3B ZERNİKE MOMENTLERİ KULLANILARAK İNSAN HAREKETLERİNİN TANINMASI

HUMAN ACTION RECOGNITION USING 3D ZERNIKE MOMENTS

OKAY ARIK

YRD. DOÇ. DR. SEMİH BİNGÖL

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Elektrik ve Elektronik Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2014

OKAY ARIK' ın hazırladığı "**3B Zernike Momentleri Kullanılarak İnsan Hareketlerinin Tanınması**" adlı bu çalışma aşağıdaki jüri tarafından **ELEKTRİK VE ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**' nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. H. Gökhan İLK
Başkan .....

Yrd. Doç. Dr. Semih BİNGÖL
Danışman .....

Doç. Dr. Ali Ziya ALKAR
Üye .....

Yrd. Doç. Dr. Yakup ÖZKAZANÇ
Üye .....

Doç. Dr. Atila YILMAZ
Üye .....

Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından YÜKSEK LİSANS TEZİ olarak onaylanmıştır.

Prof. Dr. Fatma SEVİN DÜZ
Fen Bilimleri Enstitüsü Müdürü

ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında,

- Tez içinde bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili esere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

07/01/2014

OKAY ARIK

ÖZET

3B ZERNİKE MOMENTLERİ KULLANILARAK İNSAN HAREKETLERİNİN TANINMASI

OKAY ARIK

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Danışmanı: Yrd. Doç. Dr. SEMİH BİNGÖL

Ocak 2014, 40 Sayfa

Zernike momentleri dönme hareketine karşı bağımsız olduğundan iki boyutlu görüntülerdeki şekilleri tanımada sıklıkla kullanılan bir araçtır. Benzer biçimde 3B Zernike momentlerinin genlikleri üç boyutlu görüntülerde bu özelliği gösterir. Son yıllarda yaygınlaşan optik sistemlerle insanların da üç boyutlu görüntülerini almak mümkün hale gelmiştir. Bu çalışmada insanlara ait üç boyutlu görüntüleri 3B Zernike momentleriyle analiz ederek görüntülerde yapılan hareketlerin sınıflandırılmasına yönelik yeni bir yöntem önerilmiştir. Önerilen yöntem çoklu kameralı bir hareket yakalama stüdyosunda oluşturulan i3DPost isimli veri kümesinde uygulanmıştır. Veri kümesinde bulunan yürüme, koşma, zıplama, eğilme, el sallama, hoplama ve çömelme gibi temel hareketler %99'dan fazla bir doğruluk oranıyla sınıflandırılabilmiştir.

Anahtar Kelimeler: Zernike Momentleri, Hareket Tanıma, Vücut Duruşu Tanıma, Yapay Görme, 3B Geri Çatım, Görüntü İşleme

ABSTRACT

HUMAN ACTION RECOGNITION
USING 3D ZERNIKE MOMENTS

OKAY ARIK

Master of Science, Department of Electrical and Electronics Engineering

Supervisor: Assist. Prof. Dr. SEMİH BİNGÖL

January 2014, 40 Pages

Because they are invariant under rotation, Zernike moments are widely used for shape recognition in two-dimensional images. Similarly, magnitudes of 3D Zernike moments have the analogous property in three dimensional images. In recent years, capturing 3D images of humans has become feasible as a result of advances in optical technologies. In this work, we propose a new method to classify human actions which have been recorded as three dimensional images by using 3D Zernike moments. We have applied our method to the i3DPost Multi-view Human Action Dataset. We were able to classify the actions in the dataset into main activities such as walking, running, jumping, bending, hand-waving, jumping in place and sitting with an accuracy of greater than 99 %.

Keywords: Zernike Moments, Action Recognition, Pose Recognition, Machine Vision, 3D Reconstruction, Image Processing

TEŞEKKÜR

Bu çalışmam boyunca yardım ve desteğini esirgemeyen değerli hocam Yrd. Doç. Dr. Semih Bingöl'e teşekkür ederim. Önerileri ve yönlendirmeleri çalışmama yön verdi.

Ailem ve dostlarım bu süreçte hep yanımda oldular, teşekkürlerimi iletiyorum.

Son olarak başta Dr. Hansung Kim olmak üzere bu çalışmayı yapmama olanak sağlayan i3DPost veri kümesinin oluşumunda emeği geçen bütün bilim insanlarına teşekkür etmeliyim.

İÇİNDEKİLER

Sayfa

SİMGELER VE KISALTMALAR

Simgeler

$(\cdot)^*$	Karmaşık Eşlenik (*Complex Conjugate*)
$(\cdot)[n]$	Vektörün n'inci bileşeni

Kısaltmalar

3B	Üç Boyutlu
LIDAR	*Light Detection and Ranging*
ToF	*Time of Flight*
mo-cap	*Motion Capture* (Hareket Yakalama)
fps	*Frame per Second* (Çerçeve Bölü Saniye)
.png	*Portable Network Graphics*
ascii	*American Standard Code for Information Interchange* (Amerikan Standart Kodlama Sistemi)
.ntri	*Triangle File Format*
.txt	Metin Dosyası
voxel	*Volumetric Pixel* (Hacimsel Piksel)
blob	*Binary Large Object*

1. GİRİŞ

Son dönemlerde insan makine etkileşimini daha verimli hale getirme arayışları hızlanmıştır. Geçmişte bu ihtiyaç klavye, fare, joystick ve uzaktan kumanda gibi geleneksel elektro-mekanik yardımcı cihazlarla sağlanmaktaydı. Ancak, gelişen teknolojiyle birlikte, doğrudan insan hareketleriyle makinelere komut verilebilmesi ve makinelerce yaratılan sanal gerçeklikle etkileşime girilebilmesi mümkün hale gelmiştir.

Bu amaçla, insan hareketlerinin algılanıp tanınması teknolojinin önemli hedeflerinden biri olmuştur. Ancak insan vücudu üç boyutlu bir yapı olduğundan bu bilgiyi almanın en sağlıklı yolu yine üç boyutlu bir veri girdisiyle mümkün olacaktır. İnsan vücudunun duruşunu üç boyutlu olarak algılama işlemine hareket yakalama (*motion capture*) adı verilir. Bu amaçla tasarlanmış sistemler iki ayrı teknolojiye dayanır: Hareket algılayıcılar ve optik sistemler.

Hareket algılayıcılar, nesnelerin anlık ivmelerini ölçen ivmeölçerler ve doğrultularını veren jirometrelerden oluşur. Bu algılayıcılar sayesinde üzerine yerleştirildikleri nesnelerin üç boyutlu uzaydaki konum ve doğrultularını elde etmek mümkündür. Kişinin eklem bölgelerine bu algılayıcılar yerleştirilerek eklem noktalarının üç boyutlu uzaydaki yörüngeleri çıkartılabilir. Böylece kişinin zamana göre değişen iskelet modeli elde edilmiş olur.

Optik sistemler ise üçgenleme (*triangulation*) esasına dayanır. Farklı konumlarda bulunan iki ayrı gözlemci aynı nesneyi farklı açılarda görür. Bu açısal farklılıktan (paralaks) yararlanarak nesnenin konumunu kestirmek mümkündür. Bu yöntemle birden fazla kamera kullanarak nesnelerin 3B geri çatımı (*3D reconstruction*) yapılabilmektedir. Ancak iki ya da daha çok kameradan elde edilen görüntülerdeki noktaların doğru olarak eşleştirilmesi zorunludur.

Bu işlemi kolaylaştırmak için kameralarca kolaylıkla tespit edilip takip edilebilecek işaretler (*marker*) kullanılabilir. Hareket yakalama için bu işaretler tıpkı hareket algılayıcılarda olduğu gibi kişinin eklem noktalarına yerleştirilir. Böylece bu noktaların konumlarını kestirmek ve kişinin iskelet modelini çıkartmak mümkün olur. Ancak hareket yakalama süresince işaretler her zaman en az iki kamera tarafından görülebilir olmalıdır. Bu sebeple iki kamera yerine hareketin yapılacağı platformun çevresine en az 6-8 kamera yerleştirilir. Kameraların yakalayacağı işaretler aktif ve pasif olarak ikiye ayrılır. Aktif

işaretler kameralarca kolay algılanabilmek için belirli bir ışık sinyali yayarlar. Pasif olanları ise şekil, renk ve yansıtıcılık gibi özellikleri sayesinde fark edilirler.

Yukarıda anlatılan iki yöntemde de kişinin iskelet modeli doğrudan elde edilebilmektedir. Ancak iki durumda da kişiye bir takım özel teçhizat giydirmek gereklidir. Bu zorunluluğu ortadan kaldıran üçüncü yöntem ise yine optik temelli olan işaretsiz hareket yakalamadır (*markerless motion capture*). Bu yöntemde kişiye özel işaretler yerleştirmek yerine hareket yakalama stüdyosunun arka planı belirli bir renkle kaplanır. Bu çalışmada kullanılan i3DPost veri kümesi [1] de böyle bir hareket yakalama stüdyosu ile oluşturulmuştur. Bölüm 2.1'de detaylı olarak anlatılacak olan veri kümesi çeşitli deneklerce yapılan muhtelif hareketlerin görüntülerinden oluşmaktadır. Bu yöntemde özel arka plan sayesinde kişinin görüntü içindeki sınırları kolayca çıkartılır ve Bölüm 2.2 ve 2.3'te verilecek bir dizi yöntemle kişinin üç boyutlu görüntüsü elde edilir. Bu yöntemde farklı olarak doğrudan kişinin iskelet modeli oluşturulamaz. Elde edilen 3B görüntünün tekrar işlenmesi ve bir iskelete oturtulması gereklidir. Bununla beraber kişinin harici bir donanım taşıma zorunluluğu ortadan kalkmıştır. Bu özelliğiyle de bu yöntem günlük hayata uygulanabilme özelliğiyle öne çıkmaktadır.

Bu yöntemin zayıf yönü olan özel arka plan zorunluluğu daha gelişmiş arka plan çıkartma (*background subtraction*) algoritmalarıyla aşılabilir. Bir diğer çözüm ise üçgenleme işlemi için iki kamera kullanmak yerine kameralardan birini aktif bir optik sistemle; lazer ya da projektörle değiştirmektir. Bu sayede lazer ya da projektörün gönderdiği ışık sinyali kamera tarafından daha kolay yakalanacaktır. 3B lazer tarayıcılar bu yöntemle sabit nesnelerin 3B geri çatımını hassasiyetle kotarabilmektedir. Benzer biçimde yapısal ışık (*structural light*) yönteminde projektör yardımıyla sabit nesnenin üzerine birden fazla desen (*pattern*) yansıtarak 3B geri çatım yapılmaktadır. Ancak hareketli nesneler için bu işlemin yapılabilmesi ancak projektörün yansıtacağı sabit bir desenle mümkün olur. PrimeSense firmasının geliştirdiği Kinect 3B sensörü bu yöntemle gerçek zamanlı hareket yakalama yapabilmektedir. İlerleyen yıllarda, bugün 3B geri çatımda kullanılan LIDAR (*Light Detection and Ranging*) teknolojisi ve ToF (*time of flight*) kameralar da hareket yakalama amacıyla kullanılabilir.

Bu tez çalışmasının amacı, yukarıda özetlenmeye çalışılan yöntemlerle ya da tamamen farklı teknolojilerle bir biçimde elde edilmiş 3B insan görüntülerinin 3B Zernike momentleri kullanılarak analiz edilmesi ve görüntülerde yapılan hareketlerin tanınmasıdır.

3B görüntüsünden iskelet modelinin çıkarılması esas olarak bir parametre kestirim problemidir. Önce iskelet modeli cebirsel olarak ifade edilir. Modelin parametreleri vücut duruşunu belirleyen eklem açıları olacaktır. Ek olarak vücut ölçüleri de birer parametre olarak tanımlanabilir. Ardından 3B görüntü üzerindeki noktaların modele en iyi oturduğu parametreler kestirilmeye çalışılır.

I. Mikic ve diğerleri [2] bu ilkeye dayanan bir hareket tanıma yöntemi önermişlerdir. Yöntemin ilk adımı olan tespit algoritmasında başlangıç görüntüsü için en uygun parametreler kestirilir. Ardından gelen takip algoritmasında ise bir önceki görüntüye ait parametreler başlangıç değeri olarak alınıp yeni görüntü için parametreler güncellenir. Sistem ani hareketler karşısında kararlı olmadığından sıklıkla parametreler sıfırlanarak hesaplama süresi daha uzun olan tespit algoritması geri çağırılmaktadır.

Hareket tanıma vücut duruşu, ya da poz tanımaya oranla daha karmaşık bir iştir. Temel olarak vücut duruşu anlık bir durumken, hareket belirli bir süre içerisinde gerçekleşir. Bu nedenle eğer hareket duruş üzerinden tanınacak ise duruş modelinin parametreleri zaman ekseni boyunca belirli bir uzunlukta pencere üzerinden analiz edilmelidir. Öte yandan, vücut duruşu tanıma evresini hiç kullanmadan doğrudan hareket tanıması yapabilmek de mümkündür.

Hareket ve/veya duruş tanıma gerek iki boyutta, gerekse üç boyutta yapılabilir. Bu çalışmada da kullanılan i3DPost veri kümesi üzerinde A. Iosofidis ve diğerlerinin yaptığı dört farklı çalışmada [3, 4, 5, 6] ve B. Mahasseni ve S. Todorovic'in çalışmalarında [7] hareket tanıma için iki boyutlu görüntüler kullanılmıştır. Bu çalışmalarda görüntülerdeki hareket bölgesi belirli şablonlarla karşılaştırılmış buna göre hareket tanıması yapılmıştır. Hareket yöneliminin kamera açısından bağımsız olmasının yarattığı sorun hareket yakalama stüdyosunu her açıdan izleyebilen sekiz kamerayla giderilmiştir.

Yine i3DPost veri kümesi üzerinde B. Holte ve diğerlerince yapılan iki ayrı çalışmada [8, 9] ise üç boyutlu görüntülerle hareket tanıma yapılmıştır. Bu çalışmalarda kısaca üç boyutlu görüntülerdeki optik akış (*optical flow*) çıkartılmış, sonrasında bu üç boyutlu veriler, bu çalışmada kullandığımız 3B Zernike fonksiyonlarının da içeriğinde bulunan küresel harmonikler yardımıyla analiz edilmiş ve hareket tanıma yapılabilmiştir.

3B görüntüden duruş tanıma için bir diğer yol ise bu çalışmanın temel varsayımını doğrular nitelikte olan D. Berjón ve F. Morán'ın 3B Zernike momentlerini kullanarak

vücut duruşunu kestirdikleri çalışmadır [10]. Her ne kadar tezlerini gerçek bir ortamda değil bilgisayar ortamında oluşturulmuş bir benzetim modeliyle sınamış olsalar da, bu çalışma bize 3B Zernike momentlerinin vücut duruş bilgisini taşıdığını işaret etmekte ve hareket tanımada da kullanılabileceği ipucunu vermektedir. Bu tez çalışmasının temel fikri de, 3B Zernike momentleri kullanılarak, duruş tanıması yapmaksızın doğrudan hareket tanıma yapılabileceğidir.

İnsan hareketleri üç boyutlu ortamda gerçekleşirken, kamera, bunun sadece iki boyutlu bir izdüşümünü kaydedebilmektedir. Dolayısıyla, elde edilen kayıt üç boyutlu hareketin kameranın açısına bağlı olarak iki boyuttaki izdüşümü olup hareket hakkında bütün bilgiyi içermeyecektir. Sonuç olarak, yönelimden bağımsız hareket tanıma için iki ya da daha fazla kamera kullanılması gerekmektedir. Nitekim, yukarıda bahsedilen iki boyutlu hareket tanıma çalışmalarında, hareket yöneliminin kamera açısından bağımsız olma zorunluluğu, hareket yakalama stüdyosunu her açıdan izleyebilen sekiz kamera sayesinde sağlanmıştır.

Holte ve diğerleri yaptıkları kapsamlı karşılaştırmada üç boyutlu hareket tanımanın iki boyutlu hareket tanımaya göre genelde performans açısından daha üstün olduğu, ancak daha fazla işlem yükü gerektirdiği sonucuna varmışlardır [11]. Geçen on yılda iki boyutlu verilere dayalı hareket tanıma hakkında yapılmış çok sayıda çalışma ve geliştirilmiş onlarca farklı yöntem mevcutken, üç boyutlu hareket tanıma konusunda yapılmış çalışma hala çok azdır. Bu konuda yapılmış yayınların karşılaştırmalı bir özeti Holte ve diğerlerinin makalesinde bulunabilir [11].

Bu tez çalışmasının amacı, yukarıda özetlenmeye çalışılan yöntemlerle ya da tamamen farklı teknolojilerle elde edilmiş 3B insan görüntülerinin 3B Zernike momentleri kullanılarak analiz edilmesi ve görüntülerde yapılan hareketlerin tanınmasıdır. İki boyutlu hareket tanıma yöntem ve çalışmaları bu tezin ilgi alanı dışında olup sadece performans karşılaştırması yapılırken bahsedilecektir.

3B Zernike momentleri N. Canterakis [12] tarafından tanımlanmıştır. İki boyutlu uzayda tanımlı olan Zernike momentleri, genlikleri dönme hareketine karşı bağımsız olduğundan görüntüler üzerinde yazı karakterleri gibi şekilsel yapıları algılamada sıklıkla kullanılan bir araçtır. Benzer biçimde 3B Zernike momentlerinin genlikleri de üç boyutlu uzayda dikey eksen etrafındaki dönme hareketine karşı bağımsızlık özelliği gösterir. Momentlerin bu özelliği, vücut duruşunun da dikey eksen etrafındaki dönme hareketinden bağımsız olması

ile uyumludur. Momentler ve özellikleri hakkında daha detaylı bilgi Bölüm 3'te verilmiştir.

Bu çalışmadaki temel varsayım, her vücut duruşunun moment uzayındaki ayrı bir nokta tarafından temsil edilebileceğidir. Bu sebeple moment uzayındaki bu noktaların zaman içinde takip ettiği yörüngeler incelenmiş; hareket algılama işlemi, yapılan hareketle ilintili bu yörüngeler üzerinde belirli örüntüler çıkartmak üzerine kurulmuştur.

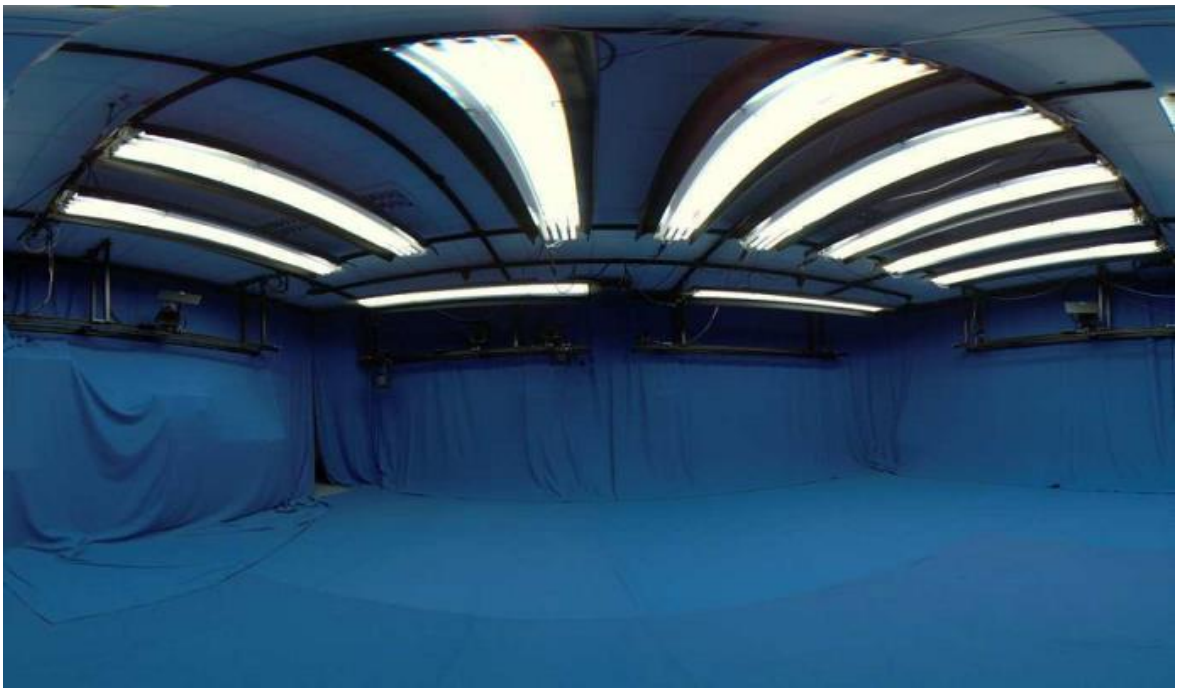
Tez çalışması sürecinde önce veri kümesinden Bölüm 2'de anlatılan yöntemlerde 3B görüntüler elde edilmiştir. Ardından bu görüntülerin, Bölüm 3'te detayları verilen 3B Zernike momentleri hesaplanmıştır. Elde edilen moment verileri üzerinde yapılan inceleme ve hareket tanıma işlemine yönelik algoritmanın geliştirilmesi Bölüm 4'de yer almaktadır. Son olarak elde edilen sonuçlar ve hareket tanıma işleminin başarımı Bölüm 5'te verilip tartışılmıştır.

2. 3B GÖRÜNTÜ VERİSİNİN OLUŞTURULMASI

3. Bölümde detaylı olarak anlatılacak olan 3B Zernike momentleri üç boyutlu fonksiyonlar ya da üç boyutlu görüntüler üzerinden hesaplanır. İnsan hareketlerinin 3B Zernike momentleriyle tanınabilmesi için insan vücudunu ifade eden üç boyutlu görüntülere ihtiyaç vardır.

2.1 İnsan Hareket Veri Kümesi

Bu tez çalışmasında N. Gkalelis ve diğerleri tarafından oluşturulan "**i3DPost Multi-view Human Action Datasets**" isimli veri kümesi kullanılmıştır [1, 13]. Veri kümesi özel bir stüdyoda 8 kamera tarafından kaydedilen görüntülerden oluşmaktadır. Kameralar stüdyonun ortasındaki genişçe bir alanı bütün açılardan görecek ve stüdyonun etrafını çevreleyecek şekilde yerleştirilmiştir (Şekil 2.1).

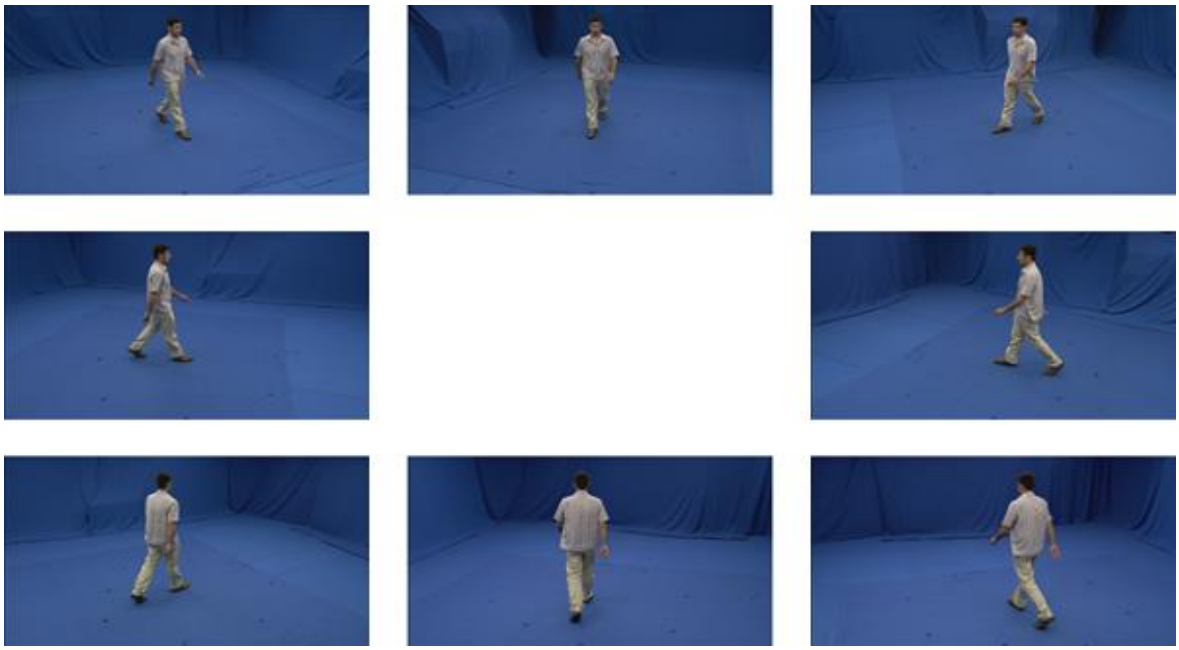


Şekil 2.1. Hareket Yakalama Stüdyosu

Hareket yakalama stüdyolarında kişi çoğunlukla eklem bölgelerinde kameralarca kolaylıkla yakalanabilecek işaretler bulunan özel bir kıyafet giyer. Bir başka yöntem ise kişiye herhangi bir özel kıyafet giydirilmeden arka planın belirli bir renge boyanmasıdır. Bu sayede kişinin görüntü içindeki sınırları kameralarca kolaylıkla ayırt edilebilir. Veri kümesini oluşturmakta kullanılan stüdyoda da bu yöntem uygulanmış, stüdyonun duvarları ve zemini mavi renkli kumaşla kaplanmıştır. Veri kümesi 8 deneğe ait 12 çekimden

oluşmaktadır. Bu çekimlerde yürüme, koşma, zıplama, eğilme, el sallama, hoplama, çömelme ve bu eylemlerin çeşitli birleşimleri bulunmaktadır. Ancak son iki çekim iki deneğin karşılıklı etkileşimli eylemlerini içerdiğinden çalışmamıza dâhil edilmemiş ve geri kalan 10 çekim üzerinde çalışılmıştır. Denekler farklı vücut ölçülerinde ve cinsiyetlerdedir (6 erkek ve 2 kadın). Sonuç olarak çalışmada toplamda 80 ayrı çekim kullanılmıştır.

Veri kümesinde 8 kameraya ait 25 fps'de alınmış 1920×1080 çözünürlüğünde .png formatında video kareleri bulunmaktadır (Şekil 2.2).



Şekil 2.2. Kameralardan elde edilen kareler

Kullandığımız veri kümesine ilişkin makalelerde [13, 14] bu veri kümesi kullanılarak 3B yüzey verisi (*3D mesh data*) elde etme çalışması da yapılmıştır. Veri kümesinin içinde bu çalışmanın sonucu elde edilen 3B yüzey verisi de bulunmaktadır. 3B yüzey verisi her kare için ASCII formatında .ntri uzantılı dosyalarda tutulmuştur. Dosyalar iki bölümden oluşmaktadır. İlk bölümde yüzey üzerindeki köşe noktalarının üç boyutlu koordinatları bulunurken ikinci bölümde yüzeyi oluşturan çokgenlerin köşe numaraları verilmiştir.

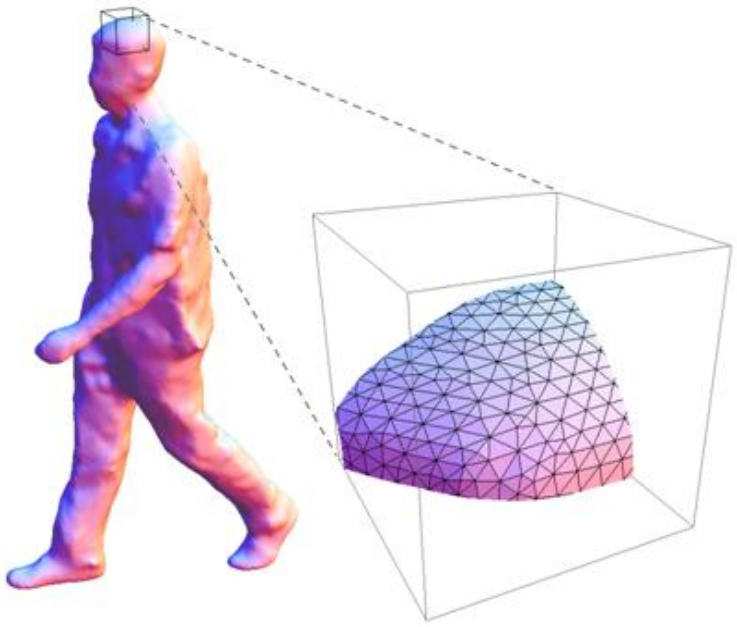
2.2 3B Yüzey Verisinden 3B Görüntü Verisinin Oluşturulması

Yukarıda değinildiği gibi i3DPost veri kümesinin içinde 3B yüzey verisi bulunmaktadır. Ancak çalışmada önerildiği biçimde bu verilerin 3B Zernike momentleriyle kullanılabilmesi için 3B görüntü formunda olması gereklidir. 3B görüntü formundan kasıt tıpkı iki boyutlu standart görüntülerin matris yapısıyla ifade edilebilmeleri gibi üç boyutlu

matris yapısıyla ifade edilebilen görüntülerdir. Standart iki boyutlu görüntüler piksel adı verilen küçük karelerden oluşurken 3B görüntüler ise voxel adlı küçük küplerden oluşur.

Voxel boyutları doğrudan 3B görüntünün çözünürlüğünü ve kalitesini belirler. Voxel olarak çok büyük küpler seçilecek olursa elde edilecek 3B görüntü insan vücudunun şeklini temsil etmekten uzak olacaktır. Çok küçük küplerin seçilmesi durumundaysa işlenecek veri artacak, işlem süresi uzayacaktır. Çalışmada bir kenarı 5 santimetrelik voxellerle çalışılmıştır. Bu boyut kol ve bacak gibi görece ince vücut uzuvlarını yakalayabilecek kadar küçüktür.

Elimizde ise veri kümesindeki yüzey verisinden gelen üç boyutlu modelin yüzeyinde rastgele dağılmış noktalar vardır (Şekil 2.3). Bu yüzey noktalarının koordinatları voxel boyutunun katlarına yuvarlanarak yüzey bilgisi kolayca 3B görüntüsüne dönüştürülebilir. Ancak elde edilecek kabuk görüntüsünün içi beklenileceği gibi boş olacaktır. Bu da insan vücudunun gövde gibi kalın bölümlerinin moment hesabında ağırlığını azaltıp yüzeysel dalgalanmaların yüksek olduğu eklem bölgelerinin ağırlığını artırabilir. Bunlar sonucunda elde edilen momentler benzer vücut duruşlarında benzer sonuçlar vermekten uzak olacaktır. Bu da hareket tanımayı güçleştirecektir. Bu nedenle elde edilen kabuk görüntüsünün içi doldurulmalıdır.

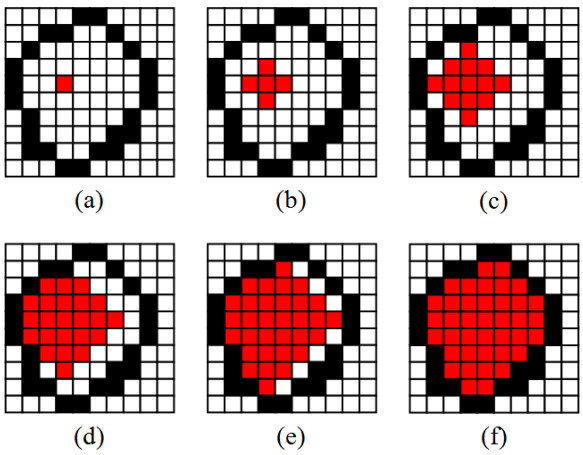


Şekil 2.3. 3B yüzey verisi ve yüzeye ait detay

2.2.1 3B Kabuk Görüntüsünün Doldurulması

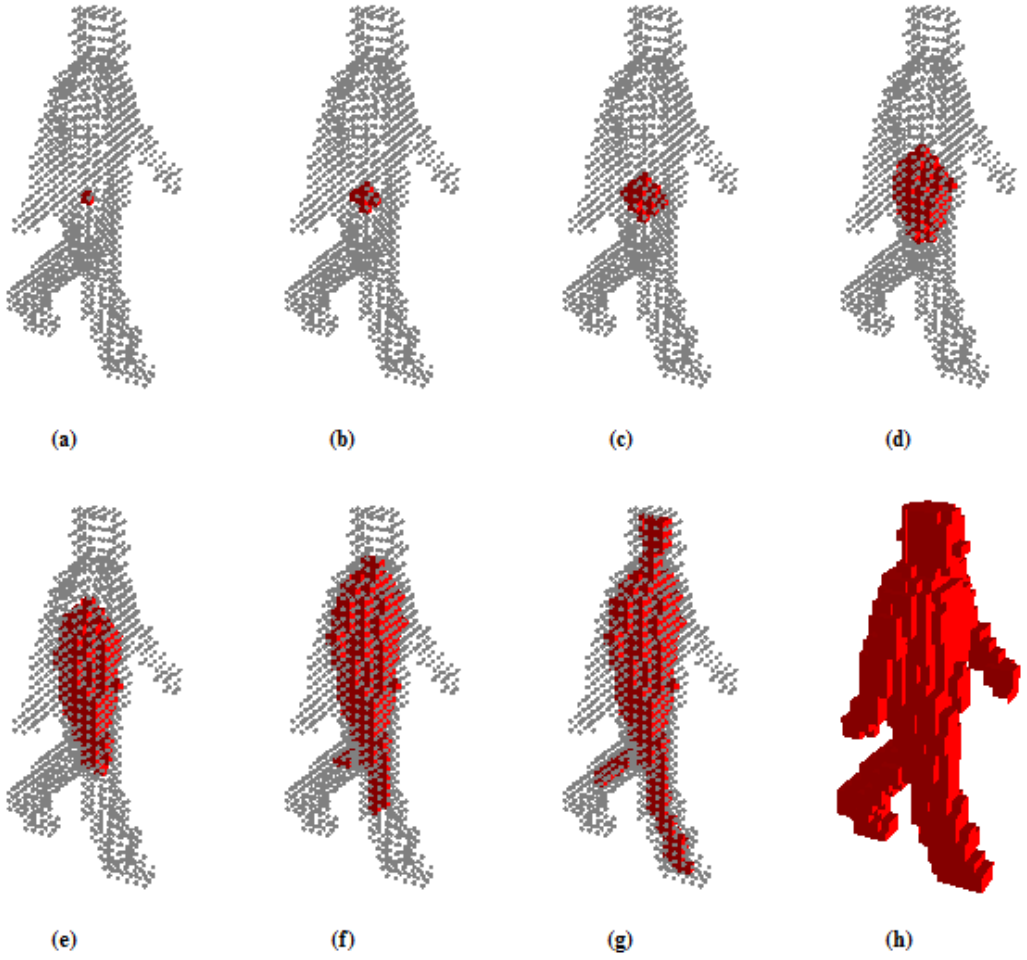
3B kabuk görüntüsünün içinin doldurulması morfolojik bir problemdir [15]. Herhangi bir voxel, verilen bir kabuk görüntüsünün iç ya da dış bölgesinde ise o voxele morfolojik olarak komşu olan diğer voxeller de aynı bölgede ya da kabuğun üzerinde olacaklardır. Bu durumda iç bölgede olduğundan emin olunan bir tohum voxelle başlanıp bir iç görüntü tanımlanabilir. Bu iç görüntü morfolojik genişleme işlemine tabi tutularak tohum voxelin etrafındaki komşu voxeller de iç görüntüsüne katılabilir. Kabukla kesişme ihtimaline karşı bu iç görüntüden kabuk görüntüsünü çıkarmak gerekecektir. Bu işlemleri tekrarlayarak iç görüntüyü her defasında büyütmek ve tamamen iç bölgeyi kaplamasını sağlamak mümkündür. Ancak kabuk görüntüsü basit bir kapalı yüzey olmak yerine kendiyle kesişen bir yapıda olabilir. Bu durumda yinelemeli olarak yürüttüğümüz işlem hiçbir zaman bütün iç bölgeye ulaşamaz. Ancak uygulamada bu durum sadece kafa, kol ve bacaklarda küçük boşluklara yol açmaktadır ve bu nedenle ihmal edilebilir. Yinelemeli algoritmamızı durdurma kıstası olarak artan voxel sayısı alınabilir. İç görüntü doyuma ulaştığında artış sıfır olacaktır ve bu noktada algoritma durur.

Morfolojik kabuk doldurma işlemini daha basit olarak iki boyutlu bir görüntü üzerinde inceleyebiliriz. Şekil 2.4'te bu süreç gösterilmiştir. (a)'da kabuk görüntüsü siyahla, seçilen tohum piksel kırmızı ile gösterilmiştir. (b)'de ve (c)'de sırasıyla yapılan morfolojik genişletme işlemini görmekteyiz. Bu noktaya kadar kabukla kesişim yoktur. Ancak (c)'deki iç görüntünün genişletilmesi kabukla kesişeceğinden genişletilmiş görüntüden kabuk görüntüsü çıkarılır ve (d)'deki görüntü elde edilir. İşleme devam edilecek olursa nihayet (f)'de iç görüntü bütün iç bölgeyi kaplamış olur. Bu noktadan sonra yapılacak yinelemeler iç görüntünün alanını büyütmeyecek ve aynı çıktıyı verecektir. Burada algoritma durur.



Şekil 2.4. İki boyutta kabuk doldurma işlemi

Şekil 2.5'te 3B kabuk görüntüsünün içinin doldurulma süreci izlenebilir. (a)'da kabuk görüntüsünü temsil eden gri noktalar ve ortadaki kırmızı tohum voxeli görülmektedir. (b)'den (g)'ye kadar yinelemeli algoritma sırasıyla 1, 2, 5, 10, 15 ve 25 kez tekrarlanmıştır. İç görüntü bu noktada doyuma ulaşmıştır. Bu noktadan sonra yapılacak tekrarlar iç görüntüsüne yeni voxeller katmayacaktır. Burada algoritma durur. Son olarak (h)'de iç görüntü ile kabuk görüntüsünün birleşimi olan hedeflenen 3B görüntüsü görülmektedir.



Şekil 2.5. 3B kabuk görüntüsünün içinin doldurulması

2.2.2 Tohum Voxelin Seçimi

Burada kritik bir nokta tohum voxelinin seçimidir. Tanımı gereği iç bölgede olduğu bilinen bir voxel olmalıdır. Aksi takdirde yinelemeli doldurma algoritması sonucunda iç bölge olarak kestirilmeye çalışılan 3B görüntü dış bölgede sınırsızca büyür. Burada voxel sayısı bir kıstas olarak ortaya konulabilir. Seçtiğimiz 5 cm kenarlı voxel boyutlarıyla ortalama bir insan vücudu 2000 civarı voxelle ifade edilebilir. Güvenli bir eşik olarak 5000 seçilebilir.

Yinelemeli doldurma algoritması bu eşiğin üzerinde voxel çıkartırsa seçilen tohum voxelinin dış bölgeden seçildiğinden emin olunabilir. Ancak yine de tohum voxel seçimi için bir yönteme ihtiyaç vardır zira yukarıdaki eşikleme yöntemiyle seçilen tohum voxeli test edilebilse de bütün noktaları bu şekilde test etmek uygulanabilir değildir. Bu noktada en basit yaklaşım tohum voxeli olarak kabuk görüntüsünün ağırlık merkezini seçmek olabilir. İçbükey katılarda ağırlık merkezi her zaman iç bölgede bulunsa da insan vücudu içbükey değildir. Ancak voxellerin çoğunlukla gövdede toplandıkları göz önüne alınırsa ve gövdenin de içbükey bir yapısı olduğu düşünülürse bu yaklaşımın çok da yanlış olmayacağı görülebilir. Nitekim yürüme, koşma gibi pek çok harekette bu yöntem başarılı sonuç verse de eğilme, çömelme gibi hareketlerin kimi karelerinde hatalı sonuçlar vermektedir. Zira kol ve bacak gibi vücut uzuvları bu duruşlarda ağırlık merkezini iç bölgenin dışına çıkarmaktadır. Çözüm olarak kol ve bacak gibi ince yapıları yok etmeye yarayan morfolojik erozyon yöntemi uygulanabilir. Ancak söz konusu kareye ait uygulanacak 3B görüntü doğal olarak hâlihazırda bulunmadığından bir önceki kareye ait 3B görüntü kullanılabilir. Bu çözüm de eğilme ve çömelme hareketlerinde başarılı olsa da koşup düşme hareketinde hatalar vermektedir. Bunun nedeni ise hareketin hızlı olmasından dolayı kullanılan bir önceki kareye ait verinin şimdiki veriyi temsil etmekten uzak olmasıdır. Buna çözüm olarak ise son iki kareye ait verileri kullanarak yapılan ağırlık merkezi kestirimi kullanılmıştır. Burada da yaklaşım, kısaca deneğin sabit hızla koştuğu varsayımıyla son iki karedeki ağırlık merkezinden son karedeki ağırlık merkezinin kestirilmesi olmuştur.

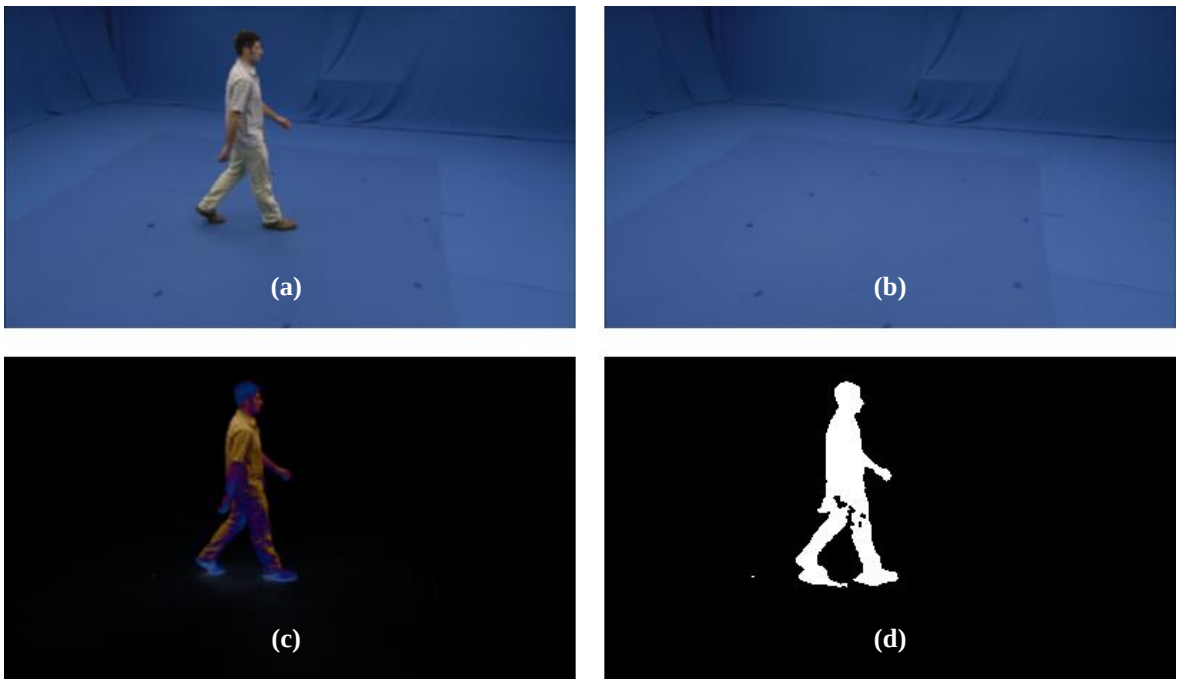
2.3 Görsel Kabuk Algoritması ile 3B Görüntünün Oluşturulması

Yukarıda bahsedilen yöntemlere rağmen veri kümesinde yine de hatalı sonuçlar veren kareler bulunmuştur. Bu nedenle kullanılan veri kümesindeki yüzey verisinin üretiminde de [13] kullanılan daha temel bir yönteme de ihtiyaç duyulmuştur: Görsel Kabuk Algoritması (*Visual Hull Algorithm*) [14, 16, 17].

2.3.1 Siluet Çıkartma İşlemi

Görsel kabuk algoritması siluet çıkartmaya dayalıdır. Siluet çıkartma işlemi istenilen nesnenin görüntü üzerinde kapladığı bölgenin bulunması olarak özetlenebilir [18]. i3DPost veri kümesinin oluşturulduğu stüdyonun mavi renkli arka planı bu işlemi kolaylaştırmak için tasarlanmıştır. Bu görüntü işleminde yapılacak en temel yaklaşım veri kümesinde verilen boş arka plandan gelen görüntüyü çıkartmaktır. Sonrasında elde edilen fark

görüntüsü eşikleme işleminden geçirilir ve ikil (*binary*) siluet görüntüsüne ulaşılır (Şekil 2.6). Morfolojik kapama işlemiyle siluet üzerindeki küçük delikler kapatılabilir.



Şekil 2.6 Yeni görüntünün (a) boş arka plan görüntüsünden (b) çıkarılıp fark görüntüsünün (c) eşiklenmesinden elde edilen siluet görüntüsü (d).

Uygulamada ham görüntü üzerindeki gürültüyü azaltmak için önce yumuşatma işlemi uygulanmıştır. Sonrasında elde edilen fark görüntüsü için eşik değeri olarak 0 – 255 aralığı içinden 18 uygun bulunmuştur. Son olarak uygulanan morfolojik kapama işleminde 5 piksel yarıçaplı dairesel bir maske kullanılmıştır.

2.3.2 Görsel Kabuk Algoritması

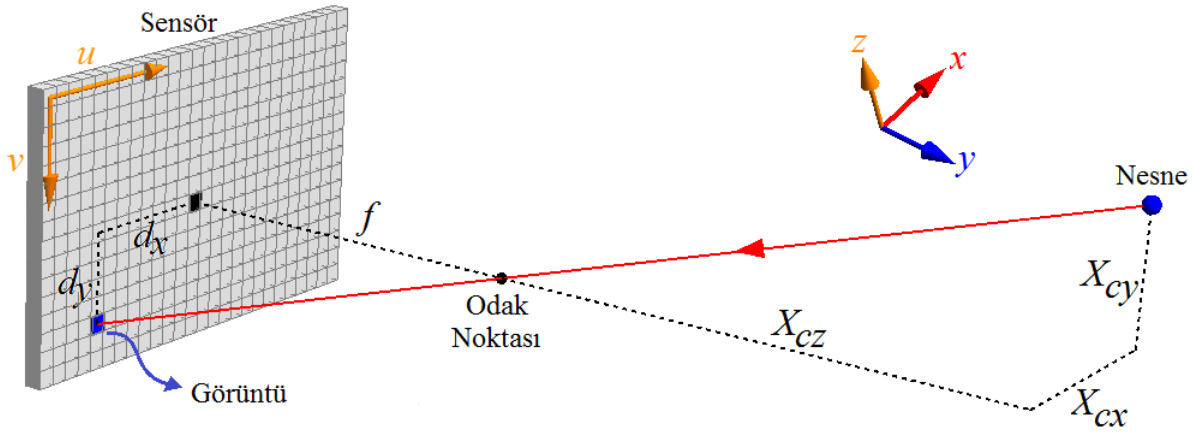
Siluet görüntüsünün ardından görsel kabuk kavramına giriş yapabiliriz. Bilindiği gibi siluet görüntüsündeki beyaz pikseller görüntü uzayındaki nesnenin bulunduğu bölgeyi gösterir. Ancak görüntü üzerindeki herhangi bir nokta (piksel) üç boyutlu gerçek uzayda birden fazla noktaya (voxel) karşılık gelir. Bu noktalar görüntünün alındığı kameranın optik odağından geçen doğrular üzerinde herhangi bir yerde olabilir. Zira kamera görüntüsü uzaklık bilgisini taşımaz. Bu durumda tepe noktası kameranın odak noktası olan ve nesnenin silueti biçiminde genişleyen konik bir bölgeden bahsedebiliriz. Bu bölge, silueti çıkarılan nesnenin gerçek uzayda bulunabileceği noktalar kümesini verecektir. Bu bölge tek bir kamerayla elde edildiğinde fazla anlam taşımaz. Görsel kabuk algoritmasının temel

fikri ise birden fazla kamera kullanarak elde edilen konik bölgeleri kesiştirmektir. Bu sayede nesnenin gerçek uzayda kapladığı hacme yaklaşan bir bölge elde edilebilir [16].

Kesiştirme işlemi elde edilmek istenen veri türüne göre değişkenlik gösterir. 1974'te B. Baumgart detaylı bir 3B yüzey verisi elde etmek için yeni bir yöntem önermiştir [16]. Ancak çalışmamızda hacimsel 3B görüntüsüne ihtiyaç duyduğumuz için M. Potmesil tarafından 1987'de sunulan hacimsel tarama yöntemi uygulanmıştır [17]. Bu yöntemde üç boyutlu gerçek uzayda belirli sıklıklarla alınan örnek noktalar her kamera için görüntü uzayına haritalanır. Noktanın bütün kamera görüntülerindeki izdüşümü o görüntüdeki siluetin içine düşüyorsa söz konusu noktanın görsel kabuk bölgesinin içinde yer aldığına karar verilebilir.

2.3.3 Kamera Görüntüleri için İğne Deliği Modeli

Üç boyutlu gerçek uzayla iki boyutlu görüntü uzayı arasındaki haritalama iğne deliği modeliyle açıklanır [19, 20]. Bu modele göre gerçek uzaydaki bir nesnenin görüntüsü, nesneden yansıyan ve doğrusal ilerleyen ışık ışınının iğne deliğinden (kameranın odağından) geçip karanlık odanın (*camera obscura*) içindeki duvara (kamera sensörü) düşmesiyle oluşur (Şekil 2.7).



Şekil 2.7. Kamera için iğne deliği modeli

Buna göre gerçek uzayla görüntü uzayı arasındaki basit ilişki söyle ifade edilebilir:

$$d_x = f \cdot \boldsymbol{x}_{cx}/\boldsymbol{x}_{cz} \quad (2.1)$$

$$d_y = f \cdot \boldsymbol{x}_{cy}/\boldsymbol{x}_{cz} \quad (2.2)$$

Burada $\boldsymbol{X}_c = (\boldsymbol{x}_{cx}\ ,\ \boldsymbol{x}_{cy}\ ,\ \boldsymbol{x}_{cz})$ vektörü, gerçek uzaydaki bir noktanın orijini optik sistemin odağında yer alan ve yönelimi kamera sensörüne dik bir koordinat sistemiyle ifade edilmiş halidir. d_x ve d_y ise bu noktanın görüntü uzayındaki izdüşümünü ifade eden görüntü koordinatlarıdır. Ancak görüntü uzayının koordinatının orijini de optik sistemin sensör üzerindeki dik izdüşümünde bulunur. f ise odak uzaklığını ifade eder.

Bununla birlikte uygulamada kullanılan koordinat sistemleri hem gerçek uzayda hem de görüntü uzayında yukarıda tanımlananlardan farklıdır. Gerçek uzayda kullanılan koordinat sistemi, bütün kameralar için ortak olmak üzere başlangıç noktası ve yönelim olarak kamerayı değil zemini ya da dış ortamdaki nesneleri referans alır. Bu nedenle iki koordinat sistemi arasında dönme ve öteleme dönüşümleri gereklidir. $\boldsymbol{R}$ dönme matrisi ve $\boldsymbol{t}$ öteleme vektörü olmak üzere:

$$\boldsymbol{X}_c = \boldsymbol{R} \cdot \boldsymbol{X} + \boldsymbol{t}\ . \tag{2.3}$$

Sensör tarafında ise kullanılan koordinat sistemi sağ üst köşeyi referans alır. Bunun için öteleme yeterli olacaktır. Burada şimdiye kadar ihmal ettiğimiz bir başka etki de devreye girer: yarıçapsal bozulma (*radial distortion*). Zira kamerada görüntü iğne deliğinden değil mercek sisteminden geçerek sensör üzerine düşer. Bu nedenle de ışık ışınının gelen açısı ile kırılan açısı eşit değildir. Bu eşitsizlik ancak iki ya da daha yüksek dereceden polinomlarla modellenebilir. c_x ve c_y öteleme miktarı ve k_1 birinci dereceden yarıçapsal bozulma katsayısı olmak üzere:

$$r = \sqrt{d_x^2 + d_y^2} \tag{2.4}$$

$$u = c_x + d_x(1 + k_1\, r) \tag{2.5}$$

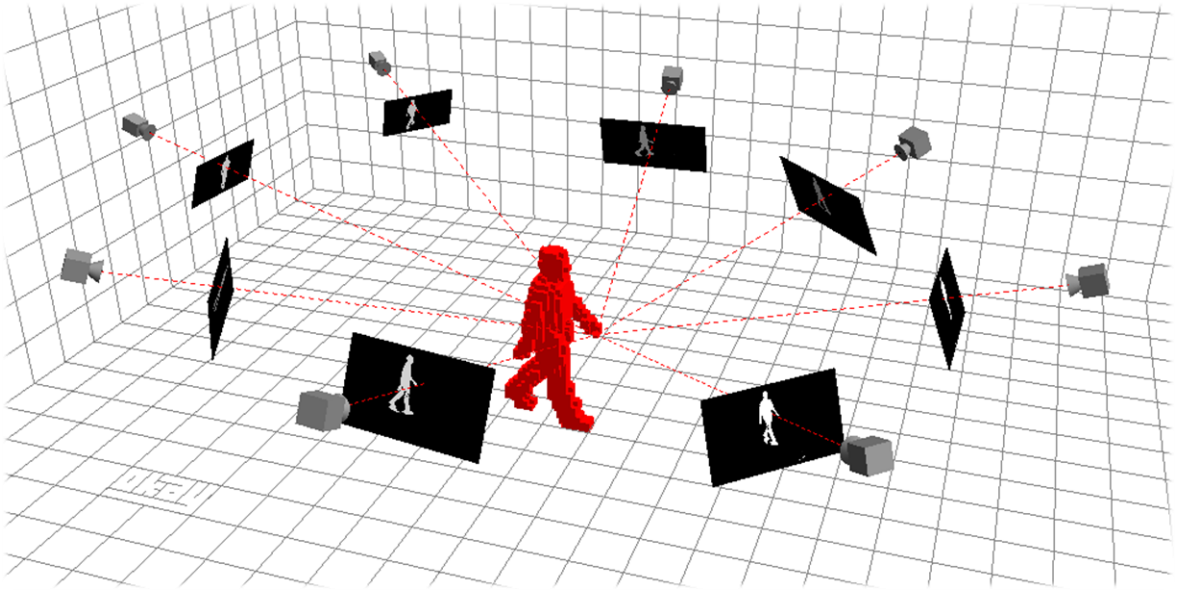
$$v = c_y + d_y(1 + k_1\, r)\ . \tag{2.6}$$

Çalışmamızda kullandığımız i3DPost veri kümesinde [1] kullanılan bütün kameralar önceden ortak bir gerçek uzay koordinat sistemine göre kalibre edilmiştir ve yukarıda geçen parametreler veri kümesinde ascii formatında .txt uzantılı dosyalarda mevcuttur.

2.3.4 Hacimsel Tarama İşlemi

Hacimsel tarama işleminden bir örnek Şekil 2.8'de görülebilir. Şekilde benzetimi yapılmış üç boyutlu gerçek uzay ve üzerinde bulunan kameralar gösterilmiştir. Deneğin sol eli civarındaki bir voxelin hacimsel kabuğun içinde yer alıp almadığı sorgusu yapılmaktadır. Bu amaçla voxelin bulunduğu noktadan kamera odaklarına kesikli olarak ve kırmızı renkle

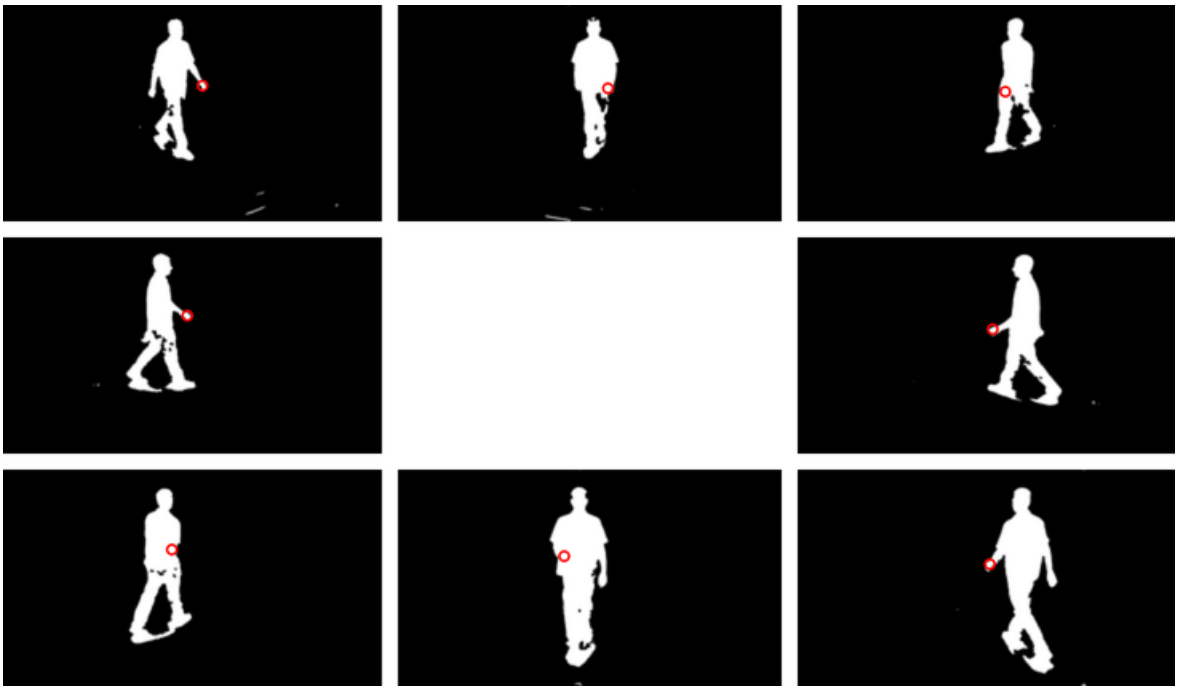
gösterilen doğrular çekilmiştir. Siluet görüntüleri kameraların kalibrasyon değerlerine uygun olarak yerleştirilmiştir ve bu doğruların görüntü düzlemini kestiği noktalar söz konusu voxelin görüntü uzayındaki izdüşümleridir.



Şekil 2.8. Hacimsel tarama işlemi

Bu kesişim noktaları Şekil 2.9'da kırmızı halkalar içinde gösterilmiştir. Görülebileceği gibi kesişim noktalarındaki piksel değerlerinin tamamı da beyazdır. Kısaca bu voxel, görsel kabuğun içindedir ve Şekil 2.8'deki kırmızı küplerle ifade edilen 3B görüntünün içine dâhil edilebilir.

Bütün bir 3B görüntüyü elde etmek için stüdyodaki bütün voxelleri sorgulamak gerekecektir. Ancak bu işlem uzun bir hesaplama süresi aldığından verimli değildir. Bunun yerine sorgulanan voxel bölgesi sınırlanabilir. İnsan vücudunu temsil eden 3B görüntünün 2×2×2 metre ölçülerinde bir küpü aşmayacağı varsayılabilir. Bu durumda bir önceki 3B görüntünün ağırlık merkezini merkez alan bu boyutlarda bir küple sorgulanacak alan sınırlandırılabilir ve hesaplama süresi kısalır.



Şekil 2.9. Sol el civarındaki bir voxelin 8 siluet görüntüsü üzerindeki izdüşümü

3. 3B ZERNİKE MOMENTLERİ

Voxel verisi üç boyutlu uzay içinde belirli bir biçimde dağılmış bir nokta bulutu olarak ele alınabilir. Kuşkusuz bu noktaların hangisinin insan vücudunun hangi bölgesine (gövde, baş, kol ve bacaklar) karşılık geldiği bilinmemektedir. Vücut duruş algılaması, I. Mikic ve diğerlerinin [2] önerdiği gibi bu noktaları eklemli vücut parçalarına kümeleyerek de yapılabilir. Ancak bu ve benzeri teknikler karmaşık hesaplama yöntemleri yüzünden uzun hesaplama süreleri gerektirir. Üstelik temelinde yatan tespit ve takip mekanizması yüzünden sistemin çıktısı oldukça kararsızdır ve sıklıkla takip süreci kaçırıldığından parametreler her defasında yeniden sıfırlanır ve daha uzun süren tespit işlemi tekrar çağırılır.

Bu tez çalışmasında, voxel dağılımının momentlerine dayanan yeni bir yöntem önerilmektedir. Momentler nokta dağılımlarının şekilsel özellikleri yansıtan niceliklerdir. Bir nokta kümesinin momentleri, bu kümenin dağılımının ağırlık merkezi, varyans, çarpıklık ve basıklık gibi şekilsel özelliklerini ifade edebilir. Bu durumda voxel verisi de böyle bir nokta kümesi olarak ele alınabilir ve dağılımının şekilsel özellikleri momentleri üzerinden değerlendirilebilir. Zira her vücut duruşunu ifade eden voxel dağılıma karşılık gelecek bir moment kombinasyonu mevcuttur. Netice olarak sadece voxel verisinin momentleri izlenerek vücut duruşu algılanabilir.

Ancak vücut duruş algılamasını gerçekleştirebilmek için kullanılacak momentlerin bir takım özellikleri olması gereklidir. Öncelikle vücut duruşu, duruş yöneliminden bağımsızdır. Örneğin kuzeye ve batıya yürüyen iki insana ait voxel verilerinin moment kombinasyonları benzer olmalıdır. Bu nedenle aradığımız momentler dikey eksen etrafında yapılan dönme hareketinden bağımsız olmalıdır.

Dahası momentler sistemde büyük yer kaplayan voxel verilerini sıkıştırmalı, daha az sayıda nicelikle söz konusu vücut duruşunu temsil ederek otomatik tanıma işlemini kolaylaştırmalıdır. Bu noktada momentleri, bir fonksiyonun fonksiyonlar uzayı üzerinde belirli baz fonksiyonları cinsinden doğrusal ifadesi olarak düşünebiliriz. Buna göre bu baz fonksiyonları fonksiyon uzayını tarıyorsa (*span*) momentler aracılığıyla ilk fonksiyonu yeniden inşa etmek (*reconstruction*) mümkün olacaktır. Bir başka değişle ilk fonksiyonun taşıdığı bilgi ile moment dizisinin taşıdığı bilgi eşdeğerdir. Ancak uygulamada sonlu

sayıda moment hesaplanabilir ve bu nedenle bilginin temsilinde bir miktar hata bulunur. bu hatayı en aza indirmenin yolu baz fonksiyonlarının birbirine dik olmasıdır (*orthogonality*).

N. Canterakis'in 1999 yılında tanımladığı 3B Zernike momentleri [12] bu iki özelliğe de sahiptir:

- Dikey eksende dönme hareketinden bağımsızlık
- Baz fonksiyonlarının dikliği

3.1 3B Zernike Fonksiyonları

3B Zernike momentlerinin hesaplanmasında kullanılan baz fonksiyonları olan 3B Zernike fonksiyonları birim küre içinde tanımlanmıştır. Üç boyutlu uzayda tamlık (*completeness*) özelliğine haiz olduklarından aşağıda n, l ve m olarak verilen üç ayrı indeksle sıralanırlar [21, 22]:

$$Z_{n,l,m}(\boldsymbol{X}) = \sum_{v=0}^{k} Q_{k,l,v}\, |\boldsymbol{X}|^{2v}\, e_{l,m}(\boldsymbol{X}) \tag{3.1}$$

$\boldsymbol{X} = [x \quad y \quad z]^T$ olmak üzere birim küre içindeki bir noktanın Kartezyen koordinatlarından oluşan bir vektördür. n temel indeks olup 0'dan hesaplanacak en yüksek dereceye kadar artabilir. k ise $(n-l)/2$ ifadesine eşit olup tamsayıdır. İkinci indeks l ise n'den başlayarak 2'şer azalan negatif olmayan bir tamsayıdır. Böylece k'nın her zaman tamsayı kalması da sağlanır. Son olarak m, $-l$ ile l arasında değişen bir tamsayıdır. Denklem temel olarak iki kısımdan oluşur. Birinci kısımda yer alan seri, yarıçap $r = |\boldsymbol{X}|$ cinsinden bir polinomdur. Polinomun katsayıları olan $Q_{k,l,v}$ şöyle hesaplanır:

$$Q_{k,l,v} = \frac{(-1)^k}{2^{2k}} \sqrt{\frac{2l+4k+3}{3}} \binom{2k}{k} (-1)^v \frac{\binom{k}{v}\binom{2(k+l+v)+1}{2k}}{\binom{k+l+v}{k}} \tag{3.2}$$

İkinci kısmı oluşturan $e_{l,m}(\boldsymbol{X})$ fonksiyonlarına harmonik polinomlar adı verilir. Tanımı ise aşağıdaki gibidir[*]:

[*] Harmonik polinomlar, $Y_{l,m}(\theta,\varphi)$ küresel harmonik olmak üzere $e_{l,m}(\boldsymbol{X}) = r^l \cdot Y_{l,m}(\theta,\varphi)$ eşitliğiyle ifade edilir. Burada r, θ ve φ, $\boldsymbol{X}$ vektörünün küresel koordinatlarıdır. Gerek M. Novotni ve R. Klein'ın makalesindeki [21] gerekse bu makaleye referans yapan K. Hosny ve M. Hafez'in makalesindeki [22] harmonik polinom tanımları hatalı olup doğrusu Eş. (3.3)'te verilendir.

$$e_{l,m}(\boldsymbol{X}) = C_{l,m}\left(\frac{ix-y}{2}\right)^{m} z^{l-m} \sum_{\mu=0}^{\left\lfloor\frac{l-m}{2}\right\rfloor} \binom{l}{\mu}\binom{l-\mu}{m+\mu}\left(-\frac{x^2+y^2}{4z^2}\right)^{\mu} \tag{3.3}$$

Burada $C_{l,m}$ normalizasyon katsayısı olup, 3B Zernike fonksiyonlarının ortonormal olmalarını sağlar:

$$C_{l,m} = \frac{\sqrt{(2l+1)\,(l+m)!\;(l-m)!}}{l\,!} \tag{3.4}$$

Negatif m değerleri için normalizasyon katsayısı değişmez: $C_{l,m} = C_{l,-m}$. Ancak harmonik polinomlar negatif m değerleri için aşağıdaki gibi tanımlanır:

$$e_{l,-m}(\boldsymbol{X}) = (-1)^{m}\left(e_{l,m}(\boldsymbol{X})\right)^{*} \tag{3.5}$$

(4.5) eşitliğinde $(\cdot)^{*}$ ifadesi karmaşık eşleniği göstermektedir. Harmonik polinomların bir başka özelliği ise yarıçap ve küresel açılar cinsinden terimlere ayrışabilmesidir:

$$e_{l,m}(\boldsymbol{X}) = r^{l} \cdot Y_{l,m}(\theta,\varphi) \tag{3.6}$$

Burada r, θ ve φ, Kartezyen koordinatları $\boldsymbol{X}$ vektörünce tanımlanan noktanın sırasıyla küresel koordinatlardaki yarıçap, enlem ve boylam değerleridir. Küresel açılarla tanımlanan $Y_{l,m}(\theta,\varphi)$, küresel harmonik olarak isimlendirilir. Küresel harmonikler birim küre üzerinde ortogonaldirler ve fizik, kimya ve bilgisayar grafikleri gibi farklı alanlarda kullanılmaktadır. Şekil 3.1'de ilk 3 dereceye kadar olan küresel harmonikler görülebilir.

3.2 Momentlerin Hesaplanışı ve Yeniden İnşa

Sonuç olarak yine birim küre içinde tanımlanan herhangi bir $f(\boldsymbol{X})$ fonksiyonu için 3B Zernike momentleri yukarıda belirtilen fonksiyonlar üzerinden şu şekilde tanımlanır [22]:

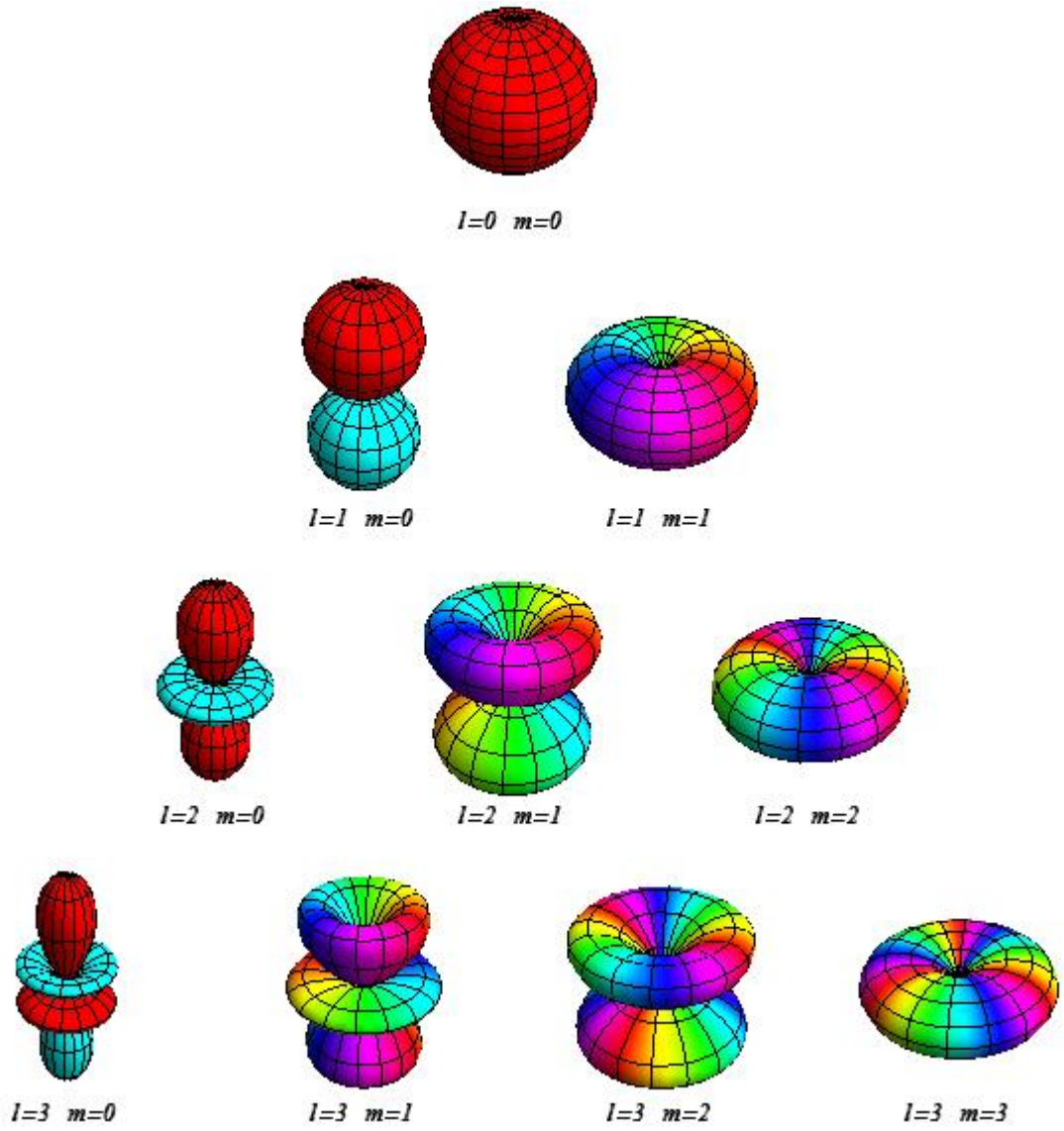
$$\Omega_{n,l,m} = \frac{3}{4\pi}\int_{|\boldsymbol{X}|\leq 1} f(\boldsymbol{X}) \cdot \left(Z_{n,l,m}(\boldsymbol{X})\right)^{*} d\boldsymbol{X} \tag{3.7}$$

Bununla birlikte uygulamada sürekli fonksiyonlarla değil nokta dağılımlarıyla çalışılacağı için moment denklemi bir miktar değişiklik gerektirir. Bu amaçla nokta dağılımı, o noktaları merkez alan dürtü (*impulse*) fonksiyonlarının toplamı olarak ifade edilebilir. Dürtülerin büyüklüğü voxellerin hacmi kadardır. $\boldsymbol{X_i}$'lerden oluşan bir nokta kümesi için, $\boldsymbol{\Delta V}$ birim voxel hacmi olmak üzere nokta dağılım fonksiyonu şöyle olacaktır:

$$f(\boldsymbol{X}) = \sum_{\mathrm{i}} \boldsymbol{\Delta V}\, \delta(\boldsymbol{X} - \boldsymbol{X_i}) \tag{3.8}$$

Eş. 3.8'de tanımlanan dağılım fonksiyonu, momentlerin tanımını veren Eş. 3.7'deki yerine konulursa momentleri tanımlayan yeni eşitlik aşağıdaki gibi olacaktır:

$$\Omega_{n,l,m} = \frac{3\boldsymbol{\Delta V}}{4\pi} \sum_{\mathrm{i}} \left(Z_{n,l,m}(\boldsymbol{X_i}) \right)^{*} \tag{3.9}$$

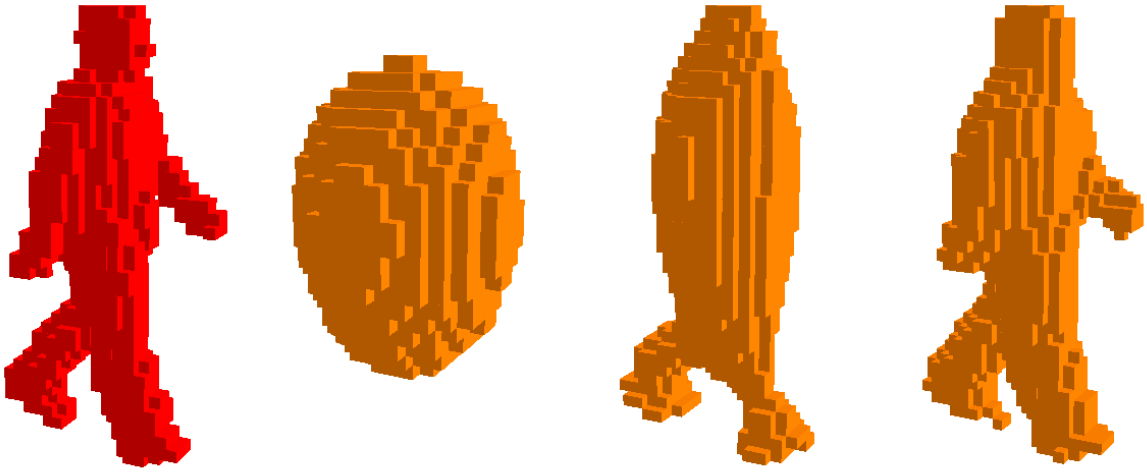


Şekil 3.1. İlk 3 dereceye kadar küresel harmonikler. Yarıçap mutlak değeri ifade ederken renk ise argumanı ifade etmektedir.

Yukarıda belirtildiği gibi momentleri kullanarak baştaki fonksiyonu yeniden inşa etmek (*reconstruction*) mümkündür. 3B Zernike fonksiyonları ortonormal oldukları için yeniden inşa, aynı fonksiyonlarla yapılabilir:

$$\hat{f}(\boldsymbol{X}) = \sum_{n}\sum_{l}\sum_{m} \Omega_{n,l,m}\, Z_{n,l,m}(\boldsymbol{X}) \tag{3.10}$$

Şekil 3.2'de farklı derecelerde elde edilmiş yeniden inşa sonuçları görülmektedir. (a) orijinal fonksiyonu (3B görüntü) ifade eder. Sırasıyla (b), (c) ve (d) ise 5., 10. ve 20. derecelerde üretilen yeniden inşa fonksiyonlarıdır. Bu fonksiyonları sürekli olarak hesaplamak mümkün olmadığından orijinal görüntünün örnekleme sıklığıyla hesaplanmış, çıkan değerler ikil (*binary*) bir görüntü elde etmek için 0.3 değeriyle eşiklenmiştir.



(a) **(b)** **(c)** **(d)**

Şekil 3.2. Farklı dereceler için yeniden inşa sonuçları: orijinal 3B görüntü (a) ve bu görüntünün 5. dereceye (b), 10. dereceye (c) ve 20. dereceye (d) kadar momentlerle yapılan yeniden inşa görüntüleri

3.3 Momentlerin Dönüş ve Simetriye Karşı Değişmezliği

Esasen 3B Zernike momentleri, formüllerden de görüleceği gibi dönme hareketi altında sabit değildir. Aksi takdirde üç boyutlu uzayda tamlık özelliği de gösteremez. Ancak hesaplanan momentlerin mutlak değerleri alınarak bu değişmezlik özelliği sağlanabilir.

$\boldsymbol{X}'$ noktası $\boldsymbol{X}$ noktasının dikey eksende θ açısı kadar saat yönünün tersinde döndürülmüş hali olsun. Bu durumda bu iki noktadaki 3B Zernike fonksiyon değerleri arasındaki ilişki şöyle olacaktır:

$$Z_{n,l,m}(\boldsymbol{X}') = \sum_{v=0}^{k} Q_{k,l,v}\, |\boldsymbol{X}'|^{2v}\, e_{l,m}(\boldsymbol{X}') \tag{3.11}$$

$$= \sum_{v=0}^{k} Q_{k,l,v}\, r'^{\,2v} \left\{ C_{l,m} \left(\frac{ix' - y'}{2} \right)^{m} z'^{\,l-m} \sum_{\mu=0}^{\left\lfloor \frac{l-m}{2} \right\rfloor} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \left(-\frac{x'^{2} + y'^{2}}{4z'^{2}} \right)^{\mu} \right\} \tag{3.12}$$

$$= \sum_{v=0}^{k} Q_{k,l,v}\, r^{2v} \left\{ C_{l,m} \left(\frac{ix - y}{2} \right)^{m} e^{im\theta}\, z^{l-m} \sum_{\mu=0}^{\left\lfloor \frac{l-m}{2} \right\rfloor} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \left(-\frac{x^{2} + y^{2}}{4z^{2}} \right)^{\mu} \right\} \tag{3.13}$$

$$= e^{im\theta}\, Z_{n,l,m}(\boldsymbol{X}) \tag{3.14}$$

$\boldsymbol{A}$ bir nokta kümesi ve $\boldsymbol{A'}$ onun dikey eksende θ açısı kadar saat yönünün tersinde döndürülmüş hali olsun. Şimdi yukarıda gösterilen ilişkiyi de kullanarak bu dağılımların 3B Zernike momentlerinin mutlak değerlerine bakılacak olursa:

$$\left|\Omega_{n,l,m}(\boldsymbol{A'})\right| = \left| \frac{3\boldsymbol{\Delta V}}{4\pi} \sum_{j} \left(Z_{n,l,m}(\boldsymbol{X}'_j) \right)^{*} \right| = \left| \frac{3\boldsymbol{\Delta V}}{4\pi}\, e^{-im\theta} \sum_{j} \left(Z_{n,l,m}(\boldsymbol{X}_j) \right)^{*} \right| = \left|\Omega_{n,l,m}(\boldsymbol{A})\right| \tag{3.15}$$

Böylece bu momentlerin mutlak değerlerinin dönme hareketi altındaki değişmezlikleri ispatlamış olur. Bu özellik vücut duruşu algılamada kilit bir rol oynamaktadır.

Momentlerin mutlak değerlerinin bir diğer özelliği ise ayna simetrisine karşı da değişmez oluşlarıdır. Ancak simetri düzlemi dikey eksenin üzerinden geçmelidir. Bu genel ifadeyi ispatlamak için **y-z** düzlemi gibi belirli bir düzlem için bu durumu ispatlamak yeterlidir. Zira dikey eksenden geçen herhangi bir düzleme göre ayna simetrisi, dikey eksenden geçen belirli bir düzleme göre ayna simetrisi ile dikey eksen etrafında dönme hareketinin birleşimi olarak ele alınabilir. Nitekim dikey eksen etrafında dönme hareketine göre momentlerin değişmezliği yukarıda ispatlanmış bir özelliktir.

O halde $\boldsymbol{X'}$ noktası $\boldsymbol{X}$ noktasının **y-z** düzlemine göre ayna simetrisi olsun. Kuşkusuz bu durumdan sadece $\boldsymbol{X}$ noktasının **x** bileşeni etkilenecek ve negatifine dönüşecektir. 3B Zernike fonksiyonlarında aldıkları değerlere bakılacak olursa:

$$Z_{n,l,m}(\boldsymbol{X'}) = \sum_{v=0}^{k} Q_{k,l,v}\, |\boldsymbol{X'}|^{2v}\, e_{l,m}(\boldsymbol{X'}) \tag{3.16}$$

$$= \sum_{v=0}^{k} Q_{k,l,v} \, |\boldsymbol{X}|^{2v} \left\{ C_{l,m} \left(\frac{i(-x) - y}{2} \right)^{m} z^{l-m} \sum_{\mu=0}^{\left\lfloor \frac{l-m}{2} \right\rfloor} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \left(-\frac{(-x)^2 + y^2}{4z^2} \right)^{\mu} \right\} \quad (3.17)$$

$$= \sum_{v=0}^{k} Q_{k,l,v} \, |\boldsymbol{X}|^{2v} \left\{ C_{l,m} \left(\frac{(ix - y)^*}{2} \right)^{m} z^{l-m} \sum_{\mu=0}^{\left\lfloor \frac{l-m}{2} \right\rfloor} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \left(-\frac{x^2 + y^2}{4z^2} \right)^{\mu} \right\} \quad (3.18)$$

$$= \left(Z_{n,l,m}(\boldsymbol{X}) \right)^{*} \quad (3.19)$$

Bu kez de $\boldsymbol{A}$ bir nokta kümesi ve $\boldsymbol{A}'$ onun **y-z** düzlemine göre simetrisi olsun. Bu durumda 3B Zernike momentlerinin mutlak değerleri arasındaki ilişki şöyle olacaktır:

$$|\Omega_{n,l,m}(\boldsymbol{A}')| = \left| \frac{3\boldsymbol{\Delta V}}{4\pi} \sum_{j} \left(Z_{n,l,m}(\boldsymbol{X}_j') \right)^{*} \right| = \left| \left(\frac{3\boldsymbol{\Delta V}}{4\pi} \sum_{j} \left(Z_{n,l,m}(\boldsymbol{X}_j) \right)^{*} \right)^{*} \right| = |\Omega_{n,l,m}(\boldsymbol{A})| \quad (3.20)$$

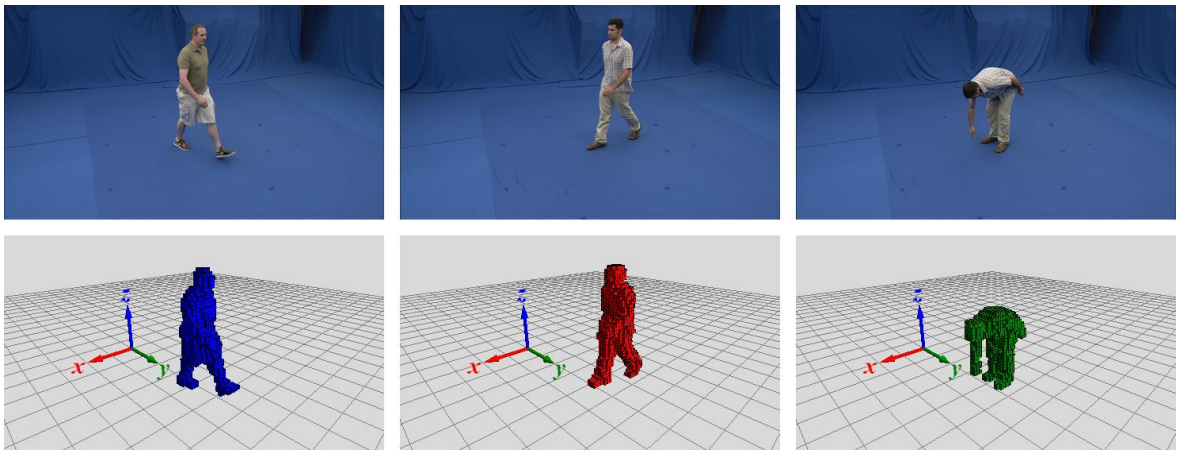
Dikey eksen etrafındaki dönme hareketine karşı değişmezlik özelliğiyle birlikte bu eşitlik dikey eksenden geçen bütün düzlemlere göre simetri altında değişmezliği ispatlar. Uygulamada bunun anlamı şu olacaktır: insan vücudunun da simetrik olduğu göz önüne alınırsa simetrik hareketler benzer moment kombinasyonları verecektir. Örneğin yürüme hareketinde sağ adımla sol adımın atıldığı vücut duruşları moment uzayında eşdeğer olacaktır.

4. İNSAN HAREKETLERİNİN TANINMASI

Bu noktaya kadar veri kümesi kullanılarak 3B görüntüler hesaplanmıştır ve 3B Zernike momentleri ve özellikleri incelenmiştir. 3B Zernike momentlerinin dönme hareketine karşı bağımsız oluşları ve fonksiyonların dikliğinin insan vücut duruşunun algılanmasında kullanılabileceği belirtilmiştir. Burada kastedilen benzer vücut duruşuna sahip ancak farklı yönlere bakan iki insana ait 3B Zernike momentlerinin yakın değerlerde olacağıdır. Yine Bölüm 3.3'te gösterildiği gibi 3B Zernike momentleri ayna simetrisine karşı da bağımsızdır. Bu nedenle vücut duruşları birbirinin simetriği de olabilir.

4.1 Benzer Vücut Duruşlarına Ait Momentlerin Karşılaştırılması

Yukarıda izah edilen çıkarımlar i3DPost veri kümesinden seçilen üç örneğin üzerinde sınanmıştır. Şekil 4.1'de veri kümesinden alınan üç örnek gözükmektedir. Görüntüler aynı kameradan alınmıştır, dolayısıyla benzer duruşlarda olan (a) ve (b)'deki denekler birbirinden farklı yönlere bakmaktadır. Üstelik duruşları birbirinin aynısı değil simetriğidir. (c)'de ise (b)'de de gözüken deneğin farklı bir vücut duruşu bulunmaktadır ve kontrol amacıyla alınmıştır. Dolayısıyla bu üç veriden faydalanarak varsayımların doğruluğunu sınanabilir.



(a) **(b)** **(c)**

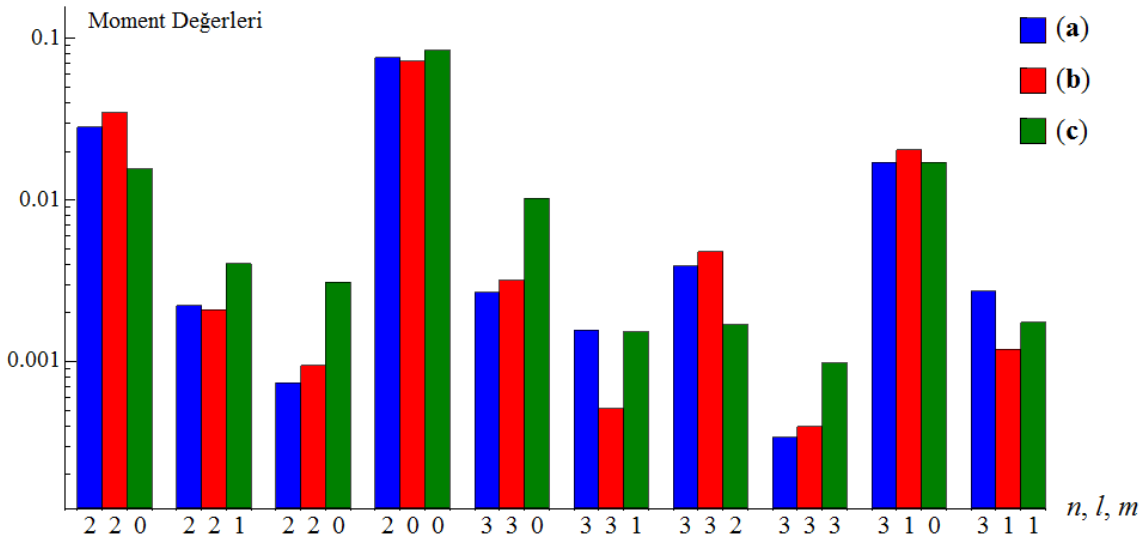
Şekil 4.1. Veri kümesinden alınan üç örnek. İki ayrı deneğe ait olan (a) ve (b)'de vücut duruşları benzer olmakla beraber yönelimleri farklıdır, üstelik duruşlar birbirinin aynısı değil simetriğidir. (c) ise (b) ile aynı deneğe ait olmakla beraber tamamen farklı bir vücut duruşundadır.

Belirtilmesi gereken bir nokta ise 3B Zernike momentlerinin öteleme ve ölçekleme altında bağımsız olmadıklarıdır. Momentleri öteleme karşısında bağımsız kılabilmek için 3B görüntülerin ağırlık merkezlerini orijine taşınabilir. Bu işlem aynı zamanda ağırlık

merkezinin Kartezyen koordinatlarını veren 2. ve 3. momentleri ($n = 1,\ l = 1,\ m = 0$ ve $n = 1,\ l = 1,\ m = 1$) de sıfırlar. İlk moment ($n = 0,\ l = 0,\ m = 0$) ise 3B görüntüdeki nokta sayısıyla orantılı olduğundan vücut duruşunu belirleyen bir özellik vermez. Bu nedenlerle yapılan sınıflandırma işlemlerinde ilk üç moment değerlendirilmemiş, dördüncü momentten itibaren hesaplamalar yapılmıştır.

Ortalama yetişkin insan ölçüleriyle çalışıldığı için ölçekleme şimdilik ihmal edilebilir. Ancak daha ilerde görüleceği gibi hareket algılamada yapılacak standardizasyon işlemleri bu etkiyi de tamamen ortadan kaldıracaktır. Ölçekleme için söylenebilecek bir diğer nokta ise momentlerin daha önce belirtildiği gibi birim küre içinde tanımlı olduklarıdır. Bu nedenle 3B görüntüler ağırlık merkezlerini merkez alan 1 metre yarıçapında bir küre üzerinden ölçeklendirilmiş, nadiren de olsa bu kürenin dışında kalan voxeller ihmal edilmiştir.

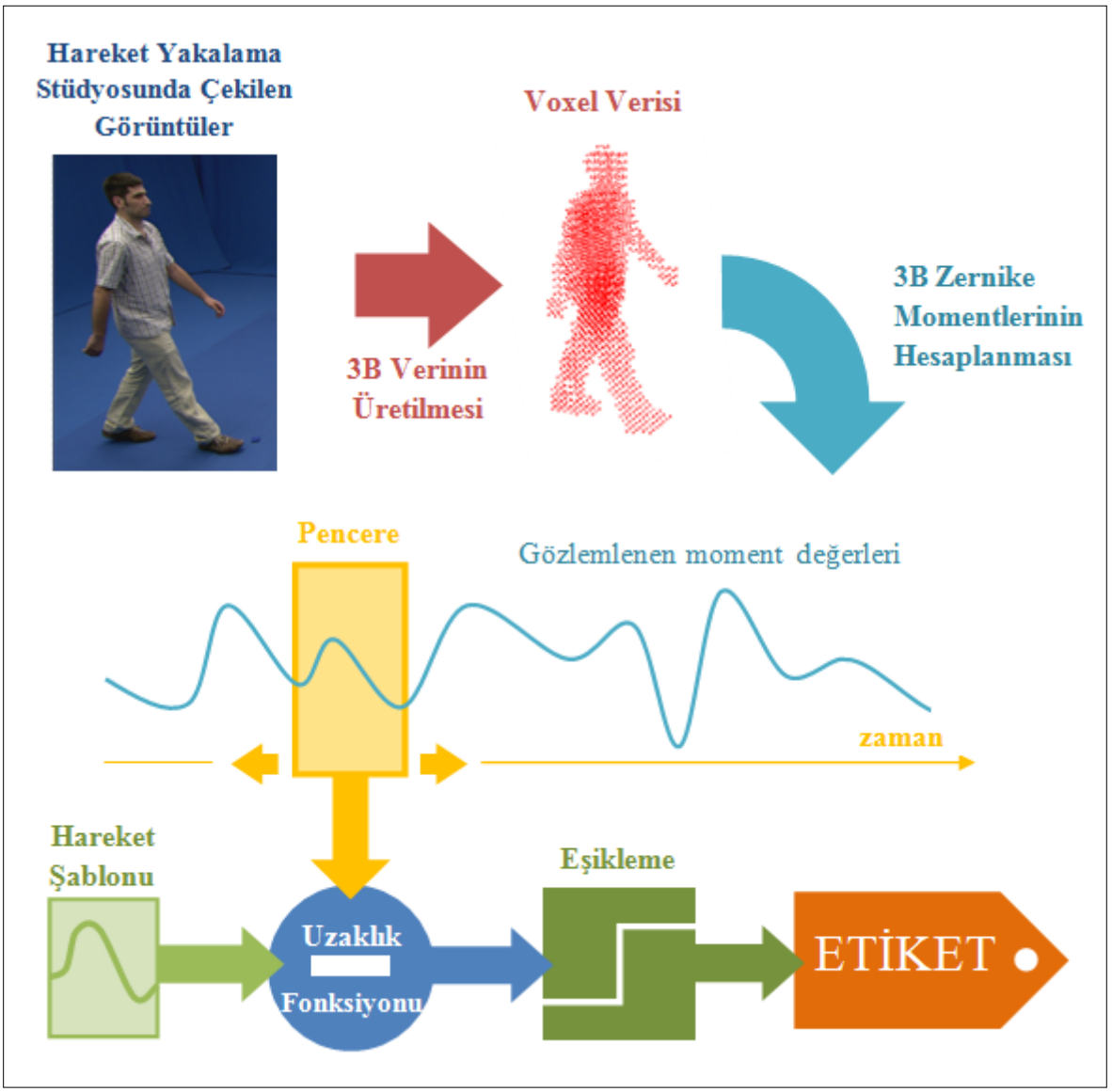
Şekil 4.2'de üçüncü dereceye kadar olan momentler için bu üç örneğe ait değerler gösterilmektedir. Momentlerin bir kısmında benzer vücut duruşlarını ifade eden (a) ve (b) verileri yakın değerler almış (c) onların daha uzağına düşmüştür. Bu durumda sadece bu moment değerlerine bakarak belirli vücut duruşlarını sınıflandırabilmek mümkün olabilir. Bu yönde D. Berjón ve F. Morán'ın [10] yine 3B Zernike momentlerini kullanarak yaptıkları bir çalışma olsa da tezlerini sadece bilgisayarda oluşturulmuş bir benzetim modeli üzerinde test etmişlerdir.



Şekil 4.2. Üç örneğe ait moment değerleri

4.2 Hareket Tanıma İşlemi

Öte yandan hareket tanıma görevi vücut duruşu algılamaktan daha karmaşık bir işlemdir. En önemli fark vücut duruşu anlık bir durumken, hareket belirli bir süre içerisinde gerçekleşir. Bu nedenle momentler anlık değerleri ile değil, belirli bir zaman genişliğindeki bir pencere üzerinden izlenmeli ve pencerenin belirlediği zaman aralığı belirli bir kurala göre etiketlenmelidir. Kural, gözlemlenen momentlerle hareket şablonu arasındaki uygun bir uzaklık fonksiyonu üzerinden tanımlanabilir. Ardından belirli deneysel eşik değerleri kullanarak pencere altında kalan zaman aralığı etiketlenebilir. Bütün çekimi etiketlemek içinse pencere ileriye kaydırılarak işlemler tekrarlanır. Şekil 4.3'teki akış diyagramında bu işlem özetlenmiştir.

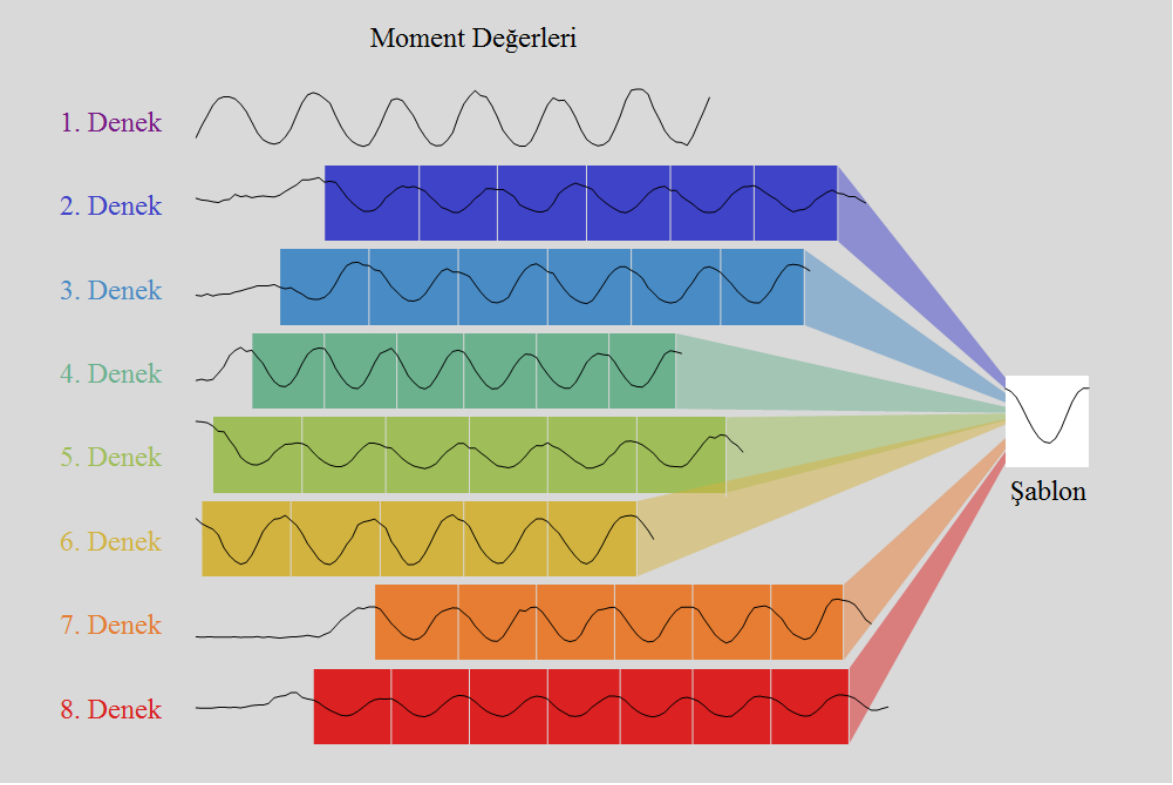


Şekil 4.3. Hareket tanıma sürecine ait akış diyagramı

4.2.1 Hareket Şablonlarının Oluşturulması

Daha önce de belirtildiği gibi i3DPost veri kümesi 8 deneğe ait 7 temel hareket barındırmaktadır. Veri kümesinin küçük boyutta olması sebebiyle verileri öğrenme ve sınama kümelerine ayırmak yerine 'birini-dışarıda-bırak' (*leave-one-out*) yöntemi kullanılmış; her deneği birer kez sınama kümesine koyarak geride kalanlarla öğrenme kümesi oluşturulmuştur. Böylelikle ileri sürülen yöntemin sağlamlığı daha güvenilir bir biçimde denenmiştir.

Sınama kümesi için ayrılacak denek seçildikten sonra geride kalan deneklere ait verilerle harekete ait şablon üretilebilir. Bu amaçla, öncelikle öğrenme verilerinin, içinde söz konusu hareketin tekrarlandığı parçalara ayrılması gerekir. Şekil 4.4'te yürüme hareketine ait şablon çıkarma işlemi görülmektedir. 1. denek sınama kümesindedir. Şekilde sadece 10. momentlerin ($n = 3, l = 3, m = 2$) zamanla değişen dalgabiçimleri çizilmiştir.



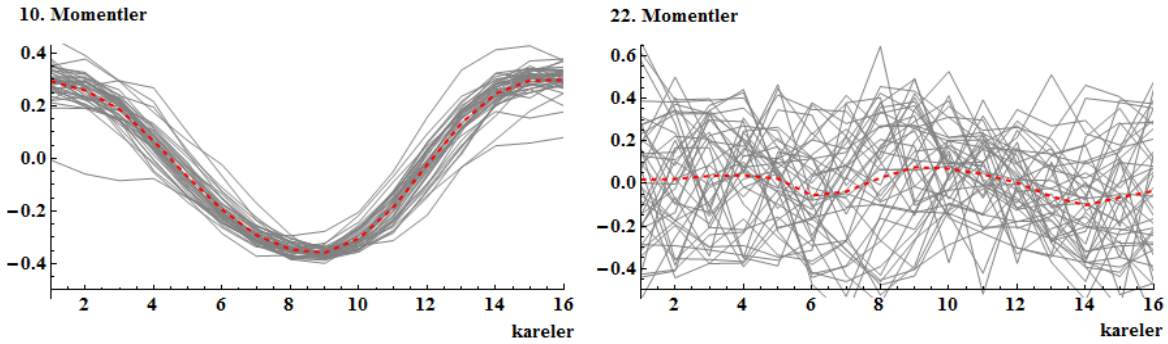
Şekil 4.4. Öğrenme kümesinden hareket şablonunun üretilmesi

Şekilde renkli dikdörtgenler öğrenme kümesi için seçilen parçaları göstermektedir. Doğal olarak bu parçalar eşit uzunlukta değildir. Bu nedenle doğrusal zaman normalizasyonu kullanılarak bütün parçalar ortalama uzunluklarına getirilmiştir. Şablon oluşturmak için en basit yaklaşım parçaların ortalamasını almak olacaktır. Ancak gözlemlerimiz, parçaların

şekilleri birbirine benzemekle beraber yakın ortalama değere ve varyansa sahip olmadıkları yönündedir. Örneğin 1. ve 6. deneklere ait momentlerin salınım genliği yüksekken şekilde görüleceği gibi 5. ve 8. deneklerinkiler düşüktür. Bu nedenle parçaları standardize ederek hepsinin sıfır-ortalama ve birim-varyanslı hale getirilmesine karar verilmiştir. Ardından standardize edilmiş parçaların ortalaması alınarak hareket şablonu oluşturulmuştur.

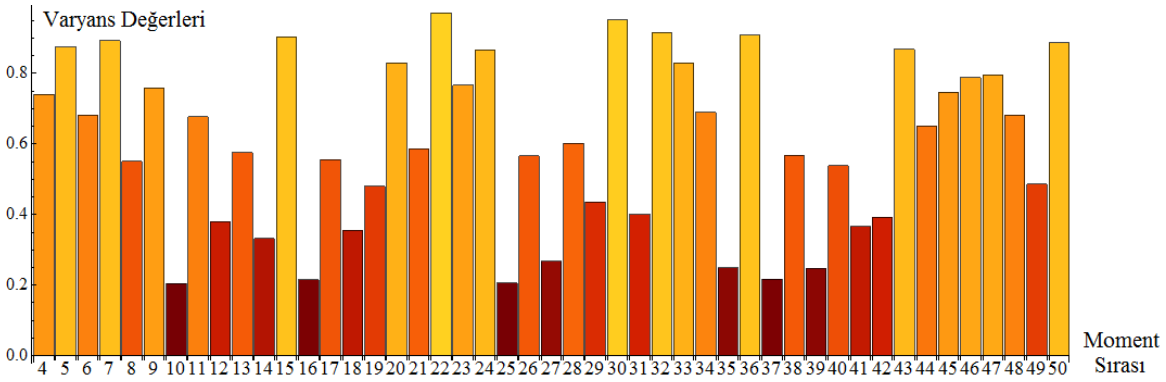
4.2.2 Uzaklık Fonksiyonu

Öğrenme kümesinde gözlemlenen bir diğer olgu ise parçaların bütün momentlerinin aynı ölçüde birbirine benzemedikleridir. Şekil 4.5'te yürüme hareketine ait standardize edilmiş parçaların 10. ($n = 3, l = 3, m = 2$) ve 22. ($n = 4, l = 0, m = 0$) momentleri ve onların ortalama değerleri (kırmızı, kesikli çizgiler) görülebilir.



Şekil 4.5. Standardize edilmiş parçaların farklı momentlerine ait değerler

10. momentler belirli bir örüntüye uyarken 22. momentler rassal bir değişim göstermektedir. Bu durumda 22. moment gibi yüksek varyansa sahip momentlerin hareket algılamada yardımcı olabilecek herhangi bir örüntü taşımadıkları açıktır. Üstelik yukarıda bahsedilen uzaklık fonksiyonunda örüntü taşıyan momentlerle beraber aynı ağırlıkta hesaba katılacaklarsa tanıma performansını düşüreceklerdir. Öğrenme kümesinde yüksek varyansa sahip momentlerin uzaklık fonksiyonundaki ağırlıklarını düşürmek sorunu çözebilir. Bu nedenle uzaklık fonksiyonunda Öklit uzaklığı yerine Mahalanobis uzaklığının kullanılması tercih edilmiş, böylece rastgele değişen momentlerin hareket algılamayı bozmasının önüne geçilmiştir. Mahalanobis uzaklığı için varyans matrisi hesaplanırken yine öğrenme kümesinin standardize edilmiş parçaları kullanılmış ve momentler arası korelasyon, moment fonksiyonlarının dikliği sebebiyle sıfır kabul edilmiştir. Şekil 4.6'da yürüme hareketi için 6. dereceye kadar momentlerin varyans değerleri gösterilmektedir.



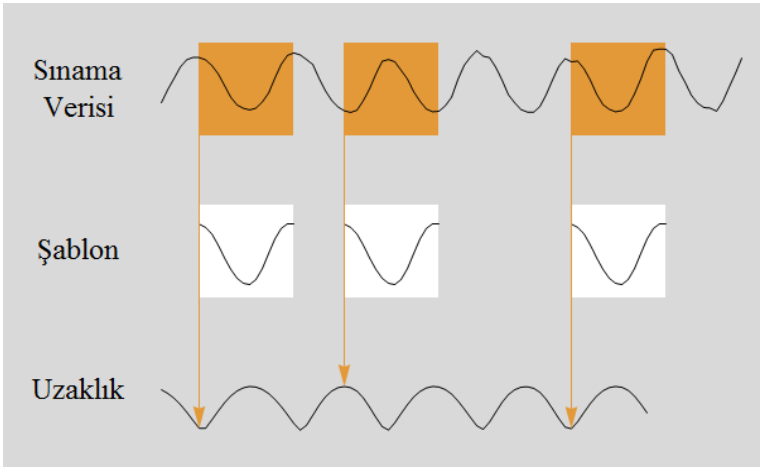
Şekil 4.6. Momentlerin varyansları

Uzaklık fonksiyonuna sokmadan önce, hem ortalama alma işleminden dolayı birim-varyans özelliğini kaybeden hareket şablonu hem de pencere altında kalan test verisi standardize edilmiştir. Son olarak hesaplanan uzaklık, farklı süreler alan hareketler arasında karşılaştırma yapılabilmesi için şablon süresine, farklı moment dereceleri arasında performans karşılaştırması yapılabilmesi için ise kullanılan moment sayısına bölünmüştür. Uzaklık fonksiyonu $u(x,t,v)$; x pencere altında kalan sınama verisi, t şablon ve v varyans değerleri olmak üzere aşağıdaki eşitlikle ifade edilmiştir. T şablon süresi, M kullanılan moment sayısıdır.

$$u(x,t,v) = \sqrt{\sum_{i=1}^{T}\sum_{m=0}^{M} \frac{(x_{i,m}-t_{i,m})^2}{v_m^2} \Big/ (T \times M)} \tag{4.1}$$

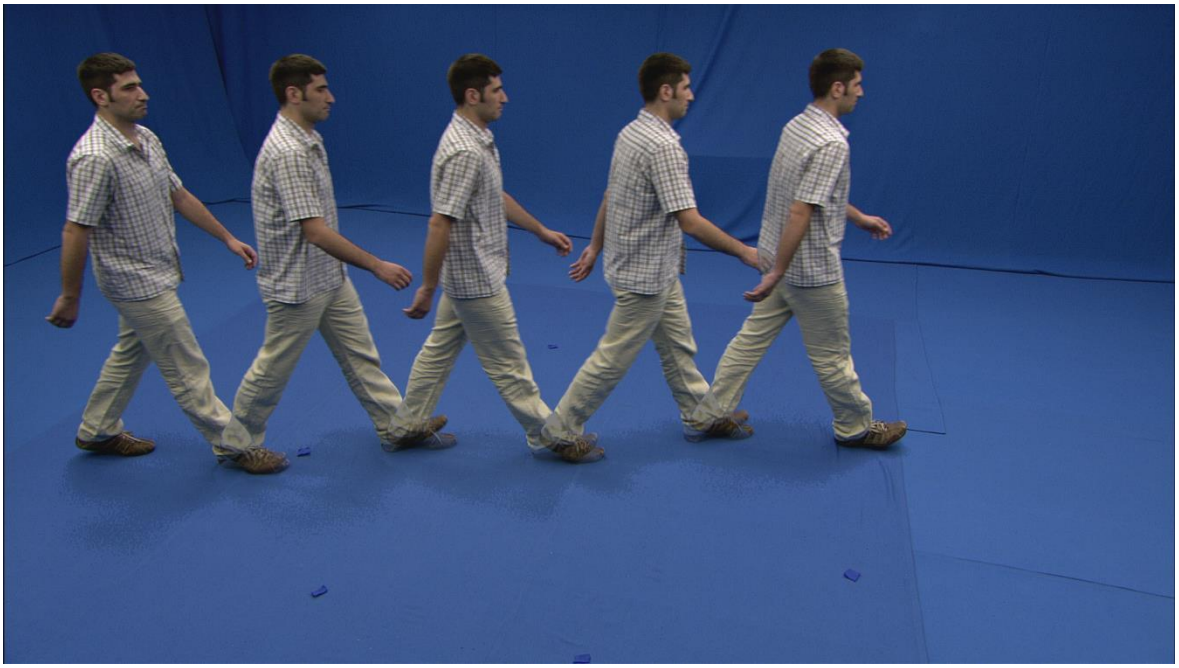
4.2.3 Uzaklık Dizileri

Artık hareketi tanımak için kullanılacak şablona ve uzaklık fonksiyonuna sahip olunduğuna göre 1. deneğe ait sınama verisine geliştirilen yöntem uygulanabilir. Şekil 4.7'de bu işlem görülmektedir. Uzaklık dizisinin hesaplanması için her defasında şablonla aynı uzunluğa sahip olan pencere sınama verisi üzerinde bir adım sağa kaydırılır. Ardından pencere altında kalan sınama verisiyle hareket şablonu standardize edilerek uzaklık fonksiyonuna sokulur ve uzaklık dizisinin yeni terimi hesaplanır.



Şekil 4.7. Uzaklık dizisinin oluşturulması (Basitleştirmek için sınama verisi ve şablonun sadece 10. momentleri çizilmiştir.)

Uzaklık dizisinde görülen 5 adet yerel minimum noktasına karşılık gelen başlangıç çerçeveleri Şekil 4.8'de görülmektedir. Şekilden de görüleceği gibi uzaklık dizisinin işaret ettiği yürüme adımları sağ ya da sol ayakla başlayabilmektedir.



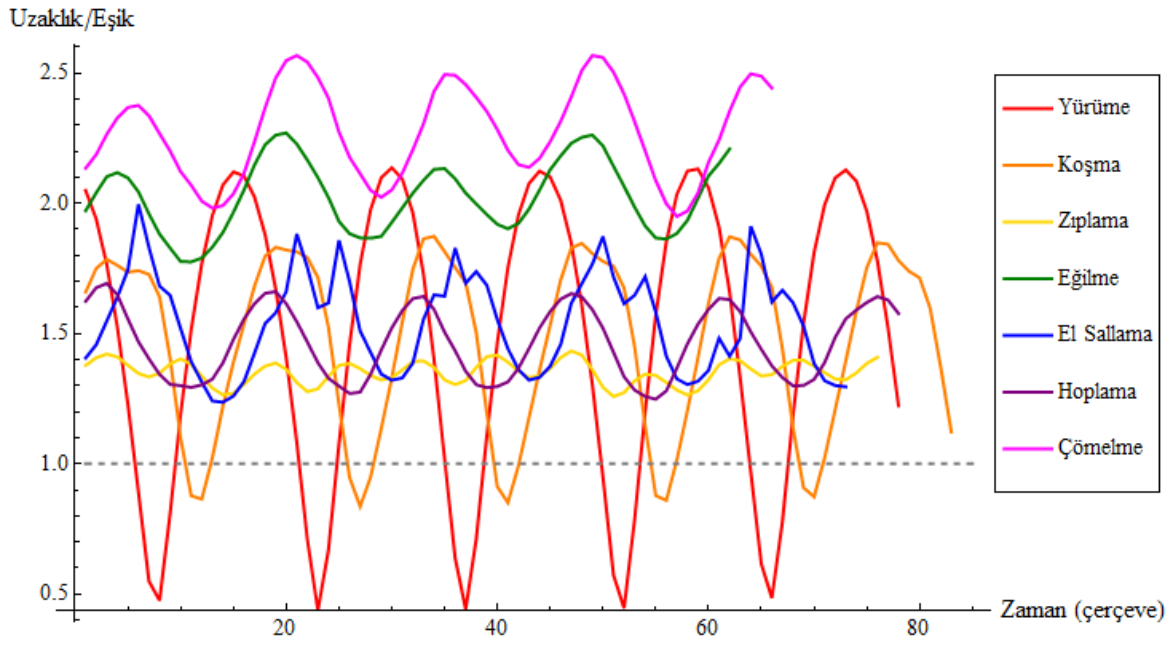
Şekil 4.8. Uzaklık dizisinin yerel minimum noktalarına karşılık gelen başlangıç çerçeveleri

Öğrenme verisindeki hareketin tekrarlandığı parçaların farklı uzunlukta olduğu daha önce belirtilmişti. Bu, kuşkusuz sınama verisinde karşılaşılabilecek aynı hareketin de farklı uzunlukta olabileceği anlamına gelmektedir. Ancak uzaklık dizisini hesaplarken sınama verisindeki hareketin şablonla aynı uzunlukta olduğunu varsayılmıştır. Bu varsayım büyük

ölçüde doğrudur zira bütün hareketlerde öğrenme parçaları da dâhil hareket uzunlukları arasındaki fark ihmal edilecek düzeydedir; bir hareket hariç: el sallama. Burada çözüm olarak şablon eşleme işlemi biraz değiştirilmiştir. Sınama verisinin üzerine şablonla aynı uzunlukta bir pencere koymak yerine şablon uzunluğunun 0.2 ila 2 katı arasında 0.1'lik adımlarla değişen çeşitli uzunlukta pencereler kullanılmıştır. Ardından doğrusal zaman normalizasyonuyla pencereler altındaki veriler şablonla eşit uzunluğa getirilerek uzaklık fonksiyonuna sokulmuştur. Buradan elde edilen uzaklıkların en küçüğü uzaklık dizisinin yeni terimi olarak belirlenmiştir. Bu yöntem diğer hareketlerde kullanmamıştır zira hesaplama süresini tahmin edileceği gibi ciddi anlamda uzatmaktadır.

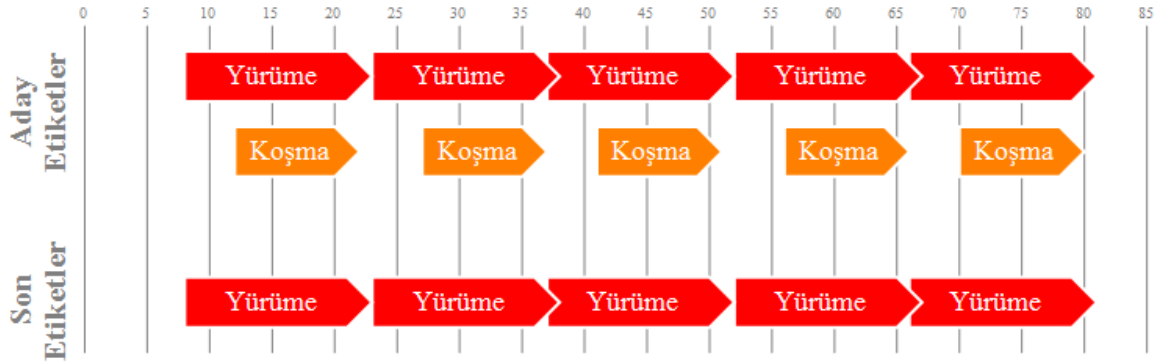
4.2.4 Eşikleme İşlemi ve Etiketleme

Bu noktaya kadar her sınama verisi için 7 temel harekete ait uzaklık dizisi hesaplanmıştır (Şekil 4.9). Bu noktadan itibaren daha önce de belirtildiği gibi (Şekil 4.3) eşikleme işlemi gelecektir. Şekil 4.7'den de gözlemleneceği üzere sınama verisinde söz konusu hareket tekrarlandıkça uzaklık değerleri düşmektedir. Bu yerel minimum noktalarının bulunması için dizi, deneysel olarak belirlenen bir eşikten geçirilir. Elde edilen ikil dizi, birbiriyle temas etmeyen parçalara (*blob*) ayrılır. Ardından parçalar içindeki genel minimumlar bulunur. Bu noktalar aday etiketlerin konumlarını verecektir.



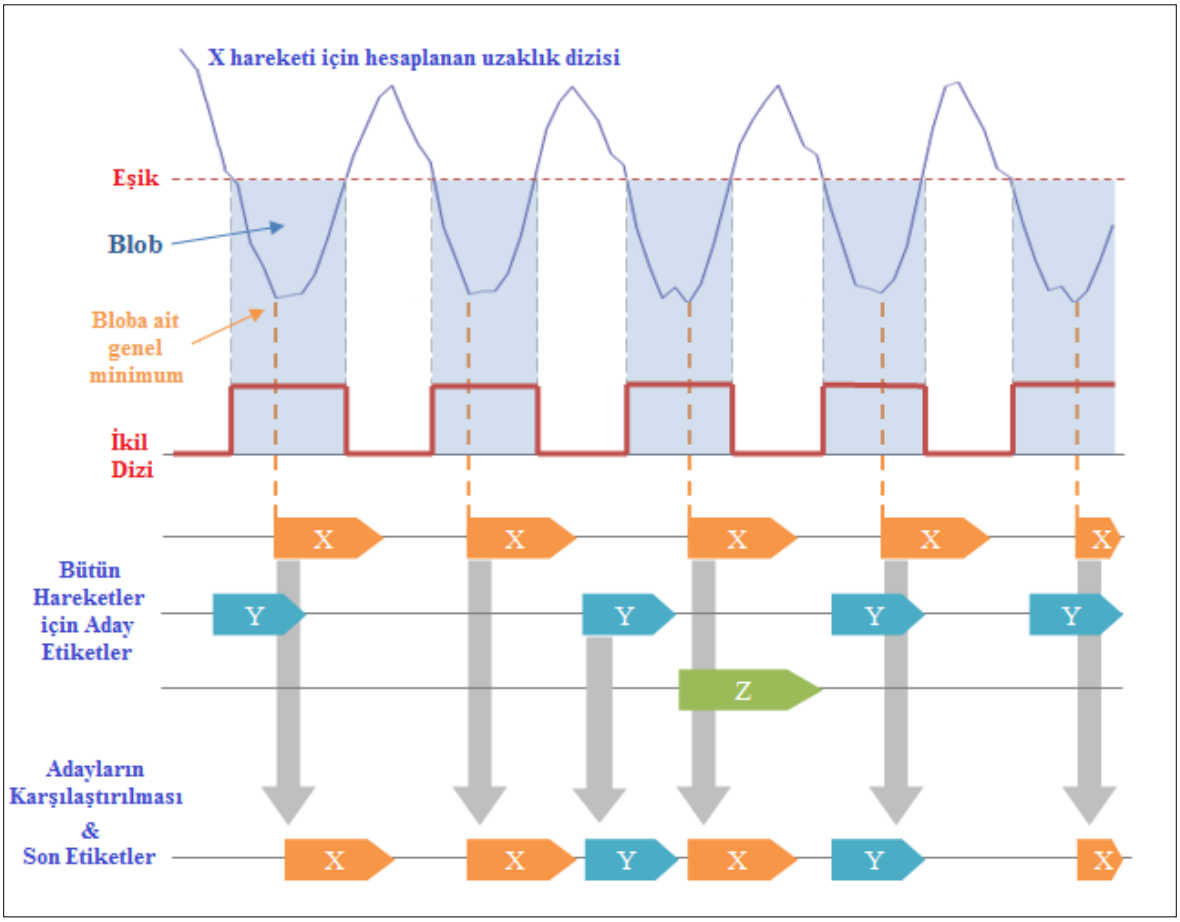
Şekil 4.9. Aynı sınama verisi için hesaplanmış farklı hareketlere ait uzaklık / eşik değerleri

Şekil 4.9'da aynı sınama verisi için hesaplanan ölçeklendirilmiş uzaklık dizisi değerleri görülebilir. Uzaklık verileri harekete ait eşik değerlerine bölünerek ölçeklendirilmiştir. Bu gösterimde yeni ortak eşik değeri 1 olmuştur. 1'in altına düşen parçaların çukur noktaları aday etiketlerin başlangıç noktalarını vermektedir. Görüleceği gibi bu uzaklık değerlerinden beşer adet yürüme ve koşma aday etiketi elde edilmiştir (Şekil 4.10).



Şekil 4.10. Aday ve son etiketler

Ardından birden fazla hareketin aynı anda gerçekleşmeyeceği varsayılarak bu aday etiketlerden kesişenler için bir kıyaslama yapılmıştır. Ölçüt olarak da yerel minimum değerlerinin daha önce kullanılan eşik değerlerine oranı alınmış ve en küçük değere sahip olanı son etiket olarak belirlenmiştir. Şekil 4.9 ve Şekil 4.10'daki örneğe dönülecek olursa yürüme ve koşma hareketlerine ait aday etiketler karşılıklı olarak kesişmektedir. Ancak Şekil 4.9'da görüldüğü üzere yürüme hareketine ait uzaklık/eşik dizisinin çukur değerleri koşmaya ait değerlere göre daha düşüktür. Bu nedenle son etiketler yürüme etiketleri olmuştur. Herhangi bir etiketle kesişmeyen aday etiketi doğrudan son etiket olarak atanmıştır. Eşikleme ve etiketleme algoritmasının temel akışı Sekil 4.11'de görülebilir.



Şekil 4.11. Eşikleme işlemi ve hareket etiketleme

5. SONUÇLAR

i3DPost veri kümesindeki 8 ayrı deneğe ait 10'ar hareketin tamamı da sınama verisi olarak kullanılabildiği için etiketlenebilmiştir. Ardından, yapılan bu otomatik etiketleme işlemi çekim karelerine yerleştirilerek görsel olarak doğrulukları değerlendirilmiştir. Elde edilen sonuçlar Çizelge 5.1'de verilen karışıklık matrisinde (*confusion matrix*) özetlenmiştir.

Çizelge 5.1: Hareketlerin 6. dereceye kadar olan momentlerle etiketlenmesi sonucunda elde edilen karışıklık matrisi

		Etiketlenen Hareket							
		Yürüme	Koşma	Zıplama	Eğilme	El sallama	Hoplama	Çömelme	Etiket Yok
Gerçek Hareket	Yürüme	83	0	0	0	0	0	0	1
	Koşma	0	56	0	0	0	0	0	3
	Zıplama	0	0	33	0	0	1	0	3
	Eğilme	0	0	0	8	0	0	0	0
	El sallama	0	0	0	0	34	0	0	2
	Hoplama	0	0	1	0	0	60	1	5
	Çömelme	0	0	0	0	0	0	14	2
	Hareket Yok	0	9	0	2	3	2	1	–

Çizelge 5.1'de satırlar deneklerin gerçekte yaptıkları hareketleri ifade ederken sütunlar o hareketlerin önerilen yöntemce nasıl etiketlendiğini göstermektedir. Çizelgenin 'Etiket Yok' başlıklı son sütunu etiketlenememiş hareketleri gösterirken çizelgenin 'Hareket Yok' başlıklı son satırı ise denek bir harekette bulunmuyorken basılan hatalı etiketleri göstermektedir. Görülebileceği gibi karıştırılan hareketler birbirine benzer duruşlar içeren hareketlerdir. Ağırlık merkezi takip edilerek bu karışıklıkların bir bölümü kolayca giderilebilir.

6. dereceye kadar olan momentlerin hesaplanmasıyla elde edilen sonuçlar yukarıdaki gibi olmakla beraber daha az moment kullanılarak ve öğrenme kümesinden daha az parça kullanılarak da etiketlemeler yapılmıştır. Bu etiketlemelerin başarılarının ölçülmesi ve karşılaştırma yapılabilmesi amacıyla doğruluk oranları (*accuracy*) hesaplanmıştır (Çizelge 5.2).

Çizelge 5.2: Farklı etiketlemelerin doğruluk değerleri

Parça miktarı (denek başına)	Tamamı	Tamamı	Tamamı	Tamamı	Tamamı	Yarısı	Teki
Moment adedi	2. dereceye kadar (4 adet)	3. dereceye kadar (10 adet)	4. dereceye kadar (19 adet)	5. dereceye kadar (31 adet)	6. dereceye kadar (47 adet)	6. dereceye kadar (47 adet)	6. dereceye kadar (47 adet)
Doğruluk oranı	**%88.40**	**%97.16**	**%97.86**	**%98.64**	**%99.12**	**%98.86**	**%96.66**

Doğruluk oranı hesaplanırken, her hareket için doğru etiketleme sayısı toplam o hareket için kestirilen toplam etiket sayısına bölünmüş; ardından hesaplanan bu doğruluk oranlarının eşit ağırlıkta aritmetik ortalaması alınmıştır. Çizelgeden de görülebileceği gibi 3. dereceye kadar hesaplanan momentlerle %97'nin üzeri bir doğruluk elde etmek mümkünken 2. dereceye kadar olan momentlerde doğruluk oranı sert bir biçimde %88 civarına düşmüştür.

Bu çalışmada kullanılan i3DPost veri kümesiyle başka hareket tanıma çalışmaları da yapılmıştır. A. Iosofidis ve diğerlerinin yaptığı dört farklı çalışmada [3, 4, 5, 6] iki boyutlu görüntüler üzerinden görüntü işleme temelli yöntemler uygulanmış; %94 - %100 arası bir doğruluk oranıyla hareket etiketlemesi yapılabilmiştir. Yine B. Mahasseni ve S. Todorovic'in benzer temelde bir yöntem önerdikleri çalışmalarında [7] yapılan hareket etiketlemesinde %95'in üzerinde bir doğruluk oranı yakalanabilmiştir. B. Holte ve diğerlerince yapılan iki ayrı çalışmada [8, 9] ise diğer çalışmalardan farklı olarak bu çalışmada olduğu gibi üç boyutlu verilerden faydalanılmış, yapılan hareket etiketlemelerinde sırasıyla yaklaşık %92 ve %98 oranında doğruluk oranı elde edilebilmiştir. i3DPost veri kümesiyle yapılan çalışmaların sonucunda elde edilen doğruluk oranları Çizelge 5.3'te verilmiştir.

Çizelge 5.3: i3DPost veri kümesiyle daha önce yapılmış hareket etiketleme çalışmaları ve doğruluk oranları

Çalışma	[3]	[4]	[5]	[6]	[7]	[8]	[9]
Doğruluk Oranı	**%94.87**	**%94.37**	**%96.34**	**%100**	**%95.78**	**%92.19**	**%98.44**

Çizelge 5.3'ten görülebileceği gibi, önerdiğimiz yöntem basitliğine rağmen literatürdeki yöntemlerin çoğundan daha iyi sonuç vermekte, literatürdeki en başarılı yöntemle de neredeyse başa baş performans göstermektedir.

Bu çalışmada, hesaplamalar Intel Core i5 M460 işlemcili, 2.88 GHz hızında ve 4GB RAM hafızasına sahip bir dizüstü bilgisayarda yapılmıştır[*]. 3B verinin üretilmesinden sonra hesaplama süresi olarak en büyük bölümü moment hesaplamaları almıştır. Bu noktada K. Hosny ve M. Hafez [22] tarafından momentleri daha kısa sürelerde hesaplamanın bir yöntemi önerilmişse de bu çalışmada uygulamamıştır. Her bir çerçeve başına[**] hesaplama süreleri Çizelge 5.4'te görülebilir.

Çizelge 5.4: Hesaplama süreleri

Moment Hesaplaması	6. dereceye kadar (47 adet)	10.58 saniye
	5. dereceye kadar (31 adet)	7.57 saniye
	4. dereceye kadar (19 adet)	4.33 saniye
	3. dereceye kadar (10 adet)	2.11 saniye
	2. dereceye kadar (4 adet)	0.85 saniye
Eşikleme ve Etiketleme İşlemi		6.27 milisaniye

5. dereceye kadar olan momentlerle yapılan etiketleme işlemi aşağıda bulunan bağlantıdaki videodan izlenebilir: http://www.ee.hacettepe.edu.tr/~semih/zernike/action.wmv

Elde edilen sonuçlar çeşitli yollarla iyileştirilebilir. Ancak bu yollar hesaplama yükünü de artıracaktır. Birinci yol 3B görüntünün çözünürlüğünü artırarak kullanılacak voxel boyutunu 2 ya da 1 cm'ye indirmek olabilir. Böylelikle 3B görüntünün görüntüsü alınan deneği temsil etme gücü artacaktır. Bir diğer yol ise kullanılan moment sayısını artırmak olabilir. Momentler, üretildikleri 3B görüntü hakkında, kendilerinden üretilecek yeniden inşa görüntüsü kadar bilgi taşımaktadır. 3. Bölümdeki Şekil 3.2'ye tekrar bakılacak olursa, 5. dereceden momentlerle ancak yumurta benzeri bir görüntü elde edilebiliyorken 10. derecede kişinin bacakları kabaca belirmektedir. Ancak 20. derecede insan benzeri bir

[*] Bütün hesaplamalarda ve çizimlerde *Mathematica* programı kullanılmıştır. Üretilen uzun verilerin düzenlenmesinde *textfixer.com* adlı siteden ve *replacetext* programından, görüntü dizileriyle ilgili işlemlerde ise *Irfanview* programından faydalanılmıştır.

[**] Çerçeveler daha önce belirtildiği gibi kameralarca 40 milisaniyede (25 fps) alınmaktadır.

form elde edilebilmiştir. Nitekim Çizelge 5.2'de de kullanılan moment sayısının başarıma etkisi görülebilir. Daha zengin bir veri kümesi kullanmak da hareketlerin öğrenilmesini iyileştireceğinden başarımı yükseltecektir. Öğrenme parçalarının sayısı azaldıkça başarımın da düştüğü Çizelge 5.2'de görülmektedir. Son olarak burada kullanılan doğrusal şablon eşleme yöntemleri yerine kullanılacak daha gelişmiş doğrusal olmayan şablon eşleme yöntemleri, basit doğrusallık varsayımının yol açtığı çarpıklığı giderebilir.

Özetle bu çalışma, 3B Zernike momentlerini kullanmanın, insan hareketlerinin tanınmasında geçerli ve mevcut yöntemlere göre kaba hareketler için hesaplamada daha verimli bir seçenek sağladığını göstermiştir. Önerilen yöntemin en önemli artısı ise karmaşık takip algoritmaları yerine basitçe dalga şekli eşlemeye (*waveform matching*) dayalı olmasıdır.

KAYNAKLAR

[1] University of Surrey, CERTH-ITI, i3DPost Multi-view Human Action Datasets, http://kahlan.eps.surrey.ac.uk/i3dpost_action (**2013**).

[2] Mikic, I., Trivedi, M., Hunter, E., Cosman, P., Articulated Body Posture Estimation from Multi-Camera Voxel Data, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 455-462, **2001.**

[3] Iosofidis, A., Tefas, A., Pitas, I., View-invariant Action Recognition Based on Artificial Neural Networks, *IEEE Transactions on Neural Networks and Learning Systems, vol.23, no.3,* 412-424, **2012.**

[4] Iosofidis, A., Tefas, A., Pitas, I., Nikolaidis, N., Multi-view Human Movement Recognition Based on Fuzzy Distances and Linear Discriminant Analysis, *Computer Vision Image Understanding, vol.116, no.3,* 347-360, **2012.**

[5] Iosofidis, A., Tefas, A., Pitas, I., Multi-view Action Recognition Based on Volumes, Fuzzy Distances and Cluster Discriminant Analysis, *Signal Processing, vol.93, no.3,* 1445-1457, **2013.**

[6] Iosofidis, A., Tefas, A., Pitas, I., Minimum Class Variance Extreme Learning Machine for Human Action Recognition, *IEEE Transactions on Circuits and Systems for Video Technology, vol.23, no.11,* 1968-1979, **2013.**

[7] Mahasseni, B., Todorovic, S., Latent Multitask Learning for View-invariant Action Recognition, in *IEEE International Conference on Computer Vision*, 3-6 December, Sydney, Australia, **2013.**

[8] Holte, M., Moeslund, T., Nikolaidis, N., Pitas, I., 3D Human Action Recognition for Multi-view Camera Systems, *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission,* 342-349, **2011.**

[9] Holte, M., Chakraborty, B., Gonzàlez, J., Moestlund, T., A Local 3D Motion

Descriptor for Multi-view Human Action Recognition from 4D Spatio-temporal Interest Points, *IEEE Journal of Selected Topics in Signal Processing, vol.6, no.5,* 553-565, **2012.**

[10] Berjón, D., Morán, F., Fast Human Pose Estimation Using 3D Zernike Descriptors, *IS&T/SPIE Electronic Imaging, Vol. 8290,* 82900K-1-6, **2012.**

[11] Holte, M., B., Trivedi, M., M., Moeslund, T., B., Human Pose Estimation and Activity Recognition From Multi-View Videos: Comparative Explorations of Recent Developmants, *IEEE Journal of Selected Topics in Signal Processing Vol.6 No.5,* **2012.**

[12] Canterakis, N., 3D Zernike Moments and Zernike Affine Invariants for 3D Image Analysis and Recognition, *11th Scandinavian Conference on Image Analysis,* 7-11 July, Kangerlusssuaq, Greenland, 85-93, **1999.**

[13] Gkalelis, N., Kim, H., Hilton, A., Nikolaidis, N., Pitas, I., i3DPost Multi-view and 3D Human Action/Interaction Database, *Proceedings of the Conference for Visual Media Production,* 159-168, **2009.**

[14] Starck, J., Hilton, A., Surface Capture for Performance-Based Animation, *IEEE Computer Graphics and Applications 27 (3),* 21-31, **2007.**

[15] Shih, F., Y., Image Processing and Mathematical Morphology, CRC Press, **2009.**

[16] Baumgart, B., G., Geometric Modeling for Computer Vision, Stanford University, Stanford, **1974.**

[17] Potmesil, M., Generating Octree Models of 3D Objects from Their Silhouettes in a Sequence of Images, *Computer Vision, Graphics, and Image Processing,* 1-29, **1987.**

[18] Jähne, B., Digital Image Processing, Springer, **2002.**

[19] Cyganek, B., Siebert, P., An Introduction to 3D Computer Vision Techniques

and Algorithms, J. Wiley & Sons, Chichester, **2009.**

[20] Hartley, R., Zisserman, A., Multiple View Geometry in Computer Vision, Cambridge University Press, **2004.**

[21] Novotni, M., Klein, R., Shape Retrieval Using 3D Zernike Descriptors, *Computer Aided Design,* vol. 36, no. 11, 1047-1062, **2004.**

[22] Hosny, K., Hafez, M., An Algorithm for Fast Computation of 3D Zernike Moments for Volumetric Images, *Mathematical Problems in Engineering*, **2012.**

Human Action Recognition Using 3D Zernike Moments

Okay Arık and A.Semih Bingöl
Hacettepe University
Department of Electrical and Electronics Engineering
Ankara, Turkey
semih@ee.hacettepe.edu.tr

***Abstract*— In this work, 3D Zernike moments have been used to classify 7 basic coarse human actions in markerless 3D video sequences. The time trajectories of the Zernike moments of the moving subject have been taken as features. Even though Zernike moment orders of about 15 to 20 are required to characterize and/or reconstruct a general 3D image with reasonable fidelity, it has been found that fewer number of moments are sufficient for satisfactory action classification, due to the accumulative nature of video data. In our work, we have obtained greater than 95% recognition accuracy using as low as 3rd order Zernike moments, over the 7 basic actions considered. Recognition accuracy increased to more than 98% with 5th order moments.**

***Index Terms*—Action Recognition, Pose estimation, Zernike Moments**

I. Introduction

In recent years, human motion recognition has begun to attract a lot of attention as a more sophisticated mode of human-machine interaction. As human body pose forms a three dimensional shape, reliable machine recognition needs three dimensional input data which can be obtained using 3D laser scanners, LIDARs, multi-camera motion capture studios, etc.

In this work, we have used the i3DPost Multi-View Human Action Datasets* to classify 7 basic human actions. The datasets have been generated using 8 cameras in a studio which has blue colored walls and floor in order to ease silhouette extraction in markerless motion capture [1]. The datasets consist of 10 distinct motion videos for each subject and a total of 8 subjects. The motion videos have been captured at 25 fps with 1920×1080 resolution. They include different actions such as walking, running, jumping, bending, hand waving, jumping in place, sitting and some combinations of these. Out of these 10 motions, we concentrated only on those containing the 7 basic actions, namely walking, running, jumping, bending, hand waving, jumping in place, and sitting. Action combinations have not been used for training. However, we did include them in the test set in order to make optimum use of the small database. The subjects have different body sizes and genders (6 men and 2 women).

To obtain 3D volumetric data (voxel data), we used the surface information contained in the dataset. Our voxel data consist of a set of cubes (voxels) having 5 centimeter edges. Higher resolution (smaller voxels) would have led to excessive information and extended calculation time. On the other hand, lower resolution (larger voxels) would have been insufficient to represent the geometric form of the body shape. Voxel data can also be interpreted as a distribution of a set of points in 3D space. Obviously, it is unknown which point of the set belongs to which part of the body like torso, head, arms or legs. Pose and action recognition can be achieved by clustering these points into body parts as described by Mikic et al. [2]. However, these techniques usually involve very heavy computational load. Moreover, due to the detection and tracking mechanisms necessary for action recognition, the output may become unstable and might need parameter resets and re-detection.

In our work, we propose an alternative method for coarse human pose and action recognition based on 3D Zernike moments of the point distribution of the voxel data. Moment is a quantitative measure of the shape of a set of points and distributions can be characterized by their moments. This implies that the geometric shape of human body in any pose corresponds to a particular moment combination. Therefore, just by monitoring moments, human pose estimation could be possible.

II. 3D Zernike Functions and Moments

The use of moment invariants for 3D pose estimation is not new and has been proposed as early as 1991 [3]. However use of Zernike moments offers some important advantages because they can be represented as the coefficients of an orthogonal expansion, and their norms are invariant under rotation around vertical axis.

Human pose is invariant with respect to the direction of the body. For instance, moment combinations of voxel data which belong to two walking men towards north and west must be nearly same. For this reason, the selected features must be invariant under rotation around the vertical axis. Moreover, the features should be efficient in that they must compress the huge amount of voxel data into as small a feature vector as possible while still retaining the shape information. Ideally, if reconstruction functions of moments form a complete set, any

* The datasets can be downloaded free of charge for academic purposes from: http://kahlan.eps.surrey.ac.uk/i3dpost_action

distribution can be reconstructed from its moments. In practice, a finite number of moments are used to approximate the distribution. In order to minimize the approximation error, reconstruction functions must be orthogonal to each other. Hence, use of 3D Zernike moments seems appropriate for human pose and action recognition.

3D Zernike moments are defined as projections of the 3D image onto a set of polynomials, which are called Zernike polynomials (or Zernike functions) and which have been defined by Canterakis [4]. The Zernike functions are defined on a unit ball and have 3 indices n, l, and m as given below:

$$Z_{n,l,m}(\boldsymbol{X}) = \sum_{v=0}^{k} Q_{k,l,v} \, |\boldsymbol{X}|^{2v} \, e_{l,m}(\boldsymbol{X}) \qquad (1)$$

where $k = (n - l)/2$. Here, n is the primal index and varies from zero to maximum moment order. The second index l is a nonnegative integer such that it is less than or equal to n. Furthermore, $n - l$ is always even, so k is always a nonnegative integer. Finally, index m varies from $-l$ to l. In Eq. (1), $\boldsymbol{X}$ is the vector of Cartesian coordinates x, y and z.

The polynomials $e_{l,m}(\boldsymbol{X})$ are called harmonic polynomials and are defined as:

$$e_{l,m}(\boldsymbol{X}) = C_{l,m} \left(\frac{ix - y}{2}\right)^{m} z^{l-m} \sum_{\mu=0}^{\left\lfloor \frac{l-m}{2} \right\rfloor} \binom{l}{\mu} \binom{l-\mu}{m+\mu} \left(-\frac{x^2 + y^2}{4z^2}\right)^{\mu} \qquad (2)$$

where r is the radius, i.e., $r = |\boldsymbol{X}|$ and $C_{l,m}$ is a normalization factor given by:

$$C_{l,m} = \frac{\sqrt{(2l+1)\,(l+m)!\,(l-m)!}}{l!} \qquad (3)$$

For negative values of m, the normalization factor is identical, i.e., $C_{l,m} = C_{l,-m}$ while harmonic polynomials for negative m are defined as:

$$e_{l,-m}(\boldsymbol{X}) = (-1)^{m} \left(e_{l,m}(\boldsymbol{X})\right)^{*} \qquad (4)$$

where (.)* denotes the complex conjugate.

The harmonic polynomials $e_{l,m}(\boldsymbol{X})$ given in Eq. (2) are derived from the spherical harmonics $\boldsymbol{Y}_{l,m}(\theta,\varphi)$ as

$$e_{l,m}(\boldsymbol{X}) = r^{l} \boldsymbol{Y}_{l,m}(\theta,\varphi). \qquad (5)$$

The spherical harmonics, shown here in polar coordinates θ and φ in accordance with common practice, form an orthogonal set within the unit ball and have important applications in physics, chemistry, seismology and computer graphics [5]. Shapes of spherical harmonics up to third degree have been plotted in Figure 1. The harmonic polynomials of Eq. (2) have been expressed in Cartesian coordinates x, y and z.

The coefficients $Q_{k,l,v}$ in Eq. (1) are chosen to guarantee orthonormality within the unit ball:

$$Q_{k,l,v} = \frac{(-1)^{k}}{2^{2k}} \sqrt{\frac{2l + 4k + 3}{3}} \binom{2k}{k} (-1)^{v} \frac{\binom{k}{v}\binom{2(k+l+v)+1}{2k}}{\binom{k+l+v}{k}} \qquad (6)$$

The 3D Zernike moments of a 3D function $f(\boldsymbol{X})$ are defined as the projection of that function onto the 3D Zernike functions:

$$\Omega_{n,l,m} = \frac{3}{4\pi} \int_{|\boldsymbol{X}| \le 1} f(\boldsymbol{X}) \cdot \left(Z_{n,l,m}(\boldsymbol{X})\right)^{*} d\boldsymbol{X} \qquad (7)$$

The Zernike functions form an orthonormal basis within the unit ball (or sphere) and, conversely, the 3D function $f(\boldsymbol{X})$ can be reconstructed from its Zernike moments. Since a finite number of moments have to be used in practice, the reconstructed function will be an approximation to the original:

$$\hat{f}(\boldsymbol{X}) = \sum_{n} \sum_{l} \sum_{m} \Omega_{n,l,m} \, Z_{n,l,m}(\boldsymbol{X}) \qquad (8)$$

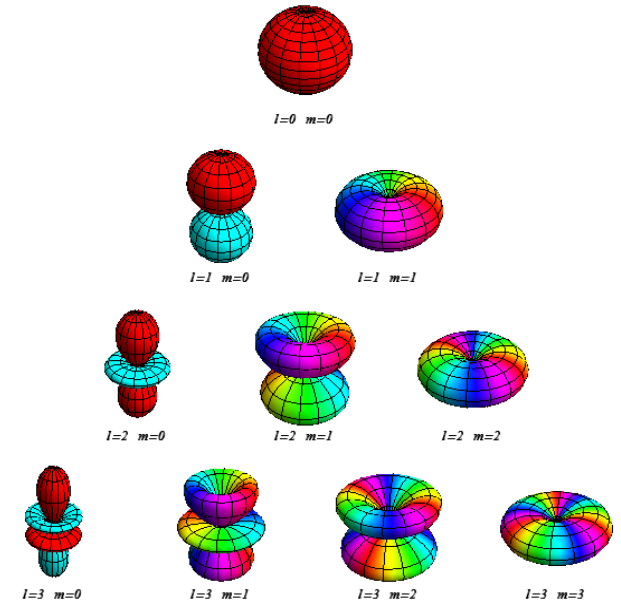


Fig. 1 Spherical harmonics up to 3rd degree. Radius shows the magnitude and color shows the phase of the spherical harmonics.

Computation of Zernike moments involves a rather hefty load. Novotni and Klein [6] describe a method to obtain Zernike moments from geometric moments, details of which are not elaborated here. In addition, although we have not used their method in this work, we must note here that in a recent paper Hosny and Hafez have proposed a faster method for moment calculations which substantially reduces the computation time [7].

For action recognition we use discrete binary 3D images for which Zernike moment computations are somewhat simplified:

$$\Omega_{n,l,m} = \frac{3\Delta V}{4\pi} \sum_{i} \left(Z_{n,l,m}(\boldsymbol{X}_i)\right)^{*} \qquad (9)$$

where the summation runs over all voxels $\boldsymbol{X}_i$ that belong to the body and ΔV is the voxel size.

If $\boldsymbol{X}'$ denotes the rotated version of a point $\boldsymbol{X}$ within the unit ball by angle θ around the z-axis, it can easily be shown

that the Zernike functions at these two points will be related by:

$$Z_{n,l,m}(\boldsymbol{X}') = e^{im\theta}\, Z_{n,l,m}(\boldsymbol{X}) \quad . \tag{10}$$

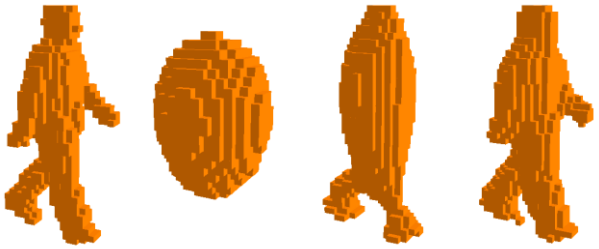
Hence, if $\boldsymbol{A}$ is a cloud of points defining a particular shape in the unit ball and $\boldsymbol{A}'$ is its rotated version around the z-axis, the 3D Zernike moments of these distributions will also be related by:

$$\begin{aligned} |\Omega_{n,l,m}(\boldsymbol{A}')| &= \left| \frac{3\Delta V}{4\pi} \sum_j \left(Z_{n,l,m}(\boldsymbol{X}_j') \right)^* \right| \\ &= \left| \frac{3\Delta V}{4\pi}\, e^{-im\theta} \sum_j \left(Z_{n,l,m}(\boldsymbol{X}_j) \right)^* \right| = |\Omega_{n,l,m}(\boldsymbol{A})| \end{aligned} \tag{11}$$

which proves that the magnitudes of these moments are invariant with respect to rotation around the z-axis. In a similar fashion, it can be shown that they are also invariant for mirror image shapes, as long as the plane of symmetry includes the z-axis. These properties turn out to be very useful in action classification. For example, in walking action, a step forward with the right leg ideally gives the same moment values as a step with the left leg, provided the origin is successfully identified.

III. Human Action Recognition

Equations (7) and (8) imply that Zernike moments are essentially the coefficients of an orthogonal series expansion of a 3D image with the Zernike functions as the basis functions. In order to give an idea about the quality of the reconstruction using a finite series, Figure 2 (a) shows the voxelized original 3D still image of a walking subject and (b), (c) and (d) shows its reconstructions using Zernike moments up to orders $n = 5$, 10 and 20, respectively.



(a) (b) (c) (d)

Fig. 2 (a) Original 3D image and its reconstructions using Zernike moments up to order (b) 5, (c) 10 and (d) 20

This figure suggests that moments up to about order 20 are needed to reconstruct 3D still images with reasonable fidelity. This result agrees with the results obtained by Novotni and Klein [6].

Berjón and Morán [8] have used Zernike moments for human pose estimation based on an articulated model and they also report satisfactory results with moment order $n = 20$ (121 different moments). However, they have not run any tests with real human subjects and have only considered pose estimation.

Action recognition is a more complex task than pose estimation. The foremost difference is the fact that while pose is an instant state, action involves duration. Hence, in order to perform action recognition using moments, we need to monitor moment values in a window of certain length and then label the action in this interval according to a rule. The rule can be formulated using a suitable distance function between the observed moment trajectories and the moment trajectory templates of the action. Then, by using some empirical thresholds, the action inside the window can be labeled. To label the whole sequence, the window is shifted along the time axis and the process is repeated. The procedure that we have used in our work is summarized in Figure 3.

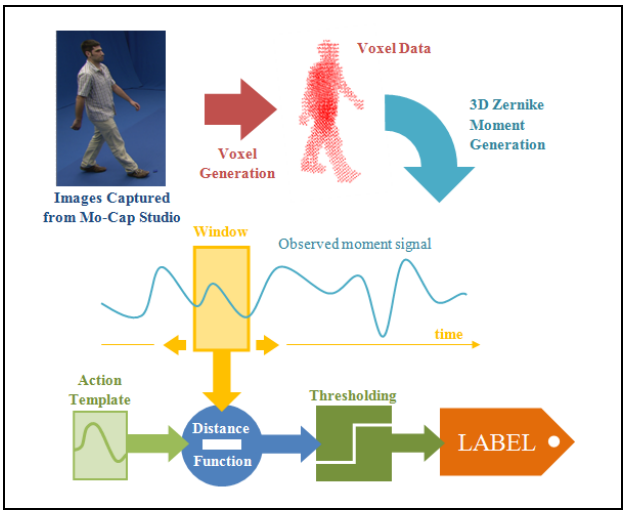


Fig. 3 Flow diagram of the proposed action recognition scheme

The amount of preprocessing before moment calculations is minimal: First, the centroid of the 3D image is calculated and becomes the origin. The unit ball, in which all moment calculations are carried out, is then defined as a sphere of one meter radius having the same origin. Rarely, some voxels belonging to the subject may lie outside the unit ball in which case they are simply neglected.

As mentioned above, our dataset consists of 7 motion videos from 8 subjects. Due to the small size of the dataset, we have not divided it into separate training and test sets but used the leave-one-out method in order to obtain more reliable results. That is, we picked all 7 motion videos belonging to one subject as the test set, and lumped all other subjects into the training set. After classifying the test set, we then repeated the procedure for each subject.

After picking the test subject, we focus on the remaining 7 subjects and generate a reference template for the action under consideration. We begin by manually marking the beginning and ending frames of the action within the motion videos. Usually, there are many such segments within a motion video and all of them are marked. Figure 4 depicts the generation of template arrays for walking action assuming that Man1 is the test subject. Each segment in this figure corresponds to a single step of the subject. Only the trajectories of the 10th

moment have been plotted (n = 3, l = 3, m = 2) in Figure 4. The same procedure is repeated for all moment orders in order to generate the full walking action template. The trajectories are computed every frame at a rate of 25 fps. Colored rectangles show selected segments from the training set. Clearly these segments will not have the same length. We have chosen to use linear time normalization to normalize the segments to their average duration.

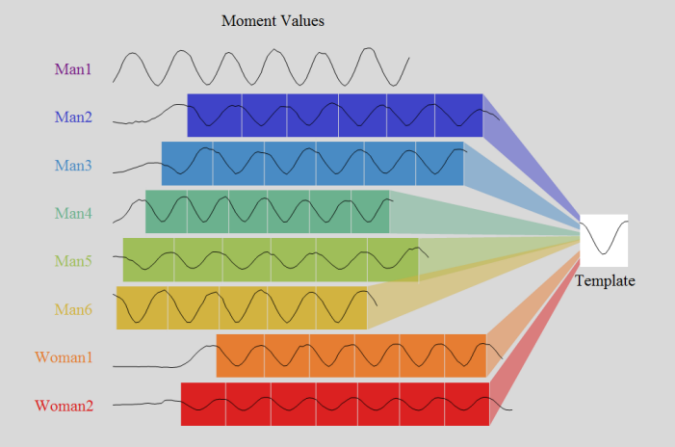


Fig.4 Generation of template array from the training set

At this stage we have several segments, which belong to the same action of different subjects, all having the same duration. To obtain the template trajectory, the obvious approach is to average the segmented trajectories. However, we observed that although the segmented trajectories usually resemble each other, their means and variances may be quite different. For example, while 10th moments of Man1 and Man6 have higher oscillation amplitudes, moment trajectory oscillations of Man4, Man5 and the two women are markedly smaller, as can be seen in Figure 4. Hence, before averaging the moment signals, they were first processed to make them zero mean and unit variance within their normalized duration. Then, by averaging the normalized trajectories, we obtained the action templates for each action (walk, run, jump, bend, hand-wave, jump in place, and sit).

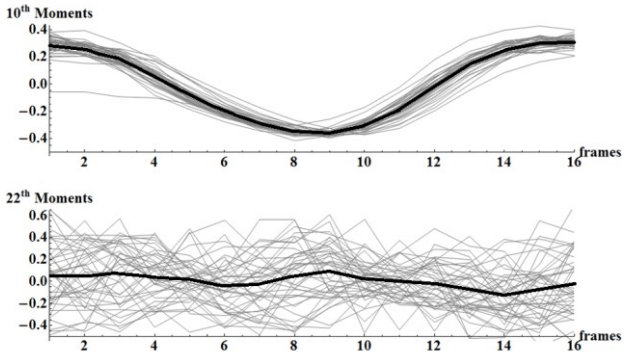


Fig.5 Standardized segmented arrays belonging to different moments

While computing the distance function, we used the zero-mean, unit variance versions of the template and test trajectories. We also observed that for some moment orders the trajectories do not resemble each other at all. In Figure 5, normalized trajectories of the 10th and 22nd (n = 4, l = 0, m = 0) moments are plotted. In this figure, the heavy lines show the mean trajectory (i.e. the template trajectory). The noise-like variations in the trajectories of the 22nd moment implies that this moment is not very suitable for classification purposes. It is therefore advisable to exclude this moment from distance calculations or, alternatively, reduce its weight. One possible approach to reduce its effect on distance is to put a weight that is inversely proportional to the variance of the trajectories. Consequently, we decided to use the Mahalanobis distance in the distance function instead of Euclidean distance. To obtain the covariance matrix, we used the training set and assumed zero covariance between components corresponding to distinct moments due to the orthogonality of the Zernike functions. Figure 6 depicts the variances of moments of various orders for a particular training set.

Finally, the calculated Mahalanobis distance is divided by template duration in order to nullify the effect of varying template durations belonging to different training sets. The calculated distance is also divided by the total number of moments in order to compare performances of different moment orders.

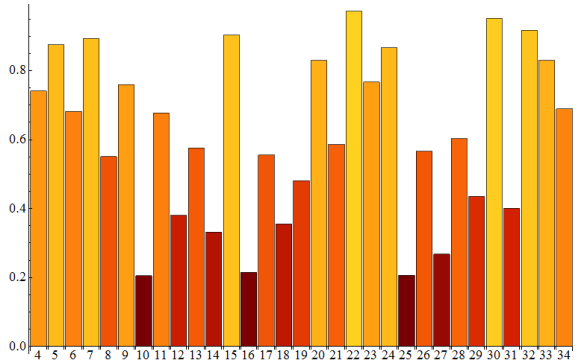


Fig.6 Variances of Moments (n=5)

After obtaining template trajectories and moment variances, we can now compute the distance between the template trajectories and the test data, which belong to Man1 in our example. This operation is shown in Figure 7. The template is aligned with the first frame of the test data, a distance value is computed, this value is stored in an array, the template is shifted one frame in time, and the process is repeated until the end of the test trajectory is reached.

Note that although we had normalized action segment durations while generating template trajectories, possible duration differences between the test and template actions have not been accounted for. In other words, during template matching we assumed that the action segment, which we want to label, is of the same length as the template. Although this assumption is clearly incorrect, we observed that variation between segment durations were small enough so that a more sophisticated matching scheme is not warranted. However,

there was one exception: Hand waving. We had to modify the template matching process for this action. Instead of calculating a single distance array with a single nominal-length template, templates of various length were generated by linearly interpolating the original and more than one distance array was calculated (19 to be exact). Lengths from 0.2 to 2 times the nominal template length were used with the steps of 0.1 times the nominal template length. Among those 19 distance arrays, the one which gave the minimum distance value became the final distance array. We have not used this revised template matching scheme for other actions because it significantly increases the computation time.

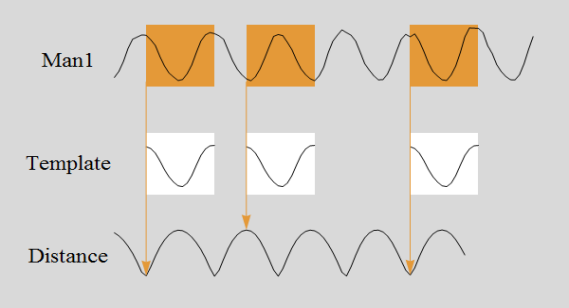


Fig.7 Generation of the distance array

Until now we considered generation of the distance arrays as the output of the template matching process. For each test video, we now have 7 distance arrays corresponding to 7 basic actions. At the frames where the action begins, there exist local minima in the distance array of the correct action as shown in Figure 7. In order to label the actions in the video, we first threshold the distance array and create disjoint "blobs" which consist of contiguous frames with distance values less than the threshold. The threshold values have been determined empirically. Then, we find the global minimum in each blob and put a candidate label there. This process is then repeated for all actions. The procedure is summarized in Figure 8. As shown in this figure, this procedure may yield many potential candidates for the same segment.

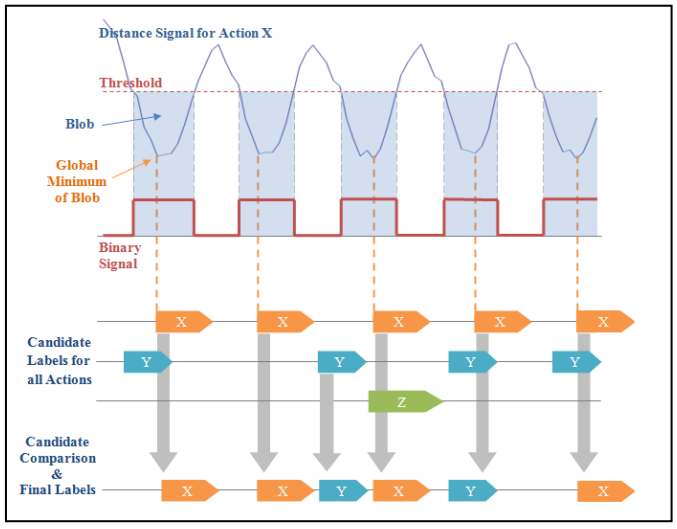


Fig.8 Thresholding and action labeling

To prune spurious labels, we assume that actions cannot occur simultaneously. Therefore, we look for intersecting candidate labels, i.e., different labels in overlapping blobs. If there is more than one label in intersecting blobs, we compare ratios of local minima to the thresholds. The candidate label for which this ratio is smaller wins and becomes the final label. Candidate labels having no intersection become final label automatically.

IV. Results

Using the method described above, we classified the 7 basic actions of the 8 subjects. We experimented with 3D Zernike moment orders of $n = 2$ to 6. The confusion matrix for moment order $n = 6$ is given in Table I. As expected, actions involving similar poses like walking/running and jumping/ jumping in place are the most commonly confused actions. Labelling results for $n = 5$ have also been given in a video [9].

TABLE I. CONFUSION MATRIX FOR $n = 6$ (PERCENTAGES)

		Labelled as						
		Walk	Run	Jump	Bend	Wave	Jump iP	Sit
Actual Action	Walk	100	0	0	0	0	0	0
	Run	5	95	0	0	0	0	0
	Jump	0	0	97	0	0	3	0
	Bend	0	0	0	100	0	0	0
	Wave	0	0	0	0	100	0	0
	Jump iP	0	0	3	0	0	95	2
	Sit	0	0	0	0	0	0	100

* Jump iP stands for Jumping in Place

Our results indicate that there is little difference between moment orders of 4 and 5, both of them yielding recognition accuracies greater than 97%. Lowering the order deteriorates performance, but even for n=3 we have greater than 95% accuracy. Going up to order 6 does not yield any appreciable gain in accuracy, possibly due to the curse of dimensionality stepping in as the number of features increases. However, going from order 5 to 6 does decrease the miss rate by more than 1%. Recognition accuracy values are given in Table II.

TABLE II. TEST RESULTS

Zernike Moment Order *n*	Number of Different Moments Used	Recognition Accuracy	Time needed to calculate moments (sec)
2	4	88.07 %	0.85
3	10	95.76 %	2.11
4	19	97.04 %	4.33
5	31	98.13%	7.57
6	47	98.16%	10.58

The rightmost column of Table II gives the time needed to calculate the Zernike moments *for each frame* of the video sequence on a PC with an Intel core i5 M460 processor running at 2.88 GHz and having 4GB RAM. Moment calculations by far dominate the computation time in our *Mathematica*

implementation. In comparison, distance calculation and labelling takes only about 6.27 milliseconds per frame (except for hand-waving). However, as noted above, the fast moment calculation algorithm of Hosny and Hafez [7] may significantly reduce computation times.

V. Discussion

Previous results indicate that higher order Zernike moments are necessary in order to characterize or reconstruct a 3D still image with reasonable fidelity. Hence, the fact that very good performance has been achieved in action recognition with low moment orders warrants further discussion.

As shown in Figure 2, moment order 5 is too low to faithfully reconstruct a still object. Order 10 barely conveys the idea of a walking human and in order to get anything like a real human figure, we need to go up to moment order $n = 20$. Then, how to explain the good recognition performance? We believe that the key here is the accumulation over time of minute differences. For example, the moment values corresponding to a single pose of the walking and running man are so close that these two poses cannot be reliably discriminated using a single snapshot, especially in the presence of noise (voxelization, computational, etc). On the other hand, although the moment values are close, they are not identical, and as these small differences accumulate over the duration of the segment (i.e., many frames), much better discrimination becomes possible.

Of course, these arguments are valid for recognition of coarse actions. Finer movements would probably require higher order moments along with their associated problems, in which case alternative methods might be preferable.

The current implementation can be improved in some ways, albeit at the expense of computational complexity. The first improvement may come from increasing 3D image resolution, i.e., decreasing the voxel size. Decreasing voxel edges from 5cm to about 2 or 1cm would presumably lead to better performance, but the computational load will increase with the cube of the resolution improvement. Employing higher order moments might help improve performance, although this is not readily evident from our results. Experimentation with a larger database and possibly higher resolution would be decisive here. Finally, more sophisticated nonlinear template matching methods might be employed instead of our simple linear warps.

TABLE III. COMPARISON OF ACTION RECOGNITION ACCURACIES OBTAINED ON THE I3DPOST DATABASE

Method in	[10]	[11]	[12]	[13]	[14]	[15]	[16]
Accuracy	92.19	98.44	94.87	94.37	96.34	100	95.78

Our results compare very well with recent results obtained on the same database. Recognition accuracies between 92.19% and 100% have been reported on the simple action classes of the i3DPost database as summarized in Table III. Our work indicates that action recognition using Zernike moments may provide a viable alternative to existing schemes, at least for coarse actions. One of the biggest merits of our method is that it does not require complicated temporal tracking algorithms but relies on simple matching of distance waveforms.

Acknowledgment

We would like to thank Dr. Hansung Kim for allowing us to use the i3DPost Multi-View Human Action Datasets in this work and the anonymous reviewers for their helpful comments.

References

[1] N. Gkalelis, H. Kim, A. Hilton, N. Nikolaidis and I. Pitas, "i3DPost multi-view and 3D human action/interaction database," Proc. CVMP, pp. 159-168, 2009.

[2] I. Mikic, M. Trivedi, E. Hunter and P. Cosman, "Articulated body posture estimation from multi-camera voxel data," Proc. CVPR (1), pp. 455-462, 2001.

[3] M. C. Lu, C. H. Lo and H. S. Don, "A neural network approach to 3-D object identification and pose estimation," *1991* IEEE Joint Conference on Neural Networks, vol. 3, pp. 2600-2605, 1991.

[4] N. Canterakis, "3D Zernike moments and Zernike affine invariants for 3D image analysis and recognition," 11th Scandinavian Conf. on Image Analysis, 1999.

[5] G. B. Arfken, H. J. Weber and F. E. Harris, Mathematical Methods for Physicists, Seventh Edition: A Comprehensive Guide, Academic Press, 2013.

[6] M. Novotni and R. Klein, "Shape retrieval using 3D Zernike descriptors," Computer-Aided Design, vol. 36, no. 11, pp. 1047-1062, 2004.

[7] K. Hosny and M. Hafez, "An algorithm for fast computation of 3D Zernike moments for volumetric images," Mathematical Problems in Engineering, vol. 2012, Article ID 353406, 2012.

[8] D. Berjón and F. Morán, "Fast human pose estimation using 3D Zernike descriptors," Proc. SPIE-IS&T Electronic Imaging, SPIE Vol. 8290, March 2012.

[9] http://www.ee.hacettepe.edu.tr/~semih/zernike/action.wmv

[10] M. Holte, T. Moeslund, N. Nikolaidis and I. Pitas, "3D human action recognition for multi-view camera systems," in Proc. 3DIMPVT, pp. 342-349, 2011.

[11] M. Holte, B. Chakraborty, J.Gonzàlez, T. Moeslund, "A local 3-D motion descriptor for multi-view human action recognition from 4-D spatio-temporal interest points," IEEE J. Sel. Topics Signal Process., vol.6, no.5, pp. 553-565, Sept.2012.

[12] A. Iosofidis, A. Tefas and I. Pitas, "View-invariant action recognition based on artificial neural networks," IEEE Trans. Neural Net. Learning Syst., vol.23, no.3, pp.412-424, Mar. 2012.

[13] A. Iosofidis, A. Tefas, N. Nikolaidis and I. Pitas, "Multi-view human movement recognition based on fuzzy distances and linear discriminant analysis," Comput. Vis. Image Understanding, vol.116, no. 3, pp. 347-360, 2012.

[14] A. Iosofidis, A. Tefas and I. Pitas, "Multi-view action recognition based on volumes, fuzzy distances and cluster discriminant analysis," Signal Process., vol. 93, no. 6, pp. 1445-1457, 2013.

[15] A. Iosofidis, A. Tefas and I. Pitas, "Minimum class variance extreme learning machine for human action recognition," IEEE Trans. Circuits. Syst. for Video Technol., vol 23, no.11, pp.1968-1979, Nov. 2013.

[16] B. Mahasseni and S. Todorovic, "Latent multitask learning for view-invariant action recognition," in ICCV 2013, Sydney, 2013.

ÖZGEÇMİŞ

Kimlik Bilgileri

Adı Soyadı	:	Okay Arık
Doğum Yeri	:	Ankara
Medeni Hali	:	Bekar
E-posta	:	okayarik@gmail.com
Adresi	:	Karşıyaka Sokak No: 40/20 Dikmen 06460 Ankara / TÜRKİYE

Eğitim

Lise	:	Hacı Ömer Tarman Anadolu Lisesi	(1997 – 2004)
Lisans	:	Orta Doğu Teknik Üniversitesi Elektrik ve Elektronik Mühendisliği	(2004 – 2009)
Yüksek Lisans:		Hacettepe Üniversitesi Elektrik ve Elektronik Mühendisliği	(2011 – 2014)

Yabancı Dil ve Düzeyi

İngilizce İyi

İş Deneyimi

A Bilgi Teknolojileri	Tasarım Mühendisi	(2010 – 2012)
ARTU Bilgi Teknojileri	Kurucu	(2013 –)

Deneyim Alanları

Görüntü İşleme, Kamera Kalibrasyonu, 3B Modelleme, Mathemetica, Java, OpenCV

Tezden Üretilmiş Projeler ve Bütçesi

–

Tezden Üretilmiş Yayınlar

–

Tezden Üretilmiş Tebliğ ve/veya Poster Sunumu ile Katıldığı Toplantılar

Arık, O., Bingöl, S., Human Action Recognition Using 3D Zernike Moments, *11th International Multi-Conference on Systems, Signals and Devices*, 11-14 February, Castelldefels-Barcelona, Spain, 2014

CURRICULUM VITAE

Credentials

Name, Surname	:	Okay Arık
Place of Birth	:	Ankara
Marital Status	:	Single
E-mail	:	okayarik@gmail.com
Address	:	Karşıyaka Sokak No:40/20 Dikmen 06460 Ankara / TURKEY

Education

High School	:	Hacı Ömer Tarman Anatolian High School	
BSc.	:	Middle East Technical University Electrical and Electronics Engineering	(2004 – 2009)
MSc.	:	Hacettepe University Electrical and Electronics Engineering	(2011 – 2014)

Foreign Languages

English Advanced

Work Experience

A Information Technologies	Design Engineer	(2010 – 2012)
ARTU Information Technologies	Founder	(2013 –)

Areas of Experience

Image Processing, Camera Calibration, 3D Modeling, Mathematica, Java, OpenCV

Projects and Budgets

–

Publications

–

Oral and Poster Presentations

Arık, O., Bingöl, S., Human Action Recognition Using 3D Zernike Moments, *11th International Multi-Conference on Systems, Signals and Devices*, 11-14 February, Castelldefels-Barcelona, Spain, 2014