T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÜÇ BOYUTLU İSKELET VERİLERİNDEN METRİK ÖĞRENME TABANLI HAREKET TANIMA

ŞEYMA YÜCER
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

GEBZE
2018

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÜÇ BOYUTLU İSKELET VERİLERİNDEN METRİK ÖĞRENME TABANLI HAREKET TANIMA

ŞEYMA YÜCER
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
PROF. DR. YUSUF SİNAN AKGÜL

GEBZE
2018

T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

METRIC LEARNING BASED ACTION RECOGNITION FROM THREE DIMENSIONAL SKELETON DATA

ŞEYMA YÜCER

A THESIS SUBMITTED FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF COMPUTER ENGINEERING

THESIS SUPERVISOR

PROF. DR. YUSUF SİNAN AKGÜL

GEBZE

2018

GEBZE TEKNİK ÜNİVERSİTESİ	**YÜKSEK LİSANS JÜRİ ONAY FORMU**

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 17/01/2018 tarih ve 2018/04 sayılı kararıyla oluşturulan jüri tarafından 09/02/2018 tarihinde tez savunma sınavı yapılan Şeyma Yücer'in tez çalışması Bilgisayar Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE
(TEZ DANIŞMANI): Prof. Dr. Yusuf Sinan AKGÜL

ÜYE : Doç. Dr. Behçet Uğur Töreyin

ÜYE : Yrd. Doç. Dr. Yakup Genç

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ...…/……/…… tarih ve ……./…..... sayılı kararı.

İMZA/MÜHÜR

Doç. Dr. Arif Çağdaş AYDINOĞLU
Gebze Teknik Üniversitesi
Fen Bilimleri Enstitüsü Müdürü

ÖZET

İnsan hareketlerinin analizi bilgisayarla görme alanının önemli problemlerinden biridir. Bu problem, hareketlerin bilgisayar tarafından analiz edilmesi olarak tanımlanabilir. Hareketlerin tanınması, fizik tedavi, güvenlik, eğlence ve biyometri gibi pek çok alana katkı sağlayacaktır. Literatür çalışmaları, günlük (yeme, içme, oturma vb.) ya da spor (koşma, dalma, bisiklet sürme vb.) hareketlerin tanınması, hasta hareketlerinin analizi, biyometrik hareket verileri ile kişi tanıma veya gözetim amaçlı şüpheli hareket tespiti için yöntemler sunmaktadır.

İnsan hareketi verileri 2B veya 3B görüntülerden elde edilmektedir. 3B veriler, 2B verilere ek olarak görüntüdeki piksellerin derinlik bilgilerini de içermektedir. 3B hareket görüntüleri üzerinde bulunan iskelet eklem koordinatları, hareketleri daha verimli ve daha doğru ifade etmektedir.

Bu tez kapsamında, 3B iskelet görüntüleri kullanılarak insan hareketlerinin analizi için iki ayrı yöntem önerilmiştir. Bu yöntemlerden ilki, hareketler için gösterim tabanlı bir çözüm sunmaktadır. Geometrik eklem çantası olarak adlandırdığımız yöntem, 3B iskelet görüntülerinin zamansal ve geometrik özniteliklerini çıkartıp, SoftMax yöntemi ile sınıflandırmaktadır.

Önerdiğimiz ikinci yöntem ise, derin ağ tabanlıdır. Tasarladığımız İkiz LSTM-DML ağı, eylemlerin birbiri ile olan ilişkisini öğrenerek hareketleri tanımaktadır. Ağ iki farklı hareketi girdi olarak almaktadır. Her bir hareketin ağdaki LSTM alt ağları sayesinde zamansal öznitelikleri çıkarılmaktadır. Çift yönlü olarak parametre paylaşımı ile eğitilen ağ, hareketlerin derin metriklerini öğrenmektedir. Böylece baştan sona çalışan ağ, derin metrikleri kullanarak hareketleri sınıflandırabilmektedir. Sınıflandırma dışında hareketlerin benzerliklerini çıkarabilen yöntem diğer çalışmalara kıyasla daha genelleştirilebilirdir.

Bu tez için iç mekânda spor ve gündelik hareketlerden oluşan GTU Action 3D veri kümesi oluşturulmuştur. Yöntemlerimiz kendi veri kümemize ek olarak Florence Action 3D, Microsoft, NTU RGB+D veri kümeleri üzerinde test edilmiş ve literatür çalışmaları ile karşılaştırılmıştır.

Anahtar Kelimeler: Hareket Tanıma, Öznitelik Çıkarımı, İkiz Ağlar, Yapay Sinir Ağları, Derin Metrik Öğrenme

SUMMARY

Human action recognition, one of the crucial problems of the computer vision, is analyzing human actions via computers. Action recognition affects many research areas such as physiotherapy, surveillance, multimedia, biometrics and so on. Related works mainly focus on the patients' actions analysis, person recognition, surveillance for suspicious movements detection and daily activities (such as eating, drinking, sitting, etc.) and sports actions (such as running, diving, cycling, etc.) recognition.

Human action data is acquired from 2D and 3D images. In addition to 2D images, 3D images contain the depth information for each pixel. Skeleton joint coordinates on 3D images provides a more efficient and more reliable way to represent human actions.

In this work, we present two different methods for analyzing human actions on 3D skeleton images. The first method is a representation-based solution for human action recognition problem. This method, called the geometric bag of joints based human action recognition, utilizes the spatial-temporal behavior of 3D skeleton frames and uses SoftMax regression method to classify human actions.

The other method is a deep neural network-based method. For this method, we designed a Siamese LSTM-DML network which learns the relationship between actions and recognizes human actions using this relationship information. For each action, sub-LSTM networks extract the temporal features of human actions. Using these features, the network is trained with two-way parameter sharing and learns the deep metrics of actions. Therefore, the end-to-end trainable network can classify human actions. Unlike the other method, this method is more generalizable and can learn the relationship between human actions.

As a part of this study, we created a GTU Action 3D dataset that contains daily indoor activities and indoor sports actions. Our methods are tested against Florence Action 3D, MSR Action3D, NTU RGB+D datasets as well as our own dataset. Also, we compared our methods to related works.

Key Words: Action Recognition, Feature Extraction, Siamese Networks, Neural Networks, Deep Metric Learning

TEŞEKKÜR

Lisans ve yüksek lisans eğitimim boyunca yardımları, bilgileri ve tecrübeleri ile bana sürekli destek olan, tez çalışmam boyunca ilgisini, yönlendirmelerini esirgemeyen danışmanım Prof. Dr. Yusuf Sinan AKGÜL hocama, manevi destekleri ve bana olan inançları için aileme, her anımı paylaştığım, her zorlukta yanımda olan ve bu tezin yazımında sayısız yardımı olan sevgili eşim Furkan Tektaş'a en içten teşekkürlerimi sunarım.

İÇİNDEKİLER

	Sayfa
ÖZET	viii
SUMMARY	ix
TEŞEKKÜR	x
İÇİNDEKİLER	xi
SİMGELER ve KISALTMALAR DİZİNİ	xii
ŞEKİLLER DİZİNİ	xiv
TABLOLAR DİZİNİ	xv
1. GİRİŞ	1
2. LİTERATUR ÇALIŞMALARI	4
2.1. 2B Veriler Kullanılarak Yapılan Çalışmalar	*5*
2.1.1. 2B Temsil Tabanlı Çalışmalar	5
2.1.2. Derin Ağ Tabanlı Çalışmalar	7
2.2. 3B Veriler Kullanılarak Yapılan Çalışmalar	*9*
2.2.1. 3B Derinlik Haritaları Tabanlı Çalışmalar	9
2.2.1.1. Gösterim Tabanlı Çalışmalar	9
2.2.1.2. Derin Ağ Tabanlı Çalışmalar	11
2.2.2. İskelet Tabanlı Çalışmalar	11
3. 3 BOYUTLU HAREKET VERİLERİ	15
3.1. Veri Kümeleri	*15*
3.1.1. GTU Action 3D	15
3.1.2. Florence Action 3D	17
3.1.3. MSR Action3D	18
3.1.4. NTU RGB+D	19
3.2. İskelet Görüntüleri Üzerinde Normalleştirme Yöntemleri	*21*
3.2.1. Normalleştirme Yöntemleri	23
3.2.1.1. İskelet Görüntü Normalleştirmesi:	23
3.2.1.2. Veri Kümesi Normalleştirmesi:	23
4. GEOMETRİK EKLEM ÇANTASI YÖNTEMİ İLE HAREKET TANIMA	24

4.1. Yöntem *25*
4.1.1. Geometrik Öznitelikler 25
4.1.2. Kelime Çantası Yöntemi 27
4.1.3. Hareketlerin Sınıflandırılması 30
4.2. Sonuçlar 30
5. OTO KODLAYICILAR İLE HAREKETLERİN GÖSTERİMLERİNİ OLUŞTURMA 32
5.1. Oto Kodlayıcılar *32*
5.2. Yöntem ve Sonuçlar *34*
6. DERİN AĞLAR İLE HAREKETLERİN SINIFLANDIRILMASI 37
6.1. SoftMax Sınıflandırıcı *37*
6.2. Çok Katmanlı Yapay Sinir Ağları MLP *38*
6.3. Aşırı Öğrenme Makinası (ELM: Extreme Learning Machine) *38*
6.4. LSTM Ağları *39*
6.5. Sınıflandırıcıların Sonuçları *40*
7. İKİZ LSTM AĞLARI İLE HAREKET TANIMA 43
7.1. İkiz LSTM Ağı *44*
7.2. Hareket İkililerinin Oluşturulması *45*
7.3. İkiz LSTM Ağları için Sınıflandırıcı Modülü *46*
7.4. Uçtan Uca İkiz LSTM DML Sınıflandırıcı Ağı *46*
7.5. Sonuçlar 51
7.5.1. GTU Action 3D Sonuçları 51
7.5.2. Florence Action 3D Sonuçları 54
8. SONUÇLAR 57
KAYNAKLAR 61
ÖZGEÇMİŞ 70
EKLER 71

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler ve Kısaltmalar		Açıklamalar
2B	:	2 Boyutlu
3B	:	3 Boyutlu
Vs.	:	Versiyon
Vb.	:	Ve benzeri
LSTM	:	Uzun Kısa Süreli Hafıza (Long Term Short Term Memory)
CNN	:	Evirişimsel Sinir Ağları (Convolutional Neural Networks)
RNN	:	Tekrarlayan Sinir Ağları (Recurrent Neural Networks)
STV	:	Uzay Zaman Hacmi (Space Time Volume)
RGB+D	:	Kırmızı Yeşil Mavi Derinlik (Red Green Blue Depth)
NERF	:	Non-Euclidean Relational Fuzzy
KYM	:	Kırmızı Yeşil Mavi
ReLU	:	Rectified Linear Unit
YGK	:	Yazılım Geliştirme Kiti
TBA	:	Temel Bileşen Analizi
YSA	:	Yapay Sinir Ağı
ÇSS	:	Çok Sınıflı Sınıflandırıcı
FC	:	Tam Bağlı (Fully Connected)
DBN	:	Derin Kanı Ağları (Deep Belief Network)

ŞEKİLLER DİZİNİ

Şekil No:		Sayfa
1.1:	Hareket verilerinin çeşitleri ve verilerin toplandığı cihazlar.	2
2.1:	Literatür çalışmalarının veri yapısına ve yöntemine göre gruplanması.	6
2.2:	2B veriler kullanılarak yapılan çalışmaların gruplanması.	7
2.3:	3B veriler kullanılarak yapılan çalışmaların gruplanması.	11
3.1:	Florence Action 3D veri kümesinden örnek görüntüler.	18
3.2:	MSR Action3D veri kümesinden örnek görüntüler.	19
3.3:	NTU RGB+D veri kümesinden örnek görüntüler.	20
3.4:	Veri kümeleri iskelet görünümleri.	23
4.1:	Özniteliklerin gösterimi a. Eklem koordinatları, b. Eklem koordinat farkları, c. Eklem doğru uzaklıkları.	29
4.2:	Kelimelerin oluşturulması.	30
5.1:	2 Katmanlı Oto kodlayıcı genel görünümü.	34
5.2:	Oto kodlayıcı LSTM yönteminin genel görünümü.	35
5.3:	Test anında iki iskelet görüntüsünün oto kodlayıcı girdileri ve çıktıları sağ taraf girdiler, sol taraf çıktıları.	36
5.4:	Test anında üst kısımdaki oturma ve alt kısımdaki saç tarama hareketlerinin oto kodlayıcı girdileri ve çıktıları.	37
6.1:	Çok Katmanlı Yapay Sinir Ağı ile hareketlerin sınıflandırılması.	39
6.2:	LSTM ağları ile hareketlerin sınıflandırılması.	41
7.1:	Hareket ikilileri arasındaki benzerliği bulan İkiz LSTM ağı.	45
7.2:	İkiz LSTM DML sınıflandırıcı ağı genel gösterimi.	49
7.3:	Uçtan Uca İkiz LSTM DML sınıflandırıcı ağı genel gösterimi.	51
7.4:	GTU Action 3D İkiz LSTM Ağı ile İkiz LSTM DML kıyaslama.	54
7.5:	GTU Action 3D veri kümesinin İkiz-LSTM DML ağı karışıklık matrisi.	55
7.6:	Florence Action 3D İkiz LSTM Ağı ile İkiz LSTM DML kıyaslama.	57

TABLOLAR DİZİNİ

Tablo No:		Sayfa
2.1:	2B verilerle yapılan derin ağ tabanlı çalışmalar.	8
2.2:	3B derinlik haritaları ile yapılan derin ağ tabanlı çalışmalar.	12
2.3:	3B iskelet verileri ile yapılan derin ağ tabanlı çalışmalar.	14
3.1:	3B insan hareketi veri kümeleri.	16
3.2:	GTU Action 3D veri kümesi hareket sınıfları.	16
3.3:	Florence Action 3D veri kümesi hareket sınıfları.	18
3.4:	Microsoft Action 3D veri kümesi hareket sınıfları.	18
3.5:	NTU RGB+D veri kümesi hareket sınıfları.	20
4.1:	Öznitelik gösterimleri ve formülleri.	27
4.2:	Özniteliklerin veri kümelerine göre başarımları.	31
6.1:	Sınıflandırıcıların veri kümeleri başarımları.	41
7.1:	GTU Action 3D veri kümesi üzerinde elde edilen başarımlar.	52
7.2:	Florence Action 3D başarımlarının karşılaştırılması.	55

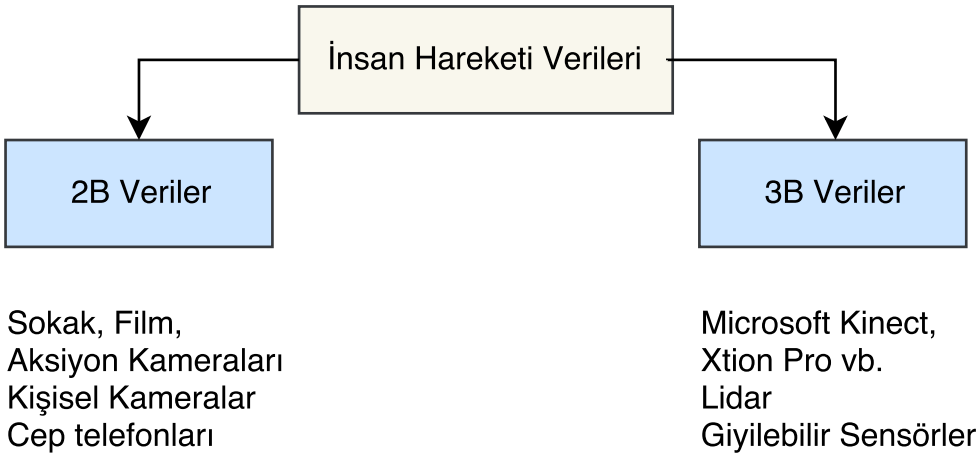
1. GİRİŞ

İnsan hareketlerinin analizi çalışmaları, son 20 yılda kameraların hayatımızdaki rolünün artışına paralel olarak önem kazanmıştır. Hareketlerimizin bilgisayarlar tarafından tanınmasını sağlayan uygulamalar, sosyal yaşantımızı önemli oranda [1] etkilemektedir. Örneğin, sağlık alanında, fiziksel hastalıkları ve duruş bozukluklarını düzeltmek için hastalara iyileştirici tedaviler bu alandaki çalışmalar [2], [3] ile sağlanmaktadır. Otonom robot ve araçlarda çevredeki insan hareketlerini tanıyan çalışmalar, [4]–[6] robotların ve araçların karar mekanizmalarını etkilemektedir. Sokakta, bankada veya toplu ortamlarda uygunsuz ve yasal olmayan hareket tespitinde [7], film, dizi veya video üzerinde hareket tabanlı özet çıkarmada [8], [9] yine insan hareketlerinin tanınması önemli bir rol oynamaktadır. Hareketlerimiz aynı zamanda biyometrik olduğu için biyometri alanında da hareket tanıma [10], [11] çalışılabilmektedir. Hareket tanıma konusunda yapılmış literatür çalışmalarından Bölüm 2'de daha detaylı bahsedilecektir. Bu konu aynı zamanda poz hesaplama, yüz tanıma ve duygu analizi gibi alt konulara da ayrılabilmektedir.

Tüm bu çalışmalar 2B ve 3B veriler üzerinde gerçekleştirilmektedir.

Şekil 1.1'de bu verileri toplayan cihazlar gösterilmiştir. 2B veriler kameralardan alınan görüntüler veya görüntü dizileridir. Bunlar kişisel kameralar, cep telefonları veya özel amaçlar için üretilen kameralar olabilir. Ayrıca bazı özel kameralar hem 2B hem 3B görüntü verebilmektedir. Örneğin film çekimlerinde kullanılan bazı kameralar, 2B görüntülerin yanında stereo özelliği ile 3B geri çatımda yapabilmektedir [12].

3B veriler ise genellikle derinlik kameraları ile sağlanmaktadır. Derinlik kameralarının maliyeti hızla gelişen teknolojilerle birlikte düşmektedir. 3B derinlik kameraları KYMD, birbirine dik 3 farklı eksende değerlere sahip görüntüler sunmaktadır. Derinlik kameralarının yanı sıra, giyilebilir sensörler ve cep telefonu sensörleri ile de 3B hareket verileri toplanabilmektedir.



Şekil 1.1: Hareket verilerinin çeşitleri ve verilerin toplandığı cihazlar.

Bu tez çalışmasında, 3B veriler kullanılarak hareket analizi ve tanıma yapılmıştır. Hareket analizi için başarılı hareket gösterimleri oluşturma ve hareket metriklerini bulma tabanlı çalışmalar önerilmiştir.

Tez kapsamında Microsof Kinect 2 cihazı ile 508 hareketten oluşan GTU Action 3D iskelet veri kümesi toplanmıştır. 10 farklı kişiden oluşan veri kümesi, evde gerçekleştirilebilecek günlük spor hareketlerini barındırmaktadır. GTU Action 3D veri kümesine ek olarak literatürde kullanılan farklı veri kümeleri de tekniklerimizin test edilmesinde kullanılmıştır. Tez dahilindeki tüm çalışmalar, iskelet eklem koordinat verileri kullanılarak yapılmıştır. İskelet eklem koordinatları derinlik sensörleri tarafından gerçek zamanlı olarak elde edilebilmektedir [13]. Kullanılan veri kümeleri ve iskelet verileri ile ilgili detaylı bilgiler Bölüm 3'te verilecektir.

Bölüm 4'te Geometrik Eklem Çantası yönteminden bahsedilmektedir. Bu çalışmada, iskelet görüntülerine ait gösterim oluşturma amacıyla çeşitli geometrik öznitelikler denenmiş ve bulunmuştur. Geometrik öznitelikler, iskelet yapısına ait uzamsal bilgiler ve vücut bölgelerindeki ilişkisel bilgileri içermektedir. Hareketlerden çıkarılan öznitelikler zaman dilimlerine göre parçalanmış ve kelime çantası yöntemi için kelimeler haline getirilmiştir. Kelimelerin histogram vektörleri hesaplanıp sınıflandırılmıştır. Tezin bu çalışması gösterim tabanlı çalışmalardan [14] 'ü ve [15]'i baz alarak gerçekleştirilmiştir.

Bölüm 5'te Derin ağ yapılarından olan Oto kodlayıcılar kullanılarak hareket gösterimleri oluşturulmaya çalışılmıştır. Bölüm 6'da temel derin ağları sınıflandırıcı olarak kullanarak hareketler sınıflandırılmış ve ağlar sonuçlara bakılarak karşılaştırılmıştır. Karşılaştırılma sonuçları sonraki çalışmaları etkilemiştir.

Bölüm 7'de hareketlerin analizi ve sınıflandırması için oluşturduğumuz İkiz LSTM ağları anlatılmıştır. Derin metrik öğrenme tabanlı çalışmanın [16] amacı hareketler arasındaki metrikleri otomatik olarak öğrenen özgün bir model oluşturmaktır. Tasarlanan model İkiz LSTM-DML, verilen iki hareket girdisi arasındaki ilişkiye ait derin metrikleri çıkarmaktadır. Hareketler ikiz ağda karşılıklı kıyaslanırken model eğitilmektedir. Hareketlerin zamansal ve uzamsal özniteliklerini modelin iç yapısı otomatik olarak çıkarmaktadır. Model, sadece hareket tanıma için değil farklı görüntü dizilerinde değişik amaçlar için de kullanılabilecektir.

Son bölümde paylaşılan çalışmaların sonuçlarında görülmüştür ki paylaşımlı ağlar sadece doğrulama problemleri için değil sınıflandırma problemleri için de başarılı bir şekilde çalışabilmektedir. Yapının derin metrikleri veriye uygun bir ağ ile başarılı bir şekilde çıkarılırsa bu metrikler sonrasında farklı uygulamalar içinde kullanılabilmektedir. Tezin sonrasında yapılabilecek çalışmalar; bu modelin uçtan uca sınıflandırıcı olarak çalışan yapısının iyileştirilmesi, farklı veri kümelerinin bir arada kullanılması ve öğrenilmemiş bir hareketin sınıflandırılmaya çalışılması şeklinde sıralanabilir.

2. LİTERATÜR ÇALIŞMALARI

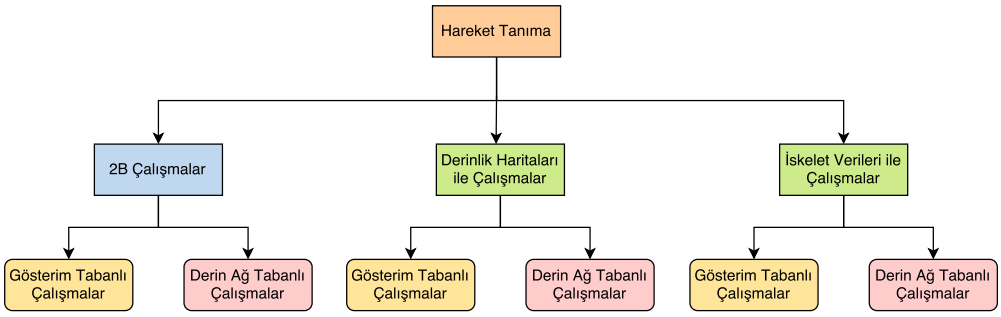
İnsana ait hareketlerin tanınması son 20 yıldır aktif olarak çalışılan bilgisayarla görme konularından biridir. Hareket tanıma problemi, 2B ve 3B veriler üzerinde nesne tanıma, insan-makina etkileşimi, video izleme, alan adaptasyonu ve semantik bölümleme gibi birçok tamamlayıcı araştırma alanındaki gelişmelere bağlıdır [17]. Bu gelişmeleri hızla takip edip ortaya bu konuda yeni bakış açıları sunan ve başarılı yöntemler geliştiren pek çok çalışma mevcuttur. Bu çalışmalar sağlık, güvenlik, eğitim, eğlence ve robotik gibi önemli birçok alana katkı sağlamaktadır [18], [19].

Eylem tanıma çalışmaları, benzer hareketler arasındaki farkları bulup, farklı yerde ve sürede gerçekleştirilen aynı hareketleri tespit edebilmektedir. Farklı boyda ve kiloda insanlar tarafından gerçekleştirilen günlük, sportif, artistik veya etkileşimli hareketler yüksek doğrulukla sınıflandırılabilmektedir [20], [21].

Bu bölümde, 2B ve 3B verileri kullanarak hareket tanıma yapan çalışmalardan bahsedilecektir. Literatürde hareket tanıma ile ilgili oldukça fazla sayıda çalışma mevcuttur. Bu çalışmaları, açık bir şekilde anlatmak için kullanılan verilerin yapısına ve sunulan yöntemlere göre gruplandırdık.

Şekil 2.1'de hareket tanıma problemi için yapılmış çalışmalar gruplanmıştır. Çalışmalar önce kullandıkları verilerin boyutuna göre 2B ve 3B olarak ayrılmıştır. 3B verilerle yapılan çalışmalar derinlik haritaları ve iskelet verilerini kullanmalarına göre bir alt gruba ayrılmıştır. Sonrasında çalışmalar, kullanılan tekniklere göre gruplanmış, gösterim tabanlı ve derin ağ tabanlı yöntemler ayrı başlıklarda incelenmiştir. Gösterim tabanlı ve derin ağ tabanlı yöntemleri sunan çalışmalar aynı verileri kullanmış olmalarına rağmen birbirlerinden oldukça farklı yaklaşımlar sergilemektedirler.

Bu tez kapsamında, 3B iskelet verileri kullanılarak hem gösterim tabanlı hem de derin ağ tabanlı yöntemler sunulmuştur.



Şekil 2.1: Hareket tanıma konusunda yapılmış literatür çalışmalarının veri yapısına ve yöntemine göre gruplanması.

2.1. 2B Veriler Kullanılarak Yapılan Çalışmalar

2B görüntüler, maliyetleri oldukça düşük kameralar tarafından çok amaçlı toplanmaktadır. Bu görüntüler üzerinde yapılan hareket tanıma çalışmaları her geçen gün büyük bir ilerleme göstermektedir [22]. Günümüzde, hemen hemen tüm günlük faaliyetler için milyonlarca videodan öğrenen gerçek zamanlı [23] çözümler mevcuttur. Ayrıca var olan Youtube videoları, Hollywood filmleri gibi farklı kaynaklar üzerinden seçilen hareketler orijinal tekniklerle [1] tanınabilmektedir. Trafik ve güvenlik kameraları gibi farklı amaçlar için kurulan sistemler, toplumdaki rahatsız edici davranışları otomatik olarak bulmak için bu çalışmalarda kullanılmaktadır [24].

Şekil 2.2'de bahsedilen çalışmalar yöntemlerine göre ayrılmıştır. İlk çalışmalar hareketin karakteristik özelliklerini çıkararak başarılı temsiller üretmeyi amaçlamıştır. Bu temsillerin başarısı sınıflandırma başarılarını da artırmaktadır. Son beş yılda derin ağ yapılarının büyük gelişmeleri ile çalışmalar, bu ağların dinamiklerinde iyileştirme yaparak kendi kendine temsiller çıkarıp sınıflandıran çözümler sunmaya doğru kaymıştır.

2.1.1. 2B Temsil Tabanlı Çalışmalar

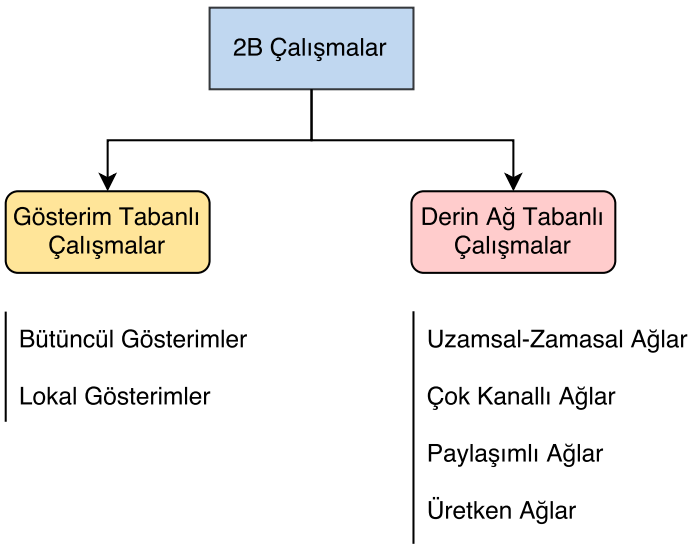
İnsan hareketini anlamak için öncelikle bir hareketin dinamiğini anlamak gerekir [25]. Hareketi ortaya çıkaran vücut yapısı, eklemlerin birlikte ve ayrı ayrı hareketleri, dinamiklerin sadece bir kısmıdır. Bu dinamiklerin başarılı bir şekilde temsil edilmesi ise, bu hareketin bilgisayar tarafından anlaşılmasını sağlamaktadır. Dinamiklerin ayrı ayrı ya da tümüyle ele alınmasına göre çalışmalar [17]' de sınıflandırılmıştır. Bütünsel temsiller, insan vücudunun yapısını, şeklini ve hareketlerinin temsilini

bütüncül bir şekilde çıkarılmasına dayanmaktadır. Lokal temsiller, bölgesel özniteliklerin çıkarılmasına dayanmaktadır.

Hareketin bütününü ele alan yöntemlerin birini eylemlerin diferansiyel özelliklerini çıkararak [26] yapmıştır. Uzay-Zaman Hacmi (STV) diye adlandırdığı sistem, nesne konturlarını zaman ekseni boyunca istifleyerek oluşturmaktadır.

[27] yaptığı çalışmada ise yer ve zamana bağlı MEI şablonları hesaplanır, bu şablonların oluşturduğu siluetler esasında bir 3B hareket şeklidir. Burada hareketleri sınıflandırmak için, oluşan 3B yüzeyin içindeki her sınır noktasının 2B haritası oluşturulmaktadır. Hareketlerin 3B hacimlerinin bulunması, farklı açılardan elde edilen görüntüler üzerinde yöntemin çalışmasını sağlamaktadır.

Bütünsel temsil araştırmaları, hareket tanımada, eylemlerin mekânsal ve zamansal yapısını korumaya daha fazla olanak sağlamaktaydı. Ancak son yıllarda araştırmalar daha çok lokal temsillere ve derin ağlar tabanlı tekniklere kaymıştır [28], [29]. Hareketlerin daha detaylı ve küçük parçalar halinde incelenmesinin, dinamiklerin çıkarılmasına daha faydalı olacağı doğrultusundaki yaklaşımlar öne sürülmüştür [30], [31].



Şekil 2.2: 2B veriler kullanılarak yapılan çalışmaların gruplanması.

Lokal temsiller vücudun bölümlerine, hareketin zaman dilimi içindeki parçalarına göre çıkarılabilmektedir. İki yönlü dikey değişikliklerin bulunması için Harris Köşe detektörlerini kullanan çalışmalar, [32] [33] hareketlerin köşelerdeki değişimlerini yakalamayı amaçlamıştır. Görüntülerden farklı olarak, hareket videoları daha kontrolsüz ortamlarda elde edilmektedir. Bu nedenle, önemli

öznitelikleri kaybetmemek için videoların işlenmesine özen gösterilmelidir. Örneğin, titreyen bir kamera, bir dizi alakasız noktaya odaklayabilir ve bunları kaydedebilir. Bu sorunu çözmek için [34], istatistiksel bir yöntem geliştirerek alakasız verileri budayan bir sistem önermiştir. Ayrıca, hareketin gerçekleştiği piksellere komşu pikseller, hareketin tanıma için işe yarar bilgiler barındırmaktadır. Bu nedenle o noktalardaki mekânsal ve zamansal verilerde çalışmada kullanılmıştır.

Hareketi en iyi ifade eden lokal temsiller, hareketin optik akışına bakarak [8], hiyerarşik olarak eylemleri bölerek [35] vb. tekniklerle bulunmuştur. Çalışmaların pek çoğunda 2B görüntülerin 3B yapılar haline getirildiği dikkat çekmektedir [27]. 3B yapısı, hareketin derinlik bilgilerinden faydalanmasına olanak sağlamaktadır. Bu da farklı açılarda oluşan ve kameraya doğru olan hareketlerin tespitini kolaylaştırmaktadır.

Bu çalışmalar devam ederken bu temsilleri otomatik olarak gerçekleştiren derin ağ tabanlı sistemler ortaya çıkmış, nesne tanıma, nesne takip etme gibi alanlarda başarılı sonuçlar elde edilmiştir [36]. Derin ağlar sayesinde, bir hareketin gerçekleştiği önemli anları ve önemli bölgelerin yerlerini yakalayan sistemler hareket verilerini birçok katmanda değerlendirebilmektedir.

2.1.2. Derin Ağ Tabanlı Çalışmalar

Derin ağ tabanlı çalışmalar kullanılan ağların yapısına göre 4'e ayrılmıştır. Ağın yapısal özelliklerine göre; Zamansal ve Mekânsal Ağlar, Çok Kanallı Ağlar, Üretken Ağlar ve Paylaşımlı Ağlar. Ağların türü, çalışmanın tüm yaklaşımına etki etmektedir. Tüm yöntemler; adı, kullanılan ağ mimarisi ve yılı ile Tablo 2.1'de gösterilmiştir.

Zamansal ve mekânsal ağlar [37]–[39], mekânsal bilgileri etkili bir şekilde çıkaran CNN'leri veya zamansal bilgiyi içerisinde tutarak öğrenen RNN'leri içermektedir. CNN'ler derin öznitelikleri başarılı bir şekilde çıkarabilirken, ilk katmanları düşük seviyeli özellikleri verebilmektedir. RNN'ler her bir zaman dilimindeki nöronların sağladığı bilgileri bir sonraki zaman dilimine aktararak hareket videolarının zamansal olarak anlaşılmasına olanak sağlar.

Çok kanallı ağlar [40] ise bu modelleri farklı aşamalarda kullanarak daha geniş bir yapı sunmaktadır. Bu yapılar şu amaçlarla kullanılmıştır.

- Ön eğitim yaparak, sınıflandırma başarısını artırmak,

- Birden fazla özniteliği ayrı ayrı öğrenip, hepsini birleştirmek,
- Hareketi ve dışındaki başka bir öğeyi de anlamlandırabilmek

Üretken ağlar [41], [42] denetimsiz öğrenme yapabilmek ve veri dengesini sağlamak için, paylaşımlı ağlar [43] ise hareketler arasındaki ilişkileri ve benzerlikleri bulabilmek için kullanılmıştır.

Tablo 2.1: 2B verilerle yapılan derin ağ tabanlı çalışmalar.

Çalışmanın Adı	Kullanılan Modeller	Yılı	Model Tipi
Sequential Deep Learning for Human Action Recognition [38]	CNN ve LSTM	2011	Uzamsal-Zamansal Ağlar
Convolutional Neural Networks for Human Action Recognition [39]	CNN	2013	
Long-Term Recurrent Convolutional Networks for Visual Recognition And Description [37]	LRCN: CNN ve LSTM	2016	
Two-Stream Convolutional Networks for Action Recognition In Videos [40]	Çoklu CNN	2014	Çok Kanallı Ağlar
Unsupervised Learning Of Video Representations Using Lstms [42]	LSTM KODLAYICI	2015	Üretken Ağlar
Deep Multi-Scale Video Prediction Beyond [41]	OTO-KODLAYICI	2015	
Extreme Low-Resolution Activity Recognition with Multi-Siamese Embedding Learning [43]	İkiz CNN	2017	Paylaşımlı Ağlar

Paylaşımlı ağlar her bir öğrenme adımında girdileri farklı olan ağların parametrelerini paylaşmasından ötürü bu isme sahiptir. İçerisinde kullanılan ağın öğrenme kapasitesine bağlı olarak hareket görüntüleri arasındaki benzerliği öğrenmeye çalışmaktadırlar.

Derin ağların çıkışından itibaren hızla gelişmesi ve umut verici sonuçların elde edilmesi hareket tanıma çalışmalarının yönünü değiştirmiştir. Büyük boyutta veriye ihtiyaç duyan derin ağlar, 2B ve 3B görsel hareket verileri için ideal yöntemler

sunmaktadır. Hareket tanımanın yanında, obje tanıma, poz hesaplama gibi birçok önemli problem bu ağlarda beraber çözülebilmektedir.

2.2. 3B Veriler Kullanılarak Yapılan Çalışmalar

Derinlik kameraları, 2009 yılından itibaren yaygınlaşmaya başlayan, üzerinde oldukça yoğun çalışılan bir teknolojidir. Azalan maliyetleri sayesinde hareket tanıma çalışmaları hızla 3B veriler üzerine kaymıştır. Ayrıca artık birçok sisteme çift kamera konularak stereo 3B görüntü elde edilebilmektedir.

RGB+D kameralar görüntü üzerindeki her pikselin kameraya olan uzaklık değerini de vermektedir. Bu şekilde derinlik değerlerinden oluşan görüntülere derinlik haritası denmektedir.

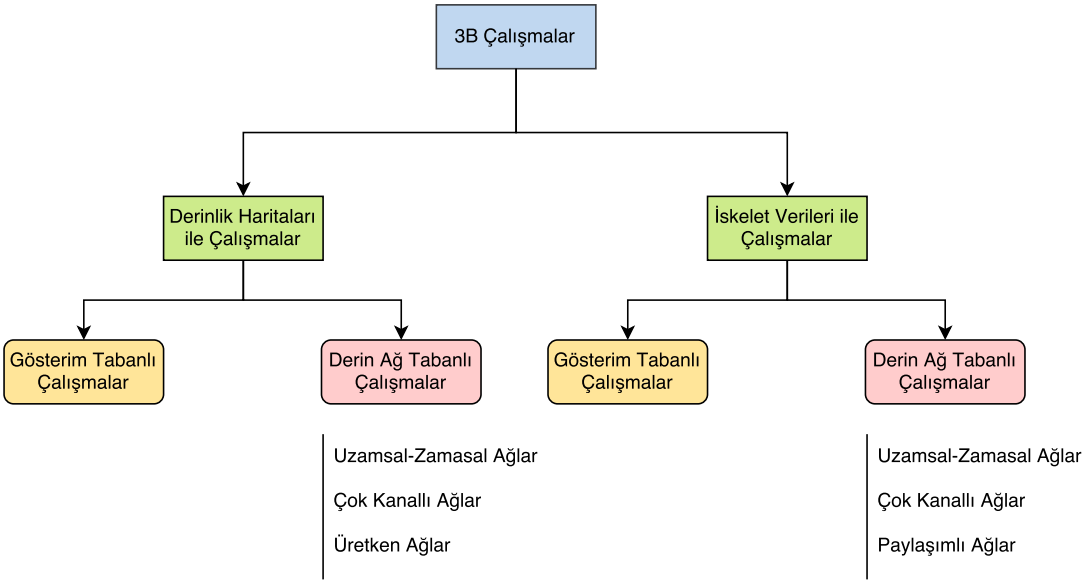
3B insan vücut yapısı, eylemlerin derinlik bilgisiyle daha iyi ifade edilmesine sebep olmaktadır. Ayrıca derinlik kameralarından, 3B eklem koordinatları, gerçek zamanlı olarak elde edilmektedir.

Çalışmalar, Şekil 2.3 teki gibi RGB+D kameralardan elde edilen verilerin derinlik-haritası veya iskelet eklem verileri olmalarına göre ikiye ayrılabilir. İki tür veriyi kullanarak yapılan çalışmalar 1. kısımda ele alınacaktır.

2.2.1. 3B Derinlik Haritaları Tabanlı Çalışmalar

2.2.1.1. Gösterim Tabanlı Çalışmalar

2B Gösterim tabanlı çalışmalar gibi derinlik verilerini kullanarak hareketlerin özniteliklerini çıkarmayı amaçlayan çalışmalar yapılmıştır. Bu çalışmalardan biri ile [44] oluşturulmuş veri kümelerinin başında gelen MSR Action3D veri kümesi, derinlik kamerası olarak kullanılan Kinect 1'den alınan görüntülerle oluşturulmuştur. Bu görüntüler arasından hareket verilerinin çıkarılması için basit bir projeksiyon tabanlı seçme yöntemi kullanılmıştır. 3 boyutlu düzlemlerin hepsi için insanın noktasal kontörleri tespit edilmiştir. Bu kontörlerin üzerinden seçilen 3B noktalar, çizge ile ifade edilip, NERF C-Means tekniğiyle öğrenilmiştir. Bu veriler üzerinde yapılan ilk çalışma olması bakımından önemlidir.



Şekil 2.3: 3B veriler kullanılarak yapılan çalışmaların gruplanması.

Hareketleri ifade etmek için sadece insan üzerindeki 3B nokta kümelerini değil, eklemelere yakın 3B noktasal alanları da kullanan [45] çalışması yapılmıştır. Bu çalışma, hareketlerle ilişki çevresel verileri çıkarak, nesnesel farklılıkta oluşan hareketleri de (örneğin su içmekle, pizza yemek gibi) tanıyabilmektedir. Buna ek olarak hareketi ifade ederken tüm eklem noktalarını kullanmak yerine, yer değişiminin büyük olduğu bölgelerdeki veriler bulunup, kullanılmıştır. Hareketler yine çizgeyle ifade edilip, çoklu çekirdek tekniğiyle sınıflandırılmıştır.

Derinlik haritası tabanlı çalışmalar [46]–[48], hareketi en açık şekilde ifade eden verileri çıkarıp, farklı gruplama teknikleri kullanarak devam etmiştir. [48] çalışması hareketlerin tanınması için derinlik haritalarının dik düzlemleri üzerinde çalışmayı önermiştir. Bir hareket için her düzlemin değişimini ayrı ayrı ele alarak 3B hareket haritaları oluşturulmuştur. Bu haritaların HOG bilgileri çıkarılıp, doğrusal f DVM ile öğrenilmiştir. Bunların dışında histogram odaklı çalışmalar [46], [47] devam etmiş, derinlik verilerinden hareketlere özgü nitelikler çıkartılmış ve popüler veri kümeleri üzerinde üstün başarılar [49] elde edilmiştir.

Derinlik haritaları kullanılarak çalışma zamanı performansı öncelikli çalışmalarda [50], [51] yapılmış, diğer tekniklerin performansları ile kıyaslanmıştır. Gerçek zamanlı sistem [51], hareketlerin dominant görüntülerini 3 boyutlu düzleme yansıtıp, hareketin başından sonuna kadar ele alarak ilerlemiştir.

2.2.1.2. Derin Ağ Tabanlı Çalışmalar

Bölüm 2.1.1'de yapıldığı gibi bu çalışmalar da 4 başlık altında incelenmiştir. Çalışmalar Tablo 2.2'de özetlenmiştir. Hareketlerin özniteliklerini çıkarırken ve sınıflarını bulurken bütüncül bir sistem sunan derin ağlar, 2B görüntü ve videolarda kullanıldığı gibi 3B derinlik haritalarında sıkça kullanılmaktadır.

Zamansal ve mekânsal ağ filtreleri hareketin aktivasyon bölgelerini bulmayı amaçlar. Bu aktivasyon RGB+D değerlerin değişimi veya görüntülerdeki geometrik açıların değişimi gibi çok farklı öznitelik tipinden [52] oluşabilir.

Derinlik haritalarının birçok gereksiz bilgiyi içermesi çalışmalarda bir ön temsil çıkarma ihtiyacını doğurmaktadır. Hareketi hiçbir ön işleme tabi tutmadan girdi olarak kullanan ağ yapıları olduğu gibi, farklı bir uzayda temsiller oluşturup bu temsilleri kullanan çalışmalar da yapılmıştır. Örneğin çalışma [53] SFAM adı verilen 6 adet el ile oluşturulan öznitelik CNN ağına verilip sınıflandırılmıştır. Ağın neleri öğrenmesi amaçlanırsa;

- Ya girdi olarak o öznitelikler verilmelidir.
- Ya da öğrenmesi amaçlanan bilgiler doğrultusunda ağın yapısı kurgulanmalı, hata fonksiyonu belirlenmelidir.

Çok kanallı ağ [54] yapıları derinlik görüntü dizilerinin üzerinde zaman ve yer bilgilerini ayrı ayrı çıkarmak için kullanılabilirler. Hem RGB+D videoları hem de iskelet verisi gibi 2 farklı veri tipini kullanmak için de bu ağlar [55], [56] idealdir. İskelet verisi derinlik haritalarında ağın nereye odaklanması gerektiği hakkında ağa ipucu verirken, derinlik haritaları çevresel detayları sunmaktadır.

Üretken ağlar, [57] denetimsiz öğrenme yapmak veya bir başka modelin daha iyi öğrenmesi için veri miktarını artırmak için kullanılabilir.

Bu veriler ile paylaşımlı ağlar üzerinde yapılmış bir çalışmaya rastlanamamıştır. Bu açıdan Bölüm 7'de sunduğumuz çalışma özgün bir yapıya sahiptir.

2.2.2. İskelet Tabanlı Çalışmalar

Derinlik kameralarından elde edilen iskelet görüntüleri eklem pozisyon bilgilerinden ibarettir. Bu pozisyon bilgisinin pek çok avantajı vardır. Bunlar;

- Az miktardaki veri sayesinde yöntemin çalışma zamanını iyileştirir.
- Tam olarak insanın üzerinde bulunan 3B eklem bilgisini sunduğu için gereksiz çevresel bilgileri içermez.
- Önişlem gereksinimi derinlik görüntülerine göre daha azdır.

İskelet görüntülerindeki eklem sayıları cihaza ve yazılıma bağlı olmak üzere değişmektedir. Ancak çalışmalarda çoğunlukla cihaz olarak Microsoft Kinect kullanılmıştır. Bu da cihazın hata oranının çalışma üzerine olan etkisini ortadan kaldırmaktadır. İskelet verileri, 2B ve 3B görüntü verilerinin eksik yanlarını büyük oranda gidermektedir. Avantajlarının çok olması nedeniyle iskelet verilerini kullanan çalışmaların miktarı fazladır. Bu kısımda, yukarıda belirttiğimiz şekilde çalışmaları gruplandırıp detaylı şekilde anlatacağız.

3B iskelet görüntülerinden öznitelik çıkararak sınıflandırma yapan çalışmalar hareketin detaylı temsillerini oluşturmayı amaçlamıştır ve eklemlerin pozisyonlarını kullanmışlardır. İskelet pozisyonlarını çizge ile ifade eden çalışma [58] ve ağaç ile ifade eden çalışma [59] bunlara birer örnektir. Hareketin en iyi temsillerini oluşturmaya çalışan araştırmalar [60], sınıflandırma başarısının temsil başarısına büyük etkisinin olduğunu göstermeyi amaçlamıştır. Bu sebeple derin ağ yapılarını kullanırken bile verilerin geometrik temsillerini [14] çıkarmışlardır.

Tablo 2.2: 3B derinlik haritaları ile yapılan derin ağ tabanlı çalışmalar.

Çalışmanın Adı	Kullanılan Yöntem	Yılı	Model Tipi
Learning Discriminative Representations From RGB-D Video Data [52]	DBN	2013	Uzamsal-Zamansal Ağlar
Scene Flow To Action Map: A New Representation For RGB-D Based Action Recognition With Convolutional Neural Networks [53]	SFAM's CNN	2017	
Human Action Recognition İn Rgb-D Videos Using Motion Sequence İnformation And Deep Learning [54]	CNN T-Sen Visualization 2B ve 3B Görüntü Birlikte	2017	Çok Kanallı Ağlar

Learning Action Recognition Model From Depth And Skeleton Videos [56]	CNN İskelet Ve RGBD	2017	
Multimodal Multipart Learning For Action Recognition İn Depth Videos [55]	LSTM	2016	
Unsupervised Learning Of Long-Term Motion Dynamics For Videos [57]	LSTM Kodlayıcı Decoder Codebook	2017	Üretken Ağlar

Derin ağ tabanlı çalışmalar ise Tablo 2.3'de adı ve açıklaması ile verilmiştir. Tabloya çalışmaların en çok kullanılan veri kümesi üzerindeki çapraz denek başarım değerleri de eklenmiştir. Bu çalışmaların yoğun bir şekilde sürdürülme nedenlerinin bir diğeri de oluşturulan orijinal mekanizmaların sadece hareket verileri ile sınırlı kalmayıp başka veriler üzerinde genelleştirilebilir olabilmesidir.

Bu tez kapsamında iskelet verileri ile hareket tanıma için bir derin ağ modeli [16] önerilmiştir. Bu ağ hareketler arasındaki metrikleri öğrenmektedir. İki hareket arasındaki benzerlik metrikleri, zaman ve uzam farkları göz önünde bulundurularak çıkarılmaktadır. Derin benzerlik metriklerin bir çok avantajı vardır. Hareket tanıma problemi dışında benzerlik bulma, bilinmeyen hareketleri çıkarma gibi pek çok uygulama alanı oluşturulabilmektedir. Metrikleri çıkarabilen modülün yanında, tanıma yapan bir sınıflandırma modülü de çalışmada sunulmuştur. Bu iki modülü beraber eğiten Uçtan Uca İkiz LSTM Ağı ise tezin ana çalışması olarak Bölüm 7'de anlatılacaktır.

Bunlara ek olarak oluşturduğumuz model farklı problemler için de uygundur. Yapının başarısı sadece hareket verileri üzerinde değil içindeki alt ağların kullanılabildiği tüm veri türlerinde elde edilebilmektedir.

Önerdiğimiz yapı bu alanda bulunan paylaşımlı ağlara da yegâne örnektir. Tablodan görülebileceği üzere bu alanda zamansal ve mekânsal çok akışlı veya tek akışlı pek çok yöntem önerilmiştir. Bu yöntemlerin başarısı, test edilen verilerin yapısal özelliklerine sıkı sıkıya bağlıdır. Bizim yöntemimiz ise bu çalışmalara göre daha genelleştirilebilir ve daha uygulanabilirdir.Bu bölümde hareket tanıma ile ilgi yapılan çalışmalara değinilmiştir. Bu alanda yapılan araştırma miktarı oldukça fazla olduğu için bu bölümde sadece son beş yıla ağırlık verilmiştir. Bir sonraki kısımda tez kapsamında kullanılan veriler, hareket görüntüleri ve detaylarına yer verilecektir.

Tablo 2.3: 3B İskelet verileri ile yapılan derin ağ tabanlı çalışmalar.

Çalışmanın Adı	Kullanılan Yöntem	NTU RGB+D Başarı CS – CV		Yılı	Model Tipi
Skepxels: Spatio-temporal Image Representation of Human Skeleton Joints for Action Recognition [61]	Hareketlerin hem hız pencereleri hem lokasyon pencereleri oluşturulup CNN ile sınıflandırılmıştır.	82,3	89,2	2017	Uzamsal-Zamansal Ağlar
NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis [62]	Vücut bölümlerini ayrı girdi olarak alan LSTM yapısı kullanılmıştır.	62,93	70,27	2016	
On geometric features for skeleton-based action recognition using multilayer LSTM networks [47]	Eklemleri nokta, doğru ve düzlem olarak ifade edip farkları ile LSTM ağı modellenmiştir.	70,26	82,39	2017	
Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition [48]	Eylemleri ağaç yapısı ile ifade edip hareket tanıma yapılmıştır.	69,2	77,7	2016	
Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition [63]	Hiyerarşik olarak adım adım öznitelikler RNN ağı tarafından bulunarak hareket tanıma yapılmıştır	-	-	2015	
Deep learning on lie groups for skeleton-based action recognition [60]	Lie grup elementleri ile LSTM ağı eğitilmiş LieNET adında ağ sunulup kullanılmıştır.	69,2	77,7	2017	
Interpretable 3D Human Action Analysis with Temporal Convolutional Networks [64]	Resnet ile hareketlerin zamansal özniteliklerini öğrenen ağ tasarlanmışdır.	74,3	83,1	2017	
Global context-aware attention LSTM networks for 3d action recognition [65]	Kompleks LSTM ağ yapısı ile ağı hem zamansal hemde bölgesel öğrenip sınıflandırma yapılmıştır.	74,4	82,8	2017	Çok Kanallı Ağlar
Two-Stream 3D Convolutional Neural Network for Skeleton-Based Action Recognition [66]	CNN ağını iki farklı özelliği çıkarmak için eğitmiştir.	66,85	72,58	2017	
Pose-conditioned Spatio-Temporal Attention for Human Action Recognition [67]	Hem derinlik görüntülerini hem iskelet pozlarını kullanarak çoklu bir yapı sunulmuştur.	84,8	90,6	2017	
3D Human Action Recognition with Siamese-LSTM Based Deep Metric Learning [16]	İkiz ağ yapısıyla baştan sona benzerlikler öğrenerek sınıflandırma yapan model önerilmiştir.	-	-	2018	Paylaşımlı Ağlar

3. ÜÇ BOYUTLU HAREKET VERİLERİ

RGB+D kameralarının maliyetlerinin düşmesi ile 3 boyutlu veriler üzerinde çalışmalar yoğunlaşmıştır. Bu kameralar sayesinde derinlik verilerinden faydalanılarak hareket tespiti yapılabilmektedir. Yapılan başarılı çalışmalarla birlikte [13] bu cihazlar insan iskeletine ait eklemlerin koordinatlarını gerçek zamanlı tespit edilebilmektedir. Eklem koordinat verilerinin tespiti sayesinde insanın içinde bulunduğu ortama ait olup, insan hareketleriyle alakası olmayan veriler elenebilmektedir. Böylece kullanılan veriler küçülmekte ve meydana gelen performans artışıyla gerçek zamanlı sistemler oluşturulabilmektedir.

Tablo 3.1'de son on yıl içinde toplanılan 3B veri kümeleri gösterilmektedir. Veri kümeleri, farklı sayıda hareketin derinlik haritalarını ve 3B iskelet eklem koordinat dizilerini içermektedir. Veriler büyük çoğunlukla iç mekânda gerçekleşen gündelik ve spor hareketlerden oluşmaktadır. Her hareket, birbirinden farklı vücut yapılarına sahip kişiler tarafından birçok defa gerçekleştirilmiştir. Test kümeleri, bu kişilerden birine ait tüm hareketlerin, diğer kişilerin hareketlerinden ayrılmasıyla oluşturulmuştur. Bazı veri kümeleri birden fazla cihaz kullanılarak toplanmıştır. Bu sayede, sunulan yöntemin farklı görüş açılarındaki performansı test edilebilmektedir.

Kullanılan cihaz ve yazılım, öznelerin üzerinde tespit edilen eklem sayılarını etkilemektedir. Microsoft Kinect'in 1. Versiyonundan 20 adet 3B eklem alınabilirken, 2. Versiyonundan 25 adet 3B eklem alınabilmektedir.

3.1. Veri Kümeleri

Bu bölümde, 3 boyutlu insan hareketlerinden oluşan veri kümeleri hakkında genel bilgi verilecek, daha sonra kendi oluşturduğumuz ve yöntemlerimizi denediğimiz veri kümeleri detaylıca açıklanacaktır.

3.1.1. GTU Action 3D

Bu veri kümesi, tez çalışması kapsamında Kinect 2 cihazı ile toplanmıştır. Toplam 14 tip hareket 10 farklı kişi tarafından gerçekleştirilip kaydedilmiştir. Toplamda 508 adet hareket bulunmaktadır. Veri kümesi genellikle kapalı alanda gerçekleştirilebilecek aerobik spor hareketlerini içermektedir. Tablo 3.2'de bu

hareketler gösterilmiştir. Veri kümesini toplarken ki motivasyonumuz Kinect 2 cihazının avantajlarından biri olan 25 eklem noktasını diğer az sayılı ekleme sahip veri kümeleriyle kıyaslamaktı.

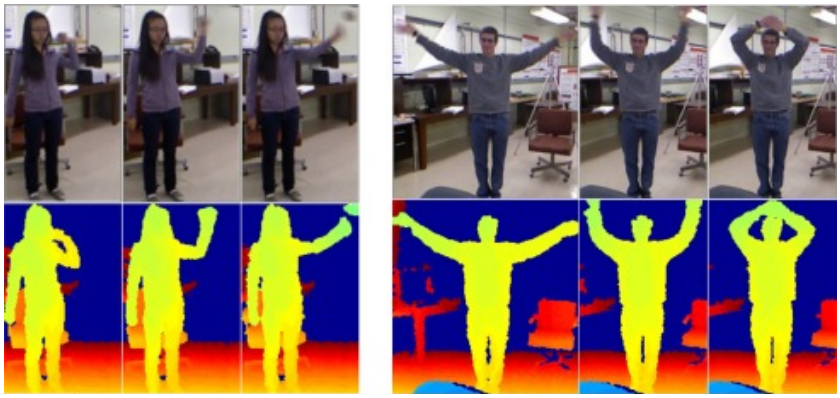
Tablo 3.1: 3B insan hareketi veri kümeleri.

Veri Kümesi	Hareket Sayısı	Hareket Sınıfı	Özne Sayısı	Kamera Sayısı	Cihaz	İçerik	Yıl
MSR Action3D [44]	567	20	10	1	-	RGB+D+3DJoints	2010
CAD-60 [68]	60	12	4	-	Kinect v1	RGB+D+3DJoints	2011
RGBD-HuDaAct [69]	1189	13	30	1	Kinect v1	RGB+D	2011
MSRDailyActivity 3D [45]	320	16	10	1	Kinect v1	RGB+D+3DJoints	2012
Act42 [70]	6844	14	24	4	Kinect v1	RGB+D	2012
CAD-120 [71]	120	10+10	4	-	Kinect v1	RGB+D+3DJoints	2013
MSR 3D Action Pairs [47]	360	12	10	1	Kinect v1	RGB+D+3DJoints	2013
Multiview 3D Event [72]	3815	8	8	3	Kinect v1	RGB+D+3DJoints	2013
Florence Action 3D [15]	215	9	10	1	Kinect v1	3DJoints	2013
Northwestern-UCLA [73]	1475	10	10	3	Kinect v1	RGB+D+3DJoints	2014
UWA3D Multiview [46]	~900	30	10	1	Kinect v1	RGB+D+3DJoints	2014
Office Activity [74]	1180	20	10	3	Kinect v1	RGB+D	2014
UTD-MHAD [75]	861	27	8	1	Kinect v1+WIS	RGB+D+3DJoints+ID	2015
UWA3D Multiview II [76]	1075	30	10	5	Kinect v1	RGB+D+3DJoints	2015
GTU Action 3D	508	14	10	1	Kinect v2	3DJoints	2015
NTU RGB+D [62]	56880	60	40	80	Kinect v2	RGB+D+IR+3DJoints	2016

Tablo 3.2: GTU Action 3D veri kümesi hareket sınıfları

1	Kolları Açıp Kapamak
2	Sağ El Sallamak
3	Sağa Sola Bel Esnetmek
4	Yürümek
5	Sağ Ayak Esnetmek
6	Sol Ayak Esnetmek
7	Sol El Sallamak
8	Öne Sağ Sol İlerlemek
9	Çömelmek
10	Sandalyede Oturup Kalkmak
11	Oturup Alkışlamak
12	Bel Çevirmek
13	Sağa Sola 8 Adım Hareket Etmek
14	Boyun Gevşetmek

3.1.2. Florence Action 3D



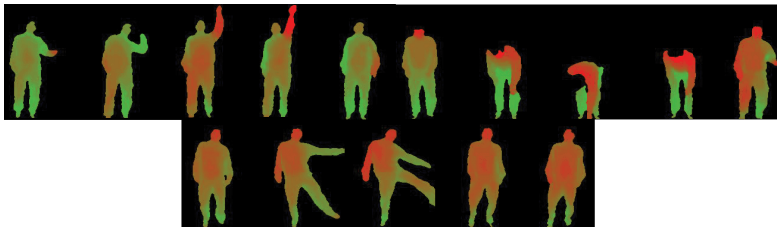
Şekil 3.1: Florence Action 3D veri kümesinden örnek görüntüler, üst kısım RGB, alt kısım RGB+D görüntülerini göstermektedir.

Floransa Üniversitesi tarafından Kinect kullanılarak toplanmıştır. Toplam 9 tip hareket 10 farklı kişi tarafından gerçekleştirilip kaydedilmiştir. Toplam 215 harekettir. Hareket sayısının az olması nedeniyle zorlayıcı bir veri kümesidir. Tablo 3.3'de bu veri kümesine ait hareket sınıfları gösterilmiştir.

Tablo 3.3: Florence Action 3D veri kümesi hareket sınıfları

1	El Sallamak
2	Şişeden İçmek
3	Telefonu Cevaplamak
4	Alkışlamak
5	Bağcık Bağlamak
6	Oturmak
7	Ayağa Kalkmak
8	Saate Bakmak
9	Eğilmek

3.1.3. MSR Action3D



Şekil 3.2: MSR Action3D veri kümesinden örnek RGB+D görüntüler.

Microsoft araştırmacıları tarafından 2010 yılında Kinect 1 kullanılarak toplanmıştır. Üzerinde en çok çalışılan veri kümelerinin başında gelmektedir. 20 farklı hareket sınıfı vardır. 10 farklı kişi tarafından toplam 567 hareket gerçekleştirilmiştir. Hareket sınıfları Tablo 3.4'deki gibidir.

Tablo 3.4: Microsoft Action 3D veri kümesi hareket sınıfları.

1	Yukarı Kol Sallamak
2	Yana Kol Sallamak
3	Çakmak
4	El Tutmak
5	İleri Yumruk Atmak
6	Yükseğe Fırlatmak
7	X Çizmek
8	Tik Çizmek
9	Daire Çizmek
10	El Çırpmak
11	İki El Sallamak
12	Yandan Yumruk Atmak
13	Bel Çevirmek

14	İleri Tekme Atmak
15	Yana Tekme Atmak
16	Koşmak
17	Teniste Savurmak
18	Teniste Servis Atmak
19	Golf Oynamak
20	Tutup Atmak

3.1.4. NTU RGB+D

Şekil 3.3: NTU RGB+D veri kümesinden örnek iskelet, RGB, RGB+D görüntüleri.

Nanyang Teknik üniversitesi tarafından Kinect 2 cihazıyla 2016 yılında toplanmıştır. Hareket sayısı, kamera sayısı ve hareket sınıfı sayısı bakımından toplanan en büyük veri kümesidir. 3 boyutlu derinlik görüntüleri dışında Şekil 3.3'de gösterildiği gibi kızılötesi görüntüler de mevcuttur.

Denek testleri dışında, kamera görüş açısı testleri de yapılabilmektedir. Bunun için aynı alandan üç farklı yatay görüntü elde edilmiş, aynı anda üç kamera kullanılmıştır. Her bir kurulum için, üç kamera aynı yükseklikte, ancak üç farklı yatay açıda yerleştirilmiştir. Bu açılar sırasıyla -45 °, 0 °, + 45 °'dir. Her denekten, bir kez sol kameraya doğru ve bir kez de ortadaki kameraya doğru her eylemi gerçekleştirmesi istenmiştir. Bu şekilde, iki ön görüntü, bir sol yan görünüm, bir sağ yan görünüm, bir sol taraf 45 derece görünümü ve bir sağ 45 derece görünümü yakalanmıştır.

60 adet hareket sınıfı vardır. Bunlar 3 ana gruba ayrılabilir, 40'ı günlük hareketlerden, 9'u sağlıkla ilgili hareketlerden, kalanı ise etkileşimli hareketlerden oluşmaktadır. Bunlar Tablo 3.5'te verildiği gibidir.

Tablo 3.5: NTU RGB+D veri kümesi hareket sınıfları.

1	Su İçmek
2	Yiyecek / Aperatif Yemek Yemek
3	Diş Fırçalamak
4	Saç Taramak
5	Düşürmek
6	Almak
7	Atmak
8	Oturmak
9	Ayağa Kalkmak (Oturma Pozisyonundan)
10	Alkışlamak
11	Okumak
12	Yazmak
13	Kâğıt Yırtmak
14	Ceket Giymek
15	Ceket Çıkarmak
16	Ayakkabı Giymek
17	Ayakkabı Çıkarmak
18	Gözlük Takmak
19	Gözlüklerini Çıkarmak
20	Şapka Takmak
21	Şapka Çıkartmak
22	Neşelenmek
23	El Sallamak
24	Bir Şeyler Tekmelemek
25	Cebe Bir Şey Koymak / Cebinden Bir Şeyler Çıkarmak
26	Atlamak (Tek Ayak Atlama)
27	Zıplamak
28	Telefon Cevaplamak
29	Telefonla / Tabletle Oynamak
30	Bir Klavyeyle Yazmak
31	Parmağıyla Bir Şeyi İşaret Etmek
32	Kendi Fotoğrafını Çekmek
33	Saati Kontrol Etmek
34	İki Elini Ovmak
35	Başını Sallamak
36	Kafa Sallamak
37	Yüzü Silmek
38	Selamlamak
39	Ellerini Bir Araya Getirmek
40	Elleri Çaprazlamak (Dur Demek)
41	Öksürmek Hapşırmak
42	Sendelemek
43	Düşmek
44	Kafaya Dokunmak (Baş Ağrısı)
45	Göğse Dokunmak (Mide Ağrısı / Kalp Ağrısı)

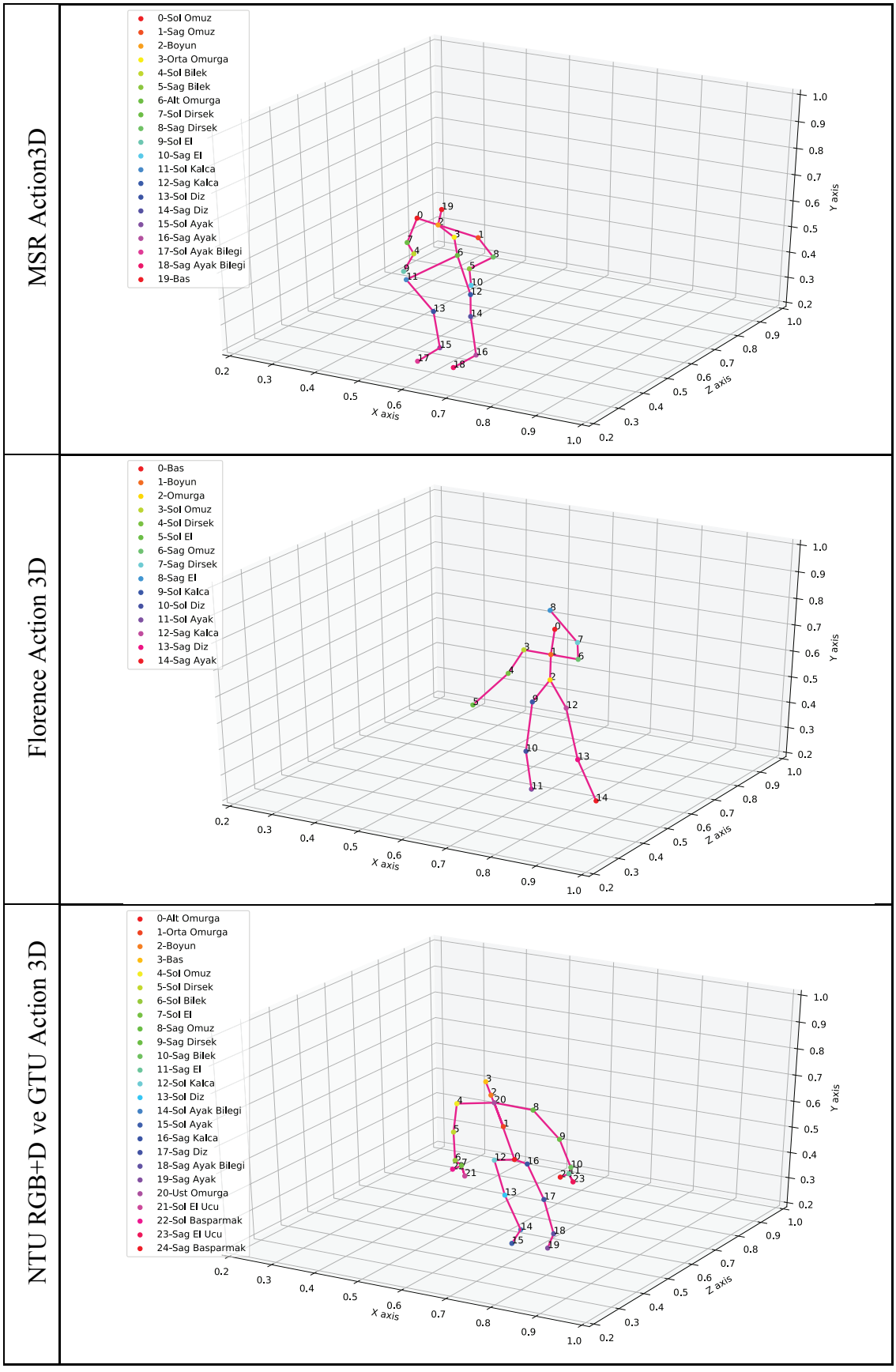
46	Sırta Dokunmak (Sırt Ağrısı)
47	Boyuna Dokumak (Boyun Ağrısı)
48	Bulantı veya Kusmak
49	Hava Yapmak / Sıcak Hissetmek (El Veya Kâğıt İle)
50	Başkasına Vurma / Tokatlamak
51	Başkasını Tekmelemek
52	Başkasını İtmek
53	Başkasının Arkasına Vurmak
54	Diğer Kişiyi İşaret Etmek
55	Başkasına Sarılmak
56	Başkasına Bir Şeyler Vermek
57	Başkasının Cebine Dokunmak
58	El Sıkışmak
59	Birbirine Doğru Yürümek
60	Birbirinden Uzaklaşmak

3.2. İskelet Görüntüleri Üzerinde Normalleştirme Yöntemleri

İskelet görüntüleri vücut eklemlerine ait 3B koordinat bilgilerini içermektedir. Bu koordinatlar, kameranın yatay-dikey eksenindeki *x, y* ve kameraya olan uzaklık mesafesi olan *z* değeridir.

Veri kümeleri, bu cihazlardan toplanan 3B eklem verilerinden ve/veya derinlik görüntülerinden oluşmaktadır. Aşağıdaki Şekil 3.4'te tez kapsamında kullanılan veri kümelerinin örnek iskelet görüntüleri verilmiştir. Görüntülerin yanında eklemlerin hangileri olduğu belirtilmiştir. MSR Action3D veri kümesi Kinect vs. 1 cihazı ile Windows üzerinde, Florence Action 3D veri kümesi yine Kinect vs 1 cihazı ile Linux işletim sisteminde OpenNI YGKsı ile, GTU Action 3D ve NTU RGB+D veri kümeleri ise Kinect vs. 2 cihazı ile Windows üzerinde toplanmıştır.

Birbirinden farklı eklem verilerini aynı şekilde ifade edebilmek, sonraki çalışmalarda bu verileri birlikte kullanabilmek ve hata hesabını kolaylaştırmak için veriler üzerinde normalleştirme yapılmıştır. Normalleştirme yöntemleri Bölüm 3.2.1'de anlatılmıştır.



Şekil 3.4 Veri kümeleri iskelet görünümleri a. MSR Action3D, b. GTU Action 3D ve NTU RGB+D c. Florence Action 3D.

3.2.1. Normalleştirme Yöntemleri

3.2.1.1.İskelet Görüntü Normalleştirmesi

Tüm veri kümelerindeki her görüntü için gerçekleştirilmiştir. Her eklemin 3B koordinat değerinden alt omurga koordinat değeri çıkarılmıştır. Normalleştirme algoritması Algoritma 3.1'de gösterildiği gibidir.

Algoritma 1 İskelet eklemleri normalizasyonu

```
1: procedure FrameNormalization(frame,spid)
2:     spineJoint = frame[spid]
3:     normalizedFrame = [ ]
4:     for each joint in frame do
5:         normalizedJoint = joint - spineJoint
6:         insert normalizedJoint into normalizedFrame
7:     end for
8:     Return normalizedFrame
9: end procedure
```

Algoritma 3.1. İskelet görüntülerini normalleştirme yöntemi.

3.2.1.2. Veri Kümesi Normalleştirmesi

Normalleştirilmiş 3B eklem koordinat değerlerinden bazıları negatif bazıları pozitiftir. Ayrıca derinlik değeri ile *x-y* koordinat değerleri arasındaki fark fazladır. Bu nedenle tüm veri kümesindeki eklem değerleri, 0 ile 1 arasına orantılı bir şekilde yerleştirilmiştir. Normalleştirme yönteminin adımları Algoritma 3.2'daki gibidir.

Algoritma 2 Veri kümesi normalizasyonu

```
1: procedure DatasetNormalization(data)
2:     maxPoint = max value in data
3:     minPoint = min value in data
4:     for each action in data do
5:         normalizedAction = (action+ |minPoint| )/(maxPoint+ |minPoint| )
6:         insert normalizedAction into normalizedData
7:     end for
8:     Return normalizedData
9: end procedure
```

Algoritma 3.2. Veri kümelerini normalleştirme yöntemi.

4.GEOMETRİK EKLEM ÇANTASI YÖNTEMİ İLE HAREKET TANIMA

Tezin bu bölümünde, geometrik eklem çantası yöntemi anlatılmıştır. Bu çalışmada küçük boyutlu veri kümelerindeki hareketlerin yeni bir gösterimi oluşturularak sınıflandırılması yapılmıştır.

3B iskelet görüntü dizileri üzerinde hareket tanıma konusunun temel problemi, eklem hareketlerinin gösterim şeklidir. İskelet görüntülerindeki eklem koordinat verilerini işleyerek değerli öznitelikler elde edilebilmektedir. Bu öznitelikler zamansal ve uzamsal olarak değişik açılardan veriye ait bilgiler taşımaktadır. Bu bilgileri çıkarabilmek amacıyla pek çok çalışma [25], [35], [77], [78] hareket verilerini önce bir ön işleme tabi tutmuş, sonra sınıflandırmıştır.

Bizde bu çalışmada hareket özniteliklerini çıkarmak için eklem koordinatlarından geometrik öznitelikler çıkartıp, bu öznitelikleri kelime çantası yöntemi ile kullandık.

Kelime çantası yöntemi genellikle doğal dil işleme problemlerinde kullanılan standart öznitelik çıkarma tekniğidir. Yöntem basitçe kelime çantası, var olan cümlelerdeki kelimelerden sözlük oluşturup, cümlelerin sözlüğe göre konumunun hesaplanmasıdır. Kelime çantası yöntemi tamamen kelimelerin yapısına bağlı olarak kompleks veya basit olabilir. İnsan hareketi öznitelikleri çıkarmada da [15], [44] kullanılmaktadır. Burada oluşturulan sözlük çok kompleks öznitelikler olabileceği gibi basit eklem pozisyonları da olabilir.

Yaptığımız kelime çantası yöntemi tabanlı çalışmada, iskelet koordinatlarından elde edilen geometrik öznitelikler eklemlerin oluşturduğu doğrular ve eklem koordinat noktaları ile hesaplanmıştır. Elde ettiğimiz öznitelikler kelime çantası yönteminde kelime olarak kullanılmıştır. Kelimeler tüm veri kümesi için gruplandırılıp, bir geometrik öznitelik sözlüğü oluşturulmuştur. Sözlüğe göre hareketlerin kelime grup numaralarının histogram değerleri hesaplanmıştır. Sonrasında histogram değerlerinin oluşturduğu vektörler sınıflandırılmıştır.

Yöntemimizi kendi veri kümemiz GTU Action 3D, MSR Action3D ve Florence Action 3D verileri üzerinde test ettik. Yapılan testlerin sonuçları bölüm sonunda gösterilmiştir.

4.1. Yöntem

Bu kısımda yöntemin detayları 3 ana başlıkta anlatılacaktır. Bunlar;

i) Geometrik Öznitelikler
ii) Kelime Çantası Yöntemi
iii) Hareketlerin Sınıflandırılması

Önce kelimelerin hareket verilerinden nasıl çıkarıldığı anlatılacaktır. Sonrasında bu kelimelerden oluşan sözlüğe, sözlüğün özelliklerine ve kelime çantası yöntemine uygulanışına değinilecektir. Son kısımda ise bu gösterimin sınıflandırılması yani hareketleri özniteliklerden tanıma kısmından bahsedilecektir.

4.1.1. Geometrik Öznitelikler

Kelime çantası tabanlı yöntemde kullanılacak öznitelikler bu bölümde anlatılmıştır. Bu öznitelikleri seçerken dikkate alınan hususlar,

- Bir hareket eklemler kullanılarak meydana gelir. Bu sebeple eklem koordinatları kendi başına birçok şeyi ifade etmektedir. Hareketin meydana geldiği pozisyon bilgisi hareket tanımlama için yararlı ve özet bir bilgidir.
- Hareketler zaman içinde gerçekleşir. Hareketlerin süresi her örnekte farklıdır. Bu tür zamana bağlı, sıralı verilerin sınıflandırılmasında var olan tüm zorluklar burada da geçerlidir. Hareketin süresi, sırası gibi parametreler hareketin gösteriminde oldukça önemlidir.

İnsanın vücut bölgeleri; omurga, sağ kol, sol kol, sağ bacak, sol bacak olmak üzere 5 parçadan oluşur. Her hareket tipi vücut bölgelerinin birkaçının aktif kullanılmasıyla meydana gelir. Her harekette eklemlerin bazılarında büyük pozisyon değişimleri olur. Bununla beraber o eklemin komşuları olan diğer eklemlerde bu değişimden etkilenmektedir. Örneğin sağ el sallama hareketi, sağ elin sallanması ile gerçekleşir. El sallanırken omurga ve baş bir miktar hareket edebilir.

Bu durumların hepsini dikkate alan uzamsal ve zamansal hassasiyete sahip bir sistem geliştirmek için eklem koordinat değerlerinin geometrik özniteliklerini hesapladık.

Bu öznitelikler sırasıyla, normalleştirilmiş eklem koordinatları (ek), eklem koordinat farkı (ekf) ve eklem doğru uzaklıklarıdır (edu). Bir 3B hareketin $H_1 = \{ek^1, ek^2, ek^3, ek^4 \ldots, ek^T\}$, eklem koordinat dizisi girdi verileridir. T her bir hareket için farklı olmak üzere bir hareketteki toplam iskelet görüntü sayısını belirtmektedir. $ek^t = \{j_1^t, j_2^t, \ldots j_N^t\}_p$, N adet 3B eklem koordinat değerlerinden oluşan tek bir iskelet görüntüsüdür. N sayısı, veri kümesinde kullanılan cihaza bağlı olan eklem sayısıdır. $j_n^t = \{x_n, y_n, z_n\} \in R^3$, ise 3B tek bir eklem noktasıdır. Kısaca her hareket farklı sayıda 3B eklem koordinat dizilerinden oluşmaktadır.

İlk kullandığımız öznitelik, bu bahsettiğimiz eklem koordinatları ek^t, t anında kameradan gelen gerçek eklem koordinat değerlerine Bölüm 3.2.1 de anlatılan normalleştirme yöntemlerinin uygulanmış halidir. Tablo 4.1'de öznitelik gösterimleri ve formülleri ile birlikte verilmiştir.

İkinci kullandığımız öznitelik, eklem koordinat farkları $ekf^{tt'}$, iki eklem arasındaki Öklid mesafesidir. Bu iki eklem bizim yöntemimizde, iki farklı t ve t' anlarındaki aynı eklemi ifade etmektedir. Bu t ve t' anları artan sıralı değerlerdir. Yani t' her zaman t den sonra ki bir andır. t ve t' anları arasında ki fark değişkendir. Örneğin 30 eklem koordinat dizisinden oluşan bir hareket için: t ve t' *(0,1), (0,2), ,(0,4) ,(0,5), (1,2), (1,3), (1,4), (1,5),}* gibi değerler olabilir. Bir $ekf^{(1,3)} = \sqrt{ek^{1^2} - ek^{3^2}}$ şeklinde hesaplanmaktadır.

Son öznitelik, Eklem Doğru Uzaklıkları edu^t, t anında vücut bölgelerinin merkezindeki eklemlerin diğer bölgelerdeki doğrulara uzaklığıdır. Burada doğrular sırasıyla (c)'de gösterilen: sağ kol doğrusu, sol kol doğrusu, sağ ayak doğrusu, sol ayak doğrusu ve gövde doğrusudur.

Örneğin sağ dirseğin sol bacak doğrusuna olan en kısa mesafesidir. Burada doğru-nokta uzaklığı formülü Tablo 4.1'in ikinci satırında gösterildiği gibi hesaplanmıştır. Önce vücudun seçilen bölgesinden iki nokta tespit edilmiş sonra bunun üzerinden geçen doğru bulunmuştur. $L_{j_n^t \mapsto j_{n''}^t}$, aynı vücut bölgesinde bulunan iki farklı eklemin oluşturduğu doğruyu ifade etmektedir. $L_{j_n^t \mapsto j_{n''}^t}$ ile belirlenen eklem j_n^t arasındaki en kısa mesafe hesaplanmıştır.

Tablo 4.1: Öznitelik gösterimleri ve formülleri.

Öznitelik Sembolü	Öznitelik Adı	Öznitelik Formülü
ek^t	Eklem Koordinatları	$ek^t = \{ j_1^t, j_2^t, \dots \ j_N^t \}$
$ekf^{tt'}$	Eklem Koordinat Farkları (Zamana Bağlı)	$ekf^{tt'} = \left\| \overrightarrow{ek^t ek^{t'}} \right\| \ \forall t < t'$
edu^t	Eklem Doğru Uzaklıkları	$edu^t = (j_n^t, j_{n'}^t \rightarrow j_{n''}^t) \ \frac{L_{j_{n'}^t \rightarrow j_{n''}^t}}{\left\| \overrightarrow{j_{n'}^t j_{n''}^t} \right\|}$

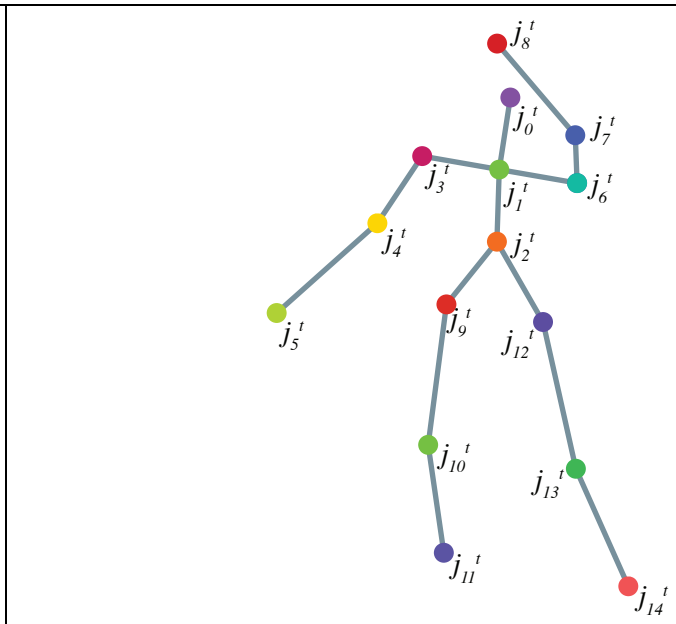
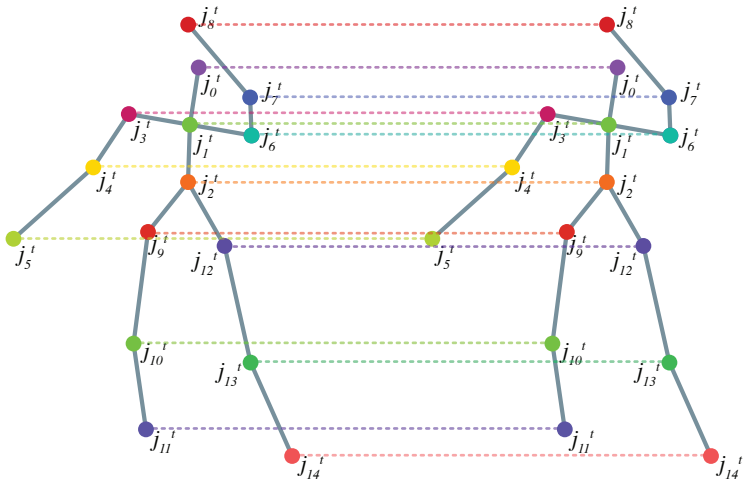
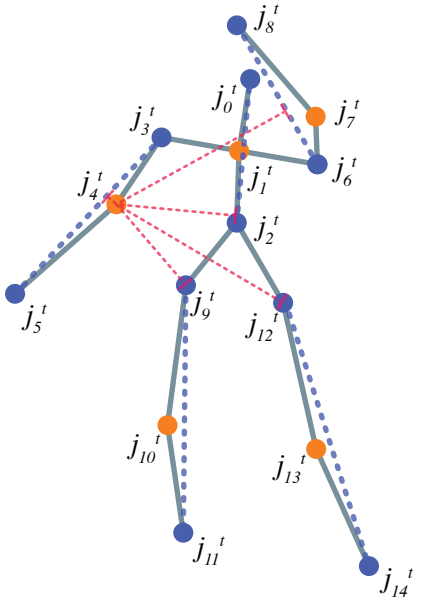
Şekil 4.1'de öznitelikler görsel olarak anlatılmıştır. (a) *t* anında normalleştirilmiş eklem koordinatlarının ek^t hepsini ayrı ayrı renklendirerek , (b) iki farklı anda, aynı eklemin zaman bağlı Öklid değişim değerini (c) ise *t* anında sol dirsek ekleminin belirtilen vücut doğrularına olan uzaklık değerini göstermektedir.

Bu öznitelikler dışında eklem açıları, bölgesel koordinat farkları gibi birkaç öznitelik daha test edilmiştir. Ancak bu özniteliklerden beklenen başarı elde edilememiştir.

4.1.2. Kelime Çantası Yöntemi

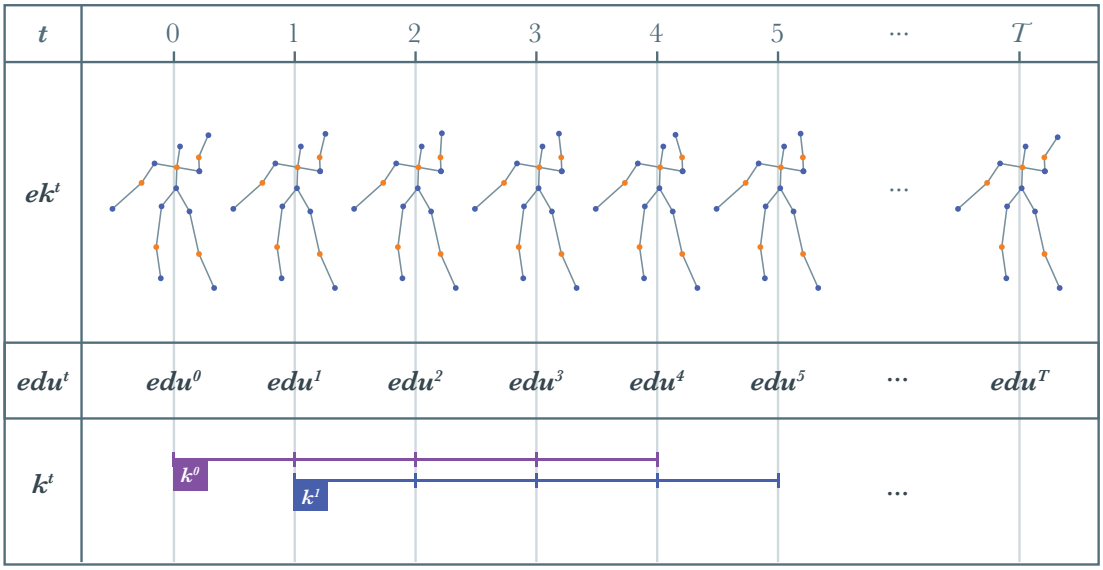
Öznitelikleri çıkarmada güçlü bir teknik olan kelime çantası yöntemi, hareket verilerinden çıkarılan öznitelikler kullanılarak uygulanmıştır. Kelime çantası yöntemi için gerekli olan kelimeler, Bölüm 4.1.1 de bahsedilen özniteliklerin belirli zaman aralıklarındaki dizileridir.

Örneğin Şekil 4.2'de gösterildiği gibi *T* sürede gerçekleşmiş bir hareketimiz olsun. Bu hareketten elde edeceğimiz kelimeler için edu^t özniteliği kullanalım. Kelime uzunluğu *u*, ne kadar süre içerisindeki özniteliklerin alınacağını belirtmektedir. Bu örnek için *u*'yu 5 seçelim. k^0 kelimesi; *t=0,1,2,3,4* anlarındaki edu^t özniteliklerinden oluşmaktadır. $k^0 = \{ edu^0, edu^1, edu^2, edu^3, edu^4 \}$ şeklinde gösterilebilir. k^1 kelimesi bir sonraki andan başlamak üzere yine 5 uzunluklu $k^1 = \{ edu^1, edu^2, edu^3, edu^4, edu^5 \}$ bir kelimedir.

ek^t Eklem Koordinatları	
$ekf^{tt'}$ Eklem Koordinat Farkları (Zamana Bağlı)	
edu^t Eklem Doğru Uzaklıkları	

Şekil 4.1: Özniteliklerin gösterimi a. Eklem koordinatları, b. Eklem koordinat farkları, c. Eklem doğru uzaklıkları.

Bu şekilde T anına kadar veri kümesinin tümündeki hareketler için kelimeler belirlenmektedir. Oluşan her kelime hem zamansal bir bilgiyi hem de özniteliğin yapısından gelen geometrik uzamsal bir bilgiyi taşımaktadır.



Şekil 4.2: Kelime çantası yöntemi için kelimelerin oluşturulması.

T süreli bir harekette bulunan kelime sayısının formülünü şu şekilde ifade edebiliriz. u kelimenin uzunluğu olmak üzere, m bir hareketteki toplam kelime sayısı $m = T - u + 1$ dir. Bu m adet kelime hem zamana özel hem de bölgeye özel bilgiler içermektedir.

Her kelime birbirinden farklı ya da aynı olabilir. Kelimelerin sınıfı k-merkezli gruplama algoritması ile belirlenir. Burada belirlenen k değeri oluşacak sözlükteki toplam farklı kelime sayısıdır.

Oluşan kelimeler tüm veri kümesinin sözlüğünü oluşturmaktadır. Her hareket kelimelerden oluşan cümlelere benzer. Buradaki fark, hareketlerin sahip olduğu kelime sayısı, kullandığımız cümlelerin sahip olduklarından çok daha fazla olmasıdır. Aynı zamanda her hareketin sahip olduğu kelime sayısı birbirinden çok farklıdır ve bu sayıda hareket için değerli bir bilgidir. Bu kelimelerin sayısını ve türünü kullanmak için hareketlerde bulunan kelime tiplerinin histogramları çıkartılarak, sözlükteki toplam farklı kelime sayısı uzunluğunda bir vektör oluşturulmuştur. Sonraki sınıflandırma bölümünde sınıflandırılan vektörlerde bu kelime histogramlarıdır.

4.1.3. Hareketlerin Sınıflandırılması

Kelimelerin k-merkezli ile gruplanıp, hareketlerin kelime histogramları hesaplanmıştır. Bu histogram vektörleri, sınıflandırma için kullanılacak öznitelik vektörleridir. Test verilerinin de öznitelikleri çıkarıldıktan sonra kelimeler k-merkezli ile merkezlere en yakın olan kelime grubu etiketi ile etiketlenmiştir. Her bir test hareketi için sahip olduğu kelime tipleri ve o kelimelerin sayısı sınıflandırmayı sağlayan verilerdir. Histogram vektörleri oluşturulduktan sonra eğitim verileri ile bir SoftMax sınıflandırıcı eğitilmiştir. Sonrasında eğitilen modele test histogram vektörleri sorulup başarımlar hesaplanmıştır. Sınıflandırıcı olarak SVM' de kullanılmıştır. Başarımı SoftMax'den düşük gelmiştir.

4.2. Sonuçlar

Üç veri kümesinde yapılan testlerin çapraz denek başarım sonuçları Tablo 4.2'de gösterilmiştir. Florence Action 3D, daha az eklem ve hareket sayısına sahip oldukça küçük bir veri kümesidir. Florence Action 3D verilerinden 300 kelimeli sözlükten oluşturulmuş ve 5 uzunluklu kelimeler kullanılmıştır. SoftMax modelimiz için 5000 devirde 32 toptan boyutu ile eğitilmiştir. Sınıflandırma başarısı en yüksek eklem farkları $ekf^{tt'}$ özniteliğinde yaklaşık olarak %88 hesaplanmıştır.

GTU Action 3D veri kümesinde de her bir öznitelik için 300 kelimeli sözlük kullanılmıştır. Her kelimenin uzunluğu 9'dur. SoftMax için 50000 devirde 32 toptan boyutu kullanılarak eğitilmiştir. Tüm öznitelikler kullanılarak %96 doğrulukla sınıflandırma gerçekleştirilmiştir.

MSR Action 3D veri kümesi 20 hareket grubuna sahiptir. Diğer iki veri kümesine göre daha kompleks hareketleri içermektedir. Bu veri kümesi için 300 kelimeli bir sözlük kullanılmıştır. Kelimelerin uzunluğu 10'dur. Eklem koordinatlarını kullanarak elde ettiğimiz başarım diğer özniteliklere göre daha yüksektir. Bu veri kümesinde ki başarımımızın düşük olma sebebi sözlük boyutumuzun yeterince büyük olmaması olabilir.

Tablo 4.2: Özniteliklerin veri kümelerine göre başarımları.

Öznitelik Adı	**Florence Action 3D**	**GTU Action 3D**	**MSR Action3D**
ek^t	0,871	0,955	**0,772**
$ekf^{tt'}$	**0,880**	0,962	0,724
edu^t	0,620	0,963	0,653
$hepsi$	0,872	**0,965**	0,734
Feature Combinations [77]	0,94	-	0,97

Bu çalışmada hareket tanıma için gösterim tabanlı bir yöntem önermiş olduk. Çalışmanın yapıldığı dönemde literatür çalışmalarının [15] ve [44] gibi elde ettiği veri kümeleri başarımlarına yakın sonuçlar elde ettik. Ancak en yüksek başarımlı çalışmayı [77] sonuçlarımız geçememiştir.

Bununla beraber geçtiğimiz son yıllarda derin öğrenme teknikleri ile elde edilen başarımlar çok yüksektir. Ancak derin öğrenme teknikleri halen küçük boyutlu Florence Action 3D gibi veri kümelerinde iyi sonuçlar verememektedir. Gösterim tabanlı çalışmalar ise bu tarz küçük veri kümelerinde daha iyi çalışmaktadır.

Bu bölümden sonra hareket tanıma problemi için önerdiğimiz diğer derin ağ tabanlı yöntemlere yer verilecektir. Yaptığımız çalışmaların küçük veri kümeleri üzerindeki başarımları da teze eklenecektir.

5. OTO KODLAYICILAR İLE HAREKETLERİN GÖSTERİMLERİNİ OLUŞTURMA

Derin ağ yapılarının gelişmesi ile görüntüler üzerinde uzamsal ve/veya zamansal öznitelikleri çıkaran LSTM, Oto kodlayıcı, LSTM kodlayıcı, CNN ve MLP benzeri ağlar kullanılmaya başlanmıştır. Bu ağ yapıları ile başarılı gösterimler ve sınıflandırma sonuçları Tablo 2.3 deki gibi çalışmalar tarafından elde edilmiştir.

İskelet koordinat verileri, hareketin gerçekleştiği eklem bölgelerine ait 3D pozisyon verilerini içermektedir. Eklem pozisyonları hareketi tanımlayıcı veriler olmasına rağmen hareket tanıma problemi için belirli zorluklar hala mevcuttur. Zorluklara örnek olarak, farklı eklem uzunluklarına sahip insanlar ve farklı hareketlerin birbirine olan benzerlikleri verilebilir. Bu zorluklar için hareketlere ait gelişmiş öznitelikler oluşturmak hareketlerin sınıflandırma başarısını artırabilmektedir.

Bu nedenle bizde gösterim tabanlı bir Oto kodlayıcı LSTM ağı geliştirdik. Bu bölümde hareket analizi ve öznitelik çıkarma için yaptığımız bu ağ anlatılacaktır. Bu çalışma sonuçlarından beklenen başarı elde edilememiştir. Ancak bir sonraki oluşturduğumuz sisteme yol göstermesi bakımından önemlidir.

5.1. Oto Kodlayıcılar

Oto kodlayıcılar girdi ve çıktı verisi arasındaki özdeşliği bulan fonksiyonu öğrenmeye çalışan ağlardır. Esasında çalışma yapısı verilen girdiye en benzer çıktıyı üretebilen fonksiyonun bulunmasına dayanmaktadır. Bu fonksiyon farklı kısıtlamalar ile girdiye benzer çıktıyı üretebildiği için veriye ait önemli yapısal bilgileri ortaya çıkarabilmektedir. Bu kısıtlamalar girdi-çıktı verilerinin boyutuna göre daha düşük boyutta katmanlardan geçiş ya da modele ait limitler olabilmektedir.

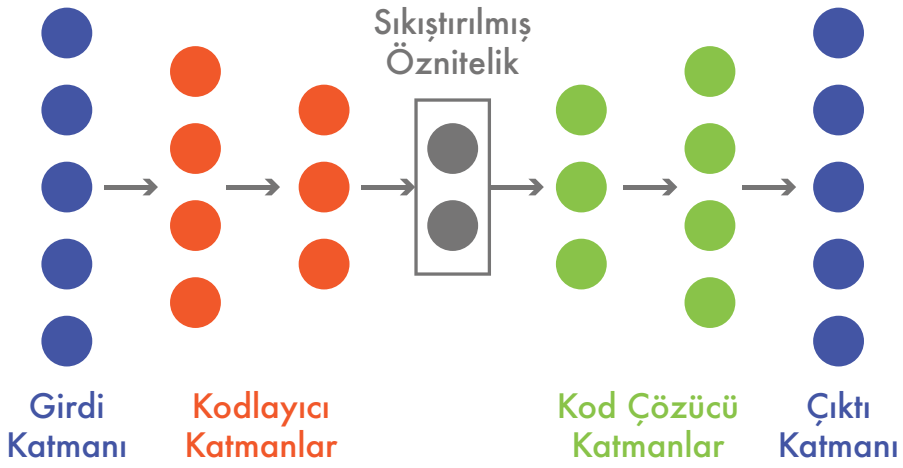
Örneğin 16x16 piksellik bir görüntüye benzer görüntü üretebilen bir oto kodlayıcı tasarlandığında, bu ağın yapısal özellikleri; katman sayısına, katmanlarda ki nöron sayısına ve nöronlardaki aktivasyon fonksiyonlarına bağlıdır. Girdi katmanında 16x16'dan 256 adet nöron bulunmaktadır. Aradaki öznitelik katmanındaki nöron sayısı, girdi katmanındaki nöron sayısından daha küçük ise mesela 100 ise, benzer 256 piksellik görüntüyü üretebilmek için ağın gerekli olan önemli görüntü özniteliklerini bu 100 nöronu kullanarak çıkarması gerekmektedir.

Oto kodlayıcılar TBA gibi özellikle veri sıkıştırmada, öznitelik çıkarmada sıklıkla kullanılmaktadır. Bunun yanı sıra veriyi daha büyük boyuta çıkarıp seyrek veri üzerinde işlem yapmak için de oto kodlayıcılar kullanılmaktadır.

Aktivasyon fonksiyonu; doğrusal veya doğrusal olmayan nöron üzerindeki fonksiyondur. Girdiye uygulanarak nöron çıktısını oluşturur.

Ara katmanlar önceki katmandan veriyi alıp nöronlardaki aktivasyon fonksiyonlarından geçirip sonraki katmana veriyi ileten yapılardır. Katmanların sayısı azaldıkça model ve çıkarılan özniteliklerin yapısı basitleşir.

Şekil 5.1'de iki katmanlı basit bir oto kodlayıcı gösterilmiştir. Girdi katmanında girdi verisinin boyutları sayısında nöron bulunur. Diğer ara katmanlarda bulunan nöronlar aktivasyon fonksiyonları ile veri üzerinde işlemler yapar. Ortada çıkarılan sıkıştırılmış öznitelik katmanı veriye ait küçük boyutta öznitelikleri temsil eder. Bu kısım gösterim oluşturmada, öznitelik çıkarmada kullanılır. Kod çözücü katmanlar sıkıştırılmış öznitelikten girdiye benzer çıktıyı üretmeye çalışır. Çıktı girdinin tamamen aynısı olmayabilir. Girdi ve çıktı arasındaki benzerlik ağdaki fonksiyonun başarılı bir şekilde bulunduğunun göstergesidir.



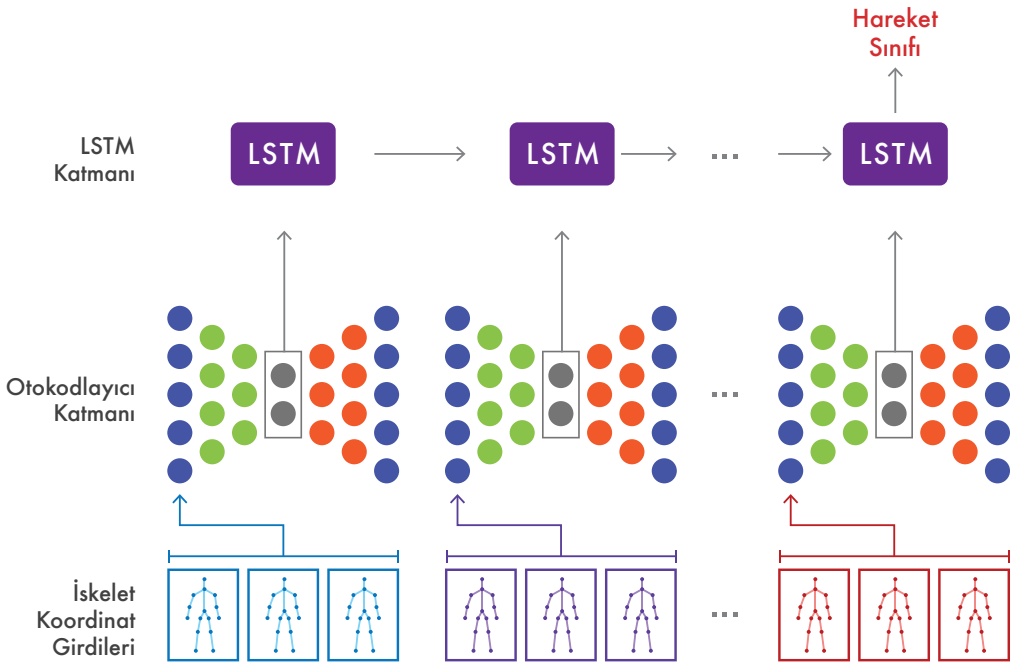
Şekil 5.1: 2 Katmanlı oto kodlayıcı genel görünümü.

Biz de hareket iskelet koordinatlarının temel özniteliklerini çıkarmak için bir oto kodlayıcı tasarladık. İskeletin uzamsal yapısıyla ilgili bilgileri çıkarıp sınıflandırma ve sonraki ağ için bunları kullanmayı amaçladık. Bu kısımla ilgili detaylar bir sonraki bölümde anlatılacaktır.

5.2. Yöntem ve Sonuçlar

Hareketleri anlık olarak veya belirli zaman periyodlarındaki hareketleri oto kodlayıcılara vererek uzamsal ve zamansal öznitelikleri elde edilmeye çalışılmıştır. Her bir *t* anındaki iskelet görüntülerini oto kodlayıcıdan geçirip LSTM ağları ile sınıflandırılmıştır. Sistemin genel görünümü

Şekil 5.2'de gösterilmiştir. İskelet koordinatlarını ardışık üçlüler halinde oto kodlayıcılardan geçirip, aradaki sıkıştırılmış öznitelikler ile LSTM ağları beslenmiştir. LSTM ağları zamansal öznitelik çıkarmak için kullanılmıştır.

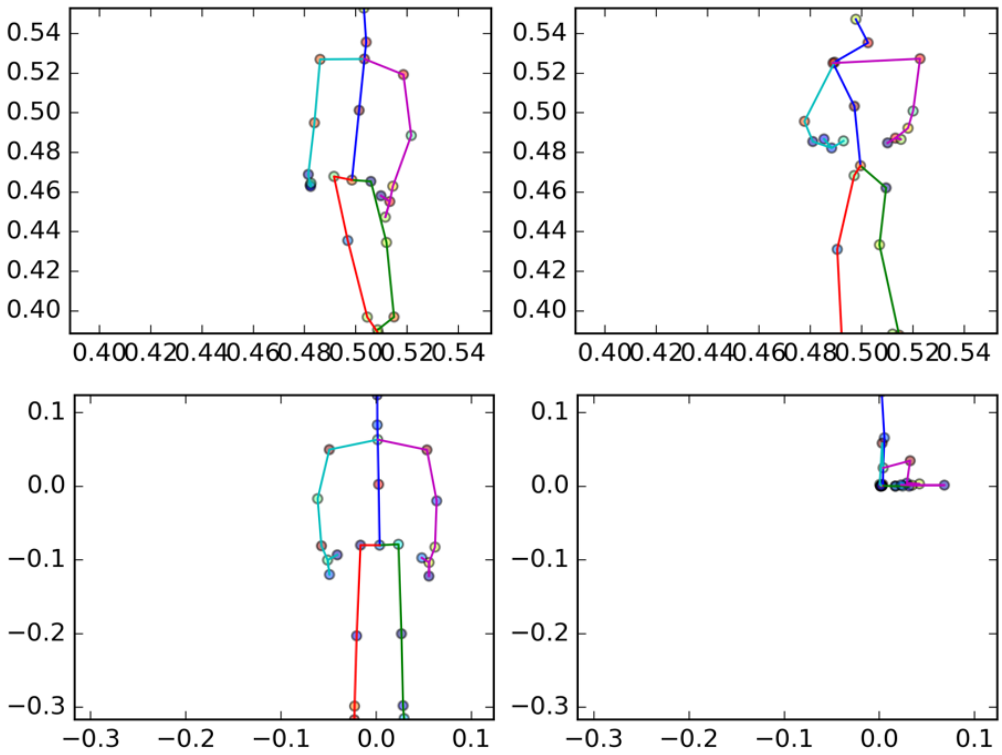


Şekil 5.2: Oto kodlayıcı LSTM yönteminin genel görünümü.

Çalışmanın oto kodlayıcı çıktıları, girdilere benzer çıkmadı. Koordinat dizilerinden oluşan girdiler üzerindeki temel ilişkiler oto kodlayıcılar ile elde edilemedi. Bir hareketin *t* anındaki iskelet koordinat girdisi ile oto kodlayıcı çıktısı aşağıda Şekil 5.3'de gösterilmiştir. Şeklin üst kısmında iskeletin girdi ve çıktı görünümü birbirine bir miktar benzemektedir ancak alttaki kısımda girdi ve çıktı arasında herhangi bir benzerlik bulunmamaktadır. Benzerliğin azlığı sınıflandırma başarısını da düşürmektedir.

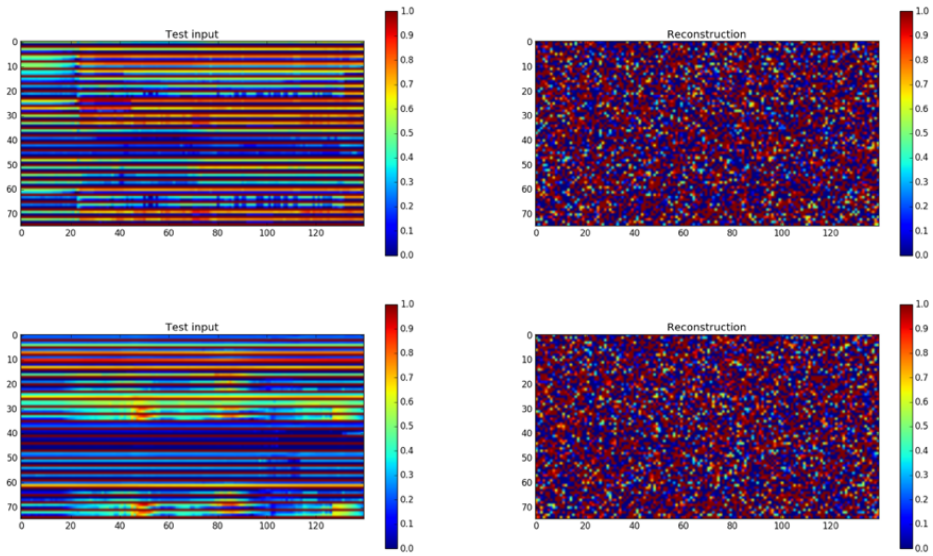
Şekil 5.4'te iki farklı hareket oto kodlayıcıya verilmiştir. Hareket girdisinin çıktıları ve hata fonksiyonun değerinin yüksekliği, belirli periyotlardaki hareketler ile beslenen oto kodlayıcı ağlarının başarılı çalışmadığını göstermektedir. Hareketlerden

ilki oturma hareketi iken ikincisi saç tarama hareketidir. Normalleştirilmiş girdiler 1 ile 0 arasında ki değerlerine göre renklendirilmiştir.



Şekil 5.3: Test anında iki iskelet görüntüsünün oto kodlayıcı girdisileri ve çıktıları sağ taraf girdiler, sol taraf çıktıları, üsteki örnek çıktısı girdiye bir miktar benzerken alttaki örnek çıktısı girdiye benzememektedir

Başarılı şekilde oluşturamadığımız iskelet gösterimlerinin LSTM sınıflandırıcı çapraz denek başarımları %15 civarında gelmiştir. İskelet koordinatlarını herhangi bir ön işlem uygulamadan beslediğimiz LSTM ağlarının sınıflandırma başarımları daha yüksek gelmiştir. LSTM ağlarının çalışma yapılarında ve başarımlarına sonraki bölümlerde yer verilecektir.



Şekil 5.4: Test anında üst kısımdaki oturma ve alt kısımdaki saç tarama hareketlerinin oto kodlayıcı girdileri ve çıktıları

6. DERİN AĞLAR İLE HAREKETLERİN SINIFLANDIRILMASI

Bölüm 5'te oto kodlayıcılar ile başarılı bir gösterim elde edilemeyince, bu bölümde temel derin ağları kullanarak, hareket verileri analiz edilip hareketler tanınmaya çalışılmıştır. Derin ağların otomatik olarak öznitelik çıkarabildiği pek çok çalışma da gösterilmiştir [79]. Zamana bağlı iskelet görüntülerinden oluşan insan hareketi verileri de bu derin ağ yapılarının doğasında olan işlemler ile otomatik sınıflandırılabilmektedirler. Bu kısımda temel derin ağlar sınıflandırıcı olarak kullanılmış ve sonuçlar paylaşılmıştır.

6.1. SoftMax Sınıflandırıcı

Bu kısımda makine öğrenmesinin en temel sınıflandırıcılarından olan DVM ve SoftMax sınıflandırıcıları denenerek karşılaştırılmıştır.

DVM: doğrusal veya doğrusal olmayan fonksiyonlarla veriler üzerinde sınıflandırma yapmaktadır. Eğitim verilerindeki herhangi bir noktadan en uzak olan iki sınıf arasında bir karar sınırı bulur. Bulunan sınır iki sınıfın da tüm noktalarına uzaklıklarını eşit miktarda maksimize etmektedir. Denklem 6.1'de DVM hata hesaplama fonksiyonu gösterilmiştir. $s_j = f(x_i, W)_j$ j-inci sınıfın skoru olmak üzere, i-inci örneğin verisi x_i,sınıf etiketi y_i'dir. Δise yanlılık için kullanılan hiper parametredir.

DVM'lerde kullanılan çekirdek fonksiyonu başarımı büyük miktarda etkilemektedir. Çekirdek fonksiyonu verinin yapısal özelliklerine göre seçilmelidir. Hareketleri sınıflandırmak için hem doğrusal çekirdek fonksiyonunu hem de Gauss çekirdek fonksiyonunu denenip sınıflandırma sonuçları Tablo 6.1'de gösterilmiştir.

$$L_{DVM} = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + \Delta) \tag{6.1}$$

SoftMax: İkili Lojistik regresyonun çok sınıf için genelleştirilmiş halidir. Denklem 6.2'de gösterilen skor fonksiyonu sayesinde sınıflara ait benzerlik skoru veren DVM'in aksine normalleştirilmiş sınıf olasılıklarını vermektedir. f_j j-inci elementin tüm sınıf skorlarının vektörüdür.

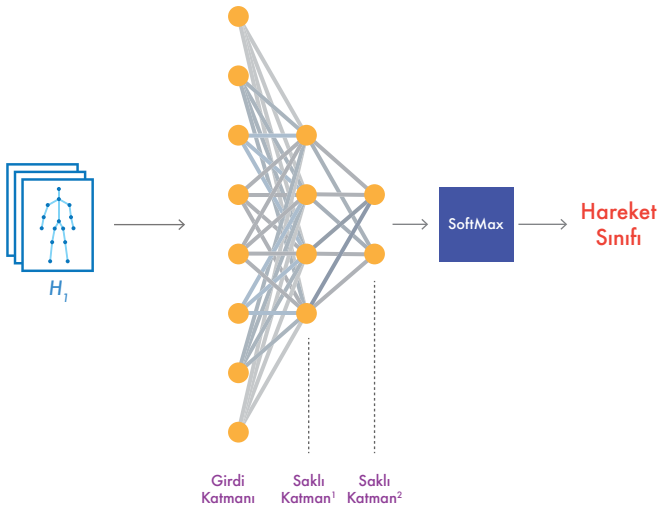
Hareketleri sınıflandırmak için SoftMax ağı hem kendi başına hem de diğer ağ yapılarının son katmanlarında kullanılmıştır.

$$L_{SoftMax} = -\log\left(\frac{e^{f_{y_i}}}{\sum_{j \neq y_i} e^{f_j}}\right) \tag{6.2}$$

6.2. Çok Katmanlı Yapay Sinir Ağları MLP

İleri beslemeli yapay sinir ağlarıdır. Birden fazla ara katmana sahiptirler. Her bir katmandan geçildikten sonra skor hesaplanır. Skor değerine göre geri yayma yapılır. Her geri yayma, hesaplanan skor değerini küçültmeye yönelik parametre optimizasyonudur. Her adımda parametreler güncellenir ve model eğitilir. Skor fonksiyonu modelin ne kadar doğrulukla çalıştığını gösteren değeri üretmektedir.

Her bir katmanda bulunan nöronlar doğrusal olmayan aktivasyon fonksiyonlarına (sigmoid ve tanjant gibi) sahiptirler. Hareketlerin sınıflandırılması için Şekil 6.1'de gösterilen sistem oluşturulmuştur. Ara katmandan elde edilen uzamsal öznitelikler SoftMax ile sınıflandırılmış sonuçları Tablo 6.1'e eklenmiştir.



Şekil 6.1: Çok Katmanlı Yapay Sinir Ağı ile hareketlerin sınıflandırılması.

6.3. Aşırı Öğrenme Makinası (ELM)

Sınıflandırma, regresyon [80], kümeleme, sıkıştırma veya öznitelik öğrenme için kullanılan ileri beslemeli sinir ağlarıdır [81]. Tek veya çok katmanlı olabilirler.

Parametreler iyileştirilmeye çalışılmaz, geri yayma yoktur. Parametreler rastgele atanır veya bir önceki katmandan alınırlar [82]. Literatürde DVM'lere göre daha iyi çalıştığı gösterilmiştir. Geri yayma yapan ağlara göre oldukça hızlı çalışırlar. Çoğu durumda tek katmanlı olarak kullanırlar. Ancak veriyi farklı bir uzaya çıkaran pek çok çekirdek fonksiyon kullanılabilmektedir. Sigmoid, tanjant, bulanık, çoklu kuadratik, Gauss vb. dahil olmak üzere birçok aktivasyon fonksiyonu bu ağlarda kullanılabilmektedir. Ayrıca DVM'lerdeki gibi çekirdek fonksiyonları bu ağlarda mevcuttur.

Aşırı Öğrenme Makinası seçilen hareket veri kümelerinde diğer derin ağlarla kıyaslanmak için kullanılmıştır. Normalize edilmiş hareket verileri tek katmanlı ELM ağları ile sınıflandırıp, sınıflandırma sonuçları Tablo 6.1'e eklenmiştir.

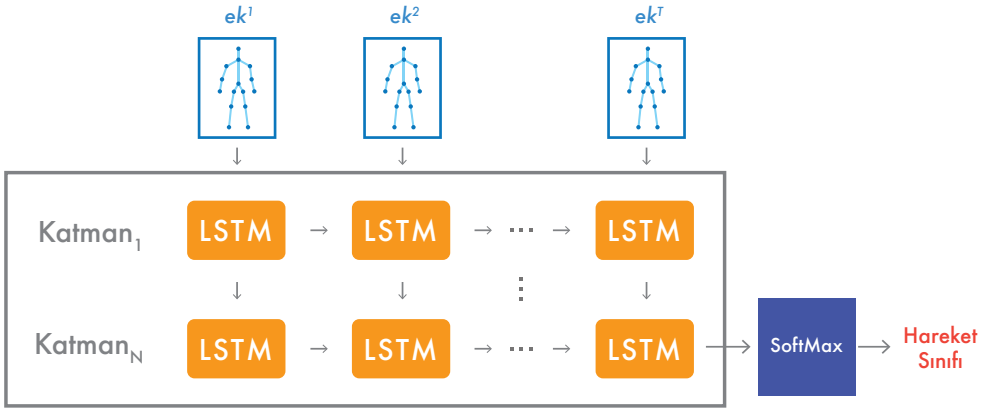
6.4. LSTM Ağları

Zamana bağlı, seri oluşturan veriler için kullanılan tekrarlı sinir ağlarıdır. Doğadaki çoğu veri zamana bağlı veya sıralıdır. Konuşurken kurulan cümleler, dinlenen şarkılar, kaydedilen videolar, sinyaller gibi pek çok veri sıralı olarak zamana bağlı ilerlemektedir. Bu veriler bir bütün halinde anlam ifade etmektedir. Koşma hareketi, başlangıç anından bitiş anında kadar oluşan görüntü dizilerinin tamamıyla koşmayı tanımlar.

Klasik yapay sinir ağları bir görüntü verisi üzerindeki uzamsal filtreleri çıkarır. Bu görüntünün öncesindeki görüntü ile ilgilenmez. RNN'ler ise bu yapılardan farklı olarak içinde veri devamlılığını sürekli sağlayan döngülere sahiptir. Bu döngüler, verinin bir *t* anından diğer bir *t'* anına geçmesine izin verir. Döngüler açıldığında her biri, sinir ağlarına benzer RNN hücreleridir. Zincir halinde sıralanan yapısı, liste ve sıralı veriler için oldukça uygundur. Her bir hücre bir *t* anındaki veri girdisini alırken, öncesindeki hücreden geçmiş *t* anlarına ait bilgileri de almaktadır. Bu özelliği ile diğer yapay sinir ağlarından ayrılmaktadır.

LSTM'ler RNN'lerin özelleşmiş halidir. Her bir LSTM hücresi döngü halinde kullanılarak LSTM ağlarını oluşturur. LSTM hücrelerinde bulunan kapılar sayesinde girdi verisinin ve önceki hücreden aktarılan verinin akışı sağlanır. Bir LSTM hücresinde; girdi kapısı, çıktı kapısı, sigmoid kapısı, unut kapısı gibi kapılar bulunabilir. Bu kapıların fonksiyonları ve yapıları gerektiği gibi değiştirilebilir. Girdi kapısı, girdi verisini aktarmak için, çıktı kapısı, çıktı verisini aktarmak için kullanılır.

Unut kapısı önceki hücreden gelen verinin ne kadarının unutulması gerektiğine karar vermek için kullanılır. Sigmoid kapısı, çıktının sigmoid fonksiyonuna verilip son halini alması için kullanılabilir. Sigmoid yerine tanjant fonksiyonu da kullanılabilir. Kapılar artırılabilir ve azaltılabilir. Tıpkı klasik sinir ağlarında ki gibi içlerindeki fonksiyonlar değiştirilebilirdir.



Şekil 6.2: LSTM ağları ile hareketlerin sınıflandırılması.

İnsan hareketi verileri de farklı zaman aralıklarında ve farklı sürelerde kaydedilmiştir. LSTM ağlarının doğal yapısına uygun zamana bağlı sıralı verilerdir. Klasik sinir ağları girdi verilerinin boyutlarını sabit olarak almaktadır. LSTM'ler ise içlerinde farklı sayıda hücre ile sabit boyutlu olmayan verilerde de çalışabilmektedir. Bu nedenle her hareketin farklı uzunluklarda olması bu ağlarda herhangi bir problem oluşturmamaktadır.

Bu kısımda tek katmanlı LSTM kullanılarak hareketler sınıflandırılmıştır. Normalleştirilmiş her bir iskelet görüntüsü bir LSTM hücresine verilmiştir. Bu sayede her görüntü, sonrasındaki LSTM hücresini etkilemektedir.

Şekil 6.2'de *N* katmanlı LSTM sınıflandırıcı gösterilmiştir. Son hücreden çıkan vektör SoftMax'e verilip, sınıflandırma sonuçları Tablo 6.1'e eklenmiştir.

6.5. Sınıflandırıcıların Sonuçları

Tablo 6.1'de bahsedilen sınıflandırıcıların hareket verilerindeki başarımları gösterilmiştir. İlk satırda yer alan destek vektör makinaları radyan tabanda Gaus çekirdek fonksiyonunu kullanmaktadır. İkinci satırda ise doğrusal bir çekirdek

fonksiyonu kullanılmıştır. Doğrusal DVM'in başarısı diğer sınıflandırıcılara göre oldukça yüksektir.

SoftMax modelimiz için eşit boyutta girdi vektörü beklenmektedir. [83] Çalışmasındaki gibi hareketlerin eksik sayıdaki iskelet görüntüleri yerine sıfır değerli boş görüntüler eklenmiştir. SoftMax modülü her veri kümesine bağlı olarak farklı epoklarda 32 toplu iş boyutunda çalıştırılmış sonuçlar üçüncü satırda gösterilmiştir. Doğrusal DVM'in başarımını geçememiştir. Daha yüksek epoklarda eğitildiğinde aşırı öğrenme sorunu ile karşılaşılmaktadır.

Tablo 6.1'in dördüncü satırında Aşırı Öğrenme Makinaları ile elde edilen sınıflandırma sonuçları gösterilmiştir. Çoklu kuadratik aktivasyon fonksiyonu RBF de çekirdek 256 adet nöronlu 1 katmanlı bir yapı tasarlanmıştır.

Çok katmanlı sinir ağları için 2 katmanlı basit bir ağ tasarlanmıştır. İlk katmanda 128 nöron ikinci katmanda 64 nöron bulunmaktadır. Başarımları Tablo 6.1'de beşinci satırda gösterilmiştir. Verilerin miktarı nedeniyle aşırı öğrenme problemi ortaya çıkmıştır. Modelden daha başarılı sonuçlar yapı ile ilgili daha fazla değişiklik yapıldığında elde edilebilirdi. Ancak bu tez kapsamında biz LSTM yapılarının daha doğru kullanılmasıyla ilgilenmekteyiz.

LSTM ağları herhangi bir veri değişikliğine (sıfırla görüntü doldurma gibi) gerek kalmaksızın çalışmaktadır. Basit 128 nöronlu bir katmanlı LSTM ağından elde edilen sınıflandırma sonuçları altıncı satırda gösterilmiştir. Diğer sınıflandırıcılara göre oldukça başarılı skorlar elde edilmiştir. Bunun temel nedeni hareket verilerinin sıralı iskelet görüntülerinden oluşmasıdır. LSTM ağlarının doğal yapısına hareket verilerinin uyduğu başarımlardan anlaşılmaktadır.

Tablo 6.1: Sınıflandırıcıların veri kümeleri başarımları.

Veri Kümeleri / Sınıflandırıcılar	GTU Action 3D	Florence Action 3D	MSR Action 3D	NTU RGB+D
SVM	0,48	0,23	0,12	
Doğrusal SVM	0,844	0,731	0,583	
SoftMax	0,759	0,616	0,47	
ELM	0,745	0,655	0,226	
MLP	0,60	0,54	0,36	
1-K LSTM	**0,902**	**0,77**	**0,70**	**0,59**

Bu bölümde derin ağları sınıflandırıcı olarak kullanarak hareket tanıma problemine uygun ağ yapılarını bulmayı amaçladık. Hareketlerin zamana bağlı verilerden oluşması bize en uygun yapının bu ağlar içinde LSTM ağları olduğunu gösterdi. Bundan sonra gerçekleştireceğimiz çalışma LSTM ağlarının ortaya çıkardığı metrikleri kullanmaktadır. Bu metrikler hareketlerin hem zamansal bilgilerini hem de başka hareketlerle ilişkilerini içermektedir. Çalışmanın detayları sonraki bölümde anlatılacaktır.

7. İKİZ LSTM AĞLARI İLE HAREKET TANIMA

Bu bölümde tezin ana çalışması [16] olan çoklu ikiz LSTM ağlarından bahsedilecektir. Derin öğrenme tabanlı yöntem dört ana başlıkta incelenecektir. Bu başlıklar;

i) İkiz LSTM Ağı
ii) Hareket İkili Girdileri
ii) İkiz LSTM Ağları için Sınıflandırıcı Modülü
iv) Uçtan uca ikiz LSTM Sınıflandırıcı Ağı

İkiz ağlar, iki tane derin ağ yapısını içinde barındıran ileri beslemeli ağlardır. İçerisinde bulunan derin ağ yapısının öğrendiği parametreleri paylaşarak ileri ve geri yönlü besleme yapar. Bu ağlar iki girdi arasındaki ilişkiyi öğrenmektedirler. Bu ilişkiye ait metrikler sınıflandırma, eşleme, gruplama gibi pek çok amaçla kullanılabilmektedirler. Bizde hareketleri analiz etmek amacıyla bu ağları kullandık. Bu sayede hem benzerlikleri bulabilen hem de hareketleri tanıyan bir ağ geliştirmiş olduk.

Hareket verilerine en uygun model yapısının LSTM ağları olduğu Bölüm 6'da sınıflandırma başarımlarından anlaşılmıştır. Bu nedenle ikiz ağın içinde çok katmanlı LSTM ağları kullanılmıştır. İki hareket her bir LSTM bloğuna verilerek eğitilmiştir. İkiz LSTM ağı bu hareket verileri arasındaki benzerliği öğrenmektedir. Bu ağın detaylarına Bölüm 7.1'de değinilecektir.

İkiz ağlar iki adet girdiyi alır ve bu girdiler arasındaki ilişkiyi öğrenmeye çalışırlar. Bu girdiler görüntü, video sinyal gibi pek çok farklı tipte veri olabilir çünkü ikiz ağın içinde bulunan modelin yapısı da değişkendir. Bölüm 7.2'de İkiz LSTM ağları için girdi ikililerinin nasıl oluşturulduğu anlatılacaktır.

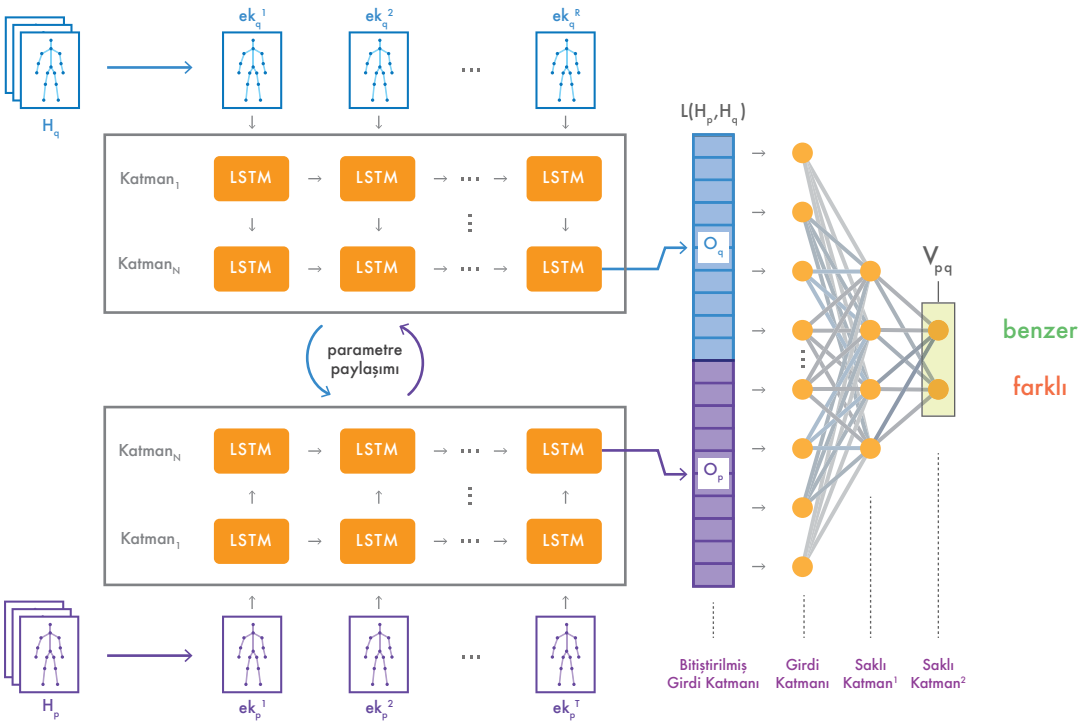
İkiz LSTM ağları hareketler arasındaki ilişkiyi bulmaktadır, ancak hareket tanıma problemi temelde hareketlerin sınıflarını bulmaya dayanmaktadır. Bu doğrultuda bizde ikiz LSTM ağının çıktılarını kullanarak hareketleri tanıyan bir sınıflandırıcı modül geliştirdik. Bölüm 7.3'te bu modül ile ilgili detaylar yer alacaktır.

Bölüm 7.4'te sınıflandırıcı modülü ile ikiz LSTM modülünü birleştirip oluşturduğumuz uçtan uca çalışan ikiz LSTM DML sınıflandırıcı ağından

bahsedilecektir. Son bölümde ise bu ağlardan elde edilen başarımlar paylaşılıp yorumlanacaktır.

7.1. İkiz LSTM Ağı

Bu bölümde, hareketlerin derin metrik bilgilerini öğrenerek birbirlerine benzeyip benzemediklerini bulan İkiz LSTM ağı anlatılacaktır. Veriler arasındaki benzerlik bilgisi birçok uygulama için gerekli ve kullanışlıdır [84]. Veri kümeleri içindeki hareketlerin her birinin arasındaki metrikleri bulma problemi klasik sınıflandırma probleminden çok daha genel bir problemdir. Verilerin sınıflarını bulma yerine onları analiz etme imkanı sunar. Bu sebeple bu bölümde, 3B hareket ikililerini girdi olarak alıp, aralarındaki benzerlik metriklerini öğrenen bir ikiz ağ tasarlanmıştır.



Şekil 7.1: Hareket ikilileri arasındaki benzerliği bulan İkiz LSTM ağı.

Şekil 7.1'de bu modülün genel yapısı gösterilmiştir. Bu modül iki adet normalleştirilmiş 3B hareketi, $H_p = \{ek_p^1, ek_p^2, ek, \ldots ek_p^T\}$ ve $H_q = \{ek_q^1, ek_q^2, ek_q^3, \ldots ek_q^R\}$, girdi olarak almaktadır. T ve R sırasıyla, H_p *ve* H_q hareketlerinin sahip olduğu toplam görüntü sayılarıdır. $ek_p^t = \{j_1^t, j_2^t, \ldots j_N^t\}_p$ ise t anında N adet 3B eklemden

oluşan tek bir iskelet görüntüsüdür. $j_n^t = \{x_n, y_n, z_n\} \in \mathbb{R}^3$, ise 3B tek bir eklem noktasıdır.

T ve *R* her bir hareket için farklı değerde olabilir. Bu nedenle metrik öğrenen modülümüzün temel yapı taşı olarak LSTM hücreleri kullanılmıştır. 2 LSTM bloğundan her biri bir hareketi alır ve bir çıktı vektörü üretir. Oluşan bu iki çıktı vektörü $O_p \in \mathbb{R}^M$ and $O_q \in \mathbb{R}^M$ 'dir. Bu vektörlerin boyutları *T* ve *R*'nin aksine sabit ve eşittir.

Bu iki O_p and O_q vektörlerinin bitiştirilmiş çıktılarını $L(H_p, H_q) \in \mathbb{R}^{2M}$ fonksiyonu ile üretilmiştir. $L(H_p, H_q)$, iki harekete ait derin benzerlik özniteliklerini taşımaktadır. Bu vektör ile 2 katmanlı bir sinir ağı denklemde gösterildiği gibi beslenmiştir.

$$D(L(H_p,H_q))=V_{pq}, \tag{7.1}$$

D çok katmanlı ağ olup, hareketlerin birbirine benzeyip benzemediklerini belirten 2 boyutlu bir $V \in \mathbb{R}^2$ vektörü üretmektedir.

$$V_{pq} = Softmax(b^3 + W^3 ReLU(b^2 + W^2 ReLU(b^1 + W^1 L(H_p,H_q)))), \tag{7.2}$$

W network ağırlıkları, *b* yanlılık parametresi, ReLU olmak üzere V_{pq} model formülü Denklem 2'de gösterilmiştir.

İkiz LSTM modülü hareket ikililerini kullanarak derin benzerlik metriklerinin çıkarılıp öğrenilmesini amaçlamaktadır. Bu metrikler kullanılarak sınıflandırma, hareketlerin analizi, etiketsiz gruplama gibi pek çok farklı uygulama geliştirilebilir. Sınıflandırma yapan modellere göre çok daha genelleştirilebilir bir modeldir.

7.2. Hareket İkililerinin Oluşturulması

Bu bölümde ikiz LSTM ağlarının girdileri olarak kullanılacak hareket ikililerinin oluşturulmasından bahsedilecektir. Paralel olarak eğitilen iki LSTM modülünün her biri bir hareketi girdi olarak almaktadır. İkiz LSTM ağlarının etiketleri 1 veya 0 olabilir. Hareket ikililerinin sınıfları aynı olduğu durumda etiketler 1 aynı olmadığı durumda etiketler 0 değerini almaktadırlar. Benzer ikililere pozitif ikililer, benzer olmayan ikililere negatif ikililer diyebiliriz.

İkiz ağların başarılı eğitilmesinde negatif pozitif ikili oranının rolü büyüktür. Biz bu ağları eğitirken oranı veri kümelerindeki sınıf miktarlarına göre seçtik. Örneğin

Florence Action 3D veri kümesinde 9 farklı hareket sınıfı mevcuttur. Bu nedenle her eğitim hareketini bir pozitif sekiz negatif hareketle eşleyerek ikilileri oluşturduk. Ne kadar miktarda ikili oluşturacağımıza veri kümesinin büyüklüğüne bakarak karar verdik. Küçük boyuttaki veri kümelerinde oluşabilecek tüm ikilileri kullanırken büyük veri kümelerinde rastgele seçilimler ile ikili sayısını azalttık. Veri kümelerindeki ikili miktarlarının toplam hareket miktarından daha fazla olması derin ağların büyük veri ihtiyacı problemini gidermektedir. Ayrıca doğrulama kümesi için ekstra oluşturduğumuz ikililer, eğitim verilerinde bir azaltma meydan getirmemiştir.

7.3. İkiz LSTM Ağları için Sınıflandırıcı Modülü

Daha önce belirttiğimiz gibi, hareket tanıma sistemlerinin çıktılarının hareket sınıfı olmaları gerekmektedir. İkiz LSTM modülünün çıktısı hareket sınıfları değil hareket benzerliklerini belirten 2 boyutlu bir vektördür. Sınıflandırıcı modülünün görevi, bir test hareketini H_p diğer hareketlerle $H_{q1},H_{q2},H_{q3},H_{q4},\ldots, H_{qk}$ k diğer hareketlerin sayısı olmak üzere, karşılaştırıp hareketlerin sınıflarını bulmaya çalışmaktır. Şekil 7.2'de gösterildiği gibi İkiz LSTM ağlarının çıktılarını V_{pq_1}, V_{pq_2}, V_{pq_3},... V_{pq} bitiştirerek bir $G \in R^{2k}$ vektörü elde edilmektedir. Bu G vektörü ile beslenen sınıflandırıcı modülün çıktısı hareketlerin çıktısıdır. Burada kullanılan sınıflandırıcı olarak KNN ve DVM denenmiştir.

Önceki İkiz LSTM modülü, hareket çiftlerini girdi olarak almakta ve çıktı olarak benzer-farklı etiketleri vermektedir. ikiz LSTM modülünün eğitilmesinde hareket sınıflarına ihtiyaç yoktur. Ancak bu sınıflandırıcı modülü için hareket sınıflarına ihtiyaç duyulmaktadır.

7.4. Uçtan Uca İkiz LSTM DML Sınıflandırıcı Ağı

Hareketlerin birbiriyle olan ilişkilerinin öznitelikleri, ikiz LSTM ağlarının öğrendiği derin metriklerden çıkarılabilmektedir. Derin metrikleri çıkarılan hareketler üzerinde sınıflandırma, gruplama ve hatta veri birleştirmesi yapılabilir.

Biz bu derin metrikleri çıkaran ağ bloklarını hareket tanıma problemi için kullandık. Geliştirdiğimiz Uçtan Uca İkiz LSTM ağı ile hareketlerin sınıflandırmasını yaptık.

Şekil 7.3'de İkiz LSTM bloklarını içeren, uçtan uca eğitilebilen sınıflandırıcı modelin genel yapısı gösterilmiştir.

Birden fazla ikili için (H_p,H_{q1}), (H_p,H_{q2}) ,..., (H_p,H_{qk}) girdilerinin her birinin $L(H_p,H_{q1})$,$L(H_p,H_{q2})$, ..., $L(H_p,H_{qk})$ blok çıktıları elde edilmiştir. Burada k değeri bir örnek girdi için gereken ikili sayısıdır, bir başka deyişle ikili listesinin uzunluğudur. Bundan dolayı k her veri kümesindeki hareket tipi sayısıdır.

İkiz LSTM bloğunun Bölüm 7.2'deki son ara katmanını ve SoftMax katmanını kaldırarak Denklem 3'te gösterilen D' ağını elde ettik.

$$D' = (Tanh(b^1 + W^1 L(H_p,H_q))) = V_{pq}, \tag{7.3}$$

D' ağında ReLU kullanmak yerine daha iyi sonuç verdiğini gözlemlediğimiz hiperbolik tanjant fonksiyonunu kullandık. Buradaki D' ağının çıktısı, her bir ikilinin derin benzerlik metrik vektörleridir. Bu vektörler (H_p,H_{q1}) hareket girdileri için V_{pq1} olarak ifade edilmektedir.

C bitiştirilme operasyonu olmak üzere, Denklem 4'te gösterildiği gibi bu vektörler bitiştirilip G vektörü elde edilir.

$$G = C(V_{pq1}, V_{pq2}, \dots, V_{pqk}) \tag{7.4}$$

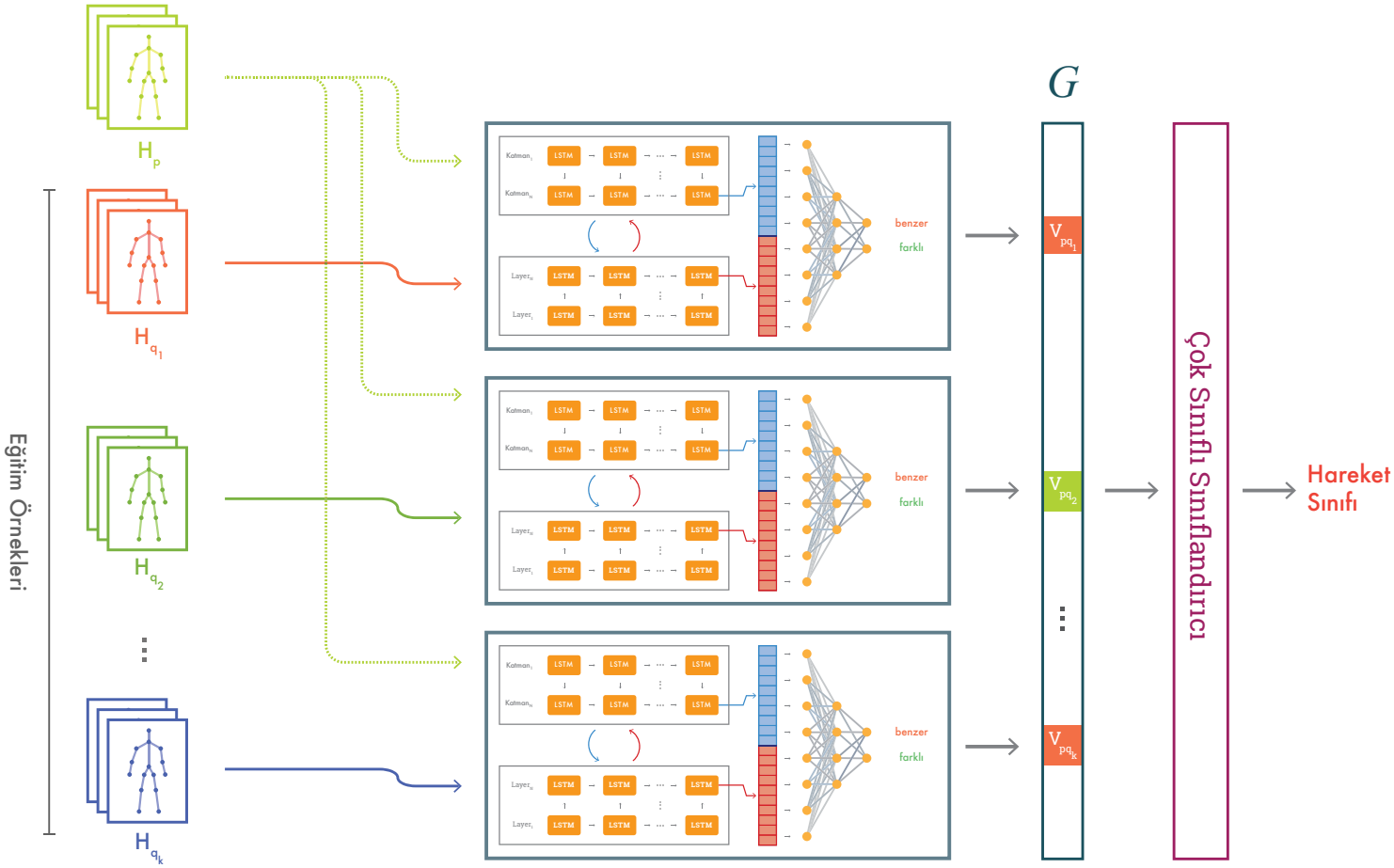
$$S = Softmax(Tanh(b^1 + W^1 G)), \tag{7.5}$$

Denklem 5'te tanımladığımız S ağına G vektörü verilmiştir. S, tek ara katmanlı yapay sinir ağıdır. Bu ağın çıktısı hareketin ait olduğu sınıftır.

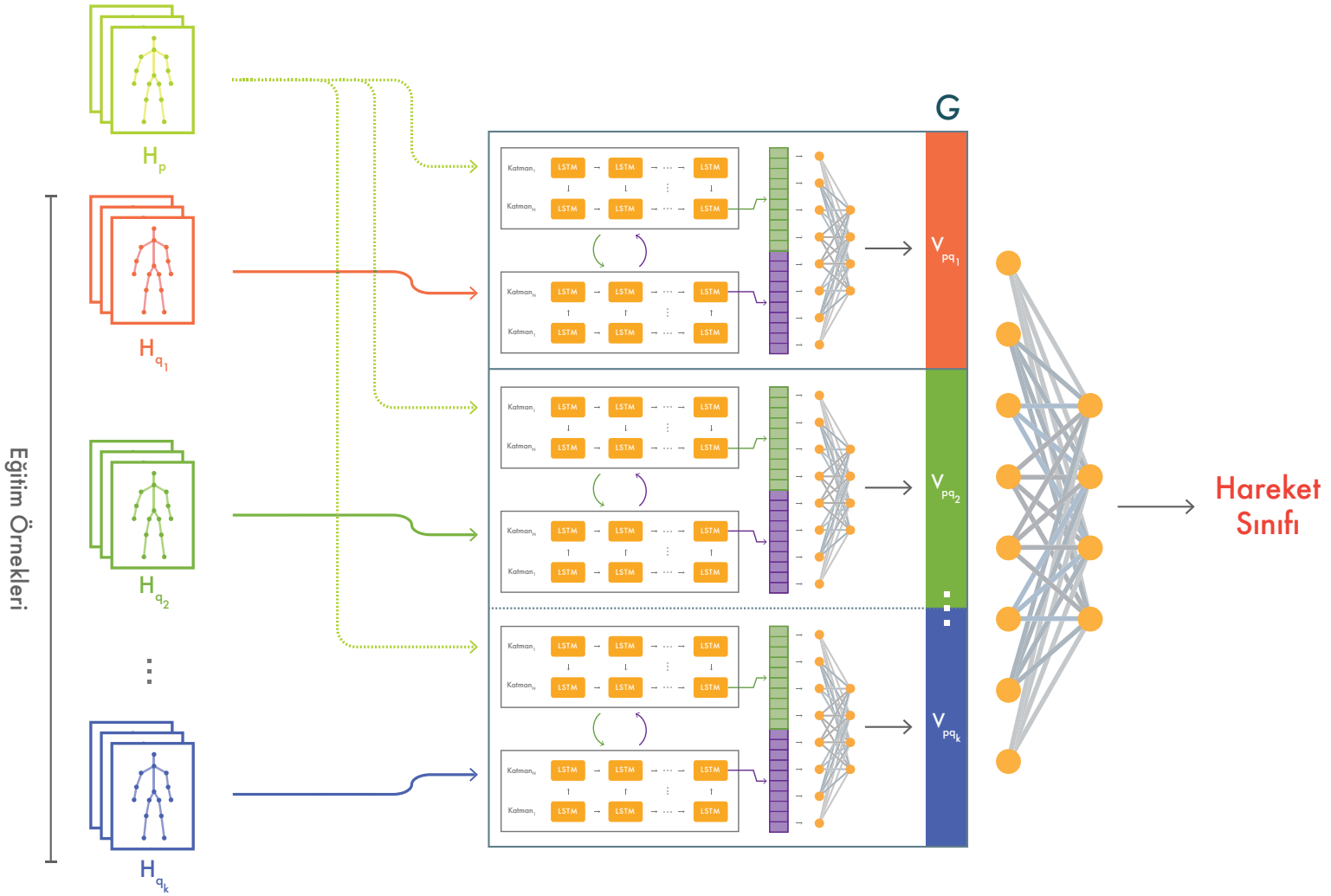
İkiz LSTM ağı girdi hareketlerinin sınıflarının birbiri ile aynı olup olmadığını öğrenirken, Uçtan Uca İkiz LSTM DML ağı hareketlerin sınıfını öğrenir. İkiz LSTM ağları farklı veri kümelerini birleştirip, beraber kullanabilir. Çünkü ikiz LSTM ağının hareket sınıf etiketine ihtiyacı yoktur. Fakat Uçtan Uca İkiz LSTM DML ağı her bir veri kümesi için ayrı ayrı eğitilmesi gerekir.

Uçtan Uca İkiz LSTM DML ağı, içindeki ağ bloklarının tümüyle beraber eğitilir. Her ileri yönlü beslemede skor fonksiyonu, en son katmanda elde edilen

vektörün standart SoftMax hata sınıf olasılık değerini verir. Bu değere göre geri yayma ile parametreler güncellenir.



Şekil 7.2: İkiz LSTM DML sınıflandırıcı ağı genel gösterimi.



Şekil 7.3. Uçtan Uca İkiz LSTM DML sınıflandırıcı ağı genel gösterimi.

7.5. Sonuçlar

Bu bölümde tez kapsamında yapılan çalışmaların kullanılan veri kümeleri üzerindeki sonuçları ayrı ayrı bölümlerde verilmiştir. Tez boyunca gerçekleştirilen çalışmaların başarımları ile literatürdeki çalışmaların başarımları kıyaslanmıştır.

7.5.1. GTU Action 3D Sonuçları

GTU Action 3D, 14 farklı eylem sınıfından oluşan 508 harekete sahip iskelet tabanlı bir veri kümesidir. Veri kümesi iç mekan gündelik hareketlerinden ve aerobik hareketlerinden oluşmaktadır. Bu veri kümesi tez çalışmaları için Kinect 2 cihazı ile toplanmıştır. Bu bölümde, tez boyunca yapılan çalışmaların GTU Action 3D üzerindeki başarımları gösterilecektir.

Bölüm 4'te bahsedilen gösterim tabanlı yöntemin başarımı Tablo 7.1'in ilk satırında gösterilmiştir. Bu başarım Bölüm 4'teki geometrik özniteliklerinin tamamının kullanılmasıyla elde edilmiştir. Başarım sonucu, özniteliklerin başarılı şekilde çıkarıldığını göstermektedir.

İkinci ve üçüncü satırda makine öğrenmesinin temel denetimli sınıflandırıcısı olan DVM denenmiştir. Doğrusal DVM başarımından veri kümesinin çok zorlayıcı hareketler içermediği anlaşılmaktadır. Burada amaç olabilecek en yüksek doğrulukla bu hareketleri tespit edebilmektir.

Tablo 7.1' de çok katmanlı LSTM ağlarının başarımının öncesinde denenen derin ağlara göre çok daha yüksek olduğu görülmektedir. Buradaki başarımdan daha yüksek bir sonuç elde etmek için oluşturduğumuz, bu bölümde yer alan İkiz LSTM DML ağının sonuçları da son satırda yer almaktadır. Kullanılan diğer sınıflandırıcı yöntemlerinin hepsinden daha yüksek bir başarım elde edilebilmiştir.

İkiz-LSTM DML ağının yapısı sırasıyla; {LSTM (75,200,2) - LSTM (200,200,2) - CONCAT (400,1) - FC (400,300) – ReLU - FC (300, 150) - ReLU-FC (300,50) - ReLU-FC (50,2)}.

Burada iki katmanlı LSTM ağlarının bir katmanlı LSTM ağından daha başarılı çalıştığı tespit edilmiş ve iki katmanlı LSTM ağı kullanılmıştır. Katman sayısı arttıkça eğitim süresi artmaktadır. Bu nedenle üç katmanlı LSTM denenmiş sonuçlarda büyük bir iyileşme göstermeyip, performansı düşürdüğü için kullanılmamıştır. Bu veri kümesinin bir iskelet üzerindeki 3B eklem sayısı 25 olduğu

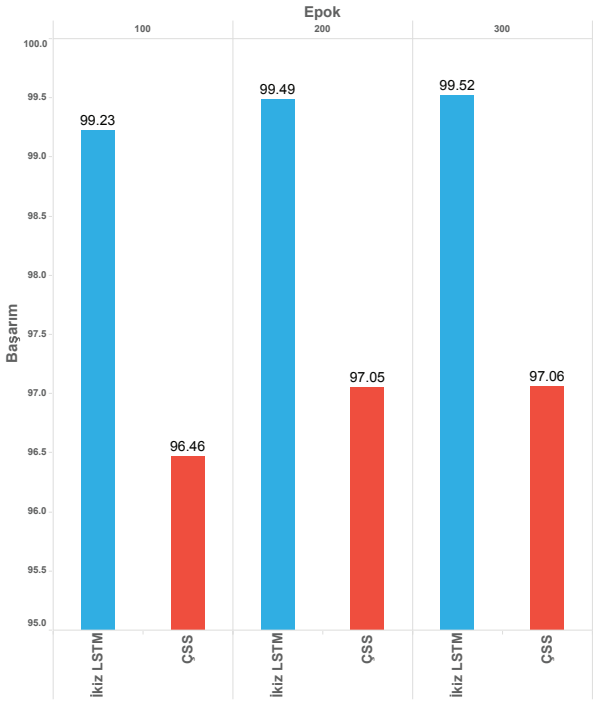
için LSTM girdisi 75 olarak verilmiştir. 2 katmanlı LSTM'de 200 adet nöron bulunmaktadır. İki adet hareketin her bir andaki iskelet verileri ayrı ayrı LSTM hücrelerine verilmektedir. LSTM ağlarının son katmanlarının son hücrelerinden çıkan çıktı vektörleri bitiştirilip, sırasıyla 3 katmanlı 300,150,50 nöronlu tam bağlı sinir ağına verilmiştir. Ağın yapısında bulunan FC, tam bağlı bir sinir ağı katmanını ifade etmektedir. ReLU bu katmanın nöronlarındaki aktivasyon fonksiyonu, CONCAT ise bitiştirme fonksiyonu demektir.

Tablo 7.1. GTU Action 3D veri kümesi üzerinde elde edilen başarımlar.

Yöntem	**Başarım**
Geometrik eklem Çantası Yöntemi	0,965
DVM	0,48
Doğrusal DVM	0,844
SOFTMAX	0,759
ELM	0,745
MLP	0,60
1-Layer LSTM	0,902
2-Layer LSTM	0,955
İkiz-LSTM DML	**0,9706**

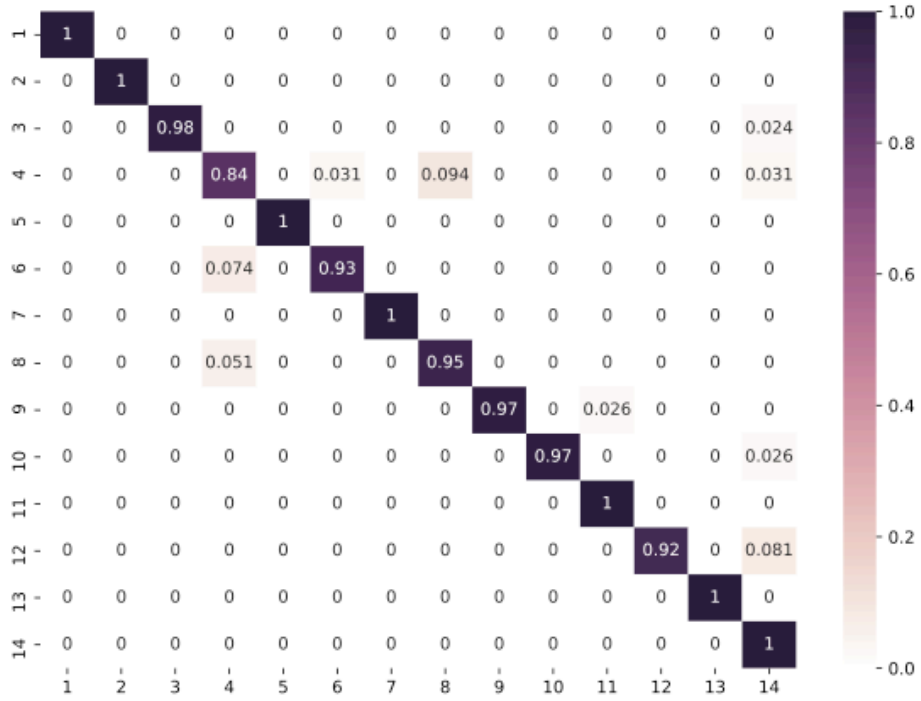
İkiz LSTM DML ağında bulunan bir alt ağ olan İkiz LSTM ağının başarım performansı ile İkiz LSTM DML ağının performansı

Şekil 7.4'de gösterilmiştir. İkiz LSTM ağlarındaki küçük bir başarım değişimi sınıflandırıcı kısmını büyük oranda etkilemektedir. Bunun sebebi İkiz Ağlarda öğrenilen derin metriklerle sınıflandırmanın gerçekleştirilmesidir.



Şekil 7.4. GTU Action 3D veri kümesi üzerinde İkiz LSTM Ağı ile İkiz LSTM DML sınıflandırıcı ağının başarımlarının karşılaştırılması.

Şekil 7.5'de İkiz-LSTM DML ağı ile elde edilen %97 başarıyla tanınan hareket kümesinin karışıklık matrisi gösterilmiştir. Matristen de görülebildiği gibi yürüme hareketi, sağa sola adım hareketi ile karışmıştır. Şekilde verilen numaralara göre hareketler sırasıyla 1.Kolları Açıp Kapama, 2.Sağ El Sallama, 3.Sağa Sola Bel Esnetme, 4.Yürüme, 5.Sağ Ayak Esnetme, 6.Sol Ayak Esnetme, 7.Sol El Sallama, 8.Öne Sağ Sol İlerleme, 9.Çömelme, 10.Sandalyede Oturup Kalkma, 11.Oturup Alkışlama, 12.Bel Çevirme, 13.Sağa Sola 8 Adımlık Hareket, 14.Boyun Gevşetme'dir.



Şekil 7.5.GTU Action 3D veri kümesinin İkiz-LSTM DML ağı karışıklık matrisi.

7.5.2. Florence Action 3D Sonuçları

Florence Action 3D, 9 farklı eylem sınıfından oluşan 217 harekete sahip küçük bir veri kümesidir. Bu bölümde, tez boyunca yapılan çalışmaların Florence Action 3D üzerindeki başarımları Tablo 7.2'de gösterilecektir.

Tablo 7.2'nin ilk 5 satırında bulunan başarımlar Florence veri kümesinde elde edilmiş gösterim tabanlı çalışmaların sonuçlarıdır. Bu çalışmalar veri kümesine has öznitelikleri bularak hareketleri sınıflandırmaktadır. Bölüm 4'de anlatılan Geometrik Eklem Çantası Yönteminin başarımları bu çalışmaların sonuçlarına yakın gelmiştir.

Tablonun 7. ve 13. Satırlarında standart makine öğrenmesi ve derin öğrenme teknikleri denenmiştir. Veri kümesinin 2 katmanlı LSTM sonuçları 1 katmanlı LSTM sonuçlarından iyi gelmemiştir. Bu veri kümesi için LSTM katman sayısını artırmak sonuçlarda iyileştirme yapmamaktadır.

Son satırda İkiz-LSTM DML ağının hareket tanıma başarım değeri verilmiştir. Sonuç Denediğimiz diğer yöntemlerden daha iyidir. Literatürde elde edilmiş en yüksek başarımı geçememiş olsa da sonuçlar umut vaat edicidir. Bu veri kümesi küçük boyutta olduğu için üzerinde derin öğrenme tabanlı bir çalışma

gerçekleştirilmemiştir. Bu açıdan çalışma, küçük veriler içinde çalışabilen derin ağ tabanlı bir yöntem sunmaktadır.

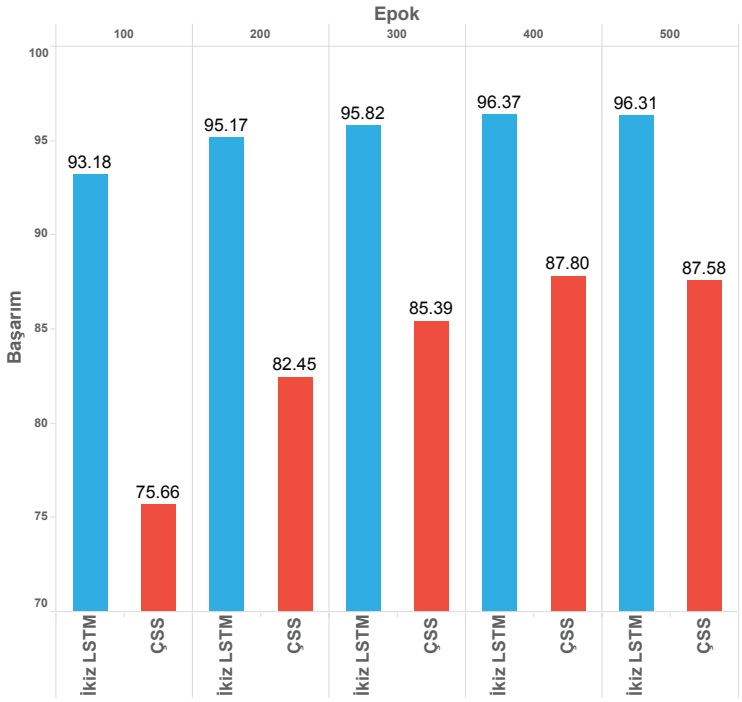
Tablo 7.2. Florence Action 3D veri kümesi üzerinde elde edilen başarımlar ve literatür çalışmalarının başarımları

Yöntem	**Başarım**
Multi-part Bag-of-Poses [15]	0,82
Riemannian Manifold [85]	0,8704
Latent Variables [86]	0,8967
Lie Group [60]	0,9088
Feature Combinations [77]	0,9439
Geometrik Eklem Çantası Yöntemi	0,88
DVM	0,233
Doğrusal DVM	0,731
ELM	0,655
MLP	0,54
Softmax	0,616
1-Layer LSTM	0,77
2-Layer LSTM	0,723
İkiz-LSTM DML	**0,8951**

Florence Action 3D veri kümesi için İkiz-LSTM DML ağının yapısı: {LSTM (45,128,2) - CONCAT (256,1) - FC (256,128)- ReLU- FC (128, 64)- ReLU-FC (64,2)} şeklindedir.

Tek katmanlı LSTM ağlarına 15 adet 3B ekleme sahip iskelet verileri anlık olarak sırayla verilmiştir. İkiz LSTM'lerin çıktıları CONCAT fonksiyonuyla bitiştirilerek elde edilen vektörler 2 katmanlı sinir ağı beslenmiştir. YSA'dan çıkan benzerlik vektörleri KNN sınıflandırıcına verilip hareket sınıfları bulunmuştur. Test anında bir test hareketi ile eğitim hareketleri eşleşmiş ve sonuçları üzerinden sınıflandırma yapılmıştır. Tablo 7.2'de Veri kümesinin çapraz denek testi sonuçları için ikiz LSTM ağı 200 ve 400 epokta eğitilmiştir. Eğitim hata değerine göre bazı denekler için model 200 epok eğitilmiş, bazıları için model 400 epok eğitilmiştir.

Şekil 7.6'da İkiz LSTM ağı ile İkiz LSTM DML Sınıflandırıcı ağının başarımları belirlenen epoklarda kıyaslanmıştır. Daha öncede belirtildiği gibi metrik öğrenme başarımı sınıflandırma başarısını büyük oranda etkilemektedir.



Şekil 7.6. Florence Action 3D veri kümesi üzerinde İkiz LSTM Ağı ile İkiz LSTM DML sınıflandırıcı ağının başarımlarının karşılaştırılması

Tezin son çalışması olan İKİZ LSTM DML ağlarından bu bölümde bahsedilmiş, sonuçlar paylaşılmıştır. Bu yöntemin, literatürde var olan çalışmalardan daha avantajlı yanları, metrik öğrenme tabanlı bir hareket tanıma çözümü sunması nedeniyle vardır. Bunlar, az boyutta veri üzerinde çalışılabilme esnekliği, birden fazla veri kümesinin bir arada kullanılabilmesi ve bilinmeyen hareketlerin de tanınabilmesi olarak sıralanabilir. Florence Action 3D veri kümesinde az sayıda veri olmasına rağmen önemli bir başarı bu veri kümesi ile elde edilebilmiştir. Bunun sebebi hareketleri ağa ikililer halinde verebilmemizdir. İkiz LSTM ağının yapısı gereği hareket veri kümelerinde aynı veya farklı hareketler olmalarına rağmen bu veri kümeleri beraber kullanılabilmektedir. Ve son olarak metrik öğrenme sayesinde eğitim anında öğrenilmemiş hareketler test anında diğer hareketlerle olan ilişkisine göre bu ağlar sayesinde tanınabilirdir.

8. SONUÇLAR

İnsan hareketlerinin bilgisayarlar tarafından tanınması bilgisayarla görme alanının önemli konularından biridir. 2B ve 3B sensörler aracılığıyla toplanan hareket verilerinin miktarı gün geçtikçe artmaktadır. Bu verileri otomatik anlamlandırma ihtiyacı da bu artışa paralel olarak ilerlemektedir. Hareketlerin bilgisayarlar tarafından tanımlanabilmesi bir çok alanda hayatımızı kolaylaştıracaktır.

Literatürde bu amaçlar doğrultusunda yapılmış çok sayıda hareket tanıma çalışması mevcuttur [1]. Film kayıtlarını kullanarak, filmler üzerinde otomatik özet çıkarmadan [87], fizik tedavi hasta hareket verilerinden duruş bozukluğu tespitine [3] kadar bir çok başarılı yöntem sunulmuştur.

3B sensörlerin yaygınlaşmasıyla beraber hareketleri daha iyi tanımlayan veriler üretilmeye başlanmıştır. Son beş yılda 3B sensörlerle çok sayıda veri kümesi [15], [44], [45], [47], [62], [68]–[72], [88] toplanmıştır. Bu veri kümelerinin içerisinde bulunan hareketler genellikle yeme içme gibi gündelik ve koşma, zıplama gibi spor hareketlerinden oluşmaktadır. Ayrıca veri kümelerinin içinde, cep karıştırma, birine vurma, tekme atma gibi toplum güvenliği için tespiti önemli hareket tipleri de bulunmaktadır.

Bu tez kapsamında 3B veri kümeleri kullanılarak hareket tanıma problemi için yeni çözümler sunulmuştur. Bunlardan ilki hareket verilerinin başarılı gösterimlerini oluşturmak amacıyla gerçekleştirilen, geometrik eklem çantası yöntemidir. 3B eklem koordinatlarının geometrisinden çıkartılan öznitelikler, kelime çantası yöntemindeki kelimeler olarak kullanılmıştır. Hareketler birer cümle haline getirilmiş, var olan tüm hareketlerdeki kelimeler denetimsiz olarak gruplanmıştır. Sonrasında hareketlerin sahip oldukları kelimelerin grup bilgileri ile hareketler sınıflandırılmıştır. Kelime çantası yöntemi literatürde hareket tanıma problemi için sık kullanılan [15], [44] bir tekniktir. Yöntemin özgün tarafı kelimelerin oluşturulma şeklidir. Kelimeler, eklemlerin geometrik uzamsal özniteliklerinin bir an içinde değil belirli zaman aralıklarında alınmasıyla oluşturulmuştur. Böylece oluşan kelimeler hem uzamsal hem zamansal bilgileri içeren özniteliklerdir. Bölüm 4'te yer alan gösterim tabanlı geometrik eklem çantası yönteminin başarım sonuçları literatürdeki benzer gösterim tabanlı çalışmalara yetişmiştir.

Geometrik kelime çantası yönteminin devamında gerçekleştirilen çalışmalar derin ağ tabanlıdır. Bilgisayarlardaki artan hesaplama kapasitesi ve yüksek işlem hacmi ile derin ağların kullanımı yaygınlaşmıştır. Özellikle bilgisayarla görme konularının başında gelen obje tanıma [89], el yazısından sayı sınıflandırma [90], gerçek zamanlı poz hesaplama [91] ve görüntü etiketleme [92] problemlerinde büyük gelişmeler derin ağlar sayesinde meydana gelmiştir. Görüntülerde, yazılarda ve zamana bağlı sıralı dizilerin analizinde elde edilen yüksek başarımlar, hareket tanıma problemi için tez kapsamında sunulan yöntemlerin tekniklerini geleneksel makine öğrenmesi tekniklerinden derin öğrenme tekniklerine sürüklemiştir. Bu ağların, kendi yapılarına ve girdi verilerine göre özel öznitelikler [93] çıkarabildiği bilinmektedir. Bu doğrultuda hareket verilerinden öznitelik çıkarmak için ilk olarak bir oto kodlayıcı model geliştirilmiştir. Oto kodlayıcılar girdiye benzer veriyi üretmek için eğitilen, genellikle girdiden daha düşük boyutta öznitelik çıkarmak için kullanılan ağlardır. Bu çalışmada sıralı olarak belirli zaman aralıklarındaki iskelet eklemlerinden öznitelik çıkarması için kullanılmıştır. Yöntem, oto kodlayıcıların ara katmanındaki öznitelik vektörlerinin LSTM ağları ile sınıflandırması üzerine idi. Ancak oto kodlayıcılar ile iskelet eklem koordinatlarının öznitelikleri başarılı bir şekilde çıkarılamamıştır. Bu durumun en büyük sebebi zamana bağlı sıralı verilerin yapısını, oto kodlayıcıların öğrenememesidir. Yöntemin detayları ve sonuçları Bölüm 5'te sunulmuştur.

Oto kodlayıcılarla elde edilen iskelet gösterimleri hareketleri yeterince iyi tanımlamamıştır. Bu nedenle, öncelikle derin ağ modellerinin yapısal özellikleri ve çıkartabildikleri özniteliklerin başarımları incelenmiştir. Bölüm 6'da derin ağ yapıları hareket verileri üzerinde denenmiştir. Denenen ağlar sırasıyla DVM, SoftMax, Aşırı Öğrenme Makinası, Çok Katmanlı Yapay Sinir Ağı ve LSTM'dir Sınıflandırma başarımlarına göre ağları kıyaslanarak hareket verilerine en uygun ağın LSTM ağları olduğu gösterilmiştir. Denenen ağların hareket girdi verileri tüm ağlar için aynıdır. Yalnızca LSTM ağlarında hareketler arasındaki süre farkı için ekstra bir işlem gerekmemiştir. Diğer ağlarda kısa süren harekeler için sıfırla doldurma yapılmıştır. Bölüm 6'da detaylarıyla yer alan çalışma ile LSTM ağlarının zamana bağlı sıralı hareket verileri üzerinde başarılı bir şekilde çalıştığı ortaya çıkmıştır. Böylece bundan sonraki çalışmalar LSTM'lerin kullanılmasına karar verilmiştir.

Tezin ana derin metrik öğrenme tabanlı çalışması, Bölüm 7'de anlatılmıştır. Metrik öğrenme, ikili girdiler arasında uygun benzerlik metriklerini bulmadır [94].

Metrikler bulunurken benzer olmayan girdi ikililerinde istenen uzaklık da korunmaya çalışılır. Derin metrik öğrenme ise iki alt ağdan oluşan bir derin ağın hata fonksiyonunun girdiler arasındaki metriğe göre optimize edilerek eğitilmesidir. İki girdi paralel olarak iki ayrı ağa verilir. Ağlar ileri besleme sırasında birbirleriyle parametrelerini paylaşır. Ağlardan elde edilen metrikler benzerliklerine göre karşılaştırılarak hata hesaplanır ve hatayı minimize etmek için geri yayma yapılır. Böylelikle ikili ağın öğrenmeye çalıştığı şey girdiler arasındaki benzerlik metrikleridir. Öğrenilen metriklerle kişi yeniden kimlik tespiti [95], yüz doğrulama [96] ve görüntü sınıflandırma [97] konularında çalışmalar yapılmıştır. Metrikleri iyi öğrenilen veriler üzerinde sınıflandırma, doğrulama veya gruplama gibi pek çok işlem gerçekleştirilebilir.

Tezin bu kısmında metrik öğrenme ile hareketler arasındaki derin benzerlik metriklerini öğrenen bir yapı tasarlanmıştır. 3B iskelet eklem verileriyle bu ağlar literatürde ilk defa kullanılmaktadır. Çoğunlukla tekil görüntüler üzerinde yapılan çalışmalarda, ağ olarak uzamsal öznitelikleri başarılı bir şekilde çıkaran CNN'ler tercih edilmiştir. Ancak hareket verileri tekil görüntü değil, sıralı 3B iskelet eklem dizileridir. Bu nedenle oluşacak derin metrik ağın içerisinde hareket verilerine uygun olan LSTM'ler kullanılmıştır. LSTM'lerin yapısal özelliklerinin avantajları ve sıralı verilere uygunluğu Bölüm 6'da anlatılmıştır.

Hareketler arasındaki benzerliği bulan bu ağa basitçe İkiz LSTM ağı ismi verilmiştir. İkiz LSTM ağının iki hareket girdisi aynı hareket sınıfına ait ise benzer etiketini, ayrı hareket sınıfına aitse farklı etiketini almaktadır. İkiz LSTM ağının benzerlik bulma test başarımları yüksektir. Bu ağ yapısı itibariyle hareket sınıflarına ihtiyaç duymamaktadır. Hareket girdilerinin sınıflarının aynı olup olmadığı bilgisiyle eğitilmektedir. Bu sayede veri kümelerinden bağımsız iki hareket değerlendirilebilmektedir. Farklı veri kümelerinde ki aynı hareketler kullanış amacına göre ayrılabilir veya birleşebilir. Var olan veri kümeleri birleştirilerek büyük boyutlarda hareket verileri analiz edilebilir. Çift halinde girdi ihtiyacı nedeniyle küçük veri kümeleri de ikili kombinasyonlarda artırılıp derin ağ içinde kullanılabilecektir.

Literatürde hareket tanıma, hareketlerin sınıflarını tahmin etmeye dayanan bir problemdir. Bu problemin çözümü için ikiz LSTM modülünün var olan avantajlarını kullanan ayrı bir sınıflandırıcı modül tasarlanmıştır. İkiz LSTM ağlarının benzerlik tespitine göre ağ, hareketlerin sınıflarını tespit etmektedir. Sınıflandırıcı modülünün

başarımı ikiz LSTM ağlarının benzerlik bulma başarımına fazlasıyla bağlıdır. Bu modülün hareket tanıma sonuçları Bölüm 7'de bulunmaktadır.

Son olarak hareketleri tanımak için ayrı çalışan ikiz LSTM ve sınıflandırıcı modülü yerine ikiz LSTM'leri kullanarak uçtan uca eğitilen bir sınıflandırıcı ağı oluşturulmuştur. Uçtan Uca İkiz LSTM DML adındaki ağ hareket ikililerini liste olarak alıp, birden fazla ikili arasındaki benzerlik metriklerini kullanarak hareket sınıfı tahmini yapmaktadır. Ağın sınıflandırma başarımı iyileştirilmeye ihtiyaç duymaktadır. Ağ için gerekli olan ikili hareket listelerinin oluşturulma şekli ağın eğitimini fazlasıyla etkilemektedir.

Bundan sonra yapılacak çalışmalar uçtan uca ikiz LSTM DML ağının iyileştirilmesi ve farklı veri kümelerinin birleştirilip denenmesi üzerine olacaktır. Ayrıca eğitim anında belirtilmeyen hareketlerin test anında kullanılması çalışmaya farklı bir bakış getirecektir.

KAYNAKLAR

[1] Poppe R., (2010), "A survey on vision-based human action recognition", *Image and Vision Computing*, 28, 6, 976–990.

[2] Moeslund T. B., Hilton A., ve Kr??ger V., (2006), "A survey of advances in vision-based human motion capture and analysis", *Computer Vision and Image Understanding*, 104, 2–3 SPEC. ISS. 90–126.

[3] Ar I. ve Akgul Y. S., (2014), "A computerized recognition system for the home-based physiotherapy exercises using an RGBD camera", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22, 6, 1160–1171.

[4] Piyathilaka L. ve Kodagoda S., (2015), "Human activity recognition for domestic robots", *Springer Tracts in Advanced Robotics*, 395–408.

[5] Forlizzi J. ve DiSalvo C., (2006), "Service robots in the domestic environment", *Proceeding of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction - HRI '06*, 258.

[6] Zhu C. ve Sheng W., (2011), "Wearable sensor-based hand gesture and daily activity recognition for robot-assisted living", *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 41, 3, 569–573.

[7] Ramanathan M., Yau W.-Y., ve Teoh E. K., (2014), "Human Action Recognition With Video Data: Research and Evaluation Challenges", *IEEE Transactions on Human-Machine Systems*, 44, 5, 650–663.

[8] Laptev I., Marszałek M., Schmid C., ve Rozenfeld B., (2008), "Learning realistic human actions from movies", *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*.

[9] Wagner D. D., Dal Cin S., Sargent J. D., Kelley W. M., ve Heatherton T. F., (2011), "Spontaneous Action Representation in Smokers when Watching Movie Characters Smoke", *Journal of Neuroscience*, 31, 3, 894–898.

[10] Drosou A., Ioannidis D., Tzovaras D., Moustakas K., ve Petrou M., (2015), "Activity related authentication using prehension biometrics", *Pattern Recognition*, 48, 5, 1743–1759.

[11] Drosou A., Moustakas K., Ioannidis D., ve Tzovaras D., (2011), "Activity related biometric authentication using Spherical Harmonics", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*.

[12] Hadfield S., Lebeda K., ve Bowden R., (2017), "Hollywood 3D: What are the Best 3D Features for Action Recognition?", *International Journal of Computer Vision*, 121, 1, 95–110.

[13] Shotton J. *vd.*, (2011), "Real-time human pose recognition in parts from single depth images", *CVPR 2011*, 1297–1304.

[14] Zhang S., Liu X., ve Xiao J., (2017), "On geometric features for skeleton-based action recognition using multilayer LSTM networks", *Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017*, 148–157.

[15] Seidenari L., Varano V., Berretti S., Del Bimbo A., ve Pala P., (2013), "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 479–485.

[16] Yucer S. ve Akgul Y. S., "3D Human Action Recognition with Siamese-LSTM Based Deep Metric Learning", Journal of Images and Graphics.

[17] Herath S., Harandi M., ve Porikli F., (2017), "Going deeper into action recognition: A survey", *Image and Vision Computing*, 60, 4–21.

[18] Chen C., Jafari R., ve Kehtarnavaz N., (2015), "A survey of depth and inertial sensor fusion for human action recognition", *Multimedia Tools and Applications*, 1–21.

[19] Perez-Sala X., Escalera S., Angulo C., ve Gonz??lez J., (2014), "A survey on model based approaches for 2D and 3D visual human pose recovery", *Sensors (Basel, Switzerland)*, 14, 3, 4189–4210.

[20] Chen C., Jafari R., ve Kehtarnavaz N., (2016), "A Real-Time Human Action Recognition System Using Depth and Inertial Sensor Fusion", *IEEE Sensors Journal*, 16, 3, 773–781.

[21] Minnen D., Starner T., Ward J. A., Lukowicz P., ve Tröster G., (2005), "Recognizing and discovering human actions from on-body sensor data", *IEEE International Conference on Multimedia and Expo, ICME 2005*, 1545–1548.

[22] Duta I. C., Jasper J. R., Ionescu B., Aizawa K., G. Hauptmann A., ve Sebe N., (2017), "Efficient human action recognition using histograms of motion gradients and VLAD with descriptor shape information", *Multimedia Tools and Applications*, 76, 21, 22445–22472.

[23] Chakraborty B., Bagdanov A. D., ve Gonzalez J., (2009), "Towards Real-Time Human Action Recognition", *Pattern Recognition and Image Analysis, Proceedings*, 425–432.

[24] Wu Z. M. ve Ng W. W. Y., (2014), "Human action recognition using action bank and RBFNN trained by L-GEM", *International Conference on Wavelet Analysis and Pattern Recognition*, 30–35.

[25] Marr D. ve Vaina L., (1982), "Representation and recognition of the

movements of shapes.", *Proceedings of the Royal Society of London. Series B, Containing papers of a Biological character. Royal Society (Great Britain)*, 214, 1197, 501–524.

[26] Yilmaz A. ve Shah M., (2005), "Actions sketch: A novel action representation", *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, 984–989.

[27] Gorelick L., Blank M., Shechtman E., Irani M., ve Basri R., (2007), "Actions as space-time shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29, 12, 2247–2253.

[28] Simonyan K. ve Zisserman A., (2014), "Very Deep Convolutional Networks for Large-Scale Image Recognition", *ImageNet Challenge*, 1–10.

[29] Wang H. ve Schmid C., (2013), "Action recognition with improved trajectories", *Proceedings of the IEEE International Conference on Computer Vision*, 3551–3558.

[30] Matikainen P., Hebert M., ve Sukthankar R., (2009), "Trajectons: Action recognition through the motion analysis of tracked features", *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops 2009*, 514–521.

[31] Dollár P., Rabaud V., Cottrell G., ve Belongie S., (2005), "Behavior recognition via sparse spatio-temporal features", *Proceedings - 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, VS-PETS*, 65–72.

[32] Harris C. ve Stephens M., (1988), "A Combined Corner and Edge Detector", *Procedings of the Alvey Vision Conference 1988*, 23.1-23.6.

[33] Laptev I., (2005), "On space-time interest points", *International Journal of Computer Vision*, 107–123.

[34] Liu J., Luo J., ve Shah M., (2009), "Recognizing realistic actions from videos in the Wild", *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, 1996–2003.

[35] Yang S., Yuan C., Wu B., Hu W., ve Wang F., (2015), "Multi-feature max-margin hierarchical Bayesian model for action recognition", *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1610–1618.

[36] Karpathy A., (2014), "What I learned from competing against a ConvNet on ImageNet", *Andrej Karpathy Blog*, 5, 1–15.

[37] Donahue J. *vd.*, (2017), "Long-Term Recurrent Convolutional Networks for Visual Recognition and Description", *IEEE Transactions on Pattern Analysis*

and Machine Intelligence, 39, 4, 677–691.

[38] Baccouche M., Mamalet F., ve Wolf C., (2011), "Sequential deep learning for human action recognition", *Proc. Int. Conf. Human Behavior Understanding (HBU)*, 29–39.

[39] Ji S., Yang M., Yu K., ve Xu W., (2013), "3D convolutional neural networks for human action recognition", *IEEE transactions on pattern analysis and machine intelligence*, 35, 1, 221–31.

[40] Girshick R. *vd.*, (2016), "Two-Stream Convolutional Networks for Action Recognition in Videos", *arXiv preprint arXiv:1406.2199*, 9905, i, 1–11.

[41] Ng J. Y. H., Hausknecht M., Vijayanarasimhan S., Vinyals O., Monga R., ve Toderici G., (2015), "Beyond short snippets: Deep networks for video classification", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 4694–4702.

[42] Srivastava N., Mansimov E., ve Salakhutdinov R., (2015), "Unsupervised Learning of Video Representations using LSTMs", *Bmvc2015*, 2009.

[43] Ryoo M. S., Rothrock B., ve Fleming C., (2017), "Privacy-Preserving Egocentric Activity Recognition from Extreme Low Resolution", *Proceedings of the AAAI Conference on Artificial Intelligence*, 1–8.

[44] Li W., Zhang Z., ve Liu Z., (2010), "Action recognition based on a bag of 3D points", *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010*, 9–14.

[45] Wang J., Liu Z., Wu Y., ve Yuan J., (2012), "Mining actionlet ensemble for action recognition with depth cameras", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1290–1297.

[46] Rahmani H., Mahmood A., Huynh D. Q., ve Mian A., (2014), "HOPC: Histogram of Oriented Principal Components of 3D pointclouds for action recognition", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 742–757.

[47] Oreifej O. ve Liu Z., (2013), "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 716–723.

[48] Yang X., Zhang C., ve Tian Y., (2012), "Recognizing actions using depth motion maps-based histograms of oriented gradients", *Proceedings of the 20th ACM international conference on Multimedia - MM '12*, 1057.

[49] Ji X., Cheng J., Tao D., Wu X., ve Feng W., (2017), "The spatial Laplacian and temporal energy pyramid representation for human action recognition using depth sequences", *Knowledge-Based Systems*, 122, 64–74.

[50] He L., Wang G., Liao Q., ve Xue J.-H., (2015), "Depth-images-based pose estimation using regression forests and graphical models", *Neurocomputing*, 164, 210–219.

[51] Chen C., Liu K., ve Kehtarnavaz N., (2013), "Real-time human action recognition based on depth motion maps", *Journal of Real-Time Image Processing*, 1–9.

[52] Liu L. ve Shao L., (2013), "Learning discriminative representations from RGB-D video data", *IJCAI International Joint Conference on Artificial Intelligence*, 1493–1500.

[53] Wang P., Li W., Gao Z., Zhang Y., Tang C., ve Ogunbona P., (2017), "Scene Flow to Action Map: A New Representation for RGB-D based Action Recognition with Convolutional Neural Networks".

[54] Ijjina E. P. ve Chalavadi K. M., (2017), "Human action recognition in RGB-D videos using motion sequence information and deep learning", *Pattern Recognition*, 72, 504–516.

[55] Shahroudy A., Ng T. T., Yang Q., ve Wang G., (2016), "Multimodal Multipart Learning for Action Recognition in Depth Videos", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38, 10, 2123–2129.

[56] Lin Y. Y., Hua J. H., Tang N. C., Chen M. H., ve Liao H. Y. M., (2014), "Depth and skeleton associated action recognition without online accessible RGB-D cameras", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2617–2624.

[57] Wang X. ve Gupta A., (2015), "Unsupervised Learning of Visual Representations Using Videos", *Iccv*, 2794–2802.

[58] Wang P., Yuan C., Hu W., Li B., ve Zhang Y., (2016), "Graph based skeleton motion representation and similarity measurement for action recognition", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 370–385.

[59] Liu J., Shahroudy A., Xu D., Kot Chichung A., ve Wang G., (2017), "Skeleton-Based Action Recognition Using Spatio-Temporal LSTM Network with Trust Gates", *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[60] Vemulapalli R., Arrate F., ve Chellappa R., (2014), "Human action recognition by representing 3D skeletons as points in a lie group", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 588–595.

[61] Liu J., Akhtar N., ve Mian A., (2017), "Skepxels: Spatio-temporal Image Representation of Human Skeleton Joints for Action Recognition", November.

[62] Lai K., Bo L., Ren X., ve Fox D., (2011), "A large-scale hierarchical multi-view RGB-D object dataset", *Proceedings - IEEE International Conference on Robotics and Automation*, 1817–1824.

[63] Du Y., Wang W., ve Wang L., (2015), "Hierarchical recurrent neural network for skeleton based action recognition", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1110–1118.

[64] Kim T. S. ve Reiter A., (2017), "Interpretable 3D Human Action Analysis with Temporal Convolutional Networks", *arXiv preprint arXiv*.

[65] Liu J. *vd.*, (2017), "Global Context-Aware Attention LSTM Networks for 3D Action Recognition", *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, PP, 99, 1.

[66] Liu H., Tu J., ve Liu M., (2015), "Two-Stream 3D Convolutional Neural Network for Human Skeleton-Based Action Recognition", *Journal of Latex Class Files*, 14, 8, 1–5.

[67] Baradel F., Wolf C., ve Mille J., (2017), "Pose-conditioned Spatio-Temporal Attention for Human Action Recognition", *arXiv*.

[68] Sung J., Ponce C., Selman B., ve Saxena A., (2011), "Human Activity Detection from RGBD Images", *IEEE International Conference on Robotics and Automation*, 842–849.

[69] Ni B., Wang G., ve Moulin P., (2011), "RGBD-HuDaAct: A color-depth video database for human daily activity recognition", *Proceedings of the IEEE International Conference on Computer Vision*, 1147–1153.

[70] Cheng Z., Qin L., Ye Y., Huang Q., ve Tian Q., (2012), "Human daily action analysis with multi-view and color-depth data", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 52–61.

[71] Koppula H. S., (2013), "Learning Spatio-Temporal Structure from RGB-D Videos for Human Activity Detection and Anticipation", *International Conference on Machine Learning (ICML)*, 28, 792–800.

[72] Wei P., Zhao Y., Zheng N., ve Zhu S. C., (2013), "Modeling 4D human-object interactions for event and object recognition", *Proceedings of the IEEE International Conference on Computer Vision*, 3272–3279.

[73] Wang J., Nie X., Xia Y., Wu Y., ve Zhu S. C., (2014), "Cross-view action modeling, learning, and recognition", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2649–2656.

[74] Wang K., Wang X., Lin L., Wang M., ve Zuo W., (2014), "3D Human Activity Recognition with Reconfigurable Convolutional Neural Networks", *Acm Mm*, 97–106.

[75] Chen C., Jafari R., ve Kehtarnavaz N., (2015), "UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor", *Proceedings - International Conference on Image Processing, ICIP*, 168–172.

[76] Rahmani H., Mahmood A., Huynh D., ve Mian A., (2016), "Histogram of Oriented Principal Components for Cross-View Action Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38, 12. 2430–2443.

[77] Carbonera Luvizon D., Tabia H., ve Picard D., (2016), "Learning features combination for human action recognition from skeleton sequences", *Pattern Recognition Letters*.

[78] Du Y., Fu Y., ve Wang L., (2016), "Representation Learning of Temporal Dynamics for Skeleton-Based Action Recognition", *IEEE Transactions on Image Processing*, 25, 7, 3010–3022.

[79] Song S., Lan C., Xing J., Zeng W., ve Liu J., (2016), "An End-to-End Spatio-Temporal Attention Model for Human Action Recognition from Skeleton Data".

[80] Huang G.-B., Zhou H., Ding X., ve Zhang R., (2012), "Extreme learning machine for regression and multiclass classification", *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics*, 42, 513–29.

[81] Tang J., Deng C., ve Guang G.-B., (2015), "Extreme learning machine for multilayer perceptron", *IEEE Transactions on Neural Networks and Learning Systems*, 27, 4, 809–821.

[82] Huang G.-B. *vd.*, (2006), "Extreme learning machine: Theory and applications", *Neurocomputing*, 70, 1–3, 489–501.

[83] Chao Li, Shouqian Sun, Xin Min, Wenqian Lin, Binling Nie X. Z., (2017), "End-To-End Learning of Deep Convolitional Neural Network for 3D Human Action Recognition", July, 609–612.

[84] Shaham U. ve Lederman R. R., (2018), "Learning by coincidence: Siamese networks and common variable learning", *Pattern Recognition*, 74, 52–63.

[85] Devanne M., Wannous H., Berretti S., Pala P., Daoudi M., ve Del Bimbo A., (2015), "3-D Human Action Recognition by Shape Analysis of Motion Trajectories on Riemannian Manifold", *IEEE Transactions on Cybernetics*, 45, 7, 1340–1352.

[86] Anirudh R., Turaga P., Su J., ve Srivastava A., (2015), "Elastic functional coding of human actions: From vector-fields to latent variables", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3147–3155.

[87] Rohrbach A., Rohrbach M., ve Schiele B., (2015), "The long-short story of movie description", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 209–221.

[88] Sung J., Ponce C., Selman B., ve Saxena A., (2012), "Unstructured human activity detection from RGBD images", *Proceedings - IEEE International Conference on Robotics and Automation*, 842–849.

[89] He K., Zhang X., Ren S., ve Sun J., (2016), "Deep Residual Learning for Image Recognition", *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

[90] Wan L., Zeiler M., Zhang S., LeCun Y., ve Fergus R., (2013), "Regularization of neural networks using dropconnect", *Icml*, 1, 109–111.

[91] Cao Z., Simon T., Wei S.-E., ve Sheikh Y., (2016), "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields".

[92] Karpathy A. ve Fei-Fei L., (2017), "Deep Visual-Semantic Alignments for Generating Image Descriptions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39, 4, 664–676.

[93] Erhan D., Bengio Y., Courville A., ve Vincent P., (2009), "Visualizing higher-layer features of a deep network", *Bernoulli*, 1341, 1–13.

[94] Wang J., Zhou F., Wen S., Liu X., ve Lin Y., (2017), "Deep Metric Learning with Angular Loss".

[95] Yi D., Lei Z., Liao S., ve Li S. Z., (2014), "Deep metric learning for person re-identification", *Proceedings - International Conference on Pattern Recognition*, 34–39.

[96] Hu J., Lu J., ve Tan Y. P., (2014), "Discriminative deep metric learning for face verification in the wild", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1875–1882.

[97] Lu J., Wang G., Deng W., Moulin P., ve Zhou J., (2015), "Multi-manifold deep metric learning for image set classification", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1137–1145.

ÖZGEÇMİŞ

Şeyma Yücer 1993 yılında Tokat'ta doğdu. 2011 yılında başladığı Gebze Teknik Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünü 2015 yılında başarıyla tamamlayarak aynı yıl yüksek lisans eğitimine Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında başladı. 2015 yılından bu yana Gebze Teknik Üniversitesinde araştırma görevlisi olarak çalışmaktadır.

EKLER

Ek A: Tez Çalışması Kapsamında Yapılan Yayınlar

S. Yucer ve Y. S. Akgul, 2018, "3D Human Action Recognition with Siamese-LSTM Based Deep Metric Learning", Journal of Image and Graphics.

Ek B: Tez Terimleri Sözlüğü: İngilizce – Türkçe

A	
Accuracy	Başarım
Action	Hareket
Action Class	Hareket Sınıfı
Activation	Etkilinim
Active	Etkin
Algorithm	Algoritma
Analysis	Analiz
Analytical	Çözümsel
And	Ve
Appearance	Görünüş
Approach	Yaklaşım
Array	Dizi
Aspect Ratio	En-Boy Oranı
Auto encoder	Oto Kodlayıcı
Axis	Eksen
B	
Background	Arka Plan
Backpropagation	Geri Yayma
Backward	Geriye Doğru
Bag Of Words	Kelimeler Çantası
Base	Taban
Batch	Toptan

Batch Size	Toptan Boyutu
Bayesian Networks	Bayes Ağları
Bias	Yanlılık
Binary	İkili
Binding	Değer Atama
Biometric	Biyometri
Bottom-Up	Aşağıdan-Yukarı
	C
Camera	Kamera
Capacity	Kapasite
Cascade Art	Arda Sıralı, Kademeli
Cell	Hücre
Classification	Sınıflandırma
Classifier	Sınıflandırıcı
Clip	Klip
Clustering	Kümeleme
Combination	Kombinasyon
Computer Vision	Bilgisayarla Görme
Concatanation	Bitiştirmek
Confusion	Karışıklık, Hata
Confusion Matrix	Karışıklık Matrisi
Connected	Bağlı
Constant	Sabit
Constraint	Kısıtlama
Contour	Kontur
Contrast	Kontrast
Conventional	Geleneksel
Convolutional	Evirişim
Corner	Köşe
Cost	Bedel
Cross-Entropy	Çapraz Düzensizlik
Cuboid	Kübik
Curve	Eğri

D	
Dataset	Veri Kümesi
Decision	Karar
Depth	Derinlik
Descriptor	Tanımlayıcı
Detector	Detektör
Discrete	Ayrık
Distribution	Dağılım
E	
Eigenshape	Öz şekil
Encoder	Kodlayıcı
Entropy	Düzensizlik
Epoch	Epok
Equation	Denklem
Error	Hata
Estimator	Kestirici
Euclidian	Öklid
Euclidian Distance	Öklid Uzaklığı
Evaluation	Değerlendirme
Evidence	Kanıt, Delil
Exercise	Egzersiz
F	
Feature	Öznitelik
Feature Descriptor	Öznitelik Tanımlayıcısı
Feature Extraction	Öznitelik Çıkarma
Feed Foreward	İleri Yönlü
Fisher Discriminant Analysis	Fisher Ayırma Analizi
Fit	Oturma Uydurma
Foreground	Ön Plan
Frame	Çerçeve
Function	İşlev
Fusion	Kaynaştırma
G	

Generative	Üretken
Generilization	Genelleme
Global	Küresel, Global
Gradient	Gradyan
Graph	Çizge
	H
Hidden	Saklı
Hiearchical	Ağaç Yapılı
High Order	Üst Düzey
High-Level	Yüksek Seviye
Hinge Loss	Menteşe Yitimi
Histogram	Histogram
Histogram Of Gradients	Gradyanlar Histogramı
	I
Image	Görüntü
Image Processing	Görüntü İşleme
Index	Dizin
Interest Point	İlgi Noktası
Input	Girdi
Interpolate	Ara Değerleme
Iteration	Yineleme
	J
Joint	Eklem
Joint Probability	Birleşik Olasılık
	K
K-Means Clustering	K-Merkezli Kümeleme
Kernel	Çekirdek
Key Pose	Anahtar Poz
	L
Layer	Katman
Leave-One-Out	Birini Dışarda Bırakma
Linear	Lineer, Doğrusal
Logical Regression	Mantıksal Regresyon

Loss	Hata, Yitim
Low-Level	Düşük Seviye
	M
Machine Learning	Makine Öğrenmesi
Magnitude	Büyüklük
Matrix	Matris
Maximation	Büyütme
Mean	Ortalama
Measure	Ölçü
Metric	Metrik
Minimization	Küçültme
Model	Model
Motion	Hareket
Movie	Film
Multi Class Classification	Çok Sınıflı Sınıflandırıcı
Multi-Kernel Learning	Çoklu Çekirdek Öğrenmesi
Multiple	Çoklu
	N
Nearest Neighbor İnterpolation	En Yakın Komşuluk Ara Değerlemesi
Neural Networks	Yapay Sinir Ağları
Neuron	Nöron
Node	Düğüm
Normalization	Normalleştirme
Numeric	Sayısal
	O
Object	Nesne
Optical Flow	Optik Akış
Optimization	Eniyileme
Orientation	Oryantasyon
Output	Çıktı
Over Training	Aşırı Eğitme
Overfitting	Aşırı Öğrenme
	P

Pattern	Örüntü
Perceptron	Algılayıcı
Performence	Başarım
Physiotherapy	Fizik Tedavi, Fizyoterapi
Pixel	İmge Noktası, Piksel
Pose	Duruş
Positive	Pozitif
Prediction	Tahmin Edilen
Prediction	Tahmin Etme
Principal	Temel
Principal Component Analysis	Temel Bileşen Analizi
Principle	İlke
Private	Özel
Probability	Olasılık
Pruning	Budama
Pseudocode	Örnek Kod
R	
Random	Rastgele
Random Forest	Rastgele Orman
Real Time	Gerçek Zamanlı
Reccurent	Özyineli
Reccurent Neural Network	Özyineli Sinir Ağları
Recognition	Tanıma
Rectification	Doğrultma
Regression	Regresyon
Repetition	Tekrar
Rotation	Dönme
S	
Sample	Örneklemek,Örneklem
Sampling	Örnekleme
Scale	Ölçek
Scaling	Ölçekleme
Scene	Görüntü

Segmentation	Bölütleme
Sequence	Sıra
Sequential	Sıralı
Set	Küme
Short Term	Kısa Zamanlı
Silhouette	Silüet
Sparse	Seyrek
Spatial	Uzamsal
Spatio-Temporal	Uzam-Zamansal
Split	Bölme
Stable	Kararlı
String	Karakter Dizisi
Supervised	Denetimli
Support	Destek
Support Vector Machine	Destek Vektör Makinesi
Surveillance	Gözetim
Symbol	Sembol
T	
Testing Set	Test Kümesi
Threshold	Eşik Seviyesi
Time Series	Zaman Dizisi
Top-Down	Yukarıdan-Aşağı
Tracking	Takip Etme
Training Set	Eğitim Kümesi
Transform	Dönüşüm
Transpose	Devrik
U	
Uniform	Tek Düze
Unsupervised	Denetimsiz
V	
Validation	Geçerleme
Velocity	Hız
Video	Video

Ek C: Tez Terimleri Sözlüğü: Türkçe - İngilizce

A	
Ağaç Yapılı	Hiearchical
Algılayıcı	Perceptron
Algoritma	Algorithm
Anahtar Poz	Key Pose
Analiz	Analysis
Ara Değerleme	Interpolate
Arda Sıralı, Kademeli	Cascade Art
Arka Plan	Background
Aşağıdan-Yukarı	Bottom-Up
Aşırı Eğitme	Over Training
Aşırı Öğrenme	Overfitting
Ayrık	Discrete
B	
Bağlı	Connected
Başarım	Accuracy
Başarım	Performence
Bayes Ağları	Bayesian Networks
Bedel	Cost
Bilgisayarla Görme	Computer Vision
Birini Dışarda Bırakma	Leave-One-Out
Birleşik Olasılık	Joint Probability
Bitiştirmek	Concatanation
Biyometri	Biometric
Bölme	Split
Bölütleme	Segmentation
Budama	Pruning
Büyüklük	Magnitude
Büyütme	Maximation
C	
Çapraz Düzensizlik	Cross-Entropy

Çekirdek	Kernel
Çerçeve	Frame
Çıktı	Output
Çizge	Graph
Çok Sınıflı Sınıflandırıcı	Multi Class Classification
Çoklu	Multiple
Çoklu Çekirdek Öğrenmesi	Multi-Kernel Learning
Çözümsel	Analytical
	D
Dağılım	Distribution
Değer Atama	Binding
Değerlendirme	Evaluation
Denetimli	Supervised
Denetimsiz	Unsupervised
Denklem	Equation
Derinlik	Depth
Destek	Support
Destek Vektör Makinesi	Support Vector Machine
Detektör	Detector
Devrik	Transpose
Dizi	Array
Dizin	Index
Doğrultma	Rectification
s	Linear
Dönme	Rotation
Dönüşüm	Transform
Duruş	Pose
Düğüm	Node
Düşük Seviye	Low-Level
Düzensizlik	Entropy
	E
Egzersiz	Exercise
Eğitim Kümesi	Training Set

Eğri	Curve
Eklem	Joint
Eksen	Axis
En Yakın Komşuluk Ara Değerlemesi	Nearest Neighbor İnterpolation
En-Boy Oranı	Aspect Ratio
Eniyileme	Optimization
Epok	Epoch
Eşik Seviyesi	Threshold
Etkilinim	Activation
Etkin	Active
Evirişim	Convolutional
	F
Film	Movie
Fisher Ayırma Analizi	Fisher Discriminant Analysis
Fizik Tedavi, Fizyoterapi	Physiotherapy
	G
Geçerleme	Validation
Geleneksel	Conventional
Genelleme	Generilization
Gerçek Zamanlı	Real Time
Geri Yayma	Backpropagation
Geriye Doğru	Backward
Girdi	Input
Görüntü	Image
Görüntü	Scene
Görüntü İşleme	Image Processing
Görünüş	Appearance
Gözetim	Surveillance
Gradyan	Gradient
Gradyanlar Histogramı	Histogram Of Gradients
	H
Hareket	Action

Hareket Sınıfı	Action Class
Hata	Error
Hata, Yitim	Loss
Hız	Velocity
Histogram	Histogram
Hücre	Cell
	İ
İkili	Binary
İleri Yönlü	Feed Foreward
İlgi Noktası	Interest Point
İlke	Principle
İmge Noktası, Piksel	Pixel
İşlev	Function
	K
K-Merkezli Kümeleme	K-Means Clustering
Kamera	Camera
Kanıt, Delil	Evidence
Kapasite	Capacity
Karakter Dizisi	String
Karar	Decision
Kararlı	Stable
Karışıklık Matrisi	Confusion Matrix
Karışıklık, Hata	Confusion
Katman	Layer
Kaynaştırma	Fusion
Kelimeler Çantası	Bag Of Words
Kestirici	Estimator
Kısa Zamanlı	Short Term
Kısıtlama	Constraint
Klip	Clip
Kodlayıcı	Encoder
Kombinasyon	Combination
Kontrast	Contrast

Kontur	Contour
Köşe	Corner
Kübik	Cuboid
Küçültme	Minimization
Küme	Set
Kümeleme	Clustering
Küresel, Global	Global
	L
Lineer	Linear
	M
Makine Öğrenmesi	Machine Learning
Mantıksal Regresyon	Logical Regression
Matris	Matrix
Menteşe Yitimi	Hinge Loss
Metrik	Metric
Model	Model
	N
Nesne	Object
Normalleştirme	Normalization
Nöron	Neuron
	O
Olasılık	Probability
Optik Akış	Optical Flow
Ortalama	Mean
Oryantasyon	Orientation
Oto Kodlayıcı	Auto encoder
Oturma Uydurma	Fit
	Ö
Öklid	Euclidian
Öklid Uzaklığı	Euclidian Distance
Ölçek	Scale
Ölçekleme	Scaling
Ölçü	Measure

Ön Plan	Foreground
Örnek Kod	Pseudocode
Örnekleme	Sampling
Örneklemek, Örneklem	Sample
Örüntü	Pattern
Öz şekil	Eigenshape
Özel	Private
Öznitelik	Feature
Öznitelik Çıkarma	Feature Extraction
Öznitelik Tanımlayıcısı	Feature Descriptor
Özyineli	Reccurent
Özyineli Sinir Ağları	Reccurent Neural Network
	P
Pozitif	Positive
	R
Rastgele	Random
Rastgele Orman	Random Forest
Regresyon	Regression
	S
Sabit	Constant
Saklı	Hidden
Sayısal	Numeric
Sembol	Symbol
Seyrek	Sparse
Sınıflandırıcı	Classifier
Sınıflandırma	Classification
Sıra	Sequence
Sıralı	Sequential
Silüet	Silhouette
	T
Taban	Base
Tahmin Edilen	Prediction
Tahmin Etme	Prediction

Takip Etme	Tracking
Tanıma	Recognition
Tanımlayıcı	Descriptor
Tek Düze	Uniform
Tekrar	Repetition
Temel	Principal
Temel Bileşen Analizi	Principal Component Analysis
Test Kümesi	Testing Set
Toptan	Batch
Toptan Boyutu	Batch Size
	U
Uzam-Zamansal	Spatio-Temporal
Uzamsal	Spatial
Üretken	Generative
Üst Düzey	High Order
	V
Ve	And
Veri Kümesi	Dataset
Video	Video
Yaklaşım	Approach
Yanlılık	Bias
Yapay Sinir Ağları	Neural Networks
Yineleme	Iteration
Yukarıdan-Aşağı	Top-Down
Yüksek Seviye	High-Level
	Z
Zaman Dizisi	Time Series