

**T.C.
ERCIYES ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**DERİN ÖĞRENME İLE İNSAN EDİMLERİNİN
TANINMASI**

**Hazırlayan
Tayyip ÖZCAN**

**Danışman
Doç. Dr. Alper BAŞTÜRK**

Doktora Tezi

**Ocak 2020
KAYSERİ**

**T.C.
ERCIYES ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**DERİN ÖĞRENME İLE İNSAN EDİMLERİNİN
TANINMASI**

**Hazırlayan
Tayyip ÖZCAN**

**Danışman
Doç. Dr. Alper BAŞTÜRK**

Doktora Tezi

**Ocak 2020
KAYSERİ**

BİLİMSEL ETİĞE UYGUNLUK

Bu çalışmadaki tüm bilgilerin, akademik ve etik kurallara uygun bir şekilde elde edildiğini beyan ederim. Aynı zamanda bu kural ve davranışların gerektirdiği gibi, bu çalışmanın özünde olmayan tüm materyal ve sonuçları tam olarak aktardığımı ve referans gösterdiğimi belirtirim.

Tayyip ÖZCAN

İmza:



YÖNERGEYE UYGUNLUK

“Derin Öğrenme ile İnsan Edimlerinin Tanınması” adlı Doktora tezi, Erciyes Üniversitesi Lisansüstü Tez Önerisi ve Tez Yazma Yönergesi’ne uygun olarak hazırlanmıştır.



Tezi Hazırlayan
Tayyip ÖZCAN



Danışman
Doç. Dr. Alper BAŞTÜRK



Bilgisayar Mühendisliği ABD Başkanı
Prof. Dr. Veysel ASLANTAŞ

Doç. Dr. Alper BAŞTÜRK danışmanlığında **Tayyip ÖZCAN** tarafından hazırlanan “**Derin Öğrenme ile İnsan Edimlerinin Tanınması**” adlı bu çalışma, jürimiz tarafından Erciyes Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında **Doktora** tezi olarak kabul edilmiştir.

29/01/2020


JÜRİ :

- Danışman : Doç. Dr. Alper BAŞTÜRK
Üye : Prof. Dr. Mehmet Emin YÜKSEL
Üye : Prof. Dr. Engin AVCI
Üye : Doç. Dr. Muhammed Fatih TALU
Üye : Doç. Dr. Celal ÖZTÜRK

Basturk
M. Emin Yücel
E. Avcı
Fatih Talu
Celal Öztürk

ONAY :

Bu tezin kabulü Enstitü Yönetim Kurulunun 04/02/2020 tarih ve 2020/05 sayılı kararı ile onaylanmıştır.


04/02/2020
Prof. Dr. Mehmet AKKURT
Enstitü Müdürü

TEŞEKKÜR

"Derin Öğrenme ile İnsan Edimlerinin Tanınması" konulu tez çalışmamın gerçekleştirilmesinde desteklerini esirgemeyen danışman hocam Doç. Dr. Alper BAŞTÜRK'e teşekkürü bir borç bilirim.

Optimizasyon konusundaki başarı ve bilgisiyle bizlere örnek olan Prof. Dr. Derviş KARABOĞA hocama, birlikte yol aldığımız öğretim elemanı dostlarıma teşekkür ederim.

Hayatım boyunca desteğini esirgemeyen ve varlığı sırtımda bir dağ olan babam Mustan ÖZCAN başta olmak üzere, babannem Alime ÖZCAN, abim Nadir ÖZCAN, kardeşlerim Emine Büşra ve Hamdi Buğra ÖZCAN, amcalarım Sami ve Sadettin ÖZCAN'a tüm kalbimle teşekkür ederim.

Başarıma inancı hep var olan yol arkadaşım kıymetli eşim İclal ÖZCAN'a ve hayata sınıksız sarılmamı, her türlü zorlukla mücadelede güçlü olmamı sağlayan yaşam kaynağım canım kızım Emine Defne ÖZCAN'a sonsuz teşekkür ederim.

Kalbimde bir ömür yaşayacak sevgili annem, seni gururlandırmaya devam edeceğim...

DERİN ÖĞRENME İLE İNSAN EDİMLERİNİN TANINMASI

Tayyip ÖZCAN

Erciyes Üniversitesi, Fen Bilimleri Enstitüsü

Doktora Tezi, Ocak 2020

Danışman : Doç. Dr. Alper BAŞTÜRK

ÖZET

Edim; hareket, eylem kelimelerinin eş anlamlısı olarak kullanılabilirken bir diğer anlam olarak ise insan bilincinin bir amaca yönelik davranışlarını ve eylemlerini ifade eden terimdir. İnsan edimlerinin yorumlanması, veri anlamlandırma çalışmalarının önemli bir aşamasını oluşturmaktadır.

İnsan edimlerini tanıma, sadece insanların günlük hareketlerinin tanınmasını kapsamaz. El hareketlerinden işaret dili tanıma, yüz mimik hareketlerinden duygu tanıma ve dudak hareketlerinden dudak okuma insan edimlerini tanıma uygulamaları olarak kabul edilebilir.

Gelişen teknoloji ile derin öğrenme yöntemleri literatürdeki kullanımını yoğunlaştırmıştır. Genellikle başarılı sonuçların elde edilmesini sağlayan derin öğrenme metodlarından evrişimli sinir ağı (convolutional neural network, CNN) ve yığılanmış özdevinimli kodlayıcılar (stacked autoencoders, SAE) insan hareketlerini tanımak için sıklıkla kullanılmaktadır. CNN ve SAE metodlarının başarılarını etkileyen önemli faktörlerden birisi bu metotlara ait hiperparametrelerin belirlenmesidir. Bu parametrelerin değer seçiminin kullanıcı deneyimine bırakılması zamansal kayba sebebiyet vermekle beraber optimum parametre seçiminin yapılabilmesi oldukça güçtür. Bu nedenlerden dolayı, bu tez çalışmasında CNN ve SAE metodlarının hiperparametrelerini belirlemede global ve sezgisel optimizasyon algoritmaları kullanılmıştır. Grid arama (grid search, GS) ve rastgele arama (random search, RS) yöntemleri global arama yöntemleri olarak kullanılırken, yapay arı koloni algoritması (artificial bee colony, ABC), genetik algoritma (genetic algorithm, GA), diferansiyel gelişim algoritması (diferential evolution algorithm, DGA), parçacık sürü optimizasyonu (particle swarm optimization, PSO) algoritması ve bu tez çalışmasında geliştirilen hibrit bir algoritma (hAbcPso) sezgisel arama yöntemleri olarak kullanılmıştır. CNN metodunun kullanıldığı çalışmalarda yeni bir CNN mimarisinin kurulmasının yanında transfer öğrenme destekli CNN metodları

sunulmuştur. Böylelikle, ön eğitilmiş CNN modelleri kullanılarak daha etkili ve kolay bir CNN mimarisi kurulumu sağlanmıştır.

Bu tez çalışmasının amacı, derin öğrenme yöntemlerinin işaret dili tanıma, insan günlük hareketi tanıma, duygu tanıma ve dudak okuma çalışmaları üzerindeki performanslarını araştırmak ve geliştirmektir. Performansların iyileştirilmesi için veri hazırlama (data preparation), imge ön işleme (image preprocessing), transfer öğrenme (transfer learning), hiperparametre optimizasyonu (hyperparameter optimization) gibi yöntemlerin bir veya bir kaçını ilgilendiren çalışmalarda uygulanmış ve birleşik yöntemler önerilmiştir.

Tez çalışmasının birinci kısmında derin öğrenme ile işaret dili tanıma gerçekleştirilmiştir. Bu amaçla rakam ve harf düzeyinde işaret dili tanıma ve hastane acil servisleri için kelime düzeyinde işaret dili tanıma uygulamaları geliştirilmiştir. Yapılan deneysel çalışmalarda harf ve rakam düzeyinde işaret dili tanıma amacıyla kullanılan veri setleri için literatürde bilinen en iyi sonuçlar elde edilmiştir. Hastane ortamında işaret dili tanıma için kelime düzeyinde yeni bir veri seti geliştirilmiş ve derin öğrenme metodu tabanlı önerilen yöntemle deneysel çalışmaları gerçekleştirilmiştir.

Tez çalışmasının ikinci kısmında derin öğrenme ile günlük hareket tanıma uygulamaları geliştirilmiştir. Birinci uygulamada, hiperparametre optimizasyonu destekli SAE tabanlı geliştiren yöntemle sensör tabanlı veriler üzerinden insan hareketlerinin sınıflandırılması sağlanmıştır. Birini dışarıda bırak çapraz doğrulama (leave one out cross validation, LOOCV) test tipi kullanılarak yapılan çalışmalarda UCI Harf veri seti için literatürde bilinen en iyi başarı oranı elde edilirken WISDM veri seti için literatürde bilinen en iyi sonuca yakın bir doğruluk oranı değeri elde edilmiştir. Bu uygulama için geliştirilen hAbcPso hibrit algoritması destekli SAE'nin diğer optimizasyon algoritması destekli SAE yapılarına göre daha başarılı sonuçlar verdiği istatistiksel analizlerle desteklenmiştir. Tezin bu kısmında yapılan diğer uygulamada statik imge tabanlı veriler üzerinden hareket tanıma çalışması gerçekleştirilmiştir. ABC ve transfer öğrenme destekli CNN mimarisi tabanlı önerilen yöntemle Stanford 40 veri seti üzerinde literatürde bilinen en iyi başarı oranı elde edilmiştir.

Tez çalışmasının üçüncü kısmında derin öğrenme ile duygu tanıma uygulamaları gerçekleştirilmiştir. Yapılan ilk çalışmada öncelikle yeni bir veri seti oluşturulmuştur. Bu

veri seti için veri toplama işlemi internet arama motorları yardımı ile gerçekleştirilmiştir. Bu uygulamada, veri ön işleme adımlarının ve veri çoğullama (data augmentation, DA) işlemlerinin sınıflandırma başarımı üzerindeki etkisi analiz edilmiştir. Diğer çalışmada genel kullanıma açmak üzere ERUFER adında yeni bir veri seti oluşturulmuştur. Bu veri seti 10 sınıf içermekte ve katılımcılar gönüllülerden oluşmaktadır. Veri ön işleme, ResNet 18, ve hiperparametre optimizasyonu destekli CNN tabanlı önerilen yöntemle ERUFER ve JAFFE veri setleri üzerinde deneysel çalışmalar gerçekleştirilmiştir. ERUFER veri seti için başarılı sonuçlar elde edilmiştir. Diğer taraftan JAFFE veri seti için literatürde bilinen en iyi başarı oranı elde edilmiştir.

Tezin son kısmında derin öğrenme ile dudak okuma uygulaması geliştirilmiştir. Bu çalışmada, AvLetters veri setinde yer alan zaman serisi veriler bir araya getirilerek tekil imge haline getirilmiştir. Veri çoğullama adımları uygulanarak veri seti genişletilmiş ve başarı oranı artırılmıştır. Yapılan deneysel çalışmalarla başarılı sonuçlar elde edilip literatürde yer alan en başarılı yöntemlerle kıyaslanabilir duruma getirilmiştir.

Anahtar Kelimeler: Derin öğrenme, Evrişimli sinir ağları, Yığınlanmış özdevimli kodlayıcılar, Görüntü işleme, Akıllı sezgisel optimizasyon algoritmaları, Yapay arı koloni algoritması, Genetik algoritma, Diferansiyel gelişim algoritması, Parçacık sürü optimizasyon algoritması, hAbcPso hibrit algoritma, Küresel arama algoritmaları, Grid arama, Rastgele arama, İnsan hareketlerini tanıma, Yüz ifadelerinden duygu tanıma, El işaret dili tanıma, Dudak okuma.

HUMAN ACTION RECOGNITION USING DEEP LEARNING

Tayyip ÖZCAN

Erciyes University, Graduate School of Natural and Applied Sciences

Ph.D. Thesis, January 2020

Supervisor: Assoc. Prof. Dr. Alper BAŞTÜRK

ABSTRACT

Action can be used as a synonym for movement and activity words. As another meaning, it is the term that expresses the behavior and actions of human consciousness for a purpose. Interpretation of human actions constitutes an important stage of data interpretation studies.

Recognition of human actions does not only include recognition of the daily movements of people. Sign language recognition from hand movements, emotion recognition from facial gesture movements and lip reading applications from lip movements can be considered as recognition of human actions.

Deep learning methods have intensified its use in the literature by using developing technology . Convolutional neural networks (CNN) and stacked autoencoders (SAE) are used often to recognize human actions. One of the important factors affecting the success of CNN and SAE methods is the determination of hyperparameters of these methods. Leaving the selection of these parameters to the user experience causes time loss and also it is very difficult to make the optimum parameter selection. For this reason, global and intelligent heuristic optimization algorithms were used to determine the hyperparameters of CNN and SAE methods in this thesis. Grid search (GS) and random search (RS) methods are used as global search methods. On the other hand, artificial bee colony algorithm (ABC), genetic algorithm (GA), differential evaluation algorithm (DE), particle swarm optimization algorithm (PSO) and a novel developed hybrid algorithm (hAbcPso) were used as heuristic search methods. Despite a new CNN architecture was developed for one of the studies in this thesis, transfer learning supported CNN methods are preferred for other CNN based experiments. In this way, a more effective and easier development of CNN architecture is provided by using pre-trained CNN models.

The aim of this thesis is to investigate and improve the performance of deep learning methods on sign language recognition, human daily action recognition, emotion

recognition and lip reading studies. In order to improve performance, one or more of the methods such as data preparation, image preprocessing, transfer learning, hyperparameter optimization have been applied in related studies and combined methods have been proposed.

In the first part of the thesis, sign language recognition were realized by using deep learning. For this purpose, digit and letter level sign language recognition was developed first. Then word level sign language recognition for using hospital emergency services was developed. To the best of our knowledge, the best known results are obtained for the data sets used for letter and digit level sign language recognition. A new word level sign language recognition data set was collected for hospital environment. The experimental studies for the data set were conducted with the proposed method based on the deep learning method.

In the second part of the thesis, daily action recognition applications by using deep learning were developed. In the first application, the SAE-based proposed method supported by hyperparameter optimization provides the classification of human actions on sensor-based data. The best success rate in the literature for UCI Har data set was obtained for leave one out cross validation test type, while the accuracy of the WISDM data set was close to the best result. The hAbcPso hybrid algorithm supported SAE, which is developed for this application, gave more successful results than the other optimization algorithm supported SAE structures. The results are supported by statistical analyzes. In the other application of this part of the thesis, action recognition study has been performed for static image based data. To the best of our knowledge, the best success rate in the literature was obtained for the Stanford 40 data set by using the proposed method which includes both ABC and transfer learning supported CNN.

In the third part of the thesis, emotion recognition applications based on deep learning were carried out. In the first study, a new data set was created for the first step. Data collection for this data set was performed with the help of internet search engines. In this application, the classification performances of data preprocessing steps and data augmentation operations are analyzed. A new data set named ERUFER was created for public use for the other study. This data set consists of 10 classes and the subjects are

volunteers. Experimental studies were carried out on ERUFER and JAFFE data sets by using the proposed method based on data preprocessing, ResNet 18, and hyperparameter optimization supported CNN. Successful results were obtained for the ERUFER data set. On the other hand, the best success rate in the literature for the JAFFE data set was obtained.

In the last part of the thesis, lip reading application was developed by using deep learning. In this study, the time series data in the AvLetters data set were combined into single images. The data set was expanded by applying data augmentation steps and the success rate was increased. Successful results have been obtained with the experimental studies and made comparable with the most successful studies in the literature.

Keywords: Deep learning, Convolutional neural networks, Stacked autoencoders, Image processing, Intelligent heuristic optimization algorithms, Artificial bee colony algorithm, Genetic algorithm, Differential evaluation algorithm, Particle swarm optimization algorithm, hAbcPso hybrid algorithm, Global search algorithms, Grid search, Random search, Human action recognition, Facial expression recognition, Hand gesture recognition, Lip reading.

İÇİNDEKİLER

DERİN ÖĞRENME İLE İNSAN EDİMLERİNİN TANINMASI

BİLİMSEL ETİĞE UYGUNLUK SAYFASI	i
YÖNERGEYE UYGUNLUK SAYFASI	ii
KABUL VE ONAY	iii
TEŞEKKÜR	iv
ÖZET	v
ABSTRACT	viii
İÇİNDEKİLER	xi
KISALTMALAR	xvi
TABLolar LİSTESİ	xviii
ŞEKİLLER LİSTESİ	xxii
GİRİŞ	1

1. BÖLÜM

LİTERATÜR TARAMASI

1.1. İşaret Dili Tanıma Üzerine Yapılan Çalışmalar	9
1.2. Günlük Hareket Tanıma Üzerine Yapılan Çalışmalar	11
1.3. Duygu Tanıma Üzerine Yapılan Çalışmalar	12
1.4. Dudak Okuma Üzerine Yapılan Çalışmalar	14
1.5. Ön Eğitilmiş Modeller Üzerine Yapılan Çalışmalar	15
1.6. Parametre Optimizasyonu Üzerine Yapılan Çalışmalar	16

2. BÖLÜM

METOTLAR VE MATERYALLER

2.1. Problem Çözümlerinde Kullanılacak Derin Öğrenme Algoritmaları	18
2.1.1. Evrişimli Sinir Ağları	19
2.1.1.1. Transfer Öğrenme	27
2.1.2. Özdevinimli Kodlayıcılar	32
2.2. Problem Çözümlerinde Kullanılacak Optimizasyon Algoritmaları	34

2.2.1. Yapay Arı Koloni Algoritması	34
2.2.2. Genetik Algoritma	37
2.2.3. Diferansiyel Gelişim Algoritması	39
2.2.4. Parçacık Sürü Optimizasyon Algoritması	41
2.2.5. Hibrit PSO Destekli ABC Algoritması	43
2.2.5.1. hABCPSO Algoritması Çalışma Prensipleri	43
2.2.6. Global Arama Teknikleri	45
2.2.6.1. Grid Arama Yöntemi	45
2.2.6.2. Rastgele Arama Yöntemi	46

3. BÖLÜM

DERİN ÖĞRENME İLE İŞARET DİLİ TANIMA

3.1. El Hareketlerini Tanımak İçin Sezgisel Optimizasyon ile Transfer Öğrenme Tabanlı Evrişimli Sinir Ağları	47
3.1.1. Giriş	47
3.1.2. Kullanılan Veri Setleri	49
3.1.2.1. SLDD Veri seti	49
3.1.2.2. TMGRD Veri seti	50
3.1.3. Metodoloji	51
3.1.3.1. Önerilen Yöntem	52
3.1.4. Deneysel Çalışmalar	53
3.1.4.1. Deneysel Kurulum	53
3.1.4.2. Deneysel Sonuçlar	55
3.1.5. Değerlendirme	69
3.2. ERUSLR: Yeni Bir Türkçe İşaret Dili Veri Seti ve Evrişimli Sinir Ağı ile Tanınması	71
3.2.1. Giriş	71
3.2.2. Kullanılan Veri Seti	73
3.2.2.1. ERUSLR Veri seti	73
3.2.3. Metodoloji	75

3.2.4. Deneysel Çalışmalar	78
3.2.4.1. Deneysel Kurulum	78
3.2.4.1.1. Donanımsal Kurulum	78
3.2.4.1.2. Parametre Kurulumu	78
3.2.4.2. Deneysel Sonuçlar	79
3.2.5. Değerlendirme	83

4. BÖLÜM

DERİN ÖĞRENME İLE GÜNLÜK HAREKET TANIMA

4.1. Yeni bir Hibrit Sezgisel Algoritma Kullanarak Derin Öğrenme ve Yapısal Optimizasyon ile Sensör Tabanlı İnsan Hareketi Tanıma	85
4.1.1. Giriş	85
4.1.2. Kullanılan Veri Setleri	86
4.1.2.1. WISDM Veri Seti	87
4.1.2.2. UCI Har Veri Seti	87
4.1.3. Metodoloji	87
4.1.3.1. Önerilen Yöntem	88
4.1.4. Deneysel Çalışmalar	90
4.1.4.1. Deneysel Kurulum	90
4.1.4.2. Deneysel Sonuçlar	91
4.1.5. Değerlendirme	99
4.2. Statik Tabanlı Günlük Hareket Tanımda Öneğitimli CNN Ağları İçin Performans İyileştirilmesi	100
4.2.1. Giriş	100
4.2.2. Kullanılan Veri Seti	101
4.2.2.1. Stanford 40 Veri seti	102
4.2.3. Metodoloji	102
4.2.4. Deneysel Çalışmalar	104
4.2.4.1. ABC Algoritmasının Kontrol Parametre Seçimi	104
4.2.4.2. Donanımsal Kurulum	104
4.2.4.3. AlexNet Tabanlı Yöntem İçin Deneysel Sonuçlar	105

4.2.4.4.	VGG16 Tabanlı Yöntem İçin Deneysel Sonuçlar	105
4.2.4.5.	VGG19 Tabanlı Yöntem İçin Deneysel Sonuçlar	108
4.2.4.6.	GoogLeNet Tabanlı Yöntem İçin Deneysel Sonuçlar	108
4.2.4.7.	ResNet50 Tabanlı Yöntem İçin Deneysel Sonuçlar	111
4.2.4.8.	NASNet-Large Tabanlı Yöntem İçin Deneysel Sonuçlar . . .	111
4.2.5.	Değerlendirme	115

5. BÖLÜM

DERİN ÖĞRENME İLE DUYGU TANIMA

5.1.	Evrişimli Sinir Ağı Kullanarak Statik Görüntü Tabanlı Duygu Tanıma	117
5.1.1.	Giriş	117
5.1.2.	Kullanılan Veri Seti	118
5.1.2.1.	Duygu Tanıma Veri seti	118
5.1.3.	Metodoloji	119
5.1.3.1.	Veri Ön İşleme	119
5.1.4.	Deneysel Çalışmalar	120
5.1.5.	Değerlendirme	123
5.2.	Transfer Öğrenimine ve Hiperparametre Optimizasyonuna Dayalı Evrişimli Sinir Ağları Kullanarak Statik Yüz İfadesi Tanıma	124
5.2.1.	Giriş	124
5.2.2.	Kullanılan Veri Seti	126
5.2.2.1.	ERUFER Veri Seti	126
5.2.2.2.	JAFFE Veri Seti	128
5.2.3.	Metodoloji	129
5.2.3.1.	Veri Ön İşleme	130
5.2.4.	Deneysel Çalışmalar	133
5.2.4.1.	Deneysel Kurulum	133
5.2.4.1.1.	Donanımsal Kurulum	133
5.2.4.1.2.	Parametre Kurulumu	133
5.2.4.2.	Deneysel Sonuçlar	133

5.2.4.2.1. JAFFE Veri Seti için Deneysel Sonuçlar	134
5.2.4.2.2. ERUFER Veri Seti için Deneysel Sonuçlar	135
5.2.5. Değerlendirme	137

6. BÖLÜM

DERİN ÖĞRENME İLE DUDAK OKUMA

6.1. Öneğitilmiş Model Destekli - Desteksiz Evrişimli Sinir Ağları Kullanarak Dudak Okuma	138
6.1.1. Giriş	138
6.1.2. AvLetters Veri seti	140
6.1.3. Deneysel Çalışmalar	141
6.1.3.1. Veri Hazırlama	141
6.1.3.2. Tasarlanan CNN Modeli	143
6.1.3.3. Farklı “minibatch boyutu” Değerlerine Göre Tasarlanan ve Ön Eğitilmiş Model Destekli CNN Karşılaştırılması	143
6.1.4. Değerlendirme	148

7. BÖLÜM

TARTIŞMA, SONUÇ ve ÖNERİLER

7.1. Tartışma ve Sonuçlar	149
7.2. Öneriler	151
KAYNAKLAR	153

8. BÖLÜM

EKLER

ÖZGEÇMİŞ	175
--------------------	-----

KISALTMALAR

CNN	: Evrişimli sinir ağı
AE	: Özdevinimli kodlayıcı
SAE	: Yığınlanmış özdevinimli kodlayıcı
GS	: Grid arama
RS	: Rastgele arama
ABC	: Yapay arı koloni algoritması
GA	: Genetik algoritma
DE	: Diferansiyel gelişim algoritması
PSO	: Parçacık sürü optimizasyon algoritması
hABCPSO	: PSO destekli ABC temelli hibrit algoritma
COA	: Guguk kuşu optimizasyon algoritması
BAO	: Bayesian optimizasyonu
LOOCV	: Birini dışarıda bırak çapraz doğrulama
SVM	: Destek vektör makineleri
LS-SVM	: En küçük kareler destek vektör makineleri
HMM	: Saklı Markov modelleri
HOG	: Yönlü gradyanlar histogramı
PCA	: Temel bileşen analizi
kNN	: K-en yakın komşuluk
STS	: Uzun-zaman betimleyicileri yaklaşımı
DL	: Derin öğrenme
HCI	: İnsan bilgisayar etkileşimi
HAR	: İnsan hareketi tanıma
DHAR	: günlük insan hareketi tanıma
HGR	: El hareketi tanıma
SLR	: İşaret dili tanıma
SLDD	: İşaret dili rakamlar veri seti
ERUSLR	: Erciyes Üniversitesi işaret dili tanıma
TMGRD	: Thomas Moeslund'un mimik tanıma veri seti
ER	: Duygu tanıma

FER	:	Yüz ifadeleri tanıma
ERUFER	:	Erciyes Üniversitesi yüz ifadesi tanıma
LR	:	Dudak okuma
DA	:	Veri çoğullama
RBM	:	Kısıtlı Boltzman makinaları
RNN	:	Tekrarlayan sinir ağı
LSTM	:	Uzun kısa vadeli hafıza
QC-SVRC	:	Kalite kontrol örnekleri ve destek vektör regresyonu
MSE	:	Ortalama karesel hata
SCG	:	Ölçeklenmiş eşlenik gradyan

TABLOLAR LİSTESİ

Tablo 2.1.	CNN eğitim seçenekleri.	26
Tablo 3.1.	SLDD veri setindeki rakamların toplam örnek sayıları.	49
Tablo 3.2.	TMGRD veri setindeki harflerin toplam örnek sayıları.	50
Tablo 3.3.	ABC optimizasyon algoritması kontrol parametreleri.	53
Tablo 3.4.	GA optimizasyon algoritması kontrol parametreleri.	55
Tablo 3.5.	PSO optimizasyon algoritması kontrol parametreleri.	55
Tablo 3.6.	SLDD veri seti üzerinde optimizasyon algoritmaları için 30'ar koşmaya ait doğruluk oranları.	56
Tablo 3.7.	Optimizasyon algoritmaları tarafından elde edilen en iyi hiperparametre değerleri ve bu hiperparametreleri SLDD veri seti için kullanan CNN'lerin başarı oranları.	56
Tablo 3.8.	SLDD veri seti için önerilen yöntemlerin ve mevcut yöntemin performans karşılaştırmaları.	57
Tablo 3.9.	SLDD veri seti için algoritmalara ait hata, standart sapma, hesaplama süresi (sn) ve ortalama fval değerleri.	57
Tablo 3.10.	$T_0 = 0,076805$ eşitliği elde edildiği durumda SLDD veri seti için algoritma karmaşıklık değerleri.	58
Tablo 3.11.	SLDD veri seti için Kolmogorov-Smirnov normalite test sonuçları.	62
Tablo 3.12.	SLDD veri seti için Kruskal-Wallis parametrik olmayan test sonuçları.	62
Tablo 3.13.	SLDD veri seti için Mann-Whitney U parametrik olmayan test sonuçları.	62
Tablo 3.14.	TMGRD veri seti üzerinde optimizasyon algoritmaları için 30'ar koşmaya ait doğruluk oranları.	63
Tablo 3.15.	Optimizasyon algoritmaları tarafından elde edilen en iyi hiperparametre değerleri ve bu hiperparametreleri TMGRD veri seti için kullanan CNN'lerin başarı oranları.	63

Tablo 3.16.	TMGRD veri seti için önerilen yöntemlerin ve mevcut yöntemlerin performans karşılaştırmaları.	64
Tablo 3.17.	TMGRD veri seti için algoritmalara ait hata, standart sapma, hesaplama süresi (sn) ve ortalama fval değerleri.	64
Tablo 3.18.	$T_0 = 0,076805$ eşitliği elde edildiği durumda TMGRD veri seti için algoritma karmaşıklık değerleri.	65
Tablo 3.19.	TMGRD veri seti için Kolmogorov-Smirnov normalite test sonuçları.	69
Tablo 3.20.	TMGRD veri seti için Kruskal-Wallis parametrik olmayan test sonuçları.	69
Tablo 3.21.	TMGRD veri seti için Mann-Whitney parametrik olmayan test sonuçları.	69
Tablo 3.22.	Literatürde sıklıkla kullanılan kelime düzeyinde işaret dili tanıma veri setleri.	71
Tablo 3.23.	ERUSLR veri setinde yer alan sınıflar ve toplam örnek sayıları. . . .	75
Tablo 3.24.	Grid aramada kullanılan parametre değerleri.	78
Tablo 3.25.	Grid arama yönteminde belirlenen parametre değerleri ve hesaplanan doğrulama hataları.	80
Tablo 3.26.	Rastgele arama yönteminde belirlenen parametre değerleri ve hesaplanan doğrulama hataları.	81
Tablo 3.27.	Global arama destekli - desteksiz parametre seçimi ve doğruluk oranları.	83
Tablo 4.1.	Kullanılan optimizasyon algoritmalarının kontrol parametreleri. . . .	90
Tablo 4.2.	WISDM veri seti için gerçekleştirilen 30 koşmaya ait hata değerleri. .	91
Tablo 4.3.	Optimizasyon algoritmaları tarafından seçilen en iyi yapısal parametreler ve WISDM veri setinin ilk gönüllüsü için hesaplanan hatalar.	92
Tablo 4.4.	WISDM veri seti için genel sonuç değerlendirme.	93
Tablo 4.5.	UCI Har veri seti için gerçekleştirilen 30 koşmaya ait hata değerleri. .	94

Tablo 4.6.	Optimizasyon algoritmaları tarafından seçilen en iyi yapısal parametreler ve UCI Har veri setinin ilk gönüllüsü için hesaplanan hatalar.	95
Tablo 4.7.	UCI Har veri seti için genel sonuç değerlendirme.	96
Tablo 4.8.	WISDM veri seti için Kolmogorov-Smirnov normalite test sonuçları.	96
Tablo 4.9.	WISDM veri seti için Kruskal Wallis test istatistikleri.	97
Tablo 4.10.	WISDM veri seti için ortalama sıralama değerleri.	97
Tablo 4.11.	WISDM veri seti için ikili algoritmalarda Mann-Whitney U test sonucu.	97
Tablo 4.12.	UCI Har veri seti için Kolmogorov-Smirnov normalite test sonuçları.	98
Tablo 4.13.	UCI Har veri seti için Kruskal Wallis test istatistikleri.	98
Tablo 4.14.	UCI Har veri seti için ortalama sıralama değerleri.	98
Tablo 4.15.	UCI Har veri seti için ikili algoritmalarda Mann-Whitney test sonucu.	99
Tablo 4.16.	ABC optimizasyon algoritması kontrol parametreleri.	104
Tablo 4.17.	AlexNet ile transfer öğreniminin Stanford 40 veri setindeki başarımı.	106
Tablo 4.18.	VGG16 ile transfer öğreniminin Stanford 40 veri setindeki başarımı. .	107
Tablo 4.19.	VGG19 ile transfer öğreniminin Stanford 40 veri setindeki başarımı. .	109
Tablo 4.20.	GoogLeNet ile transfer öğreniminin Stanford 40 veri setindeki başarımı.	110
Tablo 4.21.	ResNet50 ile transfer öğreniminin Stanford 40 veri setindeki başarımı.	112
Tablo 4.22.	NASNet-Large ile transfer öğreniminin Stanford 40 veri setindeki başarımı.	113
Tablo 4.23.	Ön eğitilmiş modeller için seçilen parametreler ve doğruluk oranları. .	114
Tablo 4.24.	Önerilen ve var olan yöntemlerin performans karşılaştırması.	115
Tablo 5.1.	Duygu tanıma veri seti.	119
Tablo 5.2.	Transfer öğrenme destekli CNN için kullanılan parametreler ve değerleri.	120

Tablo 5.3.	Duygu tanıma veri seti için yöntemlerin ortalama ve en iyi başarımlar oranları.	121
Tablo 5.4.	Literatürde sıklıkla kullanılan FER veri setleri ve özellikleri.	124
Tablo 5.5.	ERUFER veri setindeki sınıflara ait örnek sayıları.	127
Tablo 5.6.	PSO algoritmasının kontrol parametreleri.	133
Tablo 5.7.	JAFFE veri setinde 30 koşma için başarı oranları.	134
Tablo 5.8.	JAFFE veri seti üzerindeki önerilen yöntem ile mevcut çalışmalar arasındaki karşılaştırma.	135
Tablo 5.9.	ERUFER veri setinde 30 koşma için başarı oranları.	136
Tablo 6.1.	Tasarlanan, AlexNet ve GoogLeNet destekli CNN modellerinin başarı karşılaştırması.	145
Tablo 6.2.	Önerilen yöntemin literatürdeki çalışmalarla performans karşılaştırması.	147

ŞEKİLLER LİSTESİ

Şekil 2.1.	Evrişim süreci.	19
Şekil 2.2.	Ortalamaya ortaklama işlemi.	20
Şekil 2.3.	Maksimuma ortaklama işlemi.	20
Şekil 2.4.	Minimuma ortaklama işlemi.	20
Şekil 2.5.	ReLU temel aktivasyon fonksiyonu.	21
Şekil 2.6.	ELU temel aktivasyon fonksiyonu.	22
Şekil 2.7.	Sigmoid temel aktivasyon fonksiyonu.	22
Şekil 2.8.	Örnek bir düzleştirme işlemi.	23
Şekil 2.9.	Dropout düzenleme tekniği.	24
Şekil 2.10.	Dropconnect düzenleme tekniği.	24
Şekil 2.11.	Evrişimli sinir ağı mimarisi.	25
Şekil 2.12.	Transfer öğrenme süreci.	28
Şekil 2.13.	AlexNet mimarisi.	29
Şekil 2.14.	GoogLeNet mimarisi.	30
Şekil 2.15.	VGG mimarisi.	31
Şekil 2.16.	ResNet mimarisi.	31
Şekil 2.17.	NASNet arama alanının şematik diyagramı.	31
Şekil 2.18.	Örnek bir AE mimarisi.	32
Şekil 2.19.	Örnek bir SAE mimarisi.	34
Şekil 2.20.	Kovadaki işçi, gözcü ve kaşif arılar.	35
Şekil 2.21.	ABC akış şeması.	36
Şekil 2.22.	GA akış şeması.	39

Şekil 2.23.	DE akış şeması.	41
Şekil 2.24.	PSO akış şeması.	42
Şekil 2.25.	Grid arama için arama uzayındaki parametre setleri.	45
Şekil 2.26.	Rastgele arama için arama uzayındaki parametre setleri.	46
Şekil 3.1.	SLDD veri seti örnekleri.	50
Şekil 3.2.	TMGRD veri seti örnekleri.	51
Şekil 3.3.	ABC optimizasyon algoritması hiperparametre ayarlayıcı olarak kullanıldığında önerilen yöntemin akış şeması.	54
Şekil 3.4.	AlexNet + ABC destekli CNN metodunun SLDD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.	58
Şekil 3.5.	AlexNet + GA destekli CNN metodunun SLDD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.	59
Şekil 3.6.	AlexNet + PSO destekli CNN metodunun SLDD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.	59
Şekil 3.7.	AlexNet + ABC destekli CNN metodunun SLDD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.	60
Şekil 3.8.	AlexNet + GA destekli CNN metodunun SLDD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.	60
Şekil 3.9.	AlexNet + PSO destekli CNN metodunun SLDD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.	61
Şekil 3.10.	AlexNet + ABC destekli CNN metodunun TMGRD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.	65
Şekil 3.11.	AlexNet + GA destekli CNN metodunun TMGRD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.	66
Şekil 3.12.	AlexNet + PSO destekli CNN metodunun TMGRD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.	66
Şekil 3.13.	AlexNet + ABC destekli CNN metodunun TMGRD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.	67

Şekil 3.14.	AlexNet + GA destekli CNN metodunun TMGRD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.	67
Şekil 3.15.	AlexNet + PSO destekli CNN metodunun TMGRD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.	68
Şekil 3.16.	Üç farklı açıdan alınan örnekler.	74
Şekil 3.17.	ERUSLR veri setinde yer alan gönüllüler.	74
Şekil 3.18.	ERUSLR veri seti için önerilen birleşik yöntem.	76
Şekil 3.19.	Bir örnek üzerinde ön işleme çalışması.	77
Şekil 3.20.	Grid arama destekli modelin karmaşıklık matrisi.	82
Şekil 3.21.	Rastgele arama destekli modelin karmaşıklık matrisi.	82
Şekil 3.22.	Global arama desteksiz modelin karmaşıklık matrisi.	83
Şekil 4.1.	İlgili problem için tasarlanan temsili SAE mimarisi.	88
Şekil 4.2.	Stanford 40 veri setinden örnek kareler.	102
Şekil 4.3.	Önerilen yönteme ait işlem süreci.	103
Şekil 5.1.	Veri ön işleme adımları.	120
Şekil 5.2.	CNN+OV metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.	121
Şekil 5.3.	CNN+YT metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.	122
Şekil 5.4.	CNN+YT+M metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.	122
Şekil 5.5.	CNN+YT+M+DA metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.	123
Şekil 5.6.	Mutlu ifadesi için bir gönüllünün farklı pozları.	127
Şekil 5.7.	ERUFER veri seti için tüm katılımcıları içeren örnek resimler.	128
Şekil 5.8.	JAFFE veri seti için tüm katılımcıları içeren örnek resimler.	129
Şekil 5.9.	Bir örnek üzerinde veri ön işleme uygulamaları.	131
Şekil 5.10.	Önerilen yöntemin mimarisi.	132

Şekil 5.11.	JAFFE veri seti için karmaşıklık matrisi.	135
Şekil 5.12.	ERUFER veri seti için karmaşıklık matrisi.	136
Şekil 6.1.	Örnek bir imge (ön hazırlıktan sonra AvLetters - A harfi).	140
Şekil 6.2.	AvLetters, çoğaltılmış veri - A harfi.	142
Şekil 6.3.	AvLetters artırılmış veri matrisi.	142
Şekil 6.4.	Tasarlanan CNN modeli.	144
Şekil 6.5.	Tasarlanan CNN için karmaşıklık matrisi.	146
Şekil 6.6.	AlexNet destekli CNN için karmaşıklık matrisi.	146
Şekil 6.7.	GoogLeNet destekli CNN için karmaşıklık matrisi.	147
Şekil 8.1.	Veri seti oluşturulması için gerekli etik kurulu izni belgesi.	174

GİRİŞ

Eylem kelimesinin eş anlamlısı olarak kullanılabilen edim, Moeslund'un çalışmasına göre, primatif eylem, eylem ve aktivite olarak üç grupta toplanmaktadır [1]. Primatif eylemler bir uzuv hareketiyle ifade edilirken, döngüsel primatif eylemlerden oluşan harekete ise eylem adı verilmiştir. Aktivite ise birkaç eylemin bir arada içerildiği anlamlı bir olay bütünüdür. "Sol bacak ileri" denildiğinde bir primatif eylem gerçekleştirilecekken "koşma" ise bir eylemdir. "Engelli koşu" ise hem zıplama hem koşmayı barındıran bir aktivitedir. Primatif eylem, eylem ve aktivite sırasıyla düşük, orta ve yüksek seviye sınıflandırma çalışmaları olarak ifade edilmektedir [2].

İnsan edimlerini tanıma; video ve görüntülerdeki hareketlerin belirlenmesi, sensör tabanlı verilerin sınıflandırılması, belirli bir hareket ile ilgili video ve görüntülerin aranması vb. konularda yaygın olarak kullanılmaktadır [3]. Güvenlik uygulamalarında, akıllı ev sistemlerinde, şüpheli hareketlerin tespitinde, oyun endüstrisinde, mimik analizinde, dudak okumada, işaret dili tanıma gibi uygulamalarda insan edimlerini tanıma ve yorumlama en önemli parçayı oluşturmaktadır.

İnsan edimlerini sınıflandırmada birçok metot üzerine çalışılmıştır. Destek vektör makineleri (support vector machines, SVM), saklı Markov modelleri (hidden Markov models, HMM), yönlü gradyanlar histogramı (histograms of oriented gradients, HOG), temel bileşen analizi (principal component analysis, PCA), k-en yakın komşuluk (k-nearest neighbor, kNN), uzam-zaman betimleyicileri yaklaşımı (spatio-temporal descriptors approach, STS), derin öğrenme (deep learning, DL) algoritmaları gibi yöntemler, bahsedilen metotların ilgi gören ve çokça kullanılan kısımlarını oluşturmaktadır.

Video görüntülerinin artması ve büyük verilerin oluşması, insan edimlerinin sınıflandırılmasında DL metotlarına ciddi bir ihtiyacın doğmasına neden olmuştur.

Gelişen grafik sistemleri, paralel ve dağıtık sistemlerin oluşturulması büyük verilerin DL metotlarıyla işlenmesinde kolaylık sağlamaktadır.

Bu tez çalışmasında, insan edimlerinin tanınmasında alt kategoriler olarak kabul edilebilen; işaret dili tanıma, insan günlük hareketi tanıma, duygu tanıma ve işaret dili tanıma uygulamaları geliştirilmiştir. Yapılan deneysel çalışmalarda, DL yöntemlerinden evrişimli sinir ağı (convolutional neural networks, CNN) ve yığılanmış özdevinimli kodlayıcılar (stacked autoencoders, SAE) önerilen yöntemlerin sınıflandırıcı bölümlerini oluşturmaktadır. Yapılan deneysel çalışmalarda sınıflandırma performansını artırmak için veri hazırlama, imge ön işleme, veri çoğullama, görüntü işleme, transfer öğrenme, evrimsel optimizasyon ve global arama tekniklerinden bir veya daha fazlası farklı uygulamalar için kullanılmıştır. Bu şekilde oluşturulan bütünleşik metotlar önerilmiş ve ilgili problemlerde farklı test tiplerinde uygulanmıştır. Deneysel sonuçlara göre ya literatürdeki en iyi başarı ya da en iyi başarıya çok yakın performanslar elde edilmiştir.

İnsanların birbirleriyle iletişim kurması için ortak bir dil gerekmektedir. Konuşma engelli bireyler arasında dil, vücut dili veya hissel etkileşimle gerçekleştirilmesi sağlanan iletişim dili işaret dilidir. Bu tez çalışmasının işaret dili tanıma (sign language recognition, SLR) kısmında iki farklı uygulama geliştirilmiştir. İlk uygulamada harf ve rakam düzeyinde işaret dili tanıma uygulanmıştır. Bu uygulamada DL yöntemlerinden CNN, sınıflandırıcı olarak kullanılmış ve önerilen yöntemin merkezinde yer almıştır. Yeni bir CNN mimarisi geliştirmekten daha etkili ve kolay olan yöntem ön eğitilmiş model kullanarak transfer öğrenme sağlanmasıdır. Literatürde sıklıkla kullanılan AlexNet [4] ön eğitilmiş modeli bu uygulamada transfer öğrenme amaçlı kullanılmıştır. CNN modelinin başarısını doğrudan etkileyen bir başka faktör CNN'in eğitim parametrelerinin (hiperparametreler, hyperparameters) seçimidir. Bu seçimin kullanıcı deneyimine bırakılması işlem süresi karmaşıklığını artırmakla beraber, optimum sonucu bulamama riskini ortaya çıkarmaktadır. Bu nedenle, AlexNet destekli CNN yapılarının hiperparametreleri, sezgisel optimizasyon algoritmalarıyla optimize edilmiştir. Yapay arı koloni (artificial bee colony, ABC) algoritması [5], genetik algoritma (genetic algorithm, GA) [6] ve parçacık sürü optimizasyonu (particle swarm optimization, PSO) algoritması [7] hiperparametrelerin optimizasyonunda kullanılmıştır. CNN temelli, transfer öğrenme ve parametre optimizasyonu destekli önerilen yöntemler işaret dili rakamlar veri seti

(sign language digits dataset, SLDD) [8] ve Thomas Moeslund'un mimik tanıma veri seti (Thomas Moeslund's gesture recognition dataset, TMGRD) [9] üzerinde test edilmiş ve başarılı sonuçlar elde edilmiştir. Deneysel sonuçlara göre ABC destekli yöntemle SLDD veri seti için literatürde bilinen en başarılı (%94,2 başarı oranlı) çalışma [10] geride bırakılarak %98,4 oranında başarı elde edilmiştir. ABC destekli yöntemle TMGRD veri seti için literatürde bilinen en başarılı (%94,33 başarı oranlı) çalışma [11] geride bırakılarak %98,09 oranında başarı elde edilmiştir.

İşaret dili tanıma için yapılan bir diğer uygulamada ilk olarak genel kullanıma açılacak yeni bir veri seti oluşturulması amaçlanmıştır. Bu doğrultuda hastanelerin acil servislerinde kullanılmak üzere Erciyes Üniversitesi işaret dili tanıma (Erciyes University sign language recognition, ERUSLR) adında yeni bir veri seti oluşturulmuştur. 25 kelime, 49 engelli gönüllü tarafından birden fazla kez tekrarlanmış ve farklı açılardan kaydedilmiştir. Veri setinde toplamda 13222 örnek bulunmaktadır. Önerilen yöntemin ilk aşaması olarak video formatındaki örnekler ön işlemden geçirilmektedir. Videoların çerçevelere ayrılması, çerçevelerin sıralı halde tekil imgeyi oluşturması ve tekil imgenin RGB formattan gri formata çevrilmesi ön işlemin adımlarıdır. Sınıflandırma problemlerinde yaygın olarak kullanılan CNN metodu bu çalışmanın sınıflandırma aşamasında kullanılmıştır. Önerilen modelin bir diğer aşamasında ise parametre optimizasyonu gerçekleştirmek için literatürde yaygın olarak kullanılan global arama teknikleri tercih edilmiştir. Bu uygulamada grid arama (grid search, GS) ve rastgele arama (random search, RS) yöntemleri, GoogLeNet [12] tabanlı CNN modelinin eğitim parametrelerini optimize etmek için kullanılmıştır. Deneysel sonuçlar incelendiğinde, GS destekli GoogLeNet tabanlı CNN modeli ile %92,63 başarı oranı elde edilmiş ve diğer yöntemlerden elde edilen başarı oranları geride bırakılmıştır.

Günlük insan hareketi tanıma (daily human activity recognition, DHAR) akademik çalışmalarda sıklıkla kullanılan popüler bir konudur. Sağlık, yaşam destekçisi, yaşlı bakımı gibi uygulamalarda günlük hareket tespiti yaygın bir kullanıma sahiptir. DHAR için hem görsel hem de sensör tabanlı veriler kullanılabilir. Görsel veriler video verileri, statik imgeler, iskelet gibi görüntüleri içerirken, sensör tabanlı veriler ivmeölçer, jiroskop gibi sensörlerden alınan sayısal verilerden elde edilmektedir. Tezin bu kısmında sensör tabanlı verilerden DHAR ve statik imge tabanlı DHAR üzerine iki farklı uygulama

gerçekleştirilmiştir. İlk uygulamada, hiperparametre optimizasyonu (hyperparameter optimization) destekli SAE kullanılarak sensör tabanlı veriler sınıflandırılmıştır. Bu çalışmada SAE sınıflandırma aşamasında kullanılırken, ABC [5], DE [13], PSO [7] ve GA [6] algoritmaları SAE'nin hiperparametrelerinin optimizasyonunda kullanılmıştır. Bu çalışmada hiperparametre optimizasyonu gerçeklemek için ayrıca ABC temelli PSO destekli hABCPSO adında yeni bir hibrit algoritma sunulmuştur. Önerilen yöntemler literatürde yaygın olarak kullanılan human activity recognition using smartphones (UCI Har) [14] ve wireless sensor data mining (WISDM) [15] veri setleri üzerinde test edilmiştir. Birini dışarıda bırak çapraz doğrulama (leave one out cross validation, LOOCV) test tipi kullanılarak elde edilen deneysel sonuçlara göre hABCPSO destekli yöntemle UCI Har veri seti için literatürde bilinen en iyi sonuç elde edilirken, WISDM veri seti için bilinen en iyi başarı oranına yakın bir performans elde edilmiştir. Parametre optimizasyonunda sezgisel algoritmalar kullanıldığından dolayı her bir algoritma 30 defa çalıştırılmış ve tüm sonuçlar istatistiksel analizlerle ayrıca incelenmiştir. İstatistiksel analizlere göre her iki veri seti için hABCPSO yönteminin diğer algoritmalarından daha başarılı olduğu tespit edilmiştir.

Bir diğer uygulamada statik imge tabanlı DHAR gerçekleştirilmiştir. Bu çalışmada önerilen yöntem üç ana aşamadan oluşmaktadır. İlk olarak imgeler ön işlemden geçirilmektedir. İmgelerden anlamlı kısımların kırılması, gerçekleştirilen ön işlem adımındır. Önerilen yöntemin ikinci aşamasında transfer öğrenme destekli CNN ile sınıflandırma çalışması gerçekleştirilmektedir. Yaygın olarak kullanılan ön eğitilmiş modellerden bazıları AlexNet [4], VGG16 [16], VGG19 [16], GoogLeNet [12], ResNet50 [17] ve NASNet-Large [18] olarak sıralanabilir. Tüm bu ön eğitilmiş modeller transfer öğrenme amaçlı kullanılmış ve kullanılan ön eğitilmiş modeller arasında performans karşılaştırması yapılmıştır. Önerilen yöntemin son aşaması hiperparametre seçimidir. CNN'in başarı oranını artırmak için transfer öğrenme destekli CNN modellerinin hiperparametreleri ABC algoritması ile seçilmiştir. Önerilen yöntem Stanford 40 veri seti üzerinde test edilmiş ve NASNet-Large temelli yöntemle %87,78'lik başarı oranı elde edilmiştir. Elde edilen başarı oranı ile literatürde bilinen en iyi başarı oranını veren çalışma [19] geride bırakılmıştır.

Tez çalışmasının bir diğer aşamasında duygu tanıma (emotion recognition, ER) üzerine

uygulamalar geliştirilmiştir. Duygu, bir olayın durumun insan iç dünyasında oluşturduğu etkidir. Duygu tanıma ise insanların olaylara karşı tepkisinin makine öğrenmesi yöntemleri ile sınıflandırılmasıdır. Kızgınlık, mutluluk, üzüntü, iğrenme, korku ve şaşkınlık literatürde bilinen 6 temel duygu ifadesidir [20]. İnsan bilgisayar etkileşimi (human computer interaction, HCI) sisteminin alt kategorisi olan yüz ifadesi tanıma (facial expression recognition, FER), pazarlama, psikoloji, ticaret, askeri ve güvenlik gibi farklı alanlarda kullanılabilir. FER amacı için geliştirilen ilk uygulamada, internet arama motorları yardımı ile 500 örnekten oluşan ve 5 sınıf içeren bir veri seti oluşturulmuştur. Elde edilen veri seti farklı ön aşamalardan geçirilmiştir. İlk olarak Viola Jones yöntemi [21] ile yüz tespiti sağlanıp, daha sonra yüz tespiti yapılan imgeye maskeleme işlemi uygulanmıştır. Maskeleme aşamasında imge, önce gri formata çevrilmiş daha sonra gri formata çevrilmiş imgeye keskinleştirme işlemi uygulanmıştır. Bir diğer ön aşama olan veri çoğullamada farklı görüntü işleme teknikleri kullanılarak orijinal verilerden farklı veriler elde edilmiş ve sistemin eğitiminde kullanılmıştır. Gerçekleştirilen her bir aşama için sistemin performansı ölçülmüş ve her bir aşamanın sistem performansına etkisi değerlendirilmiştir. AlexNet destekli CNN modelinin sınıflandırıcı olarak kullanıldığı deneysel çalışmaların sonucuna göre, tüm aşamaların birlikte uygulandığı yöntem diğer yöntemlere göre daha başarılı sonuç vermiştir.

Tez çalışmasında gerçekleştirilen bir diğer FER çalışmasında ilk olarak literatüre kazandırmak için yeni bir veri seti oluşturulması amaçlanmıştır. Erciyes Üniversitesi yüz ifadesi tanıma (Erciyes University facial expression recognition, ERUFER) adındaki veri seti, 110 katılımcıdan 10 duygu tipi için farklı pozlarla 9005 örnek alınarak geliştirilmiştir. Uygulama için önerilen yöntemde ilk olarak veri hazırlama aşaması gerçekleştirilmektedir. Bu aşamada, orijinal imgeden ilk olarak dlib kütüphanesinde [22] tanımlı CNN tabanlı bir yöntemle yüz hizalama sağlanmaktadır. Daha sonra RGB formatındaki imge gri formata dönüştürülmektedir. Bir sonraki adımda gürültü giderme gerçekleştirildikten daha sonra imge keskinleştirilmektedir. Veri hazırlama aşamasının son adımında ise veri çoğullama adımı gerçekleştirilmektedir. Veri hazırlama aşamasından sonra parametre optimizasyonu gerçekleştirilmektedir. Bu aşamada, ResNet 18 [23] destekli CNN modelinin hiperparametreleri global arama yeteneği yüksek olan PSO algoritması ile seçilmektedir. Önerilen yöntemin test edilmesi için ERUFER

veri setinden ayrı olarak literatürde yaygın olarak tercih edilen JAFFE veri seti de kullanılmıştır. Önerilen yöntemle 10 kat çaprazlama (10 fold cross validation) test tipine göre JAFFE veri seti için %99,53'e kadar başarı elde edilmiş olup bu başarı oranı literatürde bilinen en iyi başarı oranı olmuştur. Önerilen yöntemle ERUFER veri seti üzerinde yapılan çalışmalarda %92,56'ya kadar performans değeri elde edilmiştir.

Tez çalışmasının son aşamasında dudak okuma (lip reading ,LR) üzerine bir uygulama geliştirilmiştir. Yeni bir CNN mimarisinin tanımlanmasının yanı sıra AlexNet ve GoogLeNet destekli CNN modelleri kullanılarak AvLetters [24] veri seti üzerinde deneysel çalışmalar gerçekleştirilmiştir. Orijinalinde zaman serisi tabanlı olan bu veri seti, kullanılan bir teknikte tekil imgeler haline getirilmiştir. Farklı görüntü işleme teknikleri ile gerçekleştirilen veri çoğullamanın performans etkisi ayrıca incelenmiştir. Deneysel çalışmalarda, başarı oranına etkisi araştırılan bir diğer faktör minibatch boyutudur. Elde edilen deneysel sonuçlara göre 8 minibatch boyutuna sahip AlexNet destekli CNN'in veri çoğullama adımından geçen veri seti üzerindeki değerlendirilmesi ile en yüksek başarı oranı elde edilmiştir. Bu başarı ile bilinen en iyi başarı oranı sağlanamasa da literatürde yerini almış başarılı sayılan bazı çalışmalar geride bırakılmıştır.

Tezin Organizasyonu

Bu tez çalışması şu şekilde organize edilmiştir:

- İşaret dili tanıma, günlük insan hareketi tanıma, duygu tanıma ve dudak okuma üzerine yapılan çalışmalar için geniş bir literatür taraması Bölüm 1 ile sunulmuştur.
- Bölüm 2, yapılan tez çalışmasında kullanılan metotlar ve materyaller hakkında bilgi vermektedir. Burada, problem çözümlerinde kullanılan DL yöntemleri CNN ve SAE detaylı bir şekilde açıklanmıştır. Parametre optimizasyonunda kullanılan sezgisel optimizasyon algoritmaları ile global arama teknikleri Bölüm 2'nin sonraki kısmında sunulmuştur. Sensör tabanlı verilerden insan hareketi tanıma uygulaması için geliştirilen ABC tabanlı ve PSO destekli hibrit algoritma (hABCPSO) bu bölümde verilmiştir.
- Bölüm 3, DL yöntemleri kullanılarak yapılan işaret dili tanıma uygulamalarını içermektedir. Bu bölümde ilk olarak harf ve rakam düzeyinde işaret dili tanıma

uygulaması geliştirilmiştir. Burada, hiperparametreleri ABC, GA ve PSO algoritmaları ile optimize edilmiş AlexNet destekli CNN ile sınıflandırma yöntemi önerilmiştir. Elde edilen deneysel sonuçlar literatürde var olan en başarılı çalışmalarla karşılaştırılmış ve sunulmuştur. Bölüm 3'ün diğer kısmında genel kullanıma açmak üzere hastane ortamında kullanılması için yeni bir veri seti hazırlanmıştır. Global arama teknikleri ile eğitim parametreleri optimize edilen transfer öğrenme destekli CNN kullanılarak gerçekleştirilen deneysel çalışmalar detaylı olarak sunulmuştur.

- Bölüm 4, DL yöntemleri kullanılarak yapılan günlük insan hareketi tanıma uygulamalarını içermektedir. Bölümün ilk kısmında sensör tabanlı verilerden insan edimlerini sınıflandırma uygulaması geliştirilmiştir. Burada, yapısal parametreleri ABC, GA, DE, PSO ve hABCPSO optimizasyon algoritmaları ile optimize edilen SAE ile sınıflandırma yöntemi önerilmiştir. Elde edilen deneysel sonuçlar literatürde yer alan çalışmalarla karşılaştırılmış ve sunulmuştur. Bölümün ikinci kısmında statik imgeler üzerinden insan edimlerini tanıma üzerine bir çalışma gerçekleştirilmiştir. Literatürde yaygın kullanılan 6 ön eğitilmiş modelden ayrı ayrı transfer öğrenme gerçekleştirilmiş olup her bir durum için ayrıca ABC algoritması kullanılarak hiperparametre optimizasyonu gerçekleştirilmiştir. Yapılan deneysel çalışmalar ve elde edilen sonuçlar detaylı bir şekilde sunulmuş, çalışmanın literatürdeki çalışmalarla performans karşılaştırması değerlendirilmiştir.
- Bölüm 5, DL yöntemleri kullanılarak yapılan duygu tanıma uygulamalarını içermektedir. Bu bölümde ilk olarak yeni bir duygu tanıma veri seti oluşturup ilgili veri seti üzerinde ön işleme adımları uygulanmış ve AlexNet destekli CNN ile sınıflandırma gerçekleştirilmiştir. Yüz tespiti ve maskeleye ön işlemlerinin dışında veri çoğullama uygulamasının da başarıma etkisi incelenmiştir. Bu bölümün diğer uygulamasında, genel kullanıma açmak için 110 gönüllünün katılımıyla oluşturulan ERUFER veri seti sunulmuş ve bu veri seti ile JAFFE veri seti üzerinde uygulanan metot detaylı olarak açıklanmıştır. Yapılan deneysel çalışmalar ve elde edilen sonuçlar incelenip JAFFE veri seti için literatür kıyaslaması sunulmuştur.
- Bölüm 6, DL yöntemleri kullanılarak yapılan dudak okuma uygulamasını içermektedir. Bu uygulamada kullanılan AlexNet ve GoogLeNet destekli CNN tabanlı modellerle AvLetters veri seti üzerinde sınıflandırma yapılmıştır. Bu uygulamada ayrıca

yeni bir CNN mimarisi tasarlanmış ve ön eğitimli model destekli CNN modelleri ile karşılaştırılmıştır. Veri çoğullama ve minibatch boyutu etkenlerinin performansa etkisi deneysel çalışmalarla incelenmiş olup elde edilen deneysel sonuçlar literatürde var olan çalışmalarla karşılaştırılmış ve sunulmuştur.

- Tez çalışması ile ilgili tartışmalar, genel değerlendirme ve gelecek çalışmalar için öneriler Bölüm 7 ile verilmiştir.



1. BÖLÜM

LİTERATÜR TARAMASI

Bu tez çalışmasında insan hareketlerini tanıma; işaret dili tanıma, günlük hareket tanıma, duygu tanıma ve dudak okuma olmak üzere 4 farklı alt başlıkta gerçekleştirilmiştir. Bu 4 farklı sınıflandırma uygulamalarına ait literatür taramalarının yanı sıra, problemlerin çözümünde yer alan ön eğitilmiş modeller ve parametre optimizasyonu için literatürde var olan çalışmalar bu bölümde ele alınmıştır.

1.1. İşaret Dili Tanıma Üzerine Yapılan Çalışmalar

Harf veya rakam düzeyinde işaret dili tanıma (sign language recognition, SLR) veya bir başka deyişle el hareketi tanıma (human gesture recognition, HGR), bu alandaki uygulamaların ilk aşaması olarak kabul edilse de doğruluk oranı yüksek, başarılı çalışmalar için klasik makine öğrenmesi algoritmaları kimi durumlarda yetersiz kalmaktadır. HGR sınıflandırma çalışmaları için derin öğrenme metotları son zamanlarda sıklıkla kullanılmaktadır. Lin ve arkadaşları [25] insan el hareketlerini tanımak için bir evrişimli sinir ağı (convolutional neural networks, CNN) yöntemi önermişlerdir. Çalışmalarında, sistemin gürbüzlüğü için cilt modeli ve el pozisyonu kalibrasyonu ve oryantasyon uygulanmıştır. Ekip, bu yöntemi bir XBOX Kinect kamera ile çekilen görüntüleri içeren veri kümesinde test etmiştir. Deneylerde %95,96 başarı oranı elde edilmiştir. Bir cep telefonunun el hareketlerini bulmak ve tanımak için [26] çalışmasında bir sınıflandırma yöntemi kullanılmıştır. Ekip tarafından bir veri seti oluşturulup, farklı derin öğrenme yöntemleri ile test aşaması gerçekleştirilmiştir. Ölçek, dönme, gürültü vb. gibi değişmeyen özellikler, hareket algılamayı tanımaya yarayan önemli faktörlerdir. Flores ve ekibi [27] iki CNN mimarisi kullanarak değişmez özellikler altında statik bir el hareketi tanıma yöntemi önermişlerdir. Kullanılan veri setinde, CNN mimarileri

tarafından %95,37 ve %96,20 başarı oranlarına ulaşılmıştır. Arenas ve ekibi HGR için MatConvNet kütüphanesi [28] kullanarak CNN mimarisini oluşturmuşlardır. Altı mimari varyansı, “açık” ve “kapalı” el hareketi sınıfları içeren test verilerinin sınıflandırılması için kullanılmıştır [29]. Bir başka çalışmada, transfer öğrenme destekli CNN metodu, Cote-Allard ve ekibi [30] tarafından önerilmiştir. Önerilen CNN yöntemi, daha genel ve sağlam özellikleri tespit etmek için kullanılmıştır. Kullanılan veri setinde bu yöntemle %97,81 ortalama doğruluk elde edilmiştir. Oyedotun ve Khashman [31], SLR için üç CNN ve üç yığılanmış özdevinimli kodlayıcı (stacked autoencoders, SAE) yöntemi önermişlerdir. TMGRD veri setini kullanan ekip, yöntemlerinde en iyi sonuç olarak SAE'nin üçüncü versiyonunun kullanılmasıyla %92,83 tanıma oranı elde etmişlerdir. Başka bir çalışmada Oyedotun ve Khashman [11], yüz tanıma ve statik el hareketi tanıma için prototip içeren bir duygusal sinir ağı modeli önermişlerdir. [31] makalesine kıyasla, kullanılan yöntemle %94,33 başarı oranı elde etmişlerdir. Beşer ve arkadaşları tarafından işaret dili tanıma için bir kapsül ağı (CapsNet) [32] tabanlı yöntem kullanılmıştır. CapsNet'te veri artırma, grup büyüklüğü, evrişim katmanı ve filtre büyüklüğü değişimi gibi farklı varyasyonlar uygulanmıştır. Deneysel çalışmalarda işaret dili rakamlar veri setinde (sign language digits dataset, SLDD) %94,2 başarı oranı elde edilmiştir [10].

Bu tez sürecinde işaret dili tanıma üzerine yapılan bir başka çalışma, kelime düzeyinde işaret dili veri seti geliştirme ve derin öğrenme ile tanınması üzerinedir. SLR harf veya rakam düzeyiyle olmakla birlikte kelime ve cümle düzeyinde de gerçekleştirilmektedir. Literatürde kelime düzeyinde işaret dili tanıma üzerine yoğun çalışmalar vardır. American sign language lexicon video data set (ASLLVD) veri seti [33] kullanılarak Neidle ve ekibi [34] tarafından ASLLVD'yi ön plana çıkaran bir bilgisayar görü uygulaması geliştirilmiştir. Bir başka çalışmada, Kim ve ekibi [35] el takibi ve el poz tahmini üzerine bir yöntem gerçekleştirmiş ve yöntemlerini ASLLVD veri seti üzerinde test etmişlerdir. ASLLVD veri setini çalışmalarında kullanan Metaxas ve ekibi [36], öğrenme yöntemleri ve özellikleri birleştiren bir karma yapı tasarlayarak işaret dili tanıma üzerine çalışma gerçekleştirmiştir. Bir başka kelime düzeyinde işaret dili tanıma veri seti Oszust ve Wysocki tarafından oluşturulan Polish sign language (PSL) Kinect 30 veri setidir [37]. Tam kelime modeli ve veri odaklı alt dizi modelini kullanarak sınıflandırma yapısı geliştiren Oszust ve Wysocki [38], önerilen yöntemi yine bu veri seti üzerinde test

etmişlerdir. Kapuscinski ve ekibi, çalışmalarında [39] en yakın komşu tekniği (k-nearest neighbour, kNN) ile saklı Markov modelleri (hidden Markov models, HMM) PSL Kinect 30 veri seti üzerinde sınıflandırılmıştır. Kelime düzeyinde geliştirilen bir başka veri seti Argentinian sign language (LSA64) veri setidir [40]. Bu veri setini kullanarak çeşitli modeller geliştirilmiş ve sunulmuştur [41–43].

1.2. Günlük Hareket Tanıma Üzerine Yapılan Çalışmalar

Günlük hareket tanıma, hem görsel hem de sensör verileriyle gerçekleştirilebilen uygulamalardır. Bu tez döneminde, günlük hareket tanıma için gerçekleştirilen ilk uygulama, sensör tabanlı verileri SAE ile sınıflandırma üzerine gerçekleştirilmiştir. Uygulamada ilk olarak, bu alanda literatürde bulunan çalışmalar incelenmiştir. Almaslukh ve ekibi [44] SAE kullanarak, standart eğitim test ayırımı yapılan UCI Har veri seti üzerinde sınıflandırma yapmışlar ve başarılı bir sonuç elde etmişlerdir. Alsheikh ve ekibi [45] kısıtlı Boltzman makinaları (restricted Boltzmann machines, RBM) kullanarak çalışmalarını WISDM Actitracker veri seti [46] üzerinde test etmiştir. Uzun kısa vadeli hafıza (long short-term memory, LSTM) modeli kullanarak WISDM veri seti üzerinde çalışma yapan Chen ve arkadaşları [47] %92,1'lik bir başarı elde etmiştir. Derin CNN ile otomatik olarak optimum özellikleri bulmaya çalışan bir metot Jiang [48] ve ekibi tarafından önerilmiştir. Li ve ekibi sensör verilerinden oluşan veri seti ile sınıflandırma yapmak için SAE kullanmışlardır [49]. Ravi ve ekibi [50] WISDM veri seti üzerinde 10 kat çaprazlama yaparak %98,2'lik bir başarı elde etmiştir. Ravi ve ekibi bir başka çalışmada [51] bir dizi sığ özellikten tamamlayıcı bilgilerle birlikte eylemsiz sensör verisinden öğrenilen özellikleri bir araya getiren bir derin öğrenme metodolojisi önermişlerdir. Ronao ve Cho, UCI HAR veri seti kullanarak CNN ile sınıflandırma çalışmaları gerçekleştirmiştir [52–54]. Walse ve ekibi derin sinir ağları yardımı ile UCI Har veri setini test etmiştir [55].

Hareketsiz görüntülerden hareket algılanması, video veya sensör tabanlı verilere göre daha zor bir işlem olsa da [56, 57], bu alanda hatırı sayılır sayıda çalışma vardır. Bas ve ekibi [56], pencereler gibi hareketsiz görüntülerden nesnelere çıkarmak ve otomatik nesne tanıma görevini öğrenmek için çoklu örnek öğrenme yöntemini önermektedirler. Çoklu örnek öğrenme yöntemini derin öğrenme teknikleriyle birleştirmek önem arz etmektedir.

Yöntem, Stanford 40 veri seti ile test edilmiştir ve derin öğrenmenin doğruluğunun çoklu örnek öğrenme ile artırıldığı belirlenmiştir. Qi ve ekibi görüntü tabanlı hareket tanımda CNN potansiyellerini araştırmaktadırlar [57]. Poz ipuçları CNN uygulamasına entegre edilmiştir ve yeni bir hareket tanıma yöntemi önerilmiştir. Wang ve arkadaşları, belirli bir bölgede genel bir eylem örneği bulundurma durumunu ölçmek için bir eylemlilik tanımı geliştirmiştir. Eylemliliğin doğru ve etkili bir şekilde tahmin edilmesi video analizinde önemlidir. Yapılan çalışmada görünüm tam evrişim ağı (appearance fully convolutional network) ve hareket tam evrişim ağı (motion fully convolutional network) yapıları kullanılarak hibrit bir çalışma gerçekleştirilmiştir. Stanford 40 [58], UCF Sports [59] ve HMDB [60] veri setleriyle çalışmanın başarımı test edilmiştir [61]. Bedeni 7 parçaya bölen semantik parça eylem tanıma temelli model Zhao tarafından önerilmiştir. Bu 7 parçanın her biri için semantik parça hareketleri tanımlanmıştır. Anlık hareket tanıma, tüm bu parçaların kombinasyonundan gerçekleştirilmektedir [62]. Lavinia ve ekibi [63] 2 ve 3 derinlikte CNN modellerinin birleştirilmesi üzerine bir yöntem önermektedirler. Birleştirilmiş özellikler, sınıflandırma için rastgele orman (random forest) ve destek vektör makineleri (support vector machines, SVM) algoritmalarına gönderilmiş ve sınıflandırma işlemi gerçekleştirilmiştir.

1.3. Duygu Tanıma Üzerine Yapılan Çalışmalar

Duygu tanıma bir olaya karşı insanın iç dünyasında gerçekleşen tepki ve bu tepkinin dışa vurumudur. Psikoloji, ticari ve güvenlik alanlarında sıklıkla kullanılan duygu tanıma için literatürde yoğun çalışmalar mevcuttur. Tez sürecinde yapılan duygu tanıma uygulamaları statik imge tabanlı veriler üzerinden gerçekleşmektedir. Literatürde sık kullanılan genel kullanıma açık statik imge tabanlı veri setlerinden başlıcaları; Toronto Face Database (TFD) [64], Facial Expression Recognition (FER2013) [65], The Static Facial Expressions in the Wild (SFEW) [66], The Radboud Faces Dataset (RaFD) [67], Karolinska Directed Emotional Faces (KDEF) [68], The Japanese Female Facial Expression (JAFFE) [69] olarak sıralanabilir.

Devries ve ekibi [70], çok görevli ağ tabanlı evrişimli sinir ağını (CNN), TFD veri setindeki yüz ifadelerini tanımak için kullanmıştır. Veri setinin test kısmında %85,13 oranında başarı elde etmişlerdir. Başka bir derin öğrenme yöntemi, sıfır-sapmalı CNN,

Khorrani ve ekibi tarafından önerilmiş ve %88,6'lık başarı oranını TFD veri seti üzerinde sağlanmıştır [71]. Ding ve arkadaşları [72], TFD veri setinde en iyi başarı oranını, ince ayar tabanlı CNN kullanarak, %88,9 ile sağlamışlardır.

Bir diğer veri seti olan FER2013 üzerine Tang [73], Devries ve arkadaşları [70], Zhang ve ekibi [74], Guo ve arkadaşları [75], Kim ve ekibi [76] ve Pramerdorfer ve Kampel [77] CNN tabanlı çalışmalar gerçekleştirmişlerdir. Pramerdorfer ve Kampel en yüksek başarı oranını %75,2 ile elde etmiştir. Chang ve ekibi [78], CNN'den özellik çıkarımı içeren bir karmaşıklık algı sınıflandırma yöntemi önermiştir. FER2013 veri seti kullanılarak yapılan deneysel çalışmaların sonuçlarına göre başarılı sonuçlar elde edilmiştir. Yeni bir topluluk budama algoritması (ensemble pruning algorithm) Li ve Wen [79] tarafından önerilmiş olup kullanılan test tekniğine göre %73,39 doğruluk oranı bu veri seti üzerinde sağlanmıştır.

Videolardan statik imgeler alınarak oluşturulan SFEW veri seti üzerinde, Kim ve ekibi [80], Yu ve ekibi [81], Cai ve ekibi [82] ve Meng ve ekibi [83] CNN tabanlı yöntemlerini eğitmiş ve test etmişlerdir. Bu çalışmalar içerisinde Kim ve ekibi %61,6 oranı ile literatürde bilinen en yüksek başarıyı elde etmişlerdir.

RaFD veri seti 67 katılımcı ile laboratuvar ortamında oluşturulmuş bir veri setidir. Venkatraman ve Balasubramanian [84] AlexNet ön eğitilmiş modeli kullanarak özellik çıkarma işlemi gerçekleştirmiştir. Test verileri üzerinde gerçekleştirilen çalışmada %100 oranında başarı elde edilmiştir. Sun ve ekibi [85] RaFD veri setini üzerinde klasik ve derin öğrenme metotları kullanarak çalışmalar gerçekleştirmiş ve %93,4'e kadar başarı oranı elde etmişlerdir.

Farklı açılarla alınan pozlarla oluşturulan KDEF veri seti toplamda 4900 imgeden oluşmaktadır. Kim ve ekibi [86] CNN tabanlı bir yöntem önererek sınıflandırma çalışması gerçekleştirmişlerdir. Yapılan deneysel çalışmalar sonucunda literatürdeki en başarılı sonucu elde etmişlerdir.

Bir başka yaygın kullanılan veri seti JAFFE'dir. Lui ve ekibi [87] derin inanç ağları (deep belief networks) yöntemi tabanlı sınıflandırma yapmışlar ve %91,8 başarı oranını elde etmişlerdir. Bir diğer çalışmada, Hamester ve ekibi [88] CNN ve evrişimli özdevinimli

kodlayıcı (stacked convolutional autoencoders) kullanarak bir yöntem geliştirmiş ve JAFFE veri seti üzerinde %95,8 oranında başarı sağlamışlardır.

1.4. Dudak Okuma Üzerine Yapılan Çalışmalar

Dudak okuma; güvenlik, işaret dili anlamlandırmaya yardımcı olmak, anormallik tespiti gibi durumlarda kullanılan bir yöntemdir. Bu yöntemin sınıflandırılması için literatürde araştırmacıların çalışmaları mevcuttur. Garg ve ekibi [89], CNN ve LSTM derin öğrenme metodlarından birleştirdikleri modeli dudak okuma işlemi üzerine uygulamışlardır. Bu birleşimde, CNN özellik çıkarımında, LSTM ise sınıflandırmada kullanılmıştır. Kelime ve sözcük gruplarından oluşan MIRACL-VC1 [90] veri seti modelin test edilmesinde kullanılmıştır.

Dudak dizilerinden oluşan orijinal resmin yerine dinamik özellik resmi kullanarak CNN ile sınıflandırma gerçekleştiren Li ve ekibi [91], modeli ATR Japon konuşma veri seti [92] ile test etmiştir.

Petridis ve ekibi, LSTM destekli bir çalışmayı görsel konuşma tanıma üzerine geliştirmişlerdir [93]. Model iki akıştan oluşmaktadır. Bunlardan ilki, direk olarak ağızdan özellik çıkarılması iken ikincisi ise imgeler arasındaki değişimdir. Her bir akımın zamansal dinamikleri LSTM ile gerçekleştirilmiştir. Çalışma OuluVS2 [94] ve CUAVE [95] veri setleriyle test edilip, sonuçlar sunulmuştur.

Dong ve ekibi [96], 5 evrişim katmanı ve 256 gizli birimli LSTM kullanarak bir model geliştirmişlerdir. Metodun başarımı 2 veri setinin birleşimi ile test edilmiştir.

Derin öğrenme ile kelime düzeyinde görsel konuşma tanıma modeli Stafylakis ve Tzimiropoulos [97] tarafından önerilmiştir. LRW [98] veri seti üzerinde test edilen model, evrişimli, Resnet ve iki yönlü LSTM ağlarının bir araya getirilmesiyle oluşturulmuştur.

Takashima ve ekibi [99] ciddi duyma kaybı yaşayan insanlar için derin öğrenme destekli konuşma tanıma sistemi geliştirmişlerdir. Metotta hem ses hem görsel veriler kullanılmıştır ve çıkarılan özellikler sınıflandırma işlemi için sisteme dahil edilmiştir.

Takashima ve ekibi bir başka çalışmada [100], dudak resmi ile ses özelliklerinin

birleştirildiği bir yöntemle dudak okuma üzerine yeni bir yaklaşım önermişlerdir. Önerilen yöntem ATR Japon konuşma veri seti ile test edilmiştir.

Türkçe renk isimlerinden oluşan veri setini, Kinect aygıtı ile görüntü ve açı değerleri olarak sınıflandıran sistem Yargıç ve Doğan tarafından önerilmiştir [101]. Dudak koordinatları bilgisini Kinect kameradan alan bu ekip, noktalar arasındaki açılarını kullanmışlar ve kNN algoritması ile sınıflandırma yapmışlardır.

1.5. Ön Eğitimli Modeller Üzerine Yapılan Çalışmalar

Ön eğitimli modeller özellik çıkarma ve transfer öğrenme için araştırmacılar tarafından kullanılmaktadırlar. Akçay ve arkadaşları [102] nesne algılama ve sınıflandırma için AlexNet kullanmışlardır. Alnujaim ve ekibi tarafından el hareket tanıma işlemi için önceden eğitilmiş modellerden AlexNet ve VGG16 kullanılmıştır [103]. Han ve diğerleri sınırlı eğitim setine AlexNet, ResNet152 ve VGG16 gibi ön eğitimli modellerden gelen güçlü özellikleri aktararak yeni bir görüntü sınıflandırma yöntemi üzerinde çalışmışlardır [104]. Bussey ve diğerleri güçlü yüz tanıma için AlexNet ve VGG-Face modellerini karşılaştırmışlardır [105]. AlexNet, GoogLeNet ve ResNet, Tang ve arkadaşları tarafından Çince karakter tanıma için kullanılmıştır [106]. Xia ve ekibi [107] AlexNet ve ResNet ön eğitimli modeller ile küçük yaşta çocukların gülümsemelerinin tespit edilmesi üzerine bir çalışma gerçekleştirmiştir.

Hareket tanıma için ön eğitimli modellerin kullanıldığı çalışmalar literatürde mevcuttur. Keceli [108] özellik çıkarımı için AlexNet kullanmıştır. Gupta ve Balan, aksiyon videolarının optik akışını görselleştirmek için AlexNet benzeri bir CNN eğitmişlerdir [109]. İskelet hacmiyle hareket tanıma için AlexNet ön eğitimli modelinden özellik çıkarımı Keceli ve ekibi tarafından çalışılmıştır [110]. Han ve ekibi [111], veri kümesini yeniden modelleme stratejisini sunmak için ResNet101 kullanarak transfer öğrenimi üzerine bir çalışma gerçekleştirmiştir. ResNet temelli model kullanılarak iskelet verilerinden hareket tanıma üzerine çalışma Pham ve ekibi tarafından sunulmuştur [112].

1.6. Parametre Optimizasyonu Üzerine Yapılan Çalışmalar

Hiperparametre optimizasyonu, makine öğrenme yöntemlerinin başarı oranını arttırmada etkili bir yöntemdir. En küçük kareler destek vektör makineleri (least square support vector machines, LS-SVM) regresyon için kullanılan bir makine öğrenme yöntemidir. LS-SVM'nin performansı etkileyen faktörlerden birisi hiperparametre seçimidir. Rubio ve ekibi [113], çekirdek (kernel) ve hiperparametreleri optimize etmek için sezgisel bir yöntem içeren bir model geliştirip modeli zaman serisi uygulamasında kullanmışlardır. Bayesian optimizasyonu (Bayesian optimization, BAO) tabanlı bir yöntem olan FABOLAS, alt küme büyüklüğünün optimizasyonunda kullanılmıştır [114]. Yöntem, büyük veriler üzerinde SVM ve CNN kullanılarak gerçekleştirilmiş ve deneysel sonuçlara göre, FABOLAS ile yapılan eğitimin çok görevli Bayesian optimizasyonundan daha hızlı olduğu görülmüştür. Hiperparametre ayarlama işlemi, kullanıcı deneyimine bağlı olarak zaman alan bir işlem olduğundan, [115] çalışmasında hiperparametre optimizasyonu için başka bir BAO yöntemi önerilmiştir. Katman ve düğümlerin sayısı, öğrenme ve dropout oranları gibi parametrelerin belirlenmesi, sinir ağları için zorlu bir görevdir. Bu nedenle, Diaz ve arkadaşları tarafından parametre seçimi için türevsiz bir optimizasyon aracı geliştirilmiş ve benchmark setlerinde test edilmiştir [116]. Özellik grupları ve hiperparametreler, metin sınıflandırma için genetik algoritma (genetic algorithm, GA) kullanılarak Desmet ve Hoste [117] tarafından optimize edilmiştir. Kalite kontrol örnekleri ve destek vektör regresyonu (quality control samples and support vector regression, QC-SVRC), tolerans eşiği, ceza süresi ve çekirdek genişliği gibi hiperparametrelere sahiptir. Etkili bir sonuç elde edebilmek için, QC-SVRC'nin hiperparametreleri grid arama (grid search, GS), rastgele arama (random search, RS) ve parçacık sürü optimizasyonu (particle swarm optimization) algoritması kullanılarak optimize edilmiş ve sonuçlar sunulmuştur [118]. Hinz ve ekibi [119], CNN'nin hiperparametrelerini seçme süresini azaltmak için bir yaklaşım önermiştir. RS, parzen tahmin edicilerinin ağacı, sıralı model tabanlı algoritma konfigürasyonu ve GA, hiperparametre optimizasyonu deneyleri için kullanılmış ve iyi performans göstermiştir. Dil modellemede parametre değerlerini bulmak için optimizasyon teknikleri Rahnama ve ekibi [120] tarafından kullanılmıştır. Önerilen model, hiperparametre optimizasyonu

için kullanılmış ve genel kullanıma açık bir veri setinde test edilmiştir. Riegler ve ekibi [121], manuel arama (manual search), GS, RS ve BAO algoritmaları kullanarak girdi ve gizli katmanlar için nöron sayısı, öğrenme oranı, döngü sayısı, batch boyutu, katman sayısı gibi LSTM hiperparametrelerini optimize etmek için bir yöntem geliştirmiştir. Diferansiyel gelişim algoritması (differential evaluation algorithm) kullanarak CNN'nin evrişimli katmanlarını ayarlamak için yeni bir yöntem Stoean tarafından önerilmiştir [122]. Yöntem üç farklı sınıflandırma yöntemiyle karşılaştırılmış ve daha iyi sonuçlar elde edilmiştir. Bir başka çalışmada Maraqa ve Abu-Zaiter [123], Arapça işaret dili tanıma için tekrarlayan sinir ağı (recurrent neural network, RNN) tabanlı metot kullanarak %95'lik bir doğruluk oranı elde etmiştir. Optimizasyon algoritmaları HGR çalışmaları için de kullanılmıştır. Özellik vektörünü ve sinir ağını ayarlamak için modifiye edilmiş GA'ya dayanan çok amaçlı bir evrimsel algoritma, Chevtchenko ve ekibi tarafından önerilmiştir [124]. Massey University gesture veri seti [125] üzerinde hareket tanıma için bir Bayesian sınıflandırıcı model kullanılmıştır. Sagayam ve Hemanth [126], tanıma oranı ve hata oranı gibi performans ölçümlerinde daha iyi sonuçlar alabilmek için 1-D Gizli Markov Modeli ve yapay arı koloni (artificial bee colony, ABC) algoritmasını içeren bir hibrit model önermiş ve önerilen modeli Cambridge el hareketi veri kümesi [127] üzerinde test etmişlerdir. Öğrenme hızı, katman sayısı, nöron sayısı, aktivasyon fonksiyonu, dropout oranı ve kapılı tekrarlayan birimler hücrelerinin çekirdek ayarı, RNN'lerin hiperparametreleridir. Çalışma [128] ile RNN modeli için en uygun topolojiyi oluşturmak üzere hiperparametre optimizasyonu için RS yöntemi uygulanmıştır. Yöntemin performans analizi için işlenmemiş veri özellikleri ve daha üst seviye özellikli işaret veri setleri kullanılmıştır. ABC'nin [5], literatürde parametre optimizasyonları için kullanıldığı farklı çalışmalar da mevcuttur. Badem ve ekibi hibrit bir ABC'yi kullanarak derin sinir ağı (deep neural network) ile sınıflandırma çalışmaları yapmıştır [129]. Guguk Kuşu Algoritması (cuckoo optimization algorithm, COA) [130] ve ABC kullanılarak RNN'de ağırlık optimizasyonu gerçekleştirilmiş ve elde edilen sonuçlar karşılaştırılmıştır [131]. COA ve ABC performansının RNN ile karşılaştırılması üzerine bir başka çalışma Nawi ve ekibi tarafından gerçekleştirilmiştir [132]. ABC, klasik sınıflandırma yöntemleriyle de hiperparametre optimizasyonu için kullanılmaktadır. Chen ve ekibi [133], LS-SVM'nin hiperparametrelerini optimize etmek için ABC'yi kullanmışlardır.

2. BÖLÜM

METOTLAR VE MATERYALLER

2.1. Problem Çözümlerinde Kullanılacak Derin Öğrenme Algoritmaları

Bir makine öğrenmesi (machine learning) dalı olan derin öğrenme (deep learning), verilerin üst düzey soyutlamalarını modellemeye çalışan bir algoritmadır [134]. Bir başka deyişle derin öğrenme, bilgiyi doğrusal olmayan şekilde işlemek için çok karmaşık katmanlara sahip, makine öğrenmesi alanında gelişen bir alandır [135]. Derin adını, derin katmanlar (layers) ve hiyerarşik yapısından dolayı almaktadır. Yapay sinir ağlarına (artificial neural networks) göre daha derin bir yapıya sahip olan derin öğrenme tekniklerini günümüzde popüler kılan bazı etmenler vardır. Bunlardan en önemlisi verinin çok geniş boyutlara ulaşması ve bu veriyi eğiterek işleyecek işlemci çiplerinin geliştirilmesidir. Grafiksel işlem birimleri (GPU) devasa verilerle çalışabilme ve paralel işlemleri yürütebilme olanağıyla derin öğrenmenin gelişimine önemli katkı sağlamıştır. Derin öğrenmenin en önemli avantajlarından birisi, elle hazırlanmış özelliklerin (features), yarı danışmanlı (semi-supervised) ve danışmansız (unsupervised) özellik öğrenimi ve hiyerarşik öznitelik (özellik) çıkarımı için etkili yöntemlerle yer değiştirmesidir [135].

3 tip derin öğrenme yöntemi mevcuttur. Bunlar; üretimsel, ayrımsal ve karma modeldir [136]. Verilerin ilişkili olduğu sınıflara öznitelikler üzerinden dağılımının bayes olasılık kuralı ile gerçekleştirilmesi, üretimsel derin öğrenme yöntemleriyle sağlanır. Dolayısıyla yöntem danışmansız olarak ayrıştırıcı özellik kazanmaktadır [136].

Etiket yapısının kullanıldığı bir diğer yöntem ayrımsal derin öğrenme yöntemidir. Bu yöntemle doğrudan ayırıcı özniteliklerin belirlenmesi sağlanarak danışmanlı (supervised)

bir yapı kurulmuş olmaktadır [136].

Son katmanında etiketli verilerin, diğer katmanlarda danışmansız eğitimin kullanıldığı yöntem de karma yöntem olarak adlandırılmaktadır [136].

Literatürde başlıca kullanılan derin öğrenme modelleri; evrişimli sinir ağları ve özdevinimli kodlayıcılar olarak sıralanabilir.

2.1.1. Evrişimli Sinir Ağları

Evrişimli sinir ağları (convolutional neural networks, CNN), çok boyutlu, büyük verileri işlemek üzere özelleşmiş bir yapay sinir ağı türüdür. Evrişimli ağlar, genel matris çarpımı yerine evrişim (convolution) işlemini en az bir katmanında kullanan sinir ağlarıdır [137].

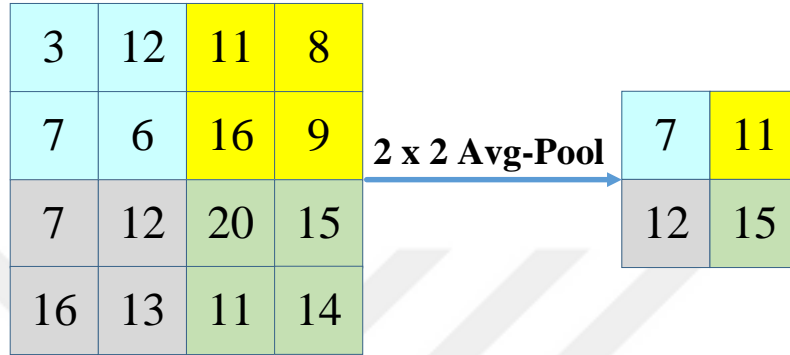
CNN'in temel bileşenleri; evrişim katmanı (convolution layer), ortaklama katmanı (pooling layer), aktivasyon fonksiyonları (activation functions), tam bağlantılı katman (fully connected layer), kayıp katman (loss layer), düzenleme (regularization) ve optimizasyondur (optimization).

Evrişim katmanı, öğrenilebilir filtre kümesini içerir ve evrişimli ağın tam bağlantılı katmanla birlikte öğrenme yeteneğine sahip olan yapısıdır. Bu katman için önemli olan parametreler; filtre genişliği (filter size), filtre sayısı (filter number) ve adım aralığıdır (stride) [4]. Örnek bir evrişim işlemi Şekil 2.1 ile sunulmuştur. Burada I , iki boyutlu imgeyi, K $h \times w$ boyuttaki filtreyi (evrişim çekirdeği (kernel) olarak da bilinir) ve $I * K$ evrişmiş özellikleri ifade etmektedir.

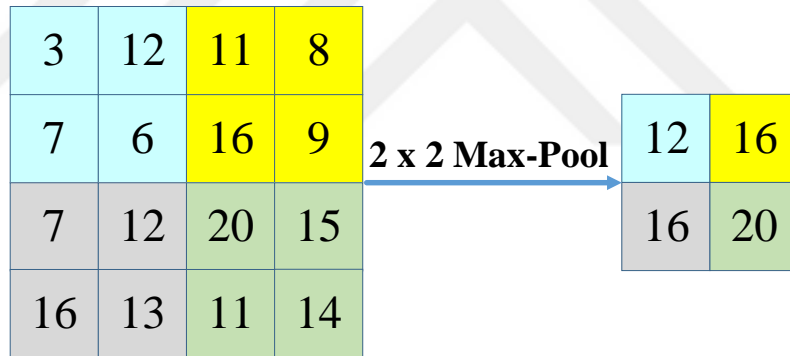
I	K	I*K																																																																														
<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td></tr> </table>	0	1	1	1	1	1	0	0	1	1	1	0	0	0	0	1	1	1	0	0	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0	0	0	1	1	0	0	1	*	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> </table>	1	0	1	0	1	0	1	0	1	=	<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>1</td><td>4</td><td>4</td><td>5</td><td>3</td></tr> <tr><td>1</td><td>2</td><td>4</td><td>3</td><td>3</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>1</td></tr> <tr><td>1</td><td>3</td><td>3</td><td>1</td><td>1</td></tr> <tr><td>3</td><td>3</td><td>2</td><td>1</td><td>1</td></tr> </table>	1	4	4	5	3	1	2	4	3	3	1	2	3	4	1	1	3	3	1	1	3	3	2	1	1
0	1	1	1	1	1																																																																											
0	0	1	1	1	0																																																																											
0	0	0	1	1	1																																																																											
0	0	0	1	1	0																																																																											
0	0	1	1	0	0																																																																											
0	1	1	0	0	0																																																																											
0	1	1	0	0	1																																																																											
1	0	1																																																																														
0	1	0																																																																														
1	0	1																																																																														
1	4	4	5	3																																																																												
1	2	4	3	3																																																																												
1	2	3	4	1																																																																												
1	3	3	1	1																																																																												
3	3	2	1	1																																																																												

Şekil 2.1. Evrişim süreci.

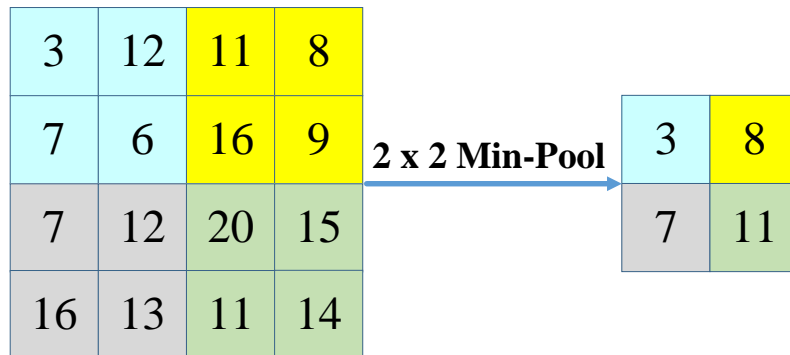
Ağın hesap maliyetini düşüren bir diğer önemli yapı ortaklama katmanıdır. Sistemi küçük pozisyon değişimlerine karşı dayanıklı hale getiren bu katmanda ortaklama işlemi genellikle ortalama, maksimum, minimum gibi işlemleri kullanmaktadır. Ortalamaya, maksimuma ve minimuma ortaklama işlemleri sırasıyla Şekil 2.2, 2.3, ve 2.4 ile gösterildiği gibidir.



Şekil 2.2. Ortalamaya ortaklama işlemi.



Şekil 2.3. Maksimuma ortaklama işlemi.

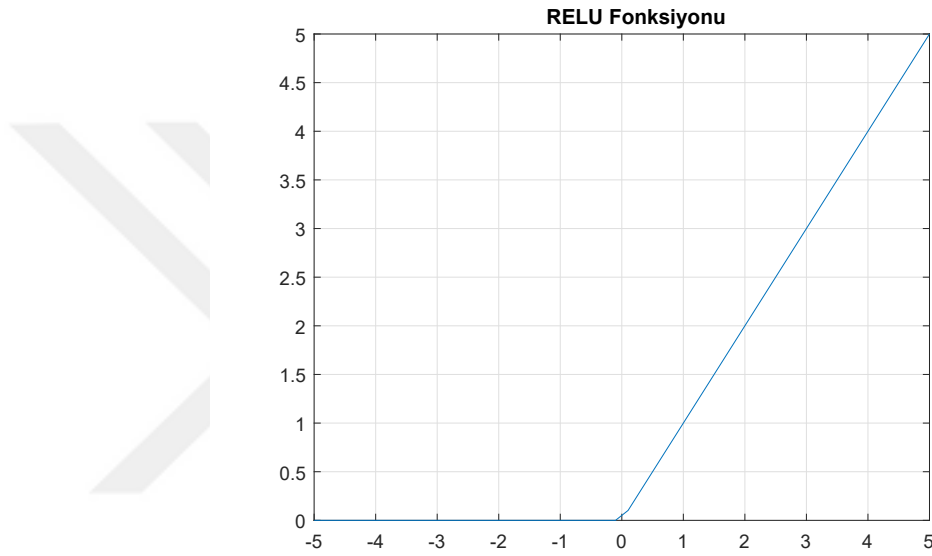


Şekil 2.4. Minimuma ortaklama işlemi.

Rectified linear unit (ReLU), exponential linear unit (ELU) ve sigmoid temel aktivasyon fonksiyonlarının seçimi CNN'in performansını önemli derece etkilemektedir. Negatif

girişleri sıfır olarak, pozitif girişleri ise değiştirmeden çıkışa yansıtan fonksiyon ReLU'dur. z değerinin giriş olarak alındığı ReLU fonksiyonu parçalı doğrusal bir fonksiyondur ve Eşitlik 2.1 ile tanımlanmıştır. ReLU fonksiyonu Şekil 2.5 ile sunulmuştur.

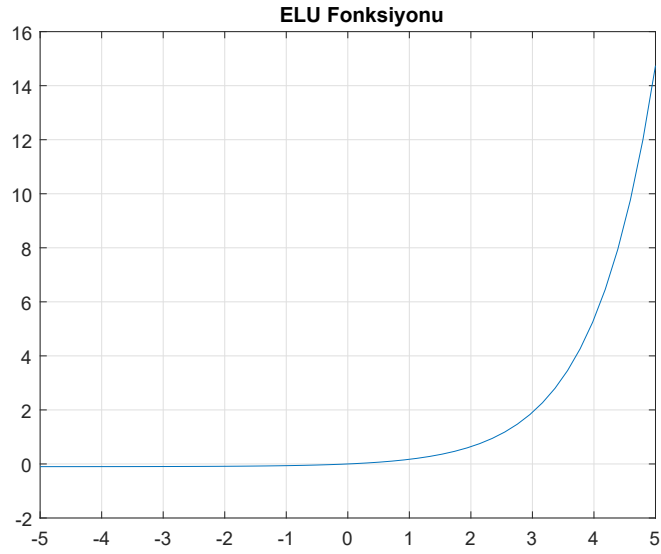
$$f_{ReLU}(z) = \max(z, 0) \quad (2.1)$$



Şekil 2.5. ReLU temel aktivasyon fonksiyonu.

ELU, sinir ağlarının daha hızlı öğrenmesini ve daha yüksek sınıflandırma doğruluğuna erişmesini sağlayan aktivasyon fonksiyonudur. ELU fonksiyonu Eşitlik 2.2 ile tanımlanmıştır. Burada λ negatif girişler için önceden tanımlanmış kontrol parametresidir. ELU aktivasyon fonksiyonu Şekil 2.6 ile sunulmuştur.

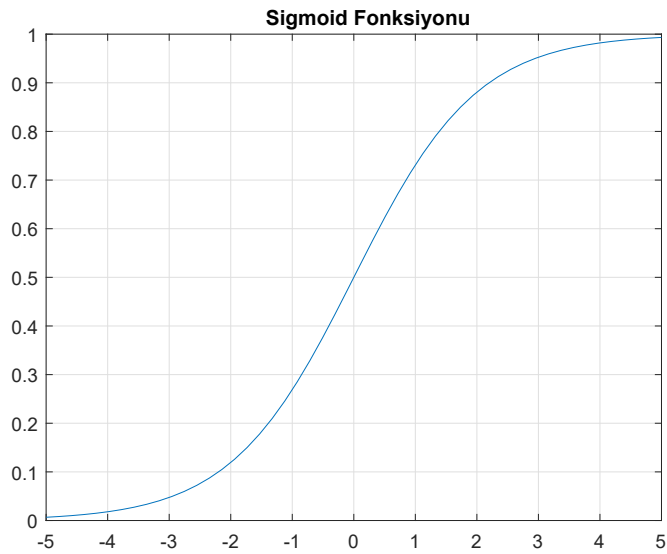
$$f_{ELU}(z) = \max(z, 0) + \min(\lambda(e^z - 1), 0) \quad (2.2)$$



Şekil 2.6. ELU temel aktivasyon fonksiyonu.

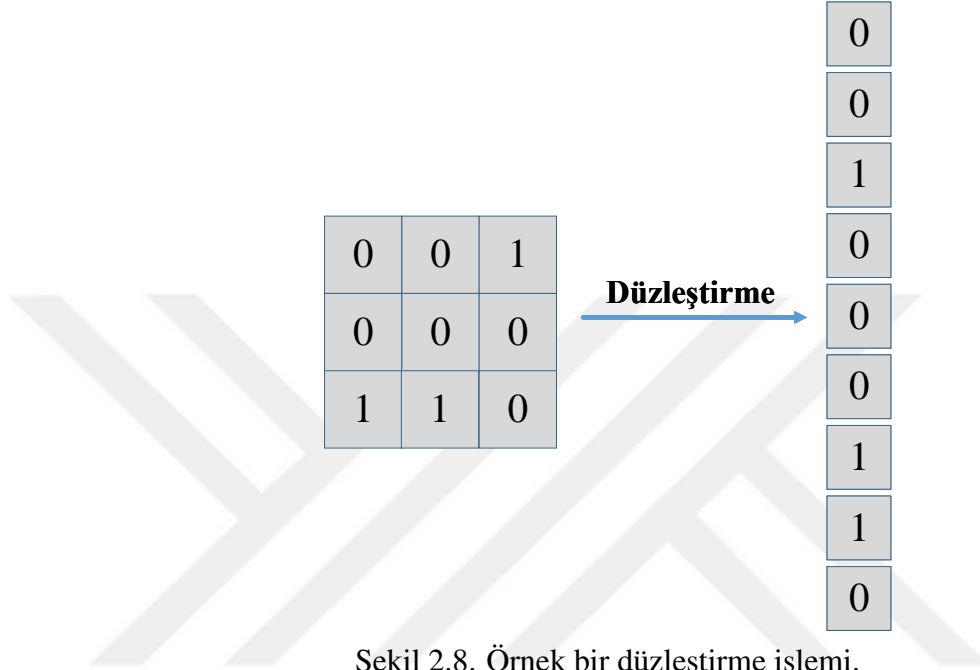
Sigmoid fonksiyonu sürekli ve türevi alınabilir bir fonksiyondur ve sıkça kullanılmaktadır. Sigmoid aktivasyon fonksiyonu Eşitlik 2.3 ile tanımlanmış ve Şekil 2.7 ile sunulmuştur.

$$f_{Sigmoid}(z) = \frac{1}{1 + e^{-z}} \quad (2.3)$$



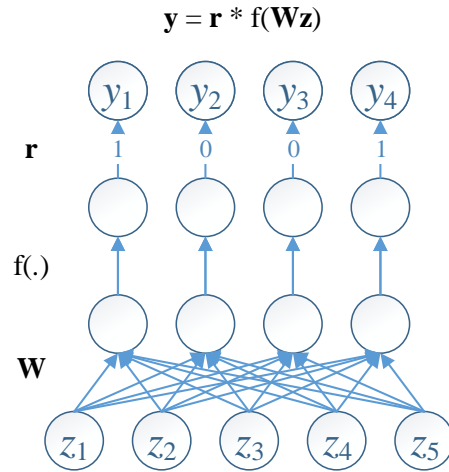
Şekil 2.7. Sigmoid temel aktivasyon fonksiyonu.

Tam bağlantılı katman,CNN’de evrişim ve ortaklama katmanından sonra gelmektedir. Bu katmanda sinirler bir önceki katmanla tam bağlantıya sahiptir. Veriler tam bağlantılı katmana gelmeden önce düzleştirme (flattening) işleminden geçirilir. Örnek bir düzleştirme işlemi Şekil 2.8 ile gösterildiği gibidir.

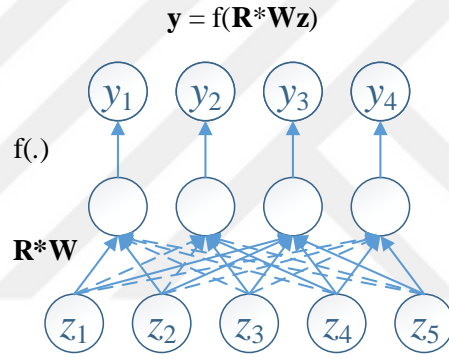


CNN’nin son katmanı olan kayıp katmanı, eğitim sırasında tahmin edilenle gerçek etiketler arasındaki sapmanın nasıl değerlendirileceğini belirlemektedir. Softmax, en sık kullanılan kayıp fonksiyonlarından birisidir.

Düzenleştirme teknikleri ile derin sinir ağları için önemli bir sorun olan aşırı öğrenme probleminin önüne geçilmesi amaçlanmaktadır. Dropout ve Dropconnect en önemli iki düzenleştirme teknikleridir [4] ve Şekil 2.9 ve 2.10 ile verilmiştir.



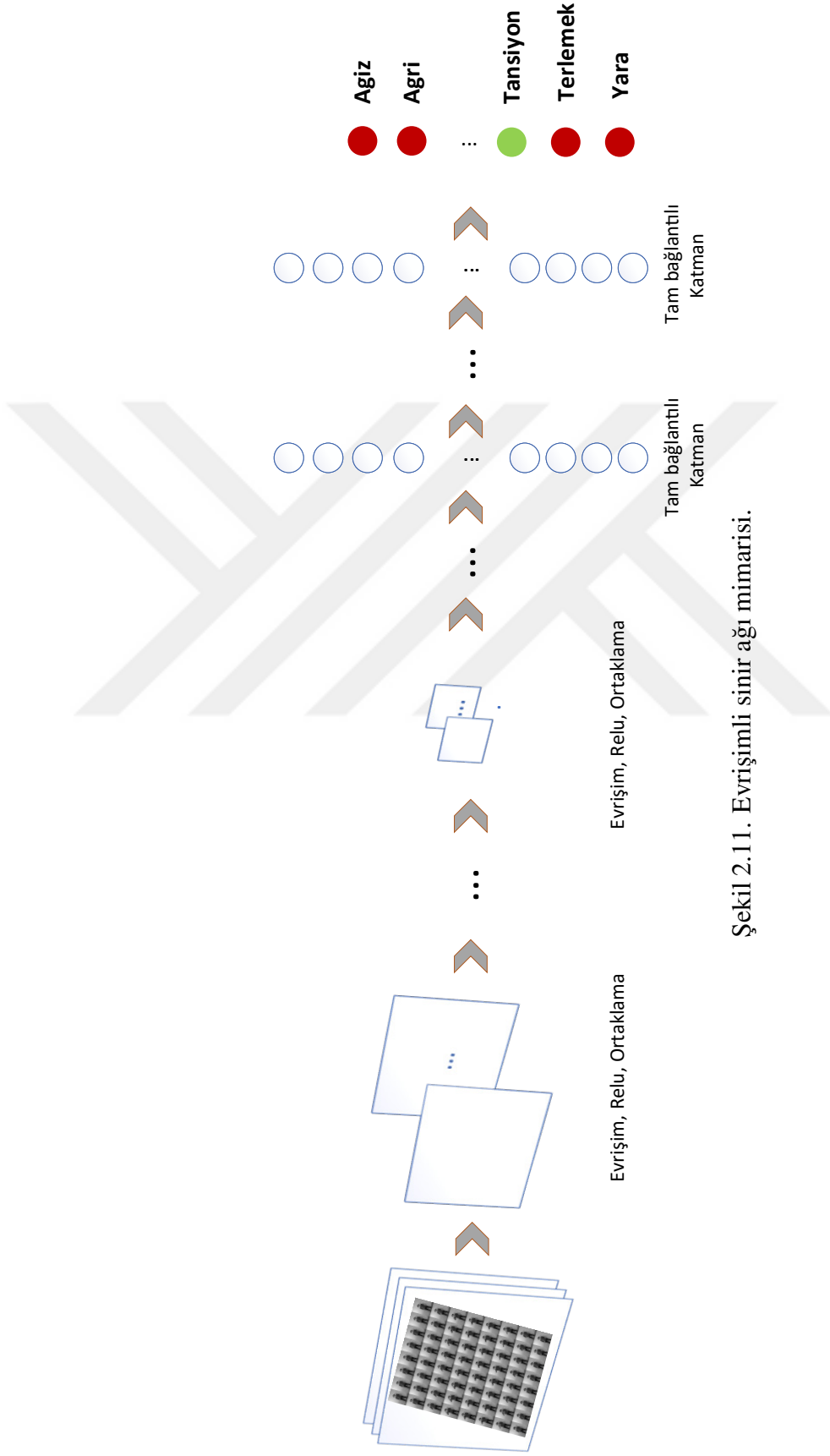
Şekil 2.9. Dropout düzenleme tekniği.



Şekil 2.10. Dropconnect düzenleme tekniği.

Şekil 2.11 ile tipik bir CNN mimarisi sunulmaktadır. AlexNet [4], GoogLeNet [12] ve ResNet [17] gibi modeller özel CNN mimarilerinden sayılmaktadır.

Derin öğrenmede, genellikle kullanıcı deneyimine bırakılan CNN eğitim parametreleri seçimlerinin, başarı oranı üzerinde önemli bir etkisi vardır. Maksimum epok (maximum epoch), minibatch boyutu (minibatch size), ilk öğrenme oranı (initial learning rate), l_2 düzenleme parametresi (l_2 regularization parameter), karıştırma tipi (shuffle) ve momentum değeri, CNN için en önemli eğitim parametrelerinden bazılarıdır. Hiperparametre olarak da adlandırılan bu eğitim parametreleri için açıklamalar ve alabilecekleri değer aralıkları Tablo 2.1 ile verilmiştir.



Şekil 2.11. Evrişimli sinir ağı mimarisi.

Tablo 2.1. CNN eğitim seçenekleri.

Seçenek Adı	Tanım	Seçenekler veya Değer Aralıkları	Varsayılan Değer	Veri Tipi
Maksimum epok	Eğitim için tanımlanan maksimum döngü sayısı.	Kullanıcı bağımlı	30	Tam sayı
Minibatch boyutu	Eğitimdeki her bir iterasyon için ihtiyaç duyulan alt veri genişliği.	Genellikle 2'nin üsleri (Önerilen)	128	Tam sayı
İlk öğrenme oranı	Öğrenme oranının başlangıç değerini ifade eder.	(0, 1)	0.01	Kayan nokta
ℓ_2 düzenleme	Çok düşük seçilmesi durumunda, eğitim işlemi uzun zaman alır, gereğinden yüksek seçilirse, optimum sonuç bulunamayabilir.	(0, 1)	0.0001	Kayan nokta
Karıştırma tipi	ℓ_2 düzenleme değeri. Bir katmandaki her parametre kümesi, ℓ_2 için bir çarpan belirtebilir. regularizer.	1 - hiç (never) 2 - bir kere (once) 3 - her epokta (every-epoch)	2 - bir kere	Dizi
Momentum	Eğitim verilerinin karıştırılıp karıştırılmadığını kontrol eden parametre.	(0, 1)	0.90	Kayan nokta

2.1.1.1. Transfer Öğrenme

Yeni bir CNN modeli geliştirmek oldukça zahmetli ve zaman isteyen bir süreçtir. Sınıflandırma, transfer öğrenme ve özellik çıkarma gibi özellikleri olan ön eğitilmiş modeller kullanmak, yeni bir CNN modeli geliştirmekten daha hızlı ve daha etkili bir yoldur. Bu tez döneminde gerçekleştirilen çalışmalarda, ön eğitilmiş modeller transfer öğrenme amacıyla kullanılmıştır.

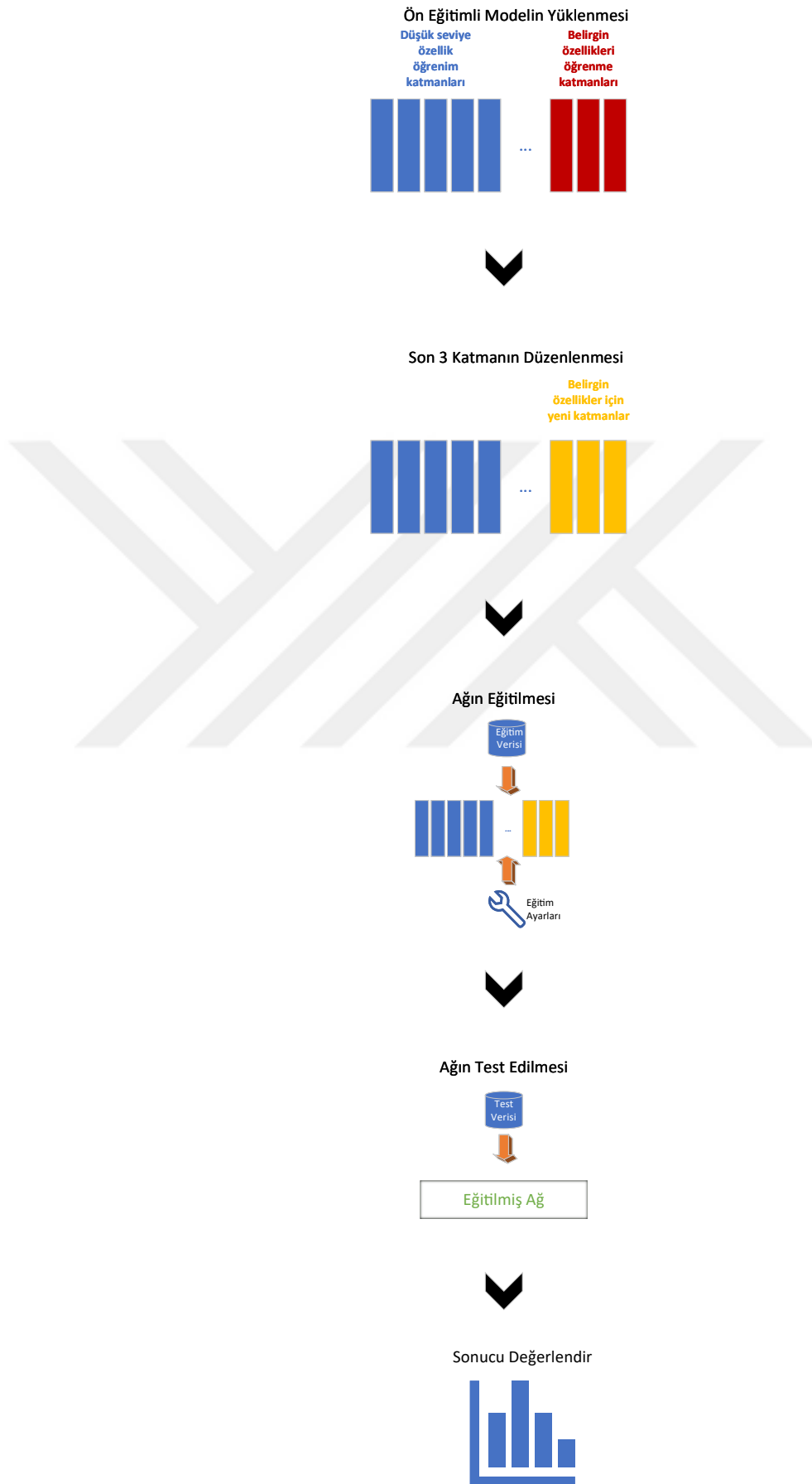
Transfer öğreniminin uygulanması için birtakım adımlar gerekmektedir [138]. Öncelikle önceden belirlenmiş model sisteme yüklenir. Ön eğitilmiş model 2 ana gruba ayrılabilir katmanlara sahiptir. İlk grup, şekiller, kenarlar, renkler vb. gibi düşük seviye özellik öğrenmeye yönelik katmanlar içerirken ikinci ana grup yüksek seviye özellik öğrenmeye yönelik katmanlar içerir. Yüklenen modelden düşük seviye özellik öğrenimini sağlayan katmanlar alınırken, daha spesifik özellik öğrenimini sağlayan katmanlar çıkarılır. Böylelikle ön eğitilmiş modelin son üç katmanı çıkarılmış olur. Son üç katman yerine kullanıcı tarafından yeni katmanlar eklenir. Oluşturulan ağ, eğitim verisi ve ayarlarıyla eğitilip ortaya eğitilmiş model çıkarılır. Eğitilmiş model test verisi ile test edilerek sonuçlar alınır. Transfer öğrenme sürecinin akış şeması Şekil 2.12 ile sunulmuştur.

AlexNet

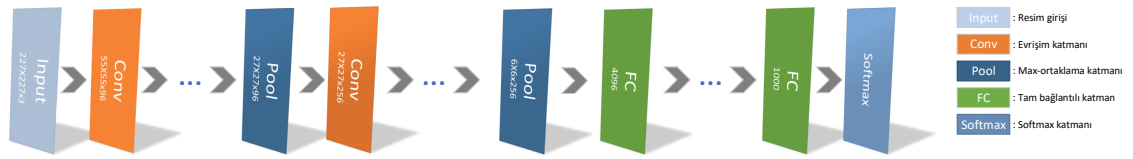
Çok geniş bir imge kütüphanesi ile eğitilen ve güçlü özellik çıkarımı gerçekleştiren AlexNet [4] ile transfer öğrenme veya özellik çıkarımı farklı problemler için uygulanabilir. Bu ağ modelinin eğitiminde ImageNet veri setinin [139] bir alt kümesi kullanılmıştır. Beş evrişim katmanı ve 3 tam bağlı katmanı olmak üzere toplamda 8 öğrenilebilir katmanı olan AlexNet, 2012 yılında yapılan ImageNet Büyük Ölçekli Görsel Tanıma Yarışması'nda (ILSVRC) birinci olmuştur. AlexNet modelinin mimarisi Şekil 2.13 ile gösterildiği gibidir.

GoogLeNet

2014 ILSVRC yarışmasını kazanan GoogLeNet modelinin [12] ILSVRC veri setinde AlexNet'e göre başarı oranı daha yüksektir. AlexNet ve VGG modellerine göre daha karmaşık yapıya sahip olan GoogLeNet modelinin mimarisi Şekil 2.14 ile gösterilmiştir.



Şekil 2.12. Transfer öğrenme süreci.



Şekil 2.13. AlexNet mimarisini.

VGG 16

ILSVRC veri seti ile eğitilen VGG16 modeli hem transfer öğrenme hem de özellik çıkarımı amacıyla kullanılabilir. VGG16 [16], 13 evrişim ve 3 tam bağlı katman olmak üzere toplamda 16 öğrenilebilir katmandan oluşur. AlexNet'ten daha geniş bir ağı sahip olan VGG16 modeli daha yavaş çalışmasına rağmen ILSVRC veri setinde başarı oranı daha yüksektir. VGG16 modelinin mimarisini Şekil 2.15 ile gösterildiği gibidir.

VGG 19

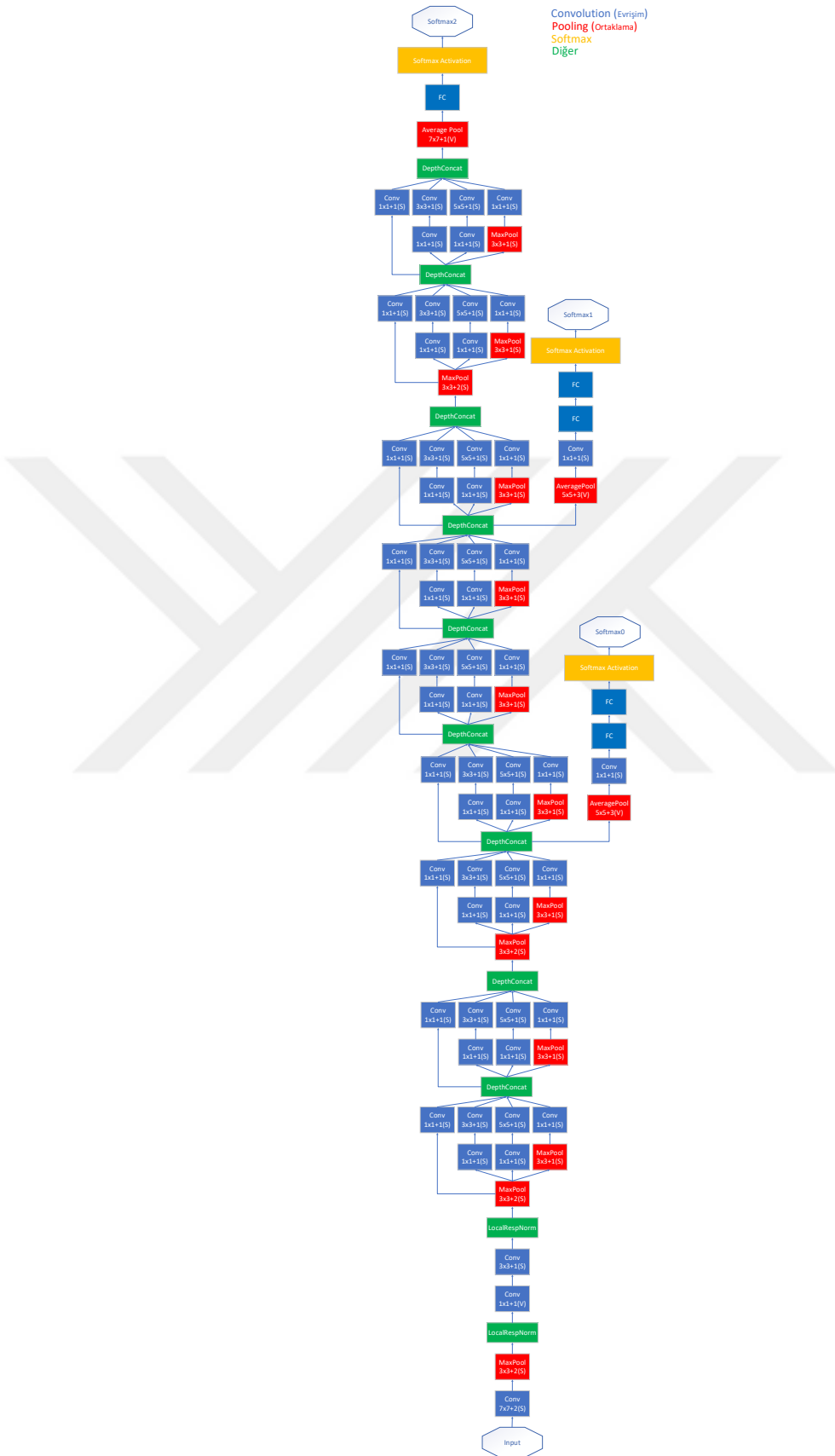
AlexNet ile karşılaştırıldığında VGG16 ile benzer özelliklere sahip olan VGG19 [16] modeli daha fazla katmana sahiptir. Bu ağ modeli, 16 evrişim katmanı ve 3 tam bağlı katman olmak üzere 19 öğrenilebilir katmandan oluşmaktadır.

ResNet 18

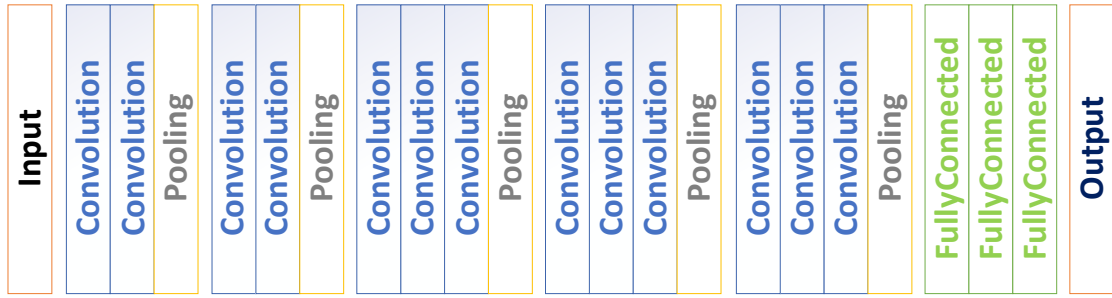
Zengin özelliklerin elde edilmesini sağlayan ResNet 18 ön eğitilmiş model, ImageNet veri setindeki bir milyondan fazla veriyi 224×224 boyutunda girdi olarak çalışmaktadır. 18 katmana sahip olmasına rağmen daha derin katmana sahip bazı modellere göre başarılı ve hızlı sonuçlar verdiği analiz edilmektedir [138].

ResNet 50

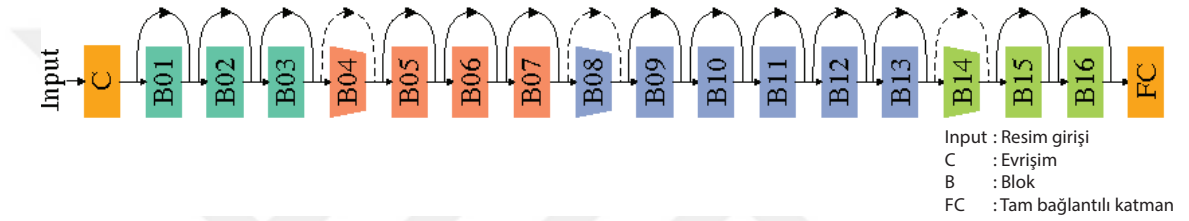
Çok derin ağların eğitimini kalıcı bağlantılarıyla gerçekleştiren ResNet [17], GoogLeNet'ten daha derin, daha büyük ve daha yavaş bir yapıya sahiptir. 50 katman derinliğinde olan bu model ile ILSVRC veri setinde GoogLeNet'e göre daha yüksek başarı elde edilmektedir. ResNet50 modelinin mimarisini [140] Şekil 2.16 ile gösterildiği gibidir.



Şekil 2.14. GoogLeNet mimarisi.



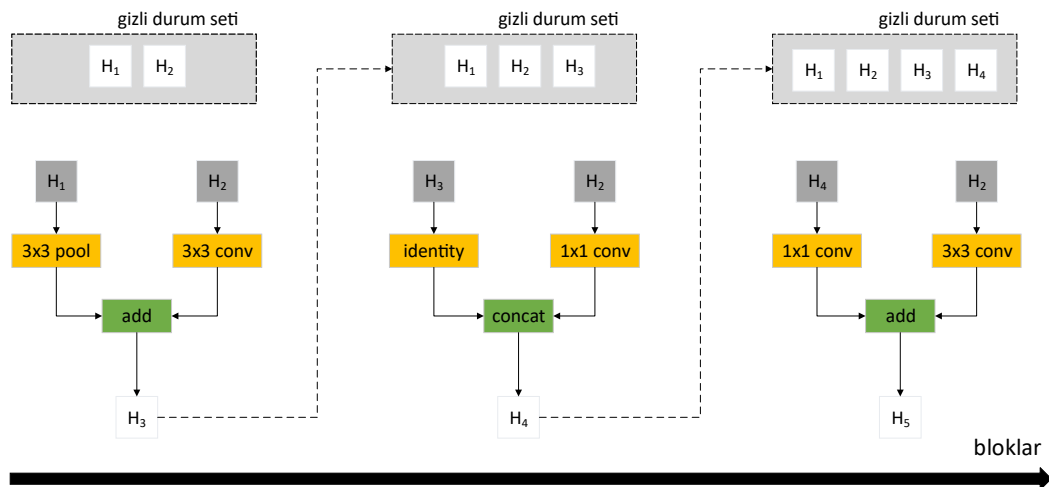
Şekil 2.15. VGG mimarisi.



Şekil 2.16. ResNet mimarisi.

NasNET-Large

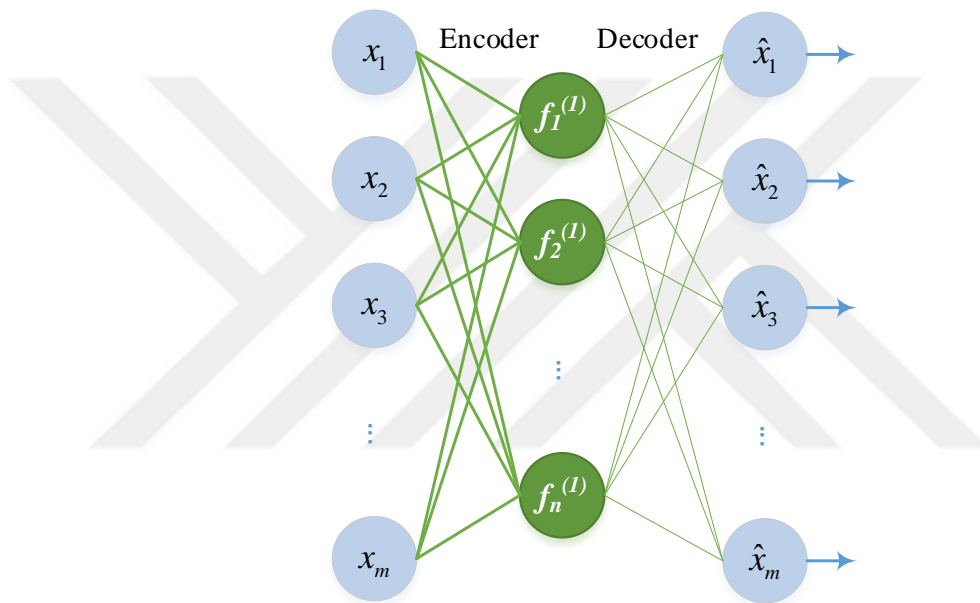
NASNet-Large [18], bu alıřmada kullanılan modeller arasında en derin ve en yavař n eęitimli modeldir. 88.9 milyon parametre ve 1244 katmana sahiptir. Modelin giriř boyutu varsayılan olarak 331×331 'dir. NASNet arama alanının řematik diyagramı [18] řekil 2.17 ile gsterildięi gibidir.



Şekil 2.17. NASNet arama alanının řematik diyagramı.

2.1.2. Özdevinimli Kodlayıcılar

Çıktısında girişi elde etmeye çalışmak üzere eğitilmiş bir sinir ağı türü olan özdevinimli kodlayıcılar (autoencoders, AE) modeli eğitimsiz öğrenme gerçekleştiren bir derin öğrenme metodudur. Geri yayılım algoritması eğitim işleminde kullanılmaktadır ve giriş verisi çıkış etiketi olarak verilmektedir [141, 142]. AE'nin en önemli kullanım amaçlarından birisi, saklı katman sayesinde verinin içinde aranan saklı öznelikleri elde etmektir. Temel bir AE mimarisi Şekil 2.18 ile sunulmuştur.



Şekil 2.18. Örnek bir AE mimarisi.

Bir AE yapısı, kodlayıcı ve kod çözücü parçaları olarak düzenlenen giriş, gizli ve çıkış katmanlarından oluşur. Kodlayıcı kısmı giriş ve gizli katmanlar arasındayken kod çözücü kısmı gizli ve çıkış katmanı arasındadır [143]. Giriş sinyalleri için nöron sayısı m ve çıkış sinyalleri için nöron sayısı n olduğunda, kodlayıcı çıkışı Eşitlik 2.4 gösterildiği gibi hesaplanır. Burada, kodlayıcı çıkışı, $\mathbf{w}_c = [w_{c1} w_{c2} w_{c3} \dots w_{cn}]^T$, giriş sinyali $\mathbf{x} = [x_1 x_2 x_3 \dots x_m]^T$, bias ağırlıkları $\mathbf{b} = [b_1 b_2 b_3 \dots b_n]^T$, bağlantıların ağırlıkları $\mathbf{W} = [\mathbf{W}_1 \mathbf{W}_2 \mathbf{W}_3 \dots \mathbf{W}_n]$ ve f aktivasyon fonksiyonu olarak ifade edilir. AE çıkışının hesaplanması Eşitlik 2.5 ile gerçekleştirilir. Burada, $\hat{\mathbf{x}} = [\hat{x}_1 \hat{x}_2 \hat{x}_3 \dots \hat{x}_m]^T$ AE çıkışını, $\hat{\mathbf{b}} = [\hat{b}_1 \hat{b}_2 \hat{b}_3 \dots \hat{b}_n]^T$ bias ağırlıklarını, $\hat{\mathbf{W}} = [\hat{\mathbf{W}}_1 \hat{\mathbf{W}}_2 \hat{\mathbf{W}}_3 \dots \hat{\mathbf{W}}_n]$ bağlantıların ağırlıklarını ifade eder.

$$\mathbf{w}_c = f(\mathbf{b} + \mathbf{W}^T \mathbf{x}) \quad (2.4)$$

$$\hat{\mathbf{x}} = \hat{f}(\hat{\mathbf{b}} + \hat{\mathbf{W}}^T \mathbf{w}_c) \quad (2.5)$$

Bir AE mimarisi birden fazla gizli katman içeriyorsa yığılanmış özdevinimli kodlayıcı (SAE) olarak adlandırılır. Örnek bir SAE mimarisi Şekil 2.19 ile sunulmuştur. Bir SAE yapısının direkt olarak doğruluğu etkileyen hiperparametreleri vardır. SAE'nin optimize edilebilir yapısal parametreleri, enkoder transfer fonksiyonu (encoder transfer function), dekoder transfer fonksiyonu (decoder transfer function), seyreklik oranı (sparsity proportion), gizli katman boyutu (hidden size), ℓ_2 ağırlık düzenleme oranı (ℓ_2 weight regularization rate) ve kayıp fonksiyonudur (loss function).

Enkoder transfer fonksiyonu parametresi, transfer fonksiyonu olarak enkoder için kullanılır. Logsig ve satlin fonksiyonları enkoder transfer fonksiyonu olarak kullanılabilir.

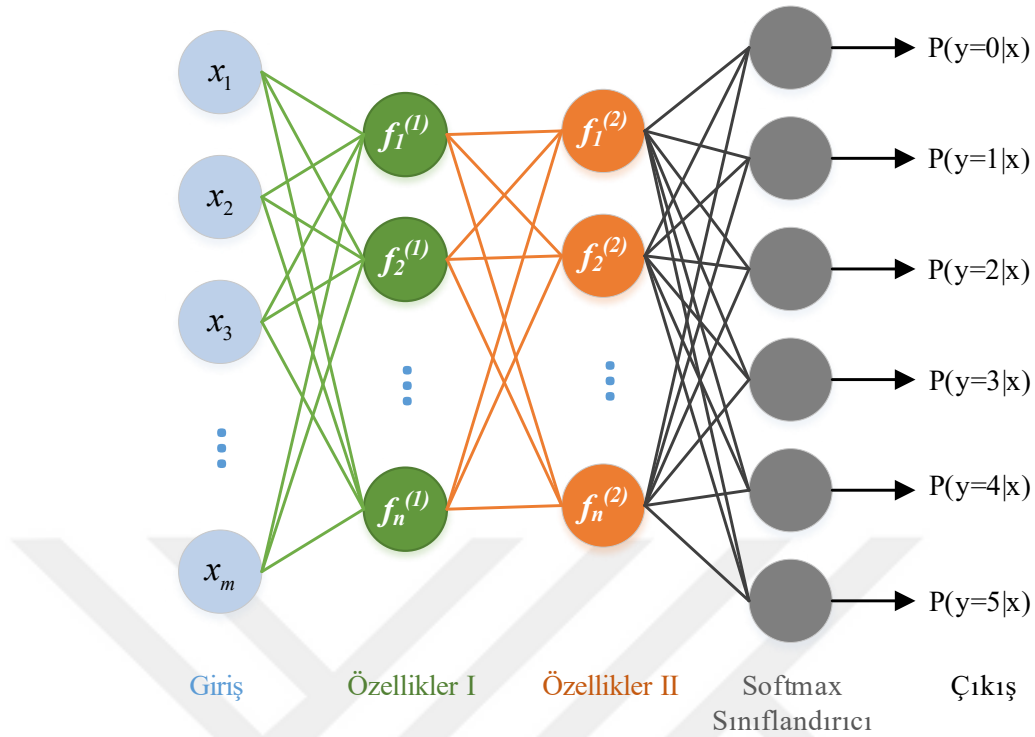
Dekoder transfer fonksiyonu, dekoder için transfer fonksiyonu olarak kullanılır. Logsig, satlin ve purelin fonksiyonları dekoder transfer fonksiyonu olarak kullanılabilir.

Seyreklik oranı, çıkışın gizli katmandan gelen seyrekliğini kontrol eden seyreklik düzenleyicinin bir parametresidir. 0 ile 1 arasında olan bu parametre için küçük değer önerilir.

Gizli katman boyutu, özdevinimli kodlayıcının gizli temsilinin boyutunu belirtir.

ℓ_2 ağırlık düzenleme oranı, ℓ_2 ağırlık düzenleyicisinin ağırlığını kontrol eden temel değerdir.

Kayıp fonksiyonu, softmax katmanının kayıp fonksiyonudur. Çapraz entropi (cross entropy) ve ortalama karesel hata (mean square error, MSE) bu parametrenin olası seçenekleridir.



Şekil 2.19. Örnek bir SAE mimarisi.

2.2. Problem Çözümlerinde Kullanılacak Optimizasyon Algoritmaları

Derin öğrenme metotlarında hiperparametre seçimleri genellikle kullanıcı deneyimine bırakılmaktadır. Bu parametrelerin doğru seçimi başarı oranını artırmaktadır. Parametrelerin seçiminde kullanılacak olan zeki optimizasyon algoritmaları bu bölümde incelenecektir.

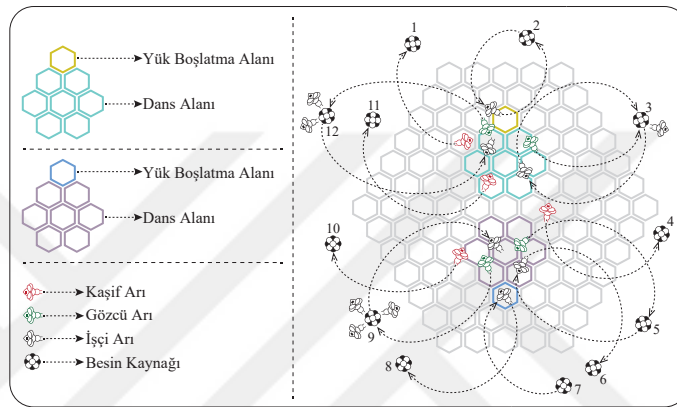
2.2.1. Yapay Arı Koloni Algoritması

Yapay arı koloni (artificial bee colony, ABC) algoritması gerçek bal arılarının davranışlarından esinlenilerek 2005 yılında Dervis Karaboğa tarafından önerilmiştir [5]. Gerçek hayattaki bal arılarının davranışları, algoritmada önerilen yapay arılarla benzerlik göstermektedir. Her bir yiyecek kaynağı, problemin olası bir çözümüne denk gelmektedir. Ayrıca algoritmada yer alan uygunluk değeri, gerçek hayattaki nektar miktarına karşılık gelmektedir [144–147]. Algoritmada tanımlı üç tip arı vardır. Bunlar; işçi arı, gözcü arı ve kaşif arılardır. Kovanda yer alan arılar Şekil 2.20 ile gösterilebilir [148].

ABC temel adımları Algoritma 2.1 ile gösterildiği gibidir [149]. İlk adımda başlangıç

parametreleri ve yiyecek kaynağı konumları tanımlanır. Başlangıç yiyecek kaynağı Eşitlik 2.6 ile belirlenir. Bu eşitlikte, x_j^{max} üst sınırı, x_j^{min} alt sınırı, x_{ij} her bir $j=1,2,\dots,D$ için çözüm vektörünü ve x_i her bir $i=1,2,\dots,SN$ için SN anlık yiyecek kaynağı popülasyonunu ifade eder. Bir durdurma kriteri sağlanana kadar işçi arı, gözcü arı ve kaşif arı fazları en iyi sonuçlar hafızada tutularak tekrarlanır [5].

$$x_{ij} = x_j^{min} + rand(0,1)(x_j^{max} - x_j^{min}) \quad (2.6)$$



Şekil 2.20. Kovandaki işçi, gözcü ve kaşif arılar.

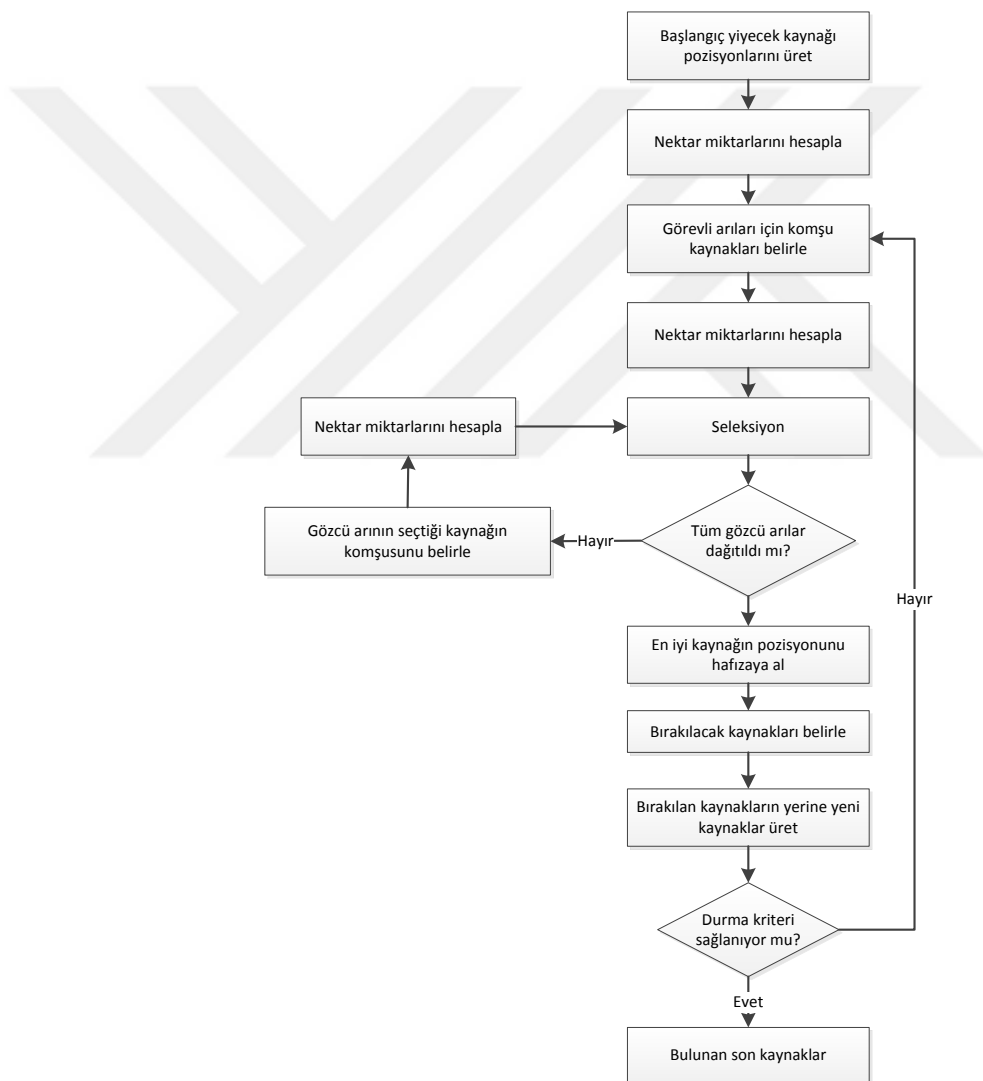
- 1: Başlangıç parametrelerinin belirlenmesi
- 2: Tüm kaşif arıların yiyecek kaynaklarına yönlendirilmesi
- 3: **repeat**
- 4: İşçi arı fazı
- 5: Gözcü arı fazı
- 6: Kaşif arı fazı
- 7: **until** Sonlandırma kriteri sağlanana kadar

Algoritma 2.1: ABC temel adımları.

İşçi arı fazında, tüm işçi arılar yiyecek kaynaklarına gönderilir. Ayrıca bu fazda Eşitlik 2.7 ile yiyecek kaynağı etrafında lokal arama gerçekleştirilir ve son olarak açgözlü seleksiyon işlemi gerçekleştirilir. Eğer x'_{ij} değeri x_{ij} değerinden daha iyiyse çözüm yer değiştirilir. Bir sonraki adım olan gözcü arı fazında, yiyecek kaynaklarının olasılık değerlerinin hesaplanması, bu olasılık değerleriyle gözcü arıların yiyecek kaynaklarına gönderilmesi, yeni olası yiyecek kaynaklarının belirlenmesi ve açgözlü seleksiyon işleminin uygulanması gerçekleştirilir. Olasılık değeri(p_i), Eşitlik 2.8 ile hesaplanır. En iyi kaynağın pozisyonu hafızaya alınır. Son faz olan kaşif arı fazında ise limit değerine ulaşan yiyecek kaynaklarında arama sonlandırılır ve bir kaşif arı yeni belirlenen yiyecek kaynağına gönderilir.

$$x'_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (2.7)$$

$$p_i = \frac{fitness_i}{\sum_{i=1}^{CS} fitness_i} \quad (2.8)$$



Şekil 2.21. ABC akış şeması.

Sürü zekası temelli algoritmalarından olan ABC algoritması az kontrol parametresine

sahip, kullanımı basit ve esnektir. Yiyecek kaynağı sayısı, maksimum döngü sayısı ve limit değeri ABC'nin temel kontrol parametreleridir [150]. Biyolojik kaynaklı ABC, Genetik Algoritma (GA), Parçacık Sürü Optimizasyon Algoritması (PSO) vb diğer popüler algoritmalara göre kimi problemlerde daha etkilidir [151, 152]. ABC'nin temel çalışma prensipleri Şekil 2.21 ile gösterilen akış diyagramı [149] ile ifade edilmektedir. Yüksek esneklik, çözümlerin popülasyonu, geniş uygulanabilirlik, objektif maliyetle başa çıkma kabiliyeti, uygulama kolaylığı, yerel çözümler özelliklerini keşfetme kabiliyeti ve sağlamlık ABC'nin avantajlarından bazıları olarak kabul edilebilir.

2.2.2. Genetik Algoritma

Biyolojik evrimin doğal süreçlerinden esinlenerek oluşturulmuş çözüm algoritmalarından birisi olan genetik algoritma (genetic algorithm, GA) 1975'te J.Holland ve arkadaşları tarafından geliştirilmiştir [6]. Genel çözümün kromozomlar şeklinde gösterimini takiben, hedef, en iyi uygunluk değerini veren gen dizilişine ulaşılmasıdır. Kullanıcı tarafından belirtilen durdurma kriteri koşulu sağlanıncaya kadar, algoritma çözüm alternatiflerini denemeye ve en uygununu aramaya devam etmektedir.

- 1: Başlangıç popülasyonunu oluştur
- 2: Popülasyondaki her bir çözümün uygunluk değerini hesapla
- 3: **repeat**
- 4: Seleksiyon
- 5: Çaprazlama
- 6: Mutasyon
- 7: **until** Sonlandırma kriteri sağlanana kadar

Algoritma 2.2: Genetik algoritma temel adımları.

GA'nın temel adımları Algoritma 2.2 ile gösterilmektedir [149]. Olası çözümlerin kodlandığı bir çözüm grubu oluşturulur. Popülasyon büyüklüğü olması gerekenden küçük seçilirse çözüm uzayının yeterince araştırılmamasına sebep olurken olması gerekenden büyük seçilmesi durumunda da çözüme ulaşmak uzun zaman almaktadır. Uygunluk fonksiyonu yardımı ile her kromozomun çözüm kalitesinin ne kadar yüksek olduğu bulunur. Popülasyonda, kaliteli çözümlerin artması, kalitesiz çözümlerin azalmasını sağlayan birim seleksiyon operatörüdür. Seçimin yapılabilmesi için kullanılan en temel yöntemler:

- Rulet Tekeri:

Popülasyondaki tüm bireylerin uygunluk değerleri toplanır. Burada her bir kromozomun seçilme şansı, kendi uygunluk değerinin tüm bireylerin uygunluk değerlerinin toplamına oranı kadardır.

- Sıralama:

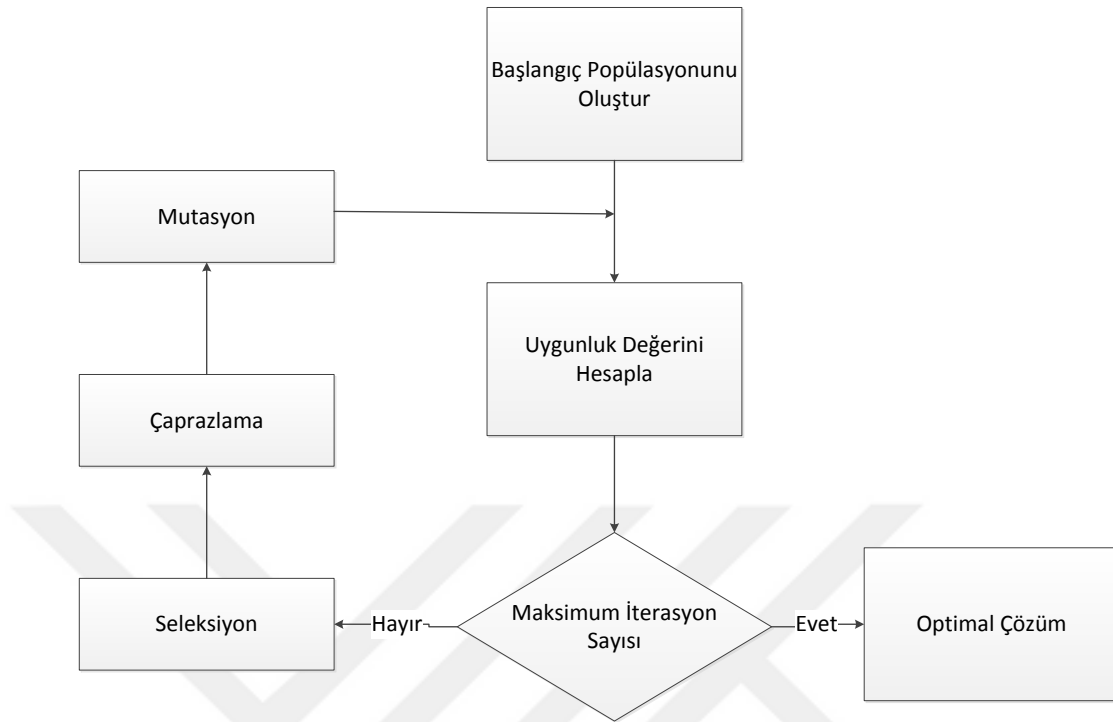
Popülasyondaki bireyler, uygunluk değerlerine göre küçükten büyüğe sıralanır. Bu sıraya göre her bir bireye sıra numarası verilir ve bu numaralara rulet tekeri yöntemi uygulanır. Bu sayede bütün kromozomlara seçilme şansı verilir.

- Turnuva:

Popülasyondaki kromozomlardan belirli bir sayıda birey rastgele olarak seçilir ve bunlardan uygunluk değeri en iyi olan kromozom ilk ebeveyn olarak alınır. Benzer şekilde bu işlem tekrar uygulanarak ikinci ebeveyn seçilir.

Yüksek kaliteye sahip bireylerle çaprazlama yapılarak daha yüksek kaliteli bireylerin üretilmesi hedeflenmektedir. Düşük çaprazlama oranı yakınsama hızını azaltırken yüksek çaprazlama oranı çözüm uzayının hızlı araştırılmasını dolayısıyla kaliteli çözümün çabuk bozulmasına neden olarak algoritmanın performansını düşürür. En önemli adımlardan birisi olan çaprazlamada, bir önceki adım olan seleksiyon aşamasında seçim süreci sonunda oluşan ebeveynler çaprazlama yapılarak yeni bireyler oluşturulur. Burada amaç, ebeveyn kromozom genlerinden uygunluk değeri daha yüksek olan çocuk kromozomlar meydana getirmeye çalışmaktır. En çok kullanılan çaprazlama yöntemi ise tek noktalı çaprazlamadır.

Seçme ve çaprazlama işlemleriyle birbirine benzer çözümler elde edilir. Çaprazlama sonucu oluşan çocuğun kromozomunda genlerden bir ya da daha fazlası yer değiştirilerek genetik çeşitliliği sağlamak ve devam ettirmek için kromozom elde edilir. Mutasyon olasılığı genellikle düşük tutulur ve mutasyon bütün çocuklara uygulanmaz, bu yüzden mutasyon etkileri kromozomlarda az görülür ve kromozomların sayısı değişmez.



Şekil 2.22. GA akış şeması.

GA'nın akış şeması Şekil 2.22 ile verilmiştir [149].

GA'nın performansının değişken olmasındaki en büyük etkenlerden birisi de kontrol parametrelerinin seçimidir. Basit GA'nın başlıca temel parametreleri: popülasyon büyüklüğü, jenerasyon sayısı, çaprazlama oranı ve mutasyon oranıdır.

2.2.3. Diferansiyel Gelişim Algoritması

Price ve Storn tarafından geliştirilen diferansiyel gelişim (differential evolution, DE) algoritması, operatörlerin benzerliği açısından genetik algoritmaya dayanan, popülasyon tabanlı sezgisel bir optimizasyon tekniğidir [13]. GA'da var olan seçim, çaprazlama ve mutasyon adımları DE'de de bulunmaktadır. GA'dan farklı olarak mutasyon işlemi diğer operatörlerden daha önce gelmektedir. GA'dan bir diğer farkı ise DE'nin nümerik olmasıdır.

DE'nin temel adımları Algoritma 2.3 ile gösterilmektedir. DE temel adımlarından ilki, kontrol parametrelerinin belirlenmesidir. Popülasyon büyüklüğü, iterasyon sayısı, alpha,

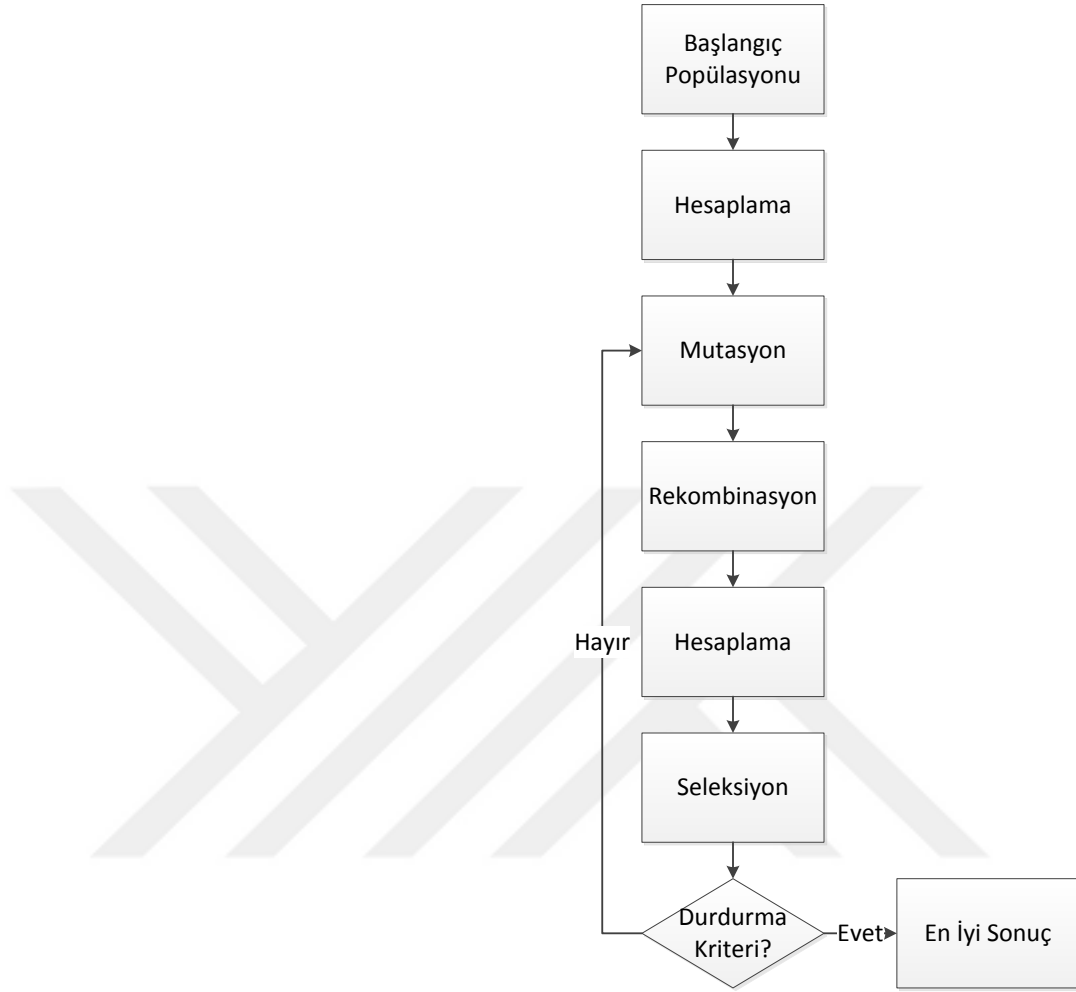
betamin ve gammadan oluşan DE kontrol parametrelerinin değerleri belirlenir. Daha sonra başlangıç popülasyonu oluşturulur. Kullanıcı tarafından belirlenen kromozom sayısı (NP), yeni kromozomların üretilmesi için kendi haricinde üç adet kromozoma daha ihtiyaç duyduğu için en az 4 olmalıdır [13]. Kromozom sayısı (NP) ile problem boyutu (D)'den başlangıç popülasyonu meydana getirilir. Bu aşamadan sonra maksimum iterasyon sayısı sağlanana kadar sırasıyla mutasyon, çaprazlama ve seleksiyon adımları gerçekleştirilir. Mevcut kromozomun bir kısım genleri üzerinde bir kısım rastgele değişim yapılması ve bu sayede araştırma uzayında aranan optimum çözüm için kromozomda farklılık oluşturulması mutasyon adımıyla gerçekleştirilmektedir. Tamamlayıcı bir işlem olan çaprazlama adımının amacı, var olan genlerden faydalanılarak yeni genleri oluşturmak suretiyle problem çözümünde başarılı bir sonuca ulaşmakta yardımcı etken olmaktır. Seleksiyon adımı ile, mevcut nesil ve üretilen yeni nesil kromozomlar değerlendirilip karşılaştırılarak yeni neslin oluşması sağlanır [153]. Seçme işlemindeki faktör, kromozomların uygunluk değerlerine bağlıdır.

- 1: Kontrol parametrelerini belirle
- 2: Başlangıç popülasyonunu oluştur
- 3: Değerlendirme
- 4: **repeat**
- 5: Mutasyon
- 6: Rekombinasyon
- 7: Değerlendirme
- 8: Seleksiyon
- 9: **until** Sonlandırma kriteri sağlanana kadar

Algoritma 2.3: Diferansiyel gelişim algoritması temel adımları.

DE'nin akış şeması Şekil 2.23 ile gösterilebilir [149].

Popülasyon büyüklüğü, iterasyon sayısı, çaprazlama sabiti ve adım boyutu DE'nin temel kontrol parametreleridir.



Şekil 2.23. DE akış şeması.

2.2.4. Parçacık Sürü Optimizasyon Algoritması

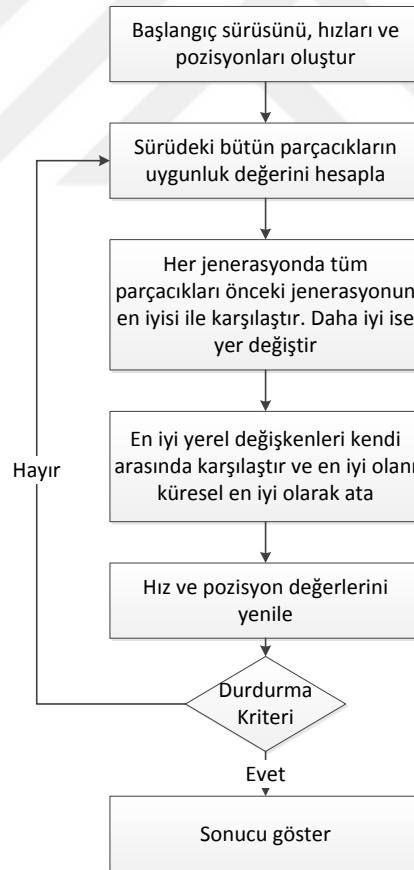
Balık ve kuş sürülerinin sosyal ve bireysel davranışlarından esinlenen parçacık sürü optimizasyon (particle swarm optimization, PSO) algoritması, Eberhart ve Kennedy [7] tarafından önerilmiştir. PSO'daki her bir parçacık olası bir çözümü temsil eder. Olası çözümlerin performansları, maliyet fonksiyonu kullanılarak hesaplanır.

PSO algoritmasının temel adımları Algoritma 2.4 ile sunulmuştur [149]. PSO'nun ilk aşamasında kontrol parametreleri belirlenir. Daha sonra, parçacıkların pozisyonları oluşturulur. Bir sonraki aşamada sürüdeki her bir parçacığın uygunluk değeri hesaplanır. Durdurma kriteri sağlanmışsa, algoritma durdurulur ve en iyi sonuç döndürülür. Aksi

takdirde, parçacıkların uygunluk değerleri değerlendirilir, yerel ve küresel en iyi sonuçlar ve pozisyonlar hafızaya alınır ve parçacıkların hızları ile konumları güncellenir. durdurma kriteri sağlanana kadar Adım 5, 6 ve 7 tekrar edilir. PSO algoritmasına ait iş akışı şeması Şekil 2.24 ile gösterildiği gibidir [149].

- 1: Kontrol parametrelerini belirle
- 2: Başlangıç parçacıklarını oluştur
- 3: Sürüdeki her bir parçanın uygunluk değerini hesapla
- 4: **repeat**
- 5: Parçacık uygunluk değerlerini geçmiş en iyi ile karşılaştır ve daha iyi ise yer değiştir
- 6: Küresel en iyiyi kontrol et
- 7: Hız ve pozisyon değerlerini yenile
- 8: **until** Sonlandırma kriteri sağlanana kadar

Algoritma 2.4: Parçacık sürü optimizasyon algoritmasının temel adımları.



Şekil 2.24. PSO akış şeması.

PSO, öğrenme faktörleri (c_1, c_2), eylemsizlik ağırlığı (w), hız sınırı (V_{max}), popülasyon

büyüklüğü ve iterasyon sayısı olan kontrol parametrelerine sahiptir. Uygulanması basit, sağlamlık, kısa hesaplama süresi, küresel optimalin bulunmasında daha yüksek olasılık ve verime sahip olmak, PSO'nun temel avantajlarıdır.

2.2.5. Hibrit PSO Destekli ABC Algoritması

ABC algoritması yerel ve global arama yönü kuvvetli olan fakat bununla beraber kimi durumlarda geç yakınsama problemine sahip olan bir zeki optimizasyon algoritmasıdır [154]. Kaşif arı fazıyla global arama yeteneğine sahip olsa da yakınsama probleminin çözümü algoritmayı hem daha kullanışlı hem de optimal sonuca daha hızlı yakınsayan hale getirecektir. Global arama yeteneği güçlü olan PSO algoritmasından faydalanmak için PSO destekli ABC temelli hibrit bir yaklaşım (hABCPSO) önerilmektedir.

Bu çalışmada, işçi, gözcü ve kaşif arı fazlarına sahip olan ABC algoritmasına ek olarak PSO fazı eklenmiştir. ABC algoritmasından çıkan yiyecek kaynağı pozisyonları, aynı iterasyonda kullanılmak üzere PSO fazına pozisyon bilgileri olarak aktarılmaktadır. PSO fazından çıkan güncel pozisyon bilgileri ise diğer iterasyonda kullanılmak üzere işçi arı fazına yiyecek kaynağı pozisyonu olarak aktarılmaktadır. Bu temel prensiple çalışan hABCPSO algoritmasına ait temel adımlar Algoritma 2.5 ile gösterildiği gibidir.

2.2.5.1. hABCPSO Algoritması Çalışma Prensibi

hABCPSO algoritmasının ilk adımında öncelikle ABC algoritması başlangıç parametrelerinden koloni büyüklüğü (NP), maksimum döngü ($maxCycle$) ve limit değerleri belirlenir. Daha sonraki adımda PSO fazı için kullanılacak PSO başlangıç parametreleri belirlenir. Bu parametrelerden parçacık sayısı NP değeriyle, iterasyon sayısı $maxCycle$ ile eşitlenir. Ayrıca öğrenme faktörleri, eylemsizlik ağırlığı, hız limiti parametreleri de belirlenir. Adım 3'te, tüm kaşif arılar yiyecek kaynaklarına yönlendirilir. Sonlandırma kriteri sağlanana kadar sırasıyla işçi arı, gözcü arı, kaşif arı ve PSO fazları sırasıyla uygulanır.

İşçi arı fazında, tüm işçi arılar yeni yiyecek kaynaklarına gönderilir. Ayrıca bu fazda Eşitlik 2.7 ile yiyecek kaynağı etrafında lokal arama gerçekleştirilir ve son olarak açgözlü seleksiyon işlemi gerçekleştirilir. Eğer x'_{ij} değeri x_{ij} değerinden daha iyiye çözümler

- 1: ABC algoritması için başlangıç parametrelerinin belirlenmesi
- 2: PSO algoritması için başlangıç parametrelerinin belirlenmesi
- 3: Tüm kaşif arıların yiyecek kaynaklarına yönlendirilmesi
- 4: **repeat**
- 5: İşçi arı fazı
- 6: Görevleri arıların yiyecek kaynaklarına yönlendirilmesi
- 7: Aç gözlü seleksiyon işleminin uygulanması
- 8: Gözcü arı fazı
- 9: Olasılık değerlerinin hesaplanması
- 10: Gözcü arıların yiyecek kaynaklarına gönderilmesi
- 11: Aç gözlü seleksiyon işleminin uygulanması
- 12: Kaşif arı fazı
- 13: En iyi yiyecek kaynağını hafızada tut
- 14: Terk edilecek yiyecek kaynakların kontrol edilmesi
- 15: Gözcü arının kaşif arıya dönüşerek yeni yiyecek kaynağına yönlendirilmesi
- 16: PSO fazı
- 17: Parçacık pozisyonlarının yiyecek kaynak pozisyonları ile güncellenmesi
- 18: Her bir parçacığın uygunluk değerinin hesaplanması
- 19: Yerel ve küresel en iyilerin güncellenmesi
- 20: Parçacıkların hız ve pozisyon güncellemeleri
- 21: Parçacık pozisyonları kullanarak yiyecek kaynaklarının güncellenmesi
- 22: **until** Sonlandırma kriteri sağlanana kadar

Algoritma 2.5: hABCPSO algoritmasının temel adımları.

yeri değiştirilir. Bir sonraki adım olan gözcü arı fazında, yiyecek kaynaklarının olasılık değerlerinin hesaplanması, bu olasılık değerleriyle gözcü arıların yiyecek kaynaklarına gönderilmesi, yeni yiyecek kaynaklarının belirlenmesi ve açgözlü seleksiyon işleminin uygulanması gerçekleştirilir. Olasılık değeri (p_i), Eşitlik 2.8 ile hesaplanır. Kaşif arı fazında ise limit değerini aşan yiyecek kaynağının yerine yeni olası yiyecek kaynağı belirlenir ve bir kaşif arı yeni belirlenen yiyecek kaynağına gönderilir. En iyi yiyecek kaynağı hafızada tutulur. Güncel yiyecek kaynağı bilgileri PSO fazına parçacık pozisyonları olarak atanır. Bu adımla birlikte PSO fazı başlatılmış olur. Her bir parçacık için uygunluk değerleri hesaplanarak yerel ve global en iyi pozisyon ve sonuç değerleri hafızaya alınır. Parçacıkların hız ve pozisyon bilgileri güncellenir. Pozisyon bilgileri yiyecek kaynakları pozisyonuna aktarılır ve bir sonraki iterasyonda kullanılmak üzere işçi arılara yönlendirilir. Durdurma kriteri sağlandığında hafızadaki en iyi sonuç ve pozisyon değeri çıktı olarak verilir.

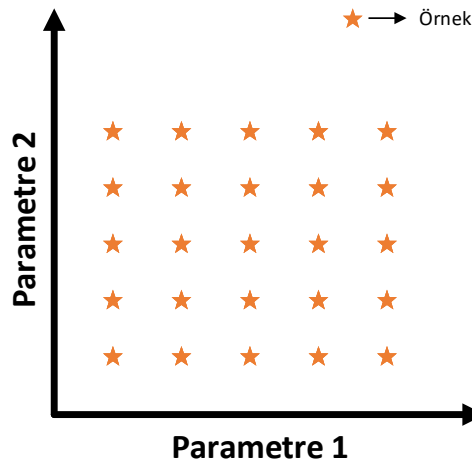
2.2.6. Global Arama Teknikleri

Bu tez çalışmasında, parametre optimizasyonu gerçekleştirilirken sezgisel optimizasyon algoritmalarının yanı sıra global arama yöntemleri de kullanılmıştır. Global arama yöntemlerinden grid ve rastgele arama metotları sırasıyla Bölüm 2.2.6.1 ve 2.2.6.2 ile detaylı bir şekilde anlatılmıştır.

2.2.6.1. Grid Arama Yöntemi

Bir modelin parametrelerini en iyilemek için kullanılan grid arama yöntemi, özellikle optimizasyon işleminin işlem sürecinin kısa olması istenen uygulamalarda sıkça tercih edilmektedir. Grid aramada optimize edilecek her bir parametrenin alt ve üst sınırları belirlenir. Daha sonra, ilgili parametreler için ilgili değer aralıklarında olası farklı değerler belirlenir. Bu değerlerin bütün kombinasyonları (Şekil 2.25 örneğinde gösterildiği gibi) ilgili modele beslenir ve sistem çalıştırılır. Uygunluk değerine göre en iyi sonucu veren parametre seti belirlenir ve grid arama yönteminin çıktısı olarak verilir.

Optimizasyon sürecinde zamansal kazanç sağlamak ve çözüm uzayının eşit aralıklarla aranması grid arama yönteminin bazı avantajları olarak sıralanabilir. Tüm bunların yanında yerel en iyiye takılma riski bu yöntemin dezavantajıdır.

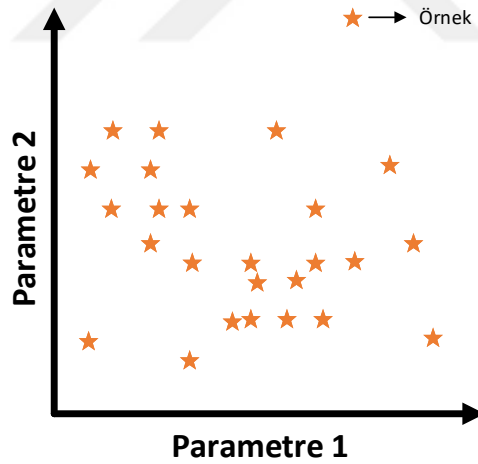


Şekil 2.25. Grid arama için arama uzayındaki parametre setleri.

2.2.6.2. Rastgele Arama Yöntemi

Bir başka global arama yöntemi olan rastgele arama yöntemi, grid arama yönteminde olduğu gibi bir modelin parametrelerini optimize etmek için kullanılır. Optimize edilecek parametrelerin belirlenmesinden sonra her bir parametrenin alt ve üst sınırları belirlenir. Parametre setleri her bir parametre için alt ve üst sınırlar arasında rastgele belirlenmektedir (Şekil 2.26 örneğinde olduğu gibi). Daha sonra belirlenen parametre setleri ilgili modele beslenir ve sistem çalıştırılır. Performans kriterine göre en iyi sonucu veren parametre seti belirlenmiş olur.

Rastgele arama yönteminin zamansal kazanç sağlama becerisinin yanında düşük bir ihtimal dahilinde olsa bile global en iyiyi bulma ihtimali vardır. Bu avantajlara karşılık, parametre değerleri alt üst sınırlar aralığında rastgele seçildiğinden dolayı çözüm uzayındaki aramalar belirli noktalara sıkışıp çözüm uzayının her bölgesinde arama gerçekleştirilemeyebilir.



Şekil 2.26. Rastgele arama için arama uzayındaki parametre setleri.

3. BÖLÜM

DERİN ÖĞRENME İLE İŞARET DİLİ TANIMA

Bilimsel çalışmalarda insan hareketi tanıma (human action recognition, HAR) önemli bir yere sahiptir. HAR'ın bir alt kategorisi olan el hareketi tanıma (hand gesture recognition, HGR), duyma engelli insanlarla iletişimde önemli bir rol oynamaktadır. Son yıllarda derin öğrenme yöntemleri ile el işareti tanıma üzerine başarılı çalışmalar gerçekleştirilmiştir. Tezin bu bölümünde derin öğrenme yöntemi kullanılarak işaret dili tanıma üzerine iki farklı çalışma uygulanmıştır.

3.1. El Hareketlerini Tanımak İçin Sezgisel Optimizasyon ile Transfer Öğrenme Tabanlı Evrişimli Sinir Ağları

3.1.1. Giriş

Örüntü tanıma, sinyal işleme, bilgisayarla görme, insan-bilgisayar etkileşimi gibi oldukça geniş araştırma alanlarına sahip olan HGR çalışmaları son zamanlarda popüler hale gelmiştir. Temel amacı, duyma yetisini kaybetmiş insanlarla iletişimi sağlamak olan HGR çalışmalarında ilk aşama olarak harf ve rakamlar üzerinden işaret dili tanıma gerçekleştirilmektedir. Popüler derin öğrenme yöntemlerinden CNN [25, 27, 29, 31], SAE [31] ve RNN [128] bu alanda sıklıkla kullanılan derin öğrenme yöntemleridir.

Bu uygulamada, statik imge tabanlı el hareketleri ile işaret dili tanıma, CNN'lere dayalı derin bir öğrenme yöntemi ile gerçekleştirilecektir. Yeni bir CNN modeli geliştirmek, zaman alıcı ve zorlu bir süreç gerektirdiğinden, yeni bir model oluşturmak yerine, transfer öğrenme yöntemlerini kullanmak yalnızca başarıda bir artışa yol açmayacak, aynı zamanda zamansal maliyetin düşmesinde de yardımcı olacaktır. Ön eğitilmiş bir model kullanmak, yeni bir CNN modeli geliştirmek yerine kolay ve hızlı bir yöntemdir. AlexNet

[4] gibi ön eğitilmiş modeller, sınıflandırma, transfer öğrenme ve özellik çıkarma gibi amaçlar için kullanılabilir. AlexNet ile transfer öğrenimi, önerilen yöntemin ilk ana kısmı olarak kullanılmıştır. Bu şekilde, kenarlar, şekiller gibi el hareketlerinin düşük seviyeli özellikleri hızlı ve etkili bir teknikle ortaya çıkarılır. Önerilen sistemin ikinci bölümünde, AlexNet destekli CNN'lerin hiperparametreleri ABC [5], GA [6] ve PSO [7] algoritmaları tarafından optimize edilmiştir. Maksimum epok, minibatch boyutu, ilk öğrenme oranı, ℓ_2 düzenleştirme parametresi, karıştırma tipi ve momentum değeri optimize edilecek olan CNN'lerin hiperparametreleridir. Önerilen yöntemin test edilmesi için genel kullanıma açık yaygın kullanılan iki veri seti tercih edilmiştir. Bu veri setleri işaret dili rakamlar veri seti (sign language digits dataset, SLDD) [8] ve Thomas Moeslund'un mimik tanıma veri setidir (Thomas Moeslund's gesture recognition dataset, TMGRD) [9]. İlgili veri setleri için detaylı bilgiler, Bölüm 3.1.2 ile verilmiştir. SLDD veri seti için %98,40 doğruluk oranı elde edilirken TMGRD veri seti için %98,09 doğruluk oranı elde edilmiştir. Bu çalışmanın ana katkıları şu şekilde sıralanabilir [155]:

- Bu çalışmada, HGR için transfer öğrenmeye dayalı ve hiperparametre optimizasyonu destekli bir CNN önerilmiştir. Önerilen CNN'in hiperparametreleri, ABC, GA ve PSO olmak üzere üç farklı optimizasyon algoritması kullanılarak ayarlanmıştır. Bilinen kadarıyla, bu çalışma literatürde HGR için bu yaklaşımı kullanan ilk çalışmadır. Önerilen CNN'lerin performansları istatistiksel testler kullanılarak karşılaştırılmıştır. İstatistiksel testler ve doğruluk oranı karşılaştırma metriğine göre en başarılı yöntemin ABC destekli olduğu sonucuna varılmıştır.
- Önerilen yöntem ile SLDD veri seti üzerinde %98,4 başarı oranı elde edilip, literatürde bilinen ve %94,2 en iyi başarı oranına sahip olan çalışma [10] geride bırakılmıştır.
- Önerilen yöntem ile TMGRD veri seti üzerinde %98,09 başarı oranı elde edilip, literatürde bilinen ve %95,33 en iyi başarı oranına sahip olan çalışma [11] geride bırakılmıştır.

Bu çalışma için gerçekleştirilen literatür taraması Bölüm 1.1 ile detaylı olarak sunulmuştur. Çalışmanın bundan sonraki planlaması şu şekildedir: Bölüm 3.1.2 ile problem çözümünde kullanılacak olan veri setleri sunulurken, Bölüm 3.1.3 ile önerilen

yöntem detaylı olarak anlatılmıştır. Bölüm 3.1.4 ile deneysel çalışmalar ve sonuçlardan bahsedilmiş olup Bölüm 3.1.5 ile çalışmanın değerlendirilmesi yapılmıştır.

3.1.2. Kullanılan Veri Setleri

Bu çalışmada önerilen yöntem sign languages digits ve TMGRD veri setleri ile analiz edilecektir. İlgili veri setlerine ait detaylı bilgiler alt bölümlerde sunulmuştur.

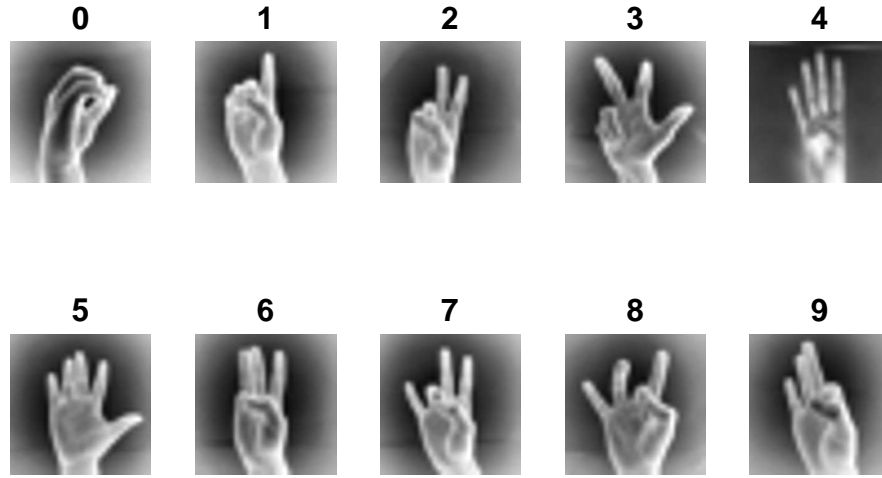
3.1.2.1. SLDD Veri seti

SLDD veri seti 0'dan 9'a kadar olan rakamların imgelerinden oluşmaktadır. Orijinal imgelerin boyutu 100×100 şeklindedir. 218 farklı gönüllünün hareketleri 10'ar defa yapmasıyla oluşturulmuştur [8]. Her bir rakama ait örnek sayısı Tablo 3.1 ile gösterildiği gibidir.

SLDD veri setinde var olan ön işlemden geçirilmiş örnek imgeler Şekil 3.1 ile gösterildiği gibidir. Veriler %70 eğitim ve %30 test için olacak şekilde [10] çalışmasındaki gibi ayrılmıştır.

Tablo 3.1. SLDD veri setindeki rakamların toplam örnek sayıları.

Rakam	Örnek Sayısı	Rakam	Örnek Sayısı	Rakam	Örnek Sayısı
0	205	4	207	8	208
1	206	5	207	9	204
2	206	6	207		
3	206	7	206		



Şekil 3.1. SLDD veri seti örnekleri.

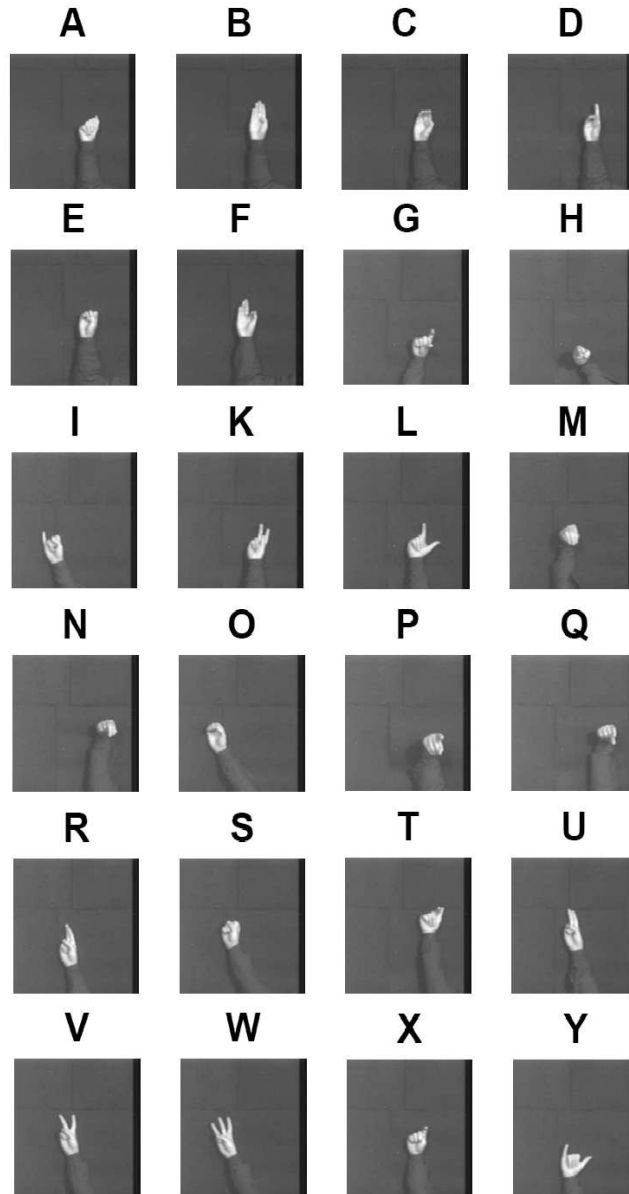
3.1.2.2. TMGRD Veri seti

TMGRD veri seti toplamda 2060 tekil imgeden oluşmaktadır. Bu veri seti 24 Amerikan işaret dili harfini içermektedir. Diğer 'J' ve 'Z' harflerini ifade edebilmek için çoklu imgeye ihtiyaç duyulduğundan bu harfler veri seti dışında tutulmuştur. İmgeler veri seti içerisinde gri formatta ve 248×256 çözünürlükte bulunmaktadır [9]. Her bir harfe ait toplam örnek sayısı Tablo 3.2 ile gösterildiği gibidir.

Çalışmada veri bölme işlemi 70% eğitim ve 30% test olacak şekilde [11] çalışmasındaki gibi ayrılmıştır. TMGRD veri setine ait her bir sınıf için örnek imge içeren yapı Şekil 3.2 ile gösterildiği gibidir.

Tablo 3.2. TMGRD veri setindeki harflerin toplam örnek sayıları.

Harf	Örnek Sayısı	Harf	Örnek Sayısı	Harf	Örnek Sayısı
A	40	I	100	R	100
B	60	K	100	S	100
C	40	L	100	T	100
D	40	M	100	U	100
E	40	N	100	V	100
F	40	O	100	W	100
G	100	P	100	X	100
H	100	Q	100	Y	100



Şekil 3.2. TMGRD veri seti örnekleri.

3.1.3. Metodoloji

CNN, sınıflandırma problemleri için popüler bir derin öğrenme metodudur. Ön eğitilmiş CNN modellerinden transfer öğrenme gerçekleştirmek, yeni bir CNN mimarisi inşa etmenin kolay bir yöntemidir. Ayrıca, transfer öğrenmeye dayalı CNN modelinin, optimuma yakın sonuçlara ulaşmak için hiperparametrelerin optimizasyonuna ihtiyaç duyulmaktadır. Hiperparametreler doğrudan performansı etkilediğinden, çözüm uzayını etkili şekilde arayabilen optimizasyon algoritmaları, hiperparametre seçimi için

kullanılabilir. Bu çalışmada kullanılan CNN metodu Bölüm 2.1.1 ile, transfer öğrenme için kullanılan AlexNet Bölüm 2.1.1.1 ile, hiperparametre optimizasyonunda kullanılan ABC, PSO ve GA algoritmaları sırasıyla Bölüm 2.2.1, 2.2.4 ve 2.2.2 ile detaylı olarak sunulmuştur.

3.1.3.1. Önerilen Yöntem

Bu çalışmada sezgisel optimizasyon algoritması destekli, AlexNet tabanlı CNN modeli kullanan bir sistem önerilmiştir. Önerilen yöntemin temel adımları Algoritma 3.1 ile verilmiştir. Algoritmanın ilk adımı için ilgili veri seti sisteme yüklenir. Ardından, yüklenen veri seti, literatürdeki mevcut çalışmalardaki kurallara uygun olarak, `train` ve `test` verisi olarak bölünür. Bir sonraki adımda, hiperparametre optimizasyonu için kullanılacak olan `train` veri seti; `trainOpt` ve `validationOpt` olarak iki yeni alt veri setine bölünür. 4. adımda, AlexNet ön eğitilmiş CNN modeli transfer öğrenimi için yüklenir. Bu yapının son 3 katmanı, sınıflandırma problemine göre güncellenir. `trainOpt`, `validationOpt` veri setleri ve CNN yapısı, görevlendirilen optimizasyon algoritması için giriş parametreleri olarak kullanılır. Hiperparametre ayarlayan optimizasyon algoritması, 7. ve 11. adımlar arasında çalıştırılır. Kullanılan optimizasyon algoritması tarafından üretilen her bir çözüm vektörü, optimize edilecek olası hiperparametre değerlerinden oluşur. Olası çözümdeki hiperparametre değerleri, CNN'nin eğitim süreci için eğitim parametreleri olarak kullanılır. Kullanılan optimizasyon algoritması tarafından üretilen hiperparametre değerleri seti için uygunluk değeri, maliyet fonksiyonu kullanılarak hesaplanır. Çalışmada kullanılan maliyet fonksiyonu Eşitlik 3.1 ile verilmiştir. Adım 12 ile görevli optimizasyon algoritması tarafından hafızada tutulan en iyi parametre seti çıktı olarak alınır. Optimize edilmiş hiperparametre değerleri ilk oluşturulan `train` veri seti ile eğitilir. Son adımda eğitilmiş model `test` verileri üzerinde test edilir ve yöntemin performansı Eşitlik 3.2 ile ölçülür.

$$\text{Hata} = 1 - \frac{\text{Doğrulama seti doğru tahmin sayısı}}{\text{Doğrulama seti tüm örnek sayısı}} \quad (3.1)$$

Adım

$$\text{Doğruluk} = \frac{\text{Test seti doğru tahmin sayısı}}{\text{Test seti tüm örnek sayısı}} \quad (3.2)$$

- 1: Veri setini yükle
- 2: Veri setini `train` ve `test` olarak böl
- 3: (`train`) eğitim verisini `trainOpt` ve `validationOpt` olarak böl
- 4: AlexNet ön eğitimli modeli yükle
- 5: Son üç katmanı ilgili probleme uygun olarak düzenle
- 6: `trainOpt`, `validationOpt` ve transfer öğrenme tabanlı CNN yapısını görevli optimizasyon algoritmasına giriş parametreleri olarak gönder
- 7: Görevli optimizasyon algoritmasının kontrol parametrelerini ata
- 8: Hiperparametrelerin alabilecekleri alt ve üst limitleri belirle
- 9: **repeat**
- 10: Görevli optimizasyon algoritmasının fazlarını gerçekleştir.
- 11: **until** Sonlandırma kriteri sağlanır
- 12: Optimize edilmiş hiperparametre değerlerini çıktı olarak ver
- 13: Optimize edilmiş hiperparametre değerlerini CNN mimarisine gönder ve modeli orijinal eğitim verisi (`train`) ile ölçekli eşlenik gradyan (scaled conjugate gradient, SCG) algoritması kullanılarak eğit
- 14: Test verisini (`test`) kullanarak yöntemin performansını hesapla

Algoritma 3.1: Optimizasyon algoritmaları kullanarak CNN yapıları için önerilen hiperparametre ayarlama mimarisi.

ABC algoritmasının hiperparametre ayarlama algoritması olarak kullanıldığı önerilen yöntemin ayrıntılı çalışma prensibi Şekil 3.3 ile sunulmuştur.

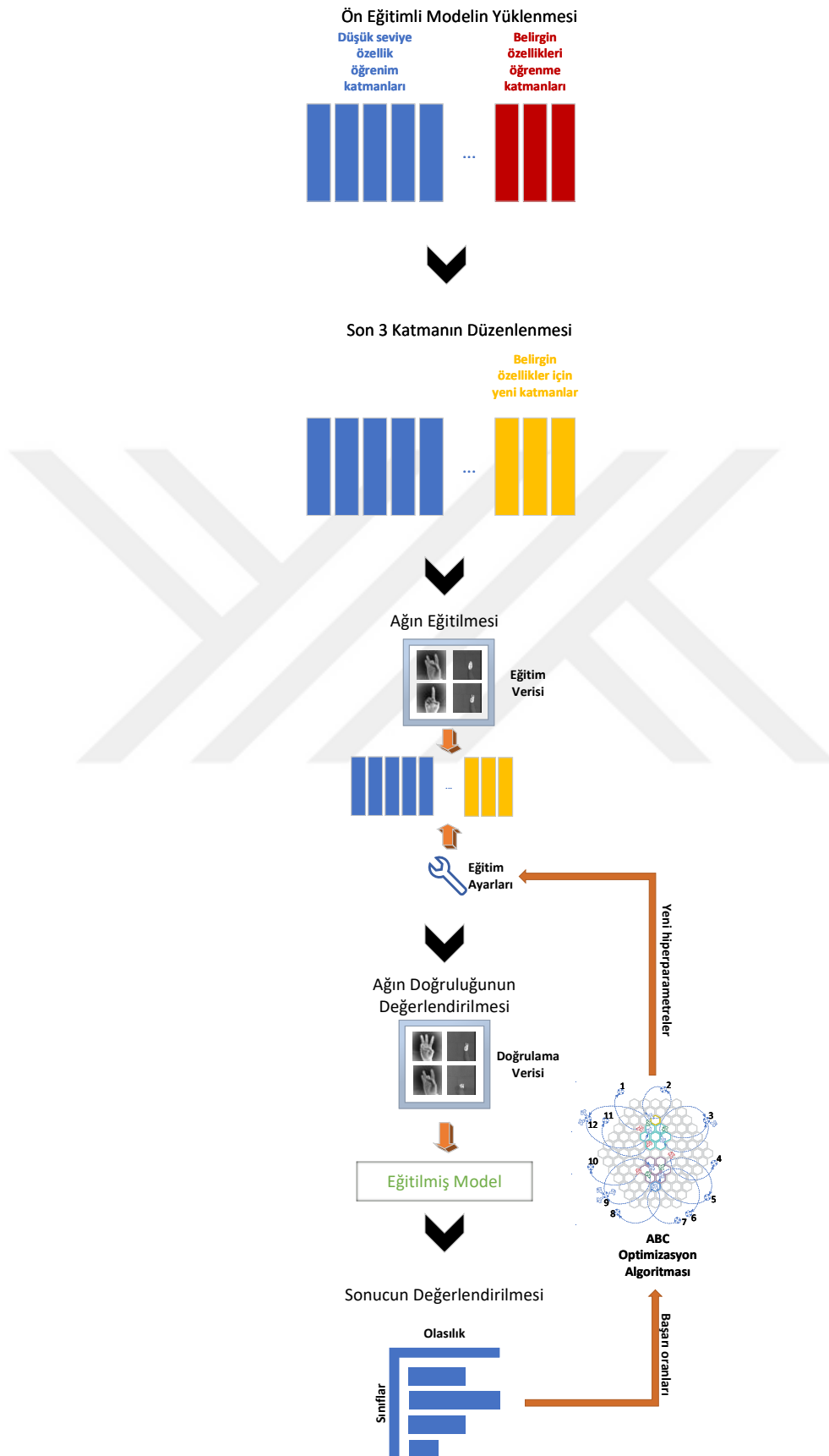
3.1.4. Deneysel Çalışmalar

3.1.4.1. Deneysel Kurulum

Önerilen yöntemin çalışması için yöntemde kullanılacak olan algoritmalara ait başlangıç parametreleri bu bölümde sunulmuştur. Yöntemde kullanılan algoritmalara ait başlangıç parametreleri Tablo 3.3, 3.4, ve 3.5 ile gösterildiği gibidir.

Tablo 3.3. ABC optimizasyon algoritması kontrol parametreleri.

Parametre	Açıklama	Değer
NP	Koloni büyüklüğü	20
$maxCycle$	Maksimum döngü sayısı	10
$limit$	Genellikle $NP * D/2$	60



Şekil 3.3. ABC optimizasyon algoritması hiperparametre ayarlayıcı olarak kullanıldığında önerilen yöntemin akış şeması.

Tablo 3.4. GA optimizasyon algoritması kontrol parametreleri.

Parametre	Açıklama	Değer
<i>ps</i>	Popülasyon büyüklüğü	20
<i>gen</i>	Jenerasyon sayısı	10
<i>cf</i>	Çaprazlama oranı	0.85
<i>mf</i>	Mutasyon oranı	0.15

Tablo 3.5. PSO optimizasyon algoritması kontrol parametreleri.

Parametre	Açıklama	Değer
<i>SwarmSize</i>	Parçacık sayısı	20
<i>MaxIter</i>	İterasyon sayısı	10
<i>c₁</i>	Öğrenme faktörü 1	1.49
<i>c₂</i>	Öğrenme faktörü 2	1.49
<i>w</i>	Durağanlık ağırlığı	0.7

Adil bir karşılaştırma yapılabilmesi için $NP * maxCycle$, $ps * gen$, ve $SwarmSize * MaxIter$ değerleri eşit tutulmuştur.

3.1.4.2. Deneysel Sonuçlar

Önerilen yöntem SLDD ve TMGRD veri setleri üzerinde test edilerek sonuçlar ayrıntılı olarak sunulmuştur. SLDD veri seti üzerinde gerçekleştirilen ABC, GA, ve PSO algoritmaları için 30'ar koşmaya ait doğruluk oranları Tablo 3.6 ile gösterildiği gibidir.

Tablo 3.6. SLDD veri seti üzerinde optimizasyon algoritmaları için 30'ar koşmaya ait doğruluk oranları.

Koşma	Doğruluk (%)			Koşma	Doğruluk (%)		
	ABC	GA	PSO		ABC	GA	PSO
1	99,27	98,31	98,31	16	98,79	98,31	98,55
2	98,79	98,55	98,31	17	96,13	96,61	98,79
3	98,31	99,27	97,34	18	97,09	98,55	98,31
4	98,79	99,03	95,88	19	99,27	96,61	95,88
5	96,13	98,31	98,06	20	98,79	99,03	97,82
6	97,09	98,55	99,27	21	98,31	99,27	95,88
7	99,03	99,03	98,55	22	99,03	97,82	98,55
8	98,31	99,52	97,82	23	98,31	98,31	98,06
9	98,79	98,06	99,52	24	98,79	97,82	95,88
10	98,79	98,55	98,55	25	98,06	98,31	99,03
11	98,79	97,82	99,76	26	98,79	97,09	99,27
12	98,06	97,09	97,82	27	98,79	99,03	98,55
13	99,03	98,55	98,31	28	98,79	99,27	98,31
14	98,06	98,06	98,55	29	98,06	97,82	98,31
15	98,79	98,06	99,03	30	99,03	98,31	98,55
ABC koşmaları için ortalama başarı					98,40		
GA koşmaları için ortalama başarı					98,30		
PSO koşmaları için ortalama başarı					98,16		

Optimizasyon algoritmaları tarafından seçilen en iyi parametreler ve bu parametreler kullanılarak önerilen yöntemin çalıştırılması ile elde edilen sonuçlar Tablo 3.7 ile sunulmuştur.

Tablo 3.7. Optimizasyon algoritmaları tarafından elde edilen en iyi hiperparametre değerleri ve bu hiperparametreleri SLDD veri seti için kullanan CNN'lerin başarı oranları.

Parametreler	ABC	GA	PSO
Maksimum epok	15	11	13
Minibatch boyutu	64	32	32
İlk öğrenme oranı	0,0054746716	0,0016000561	0,001
l_2 düzenleme	0,0004795852	0,0006019928	0,0008044368
Karıştırma tipi	bir kere	her epokta	bir kere
Momentum	0,885783117	0,8417708737	0,7893671359
Başarı Oranı	99,27%	99,52%	99,76%

Tablo 3.8 ile önerilen ve mevcut yöntemlerin doğruluk karşılaştırması sunulmaktadır.

Burada, ortalama doğruluk değerleri başarı metriği olarak kullanılmıştır.

Tablo 3.8. SLDD veri seti için önerilen yöntemlerin ve mevcut yöntemin performans karşılaştırmaları.

Yöntem	Başarı Oranı (%)
Beser et. al. (Capsule Network) [10]	94,20
Önerilen (ABC tabanlı)	98,40
Önerilen (GA tabanlı)	98,30
Önerilen (PSO tabanlı)	98,16

Elde edilen hata değerleri (en küçük, en büyük, ortalama), standart sapma, hesaplama süresi ve algoritmaların doğrulama veri seti üzerinde elde edilen 30 koşmaya ait ortalama maliyet fonksiyonu değeri (Mean Fval) Tablo 3.9 ile gösterildiği gibidir. Tablo 3.9'e göre, önerilen optimizasyon algoritması tabanlı yöntemlerin tüm çalışmalarının en iyi minimum ve maksimum hata değerleri PSO algoritması ile elde edilmiştir. Tablodan, ABC algoritmasının ortalama hata kriterine göre (0,0160 değeri ile) en başarılı yöntem olduğu söylenebilir. 0,0075 standart sapma değerine sahip GA, bu veri seti için en dirençli algoritmadır. Hesaplama zamanı başka bir performans kriteridir ve bu kriteri göre ABC algoritması en hızlı çalışan algoritmadır. Son olarak, PSO algoritması en küçük ortalama fval değerine sahiptir.

Tablo 3.9. SLDD veri seti için algoritmalara ait hata, standart sapma, hesaplama süresi (sn) ve ortalama fval değerleri.

	Algoritmalar		
	ABC	GA	PSO
Minimum Hata	0,0073	0,0048	0,0024
Maksimum Hata	0,0387	0,0339	0,0412
Ortalama Hata	0,0160	0,0170	0,0184
Standart Sapma	0,0081	0,0075	0,0105
Hesaplama Süresi	13979	14772	19809
Ortalama Maliyet	0,0048	0,0148	0,0021

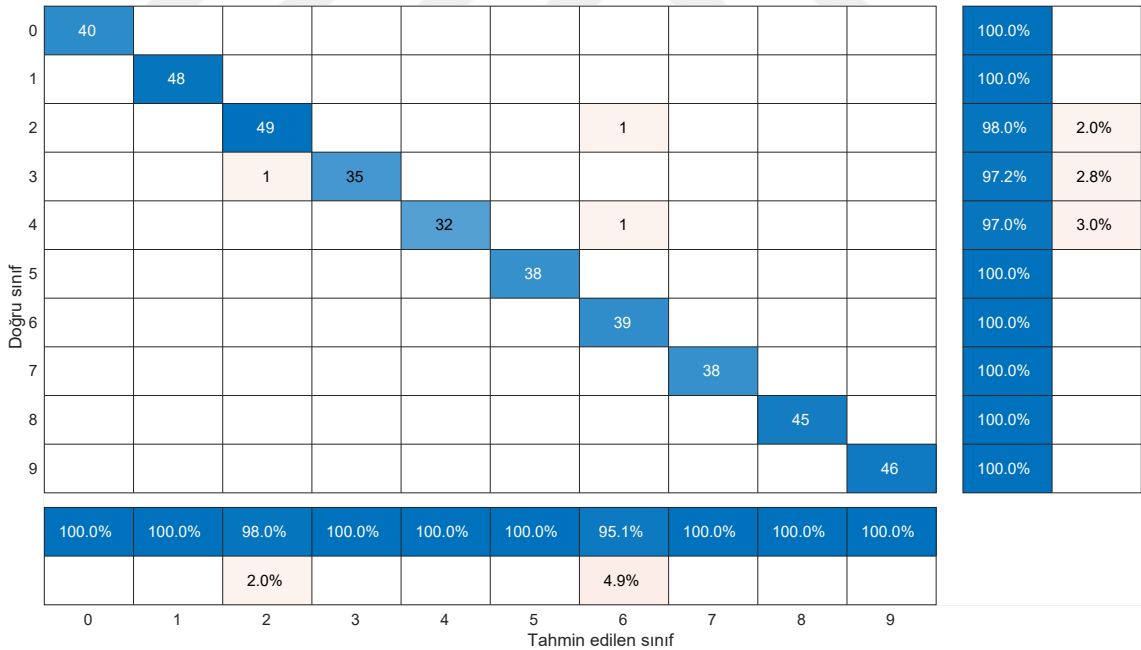
SLDD veri seti için önerilen yöntemde kullanılan tüm algoritmaların algoritma karmaşıklığı, [156] çalışmasında verilen T_1/T_0 yöntemiyle hesaplanır. İlk olarak, [156] çalışmasında sunulan test programı ile T_0 değeri hesaplanmıştır. Daha sonra, her algoritma için T_1 ortalama hesaplama süresi değerleri ölçülür. Son olarak, T_1/T_0 eşitliği kullanılarak algoritmalara ait karmaşıklık değerleri hesaplanmıştır. Elde edilen sonuçlar

Tablo 3.10 ile sunulmuştur. Karmaşıklık sonuçlarına göre, ABC optimizasyon algoritması en küçük algoritma karmaşıklığı değerine sahiptir.

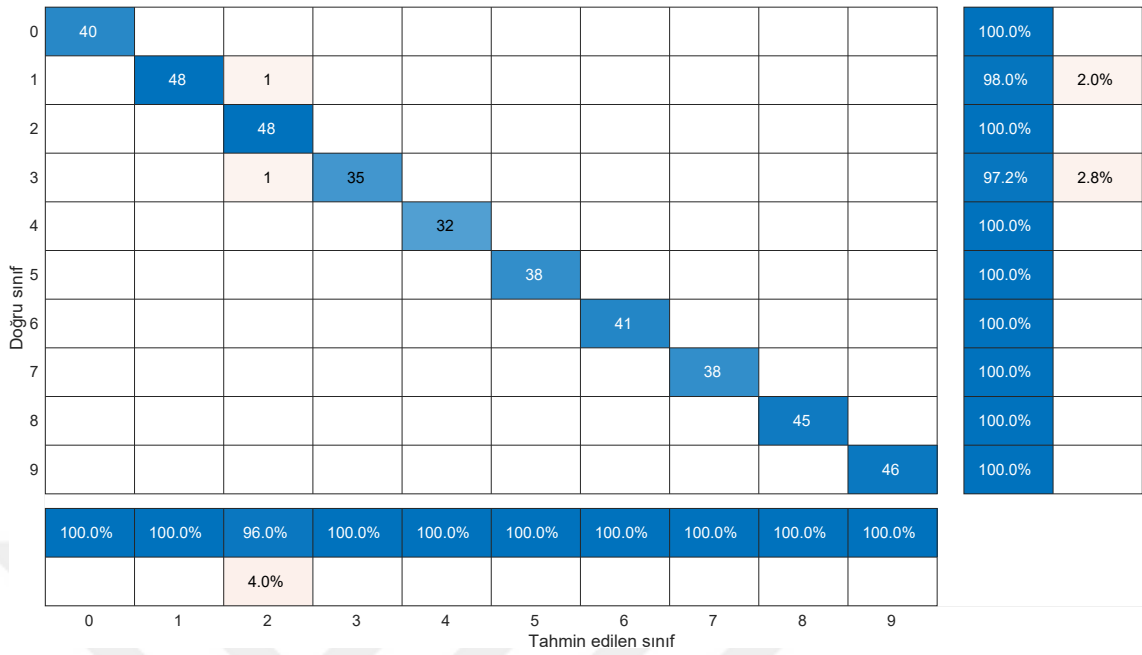
Tablo 3.10. $T_0 = 0,076805$ eşitliği elde edildiği durumda SLDD veri seti için algoritma karmaşıklık değerleri.

Algoritma	T_1	Algoritma Karmaşıklığı
ABC	13979	182000
GA	14722	191682
PSO	19809	257909

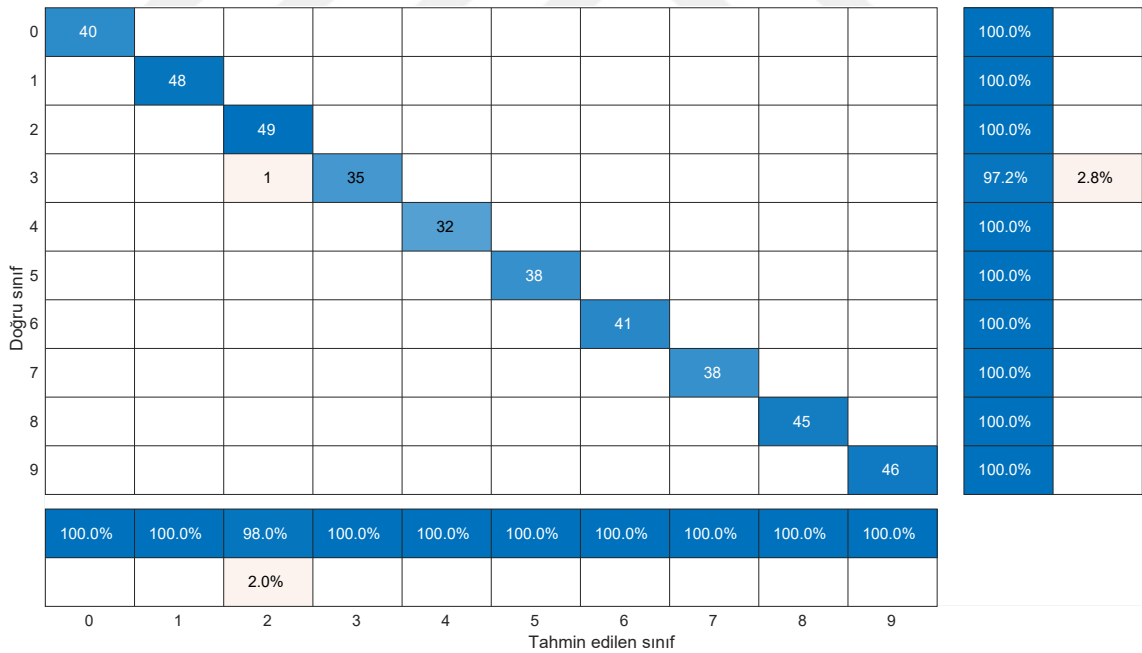
Her rakam için doğruluk oranları, karmaşıklık matrisleriyle de ifade edilebilmektedir. Hiperparametreleri SLDD veri seti için optimizasyon algoritmalarıyla (ABC, GA, PSO) ayarlanan AlexNet tabanlı CNN'lerin en iyi performanslara göre karmaşıklık matrisleri sırasıyla Şekil 3.4, 3.5 ve 3.6 ile sunulurken en kötü performanslara ait karmaşıklık matrisleri sırasıyla Şekil 3.7, 3.8 ve 3.9 ile verilmiştir.



Şekil 3.4. AlexNet + ABC destekli CNN metodunun SLDD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.



Şekil 3.5. AlexNet + GA destekli CNN metodunun SLDD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.



Şekil 3.6. AlexNet + PSO destekli CNN metodunun SLDD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.

0	40										100.0%	
1		46									100.0%	
2		2	49	1							94.2%	5.8%
3			1	33							97.1%	2.9%
4					32		2	1		1	88.9%	11.1%
5						38				1	97.4%	2.6%
6				1			39				97.5%	2.5%
7								36	4		90.0%	10.0%
8								1	40		97.6%	2.4%
9									1	44	97.8%	2.2%
	100.0%	95.8%	98.0%	94.3%	100.0%	100.0%	95.1%	94.7%	88.9%	95.7%		
		4.2%	2.0%	5.7%			4.9%	5.3%	11.1%	4.3%		
	0	1	2	3	4	5	6	7	8	9		

Şekil 3.7. AlexNet + ABC destekli CNN metodunun SLDD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.

0	40										100.0%	
1		48							1		98.0%	2.0%
2			48								100.0%	
3			1	35							97.2%	2.8%
4					32		1				97.0%	3.0%
5						38					100.0%	
6			1				40				97.6%	2.4%
7								38	8		82.6%	17.4%
8									34		100.0%	
9									2	46	95.8%	4.2%
	100.0%	100.0%	96.0%	100.0%	100.0%	100.0%	97.6%	100.0%	75.6%	100.0%		
			4.0%				2.4%		24.4%			
	0	1	2	3	4	5	6	7	8	9		

Şekil 3.8. AlexNet + GA destekli CNN metodunun SLDD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.

0	40										100.0%	
1		47	4								92.2%	7.8%
2		1	43								97.7%	2.3%
3			1	35							97.2%	2.8%
4					30				1	1	93.8%	6.2%
5						38					100.0%	
6					2		40				95.2%	4.8%
7			2				1	37	1		90.2%	9.8%
8								1	43	2	93.5%	6.5%
9										43	100.0%	
	100.0%	97.9%	86.0%	100.0%	93.8%	100.0%	97.6%	97.4%	95.6%	93.5%		
		2.1%	14.0%		6.2%		2.4%	2.6%	4.4%	6.5%		
	0	1	2	3	4	5	6	7	8	9		

Şekil 3.9. AlexNet + PSO destekli CNN metodunun SLDD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.

İstatistiksel analiz yapmak amacıyla, SLDD veri setindeki tüm hata değerleri için Kolmogorov-Smirnov normallik testi [157] yapılmıştır. Tablo 3.11 incelendiğinde, ABC ve PSO algoritmalarının anlamlılık değerleri 0,05'ten küçük olduğu için hata değerlerinin dağılımının normal olmadığı görülmektedir. Bu nedenle, parametrik olmayan testler arasından Kruskal-Wallis [158] ve Mann-Whitney U [159] testleri, yöntemlerin performansları arasında anlamlı bir fark olup olmadığını belirlemek için kullanılmıştır.

Kruskal-Wallis testi, algoritmalar arasında hata dağılımları açısından anlamlı fark olup olmadığını kontrol etmek için kullanılır. Bu testin sonuçları Tablo 3.12 ile verilmiştir. Tablo incelendiğinde anlamlılık değeri (p) 0,40398'e eşit ve 0,05'ten büyük olduğundan algoritmalar arasında anlamlı bir fark olmadığı görülmüştür. Mann-Whitney U testi yapılarak, ikili algoritmalar arasında anlamlı bir fark olup olmadığını tespit edilebilir. Tablo 3.13 incelendiğinde Mann-Whitney U test sonuçlarına göre, SLDD veri seti için kullanılan algoritmalar arasında istatistiksel olarak anlamlı bir fark olmadığı ifade edilebilir. Bu veri seti için herhangi iki algoritma arasında anlamlı fark olmadığı görülmektedir. Tablo incelendiğinde, 0,952367 anlamlılık değeri ile en çok GA ve PSO temelli sonuçlar arasında benzerlikler bulunmaktadır.

Tablo 3.11. SLDD veri seti için Kolmogorov-Smirnov normalite test sonuçları.

Algoritma	İstatistik	Anlamlılık Değeri
ABC	0,283	0,000
GA	0,140	0,138
PSO	0,223	0,001

Tablo 3.12. SLDD veri seti için Kruskal-Wallis parametrik olmayan test sonuçları.

Hata	
Chi-square	1,813
Anlamlılık değeri (p)	0,403980

Tablo 3.13. SLDD veri seti için Mann-Whitney U parametrik olmayan test sonuçları.

Algoritmalar	Anlamlılık Değeri (p)
ABC & GA	0,332133
ABC & PSO	0,179153
GA & PSO	0,952367

TMGRD veri seti üzerinde gerçekleştirilen ABC, GA, ve PSO algoritmaları için 30'ar kořmaya ait doğruluk oranları Tablo 3.14 ile gösterildiđi gibidir.

Tablo 3.14. TMGRD veri seti üzerinde optimizasyon algoritmaları için 30’ar koşmaya ait doğruluk oranları.

Koşma	Doğruluk (%)			Koşma	Doğruluk (%)		
	ABC	GA	PSO		ABC	GA	PSO
1	98,53	98,53	98,37	16	98,20	97,22	97,22
2	97,39	97,88	97,88	17	98,04	98,37	99,51
3	98,86	96,57	97,55	18	98,53	98,86	97,55
4	98,20	98,86	98,86	19	98,53	96,73	96,73
5	98,04	94,61	98,69	20	97,39	94,28	98,04
6	98,53	98,86	97,71	21	98,86	97,88	96,08
7	98,53	98,20	96,73	22	99,51	97,39	95,10
8	97,39	97,55	99,18	23	97,22	96,90	97,88
9	98,53	94,28	97,22	24	98,53	95,92	98,86
10	99,51	98,37	98,37	25	98,53	96,57	97,71
11	97,22	97,71	99,84	26	97,39	95,75	99,18
12	96,57	97,22	98,20	27	98,53	95,92	98,37
13	97,71	96,73	96,73	28	97,88	96,57	99,51
14	97,88	97,71	95,10	29	96,57	98,20	96,73
15	98,53	97,55	96,41	30	97,71	96,73	98,04
	ABC koşmaları için ortalama başarı				98,09		
	GA koşmaları için ortalama başarı				97,13		
	PSO koşmaları için ortalama başarı				97,78		

Optimizasyon algoritmaları tarafından seçilen en iyi parametreler ve bu parametreler kullanılarak önerilen yöntemin çalıştırılması ile elde edilen sonuçlar Tablo 3.15 ile sunulmuştur.

Tablo 3.15. Optimizasyon algoritmaları tarafından elde edilen en iyi hiperparametre değerleri ve bu hiperparametreleri TMGRD veri seti için kullanan CNN’lerin başarı oranları.

Parametreler	ABC	GA	PSO
Maksimum epok	21	15	25
Minibatch boyutu	128	64	128
İlk öğrenme oranı	0,0042623148	0,0037952371	0,0060928603
l_2 düzenleme	0,0009752196	0,0006271655	0,0008931835
Karıştırma tipi	bir kere	her epokta	her epokta
Momentum	0,9154817795	0,8230500711	0,9277321597
Başarı Oranları	99,51%	98,86%	99,84%

Tablo 3.16 ile önerilen ve mevcut yöntemlerin doğruluk karşılaştırmasını

sunmaktadır. Burada, diğer veri setinde olduğu gibi ortalama doğruluk değerleri başarı metriği olarak kullanılmıştır.

Tablo 3.16. TMGRD veri seti için önerilen yöntemlerin ve mevcut yöntemlerin performans karşılaştırmaları.

Yöntem	Başarı Oranı (%)
Oyedotun, Khashman (SDAE) [31]	92,83
Oyedotun, Khashman (CNN) [31]	91,33
Oyedotun, Khashman (PI-EmNN) [11]	94,33
Önerilen (ABC tabanlı)	98,09
Önerilen (GA tabanlı)	97,13
Önerilen (PSO tabanlı)	97,78

Elde edilen hata değerleri (en küçük, en büyük, ortalama), standart sapma, hesaplama süresi ve algoritmaların doğrulama veri seti üzerinde elde edilen 30 koşmaya ait ortalama maliyet fonksiyonu değeri (Mean Fval) Tablo 3.17 ile gösterildiği gibidir. Tablo 3.17'e göre, önerilen optimizasyon algoritması tabanlı yöntemlerin tüm çalışmalarının en iyi minimum hata değeri PSO, maksimum hata değeri ise GA algoritması ile elde edilmiştir. Tablodan, ABC algoritmasının ortalama hata kriterine göre (0,0191 değeri ile) en başarılı yöntem olduğu söylenebilir. 0,0074 standart sapma değerine sahip ABC, bu veri seti için en dirençli algoritmadır. Hesaplama zamanı başka bir performans kriteridir ve bu kriteri göre GA algoritması en hızlı çalışan algoritmadır. Son olarak, ABC ve PSO algoritmaları en küçük ortalama fval (0 değeri ile) değerine sahiptir.

Tablo 3.17. TMGRD veri seti için algoritmalara ait hata, standart sapma, hesaplama süresi (sn) ve ortalama fval değerleri.

	Algoritmalar		
	ABC	GA	PSO
Minimum Hata	0,0049	0,0114	0,0016
Maksimum Hata	0,0343	0,0572	0,0490
Ortalama Hata	0,0191	0,0287	0,0222
Standart Sapma	0,0074	0,0128	0,0122
Hesaplama Süresi	21293	19026	25723
Ortalama Maliyet	0	0,0123	0

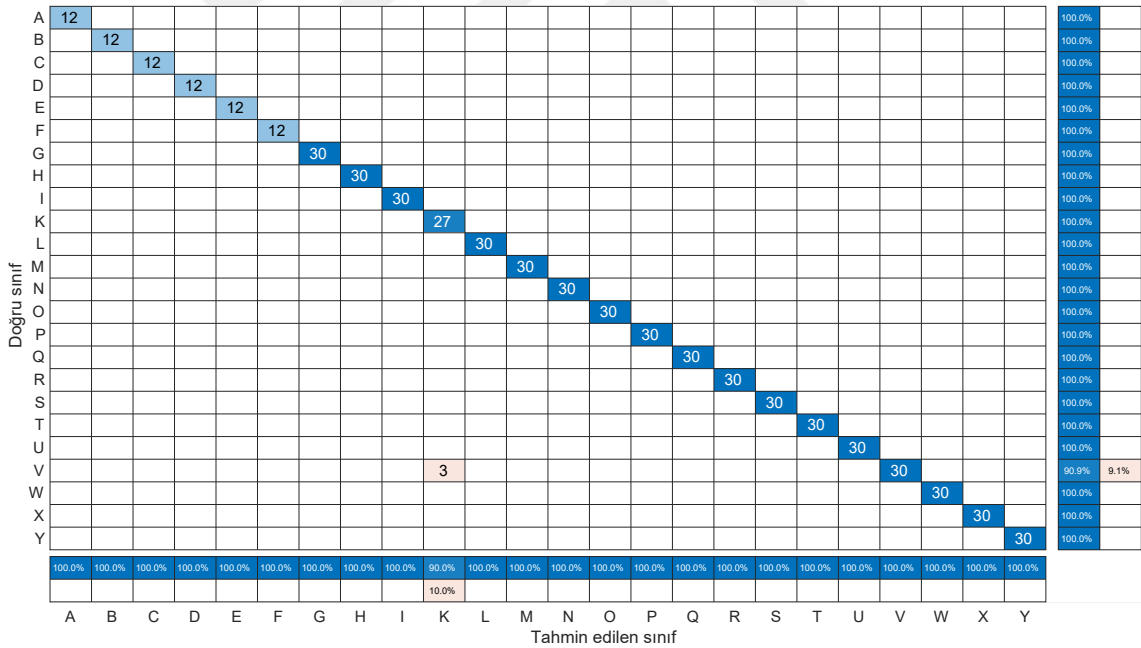
TMGRD veri seti için algoritma karmaşıklık değerleri Tablo 3.18 ile sunulmuştur. Karmaşıklık sonuçlarına göre, GA optimizasyon algoritması en küçük algoritma

karmaşıklığı değerine sahiptir.

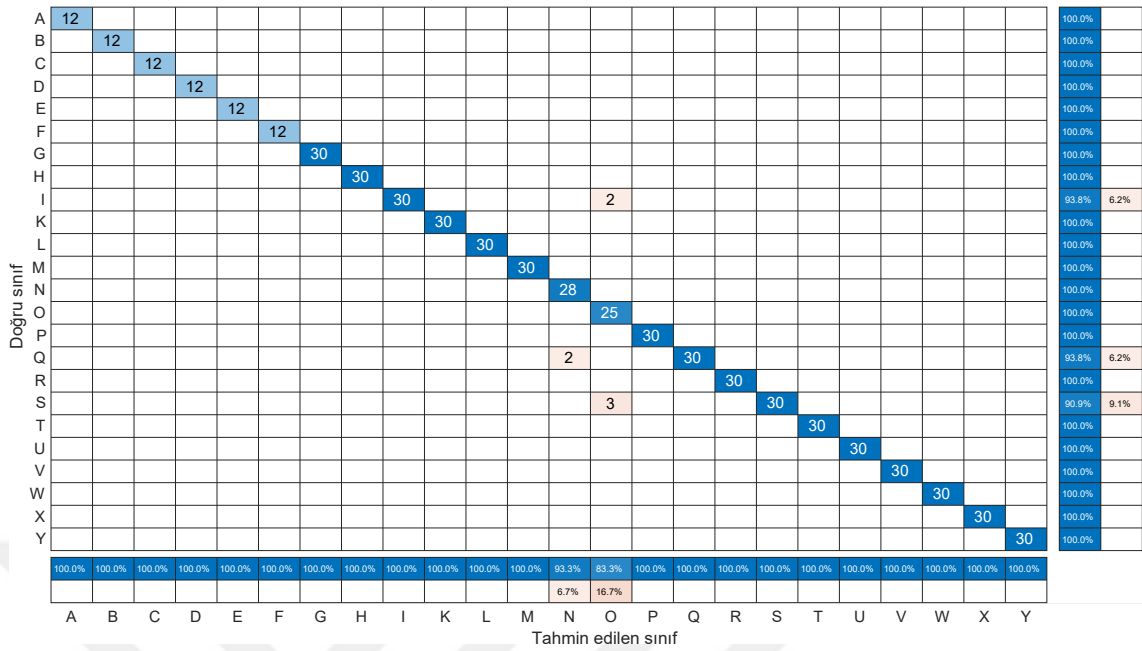
Tablo 3.18. $T_0 = 0,076805$ eşitliği elde edildiği durumda TMGRD veri seti için algoritma karmaşıklık değerleri.

Algoritma	T_1	Algoritma Karmaşıklığı
ABC	21293	277231
GA	19026	247722
PSO	25723	334919

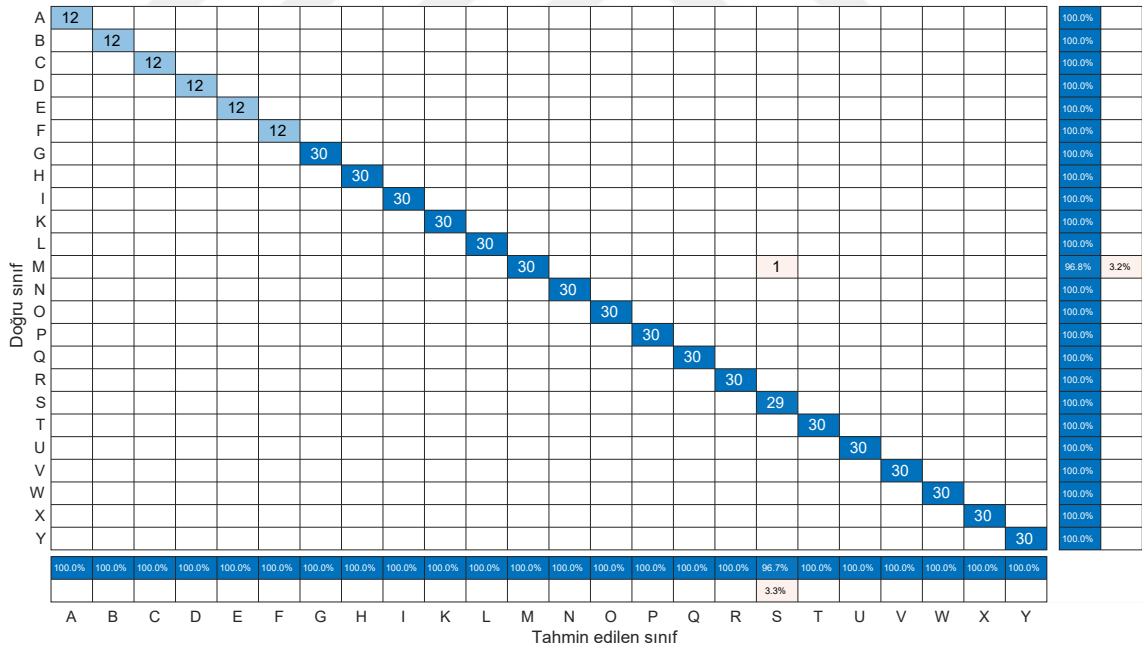
Her bir harf için doğruluk oranları, karmaşıklık matrisleriyle de ifade edilebilmektedir. Hiperparametreleri TMGRD veri seti için optimizasyon algoritmalarıyla ayarlanan AlexNet tabanlı CNN'lerin en iyi performanslara göre karmaşıklık matrisleri sırasıyla Şekil 3.10, 3.11 ve 3.12 ile sunulurken en kötü performanslara ait karmaşıklık matrisleri sırasıyla Şekil 3.13, 3.14 ve 3.15 ile verilmiştir.



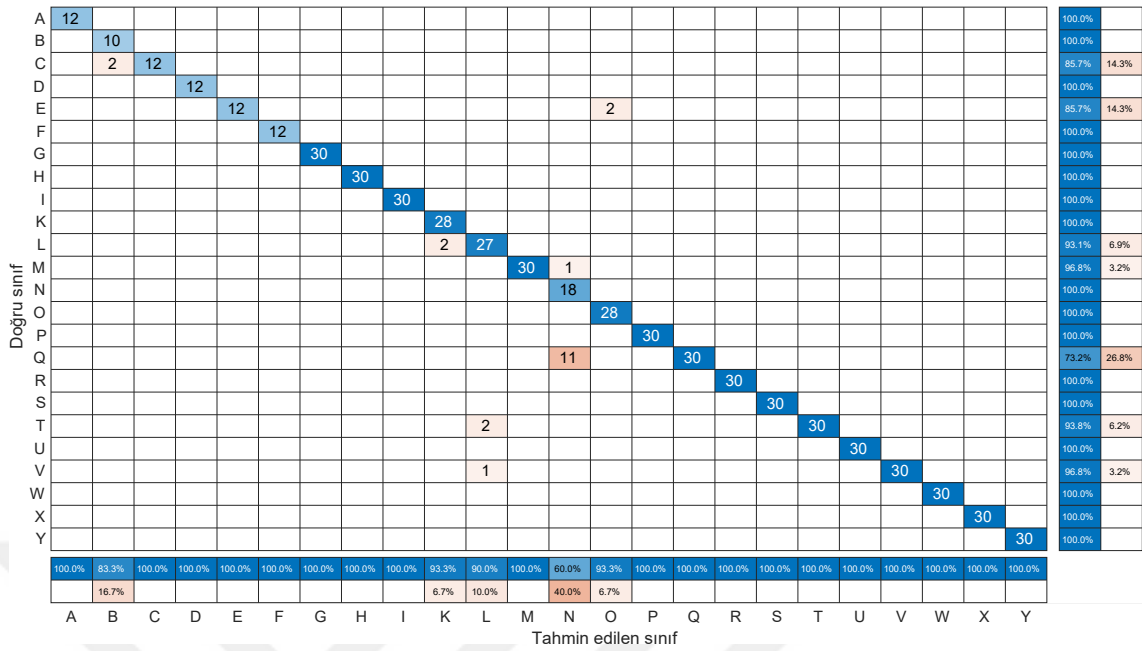
Şekil 3.10. AlexNet + ABC destekli CNN metodunun TMGRD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.



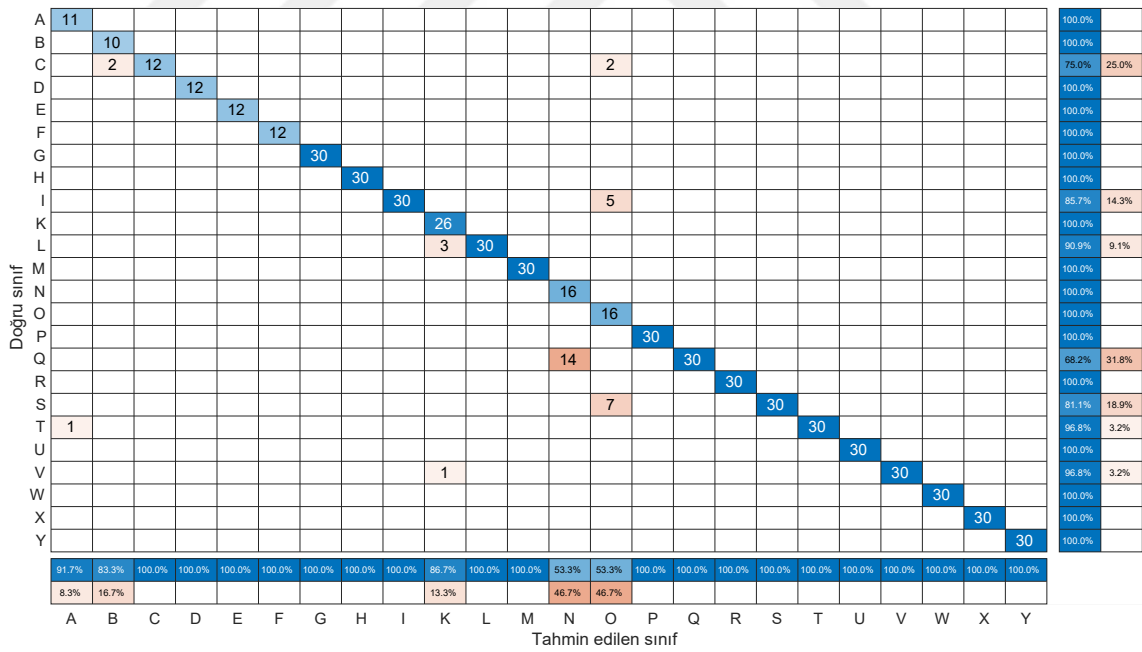
Şekil 3.11. AlexNet + GA destekli CNN metodunun TMGRD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.



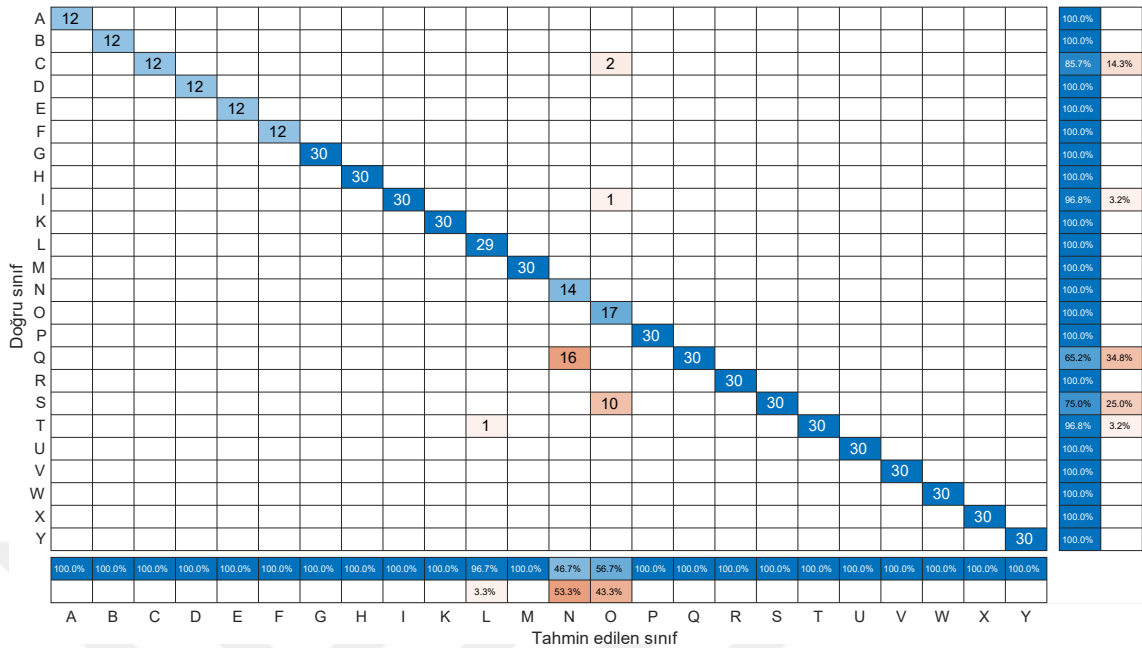
Şekil 3.12. AlexNet + PSO destekli CNN metodunun TMGRD veri seti için en başarılı sonucuna göre karmaşıklık matrisi.



Şekil 3.13. AlexNet + ABC destekli CNN metodunun TMGRD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.



Şekil 3.14. AlexNet + GA destekli CNN metodunun TMGRD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.



Şekil 3.15. AlexNet + PSO destekli CNN metodunun TMGRD veri seti için en başarısız sonucuna göre karmaşıklık matrisi.

İstatistiksel analiz yapmak amacıyla, TMGRD veri setindeki tüm hata değerleri için Kolmogorov-Smirnov normallik testi [157] yapılmıştır. Tablo 3.19 incelendiğinde, ABC algoritmasının anlamlılık değeri 0,05'ten küçük olduğu için hata değerlerinin dağılımının normal olmadığı görülmektedir. Bu nedenle, parametrik olmayan testler arasında Kruskal-Wallis [158] ve Mann-Whitney U [159] testleri, yöntemlerin performansları arasında anlamlı bir fark olup olmadığını belirlemek için kullanılmıştır. Kruskal-Wallis testi, algoritmalar arasında hata dağılımları açısından anlamlı fark olup olmadığını kontrol etmek için kullanılır. Bu testin sonuçları Tablo 3.20 ile verilmiştir. Tablo incelendiğinde anlamlılık değeri (p) 0,009223'e eşit ve 0,05'ten küçük olduğundan algoritmalar arasında anlamlı bir fark olduğu görülmüştür. Mann-Whitney U testi yapılarak, ikili algoritmalar arasında anlamlı bir fark olup olmadığı tespit edilebilir. Tablo 3.21 incelendiğinde Mann-Whitney U test sonuçlarına göre, TMGRD veri seti için kullanılan algoritmalar içerisinde ABC ve GA algoritmaları arasında anlamlı bir fark olduğu, diğer eşleşmeler için herhangi bir anlamlı fark olmadığı görülmektedir. Tablo incelendiğinde, 0,320168 anlamlılık değeri ile en çok ABC ve PSO temelli sonuçlar arasında benzerlikler bulunmaktadır.

Tablo 3.19. TMGRD veri seti için Kolmogorov-Smirnov normalite test sonuçları.

Algoritma	İstatistik	Anlamlılık Değeri
ABC	0,190	0,007
GA	0,131	0,200
PSO	0,092	0,200

Tablo 3.20. TMGRD veri seti için Kruskal-Wallis parametrik olmayan test sonuçları.

Hata	
Chi-square	9,372
Anlamlılık değeri (p)	0,009223

Tablo 3.21. TMGRD veri seti için Mann-Whitney parametrik olmayan test sonuçları.

Algoritmalar	Anlamlılık Değeri (p)
ABC & GA	0,002109
ABC & PSO	0,320168
GA & PSO	0,057912

3.1.5. Değerlendirme

Duyuma yeteneğini kaybetmiş bireylerle iletişim kurmak ve onları topluma dahil etmek önemli bir sosyal sorumluluktur. Bu çalışmada, duyma engelli insanların rakamları ve harfleri ifade etmeleri için kullandıkları el hareketleri CNN kullanılarak sınıflandırılmıştır. Bu sınıflamanın yüksek başarısı, bu insanlarla iletişimin etkinliğini ve doğruluğunu artıracaktır. Bu amaçla, sezgisel optimizasyon algoritması destekli ve AlexNet tabanlı bir derin öğrenme yapısı işaret dili sınıflandırma için tasarlanmıştır. Önerilen yöntem iki temel adımdan oluşmaktadır. Bu adımlar AlexNet destekli transfer öğrenme ve hiperparametre ayarlamadır. AlexNet kullanılarak, mimarinin son üç katmanın düzenlenmesiyle transfer öğrenme destekli yeni bir mimari oluşturulmuştur. CNN'in eğitim ayarlarında kullanılan hiperparametrelerin seçimi ise ABC, GA ve PSO kullanılarak yapılmıştır. Önerine yöntemi test etmek için SLDD ve TMGRD veri setleri kullanılmıştır. Sonuç olarak, SLDD veri seti için en iyi sonucu elde etmiş olan Beser ve

ekibinin çalışması [10] geride bırakılmıştır. Bir diğer veri seti olan TMGRD veri seti için literatürde bilinen en iyi çalışmalar [11,31] yine aynı yöntemle geride bırakılmıştır.

İleriye dönük çalışmalar için, önerilen yöntem farklı veri setleri için kullanılabilir. Bu şekilde, yöntemin gübüzlüğü daha fazla veri seti kullanılarak değerlendirilebilir. Bu çalışmada AlexNet transfer öğrenimi için kullanılmıştır. Diğer ön eğitimli CNN modelleri gelecekteki çalışmalar için de kullanılabilir. Ön eğitimli modeller, özellik çıkarıcı olarak da kullanılabilirdiğinden SVM veya diğer sınıflandırma yöntemleri, çıkarılan özellikleri sınıflandırmak için tercih edilebilir. ABC, GA ve PSO optimizasyon algoritmaları, CNN'nin hiperparametrelerini ayarlamak için kullanılmıştır. Önerilen yöntemin parametre ayarlaması için diğer optimizasyon algoritmaları gelecek çalışmalar için kullanılabilir. Son olarak, hibrit algoritmalar geliştirmek ve performanslarını literatürdeki mevcut eserlerle karşılaştırmak mümkündür.

3.2. ERUSLR: Yeni Bir Türkçe İşaret Dili Veri Seti ve Evrişimli Sinir Ağı ile Tanınması

3.2.1. Giriş

İnsanların birbirini anlayabilmesi, iletişim kurması için ortak bir dil gerekmektedir. Dil, vücut dili veya hissel etkileşimle gerçekleştirilmesi sağlanan iletişim dili işaret dilidir. Duyuma ve konuşma engelli bireyler arasında iletişim, işaret dili ile sağlanmaktadır. Engelli bireylerin kullandığı işaret dilini engelli olmayanların da bilmesi insanlar arasında etkileşimin ve iletişimin sağlanması için önem arz etmektedir. Bahsi geçen engelli bireylerin daha iyi anlaşılması ve topluma kazandırılması için işaret dili tanıma üzerine çalışmalar gerçekleştirilmektedir. İşaret dili tanıma uygulamalarının temelinde bu problem için hazırlanmış veri setleri bulunmaktadır. İşaret dili tanıma, harf/rakam, kelime ve cümle düzeyinde gerçekleştirilmektedir. Literatürde sıklıkla kullanılan kelime düzeyinde hazırlanmış veri setleri ve veri setlerine ait bazı özellikler Tablo 3.22 ile sunulmuştur.

Tablo 3.22. Literatürde sıklıkla kullanılan kelime düzeyinde işaret dili tanıma veri setleri.

Veri Seti	Ülke	Sınıf	Katılımcı	Örnek
DGS Kinect 40 [160]	Almanya	40	15	3000
GSL 20 [160]	Yunanistan	20	6	840
ASLLVD [33]	ABD	3300	6	9800
PSL Kinect 30 [37]	Polonya	30	1	300
LSA64 [40]	Arjantin	64	10	3200
DEVISIGN-D [161]	Çin	500	8	6000

Bu çalışmanın ilk aşamasında, Ek-A ile etik izni alınmış olarak hastanelerin acil servislerinde ihtiyaç duyulan 25 kelime kullanılarak yeni bir işaret dili tanıma veri seti oluşturulmuştur. Erciyes Üniversitesi işaret dili tanıma (Erciyes University sign language recognition, ERUSLR) adındaki bu yeni veri seti, 49 engelli bireyden alınan farklı sayıdaki pozlarla toplamda 13222 örnekten oluşmaktadır. Çalışmanın sonraki aşamasında veri setinde yer alan örnekler ön işlemden geçirilmiştir. Video formatındaki örnekler çerçevelere ayrılmış, daha sonra çerçeveler sıralı bir şekilde birleştirilerek tekil imge

haline getirilmiştir. Ön işlemenin son aşamasında, hareket tanımda renk faktörünün sonuca etkili bir değişim sağlamaması ve işlem sürecini artırmasından dolayı tekil imgeler gri tonlama haline çevrilmiştir. ERUSLR veri seti, eğitim ve test veri setleri olarak parçalandıktan sonra sınıflandırmada kullanılacak modelin parametrelerini en iyilemek üzere eğitim veri setinden geliştirme veri seti [162] üretilmiştir. Literatürde sıklıkla kullanılan CNN metodu [4] bu çalışmada sınıflandırıcı olarak kullanılmıştır. Yeni bir CNN modeli geliştirmekten daha kolay bir yöntem ön eğitilmiş model kullanarak transfer öğrenme gerçekleştirmektir. Dolayısıyla başarılı bir CNN ön eğitilmiş model olan GoogLeNet bu çalışmada transfer öğrenme gerçekleştirmek için tercih edilmiştir. CNN mimarisinin başarısını etkileyen bir başka faktör eğitim parametrelerinin en iyilenmesidir. Bu amaçla türetilen geliştirme veri seti, GoogLeNet destekli CNN ve yaygın kullanılan global arama yöntemleri (grid arama (GS) ve rastgele arama (RS)) kullanılarak parametre en iyileme işlemi gerçekleştirilmiştir. Orijinal eğitim veri seti ile global arama metodu çıktısı olarak elde edilen en iyi parametre seti beslemeli CNN modeli eğitilip elde edilen eğitilmiş model test verisi üzerinden test edilerek sistemin performansı ölçülmüştür. Deneysel sonuçlar incelendiğinde GS destekli GoogLeNet tabanlı CNN modeli ile %93,63 doğruluk oranı elde edilirken RS destekli GoogLeNet tabanlı CNN ile %89,81 doğruluk oranı elde edilmiştir. Bir diğer deneysel çalışma sonucunda varsayılan parametrelerin kullanıldığı GoogLeNet destekli CNN ile %88,62 başarı oranı elde edilmiştir.

Bu çalışmanın ana katkıları şu şekilde sıralanabilir:

- Hastanelerin acil servislerinde kullanılmak üzere ERUSLR adında genel kullanıma açık yeni bir işaret dili tanıma veri seti geliştirilmiştir.
- Global arama yöntemi destekli ve GoogLeNet tabanlı CNN kullanan bir sistem geliştirilmiştir.
- GS destekli GoogLeNet tabanlı CNN ile %93,61 doğruluk oranı ile en başarılı sonuç elde edilmiştir.

Bu çalışmanın kalan bölümleri şu şekilde düzenlenmiştir. Bölüm 3.2.2 ile problem çözümünde kullanılacak olan veri seti açıklanmış ve Bölüm 3.2.3 ile önerilen yöntem

açıklanmış olup Bölüm 3.2.4 ile deneysel çalışmalar detaylı bir şekilde sunulmuştur. Bölüm 3.2.5 ile bu çalışmanın değerlendirilmesi yapılacak ve gelecek çalışmalardan bahsedilmiştir.

3.2.2. Kullanılan Veri Seti

Bu çalışmada önerilen yöntem bu tez çalışmasında geliştirilen kelime düzeyinde Erciyes University sign language recognition (ERUSLR) veri seti ile analiz edilecektir. İlgili veri setine ait detaylı bilgiler alt bölümde sunulmuştur.

3.2.2.1. ERUSLR Veri seti

Hastanelerin acil servislerinde işaret dilinin anlaşılabilmesi kritik sonuçlar ortaya çıkarabilir. Bu sebepten dolayı hastanelerin acil servislerinde sıklıkla kullanılan 25 kelime belirlenmiştir. İşaret dili, herkesin kullanabileceği bir dil olmadığından gönüllü seçimi sadece engelli bireylerden yapılmıştır. 25 kelime ile oluşan 25 sınıf için 49 gönüllüyle oluşturulan bu veri seti, toplamda 13222 örnek içermektedir. ERUSLR veri seti oluşturulurken alınan örnekler 3 farklı açıdan (karşıdan (front, F), sağdan (right, R), ve soldan (left, L)) kaydedilmiştir (Şekil 3.16 ile gösterildiği gibi). Her kullanıcıdan farklı sayıda örnek alındığından dolayı bu veri seti dengesiz (unbalanced) bir yapıdadır. ERUSLR veri setindeki 49 gönüllünün yer aldığı örnek çerçeveler Şekil 3.17 ile sunulmuştur.

Veri setini oluşturan sınıflar ve bu sınıflara ait bilgiler Tablo 3.23 ile sunulmuştur. Örnek sayısında dengeli dağılıma sahip olmayan bu veri seti için farklı tipte eğitim-test verisi ayırma teknikleriyle performans analizleri gerçekleştirilebilir.



Şekil 3.16. Üç farklı açıdan alınan örnekler.



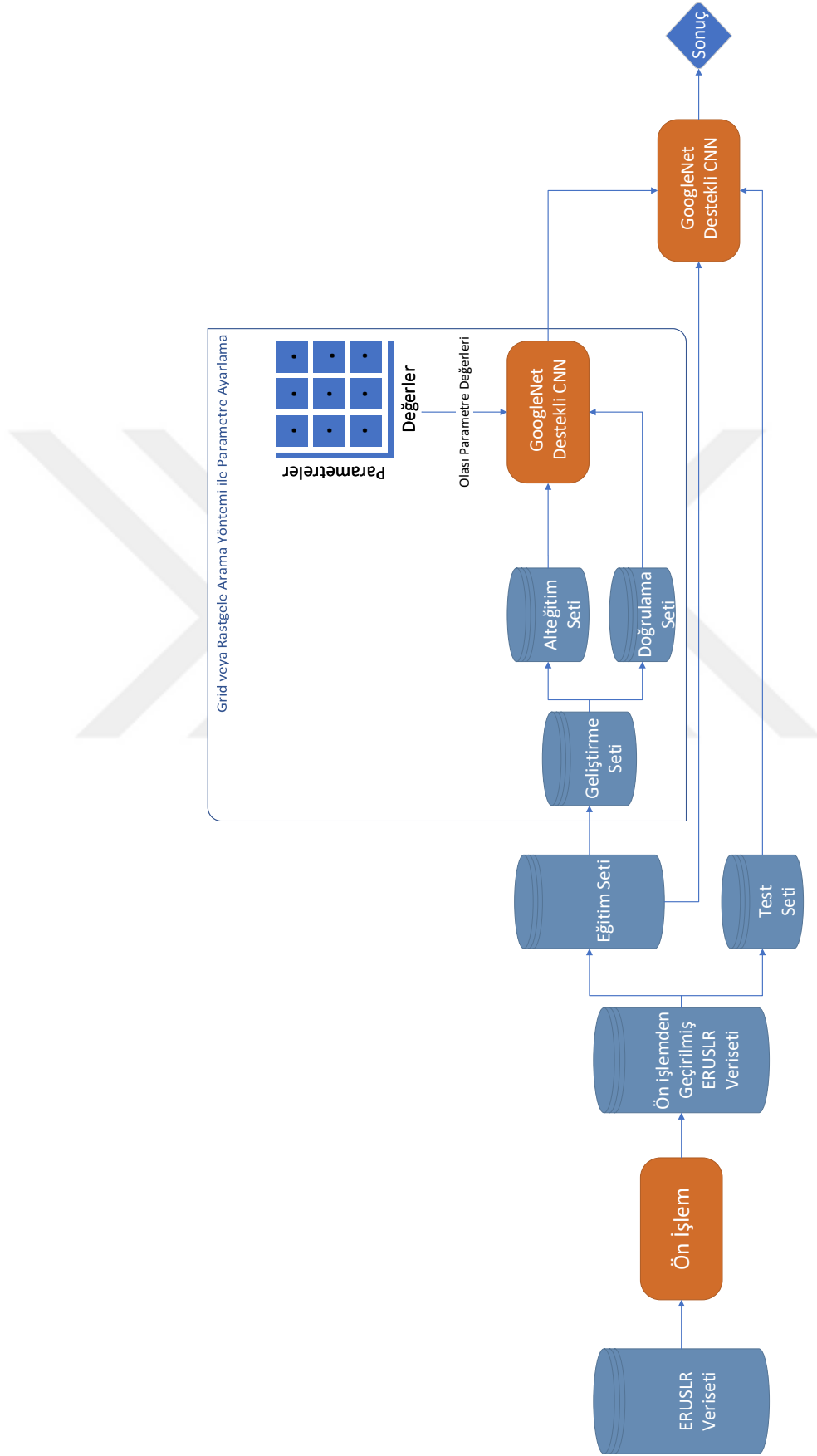
Şekil 3.17. ERUSLR veri setinde yer alan gönüllüler.

Tablo 3.23. ERUSLR veri setinde yer alan sınıflar ve toplam örnek sayıları.

Id	Sınıf	Sınıf(İngilizce)	Örnek Sayısı
1	Agiz	Mouth	621
2	Agri	Pain	393
3	Alerji	Allergy	564
4	Astim	Asthma	555
5	Ates	Fever	416
6	Bas_Agrisi	Headache	607
7	Bas_Dönmesi	Dizziness	373
8	Beyin	Brain	549
9	Bobrek	Kidney	616
10	Bogaz	Throat	571
11	Bulanti	Nausea	617
12	Grip	Flu	573
13	Halsizlik	Weakness	520
14	Ilac	Medicine	549
15	Kalp	Heart	445
16	Kan	Blood	526
17	Karin	Abdomen	554
18	Kemik	Bone	542
19	Kusmak	Spew	623
20	Nabiz	Pulse	522
21	Nefes	Breath	361
22	Seker_Hastaligi	Diabetes	634
23	Tansiyon	Blood pressure	426
24	Terlemek	Exude	530
25	Yara	Wound	499

3.2.3. Metodoloji

ERUSLR veri setinin global arama yöntemleri (GS ve RS) ile transfer öğrenme tabanlı CNN kullanılarak sınıflandırılması bu çalışmanın temel amacıdır. Uygulama için önerilen birleştirilmiş yöntem Şekil 3.18 ile sunulmuştur.



Şekil 3.18. ERUSLR veri seti için önerilen birleşik yöntem.

Önerilen metotta ilk olarak ERUSLR veri seti ön işlemden geçirilmektedir. Video formatında olan örnekler çerçevelere ayrılıp sıralı bir şekilde sıralanarak tekil imge haline gelmektedir. RGB formatındaki tekil imgenin gri formata dönüştürülmesi ön işlem sürecinin son aşamasıdır. Ön işlem aşamasına örnek bir gösterim Şekil 3.19 ile sunulmuştur.



Şekil 3.19. Bir örnek üzerinde ön işleme çalışması.

Bir sonraki aşamada, ön işlemden geçirilmiş ERUSLR veri seti %70'i eğitim %30'u test verisi olacak şekilde rastgele ayrılıp eğitim ve test veri setleri oluşturulur. Eğitim setinin %50'si kullanılarak geliştirme seti elde edilir. Parametre ayarlama, özellik çıkarma gibi işlemler için üretilen geliştirme seti [162], bu çalışmada CNN eğitim parametrelerinden ilk öğrenme oranı, l_2 düzenleme, ve momentum parametrelerinin ayarlanması için kullanılmıştır. Daha sonra, geliştirme seti %70'i eğitim %30'u doğrulama verisi olacak şekilde ayrılır. Parametre optimizasyonunda sıklıkla kullanılan GS ve RS yöntemleri bu çalışmada parametre ayarlamakla görevlendirilmiştir. Bu aşamada GS veya RS yöntemi tarafından üretilen parametre seti ile eğitim verisi GoogLeNet destekli CNN'e girdi olarak verilir. Eğitim işleminin ardından oluşan eğitilmiş model Eşitlik 3.1 kullanılarak doğrulama verisi üzerinde test edilir. Bu eşitlik ile hesaplanan doğrulama hata değeri kayıp fonksiyonu değeri olarak saklanır. GS veya RS tarafından üretilen tüm parametre setleri için bu işlem tekrarlanır. Parametre ayarlama süreci sonunda elde edilen en iyi parametre seti ve eğitim veri seti GoogLeNet destekli CNN'e girdi olarak verilir ve son eğitilmiş model elde edilir. Elde edilen bu son model Eşitlik 3.2 kullanılarak test edilir. Elde edilen çıktı doğruluk değeridir ve sistemin başarısını göstermektedir.

Önerilen yöntemde kullanılan global arama yöntemleri, evrişimli sinir ağı ve transfer

öğrenme metodu sırasıyla Bölüm 2.2.6, 2.1.1, ve 2.1.1.1 ile detaylı olarak sunulmuştur.

3.2.4. Deneysel Çalışmalar

Deneysel çalışmalar; deneysel kurulum ve deneysel sonuçlar alt başlıklarında incelenecektir.

3.2.4.1. Deneysel Kurulum

Deneysel kurulum; donanımsal kurulum ve parametre kurulumu olmak üzere iki alt başlıkta incelenmiştir.

3.2.4.1.1. Donanımsal Kurulum

ERUSLR veri seti görsel veriden oluşmaktadır. Ayrıca veri setindeki örnek sayısı oldukça yüksektir. Bu sebeplerden dolayı bütün deneysel çalışmalarda 11 GB Ram destekli NVIDIA GTX 1080 Ti Turbo donanımı kullanılmış ve önerilen yöntem koşulmuştur.

3.2.4.1.2. Parametre Kurulumu

Önerilen yöntemde, parametre ayarlaması için grid ve rastgele global arama yöntemleri kullanılmıştır. Grid aramada kullanılacak ilk öğrenme oranı, l_2 düzenleme, ve momentum parametre değerleri Tablo 3.24 ile gösterilmiştir. Diğer taraftan, rastgele arama yönteminde seçilecek ilk öğrenme oranı, l_2 düzenleme, ve momentum değerlerinin alt sınırları 0,001, 0,00001 ve 0,75 iken üst sınırları 0,01, 0,001 ve 0,95'tir.

Tablo 3.24. Grid aramada kullanılan parametre değerleri.

Parametre	Değerler				
ilk öğrenme oranı	0,001	0,003	0,005	0,007	0,01
l_2 düzenleme	0,00001	0,0005	0,0001	0,005	0,001
momentum	0,75	0,80	0,85	0,90	0,95

CNN modelinin diğer eğitim parametrelerinden olan minibatch boyutu değeri 32 ve maksimum epok değeri 100 olarak seçilmiştir. CNN'in eğitim aşamasında momentumlu

stokastik gradyan inişi yöntemi kullanılmıştır.

3.2.4.2. Deneysel Sonuçlar

Bu tez çalışmasında geliştirilen yöntem ERUSLR veri seti ile test edilmiştir. Önerilen yöntemde parametre optimizasyonunda kullanılmak üzere global arama yöntemlerinden grid arama ve rastgele arama yöntemleri tercih edilmiştir.

Global arama yöntemi olarak grid arama kullanıldığı durumda oluşabilecek tüm parametre setleri ve bu parametre setleri kullanıldığında elde edilen doğrulama hataları Tablo 3.25 ile sunulmuştur. 0,3201 doğrulama hatasını veren parametre seti grid arama sonucunda hafızada tutulan en iyi parametre değerleridir.

Global arama yöntemi olarak rastgele arama kullanıldığı durumda oluşan parametre setleri ve bu parametre setlerine ait doğrulama hataları Tablo 3.26 ile sunulmuştur. Tablo 3.26 incelendiğinde 0,3208 doğrulama hatasını veren parametre seti rastgele arama sonucunda hafızada tutulan en iyi parametre değerleridir.

GS yöntemi ile elde edilen optimum parametre seti ile eğitim verisi kullanılarak GoogLeNet destekli CNN modeli eğitilmiştir. Çıktı olarak üretilen eğitilmiş model ile test verisi üzerinde %93,61 doğruluk oranı elde edilmiştir. İlgili eğitilmiş modelin test verisi üzerindeki karmaşıklık matrisi Şekil 3.20 ile sunulmuştur.

RS yöntemi ile elde edilen optimum parametre seti ile eğitim verisi kullanılarak GoogLeNet destekli CNN modeli eğitilmiştir. Çıktı olarak üretilen eğitilmiş model ile test verisi üzerinde %89,81 doğruluk oranı elde edilmiştir. İlgili eğitilmiş modelin test verisi üzerindeki karmaşıklık matrisi Şekil 3.21 ile sunulmuştur.

Herhangi bir global arama yöntemi kullanılmadan sadece varsayılan eğitim parametre değerlerini kullanarak GoogLeNet destekli CNN modeli eğitilmiştir. Çıktı olarak üretilen eğitilmiş model ile test verisi üzerinde %88,62 doğruluk oranı elde edilmiştir. İlgili eğitilmiş modelin test verisi üzerindeki karmaşıklık matrisi Şekil 3.22 ile sunulmuştur.

Parametre ayarlayıcı desteksiz GoogLeNet tabanlı CNN modeli ile GS ve RS destekli GoogLeNet tabanlı CNN modelleri için kullanılan eğitim parametreleri değeri ve bu

Tablo 3.25. Grid arama yönteminde belirlenen parametre değerleri ve hesaplanan doğrulama hataları.

Koşma	Parametreler			Doğrulama Hatası	Koşma	Parametreler			Doğrulama Hatası
	İlk öğrenme oranı	L2 düzenleme	Momentum			İlk öğrenme oranı	L2 düzenleme	Momentum	
1	0,001	0,00001	0,75	0,3691	64	0,005	0,0001	0,90	0,3526
2	0,001	0,00001	0,80	0,4232	65	0,005	0,0001	0,95	0,4585
3	0,001	0,00001	0,85	0,3857	66	0,005	0,005	0,75	0,4456
4	0,001	0,00001	0,90	0,4009	67	0,005	0,005	0,80	0,3547
5	0,001	0,00001	0,95	0,3035	68	0,005	0,005	0,85	0,3792
6	0,001	0,0005	0,75	0,3958	69	0,005	0,005	0,90	0,5213
7	0,001	0,0005	0,80	0,4110	70	0,005	0,005	0,95	0,6301
8	0,001	0,0005	0,85	0,3850	71	0,005	0,001	0,75	0,3800
9	0,001	0,0005	0,90	0,3619	72	0,005	0,001	0,80	0,3259
10	0,001	0,0005	0,95	0,3331	73	0,005	0,001	0,85	0,3439
11	0,001	0,0001	0,75	0,3973	74	0,005	0,001	0,90	0,4348
12	0,001	0,0001	0,80	0,3915	75	0,005	0,001	0,95	0,4535
13	0,001	0,0001	0,85	0,3821	76	0,007	0,00001	0,75	0,3785
14	0,001	0,0001	0,90	0,3720	77	0,007	0,00001	0,80	0,3526
15	0,001	0,0001	0,95	0,3273	78	0,007	0,00001	0,85	0,3410
16	0,001	0,005	0,75	0,4146	79	0,007	0,00001	0,90	0,4218
17	0,001	0,005	0,80	0,3836	80	0,007	0,00001	0,95	0,9517
18	0,001	0,005	0,85	0,3764	81	0,007	0,0005	0,75	0,3915
19	0,001	0,005	0,90	0,3670	82	0,007	0,0005	0,80	0,4117
20	0,001	0,005	0,95	0,3944	83	0,007	0,0005	0,85	0,3987
21	0,001	0,001	0,75	0,3893	84	0,007	0,0005	0,90	0,3605
22	0,001	0,001	0,80	0,3468	85	0,007	0,0005	0,95	0,9517
23	0,001	0,001	0,85	0,3641	86	0,007	0,0001	0,75	0,3497
24	0,001	0,001	0,90	0,3706	87	0,007	0,0001	0,80	0,3439
25	0,001	0,001	0,95	0,3360	88	0,007	0,0001	0,85	0,3771
26	0,003	0,00001	0,75	0,3619	89	0,007	0,0001	0,90	0,4247
27	0,003	0,00001	0,80	0,3720	90	0,007	0,0001	0,95	0,5083
28	0,003	0,00001	0,85	0,3338	91	0,007	0,005	0,75	0,4196
29	0,003	0,00001	0,90	0,3684	92	0,007	0,005	0,80	0,4420
30	0,003	0,00001	0,95	0,3742	93	0,007	0,005	0,85	0,4470
31	0,003	0,0005	0,75	0,3627	94	0,007	0,005	0,90	0,5811
32	0,003	0,0005	0,80	0,3670	95	0,007	0,005	0,95	0,7931
33	0,003	0,0005	0,85	0,4182	96	0,007	0,001	0,75	0,3922
34	0,003	0,0005	0,90	0,3201	97	0,007	0,001	0,80	0,3828
35	0,003	0,0005	0,95	0,3396	98	0,007	0,001	0,85	0,3634
36	0,003	0,0001	0,75	0,3879	99	0,007	0,001	0,90	0,4146
37	0,003	0,0001	0,80	0,3699	100	0,007	0,001	0,95	0,9517
38	0,003	0,0001	0,85	0,3627	101	0,01	0,00001	0,75	0,3886
39	0,003	0,0001	0,90	0,3713	102	0,01	0,00001	0,80	0,3598
40	0,003	0,0001	0,95	0,3901	103	0,01	0,00001	0,85	0,3879
41	0,003	0,005	0,75	0,3699	104	0,01	0,00001	0,90	0,4470
42	0,003	0,005	0,80	0,3482	105	0,01	0,00001	0,95	0,9531
43	0,003	0,005	0,85	0,3576	106	0,01	0,0005	0,75	0,3518
44	0,003	0,005	0,90	0,3518	107	0,01	0,0005	0,80	0,3446
45	0,003	0,005	0,95	0,4845	108	0,01	0,0005	0,85	0,4167
46	0,003	0,001	0,75	0,3684	109	0,01	0,0005	0,90	0,4996
47	0,003	0,001	0,80	0,3605	110	0,01	0,0005	0,95	0,9517
48	0,003	0,001	0,85	0,3583	111	0,01	0,0001	0,75	0,3655
49	0,003	0,001	0,90	0,3360	112	0,01	0,0001	0,80	0,3778
50	0,003	0,001	0,95	0,3800	113	0,01	0,0001	0,85	0,3843
51	0,005	0,00001	0,75	0,3785	114	0,01	0,0001	0,90	0,5501
52	0,005	0,00001	0,80	0,3901	115	0,01	0,0001	0,95	0,9531
53	0,005	0,00001	0,85	0,3720	116	0,01	0,005	0,75	0,4859
54	0,005	0,00001	0,90	0,3951	117	0,01	0,005	0,80	0,4492
55	0,005	0,00001	0,95	0,4650	118	0,01	0,005	0,85	0,5032
56	0,005	0,0005	0,75	0,3857	119	0,01	0,005	0,90	0,6128
57	0,005	0,0005	0,80	0,3518	120	0,01	0,005	0,95	1,0000
58	0,005	0,0005	0,85	0,3821	121	0,01	0,001	0,75	0,3396
59	0,005	0,0005	0,90	0,4124	122	0,01	0,001	0,80	0,4037
60	0,005	0,0005	0,95	0,4463	123	0,01	0,001	0,85	0,3915
61	0,005	0,0001	0,75	0,3634	124	0,01	0,001	0,90	0,5912
62	0,005	0,0001	0,80	0,3381	125	0,01	0,001	0,95	0,9517
63	0,005	0,0001	0,85	0,3691					

Tablo 3.26. Rastgele arama yönteminde belirlenen parametre değerleri ve hesaplanan doğrulama hataları.

Koşma	Parametreler			Doğrulama Hatası	Koşma	Parametreler			Doğrulama Hatası
	İlk öğrenme oranı	L2 düzenleme	Momentum			İlk öğrenme oranı	L2 düzenleme	Momentum	
1	0,00455	0,000471	0,76	0,3764	64	0,006387	0,000568	0,95	0,5061
2	0,005498	0,000826	0,85	0,3612	65	0,004719	0,000468	0,83	0,3475
3	0,001799	0,000697	0,93	0,3345	66	0,00461	0,000433	0,91	0,3670
4	0,009986	0,000326	0,86	0,3720	67	0,00939	0,000465	0,93	0,9272
5	0,009237	0,000043	0,76	0,3590	68	0,009228	0,000115	0,88	0,4189
6	0,002177	0,000189	0,77	0,3706	69	0,00348	0,000089	0,76	0,3648
7	0,00397	0,000564	0,89	0,3973	70	0,003319	0,000414	0,75	0,3684
8	0,006906	0,000394	0,86	0,3893	71	0,005605	0,000573	0,80	0,3850
9	0,007986	0,000120	0,77	0,3720	72	0,006457	0,000816	0,77	0,3922
10	0,002154	0,000790	0,89	0,3778	73	0,006108	0,000950	0,84	0,3857
11	0,009903	0,000676	0,93	0,5263	74	0,007021	0,000041	0,84	0,3490
12	0,002919	0,000888	0,93	0,3785	75	0,00574	0,000710	0,87	0,3814
13	0,003499	0,000860	0,85	0,3345	76	0,008385	0,000167	0,79	0,3821
14	0,005655	0,000909	0,80	0,3482	77	0,005115	0,000446	0,90	0,3850
15	0,006591	0,000766	0,89	0,3518	78	0,007756	0,000856	0,88	0,3836
16	0,007028	0,000539	0,77	0,3886	79	0,009743	0,000275	0,77	0,3735
17	0,004905	0,000664	0,86	0,3994	80	0,008779	0,000891	0,85	0,3915
18	0,004632	0,000562	0,95	0,4376	81	0,009779	0,000849	0,87	0,4052
19	0,004142	0,000682	0,93	0,4297	82	0,006394	0,000618	0,91	0,3922
20	0,004871	0,000487	0,84	0,3901	83	0,003657	0,000712	0,93	0,3619
21	0,004493	0,000261	0,76	0,3432	84	0,004038	0,000462	0,85	0,4268
22	0,006722	0,000041	0,82	0,3619	85	0,001853	0,000767	0,78	0,3727
23	0,007065	0,000107	0,81	0,3526	86	0,00837	0,000942	0,95	0,9517
24	0,009515	0,000093	0,86	0,3807	87	0,007879	0,000973	0,85	0,3663
25	0,007912	0,000292	0,76	0,3482	88	0,001835	0,000918	0,81	0,3691
26	0,005527	0,000589	0,87	0,3583	89	0,006141	0,000741	0,82	0,3713
27	0,00304	0,000020	0,94	0,3663	90	0,006267	0,000280	0,94	0,4766
28	0,004014	0,000548	0,91	0,3417	91	0,004891	0,000084	0,89	0,3533
29	0,008252	0,000113	0,87	0,3987	92	0,005233	0,000216	0,88	0,3208
30	0,002251	0,000601	0,85	0,3879	93	0,001475	0,000222	0,82	0,3944
31	0,006093	0,000504	0,92	0,3893	94	0,00531	0,000474	0,80	0,3677
32	0,00788	0,000049	0,76	0,3273	95	0,007816	0,000959	0,77	0,3720
33	0,009211	0,000711	0,93	0,9517	96	0,001416	0,000182	0,79	0,3547
34	0,008583	0,000878	0,90	0,4131	97	0,004967	0,000065	0,90	0,3583
35	0,004955	0,000497	0,92	0,3583	98	0,009113	0,000502	0,89	0,4254
36	0,005557	0,000559	0,95	0,4333	99	0,001138	0,000033	0,77	0,3800
37	0,005117	0,000703	0,92	0,4124	100	0,005736	0,000531	0,78	0,3533
38	0,004886	0,000023	0,90	0,3720	101	0,002454	0,000614	0,88	0,3374
39	0,002828	0,000848	0,81	0,3706	102	0,001907	0,000452	0,93	0,3324
40	0,004578	0,000090	0,83	0,3259	103	0,004859	0,000703	0,77	0,3929
41	0,008258	0,000519	0,87	0,4174	104	0,007006	0,000384	0,86	0,3576
42	0,007843	0,000845	0,92	0,4268	105	0,008128	0,000611	0,87	0,4484
43	0,009407	0,000526	0,85	0,4290	106	0,008834	0,000864	0,88	0,4211
44	0,003634	0,000328	0,85	0,3922	107	0,005499	0,000086	0,91	0,3980
45	0,003569	0,000929	0,84	0,3432	108	0,00217	0,000080	0,93	0,3547
46	0,009293	0,000036	0,86	0,5061	109	0,001286	0,000519	0,83	0,3518
47	0,007984	0,000811	0,80	0,3317	110	0,009127	0,000920	0,81	0,3850
48	0,008103	0,000835	0,87	0,4074	111	0,008212	0,000831	0,87	0,3792
49	0,001103	0,000871	0,90	0,4110	112	0,005461	0,000351	0,89	0,3526
50	0,005944	0,000130	0,81	0,3295	113	0,005337	0,000840	0,82	0,3482
51	0,001808	0,000288	0,94	0,3562	114	0,008354	0,000721	0,81	0,4448
52	0,005688	0,000741	0,77	0,3208	115	0,001562	0,000639	0,80	0,3684
53	0,007653	0,000863	0,76	0,3670	116	0,0067	0,000203	0,85	0,3598
54	0,008423	0,000027	0,91	0,4275	117	0,005275	0,000729	0,82	0,3872
55	0,001866	0,000974	0,84	0,3590	118	0,004175	0,000098	0,94	0,3864
56	0,004716	0,000469	0,86	0,3396	119	0,002118	0,000903	0,92	0,3259
57	0,00599	0,000315	0,76	0,3612	120	0,001117	0,000311	0,86	0,4102
58	0,008765	0,000813	0,75	0,3547	121	0,003026	0,000719	0,81	0,3576
59	0,005056	0,000049	0,79	0,3951	122	0,009	0,000060	0,76	0,3706
60	0,005844	0,000837	0,81	0,3677	123	0,001388	0,000692	0,94	0,3562
61	0,006098	0,000967	0,83	0,4037	124	0,004714	0,000318	0,94	0,4420
62	0,002497	0,000473	0,94	0,3605	125	0,00376	0,000704	0,88	0,3381
63	0,008828	0,000845	0,85	0,4001					

	Agiz	Agri	Alerji	Astim	Ates	Bas_Agrisi	Bas_Donmesi	Beyin	Bobrek	Bogaz	Bulanti	Grip	Halsizlik	Ilac	Kalp	Kan	Karın	Kemik	Kusmak	Nabiz	Nefes	Seker_Hastaligi	Tansiyon	Terlemek	Yara	
Agiz	172	1						2		5		2		1		1		2								
Agri	1	110					2									2					2		1			
Alerji	1		159	1						1					3	1				1					2	
Astim				158				4			1										2				1	
Ates				1	118													1				2	2	1		
Bas_Agrisi	1					168	3	1	1	3		2				2				1						
Bas_Donmesi				2	3	1	103					1	1								1					
Beyin	1					2		160		2																
Bobrek	2								180		2						1									
Bogaz	2					1				159	5										4					
Bulanti	1							3	1	167							2				11					
Grip	2					1				1		167										1				
Halsizlik		1		2	4				1		1		142	1	1						2				1	
Ilac		1		2			2			3				152	2	2					1					
Kalp		1											3	122	1	2							3			
Kan	2	1	1						1		1				149	2					1					
Karın			1					2	1	9											152	1				
Kemik					4						2	2	1	3							144	1	3		3	
Kusmak	2									7							3				174					
Nabiz				1																		151		1	1	3
Nefes		3				1				1		1				1					1	100				
Seker_Hastaligi			6		1												1				1	1	179		1	
Tansiyon			2		1													5			1			119		
Terlemek													3											1	155	
Yara				1								1			4										144	

Tahmin edilen sınıf

Şekil 3.20. Grid arama destekli modelin karmaşıklık matrisi.

	Agiz	Agri	Alerji	Astim	Ates	Bas_Agrisi	Bas_Donmesi	Beyin	Bobrek	Bogaz	Bulanti	Grip	Halsizlik	Ilac	Kalp	Kan	Karın	Kemik	Kusmak	Nabiz	Nefes	Seker_Hastaligi	Tansiyon	Terlemek	Yara	
Agiz	148	1		2	3		1	3	6	2	6		2		1	10		1								
Agri	1	108		1			2				1				2							3				
Alerji			138	6							1					2	22									
Astim				162				1									1				2					
Ates				2	116							1	2		1				2			1				
Bas_Agrisi						163	5	4	3	2	1	2								1			1			
Bas_Donmesi	1					2	101	2				2					3	1								
Beyin	1					1		156				3				4										
Bobrek				1					175		1										6				2	
Bogaz	3				4					1	149	6									3	4		1		
Bulanti				3						2		160	1								15	3		1		
Grip	5							2	3	1		156									1	3		1		
Halsizlik				4	4					1			135	1							1	5	1	2	1	1
Ilac	1		1	2			2	1	3			6	1	142	2	2					1	1		1		
Kalp									4						123						3	1	2			
Kan	3		2	2				1	3	1	1	2				1	134				3	2	1		2	
Karın											3										163					
Kemik				3										1	2							151	2	1		
Kusmak										20	3										14		150			
Nabiz			2	4				1													1		148			
Nefes				2			1					2			2						1		100			
Seker_Hastaligi			5	5		3			1												4		3	167		
Tansiyon			2												1						7		5		111	
Terlemek								1					1									1	1		155	
Yara				4												1						3			142	

Tahmin edilen sınıf

Şekil 3.21. Rastgele arama destekli modelin karmaşıklık matrisi.

geliştirilmiştir. Geliştirilen CNN modelinin başarımını artırmak için ilgili modelin eğitim parametrelerini optimize etmek bu çalışmanın bir başka amacıdır. Global arama yöntemlerinden GS ve RS yöntemleri, GoogLeNet destekli CNN modelinin eğitim parametrelerini optimize etmek için kullanılmıştır. Optimize edilmiş CNN modelinin performans değerleri doğruluk oranı kriterine göre incelenmiştir. Deneysel çalışmalar sonucunda elde edilen deneysel sonuçlar incelendiğinde, GS yöntemi destekli GoogLeNet tabanlı CNN modelinin başarımı diğer yöntemlere göre daha yüksektir. Parametre ayarlayıcı kullanılmadan sadece varsayılan eğitim parametreleri ile gerçekleştirilen deneysel çalışmanın başarımı, global arama yöntemi destekli modellere göre daha düşüktür.

Parametre optimizasyonunda farklı global arama yöntemlerinin kullanılması ve performans kıyaslaması gelecek çalışmalar için planlanmaktadır. Ayrıca, önerilen metodun farklı veri setleri üzerinde değerlendirilmesi bir diğer gelecek çalışma planıdır.

4. BÖLÜM

DERİN ÖĞRENME İLE GÜNLÜK HAREKET TANIMA

Derin öğrenme ile insan edimleri tanıma üzerine olan tez çalışmasının bu bölümünde insan hareketi tanımanın (human action recognition, HAR) bir başka alt kategorisi olan günlük insan hareketi tanıma (daily human activity recognition, DHAR) ile hem görsel verilerin hem de sensör verilerinin sınıflandırılması amaçlanmıştır. Tezin bu bölümünde derin öğrenme yöntemi kullanılarak DHAR üzerine iki farklı çalışma uygulanmıştır.

4.1. Yeni bir Hibrit Sezgisel Algoritma Kullanarak Derin Öğrenme ve Yapısal Optimizasyon ile Sensör Tabanlı İnsan Hareketi Tanıma

4.1.1. Giriş

İnsan edimlerini tanımak için hem görsel hem de sensör tabanlı verilerden faydalanılmaktadır. Görsel veriler video görüntüleri, hareketsiz resimler, iskelet görüntüleri vb. verilerden oluşurken, sensör tabanlı veriler ivmeölçer, jiroskop gibi aygıtlardan alınan sayısal verilerden oluşmaktadır. Jiroskop ve ivmeölçer sensörlerinin bulunduğu cihaz aracılığı ile oluşturulan çokça veri seti literatürde yerini almıştır. Bu veri setlerini sınıflandırmak için klasik yöntemlerin yanında derin öğrenme yöntemleri de sıklıkla kullanılmaktadır. Bir derin öğrenme yöntemi olan yığılanmış özdevinimli kodlayıcılar (stacked autoencoders, SAE), sensör tabanlı verileri sınıflandırmak için kullanılmaktadır. Literatürde yaygın olarak tercih edilen WISDM [15] ve UCI Har [14] veri setlerinin sınıflandırılmasında kullanılan SAE'nin performansını doğrudan etkileyen etmenler vardır. SAE algoritmasının hiperparametreleri optimize edilerek, ilgili veri setleri üzerinden insan edimlerini tanımak bu çalışmanın amacıdır. Kullanıcı deneyimine bırakılan hiperparametre seçimi ile optimum sonucu bulmak oldukça zor

bir süreçtir. Bu çalışmada ayrıntıları Bölüm 2.2’de verilen ABC, DE, PSO, GA ve önerilen PSO destekli ABC temelli hibrit algoritma (hABCPSO) kullanılarak SAE’nin hiperparametreleri optimize edilmiş ve birini dışarıda bırak çapraz doğrulama (leave one out cross validation, LOOCV) test yöntemiyle alınan sonuçlar karşılaştırılmıştır. Her bir algoritma için 30 koşma gerçekleştirilmiş olup bu koşmaların sonuçları istatistiksel analizle incelenmiştir.

Bu çalışmanın ana katkıları şu şekildedir [163]:

- Sensör tabanlı veri setlerinde HAR için bir optimizasyon algoritması destekli SAE önerilmiştir.
- PSO destekli ABC tabanlı hibrit algoritma (hABCPSO) geliştirilmiştir.
- Geliştirilmiş hABCPSO algoritması ile bu çalışma için en gürbüz sistem elde edilmiştir.
- Önerilen hABCPSO destekli sınıflandırma yöntemi ile WISDM veri setinde %85,36 başarı sağlayan mevcut çalışmaya [164] karşı %83,31 başarı oranı elde edilmiştir.
- Önerilen hABCPSO destekli sınıflandırma yöntemi, UCI Har veri setinde %77,81 başarı sağlayan mevcut çalışmaya [164] karşı %96,34’e varan başarı oranını elde etmiştir.

Çalışmanın bundan sonraki bölümleri için ilk olarak Bölüm 4.1.2 ile problem çözümünde kullanılacak olan veri setleri açıklanmış olup, daha sonra Bölüm 4.1.3 ile çalışmada kullanılacak önerilen yöntem hakkında bilgi verilmiştir. Bölüm 4.1.4 ile yapılan deneysel çalışmalar ve elde edilen sonuçlar detaylı bir şekilde sunulmuştur. İlgili çalışma hakkında genel değerlendirme Bölüm 4.1.5 ile ifade edilmiştir.

4.1.2. Kullanılan Veri Setleri

Bu çalışmada önerilen yöntem WISDM ve UCI Har veri setleri ile analiz edilecektir. İlgili veri setlerine ait detaylı bilgiler alt bölümlerde sunulmuştur.

4.1.2.1. WISDM Veri Seti

36 katılımcı kullanılarak oluşturulan WISDM veri seti 46 dönüştürülmüş özellik ve 5424 örnek içermektedir. Bu veri setinde toplam 6 adet hareket etiketi (“walking”, “jogging”, “upstairs”, “downstairs”, “sitting”, “standing”) bulunmaktadır [15]. WISDM veri seti için eğitim test verileri ayrımı LOOCV test tipine göre düzenlenmiştir. Toplamda 36 katılımcı bulunurken, bu katılımcıların 35’i eğitim için 1’i test için kullanılacak ve 36 farklı eğitim ve test veri seti oluşturulmuştur.

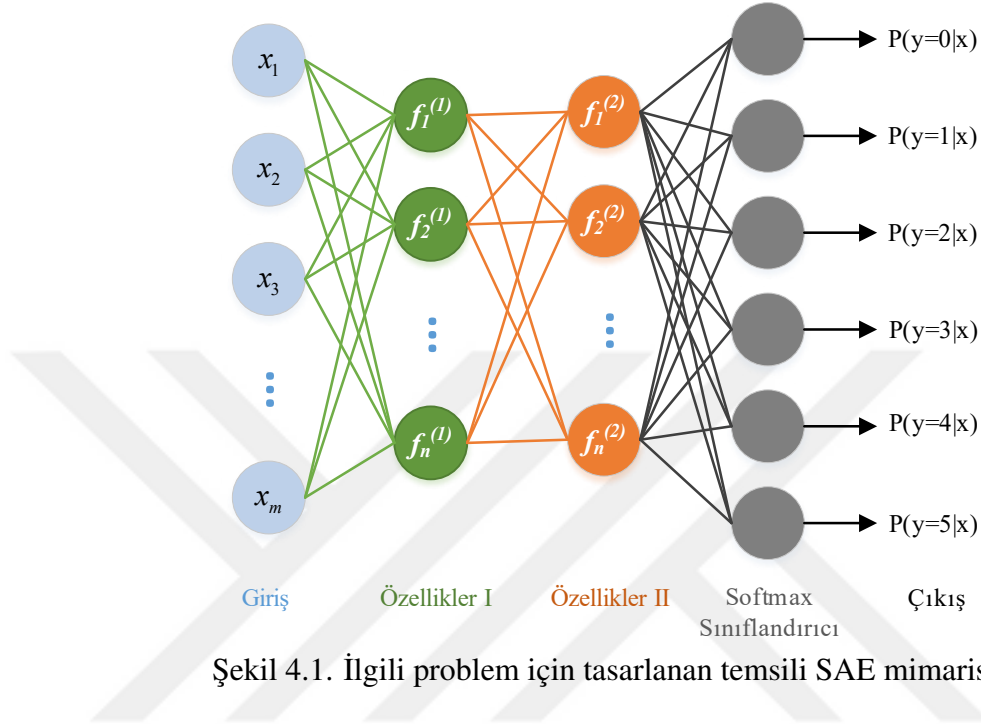
4.1.2.2. UCI Har Veri Seti

19-48 yaş arası 30 katılımcıdan alınan verilerle oluşturulan UCI Har veri seti, 561 özellik ve 10299 örnek barındırmaktadır. Her bir katılımcı 6 hareketi (“walking”, “walking_upstairs”, “walking_downstairs”, “sitting”, “standing”, “laying”) akıllı telefon kullanarak yapmaktadır. Elde edilen veriler telefonun ivmeölçer ve jiroskop aygıtlarından elde edilmektedir. Her bir özellik [-1,1] arasında normalize edilmiştir [14]. Bu veri seti WISDM veri seti ile aynı çalışmada kullanılacağından dolayı, 10299 verinin ayrımı, LOOCV test tipine uygun olarak katılımcı katılımcı ayrılacak şekilde yapılmıştır. Toplamda 30 katılımcı varken bu katılımcılardan 29’u eğitim için 1’i test için kullanılmaktadır ve bu durum her bir katılımcı tek başına test verisi olacak şekilde 30 farklı eğitim ve test veri seti oluşturulmuştur.

4.1.3. Metodoloji

SAE modeli sınıflandırma, özellik çıkarma gibi amaçlarla kullanılan bir derin öğrenme metodudur. Bölüm 2.1.2 ile ayrıntılı bilgisi verilen SAE metodu bu çalışmada optimizasyon algoritması destekli kullanılarak sensör verileri üzerinden insan hareketlerinin tanınması amacıyla kullanılmıştır. Önerilen sınıf mimarisi, Şekil 4.1 ile gösterildiği gibi iki otomatik kodlayıcı katmanı ve bir son softmax sınıflandırma katmanı kullanılarak oluşturulmuştur. İlgili SAE mimarisi için eğitim süreci üç alt aşamadan oluşmaktadır: 1. Otomatik kodlayıcıların eğitimi, 2. Softmax sınıflandırıcının eğitimi, 3. Önerilen SAE ağını oluşturmak için otomatik kodlayıcı ve softmax katmanlarının birleştirilmesi ve daha sonra tüm sistemin ayarlanması. SAE’nin başarı oranının

artırılması için yapısal parametreleri güncel zeki optimizasyon algoritmaları kullanılarak optimize edilmiştir. Bu optimizasyon sürecinde kullanılan algoritmalar Bölüm 2.2 ile ayrıntılı olarak sunulmuştur.



4.1.3.1. Önerilen Yöntem

Sensör tabanlı veriler üzerinden insan hareketlerini tanımak için geliştirilen bu yöntemde SAE derin bir sınıflandırma modeli olarak kullanılmaktadır. SAE yöntemi, enkoder transfer fonksiyonu, dekoder transfer fonksiyonu, seyreklik oranı, gizli katman boyutu, ℓ_2 ağırlık düzenleme oranı ve kayıp fonksiyonu gibi yapısal parametrelere sahiptir. Diğer sınıflandırma yöntemlerinde olduğu gibi, SAE metodunun yapısal parametreleri kullanıcı deneyimine bağlı olarak seçilirse optimum başarı oranını elde etmek çok zordur. Dolayısıyla bu çalışmada, optimizasyon algoritması kullanarak optimum SAE yapısının oluşturulması önerilmiştir.

Önerilen yöntemin temel adımları Algoritma 4.1 ile verilmiştir. İlk adım olarak, kullanılan optimizasyon algoritmasının kontrol parametreleri ayarlanır. Kontrol parametreleri ayarlanırken kullanılan algoritmalar arasında adil bir karşılaştırma için maksimum uygunluk fonksiyonu işlem sayısı aynı olmalıdır. Adım 2'de ilgili veri seti LOOCV test tipine göre eğitim ve test alt kümelerine bölünür.

Bir sonraki adımda, optimum yapısal parametreleri bulmak için görevlendirilecek optimizasyon algoritmasıyla kullanmak için `traindata` seti; `pretraindata` ve `validationdata` olarak alt kümelere ayrılır. Bir sonlandırma kriteri karşılanıncaya kadar, 5 ila 11 arasındaki adımlar tekrarlanır. Maksimum uygunluk fonksiyonu işlem sayısının elde edilmesi önerilen yöntem için sonlandırma kriteridir. Yapısal parametreler vektörünün popülasyonu, 5. Adımda kullanılan optimizasyon algoritmasına dayalı yeni çözüm üretme mekanizması kullanılarak üretilir. Algoritma popülasyon sayısına ulaşana kadar 7 ila 10 arasındaki adımlar tekrarlanır. Adım 7’de, popülasyonda sıradaki yapısal parametreler ile bir SAE yapısı kurulur. `pretraindata` ile ölçekli eşlenik gradyan (scaled conjugate gradient, SCG) algoritması kullanılarak SAE eğitimi, 8. adımda gerçekleştirilir. Bir sonraki adımda `validationdata` alt kümesi eğitilmiş SAE yapısı üzerinden test edilir ve sonuç uygunluk değeri olarak döndürülür. Adım 10’da, uygunluk değeri en iyisi ile karşılaştırılır, daha sonra en iyi uygunluk değeri ve kullanılan yapısal parametreler hafızaya alınır. En iyi yapısal parametre vektörü kullanılarak yeni bir SAE yapısı kurulur ve `traindata`, Adım 13’te SCG kullanılarak SAE ile eğitilir. Son adım olarak, eğitilmiş SAE ile `testdata` alt kümesi test işlemine girdirilir ve doğruluk oranı kriterine göre önerilen yöntemin performansı hesaplanır.

- 1: Kullanılacak optimizasyon algoritmasının kontrol parametrelerini ayarla
- 2: Tüm veri setini `testdata` ve `traindata` olarak iki farklı alt gruba ayır
- 3: `traindata` veri setini `pretraindata` ve `validationdata` adında iki farklı alt kümeye ayır
- 4: **repeat**
- 5: Kullanılan algoritmanın yeni birey üretme mekanizmasını kullanarak yapısal parametreler vektör popülasyonunu oluştur
- 6: **repeat**
- 7: Popülasyonda sıradaki yapısal parametreleri kullan ve SAE’yi ayarla
- 8: SAE metodunu `pretraindata` alt kümesi ile eğit
- 9: `validationdata` alt kümesini eğitilmiş SAE yapısına gönder ve kullanılan yapısal parametrelerin uygunluk değerini hesapla
- 10: En iyi yapısal parametreleri ve o parametrelere ait uygunluk değerini hafızada tut
- 11: **until** Popülasyon bitinceye kadar
- 12: **until** Kullanılan optimizasyon algoritmasının durdurma kriteri sağlanana kadar
- 13: En iyi yapısal parametreleri kullanarak SAE’yi ayarla ve `traindata` ile eğit
- 14: Nihai eğitilmiş SAE yapısını `testdata` alt kümesini kullanarak test et ve sistemin performansını doğruluk kriteri ile hesapla

Algoritma 4.1: Önerilen sınıflandırma yönteminin temel adımları.

Tablo 4.1. Kullanılan optimizasyon algoritmalarının kontrol parametreleri.

Algoritma	Parametre	Tanım	Veri Setleri için Değerler	
			UCI Har	WISDM
ABC	NP	Koloni büyüklüğü	20	20
	$maxCycle$	Maksimum döngü sayısı	30	60
	$limit$	Genellikle $NP * boyut / 2$	110	110
DE	NP	Pupülasyon sayısı	20	20
	$iter$	İterasyon sayısı	30	60
	CR	Çaprazlama katsayısı	0,8	0,8
	F	Mutasyon katsayısı	0,75	0,75
PSO	ps	Parçacık sayısı	20	20
	$iter$	İterasyon sayısı	30	60
	c_1	Öğrenme faktörü 1	1,49	1,49
	c_2	Öğrenme faktörü 2	1,49	1,49
	w	Eylemsizlik ağırlığı	0,7	0,7
GA	np	Populasyon sayısı	20	20
	gen	Jenerasyon sayısı	30	60
	cf	Çaprazlama katsayısı	0,8	0,8
	mf	Mutasyon katsayısı	0,2	0,2
hABCPSO	NP	Koloni büyüklüğü	20	20
	$MaxCycle$	Maksimum döngü sayısı	20	40
	$limit$	Genellikle $NP * boyut / 2$	110	110
	ss	Parçacık sayısı	10	10
	c_1	Öğrenme faktörü 1	1,49	1,49
	c_2	Öğrenme faktörü 2	1,49	1,49
	w	Eylemsizlik ağırlığı	0,7	0,7

4.1.4. Deneysel Çalışmalar

4.1.4.1. Deneysel Kurulum

Bu çalışmada sensör tabanlı verilerle insan hareketleri tanıma üzerine bir sınıflandırıcı geliştirilmiştir. Bir derin öğrenme yöntemi olan SAE, yapısal optimizasyonu zeki optimizasyon algoritmalarıyla gerçekleştirilerek WISDM ve UCI Har veri setleri üzerinde uygulanmıştır. Optimizasyon sürecinde kullanılan ABC, DE, PSO, GA ve hABCPSO algoritmalarının kontrol parametreleri Tablo 4.1 ile gösterildiği gibidir.

Test tipi olarak LOOCV yöntemi kullanıldığından 30'u UCI Har, 36'sı WISDM'den olmak üzere 66 farklı eğitim ve test setleri oluşmaktadır. Optimizasyon algoritmasının işlem süresi maliyetinin yüksek olmasından dolayı TUBITAK ULAKBİM tarafından sağlanan TRUBA çoklu bilgisayar laboratuvarı sistemi [165] kullanılmıştır.

4.1.4.2. Deneysel Sonuçlar

Optimizasyon algoritması destekli SAE kullanılarak 36 katılımcıdan oluşan WISDM veri seti ile 30 katılımcıdan oluşan UCI Har veri seti LOOCV test tekniğiyle incelenmiştir. WISDM veri setine ait tüm algoritmalar için 30’ar koşma sonucunda elde edilen hata oranları Tablo 4.2 ile sunulmuştur.

Tablo 4.2. WISDM veri seti için gerçekleştirilen 30 koşmaya ait hata değerleri.

Koşma	ABC	DE	PSO	GA	hABCPSO
1	0,2124	0,2022	0,2219	0,2213	0,1768
2	0,1993	0,2267	0,2167	0,2272	0,1921
3	0,2148	0,2401	0,2139	0,1872	0,1929
4	0,2252	0,2041	0,2235	0,2246	0,1960
5	0,2004	0,2241	0,2178	0,2200	0,1774
6	0,2178	0,2425	0,2185	0,2263	0,1836
7	0,1916	0,2549	0,1801	0,2191	0,1750
8	0,1907	0,2048	0,2182	0,1831	0,2097
9	0,1969	0,2265	0,2274	0,2165	0,1923
10	0,1921	0,2241	0,1881	0,2267	0,2062
11	0,1875	0,2373	0,1742	0,1790	0,1744
12	0,2315	0,2261	0,2250	0,2161	0,2108
13	0,2003	0,2030	0,2126	0,2243	0,2065
14	0,2047	0,2279	0,2235	0,2204	0,2071
15	0,2196	0,2287	0,2200	0,2185	0,1914
16	0,1975	0,2244	0,1969	0,2219	0,2065
17	0,1921	0,2241	0,2165	0,2220	0,2080
18	0,2291	0,2399	0,2196	0,2239	0,2148
19	0,2167	0,2379	0,1962	0,2254	0,1934
20	0,1800	0,1989	0,1720	0,1691	0,1669
21	0,2154	0,2289	0,2089	0,2152	0,1955
22	0,2049	0,2100	0,2167	0,2176	0,2004
23	0,2207	0,2305	0,1988	0,2191	0,2119
24	0,2128	0,2361	0,1956	0,2104	0,1940
25	0,2213	0,2061	0,2231	0,2161	0,2073
26	0,1907	0,2048	0,2106	0,1877	0,1831
27	0,2152	0,2211	0,1969	0,2224	0,1920
28	0,1971	0,2292	0,2174	0,2237	0,2102
29	0,1969	0,2150	0,2182	0,1872	0,2065
30	0,1921	0,2259	0,1992	0,2219	0,2104

Örnek bir alt sonuç olarak, WISDM veri setinin birinci katılımcısı için optimizasyon algoritmaları tarafından seçilen en iyi yapısal parametreler ve doğrulama verilerinde hesaplanan hata oranları Tablo 4.3 ile verilmiştir.

Tablo 4.3. Optimizasyon algoritmaları tarafından seçilen en iyi yapısal parametreler ve WISDM veri setinin ilk gönüllüsü için hesaplanan hatalar.

Parametreler	ABC	DE	PSO	GA	hABCPSO
ℓ_2 ağırlık düzenleme oranı ₁	0,000712409	0,000712409	0,003776799	0,009820524	0,008335296
seyreklik oranı ₁	0,002765411	0,002765411	0,008643709	0,004270230	0,001764923
gizli katman boyutu ₁	8	8	8	9	8
enkoder transfer fonksiyonu ₁	logsig	logsig	logsig	logsig	logsig
dekoder transfer fonksiyonu ₁	logsig	logsig	satlin	logsig	purelin
ℓ_2 ağırlık düzenleme oranı ₂	0,003034574	0,003034574	0,002974710	0,000451609	0,006031678
seyreklik oranı ₂	0,009996362	0,009996362	0,008389591	0,009702821	0,009874746
gizli katman boyutu ₂	6	6	5	7	6
enkoder transfer fonksiyonu ₂	logsig	logsig	logsig	logsig	logsig
dekoder transfer fonksiyonu ₂	satlin	satlin	satlin	logsig	logsig
kayıp fonksiyonu	crossentropy	crossentropy	crossentropy	crossentropy	crossentropy
Hata	0,1645	0,1382	0,1513	0,1579	0,1382

WISDM veri setine ait genel değerlendirme sonuçları Tablo 4.4 ile gösterildiği gibidir. Bu tabloya göre, WISDM veri seti için en düşük (maksimum, minimum, ortalama) hatalar ve standart sapma değerleri, hABCPSO destekli SAE ile elde edilmiştir.

Tablo 4.4. WISDM veri seti için genel sonuç değerlendirme.

	ABC	DE	PSO	GA	hABCPSO
Ortalama hata	0,2056	0,2235	0,2089	0,2131	0,1956
Minimum hata	0,1800	0,1989	0,1720	0,1691	0,1669
Maksimum hata	0,2315	0,2549	0,2274	0,2272	0,2148
Standart sapma	0,0137	0,0141	0,0154	0,0164	0,0133

UCI Har veri setine ait tüm algoritmalar için 30'ar koşma sonucunda elde edilen hata oranları Tablo 4.5 ile sunulmuştur.

Örnek bir alt sonuç olarak, UCI Har veri setinin birinci katılımcısı için optimizasyon algoritmaları tarafından seçilen en iyi yapısal parametreler ve doğrulama verilerinde hesaplanan hata oranları Tablo 4.6 ile verilmiştir.

UCI Har veri setine ait genel değerlendirme sonuçları Tablo 4.7 ile gösterildiği gibidir. Bu tabloya göre, UCI Har veri seti için en düşük (maksimum, minimum, ortalama) hata ve standart sapma değerleri hABCPSO destekli SAE ile elde edilmiştir.

WISDM ve UCI Har veri setlerini LOOCV test yöntemiyle çalışan ve bu test tipinde literatürde bilinen en başarılı çalışmayı gerçekleştiren ekip [164], WISDM veri setinde %85,36, UCI Har veri setinde ise %77,81'lik başarı elde etmiştir. Buna karşılık önerilen optimizasyon algoritması destekli SAE ile WISDM veri setinde %83,31'e UCI-Har veri setinde ise %96,34'e kadar başarı sağlanmıştır.

Tablo 4.5. UCI Har veri seti için gerçekleştirilen 30 koşmaya ait hata değerleri.

Koşma	ABC	DE	PSO	GA	hABCPSO
1	0,0511	0,0479	0,0487	0,0440	0,0401
2	0,0485	0,0486	0,0489	0,0476	0,0396
3	0,0514	0,0451	0,0462	0,0440	0,0403
4	0,0426	0,0490	0,0484	0,0486	0,0414
5	0,0485	0,0492	0,0462	0,0437	0,0424
6	0,0507	0,0434	0,0484	0,0444	0,0429
7	0,0485	0,0476	0,0483	0,0436	0,0397
8	0,0468	0,0423	0,0506	0,0445	0,0385
9	0,0475	0,0487	0,0468	0,0489	0,0422
10	0,0508	0,0471	0,0559	0,0466	0,0408
11	0,0495	0,0481	0,0473	0,0433	0,0403
12	0,0484	0,0496	0,0497	0,0464	0,0417
13	0,0515	0,0477	0,0470	0,0428	0,0409
14	0,0436	0,0462	0,0450	0,0418	0,0380
15	0,0498	0,0467	0,0468	0,0439	0,0409
16	0,0522	0,0468	0,0488	0,0452	0,0410
17	0,0478	0,0490	0,0484	0,0458	0,0376
18	0,0488	0,0492	0,0571	0,0487	0,0376
19	0,0461	0,0482	0,0486	0,0437	0,0396
20	0,0485	0,0504	0,0476	0,0415	0,0427
21	0,0487	0,0507	0,0486	0,0442	0,0417
22	0,0488	0,0517	0,0436	0,0451	0,0385
23	0,0468	0,0449	0,0490	0,0439	0,0400
24	0,0478	0,0507	0,0475	0,0449	0,0384
25	0,0513	0,0477	0,0538	0,0409	0,0396
26	0,0479	0,0485	0,0537	0,0449	0,0387
27	0,0486	0,0493	0,0532	0,0513	0,0366
28	0,0482	0,0463	0,0470	0,0440	0,0387
29	0,0484	0,0492	0,0479	0,0424	0,0411
30	0,0486	0,0445	0,0483	0,0427	0,0421

Tablo 4.6. Optimizasyon algoritmaları tarafından seçilen en iyi yapısal parametreler ve UCI Har veri setinin ilk gönüllüsü için hesaplanan hatalar.

Parametreler	ABC	DE	PSO	GA	hABCPSO
ℓ_2 ağırlık düzenleme oranı ₁	0,006879762	0,000975142	0,005913512	0,001187389	0,000818724
seyreklik oranı ₁	0,003417551	0,004424286	0,01	0,005856049	0,009139114
gizli katman boyutu ₁	8	6	7	5	7
enkoder transfer fonksiyonu ₁	logsig	satlin	logsig	logsig	satlin
dekoder transfer fonksiyonu ₁	purelin	satlin	logsig	logsig	satlin
ℓ_2 ağırlık düzenleme oranı ₂	0,007397451	0,006367142	0,008712199	0,004704147	0,000556032
seyreklik oranı ₂	0,008673408	0,00411798	0,01	0,009058296	0,007678787
gizli katman boyutu ₂	7	6	6	5	7
enkoder transfer fonksiyonu ₂	satlin	satlin	logsig	logsig	satlin
dekoder transfer fonksiyonu ₂	satlin	satlin	satlin	logsig	satlin
kayıp fonksiyonu	crossentropy	mse	crossentropy	crossentropy	crossentropy
Hata	0	0,0115	0	0,0029	0

Tablo 4.7. UCI Har veri seti için genel sonuç değerlendirme.

	ABC	DE	PSO	GA	hABCPSO
Ortalama hata	0,0486	0,0478	0,0489	0,0448	0,0401
Minimum hata	0,0426	0,0423	0,0436	0,0409	0,0366
Maksimum hata	0,0522	0,0517	0,0571	0,0513	0,0429
Standart sapma	0,0021	0,0022	0,0030	0,0024	0,0017

Elde edilen sonuçları istatistiksel olarak inceleyebilmek için istatistiksel analizler gerçekleştirilmiştir. WISDM veri setinde alınan sonuçlar için ilk olarak normalite testi yapılmalıdır. Yapılan normalite testi için örnek sayısı 50 ve üzeri ise Kolmogorov-Smirnov, 50 değerinin altındaysa Shapiro-Wilk testi sonucu dikkate alınır. Örnek sayısı 150 olduğu için normalite test sonucunun Kolmogorov-Smirnov kısmı ile ilgilenilecektir. Tablo 4.8 incelendiğinde Kolmogorov-Smirnov sonuçlarına göre anlamlılık değeri 0,05'in altında kalan 4 algoritma için sonuçlar normal dağılmamıştır. Bundan dolayı parametrik olmayan testlerden Kruskal Wallis ve Mann Whitney U testleri algoritma sonuçları arasında anlamlı farkın olup olmadığını tespit etmek için kullanılmıştır.

Tablo 4.8. WISDM veri seti için Kolmogorov-Smirnov normalite test sonuçları.

Algoritma	İstatistik	Anlamlılık Değeri
ABC	0,146	0,104
DE	0,182	0,012
PSO	0,221	0,001
GA	0,317	0,000
hABCPSO	0,201	0,003

WISDM veri setinin kullanıldığı uygulama için yapılan parametrik olmayan testlerde, Tablo 4.9 ile Kruskal Wallis test sonucunda anlamlılık değerinin (p) 3,1513E-9 değerine eşit olması bu algoritmalar arasında anlamlı farkın olduğunu göstermektedir. Tablo 4.10 ile her bir algoritma için ortalama sıralama değerleri verilmektedir. Bu tablo ile sezgisel olarak algoritmalar arasında anlamlı farkın olduğu görülmektedir.

Tablo 4.9. WISDM veri seti için Kruskal Wallis test istatistikleri.

Hata	
Chi-square	45,485
Anlamlılık değeri (p)	3,1513E-9

Tablo 4.10. WISDM veri seti için ortalama sıralama değerleri.

Algoritma	Ortalama sıralama
ABC	62,75
DE	111,82
PSO	74,12
GA	88,20
hABCPSO	40,62

Tablo 4.11 incelendiğinde Mann-Whitney U testinin tüm algoritmaların ikili kombinasyonları için uygulandığı ve bu algoritmaların birbirlerine göre anlamlı farka sahip olup olmadıkları gözlemlenmiştir. Tablo incelendiğinde, 0,236757 anlamlılık değeri ile en çok ABC ve PSO temelli sonuçlar arasında benzerlikler bulunmaktadır.

Tablo 4.11. WISDM veri seti için ikili algoritmalarda Mann-Whitney U test sonucu.

Algoritmalar	Anlamlılık Değeri (p)	Karşılaştırma
ABC vs DE	0,000017	ABC lehine
ABC vs PSO	0,236757	-
ABC vs GA	0,019474	ABC lehine
ABC vs hABCPSO	0,030276	hABCPSO lehine
DE vs PSO	0,000154	DE lehine
DE vs GA	0,005078	GA lehine
DE vs hABCPSO	1,7935E-7	hABCPSO lehine
PSO vs GA	0,093277	-
PSO vs hABCPSO	0,000245	hABCPSO lehine
GA vs hABCPSO	0,000010	hABCPSO lehine

WISDM veri setinde uygulandığı gibi UCI Har veri setinde de alınan sonuçlar için öncelikle normalite testi yapılmıştır. Örnek sayısı 150 olduğu için normalite test sonucunun Kolmogorov-Smirnov kısmı ile ilgilenilecektir. Tablo 4.12 incelendiğinde Kolmogorov-Smirnov sonuçlarına göre 3 algoritma için sonuçlar normal dağılmamıştır. Bundan dolayı parametrik olmayan testlerden Kruskal Wallis ve Mann Whitney U

testleri, algoritma sonuçları arasında anlamlı farkın olup olmadığını tespit etmek için kullanılmıştır.

Tablo 4.12. UCI Har veri seti için Kolmogorov-Smirnov normalite test sonuçları.

Algoritma	İstatistik	Anlamlılık Değeri
ABC	0,161	0,046
DE	0,128	0,200
PSO	0,255	0,000
GA	0,163	0,041
hABCPSO	0,103	0,200

UCI Har veri setinin kullanıldığı uygulama için yapılan parametrik olmayan testlerde, Tablo 4.13 ile Kruskal Wallis test sonucunda anlamlılık değerinin (p) 6,5207E-19 değerine eşit olması, bu algoritmalar arasında anlamlı farkın olduğunu göstermektedir. Tablo 4.14 ile her bir algoritma için ortalama sıralama değerleri verilmektedir. Yine bu tablo ile sezgisel olarak algoritmalar arasında anlamlı farkın olduğu görülmektedir.

Tablo 4.13. UCI Har veri seti için Kruskal Wallis test istatistikleri.

	Hata
Chi-square	91,437
Anlamlılık değeri (p)	6,5207E-19

Tablo 4.14. UCI Har veri seti için ortalama sıralama değerleri.

Algoritma	Ortalama sıralama
ABC	105,07
DE	96,05
PSO	101,65
GA	58,13
hABCPSO	16,60

Tablo 4.15 incelendiğinde Mann-Whitney testinin tüm algoritmaların ikili kombinasyonları için uygulandığı ve bu algoritmaların birbirlerine göre anlamlı farka sahip olup olmadıkları gözlemlenmiştir. Tablo incelendiğinde, 0,604527 anlamlılık değeri ile en çok ABC ve PSO temelli sonuçlar arasında benzerlikler bulunmaktadır.

Tablo 4.15. UCI Har veri seti için ikili algoritmalarda Mann-Whitney test sonucu.

Algoritmalar	Anlamlılık Değeri (p)	Karşılaştırma
ABC vs DE	0,216736	-
ABC vs PSO	0,604527	-
ABC vs GA	0,000002	GA lehine
ABC vs hABCPSO	3,4555E-11	hABCPSO lehine
DE vs PSO	0,549155	-
DE vs GA	0,000014	GA lehine
DE vs hABCPSO	3,838E-11	hABCPSO lehine
PSO vs GA	0,000001	GA lehine
PSO vs hABCPSO	2,8379E-11	hABCPSO lehine
GA vs hABCPSO	4,3496E-10	hABCPSO lehine

4.1.5. Değerlendirme

Sensör tabanlı verilerden insan hareketleri tanıma, görüntü tabanlı verilere göre daha zorlu bir işlemdir. Sınıflandırma problemlerindeki başarısından dolayı SAE derin öğrenme metodu, sensör verilerinden insan hareketleri tanıma üzerine kullanılmıştır. Doğruluk oranını artırmak için SAE'nin yapısal parametreleri optimizasyon algoritmaları tarafından optimize edilmiştir. Bu çalışmada ayrıca ABC temelli ve PSO destekli yeni bir hibrit algoritma (hABCPSO) önerilmiştir. ABC algoritması hem yerel hem de küresel arama yeteneğine sahip bir algoritma iken PSO algoritmasının yüksek küresel arama yeteneği bu algoritmanın kabul edilen bir avantajıdır. Küresel arama yeteneğinin daha da artması amacıyla iki optimizasyon algoritma arasında hibritleşme gerçekleştirilmiştir.

Genel kullanıma açık ve bu alanda sıklıkla kullanılan WISDM ve UCI Har veri setleri, önerilen yöntemin performans analizinde kullanılmıştır. LOOCV test tipinin kullanıldığı deneysel çalışmalar incelendiğinde hem SAE'nin başarılı bir sınıflandırıcı model olduğu hem de optimizasyon destekli SAE'nin başarı oranının arttığı tespit edilmiştir. Ayrıca, deneysel sonuçlara göre geliştirilen hABCPSO algoritması bu problem için diğer optimizasyon algoritmalarına göre daha başarılı sonuçlar vermiştir.

SAE'nin yapısını optimize etmek için farklı optimizasyon algoritmalarının kullanılması ve sensör tabanlı verilerin başka bir derin öğrenme yöntemi kullanılarak sınıflandırılması gelecekteki çalışmalar olarak planlanabilir.

4.2. Statik Tabanlı Günlük Hareket Tanımda Öneğitimli CNN Ağları İçin Performans İyileştirilmesi

4.2.1. Giriş

Bir önceki tez uygulamasında sensör tabanlı insan hareketi verileri SAE tabanlı yöntemle sınıflandırılmıştı. Sensör tabanlı verilerin dışında görüntü temelli verilerden de insan günlük hareket tanıma eylemleri gerçekleştirilebilmektedir. Bu uygulamada statik imgelerden oluşan Stanford 40 veri seti kullanılarak CNN tabanlı bir yöntemle sınıflandırma yöntemi önermek amaçlanmaktadır.

Yeni bir CNN modeli oluşturmaktan daha etkili bir yöntem, ön eğitilmiş modellerden faydalanmaktır. Ön eğitilmiş modeller; transfer öğrenme, özellik çıkarma ve doğrudan sınıflandırma gerçekleştirmek için tercih edilmektedir. En çok tercih edilen ön eğitilmiş modeller arasında yer alan AlexNet [4], VGG16 [16], VGG19 [16], GoogLeNet [12], Resnet50 [17] ve NASNet-Large [18] modelleri bu çalışmada transfer öğrenme amacıyla kullanılmıştır.

Derin öğrenme yöntemlerinin başarımını yükseltmek için tercih edilen bir başka yöntem, kullanılan metoda ait hiperparametrelerin optimize edilmesidir. Bu parametrelerin seçiminin kullanıcı deneyimine bırakılması en iyi parametre seçiminin gerçekleşmesini sağlamakta yetersiz kalabilir. Bu uygulamada, ABC algoritması [5], transfer öğrenme destekli CNN modelinin hiperparametrelerini optimize etmek üzere kullanılmıştır.

Derin öğrenme modelinin eğitiminde ve testinde kullanılacak veriler için veri ön işleme adımlarının performansa etkisi yadsınmaz. Bu çalışmada Stanford 40 veri setinde yer alan veriler için önceden tanımlı koordinatlar kullanılarak, verilerde hareketin tanımlandığı daha anlamlı kısımların kırılarak alınması sağlanmıştır. Böylelikle elde edilen verilerle daha anlamlı özellikler çıkartılarak performans artırımı amaçlanmıştır.

Stanford 40 veri setinin eğitim ve test verilerine ayrılması süreci, karşılaştırılacak çalışmalarla aynı olacak şekilde ayarlanmıştır. Buna göre, her sınıftaki ilk 100 imge eğitim, geriye kalanlar ise test verisi olacak şekilde ayrılmıştır. Bu çalışmada iki farklı

performans metriği uygulanmıştır. Bunlardan birisi sınıf bazlı ortalama doğruluk değeri olup her bir sınıf için elde edilen doğruluk oranları toplanır ve toplam sınıf sayısına bölünür. Literatürde yer alan ve karşılaştırma sürecinde kullanacağımız diğer metrik ise tüm test verilerindeki toplam doğru tahminin, test verisinde yer alan toplam örnek sayısına bölümüyle elde edilen genel doğruluktur.

Bu çalışmanın ana katkıları şu şekilde sıralanabilir:

- ABC optimizasyon algoritmasıyla optimize edilmiş, yaygın olarak kullanılan altı ön eğitilmiş model, statik görüntü tabanlı eylem tanıma için kullanılmaktadır. Bilinen kadarıyla, bu uygulama bir sezgisel optimizasyon algoritması kullanarak ön eğitilmiş 6 CNN modelinin optimizasyonunu araştırmak için gerçekleştirilen ilk çalışmadır.
- ABC algoritmasıyla optimize edilen geliştirilmiş NASNet-Large CNN modeli, literatürde mevcut yöntemlerin performansını geçerek en iyi başarı oranını elde etmiştir.
- ABC algoritmasıyla optimize edilmiş ön eğitilmiş CNN modellerinin performansları, optimize edilmemiş ön eğitilmiş CNN modellerinin performanslarından daha iyi sonuç vermektedir.

Bu çalışmanın bundan sonraki aşamaları için ilk olarak Bölüm 4.2.2 ile problem çözümünde kullanılacak olan veri seti açıklanmıştır. Daha sonra uygulanan metodoloji Bölüm 4.2.3 ile detaylı bir şekilde açıklanırken, gerçekleştirilen deneysel çalışmalar ve sonuçları Bölüm 4.2.4 ile sunulmuştur. Gerçekleştirilen uygulama ile ilgili genel bir değerlendirme Bölüm 4.2.5 ile verilmiştir.

4.2.2. Kullanılan Veri Seti

Bu çalışmada önerilen yöntem Stanford 40 veri seti ile analiz edilecektir. İlgili veri setine ait detaylı bilgiler alt bölümde sunulmuştur.

4.2.2.1. Stanford 40 Veri seti

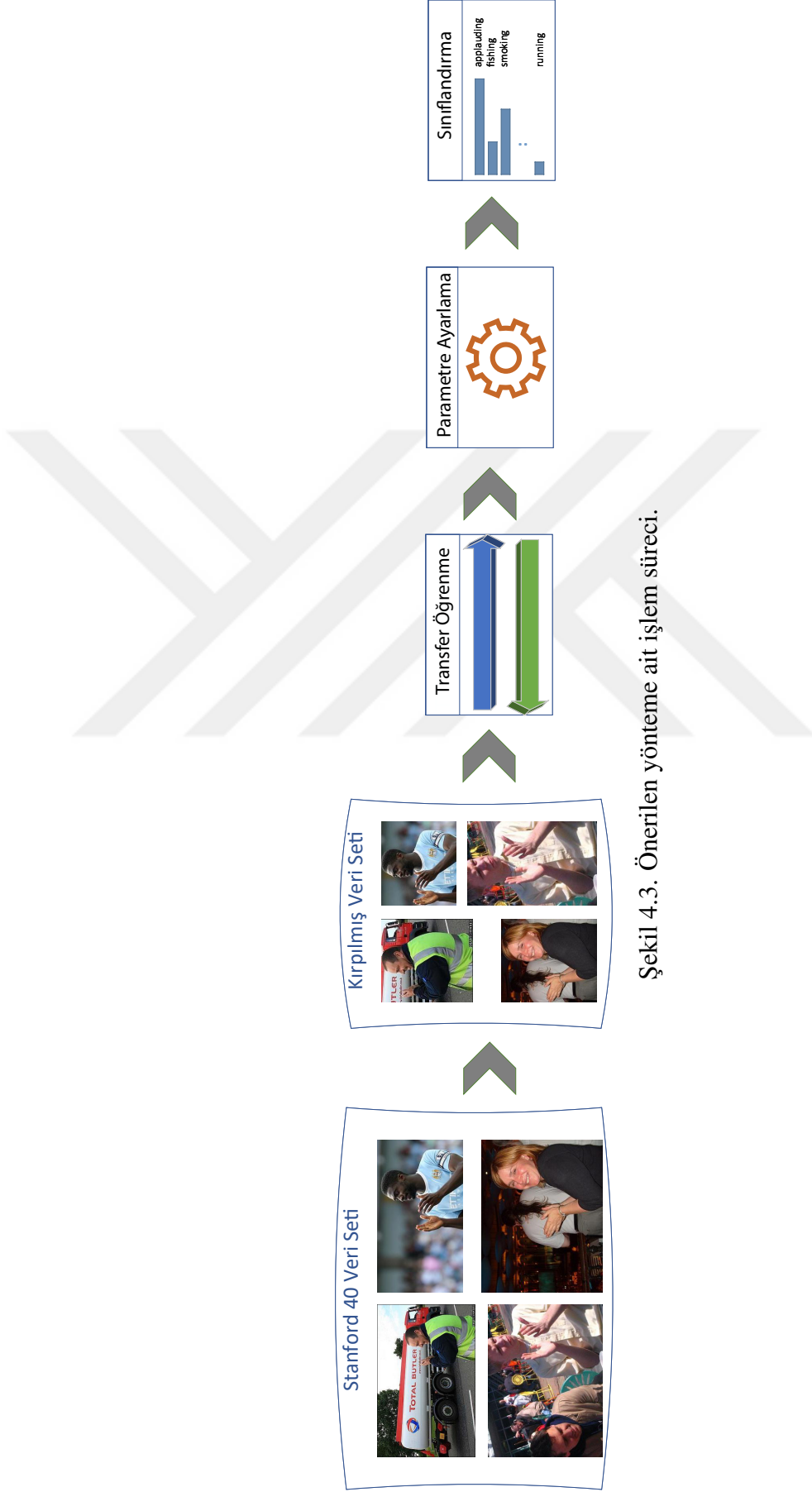
Stanford 40 veri seti [58] hareketsiz imgelerden oluşmaktadır. 9532 imgeden oluşan bu veri setinin her bir sınıftan 100 imge olacak şekilde 4000 imgesi eğitim, geriye kalan 4532 imgesi ise test verisi olarak kullanılmaktadır. Stanford 40 veri setine ait örnek ekran görüntüleri Şekil 4.2 ile gösterilmektedir.



Şekil 4.2. Stanford 40 veri setinden örnek kareler.

4.2.3. Metodoloji

Bu yazıda, görüntü ön işleme, transfer öğrenme ve hiperparametre ayarlama içeren entegre bir yöntem önerilmiştir. Önerilen yöntemin mimarisi Şekil 4.3 ile sunulmuştur. Öncelikle, görüntü ön işleme adımıyla, önceden belirlenmiş sınırlama kutusu koordinatlarını kullanarak veri集中的 tüm görüntüler için anlamlı alanın kırılması gerçekleştirilir. Daha sonra, ilgili hareket tanıma problemine özel olarak transfer öğrenme tabanlı bir CNN modeli üretilir. Daha sonraki aşamada, hiperparametre ayarlaması yapılır. Hiperparametre ayarlama aşamasında, ABC optimizasyon algoritması kullanılmaktadır. 4000 imgeden oluşturulan eğitim veri seti, alt eğitim (2400 imge) ve alt doğrulama (1600 imge) veri setlerine bölünür. Alt eğitim verileri, alt doğrulama verileri, ön eğitilmiş model destekli CNN ve ABC algoritmasından üretilen hiperparametreler uygunluk değerini hesaplamak için kullanılır. Uygunluk fonksiyonunun maliyet değeri Eşitlik 4.1 ile hesaplanmaktadır. Öte yandan, çapraz entropi kaybı, CNN'nin eğitimi için momentum ile stokastik gradyan inişi ile birlikte kullanılmaktadır.



Şekil 4.3. Önerilen yöntemle ait işlem süreci.

$$\text{maliyet} = \frac{\text{yanlış tahmin sayısı}}{1600} \quad (4.1)$$

ABC algoritmasının fazları, durdurma kriteri yerine getirilinceye kadar tekrar edilir. Optimum değere en yakın hiperparametre değerleri, ABC algoritmasının çıktısı olarak döndürülür. Daha sonra, orijinal eğitim verileri ve ABC tarafından belirlenen hiperparametre değerleri kullanılarak transfer öğrenme destekli CNN mimarisi üzerinde eğitim gerçekleştirilir. Son olarak, optimize edilmiş mimari ile elde edilen eğitilmiş modeller, test verileri üzerinde değerlendirilir.

4.2.4. Deneysel Çalışmalar

4.2.4.1. ABC Algoritmasının Kontrol Parametre Seçimi

CNN metodunun hiperparametrelerini optimize etmek için kullanılan ABC algoritmasının performansını etkileyen kendi parametreleri mevcuttur. Kontrol parametreleri olarak adlandırılan bu parametrelerin seçimi, varsayılan değerlerine yakın olarak seçilen farklı değerlerin performansına göre yapılmıştır. ABC algoritmasına ait kontrol parametreleri ve bu çalışma için seçilen değerleri Tablo 4.16 ile sunulmuştur. ABC algoritmasında kullanılmak üzere CNN hiperparametrelerine ait alt ve üst sınırlar sırasıyla [5, 4, 0,001, 0, 0001, 1, 0,7] ve [10, 11, 0,01, 0,001, 3, 0,95] olarak belirlenmiştir.

Tablo 4.16. ABC optimizasyon algoritması kontrol parametreleri.

Parametre	Açıklama	Değer
NP	Koloni büyüklüğü	10
$maxCycle$	Maksimum döngü sayısı	10
$limit$	Genellikle $NP * D/2$	30

4.2.4.2. Donanımsal Kurulum

Stanford 40 veri setindeki örnek sayısının oldukça yüksek olması ve veri setinin görüntü tabanlı olmasından dolayı deneysel çalışmalarda 4 GB Ram destekli NVIDIA GTX 960 donanımı kullanılmıştır.

4.2.4.3. AlexNet Tabanlı Yöntem İçin Deneysel Sonuçlar

Stanford 40 veri seti üzerinde AlexNet kullanarak transfer öğrenmenin başarısı Tablo 4.17 ile gösterildiği gibidir. Bu tabloya göre, ABC algoritması kullanılmadan % 78,3 başarı oranıyla “playing_violin” işlemi, ABC algoritması kullanılarak % 94,1 başarı oranıyla “rowing_a_boat” işlemi için en iyi performans elde edilir .

En kötü performans değerlendirildiğinde, ABC algoritması kullanmadan % 6,3 başarı oranı ile “playing_guitar” eylemi ve “writing_on_a_book” işlemi için ABC algoritması kullanarak % 14,4 başarı oranı elde edilmiştir.

Ortalama başarı oranı, sınıf temelli performans kriterleri için, ABC’siz % 44,1 , ABC ile % 58,4’tür. Öte yandan, genel ortalama başarı, ABC’siz % 44,9 iken ABC ile % 60,1’dir.

4.2.4.4. VGG16 Tabanlı Yöntem İçin Deneysel Sonuçlar

Stanford 40 veri seti üzerinde VGG16 kullanarak transfer öğrenmenin başarısı Tablo 4.18 ile gösterildiği gibidir. Bu tabloya göre, ABC algoritması kullanılmadan % 93,8 başarı oranıyla “holding_an_umbrella” işlemi, ABC algoritması kullanılarak % 94,4 başarı oranıyla “riding_a_horse” işlemi için en iyi performans elde edilir.

En kötü performans değerlendirildiğinde, ABC algoritması kullanmadan % 6,6 başarı oranı ile “fixing_a_car” eylemi ve “playing_guitar” işlemi için ABC algoritması kullanarak % 17 başarı oranı elde edilmiştir.

Ortalama başarı oranı, sınıf temelli performans kriterleri için, ABC’siz % 58,4 , ABC ile % 64,8’dir. Öte yandan, genel ortalama başarı, ABC’siz % 59,6 iken ABC ile % 66’dır.

Tablo 4.17. AlexNet ile transfer öğreniminin Stanford 40 veri setindeki başarımı.

Hareket	Örnek Sayısı	ABC Desteksiz Başarılı Tanıma		ABC Destekli Başarılı Tanıma	
		Sayı	Oran	Sayı	Oran
applauding	184	79	0.429	76	0.413
blowing_bubbles	159	35	0.220	86	0.541
brushing_teeth	100	26	0.260	50	0.500
cleaning_the_floor	112	75	0.670	88	0.786
climbing	195	146	0.749	173	0.887
cooking	188	58	0.309	129	0.686
cutting_trees	103	74	0.718	81	0.786
cutting_vegetables	89	35	0.393	40	0.449
drinking	156	10	0.064	31	0.199
feeding_a_horse	187	123	0.658	138	0.738
fishing	173	98	0.566	136	0.786
fixing_a_bike	128	53	0.414	88	0.688
fixing_a_car	151	73	0.483	107	0.709
gardening	99	59	0.596	67	0.677
holding_an_umbrella	192	97	0.505	149	0.776
jumping	195	87	0.446	121	0.621
looking_through_a_microscope	91	67	0.736	47	0.516
looking_through_a_telescope	103	36	0.350	53	0.515
playing_guitar	159	10	0.063	40	0.252
playing_violin	189	148	0.783	157	0.831
pouring_liquid	160	88	0.550	88	0.550
pushing_a_cart	100	34	0.340	21	0.210
reading	135	56	0.415	102	0.756
phoning	145	10	0.069	32	0.221
riding_a_bike	193	143	0.741	170	0.881
riding_a_horse	196	76	0.388	170	0.867
rowing_a_boat	85	64	0.753	80	0.941
running	151	85	0.563	99	0.656
shooting_an_arrow	114	57	0.500	85	0.746
smoking	141	41	0.291	56	0.397
taking_photos	97	30	0.309	32	0.330
texting_message	93	15	0.161	24	0.258
throwing_frisby	102	64	0.627	72	0.706
using_a_computer	130	26	0.200	59	0.454
walking_the_dog	193	133	0.689	141	0.731
washing_dishes	82	11	0.134	35	0.427
watching_TV	123	65	0.528	96	0.780
waving_hands	110	12	0.109	22	0.200
writing_on_a_board	83	55	0.663	64	0.771
writing_on_a_book	146	28	0.192	21	0.144
Sınıf Bazlı Ortalama Tanıma Performansı			0.441		0.584
Örnek Bazlı Tanıma Performansı (5532 örnek üzerinden)		2482	0.449	3326	0.601

Tablo 4.18. VGG16 ile transfer öğreniminin Stanford 40 veri setindeki başarımı.

Hareket	Örnek Sayısı	ABC Desteksiz Başarılı Tanıma		ABC Destekli Başarılı Tanıma	
		Sayı	Oran	Sayı	Oran
applauding	184	31	0.168	89	0.484
blowing_bubbles	159	76	0.478	86	0.541
brushing_teeth	100	72	0.720	66	0.660
cleaning_the_floor	112	103	0.920	97	0.866
climbing	195	147	0.754	160	0.821
cooking	188	102	0.543	103	0.548
cutting_trees	103	65	0.631	64	0.621
cutting_vegetables	89	59	0.663	53	0.596
drinking	156	53	0.340	49	0.314
feeding_a_horse	187	131	0.701	154	0.824
fishing	173	143	0.827	125	0.723
fixing_a_bike	128	82	0.641	75	0.586
fixing_a_car	151	10	0.066	133	0.881
gardening	99	84	0.848	72	0.727
holding_an_umbrella	192	180	0.938	153	0.797
jumping	195	122	0.626	129	0.662
looking_through_a_microscope	91	47	0.516	74	0.813
looking_through_a_telescope	103	66	0.641	58	0.563
playing_guitar	159	24	0.151	27	0.170
playing_violin	189	177	0.937	169	0.894
pouring_liquid	160	134	0.838	123	0.769
pushing_a_cart	100	12	0.120	27	0.270
reading	135	86	0.637	91	0.674
phoning	145	39	0.269	64	0.441
riding_a_bike	193	180	0.933	178	0.922
riding_a_horse	196	150	0.765	185	0.944
rowing_a_boat	85	75	0.882	80	0.941
running	151	96	0.636	116	0.768
shooting_an_arrow	114	84	0.737	95	0.833
smoking	141	21	0.149	62	0.440
taking_photos	97	33	0.340	43	0.443
texting_message	93	21	0.226	24	0.258
throwing_frisby	102	57	0.559	79	0.775
using_a_computer	130	44	0.338	87	0.669
walking_the_dog	193	153	0.793	151	0.782
washing_dishes	82	34	0.415	48	0.585
watching_TV	123	105	0.854	70	0.569
waving_hands	110	45	0.409	43	0.391
writing_on_a_board	83	57	0.687	65	0.783
writing_on_a_book	146	98	0.671	85	0.582
Sınıf Bazlı Ortalama Tanıma Performansı			0.584		0.648
Örnek Bazlı Tanıma Performansı (5532 örnek üzerinden)		3298	0.596	3652	0.660

4.2.4.5. VGG19 Tabanlı Yöntem İçin Deneysel Sonuçlar

Stanford 40 veri seti üzerinde VGG19 kullanarak transfer öğrenmenin başarısı Tablo 4.19 ile gösterildiği gibidir. Bu tabloya göre, ABC algoritması kullanılmadan % 95,9 başarı oranıyla “riding_a_horse” işlemi, ABC algoritması kullanılarak % 96,7 başarı oranıyla “fixing_a_car” işlemi için en iyi performans elde edilir.

En kötü performans değerlendirildiğinde, ABC algoritması kullanmadan % 19 başarı oranı ile “pushing_a_cart” eylemi ve aynı eylem için ABC algoritması kullanarak % 18 başarı oranı elde edilmiştir.

Ortalama başarı oranı, sınıf temelli performans kriterleri için, ABC’siz % 60,9 , ABC ile % 67’dir. Öte yandan, genel ortalama başarı, ABC’siz % 62,1 iken ABC ile % 68,5’tir.

4.2.4.6. GoogLeNet Tabanlı Yöntem İçin Deneysel Sonuçlar

Stanford 40 veri seti üzerinde GoogLeNet kullanarak transfer öğrenmenin başarısı Tablo 4.20 ile gösterildiği gibidir. Bu tabloya göre, ABC algoritması kullanılmadan % 97,4 başarı oranıyla “riding_a_horse” işlemi, ABC algoritması kullanılarak % 97,4 başarı oranıyla “climbing” işlemi için en iyi performans elde edilir.

En kötü performans değerlendirildiğinde, ABC algoritması kullanmadan % 30 başarı oranı ile “pushing_a_cart” eylemi ve “texting_message” işlemi için ABC algoritması kullanarak % 29 başarı oranı elde edilmiştir.

Ortalama başarı oranı, sınıf temelli performans kriterleri için, ABC’siz % 73,5 , ABC ile % 74,9’dur. Öte yandan, genel ortalama başarı, ABC’siz % 74,4 iken ABC ile % 76,1’dir.

Tablo 4.19. VGG19 ile transfer öğreniminin Stanford 40 veri setindeki başarımı.

Hareket	Örnek Sayısı	ABC Desteksiz Başarılı Tanıma		ABC Destekli Başarılı Tanıma	
		Sayı	Oran	Sayı	Oran
applauding	184	87	0.473	81	0.440
blowing_bubbles	159	84	0.528	99	0.623
brushing_teeth	100	49	0.490	59	0.590
cleaning_the_floor	112	91	0.813	97	0.866
climbing	195	153	0.785	174	0.892
cooking	188	64	0.340	131	0.697
cutting_trees	103	68	0.660	85	0.825
cutting_vegetables	89	71	0.798	66	0.742
drinking	156	30	0.192	49	0.314
feeding_a_horse	187	142	0.759	160	0.856
fishing	173	135	0.780	142	0.821
fixing_a_bike	128	96	0.750	100	0.781
fixing_a_car	151	93	0.616	146	0.967
gardening	99	71	0.717	78	0.788
holding_an_umbrella	192	151	0.786	169	0.880
jumping	195	116	0.595	152	0.779
looking_through_a_microscope	91	71	0.780	74	0.813
looking_through_a_telescope	103	45	0.437	83	0.806
playing_guitar	159	36	0.226	39	0.245
playing_violin	189	167	0.884	164	0.868
pouring_liquid	160	135	0.844	140	0.875
pushing_a_cart	100	19	0.190	18	0.180
reading	135	104	0.770	92	0.681
phoning	145	57	0.393	61	0.421
riding_a_bike	193	172	0.891	164	0.850
riding_a_horse	196	188	0.959	181	0.923
rowing_a_boat	85	78	0.918	76	0.894
running	151	93	0.616	85	0.563
shooting_an_arrow	114	86	0.754	92	0.807
smoking	141	57	0.404	27	0.191
taking_photos	97	32	0.330	39	0.402
texting_message	93	44	0.473	27	0.290
throwing_frisby	102	80	0.784	71	0.696
using_a_computer	130	36	0.277	46	0.354
walking_the_dog	193	151	0.782	162	0.839
washing_dishes	82	29	0.354	39	0.476
watching_TV	123	110	0.894	117	0.951
waving_hands	110	29	0.264	59	0.536
writing_on_a_board	83	47	0.566	55	0.663
writing_on_a_book	146	69	0.473	89	0.610
Sınıf Bazlı Ortalama Tanıma Performansı			0.609		0.670
Örnek Bazlı Tanıma Performansı (5532 örnek üzerinden)		3436	0.621	3788	0.685

Tablo 4.20. GoogLeNet ile transfer öğreniminin Stanford 40 veri setindeki başarımı.

Hareket	Örnek Sayısı	ABC Desteksiz Başarılı Tanıma		ABC Destekli Başarılı Tanıma	
		Sayı	Oran	Sayı	Oran
applauding	184	102	0.554	99	0.538
blowing_bubbles	159	104	0.654	119	0.748
brushing_teeth	100	68	0.680	69	0.690
cleaning_the_floor	112	105	0.938	94	0.839
climbing	195	182	0.933	190	0.974
cooking	188	139	0.739	112	0.596
cutting_trees	103	91	0.883	81	0.786
cutting_vegetables	89	52	0.584	68	0.764
drinking	156	57	0.365	90	0.577
feeding_a_horse	187	165	0.882	164	0.877
fishing	173	152	0.879	145	0.838
fixing_a_bike	128	106	0.828	112	0.875
fixing_a_car	151	133	0.881	138	0.914
gardening	99	70	0.707	79	0.798
holding_an_umbrella	192	179	0.932	185	0.964
jumping	195	143	0.733	162	0.831
looking_through_a_microscope	91	76	0.835	74	0.813
looking_through_a_telescope	103	69	0.670	75	0.728
playing_guitar	159	54	0.340	80	0.503
playing_violin	189	155	0.820	172	0.910
pouring_liquid	160	133	0.831	133	0.831
pushing_a_cart	100	30	0.300	45	0.450
reading	135	127	0.941	115	0.852
phoning	145	90	0.621	89	0.614
riding_a_bike	193	157	0.813	181	0.938
riding_a_horse	196	191	0.974	190	0.969
rowing_a_boat	85	79	0.929	80	0.941
running	151	110	0.728	122	0.808
shooting_an_arrow	114	98	0.860	97	0.851
smoking	141	89	0.631	67	0.475
taking_photos	97	72	0.742	57	0.588
texting_message	93	38	0.409	27	0.290
throwing_frisby	102	75	0.735	85	0.833
using_a_computer	130	96	0.738	70	0.538
walking_the_dog	193	151	0.782	164	0.850
washing_dishes	82	57	0.695	62	0.756
watching_TV	123	109	0.886	103	0.837
waving_hands	110	67	0.609	63	0.573
writing_on_a_board	83	69	0.831	70	0.843
writing_on_a_book	146	75	0.514	83	0.568
Sınıf Bazlı Ortalama Tanıma Performansı			0.735		0.749
Örnek Bazlı Tanıma Performansı (5532 örnek üzerinden)		4115	0.744	4211	0.761

4.2.4.7. ResNet50 Tabanlı Yöntem İçin Deneysel Sonuçlar

Stanford 40 veri seti üzerinde ResNet50 kullanarak transfer öğrenmenin başarısı Tablo 4.21 ile gösterildiği gibidir. Bu tabloya göre, ABC algoritması kullanılmadan % 97,4 başarı oranıyla “riding_a_bike” ve “fixing_a_car” işlemleri, ABC algoritması kullanılarak % 98,4 başarı oranıyla “riding_a_bike” işlemi için en iyi performans elde edilir.

En kötü performans değerlendirildiğinde, ABC algoritması kullanmadan % 43 başarı oranı ile “texting_message” eylemi ve “pushing_a_cart” işlemi için ABC algoritması kullanarak % 45 başarı oranı elde edilmiştir.

Ortalama başarı oranı, sınıf temelli performans kriterleri için, ABC’siz % 77,6 , ABC ile % 79,1’dir. Öte yandan, genel ortalama başarı, ABC’siz % 78,7 iken ABC ile % 80,1’dir.

4.2.4.8. NASNet-Large Tabanlı Yöntem İçin Deneysel Sonuçlar

Stanford 40 veri seti üzerinde NASNet-Large kullanarak transfer öğrenmenin başarısı Tablo 4.22 ile gösterildiği gibidir. Bu tabloya göre, ABC algoritması kullanılmadan % 100 başarı oranıyla “rowing_a_boat” işlemi, ABC algoritması kullanılarak % 99,5 başarı oranıyla “riding_a_horse” işlemi için en iyi performans elde edilir.

En kötü performans değerlendirildiğinde, ABC algoritması kullanmadan % 58.2 başarı oranı ile “waving_hands” eylemi ve “texting_message” işlemi için ABC algoritması kullanarak % 60.2 başarı oranı elde edilmiştir.

Ortalama başarı oranı, sınıf temelli performans kriterleri için, ABC’siz % 86.1 , ABC ile % 87’dir. Öte yandan, genel ortalama başarı, ABC’siz % 87 iken ABC ile % 87.8’dir.

Tablo 4.21. ResNet50 ile transfer öğreniminin Stanford 40 veri setindeki başarımı.

Hareket	Örnek Sayısı	ABC Desteksiz Başarılı Tanıma		ABC Destekli Başarılı Tanıma	
		Sayı	Oran	Sayı	Oran
applauding	184	115	0.625	117	0.636
blowing_bubbles	159	114	0.717	114	0.717
brushing_teeth	100	69	0.690	69	0.690
cleaning_the_floor	112	102	0.911	106	0.946
climbing	195	188	0.964	187	0.959
cooking	188	140	0.745	139	0.739
cutting_trees	103	96	0.932	95	0.922
cutting_vegetables	89	68	0.764	66	0.742
drinking	156	79	0.506	82	0.526
feeding_a_horse	187	169	0.904	174	0.930
fishing	173	155	0.896	154	0.890
fixing_a_bike	128	103	0.805	110	0.859
fixing_a_car	151	147	0.974	146	0.967
gardening	99	81	0.818	87	0.879
holding_an_umbrella	192	183	0.953	182	0.948
jumping	195	149	0.764	152	0.779
looking_through_a_microscope	91	82	0.901	86	0.945
looking_through_a_telescope	103	76	0.738	77	0.748
playing_guitar	159	71	0.447	91	0.572
playing_violin	189	175	0.926	176	0.931
pouring_liquid	160	146	0.912	151	0.944
pushing_a_cart	100	45	0.450	45	0.450
reading	135	106	0.785	110	0.815
phoning	145	89	0.614	83	0.572
riding_a_bike	193	188	0.974	190	0.984
riding_a_horse	196	189	0.964	192	0.980
rowing_a_boat	85	82	0.965	83	0.976
running	151	121	0.801	125	0.828
shooting_an_arrow	114	96	0.842	95	0.833
smoking	141	82	0.582	73	0.518
taking_photos	97	58	0.598	60	0.619
texting_message	93	40	0.430	44	0.473
throwing_frisby	102	81	0.794	87	0.853
using_a_computer	130	91	0.700	97	0.746
walking_the_dog	193	176	0.912	175	0.907
washing_dishes	82	65	0.793	58	0.707
watching_TV	123	107	0.870	108	0.878
waving_hands	110	58	0.527	65	0.591
writing_on_a_board	83	71	0.855	76	0.916
writing_on_a_book	146	100	0.685	105	0.719
Sınıf Bazlı Ortalama Tanıma Performansı			0.776		0.791
Örnek Bazlı Tanıma Performansı (5532 örnek üzerinden)		4353	0.787	4432	0.801

Tablo 4.22. NASNet-Large ile transfer öğreniminin Stanford 40 veri setindeki başarımı.

Hareket	Örnek Sayısı	ABC Desteksiz Başarılı Tanıma		ABC Destekli Başarılı Tanıma	
		Sayı	Oran	Sayı	Oran
applauding	184	125	0.679	139	0.755
blowing_bubbles	159	146	0.918	136	0.855
brushing_teeth	100	77	0.770	81	0.810
cleaning_the_floor	112	110	0.982	110	0.982
climbing	195	184	0.944	191	0.979
cooking	188	154	0.819	154	0.819
cutting_trees	103	96	0.932	97	0.942
cutting_vegetables	89	70	0.787	74	0.831
drinking	156	125	0.801	121	0.776
feeding_a_horse	187	174	0.930	185	0.989
fishing	173	159	0.919	159	0.919
fixing_a_bike	128	111	0.867	119	0.930
fixing_a_car	151	147	0.974	145	0.960
gardening	99	88	0.889	94	0.949
holding_an_umbrella	192	191	0.995	189	0.984
jumping	195	171	0.877	168	0.862
looking_through_a_microscope	91	82	0.901	87	0.956
looking_through_a_telescope	103	92	0.893	80	0.777
playing_guitar	159	116	0.730	110	0.692
playing_violin	189	185	0.979	184	0.974
pouring_liquid	160	159	0.994	154	0.963
pushing_a_cart	100	61	0.610	63	0.630
reading	135	125	0.926	126	0.933
phoning	145	106	0.731	109	0.752
riding_a_bike	193	190	0.984	189	0.979
riding_a_horse	196	193	0.985	195	0.995
rowing_a_boat	85	85	1.000	83	0.976
running	151	118	0.781	127	0.841
shooting_an_arrow	114	112	0.982	104	0.912
smoking	141	92	0.652	99	0.702
taking_photos	97	65	0.670	73	0.753
texting_message	93	57	0.613	56	0.602
throwing_frisby	102	94	0.922	94	0.922
using_a_computer	130	122	0.938	112	0.862
walking_the_dog	193	190	0.984	188	0.974
washing_dishes	82	73	0.890	69	0.841
watching_TV	123	116	0.943	119	0.967
waving_hands	110	64	0.582	73	0.664
writing_on_a_board	83	75	0.904	79	0.952
writing_on_a_book	146	113	0.774	120	0.822
Sınıf Bazlı Ortalama Tanıma Performansı			0.861		0.870
Örnek Bazlı Tanıma Performansı (5532 örnek üzerinden)		4813	0.870	4855	0.878

Tablo 4.23. Ön eğitimli modeller için seçilen parametreler ve doğruluk oranları.

Parametreler	ABC Desteksiz										ABC Destekli				
	AlexNet	VGG16	VGG19	GoogLeNet	ResNet50	NASNet-Large	AlexNet	VGG16	VGG19	GoogLeNet	ResNet50	NASNet-Large			
Maksimum epok	10	10	10	10	10	10	6	9	9	9	10	5			
İlk öğrenme oranı	0,001	0,001	0,001	0,001	0,001	0,001	0,001	0,0016	0,001	0,001	0,0032	0,002419			
Momentum	0,9	0,9	0,9	0,9	0,9	0,9	0,619	0,774	0,802	0,729	0,73	0,943971			
ℓ_2 düzenleme oranı	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001	0,000178	0,000292	0,000394	0,00081	0,000494	0,000150			
Karıştırma tipi	bir kere	bir kere	bir kere	bir kere	bir kere	bir kere	her epokta	her epokta	her epokta	her epokta	her epokta	bir kere			
Minibatch boyutu	11	11	11	11	11	11	10	9	9	9	11	8			
Doğruluk(%)	44,9	59,6	62,1	74,4	78,7	87,0	60,1	66,0	68,5	76,1	80,1	87,8			

AlexNet, VGG16, VGG19, GoogLeNet, ResNet50 ve NASNet-Large ön eğitilmiş modellerden transfer öğrenme ile uygulamalar geliştirilmiştir. Ayrıca, ön eğitilmiş model destekli CNN yapısının hiperparametreleri ABC optimizasyon algoritması ile seçilmiştir. ABC destekli/desteksiz ön eğitilmiş modelden transfer öğrenme gerçekleştirilerek elde edilen parametreler ve bu parametrelerin seçimi ile elde edilen sonuçlar Tablo 4.23 ile gösterilmiştir. Kullanıcı deneyimine bırakılan parametre seçimi ile optimum sonuçları bulmak oldukça güçtür. Bunun yerine ABC optimizasyon algoritması kullanarak parametre seçimi ile optimum sonuca yakın değerleri tespit etmek mümkündür.

Bu çalışmada, Stanford 40 veri setindeki performans ölçümü için genel doğruluk başarı oranı kullanılmıştır. Önerilen ve mevcut yöntemler için performans karşılaştırması Tablo 4.24 ile sunulmuştur. Tabloya göre, önerilen yöntem diğerleri arasında en iyi performansı vermektedir.

Tablo 4.24. Önerilen ve var olan yöntemlerin performans karşılaştırması.

Çalışma	Metod	Başarı oranı
Lavinia et al. (2016)	CNN'lerin füzyonu	81,15%
Bas et al. (2017)	Çoklu örnek öğrenme	66,13%
Rosenfeld and Ullman (2018)	Algılanan kavramların ağırlık toplamı	83,12%
Lavinia et al. (2019)	CNN'lerin ve iki renk uzayının füzyonu	84,27%
Xu et al. (2019)	Nonnegative Representation Classifier	81,90%
Chu and Chu (2019)	Genetik programlamaya dayalı füzyon yöntemi	61,80%
Önerilen	ABC destekli NASNet-Large tabanlı CNN	87,78%

4.2.5. Değerlendirme

AlexNet, VGG16, VGG19, GoogleNet, Resnet50 ve NASNet-Large ön eğitilmiş modellerin Stanford 40 veri seti ile çalıştırılmasında kullanılacak olan ABC destekli/desteksiz temel parametrelerle koşmalar gerçekleştirilmiştir. ABC destekli parametrelerin seçimi kullanıcı deneyimine bırakılıp farklı parametre değerleriyle koşmalar yapılarak elde edilirken ABC destekli parametrelerin seçimi algoritmaya bırakılmıştır.

Stanford 40 veri seti üzerinde, ön eğitilmiş modeller aracılığı ile yapılan ABC destekli/desteksiz transfer öğrenme çalışmalarının başarımlarını karşılaştırması bu çalışmanın amacıdır. Bu çalışmada 2 farklı performans kriteri uygulanmıştır. Sınıf bazlı başarımlar,

40 farklı sınıftan elde edilen sınıf bazlı başarının ortalaması iken, örnek bazlı başarı (genel doğruluk başarımı), toplam doğru tahmin sayısının 5532 sayısına (tüm test verisinin toplam sayısı) bölüm değerine eşittir. Bu çalışmaya göre ABC destekli/desteksiz NASNet-Large modelinden transfer eğitimin başarımı hem sınıf bazlı hem örnek bazlı başarıda en iyi sonucu vermektedir. CNN'nin hiperparametrelerinin (maksimum epok, minibatch boyutu, ℓ_2 düzenlileştirme oranı, ilk öğrenme oranı, karıştırma tipi ve momentum) ABC ile en iyilenmesi sonucunda tüm ön eğitilmiş modellerde başarı oranı artmıştır.

Gelecek çalışma olarak farklı veri setleri ve farklı optimizasyon algoritmaları kullanılarak başarı oranı üzerinden performans karşılaştırılması yapılabilir.

5. BÖLÜM

DERİN ÖĞRENME İLE DUYGU TANIMA

İnsan hareketi tanımının (human action recognition, HAR) bir diğer alt kategorisi olan duygu tanıma (emotion recognition, ER) tezin bu bölümünde incelenmiştir. Bu doğrultuda derin öğrenme yöntemi kullanılarak ER üzerine iki farklı çalışma gerçekleştirilmiştir.

5.1. Evrişimli Sinir Ağı Kullanarak Statik Görüntü Tabanlı Duygu Tanıma

5.1.1. Giriş

Bir olayın veya durumun insanın iç dünyasında oluşturduğu etki duygudur. Mutlu, üzgün, öğrenmiş, korkmuş, sinirlenmiş gibi duygular temel kabul edilen duygulardır [166]. ER ise insanların olaylara karşı duygusal tepkisinin derin öğrenme veya klasik yöntemler gibi çeşitli sınıflandırıcılar ile yapay zeka tarafından anlaşılır hale getirilmesidir.

Gelişen teknoloji ile derin öğrenme yöntemleri sınıflandırma problemlerinde sıklıkla tercih edilmektedir. Yeni bir evrişimli sinir ağı (convolutional neural network, CNN) mimarisi kurmaktan daha etkili ve hızlı olan yöntem transfer öğrenmedir (transfer learning) [167]. AlexNet, GoogLeNet, ResNet gibi ön eğitilmiş modeller transfer öğrenme, sınıflandırma ve özellik çıkarıcı olarak kullanılmaktadır. Bu çalışmada AlexNet ön eğitilmiş model yardımı ile transfer öğrenme destekli CNN, sınıflandırıcı olarak kullanılmıştır.

Yüz ifadelerinden ER gerçekleştirilmesi iki farklı temel yolla gerçekleştirilebilmektedir. Bu yollar, statik imge ile duygu tanıma ve dinamik sekanslarla duygu tanıma olarak sıralanabilir [20]. Statik görsel veriden yararlanılarak duygu tanıma [168–170]

çalışmalarında gerçekleştirilmiş ve sunulmuştur.

Bu çalışmada ilk olarak eğitim ve test aşamalarında kullanılmak üzere imge arama motorları yardımı ile yeni bir duygu tanıma veri seti oluşturulmuştur. Örnek sayısının derin öğrenme için yeterli olmadığı varsayıldığından, veri ön işlemenin (data preprocessing) ve veri çoğullamanın (data augmentation, DA) sistem performansına etkisi ayrıca incelenmiştir. Her iki durumun da sistem performansını artırdığı deneysel çalışmalarla ispatlanmıştır.

Bu çalışmanın ana katkıları şu şekilde sıralanabilir:

- Yeni bir duygu tanıma veri seti oluşturulmuştur.
- Veri ön işleme ve transfer öğrenme destekli bir sınıflandırıcı önerilmiştir.

Çalışmanın bundan sonraki kısmında ilk olarak bu çalışmada kullanılmak üzere oluşturulan duygu tanıma veri seti Bölüm 5.1.2 ile sunulmuştur. Bölüm 5.1.3 ile kullanılan metodolojiden bahsedilmiş olup Bölüm 5.1.4 ile deneysel çalışmalar ve sonuçlar detaylı bir şekilde açıklanmıştır. Son olarak ilgili çalışmanın değerlendirilmesi Bölüm 5.1.5 ile verilmiştir.

5.1.2. Kullanılan Veri Seti

Bu çalışmada önerilen yöntem internet arama motoru yardımı ile oluşturulan yeni bir veri seti ile analiz edilecektir. İlgili veri setine ait detaylı bilgiler alt bölümde sunulmuştur.

5.1.2.1. Duygu Tanıma Veri seti

Görsel arama motorları kullanılarak duygu tanıma üzerine yeni bir veri seti geliştirilmiştir. Bu veri seti, temel duygu tanıma sınıflarından “korkmuş”, “mutlu”, “sakin”, “sinirli” ve “uzgun” sınıflarını içermektedir. Veri setindeki her bir sınıf, Tablo 5.1 ile de görüleceği üzere 70 adet imge eğitim verisi 30 adet imge test verisi olacak şekilde 100 imgeden oluşmaktadır.

Tablo 5.1. Duygu tanıma veri seti.

Sınıf	Eğitim Verisi Sayısı	Test Verisi Sayısı
korkmus	70	30
mutlu	70	30
saskin	70	30
sinirli	70	30
uzgun	70	30

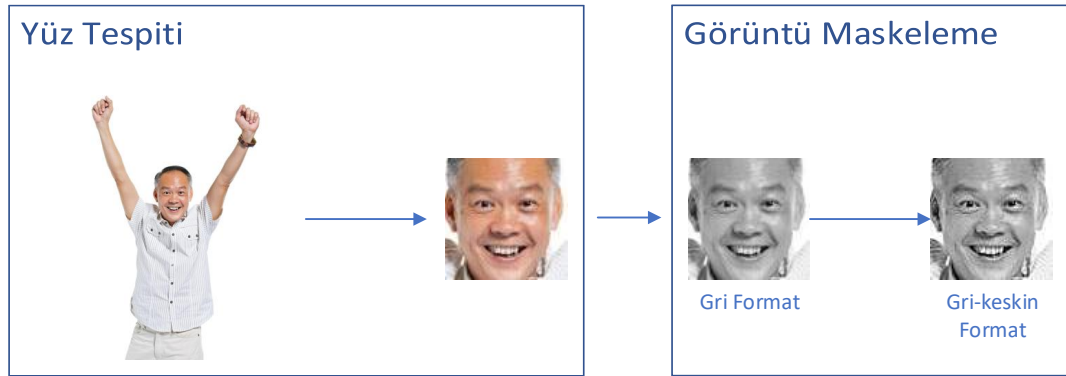
5.1.3. Metodoloji

Bu çalışmada, geliştirilen duygu tanıma veri setinin sınıflandırılması üzerine bir çalışma gerçekleştirilmiştir. Kullanılan yöntemin temeli, transfer öğrenme destekli CNN mimarisinden oluşmaktadır. Transfer öğrenme ve CNN hakkında detaylı bilgiler sırasıyla Bölüm 2.1.1.1 ve 2.1.1 ile ayrıntılı olarak verilmiştir.

Transfer öğrenme destekli CNN mimarisinin başarımını artırmak için veri ön işleme ve veri çoğullama adımları ayrıca kullanılmıştır.

5.1.3.1. Veri Ön İşleme

Veri ön işleme, verilerin farklı süreçlerden geçerek işlenmesinin sağlandığı aşamadır. Orijinal veri setine (OV) her bir ön adım uygulandıktan sonra güncellenmiş veri seti saklanmış ve bu veri seti üzerinde sınıflandırma çalışmaları gerçekleştirilmiştir. Bu çalışmada yüz tespiti (YT) ve maskeleme (M) ön işlem adımları olarak kullanılmıştır. Matlab kütüphanesinden [171] faydalanılarak Viola-Jones yöntemi [21] kullanılmış ve OV'deki her bir imgeye uygulanarak yüz tespiti gerçekleştirilmiştir. Bir diğer ön işleme adımı olarak maskeleme işlemi imgelere uygulanmıştır. Bu aşamada her bir imge öncelikle gri formata çevrilmiş daha sonra keskinleştirme işlemi uygulanmıştır [166]. Uygulanan ön işlem adımları örnek bir imge [172] üzerinden Şekil 5.1 ile sunulmuştur [166].



Şekil 5.1. Veri ön işleme adımları.

Veri ön işleme adımlarına ek olarak, veri setindeki örnek sayısının az olmasından dolayı veri çoğullama işlemi uygulanmış ve sınıflandırma çalışmaları gerçekleştirilmiştir.

5.1.4. Deneysel Çalışmalar

Bu çalışmada AlexNet destekli CNN kullanılarak sınıflandırma işlemi gerçekleştirilmiştir. İlgili modele ait eğitim parametreleri için seçilen değerler Tablo 5.2 ile sunulmuştur. Bu parametrelerin seçimi yapılırken ilgili parametrelerin varsayılan değerlerine yakın değerler seçilerek deneme yanılma yoluyla gerçekleştirilmiştir.

Tablo 5.2. Transfer öğrenme destekli CNN için kullanılan parametreler ve değerleri.

Parametre Adı	Parametre Değeri
Maksimum epok	20
Minibatch boyutu	32
İlk öğrenme oranı	0,001
Momentum	0,9
ℓ_2 düzenleme oranı	0,0001

Bu çalışmada gerçekleştirilen sınıflandırma yöntemlerine ait sonuçlar Tablo 5.3 ile sunulmuştur [166]. Bu tabloda (CNN+OV) orijinal veri ve CNN'i, (CNN+YT) yüz tespiti ve CNN'i, (CNN+YT+M) yüz tespiti, maskeleye ve CNN'i ve son olarak (CNN+YT+M+DA) yüz tespiti, maskeleye, veri çoğullama ve CNN'i ifade etmektedir. Bu yöntemlere ait yapılan 10'ar koşma için ortalama ve en iyi doğruluk değerleri

verilmiştir. Tüm metotlar içerisinde (CNN+YT+M+DA) metodu hem ortalama hem de en iyi doğruluğa göre en yüksek başarıma sahiptir.

Tablo 5.3. Duygu tanıma veri seti için yöntemlerin ortalama ve en iyi başarımlarını.

Metot	Ortalama Doğruluk (%)	En İyi Doğruluk (%)
CNN+OV	44,60	51,33
CNN+YT	67,93	72,67
CNN+YT+M	68,80	72,67
CNN+YT+M+DA	70,00	74,00

İlgili yöntemlere ait tüm koşmalar içerisindeki en iyi doğruluklara ait karmaşıklık matrisleri sırasıyla Şekil 5.2, 5.3, 5.4 ve 5.5 ile sunulmuştur.

Doğru sınıf	korkmus	18		7	4	1	60.0%	40.0%	
	mutlu	1	17	6	2	4	56.7%	43.3%	
	saskin	2	3	15	9	1	50.0%	50.0%	
	sinirli	3	2	7	14	4	46.7%	53.3%	
	uzgun	7	3	3	4	13	43.3%	56.7%	
		58.1%	68.0%	39.5%	42.4%	56.5%			
		41.9%	32.0%	60.5%	57.6%	43.5%			
		korkmus	mutlu	saskin	sinirli	uzgun			
		Tahmin edilen sınıf							

Şekil 5.2. CNN+OV metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.

Doğru sınıf	korkmus	19		8	2	1	63.3%	36.7%
	mutlu	2	28				93.3%	6.7%
	saskin	3	2	20	2	3	66.7%	33.3%
	sinirli	3	1	2	22	2	73.3%	26.7%
	uzgun	2	3	3	2	20	66.7%	33.3%
		65.5%	82.4%	60.6%	78.6%	76.9%		
		34.5%	17.6%	39.4%	21.4%	23.1%		
		korkmus	mutlu	saskin	sinirli	uzgun	Tahmin edilen sınıf	

Şekil 5.3. CNN+YT metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.

Doğru sınıf	korkmus	15		11	3	1	50.0%	50.0%
	mutlu		29		1		96.7%	3.3%
	saskin	2	2	21	2	3	70.0%	30.0%
	sinirli	4			23	3	76.7%	23.3%
	uzgun	2	2	3	2	21	70.0%	30.0%
		65.2%	87.9%	60.0%	74.2%	75.0%		
		34.8%	12.1%	40.0%	25.8%	25.0%		
		korkmus	mutlu	saskin	sinirli	uzgun	Tahmin edilen sınıf	

Şekil 5.4. CNN+YT+M metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.

Doğru sınıf	korkmus	20		6	2	2	66.7%	33.3%
	mutlu		29		1		96.7%	3.3%
	saskin	5	2	14	3	6	46.7%	53.3%
	sinirli	4		2	23	1	76.7%	23.3%
	uzgun	2	2		1	25	83.3%	16.7%
		64.5%	87.9%	63.6%	76.7%	73.5%		
		35.5%	12.1%	36.4%	23.3%	26.5%		
		korkmus	mutlu	saskin	sinirli	uzgun		
		Tahmin edilen sınıf						

Şekil 5.5. CNN+YT+M+DA metodu ile elde edilen en iyi sonucun karmaşıklık matrisi.

5.1.5. Değerlendirme

Statik imge tabanlı duygu tanıma veri setinin geliştirilmesi bu çalışmada ilk aşama olarak uygulanmıştır. Oluşturulan veri setini sınıflandırma işlemi, literatürde sıklıkla kullanılan CNN ile gerçekleştirilmiştir. Yeni bir CNN modeli geliştirmek yerine AlexNet ön eğitilmiş model kullanılarak transfer öğrenme destekli CNN mimarisi geliştirilmiştir. Bir sınıflandırma çalışmasında başarı oranını etkileyen faktörlerden birisi sınıflandırılacak verilerin işlenmesidir. Bundan dolayı oluşturulan veri seti farklı ön işlemlerden geçirilmiş ve bütün işlemler için ayrı ayrı sınıflandırma gerçekleştirilmiştir. Deneysel çalışmalar incelendiğinde ön işlem adımlarının uygulandığı yöntemlerle daha başarılı sonuçlar elde edildiği görülmüştür.

Transfer öğrenme için farklı ön eğitilmiş modellerin kullanılması, veri ön işleme sürecinde farklı metotların uygulanması ve veri çoğullamanın farklı yöntemlerle yapılması gelecek çalışmalar için planlanmaktadır.

5.2. Transfer Öğrenimine ve Hiperparametre Optimizasyonuna Dayalı Evrişimli Sinir Ağları Kullanarak Statik Yüz İfadesi Tanıma

5.2.1. Giriş

Bir durumun insanın iç dünyasındaki etkisi olarak ifade edilen duygu, bu etkinin bir veya daha fazla yüz kas hareketi ile dışa vurumudur. Kızgınlık, iğrenme, korku, mutluluk, üzüntü ve şaşkınlık literatürde bilinen 6 temel duygu ifadesidir [20]. İnsan bilgisayar etkileşimi (human computer interaction, HCI) sisteminin alt kategorisi olan yüz ifadesi tanıma (facial expression recognition, FER); pazarlama, psikoloji, ticaret, askeri ve güvenlik gibi alanlarda kullanılabilir.

FER çalışmaları video tabanlı görüntülerle gerçekleştirilmesinin yanında statik imge tabanlı da gerçekleştirilebilmektedir. Bu çalışma için Ek-A belgesinde gösterilen etik kurul izni ile geliştirilen Erciyes Üniversitesi yüz ifadesi tanıma (Erciyes University facial expression recognition, ERUFER) ve literatürde sıklıkla kullanılan statik imge tabanlı veri setleri Tablo 5.4 ile sunulmuştur.

Tablo 5.4. Literatürde sıklıkla kullanılan FER veri setleri ve özellikleri.

Veri Seti	Örnek Sayısı	Katılımcı Sayısı	Koşul	İfadeler
JAFFE	213	10	Lab	6 temel duygu artı doğal
TFD	112,234	N/A	Lab	6 temel duygu artı doğal
FER-2013	35,887	N/A	Web	6 temel duygu artı doğal
SFEW	1,766	N/A	Video	6 temel duygu artı doğal
RaFD	1,608	67	Lab	6 temel duygu artı doğal ve küçümseyen
KDEF	4,900	70	Lab	6 temel duygu artı doğal
ERUFER	9,005	110	Lab	6 temel duygu artı doğal, küçümseyen, endişeli ve heyecanlı

Bu çalışmadaki ilk amaç yeni bir statik imge tabanlı duygu tanıma veri seti geliştirilmesidir. Bu amaç doğrultusunda ERUFER veri seti oluşturulmuştur. Bu veri seti 10 sınıf üzerinden 9005 örnek içermektedir. 6 temel duygu tanımının yanında doğal, küçümseyen, endişeli ve heyecanlı sınıfları veri setinde yer almaktadır. Birini dışarıda bırak çapraz doğrulama (leave one out cross validation, LOOCV), 10 kat çaprazlama,

5 kat çaprazlama ve %70 eğitim %30 test ayrımı gibi test tipleri ERUFER veri setinin performans analizinde kullanılabilir.

Veri ön işleme, transfer öğrenme destekli CNN ve PSO destekli hiperparametre optimizasyonu yapılarının bir araya getirilmesiyle etkili bir sınıflandırma işlemi gerçekleştirmek bu çalışmanın bir diğer temel amacıdır. Veri ön işleme adımında öncelikle yüz tespiti yapıp daha sonra RGB formatındaki imgenin gri formata dönüştürülmesi sağlanır. Daha sonraki ön işleme adımlarında sırasıyla gürültü giderme, imge keskinleştirme ve veri çoğullama kullanılır.

Derin öğrenme destekli yöntemler sınıflandırma, bölütleme ve kümeleme gibi amaçlarla literatürde sıklıkla kullanılmaktadır [81, 129, 143, 173, 174]. Bu çalışmada veri ön işleme adımından sonra literatürde sıklıkla tercih edilen ve bir derin öğrenme metodu olan CNN, ERUFER veri setini sınıflandırma aşamasında kullanılır. Daha etkili ve hızlı bir CNN mimarisi kurulum yöntemi olan transfer öğrenme destekli CNN bu çalışmada tercih edilmiştir.

Sınıflandırma performansını etkileyen bir diğer önemli faktör CNN metodunun eğitim parametrelerinin seçimidir. Bu seçimin kullanıcı deneyimine bırakılması hem zamansal kayba neden olur hem de optimum parametre seçimi çok zordur. Bu nedenle CNN metodunun hiperparametreleri, global arama yeteneği yüksek PSO algoritması ile gerçekleştirilir.

Bu çalışmada önerilen yöntem ERUFER veri seti ile test edilmiştir. 10 kat çaprazlama test tipi kullanılarak yapılan simülasyonlar sonucunda %92,56 başarı oranına kadar çıktığı tespit edilmiştir. Yöntemin performansını test etmek için yapılan bir başka deneysel çalışmada, JAFFE veri seti kullanılmış 10 kat çaprazlama ve 7 sınıf bazlı senaryo üzerinden gerçekleştirilen simülasyonlar sonucunda %99,53 e kadar başarı oranı elde edildiği görülmüştür. Bu sonuç ile literatürde JAFFE veri seti üzerinde yapılan çalışmalar [175–179] doğruluk oranı metriğine göre geride bırakılmıştır.

Bu çalışmanın ana katkıları şu şekilde sıralanabilir:

- Yeni bir statik imge tabanlı yüz tanıma veri seti, ERUFER, geliştirilmiştir.

- Statik yüz ifadeleri, veri ön işleme ve hiperparametre ayarlayıcı transfer öğrenme tabanlı CNN ile sınıflandırılmıştır.
- Yapılan literatür taramasına göre, 10 kat çaprazlama ve 7 sınıf bazlı senaryoya göre JAFFE veri seti üzerinden en iyi başarı oranı elde edilmiştir.
- 10 kat çaprazlama ve 10 sınıf senaryosuna göre ERUFER veri seti üzerinde %92,56 başarı oranına kadar ulaşılmıştır.

Çalışmanın bundan sonraki kısmı şu şekilde organize edilmiştir: Bölüm 5.2.2 ile bu çalışma için geliştirilen ERUFER veri seti ve performans analizinde ERUFER dışında kullanılan JAFFE veri seti hakkında detaylı bilgi verilecektir. Bölüm 5.2.3 ile önerilen yöntem detaylı olarak sunulacak olup deneysel çalışmalar ve sonuçlar Bölüm 5.2.4 ile sunulacaktır. Son olarak, Bölüm 5.2.5 ile yapılan çalışmanın değerlendirilmesinden ve gelecek çalışmalardan bahsedilecektir.

5.2.2. Kullanılan Veri Seti

Bu çalışmada önerilen yöntem öncelikle bu tez çalışması sürecinde geliştirilen ERUFER veri seti ile analiz edilecektir. Daha sonra önerilen yöntemin gürbüzlüğünü analiz etmek için genel kullanıma açık JAFFE veri seti tercih edilecektir. İlgili veri setine ait detaylı bilgiler alt bölümde sunulmuştur.

5.2.2.1. ERUFER Veri Seti

Statik görüntüler kullanarak yüz ifadesi tanıma için yeni bir veri seti oluşturulması bu tez çalışmasının amaçlarından birisidir. Toplam 110 gönüllü ile gerçekleştirilen veri toplama işlemi, farklı pozlarda 10 farklı ifade toplanarak sağlanmıştır (Şekil 5.6 ile gösterildiği gibi). Bu veri setinde, toplamda 9,005 statik imge bulunmakta olup her bir sınıf için toplam örnek sayısı Tablo 5.5 ile sunulmuştur.



Şekil 5.6. Mutlu ifadesi için bir gönüllünün farklı pozları.

Tablo 5.5. ERUFER veri setindeki sınıflara ait örnek sayıları.

İfadeler	Örnek Sayısı	İfadeler	Örnek Sayısı
angry	906	contempt	914
disgust	726	excited	1003
fear	729	happy	1022
neutral	981	sad	915
surprised	908	worry	901

Katılımcılar farklı yaş gruplarından seçilmiş olup en genç katılımcı 18 yaşında iken, en yaşlı katılımcı 80 yaşındadır. Böylece veri seti, farklı yaş grupları için geliştirilecek çalışmalarda ayrıca kullanılabilir. Veri setindeki bazı katılımcıların görüntüleri, Şekil 5.7 ile gösterilmiştir. ERUFER veri seti, LOOCV, 10 kat çaprazlama, %70 eğitim - %30 test vb. test tipleri için kullanımı uygundur.



Şekil 5.7. ERUFER veri seti için tüm katılımcıları içeren örnek resimler.

5.2.2.2. JAFFE Veri Seti

213 görüntü içeren JAFFE veri seti [69] laboratuvar ortamında örnekler alınarak geliştirilmiştir. Bu veri setinin oluşumunda 10 Japon kadın yer almaktadır. Altı temel ifade için ifade başına 2 veya 3 imge ve ayrıca bir nötr ifadeyle görüntü alınmıştır. Veri setinde her katılımcı için birkaç resim var olduğundan, LOOCV ve 10 kat çapraz doğrulama teknikleri genellikle bu veri seti için test türü olarak kullanılmaktadır [20]. Bu veri setinde yer alan katılımcılar Şekil 5.8 ile gösterilmiştir.



Şekil 5.8. JAFFE veri seti için tüm katılımcıları içeren örnek resimler.

5.2.3. Metodoloji

Bu çalışmada hiperparametre destekli CNN kullanılarak ön işlemden geçirilmiş (Şekil 5.9 ile gösterildiği gibi) veri setleri (Şekil 5.10 üzerinde tanımlanan Veri Seti) üzerinde sınıflandırma gerçekleştirilmiştir. Önerilen yöntemin ilk aşaması olarak kullanılacak olan veri setleri ön işlemlerden geçirilir. Yüz tespitinde dlib kütüphanesinden CNN tabanlı bir fonksiyon [22] kullanılır. Daha sonra RGB formatındaki imge gri formata çevrilip, önce imgeden gürültü giderilir daha sonra keskinleştirme işlemi uygulanır. Veri ön işlemenin son adımı olarak veri çoğullama operasyonu gerçekleştirilir. Ön işleme adımından sonra Veri Seti Train Seti ve Test Seti olarak alt veri setlerine bölünür. Parametre ayarlama, özellik seçimi gibi işlemlerde kullanılmak üzere tercih edilen geliştirme seti (Dev Seti) [162], bu çalışma için parametre ayarlamak üzere tercih edilmiş ve Train Seti kullanılarak üretilmiştir. Dev Seti , Train Seti verilerinin tamamı veya bir kısmı kullanılarak oluşturulmaktadır. Bu çalışma için JAFFE veri setinde Train Seti verilerinin tamamı, ERUFER veri setinde ise verilerin bir kısmı kullanılarak Dev Seti oluşturulmuştur. Dev Seti TrainDev Seti ve TestDev Seti olarak alt veri setlerine bölünür. Önerilen yöntemin bir diğer temel adımı olan hiperparametre optimizasyonunun gerçekleştirilmesi için öncelikle kullanılacak olan parçacık sürü optimizasyon (particle swarm optimization, PSO) algoritmasının kontrol parametreleri belirlenir. Daha sonra, PSO algoritmasının uygunluk fonksiyonu

tanımlanır. Uygunluk fonksiyonunun çıktısı, `TestDev Seti` veri seti için tanıma hatası değeridir. PSO algoritmasının durdurma kriteri sağlanana kadar algoritmanın fazları tekrar tekrar çalıştırılır. Durdurma kriteri sağlandıktan sonra, hafızada tutulan en iyi parametre seti çıktı olarak verilir. Bir sonraki adımda, optimal hiperparametreler ve `Train Seti` kullanılarak nihai CNN eğitimi gerçekleştirilir ve eğitilmiş model çıktısı elde edilir. Son olarak, `Test Seti` eğitilmiş model ile test edilir ve elde edilen doğruluk oranı ile yöntemin performans analizi yapılır. Önerilen yönteme ait temel adımlar Algoritma 5.1 ile sunulmuş ve yöntemin mimarisi Şekil 5.10 ile verilmiştir.

Önerilen yöntemde yer alan metotlardan CNN Bölüm 2.1.1 ile, transfer öğrenme Bölüm 2.1.1.1 ile ve hiperparametre ayarlayıcı olarak kullanılan PSO algoritması Bölüm 2.2.4 ile detaylı bir şekilde açıklanmıştır. Bu yöntemlerin dışında önerilen sistemde yer alan veri ön işleme yöntemleri sonraki alt bölümde detaylı olarak açıklanmaktadır.

5.2.3.1. Veri Ön İşleme

Veri ön işleme, başarı oranını artırmak için görüntü ve sinyal işlemeye dayalı derin öğrenme çalışmalarında sıklıkla kullanılan bir tekniktir. Yüz tespiti, RGB görüntünün gri tonlamaya dönüştürülmesi, 2-D adaptif gürültü giderici filtreleme, görüntü netleştirme ve veri çoğullama veri ön işleme yöntemleridir.

Tüm imgeyi FER için incelemek yerine, doğruluğu artırmak için yalnızca yüz kısmının işlenmesi daha etkilidir. Bu nedenle, CNN tabanlı yüz tespiti, `dlib` kütüphanesi [22] kullanılarak gerçekleştirilir.

İmgeyi RGB renk formatından gri tonlamaya dönüştürmek için, Eşitlik 5.1 [171] ile gösterildiği gibi ağırlıklı bir R, G ve B bantları toplamı kullanılır.

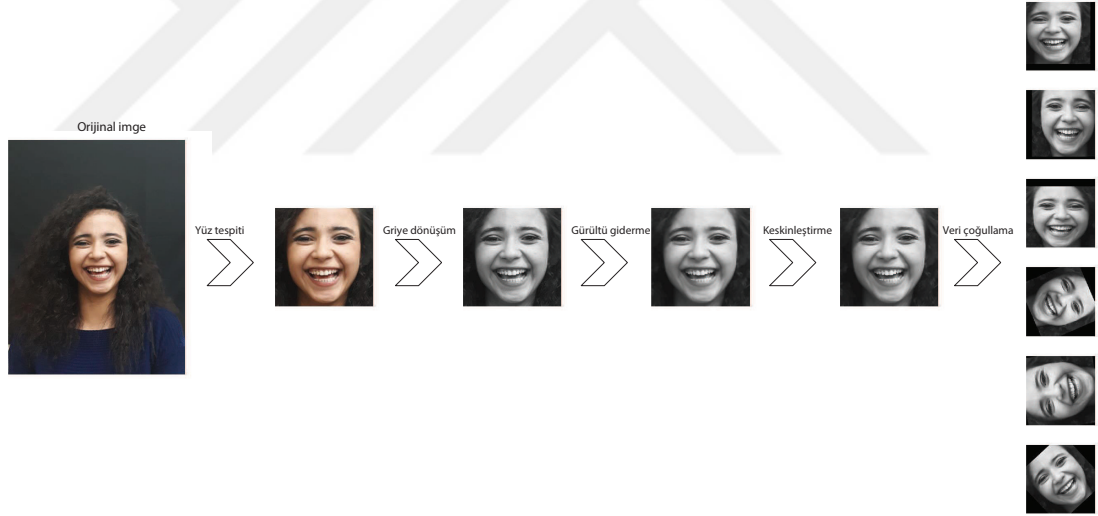
$$RGBtoGray = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B \quad (5.1)$$

Literatürde görüntülerden gürültüyü gidermek için pek çok yöntem önerilmiştir [180]. Bu yöntemlerden birisi olan Wiener filtresi, gürültü bileşenlerinin etkilerini azaltmak için kullanılır [138]. Daha sonra, maskeleye yöntemi [138] kullanılarak görüntü netleştirilir.

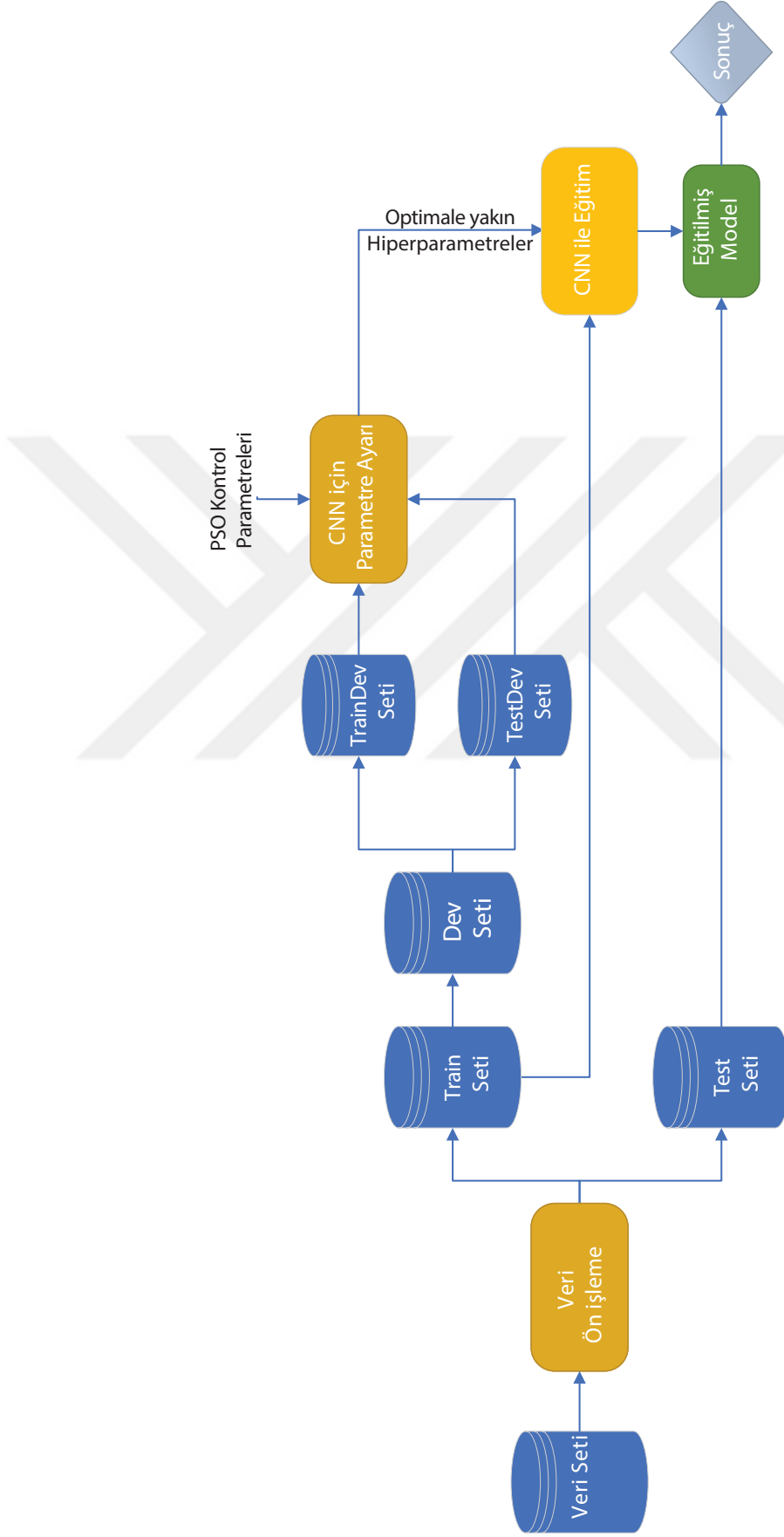
- 1: Veri Seti verileri için veri ön işleme adımlarını uygula
- 2: Veri Seti 'ni Train Seti ve Test Seti olarak böl
- 3: Train Seti veri setinden Dev Seti veri setini oluştur
- 4: Dev Seti 'ni TrainDev Seti ve TestDev Seti olarak böl
- 5: PSO algoritmasının kontrol parametrelerini tanımla
- 6: Uygunluk fonksiyonunu tanımla
- 7: PSO algoritmasının durdurma kriteri sağlanana kadar
- 8: PSO algoritmasının fazlarını koş
- 9: Optimale yakın hiperparametre değerlerini çıktı olarak al
- 10: Train Set ve ayarlanmış hiperparametrelerle ölçekli eşlenik gradyan (scaled conjugate gradient, SCG) algoritmasını kullanarak CNN'i eğit
- 11: Eğitilmiş modeli çıktı olarak al
- 12: Eğitilmiş modeli kullanarak Test Set üzerinden başarı oranını hesapla

Algoritma 5.1: Önerilen yöntemin temel adımları.

Tüm bu ön işlemlerin sonrasında, rastgele döndürme, yansıma, ölçeklendirme ve çevirme teknikleri kullanılarak veri çoğullama sağlanır.



Şekil 5.9. Bir örnek üzerinde veri ön işleme uygulamaları.



Şekil 5.10. Önerilen yöntemin mimarisi.

5.2.4. Deneysel Çalışmalar

Deneysel çalışmalar; deneysel kurulum ve deneysel sonuçlar alt başlıklarında incelenecektir.

5.2.4.1. Deneysel Kurulum

Deneysel kurulum; donanımsal kurulum ve parametre kurulumu olmak üzere üç alt başlıkta sunulmuştur.

5.2.4.1.1. Donanımsal Kurulum

ERUFER ve JAFFE veri setleri görsel verilerden oluşmaktadır. Özellikle ERUFER veri setindeki örnek sayısının oldukça yüksek olması ve görüntü işleme yöntemlerinin kullanılacağından dolayı deneysel çalışmalarda 11 GB Ram destekli NVIDIA GTX 1080 Ti Turbo donanımı kullanılmıştır.

5.2.4.1.2. Parametre Kurulumu

Önerilen yöntemde, parametre ayarlaması için PSO algoritması kullanılmıştır. Bu algoritmaya ait kontrol parametreleri Tablo 5.6 ile verilmiştir.

Tablo 5.6. PSO algoritmasının kontrol parametreleri.

Parametre	Açıklama	Seçilen Değer
<i>SwarmSize</i>	Parçacık sayısı	10
<i>MaxIter</i>	İterasyon sayısı	10
c_1	Öğrenme faktörü 1	1,49
c_2	Öğrenme faktörü 2	1,49
w	Hareketsizlik ağırlığı	0,7

5.2.4.2. Deneysel Sonuçlar

Önerilen metodoloji JAFFE ve ERUFER veri setleri üzerinde test edilmiş ve deneysel

sonuçlar sunulmuştur.

5.2.4.2.1. JAFFE Veri Seti için Deneysel Sonuçlar

JAFFE yaygın olarak kullanılan bir duygu tanıma veri setidir. Statik görüntülerden oluşan bu veri seti laboratuvar ortamında geliştirilmiştir. 6 temel ve nötr, veri setinin ifadeleridir.

Önerilen yaklaşım JAFFE veri setine uygulanmıştır. PSO bir sezgisel algoritma olduğundan dolayı, önerilen yöntem 30 kez koşturulmuştur. Tüm koşular için en iyi ve ortalama başarı oranları Tablo 5.7 ile verilmiştir. %99,53'e kadar başarı oranı elde edilmiş olup bu oran, yaptığımız literatür taramasına göre en iyi başarı oranıdır.

Tablo 5.7. JAFFE veri setinde 30 koşma için başarı oranları.

Koşma	Başarı(%)	Koşma	Başarı(%)
1	99,53	16	98,59
2	99,06	17	99,06
3	99,06	18	99,53
4	98,59	19	99,06
5	99,53	20	99,53
6	98,59	21	99,53
7	99,53	22	99,06
8	99,53	23	99,53
9	98,59	24	99,06
10	99,53	25	99,53
11	98,59	26	99,06
12	99,06	27	98,59
13	99,53	28	99,53
14	98,59	29	98,59
15	99,53	30	99,53
En iyi başarı		99,53	
Ortalama başarı		99,15	

Önerilen yöntemin ve mevcut çalışmaların başarı oranları Tablo 5.8 ile verilmiştir. Sonuçlara göre, önerilen yöntem, 10 kat çaprazlama ve 7 sınıflı senaryoda en iyi sonuca ulaşmıştır.

Tablo 5.8. JAFFE veri seti üzerindeki önerilen yöntem ile mevcut çalışmalar arasındaki karşılaştırma.

Metot	Tanıma Oranı(%)
Poursaberi ve ekibi [175]	95,04
Zavaschi ve ekibi [176]	96,20
Gogic ve ekibi [177]	98,10
Moeini ve ekibi [178]	97,30
Luo ve ekibi [179]	92,10
Önerilen	99,53

JAFFE veri setinde önerilen yöntem için karmaşıklık matrisi Şekil 5.11 ile sunulmuştur. Şekilden görülebileceği gibi, tahminler arasında yalnızca bir hata vardır.

Doğru sınıf	Angry	30						100.0%	
	Disgust		29					100.0%	
	Fear			32				100.0%	
	Happy				31			100.0%	
	Neutral					30		100.0%	
	Sad						1	30	96.8% 3.2%
	Surprise							30	100.0%
		100.0%	100.0%	100.0%	96.9%	100.0%	100.0%	100.0%	
					3.1%				
		Angry	Disgust	Fear	Happy	Neutral	Sad	Surprise	
		Tahmin edilen sınıf							

Şekil 5.11. JAFFE veri seti için karmaşıklık matrisi.

5.2.4.2.2. ERUFER Veri Seti için Deneysel Sonuçlar

Bu çalışmada geliştirilen ERUFER veri seti çok sayıda örnek içermektedir. Ayrıca, veri setinde yer alan katılımcılar 18 ile 80 yaş arasında seçilmiş olup farklı tipte uygulamalar için ideal bir yapıdadır. Bu veri setinin bir başka özelliği ise 10 sınıf barındırmasıdır. Tüm bu zorlu durumlara rağmen önerilen yöntem ile ERUFER veri seti için %92,56 başarı oranına kadar ulaşılmıştır. ERUFER veri seti için gerçekleştirilen 30 koşmanın

başarı oranları Tablo 5.9 ile sunulmuştur. ERUFER veri setinde önerilen yöntem için karmaşıklık matrisi Şekil 5.12 ile gösterildiği gibidir.

Tablo 5.9. ERUFER veri setinde 30 koşma için başarı oranları.

Koşma	Başarı(%)	Koşma	Başarı(%)
1	90,57	16	89,52
2	88,84	17	90,29
3	90,33	18	89,42
4	91,08	19	91,28
5	90,17	20	90,74
6	90,15	21	90,73
7	89,88	22	90,24
8	90,42	23	89,51
9	89,13	24	89,95
10	90,13	25	90,07
11	92,56	26	90,21
12	89,79	27	89,85
13	89,83	28	90,55
14	89,98	29	90,42
15	90,13	30	88,67
En iyi başarı			92,56
Ortalama başarı			90,15

Doğru sınıf	Angry	865	12	7	2	1		2	11		5	95.6%	4.4%
	Contempt	17	834	13	5	3	6	9	25	1	2	91.1%	8.9%
	Disgust	15	26	663	2	4	3		7		5	91.4%	8.6%
	Excited		10	4	841	14	59			68	7	83.8%	16.2%
	Fear	3		3	8	661		5	1	32	16	90.7%	9.3%
	Happy			5	17		992	3	1	3	1	97.1%	2.9%
	Neutral		1		1		2	961	12	1	3	98.0%	2.0%
	Sad	12	16	3				15	862		8	94.1%	5.9%
	Surprise		2		56	28	2	2	1	809	8	89.1%	10.9%
	Worry	5	13	1	4	4	2	4	19	2	847	94.0%	6.0%
			94.3%	91.2%	94.8%	89.9%	92.4%	93.1%	96.0%	91.8%	88.3%	93.9%	
		5.7%	8.8%	5.2%	10.1%	7.6%	6.9%	4.0%	8.2%	11.7%	6.1%		
		Angry	Contempt	Disgust	Excited	Fear	Happy	Neutral	Sad	Surprise	Worry		
		Tahmin edilen sınıf											

Şekil 5.12. ERUFER veri seti için karmaşıklık matrisi.

5.2.5. Değerlendirme

FER, askeri, ticari ve sosyal çalışmalar için sıkça kullanılan bir insan bilgisayar etkileşimi sistemi haline gelmiştir. İfadeler, sensör verilerinden ve görsel verilerden sınıflandırılabilir. Bu çalışmada, FER için entegre bir yöntem geliştirilmiştir. Veri ön işleme ve transfer öğrenme destekli CNN'nin hiperparametre ayarı bu yöntemin temel parçalarıdır. Veri ön işleme bölümü için, görüntü ilk olarak yüz tespiti yöntemi ile tespit edilmiştir. Ardından, imgelerin RGB formattan gri tonlama formatına dönüştürülmesi, gürültünün giderilmesi ve görüntü netleştirme adımları gerçekleştirilir. Veri çoğullama, veri ön işleme bölümünün son adımıdır. Bir sonraki aşama olarak transfer öğrenme tabanlı CNN'in hiperparametre ayarı gerçekleştirilmiştir. Bu aşamada ilk olarak ResNet18 ön eğitilmiş model kullanılarak transfer öğrenimi gerçekleştirilir. Daha sonra, CNN'nin eğitim parametrelerinden maksimum epok, minibatch boyutu, ilk öğrenme oranı, ℓ_2 düzenleştirme oranı, karıştırma (shuffle) tipi ve momentum olan hiperparametreler seçilir. Sistem performansını doğrudan etkileyen bu parametrelerin seçimi güçlü bir global arama yeteneği olan PSO algoritması ile gerçekleştirilmiştir. Önerilen yöntem literatürde yaygın olarak kullanılan JAFFE veri seti üzerinde uygulanmıştır. 10 kat çaprazlama ve 7 sınıf senaryosuna göre literatürde bilinen en iyi başarı oranı elde edilmiştir.

Bu çalışmada ayrıca ERUFER adında yeni bir statik imge tabanlı veri seti oluşturulmuştur. Bu veri seti, 110 katılımcıdan farklı sayıda pozlarla 9005 örnek alınarak geliştirilmiştir. Önerilen sınıflandırma metodu ERUFER veri seti için de uygulanmış ve başarılı sonuçlar elde edilmiştir.

Sonraki çalışmalar için önerilen sınıflandırma metodu farklı veri setlerinde test edilerek sistemin gürbüzlüğü incelenebilir. Bu çalışmada, ResNet18 ön eğitilmiş model, transfer öğrenimi için kullanılmıştır. Farklı ön eğitilmiş modeller kullanılarak karşılaştırmalar yapılabilir. Başka bir çalışma olarak, hiperparametre ayarında kullanılan PSO algoritması yerine farklı algoritmalar kullanılarak performans karşılaştırması yapılabilir. Son bir öneri olarak, ERUFER veri seti, diğer makine öğrenmeleri ve derin öğrenme yöntemleri ile sınıflandırma için kullanılabilir.

6. BÖLÜM

DERİN ÖĞRENME İLE DUDAK OKUMA

İnsan hareketi tanımanın (human action recognition, HAR) bu tez çalışması için incelenen son alt kategorisi dudak okumadır (lip reading, LR). Tezin bu bölümünde derin öğrenme yöntemi kullanılarak dudak okuma üzerine bir çalışma uygulanmış ve detaylı bir şekilde sunulmuştur.

6.1. Öneğitimli Model Destekli - Desteksiz Evrişimli Sinir Ağları Kullanarak Dudak Okuma

6.1.1. Giriş

HAR, insan bilgisayar etkileşiminin (human computer interaction, HCI) bir alt kategorisi olarak ele alınabilir [181]. Bir HAR uygulaması olan dudak okuma üzerine literatürde çeşitli çalışmalar yapılmıştır [89, 91, 93, 99, 101].

Dudak okumak için ses ve görüntü destekli özellikler kullanılabilir. Özellikle, resim destekli öznitelikler içeren veriler kullanıldığı uygulamalarda daha yüksek başarı oranı elde edildiği görülmektedir. Dudak okumasının başarı oranı, özellik seçiminde kullanılan tekniklerle de doğrudan ilgilidir. Gizli Markov modelleri (hidden Markov models, HMM) [90, 182], destek vektör makineleri (support vector machines, SVM) [90, 183] ve k-en yakın komşuluk (k-nearest neighbour, kNN) algoritmaları [90] en sık kullanılan klasik sınıflandırma yöntemleridir. Derin öğrenme yöntemlerinin sınıflandırma problemlerinde yaygın olarak kullanılmasının ardından bir başka sınıflandırma problemi olan dudak okuma için de derin öğrenme yöntemleri kullanılmaya başlanmıştır.

Dudak okuma işlemi alfabe, kelime ve cümle seviyelerinde uygulanmaktadır [184].

Alfabe seviyesindeki dudak okuma işleminde statik imge ve zaman serisi sınıflandırma yöntemleri kullanılabilir. Diğer taraftan, LSTM, RNN vb. sınıflandırma yöntemleri kelime ve cümle düzeyinde işlemlerde tercih edilmektedir [184].

Bu çalışmada, alfabe düzeyinde yaygın olarak kullanılan genel kullanıma açık AvLetters veri setinin [24] sınıflandırılması amaçlanmıştır. Sınıflandırma yöntemi olarak ön eğitilmiş model (pretrained model) destekli evrişimli sinir ağı (convolutional neural networks, CNN) ve yeni bir CNN modeli kullanılmıştır. Tasarlanan CNN modeli; 3 evrişim katmanı, 3 maksimuma ortaklama katmanı ve 3 düzenleme katmanı içermektedir. Diğer yandan, transfer öğrenme (transfer learning) amaçlı kullanılan AlexNet ve GoogLeNet destekli CNN yapıları oluşturulmuş ve tasarlanan CNN ile kıyaslanmıştır. Veri seti büyüklüğü, derin öğrenme yöntemlerinde başarıyı etkileyen önemli faktörlerden birisidir. Dolayısıyla bu çalışmada veri çoğullama (data augmentation, DA) teknikleri kullanılarak veri seti büyütülmüştür. Arttırılmış veri seti ve orijinal veri seti ile yapılan ve CNN modellerine dayanan deneysel çalışmalar farklı “minibatch boyutu” değerleri ile gerçekleştirilmiş ve sonuçlar detaylı bir şekilde açıklanmıştır.

Bu çalışmanın ana katkıları şu şekilde sıralanabilir:

- Bilindiği kadarıyla, AlexNet kullanarak transfer öğrenme destekli CNN, AvLetters veri setinde ilk kez kullanılmıştır.
- Kullanılan bir teknikte, problem zaman serisi verilerinden statik imge tabanlı bir dudak okuma uygulamasına dönüştürülmüştür.
- Bir CNN mimarisi önerilmiş ve transfer öğrenme destekli CNN’lerle karşılaştırılmıştır.
- Bu çalışmada önerilen yöntemler, çalışma [185]’de yer alan bazı yöntemlerden daha iyi performansa sahip olmalarının yanı sıra geliştirilmesi kolay metotlardır.

Bu çalışmanın bundan sonraki kısımları için Bölüm 6.1.2 ile çalışmada kullanılacak olan AvLetters veri seti hakkında detaylı bilgi verilmiştir. Deneysel kurulum, çalışmalar ve sonuçlar Bölüm 6.1.3 ile sunulmuştur. Son olarak, Bölüm 6.1.4 ile yapılan çalışmanın değerlendirilmesinden ve gelecek çalışmalardan bahsedilmiştir.

6.1.2. AvLetters Veri seti

Hem video hem ses kayıtlarıyla oluşturulan AvLetters veri seti, 10 farklı katılımcının alfabedeki 26 harfi 3'er kez tekrar etmesiyle meydana gelmiştir [24]. Yapılacak çalışmada görsel veriler kullanılacağından dolayı veri setinin, görüntü bilgilerini içeren *mat* dosyası dikkate alınmıştır. Her bir harfin söylenişinde oluşan resim çerçevelerinin sayısal bilgileri *mat* dosyalarında tutulmaktadır. Örneğin 'R3_Kate-lips.mat' dosyası Kate kişinin R harfini 3. tekrarının çerçeve bilgilerini tutmaktadır.

Her bir katılımcının her bir harfi söyleme süresi farklılık gösterebilir bu da örnekler arasında farklı çerçeve sayısı oluşmasına neden olmaktadır. Veri seti hazırlığında ilk aşama olarak ortalama çerçeve sayısı olarak 20 belirlenmiş ve 20'nin altında olanlar için ekleme , 20'nin üzerinde olanlar içinse çerçeve çıkarım işlemi gerçekleştirilmiştir. Her bir harf 20 çerçeveden oluştuğundan sonra dudak resimleri 5*4 formatında hareketsiz tekil imgeler haline getirilir. Böylelikle zaman serisinden oluşan veri seti hareketsiz imgelerden oluşan bir yapı haline getirilir. Ön hazırlıktan geçirilmiş veri setine ait örnek bir imge Şekil 6.1 ile gösterildiği gibidir.



Şekil 6.1. Örnek bir imge (ön hazırlıktan sonra AvLetters - A harfi).

AvLetters veri setinde 10 katılımcı 26 harfi 3'er defa tekrar ederek söylemiştir. İlk 2 telaffuz eğitim, son telaffuz ise test için kaydedilecek şekilde "TrainData" ve "TestData" klasörleri oluşturulmuştur.

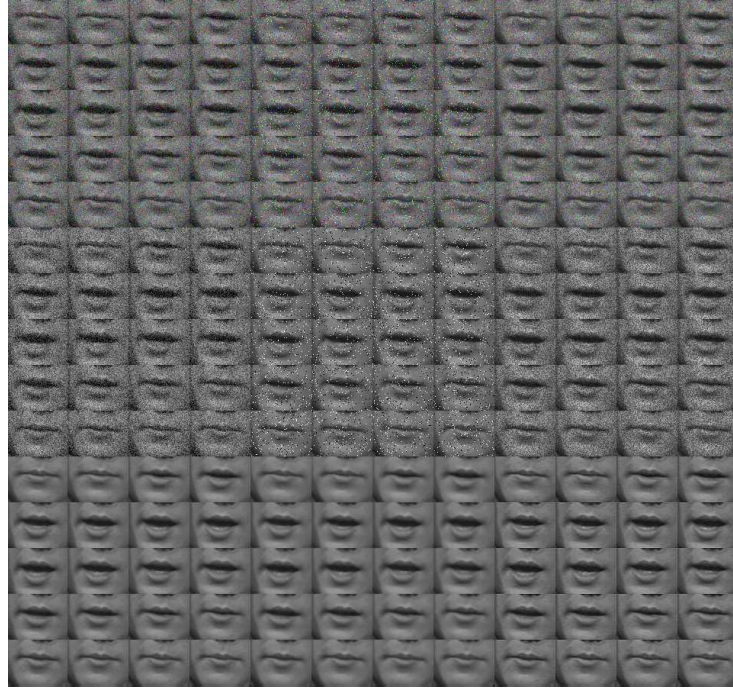
6.1.3. Deneysel Çalışmalar

Bu çalışmada AvLetters veri seti kullanılarak dudak okuma işlemi gerçekleştirilmiştir. AvLetters veri seti üzerinde yapılan çalışmada iki farklı yöntem uygulanmış olup bu yöntemlerden ilki önerilen CNN modelinin oluşturulmasıdır. Bir diğer yöntem ise kullanıcı tabanlı bir CNN yerine, AlexNet ve GoogLeNet ön eğitilmiş modellerinden transfer öğrenme yaparak performans karşılaştırması yapmaktır. Sınıflandırma başarısını yükseltmek için veri hazırlama adımları ayrıca uygulanmıştır.

Çalışmada kullanılan CNN modeli, ön eğitilmiş modeller ve transfer öğrenme süreci sırasıyla Bölüm 2.1.1 ve 2.1.1.1 ile detaylı olarak sunulmuştur. Bu çalışmanın diğer bileşenleri olan veri hazırlama ve önerilen CNN mimarisi ile deneysel sonuçlar yazının sonraki kısımlarında sunulacaktır.

6.1.3.1. Veri Hazırlama

AvLetters veri seti, sınıflandırma işlemine geçmeden önce veri ön işleme ve veri çoğaltma (DA) adımlarından oluşan veri hazırlama adımından geçmektedir. Veri ön işleme ile farklı çerçeve sayılarına sahip harf örneklerinin çerçeve sayısı eşitlenmiştir. Ön işlemde geçen veri setine veri çoğaltma teknikleri uygulanmıştır. Eğitim veri seti için uygulanan DA yöntemleri; gürültü eklemek için "gaussian", "salt and pepper" ve "speckle", keskinleştirme için "unsharp", yumuşatmak için "median" süzgeçleme olarak seçilmiştir. Ayrıca gürültülü görüntülerin RGB ve gri tonlamalı formatları da kullanılmıştır. Böylelikle, orijinal imgeden sekiz farklı şekilde yeni imgeler türetilmiştir. A harfine ait çoğaltılmış veri Şekil 6.2 ile artırılmış veri matrisi Şekil 6.3 ile gösterildiği gibidir.



Şekil 6.2. AvLetters, çoğaltılmış veri - A harfi.

Gaussian RGB Image	Salt & pepper RGB Image	Speckle RGB Image
Gaussian Grayscale Image	Salt & pepper Grayscale Image	Speckle Grayscale Image
Original Image	Median Filtered Image	Unsharp Filtered Image

Şekil 6.3. AvLetters artırılmış veri matrisi.

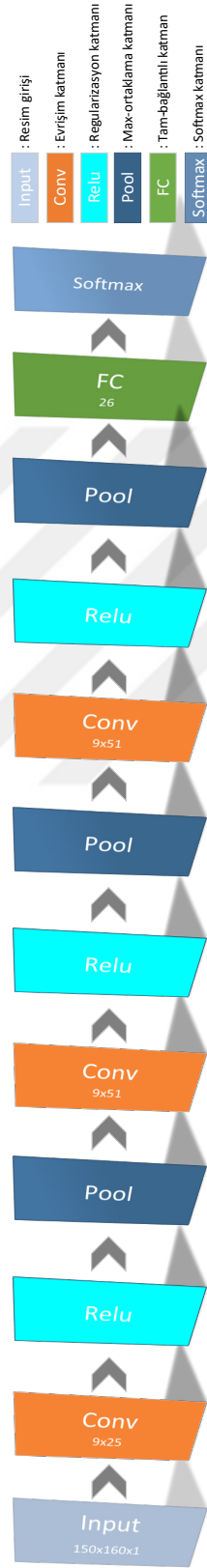
6.1.3.2. Tasarlanan CNN Modeli

Katman sayısındaki derinlikle yapay sinir ağı modellerinden ayrılan bir CNN modeli dudak okuma problemi için tasarlanmıştır. Şekil 6.4 ile sunulan ve bu çalışma için tasarlanan CNN modeli, 3 evrişim katmanı, 3 maksimuma ortaklama, 3 düzenleme katmanından oluşan bu modelle AvLetters veri seti üzerinde yapılan çalışmada %54,23'e kadar başarı elde edilmiştir.

6.1.3.3. Farklı “minibatch boyutu” Değerlerine Göre Tasarlanan ve Ön Eğitimli Model Destekli CNN Karşılaştırılması

Önerdiğimiz CNN mimarisi ile AlexNet ve GoogLeNet destekli CNN modelinin AvLetters veri seti üzerinde farklı minibatch boyutu değerleriyle elde edilen sonuç değerlerinin sunumu Tablo 6.1 ile yapılmıştır. Bu tabloya göre minibatch boyutu değeri 8 iken AlexNet destekli CNN modeli çoğaltılmış veri ile eğitildiğinde diğer modellere göre daha başarılı bir sonuç vermiştir. minibatch boyutu değeri 16 iken tarafımızca oluşturulan CNN mimarisi daha başarılı bir sonuç vermektedir. Son olarak minibatch boyutu değeri 32 iken tasarlanan CNN modeli çoğaltılmış veri ile eğitildiğinde daha başarılı sonuçlar vermektedir. Tüm çalışma sonucunda en yüksek başarı oranı minibatch boyutu değeri 8 iken AlexNet destekli CNN modeli çoğaltılmış veri ile eğitildiğinde %54,62 ile elde edilmiştir.

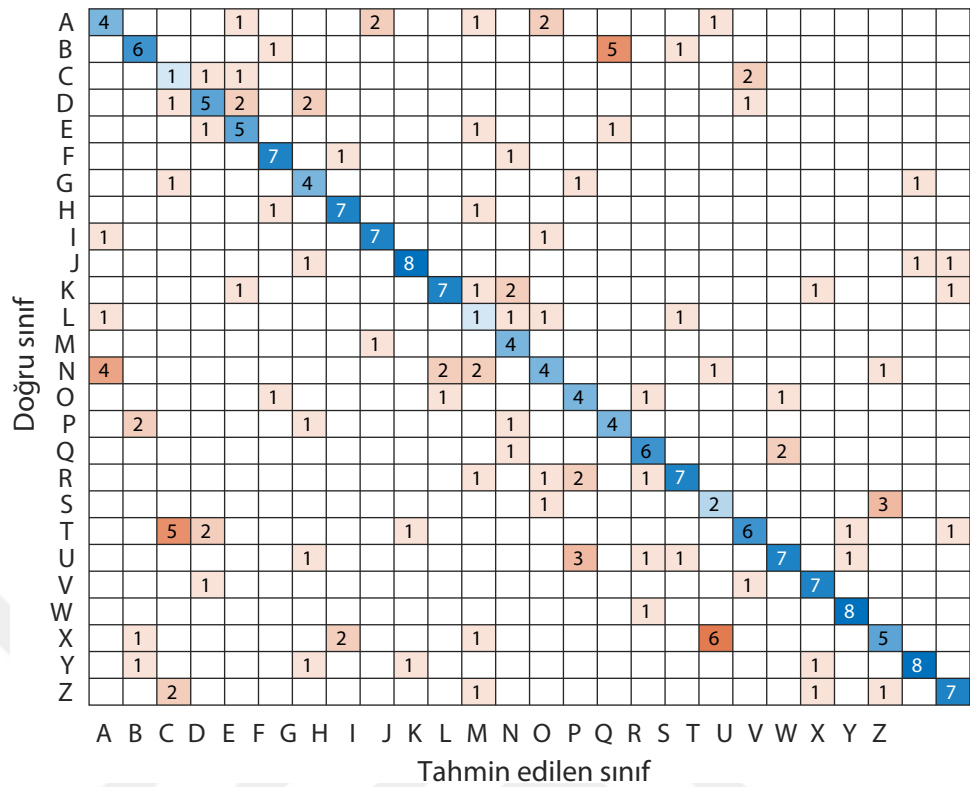
Tasarlanan ve ön eğitim destekli CNN modelleri için tüm çalışmalarda en iyi başarının elde edildiği durumlarda oluşan karmaşıklık matrisleri sırasıyla Şekil 6.5, 6.6 ve 6.7 ile verilmiştir.



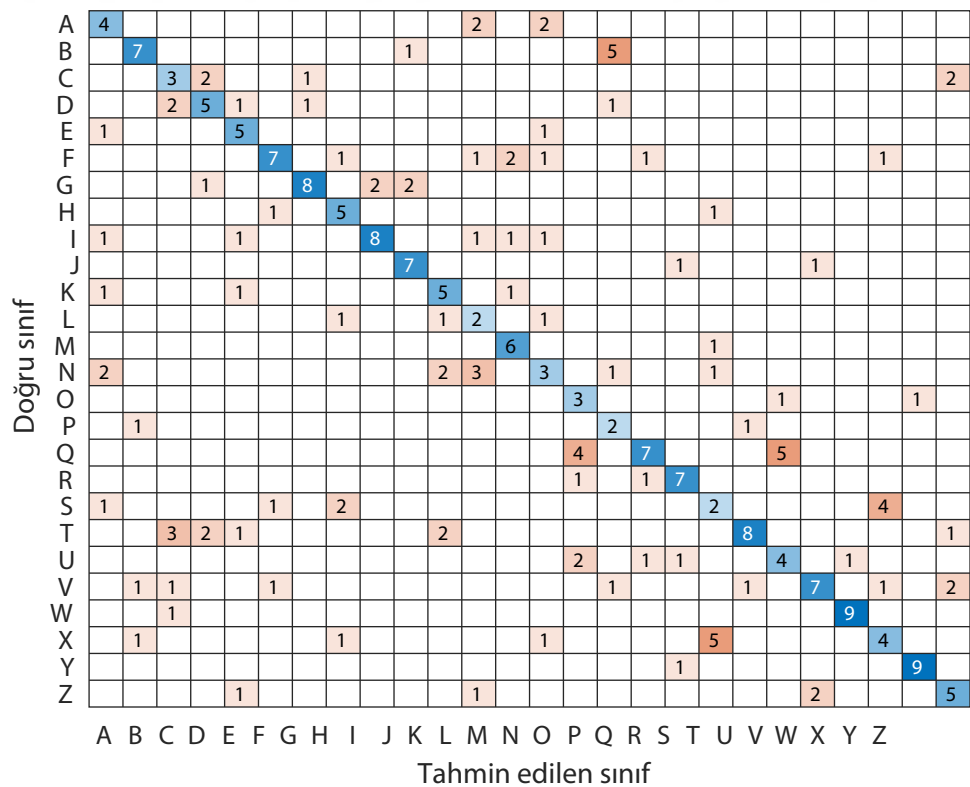
Şekil 6.4. Tasarlanan CNN modeli.

Tablo 6.1.1. Tasarlanan, AlexNet ve GoogLeNet destekli CNN modellerinin başarı karşılaştırması.

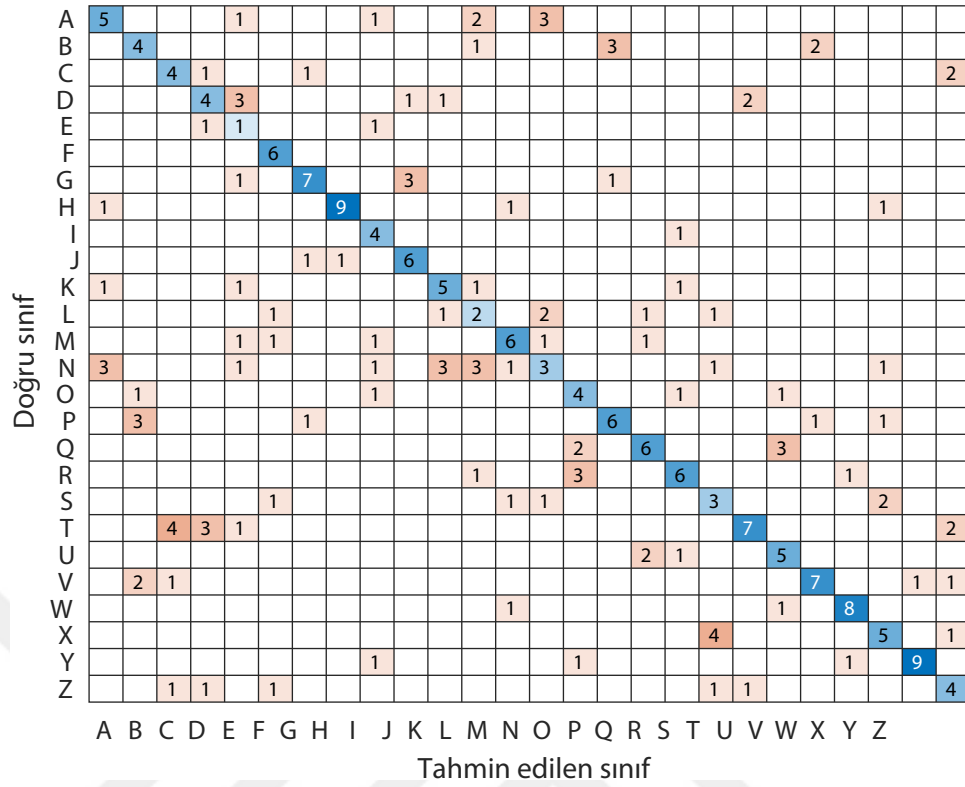
CNN Modelleri	Initial Parameters				
	Minibatch Boyutu	İlk Öğrenme Oranı	Maksimum Epok	Doğruluk	
Tasarlanan CNN	8	0.0001	100	0.0346	
Tasarlanan CNN + DA	8	0.0001	100	0.5423	
AlexNet Destekli CNN	8	0.0001	100	0.4769	
AlexNet Destekli CNN + DA	8	0.0001	100	0.5462	
GoogLeNet Destekli CNN	8	0.0001	100	0.3692	
GoogLeNet Destekli CNN + DA	8	0.0001	100	0.5231	
Tasarlanan CNN	16	0.0001	100	0.5231	
Tasarlanan CNN + DA	16	0.0001	100	0.4808	
AlexNet Destekli CNN	16	0.0001	100	0.4423	
AlexNet Destekli CNN + DA	16	0.0001	100	0.4808	
GoogLeNet Destekli CNN	16	0.0001	100	0.5192	
GoogLeNet Destekli CNN + DA	16	0.0001	100	0.5192	
Tasarlanan CNN	32	0.0001	100	0.4885	
Tasarlanan CNN + DA	32	0.0001	100	0.5385	
AlexNet Destekli CNN	32	0.0001	100	0.4577	
AlexNet Destekli CNN + DA	32	0.0001	100	0.4500	
GoogLeNet Destekli CNN	32	0.0001	100	0.3962	
GoogLeNet Destekli CNN + DA	32	0.0001	100	0.5000	



Şekil 6.5. Tasarlanan CNN için karmaşıklık matrisi.



Şekil 6.6. AlexNet destekli CNN için karmaşıklık matrisi.



Şekil 6.7. GoogLeNet destekli CNN için karmaşıklık matrisi.

Bu çalışmada kullanılan ve literatürde uygulanan derin öğrenme modelleri arasında doğruluk karşılaştırması Tablo 6.2 ile verilmiştir. Tabloya göre çalışma [186] ile özdevinimli kodlayıcı tabanlı yöntem kullanılarak literatürdeki en iyi sonuç elde edilmiştir.

Bu çalışmada kullanılan yöntemler en iyi başarı oranını veren yöntemlere yakın bir performans sergileyerek ve kimi çalışmaların başarı oranlarını geride bırakarak literatürdeki yerini almıştır [167].

Tablo 6.2. Önerilen yöntemin literatürdeki çalışmalarla performans karşılaştırması.

Metot	Doğruluk (%)
Özdevinimli kodlayıcı [186]	64,40
CNN [187]	49,90
CNN & LSTM [187]	57,70
CNN & bidirectional LSTM [187]	49,40
Tasarlanan CNN & DA	54,23
AlexNet destekli CNN & DA	54,62
GoogLeNet destekli CNN & DA	52,31

6.1.4. Değerlendirme

Bu çalışmada tasarlanan yeni bir CNN ve ön eğitim destekli CNN modelleri ile AvLetters veri seti üzerinde dudak okuma uygulaması gerçekleştirilmiştir. Aslında ses ve video tabanlı verilerden oluşan AvLetters veri setinin görüntü kısmı bu çalışmada kullanılmıştır. Dudak imgelerinden oluşan zaman serisi veriler önce farklı örnekler için aynı çerçeve sayısına eşitlenmiştir. Daha sonra bu çerçeveler sıralı şekilde birleştirilerek karesel şekle yakın tekil imgeler haline getirilmiştir. Derin öğrenmede veri sayısının başarı oranı üzerindeki etkisi dikkate alındığı için eğitim verileri için veri çoğullama teknikleri uygulanmış ve eğitim verisi 8 kat (8 farklı DA adımı uygulandığından) artırılmıştır. Deneysel çalışmalarda orijinal eğitim verisi ile DA uygulanmış veri seti ayrı ayrı değerlendirmeye alınmıştır.

Yapılan deneysel çalışmalar sonucunda, kullanıcı tabanlı ve ön eğitilmiş model destekli CNN modellerinin AvLetters veri seti üzerinde farklı başlangıç parametrelerine göre farklı sonuçlar verdiği ve mutlak galip bir modelin olmadığı görülmüştür. Bununla birlikte yapılan deneysel çalışmaların, literatürdeki yöntemlerle karşılaştırıldığında başarılı sonuçlar verdiği ve kimi başarılı yöntemleri geride bıraktığı tespit edilmiştir. Ayrıca DA'nın genellikle başarı performansını artırdığı deneysel sonuçlarla gözlemlenmiştir.

Gelecek çalışmalarda, farklı ön eğitilmiş modeller kullanılarak hem AvLetters hem de farklı veri setleriyle deneysel çalışmalar gerçekleştirilebilir. Ayrıca başlangıç parametrelerinin optimize edilmesi başarı oranını yükseltebilecek bir diğer önemli seçenektir.

7. BÖLÜM

TARTIŞMA, SONUÇ ve ÖNERİLER

7.1. Tartışma ve Sonuçlar

Bu tez çalışmasında insan edimleri 4 farklı alt başlıkta (işaret dili tanıma, günlük insan hareketi tanıma, duygu tanıma, dudak okuma) incelenmiş ve derin öğrenme yöntemleriyle performans analizi gerçekleştirilmiştir. Derin öğrenme tabanlı sınıflandırma yöntemlerinin başarımının artırılması için yapılan çalışmalarda başarıya ulaşıldığı tespit edilmiştir.

İnsan edimleri tanımanın bir alt başlığı olan işaret dili tanıma üzerine geliştirilen uygulamalar evrişimli sinir ağı (convolutional neural networks, CNN) tabanlı yöntemlerle gerçekleştirilmiştir. Yeni bir CNN modeli tasarlamak oldukça güç, zaman isteyen bir süreç gerektirmektedir. Bunun yerine ön eğitilmiş modeller kullanılarak transfer öğrenme gerçekleştirilmiş ve başarımları ortaya konulmuştur. İşaret dili ile harf ve rakam tanıma uygulamasında ABC, GA ve PSO algoritmalarının AlexNet destekli CNN modeline ait hiperparametrelerinin doğru seçimi ile başarı oranını ne şekilde etkilediği açık bir şekilde görülmektedir. Parametre seçiminde kullanılan ilgili evrimsel algoritmalar içerisinde ABC algoritmasının başarımı diğerlerine nispeten daha yüksektir. İşaret dili tanımada gerçekleştirilen diğer uygulamada, hastane ortamında kullanılmak üzere toplanan verilerle oluşturulan ERUSLR veri seti, yine transfer öğrenme destekli CNN tabanlı önerilen yöntemle sınıflandırılmıştır. Bu uygulamada, video tabanlı veriler bir teknik ile statik imge tabanlı örneklere çevrilmiştir. Literatürde sıkça karşılaşılmayan bu durum, başarısı düşük sonuçlar aldırılmamakla beraber yüksek başarımların performansı gösterimi sağlamıştır. Böylelikle literatürdeki zaman verisi tabanlı diğer uygulamalar için statik imge örneği dönüşümü gerçekleştirilip başarımlar

analizinin yapılabileceği gösterilmiştir. Uygulamanın bir diğer önemli katkısı, sezgisel optimizasyon algoritmalarının, hiperparametre seçiminde daha yüksek başarı olasılığı sağlamasına rağmen bunun yerine zamansal işlem maliyetini düşüren global arama ve rastgele arama gibi global arama tekniklerinin de başarılı olabileceği görülmüştür.

İnsan edimleri tanımanın bir diğer alt başlığı olan günlük insan hareket tanıma üzerine geliştirilen uygulamalar sonucunda hem yığınlanmış özdevinimli kodlayıcıların (stacked autoencoders, SAE) hem de CNN'nin bu alanda başarılı sonuçlar sağladığı görülmüştür. Sensör tabanlı verilerden insan hareketlerinin tanınmasını sağlayan SAE tabanlı önerilen yöntemle literatürde yaygın kullanılan UCI Har veri setinde bilinen en başarılı sonuç alınırken WISDM veri seti için bilinen en başarılı sonuca çok yakın bir performans elde edilmiştir. Bu başarının sağlanmasında SAE'nin yapısal parametrelerinin sezgisel optimizasyon algoritmalarıyla optimize edilmesi önemli bir etkidir. Bu uygulamada yaygın kullanılan optimizasyon algoritmaların yanı sıra ABC algoritmasının yerel arama, PSO algoritmasının küresel arama yeteneklerinin birleştirilmek istendiği yeni bir hibrit algoritma önerilmiş (hABCPSO) ve diğer algoritmalarla karşılaştırılmıştır. İstatiksel analizler ile desteklenen performans analizinde hABCPSO'nun diğer sezgisel algoritmalara göre daha başarılı sonuçlar verdiği saptanmıştır. İnsan günlük hareketlerinin tanıdığı diğer uygulamada hareket tanıma işlemi, statik tabanlı görsel veriler üzerinde gerçekleştirilmiştir. Altı farklı ön eğitilmiş model kullanılarak oluşturulan transfer öğrenme destekli CNN modelleri ile deneysel çalışmalar gerçekleştirilmiş ve hem kullanılan ön eğitilmiş modellerin birbirleriyle kıyaslaması gerçekleştirilmiş hem de en başarılı modelin literatürde yer alan diğer çalışmalarla performans karşılaştırılması yapılmıştır. Deneysel sonuçlara göre NASNet-Large destekli yöntemle literatürde bilinen en başarılı sonuç geride bırakılmıştır.

Tez çalışmasının bir başka bölümünde duygu tanıma uygulamaları geliştirilmiştir. Yapılan ilk uygulamada, internet arama motoru yardımıyla 5 sınıf içeren küçük boyutta bir duygu tanıma veri seti oluşturulmuş ve yüz tespiti, maskeleme gibi ön işlemlerin transfer öğrenme destekli CNN'in başarısını nasıl etkilediği incelenmiştir. Yapılan ön işlem çalışmalarının başarı oranını artırdığı tespit edilmiştir. Veri setinin az sayıda örnek içermesinden dolayı veri çoğullama adımı ayrıca uygulanmış olup ön işlem ve veri çoğullama adımlarının birlikte uygulandığı durumda en başarılı deneysel sonucun elde

edildiği görülmüştür. Bu alanda yapılan diğer duygu tanıma uygulaması için ilk olarak genel kullanıma açmak üzere yeni bir veri seti (ERUFER) oluşturulmuştur. Toplam 110 katılımcıdan alınan örneklerle oluşturulan ERUFER, 10 farklı duyguyu içermektedir. İşlem süresi kısa ama başarı oranı yüksek olan ResNet 18 ön eğitilmiş model kullanılarak transfer öğrenme destekli CNN tabanlı bir yöntem önerilmiştir. Sınıflandırma aşamasına geçmeden önce ilgili veri seti için veri hazırlama adımları uygulanmıştır. Bu aşamada yüz hizalama, gri formata dönüşüm, gürültü giderme, keskinleştirme ve veri çoğullama teknikleri uygulanmıştır. Performans yükseltmenin bir diğer aşaması olarak PSO algoritması ile ResNet 18 destekli CNN modelinin eğitim parametreleri en iyilenmiştir. Önerilen yöntemin başarımının test edilmesi için geliştirilen ERUFER veri setinin yanında literatürde sıklıkla tercih edilen JAFFE veri seti de ayrıca kullanılmıştır. Deneysel çalışmalar sonucunda ERUFER veri seti için başarılı sonuçlar elde edilmiş olup, JAFFE veri seti için 7 sınıf 10 kat çapraz doğrulama senaryosu üzerinden literatürde bilinen en iyi sonuç elde edilmiştir. Yapılan uygulamalar ile veri hazırlık aşamalarının ve eğitim parametre seçiminin başarı oranını artıran çok önemli faktörler olduğu görülmektedir.

Tez çalışmasının son bölümünde insan edimlerinin diğer alt başlığı olarak incelenen dudak okuma yer almaktadır. Burada ilk olarak zaman serisi tabanlı verilerin statik imge görüntülerine dönüştürülmesi sağlanmıştır. Sekiz farklı görüntü işleme tekniği ile veri çoğullaması gerçekleştirildikten sonra bu uygulama için tasarlanan CNN modeli ile AlexNet ve GoogLeNet destekli CNN modellerinin başarımları ayrı ayrı test edilmiştir. Farklı minibatch boyutu seçiminin başarımlara etkisi ayrıca incelenmiş olup uygulanan yöntemlerin performanslara etkisi araştırılmıştır. Veri çoğullama adımlarının başarı oranını genellikle artırdığı gözlemlenmiştir.

7.2. Öneriler

Bu tez çalışması ile bağlantılı olarak gelecekte aşağıda bulunan konuların araştırılması önerilmektedir:

Veri ön işleme adımlarının, uygulanan problemlerde başarı oranını pozitif yönde etkilediği tespit edilmiştir. İlgili uygulamalarda kullanılan ön işleme adımlarına ek olarak farklı ön işleme adımlarının uygulanması başarı oranını artırabileceğinden dolayı bu konu

üzerinde çalışmalar gerçekleştirilmesi önerilmektedir.

SAE'nin yapısal parametrelerini optimize etmek için geliştirilen ve başarılı sonuçlar üreten hABCPSO algoritmasının zamansal karmaşıklığının yüksek olma riski bulunmaktadır. Bu algoritmanın zamansal karmaşıklığını düşürmek için çalışmalar yapılması önem arz etmektedir.

Derin öğrenme ile insan edimlerinin tanınması üzerine gerçekleştirilen çalışmaların büyük bir kısmında önerilen yöntemler literatürde bilinen en iyi başarı oranını yakalamıştır. İlgili yöntemlerin farklı problemler için uygulanması önerilmektedir.

Parametre optimizasyonunda sezgisel optimizasyon ve global arama teknikleri kullanılmıştır. Kullanılan optimizasyon tekniklerinden farklı olarak diğer arama algoritmaları önerilen yöntemlerin ilgili kısımlarına eklenerek performans analizinin gerçekleştirilmesi sağlanabilir.

Yapılan uygulamaların ikisinde zaman serisi tabanlı veriler statik imgeler haline getirilmiştir. İlgili problemler için video tabanlı uygulamalar geliştirilip kıyaslamalar gerçekleştirilebilir.

KAYNAKLAR

1. Moeslund, T., Hilton, A., and Krüger, V., 2006. A survey of advances in vision-based human motion capture and analysis, (2-3 SPEC. ISS.):90–126.
2. Güçlü, A., 2012. Gerçek Zamanlı Video Görüntülerinden İnsan Hareketlerinin Tanınması ve Yorumlanması, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü.
3. Keçeli, A., 2014. Derinlik Bilgisi Kullanılarak İnsan Hareketlerinin Tanınması, Doktora Tezi, Fen Bilimleri Enstitüsü.
4. Krizhevsky, A., Sutskever, I., and Hinton, G., 2012. Imagenet classification with deep convolutional neural networks, **NIPS**, **25**:1106–1114.
5. Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization, Technical report, Erciyes University, Kayseri.
6. Holland, J.H., 1992. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, A Bradford Book.
7. Kennedy, J. and Eberhart, R., 1995. Particle swarm optimization, *Proceedings of ICNN'95-International Conference on Neural Networks*, volume 4, 1942–1948, IEEE.
8. Mavi, A., Sign Language Digits Dataset, <https://www.kaggle.com/ardamavi/sign-language-digits-dataset/home>, Çevrimiçi: Son erişim 15 Ağustos 2018.
9. Birk, H., Moeslund, T., and Madsen, C., 1997. Real-time recognition of hand alphabet gestures using principal component analysis, *The Scandinavian Conference on Image Analysis*, 261–268.
10. Beser, F., Kizrak, M.A., Bolat, B., and Yildirim, T., 2018. Recognition of sign language using capsule networks, *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 1–4.

11. Oyedotun, O.K. and Khashman, A., 2018. Prototype-Incorporated Emotional Neural Network, **IEEE Transactions on Neural Networks and Learning Systems**, **29**(8):3560–3572.
12. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A., 2015. Going deeper with convolutions, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9, IEEE, Boston.
13. Storn, R. and Price, K., 1997. Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces, **Journal of Global Optimization**, **11**(4):341–359.
14. Anguita, D., Ghio, A., Oneto, L., Parra, X., and Reyes-Ortiz, J.L., 2013. A Public Domain Dataset for Human Activity Recognition using Smartphones, *21st European Symposium on Artificial Neural Networks*, 437–442.
15. Kwapisz, J.R., Weiss, G.M., and Moore, S.A., 2011. Activity recognition using cell phone accelerometers, **ACM SigKDD Explorations Newsletter**, **12**(2):74–82.
16. Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition, **arXiv**, **1409.1556**:1–14.
17. He, K., Zhang, X., Ren, S., and Sun, J., 2016. Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778, IEEE, Las Vegas.
18. Zoph, B., Vasudevan, V., Shlens, J., and Le, Q.V., 2018. Learning transferable architectures for scalable image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8697–8710, IEEE, Salt Lake City.
19. Lavinia, Y., Vo, H., and Verma, A., 2019. New Color Fusion Deep Learning Model for Large-Scale Action Recognition, **International Journal of Computational Vision and Robotics**, **in press**:1 – 19.

20. Li, S. and Deng, W., 2018. Deep Facial Expression Recognition: A Survey, **CoRR**, **1804.08348**:1–25.
21. Viola, P., Jones, M., et al., 2001. Rapid object detection using a boosted cascade of simple features, **CVPR (1)**, **1**(511-518):3.
22. King, D.E., 2009. Dlib-ml: A Machine Learning Toolkit, **Journal of Machine Learning Research**, **10**:1755–1758.
23. He, K., Zhang, X., Ren, S., and Sun, J., 2016. Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.
24. Matthews, I., Cootes, T., Bangham, J.A., Cox, S., and Harvey, R., 2002. Extraction of Visual Features for Lipreading, **IEEE Transactions on Pattern Analysis and Machine Intelligence**, **24**(2):198–213.
25. Lin, H.I., Hsu, M.H., and Chen, W.K., 2014. Human hand gesture recognition using a convolution neural network, *Automation Science and Engineering (CASE), 2014 IEEE International Conference*, 1038–1043.
26. Alashhab, S., Gallego, A.J., and Lozano, M.Á., 2019. Hand Gesture Detection with Convolutional Neural Networks, *F. De La Prieta, S. Omatu, and A. Fernández-Caballero, Eds., Distributed Computing and Artificial Intelligence, 15th International Conference*, 45–52, Springer International Publishing, Cham.
27. Flores, C.J.L., Cutipa, A.E., and Enciso, R.L., 2017. Application of convolutional neural networks for static hand gestures recognition under different invariant features, *Proceedings of the 2017 IEEE 24th International Congress on Electronics, Electrical Engineering and Computing*, 5–8.
28. Vedaldi, A. and Lenc, K., 2015. MatConvNet: Convolutional Neural Networks for MATLAB, *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, 689–692, ACM, New York, NY, USA.

29. Arenas, J.O.P., Murillo, P.C.U., and Moreno, R.J., 2017. Convolutional neural network architecture for hand gesture recognition, *International Conference on Electronics, Electrical Engineering and Computing*, 1–4.
30. Cote-Allard, U., Fall, C.L., Campeau-Lecours, A., Gosselin, C., Laviolette, F., and Gosselin, B., 2017. Transfer learning for sEMG hand gestures recognition using convolutional neural networks, *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1663–1668.
31. Oyedotun, O.K. and Khashman, A., 2017. Deep learning in vision-based static hand gesture recognition, **Neural Computing and Applications**, **28**(12):3941–3951.
32. Sabour, S., Frosst, N., and Hinton, G.E., 2017. Dynamic Routing Between Capsules, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, 3859–3869.
33. Athitsos, V., Neidle, C., Sclaroff, S., Nash, J., Stefan, A., Yuan, Q., and Thangali, A., 2008. The american sign language lexicon video dataset, *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 1–8, IEEE.
34. Neidle, C., Thangali, A., and Sclaroff, S., 2012. Challenges in development of the american sign language lexicon video dataset (asllvd) corpus, *Proc. 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon, Language Resources and Evaluation Conference (LREC) 2012*, 1–5, Citeseer.
35. Kim, J.H., Kim, N., Park, H., and Park, J.C., 2016. Enhanced sign language transcription system via hand tracking and pose estimation, **Journal of Computing Science and Engineering**, **10**(3):95–101.
36. Metaxas, D., Dilsizian, M., and Neidle, C., 2018. Scalable ASL sign recognition using model-based machine learning and linguistically annotated corpora, *8th Workshop on the Representation & Processing of Sign Languages: Involving the Language Community, Language Resources and Evaluation Conference 2018*.

37. Oszust, M. and Wysocki, M., 2013. Polish sign language words recognition with Kinect, *2013 6th International Conference on Human System Interactions (HSI)*, 219–226.
38. Oszust, M. and Wysocki, M., 2014. Some Approaches to Recognition of Sign Language Dynamic Expressions with Kinect, 75–86, Springer International Publishing, Cham.
39. Kapuscinski, T., Oszust, M., Wysocki, M., and Warchol, D., 2015. Recognition of hand gestures observed by depth cameras, **International Journal of Advanced Robotic Systems**, **12**(4):36.
40. Ronchetti, F., Quiroga, F., Estrebou, C.A., Lanzarini, L.C., and Rosete, A., 2016. LSA64: an Argentinian sign language dataset, *XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016)*., 794–803.
41. Ronchetti, F., 2017. Thesis Overview: Dynamic Gesture Recognition and its Application to Sign Language, **Journal of Computer Science and Technology**, **17**:1–10.
42. Konstantinidis, D., Dimitropoulos, K., and Daras, P., 2018. Sign Language Recognition based on Hand and Body Skeletal Data, *2018-3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON)*, 1–4.
43. Masood, S., Srivastava, A., Thuwal, H.C., and Ahmad, M., 2018. Real-Time Sign Language Gesture (Word) Recognition from Video Sequences Using CNN and RNN, *V. Bhateja, C.A. Coello Coello, S.C. Satapathy, and P.K. Pattnaik, Eds., Intelligent Engineering Informatics*, 623–632, Springer Singapore, Singapore.
44. Almaslukh, B., AlMuhtadi, J., and Artoli, A., 2017. An effective deep autoencoder approach for online smartphone-based human activity recognition, **International Journal of Computer Science and Network Security (IJCSNS)**, **17**(4):160–165.
45. Alsheikh, M.A., Selim, A., Niyato, D., Doyle, L., Lin, S., and Tan, H.P., 2016. Deep Activity Recognition Models with Triaxial Accelerometers., *AAAI*

Workshop: Artificial Intelligence Applied to Assistive Technologies and Smart Environments, 1–6.

46. Lockhart, J.W., Weiss, G.M., Xue, J.C., Gallagher, S.T., Grosner, A.B., and Pulickal, T.T., 2011. Design considerations for the WISDM smart phone-based sensor mining architecture, *Proceedings of the Fifth International Workshop on Knowledge Discovery from Sensor Data*, 25–33, ACM.
47. Chen, Y., Zhong, K., Zhang, J., Sun, Q., and Zhao, X., 2016. *LSTM Networks for Mobile Human Activity Recognition*, 1–4.
48. Jiang, W. and Yin, Z., 2015. Human activity recognition using wearable sensors by deep convolutional neural networks, *Proceedings of the 23rd ACM international conference on Multimedia*, 1307–1310, ACM.
49. Li, Y., Shi, D., Ding, B., and Liu, D., 2014. Unsupervised feature learning for human activity recognition using smartphone sensors, *Mining Intelligence and Knowledge Exploration*, 99–107, Springer.
50. Ravi, D., Wong, C., Lo, B., and Yang, G.Z., 2016. Deep learning for human activity recognition: A resource efficient implementation on low-power devices, *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, 71–76.
51. Ravi, D., Wong, C., Lo, B., and Yang, G.Z., 2017. A Deep Learning Approach to on-Node Sensor Data Analytics for Mobile or Wearable Devices, **IEEE Journal of Biomedical and Health Informatics**, **21**(1):56–64.
52. Ronao, C.A. and Cho, S.B., 2015. Deep convolutional neural networks for human activity recognition with smartphone sensors, *International Conference on Neural Information Processing*, 46–53, Springer.
53. Ronao, C. and Cho, S.B., 2015. Evaluation of deep convolutional neural network architectures for human activity recognition with smartphone sensors, **Proc. of the KIISE Korea Computer Congress 2015**, 858–860.

54. Ronao, C.A. and Cho, S.B., 2016. Human activity recognition with smartphone sensors using deep learning neural networks, **Expert Systems with Applications**, **59**:235–244.
55. Walse, K.H., Dharaskar, R.V., and Thakare, V.M., 2016. Pca based optimal ann classifiers for human activity recognition using mobile sensors data, *Proceedings of First International Conference on Information and Communication Technology for Intelligent Systems: Volume 1*, 429–436, Springer.
56. Bas, C., Zalluhoglu, C., and Ikizler-Cinbis, N., 2017. Using deep multiple instance learning for action recognition in still images, *25th Signal Processing and Communications Applications Conf. (SIU)*, 1–4, IEEE, Antalya.
57. Qi, T., Xu, Y., Quan, Y., Wang, Y., and Ling, H., 2017. Image-based action recognition using hint-enhanced deep neural networks, **Neurocomputing**, **267**(Supplement C):475 – 488.
58. Yao, B., Jiang, X., Khosla, A., Lin, A.L., Guibas, L., and Fei-Fei, L., 2011. Human action recognition by learning bases of action attributes and parts, *2011 International Conference on Computer Vision*, 1331–1338.
59. Florida, U.O.C., 2012, UCF sports action dataset, [http :
//vision.eecs.ucf.edu/datasets/Actions.html](http://vision.eecs.ucf.edu/datasets/Actions.html), Çevrimiçi: Son erişim 20 Haziran 2019.
60. Lab, T.S., 2011, Hmdb: A large video database for human motion recognition, [http :
//serre – lab.clps.brown.edu/resources/HMDB/index.htm](http://serre-lab.clps.brown.edu/resources/HMDB/index.htm), Çevrimiçi: Son erişim 20 Haziran 2019.
61. Wang, L., Qiao, Y., Tang, X., and Van Gool, L., 2016. Actionness estimation using hybrid fully convolutional networks, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2708–2717, IEEE, Las Vegas.
62. Zhao, Z., Ma, H., and You, S., 2017. Single Image Action Recognition Using Semantic Body Part Actions, *IEEE Int. Conference on Computer Vision (ICCV)*, 3411–3419, IEEE, Venice.

63. Lavinia, Y., Vo, H., and Verma, A., 2017. Fusion based deep CNN for improved large-scale image action recognition, *Proceedings - 2016 IEEE International Symposium on Multimedia, ISM 2016*, 609–614, Institute of Electrical and Electronics Engineers Inc.
64. Susskind, J.M., Anderson, A.K., and Hinton, G.E., 2010. The Toronto face database, Technical report, University of Toronto, Department of Computer Science, ON, Canada, vol. 3.
65. Goodfellow, I.J., Erhan, D., Carrier, P.L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.H., Zhou, Y., Ramaiah, C., Feng, F., Li, R., Wang, X., Athanasakis, D., Shave-Taylor, J., Milakov, M., Park, J., Ionescu, R., Popescu, M., Grozea, C., Bergstra, J., Xie, J., Romaszko, L., Xu, B., Chuang, Z., and Bengio, Y., 2015. Challenges in representation learning: A report on three machine learning contests, *Neural Networks*, **64**:59 – 63.
66. Dhall, A., Goecke, R., Lucey, S., and Gedeon, T., 2011. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark, *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2106–2112.
67. Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D.H.J., Hawk, S.T., and van Knippenberg, A., 2010. Presentation and validation of the Radboud Faces Database, *Cognition and Emotion*, **24**(8):1377–1388.
68. Lundqvist, D., Flykt, A., and Ohman, A., 1998. The Karolinska Directed Emotional Faces – KDEF, *CD ROM from Department of Clinical Neuroscience, Psychology section*.
69. Lyons, M., Akamatsu, S., Kamachi, M., and Gyoba, J., 1998. Coding facial expressions with Gabor wavelets, *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, 200–205.
70. Devries, T., Biswaranjan, K., and Taylor, G.W., 2014. Multi-task learning of facial landmarks and expression, *Canadian Conference on Computer and Robot Vision*, 98–103, IEEE.

71. Khorrami, P., Paine, T., and Huang, T., 2015. Do deep neural networks learn facial action units when doing expression recognition?, *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 19–27.
72. Ding, H., Zhou, S.K., and Chellappa, R., 2017. Facenet2expnet: Regularizing a deep face recognition net for expression recognition, *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, 118–126, IEEE.
73. Tang, Y., 2013. Deep Learning using Support Vector Machines, **CoRR**, **1306.0239**:1–5.
74. Zhang, Z., Luo, P., Loy, C.C., and Tang, X., 2015. Learning social relation traits from face images, *Proceedings of the IEEE International Conference on Computer Vision*, 3631–3639.
75. Guo, Y., Tao, D., Yu, J., Xiong, H., Li, Y., and Tao, D., 2016. Deep neural networks with relativity learning for facial expression recognition, *2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 1–6, IEEE.
76. Kim, B., Dong, S., Roh, J., Kim, G., and Lee, S., 2016. Fusing Aligned and Non-aligned Face Information for Automatic Affect Recognition in the Wild: A Deep Learning Approach, *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1499–1508.
77. Pramerdorfer, C. and Kampel, M., 2016. Facial Expression Recognition using Convolutional Neural Networks: State of the Art, **CoRR**, **1612.02903**:1–6.
78. Chang, T., Li, H., Wen, G., Hu, Y., and Ma, J., 2019. Facial expression recognition sensing the complexity of testing samples, **Applied Intelligence**, **49**(12):4319–4334.
79. Li, D., Wen, G., Li, X., and Cai, X., 2019. Graph-based dynamic ensemble pruning for facial expression recognition, **Applied Intelligence**, **49**(9):3188–3206.

80. Kim, B.K., Lee, H., Roh, J., and Lee, S.Y., 2015. Hierarchical Committee of Deep CNNs with Exponentially-Weighted Decision Fusion for Static Facial Expression Recognition, *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, 427–434, ACM, New York, NY, USA.
81. Yu, Z. and Zhang, C., 2015. Image Based Static Facial Expression Recognition with Multiple Deep Network Learning, *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, 435–442, ACM.
82. Cai, J., Meng, Z., Khan, A.S., Li, Z., O'Reilly, J., and Tong, Y., 2018. Island loss for learning discriminative features in facial expression recognition, *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, 302–309, IEEE.
83. Meng, Z., Liu, P., Cai, J., Han, S., and Tong, Y., 2017. Identity-Aware Convolutional Neural Network for Facial Expression Recognition, *2017 12th IEEE International Conference on Automatic Face Gesture Recognition (FG 2017)*, 558–565.
84. Venkatraman, S., Balasubramanian, S., and Gera, D., 2017. Multiple face-component analysis: A unified approach towards facial recognition tasks, *2017 2nd International Conference on Man and Machine Interfacing (MAMI)*, 1–6, IEEE.
85. Sun, W., Song, Y., Jin, Z., Zhao, H., and Chen, C., 2019. Unsupervised Orthogonal Facial Representation Extraction via image reconstruction with correlation minimization, *Neurocomputing*, **337**:203 – 217.
86. Kim, T., Yu, C., and Lee, S., 2018. Facial expression recognition using feature additive pooling and progressive fine-tuning of CNN, *Electronics Letters*, **54**(23):1326–1328.
87. Liu, P., Han, S., Meng, Z., and Tong, Y., 2014. Facial Expression Recognition via a Boosted Deep Belief Network, *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 1805–1812.

88. Hamester, D., Barros, P., and Wermter, S., 2015. Face expression recognition with a 2-channel Convolutional Neural Network, *2015 International Joint Conference on Neural Networks (IJCNN)*, 1–8.
89. Garg, A., Noyola, J., and Bagadia, S., 2016. Lip reading using CNN and LSTM, Technical report, Technical report, Stanford University, CS231n project report.
90. Rekik, A., Ben-Hamadou, A., and Mahdi, W., 2014. A New Visual Speech Recognition Approach for RGB-D Cameras, *A. Campilho and M. Kamel, Eds., Image Analysis and Recognition*, 21–28, Springer International Publishing, Cham.
91. Li, Y., Takashima, Y., Takiguchi, T., and Ariki, Y., 2016. Lip reading using a dynamic feature of lip images and convolutional neural networks, *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, 1–6.
92. Kurematsu, A., Takeda, K., Sagisaka, Y., Katagiri, S., Kuwabara, H., and Shikano, K., 1990. ATR Japanese speech database as a tool of speech recognition and synthesis, **Speech Communication**, **9**:357–363.
93. Petridis, S., Li, Z., and Pantic, M., 2017. End-to-end visual speech recognition with LSTMs, *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2592–2596, IEEE.
94. Anina, I., Zhou, Z., Zhao, G., and Pietikäinen, M., 2015. OuluVS2: A multi-view audiovisual database for non-rigid mouth motion analysis, *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 1–5.
95. Patterson, E.K., Gurbuz, S., Tufekci, Z., and Gowdy, J.N., 2002. Moving-talker, Speaker-independent Feature Study, and Baseline Results Using the CUAVE Multimodal Speech Corpus, **EURASIP J. Appl. Signal Process.**, **2002**(1):1189–1201.

96. Dong, W., He, R., and Zhang, S., 2016. Digital recognition from lip texture analysis, *2016 IEEE International Conference on Digital Signal Processing (DSP)*, 477–481.
97. Stafylakis, T. and Tzimiropoulos, G., 2017. Combining Residual Networks with LSTMs for Lipreading, *CoRR*, **1703.04105**:1–5.
98. Chung, J.S. and Zisserman, A., 2016. Lip reading in the wild, *Asian Conference on Computer Vision*, 87–103, Springer.
99. Takashima, Y., Kakihara, Y., Aihara, R., Takiguchi, T., Ariki, Y., Mitani, N., Omori, K., and Nakazono, K., 2015. Audio-Visual Speech Recognition Using Convolutional Bottleneck Networks for a Person with Severe Hearing Loss, *IPSN Transactions on Computer Vision and Applications*, **7**:64–68.
100. Takashima, Y., Aihara, R., Takiguchi, T., Ariki, Y., Mitani, N., Omori, K., and Nakazono, K., 2016. Audio-Visual Speech Recognition Using Bimodal-Trained Bottleneck Features for a Person with Severe Hearing Loss., *Interspeech*, 277–281.
101. Yargıç, A. and Doğan, M., 2013. A lip reading application on MS Kinect camera, *2013 IEEE INISTA*, 1–5.
102. Akcay, S., Kundegorski, M., Willcocks, C., and Breckon, T., 2018. Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery, **13**(9):2203–2215.
103. Alnujaim, I., Alali, H., Khan, F., and Kim, Y., 2018. Hand gesture recognition using input impedance variation of two antennas with transfer learning, **18**(10):4129–4135.
104. Han, D., Liu, Q., and Fan, W., 2018. A new image classification method using CNN transfer learning and web data augmentation, **95**:43–56.
105. Bussey, D., Glandon, A., Vidyaratne, L., Alam, M., and Iftekharuddin, K., 2018. Convolutional neural network transfer learning for robust face recognition in NAO humanoid robot, *2017 IEEE Symposium Series on Computational*

Intelligence (SSCI), volume 2018-January, 1–7, Institute of Electrical and Electronics Engineers Inc.

106. Tang, Y., Wu, B., Peng, L., and Liu, C., 2017. Semi-Supervised Transfer Learning for Convolutional Neural Network Based Chinese Character Recognition, *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 441–447, CPS, Kyoto.
107. Xia, Y., Huang, D., and Wang, Y., 2017. Detecting Smiles of Young Children via Deep Transfer Learning, *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, 1673–1681, Italy, Venice.
108. Keceli, A.S., 2018. Viewpoint projection based deep feature learning for single and dyadic action recognition, **Expert Systems with Applications**, **104**:235 – 243.
109. Gupta, A. and Balan, M.S., 2018. Action Recognition from Optical Flow Visualizations, *B.B. Chaudhuri, M.S. Kankanhalli, and B. Raman, Eds.: Proceedings of 2nd International Conference on Computer Vision & Image Processing*, 397–408, Springer Singapore, Singapore.
110. Keceli, A.S., Kaya, A., and Can, A.B., 2017. Action recognition with skeletal volume and deep learning, *25th Signal Processing and Communications Applications Conf. (SIU)*, 1–4, IEEE, Antalya.
111. Han, Y., Zhang, P., Zhuo, T., Huang, W., and Zhang, Y., 2018. Going deeper with two-stream ConvNets for action recognition in video surveillance, **Pattern Recognition Letters**, **107**:83 – 90.
112. Pham, H.H., Khoudour, L., Crouzil, A., Zegers, P., and Velastin, S.A., 2018. Exploiting deep residual networks for human action recognition from skeletal data, **Computer Vision and Image Understanding**, **170**:51–66.
113. Rubio, G., Pomares, H., Rojas, I., and Herrera, L.J., 2011. A heuristic method for parameter selection in LS-SVM: Application to time series prediction, **International Journal of Forecasting**, **27**(3):725–739.

114. Klein, A., Falkner, S., Bartels, S., Hennig, P., and Hutter, F., 2016. Fast Bayesian Optimization of Machine Learning Hyperparameters on Large Datasets, **arXiv**, **54**:1–9.
115. Yao, C., Cai, D., Bu, J., and Chen, G., 2017. Pre-training the deep generative models with adaptive hyperparameter optimization, **Neurocomputing**, **247**:144–155.
116. Diaz, G., Fokoue, A., Nannicini, G., and Samulowitz, H., 2017. An effective algorithm for hyperparameter optimization of neural networks, **arXiv**, **61**(4):1–20.
117. Desmet, B. and Hoste, V., 2018. Online suicide prevention through optimised text classification, **Information Sciences**, **439-440**:61–78.
118. Sánchez-Illana, Á., Pérez-Guaita, D., Cuesta-García, D., Sanjuan-Herráez, J.D., Vento, M., Ruiz-Cerdá, J.L., Quintás, G., and Kuligowski, J., 2018. Model selection for within-batch effect correction in UPLC-MS metabolomics using quality control - Support vector regression, **Analytica Chimica Acta**, **1026**:62–68.
119. Hinz, T., Navarro-Guerrero, N., Magg, S., and Wermter, S., 2018. Speeding up the Hyperparameter Optimization of Deep Convolutional Neural Networks, **International Journal of Computational Intelligence and Applications**, **17**(02):1–15.
120. Rahnama, A.H.A., Toloo, M., and Zaidenberg, N.J., 2018. An LP-based hyperparameter optimization model for language modeling, **arXiv**, 1–11.
121. Riegler, M., 2018. HINDSIGHT: An R-based Framework Towards Long Short Term Memory (LSTM) Optimization, *Proceedings of the 9th ACM Multimedia Systems Conference*, 381–386.
122. Stoean, R., 2018. Analysis on the potential of an EA-surrogate modelling tandem for deep learning parametrization: an example for cancer classification from medical images, **Neural Computing and Applications**, 1–10.

123. Maraqa, M. and Abu-Zaiter, R., 2008. Recognition of Arabic Sign Language (ArSL) using recurrent neural networks, *2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT)*, 478–481, IEEE.
124. Chevtchenko, S.F., Vale, R.F., and Macario, V., 2018. Multi-objective optimization for hand posture recognition, **Expert Systems with Applications**, **92**:170–181.
125. Barczak, A., Reyes, N., Abastillas, M., Piccio, A., and Susnjak, T., 2011. A New 2D Static Hand Gesture Colour Image Dataset for ASL Gestures, **Res Lett Inf Math Sci**, **15**:12–20.
126. Sagayam, K.M. and Hemanth, D.J., 2018. ABC algorithm based optimization of 1-D hidden Markov model for hand gesture recognition applications, **Computers in Industry**, **99**(April):313–323.
127. Kim, T., Wong, S., and Cipolla, R., 2007. Tensor Canonical Correlation Analysis for Action Classification, *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 1–8.
128. Czuszynski, K., Ruminski, J., and Kwasniewska, A., 2018. Gesture recognition with the linear optical sensor and recurrent neural networks, **IEEE Sensors Journal**, **18**(13):5429–5438.
129. Badem, H., Basturk, A., Caliskan, A., and Yuksel, M.E., 2017. A new efficient training strategy for deep neural networks by hybridization of artificial bee colony and limited-memory BFGS optimization algorithms, **Neurocomputing**, **266**:506 – 526.
130. Rajabioun, R., 2011. Cuckoo Optimization Algorithm, **Appl. Soft Comput.**, **11**(8):5508–5518.
131. Nawi, N.M., Khan, A., Rehman, M., Chiroma, H., and Herawan, T., 2015. Weight optimization in recurrent neural networks with hybrid metaheuristic Cuckoo search techniques for data classification, **Mathematical Problems in Engineering**, **2015**:1–13.

132. Nawi, N.M., Rehman, M., et al., 2014. CSBPRNN: a new hybridization technique using cuckoo search to train back propagation recurrent neural network, *Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*, 111–118, Springer Singapore, Singapore.
133. Chen, P., Xie, Y., Jin, P., and Zhang, D., 2018. A wireless sensor data-based coal mine gas monitoring algorithm with least squares support vector machines optimized by swarm intelligence techniques, **International Journal of Distributed Sensor Networks**, **14**(5):1–21.
134. Analitik, D., 2019, Derin Öğrenme, [www.derinanalitik.com/single – post/deep – learning – nedir](http://www.derinanalitik.com/single-post/deep-learning-nedir), Çevrimiçi: Son erişim 29 Ekim 2019.
135. Hatipoğlu, P., 2016. Zaman Serilerinin Derin Öğrenme İle Sınıflandırılması, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü.
136. Özcan, H., 2014. Çok Düşük Çözünürlüklü Yüz İmgelerinde Derin Öğrenme Uygulamaları, Yüksek Lisans Tezi, Deniz Harp Okulu Deniz Bilimleri ve Mühendisliği Enstitüsü.
137. Kılıç, E., 2016. Evrişimli Sinir Ağları İle Mitotik Figür Sınıflandırması, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü.
138. Mathworks, Documentation, [https : //www.mathworks.com/help](https://www.mathworks.com/help), Çevrimiçi: Son erişim 18 Ağustos 2019.
139. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., and Fei-Fei, L., 2015. ImageNet Large Scale Visual Recognition Challenge, **International Journal of Computer Vision (IJCV)**, **115**(3):211–252.
140. Zhao, L., Wang, J., Li, X., Tu, Z., and Zeng, W., 2016. On the Connection of Deep Fusion to Ensembling, **CoRR**, **1611.07718**:1–12.
141. University, S., 2015, UFLDL Tutorial, ufldl.stanford.edu/wiki/index.php, Çevrimiçi: Son erişim 30 Ekim 2019.

142. Analitik, D., Otomatik Kodlayıcılar ve Çeşitleri, *www.derinanalitik.com/single - post/autoencoders - otomatik - kodlayicilar - nedir*, Çevrimiçi: Son erişim 29 Ekim 2019.
143. Basturk, A., Sarikaya Basturk, N., and Qurbanov, O., 2018. A Comparative Performance Analysis of Various Classifiers for Fingerprint Recognition, **Omer Halisdemir Universitesi Muhendislik Bilimleri Dergisi**, 7:504 – 513.
144. Basturk, A. and Akay, R., 2013. Performance Analysis of the Coarse-grained Parallel Model of the Artificial Bee Colony Algorithm, **Inf. Sci.**, 253:34–55.
145. Basturk, A. and Akay, R., 2012. Parallel Implementation of Synchronous Type Artificial Bee Colony Algorithm for Global Optimization, **Journal of Optimization Theory and Applications**, 155(3):1095–1104.
146. Sahin, O. and Akay, B., 2016. Comparisons of metaheuristic algorithms and fitness functions on software test data generation, **Applied Soft Computing**, 49:1202–1214.
147. Yuksel, M.E., Basturk, N.S., Badem, H., Caliskan, A., and Basturk, A., 2018. Classification of high resolution hyperspectral remote sensing data using deep neural networks, **Journal of Intelligent & Fuzzy Systems**, 34:2273–2285.
148. Aslan, S., 2016. Genomik verilerin analizinde Yapay Arı Koloni Algoritması Temelli Yeni Tekniklerin Geliştirilmesi, Doktora, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü.
149. Özcan, T., 2016. Kalman Süzgeci Parametrelerinin En İyilenmesinde Zeki Optimizasyon Tekniklerinin Başarımlarının İncelenmesi, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü.
150. Hancer, E., Ozturk, C., and Karaboga, D., 2013. Extraction of brain tumors from MRI images with artificial bee colony based segmentation methodology, *2013 8th International Conference on Electrical and Electronics Engineering (ELECO)*, 516–520, IEEE, Bursa.

151. Wang, B., 2015. A novel artificial bee colony algorithm based on modified search strategy and generalized opposition-based learning, **Journal of Intelligent & Fuzzy Systems**, **28**:1023–1037.
152. Ozturk, C., Hancer, E., and Karaboga, D., 2014. Kuresel En İyi Yapay Ari Koloni Algoritmasi ile Otomatik Kumeleme, **Gazi Universitesi Muhendislik-Mimarlik Fakultesi Dergisi**, **29**(4):677–687.
153. Karaboga, D., 2011. Yapay Zeka Optimizasyon Algoritmaları, Nobel.
154. Banharsakun, A., Achalakul, T., and Sirinaovakul, B., 2011. The best-so-far selection in Artificial Bee Colony algorithm, **Applied Soft Computing**, **11**(2):2888 – 2901.
155. Ozcan, T. and Basturk, A., 2019. Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition, **Neural Computing and Applications**, **31**(12):8955–8970.
156. Chen, Q., Liu, B., Zhang, Q., Liang, J., Suganthan, P., and Qu, B., 2014. Problem definitions and evaluation criteria for CEC 2015 special session on bound constrained single-objective computationally expensive numerical optimization, Technical report.
157. Lopes, R.H.C., 2011. Kolmogorov-Smirnov Test, 718–720, Springer Berlin Heidelberg, Berlin, Heidelberg.
158. Kruskal, W.H. and Wallis, W.A., 1952. Use of Ranks in One-Criterion Variance Analysis, **Journal of the American Statistical Association**, **47**(260):583–621.
159. Mann, H.B. and Whitney, D.R., 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other, **Ann. Math. Statist.**, **18**(1):50–60.
160. Ong, E.J., Cooper, H., Pugeault, N., and Bowden, R., 2012. Sign language recognition using sequential pattern trees, *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2200–2207, IEEE.

161. Chai, X., Wang, H., and Chen, X., 2014. The design large vocabulary of Chinese sign language database and baseline evaluations, Technical report.
162. Ng, A., 2018. Machine Learning Yearning, deeplearning.ai.
163. Ozcan, T. and Basturk, A., 2020. Human action recognition with deep learning and structural optimization using a hybrid heuristic algorithm, **Cluster Computing**, **Online First**:1–14.
164. Kolosnjaji, B. and Eckert, C., 2015. Neural Network-Based User-Independent Physical Activity Recognition for Mobile Devices, *Intelligent Data Engineering and Automated Learning – IDEAL 2015*, 378–386.
165. ULAKBİM, T., Türk Ulusal Bilim e-Altyapısı, [https : //www.truba.gov.tr](https://www.truba.gov.tr), Çevrimiçi: Son erişim 10 Ekim 2019.
166. Ozcan, T. and Basturk, A., 2019. Static Image-based Emotion Recognition Using Convolutional Neural Network, *2019 Signal Processing and Communications Applications (SIU)*, 1–4.
167. Ozcan, T. and Basturk, A., 2019. Lip Reading Using Convolutional Neural Networks with and without Pre-Trained Models, **Balkan Journal of Electrical and Computer Engineering**, 7(2):195–201.
168. Shan, C., Gong, S., and McOwan, P.W., 2009. Facial expression recognition based on local binary patterns: A comprehensive study, **Image and Vision Computing**, 27(6):803–816.
169. Liu, P., Han, S., Meng, Z., and Tong, Y., 2014. Facial expression recognition via a boosted deep belief network, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1805–1812.
170. Mollahosseini, A., Chan, D., and Mahoor, M.H., 2016. Going deeper in facial expression recognition using deep neural networks, *2016 IEEE Winter conference on applications of computer vision (WACV)*, 1–10, IEEE.
171. The Mathworks Inc., 2018. Deep Learning Toolbox (R2018b), Natick, Massachusetts.

172. Freepik, Asian Man Graphics, <https://www.freepik.com/free-photo/cheerful-old-asian-man-celebrating2861219.html>, Çevrimiçi: Son erişim 25 Ekim 2018.
173. Aghamaleki, J.A. and Ashkani Chenarlogh, V., 2019. Multi-stream CNN for facial expression recognition in limited training data, **Multimedia Tools and Applications**, **78**(16):22861–22882.
174. Basturk, A., Basturk, N.S., and Qurbanov, O., 2018. Fingerprint recognition by deep neural networks and fingercodes, *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 1–4.
175. Poursaberi, A., Noubari, H.A., Gavrilova, M., and Yanushkevich, S.N., 2012. Gauss–Laguerre wavelet textural feature fusion with geometrical information for facial expression identification, **EURASIP Journal on Image and Video Processing**, **2012**(1):1–13.
176. Zavaschi, T.H., Britto, A.S., Oliveira, L.E., and Koerich, A.L., 2013. Fusion of feature sets and classifiers for facial expression recognition, **Expert Systems with Applications**, **40**(2):646 – 655.
177. Gogić, I., Manhart, M., Pandžić, I.S., and Ahlberg, J., 2018. Fast facial expression recognition using local binary features and shallow neural networks, **The Visual Computer**, 1–16.
178. Moeini, A., Faez, K., Moeini, H., and Safai, A.M., 2017. Facial expression recognition using dual dictionary learning, **Journal of Visual Communication and Image Representation**, **45**:20–33.
179. Luo, Y., Liu, X.y., Zhang, Y., Chen, X.f., and Chen, Z., 2019. Facial expression recognition based on improved completed local ternary patterns, **Optoelectronics Letters**, **15**(3):224–230.
180. Yildirim, M.T., Basturk, A., and Yuksel, M.E., 2007. A Detail-Preserving Type-2 Fuzzy Logic Filter for Impulse Noise Removal from Digital Images, *2007 IEEE International Fuzzy Systems Conference*, 1–6.

181. Agrawal, S., Omprakash, V.R., and Ranvijay, 2016. Lip reading techniques: A survey, *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, 753–757.
182. Rekik, A., Ben-Hamadou, A., and Mahdi, W., 2015. Human Machine Interaction via Visual Speech Spotting, *S. Battiato, J. Blanc-Talon, G. Gallo, W. Philips, D. Popescu, and P. Scheunders, Eds.; Advanced Concepts for Intelligent Vision Systems*, 566–574, Springer International Publishing, Cham.
183. Rekik, A., Ben-Hamadou, A., and Mahdi, W., 2015. Unified System for Visual Speech Recognition and Speaker Identification, *S. Battiato, J. Blanc-Talon, G. Gallo, W. Philips, D. Popescu, and P. Scheunders, Eds.; Advanced Concepts for Intelligent Vision Systems*, 381–390, Springer International Publishing, Cham.
184. Fernandez-Lopez, A. and Sukno, F.M., 2018. Survey on automatic lip-reading in the era of deep learning, ***Image and Vision Computing***, **78**:53 – 72.
185. Feng, W., Guan, N., Li, Y., Zhang, X., and Luo, Z., 2017. Audio visual speech recognition with multimodal recurrent neural networks, *2017 International Joint Conference on Neural Networks (IJCNN)*, 681–688.
186. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A.Y., 2011. Multimodal deep learning, *Proceedings of the 28th international conference on machine learning (ICML-11)*, 689–696.
187. Feng, W., Guan, N., Li, Y., Zhang, X., and Luo, Z., 2017. Audio visual speech recognition with multimodal recurrent neural networks, *2017 International Joint Conference on Neural Networks (IJCNN)*, 681–688.

8. BÖLÜM

EKLER

A - Sosyal ve Beşeri Bilimler Etik Kurulu Onay Formu

BAŞVURU NO: 113

**SOSYAL VE BEŞERİ BİLİMLER ETİK KURULU
PROJE ONAY FORMU**

Projenin Adı	"Derin Öğrenme Teknikleri ile İnsan Edimlerinin Tanınması"
Projenin Niteliği	Doktora Tezi
Proje Araştırmacıları	Arş. Gör. Tayyip ÖZCAN (Sorumlu Araştırmacı) Doç. Dr. Alper BAŞTÜRK (Danışman)
Sorumlu Araştırmacının Haberleşme Bilgileri	Erciyes Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölüm Başkanlığı, 38039 Talas/KAYSERİ E-posta adresi: tozcan@erciyes.edu.tr

KARAR:

Etik Kurulumuza başvuran **Tayyip ÖZCAN'ın "Derin Öğrenme Teknikleri ile İnsan Edimlerinin Tanınması"** adlı Çalışması değerlendirilerek aşağıdaki sonuca ulaşılmıştır.

Proje etik açıdan uygun bulunmuştur.

Projenin etik açıdan geliştirilmesi gerekmektedir.

Proje etik açıdan uygun bulunmamıştır.

25/12/2018

ADI SOYADI	İMZA
Etik Kurul Başkanı	Prof. Dr. Mustafa ARGUNŞAH
Etik Kurul Başkan Yrd.	Doç. Dr. Kasım KARAMAN
Üye (Raportör)	Prof. Dr. Celal YILDIZ
Üye	Prof. Dr. Mehmet AKKURT
Üye	Prof. Dr. Mustafa BAKTIR
Üye	Prof. Dr. Mustafa DEMİRCİ
Üye	Prof. Dr. Mustafa AKDAĞ
Üye	Doç. Dr. Handan ZİNCİR
Üye	Doç. Dr. Burak ADIGÜZEL

Şekil 8.1. Veri seti oluşturulması için gerekli etik kurulu izni belgesi.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı, Soyadı : Tayyip ÖZCAN
Uyruğu : Türkiye (T.C.)
Doğum Tarihi ve Yeri : 01.06.1986 Yahyalı/KAYSERİ
Telefon : 0 352 207 66 66
Belgegeçer : 0 352 437 57 84
E-posta : tozcan@erciyes.edu.tr
Adres : Erciyes Üniversitesi Mühendislik Fakültesi
 Bilgisayar Mühendisliği Bölümü
 38039, Melikgazi KAYSERİ TÜRKİYE

EĞİTİM

Derece	Kurum	Mez.Yılı
Lise	Nuh Mehmet Baldöktü Anadolu Lisesi, KAYSERİ	2005
Lisans	Kültür Üni. Müh. ve Mim. Fakültesi Bilgisayar Müh., İSTANBUL	2010
Yüksek Lisans	Erciyes Üni. Fen Bilimleri Ens. Bilgisayar Müh. ABD, KAYSERİ	2016
Doktora	Erciyes Üni. Fen Bilimleri Ens. Bilgisayar Müh. ABD, KAYSERİ	2020

İŞ DENEYİMİ

Yıl	Kurum	Görev
2011-2011	Şahin Yazılım, KAYSERİ	Yazılım Destek
2011-2013	Merkez Çelik A.Ş., KAYSERİ	Yazılımcı
2013-Halen	ERÜ Mühendislik Fakültesi Bilgisayar Müh. Bölümü, KAYSERİ	Araştırma Görevlisi

YAYINLAR

SCI, SSCI, AHCI İndekslerine Giren Dergilerde Yayınlanan Makaleler

- **T. Ozcan**, A. Basturk, “Human action recognition with deep learning and structural optimization using a hybrid heuristic algorithm”, Cluster Computing , (Online first), pp. 1-14, 2020.
- **T. Ozcan**, A. Basturk, “Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition”, Neural Computing and Applications , vol. 31(12), pp. 8955-8970, 2019.

TR Dizinli Dergilerde Yayınlanan Makaleler

- **T. Ozcan**, A. Basturk, “Lip Reading Using Convolutional Neural Networks with and without Pre-Trained Models”, Balkan Journal of Electrical and Computer Engineering, vol. 7(2), pp.195-201, 2019.

Diğer Dergilerde Yayınlanan Makaleler

- **T. Ozcan**, A. Basturk, “Intelligent Optimization Algorithms Aided Kalman Filter Parameters Optimization and Statistical Analysis”, Sinop Üniversitesi Fen Bilimleri Dergisi, vol. 1, pp.1-15, 2016.
- S. Aslan, H. Badem, **T. Ozcan**, D. Karaboğa, A. Basturk, “Image Compression with Multi-GPU Accelerated Discrete Haar Wavelet Transform”, Kahramanmaraş Sutcu Imam University Journal of Engineering Sciences, vol. 18, pp.12-16, 2015.

Uluslararası Katılımlı Konferans Yayınları

- **T. Ozcan**, A. Basturk, “Static Image-Based Emotion Recognition Using Convolutional Neural Network”, 27th Signal Processing and Communications Applications Conference (SIU-2019), pp. 1-4, Sivas, Turkey, 2019.

- **T. Ozcan**, H. Badem, A. Basturk, “Artificial Bee Colony Algorithm Based Parameter Tuning of Kalman Filter for Object Tracking”, International Conference on Information Complexity and Statistical Modeling in High Dimensions with Applications (IC-SMHD-2016), Nevsehir, Turkey, 2016.
- **T. Ozcan**, A. Basturk, “Determination of Kalman Filter Parameters by Using Intelligent Optimization Algorithms for Object Tracking”, “Kalman Süzgeci Parametrelerinin Nesne Takibi Amacıyla Zeki Optimizasyon Algoritmalarıyla Belirlenmesi”, 1st International Conference on Engineering Technology and Applied Sciences (ICETAS-2016), pp. 1392-1396, Afyon, Turkey, 2016.
- S. Aslan, H. Badem, D. Karaboga, A. Basturk, **T. Ozcan**, “Accelerating Discrete Haar Wavelet Transform on GPU Cluster”, 9th International Conference on Electrical and Electronics Engineering (ELECO-2015), pp. 1237-1240, Bursa, Turkey, 2015.

Ulusal Konferans Yayınları

- S. Aslan, H. Badem, **T. Ozcan**, A. Basturk, D. Karaboga, “CUDA ile Hızlandırılmış İki Boyutlu Ayrık Kosinüs Dönüşümü”, 4. Ulusal Yüksek Başarımlı Hesaplama Konferansı, Ankara, Türkiye, 2015.
- S. Aslan, H. Badem, **T. Ozcan**, A. Basturk, “CUDA Platformunda İki Boyutlu Ayrık Haar Dalgacık Dönüşümü”, Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu (ASYU-2014), pp. 219-222, İzmir, Türkiye, 2014.

SCI Dizinli Dergi Sistemlerine Yüklenen İncelemedeki Çalışmalar

- **T. Ozcan**, A. Basturk, “Performance Improvement of Pretrained Convolutional Neural Networks for Action Recognition”, (Submitted), pp. 1-12, 2019.
- **T. Ozcan**, A. Basturk, “Static Facial Expression Recognition Using Convolutional Neural Networks Based on Transfer Learning and Hyperparameter Optimization”, (Submitted), pp. 1-16, 2019.
- **T. Ozcan**, A. Basturk, “ERUSLR: yeni bir Türkçe işaret dili veri seti ve evrişimli sinir ağı ile tanınması”, (Submitted), pp. 1-19, 2019.