArXiv:1811.06969*
132. Sabour, S., Frosst, N. and Hinton, G. E. (2017). Dynamic routing between capsules. In *Advances in neural information processing systems*.
133. Sharif, M., Khan, M. A., Faisal, M., Yasmin, M. and Fernandes, S. L. (2018). A framework for offline signature verification system: Best features selection approach. *Pattern Recognition Letters*.
134. Masoudnia, S., Mersa, O., Araabi, B. N., Vahabie, A.-H., Sadeghi, M. A. and Ahmadabadi, M. N. (2019). Multi-representational learning for offline signature verification using multi-loss snapshot ensemble of CNNs. *Expert Systems with Applications*, 133, 317–330.
135. Fierrez-Aguilar, J., Alonso-Hermira, N., Moreno-Marquez, G. and Ortega-Garcia, J. (2004). An off-line signature verification system based on fusion of local and global information. In *International workshop on biometric authentication*.
136. Alonso-Fernandez, F., Fairhurst, M. C., Fierrez, J. and Ortega-Garcia, J. (2007). Automatic measures for predicting performance in off-line signature. In *2007 IEEE International Conference on Image Processing*.
137. Shariatmadari, S., Emadi, S. and Akbari, Y. (2019). Patch-based offline signature verification using one-class hierarchical deep learning. *International Journal on Document Analysis and Recognition (IJDAR)*.

138. Maergner, P., Pondenkandath, V., Alberti, M., Liwicki, M., Riesen, K., Ingold, R. and Fischer, A. (2019). Combining Graph Edit Distance and Triplet Networks for Offline Signature Verification. *Pattern Recognition Letters*.
139. Maergner, P., Howe, N. R., Riesen, K., Ingold, R. and Fischer, A. (2019). Graph-based offline signature verification. *ArXiv Preprint ArXiv:1906.10401*.
140. Yapıcı, M. M., Tekerek, A. and Topaloğlu, N. (2020). Deep learning-based data augmentation method and signature verification system for offline handwritten signature. *Pattern Analysis and Applications*, 1-15.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : YAPICI, Muhammed Mutlu
 Uyruğu : T.C.
 Doğum tarihi ve yeri : 1985, Ankara
 Medeni hali : Bekar
 Telefon : 0312 863 55 78
 e-posta : mutluyapici@outlook.com.tr



Eğitim

Derece	Eğitim Birimi	Mezuniyet Tarihi
Doktora	Gazi Üniversitesi / Bilgisayar Mühendisliği	Devam Ediyor
Yüksek lisans	Gazi Üniversitesi / Elektronik-Bilgisayar Eğitimi	2013
Lisans	Ankara Üniversitesi / Bilgisayar Mühendisliği	2016
Lisans	Gazi Üniversitesi / Bilgisayar Sistemleri Öğretmenliği	2010
Lise	Balgat Mesleki ve Teknik Anadolu Lisesi	2004

İş Deneyimi

Yıl	Yer	Görev
2014-Halen	Ankara Üniversitesi	Öğretim Görevlisi

Yabancı Dil

İngilizce

Yayınlar

1. Yapıcı, M. M., Tekerek, A. and Topaloğlu, N. (2020). Deep learning-based data augmentation method and signature verification system for offline handwritten signature. *Pattern Analysis and Applications*, 1-15.

Hobiler

Bilgisayar oyunları, Yüzmek



GAZİ GELECEKTİR..