

SEMANTİK ARAMA İLE MULTİMEDYA VERİLERİN ANLAMLANDIRILMASI

**2018
DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

Oğuzhan MENEMENCİOĞLU

**SEMANTİK ARAMA İLE MULTİMEDYA VERİLERİN
ANLAMLANDIRILMASI**

Oğuzhan MENEMENCİOĞLU

**Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalında
Doktora Tezi
Olarak Hazırlanmıştır**

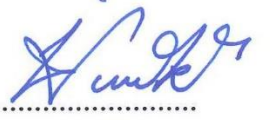
KARABÜK

Kasım 2018

Oğuzhan MENEMENCİOĞLU tarafından hazırlanan “SEMANTİK ARAMA İLE MULTİMEDYA VERİLERİN ANLAMLANDIRILMASI” başlıklı bu tezin Doktora Tezi olarak uygun olduğunu onaylarım.

Doç. Dr. İlhami Muharrem ORAK

Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı



Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir. 15/11/2018

Ünvanı, Adı SOYADI (Kurumu)

İmzası

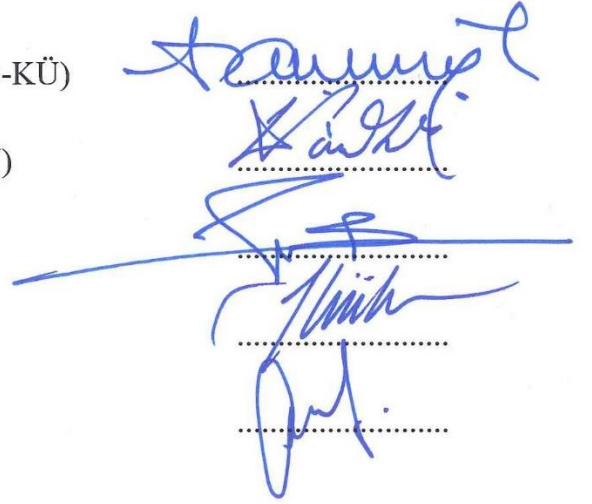
Başkan : Prof. Dr. Abdullah ÇAVUŞOĞLU (KTO-KÜ)

Üye : Doç. Dr. İlhami Muharrem ORAK (KBÜ)

Üye : Doç. Dr. Baha ŞEN (AYBÜ)

Üye : Dr. Öğr. Üyesi İlker TÜRKER (KBÜ)

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



...../...../2018

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Doktora derecesini onamıştır.

Prof. Dr. Filiz ERSÖZ

Fen Bilimleri Enstitüsü Müdürü V.





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

Oğuzhan MENEMENCİOĞLU

ÖZET

Doktora Tezi

SEMANTİK ARAMA İLE MULTİMEDYA VERİLERİN ANLAMLANDIRILMASI

Oğuzhan MENEMENCİOĞLU

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:

Doç. Dr. İlhami Muharrem ORAK

Kasım 2018, 102 sayfa

Günümüzde büyük miktarda multimedya verisi üretilmiştir. Büyük bir hızla da veri üretilmeye devam etmektedir. Ortaya çıkan veri miktarı karşısında geleneksel ve manuel görüntü işleme yöntemleri yetersiz kalmaktadır. Buna karşın, otomatik açıklama üretme yaklaşımları hem ele alınması ve çalıştırılması zor yöntemlerdir hem de salt görüntü işleme basamağında kalmaktadır. Görüntü işleme basamağından sonra çıkan veriye erişecek, veriyi ilişkilendirecek ve anlamlandıracak sistemlere ihtiyaç artmaktadır.

Multimedya işlenmesi ve erişimi: çerçevelerinde yer alan nesnelerin tespit edilmesi, tanınması ve yeniden tanınması ile etiketlenmesi, etiketlenmiş veriler üzerinden anlamsal sorgular yapılması ve bu anlamlardan bilgi üretilmesi aşamalarından oluşmaktadır. Bu aşamaların her biri bilimsel olarak ilgi çeken ve çözülmeyi bekleyen birçok problem barındırmaktadır.

Bu tez kapsamında multimedia veriden otomatik olarak elde edilen düşük seviye anahtar kelimeler ile multimedia verinin insan tarafından yorumlanması arasındaki boşluğu dolduracak bir çerçeve amaçlanmıştır.

Multimedyaadan herhangi bir bilgisayarlı görü yöntemi ile elde edilen veriler test verileri olmak üzere, benchmark verisi ve bu veriden yapay yolla elde edilen sentetik veri kullanılarak sorgulamalar yapılmış ve son kullanıcı için anlamlı-faydalı bilgi çıkarımı yapılmıştır.

Ham multimedia verisinin anlam çıkarmada sınırlı kaldığı yerde kavramsal olarak bir üst düzey bilgi katmanı olan gidiş yolu ile ilave anlamlar çıkarılmıştır. İki katmanda oluşturulan çerçevenin üzerinde tartışılmıştır.

Anahtar Sözcükler : Multimedia, SPARQL, semantik Web, semantik arama, gidiş yolu, multimedia anlamlandırma.

Bilim Kodu : 924.1.014

ABSTRACT

Ph. D. Thesis

INTERPRETATION OF MULTIMEDIA DATA WITH SEMANTIC SEARCH

Oğuzhan MENEMENCİOĞLU

**Karabük University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering**

**Thesis Advisor:
Assoc. Prof. Dr. İlhami Muharrem ORAK
November 2018, 102 pages**

Huge amount of multimedia data has been produced nowadays. Data is still and rapidly being produced. Traditional and manual methods are incapable of processing the amount of multimedia data. Although however, both automatic annotation generation methods are hard to address and to operate, and they are also on computer vision level. After computer vision level, there is an increasing need for systems to reach data, relate them and produce meaning from the retrieved data.

Handling multimedia and retrieve multimedia data consist of annotating via detection, recognizing and re-identifying the moving and non-moving objects, creating semantic queries on annotated data and producing knowledge from these semantics. Each level of these processes consists many unsolved and attractive scientific problems.

The goal of this thesis was creating a framework for filling the semantic gap between low level keywords which are retrieved automatically from multimedia data and human interpretations of multimedia data.

The queries are done, and the useful and valuable knowledge was retrieved for end-users by using test data which is obtained via any computer vision algorithm, benchmark data and synthetic data which is achieved by destructing the benchmark data.

When retrieving information from raw multimedia data is limited, the additional semantics are produced via using the upper conceptual level of information trajectories. The framework which consists of two level, is discussed.

Key Word : Multimedia, SPARQL, semantic Web, semantic search, trajectory, interpretation of multimedia.

Science Code : 924.1.014

TEŞEKKÜR

Bu tez çalışmasını konusunun belirlenmesinden başlayarak, araştırma ve tamamlanma aşamalarının sonuna kadar her birinde gösterdiği ilgi, sabır ve alakanın yanında bilimsel emek, destek ve yönlendirmeleri sonucu tamamlayabildiğim sayın hocam Doç. Dr. İlhami Muharrem ORAK’a bilgi, birikim ve tecrübesini hiç eksik etmeden rehber olduğu için sonsuz teşekkürlerimi sunarım.

Misafir araştırmacı olarak laboratuvarında dokuz ay bulduğum ve bu süre boyunca bilgi ve tecrübesinden istifade ettiğim Prof. Dr. Pascal Hitzler’e katkısından ve süre boyunca misafirperverliğinden dolayı teşekkür ederim.

Sevgili aileme manevi destekleri için kalbi teşekkürler ederim.

Özellikle sevgili eşime ve kızıma bu süreçte onlara ayırmam gereken vakitlerde tezle uğraşmama gösterdikleri tahammül ve manevi destekleri için çok müteşekkirim.

Ayrıca tez çalışmaları kapsamında “2214-A - Yurt Dışı Doktora Sırası Araştırma Burs Programı” ile destek sağlayan TÜBİTAK’a teşekkürlerimi arz ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
KABUL	ii
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ.....	xii
ÇİZELGELER DİZİNİ	xv
SİMGELER VE KISALTMALAR DİZİNİ	xvii
BÖLÜM 1	1
GİRİŞ	1
BÖLÜM 2	2
SEMANTİK WEB	2
2.1. URI	3
2.2. XML	4
2.3. RDF	5
2.4. RDFS	5
2.5. OWL.....	6
2.6. SPARQL.....	6
2.7. DEĞERLENDİRME	7
BÖLÜM 3	8
MULTİMEDYA	8
3.1. TARİHİ GELİŞİM SÜRECİ.....	8
3.2. MULTİMEDYA AÇIKLAMA	9
BÖLÜM 4	14
VERİ TEMİNİ	14

	<u>Sayfa</u>
4.1. VERİ TEMİNİ İŞLEMİ	14
4.1.1. Öğrenme Verisi.....	14
4.1.2. Sentetik Veri	16
4.1.3. Test Verisi.....	17
4.2. GÖRÜNTÜ İŞLEME YÖNTEMİ.....	17
4.2.1. Arka Plan Çıkarma Algoritması	18
4.2.2. Gauss Karışım Modelinin	18
4.3. VERİ İŞLEME	20
4.4. DEĞERLENDİRME	28
 BÖLÜM 5	 31
MODELLEME, SPARQL İLE SORGULAMA VE ANLAMLANDIRMA	31
5.1. MODELLEME	31
5.2. SPARQL İLE SORGULAMA	33
5.2.1. Yapılan Temel Sorgular.....	33
5.2.2. Yapılan Kompleks Sorgular	41
5.2.3. Yapılan Görselleştirilmiş Kompleks Sorgular.....	42
5.3. ANLAMLANDIRMA.....	46
5.4. DEĞERLENDİRME	50
 BÖLÜM 6	 51
NESNE YOLU, BENZERLİK VE ANLAMLANDIRMA ÜZERİNE İLERİ YAKLAŞIMLAR.....	51
6.1. GİDİŞ İZİ, ROTA VE YOL TANIMI	52
6.1.1. Matematiksel Formülasyon	53
6.2. MODEL.....	53
6.3. UZAKLIK	54
6.4. VERİ HAZIRLAMA.....	55
6.5. NORMALİZASYON İŞLEMİ.....	55
6.6. BENZERLİK YÜZDESİNİN HESAPLANMASI.....	56
6.7. NESNE HIZI	56
6.8. BENZERLİK.....	57

	<u>Sayfa</u>
6.8.1. PCA	57
6.8.2. PLS	60
6.8.3. K-means	61
6.9. PARAMETRE AĞIRLIKLARI İÇİN YAKLAŞIMLAR.....	63
6.10. SONUÇLAR	64
6.10.1. Doğruluk Sonuçları.....	65
6.10.2. Öklid Ağırlıklarının Doğruluk Sonuçlarına Etkisi	70
6.10.3. Giriş-Çıkış Noktaları ve Hızın Doğruluk Sonuçlarına Etkisi	77
6.10.4. Seçilen Yöntemin Farklı Parametreler ile Test Edilmesi	84
6.10.4.1. Başlangıç ve Bitiş Noktasının Etkisi.....	85
6.10.4.2. Hızın Başarıma Etkisi	86
6.10.5. Ardışık Tekrarlı Verinin Başarıma Etkisi	86
6.10.6. Rastgele Benzeşmelerin Önlenmesi	88
6.10.6. Kümeleme Sonuçları	90
 BÖLÜM 7	 95
SONUÇLAR VE DEĞERLENDİRME	95
KAYNAKLAR	97
 ÖZGEÇMİŞ	 102

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. İşlem basamakları piramidi.....	1
Şekil 2.1. Semantik katlı pasta diyagramı.....	3
Şekil 2.2. Semantik Web çatısı.	4
Şekil 2.3. Evrensel kaynak tanımlayıcı.	4
Şekil 3.1. Multimedya açıklaması.....	10
Şekil 3.2. Video yapılandırma modeli.....	11
Şekil 4.1. Öğrenme verisi sahnesi.	15
Şekil 4.2. Görüntü işleme.....	21
Şekil 4.3. Program karşılaştırma veri setinden xml okuma.	21
Şekil 4.4. Program xml okuma.....	22
Şekil 4.5. UML diyagramı.	23
Şekil 4.6. Tespit edilen her bir nesnenin toplam kaç çerçevede yakalandığı.....	23
Şekil 4.7. Yazılım ekranı.....	26
Şekil 4.8. Yazılım eliminasyon ekranı.	28
Şekil 5.1. Veri örüntüsü.	32
Şekil 5.2. ex001 Sorgusu.....	33
Şekil 5.3. ex001 Sorgu sonucu.....	34
Şekil 5.4. ex002 Sorgusu.....	34
Şekil 5.5. ex002 Sorgu sonucu.....	34
Şekil 5.6. ex003 Sorgusu.....	35
Şekil 5.7. ex003 Sorgu sonucu.....	35
Şekil 5.8. ex004 Sorgusu.....	35
Şekil 5.9. ex004 Sorgu sonucu.....	35
Şekil 5.10. ex005 Sorgusu.....	36
Şekil 5.11. ex005 Sorgu sonucu.....	36
Şekil 5.12. ex006 Sorgusu.....	36
Şekil 5.13. ex006 Sorgu sonucu.....	37
Şekil 5.14. Güncellenmiş ex006 sorgusu.	37

Sayfa

Şekil 5.15. Güncellenmiş ex006 sorgusunun sonucu.....	38
Şekil 5.16. ex006 Sorgusun test verisi (Video no:15) sonucu.	38
Şekil 5.17. ex007 Sorgusu.....	39
Şekil 5.18. ex007 Sorgu sonucu.....	39
Şekil 5.19. ex008 Sorgusu.....	40
Şekil 5.20. ex008 Sorgu sonucu.....	40
Şekil 5.21. ex009 Sorgusu.....	40
Şekil 5.22. ex009 Sorgu sonucu.....	40
Şekil 5.23. ex0086 Sorgusu.....	41
Şekil 5.24. Test verisi üzerinde ex0086 Sorgu sonucu.	42
Şekil 5.25. ex0096 Sorgusu.....	43
Şekil 5.26. Test verisi üzerinde ex0096 Sorgu sonucu.	43
Şekil 5.27. Güncellenmiş ex0096 Sorgusu.	44
Şekil 5.28. ex008610 Sorgusu.....	45
Şekil 5.29. ex009610Sorgusu.....	47
Şekil 6.1. Analiz seviyeleri ve bilgi arası ilişki.....	52
Şekil 6.2. Benzerlik hesaplanması.	56
Şekil 6.3. PCA grafiği.	58
Şekil 6.4. İşaretlenmiş PCA grafiği.	58
Şekil 6.5. Tekrarsız veri PCA grafiği.....	59
Şekil 6.6. PCA (Xc Std Dev Wighted) grafiği.	59
Şekil 6.7. PCA (Yc Std Dev Wighted) grafiği.....	60
Şekil 6.8. PLS grafiği.....	62
Şekil 6.9. PLS Exclude grafiği.....	62
Şekil 6.10. K-means grafiği.	63
Şekil 6.11. Sentetik2-doğrulaması benzerlik grafiği.....	67
Şekil 6.12. Sentetik0-doğrulaması benzerlik grafiği.....	67
Şekil 6.13. Browse3 0 – RestSlumpOnFloor 1 (9, 25) nesnesi benzerlik grafiği.	68
Şekil 6.14. Browse3 2-Browse3 3 (12, 13) nesnesi benzerlik grafiği.....	68
Şekil 6.15. Gidiş yolları a) (6, 25) ve b) (6, 9) benzerlik grafiği.	68
Şekil 6.16. Gidiş yolları a) (5, 7) ve b) (1, 31) benzerlik grafiği.	69
Şekil 6.17. Gidiş yolları a) (41, 44) ve b) (15, 19) benzerlik grafiği.	69

Sayfa

Şekil 6.18. a) (1, 6) ve b) (20, 48) gidiş yolları benzerlik grafiği.	73
Şekil 6.19. a) (15, 16), b) (1, 44) ve c) (4, 27) gidiş yolları benzerlik grafiği.	74
Şekil 6.20. a) (23, 29) ve b) (10, 47) gidiş yolları benzerlik grafiği.	75
Şekil 6.21. a) (10, 48) ve b) (47, 48) gidiş yolları benzerlik grafiği.	76
Şekil 6.22. a) (15, 33), b) (10, 20), c) (34, 44) ve d) (2,33) gidiş yolları benzerliği. .	76
Şekil 6.23 (32, 37) gidiş yolu benzerliği.	79
Şekil 6.24. LW Öklid ilk 16 en benzer gidiş yolu.....	80
Şekil 6.25. Eşit ağırlıklı Öklid ilk 16 en benzer gidiş yolları.....	80
Şekil 6.26. MW Öklid ilk 16 en benzer gidiş yolları.	81
Şekil 6.27. HW Öklid ilk 16 en benzer gidiş yolları.....	82
Şekil 6.28. SW Öklid ilk 16 en benzer gidiş yolları.....	83
Şekil 6.29. Eşit ağırlıklı Öklid başlangıç bitiş parametresi etkisi.	85
Şekil 6.30. Eşit ağırlıklı Öklid hız parametresinin etkisi.	87
Şekil 6.31. Gidiş yolları a) (9, 22), b) (9, 14), c) (4, 27), d) (8,48) benzerliği.	88
Şekil 6.32. LW Öklid Mak-Min-Std Dev sonuçları.....	89
Şekil 6.32. (devam ediyor).	90
Şekil 6.33. 1. Küme elemanları a) (1, 41) b) Sentetik0-doğrulaması c) (6, 34).....	92
Şekil 6.34. 2. Küme elemanları a) (2, 15) b) (32, 42) c) (32, 37).	92
Şekil 6.34. (devam ediyor).	93
Şekil 6.35. 3. Küme elemanları a) (3, 35) b) (36, 46).	93
Şekil 6.36. 5. Küme elemanları a) (5, 7) b) (17, 40) c) (12, 13) d) (21, 39).	93
Şekil 6.36. (devam ediyor).	94
Şekil 6.37. 7. Küme elemanları a) (9, 14) b) (22, 25).	94
Şekil 6.38. 10. Küme elemanları a) (16, 33) Sentetik2-doğrulaması b) (19, 33).....	94

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 4.1. Veri setinden seçilerek üzerinde çalışılan videolar.....	16
Çizelge 4.2. Tespit edilen her bir nesne detayı.	25
Çizelge 4.3. Her bir nesne ceza puanı karşılaştırması.....	27
Çizelge 4.4. Nesne eliminasyon sonuçları.	29
Çizelge 5.1. Test verisi sonuçları ve tepki süresi.	38
Çizelge 5.2. Güncellenmiş ex0096 Sorgusu sonucu.	44
Çizelge 5.3. ex008610 Sorgusu sonucu.	44
Çizelge 5.4. ex009610 Sorgusu sonucu.	45
Çizelge 5.5. Etiketleme doğruluk sonuçları.	48
Çizelge 5.6. Karşılaştırma verisinin tutarlılığı.	48
Çizelge 5.7. Nesnelerin hızı.	49
Çizelge 5.8. Test verisi etiketleme sonuçları.	49
Çizelge 6.1. Basit parametrelerde uygulanan Öklid ağırlıkları.....	64
Çizelge 6.2. İstatistiksel parametrelerin yer aldığı Öklid yönteminde ağırlıklar.	65
Çizelge 6.3. Öklid, eşit ağırlıklı temel 5 parametre ile gidiş yolları benzerlikleri.	66
Çizelge 6.4. LW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.....	70
Çizelge 6.5. MW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.....	71
Çizelge 6.6. HW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.	71
Çizelge 6.7. SW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.	72
Çizelge 6.8. Eşit ağırlıklı Öklid’e referansla değişen gidiş yolları sıralamaları.	75
Çizelge 6.9. LW Öklid, 10 parametre ile gidiş yolları benzerlikleri.....	78
Çizelge 6.10. Eşit ağırlıklı Öklid, 10 parametre ile gidiş yolları benzerlikleri.	78
Çizelge 6.11. MW Öklid, 10 parametre ile gidiş yolları benzerlikleri.....	79
Çizelge 6.12. HW Öklid, 10 parametre ile gidiş yolları benzerlikleri.	82
Çizelge 6.13. SW Öklid, 10 parametre ile gidiş yolları benzerlikleri.	83
Çizelge 6.14. LW, Eşit ağırlıklı, MW, HW ve SW Öklid sonuçları özeti.	84
Çizelge 6.15. Eşit ağırlıklı Öklid başlangıç bitiş parametresi etkisi.	85
Çizelge 6.16. Eşit ağırlıklı Öklid hız parametresinin etkisi.	86

	<u>Sayfa</u>
Çizelge 6.17. Ardışık tekrarlı verinin çıkarılmasının etkisi.	87
Çizelge 6.18. Eşit ağırlıklı Mak-Min-Std Dev sonuçları.	89
Çizelge 6.19. KNN ile oluşmuş kümeler.	91



SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

O	: orientation (Yönlendirme, yönü)
H	: height (Boy, yükseklik)
W	: weight (En, genişlik)
X _c	: resimdeki nesnenin mevcut yatay pozisyonu
Y _c	: resimdeki nesnenin mevcut dikey pozisyonu
$\sigma_{\min_i}$	: i nesnesinin ilk tespit edildiği çerçeve numarası
$\sigma_{\max_i}$	: i nesnesinin son tespit edildiği çerçeve numarası
$\sigma_{TotalFrameNumber_i}$	: i nesnesinin bulunduğu toplam çerçeve sayısı
D	: distance (Mesafe)

KISALTMALAR DİZİNİ

WWW	: World Wide Web (Dünyanın Geniş Web'i)
HTTP	: Hyper Text Transfer Protocol (Hiper Metin Transfer Protokol)
URI	: Uniform Resource Identifier (Evrensel Kaynak Tanımlayıcı)
URL	: Uniform Resource Locator (Evrensel Kaynak Konum Belirleyici)
XML	: Extensible Markup Language (Genişletilebilir Biçimlendirme Dili)
RDF	: Resource Description Framework (Kaynak Tanımlama Çerçevesi)
RDFS	: Resource Description Framework Schema (Kaynak Tanımlama Çerçevesi Şeması)
OWL	: Web Ontology Language (Web Ontoloji Dili)
CAL	: Coached Active Learning (Yetiştirilmiş Aktif Öğrenme)
ABC	: Active Bucket Classification (Aktif Biriktirme Sınıflandırma)
W3C	: World Wide Web Consortium (WWW Konsorsiyumu)
FPS	: Frames Per Seconds (Saniyede Çerçeve Sayısı)
PCA	: Principal Components Analysis (Temel Parçacık Analizi)

PLS : Partial Least Squares (Kısmi En Küçük Kareler)
KNN : K-Nearest Neighbors (K-En Yakın Komşu)
LWE : Low Weighted Euclidian (Düşük Ağırlıklı Öklid)
E : Euclidian (Öklid)
MWE : Medium Weighted Euclidian (Orta Ağırlıklı Öklid)
HWE : High Weighted Euclidian (Yüksek Ağırlıklı Öklid)
SWE : Special Weighted Euclidian (Özel Ağırlıklı Öklid)



BÖLÜM 1

GİRİŞ

Artan miktarda multimedya içerik günümüzde dijital formda üretilmekte, işlenmekte ve saklanmaktadır. Farklı multimedya platformlarından sadece Youtube'un 2015 verilerine göre, her ay 6 milyar saatin üzerinde video izlenmektedir; daha da önemlisi dakikada 100 saatlik yeni video Web'e yüklenmektedir [1]. Youtube artık bu şekilde bir istatistik sağlamazken, günümüzde bu miktar daha da fazla olarak gerçekleşmektedir.

Erişim için metadata olarak adlandırılan, genellikle kullanıcı tarafından elle oluşturulan metin tabanlı kaynak tanımlaması kullanılmaktadır. Metadata, kaynağın farklı özelliklerini özetleyen genel bilgidir. Metadata arama motorlarının içeriğin aranan şeyle alakalı olup olmadığına karar vermesini sağlamaktadır. Açıklama (annotation) ise içeriği anlamayı ve yorumlamayı kolaylaştırmak için video akışındaki nesneleri ve sahneleri (scene) işaretleme aracıdır [2].

Multimedya içerikten otomatik olarak hesaplanacak düşük-seviye anahtar kelimeler ile görsel sesli medyanın yorumlanması ve kullanıcı sorgularındaki anlamların zenginliği - sübjektifliği arasındaki, geniş aralığa 'Semantik Boşluk' (semantic gap) denmektedir [3]. Ortaya çıkan bu devasa multimedya veri setlerini yönetmek, anlamlandırmak, veri setlerinin içerdikleri bilgiye erişmek (knowledge retrieval) ve en önemlisi biriken multimedya veri yığınının üretildiği hızlarda faydalanmak çok etkili bilişim sistemleri tasarlayarak mümkün olacaktır. Bu amaçla metadata üzerinde semantik arama ile anlamlandırma yapılması gereklidir. Tez çalışmasında, multimedya verilerine kabul edilebilir doğrulukta ve hızlı erişimi sağlamaya yönelik bir çerçeve oluşturulması amaçlanmıştır. Bu çerçeve için gerekli multimedya açıklamanın otomatik olarak elde edilme süreci ele alınmıştır.

Multimedya açıklama üretme işlemi manuel ve/veya otomatik yapılabilen bir işlemdir. En doğru sonuçlar henüz manuel işleme sonucu üretilebilmektedir. Fakat bahsedilen veri miktarı karşısında manuel yöntem, multimedyanın üretildiği hızdan çok yavaş kalacağı için kabul edilemez olarak değerlendirilebilir. Otomatik açıklama üretme işlemi için ise çeşitli yaklaşımlar mevcuttur [4]. Otomatik açıklama çıkarma zor, ilgi çekmeye devam eden ve oldukça güncel bir konudur. Tez kapsamında CAVIAR açık karşılaştırma (benchmark public) veri setinin videoları işlenmiştir. Bu yaklaşımla amaçlanan, sonuçları bilinen ve sorgulanabilir veriler üzerinde işlem yaparak çalışmanın sonuçlarının doğruluğunu hesaplayabilmektir.

Şekil 1.1, tezde öngörülen ve araştırmaların yürütüldüğü işlem basamaklarının bir piramidini göstermektedir.



Şekil 1.1. İşlem basamakları piramidi.

BÖLÜM 2

SEMANTİK WEB

Web sadece basılı kaynaklarla sağlanan bilgi paylaşımının getirdiği kısıtları aşarak paylaşım yönteminin kökten değişmesini sağlamıştır. Web tarayıcıları kullanıcının hipertext linkler sayesinde bilgiye erişmesini sağlar. Arama motorları ise Web'deki dokümanları indeksler ve kullanıcıların arama cümleleriyle linkler arasındaki potansiyel uygunluğun anlamlandırılması için linklerin yapılarını analiz eder. Bu işlevsellik Web'in kısıtsız büyümesinin temel özelliği olarak da görülen genel, açık ve genişletilebilir doğasının sonucudur [5].

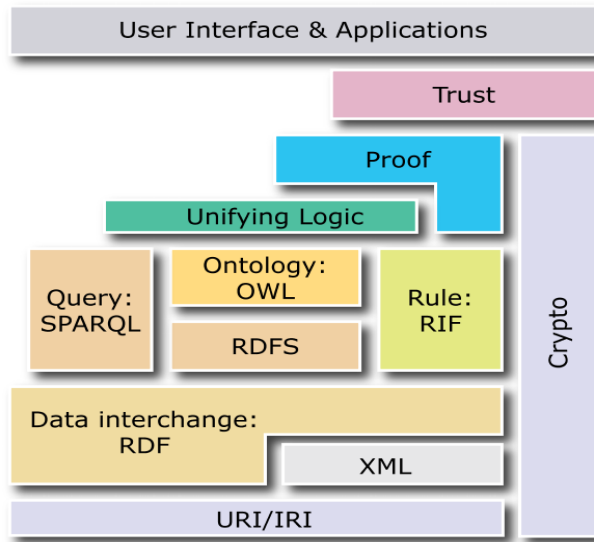
HTTP protokolünün geliştiricisi ve W3C'nin başkanı olan Tim-Berners Lee şöyle demiştir: “Web için bir hayalim var, öyle ki bilgisayarlar Web üzerindeki bütün veriyi, içerikler, linkler ve insanlarla bilgisayarlar arasındaki bütün işlemler gibi, analiz etmeye muktedir olacaklar. Henüz ortaya çıkmamış olsa da, ortaya çıktığı zaman anlamsal ağ ticaretin günlük mekanizmaları, bürokrasi ve günlük yaşamlarımız birbiri ile konuşan makineler tarafından yürütülecek. İnsanlığın asırlardır konuşup durduğu ‘akıllı ajanlar’ nihayet gerçekleşecek” [6].

Tim-Berners Lee semantik Web'i basit olarak: “Makineler tarafından doğrudan veya dolaylı olarak işlenebilen verilerin Web'i (Web of data)” şeklinde tanımlamaktadır [7].

Web 3.0 veya semantik Web: bilginin açıkça tanımlanmış bir anlama sahip olduğu, Web sayfalarının yapısının anlam ifade eden içerikte olduğu ve insanların olduğu kadar yazılım sistemlerinin de Web sayfalarını dolaşarak Web 2.0'da sadece insanların yaptıkları işlemleri gerçekleştirebildikleri, yani insanların ve yazılım sistemlerinin ortak olarak çalışabildikleri bir Web olarak tanımlanabilir (2010-) [8].

Semantik Web, dokümanları semantik anlamlar ekleyerek makine okuyabilir hale getirir. Bu yüzden Web’de dokümanların hâkimiyeti yerine veriye doğru bir dönüşüm başlamıştır. Fakat şimdiye kadar gerçekleşen değişim çok sınırlıdır. Çoğu Web sayfası farklı formatlarda biçimlendirilmemiş metin veya veri halindedir.

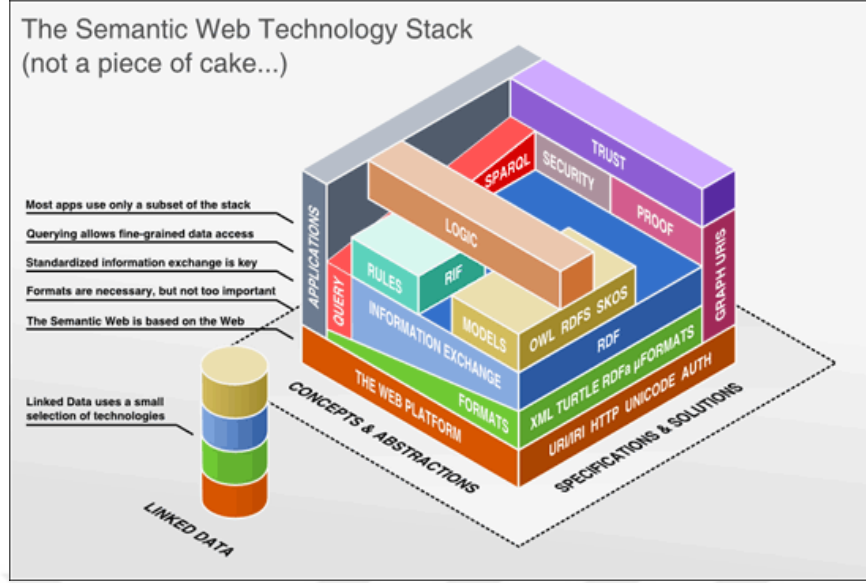
Semantik Web’in yapısı genellikle Şekil 2.1 [9] ile ifade edilmektedir. Ancak Şekil 2.2 [10]’de sunulan yığın artık yapıyı daha doğru ifade etmektedir. Katmanları birbiri üzerine konulmuş ve birbirinin üzerine inşa edilmiş biçimde düşünmek yerine, katmanların birlikte ve/veya birbirinin tamamlayıcısı olarak kullanılabildiği vurgusu öne çıkmaktadır. Ayrıca kavramların neye karşılık geldiği daha belirgin şekilde okunabilmektedir ve eklenen üçüncü boyut ile şemaya ifade tarzı olarak da derinlik eklendiği görülmektedir.



Şekil 2.1. Semantik katlı pasta diyagramı.

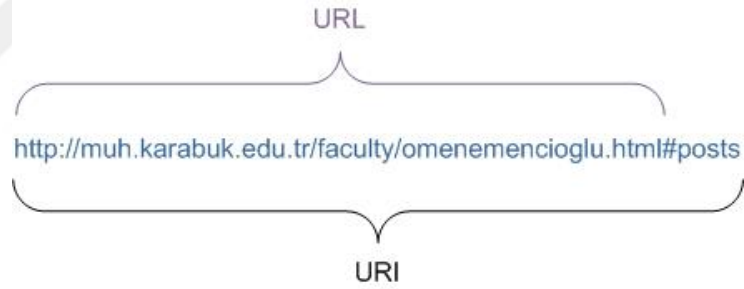
2.1. URI

URI, herhangi bir ‘şey’i veya internet üzerindeki kaynakları, tekil (unique) bir tanımlayıcı ifade kullanarak belirtmeye yarayan bir dizi metin karakterdir. URL internet üzerindeki belirli bir kaynağa ulaşmak için gerekli adresi belirtir. URI ile ifade edilen şey ise, ulaşılabilir bir Web kaynağı olabileceği gibi, herhangi bir nesne veya bir kavram da olabilir [11].



Şekil 2.2. Semantik Web çatısı.

Bir örnek URL ve URI adres Şekil 2.3’de verilmiştir.



Şekil 2.3. Evrensel kaynak tanımlayıcı.

2.2. XML

Genişletilebilir Biçimlendirme Dili (Extensible Markup Language, XML) 1986 yılında tanımlanan geliştirilmiş standart biçimlendirme dilinden türetilmiş basit, çok esnek bir metin formatıdır [12]. XML sentaktik gösterim için kullanılmaktadır [13].

2.3. RDF

Resource Description Framework (Kaynak Tanımlama Çerçevesi) Semantik Web için geliştirilen ilk dildir [13]. RDF mevcut Web'deki kaynaklara makineler tarafından erişilebilir metadata eklemeyi amaçlar [6]. RDF basit ve genel amaçlı bir metada dilidir.

RDF bilginin kavramsal olarak modellenmesini sağlayan, basit ve genel amaçlı bir üstveri dilidir. RDF özne-yüklem-nesne (subject-predicate-object) olarak şekillendirilen üçlüler (triples) topluluğundan oluşur [11,14]. Aşağıda bir örneği sunulmuştur [15].

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:a="http://description.org/schema/">
  <rdf:Description>
    <rdf:subject resource="http://www.w3.org/Home/Lassila" />
    <rdf:predicate resource="http://description.org/schema/Creator" />
    <rdf:object>Ora Lassila</rdf:object>
    <rdf:type resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Statement" />
    <a:attributedTo>Ralph Swick</a:attributedTo>
  </rdf:Description>
</rdf:RDF>
```

2.4. RDFS

RDF verileri için bir veri modelleme dağarcığı sağlar [6]. RDF için bir basit veri yazma modelidir. Aşağıda RDFS'nin bir kısmı gösterilmiştir [16].

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
```

```

<rdf:Description about="http://www.w3.org/2001/02/acls/ns#">
  <rdfs:comment>A namespace for describing Access Control Lists</rdfs:comment>
  <rdfs:comment>$Revision: 1.3 $ $Date: 2001/07/11 00:11:43 $</rdfs:comment>
  <rdfs:seeAlso resource="http://www.w3.org/2001/02/acls/acls-for-ns"/>
</rdf:Description>

<rdfs:Class ID="ResourceAccessRule">
  <rdfs:label xml:lang="en">Access Rule</rdfs:label>
  <rdfs:comment>An assertion of access privileges to a resource.</rdfs:comment>
  <rdfs:isDefinedBy resource="http://www.w3.org/2001/02/acls/ns#"/>
</rdfs:Class>
...
<!-- in progress... -->
</rdf:RDF>

```

2.5. OWL

OWL yapısal sınıfları ve Web dokümanlarında ve Web uygulamalarındaki ilişkileri tanımlayan ontoloji dilidir [17]. Nesneler, nesne grupları ve nesne grupları arasındaki ilişkiler OWL ile temsil edilir [18].

2.6. SPARQL

RDF için bir sorgu dilidir [17]. Aşağıda basit bir örnek sunulmuştur [19].

```

PREFIX ab: <http://learningsparql.com/ns/addressbook#>
SELECT ?craigEmail
WHERE
{
  ?person ab:firstName "Craig" .
  ?person ab:lastName "Ellis" .
  ?person ab:email ?craigEmail . }

```

Sorgu, adı “Craig” ve soyadı “Ellis” olan(lar)ın email adresini döndürmektedir.

2.7. DEĞERLENDİRME

Semantik Web göreceli olarak yeni bir konudur. Buna rağmen metin tabanlı (text based) uygulama ve araştırma sayısı bir hayli fazla iken, multimedya tabanında araştırma sayısı kısmen az olması gerekçesi ile etki alanı (domain) olarak multimedya seçilmiştir [4,20].

Şekil 1.1’de sunulan işlem piramidinde farklı basamaklarda işlemler yapılmış ve ilerleyen bölümlerde detaylara yer verilmiştir.



BÖLÜM 3

MULTİMEDYA

Bu bölümde semantik Web uygulamasını verisi üzerinde uygulayacağımız multimedyanın tanımı, tarihi gelişimi ve otomatik olarak işlenmesi hakkında yaklaşımlardan bahsedilmiştir.

3.1. TARİHİ GELİŞİM SÜRECİ

Multimedya ses, görüntü, grafik, yazı ve verinin aynı anda işlenebildiği ve iletilebildiği ortam olarak tarif edilmektedir. 19. yüzyılda telgrafla veri iletişimi olarak başlayan tele-iletişim, 19. yüzyılın sonunda sesli iletişim olarak devam etmiştir. 1930'dan sonra görüntülü yayınlar başlamıştır. 1980'den sonra bilgisayarda veri, ses ve görüntünün birlikte işlenmesi ile multimedya başlamıştır. 20. yüzyılın sonunda da iki yönlü görüntülü iletişim gerçekleşmiştir.

Multimedya, Douglas Engelbart ve ayrıca Nicholas Negroponte'nin çalışma gruplarıyla 1960'larda video ve bilgisayarı birleştirmek istemesiyle temelleri atılmıştır. Metin, ses, animasyon, video, grafik, film, fotoğraf, resim gibi dijital medya unsurlarından ikisinin veya daha fazlasının birleştirilmesi, yönetimi multimedya olarak tanımlanmaktadır.

Günümüzde multimedya verileri mobil cep telefonlarından ve taşınabilir multimedya cihazlarından bilgisayarlara değişebilen farklı elektronik donanımlarda bulunabilmektedir. Kullanıcılar artan bir hızla multimedya içeriği alma, oluşturma, kaydetme, gönderme, düzenleme, görüntüleme ve işleme kabiliyetine sahiptirler. Veriyi üretmek ve kaydetmek kolaylaşırken, arama yapmak ve aranan şeyle ilgili bilgiyi tespit edip erişmek ters orantılı olarak zorlaşmaktadır.

Aranan şeyle ilgili bilgiyi tespit etmek için bilgi erişim sistemleri ve bilgi erişim yaklaşımları geliştirilmektedir. Yaklaşımlardan bazıları multimedya üretilirken metadata ekleme esaslıdır, bazıları da sonradan metadata üretilmesi esaslıdır.

3.2. MULTIMEDYA AÇIKLAMA

Multimedya erişimi için multimedyanın açıklamasının üretimi, açıklamaları metadata ile ilişkilendirmek ve metadata erişimi gerekmektedir. Erişim bu basamakları ne kadar iyi yapıyorsa o kadar başarılıdır [21].

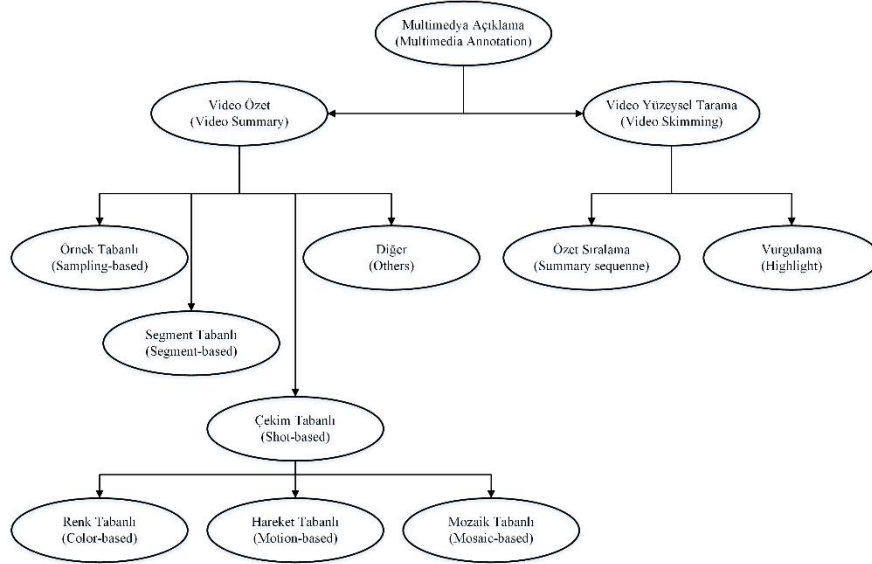
MPEG-7 formatı ile multimedya içeriğe metadata eklenmesi sağlanmaktadır. Multimedya içerik üzerinde yapılacak aramalar da bu metadata sayesinde gerçekleştirildiğinden, öncelikle MPEG formatından bahsetmekte fayda vardır. MPEG-7 ilk versiyonu MPEG (Moving Picture Experts Groups) 2002 yılında yayınlanmış ISO/IEC standardıdır. İkinci versiyonu 2006 yılında kullanıcılar ve geliştirici komiteler tarafından önerilerek yayınlanmıştır. Standart on kısımda tasarlanmış (sistemden, düşük-yüksek seviye multimedya açıklama şemasından referans yazılım ve uyumluluğa) ve multimedya içeriği tanımlamak için standartlaştırılmış araçların zengin bir kümesini sağlamaktadır. MPEG-7 standardının XML'e dayandığı bilinmektedir.

MPEG-7 formatı ile sonradan anlamlandırma yerine önceden içerikle ilgili açıklamaların multimedya eklenmesi söz konusudur. Resim gibi bazı multimedya türleri için önceden metadata eklenmesi daha kolay olsa da video, hem ses hem de çerçeveler (frame) içerdiğinden canlı yayınlarda sistematik bir anlamlandırma mekanizmasına muhtaçtır [22].

Yukarıda değinildiği üzere multimedya üretilirken metadata ekleme yöntemi ile elde edilen metadata kapsamlı olamadığı gibi çok sınırlı ve az nitelikte veri içermektedir. Kullanıcıların ve bilgi sistemlerinin multimedya üretilirken bahsedilen hızlarda metadata üretmesi de ayrıca mümkün değildir. Multimedia üzerinden geçilerek otomatik olarak metadata üretilmesi önceki yaklaşıma göre daha verimli fakat bir hayli zor bir yöntemdir.

Semantik multimedia erişim sistemleri birçok farklı çalışma alanı barındırmaktadır. Başarılı bir multimedia erişimi multimedia açıklamasının üretilmesi, metadata ile ilişkilendirilmesi ve bu metadataya erişilmesini gerektirmektedir. Videodaki kavramları otomatik olarak açıklamak hala ilgi çekici bir problemdir. Multimedyanın açıklamasını üretmek bilgisayarlı görü (computer vision) veya temel görüntü işleme alanındadır. Literatürdeki çalışmalar çoğunlukla multimedia semantiklerini elde etmekle veya farklı yöntemlerle multimedyanın açıklamasını üretmekle ilgilenmektedir. Ancak Ponnad ve Sharda SCORM'ları kullanarak bir multimedia semantik arama motoru önermiştir [23]. Gallego ve arkadaşları multimedia üzerindeki SPARQL sorgularının performansını arttırmak için bir sıkıştırma yöntemi önermiştir [24]. Fakat geri kalan çalışmalar anlam çıkarmaya odaklanmıştır.

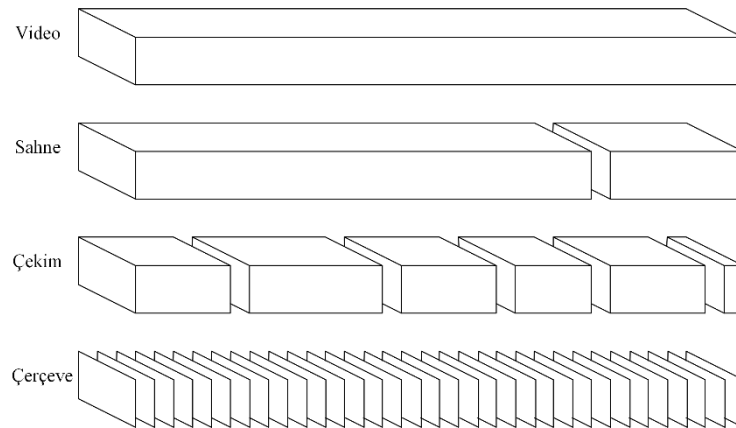
Multimedyanın manuel açıklanmasını göz ardı edersek, farklı multimedia açıklama üretme yaklaşımları mevcuttur. Li ve arkadaşları 2001 yılında video özetleme adı altında bir literatür taraması gerçekleştirip sonuçlarını çalışmalarında özetlemişlerdir [25]. Li ve arkadaşlarının çalışması ışığında, multimedyanın ele alınma alanı Şekil 3.1 ile özetlenebilir.



Şekil 3.1. Multimedia açıklaması.

Çalışmada otomatik açıklama problemi detaylı bir şekilde incelenmiştir. Mevcut yaklaşımlar bu geniş raporda tanımlanmıştır. Temel olarak iki yaklaşım vardır:

yüzeysel tarama ve özetleme. Yüzeysel tarama yaklaşımı tüm videodan orijinal sıralamada çıkarılan bazı resim ve sesleri kullanır. Yüzeysel taramanın iki tipi vardır. Özet dizisi metodu tarafından bütün videodan bir gösterim çıkarılır. Vurgulama metodu tarafından ise bir film fragmanı gibi videonun en önemli kısımları ihtiva edilir [25]. İlk yüzeysel tarama çalışmalarından olan video yapılandırma modeli Lienhart ve arkadaşları tarafından önerilmiştir. Şekil 3.2 video yapılandırma modelini göstermektedir [26]. Video özet yaklaşımı video kaynağından çıkarılmış veya üretilmiş önemli resimlerin küçük bir yığınına sağlar. Video özet yaklaşımının dört tipi mevcuttur. Örnek tabanlı anahtar çerçeve çıkarma metodu tarafından anahtar çerçeveler ya rastgele ya da homojen bir şekilde seçilir. Çekim (shot) bir video parçasıdır (segment). Çekim tabanlı anahtar çerçeve çıkarma metodu genellikle ilk çerçeveyi her bir çekimin anahtar çerçevesi olarak seçer. Tek çerçeve her çekimin içeriğini temsil etmek için yeterli olmayabilmektedir. Düşük seviyeli görsel özellikler tek anahtar çerçeve kısıdını aşarak semantik açıklama üretmek için kullanılır. Renk tabanlı yaklaşım ilk anahtar çerçeve, sonraki anahtar çerçeveler ve son anahtar çerçeve arasındaki renk histogram farkının belirli bir eşiği geçip geçmediğini değerlendirir. Hareket tabanlı yaklaşım piksel tabanlı görüntü farkını ve optik akış hesaplamasını kullanır. Mozaik tabanlı yaklaşım bütün video içeriğini iyi temsil etmek için sentezlenmiş panoramik görüntü üretir. Video parçası ya bir sahne ya bir olay hatta ya da bütün sekanstır. Her bir çekim için birkaç anahtar çerçeve ölçeklenebilir değildir, bu nedenle video parçaları kullanılır. Dalgacık dönüşümü, yüz tanıma gibi bazı başka metotlar da anahtar çerçeve tanımlamak için kullanılmaktadır [25].



Şekil 3.2. Video yapılandırma modeli.

Snoek ve arkadaşları multimedya verisinden açıklama üretmek için sadece görsel, sadece metinsel, erken birleşme, geç birleşme ve birleşik analizden oluşan beş katmanlı çekim tabanlı metot kullanmıştır. Çalışmalarında yeni araştırmacıların kendi sonuçları ile kıyaslama imkânı bulabilecekleri 101 manuel kavram sağlanmıştır [27].

Izquierdo ve arkadaşları çekim tabanlı anahtar çerçeve çıkarımı kullanmıştır. Çıkardıkları anahtar çerçeveleri açıklamak için üç farklı metot kullanılmıştır: karınca koloni optimizasyonu, parçacık sürü optimizasyonu ve dalga temelli görüntü sınıflandırıcı [28]. Argillander ve arkadaşları da multimedya içeriği otomatik olarak açıklamak için tek özelliği dikkate alarak maksimum entropi yaklaşımı ile çekim tabanlı anahtar çerçeve kullanmıştır [29].

Aktif öğrenme etiketlenmesi gereken örneklem sayısını azaltarak arama işlemini hızlandırır. Her bir arama işleminden sonra sistem ilgili ve ilgisiz örneklemeleri bir öğrenme işlemi olarak belirginleştirir. Sonraki aramalarda ilgisiz örneklemeler aynı sorgulama için yok sayılır. Böylece aktif öğrenme araştırmacılar tarafından genelde tercih edilir. Belirgin olmayan alanlarda arama yapmak yerine, yetiştirilmiş aktif öğrenme (CAL, Coached Active Learning) mekanizması belirgin alanlarda öğrenerek ve ön öğrenilmiş semantikleri kullanarak sorgulama dağılımını tahmin eder. CAL arama ile kullanma arasındaki dengeyi kurarak arama etkinliğini arttırmaya çalışır [30]. Aktif biriktirme sınıflandırma (ABC, Active Bucket Classification) yaklaşımı biriktirme temelli sınıflandırma kullanarak videoda ilgili çekimler tanımlar. Yaklaşım arama uzayını azalttığı için, yöntem sadece yüksek geri çağırılmalı video erişiminde değil, koleksiyon, sınıflandırma ve açıklama aracı olarak da faydalıdır [31].

Videodan içerik çıkarmak için ham veriler ve düşük seviyeli özellikler kullanılmaktadır. Fakat farklı alanlar için daha başarılı sonuçlara ihtiyaç vardır. Yıldırım ve arkadaşları otomatik semantik video içeriği çıkarmak için meta-ontoloji temelli ve ilave kural tanımlarını kullanarak çıkarım işlemi sağlayan çerçeveye giriş yapmaya çalışmıştır. Genel ontoloji temelli semantik meta-ontoloji modeli geliştirdikleri çerçeve tarafından kullanılır. Semantik içerikler otomatik genetik algoritma tabanlı nesne çıkarma metodu ile elde edilmektedir. Kural tabanlı ve alan bağımlı çıkarma metotlarının yerine, ontolojinin sağladığı avantajlar kullanılmaktadır.

Temelde görüntü temelli metot kullanılmıştır fakat bunun ötesinde otomatik semantik içerik çıkarmak için nesneler arası uzaysal ilişkilerle beraber olaylar arası geçici ilişkiyi kullanan ontoloji temelli yaklaşım önerilmiştir [32].

Gallego ve arkadaşları açık metadata kaynaklarını semantik olarak sorgulayabilmek için bir yapı oluşturmuştur [24]. Ponnada ve Sharda SCORM ve CORDRA standartlarını kullanarak daha zeki multimedya semantik arama motoru önermiştir. Çalışmalarında kullanıcı sorguları ile daha ilintili multimedya içeriği sağlamayı hedeflenmiştir [23]. Sorgulama ağırlıklı bir sistem yerine Liao ve arkadaşları birçok farklı türde videoyu analiz etmek için sahne temelli bir çerçeve geliştirmiştir. Çalışmalarında bir otomatik analiz çerçevesi, bir manuel oluşturma aracı ve bir HTML5 tabanlı video oynatıcı geliştirilmiştir [33].

BÖLÜM 4

VERİ TEMİNİ

Bu bölümde çalışmamızda kullanmak üzere multimedya etki alanından elde edilen verinin türleri, nasıl elde edildikleri, hali hazırda kullanılan – geliştirilen yaklaşımlar, algoritmalar ve gürültüden arındırılma süreçleri hakkında bilgiler paylaşılmıştır.

4.1. VERİ TEMİNİ İŞLEMİ

Bu çalışmada semantik arama kullanarak multimedya verilerinin anlamlandırılması sağlanmak istenmektedir. Giriş bölümünde bahsedildiği gibi farklı basamaklardan elde edilen veriler söz konusu olabilmektedir. Bu bölümde hangi veri setinin nasıl elde edildiği ve hangi amaçla kullanıldığı detaylı bir şekilde anlatılacaktır.

4.1.1. Öğrenme Verisi

Tez kapsamında geliştirilen yöntemlerin başarımlar-doğruluk açısından sonuçlarının kesin referans (ground truth) veri ile kıyaslanması için bir veri seti ihtiyacı oluşmuştur.

En yaygın olarak kullanılan multimedya veri setlerinden olan CAVIAR [34] birden çok giriş çıkış noktası olan çoklu video nesnelerinin izlerini taşımaktadır. Ayrıca kısmen kalabalık gruplar ile az kalabalık sahneleri barındıran videolardan oluşmaktadır [35,36].

CAVIAR veri seti karşılaştırma videoları barındırır. İlaveten bu videoların Java tabanlı, manuel ve interaktif bir yazılım aracı ile işlenmesi ve etiketlenmesi vasıtasıyla elde edilmiş kesin referans verisi sunar. Her video çerçevesi ve nesnesi tanımlanmış ve etiketlenmiştir. Otomatik yöntemlerle işlem yapanlar kendi sonuçlarını bu verilerle karşılaştırarak doğrulayabilir.

Amaca uygun olduđu düşünölen CAVIAR veri seti bu maksatla tercih edilmiştir. Tezin farklı bölömlerindeki işlemler bu veri seti üzerinde denenmiş ve sonuçları kıyaslanmıştır.

Şekil 4.1 de öğrenme verisinden Browse1 videosunun ilk çerçevesine ait görüntü yer almaktadır. 384 x 288 piksel boyutundaki görüntü, 25 FPS sıklığındadır.



Şekil 4.1. Öğrenme verisi sahnesi.

Veri setinden özellikle yürüme (walking), göz atma (browsing), düşme (falling) gibi temel hareketlerin olduđu, grup halinde hareket bulunmayan ve dosyalarını okumanın kısmen daha kolay olduđu 12 adet video seçilip bu videoların verileri üzerinde çalışılmıştır. 12 adet videoda toplam 45 adet hareket eden nesne bulunmaktadır. İlgili videolar Çizelge 4.1’de sunulmuştur. Videolarda Şekil 4.1’de gösterilen kadraj kullanılmıştır. Şekilde göröldüğü üzere nesnenin etrafında bir kutu (bounding box) bulunmaktadır. Veride bu kutunun bulunduđu çerçevenin numarası, nesnenin ID numarası, yönü (orientation), kutunun eni, boyu, xc-yc noktaları gibi özelliklerinin yanı sıra hareketini tanımlayan bir etiket bilgisi yer almaktadır.

Çizelge 4.1. Veri setinden seçilerek üzerinde çalışılan videolar.

ID	Video
1	Browse1
2	Browse2
3	Browse3
4	Browse4
5	Browse_WhileWaiting1
6	Browse_WhileWaiting2
7	Rest_FallOnFloor
8	Rest_SlumpOnFloor
9	Rest_WiggleOnFloor
10	Walk1
11	Walk2
12	Walk3

4.1.2. Sentetik Veri

Mevcut karşılaştırma verisinden yapay olarak elde edilmiştir. Öğrenme veri setinde mevcut nesnelerin verisinin örüntüsü bozularak elde edilmiştir. Amaç öğrenme veri setini genişletmek, kümeyi kendi içinde de öğrenme ve test verisi olarak ayırarak yapay verileri öğrenme kalanını test için kullanmaktır.

Sentetik veriyi oluştururken üç yaklaşım benimsenmiştir. Birincisi gidiş yolunun ötelenmesi, ikincisi zig-zag şeklinde bozma ve üçüncü olarak da başlangıcı bozma şeklindedir.

Birinci yaklaşımda, nesnesinin o , h , w , xc ve yc değerlerinin düzenli olarak taşınması (translation) sonucu elde edilmiştir. Nesnenin yönü hiç değiştirilmeden sadece h - w değerlerinin 3 birim ve xc - yc değerlerinin ise 5 birim arttırılması ile oluşturulmuştur. İkinci yaklaşımda nesne, orijinal verinin o , h , w , xc , yc değerlerinin 2 ile 20 birim arasında bir arttırılarak bir de eksiltiyle bozulması sonucunda üretilmiştir Üçüncü yaklaşımda ise bir noktadan sonra nesnenin verisindeki “ o ”nun -20 derece, “ h ve w ”nun -1, “ xc ”nin-3’den -8’e ve “ yc ”nin -2’den -15’e kadar değiştirilmesi ile elde edilmiştir. Nesne, orijinal verinin bir noktasına kadar hiç bozmadan korunması ve sonraki kısmın yaklaşık %20’sinin örüntüsünün değiştirilmesi ile elde edilmiştir. Verinin %80’i birebir aynı olduğu için üst üste çakışmaktadır

4.1.3. Test Verisi

Karabük Kares AVM'den kamera görüntüleri alınmıştır. Görüntüler kapı girişi ve muhtelif ardışık kameraların birer saatlik video kayıtlarından oluşmaktadır. Görüntülerin aşağıda bahsedilen yöntemlerle işlenmesi ile elde edilen veri setidir.

Öğrenme verisi üzerinde denenen yöntemlerin doğruluğu sınıdıktan sonra test verileri üzerinde de yöntemler denenerek sonuçları paylaşılmıştır.

4.2. GÖRÜNTÜ İŞLEME YÖNTEMİ

Giriş bölümünde de bahsedildiği gibi multimedya açıklama üretme işlemi manuel ve/veya otomatik yapılabilen bir işlemdir. Kesin referans verisi üretmek için genelde manuel açıklama üretme işlemi kullanılmaktadır. Fakat bahsedilen veri miktarı karşısında manuel yöntem çok yavaş kalmaktadır. Otomatik açıklama üretme işlemi için ise çeşitli yaklaşımlar mevcuttur [4]. Otomatik açıklama çıkarma zor, ilgi çekmeye devam eden ve oldukça güncel bir konudur.

Çalışma kapsamında CAVIAR karşılaştırma veri setinin videoları işlenmiştir. Amaç sonuçları bilinen ve sorgulanabilir veriler üzerinde işlem yaparak çalışmanın sonuçları karşılaştırıp doğruluğu hesaplayabilmektir. Veri setindeki her bir videonun çerçeveleri üzerinde nesneler yakalanmaya çalışılmıştır. Yapılan hesaplama, eleme vb. gibi ara işlemler ile gürültüler giderilmiştir. Neticede veri seti ile elde edilen sonuçlar karşılaştırılmış ve doğruluk değerleri sunulmuştur.

Otomatik multimedya açıklama üretmek için video özetleme yaklaşımının çekim tabanlı çerçeveler üzerinde hareket tabanlı yöntemi (Şekil 3.1) kullanılmıştır [4]. Hareket eden nesneler piksel farkı ve akış hesaplaması ile otomatik olarak tespit edilerek, zaman içindeki hareketleri takip edilmiştir.

4.2.1. Arka Plan Çıkarma Algoritması

Arka plan çıkarma algoritması bir video dizisindeki hareketli nesneleri, örneğin bir yolda hareket eden arabalar gibi, bulmak için geliştirilmiş eski bir tekniktir. Temel olarak fikir “Mevcut andaki resim, zamana göre ortalama arka plan resminden çıkarılırsa durağan olmayan nesneler kalır.” Şeklinde [37]. Mevcut andaki resmin her bir pikselinin sınıflandırma işlemi basit bir yaklaşım olduğu halde yavaş hareket eden nesneler için başarısızdır ve gölge ile hareket eden nesneleri ayıramamaktadır. t zamanında (x, y) pikseli için $I(x, y, t)$ anlık piksel değeri olduğunda, arka plan resmi en basit formunda, Denklem 4.1 uzun vadeli ortalama resmi olarak tanımlanır.

$$B(x, y, t) = \frac{1}{t} \sum_{t'=1}^t I(x, y, t') \quad (4.1)$$

Aynı zamanda aşamalı olarak Denklem 4.2 şeklinde de hesaplanabilir [37].

$$B(x, y, t) = \frac{(t-1)}{t} B(x, y, t-1) + \frac{1}{t} I(x, y, t) \quad (4.2)$$

4.2.2. Gauss Karışım Modelinin

Gauss modeli Wren ve arkadaşları tarafından sunulmuştur. Ancak piksel değerleri kompleks dağılımlara sahip olduğundan daha ayrıntılı modellere ihtiyaç duyulmuştur. Bu nedenle Friedman ve Russell tarafından arka plan çıkarma için Gauss karışım modeli önerilmiştir [38]. Çerçeve sayısı n olan ve her çerçevesindeki her bir pikselin bağımsız istatistikî işlem gördüğü ve gözlemlenen yoğunluğunun kaydedildiği, K Gauss karışımı ile t zamanda gözlenen x_t piksel yoğunluğunun olasılığı Denklem 4.3 ile modellenir.

$$\Pr(x_t) = \sum_{j=1}^K \frac{\omega_j}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_t - \mu_j)^T \Sigma_j^{-1} (x_t - \mu_j)} \quad (4.3)$$

Burada ω_j pikselin karışım modelinin j 'inci dağılımına atanmış ağırlık, μ_j ortalama ve Σ_j kovaryans matrisidir. Kolaylık için kovaryans matrisi yerine $\Sigma_j = \sigma_j^2 I$ kullanılmaktadır. Arka plan modeli belirlenirken her bir piksel tek tek ele alınır. K dağılımları büyük olan arka planların daha stabil olacağı düşüncesiyle ω_j/σ_j^2 temelinde sıralanır. Daha sonra T arka plan sürecinden elde edilmiş izin kesrinin tahmini olduğunda ilk B dağılımı denklem 4.4 ile seçilir.

$$B = \arg \min_b \left(\frac{\sum_{j=1}^b \omega_j}{\sum_{j=1}^K \omega_j} > T \right) \quad (4.4)$$

Modeli güncellemek için her piksel değeri, x_t , bir eşleşme buluncaya kadar var olan dağılımlara karşı kontrol edilir. Eşleşme bir dağılımın 2 standart sapmasının içinde bir piksel olarak tanımlanmıştır. Modelin dağıtımlarının hiçbirisi gelen değerle eşleşmezse, olası en düşük dağılım kaldırılır ve yerine başlangıçta yüksek bir varyansa ve düşük ön ağırlığa sahip bir dağılım konur.

α öğrenme değeri, $M_{j,t}$ ise eşleşen dağılım için 1 ve kalan dağılımlar için 0 olduğunda; t zamandaki her bir dağılım için dağılım ağırlığı Denklem 4.5 ile ayarlanır.

$$\omega_{j,t} = (1 - \alpha)\omega_{j,t-1} + \alpha(M_{j,t}) \quad (4.5)$$

Dağılımın parametrelerinin değiştiği hız $1/\alpha$ ile tanımlanır. $\omega_{j,t}$, piksel değerlerinin zaman 1'den t 'ye kadar model j ile eşleştiği rastlantısal düşük geçişli (eşiklenmiş) art olasılıktır.

$$\rho = \alpha * Pr(x_t | \mu_{j,t-1}, \sigma_{j,t-1}) \quad (4.6)$$

μ ve σ parametreleri eşlenmemiş dağılımlar için aynen kalır. Denklem 4.6, yukarıda bahsedildiği üzere nedensel düşük geçişli filtrenin tipiyle aynı olduğu durumda; eşleşen dağılımın parametreleri, j , Denklem 4.7 ve 4.8 şeklinde güncellenir.

$$\mu_{j,t} = (1 - \rho)\mu_{j,t-1} + \rho x_t \quad (4.7)$$

$$\sigma_{j,t}^2 = (1-\rho)\sigma_{j,t-1}^2 + \rho(x_t - \mu_{j,t})^T(x_t - \mu_{j,t}) \quad (4.8)$$

Bu, büyük deęişiklikler meydana geldiğinde hem ortalama hem de varyansın aydınlatma deęişikliklerini takip etmesini hem de arka plan dağılımını bozulmadan korumayı sağlar [39].

Hareket tahmininde Kalman filtresi kullanılmıştır. Kalman filtresi, birçok iz (track) ve veri tahmin işlerinde optimum çözüm olarak kendini ispatlamış algoritmadır. Kalman filtresi ki-kare minimize edicisi gibi düşünölüp türetilen denklemlerle, en büyük olabilirlik istatistięi geliştirerek tahmin yapar [40].

Tez kapsamında bir Gauss karışım modeli tabanlı arka plan çıkartma algoritması ile hareket eden nesneler tespit edilmiştir. nesnelerin hareket tahmininde de Kalman filtresi kullanılarak yapılmıştır. Karşılaştırma ve test videoları bu şekilde işlenmiştir. İşlenen videolardan elde veriler XML dosyalara yazılmaktadır.

4.3. VERİ İŞLEME

Yukarıda işlenen videolardan elde edilen veriler XML dosyalardan okunarak ele alınmakta ve aşağıda detayları sunulan işlemler veriler üzerinde uygulanmaktadır.

Hareket eden nesneler piksel farkı ve akış hesaplaması ile otomatik olarak tespit edilmiştir. Nesnelerin zaman içindeki hareketleri takip edilmiştir. Şekil 4.2 CAVIAR veri setinden “Browsing” tanımlamasının altındaki Browse1.mpg videosu nesne yakalama ve takip sürecini göstermektedir. Yöntemin aynı nesnenin zaman içindeki hareketlerinin yakalanmasında kullandığı ilişki sadece hareket tabanlıdır. Sadece hareket kullanılması bazı kısıtlara neden olmaktadır. Nesnenin durması ve tekrar hareket etmesi durumunda ilişki kopmaktadır. Üretilen veriler üzerinde semantik arama yapıp anlamlar çıkarılacağı için aynı nesnenin hareketleri çok önemli hale gelmektedir. Bu kısıdı aşmak için alınan tedbir ilerleyen kısımda açıklanmıştır.



Şekil 4.2. Görüntü işleme.

Görüntü işleme neticesinde elde edilen veriler XML dosyasına kaydedilmektedir. Görüntülerden elde edilen verileri okumak, işlemek, süzmek, kopan ilişkileri tekrar birleştirmek ve işlenmiş verilerden ontolojileri üretmek için bir yazılım geliştirilmiştir. Program ile veri setinden ve görüntü işlemeden sonra kaydedilen dosyalardan veri okuması yapılabilir. XML dosya okuma aşamasını gösteren ekran görüntüleri Şekil 4.3 ve Şekil 4.4’de sunulmuştur. Şekil 4.3’de Browse1 videosuna ait karşılaştırma verisinin dosyadan okunarak forma yüklenme örneğini, Şekil 4.4’de ise aynı videonun çalışmamız ile elde edilmiş sonuçları görülmektedir.

Initialize													
Read XML from CAVIAR		Read Values From Detection XML		Compute		Compare-Normalize		Create Ontologies					
dataset	frameNumber	objectID	orientation	h	w	xc	yc	appearance	movement	role	context	situation	
Browse1	0	0	174	19	48	287	277	appear	walking	walker	immobile	moving	
Browse1	0	1	173	34	54	210	242	appear	walking	walker	walking	moving	
Browse1	0	2	158	25	33	84	180	appear	walking	walker	browsing	moving	
Browse1	1	0	174	20	48	287	277	visible	walking	walker	immobile	moving	
Browse1	1	1	173	36	56	209	241	visible	walking	walker	walking	moving	
Browse1	1	2	158	25	33	84	181	visible	walking	walker	browsing	moving	
Browse1	2	0	174	21	50	286	276	visible	walking	walker	immobile	moving	
Browse1	2	1	173	37	58	208	239	visible	walking	walker	walking	moving	
Browse1	2	2	158	23	33	84	182	visible	walking	walker	browsing	moving	
Browse1	3	0	172	24	51	285	275	visible	walking	walker	immobile	moving	
Browse1	3	1	173	35	59	206	238	visible	walking	walker	walking	moving	
Browse1	3	2	158	24	33	84	183	visible	walking	walker	browsing	moving	

Şekil 4.3. Program karşılaştırma veri setinden xml okuma.

Dataset	Frame	Object	h	w	xc	yc	Age	Visible_Count	InVisible_Count
Browse1.mpg	4	1	27	44	179	229	1	1	0
Browse1.mpg	5	1	28	48	177	228	2	2	0
Browse1.mpg	6	1	29	64	175	227	3	3	0
Browse1.mpg	7	1	29	66	173	227	4	4	0
Browse1.mpg	7	2	20	33	69	170	1	1	0
Browse1.mpg	8	1	34	67	172	222	5	5	0
Browse1.mpg	8	2	21	34	68	170	2	2	0
Browse1.mpg	8	3	23	35	254	256	1	1	0
Browse1.mpg	9	1	34	70	169	222	6	6	0
Browse1.mpg	9	2	22	34	68	170	3	3	0
Browse1.mpg	9	3	34	52	252	255	2	2	0
Browse1.mpg	10	1	35	72	167	221	7	7	0

Şekil 4.4. Program xml okuma.

Şekil 4.3 ve Şekil 4.4’den anlaşılacağı üzere okunan-üretilen veriler forma yazılmaktadır. Tetiklenen hesaplama işlemi sırasıyla önce ilişkisel veri tabanı yaklaşımı ile her bir Dataset’i, çerçeveyi ve nesneyi, Şekil 4.5’de detayları sunulan ER diyagramına göre tablolara kopyalar. Daha sonra ilişkisel veri tabanı yaklaşımı ile ele alınmış veriler, ilişkileri kurularak “Algılama” (Detections) tablosuna kaydedilir.

Videodan toplam kaç çerçevede nesne yakalandığı ve yakalanan nesnelerin kaç çerçeve boyunca yakalanmaya devam ettiği istatistikî bilgisi de yazılım tarafından hesaplanmakta ve kullanıcıya çıktı olarak sunulmaktadır. Bu çıktının bir örneği, Browse1 videosu için, Şekil 4.6’da paylaşılmıştır.

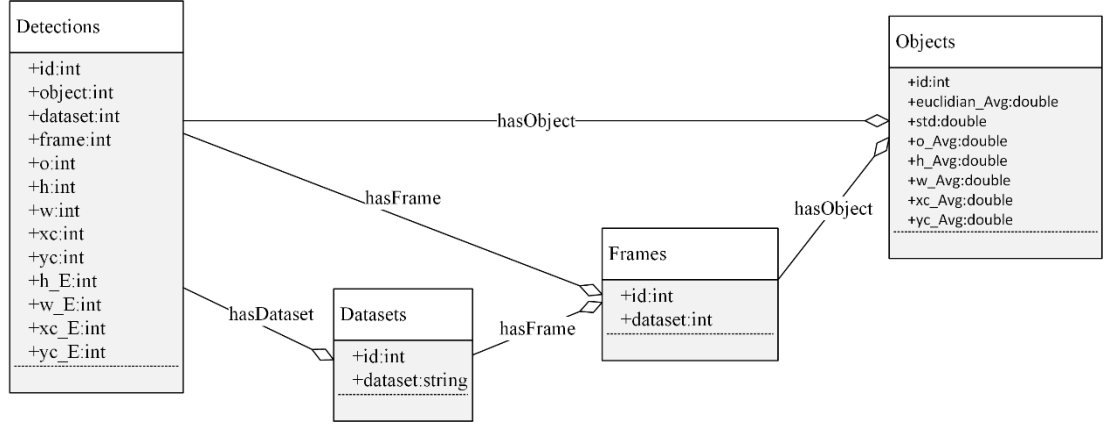
Bir yandan da her bir yakalama için (Algılama (Detections) tablosunun her bir kaydı için) bir önceki yakalama ile arasındaki Öklid uzaklığı Denklem 4.9-4.12 ile hesaplanarak tabloya kaydedilir.

$$d_{H_i} = \sqrt{h_i - h_{i-1}} \quad (4.9)$$

$$d_{W_i} = \sqrt{w_i - w_{i-1}} \quad (4.10)$$

$$d_{xc_i} = \sqrt{x_i - x_{i-1}} \quad (4.11)$$

$$d_{yc_i} = \sqrt{y_i - y_{i-1}} \quad (4.12)$$



Şekil 4.5. UML diyagramı.

```
Total Frame Count -> 672
1. object count -> 101
2. object count -> 85
3. object count -> 192
4. object count -> 66
5. object count -> 1
6. object count -> 1
7. object count -> 236
8. object count -> 161
9. object count -> 1
10. object count -> 1
11. object count -> 75
12. object count -> 1
13. object count -> 28
```

Şekil 4.6. Tespit edilen her bir nesnenin toplam kaç çerçevede yakalandığı.

n adet çerçevede toplam m adet nesne yakalandığı ve her bir nesnenin toplam yakalanma sayısının k olduğu durumda uzaklıkların ortalaması Denklem 4.13-4.16 ile hesaplanır.

$$\mu_{d_{Hj}} = \prod_{j=1}^m \left(\sum_{i=1}^n d_{Hi} \right) / k_j \quad (4.13)$$

$$\mu_{d_{Wj}} = \prod_{j=1}^m \left(\sum_{i=1}^n d_{Wi} \right) / k_j \quad (4.14)$$

$$\mu_{d_{Xcj}} = \prod_{j=1}^m \left(\sum_{i=1}^n d_{Xci} \right) / k_j \quad (4.15)$$

$$\mu_{d_{ycj}} = \prod_{j=1}^m \left(\sum_{i=1}^n d_{yci} \right) / k_j \quad (4.16)$$

Nesneleri temsil etmek için koordinat kullanılmıştır. Her bir nesnenin xc ve yc koordinat değerleri ile bir önceki kayıt ile arasındaki Öklid uzaklığı Denklem 4.17 ile hesaplanır.

$$d_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (4.17)$$

Nesne bu uzaklıkların ortalaması μ_d ile temsil edilmektedir ve Denklem 4.18 ile hesaplanır.

$$\mu_{d_j} = \prod_{j=1}^m \left(\sum_{i=1}^n d_i \right) / k_j \quad (4.18)$$

İlaveten benzer şekilde standart sapma da Denklem 4.19 ile hesaplanmıştır.

$$\sigma_{d_j} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \mu_j)^2} \quad (4.19)$$

Şekil 4.4’de okunan veriler ile hesaplama tetiklendiğinde, bahsedilen hesaplama süreçleri sonrasında elde edilen sonuçların yansıdığı yazılım ekranının bir örneği Browse1 videosu için Şekil 4.7’de, oluşan detaylar Çizelge 4.2’de sunulmuştur.

Veri setinde aynı videoda 4 adet nesne olduğu bilinmektedir. Ayrıca yukarıda bahsedildiği üzere kısıtlar nedeniyle nesnelerin durması ve yeniden hareket etmesi ayrıca kamera görüş alanından çıkıp tekrar görüntüye girmeleri nedeniyle çalışmamız tarafından yeni bir nesne olarak yakalanmaktadır. Bu noktada nesne eliminasyonu yapmak için bir yaklaşım önerilmektedir.

Çizelge 4.2. Tespit edilen her bir nesne detayı.

ID	Öklid Ort	Std	H Ort	W Ort	Xc Ort	Yc Ort
1	1,85	12,58	20,53	48,86	248,24	262,35
2	1,81	6,45	29,95	43,27	76,11	169,4
3	4,38	7,28	40,17	56,22	202,33	179,52
4	1,80	7,72	23	51,52	177,71	225,56
5	0	0	21	35	65	136
6	0	0	20	35	65	135
7	1,22	0,87	38,83	38,92	208,78	91,49
8	1,97	1,86	45,43	44,86	276,13	141,31
9	0	0	43	23	299	70
10	0	0	43	23	299	70
11	2,27	2,00	43,2	46,43	239,41	72,57
12	0	0	17	41	92	207
13	0,41	0,81	16,07	41,93	91,86	209,11

Nesne tablosundaki her nesne numarası ile bir O kümesi oluşturulur. O kümesinin her elemanı için nesnenin yakalandığı çerçeve numarası eklenerek F_r kümesi oluşturulur. Örneğin F_1 kümesi 1 ID numaralı nesnenin yakalandığı bütün çerçeve numaralarını barındırır. Burada aynı çerçevede yakalanan farklı nesnelerin birleştirilmesini önlemek için bir ceza puanı $pen_r = 100$ uygulanmıştır.

$$pen_r = \prod_{r=1}^{|O|} \prod_{i=1}^n \begin{cases} pen_r += 10, & (F_r \cap F_i) \neq \emptyset \\ pen_r, & (F_r \cap F_i) = \emptyset \end{cases} \quad (4.20)$$

Denklem 4.20 ile aynı çerçevede olan farklı nesneler için ceza puanı hesaplanır. Hesaplanan ceza puanı, eğer varsa, μ_d 'ye ilave edilir. Her bir nokta diğer noktalar ile karşılaştırılarak durum tespit edilir.

$$\prod_{r=1}^{|O|-1} \min_j \left(\prod_{j=r+1}^m \mu_{d_j} \right) \quad (4.21)$$

Burada temel yaklaşımımız sondan başlayarak ($|O| - 1, |O| - 2, \dots 1$) her bir nesnenin en yakın nesne ile birleştirilmesidir.

Çizelge 4.3 ceza puanı hesaplaması sonucunda ortaya çıkan örnek değerleri ve yaklaşımımızı göstermektedir. Örneğin 1 numaralı nesne $\{2,3,4\}$ numaralı nesneler ile aynı çerçevede yer aldığı için bu ceza puanlarına yansımıştır. 1 numaralı nesnenin

{7,8,11,13} numaralı nesneler ile kıyaslandığında hiç ceza puanı almadığından aynı çerçevelerde yer almadığı görülmektedir. Yaklaşım gereği sondan başladığı için 13 numaralı nesne en yakın nesne olan ve hiç aynı çerçevede yakalanmayan 7 numaralı nesne ile, 11-8 ile, 8-1 ile ve 7-4 numaralı nesne ile birleştirilmiştir.

Dataset	Frame	Object	h	w	xc	yc	Age	Visible_Count	InVisible_Count
Browse 1.mpg	4	1	27	44	179	229	1	1	0
Browse 1.mpg	5	1	28	48	177	228	2	2	0
Browse 1.mpg	6	1	29	64	175	227	3	3	0
Browse 1.mpg	7	1	29	66	173	227	4	4	0
Browse 1.mpg	7	2	20	33	69	170	1	1	0
Browse 1.mpg	8	1	34	67	172	222	5	5	0
Browse 1.mpg	8	2	21	34	68	170	2	2	0
Browse 1.mpg	8	3	23	35	254	256	1	1	0
Browse 1.mpg	9	1	34	70	169	222	6	6	0
Browse 1.mpg	9	2	22	34	68	170	3	3	0
Browse 1.mpg	9	3	34	52	252	255	2	2	0
Browse 1.mpg	10	1	35	72	167	221	7	7	0

Şekil 4.7. Yazılım ekranı.

İlaveten Çizelge 4.3’de görüldüğü üzere $ND = \{5,6,9,10,12\}$ nesneleri yöntemimiz ile temsil edilememiştir. Bunun nedeni sadece bir çerçevede ve bir kez yakalanmalarıdır. Bir önceki yakalama ile uzaklıkları olmadığından değer sonuçlara 0 olarak yansımıştır.

Temsil edilemeyen ND kümesindeki noktalar için Denklem 4.22 ile modellenen fark denklemini uygulanır.

$$dist_{p,j} = \prod_{p=1}^{|ND|} \prod_{j=1}^m \begin{cases} |\mu_{xc_p} - \mu_{xc_j}| & \mu_{d_p} = 0 \\ \mu_{d_p} \neq 0 \end{cases} \quad (4.22)$$

Sonrasında ND kümesinin her bir elemanı için en düşük fark Denklem 4.23 ile bulunur.

Çizelge 4.3. Her bir nesne ceza puanı karşılaştırması.

	1	2	3	4	5	6	7	8	9	10	11	12
1												
2	850.04											
3	972.53	842.57										
4	660.05	660.01	662.58									
5	100	100	100	100								
6	100	100	100	100	100							
7	0.63	0.59	3.16	0.58	100	100						
8	0.11	0.15	2.41	0.17	100	100	0.74					
9	100	100	100	100	100	100	100	100				
10	100	100	100	100	100	100	100	100	100			
11	0.42	0.46	2.11	0.47	100	100	1.05	0.30	100	100		
12	100	100	100	100	100	100	100	100	100	100	110	
13	1.45	1.40	3.97	1.39	100	100	0.82	1.56	100	100	281.86	

$$\prod_{p=1}^{|ND|} \min_j \left(\prod_{j=1}^m dist_{p,j} \right) \quad (4.23)$$

En yakın olduğu nesne ile temsil edilmeyen bu nesnenin birleştirilmesi yapılır. Böylece temsil edilmeyen her bir nesne de kendisine en yakın nesne ile birleştirilmiş olur.

Bu hesaplamaların ardından eliminasyon tamamlanmaktadır. Örnekte veri setinde olan 4 nesne elde edildiği görülmektedir. Eliminasyon yapıldıktan sonra “Objects” tablosundaki μ_d , standart sapma, μ_{d_H} vb sonuçlar da yeni nesne sayısına ve son değerlere göre güncellenmektedir. Sonuçlar da kullanıcıya çıktı olarak yansıtılmaktadır. Güncel sonuçların sunulduğu program ekranı Şekil 4.8’de gösterilmiştir. Oluşan detaylar Çizelge 4.4’de sunulmuştur.

Çizelge 4.4 ayıklanan nesnelerin ardışık olarak dönüşümünün bir örneğini sunmaktadır.

The screenshot shows the 'Initialize' window of a software application. The main table displays object detection data for 'Browse1.mpg' video. The table has columns: Dataset, Frame, Object, h, w, xc, yc, Age, Visible_Count, and Invisible_Count. The data shows 10 frames with various object counts and coordinates. Below the table, there are three sections: 'General Computation Results' showing total frame count (672) and object count (953); 'Compare Computation' showing a comparison of 4 objects with their Euclidean Avg, Std, and H Avg; and 'Detection' showing a list of detected objects with their IDs, Object names, Dataset, Frame, H, W, and xc.

Şekil 4.8. Yazılım eliminasyon ekranı.

4.4. DEĞERLENDİRME

CAVIAR veri setinden “Browsing” tanımlamasının altındaki Browse1.mpg videosu üzerinde görüntü ve veri işleme yapılmıştır. 13 nesneden birbiriyle aynı olan nesneler elendiğinde geriye toplam 4 nesne kalmıştır. Kullandığımız yöntem gereği sadece hareket eden nesneler ve hareketleri yakalandığından CAVIAR veri setindeki hareket eden nesneleri barındıran çerçeveler ile kıyaslama yapılmıştır. Browse1 videosuna ait bütün çerçeveler içerisinde genel var-yok analizinde (672-953 çerçeve) çalışmanın ön doğruluk değeri $(672 \cdot 100 / 953)$ %70,51 olarak bulunmuştur. Her bir çerçeve için yapılan var-yok analizinde ise; CAVIAR veri setinde yakalanmış nesnelerin çalışmamızda yanlışlıkla yakalanmama oranı (yalancı negatiflik – false negative) %45 olarak hesaplanmıştır. Bu durumda ön hassaslık değeri %55 olarak ortaya çıkmaktadır.

Önerilen yaklaşım tüm videolar için toplamda ve genel var-yok analizinde 7.276 çerçeveden yaklaşımımız 6.122 adet çerçevede nesne yakalamış, çalışmanın ön doğruluk değeri çerçeve bazında %84,14 olarak ortaya çıkmıştır.

12 videonun her biri için yakalanan hareketli nesne sayısı var-yok analizinde CAVIAR veri setinde yakalanmış nesnelerin çalışmamızda yanlışlıkla yakalanmama oranı

(yalancı negatiflik – false negative) 0,71 olarak hesaplanmaktadır. Böylece hassaslık değeri %29 olarak hesaplanmaktadır.

Çizelge 4.4. Nesne eliminasyon sonuçları.

Video	Nesne	Hız Tabanlı		Tahmin Tabanlı	
		Ara Düğüm	Sonuç	Ara Düğüm	Sonuç
Browse1	13	7	4	12→2	2
	12	13→7	4		2
	11	8	1	8→7→6→5	2
	10	8	1	9→8→7→6→5	2
	9	8	1	8→7→6→5	2
	8		1	7→6→5	2
	7		4	6→5	2
	6		2	5	2
	5		2		2
	4	3→2	1	3→2	1
Browse2	3	2	1	2	1
	2		1		1
Walk1	5		3	3	2
	4		3		1
	3				2
Walk2	2		3		
	4		3	3	2
	3		2		2
Browse3	5	2	1	4	1
Browse4	5		1	4→3	2
	4	3	2	3	2
	3		2		2
Browse_WhileWaiting1	2		1		1
Browse_WhileWaiting2	5		1		3
	4		2		2
	3		2		1
	2				1
Rest_FallOnFloor	7	4	2	6→5→2	1
	6	7→4	2	5→2	1
	5		2	2	1
	4		2		1
	3				1
	2				1
Rest_SlumpOnFloor	1		3		
	8	5	4	6	4
	7	6	5	6	4
	6	5	4		4
Rest_WiggleOnFloor	5		4		4
	6		2	5→4→3	2
	5	4	2	4→3	2
	4		2	3	2
Walk3	3	4	2		2
	4		1	3→2	1
	3	2	1	2	1
	2		1		1

Her bir videonun her bir çerçevesi için var-yok analizi yapmak hesaplama maliyeti açısından anlamlı değildir. Değerlendirme açısından sadece Browse1 videosuna uygulanmıştır. Nesnelerin çalışmamızda yanlışlıkla yakalanmama oranı 0,45 hassaslık değerinin %55 olduğu ifade edilmiştir.

Çalışmada [19] elde edilen verilerin standart sapma değerlerinin çok yüksek olduğu görülmüştür. Hassasiyet sonucunun da düşük olması gelecek çalışmalarda bu bölümün iyileştirilmesi üzerinde durulması gerektiğini göstermektedir.



BÖLÜM 5

MODELLEME, SPARQL İLE SORGULAMA VE ANLAMLANDIRMA

Bu bölümde üç alt bölüm yer almaktadır. Birinci bölümde öğrenme ve test veri setlerinin nasıl modellendiği aktarılacaktır. İkinci bölümde modellenmiş ve ontolojiye dönüştürülmüş veriler üzerinde geliştirilen SPARQL sorguları ve son bölümde ise bu sorgular sonucunda elde edilen anlamlandırma sonuçları sunulmuştur.

Bu bölüm çalışma [41] ve [42] ile elde edilen tüm detayları ayrıntılı bir biçimde aktarmaktadır.

5.1. MODELLEME

Öğrenme verisi olarak kullandığımız CAVIAR veri seti ve test verisi üzerinde Turtle formatında modelleme yapılmıştır. Test verisi olarak kullandığımız CAVIAR veri seti üzerinde uygulanan modelleme sonucu oluşan veri dosyaya yazılmıştır. Verinin örüntüsü Şekil 5.1’de görülmektedir. Burada dosyanın tamamı sunulamamıştır. Sadece Browse1 dosyasının içerisindeki 0. çerçeve (ilk çerçeve) ve 3 adet nesne (object) için detay görülmektedir. Normalde Browse1 videosu yaklaşık 41,5 saniye uzunluğunda, toplam 1043 adet çerçeve ve 4 adet nesne içermektedir.

Önce çerçeve sonra ilgili çerçevedeki her bir nesne birbirleriyle ilintilenerek Turtle formatında modellenmiştir. O , h , w , xc ve yc de yapıya eklenerek ontolojiye dönüştürülmüştür.

```
@prefix cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#> .
@prefix dbo: <http://semed.karabuk.edu.tr/ontology/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

cd:0 cd:frame dbo:Frame .

cd:0 dbo:frame "0"^^xsd:int .
cd:0 cd:object dbo:Object .
cd:0 cd:orientation "174"^^xsd:int .
cd:0 cd:h "19"^^xsd:int .
cd:0 cd:w "48"^^xsd:int .
cd:0 cd:xc "287"^^xsd:int .
cd:0 cd:yc "277"^^xsd:int .

cd:1 dbo:frame "0"^^xsd:int .
cd:1 cd:object dbo:Object .
cd:1 cd:orientation "173"^^xsd:int .
cd:1 cd:h "34"^^xsd:int .
cd:1 cd:w "54"^^xsd:int .
cd:1 cd:xc "210"^^xsd:int .
cd:1 cd:yc "242"^^xsd:int .

cd:2 dbo:frame "0"^^xsd:int .
cd:2 cd:object dbo:Object .
cd:2 cd:orientation "158"^^xsd:int .
cd:2 cd:h "25"^^xsd:int .
cd:2 cd:w "33"^^xsd:int .
cd:2 cd:xc "84"^^xsd:int .
cd:2 cd:yc "180"^^xsd:int .
```

Şekil 5.1. Veri örüntüsü.

Burada çerçeve ve nesne sınıfları ile ilişkilendirilen varlıklar ve özelliklerinin tanımlandığı görülebilir. Örneği sunulan örüntü ile tüm öğrenme ve test veri seti modellenerek üzerinde çalışılan ontolojiler bu şekilde elde edilmiştir. İki veri setinin benzer şekilde modellenmesi sayesinde aynı sorgular ile üzerlerinde sorgulamalar yapılması sağlanmıştır. Bu örüntü üzerinde çalıştırılan sorgular ve elde ettikleri sonuçlar aşağıda detaylandırılacaktır.

5.2. SPARQL İLE SORGULAMA

SPARQL RDF için sorgu dilidir [4]. W3C mevcut başkanı ve Web'in mucidi Tim Berners-Lee'nin "Semantik Web'i SPARQL olmaksızın kullanmaya çalışmak bir ilişkisel veri tabanını SQL olmaksızın kullanmaya çalışmak gibidir" ifadesi SPARQL'un semantic Web'deki önemini vurgulamaktadır [43].

RDF bir veri formatı değildir, veri dosyalarını kaydetmek için sentaksların seçimi ile birlikte bir modelidir. SPARQL ilişkisel veriyi sorgulamak için değil RDF veri modeline uygun veriyi sorgulamak için tasarlanmıştır.

Bir SPARQL sorgusu tipik olarak "Verilerin alt kümesinden bu şartlara uyan bilgi parçasını istiyorum." yapısındadır [19].

5.2.1. Yapılan Temel Sorgular

Ontolojiler üzerinde adım adım aşağıdaki SPARQL sorguları geliştirilmiş ve çalıştırılmıştır.

Şekil 5.2 veri setinden rastgele 10 adet çerçeveyi seçip listeleyen sorguyu göstermektedir. Sorgunun test veri seti sonuçları ise Şekil 5.3'de gösterilmiştir.

```
PREFIX cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?frame
WHERE {
    ?frame cd:frame dbo:Frame .
} LIMIT 10
```

Şekil 5.2. ex001 Sorgusu.

Veri setindeki bütün nesneleri elde etmek istediğimizde Şekil 5.4'de verilen sorgunun çalıştırılması yeterli olacaktır. Sorgunun öğrenme verisi üzerinde çalıştırıldığında dönen sonuçları Şekil 5.5 gösterilmiştir. Şekil 5.2'de ve Şekil 5.4'de verilen ex001 ve ex002 isimli sorgular geliştirilerek aşağıdaki sorgular türetilmiştir.

```

-----
| frame |
=====
| cd:383 |
| cd:888 |
| cd:240 |
| cd:790 |
| cd:170 |
| cd:344 |
| cd:811 |
| cd:201 |
| cd:128 |
| cd:493 |
-----

```

Şekil 5.3. ex001 Sorgu sonucu.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object
WHERE {
    ?object cd:object dbo:Object .
}

```

Şekil 5.4. ex002 Sorgusu.

```

-----
| object |
=====
| cd:3 |
| cd:2 |
| cd:1 |
| cd:0 |
-----

```

Şekil 5.5. ex002 Sorgu sonucu.

Videoda yer alan çerçeve sayısını sorgulamak önemlidir. Şekil 5.6’da toplam çerçeve sayısını veren sorguyu gösterilmiştir. Sorgunun öğrenme verisi sonucu Şekil 5.7’de gösterilmiştir.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT (COUNT(?frame) AS ?num_frame)
WHERE {
    ?frame cd:frame dbo:Frame .
}

```

Şekil 5.6. ex003 Sorgusu.

```

-----
| num_frame |
=====
| 1043      |
-----

```

Şekil 5.7. ex003 Sorgu sonucu.

Toplam nesne sayısını veren sorguyu Şekil 5.8’de verilmiştir. Bu sorgu bir videoda yer alan nesne sayısını döndürür. Sorgunun öğrenme verisinden Browse1 videosu üzerindeki sonucu Şekil 5.9’de gösterilmiştir.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT (COUNT(?object) AS ?num_object)
WHERE {
    ?object cd:object dbo:Object .
}

```

Şekil 5.8. ex004 Sorgusu.

```

-----
| num_object |
=====
| 4          |
-----

```

Şekil 5.9. ex004 Sorgu sonucu.

Şekil 5.10, rastgele 10 tane nesne ve bu nesnelerin yer aldığı çerçeveleri listeleen sorguyu göstermektedir. Sorgunun test veri seti üzerinde çalıştırıldığında oluşan sonuçları da Şekil 5.11’de gösterilmiştir.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object ?frame
WHERE {
    ?object cd:object dbo:Object ;
           dbo:frame ?frame .
} LIMIT 10

```

Şekil 5.10. ex005 Sorgusu.

object	frame
cd:3	"773"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"980"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"913"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"981"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"830"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"497"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"1023"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"804"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"737"^^<http://www.w3.org/2001/XMLSchema#int>
cd:3	"805"^^<http://www.w3.org/2001/XMLSchema#int>

Şekil 5.11. ex005 Sorgu sonucu.

Her bir nesnenin yer aldığı toplam çerçeve sayısı da önemli bir bilgidir. Nesnenin yer aldığı toplam çerçeve sayısını veren sorgu Şekil 5.12’de verilmiştir. Sorgunun sonuçları da Şekil 5.13’de verilmiştir. Örneğin “0” ID numaralı nesne toplam 241 adet, “2” ID numaralı nesne 1043 adet çerçevede yer almıştır.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/CAVIARdataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object (COUNT( DISTINCT?frame) AS ?num_frame)
WHERE {
    ?object cd:object dbo:Object ;
           dbo:frame ?frame .
} GROUP BY (?object)

```

Şekil 5.12. ex006 Sorgusu.

Bu sorgular ilerleyen sorgular için temel teşkil etmektedir. Sonraki sorgular bu sorgular temel alınarak geliştirilmiştir.

object	num_frame
cd:0	241
cd:1	244
cd:2	1043
cd:3	709

Şekil 5.13. ex006 Sorgu sonucu.

Aşağıda geliştirilen sorgular zaman parametresi ile çalıştırılarak sonuçlar alınmıştır. Böylece çalışan sorguya makinenin tepki süresi ölçülebilir olmakta ve kıyaslanabilir ölçek ortaya çıkmaktadır. Şekil 5.12’de sunulan ex006 sorgusu güncellenerek tekrar yazılmış ve Şekil 5.14’de sunulmuştur. Güncellenmiş sorgu ex006’nın test verisi sonucu Şekil 5.15’de sunulmuştur.

```
PREFIX cd: <http://semed.karabuk.edu.tr/ns/dataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object (COUNT( DISTINCT?frame) AS ?num_frame)
WHERE {
    ?object cd:object dbo:Object ;
            dbo:frame ?frame .
} GROUP BY (?object)
ORDER BY DESC(?num_frame)
LIMIT 4
```

Şekil 5.14. Güncellenmiş ex006 sorgusu.

Önceki bölümde sunulan sorguların (Sorgu ex003-Şekil 5.6, Sorgu ex004-Şekil 5.8, Sorgu ex006-Şekil 5.12) test verisi üzerindeki sonuçları ve çalışma süreleri Çizelge 5.1’de sunulmuştur. Sorgu ex006 tekil sonuç yerine müstakil tablo döndürdüğü için burada verilmemiş, 15 numaralı video için çalıştırılan sorgunun sonucu Şekil 5.16’de sunulmuştur.

Çizelge 5.1’de görüldüğü üzere, 1’er saatlik videolar olmasına rağmen, test verilerinde çok sayıda çerçeve ve nesne bulunmaktadır. Örneğin 1 numaralı videoda 61,693 adet çerçevede toplam 4,849 adet nesne tespit edilmiştir. 15 numaralı videodaki 2,404 ID numaralı nesne toplam 2,440 adet çerçevede yer almıştır.

```

-----
| object | num_frame |
=====
| cd:2   | 1043      |
| cd:3   | 709       |
| cd:1   | 244       |
| cd:0   | 241       |
-----
Time: 0,462 sec

```

Şekil 5.15. Güncellenmiş ex006 sorgusunun sonucu.

Çizelge 5.1. Test verisi sonuçları ve tepki süresi.

Video No	Sorgu ex003		Sorgu ex004		Sorgu ex006
	Sonuç	Süre (sn)	Sonuç	Süre (sn)	Süre (sn)
01	61693	0,56	4849	0,49	2,10
05	73482	0,59	2373	0,35	1,38
11	70286	0,55	2254	0,31	1,16
15	88236	0,96	2863	0,40	2,06

```

-----
| object | num_frame |
=====
| cd:2404 | 2440      |
| cd:1470 | 1816      |
| cd:1532 | 1556      |
| cd:1718 | 1383      |
-----
Time: 2,061 sec

```

Şekil 5.16. ex006 Sorgusun test verisi (Video no:15) sonucu.

Geliştirilen ex007 sorgusu her bir çerçevedeki toplam nesne sayısını bulmaktadır. Sorgu Şekil 5.17’de sunulmuştur. Örnek olarak test verisi üzerinde döndürdüğü

sonuçlar Şekil 5.18’de verilmiştir. 0-9 ID numaralı çerçevelerde toplam 3’er adet nesne yer almaktadır.

```
PREFIX cd: <http://semed.karabuk.edu.tr/ns/dataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?frame (COUNT( DISTINCT?object) AS ?num_object_in_frame)
WHERE {
    ?object cd:object dbo:Object ;
           dbo:frame ?frame .
} GROUP BY (?frame)
ORDER BY ASC(?frame)
LIMIT 10
```

Şekil 5.17. ex007 Sorgusu.

frame	num_object_in_frame
"0"^^<http://www.w3.org/2001/XMLSchema#int>	3
"1"^^<http://www.w3.org/2001/XMLSchema#int>	3
"2"^^<http://www.w3.org/2001/XMLSchema#int>	3
"3"^^<http://www.w3.org/2001/XMLSchema#int>	3
"4"^^<http://www.w3.org/2001/XMLSchema#int>	3
"5"^^<http://www.w3.org/2001/XMLSchema#int>	3
"6"^^<http://www.w3.org/2001/XMLSchema#int>	3
"7"^^<http://www.w3.org/2001/XMLSchema#int>	3
"8"^^<http://www.w3.org/2001/XMLSchema#int>	3
"9"^^<http://www.w3.org/2001/XMLSchema#int>	3

Time: 0,494 sec

Şekil 5.18. ex007 Sorgu sonucu.

Geliştirilen ex008 sorgusu her bir nesnenin başlangıç çerçevesini bulmaktadır. Geliştirilen model açısından başlangıç çerçevesinin bulunması önemlidir. Nesnenin kadrja ilk giriş anı-çerçevesi olarak tanımlanabilir. Ne anlama geldiği ve nasıl kullanıldığı bölümün devamında daha detaylı açıklanacaktır. Sorgu ex008, Şekil 5.19’da, sonuçları ise Şekil 5.20’de sunulmuştur.

Geliştirilen ex009 sorgusu her bir nesnenin bitiş çerçevesini bulmaktadır. Sorgu, Şekil 5.21’de, öğrenme verisi üzerinde çalışan sonuçları ise Şekil 5.22’de gösterilmiştir.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/dataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object (MIN (?frame) as ?minFrame)
WHERE {
    ?object cd:object dbo:Object ;
           dbo:frame ?frame .
} GROUP BY (?object)

```

Şekil 5.19. ex008 Sorgusu.

```

-----
| object | minFrame |
=====
| cd:3   | "334"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2   | "0"^^<http://www.w3.org/2001/XMLSchema#int>   |
| cd:1   | "0"^^<http://www.w3.org/2001/XMLSchema#int>   |
| cd:0   | "0"^^<http://www.w3.org/2001/XMLSchema#int>   |
-----
Time: 0,369 sec

```

Şekil 5.20. ex008 Sorgu sonucu.

```

PREFIX cd: <http://semed.karabuk.edu.tr/ns/dataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object (MAX (?frame) as ?maxFrame)
WHERE {
    ?object cd:object dbo:Object ;
           dbo:frame ?frame .
} GROUP BY (?object)

```

Şekil 5.21. ex009 Sorgusu.

```

-----
| object | maxFrame |
=====
| cd:3   | "1042"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2   | "1042"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:1   | "243"^^<http://www.w3.org/2001/XMLSchema#int>  |
| cd:0   | "240"^^<http://www.w3.org/2001/XMLSchema#int>  |
-----
Time: 0,400 sec

```

Şekil 5.22. ex009 Sorgu sonucu.

Sorgu ex009, her bir nesnenin son çerçevesini bulmaktadır. Yine geliştirilen model açısından nesnenin kadrajdan çıkış anını-çerçevesini bulmak da önemlidir. Sorgu ex008 ve ex009 birlikte kullanılarak geliştirilen modelin bir ayağı olarak kullanılmıştır. Detaylar ilerleyen kısımda açıklanmıştır.

5.2.2. Yapılan Kompleks Sorgular

Bölüm 5.1.1’de bahsedilen nesne ve çerçeve sayıları üzerinde düşünüldüğünde bu sayıların sorgu sonuçlarının kullanıcıya ve okuyucuya tek seferde ve bir anlam bütünlüğünde aktarılması için normalizasyon yapılması gerekmektedir. İlgili sonuçların gösteriminin yapılmasını sağlamak için yazılan sorgular revize edilerek uygulanmıştır.

Şekil 5. 23’de her bir nesnenin yer aldığı çerçevelerin toplam sayısını listeleyen ex006 sorgusu ile her bir nesnenin başlangıç çerçevesini bulan ex008 sorgusu birleştirilerek sorgu ex086 geliştirilmiştir. Sorgu en çok sayıda çerçevede yer alan ilk 10 nesnenin başlangıç çerçevesini bulmaktadır.

```
PREFIX cd: <http://semed.karabuk.edu.tr/ns/dataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object (MIN (DISTINCT?frame) as ?minFrame)
WHERE {
    {
        SELECT ?object (COUNT( DISTINCT?frame) AS ?num_frame)
        WHERE {
            ?object cd:object dbo:Object ;
                    dbo:frame ?frame .
        } GROUP BY (?object)
        ORDER BY DESC(?num_frame)
        LIMIT 10
    }
    ?object cd:object dbo:Object ;
            dbo:frame ?frame .
} GROUP BY (?object)
ORDER BY (?minFrame)
LIMIT 10
```

Şekil 5.23. ex0086 Sorgusu.

Bu sorguların sonuçlarının nasıl görüldüğünün anlaşılması için test verisi (barındırdığı çerçeve ve nesne büyüklüğü nedeniyle) üzerinde çalıştırılması gerekmektedir. Test verisi üzerinde çalıştırılmış ex0086 sorgu sonucu Şekil 5.24’de sunulmuştur.

```
-----  
| object | minFrame |  
=====  
| cd:2122 | "38816"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:2156 | "39500"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:2334 | "43047"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:2397 | "43988"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:2583 | "47404"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:2588 | "47436"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:3457 | "64687"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:3704 | "69945"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:3914 | "74338"^^<http://www.w3.org/2001/XMLSchema#int> |  
| cd:4594 | "85510"^^<http://www.w3.org/2001/XMLSchema#int> |  
-----  
Time: 1,687 sec
```

Şekil 5.24. Test verisi üzerinde ex0086 Sorgu sonucu.

Şekil 5. 25’de her bir nesnenin yer aldığı çerçevelerin toplam sayısını listeleyen ex006 sorgusu ile her bir nesnenin bitiş çerçevesini bulan ex009 sorgusu birleştirilerek sorgu ex0096 geliştirilmiştir. Sorgunu en çok sayıda çerçevede yer alan ilk 10 nesnenin bitiş çerçevesini bulmaktadır. Test verisi üzerinde çalıştırılmış ex0096 sorgu sonucu Şekil 5.26’da sunulmuştur.

5.2.3. Yapılan Görselleştirilmiş Kompleks Sorgular

Bölüm 5.2.2’de bahsedilen test videolarında yer alan ve 61,693 adet çerçeve üzerinde 4,849 adet nesne gibi sayıların sunulmasının imkansızlığı dikkate alınmıştır. Ekranda gösterilebilecek makul sayı hesaba katılarak geliştirilmiş sorguların nihai olarak görselleştirmesiyle sunulması sağlanmıştır.

Sorgu ex0096’nın güncellenerek tekrar yazılmış hali Şekil 5.27’de sunulmuştur. Nesnelerin yer aldığı çerçeve sayıları ve bu çerçeve sayılarının videoda nesnenin yer

alma süresine tekabül eden hali sorgulanmaktadır. Öğrenme verisi üzerinde çalıştırılan sorgunun sonucu Çizelge 5.2’de görülmektedir.

```
PREFIX cd: <http://semed.karabuk.edu.tr/ns/dataset#>
PREFIX dbo: <http://semed.karabuk.edu.tr/ontology/>
SELECT ?object (MAX (?frame) as ?maxFrame)
WHERE {
    {
        SELECT ?object (COUNT( DISTINCT?frame) AS ?num_frame)
        WHERE {
            ?object cd:object dbo:Object ;
                    dbo:frame ?frame .
        } GROUP BY (?object)
        ORDER BY DESC(?num_frame)
        LIMIT 10
    }
    ?object cd:object dbo:Object ;
            dbo:frame ?frame .
} GROUP BY (?object)
ORDER BY (?maxFrame)
LIMIT 10
```

Şekil 5.25. ex0096 Sorgusu.

```
-----
| object | maxFrame |
=====
| cd:2122 | "39456"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2156 | "39955"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2334 | "43498"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2397 | "44508"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2583 | "47908"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:2588 | "47939"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:3457 | "65182"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:3704 | "70578"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:3914 | "75032"^^<http://www.w3.org/2001/XMLSchema#int> |
| cd:4594 | "85986"^^<http://www.w3.org/2001/XMLSchema#int> |
-----
Time: 2,393 sec
```

Şekil 5.26. Test verisi üzerinde ex0096 Sorgu sonucu.

En çok sayıda çerçevede yer alan ilk 10 nesnenin başlangıç çerçevesini bulan sorgu ex0086'ya, başlangıç çerçevesindeki xc ve yc koordinat noktalarını ekleyen sorgu ex008610 Şekil 5.28'de sunulmuştur. Öğrenme verisi üzerindeki sonucu ise Çizelge 5.3'de gösterilmiştir.

```
PREFIX d: <http://semed.karabuk.edu.tr/ns/data#>
PREFIX fo: <http://semed.karabuk.edu.tr/ns/frameobject#>
SELECT ?object (COUNT( DISTINCT?frame) AS ?num_frame) (?num_frame / 25 as
?time)
WHERE {
    ?s fo:object ?object ;
        fo:frameInclude ?frame .
} GROUP BY (?object)
ORDER BY DESC(?num_frame)
LIMIT 10
```

Şekil 5.27. Güncellenmiş ex0096 Sorgusu.

Çizelge 5.2. Güncellenmiş ex0096 Sorgusu sonucu.

Nesne ID	Çerçeve Sayısı	Süre (sn)
0	241	9,64
1	244	9,76
2	1043	41,72
3	709	28,36

⌚ Çalışma süresi: 0,374 sn

Çizelge 5.3. ex008610 Sorgusu sonucu.

Nesne ID	Başlangıç Çerçevesi	Xc	Yc
0	0	287	277
1	0	210	242
2	0	84	180
3	334	26	164

⌚ Çalışma süresi: 0,483 sn

En çok sayıda çerçevede yer alan ilk 10 nesnenin bitiş çerçevesini bulan sorgu ex0096'ya, başlangıç çerçevesindeki xc ve yc koordinat noktalarını ekleyen sorgu ex009610 Şekil 5.29'da sunulmuştur. Öğrenme verisi üzerindeki sonucu ise Çizelge 5.4'de gösterilmiştir.

Çizelge 5.4. ex009610 Sorgusu sonucu.

Nesne ID	Bitiş Çerçevesi	Xc	Yc
0	240	314	286
1	243	30	168
2	1042	114	217
3	1042	203	97

⌚ Çalışma süresi: 0,547 sn

```
PREFIX d: <http://semed.karabuk.edu.tr/ns/data#>
PREFIX fo: <http://semed.karabuk.edu.tr/ns/frameobject#>
SELECT DISTINCT?object ?minFrame ?xc ?yc
WHERE {
  {
    SELECT ?object (MIN (DISTINCT?f) as ?minFrame)
    WHERE {
      {
        SELECT ?object (COUNT( DISTINCT?frame) AS
?num_frame)
        WHERE {
          ?s fo:object ?object ;
            fo:frameInclude ?frame .
        } GROUP BY (?object)
        ORDER BY DESC(?num_frame)
        LIMIT 10
      }
      ?s fo:object ?object ;
        fo:frameInclude ?frame .
      ?frame fo:frame ?f .
    } GROUP BY (?object)
    ORDER BY (?minFrame)
    LIMIT 10
  }
  {
    ?f fo:frame ?minFrame .
    ?d fo:frameInclude ?f .
    ?d fo:object ?object ;
      fo:xc ?xc ;
      fo:yc ?yc .
  }
} ORDER BY (?object)
LIMIT 10
```

Şekil 5.28. ex008610 Sorgusu.

5.3. ANLAMLANDIRMA

CAVIAR veri setindeki {appearance} niteliği için {appear, disappear, visible} hipotez değerleri bulunmaktadır. Bahsedilen nitelikler için ilgili değerlerin anlamlandırılması (reasoning) ile sonuç çıkarılması sağlanacaktır. Elde edilen kurallar mevcut ve gelecek ontolojilere uygulanacaktır. Örnek olarak, CAVIAR veri setindeki Browse1 videosunun 0 ID numaralı çerçevesinde nesne 0 görünmüş ve çerçeve 240'da da kaybolmuştur.

Ontolojiler üzerinde önceki bölümde sunulan SPARQL sorguları ile her bir nesne için minimum, maksimum ve toplam çerçeve numaraları σ_{min_i} , σ_{max_i} , $\sigma_{TotalFrameNumber_i}$ hesaplanmıştır. Tutarlılığı kontrol etmek için bu değerlerin hepsi bir arada yorumlanmış ve sonrasında nesnelerin etiketlenmesi önerilmiştir.

Tutarlılık için her bir videoda her bir nesne için toplam çerçeve sayısı Denklem 5.1 ile hesaplanmıştır.

$$\Sigma_{TotalFrameNumber_i} = \sigma_{max_i} - \sigma_{min_i} + 1 \quad (5.1)$$

Çerçeve σ_{min_i} ortaya çıktı, görüldü anlamında “appear” olarak; çerçeve σ_{max_i} çıktı, gözden kayboldu anlamında “disappear” olarak etiketlenebilir. Eğer denklem (5.1)'in sonucu ile $\sigma_{TotalFrameNumber_i}$ sorgu sonucu eşit ise σ_{min_i} ve σ_{max_i} arasındaki çerçeveler görünür anlamında “visible” olarak etiketlenebilir ve i nesnesi doğru etiketlenmiş nesne olarak, aksi halde ise yanlış etiketlenmiş nesne olarak sınıflandırılır.

Tezde geliştirilen ilk yaklaşım budur. Yaklaşım için hipotez bir alt basamakta (bilgisayarlı görü) yakalanan – izlenen nesnelerin bir üst katmanda etiketlenmesi doğru bir şekilde yapılır şeklindedir.

CAVIAR veri setinden seçilen 12 adet video üzerine önerilen yöntem uygulanmıştır. Birinci yaklaşıma ait doğruluk sonuçları Çizelge 5.5'de sunulmuştur. Doğruluk Denklem 5.2 ile hesaplanmıştır.

$$Dog_i = (TP_i + TN_i)/(TP_i + TN_i + FP_i + FN_i) \quad (5.2)$$

Öğrenme verisi sonuçları Çizelge 5.5’de sunulmuştur. Önerilen yöntemin yüzde yüze yakın başarımla elde ettiği görülmüştür. Bu sonuç yöntemin test verisinde kullanılabileceği anlamına gelmektedir.

```

PREFIX d: <http://semed.karabuk.edu.tr/ns/data#>
PREFIX fo: <http://semed.karabuk.edu.tr/ns/frameobject#>
SELECT DISTINCT ?object ?maxFrame ?xc ?yc
WHERE {
    {
        SELECT ?object (MAX (DISTINCT ?f) as ?maxFrame)
        WHERE {
            {
                SELECT ?object (COUNT( DISTINCT ?frame) AS
?num_frame)
                WHERE {
                    ?s fo:object ?object ;
                        fo:frameInclude ?frame .
                } GROUP BY (?object)
                ORDER BY DESC(?num_frame)
                LIMIT 10
            }
            ?s fo:object ?object ;
                fo:frameInclude ?frame .
            ?frame fo:frame ?f .
        } GROUP BY (?object)
        ORDER BY (?maxFrame)
        LIMIT 10
    }
    {
        ?f fo:frame ?maxFrame .
        ?d fo:frameInclude ?f .
        ?d fo:object ?object ;
            fo:xc ?xc ;
            fo:yc ?yc .
    }
} ORDER BY (?object)
LIMIT 10

```

Şekil 5.29. ex009610Sorgusu.

Çizelge 5.5. Etiketleme doğruluk sonuçları.

Video Adı	Sayı & Doğruluk		
	Doğru	Yanlış	Yüzde (%)
Browse_WhileWaiting1	3		100
Browse_WhileWaiting2	1		100
Browse1	4		100
Browse2	4		100
Browse3	5		100
Browse4	3		100
Rest_FallOnFloor	3		100
Rest_SlumpOnFloor	4		100
Rest_WiggleOnFloor	3		100
Walk1	4		100
Walk2	6		100
Walk3	4	1	80
Ortalama			97,7

Walk3 videosunda beş adet nesne bulunmaktadır. 2 ve 3 ID numaralı nesnelerin aynı olduğu tespit edilmiştir. 2 ID numaralı nesne 772. çerçevede yok olduktan sonra, nesne 3 ortaya çıkmaktadır. Karşılaştırma videosunu etiketleyen uzmanlar tarafından kameranın açısı nedeniyle sahnede çok küçük olmasından dolayı 1097. çerçeveye kadar gözden kaçırıldığı ve etiketlenemediği düşünülmüştür. Bu durum 772. çerçeve ile 1097. çerçeve arasında bir boşluğa neden olmuştur. Eğer bu gürültü temizlenirse, önerilen modelin doğruluğu yüzde yüz kabul edilebilir.

Çizelge 5.6. Karşılaştırma verisinin tutarlılığı.

Nesne ID	“Appear” ile *	“Disappear” ile *	** Sayısı
0	0	240	241
1	0	243	244
2	0	1042	1043
3	334	1042	709

* Etiketlenmiş çerçeve numarası

** Etiketlenmiş toplam çerçeve

Öğrenme verisinin modele göre tutarlılığı Çizelge 5.6’da gösterilmiştir. Örneğin, 0 ID numaralı nesne 0 çerçevesinde ilk defa görünmüş, en son 240 numaralı çerçevede kadrajdan çıkmış, toplamda da 241 çerçevede bulunmuştur. Farklı SPARQL sorguları ile elde edilen toplam çerçeve sayısı, görünür oldu – kayboldu olarak etiketlenen

çerçevelerin doğruluğunu göstermektedir. Çizelge 5.5 ve 5.6 birlikte değerlendirildiğinde de bir doğrulama yapmak mümkün olmaktadır.

Örnek olarak karşılaştırma veri setinden 4 adet videonun nesneleri için hız sonuçları Çizelge 5.7’de sunulmuştur. Birbirine yakın olan hız değerlerinin anlamlandırılabilceği öngörülmüştür.

Çizelge 5.7. Nesnelerin hızı.

Nesne ID	Browse1	Browse2	Browse3	Browse4
0	2,95	12,63	28,07	31,64
1	19,94	0,93	10,96	13,37
2	1,14	2,27	18,27	22,39
3	6,67	9,70	17,52	
4			18,63	

Not: Farklı videoların objeleri aynı değildir. Hız birimi piksel/saniyedir.

Öğrenme verisi üzerinde doğruluğu başarılı bulunan model test verisi üzerinde de denenmiştir. Öğrenme verisi üzerinde doğruluğu başarılı bulunan model test verisi üzerinde de örnek bir video üzerinde denenmiştir. Sorgu ex008610 ve ex009610 sonuçları birleştirilerek Çizelge 5.8’de sunulmuştur. Test verisinin modele göre tutarlılığı gözlenebilmektedir. Örneğin 2122 ID numaralı nesne 38,816 çerçevesinde ilk defa görünmüş, en son 39,456 numaralı çerçevede kadrajdan çıkmış, toplamda da 641 çerçevede bulunmuştur. Farklı SPARQL sorguları ile elde edilen toplam çerçeve sayısı, görünür oldu – kayboldu olarak etiketlenen çerçevelerin doğruluğunu göstermektedir.

Çizelge 5.8. Test verisi etiketleme sonuçları.

Nesne ID	Çerçeve Sayısı	Süre (sn)	Başlangıç Çerçevesi	Xc	Yc	Bitiş Çerçevesi	Xc	Yc	Hız (piksel/sn)
2122	641	25,64	38816	552	1	39456	202	641	28,45
2156	456	18,24	39500	133	1	39955	591	456	35,39
2334	452	18,08	43047	564	1	43498	661	452	25,52
2397	521	20,84	43988	156	1	44508	380	521	27,17
2583	505	20,2	47404	177	1	47908	197	505	24,97
2588	504	20,16	47436	369	1	47939	217	504	26,06
3457	496	19,84	64687	477	1	65182	113	496	30,97
3704	634	25,36	69945	468	1	70578	624	634	25,71
3914	695	27,8	74338	273	1	75032	576	695	27,24
4594	477	19,08	85510	201	1	85986	10	477	26,88

Videoda en çok çerçevede yer alan 10 adet nesnenin ilk ve son görüldüğü çerçevelerde yer aldıkları koordinat noktaları, kadrada bulundukları çerçeve sayısı ve giriş – çıkış arasındaki ortalama hızları ilgili sorgularla hesaplanmıştır.

5.4. DEĞERLENDİRME

Nesnelerin hareketlerinin karakteristiği düşünüldüğünde, bazı hareketler çok benzerdir. Örneğin karşılaştırma veri setinde Browse3 videosundaki nesne 2 ve 3 çok benzer bir gidiş yoluna sahiptir. Sonuçlar analiz edilerek benzer bilgi elde edilebilir.

Nesnelerin sadece hız sonuçları analiz edildiğinde, Browse1 videosundaki 0 ID numaralı nesne ile Browse2 videosundaki 1 ID numaralı nesne çok benzer hıza sahiptir. Bu nedenle nesneler benzer bir eğilim sergiler. Browse1 videosundaki nesne 1 ve Browse3 videosundaki nesne 4 çok benzer hıza sahip olmalarına rağmen çok farklı karakteristik sergiler. Bu sonuç sadece hızı dikkate alarak yapılacak değerlendirmenin gidiş yollarının benzerliğini bulmada yüksek doğruluğa ulaşamayacağını göstermektedir. Diğer taraftan hız ve ortaya çıkma-kaybolma noktalarını (x_c , y_c ve hız) birlikte dikkate almak Browse3'deki nesne 2 ve 3'ün benzerliği gibi daha önemli bilgiyi işaret etmektedir.

Test verisi bahsedilen perspektiften değerlendirildiğinde belirgin bir bilgi yoktur. Nesnelerin giriş ve çıkış pozisyonlarının değiştiğini sonuçlara bakarak görülebilmektedir.

BÖLÜM 6

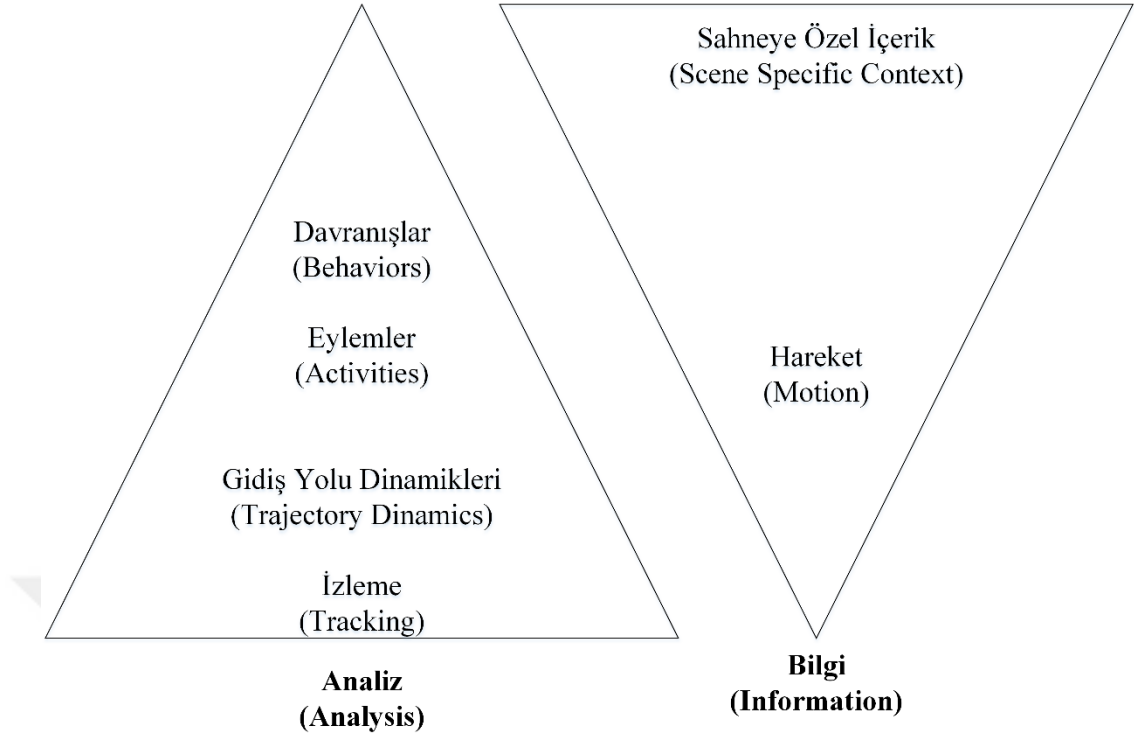
NESNE YOLU, BENZERLİK VE ANLAMLANDIRMA ÜZERİNE İLERİ YAKLAŞIMLAR

Semantik Web açısından video açıklama verilerinde yer alan sayısal değerler üzerinde çalışmanın zorlukları nedeniyle ileri düzey anlamlandırma yapmak da bir hayli zordur. Multimedya etki alanı üzerinde yapılabilecek anlamlandırma çalışmalarını arttırmak için bir arayışa girilmiştir. Bu nedenle nesnelerin gidiş yolları üzerinde durulmuştur. Gidiş yollarının benzerliklerinin tanımlanması, bu tanımlamalardan anlamlar üretilmesi ve daha özelleşmiş yaklaşımlar tezin bu bölümünde kapsamıştır.

Bütün bir video bir hikâyeyi anlatırken, bir sahne hikâyenin bir paragrafını tanımlar. Sahnenin (scene) her bir çekimi paragrafın bir cümlesidir. Her bir çerçeve (frame) ise cümlelerin kelimelerinden ibarettir [44]. Bu hiyerarşi sonraki bölümlerde içerik analizinin kavramsallaştırılmasını sağlayacaktır.

Analiz seviyeleri ve çıkarılacak bilgi arasındaki ilişki ters orantılıdır. Yüksek seviye etkinlik analizleri geniş miktarda etki alanı bilgisi gerektirirken, düşük seviye analizler çok az etki alanı bilgisini kullanır. Şekil 6.1 [45]'de sunulan piramitler bu görüşü tarif etmektedir. Analiz piramidinin en altında yer alan takip gibi düşük seviye işlemler, sadece hareket gibi çok küçük ölçekli bilgi kullanır. Analiz piramidinin üst kısımlarına çıkmak daha karmaşık olay gösterimleri sağlar fakat bunun yanında daha etki alanı bağımlı bilgi kullanımı gerektirir.

Gidiş izi analizleri düşük ve yüksek seviye analizler arasında bir ortalama sağlar. İlaveten karakteristik hareketler tekrar ederken, ilginç hareketler nadiren meydana gelir. Tekrar öğrenilen hareketler bağlamında hareket analizine imkân verir. Etki alanı bilgisine dayanmaktan ziyade, düşük seviye işaretler hareket analiz işlemini inşa etmekte kullanılır [45].



Şekil 6.1. Analiz seviyeleri ve bilgi arası ilişki.

2000'lere kadar yol tanıma probleminin aksine, tek tek video nesnelerini tespit ve takip etme ve çerçeveden çerçeveye benzerliklerini tanımlama problemi eski ve birçok araştırmacı tarafından yaygın olarak çalışılmış bir problemdir. Sonradan Makris ve Ellis tarafından rotalar, düğümler ve bir zarf ile modellenmiştir. Rotaların bazı ortalamaları düğümler ile gösterilir. Rota, gidiş izlerinin çeşitli varyasyonlarının örneklendiği bir zarf ile sınırlandırılmıştır. Rota düğümleri birbirinden yeniden örnekleme mesafesine eşit bölünme mesafesi ile ayrılmıştır. Bu araştırma ile yol tanıma denenmiş ilaveten de çoklu gidiş izleri rotalara dönüştürülmüştür. Böylece boyut azaltımı ve hareket tahmini sağlanmıştır [46].

6.1. GİDİŞ İZİ, ROTA VE YOL TANIMI

Bir gidiş izi sahne boyunca hareket eden nesnelerin konumlarının bilgisini içerir [47]. Yol nesnelerin nasıl hareket ettiklerini tanımlar [45]. Yol sahne odaklı özellik iken, rota sahne boyunca nesnelerin çok kullandığı iz olarak tanımlanır. Güvenlik sistemlerinde nesne yayalara işaret etmekle beraber bununla sınırlı değildir. İlaveten,

bir rota yayanın belirli bir başlangıç ve bitiş noktası bulunan bütün gidiş izlerini veya eğrisini gösterir [46].

6.1.1. Matematiksel Formülasyon

Takip sistemleri bir videodaki her çerçeveyi tarar ve her bir nesneyi tanımlar. S_T pozisyon, hız, görünüm veya benzer anahtar kelimelerden biri olabilir.

$$S_T = \{s_1, s_2, \dots, s_T\} \quad (6.1)$$

Denklem 6.1 dizisi nesnelerin takip edilmiş durumlarından üretilmiştir. Bir gidiş izi dinamik ölçümler kullanılarak gösterilir. Akış vektörü Denklem 6.2 olduğunda, yaygın gidiş izi gösterimi Denklem 6.3 olur.

$$f_t = [x^t, y^t, v_x^t, v_y^t, a_x^t, a_y^t]^T \quad (6.2)$$

$$F_T = \{f_1, f_2, \dots, f_T\} \quad (6.3)$$

Akış vektöründe t zamanda pozisyon, hız ve ivme sırasıyla $[x_c, y_c]$, $[v_{xc}, v_{yc}]$, $[a_{xc}, a_{yc}]$ ile gösterilir. Karakteristik olarak gürültüye neden olan ivme yerine, gösterim için genellikle sadece pozisyon veya pozisyon ve hız birlikte kullanılır. Eğer eşit boyutta iseler, iki gidiş izi arasındaki benzerlik ölçümü Denklem 6.4 ile hesaplanır [45].

$$d_E(F_i, F_j) = \sqrt{(F_i - F_j)^T (F_i - F_j)} \quad (6.4)$$

6.2. MODEL

Rotalar, benzer olsalar bile, gidiş yolları farklı hıza sahip olmaları nedeniyle farklı uzunluklara sahip olabilir [45]. Önerdiğimiz model bu durumu dikkate almaktadır. Akış vektörü Denklem 6.5 olduğunda, önerilen gidiş izi gösterimi denklem 6.6 akış dizisidir.

$$s_f = [o^f, h^f, w^f, xc^f, yc^f]^T \quad (6.5)$$

$$S_i = \{s_1, s_2, \dots, s_F\} \quad (6.6)$$

Burada f çerçevede $[xc, yc]$, $[h, w]$, $[o]$ sırasıyla pozisyon, boyut ve oriyantasyonu temsil etmektedir. Boyut azaltımı için, literatürün yerine bu çalışma yeni bir yaklaşım önermektedir. Denklem 6.5'in her elemanı Denklem 6.7 ile benzer şekilde h , w , xc , ve yc yeniden hesaplanır.

$$\mu_o = \sum_{j=1}^F o^j / F \quad (6.7)$$

Denklem 6.5, Denklem 6.8 haline dönüşür.

$$S_i = [\mu_o, \mu_h, \mu_w, \mu_{xc}, \mu_{yc}]^T \quad (6.8)$$

Önerilen boyut azaltımı uygulandıktan sonra, bütün gidiş izleri boyutları göz ardı edilerek, Denklem 6.9 ve 6.10 ile rahatlıkla karşılaştırılabilir.

$$d_E(S_i, S_j) = \sqrt{(S_i - S_j)^T (S_i - S_j)} \quad (6.9)$$

$$d_{WE}(S_i, S_j) = \sqrt{\alpha_i (S_i - S_j)^T (S_i - S_j)} \quad (6.10)$$

6.3. UZAKLIK

Uzaklık benzerlik ölçümü olarak gösterilebilir. Sayısal veriyi karşılaştırmak için temel uzaklık ölçümleri Öklid, Ağırlıklı Öklid, Mahalanobis ve Minkowski ölçümleridir. Bu metodlara ait $d(x, y)$ veya $d_\gamma(x, y)$ formülleri sırasıyla Denklem 6.11 - 6.14'de sunulmuştur [48]:

$$\sqrt{\sum_{i=1}^p (x_i - y_i)^2} \quad (6.11)$$

$$\sqrt{\sum_{i=1}^p \alpha_i (x_i - y_i)^2} \quad (6.12)$$

$$\sqrt{(x - y)^t \sum^{-1} (x - y)} \quad (6.13)$$

$$\left(\sum_{i=1}^p |x_i - y_i|^\gamma \right)^{\frac{1}{\gamma}} \quad (6.14)$$

6.4. VERİ HAZIRLAMA

Eğer bir gidiş izi göz atma, yere düşme vb. davranışlar sergileyen bir nesneye ait ise, o zaman gidiş izinde değişmeyen-tekrarlanan birçok ardışık veri olacaktır. Ardışık tekrarlı verilerin gidiş yollarının benzerliğindeki etkisi incelenmek istenmiştir. Bu nedenle veri setinden ardışık tekrarlı verinin hesaplamaya dahil edildiği ve hariç tutulduğu olmak üzere iki ayrı örüntü Denklem 6.15 ve 6.16'da oluşturulmuştur.

$$xc_i, yc_i \in [\dots, xc^f, yc^f] \text{ if } xc_i = xc_{i-1} \mid xc_i \neq xc_{i-1}, yc_i = yc_{i-1} \mid yc_i \neq yc_{i-1} \quad (6.15)$$

$$xc_i, yc_i \in [\dots, xc^f, yc^f] \text{ if } x \neq xc_{i-1}, yc_i \neq yc_{i-1} \quad (6.16)$$

Her iki örüntü de incelenip kıyaslanacaktır. Sonuçlardan tekrar eden verilerin çıkarılmasının ve elde tutulmasının etkileri tartışılacaktır.

6.5. NORMALİZASYON İŞLEMİ

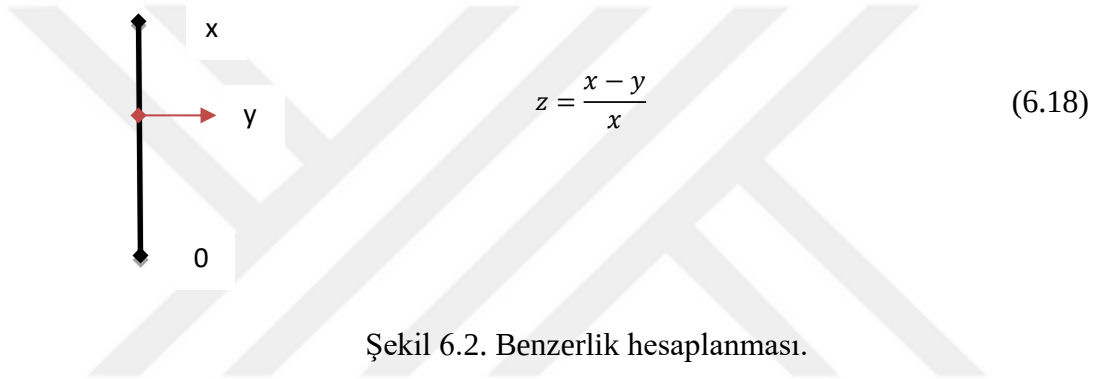
Çok farklı aralıklara sahip parametre verileri birlikte işlendiklerinde gürültüye neden olabilmektedir. Bunu önlemek için veri setine en yaygın normalizasyon yöntemi, Denklem 6.17 (min-max norm.) uygulanmıştır.

$$\partial_{norm_i} = \frac{\partial_i - \partial_{\min}}{\partial_{\max} - \partial_{\min}} (\partial_{norm_{\max}} - \partial_{norm_{\min}}) \quad (6.17)$$

0-180 arasında değişen yön, 1-85 arası değişen yükseklik, 3-68 arasında değişen genişlik ve 0-384 arasında değişen x piksel koordinat ile 0-288 arasında değişen y piksel koordinat değerleri normalizasyon uygulanarak 0-10 arasına indirgenmiştir. Böylece birbirlerine gürültü oluşturacak şekilde ağır basmaları önlenmiştir.

6.6. BENZERLİK YÜZDESİNİN HESAPLANMASI

Gidiş yollarının benzerlik yüzdesinin hesaplanması için probleme özgü bir yaklaşım geliştirilmiştir. Denklem (6. 18) x maksimum mesafe, y ölçülen mesafe olduğunda, z ile ölçülen benzerliğin yüzdesini vermektedir.



$$z = \frac{x - y}{x} \quad (6.18)$$

Şekil 6.2. Benzerlik hesaplanması.

6.7. NESNE HIZI

Videoda hareket eden nesnelerin hareketlerinin tanımlanmasında, gidiş izlerinin ve hareketin karakterinin belirlenmesinde hızlarının etkisi olduğu literatürdeki [49–51] çalışmalardan bilindiğinden, hızları hesaplanarak dikkate alınmıştır.

$$v = \frac{\sqrt{(xc_2 - xc_1)^2 + (yc_2 - yc_1)^2}}{\frac{Cerceve_{son} - Cerceve_{ilk}}{Cerceve\ Oranı}} \quad (6.19)$$

Nesnenin yer aldığı çerçeveler boyunca ortalama hızı Denklem 6.19 ile hesaplanmıştır. Hesaplanan hız piksel/saniye cinsindendir.

6.8. BENZERLİK

Elde edilen sonuçların güvenilirliğini test etmek amacıyla PCA ve PLS gibi metotlar ile kullanılan gidiş yollarının analiz edilmesi sağlanmıştır. Analiz sonuçları ile önerilen modelin sonuçları kıyaslanarak modelin güvenilirliği test edilmiştir.

6.8.1. PCA

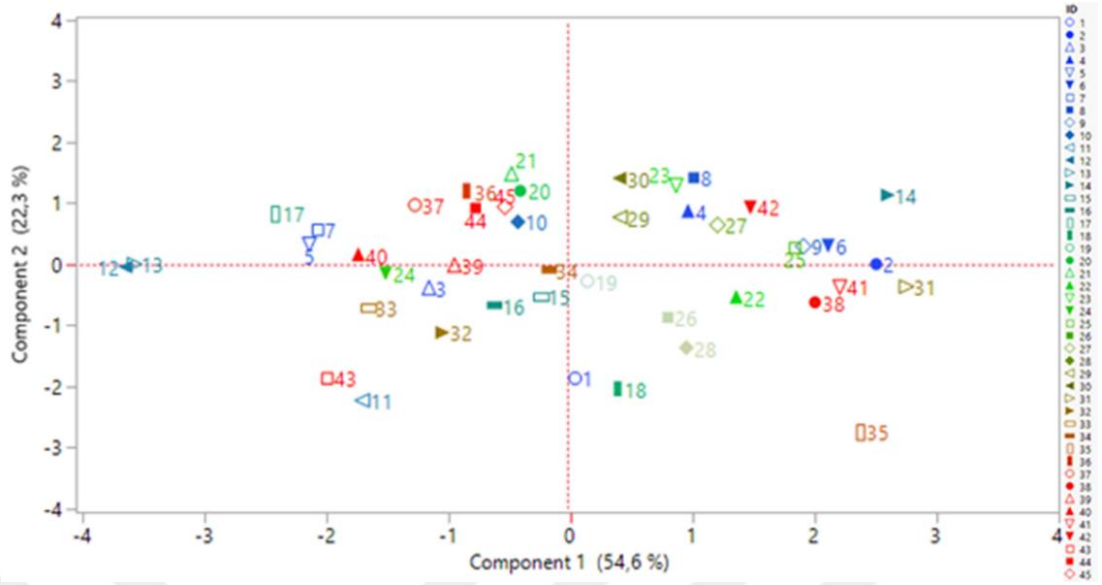
PCA (temel parçacık analizi) boyut azaltarak veriyi sıkıştırabilir ve görsel sunum sağlayarak güçlü bir analiz yapılmasına olanak verir. Algoritma tarafından ortalama hesaplanır ve her bir veri, boyutundan çıkartılarak 0 ortalamalı yeni bir veri seti elde edilir. Sonra yeni veri setinin kovaryans matrisi hesaplanır. Eğer kovaryans matris kare matris ise öz (eigen) vektörler ve kovaryans matrisin öz değerleri hesaplanır. Sonra öz vektörler öz değere göre en yüksekten en aza doğru sıralanır. Veriyi anlamlandırmak için algoritma tarafından öz vektörler ile çizgiler elde edilir. Veri setinin temel parçacığı en yüksek öz değerli öz vektör ile tanımlanır. Belirginlik sıralanmış öz vektörler ile elde edilir. Bazı parçacıklar önemsizlik nedeniyle atlanır. Bundan sonra, öz vektörlerini bütün listeden seçerek ve Denklem 6.20 matrisini oluşturarak özellik vektörü yapılandırılır.

$$FeatureVector = (eig_1, eig_2, \dots, eig_n) \quad (6.20)$$

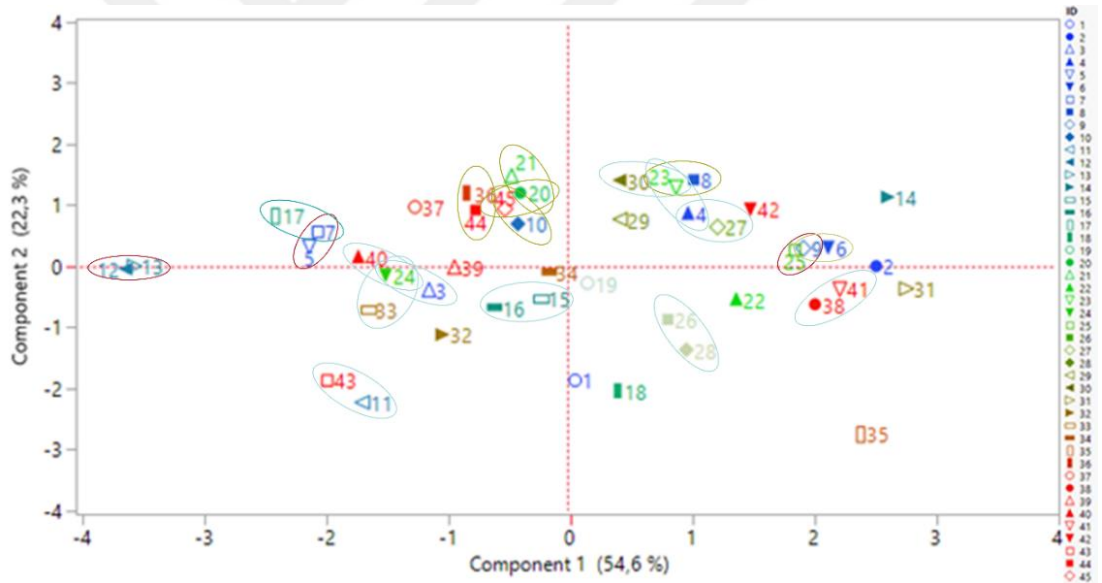
Son olarak, matris transpozu alınır ve soldaki orijinal veri seti ile çarpılarak transpozu alınır. *RowFeatureVector* transpozu alınmış öz matris ve *RowDataAdjust* ortalama ayarlanmış veri olduğu zaman, *FinalData* Denklem 6.21 ile elde edilir [52].

$$FinalData = RowFeatureVector \times RowDataAdjust \quad (6.21)$$

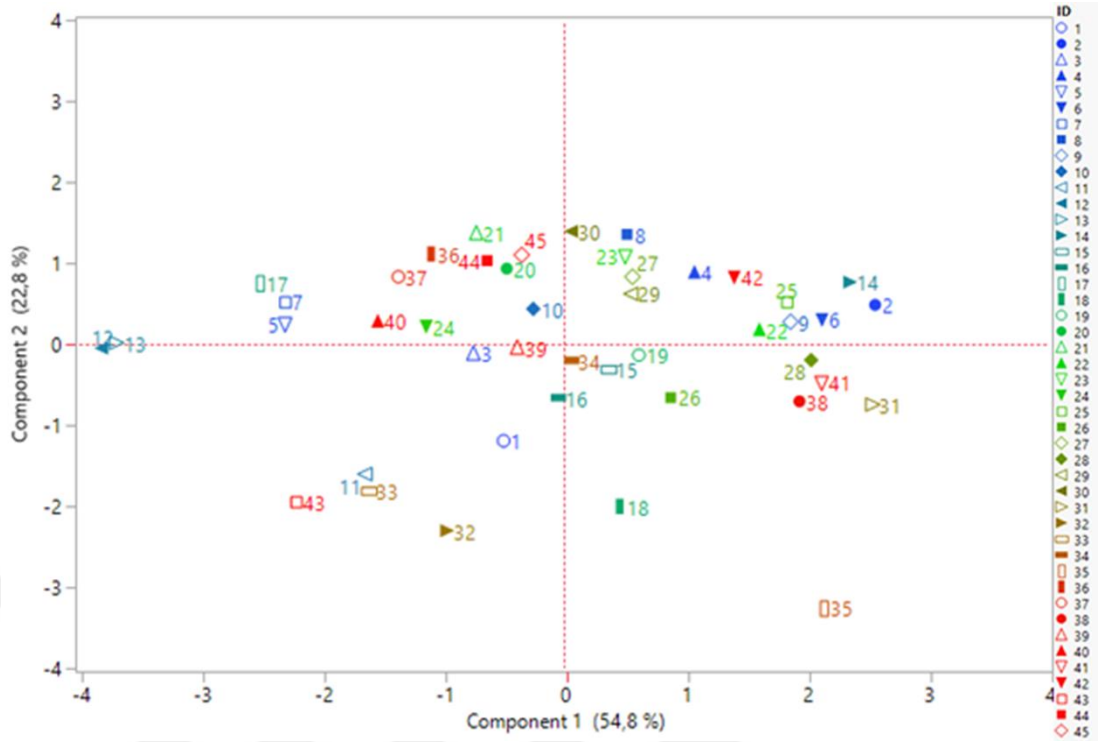
PCA JMP yazılımı kullanılarak hesaplanmış ve aşağıdaki grafikler oluşturulmuştur. Şekil 6.3, 6.4 ve 6.5 sırasıyla normal, işaretlenmiş ve tekrarsız verilerin grafiklerini, Şekil 6.6. ve 6.7 ise x_c ve y_c değerlerine göre ağırlıklandırılmış PCA grafiklerini sunmaktadır. Ağırlıklandırmanın etkisi gözlenmek istenmiştir.



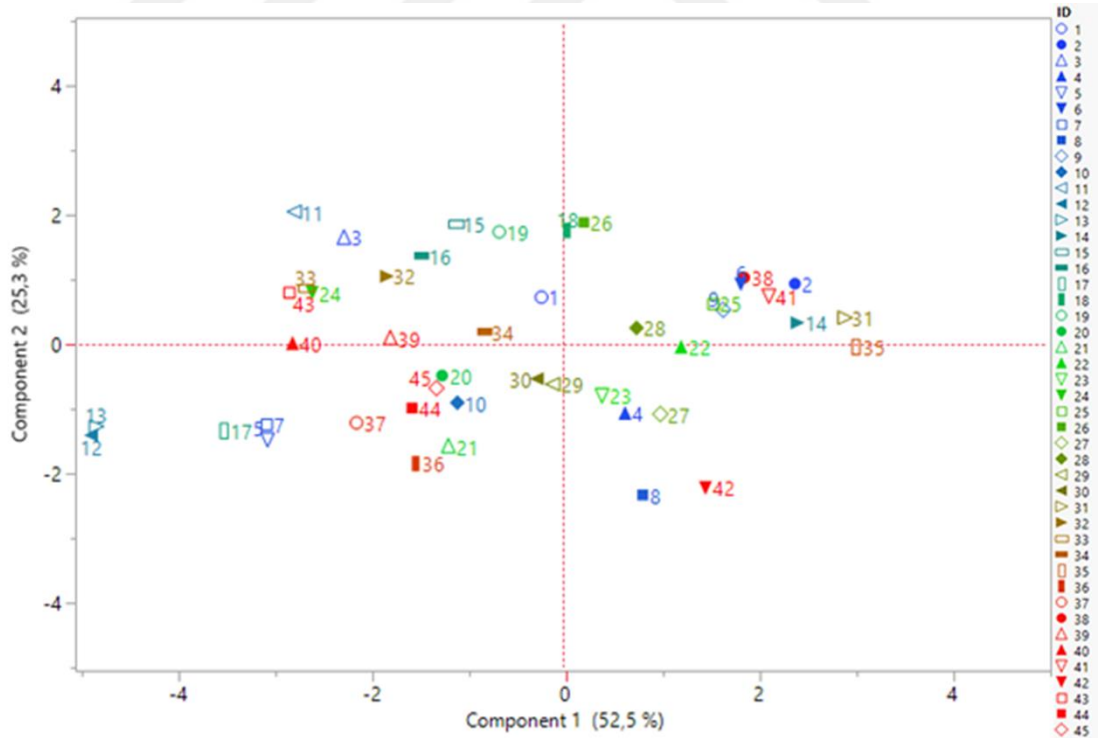
Şekil 6.3. PCA grafiği.



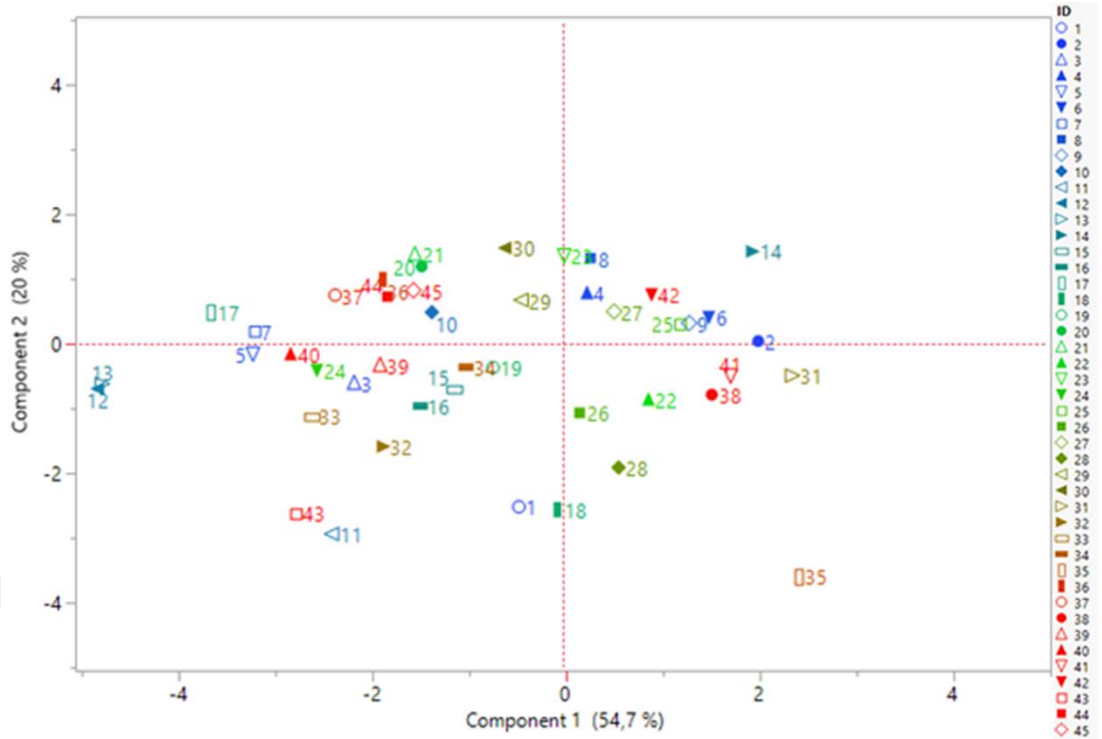
Şekil 6.4. İşaretlenmiş PCA grafiği.



Şekil 6.5. Tekrarsız veri PCA grafiği.



Şekil 6.6. PCA (Xc Std Dev Wighted) grafiği.



Şekil 6.7. PCA (Yc Std Dev Wighted) grafiği.

Grafiklerin temel parçacıklarının anlamlı olan örnekleri seçilerek bunların gidiş yollarının örüntüleri paylaşılmıştır.

6.8.2. PLS

1966 yılında Wold tarafından bulunmuş regresyon modelidir. Kimya, ekonomi ve tıp gibi birçok disiplinde yoğun olarak kullanılmaktadır [53]. PLS çok değişkenli kalibrasyon yöntemi olarak da tanımlanmaktadır [54]. K lineer skorları üreten matris; $P \in \mathcal{R}^{p \times K}$ ve $Q \in \mathcal{R}^{q \times K}$ katsayı matrisleri; $E \in \mathcal{R}^{n \times p}$ ve $F \in \mathcal{R}^{n \times q}$ rastgele hataların matrisi; $T \in \mathcal{R}^{n \times K}$ olduğunda; PLS regresyonunun temeli ($Y \in \mathcal{R}^{n \times q}$) cevap matrisinin ve ($X \in \mathcal{R}^{n \times p}$): $Y = TQ^T + F$ ve $X = TP^T + E$ tahmin matrisinin temel örtülü ayrışmasının varsayımı altında işlem yapan boyut azaltımı tekniğidir [53].

$T = XW$ gibi gizli bileşen matrisinin belirlenmesi için PLS'in ardışık optimizasyon problemlerinden $W = (w_1, w_2, \dots, w_K)$ sütunlarını bulması gerekmektedir. PLS sadece $X - Y$ arasında ilintili değil hem de X uzayında en çok değişen yönleri yakalayan vektörleri arar [53].

W_K yön matrisi iken, yön vektörlerinin K adedini kullanarak gizli bileşenler ($T_K = X\widehat{W}_K$) tahmin ettikten sonra, Q yükleri $\min_Q(\|Y - T_{KQ}^T\|_2)$ 'yi çözerek tahmin edilmiştir. $\widehat{Q}$ 'nın en küçük kareler probleminin çözümü olduğunda, bu nihai tahminci $\widehat{\beta}^{PLS} = \widehat{W}_K\widehat{Q}^T$ 'yi tetikler [53].

PLS algoritması da öğrenme veri seti üzerinde JMP yazılımı kullanılarak çalıştırılmıştır. Elde edilen şekiller paylaşılmıştır. Şekil 6.8 PLS grafiğini Şekil 6.9 ise tekrarsız verilerin PLS grafiğini ifade etmektedir.

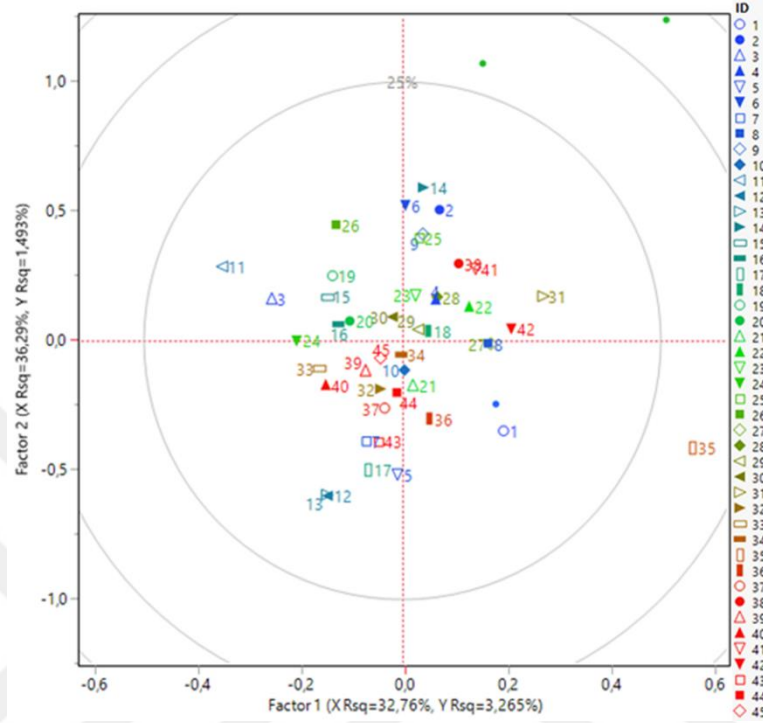
6.8.3. K-means

MacQueen güdümsüz öğrenme algoritması K-means'i verilen veri setini kümelemek için önermiştir [55]. Bir veri seti, algoritma tarafından (keyfi) k sayıda kümeye basit ve kolayca sınıflandırılır. Algoritma her bir küme için kitle merkezini tanımlar. Kitle merkezleri birbirlerine mümkün olan en uzak mesafeye konumlandırılır. Sonra, algoritma veri setindeki her bir noktayı işler, her nokta için en yakın kitle merkezini hesaplar ve her noktayı en yakın kitle merkezi ile ilişkilendirir. Bütün noktaları yerleştirdikten sonra, algoritma önceki kümelerin ağırlıklarının merkezini hesaplar, yeni merkezleri yeniden belirler ve noktaları en yakın merkeze yeniden bağlar. Algoritmanın yeterli iterasyonu sonrası, merkezler uygun yerlerini bulur. n adet veri noktasından ilgili küme merkezlerine uzaklığın göstergesi olarak $x_i^{(j)}$ veri noktası ile c_j küme merkezi arasında $\|x_i^{(j)} - c_j\|^2$ seçilen uzaklık ölçümü olduğu zaman, Denklem 6.22'de sunulan kare hata fonksiyonu hedef fonksiyonu minimize etmeye çalışır [56]. Formül $\|x - y\|$, x ve y vektörleri arasındaki Öklid mesafesi anlamına gelmektedir [57].

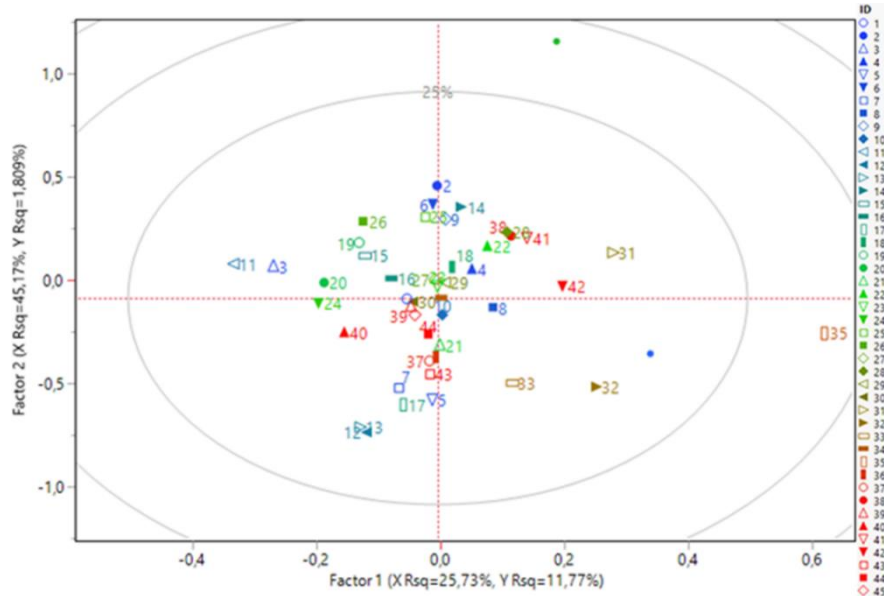
$$J = \sum_{j=1}^k \sum_{i=1}^n \|x_i^{(j)} - c_j\|^2 \quad (6.22)$$

Bu denkleme göre ve keyfi k parametresinin farklı sayılar ile denenmesi sonucu oluşan kümelere örnek olarak Şekil 6.10 ($k = 13$) olan kümenin görselini sunmaktadır. Şekil

algoritmanın JMP yazılımında öğrenme veri seti üzerinde çalıştırılması sonucu elde edilmiştir.



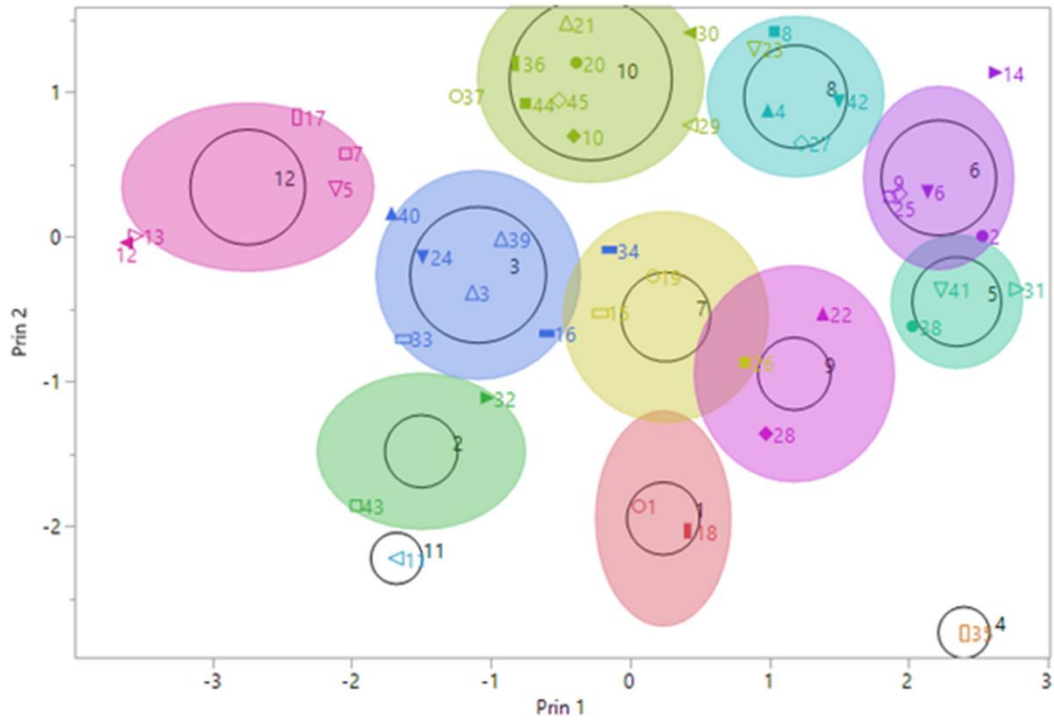
Şekil 6.8. PLS grafiği



Şekil 6.9. PLS Exclude grafiği

K-means algoritması {1, 18} gidiş yollarını 1., {32, 43} gidiş yollarını 2., {3, 16, 24, 33, 34, 39, 40} gidiş yollarını 3., {35} gidiş yolunu 4., {31, 38, 41} gidiş yollarını 5., {2, 6, 9, 25} gidiş yollarını 6., {16, 19, 26} gidiş yollarını 7., {4, 8, 27, 42} gidiş yollarını 8., {22, 28} gidiş yollarını 9., {10, 20, 21, 23, 29, 30, 36, 37, 44, 45} gidiş yollarını 10., {11} gidiş yolunu 11., {5, 7, 12, 13, 17} gidiş yollarını 12. kümeye sınıflandırmıştır. Bu sonuçlar bölümün devamında tezde önerilen yöntemin sonuçları ile kıyaslama için kullanılacaktır.

Örneğin 1. küme elemanları ele alındığında 1 ve 18 numaralı gidiş yollarını PLS birbirine çok yakın bulmamıştır fakat PCA yakın konumlandırmıştır. Başka bir örnek olarak 12. küme elemanlarını 3 algoritma da yakın ve aynı küme içerisinde bulmuştur.



Şekil 6.10. K-means grafiği.

6.9. PARAMETRE AĞIRLIKLARI İÇİN YAKLAŞIMLAR

Bölüm 6.3’de değinilen uzaklık ölçüm yöntemleri öğrenme, sentetik ve test veri setlerinde denenmiş ve çıkarımlar yapılmıştır. Sonraki kısımda sonuçlara değinilecektir.

Ağırlıklı Öklid yönteminde farklı parametrelere farklı ağırlıklar uygulanarak parametrelerin etkileri saptanmaya çalışılmıştır. Temel beş parametre için ağırlıklar Çizelge 6.1’de gösterilmiştir.

Çizelge 6.1. Basit parametrelerde uygulanan Öklid ağırlıkları.

Yaklaşım	Ağırlıklar				
	o	h	w	xc	yc
LWE	10	35	35	10	10
E	100	100	100	100	100
MWE	10	15	15	30	30
HWE	2	4	4	45	45
SWE	30	5	5	30	30

Ağırlık olarak xc ve yc referans noktası kabul edilmiştir. LWE parametre ağırlığını h ve w tarafına verdiği için düşük ağırlıklı olarak tanımlanmış yaklaşımdır. E her parametrenin eşit ağırlığa sahip olduğu yaklaşımdır. Bir başka deyişle normal Öklid mesafesi ölçümüdür. MWE xc ve yc ağırlıklarının LWE yaklaşımına göre biraz arttırılması, h ve w ’nun ağırlıklarının ise biraz azaltılması ile oluşturulmuş yaklaşımdır. HWE xc ve yc ağırlıklarının MWE yaklaşımına göre biraz arttırılması ve diğerlerinin ağırlıklarının ise azaltılması ile oluşturulmuş yaklaşımdır. SWE yaklaşımı MWE ile xc ve yc ağırlıkları aynıdır. Yön de xc ve yc ağırlıklarına eşittir fakat h ve w ’nun ağırlıkları azaltılmıştır.

Ağırlıkların dağılımından anlaşılacağı üzere parametrelerin sonuçlara etkisini gözlemlemek için farklı seçimler yapılmıştır. Parametre sayısı arttırıldığında uygulanan ağırlıklar da Çizelge 6.2’de verilmiştir. Ağırlıkların sonuçlara etkisi aşağıda paylaşılmıştır.

6.10. SONUÇLAR

Bu bölümde tezde özgün olarak önerilen ikinci yöntemin (Bölüm 6.2) uygulaması ile elde edilen sonuçlar paylaşılacaktır. Öncelikle, yöntem Bölüm 4.1.1 ve 4.1.2’de detaylarından bahsedilen öğrenme ve sentetik veri setleri üzerinde çalıştırılmıştır. İlk olarak yöntem ve algoritmanın başarımı açısından sentetik verinin dahil olduğu veri

üzerinde elde edilen sonuçlara yer verilmiştir. Önerilen yöntemin doğruluğu ve kullanılabilirliği kanıtlandıktan sonra Bölüm 6.9’da bahsedilen farklı yaklaşımların sonuçları tartışılmış ve en uygun olan seçilmiştir. Seçilen yaklaşım için parametre artırımı yapılmıştır. Farklı parametre ve ağırlıklandırma ile gidiş yollarının geometrik, dinamik yapıları ile karakteristikleri elde edilip kıyaslanmıştır. Sonrasında rastgele benzeşmeleri önleyecek ilave yaklaşımlar test edilmiştir. Nihayetinde de benzer gidiş yollarının kümelenmesi ile elde edilen sonuçlar sunulmuştur.

Çizelge 6.2. İstatistiksel parametrelerin yer aldığı Öklid yönteminde ağırlıklar.

Yaklaşım	Ağırlıklar									
	o	h	w	xc	yc	xc-yc StdDev	o min- max	h min- max	w min- max	xc-yc min- max
LWE	5	15	15	5	5	5	2,5	7,5	7,5	2,5
E	100	100	100	1000	100	100	100	100	100	100
MWE	4,17	4,17	4,17	12,5	12,5	12,5	2,08	2,083	2,083	6,25
HWE	1	2	2	15	15	15	0,5	1	1	7,5
SWE	11,25	2,5	2,5	11,25	11,25	11,25	5,63	1,25	1,25	5,63

Veri setinde toplam 45 adet nesnenin gidiş yolu bulunmaktadır. Her birinin diğerleriyle olan uzaklıkları Bölüm 6.2 ve 6.3’de bahsedilen yöntemler ile bulunmuştur. Bu hesaplamaların sayısını azaltmak ve işlem tekrarından kaçınmak için, her kıyaslama bir kere yapılmak kaydıyla, $(n(n-1)/2)$ 990 karşılaştırma yapılmıştır. 3 adet sentetik veri de ilave edilince nesne sayısı 48’e çıkmaktadır. Karşılaştırma sayısı da 1128 adet olmaktadır.

6.10.1. Doğruluk Sonuçları

Önerilen yöntemin bulduğu gidiş yollarının benzerliğinin doğruluğunu test etmek gerekmektedir. Bu, yöntemin doğruluk (accuracy) sonucunu verecektir. Önerilen yöntemin doğruluk değerlerini test etme işlemi için sentetik veri devreye girmektedir. Sentetik veri ile orijinal veri arasındaki benzerliğin- çok yakın olması (uzaklığın en az olması) beklenmektedir.

Çizelge 6.3, gidiş yolları arasında Denklem 6.8 ile hesaplanan benzerliği (uzaklıkları) ve Denklem 6.18 benzerlik yüzdesini göstermektedir. Çizelgeye 1128 sonucun hepsini

koymak mümkün olamayacağından, Bölüm 6.6'da verilen hesaplama yöntemi kullanıldığında, ilk 30 sonuç gösterilmiştir.

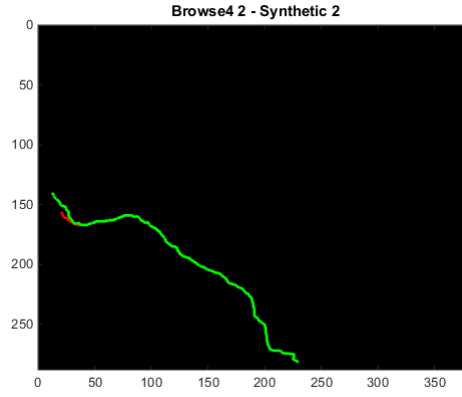
Sıralamada ilk sentetik verinin orijinal hali ile benzerliği 2. sırada kendisine yer bulmuştur. Şekil 6.11'de (16, 33) Sentetik2 ve orijinal gidiş yollarının birlikte örüntüsü (Sentetik2-doğrulaması) sunulmuştur. İlaveten x eksenini 384 değerine kadar, y eksenini ise 288 değerine kadar ölçeklendirilmiştir (384 x 288 piksel görüntü).

Kendisine 4. sırada yer bulabilen (31, 44) ikinci sentetik veri ve orijinali ile benzerliği (Sentetik0-doğrulaması) Şekil 6.12'de verilmiştir.

Üçüncü sentetik veri ve doğrulaması ilk 30 benzerlik arasına girememiştir. Benzerliği %91,16'da kalmıştır. Benzerlik sıralamasında olmaması kullanılan parametre sayısının artırılması gerekliliğine işaret ettiği sonucuna varılmıştır.

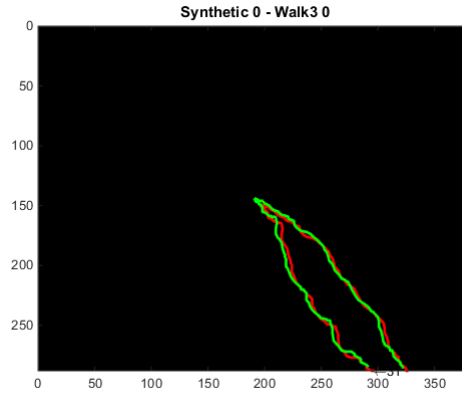
Çizelge 6.3. Öklid, eşit ağırlıklı temel 5 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)	ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)
1	9	25	0,2224	99,30	16	1	44	0,9710	96,93
2	16	33	0,2678	99,15	17	15	16	0,9824	96,89
3	12	13	0,3241	98,98	18	10	47	1,0023	96,83
4	31	44	0,6070	98,08	19	31	41	1,0058	96,82
5	6	25	0,6674	97,89	20	39	40	1,0291	96,75
6	6	9	0,6695	97,88	21	21	47	1,0301	96,74
7	5	7	0,6762	97,86	22	4	27	1,0522	96,67
8	1	31	0,6851	97,83	23	24	36	1,0705	96,61
9	41	44	0,6956	97,80	24	21	39	1,1275	96,43
10	15	19	0,7012	97,78	25	15	33	1,1294	96,43
11	20	48	0,7350	97,68	26	1	41	1,1317	96,42
12	47	48	0,7568	97,61	27	40	47	1,1369	96,40
13	10	48	0,8278	97,38	28	1	9	1,2199	96,14
14	23	29	0,9320	97,05	29	7	40	1,2299	96,11
15	1	6	0,9524	96,99	30	23	30	1,2375	96,09



Şekil 6.11. Sentetik2-doğrulaması benzerlik grafiği.

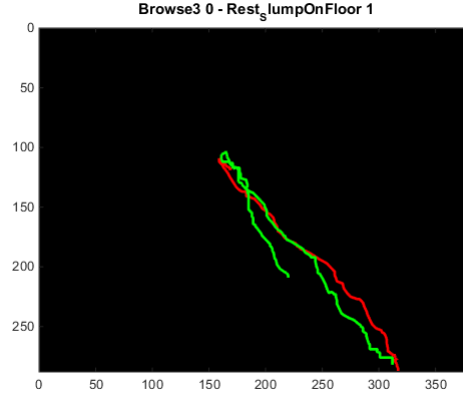
Üst sıralarda yer alan diğer gidiş yollarına bakmak gerekirse, örneğin en çok benzer çıkan (9, 25) yolları Şekil 6.13’de verilmiştir. Farklı açılardan değerlendirildiğinde bir benzerliğin olduğu yadsınamaz ancak gidiş yollarının yapısı gereği bu iki yolun en benzer olması beklenmez. Üçüncü sırada yer alan (12, 13) gidiş yolları Şekil 6.14’de sunulmuştur. 12 ve 13 gidiş yolları bazı açılardan en az Sentetik 0 ve orijinal verisinin birbirine benzediği kadar benzerdir.



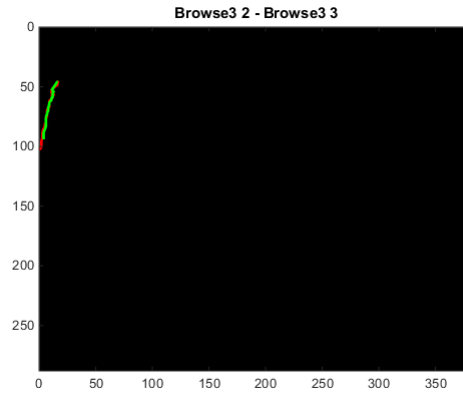
Şekil 6.12. Sentetik0-doğrulaması benzerlik grafiği.

Çizelge 6.3’deki ilk 10 adet uzaklık için şekiller (Şekil 6.11 - 6.17) sunulmuştur. İlaveten {(9, 25), (12, 13), (6, 25), (6, 9), (5, 7), (38, 41), (15, 19)} gidiş yollarının (sentetik veri (31, 32, 33. Gidiş yolları) hariç, sonradan eklenmiş 3 adet sentetik gidiş yolu olmadığından (41, 44) orijinal veri setinde (38, 41) numaralı gidiş yoludur.) benzerlikleri önceki bölümde PCA, PLS ve K-means yöntemlerinin sonuçları ile

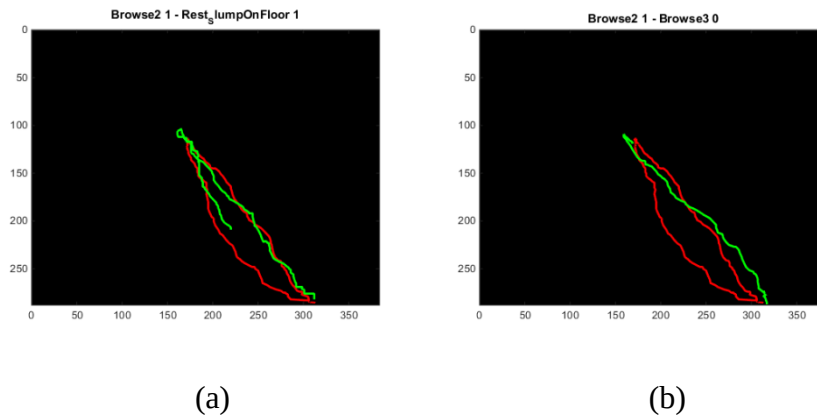
kıyaslandığında ya birbirlerine uzaklıkları çok yakın ya da aynı kümenin içinde yer aldıkları görülmektedir.



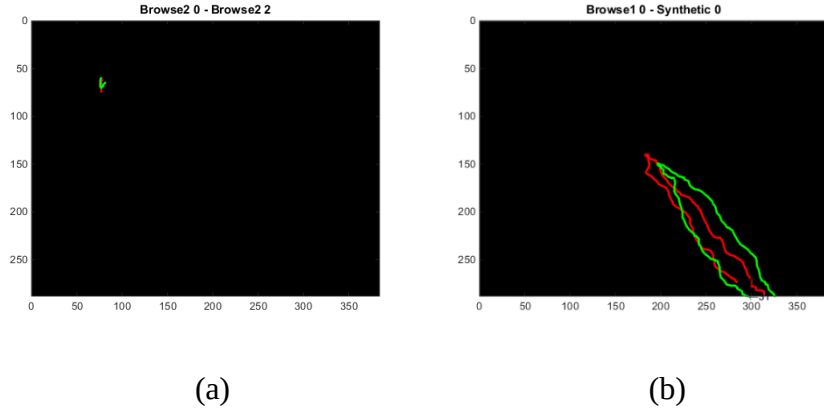
Şekil 6.13. Browse3 0 – RestSlumpOnFloor 1 (9, 25) nesnesi benzerlik grafiği.



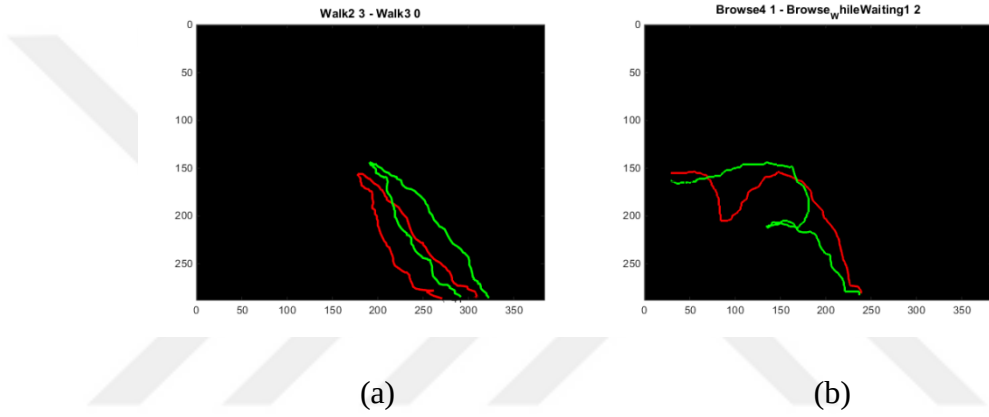
Şekil 6.14. Browse3 2-Browse3 3 (12, 13) nesnesi benzerlik grafiği.



Şekil 6.15. Gidiş yolları a) (6, 25) ve b) (6, 9) benzerlik grafiği.



Şekil 6.16. Gidiş yolları a) (5, 7) ve b) (1, 31) benzerlik grafiği.



Şekil 6.17. Gidiş yolları a) (41, 44) ve b) (15, 19) benzerlik grafiği.

Gidiş yollarının görsel olarak da geometrik yapılarının benzerliğini değerlendirmek mümkündür. Yarım olan gidiş yollarının benzerliğinin yüksek çıkması beklenmedik bir durumdur. Bu noktada ilk 10 adet benzerliğin 7'si isabetlidir. Başka bir deyişle minimum başarımın %70 olduğu değerlendirilmektedir.

Bu çıkarım ve değerlendirme ile önerilen yöntemin geliştirilebileceği, gidiş yollarının bu yöntem ile ifade edilebileceği ve benzerliklerini ölçmede kullanılabileceği ortaya koyulmuştur. Kullanılan Öklid uzaklığı ağırlıklandırılarak parametrelerin farklı etkileri üzerinde çıkarımlar yapılmıştır.

6.10.2. Öklid Ağırlıklarının Doğruluk Sonuçlarına Etkisi

Önceki bölümde sonuçları verilen hesaplamalar parametrelerin eşit ağırlıklara sahip olduğu Öklid uzaklıklarıyken, bu bölümde parametrelerin ağırlıklarının değiştirilmesinin doğruluğa etkisi araştırılmıştır. Sırasıyla LW, MW, HW ve SW sonuçları sunulmuştur. Çizelge 6.4 LW Öklid uzaklıklarını göstermiştir. Zaten eşit ağırlıklı Öklid sonuçları Çizelge 6.3’de sunulduğundan burada tekrarlanmamıştır. Çizelge 6.5 MW Öklid uzaklıklarını, Çizelge 6.6 HW Öklid uzaklıklarını ve Çizelge 6.7 ise SW Öklid uzaklıklarını sunmuştur.

Şekil 6.11’de Sentetik2-doğrulaması; LW, E, MW, HW ve SW yaklaşımları tarafından sırasıyla 2, 2, 1, 1, 3. sırada birbirine yakın bulunmuştur. Şekil 6.12’de sunulan Sentetik0-doğrulaması ise sırasıyla 14, 4, 6, 6, 4. sırada bulunmuştur. İlk gidiş yolunu bulmada MW ve HW, ikincisini bulmada da E ve SW yöntemleri daha başarılıdır.

Çizelge 6.4. LW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)	ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)
1	16	33	0,0876	99,72	16	39	40	0,4027	98,73
2	9	25	0,0986	99,69	17	2	16	0,4067	98,71
3	12	13	0,1792	99,43	18	31	34	0,4167	98,68
4	41	44	0,2446	99,23	19	24	36	0,4189	98,68
5	1	31	0,2762	99,13	20	4	27	0,4292	98,64
6	5	7	0,2875	99,09	21	40	47	0,4381	98,61
7	20	48	0,2916	99,08	22	21	47	0,4502	98,58
8	1	6	0,3027	99,04	23	1	9	0,4686	98,52
9	6	25	0,3033	99,04	24	2	33	0,4747	98,50
10	15	19	0,3083	99,02	25	1	44	0,4756	98,50
11	6	9	0,3172	99,00	26	25	44	0,4788	98,49
12	10	48	0,3284	98,96	27	21	39	0,4806	98,48
13	47	48	0,3312	98,95	28	6	31	0,4823	98,47
14	31	44	0,3425	98,92	29	9	44	0,4831	98,47
15	10	47	0,3445	98,91	30	7	17	0,4867	98,46

Çizelge 6.5. MW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	16	33	0,0887	99,72	16	15	16	0,3854	98,78
2	9	25	0,1052	99,67	17	23	29	0,3993	98,74
3	12	13	0,1329	99,58	18	15	33	0,4317	98,63
4	5	7	0,2312	99,27	19	39	40	0,4381	98,61
5	1	31	0,2416	99,24	20	1	6	0,4391	98,61
6	31	44	0,2495	99,21	21	2	16	0,4415	98,60
7	20	48	0,2742	99,13	22	31	41	0,4575	98,55
8	6	9	0,2965	99,06	23	1	41	0,4646	98,53
9	6	25	0,3131	99,01	24	10	20	0,4663	98,53
10	15	19	0,3464	98,90	25	34	44	0,4721	98,51
11	41	44	0,3666	98,84	26	31	34	0,4882	98,46
12	47	48	0,3678	98,84	27	24	36	0,4888	98,45
13	1	44	0,3727	98,82	28	2	33	0,4999	98,42
14	4	27	0,3800	98,80	29	35	36	0,5053	98,40
15	10	48	0,3813	98,79	30	21	47	0,5063	98,40

Çizelge 6.6. HW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	16	33	0,0524	99,83	16	6	25	0,3276	98,96
2	12	13	0,0998	99,68	17	2	16	0,3283	98,96
3	9	25	0,1117	99,65	18	23	29	0,3422	98,92
4	5	7	0,1132	99,64	19	6	34	0,3526	98,88
5	1	31	0,1538	99,51	20	1	41	0,3710	98,83
6	31	44	0,1839	99,42	21	22	44	0,3721	98,82
7	20	48	0,2132	99,33	22	15	19	0,3820	98,79
8	15	16	0,2442	99,23	23	6	44	0,3893	98,77
9	1	44	0,2619	99,17	24	32	37	0,3953	98,75
10	4	27	0,2623	99,17	25	35	36	0,3969	98,75
11	15	33	0,2720	99,14	26	47	48	0,4010	98,73
12	6	9	0,2864	99,09	27	10	48	0,4065	98,71
13	10	20	0,3104	99,02	28	31	34	0,4199	98,67
14	34	44	0,3132	99,01	29	9	34	0,4291	98,64
15	2	33	0,3234	98,98	30	39	40	0,4318	98,63

Şekil 6.13’da gösterilen gidiş yolları (9, 25) sırasıyla 2, 1, 2, 3, 2. sırada bulunmuştur. Gidiş yollarının benzerliğinin nasıl tanımlanacağı üzerinde tartışılacaktır. Şekilsel benzerliğin haricinde ilave parametrelerin dikkate alınması hassasiyet ve doğruluğu arttıracaktır. Örneğin (9, 25) geometrik açıdan incelendiğinde giriş-çıkış noktaları ve

karakteristiklerinin benzeştiği gözlenmektedir. Öncelikle E, sonrasında LW, MW ve SW Öklid sıralamayı en iyi bulan yöntemler olarak ortaya çıkmaktadır.

Çizelge 6.7. SW Öklid, temel 5 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	12	13	0,0995	99,69	16	31	41	0,3912	98,76
2	9	25	0,1004	99,68	17	15	33	0,3998	98,74
3	16	33	0,1449	99,54	18	22	44	0,4012	98,73
4	31	44	0,1734	99,45	19	1	41	0,4064	98,71
5	6	9	0,2806	99,11	20	10	48	0,4077	98,71
6	15	16	0,2913	99,08	21	7	40	0,4081	98,71
7	6	25	0,2935	99,07	22	14	25	0,4580	98,55
8	23	29	0,3160	99,00	23	21	47	0,4710	98,51
9	5	7	0,3166	99,00	24	9	14	0,4829	98,47
10	15	19	0,3188	98,99	25	5	40	0,4892	98,45
11	1	31	0,3338	98,94	26	4	27	0,5085	98,39
12	47	48	0,3456	98,91	27	39	40	0,5113	98,38
13	20	48	0,3620	98,86	28	1	6	0,5208	98,35
14	41	44	0,3657	98,84	29	21	39	0,5268	98,33
15	1	44	0,3886	98,77	30	24	36	0,5319	98,32

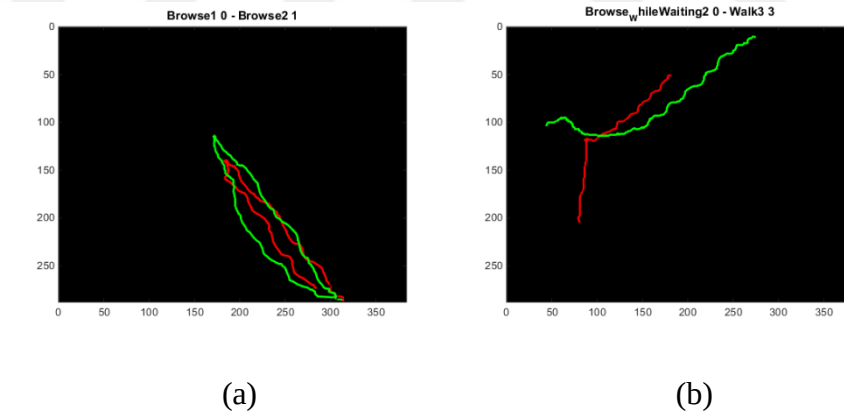
Yine üst sıralarda yer alan Şekil 6.14’de sunulan (12, 13) gidiş yolları, değerlendirilen yaklaşımlar tarafından sırasıyla 3, 3, 3, 2, 1. sırada bulunmuştur. Geometrik olarak sentetik veri kadar birbirlerine benzedikleri çıplak gözle söylenebilmektedir. Burada SW ve HW yaklaşımları bir adım öndedir.

İlk on sırada birkaç sıra farkla aynı gidiş yolları farklı yöntemlerde yer almıştır. Arada çok ufak nüanslar bulunmakla beraber, bu farklılıkların tanımlanması zordur. Tersten bakacak olursak örneğin LWE ilk on sırada diğer yaklaşımların bazılarından (Eşit ağırlıklı Öklid öncelikli denendiğinden bu noktadan sonra karşılaştırmalarda referans olarak alınmıştır.) farklı olarak (20, 48) ve (1, 6) gidiş yollarını bulmuştur. Şekil 6. 18 bu gidiş yollarını grafiksel olarak göstermektedir. Gidiş yolları (1, 6) ortalama geometrik şekilleri, giriş-çıkış noktaları ve karakteristikleri düşünüldüğünde birbirine benzerdir. Gidiş yolları (20, 48) ise geometrik şekil açısından kısmen benzer, giriş-çıkış noktaları birbirine benzemezken, karakteristik olarak da birbirine çok az benzemektedir. Böylece LWE’nin Şekil 6.12 Sentetik0-doğrulaması ve Şekil 6.15

gidiş yolları (6, 9) yerine üst sıralara taşıdığı gidiş yollarından birisi doğru iken diğeri başarısızdır.

MW diğer yaklaşımlardan bazılarında farklı olarak sadece (20, 48)'i ilk 10 sırada bulmuştur. Şekil 6.18'den anlaşılacağı üzere, genelleştirmek gerekirse sadece xc ve yc ağırlıklı ortalamanın bilgi çıkarmak için çok yeterli olamayabileceği değerlendirilmektedir. Eşit ağırlıklı Öklid'e göre kıyaslandığında ise Şekil 6.17 gidiş yolu (41, 44) yerine üst sıralara taşıdığı (20, 48) ile daha başarısızdır.

HW yaklaşımı ise (20, 48)'i farklı olarak bulmasının haricinde ilaveten (15, 16), (1, 44) ve (4, 27) gidiş yollarını da ilk ona taşımıştır. (15, 16) gidiş yollarının geometrik şekilleri ve giriş-çıkış noktaları kısmen benzerken karakteristiği farklıdır. (1, 44) gidiş yollarında geometrik şekil, giriş-çıkış noktaları ve karakteristik benzeşmektedir. (4, 27)'de ise geometrik şekil tam benzemezken, giriş-çıkış noktaları tek taraflı benzer ve karakteristik kısmen benzerdir.

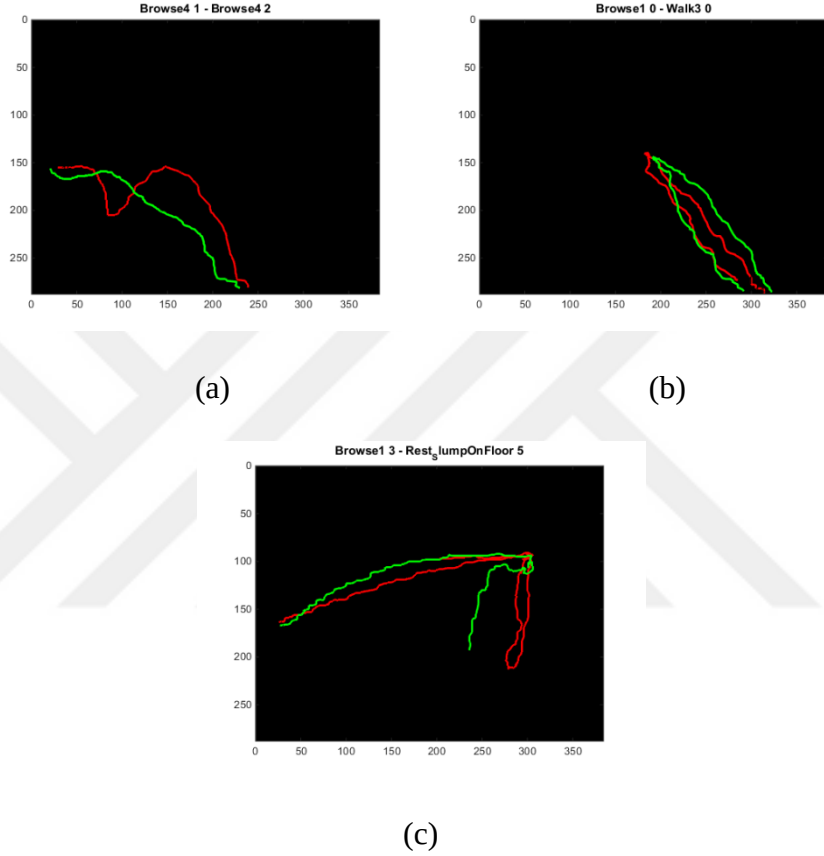


Şekil 6.18. a) (1, 6) ve b) (20, 48) gidiş yolları benzerlik grafiği.

SW yaklaşımı ise Şekil 6.19 gidiş yolları (15, 16) ve Şekil 6.20 (23, 29) gidiş yollarını da ilk on sıraya taşımıştır. (15, 16) çok başarılı değilken, (23, 29) ise geometrik şekil daha az benzeşirken, giriş - çıkış noktaları tek taraflı benzer ve karakteristik kısmen benzer olarak daha az başarılı bir benzeşmedir.

Farklı ağırlıklara sahip her yöntemin ilk 10 benzeşimi dikkate alındığında yaklaşımlar arasında belirgin bir fark ortaya çıkmamıştır. Mevcut değerlendirmeler ışığında bu

noktada eşit ağırlıklı Öklid yaklaşımının skorları daha iyidir. Yine de değerlendirmeler genişletilmek istendiğinden Çizelge 6.8 hazırlanmıştır. Eşit ağırlıklı Öklid referans alındığında ilk 15 benzerliğin diğer yaklaşımlar tarafından hangi sırada yer aldığı çizelgede gösterilmiştir.



Şekil 6.19. a) (15, 16), b) (1, 44) ve c) (4, 27) gidiş yolları benzerlik grafiği.

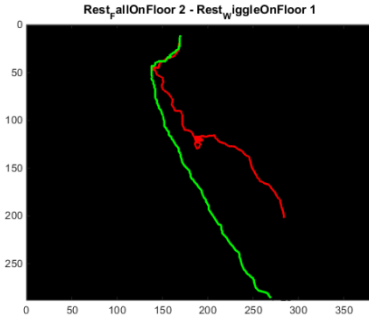
LW farklı sıralarda olmakla beraber Şekil 6.20’de verilen (23, 29) gidiş yolları hariç eşit ağırlıklı Öklid sonuçların hepsini ilk 15’de bulmuştur. (23, 29) yerine ise Şekil 6.20’deki (10, 47) gidiş yollarını bulmuştur. (10, 47) giriş – çıkış noktası ve gidiş yolunun şekline göre (23, 29)’dan daha başarısız bir benzerliktir.

MW sadece gidiş yolları (23, 29) (Şekil 6.20) ve gidiş yolları (1,6) (Şekil 6.18) hariç diğer gidiş yollarını farklı sıralarda olmakla beraber eşit ağırlıklı Öklid sonuçlarına benzer bulmuştur. Bunların yerine de Şekil 6.19 gidiş yolları (1, 44) ve (4, 27) daha

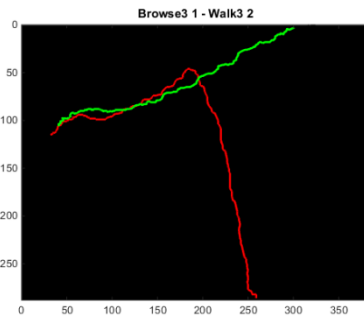
benzer bulmuştur. (1,6) ve (1, 44) gidiş yollarının her ikisi de kendi içinde benzeşmektedir. (23, 29) ile (4, 27) arasında sıralama yapmak zordur.

Çizelge 6.8. Eşit ağırlıklı Öklid'e referansla değişen gidiş yolları sıralamaları.

		E	LW	MW	HW	SW
9	25	1	2	2	3	2
16	33	2	1	1	1	3
12	13	3	3	3	2	1
31	44	4	14	6	6	4
6	25	5	9	9	16	7
6	9	6	11	8	12	5
5	7	7	6	4	4	9
1	31	8	5	5	5	11
41	44	9	4	11	31	14
15	19	10	10	10	22	10
20	48	11	7	7	7	13
47	48	12	13	12	26	12
10	48	13	12	15	27	20
23	29	14	33	17	18	8
1	6	15	8	20	39	28



(a)



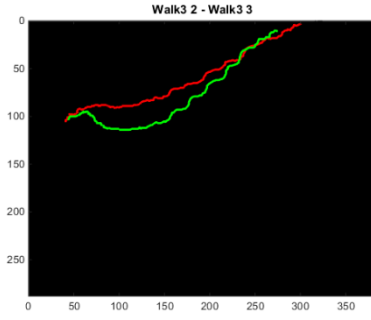
(b)

Şekil 6.20. a) (23, 29) ve b) (10, 47) gidiş yolları benzerlik grafiği.

SW sadece Şekil 6.21 gidiş yolları (10, 48) ve Şekil 6.18 gidiş yolları (1, 6) hariç diğer gidiş yollarını farklı sıralarda olmakla beraber eşit ağırlıklı Öklid sonuçlarına benzer bulmuştur. Şekil 6.19 gidiş yolları (15, 16) çok başarılı değilken (1, 44) gidiş yolu (10, 48) gidiş yoluna göre daha başarılı bir benzeşmedir.

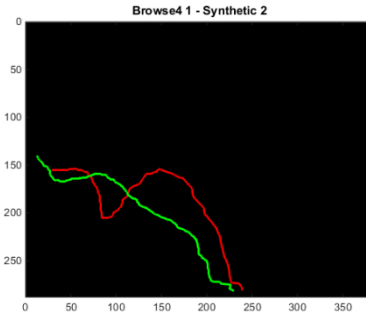


(a)

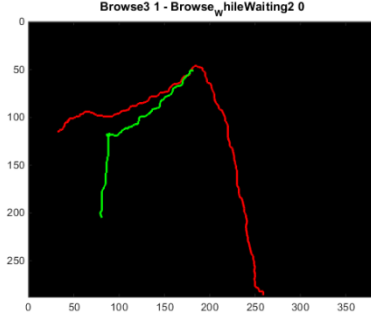


(b)

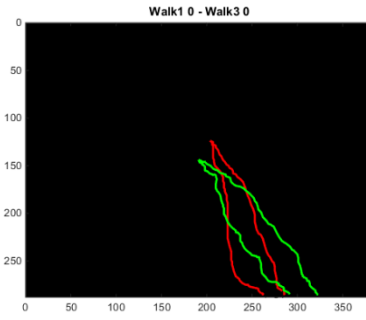
Şekil 6.21. a) (10, 48) ve b) (47, 48) gidiş yolları benzerlik grafiği.



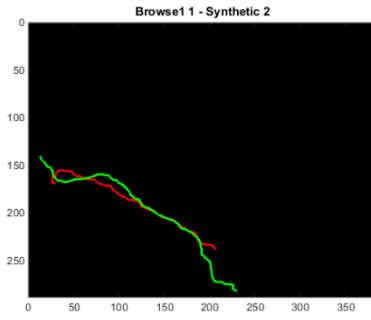
(a)



(b)



(c)



(d)

Şekil 6.22. a) (15, 33), b) (10, 20), c) (34, 44) ve d) (2,33) gidiş yolları benzerliği.

HW ise tam 7 adet gidiş yolunu farklı bulmuştur. (6, 25) (Şekil 6.15), (41, 44) ve (15, 19) (Şekil 6.17), (47, 48) ve (10, 48) (Şekil 6.21), (23, 29) (Şekil 6.20), (1, 6) (Şekil 6.18) gidiş yollarının yerine (15, 16), (1, 44) ve (4, 27) (Şekil 6.19), (15, 33), (10, 20), (34, 44) ve (2, 33) (Şekil 6.22) gidiş yollarını ilk 15 sıraya taşımıştır.

İlk 15 yolun dışında kalan ve ilk 15'e taşınan yollara tek tek odaklanınca bunlar arasında karşılaştırma ve tercih yapmak zorlaşmaktadır. Zira eşit sayıda başarısız, kısmen başarılı ve başarılı benzeşme bulunmaktadır.

Yukarıdaki detaylı kıyaslamaların ışığında genel olarak değerlendirmek gerekirse, LW, MW, HW, SW Öklid ile eşit ağırlıklı Öklid yöntemi karşılaştırıldığında bunlardan birini diğerlerine üstün kılacak bir sonuç çıkmadığından ileri testlerde tüm yaklaşımlar sunulmaya devam edilecektir.

Gidiş yollarının geometrik benzerliğinin yanında karakteristik olarak da benzerliklerini değerlendirecek ilave parametreler eklenerek tartışmalar yapılmıştır.

6.10.3. Giriş-Çıkış Noktaları ve Hızın Doğruluk Sonuçlarına Etkisi

Sentetik veri ve doğrulamalarının alt sıralarda kalması parametre sayısının artırılması gerektiği değerlendirmesine neden olmuştur. İlaveten gidiş yolunun başladığı-bittiği yerin ve hızının benzemesi sezgisel olarak doğruluğu artırır hipotezi kurulmuştur. Gidiş yolu geometrik şekline ilaveten nesnenin davranışının karakteristiği hakkında ilave bilgi edinmeyi sağlayıp sağlamayacağı test edilecektir. Parametre sayısı x_c - y_c başlangıç (giriş-çıkış noktaları) ve bitiş noktaları ile hız eklenince 5'den 10'a çıkarılmıştır. Buna göre yaklaşımların hesaplamaları tekrar yapılmıştır. Çizelge 6.9, 10 parametre ile çalıştırılan LW Öklid benzerliklerini göstermektedir. Burada en kayda değer nokta son sentetik verinin 15. sıradan da olsa benzerliğinin yukarı sıralara çıkmasıdır. Şekil 6.23 orijinali ile olan benzerlik durumunu göstermektedir.

Şekil 6.24 LW için ilk 16 benzeşen gidiş yollarını sırasıyla göstermektedir. LW'nin 5 parametrelili sonuçları ile kıyaslandığında başarımın belirgin bir şekilde arttığı gözlemlenmiştir.

Çizelge 6.10 10 parametre ile çalıştırılan eşit ağırlıklı Öklid benzerliklerini göstermektedir. Burada dikkat çeken nokta (32, 37) sentetik verisi ve orijinal verisinin benzeşimi (Sentetik1-doğrulaması) kendisine 30. sırada yer bulabilmesidir. Şekil 6.25 eşit ağırlıklı Öklid için ilk 16 gidiş yolunun görselini sunmaktadır. Bu durumun

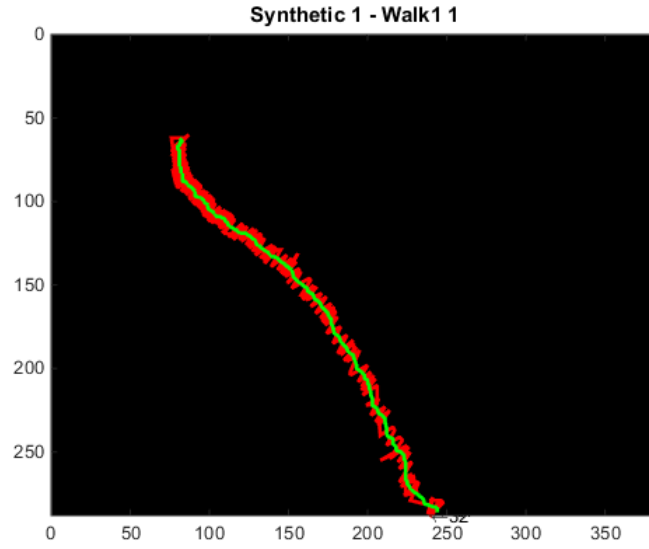
haricinde, ilk sırada bulunan (12, 13) gidiş yolları istisna, diğer sentetik veri doğrulamalarının üst sıralara taşındığı, daha doğru (birbirine daha çok benzeyen) gidiş yollarının üst sıralara taşındığı (Parametre sayısı 10 çıkarılan eşit ağırlıklı yaklaşım gidiş yolları (1, 41) 26. sıradan 9'a, (31, 41) 19. sıradan 10. sıraya, (1, 44) 16. sıradan 7. sıraya taşımıştır.) görülmüştür.

Çizelge 6.9. LW Öklid, 10 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)	ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)
1	12	13	0,3414	98,92	16	31	41	0,9960	96,85
2	31	44	0,4259	98,65	17	7	40	1,0925	96,55
3	16	33	0,6523	97,94	18	6	41	1,1539	96,35
4	1	44	0,6711	97,88	19	5	40	1,1873	96,25
5	6	31	0,6811	97,85	20	1	34	1,2014	96,20
6	1	6	0,7189	97,73	21	19	33	1,2046	96,19
7	41	44	0,7201	97,72	22	9	22	1,2468	96,06
8	6	44	0,7260	97,70	23	34	44	1,3260	95,81
9	5	7	0,7446	97,65	24	9	14	1,3943	95,59
10	1	31	0,7518	97,62	25	31	34	1,4752	95,34
11	1	41	0,8485	97,32	26	6	34	1,5450	95,11
12	35	36	0,8755	97,23	27	21	39	1,6572	94,76
13	34	41	0,9028	97,15	28	39	40	1,7806	94,37
14	16	19	0,9097	97,12	29	10	33	1,8266	94,22
15	32	37	0,9173	97,10	30	15	42	1,8292	94,22

Çizelge 6.10. Eşit ağırlıklı Öklid, 10 parametre ile gidiş yolları benzerlikleri.

ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)	ID	Nesne1	Nesne2	<i>D</i>	Benzerlik (%)
1	12	13	0,4353	98,62	16	5	40	1,7855	94,35
2	31	44	0,6577	97,92	17	34	41	1,8163	94,26
3	16	33	0,6996	97,79	18	34	44	1,8216	94,24
4	5	7	0,9638	96,95	19	19	33	1,8253	94,23
5	41	44	0,9709	96,93	20	6	41	1,8476	94,16
6	1	31	0,9789	96,90	21	9	14	1,9063	93,97
7	1	44	1,0803	96,58	22	31	34	1,9295	93,90
8	1	6	1,1542	96,35	23	21	39	1,9459	93,85
9	1	41	1,2859	95,93	24	39	40	2,0168	93,62
10	31	41	1,3236	95,81	25	9	22	2,1194	93,30
11	6	31	1,5032	95,25	26	1	34	2,1408	93,23
12	7	40	1,5059	95,24	27	2	15	2,3385	92,61
13	6	44	1,5250	95,18	28	35	46	2,4954	92,11
14	35	36	1,5319	95,16	29	6	34	2,7427	91,33
15	16	19	1,5530	95,09	30	32	37	2,8057	91,13

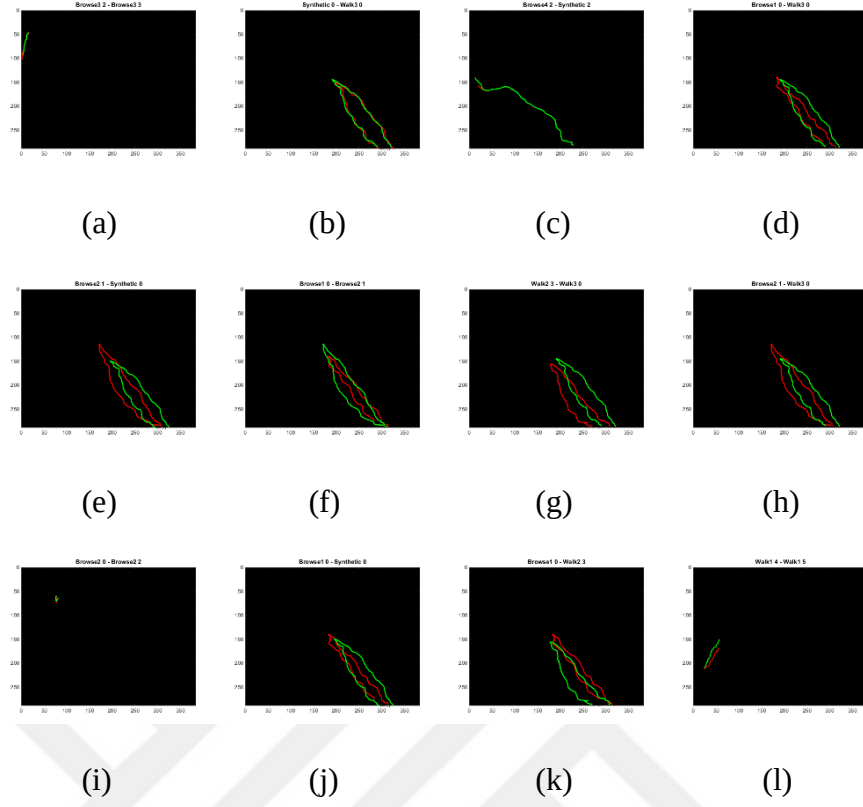


Şekil 6.23 (32, 37) gidiş yolu benzerliği.

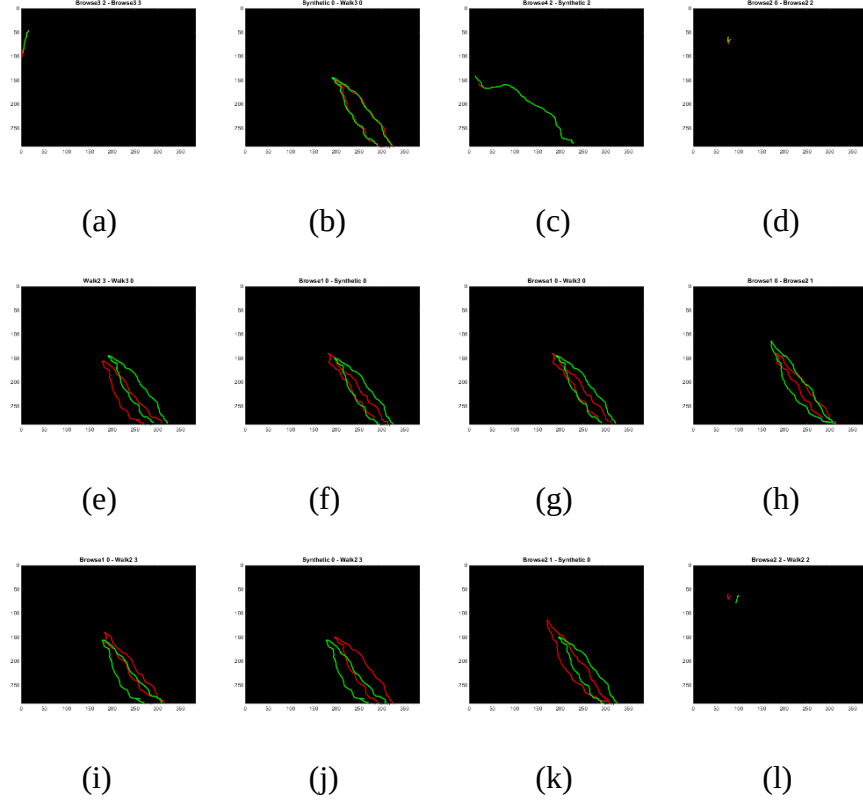
Çizelge 6.11 10 parametre ile çalıştırılan MW Öklid benzerliklerini göstermektedir. Burada dikkat çeken nokta (32, 37) Sentetik1-doğrulamasını ilk 15 benzeşimin içine sokabilmesidir. Şekil 6.26 MW Öklid için gidiş yollarının ilk 16'sının görselini sunmaktadır. MW Öklid eşit ağırlıklı Öklid'den daha iyi, LW Öklid yaklaşımlarından kısmen biraz kötü sonuçlar bulmuştur.

Çizelge 6.11. MW Öklid, 10 parametre ile gidiş yolları benzerlikleri.

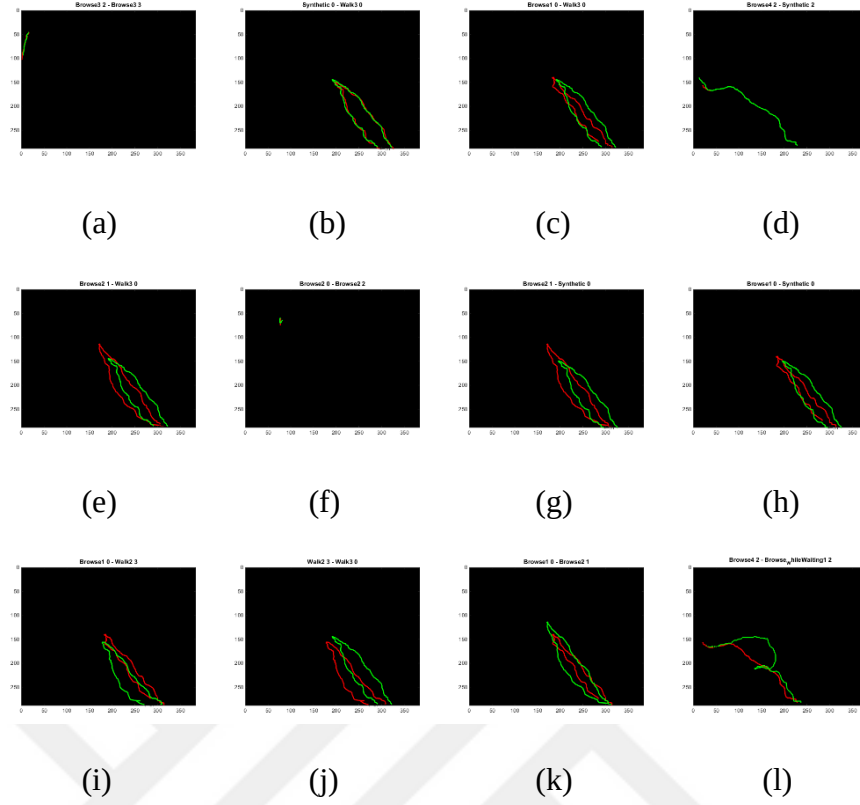
ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	12	13	0,3196	98,99	16	31	41	0,9745	96,92
2	31	44	0,3554	98,88	17	9	22	1,0141	96,79
3	1	44	0,6026	98,09	18	7	40	1,0151	96,79
4	16	33	0,6524	97,94	19	5	40	1,0224	96,77
5	6	44	0,6935	97,81	20	19	33	1,1237	96,45
6	5	7	0,7247	97,71	21	6	41	1,1398	96,40
7	6	31	0,7317	97,69	22	1	34	1,2269	96,12
8	1	31	0,7397	97,66	23	9	14	1,2788	95,96
9	1	41	0,7672	97,57	24	34	44	1,3128	95,85
10	41	44	0,7701	97,56	25	31	34	1,4969	95,27
11	1	6	0,7862	97,51	26	6	34	1,5415	95,13
12	16	19	0,8100	97,44	27	28	44	1,6186	94,88
13	35	36	0,8660	97,26	28	21	39	1,6693	94,72
14	34	41	0,9054	97,14	29	28	31	1,7055	94,61
15	32	37	0,9173	97,10	30	39	40	1,7890	94,34



Şekil 6.24. LW Öklid ilk 16 en benzer gidiş yolu.



Şekil 6.25. Eşit ağırlıklı Öklid ilk 16 en benzer gidiş yolları.



Şekil 6.26. MW Öklid ilk 16 en benzer gidiş yolları.

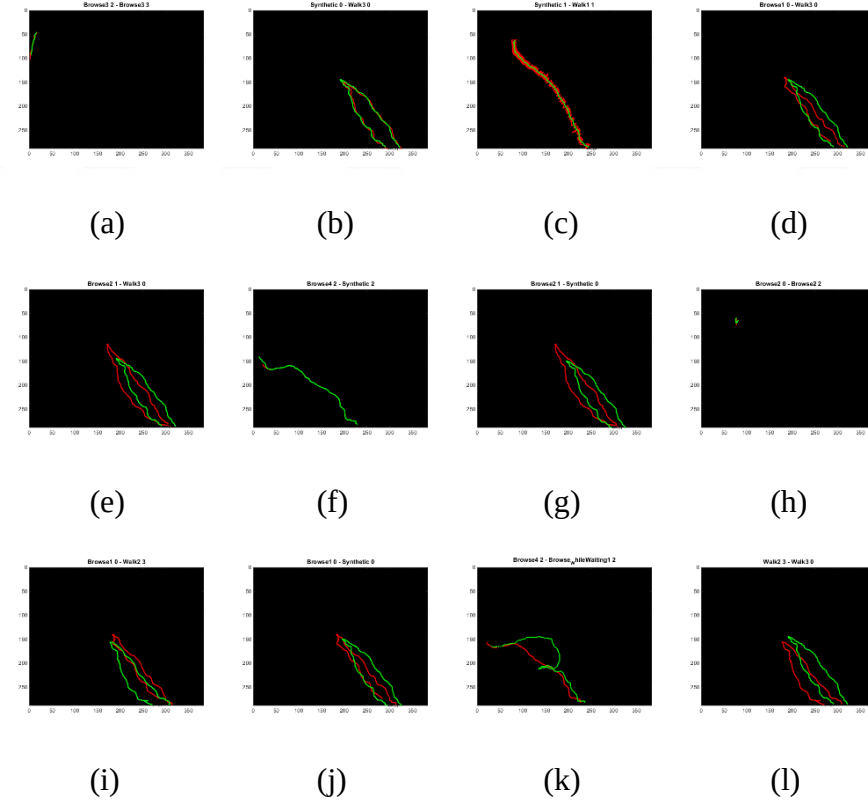
HW 10 parametre için Öklid sonuçları Çizelge 6.12’de paylaşılmıştır. Şekil 6.27 HW Öklid için ilk 16 gidiş yolunu sunmaktadır. HW (32,37) gidiş yolunu üst sıraya taşırken Sentetik2-doğrulamasını alt sıraya ötelemiştir. Buradan çıkarılan sonuç yön bozukluğunda HW’nin başarımı arttırırken, yön verisi önemliyken veya sağlıklıyken daha geride kalmıştır. Parametrenin ağırlığını düşürdüğü için bu durum beklenen sonuçtur.

SW 10 parametre için Öklid sonuçları Çizelge 6.13’de paylaşılmıştır. Şekil 6.28 SW Öklid için gidiş yollarının ilk 16’sının görselini sunmaktadır. SW Öklid eşit ağırlıklı Öklid sonuçlarına çok benzer bir sonuç bulmuştur.

Çizelge 6.14’de toplu bir özet sunulmuştur. En belirgin vurgunun HW’nin üçüncü sentetik veriyi sıralamaya sokmasıdır. HW’nin yön verisinin ağırlığını %2 alması ve sentetik verinin yön değerlerinin rastgele bozulması ve orijinalinden çok farklı olmasının buna neden olduğu değerlendirilmektedir. Ancak bu çıkarımın yan sonucu da HW ile yön haricinde kalan veriler ışığında daha güçlü çıkarım yapılabilmesidir.

Çizelge 6.12. HW Öklid, 10 parametre ile gidiş yolları benzerlikleri.

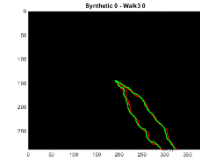
ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	12	13	0,3073	99,03	16	34	41	0,8414	97,34
2	31	44	0,3128	99,01	17	5	40	0,9193	97,09
3	32	37	0,4653	98,53	18	31	41	0,9698	96,93
4	1	44	0,5411	98,29	19	7	40	0,9755	96,92
5	6	44	0,5919	98,13	20	19	33	1,0761	96,60
6	16	33	0,6485	97,95	21	6	41	1,1033	96,51
7	6	31	0,6922	97,81	22	1	34	1,1449	96,38
8	5	7	0,6961	97,80	23	9	14	1,2174	96,15
9	1	41	0,7144	97,74	24	34	44	1,2644	96,00
10	1	31	0,7159	97,74	25	28	31	1,3458	95,74
11	16	19	0,7604	97,60	26	6	34	1,3858	95,62
12	41	44	0,8058	97,45	27	28	44	1,4237	95,50
13	35	36	0,8076	97,45	28	31	34	1,4761	95,33
14	1	6	0,8140	97,43	29	1	28	1,4882	95,29
15	9	22	0,8250	97,39	30	6	28	1,6310	94,84



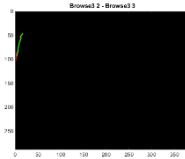
Şekil 6.27. HW Öklid ilk 16 en benzer gidiş yolları.

Çizelge 6.13. SW Öklid, 10 parametre ile gidiş yolları benzerlikleri.

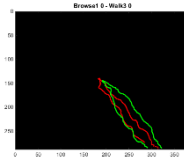
ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	31	44	0,3068	99,03	16	7	40	0,9599	96,96
2	12	13	0,3072	99,03	17	35	36	0,9817	96,90
3	1	44	0,6125	98,06	18	34	41	1,0526	96,67
4	16	33	0,6624	97,91	19	19	33	1,0819	96,58
5	16	19	0,7275	97,70	20	9	14	1,1997	96,21
6	1	41	0,7334	97,68	21	6	41	1,2016	96,20
7	5	7	0,7563	97,61	22	28	31	1,2892	95,92
8	41	44	0,7697	97,57	23	28	44	1,3186	95,83
9	1	31	0,7748	97,55	24	34	44	1,4035	95,56
10	1	6	0,8345	97,36	25	1	28	1,4276	95,49
11	6	44	0,8492	97,31	26	1	34	1,4516	95,41
12	6	31	0,9000	97,15	27	6	28	1,5023	95,25
13	5	40	0,9285	97,06	28	32	37	1,5504	95,10
14	31	41	0,9452	97,01	29	31	34	1,5865	94,98
15	9	22	0,9543	96,98	30	28	41	1,6000	94,94



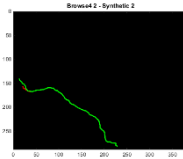
(a)



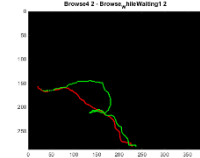
(b)



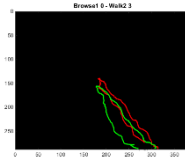
(c)



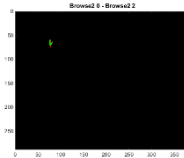
(d)



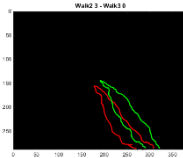
(e)



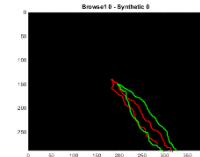
(f)



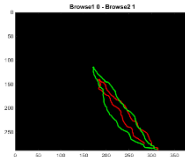
(g)



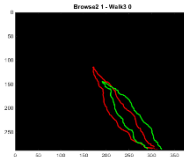
(h)



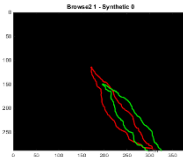
(i)



(j)



(k)



(l)

Şekil 6.28. SW Öklid ilk 16 en benzer gidiş yolları.

Eğer test verisinde yön alanı doğru üretilmiyorsa veya hiç üretilmiyorsa HW yaklaşımının bu durumda ideal yaklaşım olarak kullanılabilir olduğunu göstermektedir.

Yaklaşımların arasındaki gözlenebilir fark çok küçüktür. Yaklaşımlardan herhangi biri seçilse dahi başarımdan büyük bir kayıp yaşanmayacaktır. Ancak sonraki bölümlere eşit ağırlıklı Öklid yaklaşımı taşınarak yöntem zenginleştirilmeye devam edilmiştir.

6.10.4. Seçilen Yöntemin Farklı Parametreler ile Test Edilmesi

Bu bölümde seçilen eşit ağırlıklı Öklid yöntemi üzerinde ilk beş temel parametreye sonradan ilave ettiğimiz başlangıç-bitiş noktaları ve hız parametrelerinin benzeşmeye etkileri test edilmiştir. Her ikisinin birlikte kullanılması ile oluşan sonuçlar daha önce Çizelge 6.10 ve Şekil 6.25’de paylaşılmış ve değerlendirilmiştir.

Çizelge 6.14. LW, Eşit ağırlıklı, MW, HW ve SW Öklid sonuçları özeti.

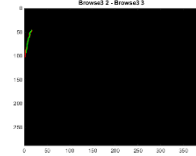
	LW	E	MW	HW	SW
1	(12, 13)	(12, 13)	(12, 13)	(12, 13)	(31, 44)
2	(31, 44)	(31, 44)	(31, 44)	(31, 44)	(12, 13)
3	(16, 33)	(16, 33)	(1, 44)	(32, 37)	(1, 44)
4	(1, 44)	(5, 7)	(16, 33)	(1, 44)	(16, 33)
5	(6, 31)	(41, 44)	(6, 44)	(6, 44)	(16, 19)
6	(1, 6)	(1, 31)	(5, 7)	(16, 33)	(1, 41)
7	(41, 44)	(1, 44)	(6, 31)	(6, 31)	(5, 7)
8	(6, 44)	(1, 6)	(1, 31)	(5, 7)	(41, 44)
9	(5, 7)	(1, 41)	(1, 41)	(1, 41)	(1, 31)
10	(1, 31)	(31, 41)	(41, 44)	(1, 31)	(1, 6)
11	(1, 41)	(6, 31)	(1, 6)	(16, 19)	(6, 44)
12	(35, 36)	(7, 40)	(16, 19)	(41, 44)	(6, 31)
13	(34, 41)	(6, 44)	(35, 36)	(35, 36)	(5, 40)
14	(16, 19)	(35, 36)	(34, 41)	(1, 6)	(31, 41)
15	(32, 37)	(16, 19)	(32, 37)	(9, 22)	(9, 22)
16	(31, 41)	(5, 40)	(31, 41)	(34, 41)	(7, 40)
17	(7, 40)	(34, 41)	(9, 22)	(5, 40)	(35, 36)
18	(6, 41)	(34, 44)	(7, 40)	(31, 41)	(34, 41)
19	(5, 40)	(19, 33)	(5, 40)	(7, 40)	(19, 33)
20	(1, 34)	(6, 41)	(19, 33)	(19, 33)	(9, 14)

6.10.4.1. Başlangıç ve Bitiş Noktasının Etkisi

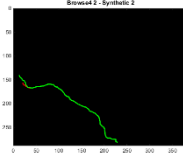
Hızın ağırlığının %0, x_c ve y_c için başlangıç – bitiş noktalarının %100 alınması ile hesaplanmaktadır. Toplamda 9 parametre kullanılmış olmaktadır. Değerler Çizelge 6.15’de, şekiller Şekil 6.29’da sunulmuştur.

Çizelge 6.15. Eşit ağırlıklı Öklid başlangıç bitiş parametresi etkisi.

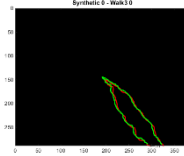
ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	12	13	0,4347	98,63	9	1	41	1,2848	95,94
2	16	33	0,6216	98,03	10	31	41	1,3232	95,82
3	31	44	0,6577	97,92	11	6	31	1,5004	95,26
4	5	7	0,8848	97,20	12	7	40	1,5047	95,24
5	41	44	0,9704	96,93	13	6	44	1,5222	95,19
6	1	31	0,9786	96,91	14	16	19	1,5224	95,19
7	1	44	1,0801	96,58	15	35	36	1,5316	95,16
8	1	6	1,1521	96,36	16	5	40	1,7297	94,53



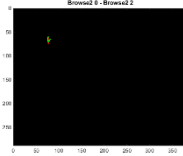
(a)



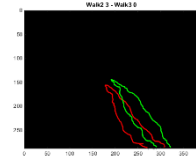
(b)



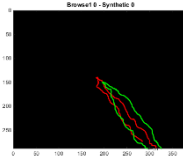
(c)



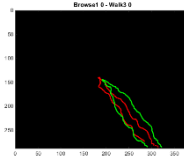
(d)



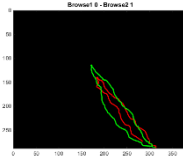
(e)



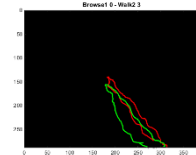
(f)



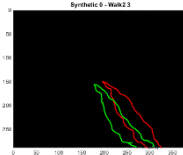
(g)



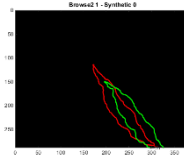
(h)



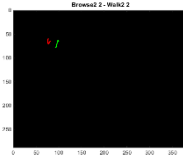
(i)



(j)



(k)



(l)

Şekil 6.29. Eşit ağırlıklı Öklid başlangıç bitiş parametresi etkisi.

Sentetik0-doğrulaması bir sıra yukarı taşınmıştır. Sentetik2-doğrulaması bir alt sıraya kaymıştır. (16, 19) bir sıra yukarı, (35, 36) bir sıra aşağı kaymıştır. Hızın dikkate alınmamasının (hızın ağırlığının 0, başlangıç – bitiş noktalarının ağırlığının %100 alınması) etkisi virgülden sonra 3. haneyi değiştirecek kadardır. Neredeyse ihmal edilebilir düzeydedir.

6.10.4.2. Hızın Başarıma Etkisi

Ağırlıkların x_c ve y_c başlangıç – bitiş noktaları için 0 ve hız için 1 alınması ile hesaplanmaktadır. Toplamda 6 parametre kullanılmış olmaktadır.

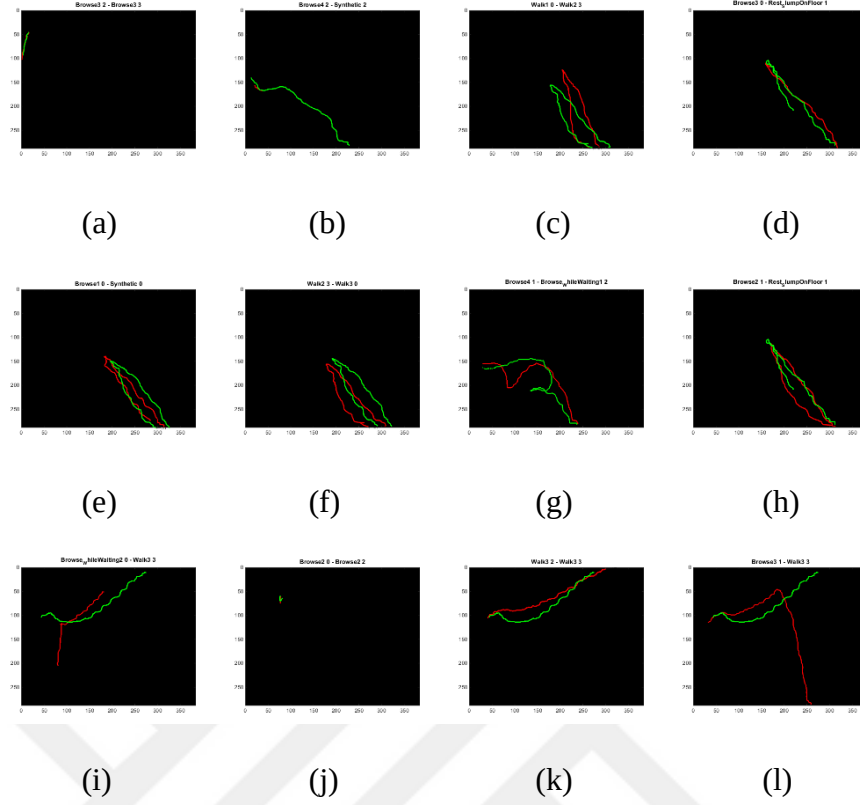
Sonuçlar Çizelge 6.16’de, gidiş yolları ise Şekil 6.30’da sunulmuştur. Bazı gidiş yollarının hem geometrik yapısı hem de hızı benzediği için görsel olarak ifade etmek kolay iken bazı gidiş yollarının sadece hızdan dolayı (hızın ağırlığının artırılması nedeniyle) benzediği görülmektedir. (15, 19), (20, 48) ve (10, 48) başarılı örnekler olarak karşımıza çıkmaktadır.

Çizelge 6.16. Eşit ağırlıklı Öklid hız parametresinin etkisi.

ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	12	13	0,3249	98,97	9	20	48	0,7488	97,63
2	16	33	0,4181	98,68	10	5	7	0,7768	97,54
3	31	44	0,6070	98,08	11	47	48	0,8251	97,39
4	9	25	0,6281	98,01	12	10	48	0,8288	97,38
5	1	31	0,6854	97,83	13	1	6	0,9550	96,98
6	41	44	0,6963	97,80	14	1	44	0,9712	96,93
7	15	19	0,7029	97,78	15	31	41	1,0063	96,82
8	6	25	0,7189	97,73	16	15	16	1,0161	96,79

6.10.5. Ardışık Tekrarlı Verinin Başarıma Etkisi

Bölüm 6.4’de bahsedilen ardışık tekrarlı verinin tekrar eden kısmının tek bir veriyle tanımlanması ile elde edilmiş yeni veri seti üzerinde Eşit ağırlıklı Öklid yaklaşımı çalıştırıldığında elde edilen sonuçlar Çizelge 6.17’de sunulmuştur.



Şekil 6.30. Eşit ağırlıklı Öklid hız parametresinin etkisi.

Çizelge 6.17. Ardışık tekrarlı verinin çıkarılmasının etkisi.

ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	12	13	0,5167	98,37	9	28	31	1,1561	96,34
2	16	33	0,6772	97,86	10	31	41	1,3519	95,72
3	31	44	0,6860	97,83	11	16	19	1,4565	95,39
4	28	44	0,8797	97,22	12	6	28	1,4829	95,31
5	5	7	0,9265	97,07	13	28	41	1,5048	95,24
6	1	6	0,9457	97,01	14	9	14	1,5138	95,21
7	41	44	0,9852	96,88	15	6	31	1,5372	95,14
8	9	22	1,1493	96,37	16	7	40	1,5897	94,97

Ardışık tekrar eden veri barındıran gidiş yolları, gözlem yapan, düşen vb. gibi eylemlere aittir. Tekrar eden verinin tekrar eden kısmı çıkarıldığında bu eylemleri barındıran gidiş yollarının benzerliklerinin artması beklenmektedir.

Genel olarak sonuçlar eşit ağırlıklı 10 parametrelilik yaklaşımının sonuçları ile, yani Çizelge 6.10 ile, kıyaslandığında iki yaklaşım da (12, 13) gidiş yollarını ilk sırada bulurken, Sentetik2-doğrulaması sıralamada bir sıra yukarıya, Sentetik0-doğrulaması

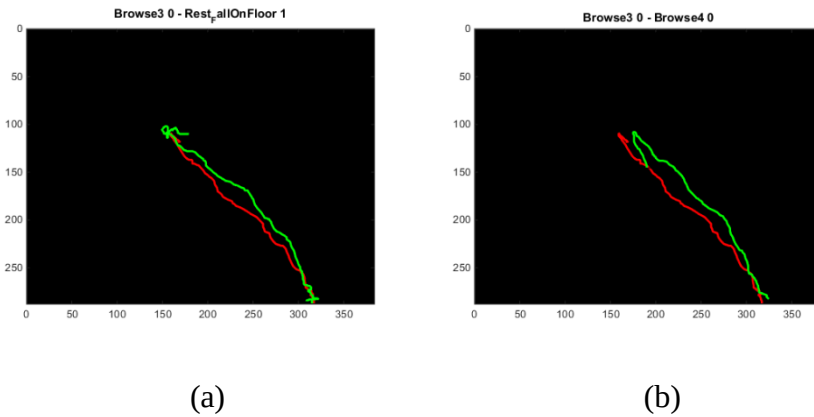
ise bir sıra aşağıya kaymıştır. Sentetik 0 ve Sentetik 2 nesnelerinin yapısı düşünüldüğünde, öteleme ve kısmi bozma usulü ile elde edildikleri için, sıralamadaki değişim negatif yönde değerlendirilebilir.

Tekrarlı ardışık verinin çıkarılması ile (9, 22) gidiş yolu 25. sıradan 8. sıraya çıkmıştır. (9, 14) gidiş yolları 21. sıradan 14. sıraya çıkmıştır. (4, 27) gidiş yolları 51. sıradan 55. sıraya gerilemiştir. (8, 48) gidiş yolları 66. sıradan 38. sıraya çıkmıştır. Gidiş yolları (9, 22) Şekil 6.31 (a)'da, (9, 14) Şekil 6.31 (b)'de, (4, 27) Şekil 6.31 (c)'de ve (8, 48) ise Şekil 6.31 (d)'de gösterilmiştir.

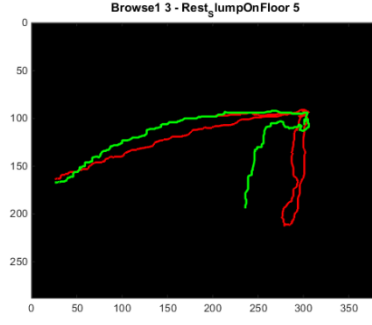
Bu yöntem değerlendirildiğinde genel olarak pozitif bir etkiye sahip olduğu gözlemlenmiştir. Bununla beraber, incelenen örnek sayısı arttırıldığı zaman daha detaylı bir değerlendirme yapılabileceği ön görülmektedir.

6.10.6. Rastgele Benzeşmelerin Önlenmesi

Rastgele benzeşmeleri önlemek için kullanılan temel 5 parametrenin maksimum ve minimum değerleri ayrıca x_c ve y_c değerlerinin standart sapmaları da hesaplamaya dahil edilmiştir. Böylece toplam 22 parametre ile işlem yapılmıştır. Sonuçlar Çizelge 6.18' de verilmiştir. Şekil 6.32 ilk 16 sonucun grafiklerini içermektedir.



Şekil 6.31. Gidiş yolları a) (9, 22), b) (9, 14), c) (4, 27), d) (8,48) benzerliği.



(c)

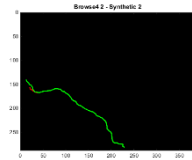


(d)

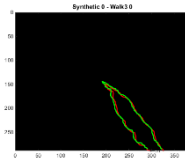
Şekil 6.31. (devam ediyor).

Çizelge 6.18. Eşit ağırlıklı Mak-Min-Std Dev sonuçları.

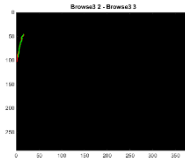
ID	Nesne1	Nesne2	D	Benzerlik (%)	ID	Nesne1	Nesne2	D	Benzerlik (%)
1	16	33	0,9259	98,30	16	39	40	2,9235	94,64
2	31	44	1,0846	98,01	17	6	41	2,9616	94,59
3	12	13	1,0985	97,98	18	5	40	3,0164	94,49
4	5	7	1,4198	97,41	19	35	46	3,1490	94,25
5	41	44	1,6546	96,96	20	21	39	3,1641	94,20
6	35	36	1,8173	96,68	21	6	31	3,3542	93,87
7	16	19	1,9897	96,35	22	6	44	3,3733	93,84
8	31	41	2,0535	96,23	23	9	14	3,4780	93,64
9	19	33	2,1918	95,99	24	6	34	3,6512	93,33
10	34	44	2,5372	95,35	25	36	46	3,6769	93,28
11	34	41	2,5757	95,28	26	9	22	3,7250	93,19
12	7	40	2,7007	95,06	27	37	42	3,8777	92,91
13	31	34	2,7292	94,99	28	4	27	4,2581	92,19
14	1	6	2,8452	94,78	29	15	42	4,2713	92,19
15	32	37	2,8834	94,73	30	14	25	4,2861	92,17



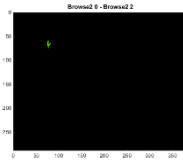
(a)



(b)

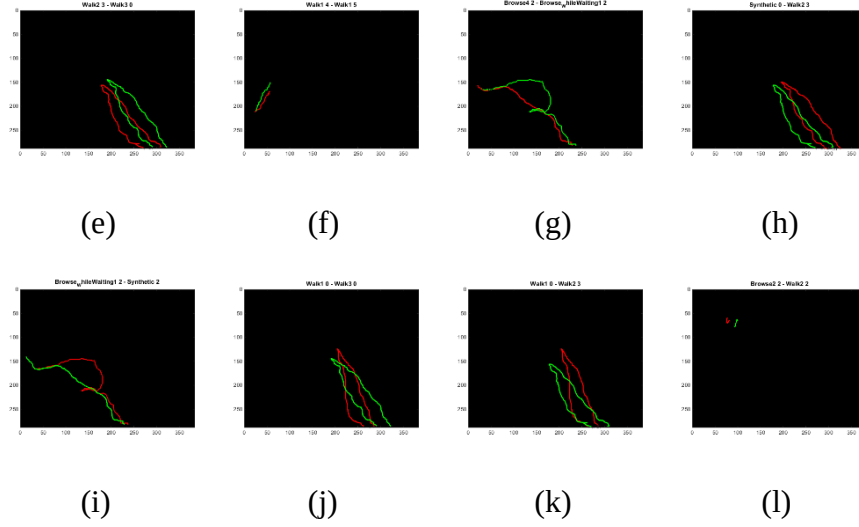


(c)



(d)

Şekil 6.32. LW Öklid Mak-Min-Std Dev sonuçları.



Şekil 6.32. (devam ediyor).

İlk 2 benzeşme sentetik verilerden oluşmaktadır. Üçüncü ise daha önce bahsi geçen (12,13) gidiş yoludur. Benzer şekilde (35, 36) da sentetik 0 gibi gidiş yolu taşınarak oluşturulmuş denebilir veya çakışık gidiş yolu diye tanımlanabilir. Sentetik1 gibi gidiş yollarını üst sıralara taşıması sonradan eklenen parametrelerin doğruluğu arttırdığını göstermektedir. Böylece rastgele benzeşim minimize edilmiştir.

6.10.6. Kümeleme Sonuçları

Yukarıdaki bölümde elde edilen sonuçlar ele alındığında farklı yolların birbirleri ile olan benzerliğinin (uzaklığının) zaten hesaplanmış olması bu sonuçlar ile kümeleme yapılabileceği ve sonuçlara farklı bir perspektiften de bakılabileceği fikrini oluşturmuştur. Buradan hareketle zaten Öklid uzaklıkları hesaplanmışken KNN algoritması ile gidiş yolları kümelendi. Farklı yaklaşımların hassasiyetlerinin farklı olduğu görülmektedir. O nedenle kümelemeyi sonlandırmak için bulunan minimum ve maksimum mesafenin %20'si eşik olarak alınmıştır. Yukarıdaki sonuçların KNN ile kümelendiği halleri bu bölümde sunulmuştur.

3,615 eşik değeri ile durdurulan KNN algoritmasının bulduğu kümeler Çizelge 6.19'da gösterilmiştir.

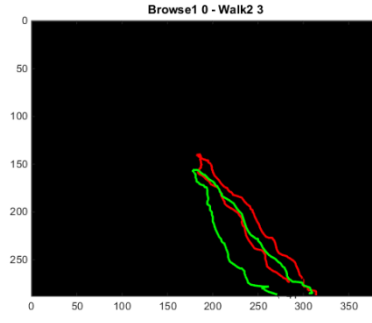
Aşağıda her kümenin elemanları görsel olarak sunulmuştur. Görsel gösterim ile yeni bilgiler ve gidiş yolları hakkında daha detaylı sonuç alınabilmektedir. İlk kümeden hareketle, önceki bölümlerde bulunan (1, 6), (1, 31), ..., (6, 34) gibi kümenin bütün elemanlarının kombinasyonunun başarımının yüksek olması ve doğruluklarının yüksek olduğunu desteklemektedir. Bütün kümelerde benzer durum gözlenmektedir. (4, 27) gidiş yollarını Şekil 6.31 c) görseli önceden 4. küme elemanlarını zaten vermiştir. Şekil 6.33-Şekil 6.38 birden çok elemanı olan kümeleri sunmuştur.

K-means algoritması tarafından bulunan kümeler ile burada KNN ile bulduğumuz kümeleri kıyaslarsak {5, 7, 12, 13, 17} kümesi 5. kümenin alt kümesidir. K-means algoritmasının aynı kümeye sınıflandırdığı 31, 41 gidiş yollarını, KNN 1. kümeye; 16, 33 gidiş yollarını KNN 10. kümeye; 9, 25 gidiş yollarını KNN 7. kümeye; 16, 19 gidiş yollarını KNN 10. kümeye; 4, 27 gidiş yollarını KNN 4. kümeye sınıflandırmıştır. 15 adet gidiş yolunun aynı kümede sınıflandığı her iki algoritma sonuçlarında örtüşmektedir. Bu örtüşme gidiş yolu bazında bir doğrulama olarak kullanılabilir.

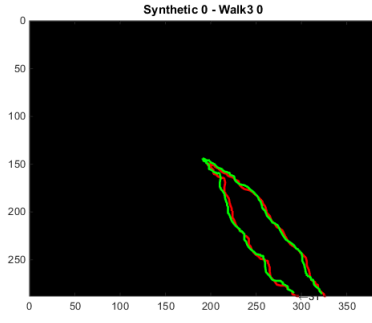
K-means için seçilen küme sayısı (k) ve KNN için eşik değerinin değiştirilebilir olduğu düşünüldüğünde her iki yöntemde farklı kümeler elde etmek mümkün olabilmektedir.

Çizelge 6.19. KNN ile oluşmuş kümeler.

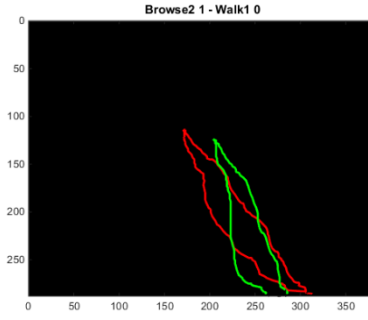
Sıra	Küme Elemanları	Sıra	Küme Elemanları
1	(1, 31, 44, 41, 6, 34)	13	(23)
2	(2, 15, 42, 32, 37)	14	(24)
3	(3, 35, 36, 46)	15	(26)
4	(4, 27)	16	(28)
5	(5, 7, 40, 17, 12, 13, 21, 39)	17	(29)
6	(8)	18	(30)
7	(9, 14, 22, 25)	19	(38)
8	(10)	20	(43)
9	(11)	21	(45)
10	(16, 33, 19)	22	(47)
11	(18)	23	(48)
12	(20)		



(a)



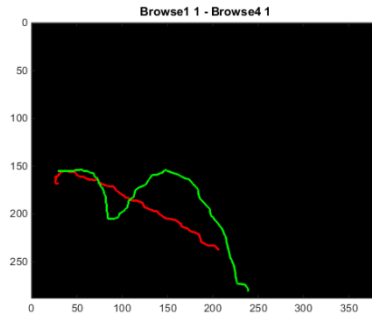
(b)



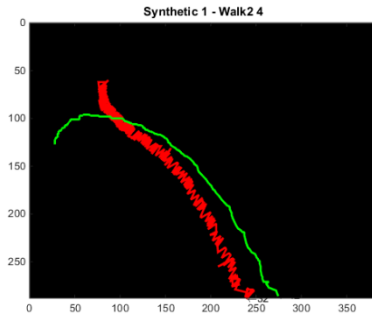
(c)

Şekil 6.33. 1. Küme elemanları a) (1, 41) b) Sentetik0-doğrulaması c) (6, 34).

4. küme (4, 27) “Browsing” etiketi ile etiketlenmiş gözlem aktivitesi içeren iki gidiş yolunu kümelemiştir. Bu anlam davranışın kompleksliği ve tespitinin güçlüğü nedeniyle çok önemli bir bilgidir.

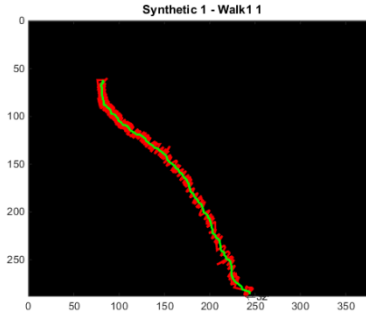


(a)



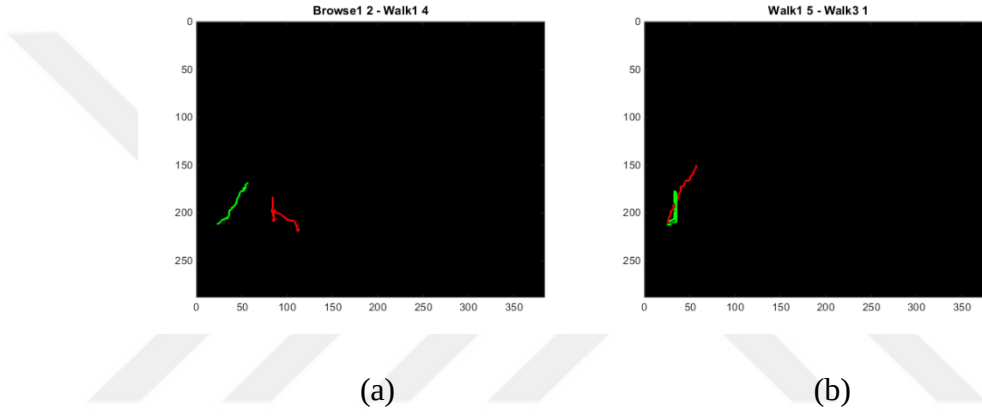
(b)

Şekil 6.34. 2. Küme elemanları a) (2, 15) b) (32, 42) c) (32, 37).

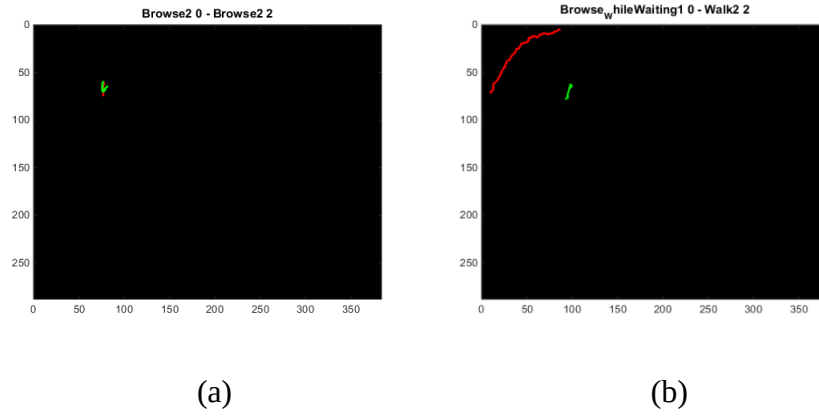


(c)

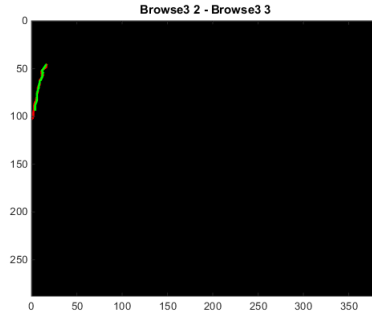
Şekil 6.34. (devam ediyor).



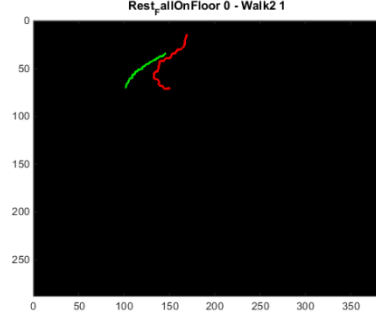
Şekil 6.35. 3. Küme elemanları a) (3, 35) b) (36, 46).



Şekil 6.36. 5. Küme elemanları a) (5, 7) b) (17, 40) c) (12, 13) d) (21, 39).

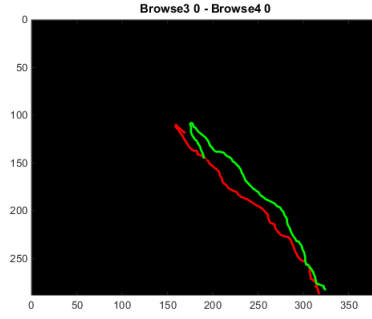


(c)

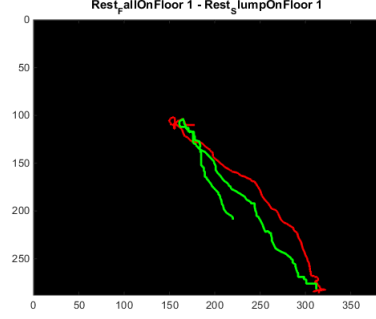


(d)

Şekil 6.36. (devam ediyor).

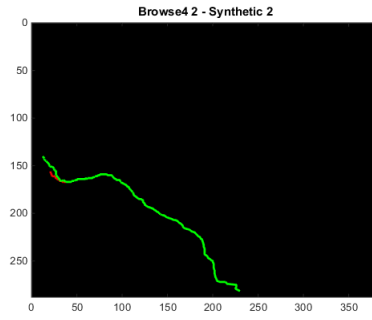


(a)

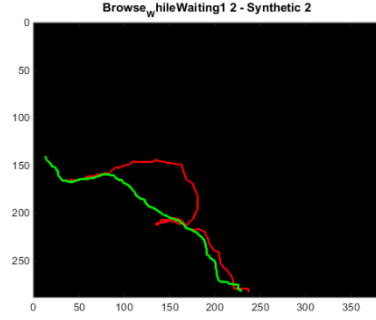


(b)

Şekil 6.37. 7. Küme elemanları a) (9, 14) b) (22, 25).



(a)



(b)

Şekil 6.38. 10. Küme elemanları a) (16, 33) Sentetik2-doğrulaması b) (19, 33).

BÖLÜM 7

SONUÇLAR VE DEĞERLENDİRME

Son kullanıcı için multimedyaadan veya multimedya veriden anlamlı, gizli kalmış bilgi çıkarılması amaçlanan bu çalışmada otomatik etiket üretme işlemi için ve gidiş yolları için iki ayrı model önerilmiştir. Bu modeller kesin referans veri seti üzerinde çalıştırılarak doğrulukları tartışılmıştır.

Birinci model ile ontolojiler üzerinde semantik sorgulama ve anlamlandırma kullanılarak giriş/çıkış noktaları tespit edilmiştir ve nesne için {appear, disappear, visible} (belirmek-görünmek, gözden kaybolmak, görünür) etiketleri başarıyla üretilebilmiştir. Bu model ile herhangi bir bilgisayarlı görü yaklaşımı ile elde edilen test verileri etiketlenebilmektedir.

İkinci model ile nesnenin yönü, (bounding box) sınır çizgilerinin boyu, eni, xc ve yc noktalarına ilaveten giriş/çıkış noktaları ile hız kullanılarak nesnelerin gidiş yollarının benzerlikleri tanımlanmış ve ölçülmüştür. Oluşturulan sentetik verinin kesin referans verisine dahil edilmesi ile çalıştırılan model, başarılı bir şekilde sentetik veriyi seçmiştir.

Doğrulamalar yapıldıktan sonra modelin başarılı olduğu görülünce modeli iyileştirmek için farklı seçenekler denenmiştir. Parametrelere farklı ağırlıklar vererek denenmiş ve en uygun parametre ağırlıkları seçilmiştir. Daha sonra kullanılan parametre önerilen modellerde oluşan gürültüleri bastırarak verimliliği arttırmak için standart sapma ve minimum-maksimum noktalar eklenerek test edilmiştir. Model kullanılarak ölçülen benzerlik mesafeleri üzerinde KNN kümelemesi ile modelde sağlanandan daha fazla bilgi üretildiği de ayrıca görülmüştür.

Ham multimedya verisinden son kullanıcının işine yarayacak bilgi çıkarımı halen açık ve güncel bir konudur. Öncelikle herhangi bir bilgisayarlı görü yöntemi ile izlenen ve tespit edilen nesneler, tezde geliştirilen ve önerilen birinci model ile etiketlenip sonrasında da ikinci model ile ilave anlamlar çıkarılabileceği düşünülmektedir.

Özellikle kümeleme neticesinde ortaya çıkan “Browsing” etkinliği gibi sonuçlar bağlamında bilginin çıkarılması anlamlı ve umut vericidir.

Bu çalışmada elde edilen bilgiler ışığında farklı karşılaştırma veri setleri üzerinde de denemeler yapılarak daha çok veri ve sonuç ile çalışmanın genişletilebileceği ön görülmektedir.

Önerilen modeller sabit kameralar ile elde edilen görüntüler için kamera bazlı gidiş yollarından rotalar tanımlanmasında kullanılabilir. Oluşturulan rotalar ile veri miktarı artınca ilave semantik sorgular ile gizli kalmış bilgilere erişmek için çalışmanın genişletilebileceği değerlendirilmektedir.

Farklı kamera görüntülerindeki nesnelerin yeniden tanınması, yüz tanıma vb. gibi özellikler eklenip ve farklı kamera haritaları kullanıldığında daha büyük ve kapsamlı davranış analizleri ve çıkarımları yapılabileceği öngörülmektedir.

KAYNAKLAR

1. "Statistics", <http://www.youtube.com/yt/press/statistics.html> (2015).
2. Gang Zhai, Fox, G. C., Pierce, M., Wenjun Wu, and Bulut, H., "ESports: Collaborative and Synchronous Video Annotation System in Grid Computing Environment", *Seventh IEEE International Symposium On Multimedia (ISM'05)*, 2005: 95–103 (2005).
3. Troncy, R., Benoit, H., and Simon, S., "Introduction", *Multimedia Semantics*, Wiley, West Sussex, 1–5 (2011).
4. Menemencioğlu, O. and Orak, İ. M., "A Review on Semantic Text and Multimedia Retrieval and Recent Trends", *International Journal Of Multimedia Data Engineering And Management*, 6 (1): 54–74 (2015).
5. İnternet: Berners-Lee, T., "Linked Data", <http://www.w3.org/DesignIssues/LinkedData.html> (2013).
6. Preethi, N. and Devi, T., "Case and Relation (CARE) based Page Rank Algorithm for Semantic Web Search Engines", *International Journal Of Computer Science Issues*, 9 (3): 329–337 (2012).
7. Berners-lee, T. I. M., Hendler, J., and Lassila, O. R. A., "The Semantic Web will enable machines to", *Scientific American*, (May): 1–4 (2001).
8. İnternet: Bratt, S., "Semantic Web: Linked Data on the Web", <http://www.w3.org/2007/Talks/0130-sb-W3CTechSemWeb/layerCake-4.png> (2014).
9. İnternet: Bratt, S., "Semantic Web: Linked Data on the Web", <http://www.w3.org/2007/Talks/0130-sb-W3CTechSemWeb/layerCake-4.png> (2014).
10. İnternet: Nowack, B., "The Semantic Web - Not a Piece of Cake...", <http://bnode.org/blog/2009/07/08/the-semantic-web-not-a-piece-of-cake> (2016).
11. İnternet: Berners-Lee, T., "Linked Data", <http://www.w3.org/DesignIssues/LinkedData.html> (2013).
12. İnternet: W3, "Extensible Markup Language (XML)", <http://www.w3.org/XML/> (2013).
13. Popitsch, N. and Haslhofer, B., "Web Semantics : Science , Services and Agents on the World Wide Web DSNotify – A solution for event detection and link

maintenance in dynamic datasets", *Web Semantics: Science, Services And Agents On The World Wide Web*, 9 (3): 266–283 (2011).

14. Internet: Brickley, D. and Guha, R. V., "RDF Schema 1.1", <http://www.w3.org/TR/rdf-schema/> (2014).
15. Internet: W3, "RDF Examples and Miscellaneous Tests", http://www.w3.org/2000/10/rdf-tests/RDF-Model-Syntax_1.0/RDF-Model-Syntax_1.0.zip (2018).
16. Internet: Prudhommeaux, E., "RDF Schema Example", <http://www.w3.org/2001/Talks/0710-ep-grid/slide21-0.html> (2018).
17. "Web Ontology Language", <http://www.w3.org/2001/sw/wiki/OWL> (2014).
18. Internet: Prud'hommeaux, E. and Seaborne, A., "SPARQL Query Language for RDF", <http://www.w3.org/TR/rdf-sparql-query/> (2014).
19. DuCharme, B., "Learning SPARQL", First Edit. Ed., *O'Reilly Media*, Sebastopol, 235 (2011).
20. Menemencioglu, O. and Orak, I. M., "A Review on Semantic Web and Recent Trends in Its Applications", *2014 IEEE International Conference On Semantic Computing*, 297–303 (2014).
21. Menemencioglu, O. and Orak, İ. M., "A Review on Semantic Text and Multimedia Retrieval and Recent Trends", *Mobile Computing and Wireless Networks: Concepts, Methodologies, Tools, and Applications, IGI Global*, 107–127 (2016).
22. Döller, M. and Kosch, H., "The MPEG-7 Multimedia Database System (MPEG-7 MMDB)", *Journal Of Systems And Software*, 81 (9): 1559–1580 (2008).
23. Ponnada, M. and Sharda, N., "Model of a Semantic Web Search Engine for Multimedia Content Retrieval", *6th IEEE/ACIS International Conference On Computer And Information Science (ICIS 2007)*, (Icis): 818–823 (2007).
24. Arias Gallego, M., Corcho, O., Fernández, J. D., Martínez-Prieto, M. a., and Suárez-Figueroa, M. C., "Compressing semantic metadata for efficient multimedia retrieval", *Lecture Notes In Computer Science (Including Subseries Lecture Notes In Artificial Intelligence And Lecture Notes In Bioinformatics)*, 8109 LNAI: 12–21 (2013).
25. Li, Y., Zhang, T., and Tretter, D., "An Overview of Video Abstraction Techniques An Overview of Video Abstraction Techniques", *Imaging*, 1–23 (2001).
26. Lienhart, R., Pfeiffer, S., and Effelsberg, W., "Video abstracting", *Communications Of The ACM*, 40 (12): 54–62 (1997).
27. Snoek, C. G. M., Worring, M., Van Gemert, J. C., Geusebroek, J.-M., and Smeulders, A. W. M., "The challenge problem for automated detection of 101

- semantic concepts in multimedia", *Proceedings Of The 14th Annual ACM International Conference On Multimedia MULTIMEDIA 06*, 421–430 (2006).
28. Izquierdo, E., Chandramouli, K., Grzegorzec, M., and Piatrik, T., "K-Space Content Management and Retrieval System", *14th International Conference Of Image Analysis And Processing - Workshops (ICIAPW 2007)*, (Iciapw): 131–136 (2007).
 29. Argillander, J., Iyengar, G., and Nock, H., "Semantic Annotation of Multimedia using Maximum Entropy Models", *Proceedings. (ICASSP '05). IEEE International Conference On Acoustics, Speech, And Signal Processing, 2005.*, 2: 153–156 (2005).
 30. Wei, X.-Y. and Yang, Z.-Q., "Coaching the exploration and exploitation in active learning for interactive video retrieval", *IEEE Transactions On Image Processing*, 22 (3): 955–968 (2013).
 31. Rooij, O. d. and Worring, M., "Active bucket categorization for high recall video retrieval", *IEEE Transactions On Multimedia*, 15 (4): 898–907 (2013).
 32. Yildirim, Y., Yazici, A., and Yilmaz, T., "Automatic Semantic Content Extraction in Videos Using a Fuzzy Ontology and Rule-Based Model", *IEEE Transactions On Knowledge And Data Engineering*, 25 (1): 47–61 (2013).
 33. Liao, C.-W., Chan, K.-H., Chang, W.-T., and Tu, S.-T., "Scene-based video analytics studio", *2014 IEEE International Conference On Semantic Computing*, (2014).
 34. Internet: Fisher, R., "CAVIAR Test Case Scenarios", <http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/> .
 35. Yang, B. and Nevatia, R., "Multi-target tracking by online learning of non-linear motion patterns and robust appearance models", *Computer Vision And Pattern Recognition (CVPR), 2012 IEEE Conference On*, 1: 1918–1925 (2012).
 36. Heili, A., López-Méndez, A., and Odobez, J. M., "Exploiting long-term connectivity and visual motion in CRF-based multi-person tracking", *IEEE Transactions On Image Processing*, 23 (7): 3040–3056 (2014).
 37. Friedman, N. and Russell, S., "Image Segmentation in Video Sequences: A Probabilistic Approach", *Proceedings Of The Thirteenth Conference On Uncertainty In Artificial Intelligence*, 175–181 (1997).
 38. Zivkovic, Z., "Improved adaptive Gaussian mixture model for background subtraction", *Proceedings Of The 17th International Conference On Pattern Recognition, 2004. ICPR 2004.*, 2 (2): (2004).
 39. Grimson, W. E. L., Stauffer, C., Romano, R., and Lee, L., "Using adaptive tracking to classify and monitor activities in a site", *Proceedings. 1998 IEEE Computer Society Conference On Computer Vision And Pattern Recognition (Cat. No.98CB36231)*, 22–29 (1998).

40. İnternet: Lacey, T. and Thacker, N., "Tutorial: The Likelihood Interpretation of the Kalman Filter", <http://www.tina-vision.net/docs/memos/1996-002.pdf> (2018).
41. Menemencioglu, O. and Orak, I. M., "Semantic querying on multimedia data", **2016 24th Signal Processing And Communication Application Conference (SIU)**, 1069–1072 (2016).
42. Menemencioglu, O. and Orak, İ. M., "An Attempt at Automatic Label Generation for Object Entry and Exit on Multimedia with a Semantic Search", 7 (1): 25–32 (2018).
43. İnternet: Herman, I., "SPARQL Is a Recommendation", https://www.w3.org/blog/SW/2008/01/15/sparql_is_a_recommendation/ (2018).
44. Bhaumik, H., Bhattacharyya, S., Nath, M. Das, and Chakraborty, S., "Hybrid soft computing approaches to content based video retrieval: A brief review", **Applied Soft Computing**, 46: 1008–1029 (2016).
45. Morris, B. T. and Trivedi, M. M., "A survey of vision-based trajectory learning and analysis for surveillance", **IEEE Transactions On Circuits And Systems For Video Technology**, 18 (8): 1114–1127 (2008).
46. Makris, D. and Ellis, T., "Path detection in video surveillance", **Image And Vision Computing**, 20 (12): 895–903 (2002).
47. Makris, D. and Ellis, T., "Learning Semantic Scene Models From Observing Activity in Visual Surveillance", **IEEE Transactions On Systems, Man And Cybernetics, Part B (Cybernetics)**, 35 (3): 397–408 (2005).
48. Lesot, M. J., Rifqi, M., and Benhadda, H., "Similarity measures for binary and numerical data: a survey", **International Journal Of Knowledge Engineering And Soft Data Paradigms**, 1 (1): 63 (2009).
49. Zhang, T., Liu, S., Xu, C., and Lu, H., "Mining semantic context information for intelligent video surveillance of traffic scenes", **IEEE Transactions On Industrial Informatics**, 9 (1): 149–160 (2013).
50. Lee, G., Mallipeddi, R., and Lee, M., "Trajectory-based vehicle tracking at low frame rates", **Expert Systems With Applications**, 80: 46–57 (2017).
51. Hu, W., Xie, D., Fu, Z., Zeng, W., and Maybank, S., "Semantic-Based Surveillance Video Retrieval", **IEEE Transactions On Image Processing**, 16 (4): 1168–1181 (2007).
52. Smith, L. I., "A tutorial on Principal Components Analysis Introduction", **Statistics**, 51: 52 (2002).

53. Chun, H. and Keleş, S., "Sparse partial least squares regression for simultaneous dimension reduction and variable selection", *Journal Of The Royal Statistical Society: Series B (Statistical Methodology)*, 72 (1): 3–25 (2010).
54. Atas, H., "Çoklu İyon-Seçici Elektrot Sistemleri Kullanılarak Farklı Su Numunelerindeki Bazı İyonların Eş Zamanlı Tayini", *Ankara Üniversitesi*, (2015).
55. Macqueen, J., "Some methods for classification and analysis of multivariate observations", *Proceedings Of The Fifth Berkeley Symposium On Mathematical Statistics And Probability*, 1 (233): 281–297 (1967).
56. "K-Means Clustering", https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html (2017).
57. İnternet: Piech, C., "K Means", <http://stanford.edu/~cpiech/cs221/handouts/kmeans.html> (2017).

ÖZGEÇMİŞ

Oğuzhan MENEMENCİOĞLU 1984 yılında Ereğli / KONYA’da doğdu. İlk okulu tamamladıktan sonra orta okula Karaman Anadolu Lisesi’ni kazanarak devam etti. Lise öğretimine Sivas Fen Lisesi’ni kazanarak devam etti ve ağırlıklı orta öğretim başarı puanı nedeniyle ikinci sınıftan ayrılarak Karaman Fatih Lisesi’ne geçiş yaptı. Dereceye girerek orta öğretimini tamamlayarak Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Bilgisayar Eğitimi bölümünde 2006 yılında lisans derecesini, Karabük Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında 2011 yılında Yüksek Lisans derecesini aldı. Karabük Üniversitesi Bilgisayar Mühendisliği Bölümü’nde 2009 yılında başladığı Araştırma Görevlisi pozisyonunda halen çalışmaya devam etmektedir. Evli ve bir çocuk babasıdır.

ADRES BİLGİLERİ

Adres : Karabük Üniversitesi

Mühendislik Fakültesi Bilgisayar Mühendisliği

Balıklarkayası Mevkii / KARABÜK

Tel : (533) 560 1396

E-posta : omenemencioglu@karabuk.edu.tr