

**TEK ADIMDA ÖĞRENME YAKLAŞIMI İLE EL
HAREKETLERİNİ SINIFLANDIRMA**

**2019
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

Esmâ ŞEKER

**TEK ADIMDA ÖĞRENME YAKLAŞIMI İLE EL HAREKETLERİNİ
SINIFLANDIRMA**

Esmâ ŞEKER

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalında

Yüksek Lisans Tezi

Olarak Hazırlanmıştır

KARABÜK

Haziran 2019

Esma ŐEKER tarafından hazırlanan “TEK ADIMDA ÖĐRENME YAKLAŐIMI İLE EL HAREKETLERİNİ SINIFLANDIRMA” baŐlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduĐunu onaylarım.

Doç. Dr. OĐuz FINDIK
Tez DanıŐmanı, Bilgisayar MühendisliĐi Anabilim Dalı



Bu çalıŐma, jürimiz tarafından oy birliĐi ile Bilgisayar MühendisliĐi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiŐtir. 12/06/2019

Ünvanı, Adı SOYADI (Kurumu)

İmzası

BaŐkan : Dr. Öğr. Üyesi Caner ÖZCAN (KBÜ)



Üye : Doç. Dr. OĐuz FINDIK (KBÜ)



Üye : Dr. Öğr. Üyesi Okan ERKAYMAZ (BEÜ)



30.06/2019

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıŐtır.

Prof. Dr. Filiz ERSÖZ
Fen Bilimleri Enstitü Müdürü V.





“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Esmâ ŞEKER

ÖZET

Yüksek Lisans Tezi

TEK ADIMDA ÖĞRENME YAKLAŞIMI İLE EL HAREKETLERİNİ SINIFLANDIRMA

Esmâ ŞEKER

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:

Doç. Dr. Oğuz FINDIK

Haziran 2019, 49 sayfa

Günümüzde insan bilgisayar etkileşimi kapsamında herhangi bir araç kullanmadan insan hareketlerini tanıma işlemi büyüyen araştırma alanlarından biridir. Bu nedenle birçok araştırmacı insan hareketlerini tanımak için çeşitli yaklaşımlar önermiştir. Bu yaklaşımların bir kısmı önceden hareket hakkında bilgi gerektirmektedir. Diğer kısmı ise büyük veri kümelerine ihtiyaç duymaktadır. Tek adımda öğrenme yaklaşımı önceden hareket hakkında bilgi veya büyük miktarda veri gerektirmez. El hareketleri insanların yaşamlarında önemli bir rol oynamaktadır. Böylece, el hareketi tanıma sistemi, insanların bilgisayarla etkileşime girmesi için yenilikçi ve doğal bir yol sunmaktadır. Bu çalışma tek adımda öğrenme yaklaşımını kullanarak el hareketlerini sınıflandırmayı amaçlamaktadır. Tek adımda öğrenme yaklaşımı bir videoda bulunan hareketi tek bir görüntü ile temsil etmektedir. Çalışmada yaklaşımı uygulamak için standart sapma (SS), hareket geçmiş görüntüsü (HGG) ve 2 boyutlu hızlı Fourier dönüşümü (2B HFD) yöntemleri kullanılmıştır. SS yöntemi ile içinde hareket

bulunduran bir video, yalnızca bir görüntüye dönüştürülmüştür. Ancak bu yöntem hareketin geçmiş bilgisini koruyamamaktadır. Bu nedenle çalışmada SS yöntemi ile birlikte HGG yöntemi de kullanılmıştır. Yöntem hareketin geçmiş bilgisini korumakla birlikte hareketin yön bilgisini de korumaktadır. Ayrıca kamera kaymasından oluşabilecek yanlış sınıflandırmayı engellemek için 2B HFD yöntemi kullanılmıştır. Bütün bu yöntemler iki farklı veri kümesi üzerinde uygulanmıştır. İlk veri kümesi, zengin fakat karmaşık bir yapıya sahip olan ChaLearn veri kümesidir. İkinci veri kümesi ise Kinect v2 kamera kullanarak bizim oluşturduğumuz karışık veri kümesidir. Veri kümeleri içinde eğitim ve test verilerini karşılaştırıp, eşleşen hareketleri sınıflandırmak için mesafe ölçümleri ve çapraz korelasyon kullanılmıştır. Bu mesafe ölçümleri Levenshtein, Frobenius ve Mahalanobis'tir. Çalışma sonuçları incelendiğinde ChaLearn veri kümesine uygulanan yöntemlerin %76.15 başarı ve karışık veri kümesine uygulanan yöntemlerin %84.1 başarı elde ettiği gözlemlenmiştir. Ek olarak Levenshtein mesafe ölçümü ile yapılan sınıflandırma, diğer mesafe ölçümleri ile yapılan sınıflandırmalardan daha doğru bir sonuç vermektedir.

Anahtar Sözcükler : Tek adımda öğrenme, Hareket tanıma, ChaLearn veri kümesi, Karışık veri kümesi, SS, HGG, 2B HFD.

Bilim Kodu : 924.1.014

ABSTRACT

M. Sc. Thesis

HAND GESTURE CLASSIFICATION WITH ONE-SHOT LEARNING APPROACH

Esmā ŞEKER

Karabük University

Graduate School of Natural and Applied Sciences

Department of Computer Engineering

Thesis Advisor:

Assoc. Prof. Dr. Oğuz FINDIK

June 2019, 49 pages

Nowadays, the process of recognizing human gestures without using any vehicle of human computer interaction is one of the growing research areas. Therefore, many researchers have suggested various approaches to recognize human gestures. Some of these approaches require prior knowledge about the gesture. The other part needs very large datasets. The one-shot learning approach does not require prior knowledge about the gesture or a large amount of data. Hand gestures play an important role in people's lives. Thus, the hand gesture recognition system provides an innovative and natural way for people to interact with the computer. This study aims to classify hand gestures using one-shot learning approach. The one-shot learning approach represents the gesture in a video with a single frame. In order to apply the approach in the study, standard deviation (STD), motion history image (MHI) and 2 dimensional fast Fourier transform (2D FFT) methods were used. A video containing

gesture within the STD method was converted into only one frame. However, this method cannot preserve the history knowledge of the gesture. Therefore, MHI method was used with STD in the study. The method preserves the historical knowledge and the direction of gesture. In addition, 2D FFT method was used to prevent misclassification from camera shift. All these methods were applied to two different datasets. The first dataset is a very rich and complex ChaLearn dataset. The second dataset is mixed dataset that we created using the Kinect v2 camera. Distance measurements and cross-correlation were used to compare training and test data in datasets and to classify matching gestures. These distance measurements are Levenshtein, Frobenius and Mahalanobis. When the results of the study were examined, it was observed that the methods applied to the ChaLearn dataset achieved success of 76.15% and the methods applied to the mixed dataset achieved 84.1% success. In addition, it is observed that Levenshtein distance measurement supplies a more accurate classification compared to other distance measurement.

Key Word : One-shot learning, Gesture recognition, ChaLearn dataset, Mixed dataset, STD, MHI, 2D FFT.

Science Code : 924.1.014

TEŐEKKÜR

Bu tez alıőmasının planlanmasında, araőtırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteęini esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında şekillendiren sayın hocam Do. Dr. Oęuz FINDIK'a;

Hayatta hiçbir zaman pes etmeme izin vermeyen babam Őeref ŐEKER'e;

Maddi manevi hiçbir yardımı esirgemedен yanımda olan ve ümidimi kaybettiğim her anda beni cesaretlendiren, kendimi şanslı hissettiren dünyanın en sabırlı annesi Sultan ŐEKER'e ve sevgili kardeşlerime;

Ve tanıdığım günden itibaren "İyi ki varsın." dedirten hayat arkadaşım Ömer AYDIN'a sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

	<u>Sayfa</u>
KABUL.....	ii
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ.....	xii
ÇİZELGELER DİZİNİ	xiii
SİMGELER VE KISALTMALAR DİZİNİ	xiii
BÖLÜM 1	1
GİRİŞ	1
BÖLÜM 2	7
YÖNTEM.....	7
2.1. OTSU EŞİK YÖNTEMİ	7
2.2. HAREKET TANIMA YÖNTEMİ.....	11
2.2.1. Standart Sapma Yöntemi	11
2.2.2. Hareket Geçmiş Görüntü Yöntemi	13
2.2.3. 2B Hızlı Fourier Dönüşümü Yöntemi	16
2.3. HAREKET SINIRLAMA YÖNTEMİ.....	18
2.4. VERİ KÜMELERİ.....	21
2.4.1. ChaLearn Veri Kümesi	21
2.4.2. Karışık Veri Kümesi	26
2.5. MESAFE ÖLÇÜMLERİ.....	30

BÖLÜM 3	34
DENEYSEL ÇALIŞMALAR	34

Sayfa

BÖLÜM 4	43
SONUÇ	43

KAYNAKLAR	44
-----------------	----

ÖZGEÇMİŞ	49
----------------	----





ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 2.1. Otsu eşik yönteminin akış diyagramı.....	8
Şekil 2.2. ChaLearn veri kümesindeki görüntülere Otsu yöntemi uygulandıktan sonraki görüntüleri	10
Şekil 2.3. Karışık veri kümesindeki görüntülere Otsu yöntemi uygulandıktan sonraki görüntüleri	10
Şekil 2.4. ChaLearn veri kümesindeki videolara SS yöntemi uygulanması sonucu oluşan görüntüler	12
Şekil 2.5. ChaLearn veri kümesinde 12. dosyadaki eğitim videolarına SS yöntemi uygulandıktan sonraki görüntüleri	15
Şekil 2.6. ChaLearn veri kümesinde 12. dosyadaki eğitim videolarına HGG yöntemi uygulandıktan sonraki görüntüleri	15
Şekil 2.7. ChaLearn veri kümesinde 1. dosyadaki 1., 2. ve 3. test videolarının grafikleri	19
Şekil 2.8. ChaLearn veri kümesinden RGB ve derinlik hareket görüntüleri	26
Şekil 2.9. ChaLearn veri kümesindeki eğitim videolarından alınan görüntüler.....	27
Şekil 2.10. Karışık veri kümesinden alınan eğitim görüntüleri.....	28
Şekil 3.1. ChaLearn veri kümesine Otsu eşik yöntemi uygulanmış görüntüler	36

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 2.1. Hareket tanıma veri kümeleri.....	23
Çizelge 2.2. ChaLearn veri kümesindeki kategoriler.....	25
Çizelge 3.1. ChaLearn veri kümesi ile kullanılan yöntemler ve başarı yüzdeleri.....	40
Çizelge 3.2. Karışık veri kümesi ile kullanılan yöntemler ve başarı yüzdeleri.....	42



SİMGELER VE KISALTMALAR DİZİNİ

KISALTMALAR

MR	: Manyetik Rezonans
ASL	: Amerikan İşaret Dili
BSL	: İngiliz İşaret Dili
JSL	: Japon İşaret Dili
ISL	: Hint İşaret Dili
GMM	: Gizli Markov Modeli
SDM	: Sonlu Durum Makineleri
DBA	: Dinamik Bayesian Ağları
KRA	: Koşullu Rastgele Alanlar
HEG	: Hareket Enerji Görüntü
HGG	: Hareket Geçmiş Görüntü
YGH	: Yönlendirilmiş Gradyanların Histogramı
OAH	: Optik Akış Histogramı
DZE	: Dinamik Zaman Eğimi
GKRA	: Gizli Koşullu Rastgele Alanlar
SS	: Standart Sapma
2B HFD	: 2 Boyutlu Hızlı Fourier Dönüşümü
AFD	: Ayrık Fourier Dönüşümü
3DIG	: 3 Boyutlu İkonik Hareket
CGD13	: ChaLearn Hareket Veri Kümesi 2013
ChAirGest	: İnsan Bilgisayar Etkileşimi İçin Orta Hava Hareketi
SKIG	: Sheffield Kinect Hareket
6DMH	: 6 Boyutlu Hareket Hareketi
G3D	: Oyun 3 Boyutlu
MSRGesture3D	: MSR Hareket 3 Boyutlu
CGD11	: ChaLearn Hareket Veri Kümesi 2011

ASLLVD : Amerikan İşaret Dili Sözlük Video Veri Kümesi
V1 : Versiyon 1
V2 : Versiyon 2



BÖLÜM 1

GİRİŞ

Görüntü işleme, ölçülmüş veya kaydedilmiş olan elektronik görüntü verilerini, elektronik ortamda amaca uygun şekilde değiştirmeye yönelik yapılan bilgisayar çalışmasıdır [1]. Günümüzde görüntü işleme birçok alanda kullanılmakta ve önemi giderek artmaktadır. Bu alanlardan bazıları; tıp, biyoloji, astronomi, fizik, mühendislik, askeri alanlar, güvenlik, savunma, endüstriyel uygulamalar, otomotiv sektörü, petrol arama, trafik, robotik ve bazı geçiş sistemleridir. Tıp alanında kullanılan görüntü işleme ile hastaların manyetik rezonans (MR) görüntülerinin işlenmesi sayesinde bazı hastalıklar teşhis edilmektedir. Biyoloji alanında morfolojik görüntü işleme teknikleri kullanılarak bazı bitkilerdeki hastalık sonucu oluşan lekeler tespit edilebilmektedir. Trafikte kullanılan kameralar ile hız tespiti, kural ihlal tespiti, plaka tanıma gibi işlemlerle cezai işlemler otomatik olarak yapılabilmektedir. Geçiş sistemlerinde, ulaşım kartları yerine parmak izi tanıma sisteminin kullanılması kolaylık sağlamaktadır [2]. Geçmiş zaman ile karşılaştırıldığında görüntü işlemenin insan hayatının tüm alanına yayıldığı görülmektedir. Ayrıca cep telefonları ve bilgisayarlarda bulunan yüz tanıma uygulamaları ile kişilerin yaşlarını ve cinsiyetlerini bilen algoritmalarda bulunmaktadır [2,3].

Görüntü işleme için iki teknik kullanılmaktadır. Bunlar; analog ve dijital görüntü işleme teknikleridir. Analog görüntü işleme teknikleri, fotokopiler ve fotoğraflar gibi basılı kopyalar için kullanılmaktadır. Görüntü analistleri, bu tekniği kullanırken çeşitli şekillerde yorumlamaktadır. Burada görüntü işleme sadece teknik bilgi ile sınırlı kalmayıp, kişinin hayal gücüne dayanmaktadır. Ayrıca analog teknik ile görüntü işlemede ham veri yani geçmişte toplanmış ve işlenmemiş görüntüler önemli bir araç olmaktadır. Analistler, tanımlamak istedikleri ürünler ile ilgili geçmiş görüntüleri sisteme öğretmektedir. Dijital görüntü işleme tekniği, görüntülerin

bilgisayar ile manipüle edilmesini sağlamaktadır. Uydu platformundan alınan görüntüler, algılayıcı hatası nedeniyle eksik bilgi içermektedir. Teknik, eksik bilgileri gidermek ve özgün bilgi elde etmek için 3 temel aşama kullanılmaktadır. Bu aşamalar; ön-işleme, geliştirme ve görüntüleme, bilgi çıkarımıdır [1,3].

Bilgiyi bilgisayara girme yöntemlerinde sistemin verimliliği, kullanılabilirliği ve kullanıcı deneyiminin önemli etkileri bulunmaktadır. İnsanlar, elektronik bilgisayarların ortaya çıktığı andan itibaren mekanik aygıtları girdi yöntemi olarak kullanılmaktadır. İnsan bilgisayar etkileşimi kavramı, Shakel'in 1959 tarihinde yayınladığı "Bilgisayarın ergonomisi" başlıklı makale ile ortaya çıkmıştır. İnsanlar o tarihten itibaren yeni insan bilgisayar etkileşimi yöntemleri geliştirmektedir [2,3]. Donanımsal hesaplama gücündeki hızlı büyüme ve makine öğrenme yöntemlerinin doğruluğu bilgisayar bilimleri araştırma insanlarının temel bakış açısını değiştirmektedir. Ancak bilgilerin makinelere aktarılması için hala çeşitli düğmeler kullanılmaktadır. İnsan bilgisayar etkileşiminde insanların sanal gerçeklikte yürüyeceği, bilgisayarlarla konuşabileceği ve makineleri kontrol etmek için çeşitli el hareketlerini kullanabileceği daha sezgisel ve verimli çalışmalara ihtiyaç duyulmaktadır [2,3,4].

Son yıllarda insan hareketlerini tanımak, insan bilgisayar etkileşimi için önemli bir araştırma konusu haline gelmiştir. İnsan hareketleri, konuşulan kelimelere ek olarak düşmanlıktan sevgiye kadar olan çeşitli duygu ve düşünceleri paylaşmaya olanak sağlamaktadır [4]. Hareket, ellerin, yüzün veya ayakların hareketlerini içeren sesli olmayan bir iletişim şekli olmaktadır. Özellikle konuşamayan ve duymayan veya duyabilen fakat konuşamayan insanlar arasında hareketler işaret dili olarak adlandırılmakta ve yaygın olarak kullanılmaktadır. Bir işaret dili, el şekilleri, vücut hareketleri veya yüz ifadeleri tarafından oluşturulan çeşitli hareketlerden oluşmaktadır. Dünyada her biri kendi hareketlerine sahip olan birçok işaret dili bulunmaktadır. Bunlardan bazıları şunlardır; Amerikan işaret dili (ASL), İngiliz

işaret dili (BSL), Japon işaret dili (JSL), Hint işaret dili (ISL). Fakat işitme engelli insanların kullandığı bu işaret dillerini, normal insanlar öğrenmediği için toplumda sınırlı kalıp işitme engelli insanlar ile normal insanlar arasındaki iletişimde soruna neden olmaktadır [4,5]. Bu nedenle insan bilgisayar etkileşimini daha verimli hale getirmek için hareket tabanlı cihazlar ön plana çıkmıştır. Bu tür cihazlar herhangi bir aygıtla bağlı kalmadan doğrudan vücut hareketlerini kullanmaktadır. Gelişen teknoloji ile birlikte hareket tabanlı cihazlar yalnızca işitme engelli insanlar için kullanılmamış diğer alanlarda da kullanılmıştır. Ancak insan hareketlerini tanıma, insan makine etkileşimi konusunda çok karmaşık araştırma alanlarından biri olmaktadır [4,5,6].

İnsan hareketlerini tanımak için çeşitli yaklaşımlar önerilmiş ve uygulanmıştır. Bu yaklaşımların bir kısmı teorikte kalmış, diğer kısmı ise pratikte istenilen başarıyı elde edememiştir. Pratikte istenilen başarıyı elde edemeyen yaklaşımların bazıları, hareket özelliklerinin kapsamlı bir tanımını kolayca çıkarmak için veri eldiven cihazları veya renk işaretleyicileri gibi ek donanım cihazları kullanmıştır [4,5,7,8]. Bu donanım cihazları insan bilgisayar etkileşiminde istenilen verimi elde edememektedir. Yaklaşımlar, hareketi tanımak için birden fazla örnek kullanan ve tek örnek kullanan olmak üzere iki gruba ayrılmıştır. Birden fazla örnek kullanan yaklaşımlardan birkaçı, gizli Markov modeli (GMM), sonlu durum makineleri (SDM), dinamik Bayesian ağları (DBA), koşullu rastgele alanlar (KRA) olmaktadır. GMM, örüntü ve hareket tanımak için kullanılan en yaygın yaklaşımlardan biri olarak kabul edilmektedir. Yaklaşım, el hareketleri arasında bir bağımlılık olmadığını kabul ederek bağımsızlık varsayımına uymaktadır. Bu varsayım, el yörüngelerinde uzun vadeli zamansal özellikleri atmaktadır. Varsayımın amacı, eğitim ve çıkarım hesaplamalarını izlenebilir duruma getirmektir. Tamamlayıcı bir ölçü olan yaklaşım, bitişik hareketler arasındaki yerel bağımlılıkları simüle etmek için bir dizi geçiş olasılığı kullanmaktadır. Yaklaşımın, eğitim veri kümesindeki hareketler ile gizli durumlar arasındaki durum ve geçiş olasılığı matrislerini tahmin etmesi gerekmektedir. Daha sonra, çıkarım görevi eğitilmiş olasılık matrislerine dayanarak ileriye doğru yayılma yoluyla gerçekleştirilmektedir [9,10]. SDM, genel bir hareketin nitel olarak farklı evrelerini modellemek için kullanılmaktadır. Bu evreler, hareketin statik başlangıç konumu, elin ve parmakların hareketin sonuna kadar düzgün bir

şekilde hareket etmesi, hareketin statik son konumu, elin ve parmakların başlangıç pozisyonuna geri dönmesi olmak üzere dört adımdan oluşmaktadır. SDM, hareketleri vektörlerin bir listesi olarak temsil etmektedir. Bu sayede vektörlerin yer değiştirmelerine dayalı olarak tablo araması kullanılarak veri kümesinde bulunan hareket, vektör modeli ile eşleştirilmektedir [11]. DBA, GMM'leri ve kalman filtrelerini birleştiren genel bir olasılık yaklaşımıdır. Bayesian ağlarının genelleştirilmiş versiyonunu temsil etmektedir. Yaklaşım, dinamik hareketlere kıyasla statik hareketler için daha yüksek bir başarı oranı göstermektedir [12]. KRA, GMM'ne benzer ayırıcı bir olasılıksal model olarak ifade edilmektedir. Yaklaşım, gözlemlenen veri noktalarının temelde özellik ve vektör gösterimlerini destekleyebildiği için güçlü ayırt edici model olarak kabul edilmektedir. Ancak gizli durum değişkenlerini kullanarak ara yapılarını yakalayamadığı için sınırlı bir yapıya sahip olmaktadır [10,12]. Bu tür yaklaşımlar, hareket sınıflarının tüm çeşitlerini kapsayacak şekilde büyük miktarda veri gerektirmektedir. Ayrıca yaklaşımlar, gereken örnek sayısı, girdi alanı, hareket kategorisi ve seçilen öğrenme yöntemi gibi faktörlere bağlı olarak değişmektedir [9,10,11,12].

Tek örnek ile hareket tanıma yaklaşımları, literatürde genel olarak tek adımda öğrenme yaklaşımları olarak bahsedilmektedir. Yaklaşımlar içinde birden fazla hareket örneğinin bulunması insan hareketlerini tanıma doğruluğunu arttırmasına rağmen tek adımda öğrenme yaklaşımlarında her sınıfın yalnızca bir eğitim örneği bulundurma zorunluluğu bulunmaktadır [13]. Tek adımda öğrenme yaklaşımları genel olarak, hareketlerin görünümü, şekli, mekansal-zamansal yönelimi, optik akışı, ilgi noktası ve hacimine göre farklılık göstermektedir [13]. Mekansal-zamansal yaklaşımlarda, hareket bölgelerini belirten ve hareketin zaman bilgisini koruyan bir hareket enerji görüntü (HEG) ve hareket geçmiş görüntü (HGG) yöntemleri kullanılmıştır. Yaklaşım ilk olarak 2 boyutlu kullanılmış, daha sonra 3 boyutlu olarak kullanılması önerilmiştir [14]. Optik akış tabanlı yaklaşımlarda hareket algılama ve yerleştirme yöntemleri ile frekans bölgesi gösterimi kullanılmaktadır [15]. Model tabanlı yaklaşımlarda insan vücudunun uzuvları tanımlanıp vücudun 2 boyutlu ve 3 boyutlu modelleri kullanılmaktadır. Modeller poz tahminlerine ve rakam merkezli ölçümlere dayanmaktadır [16]. Günümüzde yaygın olarak kullanılan tek adımda öğrenme yaklaşımlarının birkaçı şunlardır; yönlendirilmiş gradyanların histogramı

(YGH) ve optik akış histogramı (OAH), özelliklerin çantası, dinamik zaman eğimi (DZE), gizli koşullu rastgele alanlar (GKRA). YGH/OAH, tek adımda öğrenme yaklaşımlarında yaygın olarak kullanılmaktadır. YGH/OAH, hareketleri eşleştirmek için GMM yaklaşımı ile birlikte kullanılmaktadır [17]. Ayrıca YGH, el hareketlerini tanımak için kenar ve gradyan tabanlı hareket tanımlayıcıları ile birlikte uygulanabilmektedir. YGH, yerel bir objenin görünüşünün ve şeklinin, yoğunluk gradyan yönleri dağılımını iyi bir şekilde analiz edebilmektedir. Bu sayede, geometrik ve optik deformasyon üzerinde iyi bir değişkenlik sağlamaktadır [17,18,19]. Ancak, el hareketlerini yalnızca basit bir arka planda algılayabilmektedir. Bu nedenle görüntünün arka planı karışık bir yapıya sahip olduğunda başarısızlık ile sonuçlanmaktadır. OAH, iki görüntü arasındaki hareketi her durumda tahmin etmeye çalışmaktadır. OAH denklemi, iki değişkenli tek bir denklem olarak ifade edilmektedir. Tüm OAH yöntemlerinde akışı tahmin etmek için ek koşullar sağlanmaktadır [17]. Özelliklerin çantası yaklaşımı, hem RGB hem de derinlik görüntülerinin bir araya getirildiği 3B hareketleri bulunan düşük seviye özellikler için kullanılmaktadır [20]. DZE yaklaşımı, hareketlerin yörüngelerinin tahmini ölçüğüne ve konumuna ihtiyaç duymaktadır. Bir budama tekniği, hipotez miktarını azaltmak için yaklaşımda kullanılmaktadır [21,22,23]. Yaklaşım normalde şablonlardaki zamansal öge yer değiştirmelerinin üstesinden gelmek için kullanılmaktadır. Elementlerin sıralı düzeni, test veri kümesinde belirli bir varyans seviyesine sahip olmaktadır. Yaklaşım, elementlerin sırasını göz önünde bulundurmadan eğitim veri kümesinde karşılık gelen bir zaman dilimi içindeki elemanlar ile eşleştirerek varyansın üstesinden gelmektedir. Ancak, hareket yörünge uzunluğunun budama için bir ölçüt olarak kullanılması yaklaşımda hareket hızının sabit olduğu anlamına gelmektedir. Kişinin hareketleri, eğitim veri kümesindeki kişiden daha hızlı olarak test veri kümesini işaret ettiği takdirde verilerin gerçek işaret etiketlerinden bağımsız olarak, bu test verilerini budama olasılığı yüksek bir değere sahip olmaktadır [23,24,25,26]. Tek örnek ile hareketleri tanımak için DZE'yi kullanan başarılı yaklaşımlar olmuştur. Fakat bu yaklaşımlar, hareketler arasındaki mesafenin hareketlerin bitiş noktalarına ilişkin bir fikir vermesi için eş zamanlı segmentasyon ve tanıma gerektirmektedir [24,25]. Hareketler genellikle karmaşık bir alt yapıya sahip olmaktadır. Bazı araştırmacılar tarafından GKRA yaklaşımının gizli ve karmaşık olan modelleri tanımak için avantajlı olduğu ifade edilmektedir. Gizli

durumlarla hareket tanımaya yönelik mevcut yaklaşımların çoğu hareket akışlarını modellemek için GMM yaklaşımını kullanmaktadır. Gizli durumlar, hareket dizilerinin iç yapısını simüle etmek için kullanılmaktadır. Her gizli durum, önceden tanımlanmış sınıfların bir bileşeni olarak kabul edilmektedir. Yaklaşım, kontrollü ortamlarda el hareketlerinin vuruşları arasındaki benzerliklerin temel yapısını modellemek için kullanılmıştır. GKRA yaklaşımının dezavantajı, olasılık fonksiyonu dış bükey olmamaktadır. Bu nedenle gizli durum ile eğitim süreci küresel maksimuma ulaşmamaktadır [10]. Tek adımda öğrenme yaklaşımlarında hareketleri birbirinden ayırt edecek özelliklerin nasıl ortaya çıkarılacağı ve her sınıf başına yalnızca bir eğitim örneğinden ayırt edici bir hareketin nasıl öğrenileceği olmak üzere iki önemli sorun bulunmaktadır.

Anlatılan bütün yaklaşımların bazı dezavantajları bulunmaktadır. Bu dezavantajlardan birkaçı şunlardır; 1) Çok büyük sayıda veri kümelerine ihtiyaç duyulmaktadır. 2) Yaklaşımların hareket hakkında önceden bilgi sahibi olmaları gerekmektedir. 3) Yeni hareketleri tanımakta sorun yaşamaktadırlar. Diğer dezavantajlar yaklaşımlara göre farklılık göstermektedir.

El hareketleri, insan bilgisayar etkileşimi için doğal ve sezgisel bir iletişim yöntemi sağlamaktadır. Kullanıcı elinin bir cihaz yardımıyla bilgisayar ile etkileşimi sağlaması itici bir güç olmaktadır. Bilgisayarların el hareketlerini görsel olarak tanıyıp doğru bir şekilde sınıflandırma yapabilmesi için verimli insan bilgisayar arayüzlerine ihtiyaç duyulmaktadır. Ancak bu durum çeşitlilik bakımından zengin olan el hareketlerinin karmaşıklığı nedeniyle zorlu bir sorun olmaktadır. El hareketlerini tanımak ve sınıflandırmak; (i) hareketin büyüklüğüne, benzerliğine, karmaşıklığına ve çift el hareketi kullanımına, (ii) hareketin aydınlatma koşullarına, arka plan içeriğine ve tıkanmalarına, (iii) hareketin sürekli olmasına, yüz hareketleri ile benzemesine, görüntüden çıkmasına, hareket sırasında duraklamasına, (iv) hareketin boyutuna, hızına, konumuna ve yönlendirmesine bağlı olmaktadır.

Çalışmada el hareketlerini sınıflandırmak için tek adımda öğrenme yaklaşımı kullanılmıştır. Yaklaşım, görüntü edinimi, ön işleme, özellik çıkarma ve sınıflandırma olarak dört temel adımdan oluşmaktadır. Görüntü edinimi için

çalışmada iki farklı veri kümesi kullanılmıştır. İlk veri kümesi çok zengin fakat karmaşık bir yapıya sahip olan ChaLearn veri kümesidir. İkinci veri kümesi ise bizim oluşturduğumuz karışık veri kümesidir. Ön işleme adımında görüntüleri arka plandan ayırmak için Otsu eşik yöntemi uygulanmıştır. Görüntülerden özellik çıkarımı Standart sapma (SS) yöntemi, Hareket geçmiş görüntüsü (HGG) yöntemi ve 2 boyutlu hızlı Fourier dönüşümü (2B HFD) yöntemi olmak üzere üç temel yöntem ile sağlanmıştır. Son olarak çalışmada sınıflandırma işlemini gerçekleştirmek için Levenshtein, Frobenius ve Mahalanobis olmak üzere üç farklı mesafe ölçümü ve çapraz korelasyon kullanılmıştır.

BÖLÜM 2

YÖNTEM

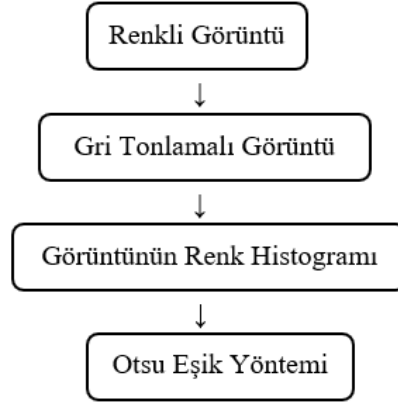
2.1. OTSU EŞİK YÖNTEMİ

Adını yöntemi geliştiren Nobuyuki Otsu'dan alan Otsu yöntemi, gri tonlamalı görüntüler üzerinde uygulanan bir eşik seviye yöntemidir. Bu yöntem kullanılırken görüntünün ön plan ve arka plan olmak üzere iki renk sınıfından oluştuğu kabul edilmektedir. Bu renk sınıflarının, sınıf içi varyans değeri tüm eşik değerleri için hesaplanmaktadır [27,28]. Sınıf içi varyans değerinin en küçük olmasını sağlayan eşik değeri, optimum eşik değeri olarak kabul edilmektedir. Sınıf içi varyans değeri minimum değerinde iken sınıflar arası varyans değeri maksimum değerinde olmaktadır. Sınıflar arasındaki varyans değerinin maksimum olması çalışmanın daha hızlı ve doğru sonuçlar vermesini sağlamaktadır [28].

Otsu eşik yöntemi gri tonlamalı görüntüler üzerinde uygulanırken yalnızca renklerin bir görüntü içinde ne kadar bulunduğunu kontrol etmektedir. Renklerin görüntü üzerindeki sayısına bakıldığı için eşik yöntemi uygulanmadan önce görüntünün renk histogramı hesaplanıp bütün işlemler bu histogram dizisi üzerinden yapılmaktadır. Yöntem, histogram üzerinde eşikleme yapılabilir en uygun konumun bulunması için kullanılmaktadır. Histogram içindeki her eleman eşik değeri kabul edilerek diğer elemanlar için hesaplamalar yapılmaktadır. Belirlenen eşik değerinden önceki elemanlar eşik öncesi, belirlenen eşik değerinden sonraki elemanlar eşik sonrası

olarak adlandırılmaktadır [27,28,29,30]. Şekil 2.1’de Otsu eşik yönteminin akış diyagramı gösterilmektedir.





Şekil 2.1. Otsu eşik yönteminin akış diyagramı.

Varyans, belli bir sayı dizisinin, bu dizinin aritmetik ortalaması etrafındaki dağılımının yorumlanabilmesini sağlayan bir ölçü olarak kabul edilmektedir. Varyans değeri, büyük ise değerlerin dağınık ve kendi ortalamalarından uzak oldukları, küçük ise kendi ortalamaları etrafında çok dağılmadıkları anlaşılmaktadır [27,28,30]. Varyans değeri Eş. 2.1 kullanılarak hesaplanmaktadır [27,28,30].

$$\sigma^2 = \sum_{i=1}^N (x_i - \bar{x}_i)^2 - Pr(x_i) \quad (2.1)$$

Burada σ^2 varyans değeri, $(x_i - \bar{x}_i)^2$ ağırlıklı ortalama değeri, $Pr(x_i)$ ise olasılık fonksiyonunu temsil etmektedir. Bir görüntü içindeki bir renk değerinin olasılığı, değer in görüntü içindeki sayısının, toplam piksel sayısına bölümünden oluşmaktadır. Sınıf içi varyans değeri Eş. 2.2 kullanılarak hesaplanmaktadır [27,28,30].

$$\sigma_G^2 = \sum_{i=0}^{N-1} (i - \mu_G)^2 P_i \quad (2.2)$$

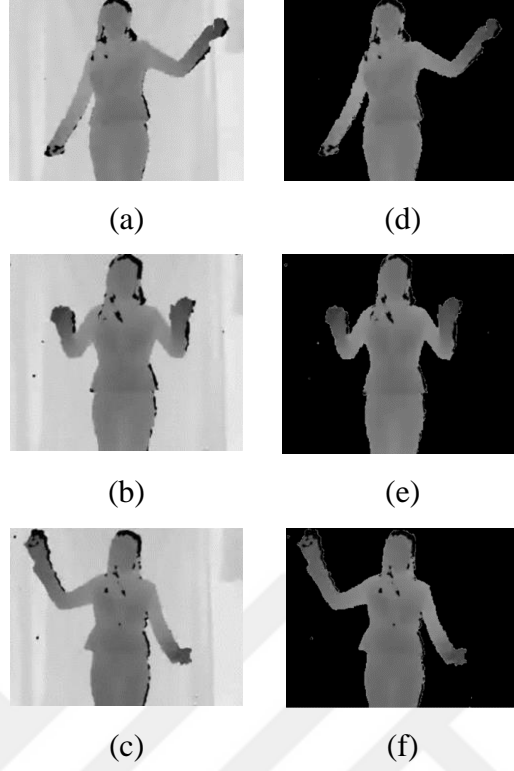
Burada σ_G^2 sınıf içi varyans değerini, μ_G sınıf içi piksellerin ortalama yoğunluğunu ve P_i , i değerinin olasılığını ifade etmektedir. Sınıflar arası varyans değeri Eş. 2.3 kullanılarak hesaplanmaktadır [27,28,30].

$$\sigma_B^2 = P_1(\mu_1 - \mu_G)^2 + P_2(\mu_2 - \mu_G)^2 \quad (2.3)$$

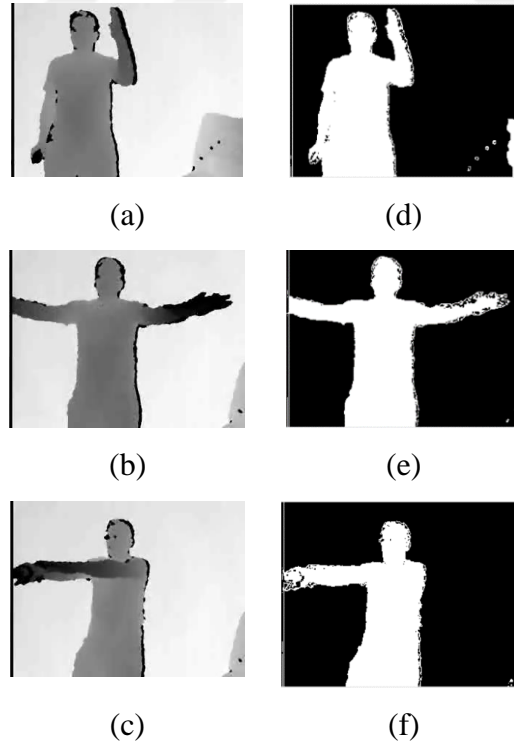
Burada σ_B^2 sınıflar arası varyans deęerini, μ_1 ilk sınıfın ve μ_2 ikinci sınıfın piksel yoęunluęunu vermektedir [27,28,30].

ChaLearn veri kümesi içindeki her dosyada hem RGB hem de derinlik videoları bulunmaktadır. Çalışmada ChaLearn veri kümesindeki derinlik videoları kullanılmıştır. Ancak derinlik videoları çalışma içinde kullanılmadan önce gri tonlamaya dönüştürülmüştür. Derinlik verilerinin gri tonlamaya dönüştürülmesi bizlere iki büyük avantaj sağlamaktadır. İlk olarak, gri tonlama, nesnenin kameraya uzaklığı ile orantılı olarak piksel yoęunluęu bilgisini vermektedir. İkincisi ise, ön planı arka plandan ayırmak için görüntünün her pikselinde gri tonlama eřięinin kullanılmasını sağlamaktadır. Çalışmada gri görüntüler elde edildikten sonra, daha doęru bir sınıflandırma yapabilmek için görüntülerin ikili görüntülere dönüştürülmesi gerekmektedir. Aslında bir gri görüntüyü ikili görüntüye çevirmek için kullanılacak yöntem oldukça basittir. Bir eşik deęeri belirlenmekte ve bu deęerin üzerindeki renkler beyaza, altındaki renkler ise siyaha dönüştürülmektedir. Ancak tüm görüntüler aynı özelliklere sahip olmamaktadır. Sabit bir eşik deęeri tüm görüntüler için geçerli sonuç üretememektedir. Bu nedenle eşik deęerinin, görüntünün renk daęılımına göre belirlenmesini sağlayacak bir yöntem ihtiyacı duyulmaktadır. Otsu eşik yöntemi, görüntüdeki renk daęılımına göre ön planın arka plandan ayrılmasını sağlamıştır.

Şekil 2.2’de ChaLearn veri kümesindeki ve Şekil 2.3’te karışık veri kümesindeki eğitim videolarından alınan görüntülerin Otsu yöntemi uygulandıktan sonraki görüntüsü gösterilmiştir.



Şekil 2.2. ChaLearn veri kümesindeki görüntülere Otsu yöntemi uygulandıktan sonraki görüntüleri.



Şekil 2.3. Karışık veri kümesindeki görüntülere Otsu yöntemi uygulandıktan sonraki görüntüleri.

2.2. HAREKET TANIMA YÖNTEMLERİ

2.2.1. Standart Sapma Yöntemi

Çalışmada tek adımda öğrenme yaklaşımı çeşitli yöntemler ile birlikte kullanılmıştır. Bu yöntemlerden biri de standart sapma (SS) yöntemidir.

Normal hayatta kullanılan standart sapma, olasılık kuramı ve istatistik bilim dallarında, bir anakütle, bir örneklem, bir olasılık dağılımı veya bir rassal değişken, veri değerlerinin yayılımının özetlenmesi için kullanılan bir ölçü olarak ifade edilmektedir [31]. Bir dizinin standart sapma değerini hesaplamak için;

1. Sayıların aritmetik ortalaması hesaplanır.
2. Her sayının aritmetik ortalamadan farkı bulunur.
3. Bulunan farkların her birinin karesi hesaplanır.
4. Farkların kareleri toplanır.
5. Elde edilen toplam, dizinin eleman sayısının bir eksiğine bölünür.
6. Bulunan sayının karekökü alınır.

SS yöntemi çalışma içinde videoları oluşturan görüntüler için kullanılmıştır. Yöntem, bir videoyu oluşturan bütün görüntülerden tek bir görüntü elde etmeyi amaçlamaktadır. Kullanılan bu yöntem normal hayatta kullanılan standart sapma yönteminden biraz farklıdır. Görüntü işlemek için kullanılan SS yönteminin algoritması aşağıda belirtilmiştir.

1. Bir videoyu oluşturan bütün görüntüler 240x320'lik matrislerden oluşmaktadır. Veri kümesindeki videolar, görüntü haline getirilmiştir.
2. Bir video içindeki tüm görüntüler toplanıp bir videoyu oluşturan toplam görüntü sayısına bölünmektedir. Böylece ortalama bir görüntü elde edilmektedir.
3. Videoyu oluşturan her görüntü ortalama görüntüden çıkarılmaktadır.
4. Elde edilen görüntülerin kareleri hesaplanıp toplanmaktadır.
5. Ortaya çıkan sonuç toplam görüntü sayısına bölünmektedir.

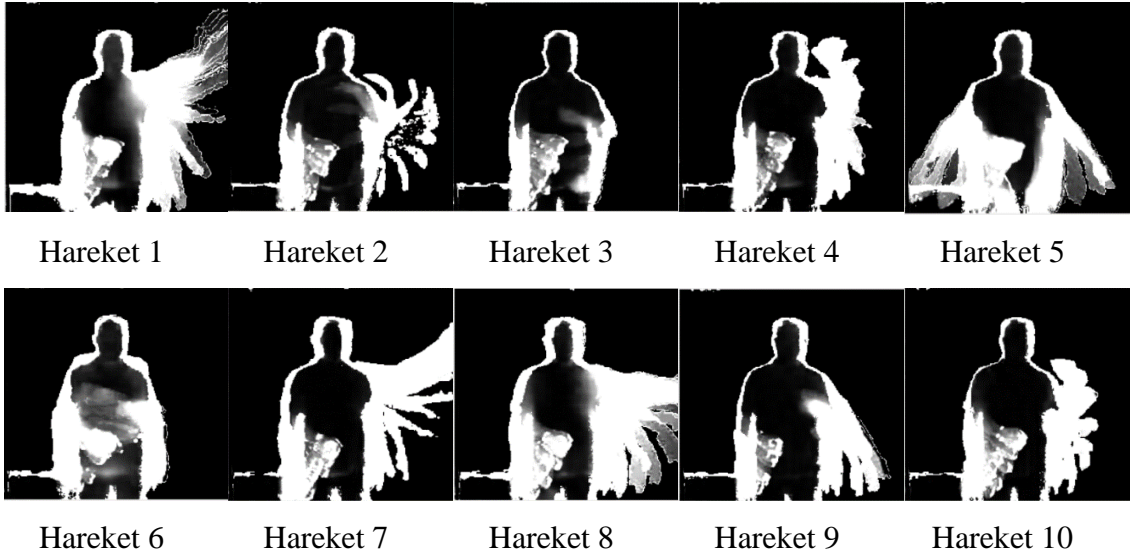
Uygulanan SS yöntemi sonucunda bir videodan bir görüntü elde edilmektedir. Şekil 2.4'te ChaLearn veri kümesindeki 4. dosyada bulunan eğitim videolarına SS yöntemi uygulanması sonucu elde edilen görüntüler gösterilmiştir. Bir video içindeki ortalama görüntü Eş. 2.4 kullanılarak hesaplanmaktadır [32].

$$\bar{I}_{xy} = \frac{\sum I_{xy}}{N} \quad (2.4)$$

Burada \bar{I}_{xy} bir videodaki ortalama görüntü, I_{xy} bir videodaki görüntülerin toplamı, N ise bir videodaki toplam görüntü sayısıdır. Görüntüler matrislerden oluşmaktadır. $x = 1, 2, 3, \dots, m$ satır sayısı, $y = 1, 2, 3, \dots, n$ sütun sayısını belirtmektedir. Bir videoya SS yönteminin uygulanması sonucu oluşan görüntü ise Eş. 2.5 kullanılarak hesaplanmaktadır [32].

$$\sigma = \frac{\sum (I_{xy} - \bar{I}_{xy})^2}{N} \quad (2.5)$$

Burada σ , SS yöntemi sonucu oluşan görüntüyü ifade etmektedir.



Şekil 2.4. ChaLearn veri kümesindeki videolara SS yöntemi uygulanması sonucu oluşan görüntüler.

Şekil 2.4'te, ChaLearn veri kümesinde bulunan 4. dosyadaki eğitim videolarının SS yöntemi uygulanmış görüntüleri gösterilmektedir. Bu dosya içinde 10 tane eğitim videosu bulunmaktadır. Videolar dalış hareketlerini temsil etmektedir. Her video içinde bir dalış hareketi bulunmaktadır. Hareketler sırasıyla dalış esnasında; “dur, orada kal”, “havam kesildi”, “havam az”, “kulağımda problem var”, “aşağıda buluşalım”, “üşüdüm”, “yardım edin”, “sen”, “ben”, “rezervdeyim” ifadelerine karşılık gelmektedir.

2.2.2. Hareket Geçmiş Görüntü Yöntemi

İnsan hareketlerini tanıma yaklaşımları, şablon eşleştirme, durum yaklaşımı ve insan davranışlarının anlamsal açıklaması olmak üzere üç gruba ayrılmaktadır. Hareket geçmiş görüntü yöntemi, şablon eşleştirme yaklaşımlarında kullanılmaktadır. Şablon eşleştirmeye dayalı yaklaşımlarda öncelikle görüntü sırası statik şekil düzenine dönüştürülmektedir. Daha sonra veri kümesine önceden kaydedilmiş olan hareketler ile karşılaştırılmaktadır. Şablon eşleştirme yaklaşımları gürültüye daha yatkın olmakla birlikte hareketlerin zaman aralıklarındaki değişikliklerine de duyarlı olmaktadır [33,34]. Ayrıca insan hareketlerini tanıma yaklaşımları, hareketlerin görünümü, şekli, optik akışı, ilgi noktası ve hacmine göre farklılık göstermektedir. Görünüm tabanlı hareket tanıma, insan vücudunda veya çevresinde herhangi bir cihaz bulunmadan bir hareketi tanımak için kullanılan en kolay hareket tanıma yöntemlerinden biri olarak kabul edilmektedir [35]. Hareket geçmiş görüntü yöntemi, görünüme dayalı şablon eşleştirme yaklaşımlarında kullanılmaktadır.

HGG yöntemi, gri tonlamalı görüntülere yoğunlaşmış baskın hareket bilgilerini korumaktadır. Bu sayede, hareket sırasını net bir şekilde gösterebilmektedir. Ayrıca yöntem gölgelere, eksik parçalara veya gürültüye karşı hassas bir yapıya sahip olmamaktadır. Bu avantajlar, yöntemi hareket analizi için uygun hale getirmektedir [36,37].

Yöntem, tek bir görüntüde kodlanmış zaman aralığını temsil etme avantajına sahip olmaktadır. Bu sayede, insan hareketlerinin zaman ölçeğini kapsamaktadır. Her pikselin yoğunluğunu geçici bir şekilde kullanarak hareket akışını veya dizisini ifade

etmektedir. Ayrıca yöntem, genel hareket kalıplarını tanımaktadır. Böylece hareketin kolayca tespit edilemeyeceği düşük ışıklı alanlarda ucuz kameralarla veya düşük güçlü CPU'larla uygulanabilmektedir. Yöntemin başka bir avantajı, hareket yönüne karşı duyarlı olmasıdır. Bu sayede, elin yukarıdan aşağıya veya aşağıdan yukarıya gitmesi gibi zıt yönlü hareketleri kolayca ayırt edebilmektedir [36,37]. HGG yöntemi, Eş. 2.6 kullanarak öncelikle gri görüntüleri ikili görüntülere dönüştürmektedir.

$$\Psi(x, y, t) = \begin{cases} 1 & \text{ise } z(x, y, t) \geq \Phi \\ 0 & \text{aksi halde,} \end{cases} \quad (2.6)$$

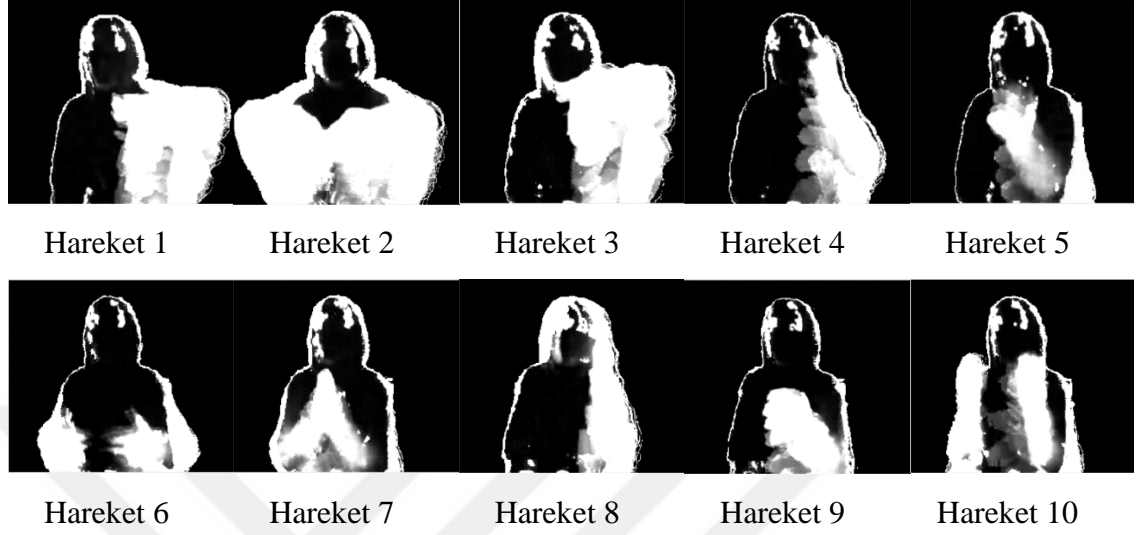
Burada $\Psi(x, y, t)$ t zamanında (x,y) piksellerine sahip olan görüntüyü, Φ eşik değerini ifade etmektedir. Eş. 2.7, HGG yöntemi kullanarak bir hareketi temsil eden tüm görüntüleri, tek bir görüntü haline getirmeyi hesaplamaktadır [36,37].

$$H_\tau(x, y, t) = \begin{cases} \tau & \text{eğer } D(x, y, t) = 1 \text{ ise} \\ \text{Max}(0, H_\tau(x, y, t - 1) - 1) & \text{aksi halde} \end{cases} \quad (2.7)$$

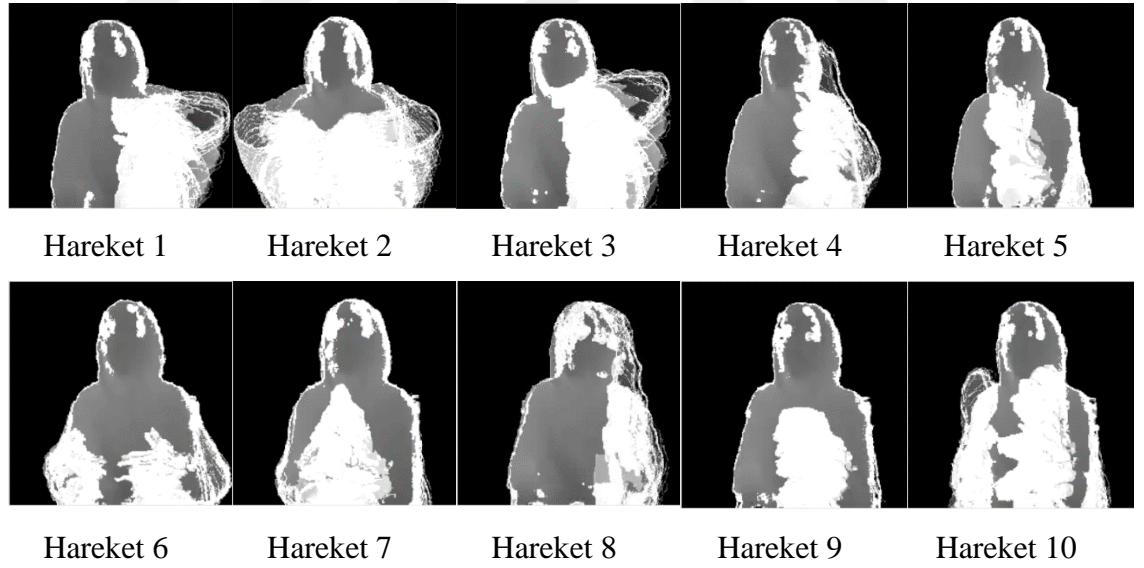
Burada τ görüntü sırası, $H_\tau(x, y, t)$ t değeri 1 ile τ arası değerleri almaktadır. Görüntüdeki piksel değeri 1 ise, o görüntünün zamansal numarası o piksele aktarılmaktadır. Eğer o piksel 0 ise, bir önceki görüntüdeki değerine bakılarak herhangi bir hareket olup olmadığı bilgisi öğrenilmektedir. Hareket varsa, bir önceki görüntünün zamansal numarası bulunan piksele aktarılmaktadır [36,37]. Sonuç olarak, yöntem sayesinde görüntüye ilişkin hareket bilgisi eklenirken bir önceki görüntüde bulunan hareket bilgisi de eklenmiş olmaktadır.

Hareket geçmiş görüntü yöntemi, tek adımda öğrenme yaklaşımı için kullanılan yöntemlerden biridir. Çalışmada SS yöntemi tek başına kullanıldığında hareketin akış yönünü göstermediğinden hareketin geçmiş bilgisi kaybolmaktadır. Bu durum yanlış sınıflandırmaya neden olmaktadır. HGG yöntemi, hareketin akış yönünü ve buna bağlı olarak hareketli parçaların geçici konum bilgisini yakalamak için uygulanmaktadır. Şekil 2.5'te ve Şekil 2.6'da ChaLearn veri kümesinde 12. dosyada bulunan eğitim videolarına SS yöntemi ve HGG yöntemi uygulandıktan sonraki görüntüleri gösterilmektedir. Şekiller incelendiğinde SS yönteminin hareketin akış

yönü bilgisini kaybettiği ancak HGG yönteminin hareketin akış yönü bilgisini koruduğu net bir şekilde görülmektedir.



Şekil 2.5. ChaLearn veri kümesinde 12. dosyadaki eğitim videolarına SS yöntemi uygulandıktan sonraki görüntüleri.



Şekil 2.6. ChaLearn veri kümesinde 12. dosyadaki eğitim videolarına HGG yöntemi uygulandıktan sonraki görüntüleri.

Şekil 2.5'te ve Şekil 2.6'te, ChaLearn veri kümesinde bulunan 12. dosyadaki eğitim videolarının SS ve HGG yöntemi uygulanmış görüntüleri gösterilmektedir. Her dosya içinde 10 tane eğitim videosu bulunmaktadır. Videolar italyan el hareketlerini temsil etmektedir. Her video içinde bir italyan el hareketi bulunmaktadır. Şekillerde

bulunan hareketler sırasıyla; “git başımdan”, “birlikte”, “buraya gel”, “güzel”, “umrumda değil”, “ne topları”, “leydim”, “sen delisin”, “mükemmel”, “ne istiyorsun” ifadelerine karşılık gelmektedir.

2.2.3. 2B Hızlı Fourier Dönüşümü Yöntemi

Fourier dönüşümü, herhangi bir işaretin, farklı genlik ve frekanstaki cosinus ve sinus’lerin toplamı şeklinde yazılabileceğini ifade etmektedir. Dönüşüm, k uzayında tanımlı bir işaretin içerdiği frekans bileşenleri ve ağırlıkları cinsinden tanımlanma imkanı sağlamaktadır. Dönüşüm, kullanmanın iki temel nedeni bulunmaktadır. İlki, işlem kolaylığı sağlamaktadır. İkinci ise, bulunduğumuz uzayda doğrudan görülmesi imkansız olan bileşenleri görülebilir hale getirmektedir [38,39]. Fourier dönüşümü, sonsuz uzunluktaki bir dizi için tanımlanmaktadır. Ancak bu dizinin sınırlandırılması ve sınırlı sayıda nokta için değerlendirilmesi gerekmektedir. Ayrık fourier dönüşümü (AFD), bu sorunu gidermektedir. AFD, ayrık zamanlı sinyal işleme algoritma ve sistemlerinin analizi, tasarımı, gerçekleştirilmesi ile doğrusal filtreleme, korelasyon analizi ve spektrum analizi gibi sinyal işleme uygulamalarında önemli bir rol oynamaktadır. AFD, Fouier dönüşümünün eşit aralıklı frekanslardaki örneklerine bire bir benzemektedir. Eşt. 2.8 kullanılarak ayrık Fourier dönüşümü (AFD) hesaplanmaktadır [38,39].

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk} \quad 0 \leq k \leq N-1 \quad (2.8)$$

Burada $W_n = e^{-j(2\pi/N)}$ ’dir. AFD hesaplanmasında her bir $X(k)$ değeri için N adet karmaşık çarpma ve $N-1$ adet karmaşık toplama işlemi yapılmaktadır. Bu nedenle N adet AFD değeri bulunurken, N^2 adet çarpma ve $N(N-1)$ adet toplama işlemi gerekmektedir. Ayrıca her karmaşık çarpma işlemi, dört gerçel çarpma ve iki gerçel toplama işlemi ve her karmaşık toplama işlemi, iki gerçel toplama işlemi ile gerçekleşmektedir. Sonuç olarak, N ’nin büyük olması durumunda AFD yönteminin uygulanması çok fazla miktarda işlem yapılmasına neden olmaktadır [38,39]. Bu nedenle, AFD yöntemi için gerekli işlem miktarını azaltacak olan hızlı Fourier dönüşümü (HFD) yöntemi geliştirilmiştir. Bu yöntem ile AFD yöntemi için

yapılması gereken işlem sayısı büyük ölçüde azaltılarak işlem kolaylığı sağlanmıştır. HFD, AFD hesaplaması için etkili ve ekonomik bir yöntemdir [39,40].

HFD yöntemi, görüntü işlemede kullanılan önemli yöntemlerden biridir. Yöntem, uzaysal alanda başarılması zor işlemleri, frekans alanında başarabilmektedir. Bu sayede görüntüyü oluşturan frekans bileşenlerini birbirinden ayırt edebildiği için değişik derecelerden alçak ve yüksek filtreleme işlemlerini kolaylıkla yapabilmektedir [39,40].

El hareketlerini sınıflandırmak için tek adımda öğrenme yaklaşımı çeşitli yöntemler ile birlikte çalışmada uygulanmıştır. Bu yöntemlerden biri de 2B HFD yöntemidir. Yöntem çalışma içinde tek başına kullanılmamıştır. SS yönteminden önce ve HGG yönteminden sonra kullanılmıştır. SS yönteminde önce 2B HFD yöntemini uygulamanın avantajı, kişinin derinlik değerleri frekans alanına alındığında, kişinin pozisyonu önemsiz hale gelmektedir. Böylece, kameranın herhangi bir dikey veya yatay hareketi frekans alanında geçersiz sayılmaktadır. Örneğin, kamera eğitim verilerinde sabit, fakat test verilerinde hareketli olması yanlış sınıflandırmaya neden olmaktadır. Ancak 2B HFD yönteminin, SS yönteminden önce kullanılması yanlış sınıflandırma olmasını engellemiştir. Bununla birlikte, arka plan çıkarılmış derinlik verilerinin, görüntüler arasında SS yönteminden önce spektral alana dönüştürülmesi, zaman çözünürlüğünü baskılayacak ve böylece kamera hareketinin etkisini azaltacaktır. Bu sayede eşleştirme için eğitim ve test verilerini benzer konuma getirecektir [39,40]. 2B HFD yöntemi Eş. 2.9 kullanılarak hesaplanmaktadır.

$$F_{uv}(t) = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I_{xy}(t) e^{-2i\pi((ux/M)+(vy/N))} \quad (2.9)$$

Burada $M \times N$ görüntünün satır ve sütun sayısını, $I_{xy}(t)$ t zamanındaki görüntünün (x,y) pikselindeki değerini, üstel terim ise Fourier alandaki her bir $F_{uv}(t)$ noktasına karşılık gelen temel fonksiyonu ifade etmektedir [39,40].

2.3. HAREKET SINIRLAMA YÖNTEMİ

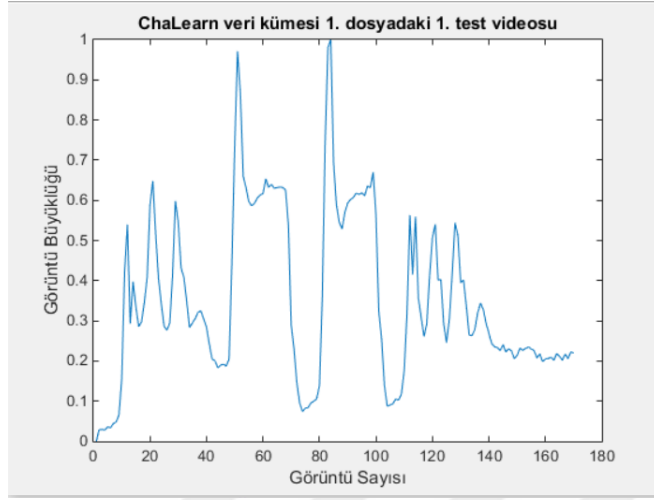
ChaLearn veri kümesi 9 kategoriden oluşmaktadır. Bu kategorilerin içinde çeşitli el ve kol hareketleri bulunmaktadır. Veri kümesi içinde hem RGB hem de derinlik videoları bulunmaktadır. Çalışmada veri kümesinde bulunan derinlik videoları kullanılmıştır. Veri kümesi, her birinde 8 ile 12 eğitim videosu ve 35 ile 39 test videosu bulunan 20 dosyadan oluşmaktadır. Eğitim veri kümesindeki her videoda yalnızca bir hareket bulunurken, test veri kümesindeki her videoda 1 ile 5 arasında hareket bulunmaktadır. Bu nedenle hareketleri sınıflandırmak için öncelikle test videolarında bulunan farklı hareketleri birbirinden ayırt etmek gerekmektedir. ChaLearn veri kümesindeki her hareket mutlaka başlangıç pozisyonuna geri dönmektedir. Bu durum, test videolarındaki farklı hareketleri ayırmayı kolaylaştırmaktadır. Eş. 2.10 kullanılarak bir test videosundaki hareketleri ayırma işlemi gerçekleştirilmektedir [32].

$$\begin{aligned} P_1 &= (I_{11}(t) - I_{11}(1))^2 + (I_{12}(t) - I_{12}(1))^2 + (I_{13}(t) - I_{13}(1))^2 + \dots + (I_{1n}(t) - I_{1n}(1))^2. \\ P_2 &= (I_{21}(t) - I_{21}(1))^2 + (I_{22}(t) - I_{22}(1))^2 + (I_{23}(t) - I_{23}(1))^2 + \dots + (I_{2n}(t) - I_{2n}(1))^2. \\ P_3 &= (I_{31}(t) - I_{31}(1))^2 + (I_{32}(t) - I_{32}(1))^2 + (I_{33}(t) - I_{33}(1))^2 + \dots + (I_{3n}(t) - I_{3n}(1))^2. \\ &\dots \dots \dots \\ P_m &= (I_{m1}(t) - I_{m1}(1))^2 + (I_{m2}(t) - I_{m2}(1))^2 + (I_{m3}(t) - I_{m3}(1))^2 + \dots + (I_{mn}(t) - I_{mn}(1))^2. \end{aligned} \quad (2.10)$$

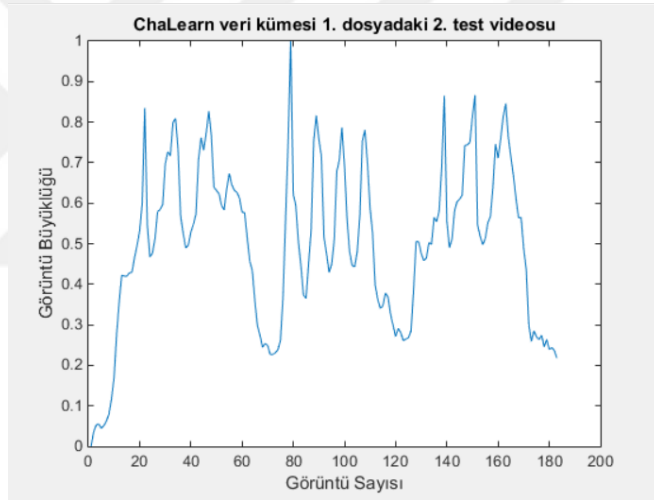
Burada $I_{xy}(t)$ $t = 1, 2, 3, \dots, T$, bir videoyu oluşturan görüntülerin her birinin t zamanındaki matrisidir. Videoyu içindeki her görüntü ilk görüntüden çıkarılmış, karesi alınmış ve toplanmıştır [32]. Eş. 2.11 ile bir videodaki ayırma işlemi yapıldıktan sonraki satır işlemi hesaplanmaktadır.

$$P_s = \sum_{k=1}^m P_k \quad (2.11)$$

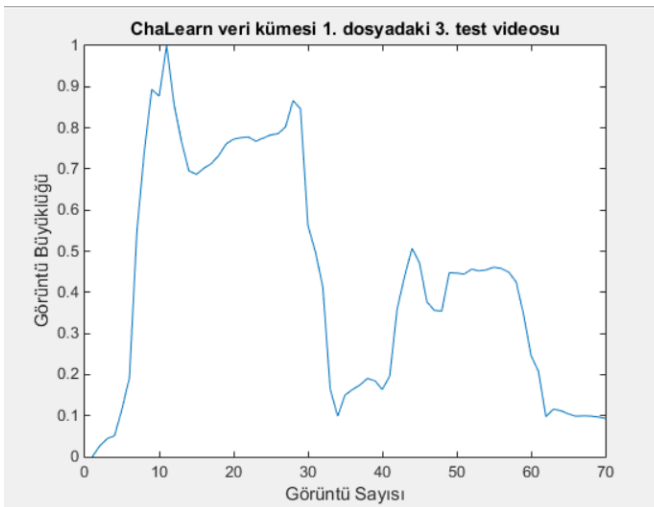
Burada matrislerin satırları toplanarak görüntü elde edilmektedir [32]. Yukarıda belirtilen formüller teorikte kalmış, pratikte başarılı olamamıştır. Şekil 2.7'de veri kümesindeki 1. dosyada bulunan 1., 2. ve 3. test videosunun grafiği verilmiştir.



(a)



(b)



(c)

Şekil 2.7. ChaLearn veri kümesinde 1. dosyadaki 1., 2. ve 3. test videolarının grafikleri.

Şekil 2.7 incelendiğinde grafiklerdeki satır bir videodaki görüntü sayısını, sütun ise bir videodaki hareketlerin büyüklüğünü ifade etmektedir. Ayrıca test videolarının Şekil 2.7.(a)'da 4, Şekil 2.7.(b)'de 3 ve Şekil 2.7.(c)'de ise 2 hareket olduğu görülmüştür. Şekil 2.7.(a)'daki grafikte birinci ve dördüncü hareket ile ikinci ve üçüncü hareketin aynı olduğu, Şekil 2.7.(b)'deki grafikte birinci ve üçüncü hareketin aynı olduğu, Şekil 2.7.(c)'deki grafikte ise varolan iki hareketinde birbirinden tamamen farklı olduğu gözlemlenmiştir. Bu hareketlerin doğru bir şekilde ayrılması gerekmektedir. Çalışmada hareketleri ayırmak için pratikte başarılı olan bir yöntem geliştirilmiştir. İlk olarak test videosu görüntülere ayrılmıştır. Eşit 2.10'de bütün görüntülerin ilk görüntüden çıkartılmasının yeterli olduğu belirtilmiştir. Fakat video görüntülere ayrıldıktan sonra görüntüler incelendiğinde ilk 10 ile 15 görüntüde kişinin hareket etmediği görülmüştür. Bu nedenle yalnızca ilk görüntünün çıkartılması yeterli olmayacaktır. İkinci olarak grafikteki yerel minimum noktaların belirtilmesi gerekmektedir. ChaLearn veri kümesindeki her hareket başlangıç pozisyonuna geri dönmektedir. Bu sayede her yerel minimum nokta bir hareketin bittiğini ifade etmektedir. Son olarak bu minimum noktalarda hareketlerin birbirinden ayrılması gerekmektedir. Bir test videosundaki birden fazla hareketin birbirinden ayrılması için yapılması gereken işlemler sırası ile aşağıda belirtilmiştir;

1. Bir video, görüntülere ayrıldıktan sonra Eşit. 2.11'deki işlemler görüntülere uygulanmaktadır.
2. Görüntülerin minimum ve maksimum değerleri hesaplanmaktadır.
3. Görüntülere normalizasyon işlemi uygulanmaktadır.
4. İlk 11 görüntüde herhangi bir hareket olmadığı için herhangi bir işlem uygulanmamaktadır.
5. Kalan görüntüler, kendinden bir önceki ve bir sonraki görüntü ile karşılaştırılmalı ve minimum değer yakalandığında hareket tamamlanmalıdır.
6. Ancak her minimum noktada hareketin tamamlandığı anlamına gelmemektedir. Bu nedenle belirli bir eşik değerinin altındaki minimum değerlere göre hareketler birbirinden ayrılmaktadır.

2.4. VERİ KÜMELERİ

2.4.1. ChaLearn Veri Kümesi

Son yıllarda algoritmaları geliştirmek, eğitmek ve değerlendirmek için halka açık birçok veri kümesi oluşturulmuştur. Ancak hareket tanıma verileri, algoritmaların geliştirilmesi için kendi özel veri kümelerine ihtiyaç duymaktadır. El hareketi tanıma sistemlerinin doğru sonuçlar verebilmesi için, çeşitli araştırma grupları tarafından birçok veri kümesi oluşturulmuştur. Ancak bu veri kümeleri, elin statik duruşları veya dinamik hareketleri için sınırlı kalmaktadır. Mevcut veri kümeleri içinde bulunan kısıtlı hareketler, kontrollü aydınlatma, tutarlı arka plan ve vücut hareketinin takibini kolaylaştırmak için renkli eldivenler kullanarak tek elle elde edilen hareketlerden oluşmaktadır. Bu nedenle bu veri kümeleri çeşitli hareketleri tanımada yetersiz kalmaktadır. Multimedya ara yüzleriyle ilgili daha anlamlı el hareketleri içeren veri kümeleri bulunmaktadır. Ancak bu veri kümelerinin bir kısmı halka açık olmamaktadır [41,42,43,44].

Gelişen teknolojiler ve mevcut veri kümelerinin sınırlamaları, daha sonra geliştirilen veri kümelerinin nasıl ve ne amaçla toplanması gerektiği konusunda aydınlatıcı olmaktadır. Günümüzde yaygın olarak kullanılan birçok hareket tanıma veri kümesi bulunmaktadır. Çizelge 2.1, günümüzde bulunan 10 hareket veri kümesine genel bir bakış sunmaktadır [41,42,43,44,45,46,47,48,49,50,51,52]. Çizelgede veri kümelerinin adı veya kısaltması ve referans bilgileri verilmektedir. Yıl, veri kümesinin hangi tarihte ortaya çıktığını belirtmektedir. Sensör, sensörün çevreye mi yoksa kullanıcıya mı yerleştirildiğini göstermektedir. Çoğu veri kümesi, ortamda sabit konumda bulunan bir video kameradan alınan görüntülerden oluşmaktadır. Sadece birkaç veri kümesi, çevre ve giyilebilir sensörlerin birleşiminden oluşan alternatif seçenekler kullanmaktadır. İki veri kümesi çevresel ve giyilebilir sensörlerden elde edilen verilerden oluşmaktadır. ChAirGest veri kümesi, ortamda sabitlenmiş bir RGB kamera ile kullanıcının kolunda bulunan bir cihaz sayesinde elde edilen verilerden oluşmaktadır [44]. 6DMG veri kümesi, hem elin hareketini hem de alandaki konumunu elde etmek için el tipi kontrol cihazı ve optik izleyici

kullanılmaktadır. Bu tür seçenekler, ortak malzeme üzerindeki her iki yaklaşımın karşılaştırılmasını ve birleştirilmesini sağlamaktadır [45]. Birçok veri kümesi içinde birden fazla hareket türü bulunmasına rağmen bazı veri kümeleri yalnızca bir hareket türüne odaklanmıştır. 3DIG veri kümesi, sadece ikonik hareketlerden oluşmaktadır. Ayrıca kullanıcılar bir nesneyi anlatmak için istedikleri hareketleri yapmakta özgür olmaktadır. Bu veri kümesinde sınıflandırma yapılmasındaki amaç kullanıcının tasvir ettiği nesneyi tanımaktır [42]. Tanım, veri kümelerinin genel olarak hangi hareketlerden oluştuğunu ifade etmektedir. Erişim durumu ise veri kümelerinin halkın erişimine açık olup olmadığını belirtmektedir. Veri kümeleri arasında büyük farklılıklar gözlemlenmektedir. El hareket türlerine göre veri kümeleri ayırt edilebilmektedir. Veri kümeleri, işaret dili, kültürel işaretler veya askeri hareketlerden oluşmaktadır. Bazı veri kümeleri ise oyun uygulamaları veya insan-bilgisayar etkileşimi gibi belirli alanlarda kullanılan hareketlerden oluşmaktadır. Bu hareket türleri genellikle ikonik, sembolik veya mecazi hareketlere dayanmaktadır [42,47,51].

ChaLearn veri kümesi tek adımda öğrenme yaklaşımı için kullanılan bir veri kümesidir. Bu veri kümesi çeşitli hareket türlerine erişim sağlayarak halka açık diğer veri kümelerini birçok bakımdan tamamlamaktadır. Veri kümesi tüm vücut hareketleri, pozlar veya faaliyetler içeren veri kümelerinin aksine, bir kameraya bakan kullanıcıların üst vücut hareketlerine odaklanmaktadır [41,51,52]. Ayrıca yine diğer veri kümelerinden farklı olarak, göstergebilimsel, dilbilimsel, sosyolojik, psikolojik ve estetik açıdan araştırma yapma imkanı sunmaktadır. Veri kümesi içinde 54.000'den fazla hareket bulunmaktadır. Böylece, eller ve kollar üzerinde eşit görülmemiş bir veri kümesi sunmaktadır [51].

ChaLearn veri kümesi oluşturulurken öncelikle hareketin türünden ziyade hareketin işlevini kontrol eden Kendon sürekliliğinden esinlenilmiştir. Kendon sürekliliği, beş hareket kategorisini dikkate alarak hareket alanını tek boyutlu olarak temsil etmektedir. Bu hareket kategorileri sırası ile; (i) hareketlerle anlatma, (ii) dil hareketleri, (iii) pandomim, (iv) amblemler, (v) işaret dilinden oluşmaktadır. Bu psikodilsel sınıflandırma, hareketler gerçekleşirken artan seviye bilincini ve soyutlamayı temsil etmektedir. Ancak ChaLearn veri kümesinde bilinçli bir şekilde

konuşma, herhangi bir faaliyet kalıbı olmadan bilinçaltı tarafından gerçekleştirilen beden dili hareketleri, danslarla veya diğer sanatsal performanslarla yapılan hareketler olmak üzere çeşitli hareket kategorileri de bulunmaktadır.

Çizelge 2.1. Hareket tanıma veri kümeleri (Erişim için gösterim: Genel veya Genel Değil).

İsim	Yıl	Referans	Sensör	Tanım	Erişim
3DIG	'13	1	Çevre	3B ikonik hareketler	G
ASL	'13	-	Çevre	Amerikan işaret dili har.	GD
CGD13	'13	2	Çevre	Konuşmaya eşlik eden kültürel hareketler	G
ChAirGest	'13	1	Çevre/ Giyilebilir	İnsan bilgisayar etkileşim hareketleri	GD
SKIG	'13	5	Çevre	Farklı aydınlatmalı RGB verilerinden hareketler	G
6DMG	'12	2	Çevre/ Giyilebilir	Konum, yönlendirme, ivme ve açısal hız dahil olmak üzere kapsamlı hareketler	G
G3D	'12	7	Çevre	Oyun hareketleri	GD
MSRGesture3D	'12	17	Çevre	El derinlik verilerinden işaret dili hareketleri	G
CGD11	'11	17	Çevre	Geniş kapsamlı hareketler	G
ASLLVD	'08	17	Çevre	İşaret dili hareketlerini farklı açılardan bulunduran veri kümesi	GD

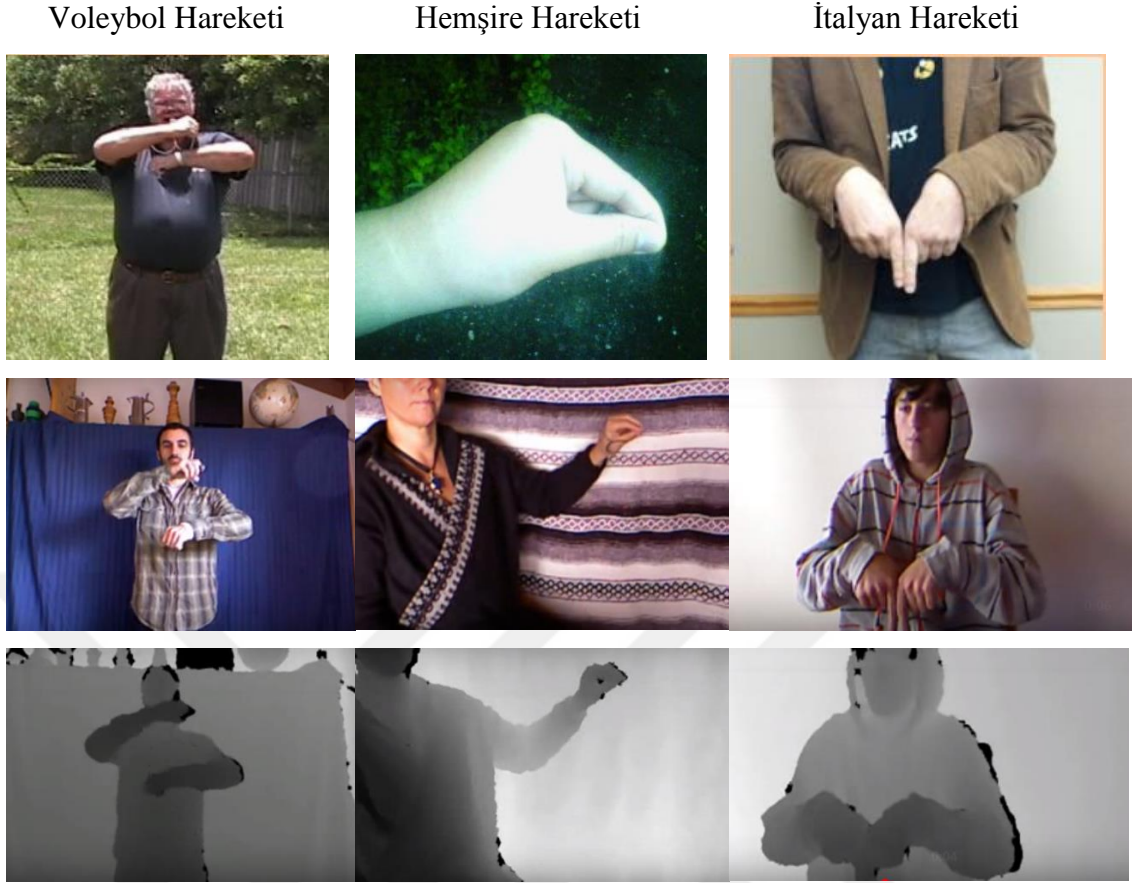
Bu nedenle Kendon sürekliliği tam olarak veri kümesini temsil edememektedir. Ayrıca Kendon sürekliliğinde hareketi tek boyutlu olarak temsil ederek hareket yönünün karıştırılmasına neden olmaktadır. ChaLearn veri kümesi içinde zorunlu konuşma varlığının azalması, dil özelliklerinin varlığının artırılması ve çeşitli hareketlerin bulunması veri kümesinin çok boyutlu bir yapıya gelmesini sağlamaktadır [41,51,52]. Bu boyutlar ifade, iletişim ve eylem olmak üzere üç gruba

ayrılmaktadır. ChaLearn veri kümesi 9 kategoriden oluşmaktadır. Bu kategoriler; 1) etkinlik, 2) pandomim, 3) dans, 4) beden dili, 5) hareketlerle anlatma, 6) ikonik hareketler, 7) amblemler, 8) işaret dili, 9) sinyaller'dir [51,52]. Çizelge 2.2'de ChaLearn veri kümesi içindeki kategoriler örnekler ile gösterilmiştir. İfade boyutunda üç hareket kategorisi bulunmaktadır. Sinyal hareketleri, kullanıcı tarafından bilinçli olarak gerçekleştirilip algılanmaktadır. Bu hareket türü aktif hareket türü olarak adlandırılmaktadır. Beden dili hareketlerini kullanıcı bilinçli olarak gerçekleştirilmemektedir. Son olarak, insanları ve nesnelere işaret eden ve ritmik hareketlerin de dahil olduğu hareketlerle anlatma adı verilen konuşmayı vurgulamak için kullanılan yarı bilinçli hareketler bulunmaktadır. İletişim boyutunda sanatsal olarak dans hareketleri ve dilsel olarak işaret dili hareketleri bulunmaktadır. Ayrıca bu boyutta hem sanatsal hem de dilsel hareketleri içeren amblemler de bulunmaktadır. Eylem boyutunda etkinlik hareketleri, ikonik hareketler ve pandomim hareketleri yer almaktadır [41,51,52].

ChaLearn veri kümesindeki veriler, kızıl ötesi sensör kullanan bir Microsoft Kinect™ kamera ile kaydedilmektedir. Kameranın mekansal çözünürlüğü, yüz ifadelerini deşifre etmek veya parmak pozlarını algılamak için kullanılan derinlik bilgilerini sınırlamaktadır. Ancak RGB modeli yeterli çözünürlüğü sağlamaktadır. Bu nedenle, veri kümesi hem normal video kümesine hem de 3B video kümesine sahiptir [51,52]. Şekil 2.8'de veri kümesindeki bazı hareketlerin RGB ve derinlik görüntüleri gösterilmektedir. Şekilde ilk satır hareket türlerini, ikinci satır hareketleri, üçüncü satır ChaLearn veri kümesi içindeki RGB görüntülerini ve son satır ise veri kümesi içindeki derinlik görüntülerini ifade etmektedir. Veri kümesi her bir dosya içinde 47 RGB videosu ve 47 derinlik videosu bulunan 20 dosyadan oluşmaktadır. RGB ve derinlik videolarının 8-12 videosu eğitim videosu ve geri kalan videolar ise test videolarından oluşmaktadır. Test videoları, eğitim videolarından rastgele seçilen 1-5 el hareketlerinin birleşmesiyle meydana gelmektedir [51,52].

Çizelge 2.2. ChaLearn veri kümesindeki kategoriler.

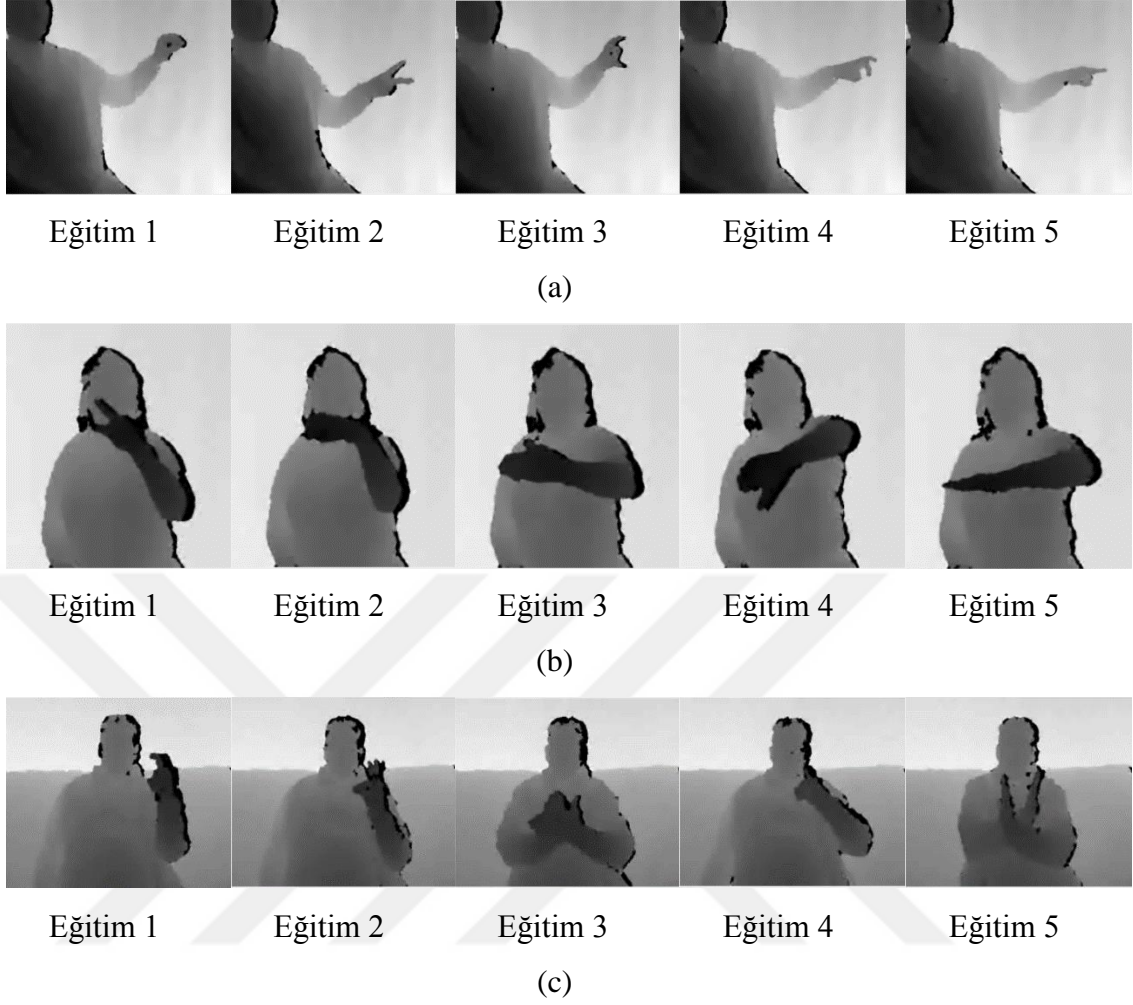
Kategoriler	Örnekler
Etkinlik	Yazı Yazmak, Telefona Cevap Vermek 
Pantomim	İp Çekmek, Merdiven Çıkmak 
Dans	Tango, Bale, Cha Cha 
Beden Dili	Baskınlık, Yalan Söyleme, Endişe 
Hareketler ile Anlatma	Konuşmayı Vurgulamak için Yapılan El Hareketleri 
İkonik Hareketler	İtalyan Hareketleri 
Amblemler	Siyasi ve Politik Hareketler 
İşaret Dili	İşitme Engelli İnsanlar için Hareketler 
Sinyaller	Dalış Hareketleri 



Şekil 2.8. ChaLearn veri kümesinden RGB ve derinlik hareket görüntüleri.

2.4.2. Karışık Veri Kümesi

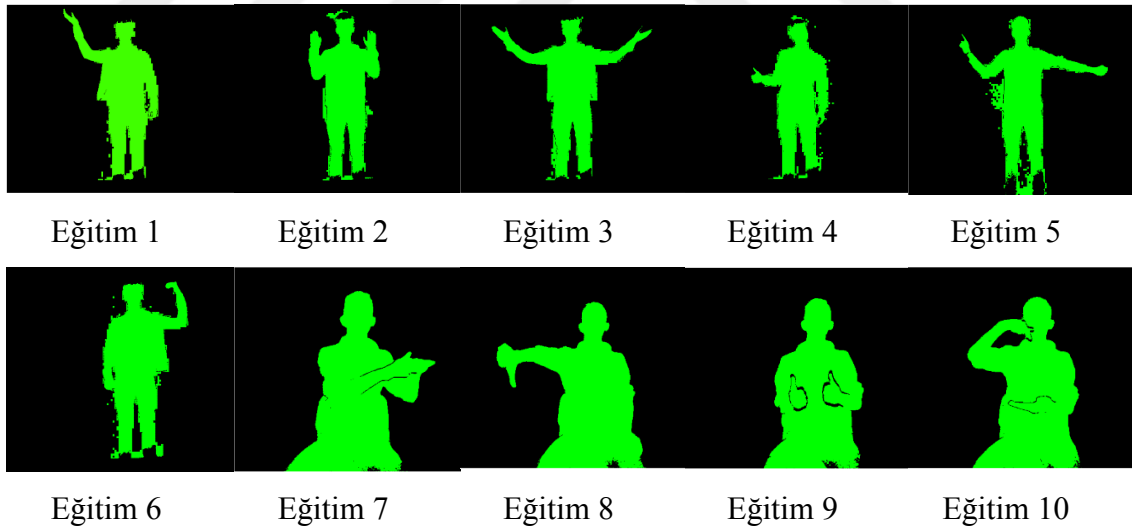
Çalışma içinde iki farklı veri kümesi kullanılmıştır. Bunlardan ilki ChaLearn veri kümesi, ikincisi ise bizim tarafımızdan oluşturulan karışık veri kümesidir. ChaLearn veri kümesi, verileri kaydederken Kinect versiyon1 (v1) kamera kullanarak yalnızca insan vücudunun üst kısmını kaydetmektedir [51]. Veri kümesi, hem RGB hem de derinlik videoları içeren 20 dosyadan oluşmaktadır. Her dosyada 8 ile 12 eğitim videosu ve 35 ile 39 test videosu bulunmaktadır. Çalışma içinde veri kümesinin derinlik videoları kullanılmıştır. Şekil 2.9'da ChaLearn veri kümesi içinde bulunan 10., 11. ve 19. dosyalardaki 5 eğitim görüntüsü gösterilmektedir.



Şekil 2.9. ChaLearn veri kümesinde eğitim videolarından alınan görüntüler.

Şekil 2.9’da, ChaLearn veri kümesinde bulunan sırasıyla 10., 11. ve 19. dosyadaki eğitim videolarının 5 tane eğitim görüntüsü gösterilmektedir. Dosya 10’da cerrah ile hemşirelerin kendi aralarında kullandıkları el hareketleri yer almaktadır. Şekil 2.9.(a)’da bulunan hareketler sırasıyla; “yara ağzını açık tutan alet”, “düz makas”, “şırınga”, “kavisli makas”, “cerrh bıçağı” ifadelerini temsil etmektedir. Dosya 11 içinde müzik nota hareketleri yer almaktadır. Şekil 2.9.(b)’deki hareketler sırasıyla; “re”, “do”, “sol”, “fa”, “mi” ifadelerine karşılık gelmektedir. Dosya 19 ise çeşitli kültürlerdeki el hareketlerine sahip olmaktadır. Şekil 2.9. (c)’de bulunan hareketler sırasıyla; “horoz”, “bir aslan yüzü”, “döner disk”, “kuğu kanadı”, “hayırlı işareti” olmaktadır [51,52].

Karışık veri kümesinde Kinect versiyon 2 (v2) kamera kullanılmıştır. Bu veri kümesinde çeşitli türlerde hareketler bulunmaktadır. Kamera ile kişiden veriler alınırken bir arayüz kullanılmıştır. Bu arayüz C# programlama dilinde yazılmış olup, kullanıcıdan alınan hareket videosunu veri kümesine görüntü olarak kaydetmektedir. Şekil 2.10'da veri kümesinde bulunan eğitim görüntüleri gösterilmektedir. Veri kümesi içinde birden fazla hareket bulunduran test verilerindeki görüntüler kişinin başlangıç pozisyonuna dönmesi ile birbirinden ayrılmaktadır. Test veri kümesinde öncelikle bir video için kişiden alınan hareketler bir dosyaya kaydedilmektedir. Test verilerinde alınan hareketin bir sınırlaması bulunmamaktadır. Kişi bir video içinde istediği kadar hareketi yapabilmektedir. Fakat kişi her hareket sonunda mutlaka başlangıç pozisyonuna geri dönmektedir. Aksi halde hareketler birbirinden ayrılırken sorun oluşmaktadır. Daha sonra dosyaya kaydedilen görüntüler tek tek incelenmektedir. Kişinin başlangıç pozisyonu tekrar yakalandığında arada görüntüler bir hareket olarak kabul edilmektedir. Son olarak bu görüntüler SS ve HGG yöntemleri uygulanarak tek bir hareket görüntüsü haline getirilmektedir.



Şekil 2.10. Karışık veri kümesinden alınan eğitim görüntüleri.

İki veri kümesinde de Kinect kameranın farklı versiyonları kullanılmıştır. Bu iki kameranın farklılıkları aşağıdaki şekilde sıralanmaktadır [53];

1. Kinect v2, v1 kameraya göre daha yüksek bir çözünürlüğe sahiptir.

2. Kinect v2, v1'e göre yüz tanımak ve hareket izlemek için daha hassas bir yapıya sahiptir.
3. Kinect v2, belirli nesnelerin özelliklerini ve hareketlerini belirlemek için "uçuş süresi" teknolojisini kullanmaktadır.
4. Uçuş süresi teknolojisi v2 kameraya, nesnelere, iyi aydınlatılmış bir odada ne kadar net görüyorsa tamamen karanlık bir ortamda da nesnelere o kadar net görebilme imkanı sağlamaktadır.
5. Kinect v1, 480 Hd çözünürlüğe sahipken, v2 1080 Hd çözünürlüğe sahiptir.
6. Kinect v2, saniyede 2gb'lık veri işleyebilmektedir. Kamera içinde bulunan usb 3.0 özelliği veri aktarımı için 10 kat daha hızlı geniş bant ve %60 daha geniş bir görüş alanı sağlamaktadır.
7. Kinect v2 ile insan baş parmakları dahil olmak üzere 6 insandan 20 eklem tespit edilip izlenebilmektedir. Buna karşılık v1 kamera ile yalnızca 2 kişiden 20 eklem izlenebilmektedir. Ek olarak v2 kullanılarak çok önemli biyometrik verilerle birlikte kalp atış hızları, yüz ifadeleri ve uzuvlardaki ağırlıklar tespit edilebilmektedir.
8. Kinect v1, parmakları ayrı ayrı izlemeye uygun değilken, v2 ile parmak hareketleri izlenebilmektedir.

Karışık veri kümesi iki büyük avantaj sağlamaktadır. İlk avantajı, ChaLearn veri kümesi ile yapılan çalışmada doğrudan video kullanılmaktadır. Bu nedenle, video içinde bulunan bazı gereksiz görüntülerde çalışmaya dahil olup yanlış sınıflandırma yapılmasına neden olmaktadır. Karışık veri kümesi, doğrudan görüntüleri kullanılmaktadır. Bu sayede gereksiz görüntülerin olmadığı daha doğru bir sınıflandırma yapılabilmektedir. İkinci avantajı ise, karışık veri kümesinin verileri görüntülerden oluştuğu için test verilerindeki hareketleri birbirinden ayırmak için herhangi bir işlem yapmamasıdır. Bu sayede hareket karşılaştırılması yapılırken hata oranını daha düşük seviyede olmaktadır. Ek olarak, karışık veri kümesinde eğitim verileri oluşturulurken bir hareketin 4 defa tekrarı alınmıştır. Böylece, daha iyi bir eğitim sağlanarak kamera kaymasından kaynaklanabilecek yanlış sınıflandırma engellenmiştir.

2.5. MESAFE ÖLÇÜMLERİ

Çalışma içinde, eğitim verileri ile test verilerini karşılaştırmak için 3 farklı mesafe ölçümü kullanılmıştır. Bu mesafe ölçümleri, Levenshtein, Frobenius ve Mahalanobis'tir. Ayrıca bu mesafe ölçümlerinden farklı olarak hareketleri karşılaştırmak için çapraz korelasyon da kullanılmıştır.

Korelasyon, ölçülen iki miktar arasında doğrusal bir ilişkinin mevcut olma olasılığını belirleme yöntemi olarak kullanılmaktadır. İlk olarak Karl Pearson, Pearson ürün moment korelasyon katsayısını (r) keşfetmiştir. Bu korelasyon katsayısı, ilk resmi korelasyon katsayısı olarak bilinmektedir. Gelişen teknoloji ile birlikte katsayı, istatistiksel analiz, örüntü tanıma ve görüntü işleme alanlarında yaygın olarak kullanılmaktadır [54,55].

Çapraz korelasyon ile hareketlerin karşılaştırılmasındaki amaç, gri tonlamalı görüntüler arasındaki hesaplanan korelasyon katsayılarının en büyüğü ile iki görüntü arasındaki benzerliği bulmaktır. Korelasyon iki görüntü arasında gerçekleşmektedir. Bu görüntülerden ilki eğitim ikincisi test görüntüsünden oluşmaktadır. İki görüntü arasında korelasyon hesaplanırken, test görüntüsü piksel piksel kaydırılıp, her konumda eğitim ile test görüntüsü arasındaki korelasyon katsayısı hesaplanmaktadır. En büyük korelasyon katsayısı, eğitim ve test görüntüsü arasındaki en iyi eşleşen bölgeyi göstermektedir [54,55]. Eş. 2.12 kullanılarak iki görüntü arasındaki korelasyon katsayısı hesaplanmaktadır [60].

$$r = \frac{\sum_{k=0}^{m-1} \sum_{l=0}^{n-1} (A_{kl} - \bar{A})(B_{kl} - \bar{B})}{\sqrt{\sum_{k=0}^{m-1} \sum_{l=0}^{n-1} (A_{kl} - \bar{A})^2} \sqrt{\sum_{k=0}^{m-1} \sum_{l=0}^{n-1} (B_{kl} - \bar{B})^2}} \quad -1 \leq r \leq 1 \quad (2.12)$$

Burada r korelasyon katsayısını, A_{kl} eğitim görüntüsünü, B_{kl} test görüntüsünü, \bar{A} eğitim görüntüsünün aritmetik ortalamasını, \bar{B} test görüntüsünün aritmetik ortalamasını temsil etmektedir. İki görüntü birbirine benziyorsa $r = 1$, benzerlik bulunamadıysa $r = 0$ veya iki görüntü birbirine tamamen zıt ise $r = -1$ değerine sahip olmaktadır [54,55].

Korelasyon katsayısı, çalışma içinde iki yarar sağlamaktadır. İlki, 2 boyutlu iki görüntünün tek bir sayısal değer ile karşılaştırılmasını sağlamaktadır. İkincisi ise, görüntüdeki x ve y koordinatları değişmez bir yapıya sahiptir. Bu sayede, katsayı bir görüntüdeki parlaklık veya kontrasttaki tek tip değişikliklere karşı duyarsız olmaktadır. Fakat korelasyon katsayısı, yoğun bir hesaplamaya sahip olması nedeniyle görüntü kaydı için kullanışlılığını sınırlamaktadır. Ayrıca görüntü sistemlerinde kaçınılmaz olarak meydana gelen görüntü eğiltme veya çarpma gibi etkilere karşıda hassas bir yapıya sahip olmaktadır [55].

Kelime dizelerini benzerlik açısından karşılaştırmak, benzer dizelerin eşleşmesini sağlamak için çekirdek bir yapıyı oluşturmaktadır. Düzenleme mesafesi olarak bilinen yapı, iki dizenin birbirine ne kadar benzer olduğunu ölçmek için yaygın olarak kullanılmaktadır. Benzerlik, bir dizeyi diğerine dönüştürmek için kaç işlem gerektiğiyle ölçülmektedir. Yapılacak olan işlemler, bir dizeyi başka bir dizeye dönüştürmek için bir dizedeki bir veya birden fazla karakterin yerleştirilmesini, silinmesini, aktarılmasını ve değiştirilmesini içermektedir. Verilen iki dizi arasındaki düzenleme mesafesini ölçmek için birçok mesafe ölçümü bulunmaktadır. Levenshtein mesafe ölçümü, gelişen teknoloji ile birlikte en sık kullanılan düzenleme mesafe ölçümlerinden biri olmaktadır [56,57].

Levenshtein mesafe ölçümü, iki dizi, iki kelime, iki cümle gibi varlıklar arasında ekleme, silme ve yer değiştirme gibi işlemleri gerçekleştirmektedir. Bu mesafe ölçümü, iki kelimeyi harf harf karşılaştırarak, bir kelimenin diğerine dönüştürülmesi için gerekli işlem maliyetini hesaplamaktadır [57,58]. Örneğin; kazak – kazan kelimeleri arasındaki mesafe maliyeti 1 olmaktadır. Çünkü harfleri ve harf sayıları aynı olmasına rağmen son harflerinin farklı olduğu görülmektedir. Arap – araba kelimeleri arasındaki mesafe maliyeti ise 2 olmaktadır. Çünkü p harfi b ile değişmiş ve a harfi eklenmiştir. Mesafe ölçümünün maliyetini hesaplamak için Eş. 2.13 kullanılmıştır [57,58].

$$lev_{a,b}(i,j) = \begin{cases} \max(i,j) & \min(i,j) = 0 \text{ ise} \\ \min \begin{cases} lev_{a,b}(i-1,j) + 1 \\ lev_{a,b}(i,j-1) + 1 \\ lev_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{aksi halde} \end{cases} \quad (2.13)$$

Burada $1_{(a_i \neq b_j)}$ gösterge fonksiyonudur. Fonksiyon $a_i = b_j$ ise 0'a değilse 1'e eşittir. $lev_{a,b}(i,j)$, a 'nın ilk karakteri i ile b 'nin ilk karakteri j arasındaki mesafedir. Minimumdaki birinci eleman a 'dan b 'ye silme, ikinci eleman ekleme ve üçüncü eleman da yer değiştirme işlemi yapmaktadır [57,58].

Levenshtein mesafe ölçümü, çalışmada eğitim ve test görüntülerini karşılaştırmak için kullanılmıştır. Eğitim veri kümesindeki görüntü V , test veri kümesindeki görüntü Y ise mesafe ölçümü $L(V,Y)$ şeklinde ifade edilmektedir. $L(V,Y)$ maliyeti düşük ise görüntüler birbirine benzer, değil ise görüntüler birbirinden farklı kabul edilmektedir.

Çalışma içinde Frobenius mesafe ölçümüdür ikinci mesafe ölçümü olarak kullanılmıştır. Bu mesafe ölçümü, Öklid mesafe ölçümünün bir alt sınıfıdır ve bir matristeki tüm değerlerin karelerinin toplamının karekökünü almaktadır [59]. Çalışmada eğitim veri kümesindeki görüntü ile test veri kümesindeki görüntü, bu mesafe ölçümü ile karşılaştırılmıştır. Matrislerden oluşan birbirinden tamamen farklı iki görüntünde, bu mesafe ölçümü ile yapılan işlem sonucunda aynı değeri verdiği gözlemlenmiştir. Bu durum yanlış sınıflandırma olmasına neden olmaktadır. Sonuç olarak bu mesafe ölçümünden beklenen başarı oranı elde edilememiştir. Eşit. 2.14 kullanılarak Frobenius mesafe ölçümü hesaplanmaktadır [59].

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} \quad (2.14)$$

Burada bir görüntüyü oluşturan matrisin tüm değerlerinin karelerinin toplamının karekökü alınmıştır [59].

Öklid mesafe ölçümü, uzayda iki nokta arasındaki mesafeyi ölçmek için kullanılan en temel mesafe ölçümü olarak kabul edilmektedir. Ancak bu mesafe ölçümü, mesafesi hesaplanacak iki nokta dışındaki bilgileri hesaba katmamaktadır. Ayrıca bu noktaların tanımlı olduğu eksenlerin birimlerini eşit kabul etmektedir [60,61]. Eksenlerin birimleri genel olarak birbirinden farklı değerlere sahip olmaktadır.

Mahalanobis mesafe ölçümü, noktalar arasındaki mesafe ölçümüne ek olarak kovaryans matris değerini de hesaba katmaktadır. Böylece, başarı oranı yüksek bir sınıflandırma işlemi yapılmaktadır. Eşit. 2.15 kullanılarak Mahalanobis mesafe ölçümü hesaplanmaktadır [60,61].

$$D = \sqrt{(x_i - \alpha)^T S^{-1} (x_i - \alpha)} \quad (2.15)$$

Burada x_i değer vektörüdür. Eşit. 2.16'de α ifadesi ve Eşit. 2.17'da S ifadesi hesaplanmaktadır

$$\alpha = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.16)$$

Burada α ortalama vektördür.

$$S = \frac{1}{n} \sum_{i=1}^n (x_i - \alpha_i)(y_j - \alpha_j) \quad (2.17)$$

Burada S kovaryans matrisidir. Sonuç olarak, bu mesafe ölçümü, değer vektörü ile ortalamaların farkının, kovaryans matrisinin tersi ve yine değer vektörü ile ortalamalarının farkının tersyüzünün çarpımlarının karekökü olarak hesaplanmaktadır [60,61].

Mahalanobis mesafe ölçümü, görüntü işlemede de kullanılmaktadır. Bu nedenle, çalışmada son olarak bu mesafe ölçümü kullanılmıştır. İlk olarak, eğitim görüntülerinin ortalama ve standart sapma değerleri hesaplanmaktadır. Daha sonra test görüntülerinin ortalama ve standart sapma değerleri mesafe ölçüm formülünde yerine yerleştirilmektedir. Böylece, eğitim görüntüsü ile test görüntüleri arasında Mahalanobis uzaklığı elde edilmektedir. Son olarak, Mahalanobis uzaklığı en düşük olan eğitim ve test görüntüsü birbirine benzemekte, aksi halde görüntüler birbirinden farklı olmaktadır.

BÖLÜM 3

DENEYSEL ÇALIŞMALAR

Çalışma içinde tek adımda öğrenme yaklaşımı çeşitli yöntemler ile birlikte kullanılarak el hareketlerini sınıflandırmayı amaçlamaktadır. Bu amaç doğrultusunda hareketleri video içinden alıp tek bir görüntü haline getirmek için 3 temel yöntem kullanılmaktadır. Bu yöntemler, SS, HGG ve 2B HFD yöntemleri olmaktadır. Ayrıca hareketleri sınıflandırmak için Levenshtein, Frobenius ve Mahalanobis olmak üzere 3 farklı mesafe ölçümü ve çapraz korelasyon kullanılmaktadır.

Çalışmada SS yöntemi kullanılarak veri kümesindeki her video, bir görüntü haline getirilmektedir. Ancak SS yöntemi incelendiğinde hareketin geçmiş bilgisini ve yön bilgisini koruyamadığı gözlemlenmektedir. Bu durum hatalı sınıflandırmaların olmasına neden olmaktadır. Hatalı sınıflandırmayı önlemek için çalışmada HGG yöntemi kullanılmaktadır. Yöntem, tek bir görüntüde kodlanmış zaman aralığını temsil etmektedir. Bu sayede, insan hareketlerinin zaman ölçeğini kapsamaktadır. Her pikselin yoğunluğunu geçici bir şekilde kullanarak hareket akış yönünü ifade etmektedir. Ayrıca yöntem, genel hareket kalıplarını tanımaktadır. Böylece hareketin kolayca tespit edilemeyeceği düşük ışıklı alanlarda ucuz kameralarla veya düşük güçlü CPU'larla uygulanabilmektedir. Yöntemin başka bir avantajı, hareket yönüne karşı duyarlı olmaktadır. Bu sayede, elin sağdan sola veya soldan sağa hareketi gibi zıt yönlü hareketleri kolayca ayırt edebilmektedir. Son olarak 2B HFD yöntemi de kamera kaymalarından oluşabilecek hatalı sınıflandırmayı engellemek için çalışma içinde kullanılmaktadır. Yöntem çalışma içinde tek başına kullanılmamıştır. SS yönteminden önce ve HGG yönteminden sonra kullanılmıştır. 2B HFD yönteminin, SS yönteminden önce ve HGG yönteminden sonra kullanılması yanlış sınıflandırma olmasını engellemiştir. Bununla birlikte, arka plan çıkarılmış derinlik görüntülerinin SS yönteminden önce spektral alana dönüştürülmesi, zaman çözünürlüğünü baskılayacak

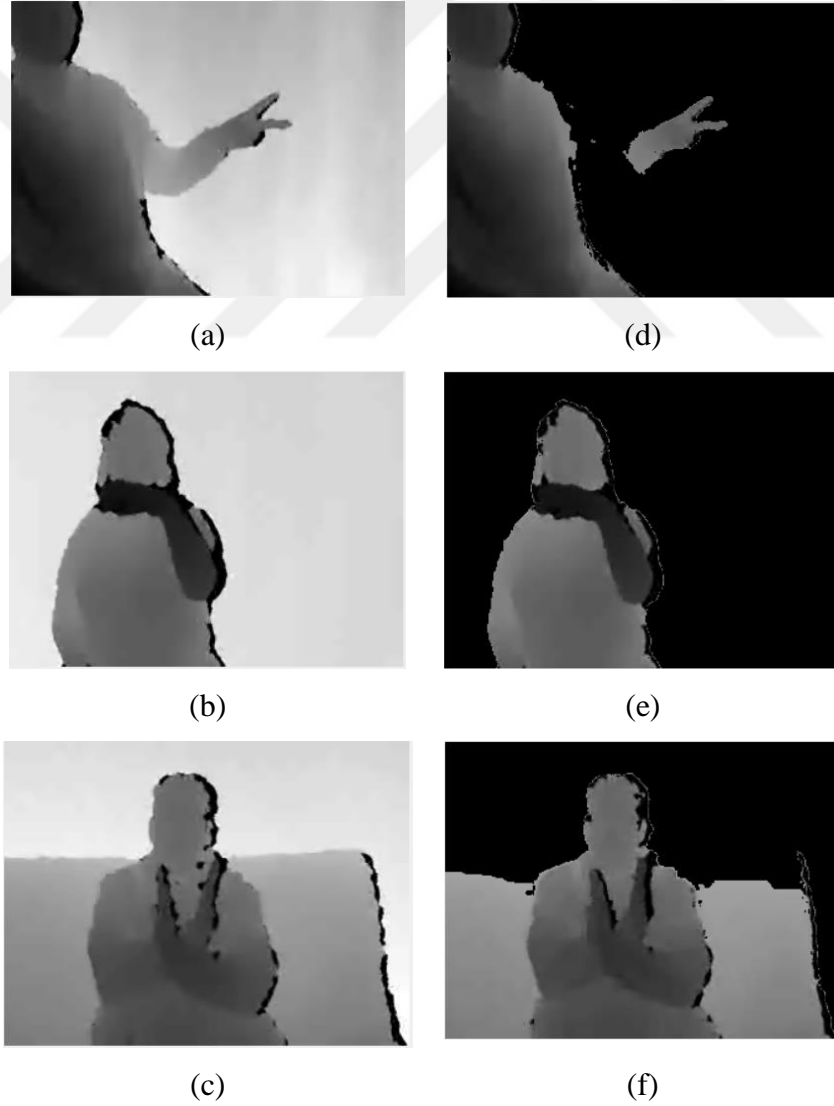
ve böylece kamera hareketinin etkisini azaltacaktır. Bu sayede eşleştirme için eğitim ve test verilerini benzer konuma getirecektir.

Çalışma içinde ChaLearn ve bizim oluşturduğumuz karışık veri kümesi olmak üzere iki farklı veri kümesi kullanılmıştır. ChaLearn veri kümesinde 9 farklı hareket türü bulunmaktadır. Veri kümesindeki veriler Kinect v1 kamera kullanılarak kaydedilmiş 20 dosyadan oluşmaktadır. Her dosya içinde 47 RGB ve 47 derinlik videoları bulunmaktadır. Hem RGB hem de derinlik videoları 8 ile 12 tane eğitim videosu ve 35 ile 39 tane test videosundan oluşmaktadır. Çalışma içinde doğrudan derinlik videoları kullanılmıştır. Tek adımda öğrenme yaklaşımı tek bir görüntüden el hareketi sınıflandırmayı amaçlamaktadır. Bu nedenle yaklaşımı uygulayabilmek için öncelikle ChaLearn veri kümesindeki videoların görüntülere ayrılmasına ihtiyaç duyulmaktadır. Görüntüler derinlik görüntüler olduğu için öncelikle gri tonlamalı görüntülere daha sonra Otsu eşik yöntemi kullanılarak ikili görüntülere dönüştürülmektedir. Yöntem gri tonlamalı görüntüler üzerinde uygulanırken yalnızca renklerin bir görüntü içinde ne kadar bulunduğunu kontrol etmektedir. Renklerin görüntü üzerindeki sayısına bakıldığı için eşik yöntemi uygulanmadan önce görüntünün renk histogramı hesaplanıp bütün işlemler bu histogram dizisi üzerinden yapılmaktadır. Yöntemde görüntünün iki renk sınıfından oluştuğu kabul edilmektedir. Bu sınıflarda sınıf içi varyans değerinin minimum olması ve sınıflar arasındaki varyans değerinin maksimum olması sağlanarak hareket görüntüsünün arka ekrandan ayrılması gerçekleştirilmiştir. Bu durum daha hızlı ve daha doğru bir sınıflandırma işlemi yapabilmek için SS VE HGG yöntemlerine kolaylık sağlanmaktadır. Şekil 3.1’de ChaLearn veri kümesine Otsu yöntemi uygulanmış görüntüler bulunmaktadır. Şekil 3.1.(a),(b),(c)’deki görüntüler veri kümesi içindeki sırasıyla 10., 11. ve 19. Dosyadaki eğitim derinlik görüntülerini, Şekil 3.1.(d), (e), (f) ise bu derinlik görüntülerine Otsu yöntemi uygulandıktan sonraki görüntülerini göstermektedir. Şekil 3.1(a), 10. dosyada bulunan “düz makas” ‘ı, Şekil 3.1.(b) 11. dosyada bulunan “do” ‘yu ve Şekil 3.1.(c) ise 19. dosyada bulunan “hayırlı işaretini” ‘ni temsil etmektedir.

Çalışma içinde ikinci veri kümesi bizim oluşturduğumuz karışık veri kümesi olmaktadır. Bu veri kümesinde kullanıcıdan alınan hareketler Kinect v2 kamera

kullanılarak veri kümesine kaydedilmiştir. Kullanıcıdan alınan hareket videoları bir arayüz ile veri kümesine görüntü olarak aktarılmaktadır. Bu arayüz C# programlama dilinde yazılmış olup, kullanıcıdan alınan hareket videosunu veri kümesine görüntü olarak kaydetmektedir.

ChaLearn veri kümesi içindeki test videolarının her birinde 1 ile 5 arasında hareket bulunmaktadır. Hareketlerin doğru şekilde birbirinden ayrılması hatalı sınıflandırmaya neden olmaktadır. ChaLearn veri kümesindeki bütün hareketler başlangıç pozisyonuna geri dönmektedir. Bu durum, bir video içindeki farklı hareketleri birbirinden ayırt etmeyi kolaylaştırmaktadır.



Şekil 3.1. ChaLearn veri kümesine Otsu eşik yöntemi uygulanmış görüntüler.

Çalışmada eğitim videolarında tek hareket bulunurken test videolarında birden fazla hareket bulunmaktadır. Bu nedenle doğru sınıflandırma yapabilmek için ilk olarak test videolarındaki farklı hareketlerin birbirinden ayrılması gerekmektedir. Hareketleri ayırmak için çalışmada hareketleri sınırlama yöntemi kullanılmıştır. Yöntem ile öncelikle test videoları grafikler haline getirilmiştir. Daha sonra grafikler incelenip yerel minimum noktaları belirlenmiştir. Her yerel minimum noktası bir hareketin bittiğini ifade etmektedir. Bu sayede yerel minimum noktalarından görüntüler birbirinden ayrılmaktadır. Her bir görüntü kümesi bir hareketi temsil etmektedir. Karışık veri kümesinde verilerin görüntü olarak kaydedilmesi nedeniyle hareket sınırlama yöntemine ihtiyaç duyulmamıştır.

El hareketlerinin sınıflandırma işlemini yapmak için Levenshtein, Frobenius ve Mahalanobis olmak üzere üç farklı mesafe ölçümü ve çapraz korelasyon kullanılmıştır. Sınıflandırma işlemi eğitim ve test veri kümelerindeki görüntülerin karşılaştırılıp eşleşmesi ile gerçekleşmektedir. Test veri kümesindeki bir videoda bulunan hareketler birbirinden ayrıldıktan sonra o hareketi oluşturan görüntüler SS ve HGG yöntemleri ile tek bir görüntü haline getirilmektedir. Hareket görüntüsü 20 dosyada bulunan birbirinden farklı eğitim görüntüleri ile mesafe ölçümleri sayesinde karşılaştırılmaktadır. Eğitim ve test görüntüsü eşleşme sağladığında test görüntüsü o dosya içine aktarılmaktadır. Doğru eşleşme olduğu takdirde çalışmanın başarı oranı artmakta aksi takdirde çalışmanın başarı oranı düşmektedir.

Çapraz korelasyon, eğitim ile test verisi arasındaki korelasyon katsayısını hesaplamaktadır. Korelasyon, gri tonlamalı görüntüler arasındaki hesaplanan korelasyon katsayılarının en büyüğü ile iki görüntü arasındaki benzerliği bulmaktadır. Bu görüntülerin ilki eğitim ikincisi test görüntüsünden oluşmaktadır. İki görüntü arasında korelasyon hesaplanırken, test görüntüsü piksel piksel kaydırılıp, her konumda eğitim ile test görüntüsü arasındaki korelasyon katsayısı hesaplanmaktadır. Katsayı değeri ne kadar büyük ise görüntüler birbirine o kadar benzer olmaktadır. Levenshtein mesafe ölçümü, iki dizi, iki kelime, iki cümle gibi varlıklar arasındaki işlemleri gerçekleştirmektedir. Bu mesafe ölçümü, iki kelimeyi harf harf karşılaştırarak, bir kelimenin diğerine dönüştürülmesi için gerekli işlem maliyetini

hesaplamaktadır. Gerekli işlemler, bir dizeyi başka bir dizeye dönüştürmek için bir dizedeki bir veya birden fazla karakterin yerleştirilmesini, silinmesini, aktarılmasını ve değiştirilmesini içermektedir. Mesafe ölçümü, normal hayatta kelimeleri karşılaştırmak için kullanılırken çalışmada görüntüleri karşılaştırmak için kullanılmıştır. Eğitim ile test verilerinde görüntüleri oluşturan matrislerin bütün elemanlarını tek tek karşılaştırarak benzerliliği hesaplamaktadır. Bu sayede sınıflandırma işleminde yüksek bir başarı oranına sahip olmaktadır. Frobenius mesafe ölçümü, görüntüyü oluşturan matrise yalnızca bir işlem uygulayıp matrizen tek değer almaktadır. Bu durumda birbirinden tamamen farklı iki görüntüde aynı sonucu vermektedir. Sonuç olarak bu mesafe ölçümü, hatalı sınıflandırmalara neden olarak düşük bir başarı oranına sahip olmaktadır. Mahalanobis mesafe ölçümü, iki görüntü arasındaki yalnızca uzaklığı değil o görüntülerin ortalamasını ve standart sapmasını da hesaba katmaktadır. Bu sayede, Levenshtein mesafe ölçümünden sonra ikinci en yüksek başarı oranı gösteren mesafe ölçümü olmaktadır.

Çalışmada ChaLearn veri kümesindeki videolar birbirinden ayrılıp görüntü haline getirildikten sonra 3 temel işlem olarak bilinen SS, HGG v 2B HFD yöntemleri 9 farklı şekilde çalışmaya uygulanmıştır. Bütün yöntemlerde öncelikle videolar görüntülere dönüştürülmüş ve Otsu eşik yöntemi uygulanmıştır. Yapılan 9 farklı işlem aşağıda belirtilmiştir;

1. ChaLearn veri kümesinde bulunan test verilerindeki farklı hareketler birbirinden ayrıldıktan sonra, eğitim ve test verilerine yalnızca SS yöntemi uygulanıp çapraz korelasyon ile karşılaştırılmış ve eşleştirilmiştir.
2. HGG yöntemi görüntülere karşı duyarsız olmaktadır. Ayrıca gri tonlamalı görüntülere yoğunlaşmış baskın hareket bilgisini korumaktadır. Bu sayede ChaLearn veri kümesindeki test verilerinde bulunan farklı hareketler birbirinden ayrıldıktan sonra eğitim ve test verilerine yalnızca HGG yöntemi uygulanmış ve çapraz korelasyon ile karşılaştırılıp eşleştirilmiştir.
3. Kamera kayması yanlış sınıflandırmaya neden olmaktadır. Bu nedenle ilk olarak veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrılmıştır. Daha sonra eğitim ve test verileri birbiri ile

karşılaştırılmadan önce 2B HFD uygulanmıştır. Son olarak SS yöntemi uygulanmış ve çapraz korelasyon ile karşılaştırma ve eşleştirme yapılmıştır.

4. Veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrılmış ve HGG yönteminden sonra yanlış sınıflandırmayı önlemek için 2B HFD yöntemi uygulanmıştır. Test ve eğitim verilerini karşılaştırmak için çapraz korelasyon kullanılmıştır.
5. Veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrıldıktan sonra kamera kaymasından oluşabilecek yanlış sınıflandırmayı önlemek için ilk olarak 2B HFD daha sonra ise SS yöntemi uygulanmıştır. Eğitim ve test verilerindeki görüntüleri karşılaştırmak ve sınıflandırmak içinde Levenshtein mesafe ölçümü kullanılmıştır.
6. İlk olarak veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrılmış ve HGG yönteminden sonra yanlış sınıflandırmayı önlemek için 2B HFD yöntemi uygulanmıştır. Daha sonra eğitim ve test verilerini karşılaştırmak ve eşleştirmek için Levenshtein mesafe ölçümü kullanılmıştır.
7. Çalışmada ilk olarak hareket sınırlama yöntemi ile veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrılmıştır. Daha sonra 3 temel yöntem olarak bilinen SS, HGG ve 2B HFD yöntemleri birlikte kullanılmıştır. Yöntemler kullanılırken SS yönteminden önce ve HGG yönteminden sonra 2B HFD yöntemi kullanılmıştır. SS yöntemi uygulandıktan sonra yapılan karşılaştırma sonucu oluşan hatalı eşleşme HGG yöntemi kullanılarak düzeltilmiştir. Karşılaştırma ve sınıflandırma işlemlerini yapmak içinde Levenshtein mesafe ölçümü kullanılmıştır.
8. Çalışmada hareket sınırlama yöntemi ile veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrılarak 3 temel yöntem olarak bilinen SS, HGG ve 2B HFD yöntemleri birlikte kullanılmıştır. Yöntemler kullanılırken SS yönteminden önce ve HGG yönteminden sonra 2B HFD yöntemi kullanılmıştır. SS yöntemi uygulandıktan sonra yapılan karşılaştırmada hatalı eşleşmeler meydana gelmiştir. Bu sorun HGG yöntemi kullanılarak düzeltilmiştir. Karşılaştırma ve sınıflandırma işlemlerini yapmak içinde Frobenius mesafe ölçümü kullanılmıştır.

9. Öncelikle veri kümesi içinde bulunan test verilerindeki farklı hareketler birbirinden ayrılmıştır. Daha sonra 3 temel yöntem olarak bilinen SS, HGG ve 2B HFD yöntemleri birlikte kullanılmıştır. Yöntemler kullanılırken SS yönteminden önce ve HGG yönteminden sonra 2B HFD yöntemi kullanılmıştır. SS yöntemi uygulandıktan sonra yapılan karşılaştırma sonucu oluşan hatalı eşleşme HGG yöntemi kullanılarak düzeltilmiştir. Karşılaştırma ve sınıflandırma işlemlerini yapmak için Mahalanobis mesafe ölçümü kullanılmıştır.

Çalışmada kullanılan yöntemler ve başarı oranları Çizelge 3.1’de gösterilmektedir. İlk sütun çalışma içinde kullanılan yöntemleri, ilk satır ise çalışmada belirtilen mesafe ölçümlerini göstermektedir. Çizelge içinde bulunan her değer veri kümesi içinde bulunan videoyu tek bir görüntüye çevirmek için kullanılan yöntemin, eğitim ve test görüntülerini karşılaştırmak ve eşleştirmek için kullanılan mesafe ölçümleri ile ne oranda başarı verdiğini göstermektedir.

Çizelge 3.1. ChaLearn veri kümesi ile kullanılan yöntemler ve başarı yüzdeleri.

Mesafe Ölç. / Yöntemler	Çapraz Korelasyon	Levenshtein Mesafe Ölç.	Frobenius Mesafe Ölç.	Mahalanobis Mesafe Ölç.
SS Yöntemi	50.22	40.25	20.63	33.45
HGG Yöntemi	53.69	50.22	25.66	36.78
SS Yöntemi Öncesi 2B HFD Yöntemi	51.68	55.53	34.58	40.89
HGG Yöntemi Sonrası 2B HFD Yöntemi	56.64	63.55	38.67	50.32
SS Yön. Öncesi 2B HFD Yön. Ve HGG Yön. Sonrası 2B HFD Yöntemi	55.61	76.15	41.7	52.2

Çalışmada 3 temel yöntem olarak adlandırdığımız SS, HGG ve 2B FFT yöntem ile videolar tek bir görüntü haline getirilmiştir. Daha sonra çapraz korelasyon ve

Levenshtein, Frobenius ve Mahalanobis mesafe ölçümleri ile eğitim ve test verileri karşılaştırılmış ve eşleşen görüntüler aynı dosya içine yerleştirilmiştir. Çalışma yapılırken bu yöntemler ve mesafe ölçümleri ile birçok işlem gerçekleştirilmiştir. Çizelge incelendiğinde 7. işlem türünün diğer işlemlerle karşılaştırıldığında daha yüksek bir başarı oranına sahip olduğu görülmüştür. 7. işlem SS, HGG ve 2B HFD yöntemlerini bir arada kullanmıştır. SS yöntemi hareket geçmiş bilgisini ve yön bilgisini kaybetmesi nedeniyle yanlış eşleşmelere neden olmaktadır. HGG yöntemi bu işlemde kullanılarak sorunu çözmektedir. Sonuç olarak en doğru sınıflandırma 7. işlem ile yapılmıştır. Hareket geçmiş bilgisi ve yön bilgisini koruduğu için ikinci en yüksek başarı oranına 6. işlem sahip olmaktadır.

Karışık veri kümesinde SS, HGG ve 2B HFD yöntemleri 9 farklı işlem olarak yine çalışmada uygulanmıştır. Fakat veri kümesinde kamera ile kişiden hareket verileri alınırken bir arayüz kullanılmıştır. Bu arayüz C# programlama dilinde yazılmış olup, kullanıcıdan alınan hareket videosunu veri kümesine görüntü olarak kaydetmektedir. Bu nedenle veri kümesine uygulanan 9 farklı işlemin hiçbirinde hareket sınırlama yöntemi kullanılmamıştır. Çalışmada işlemler uygulanmadan önce hepsinde Otsu eşik yöntemi uygulanmıştır. Bu sayede ön planda bulunan hareket görüntüsü arka plandan ayrılmıştır. Veri kümesi içinde birden fazla hareket bulunduran test verilerindeki görüntüler kişinin başlangıç pozisyonuna dönmesi ile birbirinden ayrılmaktadır. Test veri kümesinde öncelikle bir video için kişiden alınan hareketler bir dosyaya kaydedilmektedir. Test verilerinde alınan hareketin bir sınırlaması bulunmamaktadır. Daha sonra dosyaya kaydedilen görüntüler tek tek incelenmektedir. Kişinin başlangıç pozisyonu tekrar yakalanana kadar olan görüntüler bir hareket olarak kabul edilmektedir. Son olarak bu görüntüler SS ve HGG yöntemleri ile tek bir görüntü haline getirilmektedir.

Çizelge 3.2, karışık veri kümesiyle kullanılan işlemleri ve başarı oranlarını göstermektedir. İlk sütun çalışma içinde kullanılan yöntemleri, ilk satır ise çalışmada belirtilen mesafe ölçümlerini göstermektedir. Çizelge içinde bulunan her değer veri kümesi içinde bulunan videoyu tek bir görüntüye çevirmek için kullanılan yöntemin, eğitim ve test görüntülerini karşılaştırmak ve eşleştirmek için kullanılan mesafe ölçümleri ile ne oranda başarı verdiğini göstermektedir. Çalışma yapılırken yine bu

yöntemler ve mesafe ölçümleri ile birçok işlem gerçekleştirilmiştir. Karışık veri kümesi ile yapılan çalışmada 7. işlemin daha yüksek bir başarı oranına sahip olduğu gözlemlenmiştir. Ayrıca, karışık veri kümesi ile yapılan işlemlerde başarı oranlarının ChaLearn veri kümesindeki başarı oranlarından daha yüksek olduğu görülmektedir. ChaLearn veri kümesi ile yapılan çalışmada doğrudan video kullanılmaktadır. Bu durum, gereksiz görüntülerin çalışma içine dahil edilmesine ve yanlış sınıflandırma olmasına neden olmaktadır. Karışık veri kümesinde doğrudan görüntünün kullanılmasıyla oluşabilecek gereksiz görüntüler çalışmada yer almamıştır. Bu durum, daha doğru bir sınıflandırma yapılmasını sağlamıştır. Ek olarak, karışık veri kümesinin eğitim verilerinde bulunan her hareketin dört defa tekrarı alınmıştır. Böylece daha iyi bir eğitim sağlanarak hatalı sınıflandırma olması engellenmiştir.

Çizelge 3.2. Karışık veri kümesi ile kullanılan yöntemler ve başarı yüzdeleri.

Mesafe Ölç.	Çapraz Korelasyon	Levenshtein Mesafe Ölç.	Frobenius Mesafe Ölç.	Mahalanobis Mesafe Ölç.
SS Yöntemi	53.90	45.66	32.11	37.56
HGG Yöntemi	58.23	50.88	35.21	42.22
SS Yöntemi Öncesi 2B HFD Yöntemi	55.45	60.1	41.03	46.43
HGG Yöntemi Sonrası 2B HFD Yöntemi	63.34	72.47	44.31	51.55
SS Yön. Öncesi 2B HFD Yön. Ve HGG Yön. Sonrası 2B HFD Yöntemi	67.04	84.1	49.65	56.2

Çalışmada iki dezavantaj bulunmaktadır. İlki, vücut önünde yapılan hareketler ve parmak hareketleri kullanılan yöntemler ile istenilen netlikte algılanamamaktadır. Bu nedenle ChaLearn veri kümesi içinde parmak hareketlerini bulduran dosyaların başarı oranı diğer dosyalara kıyasla düşük olmaktadır. İkincisi ise, karmaşık veri kümesi yeni hareketlere açıktır. Fakat veri kümesi içinde az sayıda veri bulunmaktadır.

BÖLÜM 4

SONUÇ

El hareketleri, insan bilgisayar etkileşimi için doğal ve sezgisel bir iletişim yöntemi sağlamaktadır. Çalışmada el hareketlerini sınıflandırmak için tek adımda öğrenme yaklaşımı kullanılmıştır. Yaklaşım, görüntü edinimi, ön işleme, özellik çıkarma ve sınıflandırma olarak dört temel adımdan oluşmaktadır. Görüntü edinimi için çalışmada iki farklı veri kümesi kullanılmıştır. İlk veri kümesi çok zengin fakat karmaşık bir yapıya sahip olan ChaLearn veri kümesi ve ikinci veri kümesi ise bizim oluşturduğumuz karışık veri kümesi olmaktadır. Ön işleme adımında görüntüleri arka plandan ayırmak için Otsu eşik yöntemi uygulanmıştır. Görüntülerden özellik çıkarımı için SS, HGG ve 2B HFD yöntemleri kullanılmıştır. SS yöntemi ile içinde hareket bulunan bir video, yalnızca bir görüntüye dönüştürülmüştür. Ancak bu yöntem hareketin geçmiş bilgisini koruyamamaktadır. HGG yöntemi hareketin geçmiş bilgisini korumakla birlikte hareketin yön bilgisini de korumaktadır. Ayrıca kamera kaymasından oluşabilecek yanlış sınıflandırmayı engellemek için 2B HFD yöntemi kullanılmıştır. Bütün yöntemler iki farklı veri kümesi üzerinde uygulanmıştır. Son olarak veri kümeleri içindeki eğitim ve test verilerini karşılaştırıp, eşleşen hareketleri sınıflandırmak için Levenshtein, Frobenius ve Mahalanobis mesafe ölçümleri ve çapraz korelasyon kullanılmıştır.

Çalışma, hareketler hakkında önceden bilgi sahibi olmayı gerektirmemek ile birlikte veri kümesine eklenen yeni hareketleri de kolaylıkla tanımaktadır. Çalışma sonuçları incelendiğinde ChaLearn veri kümesine uygulanan yöntemlerin %76.15 başarı ve karışık veri kümesine uygulanan %84.1 başarı elde ettiği gözlemlenmiştir. Ek olarak SS, HGG ve 2B HFD yöntemleri ile birlikte Levenshtein mesafe ölçümü ile yapılan sınıflandırma, diğer mesafe ölçümleri ile yapılan sınıflandırmalardan daha yüksek başarı oranına sahip olmaktadır.

KAYNAKLAR

1. Gonzalez, R. C. and Woods, R. E., "Digital image processing", *Addison-Wesley Longman Publishing Co. Inc.*, (2001).
2. Ahad, M. A. R., Tan, J. K., Kim, H. S. and Ishikawa, S., "Human activity recognition various paradigms", *Internat. Conf. on Control Automation and Systems*, 1896-1901 (2008).
3. Bobick, A. and Davis, J., "An appearance-based representation of action", *Proc. 1996 International Conference on Pattern Recognition*, 7270:307-312 (1996).
4. Sharma, R. P. and Verma, G. K., "Human computer interaction using hand gesture", *Procedia Computer Science*, 54: 721-727 (2015).
5. Kiliboz, N. C. and Gdkbay, U., "A hand gesture recognition technique for human-computer interaction", *Journal of Visual Communication and Image Representation*, 28:97-104 (2015).
6. Feng, Z., Yang, B., Lv, N., Xu, T., Yin, J., Zhao, X. and Feng, S., "Motion-towards-each-other-based hand gesture initialization", *Pattern Recognition*, 48:4049-4056 (2015).
7. Ahad, M. A. R., "Computer vision and action recognition: a guide for image processing and computer vision community for action understanding", *Atlantis Ambient and Pervasive Intelligence*, Atlantis Press, (2011).
8. Pisharady, P. K. and Saerbeck, M., "Recent methods and databases in vision-based hand gesture recognition: A review", *Computer Vision and Image Understanding*, 141: 152-165 (2015).
9. Beh, J., Han, D. K., Durasiwami, R. and Ko, H., "Hidden markov model on a unit hypersphere space for gesture trajectory recognition", *Pattern Recognition Letters*, 36: 144-153 (2014).
10. Belgacem, S., Chatelain, C. and Paquet, T., "Gesture sequence recognition with one shot learned CRF/HMM", *Image and Vision Computing*, 61: 12-21 (2017).
11. Hong, P., Turk, M., Huang, T. S., "Gesture modeling and recognition using finite state machines", *IEEE Conference on Face and Gesture Recognition*, 410-415 (2000).
12. Suk, H.-II, Sin, B.-Kee and Lee, S.-Whan, "Hand gesture recognition based on dynamic Bayesian network framework", *Pattern Recognition*, 43: 3059-3072 (2010).

13. Cabrera, M. E. and Wachs, J. P., "A human-centered approach to one-shot gesture learning", *Frontiers in Robotics and AI*, (2017).
14. Weinland, D., Ronfard, R., Boyer, E., "Free viewpoint action recognition using motion history volumes", *Computer Vision and Image Understanding*, 104 (2): 249-257 (2006).
15. Mahbub, U., Imtiaz, H., Ahad, M. A. R., "An optical flow based approach for action recognition", *14th Internat., Conf. on Computer and Information Technology*, 646-651 (2011).
16. Krishnan, R. and Sarkar, S., "Conditional distance based matching for one-shot gesture recognition", *Pattern Recognition*, 48: 1302-1314 (2015).
17. Konency, J. and Hagara, M., "One-shot-learning gesture recognition using HOG-HOF features", *Journal of Machine Learning Research*, 15: 2513-2532 (2014).
18. Gao, J., Cao, Q., "Adaptive HOG-LBP based learning for palm tracking", *2nd International Conference on Computer and Information Application*, Paris, (2012).
19. Feng, K., Yuan, F., "Static hand gesture recognition based on HOG characters and support vector machines", *2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation*, Canada, (2013).
20. -Vela, A. H., Bautista, M. Á., -Sala, X. P., - López, V. P., Escalera, S., Baró, X., Pujol, O. and Angulo, C., "Probability-based dynamic time warping and bag-of-visual-and-depth-words for human gesture recognition in RGB-D", *Pattern Recognition Letters*, 50: 112-121 (2014).
21. Premaratne, P., Yang, S., Vial, P. and Ifthikar, Z., "Centroid tracking based dynamic hand gesture recognition using discrete hidden markov models", *Neurocomputing*, 228: 79-83 (2017).
22. Keogh, E., "Exact indexing of dynamic time warping", *28th International Conference on Very Large Data Bases*, 406-417 (2002).
23. Lemire, D., "Faster retrieval with a two-pass dynamic-time-warping lower bound", *Pattern Recognition*, 42 (9): 2169-2180 (2009).
24. Salvador, S., Chan, P., "Toward accurate dynamic time warping in linear time and space", *Intell. Data Anal.*, 11 (5): 561-580 (2007).
25. Cheng, H., Dai, Z., Liu, Z. and Zhao, Y., "An image-to-class dynamic time warping approach for both 3D static and trajectory hand gesture recognition", *Pattern Recognition*, 55: 137-147 (2016).

26. Tang, J., Cheng, H., Zhao, Y. and Guo, H., “Structured dynamic time warping for continuous hand trajectory gesture recognition”, *Pattern Recognition*, 80: 21-31 (2018).
27. Vijay, P. P. and Patil, N. C., “Gray scale image segmentation using Otsu thresholding optimal approach”, *Journal for Research*, 02 (2016).
28. Yu, C., Dian-ren, C., Yang, L. and Lei, C., “Otsu’s thresholding method based on gray level-gradient two-dimensional histogram”, *2nd International Asia Conference on Informatics in Control, Automation and Robotics*, (2010).
29. Liu, D., Yu J., “Otsu method and k-means”, *2009 Ninth International Conference on Hybrid Intelligent Systems*, Shenyang, China, (2009).
30. Otsu, N., “A threshold selection method from gray-level histograms”, *IEEE Transactions on Systems, Man, and Cybernetics*, 9: 62-66 (1979).
31. Shibo, X., Yali, W., Wenqing, L., “Standard deviation control chart based on weighted standard deviation method”, *Proceedings of the 33rd Chinese Control Conference*, China, (2014).
32. Mahbub, U., Imtiaz, H., Roy, T., Rahman, M. S. and Ahad, M. A. R., “A template matching approach of one-shot-learning gesture recognition”, *Pattern Recognition Letters*, 34: 1780-1788 (2013).
33. Bobick, A. F. and Davis, J. W., “The recognition of human movement using temporal templates”, *IEEE Trans. Pattern Anal. Machine Intell.*, (23)3: 257-267 (2001).
34. Shan, C., Wei, Y., Qiu, X. and Tan, T., “Gesture recognition using temporal template based trajectories”, *Proceedings of the 17th International Conference on Pattern Recognition*, 1051-14651 (2004).
35. Rautaray, S. S. and Agrawal, A., “Vision based hand gesture recognition for human computer interaction: survey”, *Artificial Intelligence Review*, 43: 1-54 (2015).
36. Mahbub, U., Roy, T., Rahman, M. S., Imtiaz, H., Serikawa, S. and Ahad, M. A. R., “One-shot-learning gesture recognition using motion history based gesture silhouettes”, *International Conference on Industrial Application Engineering*, (2012).
37. Ahad, M. A. R., Tan, J. K., Kim, H. and Ishikawa S., “Motion history image its variants and applications”, *Machine Vision and Applications*, 23: 255-281 (2012).
38. Bongiovanni, G., Corsini, P., Frosini, “Two dimensional properties of discrete Fourier transform”, *Calcolo*, 13 (2): 131-143 (1976).

39. Mahmood, F., Toots, M., Öfverstedt, L., and Skoglund, U., “Algorithm and architecture optimization for 2D discrete Fourier transforms with simultaneous edge artifact removal”, *International Journal of Reconfigurable Computing*, (2018).
40. Rzeszotarski, M. S., Royer, F. L., Gilmore, G. C., “Introduction to two-dimensional Fourier analysis”, *Behavior Research Methods & Instrumentation*, 15 (2): 308-318 (1983).
41. Ruffieux, S., Lalanne, D., Mugellini, E. and Khaled, O. A., “A survey of datasets for human gesture recognition”, *Human-Computer Interaction Advanced Interaction Modalities and Techniques*, 8511: 337-348 (2014).
42. Internet: 3D Iconic Gesture Dataset (3DIG), “3D iconic gesture dataset (3DIG)”, <http://projects.ict.usc.edu/3dig/> .
43. Internet: Kaggle, “ASL alphabet”, <https://www.kaggle.com/grassknotted/asl-alphabet>, (2016).
44. Internet: ChAirGest 2013, “Challenge for multimodal mid-air gesture recognition for close hci”, <https://project.heia-fr.ch/chaigest/Pages/Overview.aspx> (2016).
45. Internet: Pgram, “Shefeld kinect gesture (SKIG) dataset”, <http://www.pgram.com/dataset/shefeld-kinect-gesture-skig-dataset/>, (2015).
46. Chen, M, AlRegib, G., Juang, B., “6DMG: A new 6D motion gesture database”, *Proceedings of the Second ACM Multimedia Systems Conference*, USA, (2012).
47. Bloom, V., Makris, D., Argriyou, V., “G3D: A gaming action dataset and real time action recognition evaluation framework”, *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, USA, (2012).
48. Internet: American Sign Language Lexicon Video Dataset (ASVLLD) circa 2008, “American sign language lexicon video dataset (ASVLLD) circa 2008”, <http://csr.bu.edu/asl/asllvd/annotate/index-cvpr4hb08-dataset.html>, (2008).
49. Athitsos, V., Neidle, C., Sclaroff, S. and Nash, J., “The American sign language lexicon video dataset”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, (2008).
50. Internet: Jchen, “3D human recognition dataset”, <https://beckschen.github.io/2018/05/13/HA-Dataset/>, (2018).
51. Guyon, I., Athitsos, V. and Jangyodsuk, P., “The ChaLearn gesture dataset (CGD 2011)”, *Machine Vision and Application*, 25: 1929-1951 (2014).

52. Wan, J., Li, S. Z., Zhao, Y., Guyon, I. and Escalera, S., “ChaLearn looking at people RGB-D isolated and continuous datasets for gesture recognition”, ***IEEE Conference Computer Vision and Pattern Recognition Workshops***, (2016).
53. Internet: Zugara, “How does the kinect 2 compare to the kinect 1?”, <http://zugara.com/how-does-the-kinect-2-compare-to-the-kinect-1> (2009).
54. Wang, X., Mueen, A., Ding, H., Trajcevski, G., Scheuermann, P., Keogh, E., “Experimental comparison of representation methods and distance measures for time series data”, ***Data Min. Knowl. Discov.***, 26 (2): 275-309 (2013).
55. Huang, Z, Zheng, S., “The cross-correlation properties of composite system”, ***Scientific Reports***, 1297 (2018).
56. Levenshtein, V. I., “Binary codes capable of correcting deletions, insertions, and reversals”, ***Soviet Physics Doklady***, 163: 845-848 (1965).
57. Konstantinidis, S., “Computing the edit distance of a regular language”, ***Information and Computation***, 205: 1307-1316 (2007).
58. Teodoro, B. T., Bernardes, J. and Digiampietri, L. A., “Skin color segmentation and Levenshtein distance recognition of BSL signs in video”, ***2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)***, Brazil, (2017).
59. Internet: Wikipedia, “Frobenius method”, https://en.wikipedia.org/wiki/Frobenius_method, (2009).
60. De Maesschalck, R., Jouan-Rimbaud, D., Massart, D. L., “The mahalanobis distance”, ***Chemometrics and Intelligent Laboratory Systems***, 50 (1): 1-18 (2000).
61. Fan, Z., Ni, M., Sheng, M., Wu, Z., Xu, B., “Principal component analysis integrating mahalanobis distance for face recognition”, ***2013 Second International Conference on Robot, Vision and Signal Processing***, Japan, (2013).

ÖZGEÇMİŞ

Esmâ ŞEKER 1992 yılında Sivas’da doğdu. İlk ve orta öğrenimini aynı şehirde tamamladı. Sivas Selçuk Anadolu Lisesi’nden mezun oldu. 2010 yılında Eskişehir Anadolu Üniversitesi Bilgisayar Mühendisliği Bölümü’nde öğrenime başlayıp 2015 yılında iyi derece ile mezun oldu. 2015 yılında Karabük Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda yüksek lisans eğitimine başladı. 2015 yılında Karabük Üniversitesi’nde göreve başladı ve halen aynı yerde çalışmaya devam etmektedir.

ADRES BİLGİLERİ

Adres : Karabük Üniversitesi
Mühendislik Fakültesi/Mekatronik Mühendisliği
Balıklarkayası Mevkii / KARABÜK
Tel : (541) 485 4758
E-posta : esmaseker@karabuk.edu.tr