

**ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

DOKTORA TEZİ

ÇOK KAMERALI NESNE TAKİBİ TABANLI VIDEO SİNOPSİS

Kemal Batuhan BAŞKURT

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**ANKARA
2019**

Her hakkı saklıdır

TEZ ONAYI

Kemal Batuhan BAŞKURT tarafından hazırlanan “Çok kameralı nesne takibi tabanlı video sinopsis” adlı tez çalışması 29/01/2019 tarihinde aşağıdaki jüri tarafından oy birliği ile Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Danışman : Prof. Dr. Refik SAMET

Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı



Jüri Üyeleri :

Başkan : Prof. Dr. Suat ÖZDEMİR

Gazi Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı



Üye : Prof. Dr. Refik SAMET

Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı



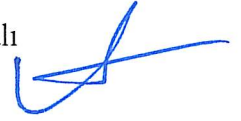
Üye : Doç. Dr. Murat Hüsnü SAZLI

Ankara Üniversitesi Elektrik-Elektronik Mühendisliği Anabilim Dalı



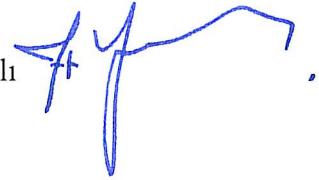
Üye : Doç. Dr. Alptekin TEMİZEL

Orta Doğu Teknik Üniversitesi Modelleme ve Simulasyon Anabilim Dalı



Üye : Dr. Öğr. Üyesi Hacer YALIM KELEŞ

Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı



Yukarıdaki sonucu onaylarım.

Prof. Dr. Atila YETİŞEMİYEN

Enstitü Müdürü

Muzaffer Arıkan'a...



ETİK

Ankara Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım bu tez içindeki bütün bilgilerin doğru ve tam olduğunu, bilgilerin üretilmesi aşamasında bilimsel etiğe uygun davrandığımı, yararlandığım bütün kaynakları atıf yaparak belirttiğimi beyan ederim.

29.01.2019



Kemal Batuhan BAŞKURT

ÖZET

Doktora Tezi

ÇOK KAMERALI NESNE TAKİBİ TABANLI VIDEO SİNOPSİS

Kemal Batuhan BAŞKURT

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Refik SAMET

Kameraların günlük hayatta kullanımının artmasıyla kayıt altında tutulan video miktarı her geçen gün artmaktadır. Etkili bir şekilde görsel tarama ve ilgilenilen olayların tüm veri içinden çıkarılması masraflı ve zaman alıcı bir sorun haline gelmektedir. Saatler süren video kayıtlarının bir kaç dakika içinde gözden geçirilmesine imkan sağlayan video sinopsis, bu soruna etkili bir şekilde çözüm üretmektedir. Video sinopsis, kaynak görüntüden çıkarılan aktivitelerin zaman düzleminde kaydırılarak en kısa zamanda en fazla aktivitenin eş zamanlı gösterimidir.

Tez kapsamında, çok kameralı nesne takibi tabanlı yeni bir çevrimiçi video sinopsis sistemi geliştirilmiştir. Literatürdeki video sinopsis çalışmaları genellikle sadece aktivite optimizasyonuna odaklanmışken tez çalışmasında, kamera görüntüsünün alınmasından video sinopsis oluşturmaya kadarki süreç bütün olarak ele alınmıştır. Kapsamlı bir literatür taraması ile geliştirilen sistemin her adımındaki sorunlar tespit edilerek video sinopsisin genel başarımını artıracak şekilde gerekli katkılar yapılmıştır. Aktivite oluşturma aşamasında çevresel koşullardan minimum düzeyde etkilenmek adına daha önce video sinopsiste kullanılmamış gürbüz bir hareket algılama yöntemi iyileştirilerek kullanılmıştır. Video sinopsisin ihtiyaçları doğrultusunda uzun süreli takibe odaklanmış yeni bir korelasyon filtresi tabanlı nesne takip yöntemi geliştirilmiştir. Algoritmik akışta en kritik adım olan aktivite optimizasyonunu gerçek zamanlı ve etkili bir şekilde gerçekleştirecek grid haritası tabanlı yeni bir çevrimiçi optimizasyon yöntemi geliştirilmiştir. Kameralar arasında nesne eşleştirmesinde coğrafi kalibrasyon yöntemi, video sinopsis çalışmalarında ilk defa kullanılmıştır. Ayrıca tüm yöntemlerin gerçek zamanlı uygulamasını sağlayan, boru hattı mimarisine dayalı yeni bir video sinopsis kütüphanesi geliştirilmiştir. Çalışma kapsamında gerçekleştirilen performans karşılaştırmaları, geliştirilen video sinopsis sisteminin ve bu sistemin alt adımları için önerilen yöntemlerin literatürdeki öncü sistem ve yöntemlerden daha iyi sonuç verdiğini göstermiştir.

Ocak 2019, 136 sayfa

Anahtar Kelimeler: Video gözetleme, video sinopsis, hareket algılama, nesne takibi, çok kameralı nesne takibi, aktivite optimizasyonu

ABSTRACT

Ph.D. Thesis

MULTI-CAMERA OBJECT TRACKING BASED VIDEO SYNOPSIS

Kemal Batuhan BASKURT

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof. Dr. Refik SAMET

Amount of recorded video increases each passing day considering rapid increment on the usage of security cameras in daily life. Efficient video browsing and retrieval become computationally expensive and time consuming. Video synopsis, which allows to review hours of video in a couple of minutes, provides an effective solution to this problem. Video synopsis is simultaneous representation of the activities by shifting them in time domain to find the shortest distribution.

In this thesis, a novel multi-camera object tracking based online video synopsis system is proposed. The whole processing pipeline starting from image capture to build video synopsis is considered and investigated in this thesis. After a comprehensive literature review, the problems in each step of the pipeline is identified and required contributions are made to increase the overall performance of video synopsis. Therefore, a motion detection method, which has not been used in video synopsis before, is improved to increase robustness against environment challenges. A novel visual object tracking method for long-term tracking according to the specified requirements of video synopsis is proposed. A novel grid-map based online activity optimization method is proposed for efficient real-time activity re-arrangement. Geographical coordinate of the activities is used to match objects among the cameras first time in video synopsis studies. Furthermore, a new video synopsis library that is based on pipeline architecture is built to perform all operations in real-time. Performance comparisons on publicly available datasets shows that the proposed system outperforms the state-of-the-art methods in the literature.

January 2019, 136 pages

Key Words: Video surveillance, video synopsis, motion detection, visual object tracking, multi-camera visual object tracking, activity optimization

TEŞEKKÜR

Çalışmalarımı yönlendiren, araştırmalarımın her aşamasında bilgi, öneri ve yardımlarını esirgemeyen danışman hocam Prof. Dr. Refik SAMET'e (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı),

Tez çalışmalarım süresince yapılan tez izleme komitesi toplantılarına katılarak fikir ve önerileriyle çalışmamın olgunlaşmasına yardımcı olan Doç. Dr. Murat Hüsnü SAZLI'ya (Ankara Üniversitesi Elektrik Elektronik Mühendisliği Anabilim Dalı) ve Dr. Öğretim Üyesi Hacer YALIM KELEŞ'e (Ankara Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı),

Değerli yorum ve önerileriyle sağladıkları katkıdan dolayı tez jüri üyeleri; Prof. Dr. Suat ÖZDEMİR'e (Gazi Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı) ve Doç. Dr. Alptekin TEMİZEL'e (Orta Doğu Teknik Üniversitesi Modelleme ve Simulasyon Anabilim Dalı),

Video sinopsis konusu ile tanışmamı sağlayan ve video işleme konusundaki bilgi ve tecrübelerini paylaşarak bu alanda kendimi geliştirmeme imkan sağlayan Kerem ÇALIŞKAN'a,

Bugüne kadar gösterdikleri eşsiz sevgi, destek, fedakarlık ve hoşgörü nedeniyle değerli aileme,

Sağladığı koşulsuz ve sınırsız manevi desteğin yanında teknik konularda zorlandığım her an desteği ile güvende hissetmemi sağlayan eşim Didem'e ve özellikle çalışmalarımı tamamlamam adına sağladığı ekstra motivasyon için oğlum Kutay'a en içten teşekkürlerimi sunarım.

Kemal Batuhan BAŞKURT

Ankara, Ocak 2019

İÇİNDEKİLER

TEZ ONAY SAYFASI

ETİK.....	i
ÖZET.....	ii
ABSTRACT	iii
TEŞEKKÜR	iv
SİMGELER VE KISALTMALAR DİZİNİ	vii
ŞEKİLLER DİZİNİ	x
ÇİZELGELER DİZİNİ	xii
1. GİRİŞ.....	1
1.1 Tezin Kapsamı	2
1.2 Tezin Özgün Değeri.....	3
1.3 Tez Taslağı	4
2. KAYNAK ÖZETLERİ VE LİTERATÜR DEĞERLENDİRMESİ.....	6
2.1 Kaynak Özetleri.....	6
2.1.1 Nesne algılama	6
2.1.2 Nesne takibi.....	8
2.1.3 Çok kameralı nesne takibi.....	12
2.1.4 Aktivite optimizasyonu	17
2.1.5 Arka plan oluşturma	26
2.1.6 Harmanlama	27
2.2 Literatür Değerlendirmesi.....	28
2.2.1 Aktivite optimizasyon tipi.....	30
2.2.2 Kamera topolojisi	37
2.2.3 Girdi veri tipi	40
2.2.4 Aktivite sınıflandırma	41
3. ÇOK KAMERALI NESNE TAKİBİ TABANLI ÇEVİRİMİÇİ VIDEO SİNOPSİS SİSTEMİNİN GELİŞTİRİLMESİ.....	43
3.1 Nesne Algılama İşlemi.....	45
3.1.1 Arka plan / ön plan kararı	47
3.1.2 Arka plan modelini güncelleme	48
3.1.3 Karar eşik değerinin dinamik güncellenmesi	49
3.1.4 Arka plan güncelleme frekansının dinamik güncellenmesi.....	50
3.1.5 Ani değişim algılayıcı	52
3.2 Hareketli Nesne Takip İşlemi.....	54

3.2.1 Korelasyon filtresi tabanlı nesne takibi.....	55
3.2.2 Alternatif korelasyon filtreleri kufllanarak gürbüz nesne takibi	58
3.3 Çok Kameralı Nesne Takip İşlemi.....	65
3.3.1 Kamera kalibrasyonu	66
3.3.2 Alternatif korelasyon filtreleri ile çok kameralı nesne takibi	70
3.4 Aktivite Optimizasyon İşlemi	75
4. BULGULAR VE TARTIŞMA	82
4.1 Nesne Algılama	82
4.2 Hareketli Nesne Takibi.....	85
4.3 Aktivite Optimizasyonu	91
4.3.1 Parametre analizi	95
4.3.2 Literatürdeki öncü yöntemlerle karşılaştırma	105
4.4 Çalışma Zamanı Performans Analizi	108
5. SONUÇ	111
5.1 Öneriler	114
KAYNAKLAR	116
EKLER.....	126
EK 1 Video Sinopsiste Kullanılan Yöntemler	127
EK 2 Korelasyon Filtresinin Elde Edilmesi	133
ÖZGEÇMİŞ.....	135

SİMGELER VE KISALTMALAR DİZİNİ

Simgeler

$I(x_{ij})$	x_{ij} pikselinin değeri
$B(x_{ij})$	x_{ij} pikseline ait arka plan modeli
$B_k(x_{ij})$	x_{ij} pikseline ait arka plan modelindeki k . sıradaki örnek
$R(x_{ij})$	x_{ij} pikseline ait karar eşik değeri
$T(x_{ij})$	x_{ij} pikseline ait arka plan modeli güncelleme frekansı
$\#_{min}$	Arka plan modelinde eşleşmesi beklenen minimum örnek sayısı
$D(x_{ij})$	x_{ij} pikseline ait arka plan model örneklerinin minimum uzaklık vektörü
$F(x_{ij})$	x_{ij} pikselinin hareket durumu
$\bar{d}_{min}(x_{ij})$	x_{ij} pikseline ait arka plan modelinin dinamikliği
$R_{inc/dec}$	Karar eşik değerinin artma/azalma oranları
R_{lower}	Karar eşik değerinin alt sınır değeri
R_{scale}	Sahne dinamikliği ile mevcut eşik değeri arasındaki oran
T_{inc}, T_{dec}	Arka plan modeli güncelleme frekansı artma/azalma oranları
T_{lower}	Arka plan modeli güncelleme frekansı alt sınır değeri
T_{upper}	Arka plan modeli güncelleme frekansı üst sınır değeri
M_{Max}	Sahnedeki hareketli piksellerin toplamı
f	Hedef nesneyi çevreleyen çerçeve
g	Korelasyon filtresinin çıktısı
h	Korelasyon filtresi
F	Fourier düzlemde girdi görüntü
H	Fourier düzlemde korelasyon filtresi
$\odot$	Frekans düzlemde eleman bazında çarpım
$*$	Karmaşık eşlenik
η	Korelasyon filtresi güncelleme katsayısı
g_{max}	Korelasyon filtresinin tepe bölgesine karşılık gelen değerler
μ_{sl}	Sidelobe bölgesinin ortalaması
σ_{sl}	Sidelobe bölgesinin standart sapması
$\mathcal{T}$	Takipçi
HA	Alternatif korelasyon filtreleri
Tr	Nesne izi
S	Takipçi durumu
P	Takipçi havuzu
P_{Un}	Sahipsiz takipçi havuzu
$t_{passive}$	Takipçinin izin verilen maksimum pasif kalma süresi
E_{conf}	Korelasyon hata eşik değeri
PSR_{min}	Kabul edilebilir minimum PSR değeri
E_{conf}	Güvenilir korelasyon hata seviyesi
m_{dev}	Nesne çerçevesinin maksimum sapma oranı
F_k	Nesnenin durum kestirim fonksiyonu
$\hat{x}_k$	Nesnenin en iyi durum kestirimi
P_k	Nesnenin durum kestiriminin kovaryansı
l	Nesne konumu
v	Nesne hızı
E	Enerji fonksiyonu
E_a	Enerji fonksiyonunun aktivite yoğunluk metriği

E_o	Enerji fonksiyonunun uzamsal kesişim metriği
E_c	Enerji fonksiyonunun kronolojik tutarlılık metriği
α	Enerji fonksiyonunun kronolojik tutarlılık metriğinin ağırlık parametresi
β	Enerji fonksiyonunun uzamsal kesişim metriğinin ağırlık parametresi
A_c	Aday aktivite
$R_{inc/dec}^{upper}$	Karar eşik değeri artma/azalma oranlarının üst sınırı
$R_{inc/dec}^{lower}$	Karar eşik değeri artma/azalma oranlarının alt sınırı
T_{inc}^{upper}	Arka plan modeli güncelleme frekansı artma oranının üst sınırı
T_{inc}^{lower}	Arka plan modeli güncelleme frekansı artma oranının alt sınırı
T_{dec}^{upper}	Arka plan modeli güncelleme frekansı azaltma oranının üst sınırı
T_{dec}^{lower}	Arka plan modeli güncelleme frekansı azaltma oranının alt sınırı

Kısaltmalar

ASMS	Ölçek ayarlı ortalama öteleme takipçisi (Scale adaptive mean-shift tracker)
CDR	Kronolojik düzensizlik oranı (Chronological disorder ratio)
CR	Yoğunlaştırma oranı (Compact ratio)
CSK	Çekirdek ile algılamalı takibin dairesel yapısı (Circulant structure of tracking-by-detection with kernels)
DCF	Seçici korelasyon filtresi (Discriminative correlation filter)
DN	Doğru negatif
DP	Doğru pozitif
DSST	Seçici ölçek uzayı takipçisi (Discriminative scale space tracker)
F1	Yakalama ve kesinlik metriklerinin harmonik ortalaması
FFT	Hızlı fourier dönüşümü (Fast fourier transform)
FOV	Görüş alanı (Field of view)
FR	Video karesi yoğunlaştırma oranı (Frame condensation ratio)
GMM	Gauss karışım modeli (Gaussian mixture model)
HBT	Hough tabanlı takipçi (Hough-based tracking)
HOG	Yönlendirilmiş gradyanların histogramı (Histogram of oriented gradients)
HSV	Renk tonu, doygunluk, parlaklık (Hue saturation value)
ICF	Tümleşik korelasyon filtresi (Integrated correlation filter)
IGN_CH	Kronolojik sıralamanın ihmalı (Ignore chronological order)
IGN_CH_GR	Kronolojik sıralamanın ihmalı ve konuma göre sınıflandırma (Ignore chronological order and group by location)
IV	Aydınlanma değişimi (Illumination variation)
JIT	Tam zamanında oluşturucu (Just-in-time renderer)
JVS	Birleşik video sinopsis (Joint video synopsis)
KLT	Lucas-Kanade takipçisi (Lucas-Kanade tracker)
LBP	Yerel ikili örüntü (Local binary pattern)
MACH	Maksimum ortalama korelasyon yüksekliği (Maximum Average Correlation Height)
MAP	Ençok sonsal yordam (Maximum a posteriori)
MIT	Çok örnekli takipçi (Multiple instance tracker)
MKL	Çoklu çekirdek öğrenmesi (Multiple kernel learning)
MOSSE	Hata karelerinin toplamının minimum çıktısı (Minimum Output Sum of Squared Error)
MPR	Çok düzeyli yama yer değiştirme (Multilevel patch relocation)

MSER	Maksimum kararlı ekstrem bölgeler (Maximally Stable Extreme Regions)
Muster	Çoklu depo takipçisi (Multi-Store Tracker)
OCC	Üst üste oturma (Occlusion)
OPBS	Çevrimiçi temel arka plan seçici (Online principal background selection)
OR	Kesişim oranı (Overlap ratio)
OTB	Nesne takip kıyaslaması (Object tracking benchmark)
PBAS	Piksel tabanlı uyarlamalı parçalayıcı (Pixel-based adaptive segmenter)
PRE_CH	Kronolojik sıralamanın korunması (Preserve chronological order)
PSR	Tepenin yan huzmeye oranı (Peak to sidelobe ratio)
RANSAC	Rastgele örnek konsensüsü (Random sample consensus)
RGB	Kırmızı yeşil mavi (Red green blue)
SAMF	Çoklu öznitelikler ile ölçek ayarlı takipçi (Scale adaptive with multiple features tracker)
SIFT	Ölçekten bağımsız öznitelik dönüşümü (Scale invariant feature transform)
SRDCF	Uzamsal düzenlenmiş seçici korelasyon filtresi (Spatially regularized discriminative correlation filter)
S-SVM	Yapılandırılmış destek vektör makinesi (Structured support vector machine)
SURF	Hızlandırılmış gürbüz öznitelikler (Speeded-up Robust Features)
SV	Ölçek değişimi (Scale variation)
SVM	Destek vektör makinesi (Support vector machine)
TLD	Takip öğrenme algılama (Tracking learning detection)
ViBe	Görsel blob çıkarıcı (Visual blob extractor)
VOT	Görsel nesne takibi (Visual object tracking)
VVH	İki ufuk noktası ve bilinen kamera yüksekliği (Two vanishing points and known camera height)
VVW	İki ufuk noktası ve bilinen genişlik (Two vanishing points and known width)
VWH	Bir ufuk noktası, bilinen genişlik ve kamera yüksekliği (One vanishing point, known width and camera height)
YN	Yanlış negatif
YP	Yanlış pozitif

ŞEKİLLER DİZİNİ

Şekil 1.1 Mevcut video sinopsis metodolojisi.....	2
Şekil 2.1 Farklı zamana ait aktivitelerin video sinopsis üzerinde eş zamanlı gösterimi (Rav- Acha vd. 2006)	17
Şekil 2.2 Aktivitelerin yeniden düzenlemesi (Rav-Acha vd. 2006).....	18
Şekil 2.3 Kesişen nesnelerin konum düzleminde kaydırılması (Nie vd. 2013)	22
Şekil 2.4 Uzamsal kesişimi engellemek için nesnelerin boyutlarının küçültülmesi (Li vd. 2016)	23
Şekil 2.5 Video sinopsis çalışmalarının yıllara göre dağılımı (2017 yılı itibariyle)	29
Şekil 2.6 Video sinopsis çalışma sayılarının optimizasyon tipi, kamera topolojisi, aktivite sınıflandırma ve girdi veri tipine göre oranları (2017 yılı itibariyle)	29
Şekil 2.7 Kronolojik tutarlılık metriğinin farklı ağırlıkları için aktivite dağılımı.....	32
Şekil 2.8 Çok kameralı video sinopsis gösterimi (Mahapatra vd. 2016).	38
Şekil 2.9 Sıkıştırılmış düzlemde video sinopsis altyapısı (Zhong vd. 2014)	40
Şekil 3.1 Geliştirilen video sinopsis sisteminin mimarisi	44
Şekil 3.2 İki boyutlu (C1,C2) renk uzayı için 6 örneklili x_{ij} piksel modelinin güncellenmesi	47
Şekil 3.3 Piksel tabanlı karar eşik değerinin $R(x_{ij})$ değişimi (Hofmann vd. 2012).....	50
Şekil 3.4 Ani değişim algılayıcısının etkisi.....	53
Şekil 3.5 Korelasyon filtresi kestirimi ile hareket blobu arasındaki fark	64
Şekil 3.6 Dünya ve kamera koordinat sistemleri arasındaki dönüşüm	67
Şekil 3.7 Kameraların görüş alanlarının sınırları	69
Şekil 3.8 Kamera kalibrasyonu akış diyagramı.....	70
Şekil 3.9 Çok kameralı nesne takibi akış diyagramı	72
Şekil 3.10 i ve j aktivitelerinin izlerine ait piksel koordinatlarının karşılaştırılması	76
Şekil 3.11 Görüntünün gridlere bölünmesi	76
Şekil 3.12 i ve j aktivitelerinin izlerine ait gridlerin karşılaştırılması.....	77
Şekil 3.13 Aktivite grid haritası	79
Şekil 4.1 Yöntemlerin karşılaştırılması: a. tüm veri kümesi üzerinde, b. aydınlanma değişimi, c. ölçek değişimi, d. engelleme.....	88
Şekil 4.2 Aydınlatma değişimi (IV), ölçek değişimi (SV), kesişim (OCC) ve tüm veri kümesi ortalaması (Hepsi) için dayanıklılık karşılaştırması.	89
Şekil 4.3 Tüm veri kümesi üzerinde ortalama kare işleme zamanlarının karşılaştırması	90
Şekil 4.4 Çok iş parçacıklı implementasyon	91
Şekil 4.5 Kullanılan veri kümelerine ait görüntüler	94
Şekil 4.6 D9 (cam1), D10 (cam2), D11 (cam3) veri kümelerinde kameraların yerleşimi ve örnek görüntüleri (Zhu vd. 2016a).....	95
Şekil 4.7 Aynı senaryo için sırasıyla IGN_CH, IGN_CH_GR, PRE_CH ve NO_CRITERIA konfigürasyonlarının video sinopsis başlangıç durumları... ..	96
Şekil 4.8 FR sonuçları	97
Şekil 4.9 CR sonuçları	97
Şekil 4.10 OR sonuçları	98
Şekil 4.11 CDR sonuçları.....	98
Şekil 4.12 Farklı boyutlardaki grid haritaları	99
Şekil 4.13 8x4 ve 8x8 grid boyutları için FR sonuçları	100

ÇİZELGELER DİZİNİ

Çizelge 2.1 Aktivite sınıflandırma yöntemleri	41
Çizelge 4.1 Hareket algılama için kullanılan performans metrikleri	82
Çizelge 4.2 Önerilen yöntemin Change Detection Dataset üzerinde PBAS ile karşılaştırılması	84
Çizelge 4.3 Sahnede aynı anda görülen nesne sayılarına göre ortalama kare işleme süresi.....	91



1. GİRİŞ

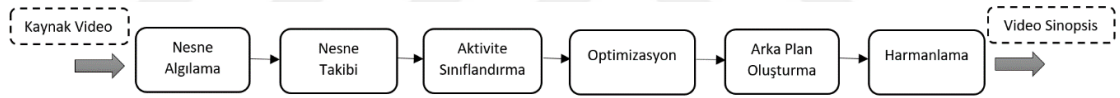
Kamera teknolojisinin ucuzlaması, dijital veri depolama birimlerinin gelişmesi, artan güvenlik ihtiyacı ve görüntü işleme konusundaki teknolojilerin belirgin bir olgunluğa ulaşması sayesinde, güvenlik kameraları ve bu kameralardan elde edilen videoları işleyen uygulamalar günlük hayatımızda her geçen gün daha fazla kullanılmaktadır. Kamera görüntüleri kapalı devre güvenlik sistemlerinde görevliler tarafından sürekli olarak izlenmektedir. Bu görüntüler, gözetleme dışında kritik alan kontrolü, trafik kural ihlali, kontrollü geçiş gibi birçok alanda insan gücüyle gerçekleştirilen işlemlerin akıllı kamera sistemleri tarafından daha güvenilir ve masrafsız bir şekilde yapılmasında kullanılmaktadır. Gözetleme veya analiz için kullanılan kamera görüntüleri, geriye dönük kanıt niteliği taşıyacak şekilde kayıt altına alınmaktadır. Kamera sayısındaki artış dikkate alındığında, kaydedilen verinin büyüklüğü her geçen gün artmakta ve kontrolü zorlaşmaktadır. İzleme yapan görevli sayısını aynı oranda artırmak sorunu çözmek için yeterli olmamaktadır.

Görsel verinin kontrolünü kolaylaştırmak adına, video görüntülerinin kullanıcı için daha anlamlı olan kısımlarının tüm veri içinden çıkarılarak video yoğunluğunu artırmayı amaçlayan yöntemler bulunmaktadır. Bu yöntemler, video özetleme ve video sinopsis olmak üzere ikiye ayrılmaktadır. Video özetleme, hareketsiz veya ilgilenilmeyen video karelerini çıkararak geriye kalan karelerin arka arkaya gösterilmesiyle özet görüntü oluşturmaktır (Smith ve Kanade 1998, Truong ve Venkatesh 2007, Chakraborty vd. 2015). Video sinopsis, kamera görüntüsündeki belli bir zamana ait (örnek olarak 24 saat) aktivitelerin daha kısa bir süre (örnek olarak 5 dakika) içinde eş zamanlı olarak gösterilmesini sağlayan aktivite tabanlı bir yöntemdir (Rav-Acha vd. 2006). Aktivite, bir nesnenin tüm video kareleri boyunca takip edilip ilişkilendirilen örneklerinin tamamını temsil etmektedir. Video sinopsis, kaynak videodan ilgilenilen aktiviteleri çıkartıp zaman düzleminde kaydırarak eş zamanlı olarak göstermeyi amaçlamaktadır. Kaynak görüntüde farklı zamanlara ait aktiviteler video sinopsiste aynı kare üzerinde eş zamanlı olarak gösterilebilmektedir. Bu da, video karesi tabanlı video özetleme yöntemlerine göre daha etkili bir özetleme performansı ortaya koymaktadır. Aktivite yoğunluğunun fazla olması sayesinde kullanıcı daha kısa sürede daha fazla olayı

gözlemleyebilmektedir. Video sinopsis, aktivite tabanlı olması sayesinde, video karesi tabanlı özetleme yöntemlerine göre özet yoğunluğu açısından daha verimli sonuçlar üretmektedir.

1.1 Tezin Kapsamı

Literatürde yer alan video sinopsis çalışmaları incelendiğinde, mevcut video sinopsis metodolojisi (Şekil 1.1) hedef nesnelerin algılanması ile başlamaktadır. Algılanan nesneler görüntüye girişinden çıkışına kadar takip edilerek aktivite olarak tanımlanmaktadır. Bazı yaklaşımlar aktiviteleri belirli kriterlere göre sınıflandırarak benzer aktiviteleri beraber göstermektedir. Seçilen aktiviteler, toplamda en kısa video sinopsisi oluşturacak şekilde zaman düzleminde kaydırılarak yeniden düzenlenmektedir. Optimal pozisyonları tespit edilen aktiviteler, oluşturulan arka plan görüntüsü üzerine yerleştirilerek video sinopsis oluşturulmaktadır.



Şekil 1.1 Mevcut video sinopsis metodolojisi

Metodolojide yer alan adımlar birbirini besleyerek devam etmektedir. Özellikle başlangıçtaki adımlarda meydana gelen hatalar algoritmik akışın sonuna kadar taşınmakta ve genel başarımı düşürmektedir. Mevcut video sinopsis çalışmalarının büyük çoğunluğu aktivite optimizasyonu adımına yoğunlaşmakta ve öncesindeki adımlardaki eksiklikleri göz ardı etmektedir. Tez kapsamında video sinopsis, sadece aktivite optimizasyon işlemi olarak değil kamera görüntüsünün alınmasından video sinopsisin oluşmasına kadarki süreci kapsayan bir bütün olarak ele alınmıştır. Dolayısıyla metodolojideki her adım incelenmiş, her birinin video sinopsis başarımı üzerindeki etkisi analiz edilmiştir. Sorunlar tespit edildikten sonra video sinopsisin başarımını en yükseğe çıkaracak şekilde metodolojideki her adımda gerekli katkılar yapılmıştır.

Bu doğrultuda, tez çalışmasında ilk olarak sabit kamera görüntüleri üzerinde ilgilenilen nesneler algılanmıştır. Nesne algılama için istatistiksel modelleme tabanlı bir hareket algılama yöntemi uygulanmıştır. Hareket olarak algılanan pikseller görüntü bölütleme yöntemi ile hareketli nesne olarak birleştirilmiştir. Hareket olarak algılanan nesneler video kareleri arasında ilişkilendirilerek takip edilmiş ve aktivite haline getirilmiştir. Kamera kalibrasyonu yardımı ile, algılanan nesnelerin kameraya göre coğrafi konumları tespit edilip takip işlemi desteklenmiştir. Kamera özelinde takip edilen nesnelerin öznitelikleri saklanarak diğer kamera görüntülerinde yakalanan nesneler ile eşleştirilmiştir. Çok kameralı nesne takibi, coğrafi kalibrasyondan gelen nesne konum bilgileri ile desteklenerek, nesnelerin maksimum hız gibi kriterleri dikkate alınarak, öznitelik bazında yanlış eşleşen nesneler elenmiştir. Nesneler takip edilip ait olduğu video kareleri ile etiketlendikten sonra istenilen zaman aralığına ait aktiviteler, geliştirilen çevrimiçi optimizasyon yöntemi ile video sinopsiste uygun yerlere yerleştirilmiştir. Bu yerleştirme işleminde, mümkün olan en fazla sayıda aktivitenin, minimum uzamsal kesişim ile en kısa sürede kullanıcıya gösterilmesi amaçlanmıştır. Nesneler zaman düzleminde kaydırılarak en uygun konumları bulunduktan sonra, sahneye ait bir arka plan görüntüsü üzerine yerleştirilerek gösterilmiştir.

1.2 Tezin Özgün Değeri

Tez kapsamında geliştirilen çok kameralı çevrimiçi video sinopsis sistemi ile, nesnelerin birden fazla sabit kamera üzerinde görsel ve coğrafi konum bilgileri dikkate alınarak takip edilmesi ve aktivitenin ait olduğu tüm kameralarda video sinopsis dahilinde gösterilmesi sağlanmıştır. Uygulanan çok kameralı nesne takip yönteminde, kamera görüntülerinin kesişmesi beklenmemektedir. Bu yönden geliştirilen sistem, görüntüsü kesişmeyen çok kameralı sistemde görsel eşleştirme ile çevrimiçi aktivite optimizasyonu yapan ilk video sinopsis sistemidir. Günlük hayatta kamu ve özel alanlarda yaygın olarak kullanılan güvenlik kameralarının görüntüsünün kesişme zorunluluğu olmadığı düşünüldüğünde, geliştirilen yöntem daha geniş bir uygulama alanına sahip olmaktadır.

Literatürde yer alan çalışmaların tamamı, video sinopsisin belirli bir adımına yoğunlaşmıştır. Tez kapsamında video sinopsis metodolojisini bir bütün olarak ele alıp, her adımda video sinopsis için en uygun sonucun elde edilmesini amaçlayan kapsamlı bir çalışma gerçekleştirilmiştir. Bu anlamda gerçekleştirilen katkılar aşağıdaki gibi listelenebilir.

1. Yeni bir çok kameralı nesne takibi tabanlı çevrimiçi video sinopsis sistemi geliştirilmiştir.
2. Çevrimiçi aktivite optimizasyonu için yeni bir yöntem geliştirilmiştir.
3. Hareketli nesne takibi için yeni bir yöntem geliştirilmiştir.
4. Video sinopsis sistemlerinde kullanılmamış yeni bir hareket algılama yöntemi iyileştirilerek kullanılmıştır.
5. Kameralar arasında aktivite eşleştirmesinde coğrafi kalibrasyon kullanılmıştır.
6. Video sinopsis sisteminde yer alan tüm işlemlerin gerçek zamanlı olarak uygulandığı boru hattı mimarisine dayalı yeni bir video sinopsis kütüphanesi geliştirilmiştir.

Uygulama alanı olarak bakıldığında tez çalışması sonucunda elde edilen video sinopsis sistemi, yaygın olarak kullanılan kapalı devre kamera sistemlerinde kullanıcıya inisiyatif alma zorunluluğu bırakmadan otomatik olarak raporlama sağlamaktadır. Bu da, insan kaynaklı hata ve ihmalleri minimuma indirmektedir. Görsel verinin taranması ve raporlanmasını kolaylaştırmaktadır. Güvenlik alanında yaygın olarak uygulanan periyodik yazılı raporlar yerine, aktivitelerin video sinopsis halinde sunulduğu görsel raporlara imkan sağlamaktadır. Tez kapsamında geliştirilen gerçek zamanlı video sinopsis sistemi, güvenlik/izleme sektörü dışında internet ortamında paylaşılan videolar üzerinde de uygulanabilmekte ve aranan içeriğe hızlı ve kolay bir şekilde ulaşılmasına imkan sağlamaktadır.

1.3 Tez Taslağı

Tezin 2. bölümünde, literatürde yer alan video sinopsis çalışmalarının detaylı özeti yer almaktadır. Video sinopsis çalışmalarında takip edilen metodolojinin her bir adımı ayrı başlıklar halinde incelenmiştir. Her adımda kullanılan yöntemler, avantaj ve

dezavantajlarıyla analiz edilmiştir. 2. bölümün devamında mevcut video sinopsis çalışmaları; kullanılan optimizasyon tipi, uygulandığı kamera topolojisi, girdi veri tipi ve aktivite sınıflandırma kategorilerine göre analiz edilmiştir. Literatürde yer alan çalışmaların yıllara göre dağılımı ve belirlenen kategorilere göre araştırma eğiliminin ne yönde değiştiği tartışılmıştır. 3. bölümde, çok kameralı nesne takibi tabanlı çevrimiçi video sinopsis sistemi önerilmiş ve her bir adımı detaylı olarak açıklanmıştır. Video sinopsisin ihtiyaçları doğrultusunda iyileştirilen hareket algılama yöntemi, uzun süreli nesne takibine uygun olarak geliştirilen nesne takip yöntemi, uygulanan çok kameralı nesne takip yöntemi ve geliştirilen çevrimiçi aktivite optimizasyon yönteminin teknik detayları paylaşılmıştır. 4. bölümde, her bir adımda geliştirilen yöntemlerin veri kümeleri üzerinde uygulaması ve elde edilen sonuçların karşılaştırması yapılmış ve sonuçlar tartışılmıştır. 5. bölümde, tez araştırmasının sonuçları ve öneriler paylaşılmıştır.

2. KAYNAK ÖZETLERİ VE LİTERATÜR DEĞERLENDİRMESİ

Bu bölümde, literatürde yer alan video sinopsis çalışmaları iki başlık altında incelenmiştir. Bölüm 2.1’de öncelikle, yayınlanmış video sinopsis çalışmalarında kullanılan yöntemler değerlendirilmiştir. Çalışmaların özgün değerleri, avantaj ve dezavantajları yorumlanmıştır. Bölüm 2.2’de, literatürde yer alan video sinopsis çalışmaları; aktivite optimizasyon tipi (çevrimiçi/çevrimdışı), kamera topolojisi (tek/birden çok kamera), girdi veri tipi (piksel/sıkıştırılmış veri) ve aktivite sınıflandırma (sınıflandırılmış/tüm aktiviteler) kategorilerine göre analiz edilmiştir. Her bir kategoride tercih edilen yöntemlerin avantaj ve dezavantajları yorumlanmıştır. Ayrıca video sinopsis çalışmalarının yıllara göre dağılımı ve bu kategorilerin dağılımdaki ağırlıkları da analiz edilmiştir. Mevcut video sinopsis çalışmalarını kapsayan detaylı bir literatür analizi sunulmuş ve çalışmaların her bir adımda hangi yöntemleri kullandığı EK 1 çizelge 1’de listelenmiştir.

2.1 Kaynak Özetleri

Bu bölümün alt başlıklarında video sinopsis çalışmalarında kullanılan nesne algılama, nesne takibi, çok kameralı nesne takibi, aktivite optimizasyonu, arka plan oluşturma ve harmanlama (stitching) yöntemleri incelenmiştir.

2.1.1 Nesne algılama

Literatürde nesne algılama, video sinopsisin ilk adımı olarak yer almaktadır. Video sinopsis çalışmalarında en yaygın kullanılan nesne algılama yöntemi hareket algılamadır. Kullanılan hareket algılama yöntemleri içinde en popülerleri ise *Gaussian Mixture Model (GMM)* yöntemidir (Stauffer ve Grimson 2000). GMM yönteminin geliştirilmiş versiyonları, *background cut* (Sun vd. 2006), *min-cut* (Kolmogorov ve Zabih 2004), *adaptive background modeling* (Stauffer ve Grimson 1999), *hierarchical background modeling* (Chen vd. 2007) ve *ViBe* (Barnich ve Van Droogenbroeck 2011) video sinopsis çalışmalarında nesne algılama için kullanılan diğer hareket algılama

yöntemleridir. EK 1 çizelge 1’de görüldüğü gibi bazı çalışmalar ardışık video karelerinin farkları veya video karesinin belirli bir zaman dilimine ait ortalama kare ile karşılaştırılması gibi daha basit yöntemler kullanmıştır. Bu basit yöntemler, dinamik arka plan nesnelere sahip yoğun aktivitenin olduğu ve aydınlatma gibi çevresel koşulların değişkenlik gösterdiği sahnelerde kötü performans göstermektedir. Pritch vd. (2009), çalışma zamanı performansını artırmak için nesne algılama frekansını düşürmeyi önermiştir. Bunun dışındaki video sinopsis çalışmaları, nesne algılamayı her video karesine uygulamıştır. Hareket algılama konusundaki literatür taramaları incelendiğinde, video sinopsis çalışmalarında şu ana kadar kullanılan hareket algılama yöntemleri arasında, çalışma zamanı ve başarımlar dengesi dikkate alındığında, en iyi sonucun ViBe yöntemi tarafından elde edildiği görülmüştür (Goyette vd. 2012).

Yukarıda bahsedilen arka plan modelleme tabanlı yöntemlerden farklı olarak, Rodriguez (2010), nesne algılama için optik akış yöntemini (Papenberg vd. 2006) kullanmıştır. Bu sayede, video sinopsisi hareketli kameralar üzerinde uygulamıştır. Bu açıdan bakıldığında, video sinopsiste kullanılan nesne algılama adımına en özgün katkıyı bu çalışma sağlamıştır. Nesne algılama için hareket dışında kullanılan diğer bir seçenek ise güdümlü öğrenme tabanlı insan algılama yöntemleridir (Kasamwattanarote vd. 2010, Mahapatra vd. 2016). Bu yöntemlerin gürültüden etkilenme ve yanlış algılama ihtimalleri daha düşük olduğu için daha başarılı sonuçlar elde edilmiştir. Fakat güdümlü öğrenme tabanlı yöntemlerde, hedef nesne tipine göre önceden eğitim yapılması gerekmektedir. Bunların dışında *quadtree* (Finkel ve Bentley 1974), *motion vector-based LBP* (Wang vd. 2013a) ve *3D graph-cut* yöntemleri sıkıştırılmış veri üzerinde nesne algılama için kullanılmıştır. Bu yöntemler piksel tabanlı yöntemlere göre daha düşük performans göstermiştir.

Sonuç olarak nesne algılama, video sinopsis algoritmik akışında bir ön-işlem olarak görülmektedir. Bu yüzden video sinopsis çalışmaları, bu adımda herhangi bir iyileştirme yapmak yerine literatürde kabul görmüş yöntemleri kullanmıştır. Buna rağmen özellikle hareket algılama konusundaki literatür taramaları, video sinopsis içinde kullanılan daha etkili yöntemlerin olduğunu ve bu sayede başarımın önemli ölçüde artacağını göstermiştir (Wang vd. 2014). Her ne kadar ön-işlem olarak görülsün de nesne algılama

adımındaki yanlış algılamalar, video sinopsisin görsel kalitesini ve optimizasyonun hesaplama karmaşıklığı doğrudan etkilemektedir. Bu yüzden, bu adımın iyileştirilmesine daha fazla efor harcanmalıdır. Bu sayede video sinopsis sisteminin genel başarımına katkı sağlanmış olacaktır.

2.1.2 Nesne takibi

Nesne takibi ile amaç, ardışık video kareleri arasında algılanan nesneleri eşleştirerek ilgili nesnenin sahneye girişinden çıkışına kadar tüm bilgilerini tutmaktır. Nesne takibi sonucunda, video sinopsisin optimizasyon aşamasına girdi olarak verilen aktiviteler oluşturulmaktadır. Bu aşama, tüm video sinopsis çalışmalarında yer alan zorunlu bir adımdır.

Video sinopsiste kullanılan nesne takip yöntemleri konusunda çeşitlilik mevcuttur. Pritch vd. (2008) tarafından önerilen *tube generation* yöntemi sonraki çalışmalar tarafından da yaygın bir şekilde kullanılmıştır. Bunun yanı sıra *sticky tracking* video sinopsis için özelleşmiş en popüler nesne takip yöntemidir (Feng vd. 2012). Video sinopsisteki hayalet nesne ve anlık kaybolup tekrar görülmeleri engellemek için uzamsal olarak kesişen nesneler, tek bir nesne olarak birleştirilerek aktivite haline getirilmiştir. Bu yöntemin önceliği, kusursuz nesne takibi yapmak yerine anlamsal bütünlüğü korumak adına aktivitenin kopmadan video sinopsiste gösterilmesidir. *Clifford worm extraction* (Rodriguez 2010), *blob sequence optimization* (Lin vd. 2015), *3D graph-cut based tracking* (Zhong vd. 2014) ve *clustered track extraction* (Mahapatra vd. 2016) video sinopsis çalışmalarında kullanılan diğer nesne takip yöntemleridir. Bu çalışmalardaki nesne takip yöntemleri, kendi optimizasyon yöntemlerine özgü geliştirildikleri için başka çalışmalarda kullanılmamıştır. Bu noktada *sticky tracking* diğerlerinden farklı olarak herhangi bir çalışmada kullanılabilmesi açısından öne çıkmıştır. EK 1 çizelge 1’de görüldüğü gibi bir çok çalışma tarafından kullanılmıştır.

Bunların dışında Kalman filtresi (Kalman 1960), parçacık (particle) filtresi (Pérez vd. 2002), *multiple object tracking* (Yang vd. 2005), *tracking with shadow detection* (KaewTraKulPong ve Bowden 2002), *multi-feature graph-based object tracking* (Taj

vd. 2006), *least mean square tracking* (Hsia vd. 2011), *multiple pedestrian tracking* (Zhang vd. 2014) ve *nearest neighbor-based tracking* (Choeychuen vd. 2006) video sinopsis çalışmalarında yaygın olarak kullanılan nesne takip yöntemleridir. Bu yöntemler, konu ile ilgili literatür taramalarında ayrıntılı olarak tartışıldığı gibi başarımları ve çalışma zamanı arasında önemli bir ödünleşime sahiptir (Kristan vd. 2015). Nesne çerçevelerinin video kareleri arasındaki *euclidean* veya *chi-square* uzaklıklarını hesaplayan veya Kalman filtresi gibi doğrusal kestirim yapan karmaşıklığı düşük yöntemler, gerçek-zamanlı çalışırken düşük başarımları göstermektedir. Hesaplama karmaşıklığı yüksek yöntemler gölge, aydınlatma değişimleri gibi çevresel koşullar ve hedef nesnenin görünüm değişimlerine karşı daha dayanıklı olmakla beraber gerçek-zamanlı çalışma performansından uzaklaşmaktadır.

Şu ana kadar yapılan video sinopsis çalışmalarında kullanılan nesne takip yöntemleri dışında literatürde birçok etkili yöntem bulunmaktadır. Bu bölümün devamında bu yöntemlerden bahsedilmiştir. Nesne takibi alanında popüler yöntemlerden biri olan TLD (Tracking Learning Detection), hedef nesneyi modellemek için çevrimiçi seçici sınıflandırıcı oluşturmuştur (Kalal vd. 2012). Sınıflandırıcıdan alınan tespit bilgisi ile optik akış takip yöntemi birleştirilerek hedef nesne takip edilmiştir. İlk karede verilen hedef nesneye ait ve nesnenin uzağında arka plana ait ikili örüntüler (binary pattern) öğrenilmiştir. Öğrenme esnasında *random ferns* kullanılmıştır. Yeni video karesi alındığında, sahnede sınıflandırıcı tarafından algılanan 50 yüksek olasılığa sahip bölge seçilmiştir. Optik akış yöntemi, bir önceki kareye ait hedef bölge ile adaylar arasında KLT (Baker ve Matthews, 2004) takibi yaparak en yüksek benzerliğe sahip aday nesne yeni kareye ait hedef nesne olarak tespit edilmiştir. Yeni nesne belirlendikten sonra nesnenin içinden ve yakınından pozitif örnekler alınırken, uzak alanlardan negatif örnekler alınarak sınıflandırıcı güncellenmiştir. Bu yöntem, HBT (Godec vd. 2013), MIT (Babenko vd. 2009) gibi sınıflandırıcı tabanlı diğer yöntemlerden farklı olarak, sadece hedef nesnenin içinden değil etrafından da pozitif örnekler almıştır. Bu sayede, bakış açısı değişimi veya engellemelerden dolayı nesnenin görünümünün değişimine karşı yöntem dayanıklılığı artmıştır.

Struck (Hare vd. 2016) yöntemi, belirlenen hedef çerçeveyi video karesi üzerinde farklı pozisyonlara kaydırarak pozitif örnekler almış ve *structured-SVM* oluşturarak nesneyi öğrenmiştir. MIT yöntemi gibi *Haar* özniteliklerini kullanmıştır. Yeni karede, hedefin önceki çerçevesinin etrafından alınan aday çerçeveler öğrenilen *s-SVM* ile karşılaştırılmıştır. Diğer yöntemlerdeki gibi en yüksek sonucu dönen aday çerçeve, hedefin yeni konumu olarak belirlenmiştir. Yeni çerçeve ile *s-SVM* güncellenirken güncel çerçevenin hala en yüksek sonucu dönmesine dikkat edilmiştir.

DSST (Discriminative Scale Space Tracker), nesne takibinde ölçek değişiminin önemine dikkat çekmiştir (Danelljan vd. 2014). Literatürde yer alan tespit tabanlı takip (tracking-by-detection) yöntemlerinin iyi sonuçlar üretmesinin yanında, ölçek değişimlerine karşı yeteri kadar dayanıklı olmadıkları için, nesne biçimindeki önemli değişimlerde takibi kaçırdıkları belirtilmiştir. CSK (Henriques vd. 2012), MOSSE (Bolme vd. 2010), *Struck* gibi seçici sınıflandırıcı tabanlı yöntemler sadece nesnenin konum değişimi (translation) üzerinden kestirimler yapmıştır. Bu yöntemde konum değişimi ve ölçek değişimini dikkate alan iki farklı seçici korelasyon filtresi (discriminative correlation filter) beraber kullanılmıştır. Tek boyutlu bir filtre ile hedef nesnenin video kareleri arasında ölçek değişimi kestirilirken, iki boyutlu başka bir filtre ile konum değişim kestirimi yapılmıştır. Ölçek kestirimi için nesne sınıflandırıcı ölçek piramidi ile eğitilmiştir. Filtrelerde HOG (Histogram of Oriented Gradients) öznitelikleri kullanılmıştır (Dalal ve Triggs 2005). Bu yöntem nesnenin konum ve ölçek değişimine karşı dayanıklı olmasının yanında gerçek-zamanlı olarak çalışmıştır.

Muster (Hong vd. 2015), psikolojide insan belleğini modellemek için kullanılan *Atkinson-Shiffrin* modelinden esinlenerek geliştirilmiş, nesnelerin uzun-vade ve kısa-vade olmak üzere 2 ayrı modelini saklayan bir takip yöntemidir. Kısa vadede ICF (Integrated Correlation Filter) yardımı ile hedef nesnelerin bir sonraki video karesindeki pozisyonu kestirilmiştir. ICF, nesnenin konum ve ölçek değişim kestirimi yapan iki aşamalı bir filtredir. Durağan sahnelerde iyi sonuç vermektedir. Hedef nesne görünümünün değiştiği durumları kontrol altında tutmak için nesneden çıkarılan SIFT öznitelikleri, kareler arasında *RANSAC* yöntemi ile eşleştirilmiştir. Kısa vade ve uzun vade takip işlemlerinden elde edilen hedef nesneler, birbiri ile karşılaştırılmış ve

aralarında ciddi bir fark yoksa kısa vadeli takip sonucu hedef nesne olarak belirlenmiştir. Kısa vadeli takipten elde edilen hedef nesne bilgileri ile uzun vadeli takip güncellenmiştir. 2 takip yönteminin çıktısı arasındaki uyumsuzluk çok fazla ise nesne görünümünde ciddi bir değişim olduğu kabul edilip uzun vadeli takip sonucu hedef nesne olarak belirlenmiştir. Bu durumda kısa vadeli takibin nesne görünümü değişiminden dolayı takibi kaybettiği düşünülmüştür. Kısa vadeli takipte kullanılan ICF, yeni hedefe göre güncellenmiş ancak uzun vadeli takip güncellenmemiştir. Hedef nesnenin kısa vade çıktısında olmayıp uzun vade çıktısında bulunan bölgeleri, engellenmiş (occluded) olarak etiketlenmiştir. Uzun vadeli takip yöntemi güncellenirken nesnenin belirlenen eşik değerinin altında engellenmiş olmasına bakılmıştır. Hedef nesnenin birden fazla model temsil edilmesi, başarıyı artırırken yöntem gerçek zamanlı çalışmadan uzaklaşmıştır.

SRDCF (Spatially Regularized Discriminative Correlation Filter Tracker), uzamsal olarak düzenlenmiş seçici korelasyon filtresi tabanlı bir nesne takip yöntemidir (Danelljan vd. 2015). Standart seçici korelasyon filtresi tabanlı takip yöntemlerinde, dairesel korelasyon sırasında alınan örnek sınırları öğrenme işleminin başarımını düşürmekte ve sınırlı alanda arama yapılmasına sebep olmaktadır. SRDCF yönteminde, hedef bölge dışında kalan alanlarda filtre katsayılarını cezalandıran uzamsal düzeltme fonksiyonu (spatial regularization) ile dairesel korelasyon sorunu çözülmüştür. Bu sayede, filtre boyutu değişmeden öğrenilen ve tespit edilen örnek sayısı artmakta, dolayısıyla filtreler daha etkili kullanılmaktadır. İki boyutlu görüntü düzlemi üzerinde işlem yapan DSST, CSK gibi DCF tabanlı yöntemler yerine *Fourier* düzleminde işlem yapılmış ve bu sayede yöntem performansının artırıldığı belirtilmiştir. *Gauss-Seidel* tabanlı tekrarlamalı optimizasyon ile korelasyon filtresi hesaplanmıştır. Korelasyon puanı en yüksek örnek, hedef nesne olarak belirlenmiştir. Her piksel, HOG özniteliklerinin yanında renk ve gri-tonlu özniteliklerden oluşan 42 boyutlu bir vektör ile temsil edilmiştir.

Bu bölümde video sinopsis çalışmalarında kullanılmış olan nesne takip yöntemlerinin yanında, video sinopsis çalışmalarında kullanılmamasına rağmen literatürde kabul görmüş öncü yöntemler de incelenmiştir. Nesne takibininin sağlıklı yapılamaması,

nesne izinin kopması ve uzamsal olarak kesişen nesnelerin yanlış eşleştirilmesi gibi durumlara sebep olduğundan dolayı doğrudan video sinopsis başarımını düşürmektedir. Takipteki kopmalardan dolayı aynı nesne için birden fazla aktivite oluşturulması, video sinopsisin anlamsal bütünlüğüne zarar vermektedir. Bu durumda aynı nesnenin farklı aktiviteler olarak aynı anda gösterilme ihtimali ortaya çıkmaktadır. Gereğinden fazla aktivite oluşturulması, aynı zamanda optimizasyon probleminin karmaşıklığını artırmaktadır. Bu yüzden video sinopsise özelleşmiş, gürbüz nesne takip yöntemleri video sinopsis başarımına olumlu katkı yapmaktadır. Video işleme alanında nesne takibinin popüler bir konu olması ve yoğun bir şekilde araştırılıyor olması, bu konudaki çalışmaların başarımı her geçen gün artırmaktadır. Özellikle çevresel koşullara karşı dayanıklı takip yöntemlerinin video sinopsis sisteminde kullanılmasının genel başarımlar üzerinde olumlu etkisi olacaktır.

2.1.3 Çok kameralı nesne takibi

Çok kameralı nesne takibi hedef nesnelerin farklı kamera görüntüleri üzerinde eşleştirilmesini amaçlamaktadır. Genel olarak aşağıdaki 4 temel adımdan oluşmaktadır:

- a. Çok kameralı kalibrasyon: Farklı kameraların koordinat sistemlerinin ortak bir koordinat sistemine taşınmasını amaçlamaktadır. Kameralar arasında hedef nesnenin tekrar tanımlanmasına yardımcı olmaktadır.
- b. Kamera ağ topolojisinin oluşturulması: Kameraların birbirine göre konum haritalarının oluşturulmakta ve bakış açılarının kesişme durumları tanımlanmaktadır. Bu sayede nesnelerin kameralar arasındaki geçiş süreleri gibi kestirimler yapılarak çok kameralı nesne takibi desteklenmektedir.
- c. Hedef nesnelerin farklı kameralarda yeniden tanımlanması (re-identification): Farklı kameralardan elde edilen görüntülerin aynı nesneye ait olup olmadığının belirlenmesidir. Nesneye ait görsel özniteliklerin yanında kalibrasyon ve kamera topolojisi sayesinde sağlanan uzam-zamansal veri ile desteklenmektedir.
- d. Çok kameralı nesne takibi: Nesne görünümünün farklı kameralar arasında eşleştirilerek takip edilmesidir.

Çok kameralı kalibrasyon: Kameraların birbiri ile durumlarına göre çeşitlilik gösteren farklı çok kameralı kalibrasyon yaklaşımları yer almaktadır. Eğer kamera görüntüleri kesişiyor ise aralarındaki homografi hesaplanabilmektedir (Brown ve Lowe 2003). Bu sayede, tüm kamera görüntüleri tek bir koordinat sisteminde temsil edilmektedir.

Kamera görüntülerinin kesişmediği durumda, ortak koordinat sistemi oluşturmak daha zorlu bir problem haline gelmektedir. Kanhere ve Birchfield (2010) tarafından belirtildiği gibi görüntüdeki perspektif geometride paralel çizgilerin kesiştiği ufuk noktası (vanishing point) ve bilinen uzunlukları kullanan farklı yöntemler (VVH-Two vanishing points and known camera height, VVW-Two vanishing points and known width, VWH-One vanishing point, known width and camera height vb.) bulunmaktadır. Bu yöntemler, özellikle trafik akış yönünde konumlandırılmış trafik kameralarında daha yüksek başarımlar göstermektedir. Yollarda yer alan paralel çizgiler ve aynı doğrultuda hareket eden araçlar, yöntemlerin daha güvenilir girdiler elde etmesini sağlamıştır.

Fakat daha kontrolsüz bir sahnede görüntüleme yapan bir kameranın coğrafi kalibrasyonunun bahsedilen yöntemlerle yapılması çok zordur. Bu tarz güvenlik kameralarında coğrafi kalibrasyon gerçekleştirmek için sahneden elde edilen daha fazla ön bilgiye ihtiyaç olmaktadır. Schoepflin ve Dailey (2007) tarafından belirtildiği gibi *vanishing* noktalar dışında kamera pozisyonunun bilinmesiyle veya sahnedeki nesne uzunluklarının bilinmesi ile kamera kalibrasyonu gerçekleştirilmiştir. Aynı çalışmada bahsedilen kalibrasyon yöntemlerinin hassasiyetleri de karşılaştırılmıştır.

Kamera ağ topolojisinin oluşturulması: Kameraların birbirine göre konumlarını ve bakış açılarını bilmek, nesnelerin kameralar arasındaki geçişini kestirmek için yardımcı bilgi olarak kullanılmaktadır (Stauffer 2003). Bir kamera görüntüsünden çıkan nesnenin çıkış yönü dikkate alındığında, hangi kamera görüntülerine hangi bölgelerden girebileceğinin bilinmesi, nesne yeniden tanımlama işlemini kolaylaştırmaktadır. Kamera görüntüsüne yeni giren nesneler, diğer kameralardan çıkıp o bölgeden girme ihtimali olan nesnelerle karşılaştırılmaktadır. Bu yolla yanlış eşleşme ihtimali düşerken eşleştirme işleminin daha hızlı gerçekleştirilmesi sağlanmaktadır. Kameraların görüş alanları arasındaki mesafenin bilinmesi ile takip edilen nesne hızı dikkate alınarak,

komşu kamera görüşüne ne zaman gireceği kestirilmektedir. Bu sayede nesnenin arama alanı hem konum hem de zaman düzleminde mümkün oldukça daraltılarak belirsizliğin minimuma indirgenmesi amaçlanmaktadır.

Hedef nesnelerin farklı kameralarda yeniden tanımlanması (re-identification):

Farklı kamera görüntülerine ait nesne örneklerinin birbiri ile görsel olarak eşleştirilmesidir. Kameralar arasında hareket eden nesnelerin renk, biçim, doku gibi görsel özelliklerinin çok fazla değişmemesi gerektiğinden yola çıkarak farklı eşleştirme yöntemleri uygulanmaktadır (Schwartz ve Davis 2009, Farenzena vd. 2010). Literatürde bu konuda yapılan çalışmalar incelendiğinde nesne görünümünün genellikle renk (colour), biçim (shape) ve doku (texture) olarak 3 sınıfa ayrıldığı görülmüştür. Bu 3 kategoriden birini kullanmak yerine genellikle üçü birlikte kullanılmıştır. Renk karşılaştırması için HSV, Lab, RGB ve Long-RGB gibi farklı renk uzayları kullanılmıştır (Wang vd. 2007). Biçim tanımlaması için yaygın olarak HOG öznitelikleri kullanılmıştır. Doku için ise SIFT, SURF, LBP, MSER gibi özniteliklerin yanında *Gabor* gibi filtreler de kullanılmıştır.

Çok kameralı nesne takibi: Kamera koordinatları ortak bir koordinat sistemine taşındıktan sonra farklı kameralarda takip edilen nesnelerin birbiri ile ilişkilendirilmesidir. *Bayesian* temelli yaklaşımlardan Kalman filtresi (Black vd. 2002), *extended Kalman* filtresi (Straw vd. 2011) ve parçacık filtresi (Kim ve Davis 2006) nesne takibinde yaygın olarak kullanılan yöntemlerdendir. *Bayesian* yaklaşımın öncülerinden biri olan Huang ve Russell (1997) tarafından önerilen yöntemde iki kamera görüntüsündeki araçları eşleştirmek için aracın renk, boyut, hız, görülme zamanı ve hareket ettiği şerit gibi bilgileri içeren bir model oluşturulmuştur. Oluşturulan model sayesinde bir kamerada takip edilen nesnenin diğer kamerada görünüm veya uzam-zamansal özelliklerinin kestirilmesi amaçlanmıştır. Daha sonra, Pasula vd. (1999) bu yaklaşımı iki kamera yerine daha fazla sayıda kamera ağına uygulanacak şekilde geliştirmiştir. Javed vd. (2003) tarafından önerilen *kernel density estimator* yöntemi ile görüntüsü kesişmeyen kameralarda, bir kameradan çıkan nesnenin konum, zaman ve hız bilgileri kullanılarak nesnenin hangi kamera görüntülerine ne zaman ve hangi konumdan gireceğini kestirilmiştir. Farklı kameralarda nesne görünüm değişimlerini

hesaplamak için nesneye ait renk histogramı kullanılmıştır. Nesnenin farklı kameralardaki renk histogram aralığı gaussian dağılımı olarak tanımlanmış ve buna göre nesne yeniden tanımlaması yapılmıştır.

Bae ve Yoon (2014), nesnelerin yerel ve global olarak ilişkilendirildiği iki aşamalı bir takip yöntemi önermiştir. Takip edilen nesneler sahnede o anki nesne algılamaları ile eşleşiyorsa güvenilir takip parçacığı (tracklet) olarak tanımlanıp takipçi, eşleşen algılama ile yerel olarak güncellenmiştir. Takip parçacığı; görünüm, biçim ve hareket özellikleriyle tanımlanmıştır. Takip parçacığının güvenilirliğine uzunluğu, engellenme durumu ve algılamalar ile benzerliğine göre karar verilmiştir. Takip edilen nesne herhangi bir algılama ile eşleştiremiyorsa engelleme ve görüntüden çıkma gibi sebeplerden dolayı takibin kopmuş olabileceği varsayılarak güvenilir olmayan takip parçacığı olarak saklanmıştır. Güvenilir olmayan takip parçacıkları belirli frekansta global olarak diğer takip ve algılamalar ile eşleştirilmeye çalışılmıştır. Tek bir nesnenin parçası olduğuna karar verilen parçacıklar birleştirilmiştir. Yerel ve global ilişkilendirme problemini çözmek için *Hungarian* algoritması kullanılmıştır (Ahuja vd. 1989). Yerel ilişkilendirme yapılması global ilişkilendirmenin karmaşıklığını azaltması açısından performans kazancı sağlamıştır. Çalışma sonuçlarına göre bu yöntem, literatürde kabul görmüş veri kümeleri üzerinde başarılı sonuçlar vermiştir.

Previtali vd. (2017), *Ptracking* adını verdikleri dağıtık parçacık filtresi (distributed particle filter) tabanlı gerçek-zamanlı çalışan bir yöntem önermiştir. Yöntem, kamera özelinde yerel kestirim ve kameralar arası global kestirim aşamalarından oluşmuştur. Tek bir kamera görüntüsünde tespit edilen nesnenin takibi *Bayesian* rekürsif kestirim problemi haline getirilip dağıtık parçacık filtresi ile çözülmüştür. Küme sayısının önceden bilinmemesi ve gerçek-zamanlı çalışma ihtiyacı yüzünden *k-means*, hiyerarşik kümeleme, QT-kümeleme gibi yaygın olarak kullanılan yöntemler yerine yeni bir yaklaşım geliştirilmiştir. Bu yaklaşım, her kümeyi bir Gaussian dağılımı olarak kabul etmiştir. Kümeleme kriteri olarak nesne yönü, hızı ve pozisyonu kullanılmıştır. Ayrıca sahnede iki nesnenin birbirine çok yaklaştığı durumlarda HSV renk histogramları kullanılarak ayrıştırılmıştır.

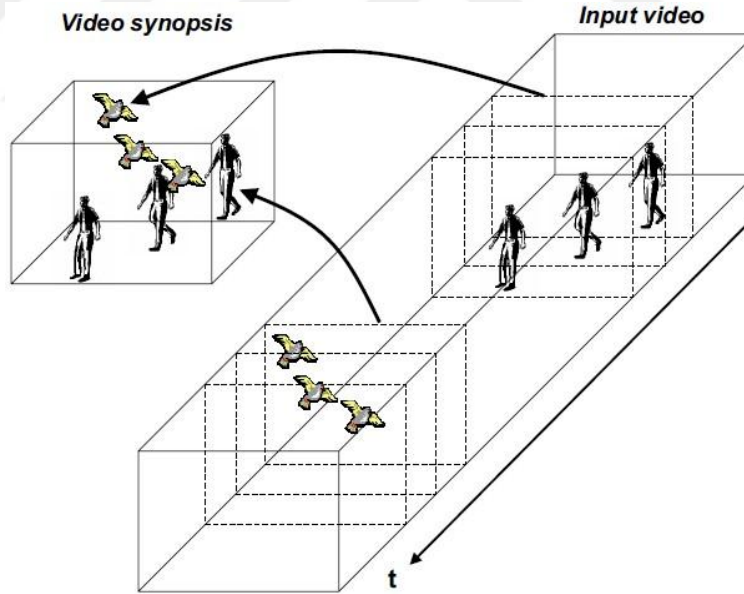
Munaro vd. (2016), derinlik kameraları üzerinde çok kameralı nesne takibi yapan bir kütüphane oluşturmıştır. Kamera kalibrasyonunda öncelikle her kamera, kalibrasyon tahtası ile kalibre edilerek dahili (intrinsic) parametreleri kestirilmiştir. Sonrasında kalibrasyon tahtası iki kamera görüntüsünün kesiştiği noktaya taşınarak kameraların birbirine göre konumları kestirilmiştir. Her karede algılanan insan konumu ile o an takip edilen nesneler arasındaki *Mahalanobis* uzaklığı dikkate alınarak eşleştirilmiştir. Kalman filtresi yardımı ile nesnelerin konumları kestirilmiş ve hızları hesaplanmıştır. Deneyler kapalı ortamda birbiri ile kesişen kamera görüntüleri üzerinde yapılmıştır. Çoklu kamera kalibrasyonunda kalibrasyon tahtasının kameralardan uzak bir konuma yerleştirilmesinin başarımı düşürdüğü belirtilmiştir. Çalışma sonunda oluşturulan takip yöntemi ile farklı derinlik kameralarının performansı karşılaştırılmıştır.

Mazzon ve Cavallaro (2013) tarafından önerilen *Social Force* modeli ile sahnenin fiziksel durumu dikkate alınarak insanların hareketleri kestirmeye çalışılmıştır. Sahnedeki engel, geçenlerin dikkatini çekecek mağaza vitrini gibi hedef nesne hareketini etkileyecek/geciktirecek tüm unsurlar tanımlanmıştır. Sahne özellikleri ile nesnenin görüldüğü süre boyunca sahip olduğu hız bilgileri kullanarak kamera görüntüleri dışındaki hareket kestirimi yapılmaya çalışılmıştır. Fakat kamera görüntüleri arasında nesne eşleştirme işleminde görsel bir öznitelik kullanılmadan sadece nesne izi kullanılmıştır. Bu da, yeniden tanımlama başarımının düşmesine sebep olmuştur.

Özellikle görüntüsü kesişmeyen kamera görüntüleri üzerinde çok kameralı nesne takibi, hala yoğun bir şekilde araştırma yapılan zorlayıcı bir problemdir. Kameralar arasında nesnelerin başarılı bir şekilde eşleştirilmesinin önünde bir çok engel bulunmaktadır. Kameralar arasındaki mesafenin fazla olması hedef nesnelerin görüntüde olmadığı zamandaki hareketinin kestirimini zorlaştırmaktadır. Bunun dışında kameraların nesne hareketine göre bakış açılarının farklı olması eşleştirme işlemini zorlaştırmaktadır. Nesneyi arkadan gören bir kameradan sonra nesneyi önden gören başka bir kamerada eşleştirme yapmak oldukça zorlu bir problemdir. Ayrıca eşleştirme için kullanılan görsel özniteliklerin kameraların görsel kalitesindeki farklılıklar yüzünden olumsuz etkilenme ihtimali de bulunmaktadır. Bu sebeplerden dolayı çok kameralı nesne takibi hala genel-geçer bir çözüm aranan güncel bir araştırma alanıdır.

2.1.4 Aktivite optimizasyonu

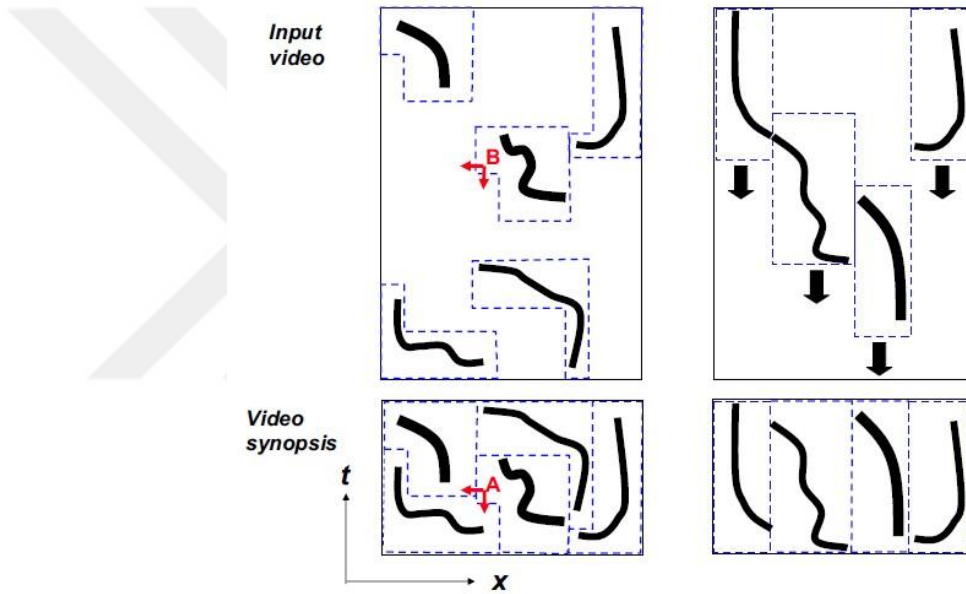
Video sinopsis, mümkün oldukça fazla aktiviteyi en kısa sürede eş zamanlı olarak göstermesini sağlayan aktivite tabanlı bir yöntemdir. Video sinopsisin temel amacı, girdi görüntüden çıkarılan her bir aktivitenin zaman düzleminde kaydırılarak, diğerleri ile minimum kesişime sahip toplamda en kısa video sinopsisi oluşturacak şekilde optimal pozisyonuna yerleştirilmesidir. Video karesi tabanlı video özetleme yöntemlerinde video, her bir kare dikkate alınarak analiz edilmektedir. Videodan ilgilenilmeyen kareler çıkarılarak kalan kareler özet video şeklinde görüntülenmektedir. Video sinopsiste ise özetlemede aktiviteler dikkate alındığı için her bir kare içindeki ilgilenilmeyen alanlar da özet yoğunluğunu artırmak adına değerlendirilmektedir. Farklı zamanlara ait aktiviteler, video sinopsis üzerinde aynı anda gösterilebilmektedir. Bu sayede video karesi tabanlı özetleme yöntemlerine göre çok daha yüksek yoğunluk artırma performansına sahip olmaktadır (Şekil 2.1).



Şekil 2.1 Farklı zamana ait aktivitelerin video sinopsis üzerinde eş zamanlı gösterimi (Rav-Acha vd. 2006)

Video sinopsis ilk olarak Rav-Acha vd. (2006) tarafından önerilmiştir. Önerdikleri yöntem çevrimiçi ve çevrimdışı olmak üzere iki aşamadan oluşmaktadır. Aktivite oluşturma ve tamamlanan aktiviteleri bir kuyruğa doldurma işlemi çevrimiçi olarak

gerçekleştirilmiştir. Sonrasında belirlenen bir zaman aralığına ait aktivitelerin yeniden düzenlenmesi (rearrangement), arka plan görüntüsü oluşturma ve aktivitelerin arka plan ile harmanlaması çevrimdışı olarak gerçekleştirilmiştir. Aktivite yoğunluğu, zamansal tutarlılık (temporal consistency) ve aktivite kesişim masraflarını hesaplayan bir global enerji fonksiyonu tanımlanmıştır. Sonrasında *Simulated Annealing* (Kirkpatrick vd. 1983) optimizasyon yöntemi kullanılarak enerji minimizasyonu gerçekleştirilmiştir. Tanımlanan enerji fonksiyonu, aktivite tekrar düzenlenmesinde istenmeyen durumları cezalandırmaktadır. Dolayısıyla fonksiyonun minimum enerjiye sahip olduğu durum, video sinopsis için optimal aktivite dağılımını temsil etmektedir (Şekil 2.2).



Şekil 2.2 Aktivitelerin yeniden düzenlenmesi (Rav-Acha vd. 2006)

Bu yöntem Pritch vd. (2007) tarafından, sürekli yayın yapan kamera görüntülerine uygulanarak geliştirilmiştir. Bu çalışmada aktiviteleri tanımlamak için *tube* terimi kullanılmış ve literatürde yaygın olarak kabul görmüştür. Daha iyi bir nesne algılama yöntemi kullanarak video sinopsis başarımını artırmayı hedeflemelerinin yanında yeni metrikler ile tanımlanan enerji fonksiyonu geliştirilmiştir.

Yukarıda bahsedilen iki çalışma video sinopsisin uygulama aşamasından çok, teorik temellerinin oluşturulmasına önem vermiştir. Yazarlar bu iki çalışmayı birleştirip genişleterek önerilen yöntemin çalışma zamanını da analiz etmiştir (Pritch vd. 2008).

Optimizasyon işlemini hızlandırmak için aktivitelerin kaydırılması 10 video karesi aralıklarla yapılmıştır. Aynı şekilde, nesne algılama için kullanılan hareket algılama yöntemi de her 10 video karesinden birine uygulanmıştır. Bunların yanında, kaynak görüntü çözünürlüğünü düşürerek performans kazancı sağlanmaya çalışılmıştır. Önerilen iyileştirmeler sonucunda yöntem, hala gerçek-zamanlı çalışma performansından uzak olsa da video sinopsisin gerçek zamanlı uygulanması konusundaki sorunları analiz eden ilk çalışma olması açısından önemlidir.

Yazarların sonraki çalışması, algılanan nesnelerin sınıflandırılarak benzer nesnelerin bir arada gösterilmesini amaçlamıştır (Pritch vd. 2009). Görünüm ve hareket öznelikleri kullanarak yapılan sınıflandırmada, benzer kişilerin aktivitelerinin beraber gösterilmesi veya aynı yöne hareket eden nesnelerin beraber gösterilmesi gibi seçenekler sunulmuştur. Bu sayede, video sinopsisi izleyen kullanıcının daha kolay takip etmesi amaçlanmıştır. Yazarlar, zaman düzleminde fazla yer kaplayan aktivitelerin kesişen başka aktivitelerin beraber gösterimini engellemesinden dolayı optimizasyon başarımını düşürdüğünü vurgulamıştır. Bu yüzden uzun aktiviteler, en fazla 50 video karesinden oluşan aktivite parçacıklarına (tubelet) ayrılarak zaman düzleminde kaydırılmıştır. Bu işlem, her ne kadar oluşturulan özetin daha yoğun olmasını sağlasa da aynı nesneye ait aktivite parçacıklarının farklı zamanlarda gösterimi olayların tutarlılığını bozmaktadır. İzleyen kişinin sahnedeki olaylar hakkında çıkarım yapması zorlaşmaktadır. Yukarıda bahsedilen çalışmalar, video sinopsis yaklaşımı ortaya atan yazarların çalışmalarıdır. Her ne kadar, yoğun aktivite barındıran videolarda çok zaman alan çevrimdışı optimizasyon, büyük boyutlarda bellek ihtiyacı, video sinopsis uzunluğunun kullanıcıya bırakılarak önemli olabilecek aktiviteleri kaçırma ihtimali gibi sınırlamaları olsa da bu çalışmalar, video sinopsis yaklaşımının olgunlaşması ve sonraki çalışmalara öncülük etmesi açısından önemlidir. Bu çalışmalar sonucunda oluşturulan video sinopsis metodolojisi, literatürde yaygın olarak kabul görmüştür.

Yukarıda belirtilen yazarlardan farklı olarak video sinopsis konusunda yapılan öncü çalışmalardan biri (Xu vd. 2008) tarafından önerilmiştir. Nesne algılama yöntemi olarak piksel değerlerinin Gaussian dağılımını dikkate alan hareket algılama yöntemi kullanılmıştır. Bu yolla algılanan aktivitelerin tamamı genel aktivite kümesi olarak

tanımlanıp hedef videoda yer alacak aktiviteler, bu küme içinden *mean shift* optimizasyon yöntemi ile yerleştirilmiştir. Optimizasyon aşamasında şu ana kadar bahsedilen yöntemlerden en önemli farkı, aktiviteler arasındaki kronolojik sıralamayı dikkate almamasıdır. Çalışmada elde edilen sonuçların Pritch vd. (2007) tarafından elde edilen sonuçlar ile karşılaştırıldığında daha yüksek başarıma sahip olduğu belirtilmiştir. Fakat literatüre bakıldığında karşılaştırılan çalışma kadar ilgi görmediği gözlemlenmektedir. Bu durumun muhtemel sebebi, kullanılan optimizasyon yönteminin yerel optimumu hedeflemesi sonucundan başarımının sınırlı koşullar altında Pritch vd. (2007) tarafından önerilen yöntemden daha iyi sonuç vermesidir.

Yildiz vd. (2008), aktivite algılama için nesne tabanlı yerine piksel tabanlı bir yaklaşım önermiştir. Düşük enerjili dikey yollar çıkartılarak video kareleri iteratif olarak küçültülmüştür. Bu sayede her bir karede yer alan aktivitelerin elde edilmesi amaçlanmıştır. Genel yaklaşımın aksine nesneleri zaman düzleminde kaydırmak yerine karelerdeki pasif alanlar çıkarılarak video özetlenmiştir. Önerilen yöntem, boru-hattı mimarisine sahip bir sistem üzerinde uygulanarak gerçek-zamanlı ve düşük bellek ihtiyacı ile çalıştırılmıştır. Sonrasında yazarlar bu çalışmayı, kullanıcının göz hareketlerini takip eden bir sistem ile entegre ederek genişletmiştir (Vural ve Akgül 2009). Kamera görüntülerini izleyen güvenlik görevlilerinin göz hareketlerinin takibi ile aktiviteler, özellikle dikkat edilen veya gözden kaçan aktiviteler olarak sınıflandırılmıştır. Video sinopsis üretilirken bu aktivite gruplarından biri kullanıcıya sunulmuştur. Nesne sınırlarını dikkate almadan yapılan piksel tabanlı analiz, video sinopsiste nesne bütünlüklerinin bozulmasına sebep olmuştur. Özellikle yoğun aktivite barındıran görüntülerde nesne-tabanlı analiz yapan yöntemlere göre video sinopsisin görsel kalitesi daha düşük olmuştur.

Rodriguez (2010) hareket algılama için kullandığı optik akış yöntemi ile kameranın sabit olması kısıtlamasını ortadan kaldırmış ve hava araçlarından alınan görüntüler üzerinde video sinopsis oluşturan bir yöntem geliştirmiştir. Diğer yöntemlerden farklı olarak algılanan aktiviteyi frekans düzlemine aktarıp, oluşturulmuş aktivite şablonlarıyla karşılaştırarak sınıflandırılmış aktivitelerin video sinopsisini oluşturmuştur. Chou vd. (2015) benzer aktiviteleri sınıflandırarak video sinopsis

oluşturan başka bir yöntem önermiştir. Sahnedeki muhtemel giriş ve çıkış bölgesi olarak dört alan belirlenmiştir. Aktiviteler sahneye giriş ve çıkış bölgelerine göre sınıflandırılmıştır. Önerilen sınıflandırma yönteminde farklı örnek frekansına, hıza ve boyuta sahip olsalar dahi benzer nesneler algılanıp aynı sınıfa dahil edilebilmiştir. Lin vd. (2015) nesnelerin sahneye giriş ve çıkış bölgelerine göre sınıflandırma yapan başka bir yöntem önermiştir. Önerilen öğrenme tabanlı anomali algılama yöntemi ile tespit edilen nesneler ilgili bölgelere göre sınıflandırılmıştır. Bu yöntemlerin ortak amacı - farklı sınıflandırma kriterleri kullanılsa da- benzer aktiviteleri video sinopsiste beraber gösterip izlenilebilirliği kolaylaştırmaktır. Video sinopsis metodolojisine yeni bir adım (algılanan aktivitelerin sınıflandırılması) eklemelerinin yanında optimizasyon aşamasına sınıflandırma metriği ekleyerek de literatüre katkı sağlamışlardır.

Fu vd. (2014), aktiviteler arasındaki etkileşimi de dikkate alan bir optimizasyon yöntemi önermiştir. Bunun için enerji fonksiyonuna hareket yapısı (motion structure) metriği eklenmiştir. Bu metrik, kaynak videoda etkileşim halinde bulunan aktivitelerin video sinopsiste de beraber gösterilmesini zorlamaktadır. Aktivite etkileşimlerinin korunması video sinopsisin anlamsal bütünlüğü desteklerken, etkileşen aktivitelerin zaman düzleminde beraber kaydırılması optimizasyon verimini düşürerek özet videoda daha fazla boşlukların olmasına sebep olmaktadır.

Kasamwattananarote vd. (2010), aktiviteleri *tunnel* olarak tanımlayıp gerçek-zamanlı aktivite algılama adımı üzerinde yoğunlaşmıştır. Aktivitelerin birbiri ile kesişim durumları, video analiz aşamasında tespit edilerek kayıtlı videonun içine gömülmüştür. Videoya gömülü kesişim bilgilerini çıkarmak için geliştirilen video görüntüleme yöntemi ile bu bilgiler kullanılarak aktivite kaydırma işlemi yapılmıştır. Çalışmada, gerçek-zamanlı bir yöntem önerildiği belirtilse de bir çok çevrimdışı optimizasyon yönteminde olduğu gibi sadece aktivite algılama yöntemi gerçek-zamanlı olarak gerçekleştirilmiştir.

Literatürdeki çalışmaların büyük bir çoğunluğunda aktiviteler zaman düzleminde kaydırılarak optimizasyon işlemi gerçekleştirilmiştir. Nie vd. (2013) aktiviteleri hem zaman hem de konum düzleminde kaydırarak kesişimleri minimuma indirmeyi

amaçlamıştır. Konum düzleminde kaydırma yapabilmek için şekil 2.3'te görüldüğü gibi aktivitenin arka alanını genişleterek yapay bir görüntü oluşturmuştur. Nesnelerin konum düzleminde kaydırılması nesne kesişimlerini azaltmakta, dolayısıyla sinopsis verimini artırmaktadır. Fakat yapay bir sahne oluşturulması olayların anlamlandırılmasını zorlaştırabilmektedir. Ayrıca büyük bir çoğunluğunda hareket olan sahnelerde önerildiği gibi konum genişletmesi yapacak uygun boş alan olmayabilir. Bu durum yöntemin sahne özelliklerine bağımlı hale gelmesine sebep olmaktadır.



Şekil 2.3 Kesişen nesnelerin konum düzleminde kaydırılması (Nie vd. 2013)

Li vd. (2016), kesişim probleminin çözümüne farklı bir yaklaşım getirerek şekil 2.4'te görüldüğü gibi nesnelerin boyutlarını küçültmüştür. Enerji fonksiyonuna her nesnenin ölçek değişimini temsil eden bir metrik eklenmiştir. Video sinopsiste nesnelerin ölçeklerinin düşürülerek gösterilmesi geleneksel yöntemlerle karşılaştırıldığında kesişimi ciddi anlamda azaltmıştır. Fakat ölçek değişimi belirli sınırlar içinde tutulsa dahi görsel kaliteyi düşüren yapay sahneler ortaya çıkmıştır. Şekil 2.4'te görüldüğü gibi sahnede yan yana duran insanın araçtan daha büyük olduğu durumlar ortaya çıkmıştır.



Şekil 2.4 Uzamsal kesişimi engellemek için nesnelerin boyutlarının küçültülmesi (Li vd. 2016)

Nesne kesişimi üzerine yoğunlaşan son çalışmalarda He vd. (2017a,b) nesne durumunu; kesişmeyen, aynı yönde kesişen ve ters yönde kesişen şeklinde tanımlayarak kesişimleri bunlara göre sınıflandırmıştır. Optimizasyon problemini kesişim durumlarını dikkate alan çizge tabanlı bir yöntem ile çözmeyi amaçlamıştır. Optimizasyon işlemini kesişim durumunu merkeze alarak gerçekleştirdiği için kesişim problemine getirilen en kapsamlı çözüm durumundadır. Fakat diğer metrikleri (aktivite, kronolojik tutarlılık, vs.) dikkate almadığı için yöntemin nihai sonucunun hala iyileştirmeye ihtiyacı vardır.

Huang vd. (2014), video sinopsisin gerçek-zamanlı olarak oluşturulabilmesi için çevrimiçi optimizasyon tekniklerinin önemine vurgu yapmıştır. Önerilen yaklaşımda görüntüdeki her piksele karşılık bir elemanı bulunan bir sinopsis tablosu tanımlanmıştır. Tabloda her pikselde bulunan aktivite kimliği ve bulunduğu video kare numarası yer almaktadır. Video akarken tespit edilen yeni aktivitenin, var olan bir aktivitenin devamı veya yeni bir aktivite olmasına göre tablo güncellenmiştir. Önerilen yöntem gerçek-zamanlı olarak çalışmaktadır. Fakat optimizasyon aşamasında işlemin gerçek-zamandan uzaklaşmasının en büyük nedeni olan uzamsal nesne kesişimlerinin tamamen ihmal edilmesi, bu yaklaşımın en büyük problemidir. Ayrıca sahneden bağımsız karar mekanizması yerine kullanıcı tarafından belirlenen eşik değerlerinin kullanılması yöntemin başarımını kısıtlamaktadır.

Zhu vd. (2014), video izleme uygulamalarının günlük hayatta kullanım senaryolarına dikkat çekerek, tek kamera görüntüsünde video sinopsis oluşturmanın yetersiz kaldığına vurgu yapmıştır. Olayların genellikle birden fazla kamera görüntüsüne dahil olmasından dolayı bakış açıları kesişen birden fazla kamera görüntüsü üzerinde algılanan nesnelerin, kameralar arasında birbiriyle ilişkilendirilmesi sonucu video sinopsis

oluşturan bir yöntem önerilmiştir. Problem, kameralar arasında takip edilen nesnelerin izlerinin eşleştirilmesi ve video sinopsis olarak ikiye ayrılmıştır. Kesişen iki kameranın her birinde elde edilen nesne izleri, kameralar arası ilişkiden hesaplanan homografi matrisleri kullanılarak iki kameranın mozaiklenmesi sonucu oluşan sanal düzleme (panoramik görüntü) aktarılmıştır. Sonrasında bu izlerin her biri, oluşturulan çizgede bir düğüm olarak gösterilip *quadratic program* tabanlı çizge eşleştirme yöntemi ile kameralar arasında eşleştirilmiştir. Bu yöntem ile kesişim alanlarında yakın zamanlarda bulunan izlerden yola çıkılarak en uygun eşleştirilme yapılmaya çalışılmıştır. Çalışmanın video sinopsis kısmında, aktivitenin yer aldığı ardışık video karelerinin genellikle birbirini tekrar eden gereksiz içeriğe sahip olduğu belirtilmiştir. Bu tekrarın oluşturulacak sinopsis videosunu gereksiz uzatmasından dolayı aktiviteye ait video karelerini sadece hareketin veya görünümün değiştiği anahtar (key) kareleri seçerek kısaltmak hedeflenmiştir. Anahtar karelerin girdi görüntü yerine belirlenmiş aktiviteler üzerinden seçilmesi, önceki yöntemlere göre daha etkili sonuçlar elde edilmesine sebep olmuştur.

Hoshen ve Peleg (2015) tarafından önerilen yöntem ile sistemde yer alan kameralar merkez ve yardımcı kameralar olarak ilişkilendirilmiştir. Merkez kamerada bir aktivite tespit edildiğinde yardımcı kameraların yakın geçmişinde algılanan aktivitelerin gerçek zamanlı olarak video sinopsisi oluşturulmuştur. Kameralar arasında herhangi bir nesne eşleştirme işlemi yapılmadan sadece aynı zamana ait aktiviteler gösterilmiştir. Buradaki amaç herhangi bir olay anında operatöre yardımcı kameralar üzerindeki özetleri de sunarak olayla ilgili karar verme sürecine yardımcı olmaktır.

Mahapatra vd. (2016) tarafından önerilen video sinopsis yöntemi görüntüsü kesişen birden fazla kameraya uygulanmıştır. Kamera görüntülerinin kesişimlerinden faydalanılarak homografi hesabı ile ortak düzlem oluşturulmuştur. Aktiviteler, ortak düzlemin kuşbakışı görüntüsü üzerinde bulundukları konumda işaretlenirken, aynı anda görüldükleri kameralara ait görüntüleri de gösterilmektedir. Bu yaklaşımda aynı zamanda aktiviteler; yürüme, koşma, eğilme, zıplama, el çırpma, tek el sallama ve çift el sallama olarak 7 kategoriye ayrılarak benzer aktivitelerin beraber gösterilmesi sağlanmıştır.

Zhu vd. (2016a), aktiviteyi kritik anların seçimi ile kısaltan çok kameralı video sinopsis yaklaşımı önermiştir. Nesnenin ilk görüldüğü, başka bir nesne ile kesiştiği, tekrar ayrıldığı ve gözden kaybolduğu durumlar kritik an olarak belirlenmiştir. Diğer çok kameralı video sinopsis yöntemlerinden farklı olarak nesneler kameralar arasında görsel özniteliklerine göre eşleştirilmiştir. Optimizasyon aşamasında kullanılan kronolojik tutarlılık metriği, sadece tek bir kamerada değil nesnelerin görüldüğü tüm kameralarda uygulanacak şekilde enerji fonksiyonu çok kameralı sisteme uygun hale getirilmiştir.

Çok kameralı video sinopsis yaklaşımları nesnelerin kamera görüntüleri arasında eşleştirilmesi ihtiyacından dolayı tek kameralı yaklaşımlara göre daha zorlu bir problemi temsil etmektedir. Buna rağmen güvenlik kameralarının dağıtık yapısı düşünüldüğünde günlük hayatta yaygın olarak kullanılan video gözetleme sistemlerine istenilen veriye etkili bir şekilde erişim anlamında daha iyi bir çözüm sunmaktadır. Çok kameralı yaklaşımları da kendi içinde değerlendirmek gerekirse kamera görüntülerinin kesişmesi şartını aramayan yaklaşımlar daha genel uygulama alanına sahiptir.

Lin vd. (2017), çalışma zamanını iyileştirmeye odaklanarak video sinopsise ait her bir adımın dağıtık olarak gerçekleştirilmesini önermiştir. Buna göre video sinopsis metodolojisinde yer alan her bir adımın ayrı bir bilgisayar üzerinde gerçekleştirildiği bir işlem havuzu oluşturulmuştur. Her bir işlem sonucunu, bir sonraki işleme ait bilgisayar ile paylaşarak tüm akış dağıtık olarak gerçekleştirilmiştir. Mimarinin dışında performans kazancı sağlamak için görüntü boyutunun küçültülmesi, saniye başına düşen kare sayısının azaltılması gibi çözümler de önerilmiştir. Bu çalışma ilk defa video sinopsise özgü bir mimari önerilmesi açısından önemlidir.

Yukarıda bahsedilen yaklaşımların tamamı girdi olarak piksel matrisi alıp işlemektedir. Video sinopsisin çalışma zamanı problemine çözüm bulmaya çalışan bir grup çalışma, kameralardan alınan sıkıştırılmış verinin sadece ihtiyaç duyulan kısımlarını çözerek işlem yapmayı önermiştir (Wang vd. 2013a,b; Zhong vd. 2014). Bu yaklaşımlar, sıkıştırılmış verinin tamamen çözülmemesinden sağlanan performans kazancı ile yöntemi gerçek-zamana yaklaştırmayı amaçlamıştır. Her ne kadar çalışma zamanı olarak kazanç sağlasalar da sıkıştırılmış veri üzerinde yapılan işlemler pikseller üzerinde

yapılan işlemler kadar başarılı sonuç vermediği için bu yöntemlerin başarımları daha düşük kalmıştır.

Son olarak Zhu vd. (2013, 2016b) görsel olmayan verinin video sinopsis içinde kullanımı üzerine çalışmıştır. Hava tahmini, trafik bilgisi ve planlanmış halka açık etkinlikler gibi bilgiler görsel veri ile ilişkilendirilmiştir. Bu sayede, aktiviteler sınıflandırılarak daha etkili video içerik analizi gerçekleştirilmiştir. Bu çalışmalarda video sinopsisin metodolojik akışı ile ilgili herhangi katkı sağlanmamıştır. Bu yönden video sinopsisten çok veri toplama ve ilişkilendirme başlıkları altında incelenmektedir. Yine de birden fazla kaynaktan elde edilen verinin video sinopsis içinde kullanımı açısından yenilikçi bir yaklaşım getirmiştir.

Bu bölümde literatürde mevcut video sinopsis çalışmaları avantaj ve dezavantajlarıyla birlikte özetlenmiştir. Mevcut çalışmaların her bir adımında hangi yöntemleri kullandığı EK 1 çizelge 1’de listelenmiştir. Görüldüğü gibi video sinopsis farklı açılardan araştırılıp sorunlara çözüm üretilmiştir. Bazı çalışmalar yöntemlerin gerçek hayatta uygulanabilirliğini artırmak için çok kameralı mimariye taşımaya odaklanırken bazıları kamera görüntülerini gerçek-zamanlı işleyebilmek için çalışma zamanını iyileştirmeye odaklanmıştır. Performans iyileştirmesini optimizasyon yöntemini çevrimiçi olarak gerçekleştirme yoluyla gerçekleştirmeye çalışan yaklaşımların yanında, girdi olarak sıkıştırılmış veriyi işleyerek kazanç sağlamaya çalışan yaklaşımlar önerilmiştir. Diğer tarafta video sinopsis metodolojisindeki her bir adımı dağıtık gerçekleştirerek gerçek zamanlı işlemi yakalamaya çalışan yaklaşımlar bulunmaktadır. Bu anlamda literatür analiz edildiğinde hala başarımlar ve çalışma zamanı performansını dengeleyen optimal çözüm aranmaktadır.

2.1.5 Arka plan oluşturma

Aktivitelerin en uygun dizilimi belirlendikten sonra sahne değişimini en iyi şekilde temsil eden, aktivitelerle uyumlu arka plan görüntüsünün oluşturulması gerekmektedir. Video sinopsis sonucunda farklı zaman periyotlarından gelen aktiviteler, aynı anda gösterilebileceği için arka plan görüntüsünün kalitesi görsel kaliteyi artırmaktadır.

Aktivitelere ait oldukları zamana ait ortalama ve ortanca video karelerinin oluşturulması en kolay ve yaygın olarak kullanılan yöntemlerdir. Video sinopsis literatüründe arka plan oluşturma üzerine yapılan tek çalışma OPBS (Online principal background selection) adı verilen yöntemi önermiştir (Feng vd. 2010). Sahnedeki her piksele ait aktivite yoğunluğunu tutan bir histogram kullanılarak daha yoğun aktivite barındıran arka plan karelerinin seçilmesi sağlanmaktadır. Bu yöntem kullanılarak oluşturulan sinopsis videoları, ortalama ve ortanca kare yöntemine göre kullanıcıya daha doğal bir görüntü izlenimi vermektedir.

Bunun dışında Nie vd. (2013) tarafından önerilen MPR (Multilevel patch relocation) yöntemi yazarlar tarafından önerilen video sinopsis yaklaşımına özgü bir yöntemdir. Literatür özetinde bahsedildiği gibi bu çalışma nesneleri zaman düzleminin yanında konum düzleminde de kaydırmıştır. Konum düzleminde kaydırılırken nesneye ait arka plan genişletilmiştir. Dolayısıyla MPR diğer video sinopsis çalışmalarında kullanılmak için uygun değildir.

Arka plan oluşturma adımının video sinopsisin aktivite yoğunluğunu üzerinde herhangi bir etkisi yoktur. Oluşturulan videonun görsel kalitesinin yükseltilmesi ve kullanıcı tarafından aktivitelerin daha kolay takip edilmesini sağlamaktadır. Bu yüzden literatürde bu konu üzerine çok fazla yoğunlaşmamakta ve işlem sonrası görsel iyileştirme olarak görülmektedir.

2.1.6 Harmanlama

Video sinopsisin son adımı olan harmanlama, gösterilecek nesnelerin oluşturulan arka plan ile uyumsuzluğunu en aza indirecek şekilde birleştirilmesini amaçlamaktadır. Bu aşamaya gelindiğinde aktivitelerin en uygun pozisyonları belirlenmiş olduğu için video sinopsisin yoğunluk azaltma başarımı üzerinde herhangi bir etkisi bulunmamaktadır. EK 1 çizelge 1’de görüldüğü gibi çalışmaların çoğunluğu herhangi bir harmanlama yöntemi kullanmadan nesneleri arka plan görüntüsü üzerinde uygun konuma yerleştirmiştir. Nesnenin kaynak videoda bulunduğu zamana ait ışıklandırma, kamera konfigürasyonundan kaynaklı görüntü değişimleri ve sahne değişimleri gibi

durumlardan dolayı video sinopsiste yerleştirileceği arka plan görüntüsü ile uyumsuzluklar olabilmektedir. Bu yüzden, harmanlama yönteminin kullanılması farklı zamanlardan gelen ve aynı kare üzerinde gösterilen nesnelerin arka plan ile uyumsuzluğunu en aza indirerek görsel kaliteyi artırmaktadır.

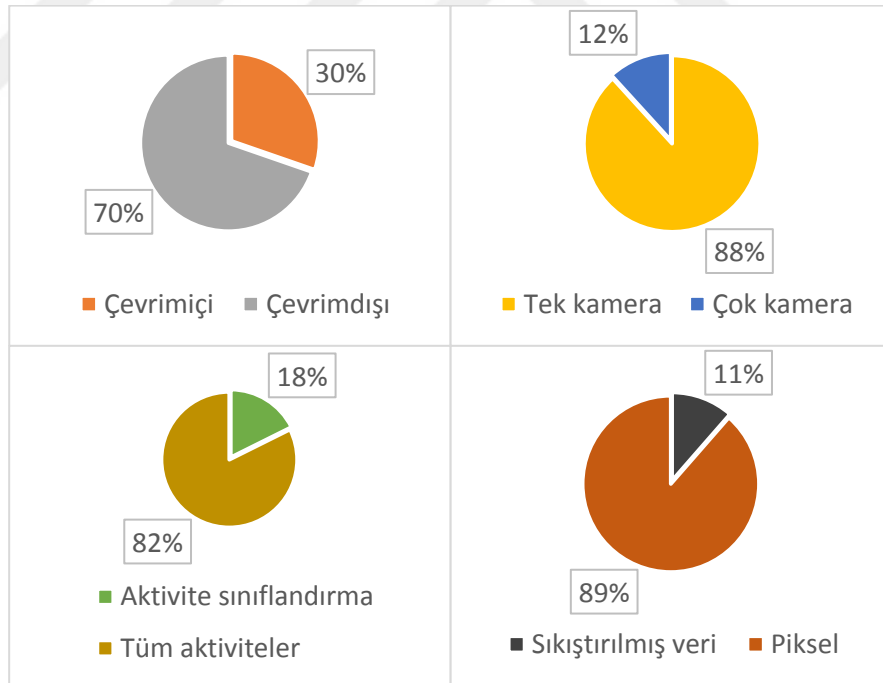
Harmanlama yöntemi kullanan çalışmaların büyük çoğunluğu bu alanda kabul görmüş olan *poisson image editing* (Pérez vd. 2003) yöntemini kullanmaktadır. Bunun dışında *JIT renderer* (Kasamwattananarote vd. 2010) ve nesne çerçevesinde kalan piksellerin arka plan ile renk ortancalarının alınması bu alanda kullanılan diğer yöntemlerdir. Harmanlama, işlem sonrası görsel iyileştirme adımı olarak görüldüğü için video sinopsise özelleşmiş bir yöntem geliştirmek yerine literatürde kabul görmüş mevcut yöntemler kullanılmıştır.

2.2 Literatür Değerlendirmesi

Bu bölümde, literatürde yer alan video sinopsis çalışmaları optimizasyon tipi (çevrimiçi/çevrimdışı), kamera topolojisi (tek/birden çok kamera), girdi veri tipi (piksel/sıkıştırılmış) ve aktivite sınıflandırma (sınıflandırılmış/tüm aktiviteler) şekline göre dört kategoriye ayrılmıştır. Bu doğrultuda, yapılan çalışmaların yıllara göre dağılımı şekil 2.5'te ve çalışmaların bu kategorilere göre oranları şekil 2.6'da gösterilmiştir.



Şekil 2.5 Video sinopsis çalışmalarının yıllara göre dağılımı (2017 yılı itibariyle)



Şekil 2.6 Video sinopsis çalışma sayılarının optimizasyon tipi, kamera topolojisi, aktivite sınıflandırma ve girdi veri tipine göre oranları (2017 yılı itibariyle)

Çevrimdışı optimizasyon uygulayan yöntemler, çevrimiçi yöntemlere göre baskın şekilde daha fazla araştırılmıştır. Çevrimiçi yöntemler, video sinopsis yaklaşımının ortaya atılmasından sonra ilk yıllardan itibaren araştırılsa da her dönem azınlıkta kalmıştır. Benzer şekilde tek kamera görüntüsü üzerinde çalışan yaklaşımlar, çok kameralı yaklaşımlara göre daha popüler olmuştur. Video sinopsisin ilk olarak 2006 yılında ortaya atılmasına rağmen ilk çok kameralı video sinopsis yaklaşımı 2014 yılında önerilmiştir. Direk olarak sıkıştırılmış veri üzerinde 2013, 2014 ve 2017 yıllarında bir kaç video sinopsis çalışması yayınlanmış olmakla beraber bunların toplam video sinopsis çalışmaları içindeki payı oldukça küçüktür. Aktiviteleri belirli kriterlere göre sınıflandırıp gösteren çalışmaların dağılımı incelendiğinde bu konuda da istikrarlı bir grafik görülmemiştir. Ağırlık, aktivitelerin herhangi bir sınıflandırma yapmadan rastgele gösterildiği çalışmalar üzerinedir.

Genel olarak sıkıştırılmış girdi verisi ve aktivite sınıflandırma gruplarında istikrarlı bir dağılım görülmezken çevrimiçi optimizasyon yapan ve çok kameralı sistemlere uygulanan video sinopsis yaklaşımlarında son yıllarda artış görülmektedir. Bu durum, video sinopsis çalışmalarının gelecekte ne yöne doğru genişleyeceği konusunda ipucu sağlamaktadır. Bu bölümün alt başlıklarında bahsedilen dört kategoriye göre video sinopsis çalışmaları analiz edilmiştir.

2.2.1 Aktivite optimizasyon tipi

Aktivitelerin en kısa sürede gösterilecek şekilde zaman düzlemindeki en uygun konumlarının bulunmasını amaçlayan optimizasyon işlemi, video sinopsisin en önemli adımını oluşturmaktadır. Bu amaç için kullanılan tüm optimizasyon yöntemleri aktivitenin kaynak videodaki zamanı ile oluşturulacak video sinopsis üzerindeki en uygun gösterim zamanı arasındaki ilişkiyi hesaplamayı hedeflemektedir. Optimizasyon problemi, genel olarak tanımlanan global enerji fonksiyonunun minimuma indirgenmesi şeklinde tanımlanmaktadır. Enerji fonksiyonunu tanımlarken aktivite yoğunluğunun azalması, aktivitelerin arka alan ile uyumsuzluğu, kronolojik sıralarının bozulması ve kesişim durumlarını cezalandıran metrikler kullanılmaktadır. Bu metrikler ile oluşturulan enerji fonksiyonunun değerini minimuma indiren aktivite haritasının en

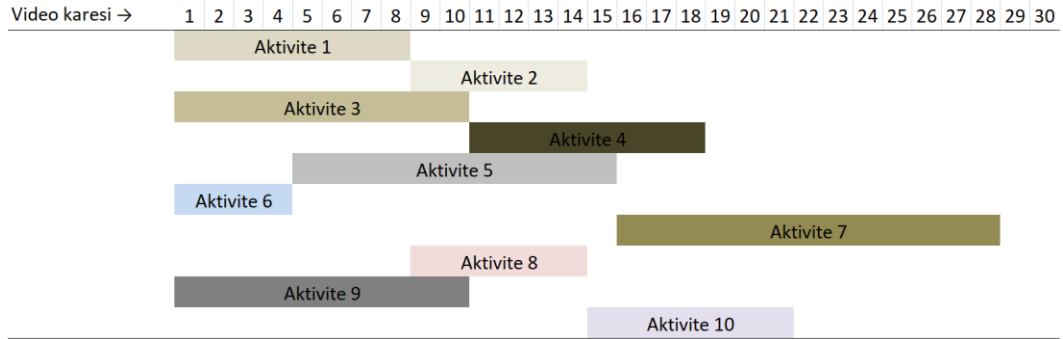
uygun çözüm olduđu kabul edilmektedir. Literatürde yaygın olarak kullanılan dört metrik aşğıdaki gibi açıklanmaktadır:

- Aktivite yoğunluğu: Mümkün olan en fazla sayıda aktivitenin video sinopsise dâhil edilmesini sağlamaktadır. Video sinopsis dışında kalan her aktivite için enerji fonksiyonunun değeri artırarak cezalandırmaktadır.
- Arka alan tutarlılığı: Nesnenin video sinopsiste uygun konuma yerleştirilmesi esnasında kaynak görüntüde sahip olduđu arka alan ile benzer bir zemine oturtulmasını garanti etmektedir. Video sinopsisin arka alanının, aktivitelerle ve sahne değışimleriyle uyumlu olmasını zorlamaktadır.
- Kronolojik tutarlılık: Video sinopsiste yer alan aktivitelerin kaynak video üzerindeki kronolojik sıralarının korunmasını amaçlamaktadır. Sıralamanın bozulmasını enerji fonksiyonunun değeri artırarak cezalandırmaktadır.
- Kesişim: Görsel kaliteyi artırmak için nesnelerin uzamsal olarak kesişmesini engellemeyi amaçlamaktadır. Video sinopsisteki nesne kesişim oranında enerji fonksiyonunun değeri artırarak bu durumu cezalandırmaktadır.

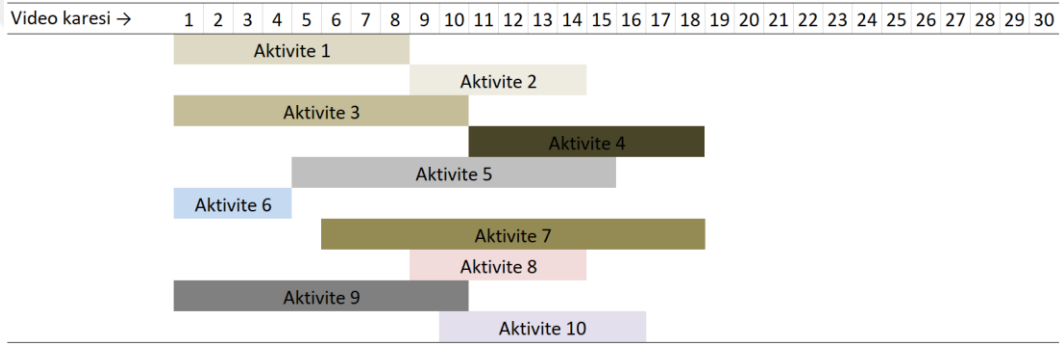
Bu metriklerden herhangi birinin ihlal edilmesi global enerji fonksiyonunun değeri artıracağı için fonksiyonun sahip olabileceği minimum değere indirgenmesi zorlaşmaktadır. Bu yüzden bu metrikleri minimum düzeyde ihlal eden en kısa aktivite dağılımı hedeflenmektedir. Literatür incelemesinde bahsedildiği gibi metriklerin kullanımı her çalışmada farklılık göstermektedir. Tamamını kullanan çalışmaların yanında sadece bir metriğe göre optimizasyon işlemi gerçekleştiren çalışmalar da bulunmaktadır.

Bu metrikler arasında aktivite yoğunluğu ve arka plan tutarlılığı her bir aktivite için bağımsız olarak hesaplanırken, kronolojik tutarlılık ve kesişim aktiviteler arasında hesaplanmaktadır. Kronolojik tutarlılık ve kesişim metrikleri için genellikle ağırlık parametreleri kullanılmaktadır. Bu parametreler sayesinde optimal çözüme giderken bu metriklerin öncelikleri ayarlanmaktadır. Aynı enerji fonksiyonunun farklı ağırlık parametreleri ile minimuma indirgendiğinde elde edilen farklı aktivite dağılımları şekil 2.7’de simule edilmiştir. Kronolojik sıraya tamamıyla sadık kalınan (şekil 2.7.a),

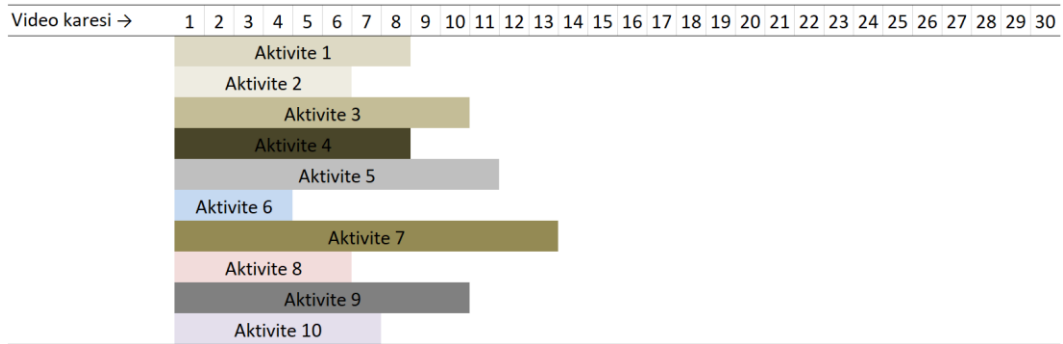
kronolojik sıralamanın bir kısmının değiştiği (şekil 2.7.b) ve tamamen ihmal edildiği (şekil 2.7.c) durumlar bu şekil üzerinde görülmektedir.



(a)



(b)



(c)

Şekil 2.7 Kronolojik tutarlılık metriğinin farklı ağırlıkları için aktivite dağılımı

- a. Kronolojik sıralamanın tamamen korunduğu durum, b. Kronolojik sıralamanın kısmen değiştiği durum, c. Kronolojik sıralamanın tamamen ihmal edildiği durum

Şekil 2.7.a'da görüldüğü gibi aktiviteleri aynı kronolojik sıra ile göstermek daha uzun video sinopsise sebep olmaktadır. Bu durumda tüm aktiviteler 28 video karesinde

gösterilmiştir. Kronolojik tutarlılık metriğinin daha düşük ağırlık metriği ile baskılandığı durumda tüm aktiviteler 18 video karesinde gösterilmiştir (Şekil 2.7.b). Diğer tarafta kronolojik tutarlılığı tamamen ihmal etmek 13 video karesi ile en kısa video sinopsisin oluşturulmasını sağlamıştır (Şekil 2.7.c). Bu senaryoda video sinopsis uzunluğunu en uzun aktivite uzunluğu belirlemiştir fakat tüm aktivitelerin aynı anda gösterilmesi görsel olarak kaotik bir durum yaratmaktadır. Her ne kadar en düşük enerjiye sahip çözümü sunsa da oluşturulan video sinopsisin görsel kalitesi çok düşük olacaktır. Bu gösterim ağırlık parametrelerinin etkisinin yanında optimizasyonda birden fazla metrik kullanmanın önemini vurgulamaktadır. Yukarıdaki örnekte kronolojik tutarlılığın yanında kesişim metriğinin de kullanılması 3. senaryodaki kaotik durumu görsel kalite ve video sinopsis uzunluğu arasında bir denge noktasına taşıyacaktır. Görüldüğü gibi aktivite optimizasyon farklı metriklerin kontrol edilmesini gerektiren lineer olmayan bir problemdir. Bu yüzden problemin çözümü için masraflı optimizasyon yöntemlerine başvurulmuştur.

Yukarıda anlatılan metriklerle istenmeyen durumları cezalandıran bir enerji fonksiyonu tanımlandıktan sonra bu fonksiyonun minimum değere sahip olduğu durumu hesaplamak için çevrimiçi ve çevrimdışı optimizasyon yöntemleri kullanılmaktadır. Çevrimiçi ve çevrimdışı optimizasyon yöntemlerinin her ikisinin de avantaj ve dezavantajları bulunmaktadır. Çevrimdışı yöntemler global optimal sonuca daha fazla yaklaştıkları için çevrimiçi yöntemlere göre aktivite yoğunluğu daha yüksek video sinopsis sunmaktadır. Fakat çevrimdışı optimizasyon işlemi özellikle aktivite sayısı arttığında oldukça masraflı hale gelmektedir. Çevrimiçi yöntemler çalışma zamanı olarak daha iyi performans sergilese de optimizasyon kalitesinin artırılmaya ihtiyacı vardır. Bu yöntemlerin farkları ve video sinopsiste kullanılan çevrimiçi ve çevrimdışı optimizasyon yöntemlerinin detayları bu bölümün alt başlıklarında anlatılmıştır.

2.2.1.1 Çevrimdışı optimizasyon

Çevrimdışı yöntemler optimizasyon işlemine başlamadan önce tüm aktivitelerin hazır bulunmasına ihtiyaç duymaktadır. Tüm veri tek seferde işlenerek global en uygunun bulunması hedeflenmektedir.

Simulated annealing (Kirkpatrick vd. 1983), video sinopsiste en yaygın kullanılan çevrimdışı optimizasyon yöntemidir. Bu yöntem, pek çok değişkene sahip fonksiyonların en büyük veya en küçük değerlerinin bulunması için tasarlanmıştır. Özellikle pek çok yerel en küçük değere sahip doğrusal olmayan fonksiyonların global optimal değerlerini bulmak için yaygın olarak kullanılmaktadır. Video sinopsiste de öncü çalışmalar bu yöntemi kullandığı için sonrasında birçok çalışma farklı enerji fonksiyonları tanımlasa da bu yöntemle minimize yapılmaktadır. *Mean shift* ve *alpha-beta swap graph-cut* (Boykov vd. 2001) benzer şekilde global küme üzerinde iterasyonlar ile optimal çözüme gitmek için video sinopsiste yaygın kullanılan diğer yöntemlerdir.

Genetik algoritmaların (Whitley 1994), son yıllarda aktivite optimizasyonunda kullanılmaya başlandığı görülmektedir. *Simulated annealing* ile karşılaştırıldığında hem aktivite yoğunluğu hem de hesaplama karmaşıklığı olarak daha iyi performans göstermesi popülerliğini artırmıştır.

Yukarıda bahsedilen optimizasyon yöntemleri literatürde farklı alanlarda halihazırda kullanılan ve video sinopsis çalışmalarına entegre edilmiş yöntemlerdir. Bunların dışında aktivitelerin tekrar düzenlenmesi için geliştirilmiş optimizasyon yöntemleri de bulunmaktadır. Pritch vd. (2009) tarafından önerilen *packing cost* yöntemi benzerliklerine göre sınıflandırılmış aktivitelerin zaman düzleminde kaydırılarak optimal pozisyonlarının bulunmasını amaçlamıştır. Aktiviteleri video sinopsise eklerken en uzun olanlara öncelik veren bir strateji izlenmiştir. Aktivitelerin tek tek birbiri ile zaman düzleminde kesişimlerini hesaplamak yerine sınıflandırılmış aktivite gruplarının kesişimleri dikkate alınmıştır. Bir sınıfa ait tüm aktiviteler tek seferde video sinopsise dahil edilmiştir. Bu sayede optimizasyon problemi aktivite sınıflandırma ve sınıfların optimal pozisyonlarını bulma olmak üzere iki aşamaya ayrılmıştır. Her bir sınıfı optimizasyon aşamasında bir aktivite gibi dikkate alınması problemin basite indirgenmesine imkan sağlamıştır.

Film map generation (Kasamwattanarote vd. 2010), nesnelerin görüntüdeki derinlikleri ve bulundukları video karelerini temsil eden bir model oluşturmuştur. Bu model yardımı

ile nesne kesiřimleri hesaplanarak zaman d zleminde kaydırılmıřtır. *Trajectory clustering* (Chou vd. 2015) ve *low-complexity range tree* (Hsia vd. 2016) y ntemleri metrik olarak sadece nesne kesiřimini kullanmıřtır. Nesneler hi kesiřim olmayana kadar iteratif olarak kaydırılmıřtır.

Abnormality-type-based optimizasyon y nteminde (Lin vd. 2015) enerji fonksiyonunu tanımlarken kronolojik tutarlılık ve kesiřim metriklerinin yanında aktivite benzerlik metrięi de kullanılmıřtır. Aktivite benzerlik metrięi benzer aktivitelerin beraber g sterilmesini zorlayarak g rsel kaliteyi artırmayı amalamıřtır.

Hsia vd. (2016) tarafından  nerilen *low-complexity range tree* y ntemi kriter olarak sadece aktivitelerin uzamsal kesiřimini dikkate alan basit bir optimizasyon y ntemi  nermiřtir. Tanımlanan enerji fonksiyonu aktivite daęılımları iin hesaplanmış ve bu iřlem aktiviteler arasında herhangi bir uzamsal kesiřim kalmayana ya da kullanıcı tarafından tolere edilebilecek kesiřimler kalana kadar iteratif olarak tekrarlanmıřtır. Bu y ntemin asıl amacı aktivite optimizasyon iřlemini iyileřtirmek yerine kullanıcının aradıęı aktiviteyi verimli bir řekilde bulmaktır.

Bu y ntemlerin en  nemli iki problemi; t m veriyi tek seferde iřlemek iin gereken y ksek miktarda bellek ihtiyacı ve t m aktivitelerin birbirine g re konumunu deęerlendirmek iin gereken iřlem s resinin uzun olmasıdır.  evrimdışı y ntemlerin hesaplama karmařıklıęı ok y ksektir ve aktivite sayısının artması ile  ssel olarak artmaktadır. T m veri iteratif olarak iřlenerek minimum enerjiye ulařılmaya alıřılmasından dolayı aktivite dizilimindeki bir deęiřiklik bir sonraki iterasyonda t m enerji fonksiyonunun tekrar hesaplanmasına gerek duymaktadır. Bu durum optimal dizilime ulařma iřleminin zaman almasına ve y ksek iřlem g c ne ihtiya duymasına sebep olmaktadır. Bu kısıtlamalar  evrimdışı y ntemlerin kamera yayın akıřları  zerinde gerek-zamanlı olarak uygulanmasının  n ne gemektedir.

2.2.1.2 Çevrimiçi optimizasyon

Çevrimiçi yöntemler her yeni aktivite algılandığında adım adım güncellenen bir optimizasyon stratejisi izlemektedir. Algılanan aktivite o an bellekte tutulan aktivite seti içinde kullanılan metriklere göre en uygun konuma yerleştirilmektedir. Çevrimdışı yöntemlerdeki gibi tüm veri üzerinden işlem yapılmadığı için yerel en uygun çözüme ulaşılmaktadır. Bu durum, oluşturulan video sinopsisin çevrimdışı yöntemlere göre daha düşük aktivite yoğunluğuna sahip olmasına sebep olmaktadır. Bunun yanında, çevrimdışı yöntemlerin gerek duyduğu yüksek bellek ihtiyacı ve zaman alan hesaplama karmaşıklığı problemleri ile karşılaşmamaktadır. Bu sayede gerçek-zamanlı olarak güvenlik kamera görüntülerine uygulanabilmektedir. Video sinopsis çalışmalarında kullanılan çevrimiçi optimizasyon yöntemleri (EK 1 çizelge 1’de * ile işaretlenmiştir) aşağıdaki gibidir.

Online tube filling (Feng vd. 2012) algılanan her bir nesneyi mevcut aktiviteler arasında kaydırarak ideal pozisyonunu aramaktadır. Mevcut aktivite kümesini tutmak için belirli sayıda video karesinden oluşan L1 ve sınırsız alana sahip L2 adı verilen iki alan oluşturulmuştur. Algılanan aktivitenin L1 içinde bir karede başlaması gerekirken uzunluğu L1’e sığmazsa L2 kullanılmıştır. Kesişim metriğinden oluşan bir enerji fonksiyonu tanımlanmıştır. Aktivitelerin kesişiminin az olduğu durumlar yerleştirmede önceliğe sahip olmaktadır. Ayrılan L1 alanının yeni aktivite için dolu olup olmadığına bakılmakta ve eğer aktivite için yeterli alan yoksa L2 alanından belirli bir miktar alan L1 alanına aktarılmaktadır. Bu sayede sinopsis video uzunluğu olay yoğunluğuna göre otomatik olarak ayarlanmaktadır.

Huang vd. (2014) tarafından önerilen sinopsis tablosu görüntüdeki her pikselin o anki durumunu temsil etmektedir. Tabloda pikseli işgal eden nesne numarası ve hangi video karelerinde burada bulunduğu bilgisi tutulmaktadır. Algılanan nesneler sinopsis tablosunda ilgili pozisyonlar boş ise yerleştirilmektedir. Aksi takdirde o pozisyonda var olan nesne ile kullanılan kesişim metriğinin en fazla izin verdiği ölçüde kesişmesi için tablonun ilgili alanı boşalana kadar bekletilmektedir. Oluşturulan sinopsis tablosu video sinopsiste gösterilen nesnelerin haritasını tutmaktadır. Jin vd. (2016) tarafından önerilen

simple tube generation yöntemi de benzer şekilde bir sinopsis tablosu tutmaktadır. Önceki yöntemden farklı olarak sinopsis tablosunda sadece ilgili pikseli işgal eden nesne numarası tutulmaktadır. Aynı anda birden fazla nesnenin aynı pikseli işgal etmesine izin verilmediği için nesnenin o konumda bulunduğu video karelerini tutmaya gerek kalmamaktadır.

Mahapatra vd. (2016) tarafından önerilen yöntem de anlık aktivite durumunu temsil eden bir tablo kullanarak optimizasyon gerçekleştirmektedir. Fakat piksel tabanlı bir yaklaşım yerine nesneleri referans alan bir yöntem önerilmektedir. Algılanan nesnenin tüm izi o an video sinopsiste yer alan nesne izleri ile karşılaştırılarak kesişim haritası çıkarılmaktadır. Algılanan nesne herhangi bir nesne ile kesişmeye kadar kaydırılmaktadır. He vd. (2017b) benzer şekilde nesnelerin kesişim haritalarını çıkaran bir yöntem önermektedir. Önceki yöntemlerden daha detaylı olarak nesne kesişimi 3 farklı şekilde tanımlanmaktadır: kesişim olmayan, aynı yönde ve farklı yönde kesişim. Mevcut aktivitelerin tamamını dikkate alarak görüntünün kesişim haritasını çıkardıktan sonra kesişim olmayan nesneleri herhangi bir zamana yerleştirirken kesişen nesneleri önceden belirlenmiş kesişim periyotlarının dışına yerleştirmektedir.

Görüldüğü gibi çevrimiçi yöntemler genellikle tanımlanan metrikler doğrultusunda sahnenin o anki aktivite durumunu temsil eden bir tablo oluşturmaktadır. Algılanan nesneler bu tablo kullanılarak uygun pozisyonlara yerleştirilmekte ve bu doğrultuda tablo güncellenmektedir.

2.2.2 Kamera topolojisi

Video sinopsis çalışmalarının çoğunluğu tek kamera görüntüsü üzerine odaklanmıştır. Çok kameralı topoloji üzerinde video sinopsis gerçekleştiren sadece dört çalışma bulunmaktadır (EK1 çizelge 1’de ** ile işaretli). Zhu vd. (2014) bakış açısı kesişen kamera görüntüleri üzerinde video sinopsis oluşturan bir yöntem önermiştir. Kamera görüntüleri arasında homografi matrislerini çıkararak tüm görüntüleri kapsayan ortak bir referans görüntü oluşturmuştur. Kameralar arasında nesnelerin ilişkilendirilmesi için görsel eşleştirme kullanmak yerine kamera kesişim alanlarında nesnelerin izlerini

eşleştirmiştir. Mahapatra vd. (2016) kesişen kamera görüntüleri üzerinde çalışan başka bir video sinopsis yöntemi önermiştir. Zhu vd. (2014) tarafından önerilen yönteme benzer şekilde homografi matrislerini kullanarak ortak bir düzlem oluşturmuştur. Video sinopsis gösteriminde aktiviteler kuş bakışı ortak düzlem üzerinde işaretlenirken görüldüğü kameraya ait görüntüsü eş zamanlı olarak gösterilmiştir (Şekil 2.8). Kamera görüntüleri arasında nesne eşleştirmesi benzer şekilde kamera kesişim alanlarında nesne izlerinin karşılaştırılması ile gerçekleştirilmiştir.



Şekil 2.8 Çok kameralı video sinopsis gösterimi (Mahapatra vd. 2016)

Yukarıdaki görüntü nesnelerin sahnenin kuş bakışı görünümündeki konumlarını temsil ederken alt tarafta her nesnenin algılandığı kameradaki görüntüsü yer almaktadır

Hoshen ve Peleg (2015) tarafından gerçekleştirilen çok kameralı video sinopsis çalışmasında belirlenen master kameradaki aktiviteler yardımcı kameralar tarafından desteklenmiştir. Kullanıcının master kamera üzerindeki aktivitelerle ilgili daha kolay çıkarım yapmasını sağlamak için aynı zamanda yardımcı kameralarda görülen

aktiviteler de eş zamanlı olarak gösterilmiştir. Kamera görüntüleri arasındaki aktiviteler arasında herhangi bir eşleştirme yapılmamıştır.

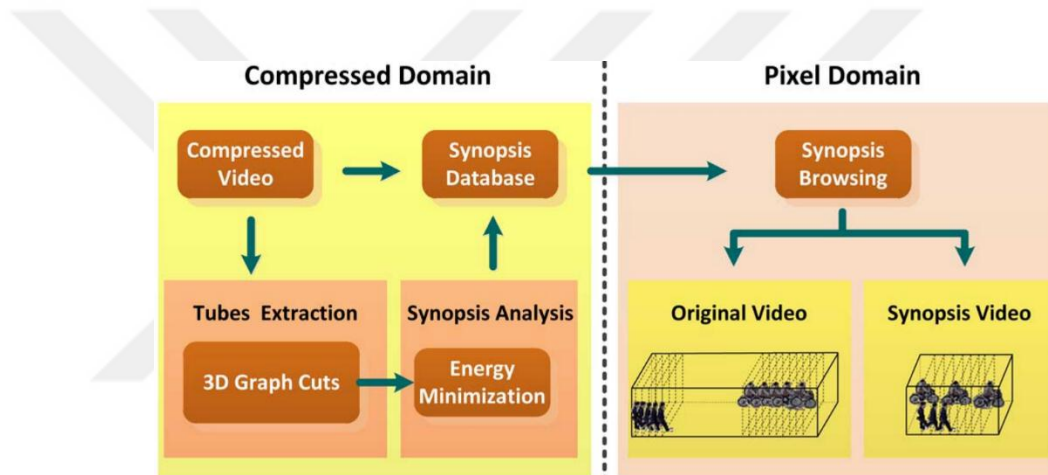
Zhu vd. (2016a) tarafından gerçekleştirilen çalışma kameraların bakış açısının kesişmesi şartı aramayan ve kameralar arasında nesneleri görsel olarak eşleştiren tek çalışmadır. Her kamerada aktiviteler ayrı olarak çıkarıldıktan sonra önerilen *joint tube rearrangement* yöntemi ile tek bir aktivite olarak birleştirilmiştir. Gösterim esnasında önceki yöntemler gibi ortak bir referans görüntü üzerinde göstermek yerine her kameraya ait aktiviteler bağımsız olarak gösterilmiştir.

Görüldüğü gibi çok kameralı video sinopsis çalışmaları hala sınırlı sayıdadır. Günlük hayatta kullanılan gözetleme kameraları genellikle birden fazla kameradan oluşan kapalı devre bir sistemde yer almaktadır ve aktiviteler genellikle birden fazla kamera görüntüsünde devam etmektedir. Herhangi bir olayın anlaşılması adına daha geniş bir bakış açısı sunması açısından çok kameralı yöntemler tek kameralı yöntemlere göre daha etkili çözüm sunmaktadır. Diğer taraftan video sinopsisi tek kamera yerine çok kameralı bir sistemde uygulamak bazı zorlukları da beraberinde getirmektedir. Kameralar arasında nesnelerin güvenilir bir şekilde takip edilmesi hala zorlu bir problemdir. Çok kameralı nesne takibindeki hatalar aynı nesnenin farklı aktiviteler olarak algılanmasına sebep olacağından dolayı video sinopsisin anlamsal bütünlüğüne zarar vermektedir. Bu da tek kameralı çözümlere göre dezavantaj oluşturmaktadır. Bunun yanında çok kameralı sistemde aktivitelerin kameralar arasında ilişkilendirilmesiyle ilgili gerekli olan ekstra metrikler sebebiyle aktivite optimizasyon işlemi daha karmaşık bir hal almaktadır.

Detayları bölüm 2.1.4'te verilen dağıtık video sinopsis modeli kapalı devre kamera sistemlerinin dağıtık yapısı ve bulut teknolojisinin hızlı gelişimi dikkate alındığında yenilikçi bir çözüm sunmuştur (Lin vd. 2017). Dağıtık mimaride gerçekleştirilen çok kameralı video sinopsis yaklaşımı teknolojik gelişmeler dikkate alındığında üzerinde daha fazla çalışma gerektiren bir konu olarak görülmektedir.

2.2.3 Girdi veri tipi

Video sinopsis çalışmalarının büyük bir çoğunluğu girdi olarak piksel değerlerini almaktadır. Bunun yanında Wang vd. (2013a,b), Zhong vd. (2014) ve Liao vd. (2017) tarafından önerilen yöntemler sıkıştırılmış veri üzerinden çalışmaktadır. Bu çalışmalar özellikle çevrimdışı optimizasyon yöntemlerinin kamera görüntüleri üzerinde gerçek-zamanlı olarak uygulanması amacıyla yöntem karmaşıklığını düşürmek için önerilmiştir. Bu sebeple şekil 2.9’da görüldüğü gibi video sinopsis adımlarının bazılarını videoyu çözmeden direk olarak sıkıştırılmış veri üzerinde yapmaya odaklanmışlardır.



Şekil 2.9 Sıkıştırılmış düzlemde video sinopsis altyapısı (Zhong vd. 2014)

Sinopsis bilgisi, video sinopsiste karşılık gelen kare indeksi, aktivite sayısı gibi bilgiler sıkıştırılmış veri üzerinden elde edildikten sonra ilgili video karesinin içine kodlanmıştır. Bu sayede bu bilgilerin saklanması için gereken bellek ihtiyacından da tasarruf edilmiştir.

Sıkıştırılmış veri üzerinde işlem yapan yöntemlerin çalışma zamanı performansı daha iyi olsa da sıkıştırılmış veri üzerinde gerçekleştirilen video analizi piksel değerleri üzerinde gerçekleştirilen yöntemlere göre daha düşük başarıma sahip olduğu için bu optimizasyon yöntemlerinin aktivite yoğunluk performansı daha düşüktür. Sıkıştırılmış veri üzerinde işlem yapan yöntemler işlenmiş veriyi video karesine kodladığından dolayı video tarama ve ilgilenilen alana hızlı erişim konusunda kullanıma daha uygun

görülmektedir. Sıkıştırılmış veri üzerinde çalışan yöntemlerin çalışma zamanı kazancı optimizasyon aşamasında kaybedilen başarımları telafi edemediği için bu tarz çalışmalar literatürde çok fazla ilgi görmemiştir. Bu yöntemler sıkıştırılmış veri üzerinde gerçekleştirilen video analizin başarımları ile ilgili bilgi veriyor olsa da optimizasyon başarımları konusunda daha fazla araştırmayı teşvik edecek kadar umut vadetmemiştir.

2.2.4 Aktivite sınıflandırma

Çizelge 2.1’de listelendiği gibi video sinopsis çalışmalarının bazıları optimizasyon aşamasından önce benzer aktiviteleri sınıflandırıp beraber göstermek için aktivite sınıflandırma adımını uygulamıştır.

Çizelge 2.1 Aktivite sınıflandırma yöntemleri

Çalışma	Sınıflandırma yöntemi
Pritch vd. (2009)	SVM (Cortes ve Vapnik 1995)
Vural ve Akgül (2009)	Eye tracking (LC Technologies)
Rodriguez (2010)	Action MACH filter (Rodriguez vd. 2008)
Chou vd. (2015)	Longest common sequences (Vlachos vd. 2002)
Lin vd. (2015)	Abnormal activity classification (Lin vd. 2015)
Mahapatra vd. (2016)	SimpleMKL (Rakotomamonjy vd. 2008)

Bu alandaki ilk çalışma göz takip yöntemi kullanarak güvenlik görevlisinin izlediği ve kaçırdığı aktiviteleri sınıflandırmıştır. Video sinopsiste bu sınıflandırmaya göre benzer aktiviteleri beraber göstermiştir (Vural ve Akgül 2009). Bu çalışmanın temel amacı güvenlik görevlisinin canlı izleme esnasında kaçırdığı aktiviteleri video sinopsiste kullanıcıya sunmaktır. Pritch vd. (2009) aktivitelerin görünüm ve hareket özniteliklerini kullanarak SVM (Cortes ve Vapnik 1995) yardımı ile aktiviteleri sınıflandırmıştır. Rodriguez (2010) *action MACH* (Rodriguez vd. 2008) filtresi kullanarak aktiviteleri koşma, bir obje alma, araca girme gibi belirli kategorilere ayırmıştır. Sonrasında kullanıcı tarafından belirlenen aktivite grubu video sinopsis içinde gösterilmiştir.

Görünüm ve hareket tipine göre sınıflandırma yapan yukarıdaki yöntemlerden farklı olarak Chou vd. (2015), *longest common sequence* algoritması vasıtasıyla aktiviteleri

uzamsal konumlarına göre sınıflandırmıştır. Sahnede potansiyel giriş ve çıkış alanları olarak dört farklı bölge belirlenmiş ve aktiviteler bu bölgelerden giriş ve çıkış yapma durumlarına göre sınıflandırılmıştır. Lin vd. (2015) sahnedeki normal aktiviteleri eğitim yoluyla öğrenen bir yöntem önermiş ve bu sayede anormal durumları tespit etmeyi amaçlamıştır. Anormal aktiviteleri tespit etmenin yanında uzamsal pozisyonlar da sınıflandırma için kullanılmıştır. Sahnede belirlenen kritik bölgeler aktivitenin akışını (sahneye giriş ve çıkışını) belirlemek için kullanılmış ve video sinopsiste gösterilen aktiviteler bu sınıflandırmaya göre belirlenmiştir. Bunların dışında, Mahapatra vd. (2016) *multiple kernel learning* (Rakotomamonjy vd. 2008) yöntemini kullanarak aktivite tipini algılamayı amaçlamıştır. İnsan hareketlerine ait görünüm öznitelikleri kullanılarak aktiviteler yürüme, koşma, eğilme, zıplama, el çırpma, tek el sallama ve iki eli de sallama gibi sınıflara ayrılmış ve aynı kategorideki aktiviteler video sinopsiste beraber gösterilmiştir.

Uygulamanın amacına göre aktivite sınıflandırma pozitif veya negatif etki edebilir. Benzer aktivitelerin video sinopsiste eş zamanlı olarak gösterilmesi kullanıcının aktivitelere daha kolay odaklanmasını ve videonun daha anlaşılır olmasını sağlamaktadır. Diğer taraftan, farklı kategorilere ait aktiviteler birbiri ile ilişkili olabilir ve kullanıcı video sinopsis üzerinde bu bağlantıyı arıyor olabilir. Bu durumda sınıflandırma yapmadan tüm aktivitelerin beraber gösterilmesi daha yararlı olacaktır. Aktivite sınıflandırma, algoritmik akışta ekstra bir adım olarak uygulanmasından dolayı video sinopsis boru hattının işlemsel karmaşıklığını artırmaktadır. Genel olarak video sinopsisin gösteriminde çeşitlilik sağlama açısından metodoloji üzerinde bir iyileştirmeden çok uygulama seviyesinde bir özellik olarak düşünülmelidir. Bu anlamda herhangi bir video sinopsis uygulamasında opsiyonel bir adım olarak kullanılabilir.

3. ÇOK KAMERALI NESNE TAKİBİ TABANLI ÇEVİRİMİÇİ VIDEO SİNOPSİS SİSTEMİNİN GELİŞTİRİLMESİ

Bu tezde yeni bir çok kameralı nesne takibi tabanlı çevrimiçi video sinopsis sistemi geliştirilmiştir. Mevcut video sinopsis çalışmaları genellikle tek kamera görüntüsü üzerine uygulanmıştır. Var olan az sayıdaki çok kameralı çevrimiçi video sinopsis yöntemi ise görüntüsü kesişmeyen kameralar arasında nesneleri görsel olarak ilişkilendirmemiştir. Bu bakış açısı, kullanıcının geniş alanda aynı nesneye ait aktiviteleri kavraması ve gözetlemesi için yetersiz kalmıştır. Bu yüzden bu tez çalışmasında hedef nesnelerin görüntüsü kesişen veya kesişmeyen farklı kamera görüntülerinde görsel olarak eşleştirilip herhangi bir kamerada görüntüye girmesinden itibaren tespit edildiği tüm kameralarda takip edilmesi ve bu takibin raporlanması amaçlanmıştır. Bu kameraların herhangi birinde algılanan aktiviteler kamera özelinde video sinopsis olarak gösterilmiştir. Nesne özelinde bakıldığında, nesne ilk görüldüğü kameranın video sinopsisine dahil edilip gösterildikten sonra, kronolojik sıra ile görüldüğü kameraların video sinopsisine dahil edilmek üzere ilgili kameranın aktivite havuzuna taşınmıştır.

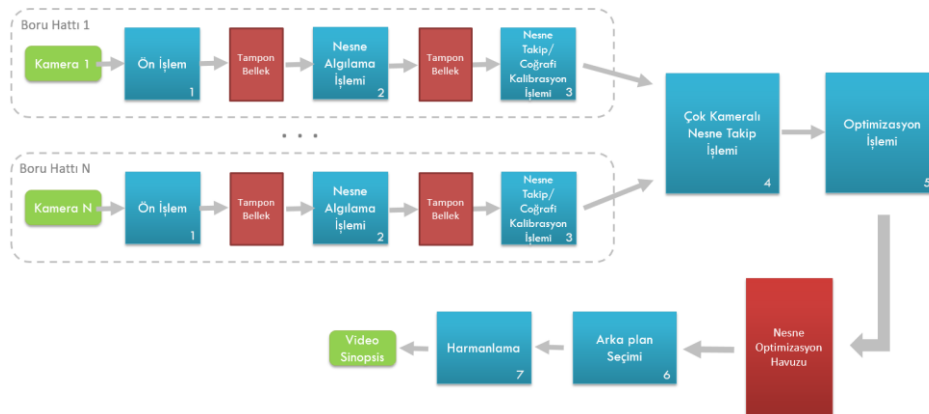
Literatürde yer alan video sinopsis çalışmaları genellikle aktivite optimizasyon aşamasına yoğunlaşmıştır. Tez kapsamında video sinopsis kamera görüntüsünün alınmasından video sinopsis oluşturmaya kadar bir bütün olarak ele alınmıştır. Video sinopsis sisteminin her bir adımı bir sonraki adıma girdi oluşturduğu için özellikle aktivite optimizasyonundan önceki adımların başarımı sonucu doğrudan etkilemektedir. Bu yüzden çok kameralı video sinopsis sisteminin her adımında video sinopsis için en uygun yöntem geliştirilmiştir. Her adımda yapılan katkılar aşağıdaki gibi özetlenmektedir.

1. Dış ortamda çevresel koşullardan minimum düzeyde etkilenmek adına literatürde kabul görmüş bir hareket algılama yöntemi iyileştirilerek kullanılmıştır.
2. Hareketli nesne takibi için yeni bir yöntem geliştirilmiştir. Video sinopsiste karşılaşılan en önemli sorunlardan bir tanesi hedef nesnelerin yeteri kadar iyi

takip edilememesi sonucunda aktivitelerin kopması ve gereksiz aktiviteler yüzünden optimizasyon aşamasının karmaşıklığının artmasıdır. Bu sorunu çözmek için hedef nesnenin görünüm değişimlerinden ve çevresel koşullardan minimum düzeyde etkilenen uzun süreli takibe odaklanmış video sinopsise özgü yeni bir nesne takip yöntemi geliştirilmiştir. Geliştirilen nesne takip yöntemi genişletilerek çok kameralı nesne takibine uygun hale getirilmiştir.

3. Kameralar arasında aktivite eşleştirmesinde coğrafi kalibrasyon video sinopsis çalışmalarında ilk defa kullanılmıştır.
4. Çevrimiçi aktivite optimizasyonu için yeni bir yöntem geliştirilmiştir. Literatürde yer alan çevrimiçi optimizasyon yöntemleri çalışma zamanı performansına odaklanırken başarımdan ödün verilmiştir. Geliştirilen çevrimiçi optimizasyon yönteminde grid haritası implementasyonu sayesinde elde edilen çalışma zamanı performans kazancı, başarıımı destekleyecek daha güçlü bir optimizasyon yönteminin tasarımında kullanılmıştır. Literatürde kullanılan optimizasyon yöntemlerinde nesne boyutu arttıkça video sinopsiste gösterilme önceliği artmaktadır. Fakat nesne öneminin boyutundan bağımsız olduğu düşünülerek geliştirilen yöntemde tüm nesnelere eşit ağırlık verilmiştir.

Tez kapsamında geliştirilen video sinopsis sisteminin mimarisi şekil 3.1’de verilmiştir. Numaralandırılmış her bir blok, sistem dâhilinde uygulanacak işlemleri göstermektedir. Bundan sonraki açıklamalarda genel mimariye ait herhangi bir yöntemden bahsedilirken şekil 3.1’de hangi bloğa karşılık geldiği, ilgili blok numarası (Blok n) ile belirtilecektir.



Şekil 3.1 Geliştirilen video sinopsis sisteminin mimarisi

Geliştirilen video sinopsis sisteminde her bir kamerada hareket algılama (Blok 2) sonucu tespit edilen nesneler görüntüde olduğu süre boyunca takip edilmektedir (Blok 3). Kameranın dahili ve harici parametreleri veya kullanıcı tarafından sağlanan kalibrasyon bilgileri kullanılarak tespit edilen nesnelerin coğrafi koordinatları da nesne etiketleme işlemine dahil edilmektedir (Blok 3). Kamera özelinde takip edilen hareketli nesneler, hareketine ait coğrafi bilgiler de göz önünde bulundurularak farklı kameralarda eşleştirilmektedir (Blok 4). Etiketlenen nesneler aktivite optimizasyon yöntemi ile gerçek zamanlı olarak zaman düzleminde optimal pozisyonları bulunacak şekilde tekrar düzenlenmektedir (Blok 5). Bu bölümün alt başlıklarında bahsedilen mimarideki her bir adım için geliştirilen yöntemlerin detayları verilmiştir.

3.1 Nesne Algılama İşlemi

Geliştirilen video sinopsis sisteminde, tipden (insan, araç, hayvan, vb.) bağımsız olarak tüm nesnelerin algılanması amaçlanmıştır. Bu yüzden nesne tipine özgü eğitim gerektiren öğrenme tabanlı yöntemler yerine hareket algılama yöntemi kullanılmıştır. Mevcut video sinopsis çalışmalarında kullanılan hareket algılama yöntemleri genellikle ardışık kareleri karşılaştıran basit yöntemlerdir. Video sinopsiste hareket algılama, algoritmik akışta ön işlem olarak görüldüğü için üzerinde çok fazla durulmamıştır. Bu da kullanılan yöntemlerin gölge, yansıma gibi çevresel koşullardan çok fazla etkilenmesi sonucunda yanlış algılamalara sebep olmaktadır. Algılanan nesne sayısının artması video sinopsis probleminin karmaşıklığını artırdığı ve çalışma performansını düşürdüğü için yanlış nesne algılamalarını minimuma indirmek başarıyı doğrudan etkilemektedir. Bundan dolayı video sinopsiste nesne algılama geliştirilmesi gereken bir alan olarak gözlemlenmiştir.

Bu anlamda istatistiksel modelleme tabanlı bir hareket algılama yönteminin (Kim vd. 2005, Barnich ve Droogenbroeck 2011) kullanılması nesne algılama kalitesini en üst seviyede tutmak adına önemli katkı sağlayacaktır. Bu yöntemler genel olarak sahnenin sabit arka planının modellenmesi, yeni piksellerin arka plan modeli ile karşılaştırılıp hareket olarak belirlenmesini kapsayan karar aşaması ve sahne değişime karşı arka plan modelinin güncellenmesi olmak üzere 3 temel adımdan oluşmaktadır. Bu 3 aşamada da

kullanıcı tanımlı parametreler kullanılmakta ve bu parametreler yöntemin sahneden bağımsız çalışmasının önüne geçmektedir. Hofmann vd. (2012) tarafından önerilen “non-parametric” yöntemler kullanıcı tanımlı parametre sayısını sınırlandırıp sahne değişimine uyumlu dinamik arka plan modelleri oluşturmaktadır. Video sinopsiste kullanılacak hareket algılama yönteminin çevre koşullarından minimum düzeyde etkilenmesi ve farklı ortamlarda yüksek başarımla çalışması gerekmektedir. Bu yüzden bahsedilen “non-parametric” yöntemler iyileştirilerek kullanıcı tanımlı parametre sayısının azaltılması planlanmıştır. Ayrıca literatürde kullanılan hareket algılama yöntemlerinin dış ortamda karşılaşılan bulut geçişi, aydınlatma değişimi (gece-gündüz geçişi), araç farları gibi sahnenin aniden değiştiği durumlara hızlı adapte olamadığı gözlemlenmiştir. Bu durumda meydana gelecek yanlış algılamalar video sinopsisi olumsuz etkileyeceğinden dolayı sahnede normal eğilimin dışında ani değişimler tespit edildiğinde arka plan modelinin mümkün olan en hızlı şekilde yeni duruma adapte olması için arka plan güncelleme frekansının en yüksek seviyeye çıkarılmasını sağlayan ani değişim modülü eklenmiştir. Bu sorunlar göz önünde bulundurulduğunda Hofmann vd. (2012) tarafından önerilen PBAS (Pixel based adaptive segmenter) yöntemi temel alınarak yukarıda bahsedilen dış ortam koşullarına dayanıklılığını artırmak adına iyileştirilmiştir.

Tez kapsamında kullanılan arka plan modelleme yöntemi beş ana adımdan oluşmaktadır (Baskurt ve Samet 2017a):

1. Arka plan/ön plan kararı;
2. Arka plan modelini güncelleme;
3. Karar eşik değerinin dinamik güncellemesi;
4. Arka plan güncelleme frekansının dinamik güncellemesi;
5. Ani değişim algılayıcı.

Kullanılan hareket algılama yönteminde ilk olarak sahnenin hareketsiz arka planını temsil eden bir model oluşturulmuştur. Yeni video karesindeki her piksel arka plan modeli ile karşılaştırılarak hareketli bir nesne olduğuna veya arka plana ait olduğuna karar verilmiştir. Yeni piksel eğer arka plan olarak etiketlenmişse arka plan modeli piksel ile güncellenmiştir. Pikselin arka plan olup olmadığına karar verilmesini sağlayan

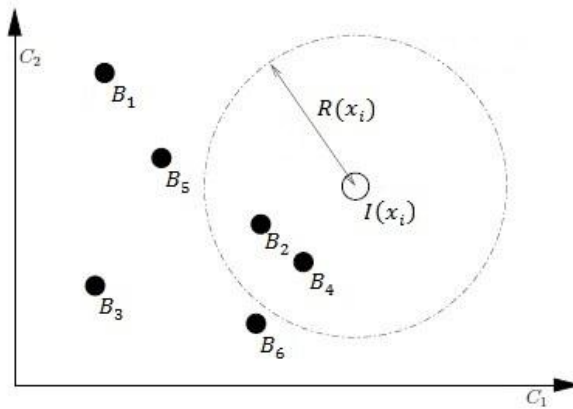
karar eşik değeri ve arka plan güncelleme frekansı sahnenin dinamikliğine göre otomatik olarak ayarlanmıştır. Son olarak ani değişim algılayıcı ile sahnede meydana gelen ani dominant değişimler algılanarak arka planın mümkün olan en hızlı şekilde yeni sahneye adapte olması sağlanmıştır. Bu bölümün alt başlıklarında her bir adımın detayları ve yapılan katkılar detaylı olarak anlatılmıştır.

3.1.1 Arka plan / ön plan kararı

Bu adım her bir pikselin arka plan modeli ile karşılaştırılarak hareketli bir nesneye mi yoksa hareketsiz arka plana mı ait olduğunun kararını vermeyi amaçlamaktadır. Arka plan modeli $B(x_{ij})$ görüntüdeki her bir pikselin (x_{ij}) geçmişte sahip olduğu rastgele N adet değerden oluşmaktadır.

$$B(x_{ij}) = \{B_1(x_{ij}), B_2(x_{ij}), \dots, B_N(x_{ij})\} \quad (3.1)$$

Karar eşik değeri ($R(x_{ij})$) ilgili renk uzayında pikselin güncel değeri ile arka plan örneği arasındaki izin verilen maksimum uzaklığı temsil etmektedir. Bir pikselin değeri ($I(x_{ij})$), arka plan modelinde karşılık gelen en az ($\#_{min}$) örnek ile karar eşik değerinden ($R(x_{ij})$) daha fazla yakın ise o piksel arka plan olarak etiketlenmektedir (Şekil 3.2).



Şekil 3.2 İki boyutlu (C_1, C_2) renk uzayı için 6 örnekle x_{ij} piksel modelinin güncellenmesi

3.1.2 Arka plan modelini güncelleme

Bir piksele ait arka plan modeli ancak o pikselin hareketsiz arka plana ait olduğuna karar verildiği durumda pikselin o anki değeri ile güncellenmektedir. Pikselin güncel değeri $I(x_{ij})$ arka plan modelinden rastgele seçilen bir örnek $B_k(x_{ij})$ yerine atanmaktadır. Sahnenin güncel durumu bu şekilde arka plan modeli tarafından öğrenilmektedir. Bahsedilen öğrenme (güncelleme) işleminin çalışma zamanı performansını dikkate alarak belirli bir frekansta yapılması gerekmektedir. Bu yüzden arka plan modeli her video karesi yerine $p = 1/T(x_{ij})$ frekansında güncellenmektedir. $T(x_{ij})$, piksele özgü arka plan güncelleme frekansını temsil etmektedir. Bu değer ilgili pikselin zaman içinde değişim sıklığına (dinamiklik) göre otomatik olarak ayarlanmaktadır. $T(x_{ij})$ 'nin detaylı tanımı bölüm 3.1.4'de verilmiştir.

Referans alınan çalışmada arka plan modelinin güncellemesinde pikselin kendi modeli dışında 8 bağlantılı komşuluğundan rastgele seçilen bir piksele (y_{ij}) ait arka plan modeli de komşu piksel değeri ile güncellenmektedir (Hofmann vd. 2012). Tez kapsamında geliştirdiğimiz yöntemde bu güncelleme işlemi komşu pikselin değeri yerine arka plan olarak etiketlenen piksel değeri ile yapılmıştır (Denklem 3.2).

$$B_k(y_{ij}) \leftarrow I(x_{ij}) \quad (3.2)$$

Rastgele seçilen komşu pikselin arka plana ait olmama ihtimali olduğu için referans yöntemdeki yaklaşım hareketli nesneye ait piksellerin de arka plana dahil edilmesine imkan sağlamaktadır. Hareketli nesneye ait piksellerin arka plan modeline dahil edilmesi modelin tutarlılığını bozacağından dolayı komşu güncelleme işleminde böyle bir değişiklik yapılmıştır.

3.1.3 Karar eşik değerinin dinamik güncellenmesi

Sahnede aynı anda hareketli nesnelere ve statik arka plana ait pikseller beraber yer almaktadır. Bunun için tüm pikseller için aynı arka plan/ön plan karar eşik değeri ve güncelleme frekansı kullanmak yöntemin kontrolsüz dış ortamda dayanıklılığını düşürmektedir. Sahnede hareketin çok olduğu dimaik alanlar için hareketli nesnenin arka plana modeline dahil edilmesi ihtimalini azaltmak için daha büyük karar eşik değerleri kullanılmalıdır. Diğer taraftan durağan alanlar için daha küçük değişiklikleri hareket olarak tanımlayabilmek adına daha düşük karar eşik değerleri kullanılmalıdır.

Her piksele ait karar eşik değerini belirlemek için pikselin ne kadar sıklıkta değiştiğini temsil eden piksel dinamikliği tanımlanmıştır. Arka plan modelinde güncellenen her bir örneğe $(B(x_{ij}))$ ait minimum uzaklık vektörü $D(x_{ij})$ ve o pikselin güncel değeri piksel dinamikliğini hesaplamak için saklanmıştır.

$$D(x_{ij}) = \{D_1(x_{ij}), \dots, D_k(x_{ij}), \dots, D_N(x_{ij})\} \quad (3.3)$$

$$d_{min}(x_{ij}) = \min_k \text{dist}(I(x_k), B_k(x_{ij})) \quad (3.4)$$

$$D_k(x_{ij}) \leftarrow d_{min}(x_{ij}) \quad (3.5)$$

Her pikselin arka plan modeline ait tutulan minimum güncelleme mesafesi geçmişinin ortalaması $\bar{d}_{min}(x_{ij})$ piksele ait arka plan modelinin dinamikliğini temsil etmektedir. Piksel dinamikliği R_{scale} parametresi ile belirlenen üst sınıra ulaştığında karar eşik değeri $R(x_{ij})$, artma/azalma parametreleri ($R_{inc/dec}$) dikkate alınarak artırılır veya azaltılır (Denklem 3.6).

$$R(x_{ij}) = \begin{cases} R(x_{ij}) \cdot (1 - R_{inc/dec}), & \text{if } R(x_{ij}) > \bar{d}_{min}(x_{ij}) \cdot R_{scale} \\ R(x_{ij}) \cdot (1 + R_{inc/dec}), & \text{else} \end{cases} \quad (3.6)$$

$R(x_{ij})$ değeri (R_{lower}) alt sınır parametresi ile makul sınırlar içinde tutulmuştur. $R_{inc/dec}$ ve R_{scale} kullanıcı tarafından sağlanan sabit parametrelerdir. $R_{inc/dec}$ eşik değerinin değişim şartını sağladıktan sonra ne kadarlık bir yüzde ile değişeceğini temsil etmektedir. R_{scale} ise sahne dinamikliği ile mevcut eşik değeri arasındaki oranı temsil etmektedir. Çok yüksek ve düşük değerlerin kullanılmasının başarımı düşüreceği belirtilmiştir. Bu parametreler doğrultusunda kestirim denklemine tekrar baktığımızda x_{ij} pikseline ait karar eşik değeri pikselin örneklerinin minimum eşleşme uzaklığının ortalamasının R_{scale} katının üstüne çıkması durumunda (sahnenin dinamik olması) $R_{inc/dec}$ oranında azaltılmakta, aksi takdirde (sahnenin durağanlaştığı kabul edilip) artırılmaktadır. Şekil 3.3’de sahneye ait her pikselin $R(x_{ij})$ değerinin farklılığı gösterilmektedir. Sağdaki görüntüde parlak alanlar yüksek $R(x_{ij})$ değerini temsil etmektedir. Sahnede dinamik alanların (rüzgarla hareket eden ağaç yaprakları) $R(x_{ij})$ değerinin daha yüksek olduğu dikkat çekmektedir. Bu da hareketli alanlardaki piksellerin arka plan modeline daha zor dahil edildiğini göstermektedir.



Şekil 3.3 Piksel tabanlı karar eşik değerinin $R(x_{ij})$ değişimi (Hofmann vd. 2012)

3.1.4 Arka plan güncelleme frekansının dinamik güncellenmesi

Karar eşik değeri dışında sahneye göre dinamik olarak değişen diğer parametre arka plan modelinin güncelleme frekansıdır ($T(x_{ij})$). Yüksek $T(x_{ij})$ değeri, pikselin daha seyrek güncellenmesi anlamına gelmektedir. Güncelleme frekansının yüksek olması

arka plan modelinin sahnedeki deęişimlere hızlı adapte olmasını saęlarken hareket ederken bir anda duran fakat tekrar hareket etme ihtimali olan nesnelerin kaybedilmesine sebep olmaktadır. Bu yüzden yazarlar güncelleme frekansının piksel deęerinin deęişim frekansı ile ilişkili olarak dinamik olarak belirlenmesini önermektedir. Hareket olarak yanlış etiketlenen pikseller genellikle hareketli arka plana sahip alanlarda bulunmaktadır. Bu yüzden eşik deęeri kestiriminde kullanılan sahne dinamikliği burada da kullanılmaktadır. Pikselin güncelleme frekansı $(1/T(x_{ij}))$, denklem 3.7’de görüldüğü gibi $T(x_{ij})$ deęerinin hesaplanması sonucu elde edilmiştir.

$$T(x_{ij}) = \begin{cases} T(x_{ij}) + \frac{T_{inc}}{\bar{d}_{min}(x_{ij})}, & \text{if } F(x_{ij}) = 1 \\ T(x_{ij}) - \frac{T_{dec}}{\bar{d}_{min}(x_{ij})}, & \text{if } F(x_{ij}) = 0 \end{cases} \quad (3.7)$$

$(F(x_{ij}) = 1)$ hareketli pikseli, $(F(x_{ij}) = 0)$ ise statik arka plan pikselini temsil etmektedir. Arka plan güncelleme frekansında kullanılan $T(x_{ij})$ deęeri, arka planın yanlış güncellemesini kontrol altında tutmak adına (T_{lower}) ve (T_{upper}) parametreleri ile sınırlandırılmıştır.

Referans alınan yöntemde karar eşik deęerinin $(R_{inc/dec})$ ve güncelleme frekansının (T_{inc}, T_{dec}) artma ve azalma parametreleri sabit deęerler olarak kullanılırken tez kapsamında geliştirilen yöntemde bu deęerler de dinamik olarak deęişmektedir. Sabit artma/azalma deęerlerinin kullanılması sahne deęişimlerine karşı yöntemin yavaş reaksiyon göstermesine ve sahneye geç uyum saęlamasına sebep olmaktadır. Bu noktada geliştirilen yöntem her bir piksel için artma/azalma parametreleri kullanarak sahne deęişimlerine tamamen adapte olmaktadır. Bu parametreler piksel dinamikliğine göre her adımda belirli bir oranda (% 1) deęişecek şekilde tasarlanmıştır. Statik arka plan pikselleri için bu deęerler artırılırken hareketli pikseller için azaltılmaktadır. Bu sayede yöntemin sahne deęişimlerine verdiği tepki deęişimin hızıyla uyumlu olarak deęişmektedir.

$$R_{inc/dec}(x_{ij}) = \begin{cases} R_{inc/dec}(x_{ij}) \cdot 0.99, & \text{if } R(x_{ij}) > \bar{d}_{min}(x_{ij}) \cdot R_{scale} \\ R_{inc/dec}(x_{ij}) \cdot 1.01, & \text{else} \end{cases} \quad (3.8)$$

$$T_{inc}(x_{ij}) = \begin{cases} T_{inc}(x_{ij}) \cdot 0.99, & \text{if } F(x_{ij}) = 1 \\ T_{inc}(x_{ij}) \cdot 1.01, & \text{if } F(x_{ij}) = 0 \end{cases} \quad (3.9)$$

$$T_{dec}(x_{ij}) = \begin{cases} T_{dec}(x_{ij}) \cdot 0.99, & \text{if } F(x_{ij}) = 1 \\ T_{dec}(x_{ij}) \cdot 1.01, & \text{if } F(x_{ij}) = 0 \end{cases} \quad (3.10)$$

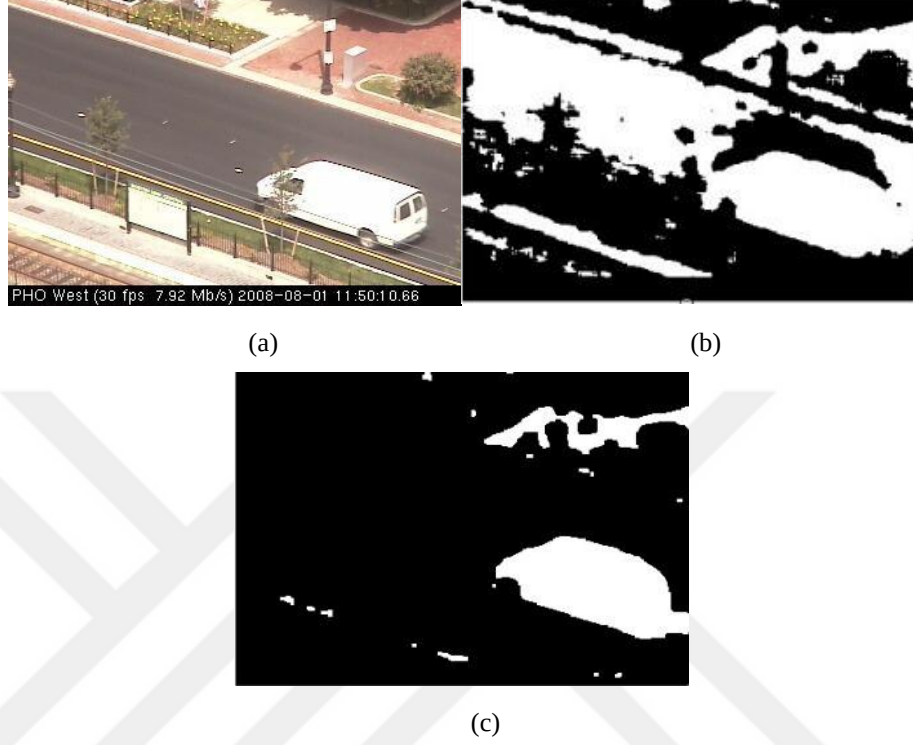
3.1.5 Ani deęişim algılayıcı

Sokak aydınlatmalarının açılıp kapatılması, gün/gece geişleri, bulut geişı veya araç farlarının sebep olduęu ani parlamalar gibi sahnede dominant deęişimlerin olduęu durumlar arka plan modelleme yöntemleri için ciddi sorunlar oluşturmaktadır. Arka plan modelinin ani deęişimlere adapte olması uzun zaman almakta bu durum da tam adaptasyon sağlanana kadar yanlış algılamalar yapılmasına sebep olmaktadır. Geliştirilen arka plan modelleme yöntemine bu durumu engellemek için ani deęişim algılayıcı adı verilen yeni bir modül eklenmiştir. Sahnede kabul edilen maksimum hareket oranı M_{Max} parametresi yardımıyla sahnedeki dominant deęişimler kontrol altında tutulmuştur. Sahnedeki hareketli piksellerin toplamı M_{Max} deęerine ulaştığında hareketli piksellerin güncelleme frekansı T_{lower} deęerine yani en yüksek güncelleme frekansına çekilmiştir. Bu sayede sahnedeki deęişimler mümkün olan en hızlı şekilde arka plan modeli tarafından öğrenilmiştir. Öğrenme işleminin tamamlanmasıyla arka plan güncelleme frekansı zaman içinde optimal seviyeye geri dönmektedir.

$$T(x_{ij}) = \begin{cases} T_{lower}, & \#_{(F=1)} > (M_{Max} \cdot \#_x) \\ T(x_{ij}), & \text{else} \end{cases}, i \in (F(x_{ij}) = 1) \quad (3.11)$$

Ani deęişim algılayıcısının referans yöntemine (PBAS) göre etkisi şekil 3.4'te görölmektedir. Kamerada görüntüsünde ani bir parlama olduktan sonra PBAS (Şekil

3.4.b) ve geliştirilen yöntemin (Şekil 3.4.c) sahnenin yeni durumuna adapte olma performansı görülmektedir.



Şekil 3.4 Ani değişim algılayıcısının etkisi

a. Girdi görüntü, b. PBAS, c. Geliştirilen yöntem

Sonuç olarak referans yöntem üzerinde yapılan değişikliklerle kullanıcı tarafından karar verilmesi gereken daha az parametreye sahip, dış ortam değişimlerine karşı daha dayanıklı bir hareket algılama yöntemi geliştirilmiştir. Geliştirilen yöntem sayesinde dış ortamda yanlış algılama durumunu minimuma indirgeyerek sonraki adımlar olan nesne takibi ve aktivite optimizasyonundaki işlem karmaşıklığını düşürmek amaçlanmıştır. Hareket algılama aşamasında elde edilen sonuçların yeterli güvenilirlikte olması video sinopsis algoritmik akışının en başında yapılan bir hatanın sonuna kadar taşınma ihtimalini minimuma indirmesi açısından önemlidir.

3.2 Hareketli Nesne Takip İşlemi

Tez kapsamında geliştirilen çok kameralı nesne takibi tabanlı çevrimiçi video sinopsis sistemi görüntüye giren herhangi bir nesnenin gerçek-zamanlı olarak takip edilmesini hedeflemektedir. Geliştirilen sistemin genel mimarisinde (Şekil 3.1) hareket algılama yönteminden (Blok 2) gelen herhangi bir nesneyi takip eden bir yöntem ihtiyacı duyulmaktadır.

Literatürde yer alan nesne takip yöntemleri, çevrimiçi ve çevrimdışı olarak sınıflandırıldığında çevrimdışı takip yöntemleri önceden öğrenilmiş veriye ihtiyaç duymaktadır. Bu durum sahnede karşılaşılan herhangi bir aktiviteyi içermesi planlanan video sinopsis sistemi için kısıtlayıcı olmaktadır. Çevrimdışı bir yöntemin kullanılması video sinopsise dahil edilecek aktivite tiplerinin önceden belirlenip sistemin buna göre eğitilmesini gerektirmektedir. Bu durum oluşturulacak video sinopsisin kapsamının daraltılmasına sebep olacaktır. Bu yüzden gerçek zamanlı çalışan ve çevrimdışı bilgiye ihtiyaç duymayan seçici korelasyon filtresi tabanlı bir yöntem geliştirilmiştir. Geliştirilen yöntem aynı anda birden fazla nesneyi takip edebilmektedir.

Bölüm 2.1.2’de bahsedildiği gibi çevrimiçi nesne takip yöntemlerinde en başarılı sonucu korelasyon filtresi tabanlı yöntemler elde etmektedir. Literatürde nesne takibinin farklı sorunlarına odaklanmış bir çok korelasyon filtresi tabanlı yöntem mevcuttur. MOSSE (Bolme vd. 2010), CSK (Henriques vd. 2012) gibi yöntemler nesnenin sadece yer değiştirmesine bakarak kestirim yapmaya çalışırken; DSST (Danelljan vd. 2014), SRDCF (Danelljan vd. 2015) gibi yöntemler ölçek değişimlerinin önemini vurgulayıp yer değiştirme filtresinin yanında bağımsız bir ölçek kestirim filtresi kullanmaktadır. Bu sayede nesnenin konumunu kestirirken ölçek kestirimi de yapılmaktadır. Bu yöntemlerin, sadece yer değiştirme kestirimi yapanlara göre daha iyi sonuç verdiği görülmektedir. Bu yüzden çalışma kapsamında geliştirilen yöntem de yer değiştirme ve ölçek kestirimi yapan iki korelasyon filtresini temel almaktadır (Baskurt ve Samet 2017b, 2018a). Yer değiştirme filtresi ile hedef nesnenin bir sonraki video karesindeki konumu kestirilirken sonrasında o konuma ait ölçek kestirimi yapılmaktadır.

Bölüm 3.2.1’de öncelikle temel alınan korelasyon tabanlı nesne takip yönteminin detayları açıklanmıştır. Sonrasında bölüm 3.2.2’de geliştirilen çoklu nesne takip yönteminin mevcut literatüre katkısı ve teknik detayları açıklanmıştır.

3.2.1 Korelasyon filtresi tabanlı nesne takibi

Nesne takibi için kullanılan korelasyon filtresi girdi olarak görüntünün Histogram of Oriented Gradients (HOG) özniteliklerini almaktadır. Bu bölümde verilen hedef nesne ile korelasyon filtresinin nasıl oluşturulduğu, yeni video karelerinde filtrenin nasıl çevrimiçi olarak güncellendiği ve ölçek kestirimi için kullanılan filtre detayları açıklanmıştır.

Korelasyon filtresinin oluşturulması: Korelasyon filtresi tabanlı takip yöntemlerinde hedef nesneyi çevreleyen çerçeve (f) kullanıcı tarafından girdi olarak verilmektedir. Verilen hedef nesne görünümü bir filtre ile modellenmektedir. Sonraki video karelerinde filtre hedef nesnenin önceki karedeki konumu etrafındaki arama penceresinde ilintilendirilerek (korelasyon) takip edilmektedir. En yüksek korelasyon çıktısına sahip konum, nesnenin yeni konumu olarak belirlenmekte ve filtre güncellenmektedir.

Bu işlemlerin ilk adımında verilen girdi ile çevrimiçi öğrenme işlemi gerçekleştirilmektedir. Başlangıç çerçevesinin rastgele 8 afin (affine) dönüşümü hesaplanarak girdi serisi (f_i) oluşturulmaktadır. Her bir girdi örneği için, tepe noktası örneğin merkezinde olan bir “Gaussian” ($\sigma = 2.0$) fonksiyonu tanımlanmaktadır. Bu fonksiyonun girdi serisi ile çarpımı sonucunda beklenen çıktı serisi (g_i) oluşturulmaktadır.

Sonrasında girdi ve oluşturulan çıktı serileri arasındaki korelasyon filtresi (h) hesaplanmaktadır. Konvolüsyon (Convolution) teoremine göre uzamsal düzlemde korelasyon, frekans düzleminde eleman bazında (element-wise) çarpıma karşılık geldiği için tüm korelasyon işlemleri frekans düzleminde gerçekleştirilip sonra tekrar uzamsal düzleme dönüştürülmektedir (Bracewell 1986). Bu sayede işlem hızı artarak takip

yönteminin gerçek-zamanlı çalışması sağlanmaktadır. Uzamsal düzlemdeki girdi görüntüleri ($F = \mathcal{F}(f)$) ve filtre ($H = \mathcal{F}(h)$) “Fast Fourier Transform (FFT) (Press vd. 1992)” ile “Fourier” düzlemine aktarılmaktadır. ' $\odot$ ' sembolü frekans düzlemde eleman bazında çarpım ve '*' sembolü karmaşık eşleniği (complex conjugate) temsil ettiğinde korelasyon filtresi denklem 3.12'deki gibi olmaktadır.

$$G = F \odot H^* \quad (3.12)$$

$$H_i^* = \frac{G_i}{F_i} \quad (3.13)$$

Bu durumda işlemsel darboğaz, FFT ve geri dönüşümü gerçekleştiren “inverse-FFT” işlemine indirgenmektedir. Yani P sayıda pikselden oluşan bir arama penceresi kullanan takipçi için hesaplama karmaşıklığı $O(P \log P)$ kadar olmaktadır.

FFT dönüşümünde görüntü ve filtre, tor (torus) yapısı üzerine eşlenmektedir. Bu yapıda görüntünün üst kenarı ile alt kenarı, sağ kenarı ile sol kenarı birleşmektedir. Bu sayede uzamsal düzlemde çerçevenin yerini değiştirmek yerine frekans düzleminde tor etrafında döndürülmektedir. Eğitim girdilerini (F_i) beklenen çıktılara (G_i) ilintileyen filtreyi bulmak için denklem 3.14'te görüldüğü gibi güncel çıktı ($F_i \odot H^*$) ile beklenen çıktısı (G_i) arasındaki hatanın kareler toplamını minimuma indirgeyen filtre hesaplanmaktadır.

$$H = \min_H \sum_i |F_i \odot H^* - G_i|^2 \quad (3.14)$$

Denklem 3.14'teki minimizasyon problemi, korelasyon filtresi H 'nin her bir elemanı için ayrı çözüldüğünde denklem 3.15'teki gibi olmaktadır. Bu işlemin detayları Ek 2'de paylaşılmıştır.

$$H = \frac{\sum_i F_i \odot G_i^*}{\sum_i F_i \odot F_i^*}, \quad H^* = \frac{\sum_i G_i \odot F_i^*}{\sum_i F_i \odot F_i^*} \quad (3.15)$$

Denklem 3.15'te elde edilen son hali ile korelasyon filtresinin pay kısmı girdi ile beklenen çıktı arasındaki korelasyondan, payda ise girdinin enerji spektrumundan oluşmaktadır.

Çevrimiçi güncelleme: Takip edilen nesnenin görünümü; rotasyon, ölçek, duruş, ışık koşulları, bakış açısı gibi etkenlerden dolayı sürekli değişmektedir. Geliştirilen yöntemin bu değişimlere karşı mümkün oldukça dayanıklı olması gerekmektedir. Bunun için verilen başlangıç çerçevesiyle eğitilen korelasyon filtresi takip boyunca nesnenin görünümü ile güncellenmektedir. i numaralı video karesinin işlendiği durumda korelasyon filtresi denklem 3.16'daki gibi pay ve paydaya ayrıştırıldığında, denklem 3.17 ve 3.18'teki gibi güncellenmektedir. Buna göre filtre, güncellenme katsayısı (η) oranında görünüm değişimlerine adapte olurken aynı zamanda nesne geçmişiyile tutarlı kalmaktadır.

$$H_i^* = \frac{A_i}{B_i} \quad (3.16)$$

$$A_i = \eta G_i \odot F_i^* + (1 - \eta) A_{i-1} \quad (3.17)$$

$$B_i = \eta F_i \odot F_i^* + (1 - \eta) B_{i-1} \quad (3.18)$$

Korelasyon filtresinin başarımını ölçmek için tepe noktasının şiddetini temsil eden “Peak to Sidelobe Ratio (PSR)” değeri kullanılmaktadır. Bu değeri hesaplamak için korelasyon çıktısı g , en yüksek değere sahip olduğu tepe bölgesi ve geri kalan değerleri kapsayan “sidelobe bölgesi” olmak üzere ikiye ayrılmaktadır. Tepe bölgesi, en yüksek değerler etrafındaki 11×11 boyutundaki alan olarak kabul edilmektedir. Bu durumda PSR değeri denklem 3.19'daki gibi hesaplanmaktadır.

$$PSR = \frac{g_{max} - \mu_{sl}}{\sigma_{sl}} \quad (3.19)$$

Bu denklemde g_{max} tepe bölgesine ait değerleri temsil ederken μ_{sl} ve σ_{sl} “sidelobe bölgesinin” sırasıyla ortalamasını ve standart sapmasını temsil etmektedir. Güvenilir PSR değeri literatürdeki konu ile ilgili çalışmalarda farklılık göstermekle birlikte korelasyon filtresi sadece PSR değeri güvenilir seviyede ise aktif konum ile güncellenmektedir. Aksi durumda aktif konum ile güncelleme gerçekleşmemektedir. Kullanılan güvenilir PSR değeri aralığı bölüm 4.2’de tartışılmaktadır.

Ölçek kestirimi: Geliştirilen takip yönteminde iki adet korelasyon filtresi oluşturulmaktadır. Bunlardan ilki nesnenin yer değişim kestirimini yaparken ikincisi ölçek kestirimini yapmaktadır. Algoritmik akışta her video karesi için nesne konum kestiriminden sonra belirlenen konumda ölçek kestirimi yapılmaktadır. Yer değiştirme filtresinde hedef nesnenin bir önceki video karesindeki konumunun etrafında arama yapılırken ölçek filtresinde oluşturulan ölçek piramidi üzerinde arama yapıp en düşük hataya sahip ölçek, nesne ölçeği olarak belirlenmektedir (Denklem 3.14 ile hesaplanmaktadır). Ölçek kestirimi sonucunda ölçek filtresi de konum kestirim filtresiyle aynı şekilde yukarıda belirtildiği gibi güncellenmektedir. Bu sayede takip yöntemi nesnenin kameraya uzaklığının değişiminden kaynaklanan ölçek değişimine daha dayanıklı olmaktadır.

3.2.2 Alternatif korelasyon filtreleri kullanarak gürbüz nesne takibi

Tez çalışması kapsamında korelasyon filtresi tabanlı yöntemler temel alınarak birden fazla nesneyi aynı anda gerçek-zamanlı olarak takip eden yeni bir yöntem geliştirilmiştir. Geliştirilen yöntem mevcut yöntemlerden farklı olarak her bir hedef nesnenin farklı görünümelerini temsil eden alternatif filtreler barındırmaktadır. Bu sayede nesnelerin görünüm değişimlerine karşı yöntem dayanıklılığı artmakta ve tez kapsamında bir sonraki aşamada geliştirilen çok kameralı nesne takibine girdi oluşturulmaktadır. Son olarak da mevcut çalışmalarda önemli bir sorun olan takip çerçevesinin nesnenin üzerinden kayması (drift) durumu, girdi olarak alınan hareket

bilgisi ile minimuma indirilmektedir. Bu bölümün alt başlıklarında bahsedilen katkılar detaylı olarak anlatılmıştır.

Hareket tetikli çoklu nesne takibi: Sahnede hareket olarak tespit edilen her bir blob (komşu hareketli piksellerin bağlı bileşen etiketleme yöntemi ile hareketli nesne haline getirilmiş hali) için bir takipçi oluşturulmaktadır. Nesne görüntüde olduğu sürece kendisi için oluşturulmuş takipçi ile ilişkilendirilmekte ve tüm bilgileri burada tutulmaktadır. Her bir takipçi ($\mathcal{T}$); ana korelasyon filtresi (H), görünüm değişimlerine karşı yöntem dayanıklılığını artıran alternatif korelasyon filtreleri $HA_a (a \in 1, 2, \dots, A)$, hedef nesnenin geçmiş konumları (XxY boyutunda bir video karesinde; $Tr_l (l \in 1, 2, \dots, L), l = (x \in 1, 2, \dots, X; y \in 1, 2, \dots, Y)$) ve aktif/pasif durumunu temsil eden S (status) değişkeninden oluşmaktadır.

$$\mathcal{T} = \{H, HA_{1,2,\dots,A}, Tr_{1,2,\dots,L}, S\} \quad (3.20)$$

Görüntüdeki tüm nesneleri temsil eden takipçiler, oluşturulan takipçi havuzunda ($P = \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_N\}$) tutulmaktadır. Görüntüde aktif olarak bulunan bir nesne ile ilişkilendirilmiş takipçi, havuzda aktif olarak etiketlenmektedir. Takip ettiği nesne görüntüden çıkan veya engellenen takipçiler havuzdan silinmeden önce pasif duruma geçmektedir. Bu takipçiler $t_{passive}$ süresi boyunca havuzda tutulmaya devam edilmektedir. Yeni gelen hareket blobları mevcut takipçiler ile eşleştirilirken aktif ve pasif olmak üzere tüm havuz içinde aranmaktadır. Bu sayede görüntüsü başka bir nesne tarafından engellenen nesneler $t_{passive}$ süre içinde tekrar görüntüye girdiklerinde aynı takipçi ile ilişkilendirilip takibine devam edilmektedir.

Geliştirilen yöntemin mevcut korelasyon filtresi tabanlı yöntemlerden önemli farklarından biri birden fazla nesneyi aynı anda gerçek-zamanlı olarak takip edebilmesidir. Geliştirilen nesne takip yöntemi girdi olarak hareket algılama sonucu elde edilen hareket bloblarını almaktadır. Alınan hareket bloblarının sayısı ve mevcut takipçilerin durumuna göre algoritma 1 uygulanmaktadır.

Algoritma 1. MultiObjectTracker

Initialize active $P \leftarrow \emptyset$ and inactive $P_{inactive} \leftarrow \emptyset$ tracker pools

```

1:   for each video frame  $F$  do
2:       Initialize updated tracker list  $T_{updated} \leftarrow \emptyset$ 
3:       Get motion blob list  $B$  by applying PBAS to  $F$ 
4:       for each motion blob  $b$  in  $B$  do
5:           Initialize matched tracker list  $T_{matched} \leftarrow \emptyset$ 
6:            $T_{matched} \leftarrow$  call Find-Match with  $b, P$ 
7:           if  $T_{matched}$  equal  $\emptyset$  then
8:                $T_{matched} \leftarrow$  call Find-Match with  $b, P_{inactive}$ 
9:           end if
10:          if  $T_{matched}$  not equal  $\emptyset$  then
11:              Find best matched candidate tracker  $T_c$  using Eq. 3.21
12:              call Update-Tracker for  $T_c$  with  $b$  using Eqs. (3.16-3.18)
13:              Push  $T_c$  to  $T_{updated}$ 
14:          else
15:              Create new tracker with  $b$  using Eq. 3.12
16:          end if
17:      end loop
18:      call Update-Tracker for each  $T$  in  $P$  and  $T$  not in  $T_{updated}$ 
19:      Update  $P_{inactive}$ 
20:  end loop

```

Algoritma 2. FindMatch

Input: motion blob b , tracker pool P

Output: matched tracker list $T_{matched}$

```

1:   for each tracker  $T$  in  $P$  do
2:       if rectangle( $b$ ) intersect rectangle( $T$ ) then
3:           Push  $T$  to  $T_{matched}$ 
4:       end if
5:   end loop

```

Algoritma 3. UpdateTracker

Input: tracker T , motion blob b

Output: tracker T

```

1:   Update correlation filter  $H$  using Eqs. (3.16-3.18)
2:   Update alternative correlation filters  $HA$  using Eq. 3.23
3:   Check boundary drifting using Eq. 3.16
4:   if boundary drifting exists then
5:       Apply boundary correction using Eqs. (3.25-3.27)
6:   end if

```

Buna göre herhangi bir video karesinde girdi olarak alınan hareket blobu varken nesne takip yöntemi dahilinde herhangi bir takipçi yoksa bu hareketli nesnelerin sahneye yeni

girdiği varsayımı yapılarak her biri için bölüm 3.2.1’de açıklanan korelasyon filtresi tabanlı bir takipçi oluşturulmaktadır.

Diğer senaryoda hem hareketli blob, hem de mevcut takipçiler varsa; mevcut takipçilere ait hedef nesne konumları, hareketli blob konumları ile karşılaştırılmaktadır. Çerçeveleri kesişen blob ve takipçiler eşleştirilip ilgili takipçinin yer değiştirme ve ölçek korelasyon filtreleri denklem 3.17 ve 3.18 ile güncellenmektedir. Bir hareket blobu birden fazla takipçi ile kesişiyorsa denklem 3.14 sonucunda en yakın korelasyon çıktısını veren takipçi ile eşleştirilip güncellenmektedir. Herhangi bir takipçi ile eşleşmeyen hareket blobları öncelikle kesişim şartı aranmadan pasif takipçiler ile eşleştirilmeye çalışılmaktadır. Pasif takipçilerden birinin korelasyon sonucunun belirlenen hata eşik değerinden ($E(r_{ror})_{conf}$) daha fazla yaklaşması durumunda bu nesnenin daha önce takip edilen fakat görüntüsü engellenmiş bir nesne olduğuna karar verilmektedir. Eşleşen takipçi güncellenerek aktif duruma geçmektedir. Yeni hareket blobunun bu aşamaya kadar herhangi bir takipçi ile eşleşmemesi durumunda yeni takipçi oluşturulmaktadır.

Yeni hareket blobu gelmemesine rağmen mevcut takipçiler varsa hareket bilgisinden bağımsız olarak temel alınan yöntemdeki gibi takipçiler güncellenmektedir. Güncellenme işlemlerinin tamamında korelasyon filtresi sonucunda elde edilen PSR değerinin kabul edilebilir seviyede (PSR_{min}) olması dikkate alınmaktadır. PSR değeri düşük olan –güvenilir olmayan- eşleşmelerde ilgili takipçi güncellenmemektedir. Bu olasılık nesnenin görüntüden çıkması veya görünümünün önemli ölçüde değişmesi durumunda gerçekleşmektedir. Nesnenin görüntüden çıkması veya engellenmesi durumuna karşı ilgili takipçi yukarıda bahsedildiği gibi pasif durumda bir süre bekletilmektedir. Bu sayede nesnelerin görünür olduktan sonra tekrar aynı takipçi ile takibe devam etmeleri sağlanmaktadır. Diğer olasılıkta nesnenin görünümünün önemli ölçüde değişmesi literatürde yer alan takip yöntemleri için ciddi bir sorundur. Bu sorunun alternatif takipçi yöntemi ile nasıl minimum seviyeye indirildiği bir sonraki alt başlık olan alternatif filtrelerin oluşturulması bölümünde anlatılmaktadır.

Oluşturulan takipçi havuzu sayesinde nesnelerin kesişmesi durumunda kesişen nesnelere ait takipçi bilgileri karşılaştırılarak en iyi kestirimin yapılması hedeflenmektedir. Kesişen nesnelerin birbiri ile karıştırılması video sinopsiste bir anda belirip kaybolan hayalet nesnelere sebep olmaktadır. Takipçilerin birbiri ile haberleşerek kesişme durumundaki performansın iyileştirilmesi ile video sinopsiste karşılaşılan bu sorun minimuma indirilmektedir. Bu durumda tek takipçi güncellemesinde kullanılan minimum hatalı korelasyon çıktısı yeterli olmamaktadır. Kesişen takipçilerden birden fazlasının aynı konumu takip sonucu olarak belirlemesi durumunda korelasyon hata çıktısı en düşük olan takipçi doğru takipçi olarak belirlenmekte (Denklem 3.21) ve güncellenmektedir.

$$H = \min_H \{H_1, H_2, \dots, H_K\} \quad (K = \text{Kesişen takipçi sayısı}) \quad (3.21)$$

Diğer kesişen takipçiler güncellenmemektedir. Bu sayede kesişme durumunda görüntüsü engellenen nesnelere ait takipçi filtrelerinin kendisine ait olmayan veri ile güncellenmesi engellenmektedir.

Alternatif filtrelerin oluşturulması: Geliştirilen çok kameralı video sinopsis sisteminde kameraların bakış açısı farklılıklarından dolayı aynı hedef nesneye ait farklı yönlerden alınan görüntülerin başarılı bir şekilde eşleştirilmesi gerekmektedir. Mevcut korelasyon filtresi tabanlı nesne takip yöntemleri genellikle hedef nesnenin tek bir modeli üzerinden kestirim yapmaktadır. Geliştirilen çok kameralı sistem dikkate alındığında nesneye ait tek bir modelinin kullanılması yetersiz kalmaktadır. Bu yüzden geliştirilen takip yöntemi, korelasyon filtresinin güvenilir sonuçlar verdiği zamanlara ait örnekler üzerinden alternatif filtreler oluşturmaktadır. Takibin kaybolduğu veya PSR çıktısının belirlenen güvenilir seviyenin (PSR_{min}) altına düştüğü durumlarda nesne görünümünün değişmiş olma ihtimaline karşı alternatif filtreler de devreye sokulmaktadır. Denklem 3.14'te belirtilen korelasyon işlemi denklem 3.22'daki gibi tüm alternatif filtrelere de uygulanmaktadır. Bu filtrelerden birinin ana filtreden daha düşük korelasyon hata çıktısı vermesi durumunda ana filtre ile yer değiştirmektedir. Bundan sonraki takip yeni filtre üzerinden devam etmektedir.

$$H = \min_H \{H, HA_a\} \quad (3.22)$$

Yöntemin gerçek-zamanlı çalışma durumu da göz önüne alınarak alternatif korelasyon filtresi sayısı a ($HA_a (a \in 1, 2, \dots, A)$) ile sınırlandırılmaktadır. HA_a listesine yeni bir filtre eklerken yeni filtrenin yeteri kadar güvenilir olmasına ve mevcut alternatif filtrelerden farklı bir görünümü temsil etmesine dikkat edilmektedir. Güvenilirlik için listede karşılaştırılan alternatif filtrenin PSR değerinden daha yüksek PSR değerine sahip olması gerekmektedir. PSR_{min} değerinin altındaki filtreler alternatif filtre listesine eklenmemektedir. Mevcut alternatif filtrelerden farklı bir görünüme sahip olması için korelasyon hata çıktısı kullanılmaktadır. Hatanın yine güvenilir hata seviyesinden (E_{conf}) düşük olmasına bakılmaktadır. Fakat bu sefer güvenilir sınırlar içinde hatası büyük olan filtre farklı görünümü temsil ettiği için alternatif filtreler arasında eklenmektedir (Denklem 3.23).

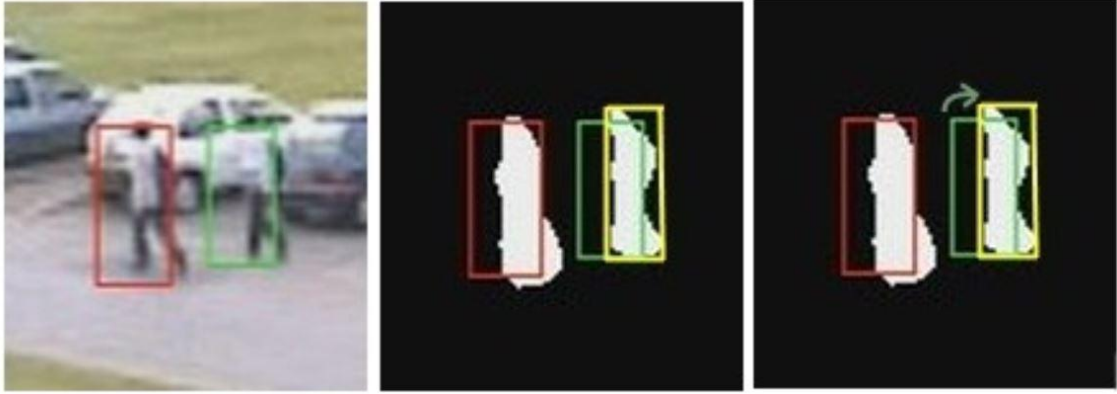
$$\begin{aligned} PSR_{H_C} &> PSR_{HA_a}, \\ |H - H_C| &< E_{conf}, \\ H_C &> HA_a \end{aligned} \quad (3.23)$$

Bu sayede yukarıda bahsedildiği gibi çok kameralı video sinopsis sisteminin altyapısının oluşturulmasının yanında geliştirilen yöntem tek kamerada nesne takip yöntemlerinde ciddi sorunlar yaratan ölçek, oryantasyon değişimi, görüntü engellenmesi gibi durumlarda takibin kaybolma ihtimalini minimuma indirmektedir.

Çerçeve kayma problemi: Takip esnasında nesne çerçevesinin zamanla nesne merkezinin dışına kayması (drift), mevcut nesne takip yöntemlerinde karşılaşılan önemli sorunlar arasında yer almaktadır. Çerçeve kayması sonucunda arka plana ait öznelilikler çerçeve içine dahil edildiği için takipçinin güvenilirliği azalmaktadır. Bir süre sonra çerçeve içinde arka planın baskın hale gelmesi durumunda takipçi nesneyi bırakıp arka planda takılı kalmaktadır.

Geliştirilen takip yönteminde girdi olarak alınan hareket bilgisi kayma durumunu minimuma indirmek için kullanılmaktadır. Hareket bilgisi görsel öznelik içermediği için tek başına nesne takibi için yetersiz kalmaktadır. Fakat sürüklenme durumu gibi görsel takibin yetersiz kaldığı durumlara destek olabilmektedir. Hareket sonucu elde edilen blob (b) ile nesne takip sonucu elde edilen hedef çerçevenin (g) merkezleri arasında hedef çerçeve genişliğine göre belirlenen maksimum sapma oranından ($m_{dev(iation)}$) büyük fark olması durumunda (Denklem 3.24) korelasyon filtresi hareket blobuna yaklaşacak şekilde zorlanmaktadır (Şekil 3.5).

$$genişlik(g) > |merkez(b) - merkez(g)| > m_{dev} * genişlik(g) \quad (3.24)$$



Şekil 3.5 Korelasyon filtresi kestirimi ile hareket blobu arasındaki fark.

Soldan sağa sırasıyla takip sonucunda elde edilen nesne çerçeveleri, nesne çerçevelerinin hareket verisi üzerinde gösterimi ve nesne çerçevesinin hareket çerçevesine (sarı çerçeve) yakınsanması

Bunun için hareket blobuna ait çerçeve ile bölüm 3.2.1’de açıklandığı gibi yeni bir güncelleme korelasyon filtresi ($H_{u(pdate)}$) oluşturulmaktadır. Denklem 3.16-3.18 aşağıdaki gibi denklem 3.25-3.27 olarak tekrar yazılmaktadır. Mevcut korelasyon filtresi i . video karesinde kendi değeri yerine H_u kullanılarak güncellenmektedir. Böylelikle H_u güncelleme oranı (η) kadar mevcut korelasyon filtresine etki ederken mevcut filtre geçmişi korunmaya devam etmektedir. Sonraki video karelerinde de hareket blobu ile hedef konum arasında merkez sapmasının devam etmesi durumunda aynı işlem uygulanarak sapmanın adım adım düzeltilmesi sağlanmaktadır.

$$H_i^* = \frac{A_u}{B_u} \quad (3.25)$$

$$A_i = \eta G_u \odot F_u^* + (1 - \eta) A_{i-1} \quad (3.26)$$

$$B_i = \eta F_u \odot F_u^* + (1 - \eta) B_{i-1} \quad (3.27)$$

Çerçeve kayma durumuna karşı geliştirilen bu yaklaşım nesne takip yönteminin desteklemek için kullanılmaktadır. Bu yaklaşımın nesne takibi üzerinde herhangi bir başarımla düşüşüne sebep olması istenmektedir. Bu yüzden hareket algılama yönteminden birden fazla nesnenin tek bir hareket blobu olarak gelme durumuna karşı hareket blobu ile filtre çerçevesi arasında hedef çerçeve genişliğinden büyük fark olduğu durumda (Denklem 3.24) bu işlem devre dışı bırakılmaktadır. Aksi takdirde hareket algılama sonucu doğru ayrıştırılamayan blobların nesne takibini olumsuz etkileme durumu oluşmaktadır.

3.3 Çok Kameralı Nesne Takip İşlemi

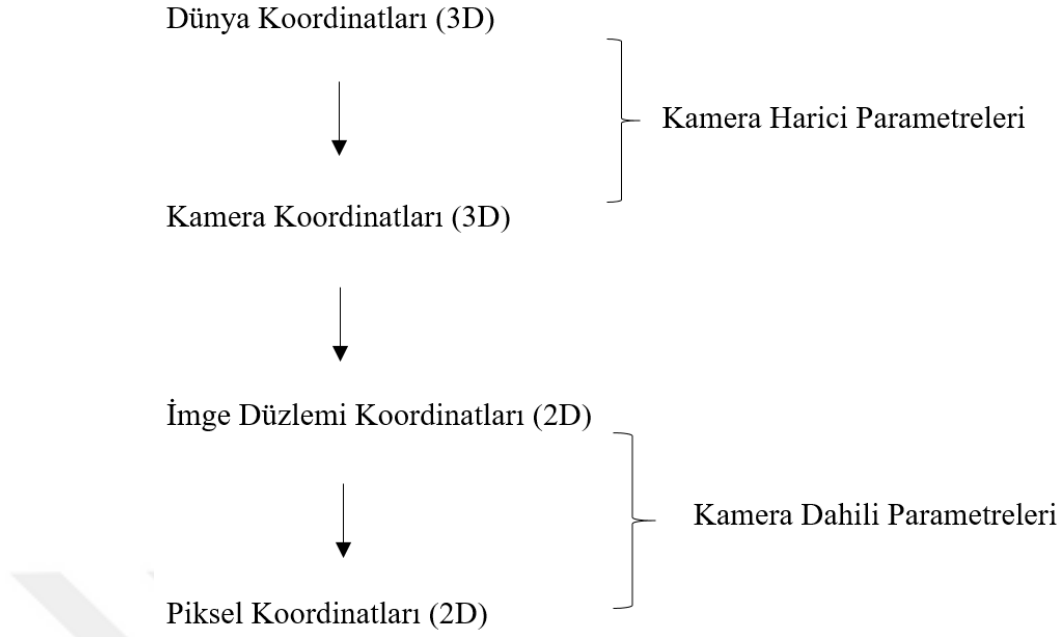
Tez kapsamında uygulanan çok kameralı nesne takip yöntemi, kamera kalibrasyon bilgileri kullanılarak takip edilen nesnelerin coğrafi konumlarının hesaplanması ile desteklenmiştir. Bu sayede nesnelerin kameralar arasında eşleştirme başarımının artırılması amaçlanmıştır. Bölüm 2.1.3'te anlatıldığı gibi sahne ve kamera ile ilgili sahip olunan veriye göre piksel koordinatları ile dünya koordinatları arasında dönüşüm yapan farklı yöntemler mevcuttur. Bu çalışma kapsamında kullanılan veri kümesinde kameranin dâhili ve harici parametreleri sağlandığı için çalışma kapsamında bu parametreleri kullanarak kamera kalibrasyonu ile piksel ve coğrafi koordinatlar arasında ilişki sağlanmaktadır. 3 boyutlu dünya koordinatlarından 2 boyutlu piksel koordinatlarına geçişte 3. boyuta ait bilgi kaybolduğu ve sadece kamera kalibrasyonu ile bu verinin geri getirilemeyeceği için nesnelerin düz bir zemin ($Z = 0$) üzerinde hareket ettiği varsayılmaktadır. Kalibrasyon tamamlandıktan sonra veri kümesinde yer alan kameraların görüş alanlarının kuşbakışı haritada coğrafi olarak nereye denk geldiğini tespit etmek için kamera görüntülerinde nesnelerin hareket edebileceği alanlar

işaretlenmektedir. Aynı zamanda kamera görüntülerinin kesişim alanları coğrafi olarak hesaplanmaktadır. Bu bilgiler nesnelerin kamera geçişlerinde kestirimini ve eşleşmesini kolaylaştırmak için takip yönteminde kullanılmaktadır.

Bölüm 3.2’de bahsedilen korelasyon filtresi tabanlı nesne takip yöntemi çok kameralı sisteme uygulanacak şekilde genişletilmiştir. Nesnelerin sonraki video karelerinde durumunu kestirmek için “Kalman” filtresi ile desteklenmiştir. Bu sayede herhangi bir kamerada algılanan nesnenin konum-zaman ilişkisinden dolayı eşleşebileceği takipçiler sınırlandırılmıştır. Bu şekilde gereksiz karşılaştırma işlemi minimuma indirilerek yöntem çalışma zamanının iyileştirilmesi sağlanmıştır. Kameraların kesişim alanında aynı anda benzer konumlarda algılanan nesnelere ait takipçiler aynı nesneyi takip etme ihtimali olduğuna dair işaretlenmiştir. Bu takipçilerden bir tanesi kamera görüntülerinden çıktığında yapılan karşılaştırma sonucunda aynı nesne olduğuna karar verilirse ilgili takipçiler birleştirilmiştir. Bu sayede takibin mümkün oldukça kesintisiz olması sağlanmaktadır. Takibi sonuçlanan nesneler video sinopsise girdi olarak verileceği için nesne izlerinin kesintisiz olması sonraki aşamada yapılacak optimizasyonu kolaylaştırmasının yanında video sinopsisin semantik bütünlüğünü desteklemektedir.

3.3.1 Kamera kalibrasyonu

3 boyutlu düzlemden 2 boyutlu piksel düzlemine aktarılan bir noktanın sadece kamera kalibrasyonu kullanılarak tekrar 3 boyutlu düzleme aktarılması mümkün değildir. 2 boyuta geçme esnasında derinlik ve ölçek bilgisi kaybolmaktadır. Derinlik ve ölçek algılaması için uygun sensörlerin (derinlik kameraları) kullanılması veya stereo vision gibi çözümler mevcuttur. Kamera kalibrasyonu ile görüntü üzerindeki bir noktanın 3 boyutlu dünya koordinatlarında karşılığını bulmak için zeminin her noktada eşit yükseklikte ($Z = 0$) olduğu varsayımı üzerinden gidilmelidir. Bu durumda noktayı, 3 boyutlu dünya koordinatlarından imge koordinatlarına taşıyan dönüşüm matrisinin tersi kullanılarak görüntüdeki noktalar ortak koordinat sistemine taşınabilmektedir. Dönüşüm adımları şekil 3.6’daki gibi tanımlanmaktadır.



Şekil 3.6 Dünya ve kamera koordinat sistemleri arasındaki dönüşüm

Harici (extrinsic) parametreler, kamera pozisyonu ve yönelimini tanımlayan parametrelerdir. Dünya koordinat merkezi ile kamera koordinat merkezi arasında öteleme (translation) ve yönelim (rotation) dönüşümü için kullanılmaktadır. Dahili (intrinsic) parametreler ise kameranın geometrik, dijital ve optik özelliklerini yansıtmaktadır. Odak uzaklığı, optik merkez, “skew” ve dünya koordinat sistemi birimi ile piksel değerleri arasındaki ilişkiden oluşmaktadır. 3 boyutlu dünya koordinatlarındaki (x_w, y_w, z_w) noktasını imge düzleminde (u, w) noktasına dönüştürmek için denklem 3.28 kullanılmaktadır.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K[R \ T] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (3.28)$$

R , kameranın dünya referans sistemine göre yönelimini temsil eden 3 elemanlı vektördür. Her bir elemanı kendi ekseninde yönelimi radyan cinsinden temsil etmektedir. T , kameranın dünya referans sistemine göre öteleme bilgisini temsil eden 3 boyutlu matristir. K ise kameranın dahili parametrelerini temsil eden 3 boyutlu matristir.

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.29)$$

(c_x, c_y) , optik merkezi (principal point) temsil etmektedir. Skew piksel düzleminde x ve y ekseninin açısal kaymasını temsil etmektedir. x ve y düzlemlerinin birbirine dik olması durumunda s (*skew*) değeri 0 olmaktadır. f_x ve f_y değerleri, odak uzaklığının (F) x ve y eksenlerinde her bir dünya koordinat biriminin kaç piksele karşılık geldiği ile çarpımına eşittir.

$$f_x = F * s_x \quad (3.30)$$

$$f_y = F * s_y \quad (3.31)$$

F , genellikle mm cinsinden, f_x ve f_y piksel cinsinden tanımlanmaktadır. Denklem 3.28'deki $K[R \ T]$ çarpımı, $tform$ matrisi olarak tanımlanırsa bu matrisin tersinin piksel koordinatı ile çarpımı dünya koordinatındaki ($Z = 0$) karşılığını vermektedir.

$$\begin{bmatrix} x_w \\ y_w \\ 0 \\ 1 \end{bmatrix} = tform^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (3.32)$$

Belirtilen kamera kalibrasyonu yardımı ile her kamera görüntüsünde takip edilen nesnelerin ortak koordinat sistemine göre konumları belirlenmektedir. Bu sayede çok kameralı nesne takibi coğrafi konum bilgisi ile desteklenmektedir.

Kalibrasyon işlemi tamamlandıktan sonra her kamera bakış alanında nesnenin hareket edebileceği alanlar elle belirlenmiştir. Şekil 3.7'de kırmızı ile işaretlenen alanlar piksel koordinatından coğrafi koordinat sistemine dönüştürülerek kameraların coğrafi görüş alanları (FOV: field of view) hesaplanmıştır.



Şekil 3.7 Kameraların görüş alanlarının sınırları

Her kameraya ait coğrafi görüş alanını (FOV) temsil eden çokgenler denklem 3.33'deki gibi tanımlanmaktadır.

$$FOV_C = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\} \quad (3.33)$$

Tüm kameralara ait coğrafi görüş alanlarını tanımlayan çokgen serisi $FOV = \{FOV_{C1}, FOV_{C2}, \dots, FOV_{CN}\}$ tanımlandıktan sonra kameraların kesişim bölgeleri de hesaplanmaktadır. Görüş alanlarının kesişimlerinin bilinmesi aynı nesnenin birden fazla takipçi ile ilişkilendirilmesinin önüne geçmektedir. Bölüm 3.3.2'de detaylı olarak anlatıldığı gibi kesişim bölgelerindeki nesneler için her bir kameradaki görüntüyü temsil edecek şekilde birden fazla takipçi oluşturulmasına izin verilmektedir. Nesnenin kesişim alanından çıkması ile sahip olduğu tüm takipçiler birleştirilmektedir. Her kamera C için diğer kameralar ile görüş alanı kesişimini temsil eden bir liste $Int_C = \{FOV_i\} (i \in 1, 2, \dots, N; i \neq C)$ tutulmaktadır. FOV_i , C kamerası ile i kamerasının kesişim alanını temsil eden çokgenin sınır noktalarından oluşmaktadır. Kesişim alanlarını hesaplarken

işlem kolaylığı için kamera görüş alanları coğrafi olarak 10 cm^2 'lik hücelere bölünerek taranmaktadır.

Şekil 3.8'de kamera kalibrasyonunda gerçekleştirilen işlemlerin akış diyagramı görülmektedir.



Şekil 3.8 Kamera kalibrasyonu akış diyagramı

3.3.2 Alternatif korelasyon filtreleri ile çok kameralı nesne takibi

Bu bölümde, bölüm 3.2'de anlatılan korelasyon filtresi tabanlı nesne takip yöntemi birden çok kameraya uygulanacak şekilde genişletilmiştir. Çok kameralı nesne takibinde kullanılan takipçi denklem 3.34'deki gibi tanımlanmıştır.

$$\mathcal{T} = \{H, Ha, Tr, Tr_g, S, C, Cp, Int\} \quad (3.34)$$

Nesneyi modelleyen ana korelasyon filtresi (H), alternatif korelasyon filtreleri (Ha) ve takipçi durumu (S) tek kameralı yöntem ile aynı içeriğe sahiptir. Tek kameralı takip yönteminde nesne izi (Tr) piksel koordinatları ile temsil edilirken, çok kameralı takip yönteminde kamera kalibrasyonu sayesinde elde edilen dönüşüm ile coğrafi koordinatlar (Tr_g) da saklanmaktadır. Nesne coğrafi izi $Tr_g = \{x, y, w, h, t\}$, hedefin bulunduğu konum koordinatları, nesne genişliği, yüksekliği ve o konumda bulunduğu zaman olarak saklanmaktadır. $Tr = \{x, y, w, h, t, C\}$, nesnenin coğrafi izinden farklı olarak her bir konumla birlikte hangi kameraya (C) ait olduğu bilgisini de saklamaktadır. Nesne boyutları piksel koordinatları için piksel cinsinden, coğrafi koordinatlar için cm cinsinden hesaplanmaktadır. Coğrafi boyutlar, o koordinattaki iki nokta arasındaki mesafenin hesaplanması ile elde edilmektedir. Konuma ait zaman bilgisinin tutulması nesnenin hızının hesaplanmasında kullanılmaktadır. Bölüm 3.3.1'de anlatıldığı gibi zeminin düz olduğu varsayımı üzerinden tüm coğrafi koordinatların

$\{x,y,0\}$ üçüncü boyutu 0 olduğu için $\{x,y\}$ olarak 2 boyutta tanımlanmaktadır. Bunun dışında nesnenin son görüldüğü kamerayı temsil eden kamera C tutulmaktadır. Hedef nesnenin geçmiş izi dikkate alınarak sonraki video karesinde olması öngörülen kameraların listesi $Cp = \{C_n, x, y\}$ kamera numarası (C_n) ve öngörülen coğrafi konum ile saklanmaktadır. Nesnenin gelecek konumlarının kestirilmesi özellikle kamera geçişlerinde doğru eşleştirme için kullanılmaktadır. $Int = \{(T_n, t)^{(i)}\}$, takip boyunca kesiştiği takipçi numarası (T_n) ve kesişme zamanı (t) serisini temsil etmektedir (i = kesişim indeksi).

Her kameraya ait ayrı birer takipçi havuzu $P(Pool)$ tanımlanmaktadır. Bu havuzda o an aktif olarak ilgili kamerada takip edilen nesneler bulunmaktadır.

$$P = \{T_1, T_2, \dots, T_N\} \quad (3.35)$$

Tüm kameralara ait takipçi havuzlarını ve o an aktif olarak herhangi bir kamerada görülmeyen nesnelerin tutulduğu havuzu $P_{Un(attended)}$ kapsayan global bir takipçi havuzu P_g tanımlanmaktadır.

$$P_g = \{P_{c1}, P_{c2}, \dots, P_{cN}, P_{Un}\} \quad (3.36)$$

Her kameraya ait alt takipçi havuzlarının tutulması algılanan nesnenin öncelikle görüldüğü kameraya ait takipçilerle karşılaştırması açısından kolaylık sağlamaktadır. Bu yolla karşılaştırma işlemi daha hızlı gerçekleştirilmektedir. Her yeni video karesinde uygulanan algoritmik akış şekil 3.9'da görülmektedir.

Karşılaştırılan takipçiler; o kameraya ait takipçiler, sahihsiz takipçiler ve algılanan nesne başka bir kamera ile kesişim alanında ise kesişen kameraya ait takipçilerdir. Algılanan nesne ile birden fazla takipçi konum olarak kesişiyor ise korelasyon hatası daha düşük olan takipçi ile eşleştirilmektedir. Konum kesişimi için nesnenin etrafında kendi genişliğinin 2 katı çapında bir çember oluşturulmaktadır.

$$Int(O, T) = \begin{cases} 1, & |L_O - L_T| < 2 * W_T \\ 0, & otherwise \end{cases} \quad (3.37)$$

O ve T sırasıyla algılanan nesne ve takipçiyi temsil ederken L_O algılanan nesne konumu, L_T takipçi konumu ve W_T takipçiye ait nesne genişliğini temsil etmektedir. Herhangi bir takipçi ile kesişmeyen nesne için yeni takipçi oluşturulmaktadır. Sonraki adımda her takipçiye ait nesne hızı ve konumu kullanılarak “Kalman” filtresi ile bir sonraki olası konumu kestirilmektedir (Welch ve Bishop 2001). “Kalman” filtresi ile nesnenin k anındaki durumu $\vec{x}_k$, denklem 3.38’teki gibi konum ($\vec{l}$) ve hız ($\vec{v}$) ile gösterilmektedir. Konum ve hız değişkenlerini rastgele “Gaussian” dağılımları olarak kabul edilmektedir. Her değişkenin rastgele dağılım merkezi olan ortalama değeri (μ) ve belirsizliğini temsil eden varyans değeri (σ^2) tanımlanmaktadır.

$$\vec{x}_k = (\vec{l}, \vec{v}) \quad (3.38)$$

Nesnenin en iyi durum kestirimi $\hat{x}_k$ ve kestirimin kovaryansı P_k denklem 3.39 ve 3.40’deki gibi tanımlanmaktadır.

$$\hat{x}_k = \begin{bmatrix} location \\ velocity \end{bmatrix} \quad (3.39)$$

$$P_k = \begin{bmatrix} \Sigma_{ll} & \Sigma_{lv} \\ \Sigma_{vl} & \Sigma_{vv} \end{bmatrix} \quad (3.40)$$

Sonrasında şu anki durumun bir sonraki duruma taşınmasını sağlayan kestirim fonksiyonu F_k tanımlanmaktadır. Kestirim için en basit anlamda iki durum arasında geçen sürede nesnenin hızına bağlı olarak aldığı yol kullanılmaktadır.

$$P_k = P_{k-1} + \Delta t v_{k-1} \quad (3.41)$$

$$v_k = v_{k-1} \quad (3.42)$$

$$\hat{x}_k = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \hat{x}_{k-1} \quad (3.43)$$

$$\hat{x}_k = F_k \hat{x}_{k-1} \quad (3.44)$$

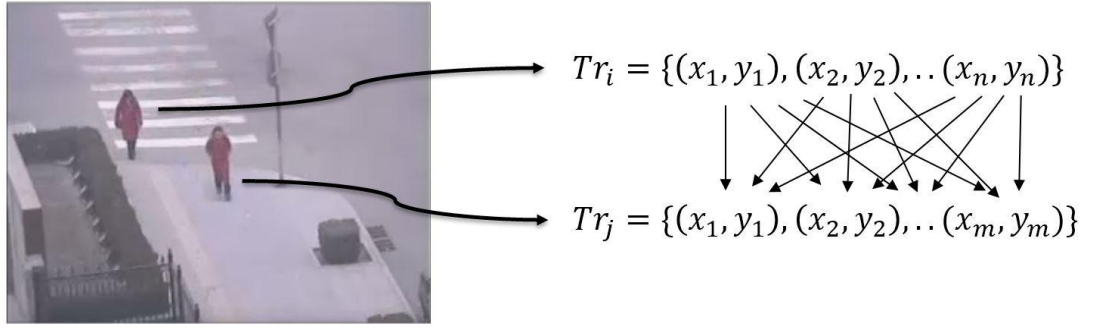
Kestirim fonksiyonu güncel tüm dağılımı bir sonraki duruma taşıyarak yeni bir dağılım oluşturmaktadır. Hala bu dağılım içinde hangi değerlerin gerçek değerler olduğu bilinmemektedir. Dağılımın ortalaması en iyi kestirim olarak kabul edilmektedir. Bir sonraki video karesinde ilgili takipçi, algılanan bir nesne ile eşleştirildiğinde $\hat{x}_k$ ve P_k güncellenmektedir.

Sahne $t_{passive}$ süresi boyunca, algılanan herhangi bir nesne ile eşleşmeyen takipçiler sahipsiz takipçi alt havuzuna (P_{Un}) aktarılmaktadır. Takipçinin durumu (S) pasif duruma getirilmektedir. Sahipsiz takipçi, takip boyunca herhangi bir takipçi ile kesişmiş ise iki takipçinin kesişim süresindeki yön, hız ve coğrafi boyutları karşılaştırılmaktadır. Karşılaştırma sonunda aynı olduğuna karar verilen takipçiler birleştirilmektedir. Havuzda yer alan sahipsiz takipçilerin karşılaştırma zamanını gereksiz uzatmaması adına belirli bir süre sonunda silinmesi gerekmektedir. Bunun için takipçilerin sahipsiz olarak kalabileceği en uzun süre t_{lost} tanımlanmaktadır. Bu sürenin sonunda hala sahipsiz olan takipçiye ait nesnelerin tekrar görüntüye girmeyeceği varsayılarak takibin tamamlandığına karar verilmektedir. Bu nesneler video sinopsiste kullanılmak üzere başka bir listeye aktarılmaktadır.

3.4 Aktivite Optimizasyon İşlemi

Aktivite optimizasyonu kaynak video üzerinde tespit edilen en fazla aktivitenin toplamda en kısa sürede gösterilmesi için aktivitelerin zaman düzleminde kaydırılarak optimal yerleşiminin bulunmasını amaçlamaktadır. Aktivite optimizasyonu hakkında detaylı açıklama bölüm 2.2.1’de yer almaktadır. Literatürde yer alan çevrimiçi aktivite optimizasyon yöntemlerinin en önemli sorunu gerçek-zamanlı çalışabilmek için enerji minimizasyonundan ödün vermek zorunda kalmalarıdır. Çevrimiçi yöntemlerin çalışma zamanı performansının artmasının önündeki en önemli darboğaz her aktivite kombinasyonu için aktivitelerin birbirine göre kesişim durumlarının hesaplanıp zaman düzleminde kaydırılmasıdır. Aktivitelere ait tüm örneklerin ilgili zamandaki konumlarının birbiri ile karşılaştırılması masraflı bir işlemdir. Kesişim hesaplama işlemi, en uygun konumu bulmak için aktiviteyi kaydırarak arka arkaya uygulanmaktadır. Bu durum özellikle aktivite sayısının yüksek olduğu durumda işlem süresinin uzamasına sebep olmaktadır.

Tez kapsamında geliştirilen çevrimiçi aktivite optimizasyon yönteminde ilk olarak bu darboğazın çözülmesine, sonrasında da buradan elde edilen çalışma zamanı performans kazancı ile daha detaylı enerji minimizasyonu yapılmasına odaklanılmıştır. Literatürde yer alan aktivite optimizasyon yöntemlerinde $W \times H$ piksel boyutunda, L adet video karesinden oluşan bir videoda i aktivitesine ait iz (trajectory) $Tr_i = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ ($N \leq L; x \in 1, 2, \dots, W; y \in 1, 2, \dots, H$) şeklinde temsil edilmektedir. İki aktivitenin kesişim durumunu kontrol etmek için ikisine ait tüm iz koordinatları şekil 3.10’da gösterildiği gibi birbiri ile karşılaştırılmaktadır.



Şekil 3.10 i ve j aktivitelerinin izlerine ait piksel koordinatlarının karşılaştırılması

Geliştirilen yöntemde karşılaştırma işlemini hızlandırmak için Şekil 3.11’de görüldüğü gibi görüntü $M \times N$ adet gride bölünmektedir (Baskurt ve Samet 2018b).



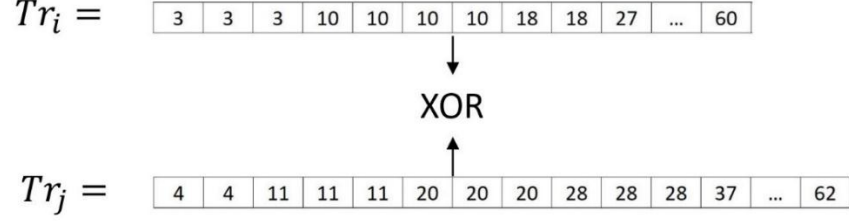
Şekil 3.11 Görüntünün gridlere bölünmesi

Grid gösterimi ile i aktivitesine ait iz denklem 3.45’teki gibi nesnenin görüldüğü video karelerinde bulunduğu grid numarası (g_N) ile temsil edilmektedir.

$$Tr_i = \{g_1, g_2, \dots, g_N\}, (N \geq L) \quad (3.45)$$

Bu şekilde nesne izi (x, y) olmak üzere 2 tamsayı yerine maksimum grid sayısının 128 ile sınırlandırılması sonucunda 1 bayta indirgenmektedir. Bu dönüşüm sonucunda aktivite izlerinin karşılaştırılması (XOR) ve zaman düzleminde kaydırılması bit işlemi

(bit-wise) olarak gerçekleştirilmektedir (Şekil 3.12). Bu sayede aktivite karşılaştırma işlemleri çok daha hızlı gerçekleştirilmektedir.



Şekil 3.12 i ve j aktivitelerinin izlerine ait gridlerin karşılaştırılması

Aktivite optimizasyonu esnasında gösterilmekte olan aktiviteleri temsil eden sinopsis aktiviteleri (A_s) ve uygun konuma yerleştirilecek aday aktiviteleri gösteren A_c olmak üzere iki aktivite kümesi tanımlanmaktadır. Yeni bir aktivite algılandığında A_c kümesine eklendikten sonra A_s içindeki olası konumlarda tanımlanan enerji fonksiyonu hesaplanarak fonksiyon değerinin minimum olduğu konum aktivitenin video sinopsis içinde gösterileceği zamanı temsil etmektedir. Enerji fonksiyonu (E) denklem 3.46'da görüldüğü gibi aktivite yoğunluğu (E_a), kesişim (E_o) ve kronolojik tutarlılık (E_c) metriklerinden oluşmaktadır.

$$E = \sum_{i \in A_s, A_c} E_a(i) + \sum_{i \in A_c, j \in A_s} [\alpha E_c(i, j) + \beta E_o(i, j)] \quad (3.46)$$

Kaynak videoda algılanan tüm aktiviteler video sinopsiste gösterildiği için E_a herhangi bir ağırlık parametresi almamaktadır. E_c için α , E_o için β ağırlık parametreleri kullanılarak bu metriklerin enerji fonksiyonunun değeri üzerindeki etkisi kontrol edilmektedir ($\alpha, \beta \in \mathbb{R} \mid 0 \leq \alpha, \beta \leq 1$). Ağırlık parametresinin 0 olması ilgili metriğin ihmal edilmesine sebep olurken, yüksek olması cezalandırdığı durumun global enerji değeri üzerindeki etkisinin aynı oranda yüksek olması anlamına gelmektedir.

E_a , mümkün olan en fazla sayıda aktivitenin video sinopsise dahil edilmesini zorlayan metriktir. Aktivitelerin video sinopsiste eklenmeyen örnekleri için tanımlanan enerji fonksiyonunun değerini artırmaktadır.

$$E_a(i) = 1 - \left(\sum_{f_s \in F_s} Area_{f_s}(i) / \sum_{f_i \in F_i} Area_{f_i}(i) \right) \quad (3.47)$$

F_s , sinopsis videodaki tüm kareleri, F_i ise kaynak videodaki kareleri temsil etmektedir. Bu metrik her aktivite özelinde hesaplanmaktadır. Diğer iki metrik aktivitelerin birbirine göre durumu dikkate alınarak hesaplanmaktadır.

E_c , aktivitelerin kaynak videodaki kronolojik sıralarını bozarak video sinopsise eklenmeleri durumunda enerji fonksiyonunun değerini artırmaktadır.

$$E_c(i, j) = \begin{cases} 0, & (l_i - l_j) \cdot (t_i - t_j) > 0 \\ 1, & otherwise \end{cases} \quad (3.48)$$

l_i ve l_j , aktivitelerin video sinopsise eklendikleri kareleri, t_i ve t_j ise kaynak videodaki başlangıç karelerini temsil etmektedir.

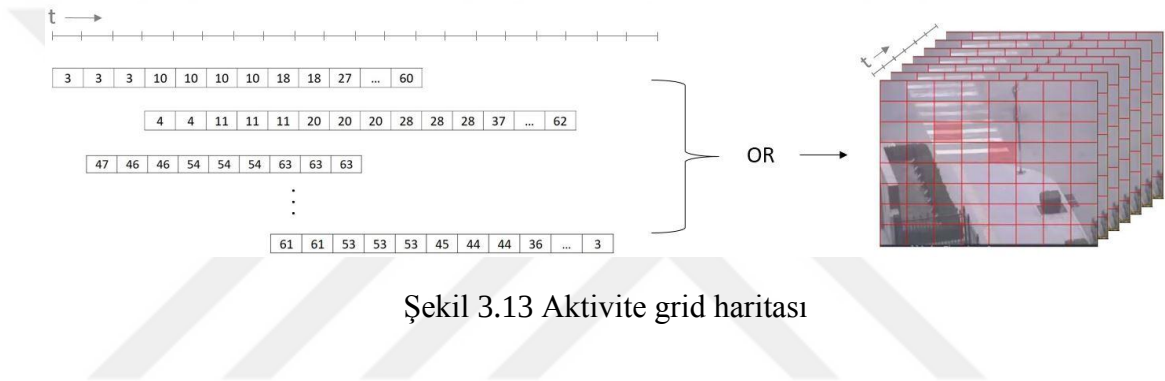
E_o , aktivitelerin birbiriyle kesişmesi durumunu cezalandıran metriktir. Her video karesinde kesişen grid sayısının, kesişen aktivite uzunluklarının ortalamasına oranı olarak hesaplanmaktadır.

$$E_o(i, j) = \sum_{t_s \in T_s} \left[\frac{Collision_{t_s}(i, j)}{(Length(i) + Length(j)) \cdot 0.5} \right] \quad (3.49)$$

Aktivitelerin kesişim durumunu hesaplarken de piksel hesabı yerine nesnelerin bulunduğu gridler dikkate alınmaktadır. Nesnenin ilgili video karesinde kapladığı alan da aynı şekilde grid hesabı ile belirlenmektedir. Aktivitelerin kesişim durumuna karar verirken kesişim eşik değeri kullanılmaktadır. Bu değer aktivitelerin en az kaç video karesi boyunca uzamsal olarak kesiştiğinde kesişen aktiviteler olarak kabul edileceğini belirlemektedir. Kesişim eşik değerinin başarımlar üzerindeki etkisi bölüm 4.3.1’de analiz edilmiştir.

Literatürdeki yöntemlerden farklı olarak tüm metrikler 0 ile 1 arasına ölçeklenmektedir. Bu sayede aktivitelerin tamamının global enerji değerine aynı oranda etki etmesi sağlanmaktadır. Örnek olarak aktivite yoğunluğu ve kesişim metriklerinde ölçekleme yerine direk olarak nesne alanları dikkate alınmış olsaydı, metrik ihlal oranı daha az olsa bile büyük nesneler küçük nesnelere göre daha fazla etki edecekti.

Denklem 3.46’da tanımlanan enerji fonksiyonunu minimuma indirgeyerek her yeni aktivitenin en uygun pozisyona yerleştirilmesi için mevcut aktivite dağılımını temsil eden grid haritası oluşturulmaktadır (Şekil 3.13).



Grid haritası video sinopsiste yer alan aktivitelerin (gösterilmekte olan ve en uygun pozisyonuna yerleştirilmiş ama başlangıç zamanı henüz gelmemiş aktiviteler) her bir video karesinde işgal ettikleri gridleri temsil etmektedir. Grid haritasının uzunluğu yeni gelen aktivitenin yerleştirildiği zamana göre dinamik olarak artmaktadır. Yeni aktivite bir sonraki video karesinden başlanarak grid haritasıyla karşılaştırılmaktadır. Karşılaştırma işleminde aktivitenin tüm izi, grid haritasında karşılaştırılan video karesinden itibaren karşılık gelen gridlerle karşılaştırılarak denklem 3.46’daki enerji fonksiyonunun değeri hesaplanmaktadır. Bu işlem aktivite haritanın sonuna gelene kadar kaydırılarak devam etmektedir. Performans kazancı için kaydırma işlemi q video karelik aralıklarla yapılmaktadır. q parametresinin aldığı değerler bölüm 4.3’te araştırma bulgularında tartışılmaktadır. İterasyon sonunda toplamda en düşük enerjili konuma aktivite yerleştirilerek grid haritası aktivite izi ile güncellenmektedir. Geliştirilen yöntemin sözde (pseudo) kodu aşağıdaki gibidir:

Algoritma 4. BuildVideoSynopsis**Input:** A_s^i, A_c^i , $i = 1, 2, \dots$, number of cameras**Output:** A_s^i Initialize object pool P_i for each camera

```

1:  while 1 do
2:      if  $P_i$  not equal  $\emptyset$  then
3:          Add objects of  $P_i$  to  $A_c^i$ 
4:      end if
5:      if  $A_c^i$  not equal  $\emptyset$  then
6:          call OptimizeActivities with  $A_s^i, A_c^i$  and current frame  $F$ 
7:          call DisplayActivities with  $A_s^i, F$ 
8:      end if
9:      Display  $F$ 
10: end while

```

Algoritma 5. OptimizeActivities**Input:** A_s, A_c, id_{cam} **Output:** A_s Initialize best object shift, $S \leftarrow 0$

```

1:  while  $A_c$  not equal  $\emptyset$  do
2:      Pick activity  $A$  from  $A_c$ 
3:       $N \leftarrow$  frame length of  $A_s$ 
4:      for  $i \leftarrow 0$  to  $N$  do
5:          Calculate energy cost  $E_i$  of current shift by Eq. (3.46)
6:      end loop
7:       $S =$  frame number of  $\min(E_i)$ 
8:      Insert  $A$  into  $S$ -th frame of  $A_s$ 
9:  end while

```

Algoritma 6. DisplayActivities**Input:** A_s , current frame F **Output:** current frame F with synopsis objects blendedInitialize list of object to be blended, $B_i \leftarrow \emptyset$

```

1:  while  $A_s$  not equal  $\emptyset$  do
2:      Get object  $O$  from  $A_s$ 
3:      Push  $O$  to  $B_i$ 
4:      Fill grid of  $O$  to display
5:       $\text{displayedInstance}[O] \leftarrow \text{displayedInstance}[O] + 1$ 
6:      if  $\text{displayedInstance}[O]$  equal  $\text{totalInstanceCount}[O]$  then
7:          Send  $O$  to  $P_{\text{nextCamera}}$ 
8:      end if
9:  end while
10: Sort  $B_i$  by position
11: Blend  $B_i$  to  $F$ 

```

Geliştirilen çevrimiçi optimizasyon yöntemi her kamera için ayrı olarak çalışmaktadır. Bölüm 3.3'te anlatılan çok kameralı nesne takibi yöntemi sayesinde her bir aktivitenin hangi kameralarda görüleceği bilinmektedir. Bu sayede herhangi bir kamera görüntüsünden çıkan aktivite görüntüsüne gireceği bir sonraki kameraya ait aday aktiviteler (A_c) arasına eklenmektedir. Bir önceki kamerada olduğu gibi bu kameraya ait mevcut grid haritasına göre en uygun zamana kaydırılarak gösterilmektedir. Aktivite kamera görüntülerinden tamamen çıkana kadar bu şekilde kameralar arasında taşınmaktadır.



4. BULGULAR VE TARTIŞMA

Bu bölümde tez kapsamında geliştirilen çok kameralı nesne takibi tabanlı çevrimiçi video sinopsis sisteminin farklı senaryolarda uygulamaları yapılmıştır. Elde edilen sonuçlar literatürdeki öncü yöntemlerin sonuçlarıyla karşılaştırılmış ve tartışılmıştır. Tez çalışması kapsamında geliştirilen nesne algılama, nesne takibi ve aktivite optimizasyonu yöntemlerine ait araştırma bulguları ve sistemin çalışma zamanı performansı bu bölümde paylaşılmıştır. Uygulama çalışmaları maksimum kamera sayısının 8, optimizasyon havuzunda bekleyen maksimum aktivite sayısının 100 ve maksimum kamera çözünürlüğünün 1920x1080 olduğu varsayımıyla gerçekleştirilmiştir.

4.1 Nesne Algılama

Bölüm 3.1’de anlatıldığı gibi nesne algılama için hareket algılama yöntemi kullanılmıştır. Geliştirilen hareket algılama yönteminin performans değerlendirmesi için çizelge 4.1’deki metrikler kullanılmıştır.

Çizelge 4.1 Hareket algılama için kullanılan performans metrikleri

Metrik	Açıklama
Yakalama (Recall)	$DP / (DP + YN)$
Kesinlik (Precision)	$DP / (DP + YP)$
F1	$(2 \times Kesinlik \times Yakalama) / (Kesinlik + Yakalama)$

(DP: Doğru Pozitif, YP: Yanlış Pozitif, YN: Yanlış Negatif, DN: Doğru Negatif)

Yakalama hareketli piksel olarak algılanan hareketli piksel sayısının hareket olarak algılanan arka plan pikseli sayısına oranını temsil etmektedir. *Kesinlik* metriği hareket olarak algılanan hareketli piksel sayısının arka plan olarak algılanan hareketli piksel sayısına oranını temsil etmektedir. *F1* ise *yakalama* ve *kesinlik* metriklerinin harmonik ortalamasını temsil etmektedir.

Karşılaştırmada “Change detection 2012” veri kümesi kullanılmıştır (Goyette vd. 2012). Veri kümesi, farklı durumlarda yöntem başarımını ölçmek adına “baseline”, “camera

jitter”, “dynamic background”, “intermittent object motion”, “shadow” ve “thermal” olmak üzere 6 kategoriden oluşmuştur. “Baseline” kategorisi yöntemi zorlayıcı durumların olmadığı basit videolardan oluşmuştur. “Camera jitter” kamera hareketinden dolayı görüntü kalitesinin düşük olduğu görüntülerden oluşmuştur. “Dynamic background” yoğun şekilde hareketli arka plana (ağaç, su, bayrak, vb.) sahip görüntülerden oluşmuştur. “Intermittent object motion” park eden araç gibi hareketliken arka plana dahil olan veya bir süre sonra hareket etmeye başlayan arka plan nesnelere sahip görüntülerden oluşmuştur. “Shadow” iç ve dış ortamda nesnelerin yoğun ve hafif şekilde gölgeye sahip olduğu görüntülerden oluşmuştur. “Thermal” ise termal kamera ile alınmış görüntülerden oluşmuştur. Yöntem karşılaştırması her bir kategori için ayrı olarak yapılmış bunun yanında tüm kategorilerin ortalaması da paylaşılmıştır. Bu veri kümesi hareket algılama konusunda öncü 44 yöntemin kapsamlı karşılaştırmasını içermektedir. Bu yüzden performans karşılaştırmasını bu veri kümesi üzerinde yapmak diğer 44 yöntem ile karşılaştırma imkanı da sağlamıştır. Performans değerlendirmesi Intel® Xeon® Quad Core 3.40 GHz işlemci gücüne ve 8 GB belleğe sahip bilgisayar üzerinde gerçekleştirilmiştir.

Geliştirilen hareket algılama yöntemi referans alınan PBAS yöntemi ile bu performans metrikleri dikkate alınarak karşılaştırılmıştır. Karşılaştırmada PBAS yönteminin yazarlar tarafından sağlanan implementasyonu kullanılmıştır. PBAS yazarları tarafından önerilen optimal parametre konfigürasyonu şu şekildedir: $\{N = 35, \#_{min} = 2, R_{inc/dec} = 0.05, R_{lower} = 18, R_{scale} = 5, T_{inc} = 1, T_{dec} = 0.05, T_{lower} = 2, T_{upper} = 200\}$.

Geliştirilen yöntemde $T_{upper} = 5$ değeri dışında aynı değerler kullanılmıştır. Bunun dışında sahne dinamizmine göre otomatik olarak değişen $R_{inc/dec}$, T_{inc} ve T_{dec} parametreleri için alt ve üst sınır değerleri belirlenmiştir: $\{R_{inc/dec}^{upper} = 0.05, R_{inc/dec}^{lower} = 0.01, T_{inc}^{upper} = 1.5, T_{inc}^{lower} = 0.5, T_{dec}^{upper} = 0.1, T_{dec}^{lower} = 0.02\}$. Yöntemlerin başarımları ve çalışma zamanı performansının karşılaştırılması toplu olarak çizelge 4.2’de verilmiştir (İyi değerler koyu ile gösterilmiştir).

Çizelge 4.2 Önerilen yöntemin Change Detection Dataset üzerinde PBAS ile karşılaştırılması

		Baseline	Camera Jitter	Dynamic Background	Intermittent Object Motion	Shadow	Thermal	Ortalama
Yakalama	PBAS	0,8259	0,7927	0,7918	0,4409	0,8447	0,6395	0,7226
	Geliştirilen Yöntem	0,8246	0,7114	0,7836	0,5238	0,8665	0,6617	0,7286
Kesinlik	PBAS	0,7698	0,4386	0,6535	0,7206	0,7283	0,7884	0,6832
	Geliştirilen Yöntem	0,8225	0,5211	0,7182	0,6308	0,7130	0,8176	0,7039
F1	PBAS	0,7820	0,5490	0,6183	0,4922	0,7772	0,6806	0,6499
	Geliştirilen Yöntem	0,8198	0,5617	0,6974	0,5379	0,7550	0,7176	0,6816
Çalışma zamanı (ms)	PBAS	25,25	33,50	32,00	16,50	23,67	17,00	24,65
	Geliştirilen Yöntem	19,75	18,5	17,33	10,5	15	10	15,18

Çizelge 4.2’de görüldüğü gibi veri kümesindeki her bir senaryonun sonucu ilgili metriğe ait ortalama sonuç ile birlikte verilmiş ve geliştirilen yöntem tüm metriklerin genel ortalamasından daha iyi sonuç göstermiştir.

Bölüm 3.1’de açıklandığı gibi görüntüdeki her piksele ait geçmiş örnekleri kapsayan ayrık bir arka plan modeli oluşturulmuştur. Ayrık arka plan modelinin sağladığı önemli bir avantaj görüntünün alt parçalara bölünüp paralel olarak işlenmesine imkan sağlaması olmuştur. Parallelleştirme işleminde dikkat edilmesi gereken nokta görüntünün alt parçalarının sınırlarındaki piksellere ait komşu örneklerin güncellemesinde başka bir alt bölgeye müdahale etme ihtiyacıdır. Yöntemin uygulamasında bu durum ilgili iş parçacıkları arasında iletişim kurularak kontrol edilmiştir. Görüntü paralel işleme için 2x2 olmak üzere 4 alt parçaya bölünmüştür. Çizelge 4.2’de görülen çalışma zamanı karşılaştırması uygulamanın etkinliğini göstermiştir. Geliştirilen yöntem PBAS’a göre genel karşılaştırmada yaklaşık olarak 38.4% daha hızlı çalışmıştır.

Sonuç olarak bu karşılaştırma sonucunda geliştirilen yöntemin PBAS’a karşı üstünlüğü gösterilmiştir. PBAS’a göre 3 parametre daha sabit değerler yerine sahne değişimine göre otomatik ayarlanabilen hale getirilmiştir. Yöntemin kullanıcı tarafından belirlenen sabit parametreler yerine otomatik ayarlamalarla koşması sahneden bağımsız olarak yüksek performans göstermesini sağlamaktadır. Video sinopsisin çevre güvenliğinde

herhangi bir gözetleme kamerası üzerine uygulanacağı düşünüldüğünde kullanılan hareket algılama yönteminin sahneden mümkün olduğunca bağımsız olarak iyi performans göstermesi önemlidir.

PBAS'ın arka plan modelinin güncellemesinde kullandığı komşu modeli güncelleme yaklaşımı hareketli piksellerin modele dahil edilme ihtimalini barındırmaktadır. Bu durumu engellemek için güncelleme mantığı değiştirilmiştir. Bunun dışında geliştirilen ani değişim algılayıcısı sayesinde arka plan modeli sahnedeki ani dominant değişimlere en hızlı şekilde adapte olarak yanlış algılama oranını minimuma düşürmüştür. Son olarak arka plan modelinin ayırık yapısından yararlanılarak gerçekleştirilen paralel uygulama ile çalışma zamanında ciddi bir kazanım sağlanmıştır. Bahsedilen başarımlar ve çalışma zamanı iyileştirmeleri ile geliştirilen yöntem PBAS ve bir çok öncü yöntemden daha iyi sonuç vermiştir. Bu çalışmanın devamında parametre konfigürasyonuna ihtiyaç duymayan tamamen otomatik bir yöntem geliştirilmesi amaçlanmaktadır. Bunun dışında veri kümesi üzerindeki “intermittent” senaryosundaki performans genel başarımları ciddi anlamda düşürmüştür. Bu senaryo üzerindeki yöntem başarımının iyileştirilmesi gerekmektedir.

4.2 Hareketli Nesne Takibi

Bu bölümde geliştirilen nesne takip yönteminin performans değerlendirmesi yapılmıştır. Geliştirilen yöntem referans alınan yöntem (DSST), Muster, Struck, TLD, ASMS (Vojir vd. 2014) ve SAMF (Li ve Zhu 2015) yöntemleri ile karşılaştırılmıştır. TLD ve Struck hedef nesnenin negatif örneklerini de saklayarak görünüm değişimlerine karşı ekstra dayanıklılık sağlayan başarımları yüksek yöntemlerdir. Muster nesneyi tek model yerine iki model ile temsil ederek başarımları artırmayı amaçlamıştır. Bu yüzden alternatif korelasyon filtrelerinin etkisini karşılaştırmak için iyi bir adaydır. ASMS, Visual Object Tracking Challenge (VOT: Kristan vd. 2015) sonuçlarına göre en yüksek doğruluğa sahip gerçek zamanlı nesne takip yöntemi olarak rapor edilmiştir. SAMF ise aynı sonuçlara göre “pooled AR-rank” kategorisinde en iyi başarımları gösteren yöntemlerden biridir. Geliştirilen video sinopsis sisteminin güvenlik kameraları üzerinde çalışması planlandığı için performans değerlendirmesi OTB veri kümesinin (Wu vd. 2015) sabit

kamera ile alınan görüntüleri üzerinde gerçekleştirilmiştir. Hareketli kameralara ait görüntüler dikkate alınmamıştır. OTB nesne takibinde performans değerlendirmesi için kullanılan 100 güncel videodan oluşmaktadır. Veri kümesindeki görüntüler aydınlatma değişimi, ölçek değişimi, engelleme, deformasyon, hareket bulanıklığı, hızlı hareket, görüntü içinde rotasyon, görüntü dışında rotasyon, görüntü dışına taşma, arka plan hareketi, düşük çözünürlük gibi 11 farklı kategoriye ayrılmıştır. Değerlendirmede VOT dahilinde de kullanılan doğruluk (accuracy) ve dayanıklılık (robustness) kriterleri dikkate alınmıştır. OTB veri kümesi ve VOT kriterleri nesne takip ile ilgili literatürde yaygın olarak kullanılmaktadır. Bu yüzden geliştirilen yöntemin bu standartlara göre karşılaştırılması global ve güncel bir değerlendirme sağlamıştır.

Doğruluk, yöntem tarafından kestirilen nesne çerçevesi ile nesneye ait referans çerçeve arasındaki kesişimin nesne takibinin devam ettiği süre boyunca elde edilen ortalamasıdır. 0 ile 1 arasına ölçeklenmiş bir değer ile temsil edilmektedir. Dayanıklılık, takip boyunca her 1000 video karesinde takipçinin nesneyi kaybetme sayısının ortalamasıdır. Nesneyi kaybetme durumu kestirilen nesne çerçevesi ile referans çerçeve arasındaki kesişimi sınırlayan bir eşik değeri ile kontrol edilmektedir. VOT değerlendirme sistemine göre takipçi nesneyi kaybettikten 5 video karesi sonra doğru pozisyondan tekrar başlatılmaktadır. Tekrar başlatma işlemini takip eden 10 video karesindeki performans değerlendirmeye dahil edilmemektedir. Doğruluk ve dayanıklılık kriterleri dışında hız metriği ile yöntemin çalışma zamanı ölçülmektedir. Hız metriği milisaniye cinsinden ortalama video karesi işleme süresi ile temsil edilmektedir. Performans karşılaştırması Intel Xeon Quad Core 3.40 GHz işlemci gücü ve 8 GB belleğe sahip bir bilgisayar üzerinde gerçekleştirilmiştir.

Geliştirilen nesne takip yönteminin uygulamasında yeni bir hareketli nesne tespit edildiğinde ilgili korelasyon filtresi oluşturulurken filtre boyutu $M \times N$, nesne boyutunun (hareket blobu) iki katı olarak belirlenmiştir. Literatürde konu ile ilgili yapılan çalışmalar da dikkate alınarak filtre güncellemesi için denklem 3.17, 3.18, 3.26 ve 3.27’te kullanılan güncelleme katsayısı (η) 0.025 olarak belirlenmiştir (Danelljan vd. 2014). Herhangi bir korelasyon filtresinin güncellenmesi için PSR değerinin belirlenen güvenilir eşik değerinin (PSR_{min}) üzerinde olması şartı aranmaktadır. Bolme vd. (2010)

tarafından yapılan deneysel çalışmalara göre *PSR* değerinin 20 ile 60 arasında olması güçlü bir filtreyi temsil etmektedir. Bu değer 7 civarına düşmesi hedef nesnenin görülmediği veya takibin yapılamadığı durumlar olarak tanımlanmaktadır.

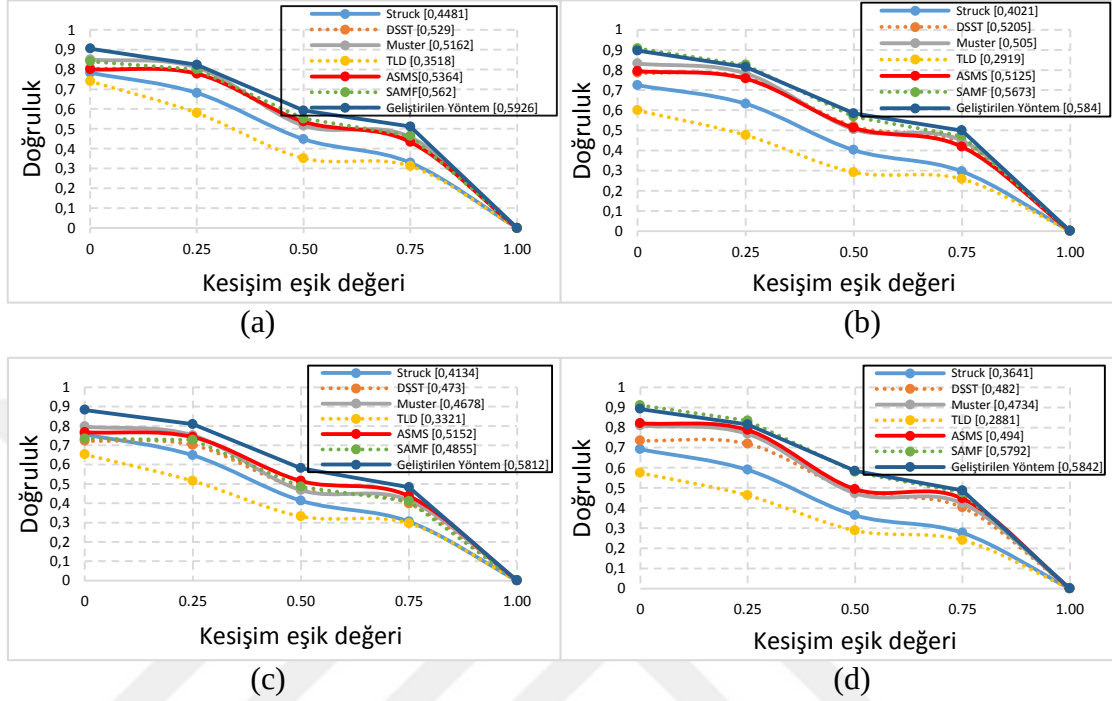
Her bir takipçi için izin verilen maksimum alternatif korelasyon filtresi sayısı 4 ile sınırlandırılmıştır. Yüksek sayıda korelasyon filtresi kullanmak hesaplama karmaşıklığı artırdığı için çalışma zamanı performansı üzerinde direk etki göstermiştir. Daha fazla sayıda korelasyon filtresi kullanmak başarıyı artırsa da çalışma zamanını olumsuz etkilemektedir. Nesnenin tekrar sahneye girmesi için kullanılan tolerans süresi $t_{passive}$ 10 saniye olarak kullanılmıştır. Bu parametre sahnede geçici olarak engellenen/görülmeyen nesneler için birden fazla takipçi oluşturulmasını engellemektedir. Bölüm 3.2.2’de anlatıldığı gibi sahneden çıkan bir nesne sahneye farklı bir noktadan tekrar girebileceği için pasif takipçilerin eşleştirilmesinde konum kesişim şartı aranmamakta ve sadece korelasyon filtresinin eşleşme durumuna bakılmaktadır. Maksimum korelasyon hata eşik değeri (E_{conf}) başlangıçta belirlenen 300 değerinden sonra çalışma esnasında dinamik olarak ayarlanmaktadır. Bunun için çalışma esnasında elde edilen korelasyon hatalarının listesi (E_{list}) tutulmaktadır. Her takipçiye ait korelasyon hataları bu listede tutulduktan sonra (E_{conf}), listeye yeni eklenen her 5. hata değeri ile birlikte denklem 4.1’deki gibi güncellenmektedir.

$$E_{conf} = median(E_{list}) * 2 \quad (4.1)$$

Çalışma zamanı performansını düşürmemek için korelasyon hata listesinin boyutu ilk giren ilk çıkar mantığında 500 ile sınırlandırılmıştır. Son olarak referans alınan yöntemle benzer şekilde hedef nesne ölçek kestirimi için 33 seviyeli 1.02 ölçek faktörüne sahip ölçek piramidi oluşturulmuştur.

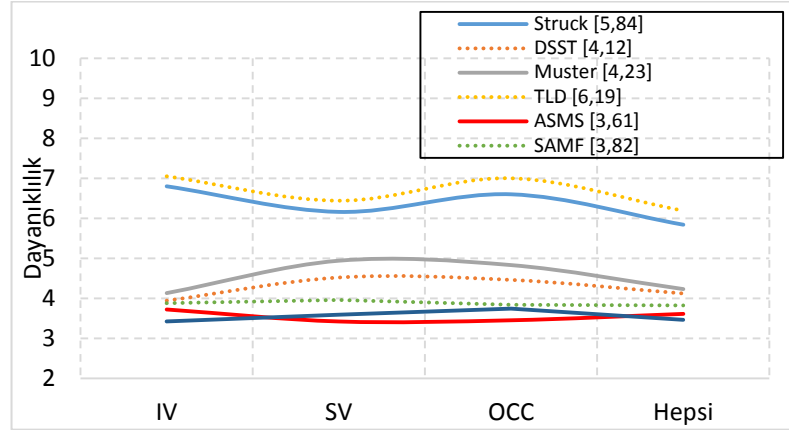
Şekil 4.1 karşılaştırılan yöntemlerin farklı kesişim eşik değerleri için aydınlatma değişimi (IV: Illumination variation), ölçek değişimi (SV: Scale variation) ve engelleme (OCC: Occlusion) kategorilerinin yanında tüm kategoriler için genel olarak doğruluk değerlerini karşılaştırmaktadır. Şekilde görüldüğü gibi daha düşük kesişim eşik

değerleri takipçinin nesneyi kaybetme ihtimalini düşürdüğü için doğruluğu artırmıştır. Şekil 4.2, tüm veri kümesi üzerindeki dayanıklılık değerlerini göstermektedir.



Şekil 4.1 Yöntemlerin karşılaştırılması: a. tüm veri kümesi üzerinde, b. aydınlanma değişimi, c. ölçek değişimi, d. engelleme.

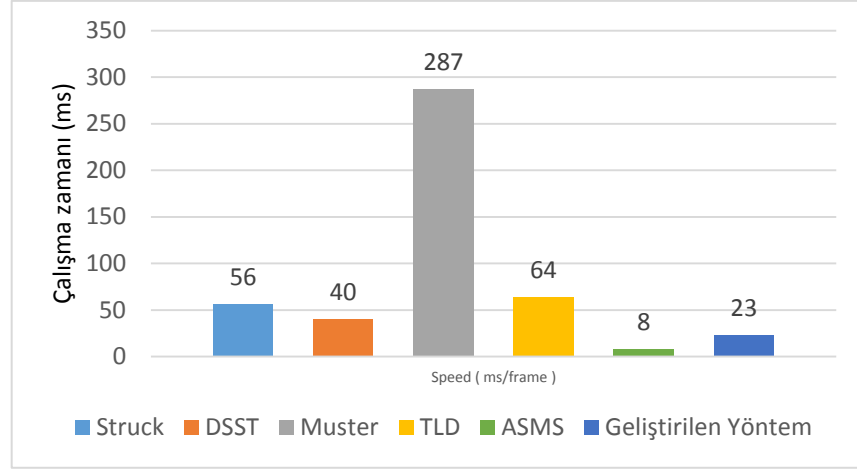
Dikey eksen 0 ile 1 arasına ölçeklenmiş doğruluk değerini yatay eksen ise kesişim eşik değerini temsil etmektedir.



Şekil 4.2 Aydınlatma değişimi (IV), ölçek değişimi (SV), kesişim (OCC) ve tüm veri kümesi ortalaması (Hepsi) için dayanıklılık karşılaştırması.

Her yonteme ait ortalama dayanıklılık bilgisi yöntem adıyla beraber paylaşılmıştır

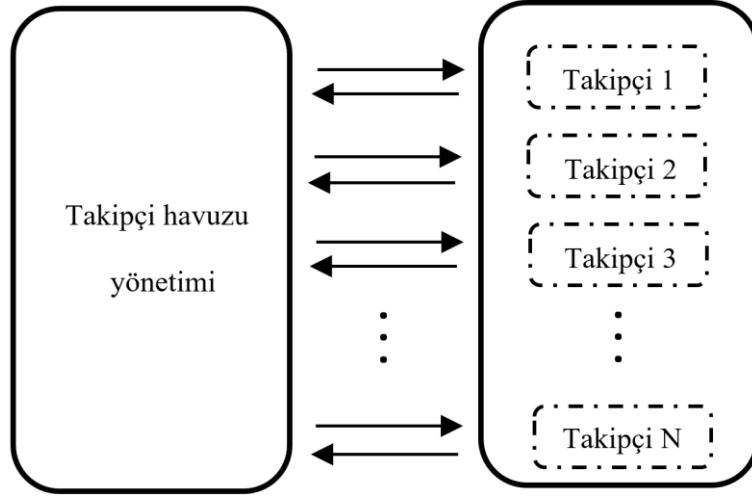
Geliştirilen yöntem kullanılan üç değerlendirme metriğine göre diğer yöntemlerden daha iyi sonuç vermiştir. Özellikle yöntemin dayanıklılık performansı kullanılan alternatif korelasyon filtrelerinin ve hareket tabanlı çerçeve kayma çözümünün başarımını göstermiştir. Bu sonuçlara göre geliştirilen yöntem, gözetleme uygulamalarında oldukça önemli olan uzun süreli takip konusunda diğer yöntemlere göre daha yüksek başarımlar göstermiştir. Karşılaştırmadan çıkarılacak başka bir sonuç da hedef nesnenin birden fazla model ile temsil edildiği yöntemler tek model kullananlara göre daha iyi sonuç göstermektedir. Birden fazla özneliği birleştiren SAMF yöntemi doğruluk karşılaştırmasında ikinci en iyi sonucu göstermiştir. Buradan çoklu modelin yanında birden fazla çeşit özneliği dikkate alınmanın çevresel şartlardan daha az etkilenecek daha dayanıklı bir yöntem ortaya çıkaracağı sonucuna varılabilir. Şekil 4.3 ise, ortalama çalışma zamanlarını göstermektedir.



Şekil 4.3 Tüm veri kümesi üzerinde ortalama kare işleme zamanlarının karşılaştırması

Çalışma zamanı karşılaştırması yöntemlerin C++ implementasyonları kullanılarak gerçekleştirilmiştir. SAMF yönteminin yazarları sadece Matlab kodunu sağladığı için çalışma zamanı karşılaştırması yapılamamıştır. Şekil 4.3'te görüldüğü gibi karşılaştırma sonucunda ASMS yöntemi en hızlı yöntemdir. Bu yöntemde kullanılan basit “mean-shift” yaklaşımı çalışma zamanı performansında avantaj sağlamıştır. Muster yönteminin çalışma zamanı performansı diğerlerine göre oldukça düşüktür. Bunun sebebi hedef nesneyi temsil etmek için birden fazla model kullanmasıdır. Tez kapsamında geliştirilen yöntemde de çoklu model kullanılmasına rağmen çalışma zamanı olarak ikinci en iyi yöntem konumundadır. Geliştirilen çok iş parçacıklı takipçi havuzu başarımları ve çalışma zamanı arasında etkili bir denge sağlamıştır.

Literatür özetinde verilen korelasyon filtresi tabanlı nesne takip yöntemlerinden bazıları gerçek-zamanlı çalışmasına rağmen bu yöntemlerin tamamı aynı anda sadece tek bir nesneyi takip etmektedir. Tez çalışması kapsamında geliştirilen çoklu nesne takip yönteminin mümkün oldukça fazla nesneyi aynı anda gerçek-zamanlı olarak takip etmesi gerekmektedir. Bunun için yöntemin programlamasında çok iş parçacıklı bir uygulama yoluna gidilmiştir. Bölüm 3.2.2’de bahsedilen takipçi havuzu ana iş parçacığı tarafından yönetilirken her takipçi ile ilgili korelasyon ve güncelleme işlemleri kendine ait bağımsız iş parçacığı üzerinde gerçekleştirilmiştir (Şekil 4.4). Takipçi kendi işlemini bitirdiğinde takip sonucunu ana iş parçacığına bildirmektedir. Bu sayede yöntemin gerçek-zamanlı olarak çalışması sağlanmıştır.



Şekil 4.4 Çok iş parçacıklı implementasyon

Çizelge 4.3, 760x570 çözünürlüğündeki görüntüler üzerinde aynı anda takip edilen nesne sayısının performans üzerindeki etkisini göstermiştir. Karşılaştırmada kullanılan nesne sayısı aralıkları işlemcinin çekirdek sayısına göre belirlenmiştir. Sahnedeki toplam nesne sayısı (her nesne ayrı bir iş parçacığı tarafından yönetilmektedir) çekirdek sayısından az olduğunda çalışma zamanı etkilenmemiştir. Nesne sayısı çekirdek sayısını geçtikten sonra çalışma zamanı performansı düşmüştür. Kaynakların paylaşılması için iş parçacıkları arasında gerçekleşen anahtarlama işlemleri toplam çalışma zamanını artırmıştır.

Çizelge 4.3 Sahnede aynı anda görülen nesne sayılarına göre ortalama kare işleme süresi

Nesne sayısı	Ortalama kare işleme süresi (ms)
1-8	17.2
8-16	23.8
>16	35.4

4.3 Aktivite Optimizasyonu

Bu bölümde geliştirilen çevrimiçi aktivite optimizasyon yönteminin farklı veri kümeleri üzerinde uygulanması sonucu elde edilen bulgular analiz edilmiştir. Analiz kapsamında ilk olarak oluşturulan farklı parametre konfigürasyonlarının ve grid haritasının

boyutunun başarımlar üzerindeki etkisi incelenmiştir. Daha sonra geliştirilen yöntemin literatürdeki öncü yöntemlerle karşılaştırılması yapılmıştır.

Video sinopsiste aktivite optimizasyon yönteminin başarımlarını ölçmek için yaygın olarak kullanılan dört metrik aşağıdaki gibidir:

- 1) **Frame condensation ratio (FR):** Video sinopsisteki toplam kare sayısının (T_s) kaynak videodaki toplam kare sayısına (T_i) oranı olarak hesaplanmaktadır. Kaynak videodaki aktivitelerin ne kadar kısa süre içinde gösterildiğinin ölçütüdür. Daha düşük FR , daha iyi aktivite optimizasyonu anlamına gelmektedir.

$$FR = T_s/T_i \quad (4.2)$$

- 2) **Compact ratio (CR):** Video sinopsiste gösterilen aktivitelere ait toplam piksel sayısının video sinopsisteki tüm pixellere oranı olarak hesaplanmaktadır. Aktivite optimizasyonunun video karesi bazında başarımlarını ölçmektedir. Daha yüksek CR , video sinopsiste kare başına düşen aktivite sayısının yüksek olmasını temsil ettiğinden daha yüksek optimizasyon başarımlarını anlamına gelmektedir.

$$CR = \frac{1}{W \cdot H \cdot T_s} \sum_{t=1}^{T_s} \sum_{x=1}^W \sum_{y=1}^H \{1 \mid \text{if } p(x, y, t) \in \text{activity}\} \quad (4.3)$$

$p(x, y, t)$, video sinopsisin t . karesinde, x . sütun y . satırdaki pikseli temsil etmektedir. W ve H sırasıyla görüntünün genişliğini ve yüksekliğini temsil etmektedir.

- 3) **Overlap ratio (OR):** Video sinopsiste yer alan aktivitelerin uzamsal olarak kesişim alanına ait piksel sayısının toplam piksel sayısına oranı olarak hesaplanmaktadır. Daha düşük OR , daha az aktivite kesişimini temsil ettiği için daha yüksek optimizasyon başarımlarını anlamına gelmektedir.

$$OR = \frac{1}{W \cdot H \cdot T_s} \sum_{t=1}^{T_s} \sum_{x=1}^W \sum_{y=1}^H \{1 \mid \text{if } p(x, y, t) \in \text{activity collision}\} \quad (4.4)$$

4) Chronological disorder ratio (CDR): Video sinopsiste kronolojik sırası bozularak gösterilen aktivitelerin (A_d) tüm aktivitelere (A_s) oranı olarak hesaplanmaktadır. Düşük *CDR*, aktivitelerin kaynak videodaki kronolojik sıralamasına daha yüksek oranda riayet edildiğini göstermektedir.

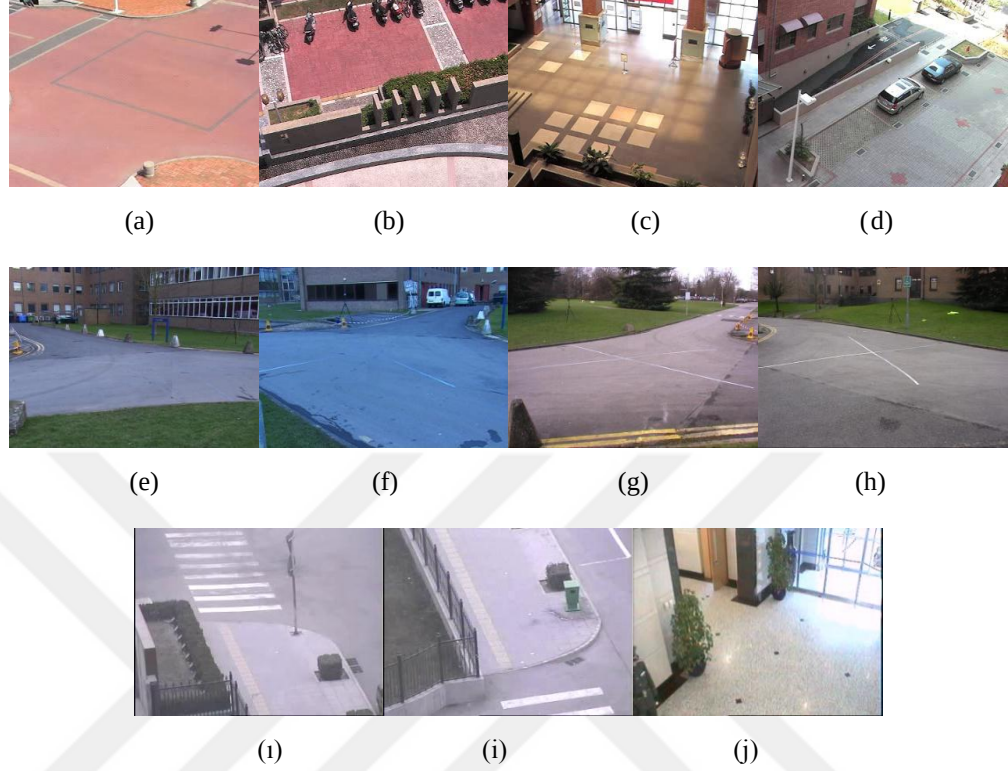
$$CDR = A_d/A_s \quad (4.5)$$

Geliştirilen yöntemin başarımını analiz etmek için bu metrikler kullanılmıştır. Bu bölümde sonuçları paylaşılan tüm uygulamalar, Intel® Xeon® Quad Core 3.40 GHz işlem gücüne ve 8 GB belleğe sahip bilgisayar üzerinde gerçekleştirilmiştir. Veri kümesi olarak literatürde yer alan video sinopsis çalışmalarında yaygın olarak kullanılan 11 farklı küme kullanılmıştır. Kullanılan veri kümelerinin çözünürlük, uzunluk, içerdiği aktivite sayısı ve ortam gibi bilgileri Çizelge 4.4'te paylaşılmıştır.

Çizelge 4.4 Performans değerlendirmesinde kullanılan veri kümeleri

No	Veri kümesi	Çözünürlük	Video kare sayısı	Aktivite sayısı	Sahne
D1	Video Synopsis Dataset Crossroad	320x240	5400	91	Dış ortam
D2	Video Synopsis Dataset Sidewalk	320x240	5400	20	Dış ortam
D3	Video Synopsis Dataset Hall	320x240	5400	23	İç ortam
D4	Video Synopsis Dataset Street	320x240	5400	22	Dış ortam
D5	PETS2009 view 5	720x576	795	28	Dış ortam
D6	PETS2009 view 6	720x576	795	28	Dış ortam
D7	PETS2009 view 7	720x576	795	28	Dış ortam
D8	PETS2009 view 8	720x576	795	28	Dış ortam
D9	MCT Dataset2 C1	320x240	24000	235	Dış ortam
D10	MCT Dataset2 C2	320x240	24000	104	Dış ortam
D11	MCT Dataset2 C3	320x240	24000	179	İç ortam

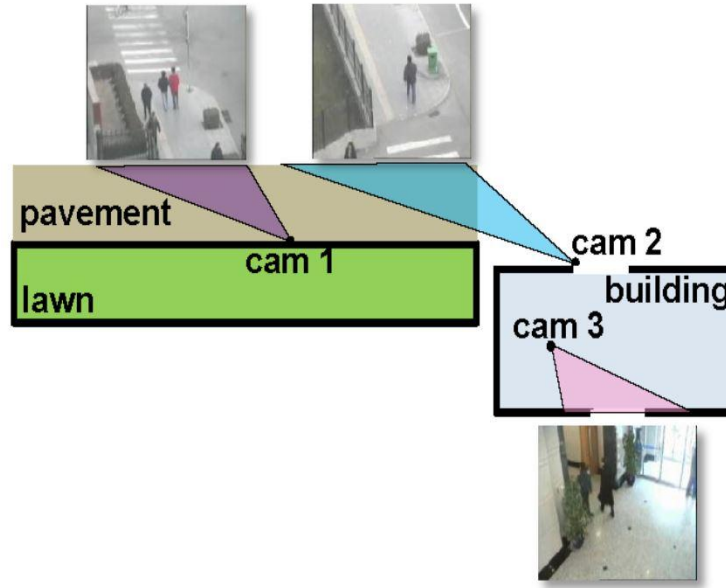
Ayrıca, şekil 4.5’te her bir veri kümesine ait örnek ortam görüntüsü paylaşılmıştır.



Şekil 4.5 Kullanılan veri kümelerine ait görüntüler.

D1: video synopsis dataset crossroad (a), D2: video synopsis dataset sidewalk (b), D3: video synopsis dataset hall (c), D4: video synopsis dataset street (d), D5: PETS2009 view5 (e), D6: PETS2009 view6 (f), D7: PETS2009 view7 (g), D8: PETS2009 view8 (h), D9: MCT Dataset2 C1 (i), D10: MCT Dataset2 C2 (i), D11: MCT Dataset2 C3 (j).

D1, D2, D3 ve D4 veri kümeleri birbirinden bağımsız kameralardan alınmış görüntülerden oluşmaktadır. D5, D6, D7 ve D8 veri kümeleri 4 farklı kameranın aynı alana baktığı bir senaryoyu temsil etmekte ve kamera görüntülerinin büyük bir kısmı birbirleriyle kesişmektedir. Bu veri kümelerindeki aktiviteler genellikle tüm kameralarda aynı anda görülmektedir. D9, D10 ve D11 veri kümeleri 3 farklı kameradan bakış açısı kesişmeyecek şekilde elde edilen görüntülerinden oluşmaktadır. Aktiviteler kameralar arasında geçiş yapmakla beraber aynı anda birden fazla kamerada görülmemektedir. D9, D10 ve D11 kameralarının yerleşimi şekil 4.6’da görüldüğü gibidir.



Şekil 4.6 D9 (cam1), D10 (cam2), D11 (cam3) veri kümelerinde kameraların yerleşimi ve örnek görüntüleri (Zhu vd. 2016a)

Geliştirilen çevrimiçi aktivite optimizasyon yönteminin ilk olarak farklı parametre konfigürasyonları için başarımı analiz edilmiş ve parametrelerin başarımlar üzerindeki etkisi incelenmiştir. Sonrasında geliştirilen yöntem literatürdeki öncü yöntemler ile tüm veri kümeleri dikkate alınarak karşılaştırılmıştır.

4.3.1 Parametre analizi

Geliştirilen çevrimiçi aktivite optimizasyon yönteminin aşağıdaki gibi 4 farklı konfigürasyonu oluşturulmuş ve bu konfigürasyonlar D9 veri kümesi üzerinde karşılaştırılmıştır:

- 1) **IGN_CH** konfigürasyonu aktivitelerin kronolojik tutarlılığını (*Denklemler 3.46'de, $\alpha = 0$*) dikkate almamaktadır.
- 2) **IGN_CH_GR**, aynı şekilde kronolojik tutarlılığı dikkate almamasının yanında görüntüyü 4 eşit (2x2) bölgeye bölerek aday aktiviteleri görüntüye giriş bölgelerine göre gruplandırmaktadır. Her bir bölgeden sırayla aday seçilerek optimizasyona dahil edilmektedir. Bu yöntemin, görüntüdeki farklı bölgelerin eşit ağırlıkta

kullanılmasını sağlayarak video sinopsisin aktivite yoğunluğunu artırması beklenmektedir.

3) **PRE_CH**, aktivitelerin kronolojik tutarlılığını dikkate alarak optimizasyon yapmaktadır. Kronolojik tutarlılık (E_c) metriğinin ağırlığı ($\alpha = 1$) şeklinde kullanılarak kronolojik tutarlılığın ihlalinde enerji değeri bu oranda artırılmaktadır (Denklem 3.46).

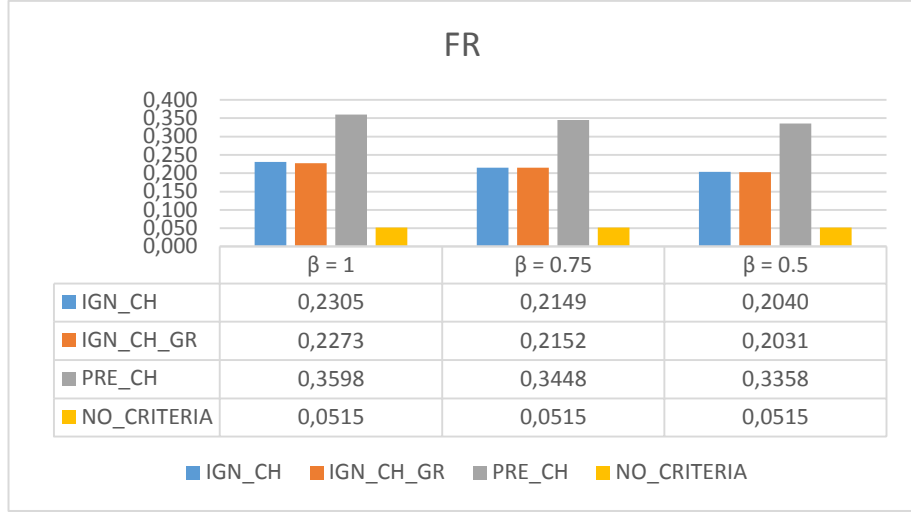
4) **NO_CRITERIA**, E_a dışındaki metrikleri dikkate almayarak aday aktiviteleri video sinopsise eklemektedir. En kısa video sinopsisi oluşturmaya rağmen özellikle kesişim metriği dikkate alınmadığı için görsel kalitesi çok düşüktür.

Yukarıdaki konfigürasyonların aynı senaryo için görsel çıktıları şekil 4.7’de görülmektedir.

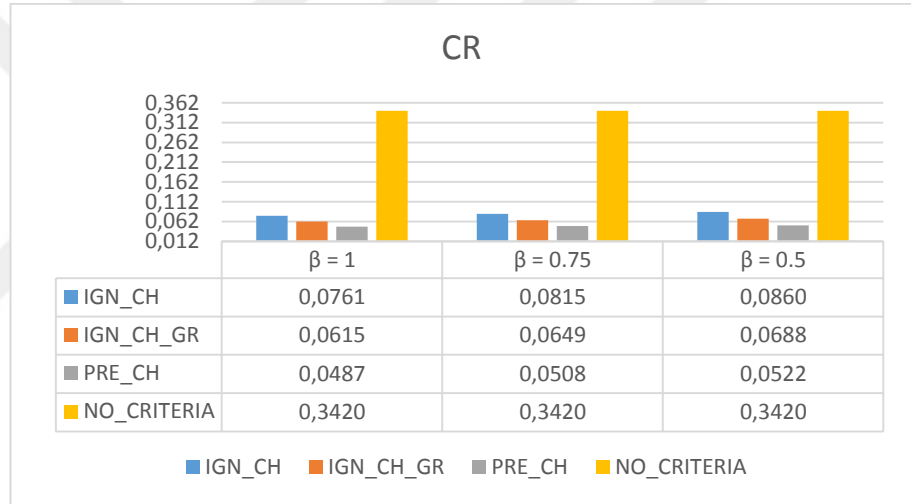


Şekil 4.7 Aynı senaryo için sırasıyla IGN_CH, IGN_CH_GR, PRE_CH ve NO_CRITERIA konfigürasyonlarının video sinopsis başlangıç durumları

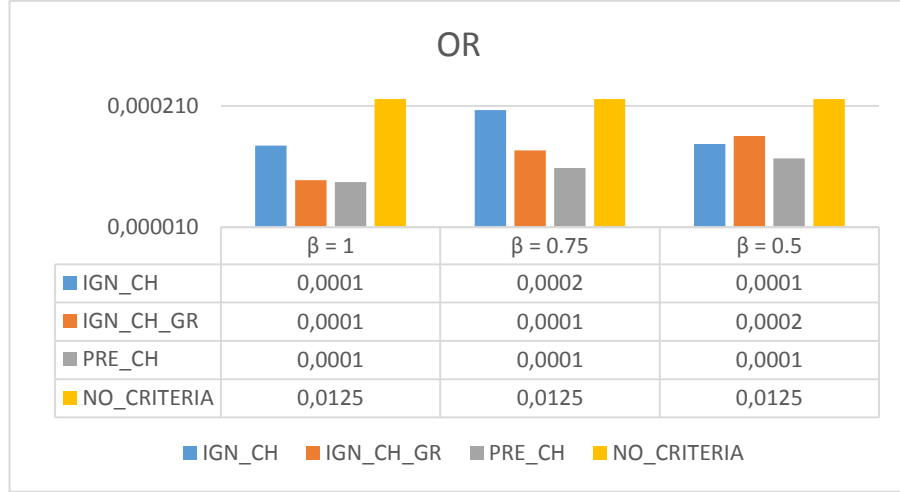
Bu konfigürasyonların başarımları sonuçları şekil 4.8-4.11’de gösterilmiştir. Şekillerde yer alan yatay eksen denklem 3.46’de yer alan E_o metriğinin ağırlığının (β) farklı değerlerini göstermektedir. $\beta = 1$ değeri bu karşılaştırmada kesişimi en yüksek oranda cezalandırırken en az aktivite kesişimine sahiptir. $\beta = 0.75$ ve $\beta = 0.5$ değerlerinde cezalandırma oranı düştüğü için oluşturulan video sinopsiste daha fazla nesne kesişimi görülmektedir.



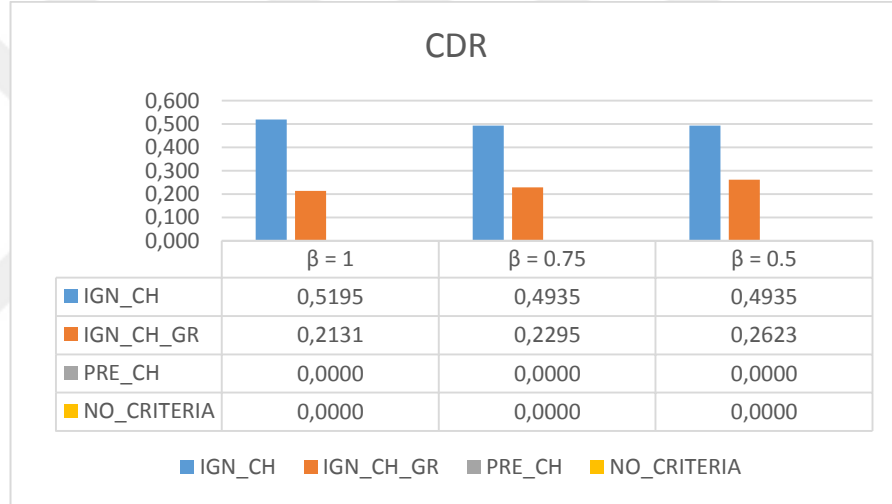
Şekil 4.8 FR sonuçları



Şekil 4.9 CR sonuçları

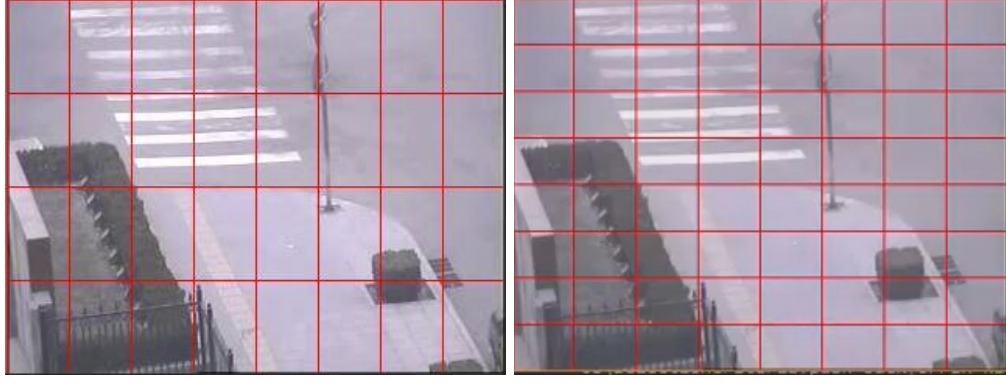


Şekil 4.10 OR sonuçları



Şekil 4.11 CDR sonuçları

Yukarıdaki sonuçlar görüntünün 8x4 olmak üzere toplam 32 gride bölündüğü durum için hesaplanmıştır. Grid boyutlarının yöntem başarımı üzerinde etkisini görmek için aynı senaryo şekil 4.12’de görüldüğü gibi görüntünün 8x8 olarak toplam 64 gride bölündüğü durum için gerçekleştirilmiştir.

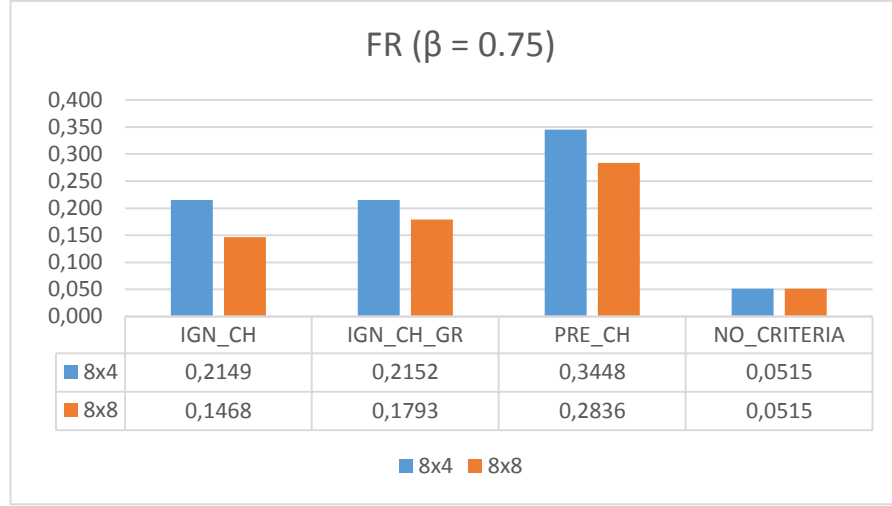


(a) 8x4

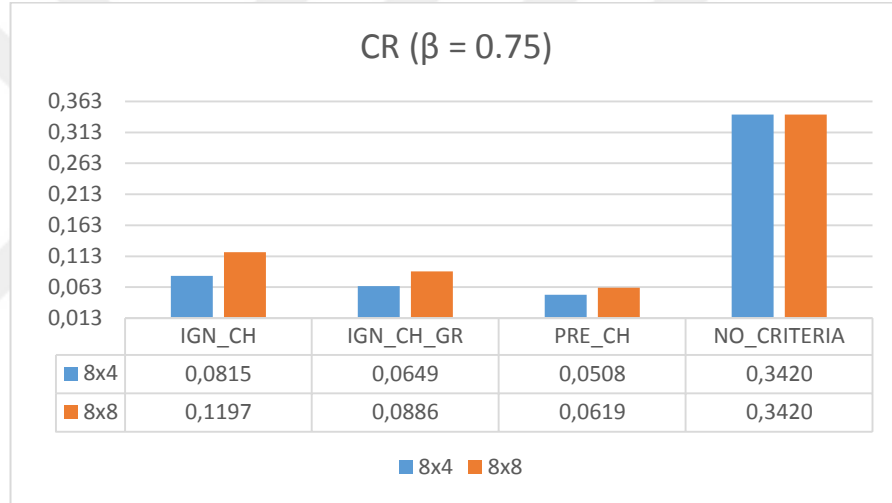
(b) 8x8

Şekil 4.12 Farklı boyutlardaki grid haritaları

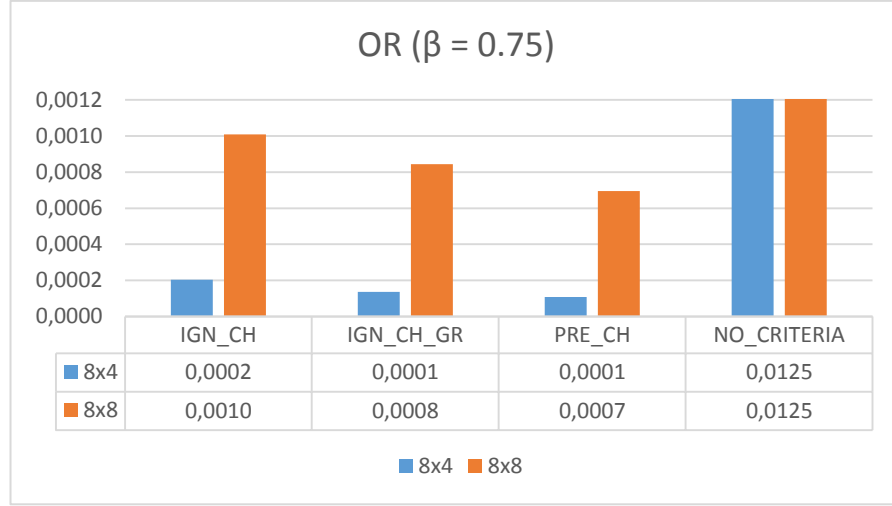
D9 veri kümesi üzerinde iki farklı grid boyutu için elde edilen sonuçlar şekil 4.13-4.16'da paylaşılmıştır. Bölüm 3.4'te bahsedildiği gibi geliştirilen yöntemde nesne izleri grid numaraları ile temsil edilip kesişim kontrolü bunun üzerinden yapılmaktadır. Bu veri kümelerinde grid haritası 8x4 boyutunda oluşturulduğunda grid boyutları nesne boyutları ile daha fazla uyuşmaktadır. Bu sebeple *OR* (Şekil 4.15) sonuçlarında görüldüğü gibi daha az kesişme yaşanmaktadır. Daha az kesişme yaşanmasından dolayı video sinopsis boyutu uzamaktadır (Şekil 4.13). 8x8 grid boyutunda aynı sayıda aktivite daha az video karesinde (daha düşük FR) daha fazla kesişim (daha yüksek *OR*) ile gösterildiği için video karelerinin aktivite yoğunluğunu temsil eden *CR* metriği daha yüksek çıkmaktadır. 8x4 grid boyutunda aktivitelerin başlangıç bölgesine göre gruplandırıldığı konfigürasyonda kronolojik tutarlılık başarımı daha yüksektir (Şekil 4.16). Şekil 4.11'de görüldüğü gibi *IGN_CH_GR*, *IGN_CH* konfigürasyonuna göre kronolojik tutarlılığı daha fazla korumaktadır. Grid boyutunun 8x4 olduğu durumda nesne kesişimi daha az olduğu için gruplandırılmış aktiviteler arasında kronolojik tutarlılığı bozma ihtimali azalmaktadır.



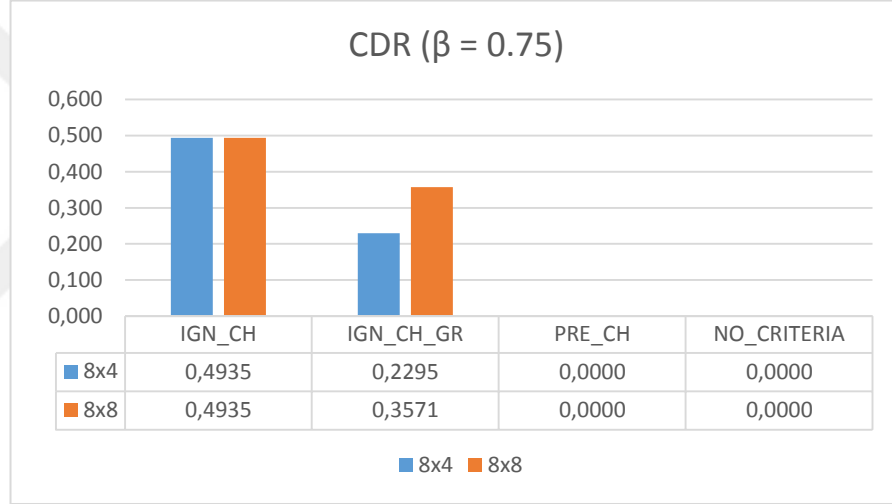
Şekil 4.13 8x4 ve 8x8 grid boyutları için FR sonuçları



Şekil 4.14 8x4 ve 8x8 grid boyutları için CR sonuçları



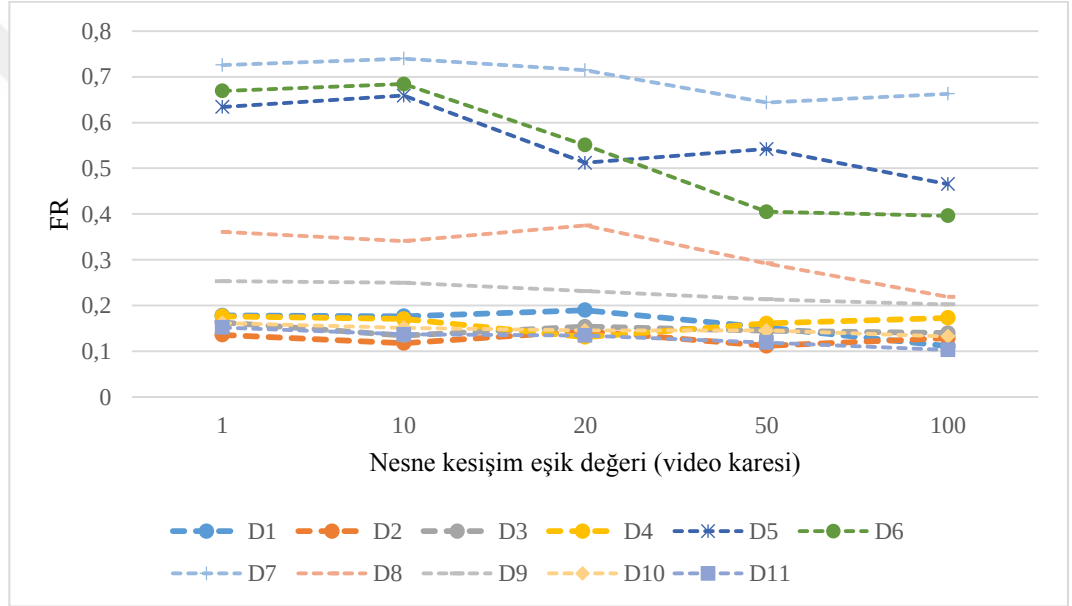
Şekil 4.15 8x4 ve 8x8 grid boyutları için OR sonuçları



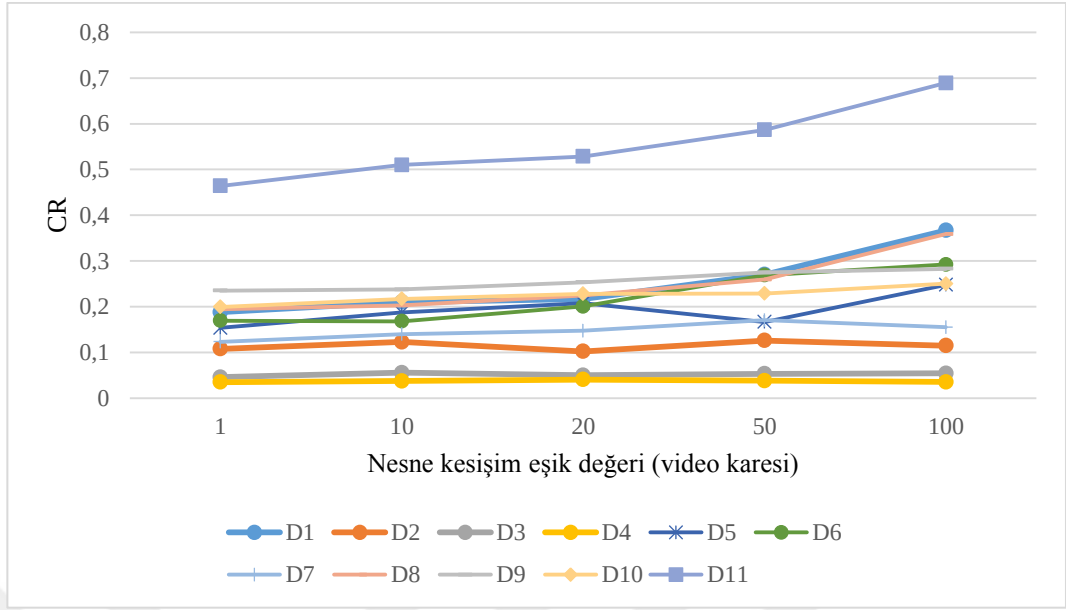
Şekil 4.16 8x4 ve 8x8 grid boyutları için CDR sonuçları

Başarım üzerindeki etkisinin analiz edilmesi gereken diğer bir parametre nesne kesişim eşik değeridir. Bu değer aktivitelerin birbiri ile uzamsal kesişimini hesaplarken izin verilen maksimum sayıdaki video karesini temsil etmektedir. İki aktivite bu eşik değerinden daha fazla sayıda video karesi boyunca kesişiyor ise bu aktiviteler birbiriyle kesişiyor olarak etiketlenmektedir. Nesne kesişim eşik değerinin küçük olması durumunda uzamsal kesişim daha az tolere edileceği için şekil 4.19’da görüldüğü gibi oluşturulan video sinopsisin OR değeri daha düşük çıkmaktadır. Bu da aktivite kesişimlerinin daha az olduğunu göstermektedir. Aynı sayıda aktivitenin daha az kesişim ile gösterilmesi sonucunda oluşturulan video sinopsisin uzunluğu artmakta

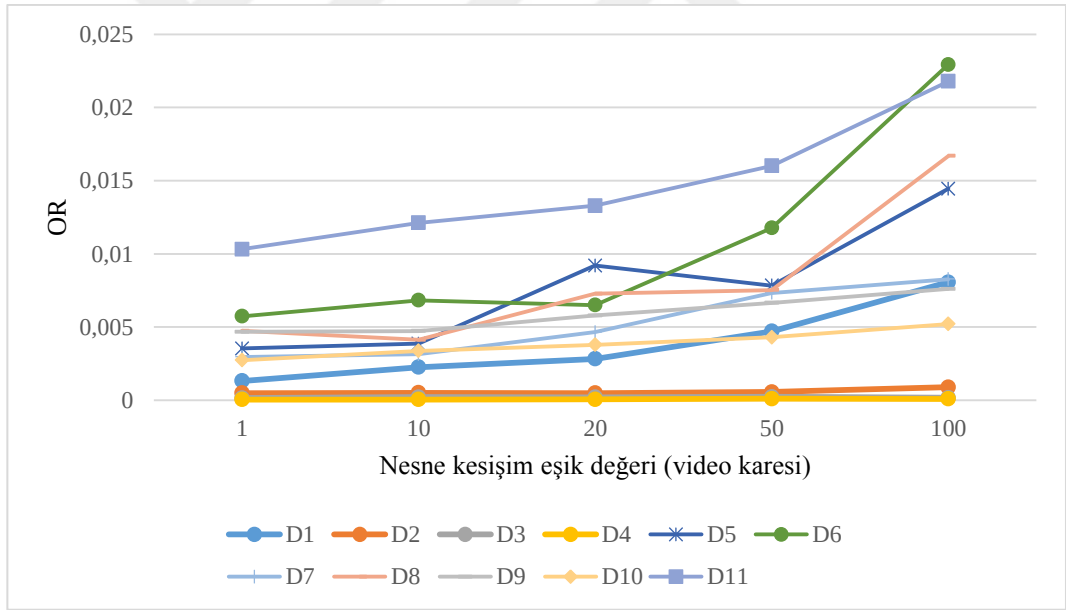
dolayısıyla şekil 4.17’de görüldüğü gibi daha yüksek *FR* değerine sahip olmaktadır. Bu durum oluşturulan video sinopsisteki video karesi başına düşen aktivite yoğunluğunu azaltmakta ve şekil 4.18’de görüldüğü gibi daha düşük *CR* değerine sahip olmasına sebep olmaktadır. İlgili şekillerde görüldüğü gibi kesişim eşik değeri arttıkça oluşturulan video sinopsiste aktivitelerin birbiriyle kesişim oranı artmakta (daha yüksek *OR*), bunun sonucunda oluşturulan video sinopsis kare başına düşen aktivite bakımından daha yoğun (daha yüksek *CR*) ve uzunluk olarak daha kısa (daha düşük *FR*) hale gelmektedir. Şekil 4.20’de görüldüğü gibi nesne kesişim eşik değerinin aktivitelerin kronolojik tutarlılığı üzerinde anlamlı bir etkisi görülmemektedir.



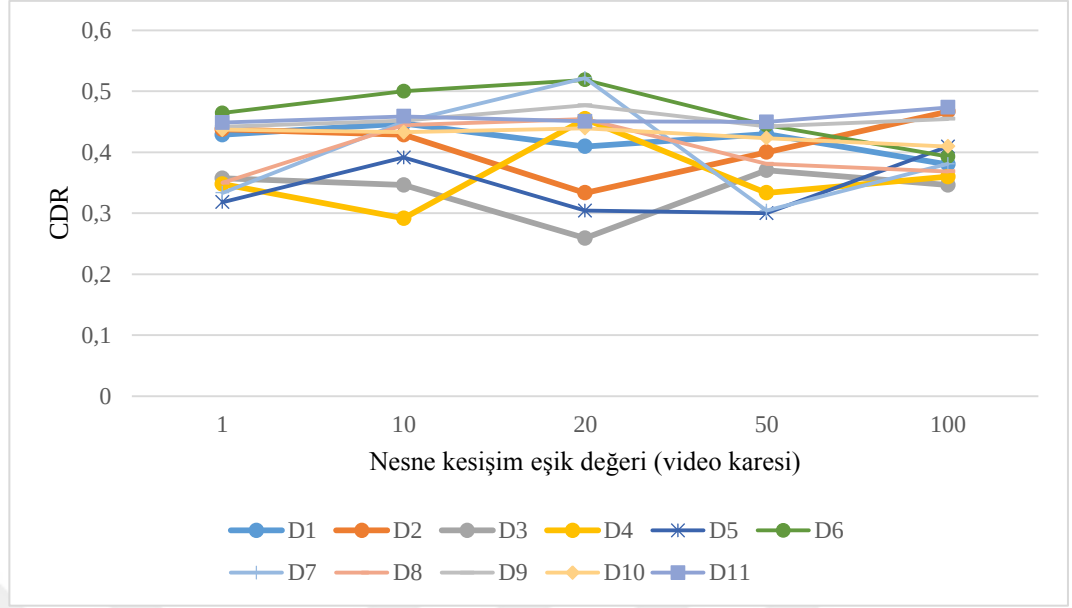
Şekil 4.17 Nesne kesişim eşik değerinin *FR* üzerindeki etkisi



Şekil 4.18 Nesne kesişim eşik değerinin CR üzerindeki etkisi



Şekil 4.19 Nesne kesişim eşik değerinin OR üzerindeki etkisi

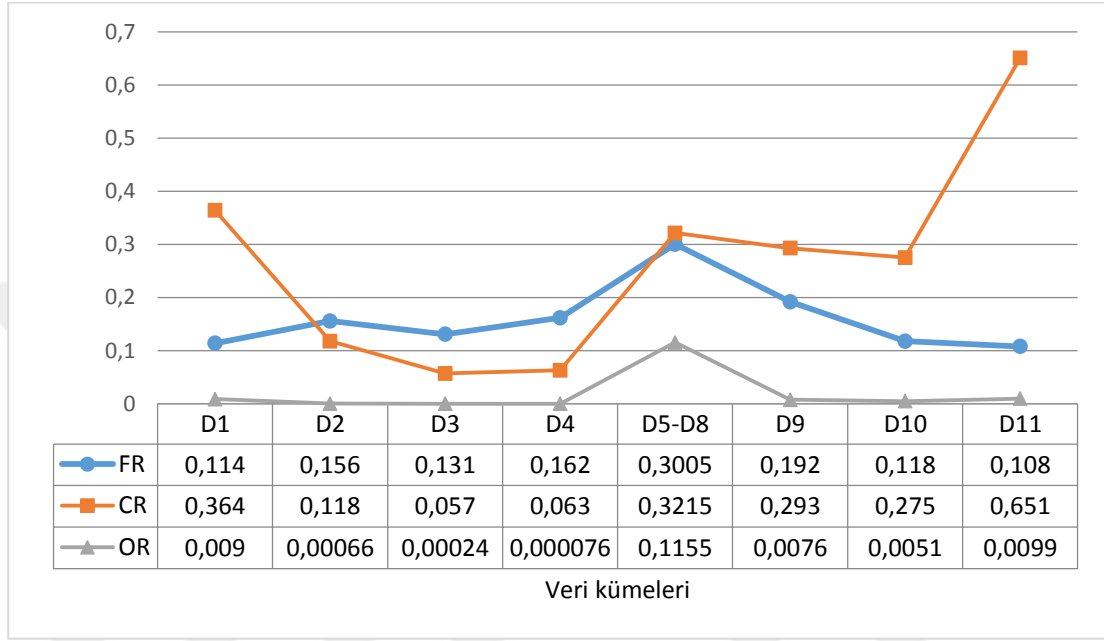


Şekil 4.20 Nesne kesişim eşik değerinin CDR üzerindeki etkisi

Geliştirilen grid haritası tabanlı çevrimiçi aktivite optimizasyon yönteminin hesaplama karmaşıklığı o an sahnede algılanan aktivite sayısı (K) ve grid haritasının video karesi olarak toplam uzunluğunun q parametresine oranı (grid haritasında her q karede bir karşılaştırma yapıldığı için) olarak belirlenen (M) uzunluğuna bağlıdır. Buna göre bir video karesini işlemenin hesaplama karmaşıklığı $O(K M)$ olmaktadır. Optimal pozisyonları hesaplanan aktiviteler grid haritasında gösterilecekleri konuma yerleştirildiği için grid haritası video sinopsisin anlık durumunu temsil etmektedir. Dolayısıyla algılanan aktivitelerin tek tek video sinopsiste yer alan aktiviteler yerine grid haritasıyla karşılaştırılması yeterli olmaktadır. Grid haritasının toplam işlemsel uzunluğu (M), video sinopsiste yer alan aktivitelerin toplam uzunluğundan (S) daha düşük olduğu için hesaplama karmaşıklığı aynı oranda azalmaktadır. T video karesine sahip bir videonun toplam hesaplama karmaşıklığı ise $O(T K M)$ olmaktadır.

4.3.2 Literatürdeki öncü yöntemlerle karşılaştırma

Bu bölümde öncelikle geliştirilen çevrimiçi optimizasyon yönteminin şekil 4.21’de görüldüğü gibi 11 veri kümesi üzerinde *FR*, *CR* ve *OR* sonuçları paylaşılmıştır.



Şekil 4.21 Geliştirilen yöntemin veri kümeleri üzerindeki *FR*, *CR* ve *OR* sonuçları

FR sonuçları incelendiğinde geliştirilen yöntemin D5, D6, D7 ve D8 veri kümelerinde performansının diğer veri kümelerine göre daha düşük olduğu görülmektedir. Bu veri kümeleri 4 kameranın farklı açılardan ortak bir alana baktığı görüntülerden oluşmaktadır. Diğer veri kümeleri güvenlik kameralarının sürekli yayın akışını temsil ederken D5-D8 veri kümelerinde görüntüler aktivitelerin görüntüye girmesi ile başlayıp görüntüden çıkmaları ile sonlanmaktadır. Aktiviteler arasında sahnenin boş olduğu herhangi bir an bulunmamaktadır. Ayrıca kullanılan kameralar sahneye yakın ve ortalama insan yüksekliğinde olduğu için aktivite kesişimi yüksek seviyededir. Bu yüzden toplam video karesi sayısındaki azalma diğer veri kümelerine göre daha düşük olmuştur. *CR* değeri sahnenin aktivite yoğunluğunun fazla olduğu (D1, D9, D10) ve kameranın sahneye yakın olması sebebiyle aktivitelerin görüntüde büyük alanlar kapladığı (D5-D8, D11) veri kümelerinde daha yüksek çıkmıştır. *OR* değeri D5-D8 veri kümelerinde yukarıda bahsedildiği gibi tüm aktivitelerin görüntünün ortasında

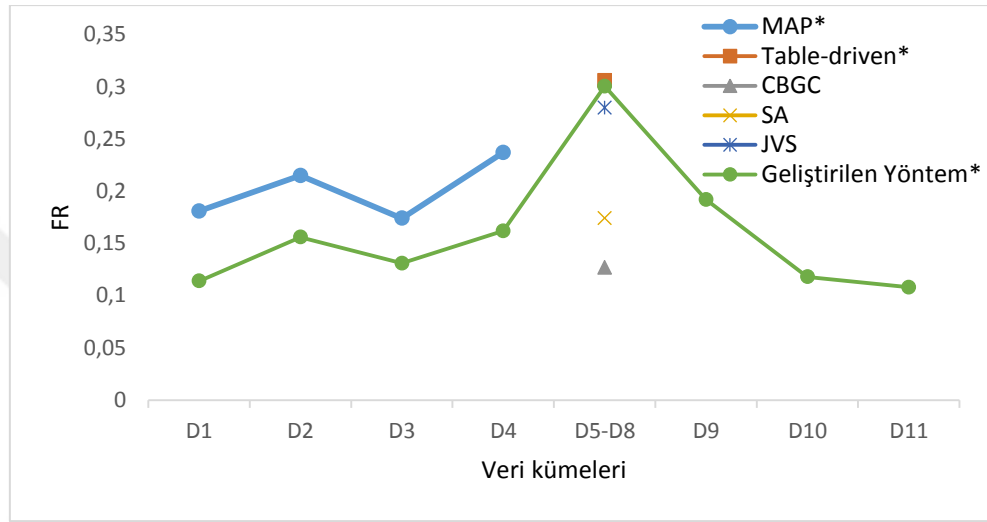
kesişmesinden dolayı diğer veri kümelerine göre normalin çok üstünde olduğu görülmüştür.

Geliştirilen aktivite optimizasyon yöntemi literatürde öncü 6 farklı yöntem ile karşılaştırılmıştır. Karşılaştırılan yöntemlerden 4'ü çevrimdışı, 2'si ise çevrimiçi yöntemlerdir. Çevrimdışı yöntemlerden ilki, Pritch vd. (2008) tarafından önerilen optimizasyon yöntemidir. Bu yöntem global bir enerji fonksiyonu tanımlayıp “graph-cut” yöntemiyle bu fonksiyonu minimuma indirgeyerek aktivitelerin optimal yerleşimini bulmaktadır. Karşılaştırmada kullanılan ikinci çevrimdışı yöntem, Zhu vd. (2016a) tarafından önerilen JVS yöntemidir. Bu yöntem, çok kameralı topolojide video sinopsis oluşturmaktadır. Üçüncü çevrimdışı yöntemde, Mahapatra vd. (2016) tarafından önerilen “contradictory binary graph coloring (CBGC)” yaklaşımı optimize edilecek aktiviteleri bir çizge üzerine yerleştirmektedir. Oluşturulan çizge, aktivitelerin kesişim durumlarına göre siyah ve beyaz olmak üzere iki renge ayrılarak kesişmeyen aktivitelerin bir grup olarak beraber gösterilmesi sağlanmaktadır. Son olarak yine Mahapatra vd. (2016) “simulated annealing (SA)” tabanlı çevrimdışı bir yöntem önermektedir.

Karşılaştırmada kullanılan çevrimiçi yöntemlerin ilki Mahapatra vd. (2016) tarafından önerilen “Table-driven” yaklaşımıdır. Bu yöntem aktivitelerin birbirine göre kesişim durumlarını saklayan bir tablo oluşturmaktadır. Yeni algılanan aktiviteler bu tablo üzerindeki aktiviteler ile karşılaştırılarak uzamsal kesişim olmayacak şekilde optimal pozisyonları hesaplanmaktadır. Diğer çevrimiçi yöntem ise Huang vd. (2014) tarafından önerilen MAP yöntemidir. Bu yöntem görüntüde her pikselin temsil edildiği bir sinopsis tablosu oluşturarak aktivite optimizasyonu problemini çözmeyi amaçlamaktadır. Sinopsis tablosu her bir pikselde bulunan aktiviteyi ve ilgili aktivitenin o pikselde en son ne zaman görüldüğünü tutarak yeni aktivitelerin minimum kesişim ile optimal pozisyona yerleştirilmesini amaçlamaktadır.

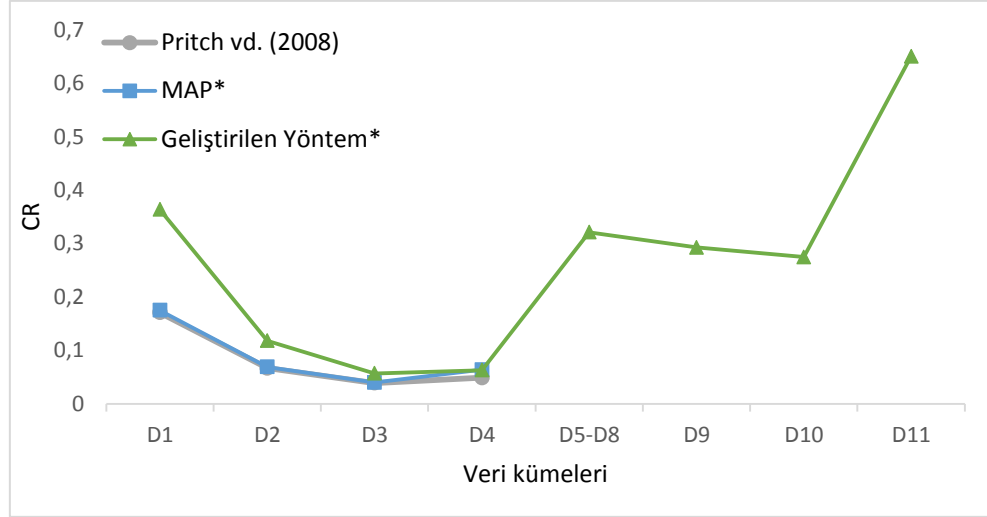
Yöntemlerin performans karşılaştırmasında *FR* ve *CR* metrikleri kullanılmıştır. Her bir metriğin veri kümeleri üzerindeki sonuçları sırasıyla şekil 4.22 ve şekil 4.23'te paylaşılmıştır. Karşılaştırmada kullanılan yöntemlerin kaynak kodları paylaşılmadığı

için yöntemler tüm veri kümeleri üzerinde uygulanamamıştır. Şekillerde, yöntemlerin önerildiği çalışmalarda paylaşılan sonuçlar kullanılmıştır. Bunun için sadece ilgili çalışmada kullanılan veri kümelerine karşılık gelen sonuçlar yer almaktadır. Çevrimiçi yöntemler şekillerde ‘*’ ile belirtilmiştir. JVS, CBGC, SA ve “Table-driven” yöntemleri birden fazla kamera görüntüsüne uygulanırken, MAP ve Pritch vd. (2008) tarafından önerilen yöntem tek kamera görüntüsü üzerinde uygulanmıştır.



Şekil 4.22 FR karşılaştırması ($\alpha = 1$, $\beta = 0.5$, kesişim eşik değeri = 100)

Şekil 4.22’de görüldüğü gibi çevrimdışı yöntemlerin oluşturduğu video sinopsis uzunluğu çevrimiçi yöntemlere daha kısadır. Çevrimdışı yöntemler tüm aktiviteleri beraber değerlendirdiği için toplamda en kısa video sinopsisi oluşturacak optimal aktivite dağılımına daha fazla yaklaşmaktadır. Geliştirilen yöntem diğer çevrimiçi yöntemler olan MAP ve “Table-driven” yöntemlerine göre daha iyi sonuç vermiştir. Aynı veri kümeleri üzerinde geliştirilen yöntem daha kısa video sinopsis oluşturmuştur.

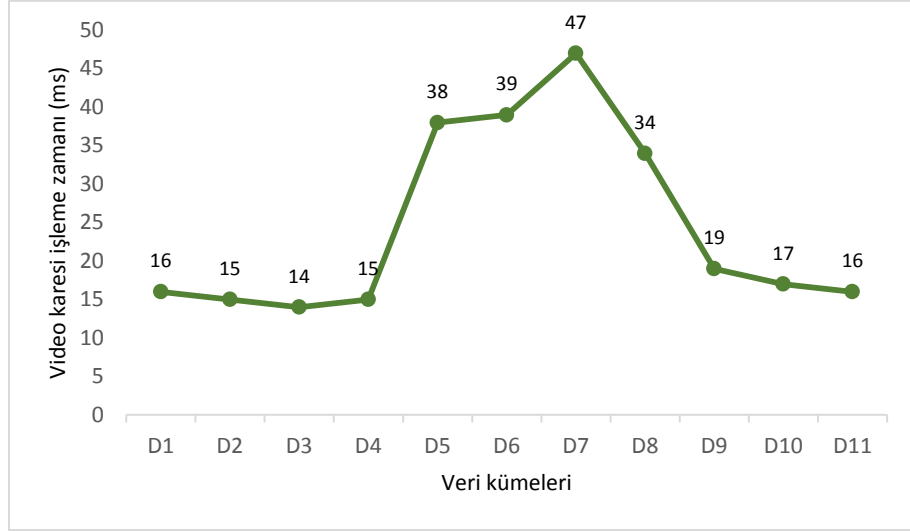


Şekil 4.23 CR karşılaştırması ($\alpha = 1$, $\beta = 0.5$, kesişim eşik değeri = 100)

Şekil 4.23’de geliştirilen yöntemin diğer iki yöntem ile CR karşılaştırması görülmektedir. Sonuçlara göre geliştirilen yöntem tarafından oluşturulan video sinopsiste video karesi başına düşen aktivite yoğunluğu daha yüksektir. Geliştirilen yöntem diğer çevrimiçi yöntemlere göre ortak olarak değerlendirilen veri kümeleri üzerinde *FR* değerlerinde %16, *CR* değerinde %73 oranında daha iyi sonuç vermiştir.

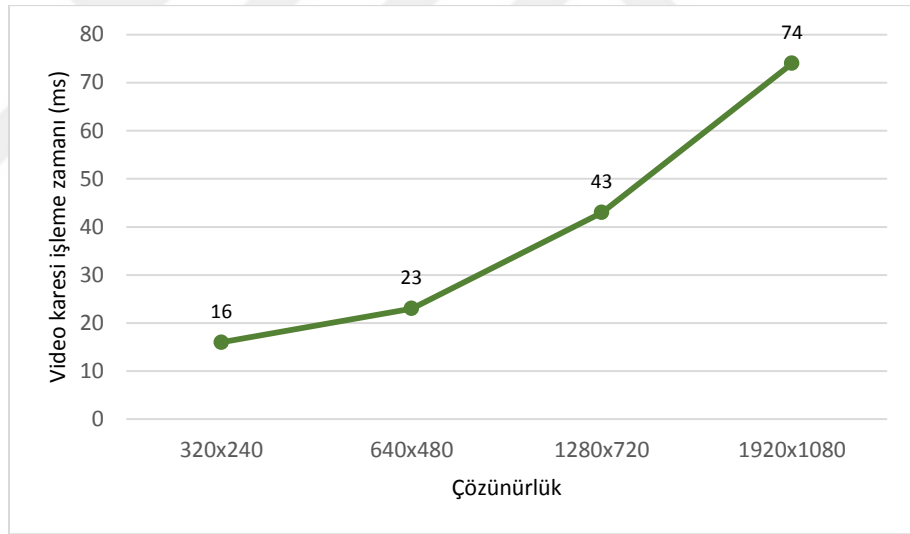
4.4 Çalışma Zamanı Performans Analizi

Bu bölümde geliştirilen video sinopsis sisteminin çalışma zamanı performansı Intel® Xeon® Quad Core 3.40 GHz işlem gücüne ve 8 GB belleğe sahip bilgisayar üzerinde analiz edilmiştir. Paylaşılan süreler video karesinin alınmasından optimizasyon sonucunda aktivitelerin optimal pozisyonlarının bulunmasını kapsayan tüm işlemleri içermektedir. İlk olarak şekil 4.24’te yöntemin 11 veri kümesi üzerindeki ortalama kare işleme süreleri paylaşılmıştır. Çizelge 4.4’te paylaşıldığı gibi çözünürlüğü diğerlerine göre daha yüksek olan D5, D6, D7 ve D8 veri kümelerinde ortalama video karesi işlem süresi daha yüksektir. Bu veri kümelerinde kameraların aktivitelere çok yakın olması dolayısıyla aktivitelerin sahnede oran olarak daha fazla alan kaplamasının da etkisi bulunmaktadır.



Şekil 4.24 Veri kümelerinin ortalama video karesi işleme süreleri (ms)

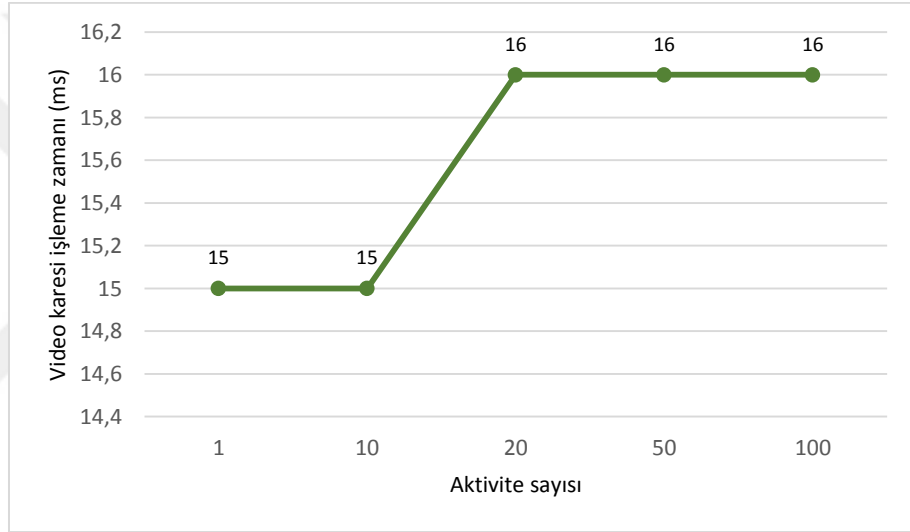
Şekil 4.25'te D1 veri kümesinin farklı çözünürlükleri için işlem süreleri paylaşılmıştır.



Şekil 4.25 Kamera çözünürlüğünün çalışma zamanı üzerindeki etkisi

320x240 çözünürlüğündeki görüntüyü işlemek için gereken süre yaklaşık 16 ms iken görüntü çözünürlüğü 1920x1080'e çıktığında işlem süresi yaklaşık 4.6 katına çıkarak 74 ms olmaktadır. Geliştirilen video sinopsis sisteminde çözünürlüğün artmasından en fazla etkilenen yöntem piksel tabanlı çalışmasından dolayı hareket algılama olarak görülmüştür.

Son olarak optimizasyon için çevrimiçi olarak bekleyen aktivite sayısının (A_c : bölüm 3.4) performans üzerindeki etkisi analiz edilmiştir. Geliştirilen sistemde kamera görüntüleri üzerinde algılanan aktiviteler sonlandırılmalarından hemen sonra aktivite optimizasyon işlemine gönderilmektedir. Aktivite optimizasyonunda optimal pozisyonunun tespiti için aynı anda bekleyen aktivite sayısındaki artış optimizasyon havuzunu genişletmektedir. Fakat uygulanan grid haritası yaklaşımı sayesinde aktivitelerin birbiri ile karşılaştırması bit seviyesinde işlemlerde yapılmaktadır. Bu sayede aktivite sayısındaki artış şekil 4.26’da görüldüğü gibi ciddi bir performans kaybına sebep olmamıştır.



Şekil 4.26 Optimizasyon için bekleyen aktivite yoğunluğunun çalışma zamanı üzerindeki etkisi (D1 veri kümesi)

Bu bölümde kamera çözünürlüğü, kamera sayısı ve aday aktivite sayısı gibi çalışma zamanını doğrudan etkileme potansiyeli olan etkenlerin geliştirilen video sinopsis sistemi üzerindeki etkisi incelenmiştir. Bahsedilen değişkenler referans alınarak geliştirilen video sinopsis sisteminin bir bilgisayar üzerindeki performans sınırlamaları belirlenmiştir.

5. SONUÇ

Tez kapsamında çok kameralı nesne takibi tabanlı çevrimiçi video sinopsis sistemi geliştirilmiştir. Her bir kamera görüntüsü üzerinde algılanan hareketli nesneler kamera özelinde ve kameralar arasında takip edilerek aktivite haline getirilmiştir. Elde edilen aktiviteler zaman düzleminde kaydırılarak en kısa sürede en fazla aktiviteyi gösterecek şekilde optimize edilmiştir. Optimal gösterim zamanları belirlenen aktiviteler her bir kamera içinde video sinopsis olarak gösterilirken, kameralar arasında görülme sıralarına göre paylaşılmıştır. Herhangi bir aktivite ilk görüldüğü kameraya ait aktivite havuzuna dahil edilip o kameraya ait video sinopsis içinde gösterildikten sonra sıradaki kameraya ait aktivite havuzuna aktarılmıştır. Bu sayede aktiviteler kamera özelinde video sinopsis olarak, kameralar arasında ise kronolojik sıra ile gösterilmiştir.

Günümüzde kamera kullanımındaki bireysel ve kurumsal anlamdaki artış dikkate alındığında görsel verinin anlamlandırılıp otomatik olarak etkili bir şekilde özetlenmesi kameraların kullanıldığı her alanda önemli bir ihtiyacı karşılamaktadır. Video kayıtları üzerinde aranılan bir olayın manuel olarak geriye yönelik taranması ve raporlanması zaman alan ve masraflı bir işlemdir. Bu anlamda geliştirilen video sinopsis sistemi çok geniş bir uygulama alanında görsel verinin etkili bir şekilde taranmasını sağlamaktadır. Video sinopsis içinde gösterilen aktiviteler sahnede algılanan herhangi bir hareket olabileceği gibi, kontrollü geçiş sistemlerinde algılanan araç plakaları veya insan yüzleri gibi farklı video analiz sistemlerinin çıktısı olabilir. Bu yönden bakıldığında görsel verinin kullanıldığı herhangi bir durumu kapsayan geniş bir kullanım alanına hitap etmektedir.

Tez çalışması sonucunda genel olarak video işleme, özel olarak ise video sinopsis alanında literatüre hem metodolojik hem de kuramsal katkılar sağlanmıştır. Metodolojik olarak aktivitelerin coğrafi konumları video sinopsis literatüründe ilk defa kullanılmış ve görüntüsü kesişmeyen kameralar üzerinde çok kameralı çevrimiçi video sinopsis yaklaşımı ilk defa araştırılmıştır. Mevcut video sinopsis yöntemlerinin tamamı nesneleri piksel koordinatlarına göre etiketlemektedir. Tez kapsamında geliştirilen yöntemde kamera kalibrasyonu ile nesnelerin kameraya göre coğrafi konumları tespit edilmiştir.

Kameranın coğrafi konumunun bilinmesiyle hedef nesneler coğrafi konumları ile de etiketlenmiştir. Coğrafi konumlarının bilinmesi nesnenin maksimum hızı gibi metrikleri kullanarak çok kameralı nesne takibini desteklemesinin yanında, belirli bir coğrafi alana ait aktivitelerin (hangi kamerada olduğu fark etmeksizin) video sinopsisinin oluşturulmasına imkan sağlamaktadır. Bu sayede güvenlik kameralarında herhangi bir olay tespit edildikten sonra coğrafi filtreleme ile çevresindeki kameralarda olayın öncesi ve sonrası kolay bir şekilde takip edilebilmektedir.

Literatürdeki video sinopsis çalışmaları incelendiğinde, çalışmalarda genellikle aktivite optimizasyon işlemi üzerine odaklanılmıştır. Bu çalışmaların aksine, tez kapsamında video sinopsis tüm algoritmik adımları ile birlikte bir bütün olarak ele alınmıştır. Genel başarıımı düşüren problemler tespit edilip her adımda gerekli iyileştirmeler yapılmıştır. Kuramsal olarak her adıma yapılan katkı aşağıdaki gibi özetlenmiştir:

1. Nesne algılama aşamasında yanlış algılamalar gereksiz aktivitelerin video sinopsise dahil edilmesine sebep olmaktadır. Bu nedenle özellikle dış ortamda gürültü ve aydınlatma değişimi, sahneye ait dinamik arka plan (bayrak, su, vb.) gibi çevresel koşullardan minimum düzeyde etkilenen bir yöntemle ihtiyaç duyulmaktadır. Bu aşamada yanlış algılama sonucu gereksiz olarak üretilen nesneler sonraki adımların çalışma zamanı performansını ve başarılarını düşürmektedir. Literatürde kabul görmüş PBAS hareket algılama yöntemi, çevresel koşullara karşı dayanıklılığı artacak şekilde iyileştirilmiştir. Dış ortamda sahnede oluşabilecek ani değişimlere karşı yöntemin dayanıklılığını artırmak adına ani değişim algılama ve bu durumda mümkün olan en hızlı şekilde değişime uyum sağlama yeteneği eklenmiştir.
2. Nesne takibi, optimizasyonu besleyecek aktiviteleri oluşturması açısından video sinopsisin önemli bir adımıdır. Hedef nesnenin görüntüye girişinden çıkışına kadar kesintisiz takip edilmesi gerekmektedir. Nesne takibinin yetersiz kalması sonucunda aynı nesne için birden fazla aktivite oluşturulması optimizasyon işleminin karmaşıklığını artırmasının yanında video sinopsiste aynı nesnenin farklı zamanlarda gösterilmesine sebep olarak anlamsal bütünlüğü bozmaktadır. Bu sorunları çözmek adına özellikle hedef nesnenin ölçek, görünüm

değişimlerine, başka nesneler tarafından engellenmesi ve çevresel koşullara karşı dayanıklı yeni bir takip yöntemi geliştirilmiştir. Uzun süreli takibi sağlamak için hedef nesnenin birden fazla model ile temsil edildiği yöntem tespit edilen sorunları çözerek bölüm 4.2’de tartışıldığı gibi literatürdeki öncü yöntemlerden daha iyi sonuç vermiştir. Tek kamera için geliştirilen bu yöntem hedeflenen çok kameralı video sinopsisi oluşturmak için görüntüsü kesişmeyen birden fazla kamera görüntüsünden nesne takibi yapacak şekilde genişletilmiştir.

3. Aktivite optimizasyonu video sinopsisin en önemli adımıdır. Optimizasyon işlemi bir grup aktivitenin zaman düzleminde toplamda en kısa sürede gösterilecek şekilde optimal dağılımını bulmayı amaçlamaktadır. Optimizasyon probleminin evrensel kümesi aktivitelerin tüm olası dağılımları olduğu için problemin çözümü oldukça karmaşık ve masraflıdır. Aktivite sayısının artmasıyla birlikte karmaşıklık üstsel olarak artmakta dolayısıyla gereken bellek ve işlem gücü de aynı oranda artmaktadır. Aktivite optimizasyonu işleminin hesaplama darboğazını tüm olası dağılımlar için tekrar tekrar hesaplanan aktivite kesişim kontrolü oluşturmaktadır. Geliştirilen grid haritası tabanlı aktivite optimizasyon yöntemi ile karşılaştırma işlemi bit seviyesine indirgenip hızlı bir şekilde gerçekleştirilmiştir. Bu sayede aktivitelerin birbiri ile kesişimleri bitsel operasyonlar ile etkili bir şekilde hesaplanarak optimizasyonun gerçek zamanlı çalışması önündeki en önemli darboğaz çözülmüştür. Buradan elde edilen çalışma zamanı kazancı ile literatürdeki çevrimiçi yöntemlerde kullanılanlara göre daha kapsamlı bir enerji fonksiyonu kullanılmıştır. Geliştirilen aktivite optimizasyon yöntemi ile video sinopsis güvenlik kameraları üzerinde gerçek zamanlı ve başarılı bir şekilde uygulanabilir hale gelmiştir. Bölüm 4.3’te paylaşılan karşılaştırma sonuçları geliştirilen yöntemin literatürdeki öncü yöntemlere göre ne kadar etkili olduğunu açıkça göstermektedir.

Yukarıda bahsedilen kuramsal katkıların dışında çevrimiçi video sinopsis oluşturmak için gereken gerçek-zamanlı performansı sağlamak adına paralel programlama teknikleri kullanılarak yeni bir video sinopsis kütüphanesi geliştirilmiştir. Algoritmik akışta yer alan her bir yöntem bu kütüphane üzerinde uygulanmıştır. Literatürde yer alan çevrimiçi video sinopsis yöntemleri optimizasyon işleminin gerçek zamanlı olarak çalışmasına odaklanmıştır. Tez çalışması ile kamera görüntüsünün alınmasından video

sinopsisin oluşturulmasına kadar tüm aşamaların gerçek zamanlı olarak gerçekleştirildiği kapsamlı bir kütüphane oluşturulmuştur. Oluşturulan kütüphanenin sonraki bilgisayarlı görü projelerinde kullanılma imkânı bulunmaktadır.

5.1 Öneriler

Geliştirilen çok kameralı video sinopsis sisteminde her bir kameraya ait işlemler ayrı iş parçacıkları üzerinde paralel olarak uygulanmaktadır (Şekil 3.1). Bahsedilen paralelleştirme aynı bilgisayar üzerinde gerçekleşmektedir. Bu paralelleştirme dağıtık bir mimaride farklı bilgisayarların işlem gücünden yararlanılarak yapılabilir. Çok kameralı nesne takip havuzu farklı bilgisayarlar tarafından beslenecek şekilde bilgisayarlar arası bir iletişim modülü ile desteklenerek dağıtık mimaride çalışır hale getirilebilir. Bu sayede mimari seviyesinde çalışma zamanı performansı artırılabilir. Bahsedilen dağıtık işleme mimarisi kapalı devre kamera sistemlerinin dağıtık yapısına da uyumlu olması açısından ağ üzerindeki işlem yükünün de tek bir bilgisayar üzerinde darboğaz oluşturmak yerine homojen olarak dağılmasını sağlayacaktır. Geliştirilen yöntemin dağıtık mimariye taşınması bulut teknolojisi üzerinde internet üzerinden paylaşılan kamera görüntüsü ve videoların daha geniş bir kapsamda etkili bir şekilde video sinopsisinin oluşturulmasına imkan sağlayacaktır. Örnek olarak ülke genelinde şehir güvenliğinde kullanılan ve aynı ağ üzerinden erişilebilen mobese kameralarının gerçek zamanlı olarak analizi ile günlük veya belirli zaman aralıklarına ait video sinopsisi üretilerek otomatik görsel raporlama sağlanabilir.

Metodolojideki adımlar incelendiğinde işlemsel darboğazı aktivite optimizasyonunun oluşturduğu görülmektedir. Hesaplama karmaşıklığı yüksek olan bu optimizasyon işlemini grafik işlemci ünitelerinin (GPU) çok çekirdekli yapısından faydalanarak daha hızlı bir şekilde gerçekleştirme imkanı bulunmaktadır. GPU teknolojisindeki son yıllardaki sıçrama dikkate alındığında, optimizasyon işleminin GPU üzerinde uygulanması çalışma zamanı performansını önemli ölçüde artırabilir. Bu anlamda, aktivite optimizasyon işlemi derin pekiştirmeli öğrenme (deep reinforcement learning) yöntemleri kullanılarak daha etkili bir şekilde yapılabilir.

Aktivite optimizasyonunda her bir nesne, grid haritası üzerinde işgal ettiği grid ile temsil edilmektedir. Grid boyutu ve nesne boyutunun birbirine göre tutarlı olması, nesne kesişimlerinin daha doğru temsil edilmesine sebep olmaktadır. Bu sayede, video sinopsisin görsel kalitesi artmaktadır. Görüntüyü sabit boyuttaki gridlere bölmek yerine uzaklık, yoğun olarak görülen nesne tipi (araç, insan, vs.) gibi kriterlere göre grid boyutlarını yerel olarak belirlemek, aktivite optimizasyon başarımını artıracaktır. Ayrıca algılanan nesneler tip, renk, hareket gibi kriterlere göre sınıflandırılarak, benzer aktivitelerin video sinopsiste beraber gösterilmesi görsel kaliteyi artıracaktır.

Hareketli nesne takibinde hareket bloklarının kopmalarından dolayı takip edilen nesne boyutu, anlık olarak yarıya düşüp tekrar eski haline dönebilmektedir. Bu durumu engellemek için nesneye ait ortalama boyut bilgisi saklanabilir. Belirlenen ortalamanın çok üstünde veya altındaki algılamalarda korelasyon filtresi anlık algılama yerine ortalama boyutu dikkate alarak, takibin daha kararlı hale gelmesi sağlanabilir.

KAYNAKLAR

- Ahuja, R.K., Magnanti, T.L. and Orlin, J.B. 1989. Network flows. Handbooks in Operations Research and Management Science. [https://doi.org/10.1016/S0927-0507\(89\)01005-4](https://doi.org/10.1016/S0927-0507(89)01005-4)
- Babenko, B., Belongie, S. and Yang, M.H. 2009. Visual tracking with online multiple instance learning. In 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009. <https://doi.org/10.1109/CVPRW.2009.5206737>
- Bae, S.H. and Yoon, K.J. 2014. Robust Online Multi-object Tracking Based on Tracklet Confidence and Online Discriminative Appearance Learning. In 2014 IEEE Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/CVPR.2014.159>
- Baker, S. and Matthews, I. 2004. Lucas-Kanade 20 years on: A unifying framework. International Journal of Computer Vision. <https://doi.org/10.1023/B:VISI.0000011205.11775.fd>
- Barnich, O. and Van Droogenbroeck, M. 2011. ViBe: A universal background subtraction algorithm for video sequences. IEEE Transactions on Image Processing. <https://doi.org/10.1109/TIP.2010.2101613>
- Baskurt, K.B. and Samet, R. 2017a. Improved adaptive background subtraction method using pixel-based segmenter. Retrieved from <https://dspace5.zcu.cz/handle/11025/29610>
- Baskurt, K.B. and Samet, R. 2017b. Robust object tracking using alternative correlation filters. In 2017 25th Signal Processing and Communications Applications Conference (SIU) (pp. 1–4). IEEE. <https://doi.org/10.1109/SIU.2017.7960179>
- Baskurt, K.B. and Samet, R. 2018. Long-term multiobject tracking using alternative correlation filters. Turkish Journal of Electrical Engineering & Computer Sciences 26.5.
- Black, J., Ellis, T. and Rosin, P. 2002. Multi view image surveillance and tracking. In Proceedings - Workshop on Motion and Video Computing, MOTION 2002. <https://doi.org/10.1109/MOTION.2002.1182230>
- Bolme, D.S., Beveridge, J.R., Draper, B.A. and Lui, Y.M. 2010. Visual object tracking using adaptive correlation filters. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/CVPR.2010.5539960>
- Boykov, Y., Veksler, O. and Zabih, R. 2001. Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence. <https://doi.org/10.1109/34.969114>

- Bracewell, R.N. 1986. The Fourier Transform and its Applications. McGraw-Hill. <https://doi.org/10.1049/ep.1965.0268>
- Brown, and Lowe. 2003. Recognising panoramas. In Proceedings Ninth IEEE International Conference on Computer Vision. <https://doi.org/10.1109/ICCV.2003.1238630>
- Chakraborty, S., Tickoo, O. and Iyer, R. 2015. Adaptive keyframe selection for video summarization. In Proceedings - 2015 IEEE Winter Conference on Applications of Computer Vision, WACV 2015. <https://doi.org/10.1109/WACV.2015.99>
- Chen, Y.T., Chen, C.S., Huang, C.R. and Hung, Y. P. 2007. Efficient hierarchical method for background subtraction. Pattern Recognition. <https://doi.org/10.1016/j.patcog.2006.11.023>
- Choeychuen, K., Kumhom, P. and Chamnongthai, K. 2006. An Efficient Implementation of the Nearest Neighbor Based Visual Objects Tracking. In 2006 International Symposium on Intelligent Signal Processing and Communications. <https://doi.org/10.1109/ISPACS.2006.364723>
- Chou, C.L., Lin, C.H., Chiang, T.H., Chen, H.T. and Lee, S.Y. 2015. Coherent event-based surveillance video synopsis using trajectory clustering. In 2015 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2015. <https://doi.org/10.1109/ICMEW.2015.7169855>
- Cortes, C. and Vapnik, V. 1995. Support-vector networks. Machine Learning, 20(3), 273–297. <https://doi.org/10.1007/BF00994018>
- Dalal, N. and Triggs, B. 2005. Histograms of oriented gradients for human detection. In Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005. <https://doi.org/10.1109/CVPR.2005.177>
- Danelljan, M., Hager, G., Khan, F.S. and Felsberg, M. 2015. Learning spatially regularized correlation filters for visual tracking. In Proceedings of the IEEE International Conference on Computer Vision. <https://doi.org/10.1109/ICCV.2015.490>
- Danelljan, M., Häger, G., Shahbaz Khan, F., and Felsberg, M. (2014). Accurate Scale Estimation for Robust Visual Tracking. In Proceedings of the British Machine Vision Conference 2014. <https://doi.org/10.5244/C.28.65>
- Dollar, P., Appel, R., Belongie, S. and Perona, P. 2014. Fast feature pyramids for object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. <https://doi.org/10.1109/TPAMI.2014.2300479>
- Farenzena, M., Bazzani, L., Perina, A., Murino, V. and Cristani, M. 2010. Person re-identification by symmetry-driven accumulation of local features. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/CVPR.2010.5539926>

- Feng, S., Lei, Z., Yi, D. and Li, S.Z. 2012. Online content-aware video condensation. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/CVPR.2012.6247913>
- Feng, S., Liao, S., Yuan, Z. and Li, S. Z. 2010. Online principal background selection for video synopsis. In Proceedings - International Conference on Pattern Recognition. <https://doi.org/10.1109/ICPR.2010.13>
- Finkel, R.A. and Bentley, J.L. 1974. Quad trees a data structure for retrieval on composite keys. *Acta Informatica*. <https://doi.org/10.1007/BF00288933>
- Fu, W., Wang, J., Gui, L., Lu, H. and Ma, S. 2014. Online video synopsis of structured motion. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2013.12.041>
- Girshick, R. 2015. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision. <https://doi.org/10.1109/ICCV.2015.169>
- Godec, M., Roth, P. M. and Bischof, H. 2013. Hough-based tracking of non-rigid objects. *Computer Vision and Image Understanding*. <https://doi.org/10.1016/j.cviu.2012.11.005>
- Goyette, N., Jodoin, P.M., Porikli, F., Konrad, J. and Ishwar, P. 2012. changedetection.net: A new change detection benchmark dataset. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. <https://doi.org/10.1109/CVPRW.2012.6238919>
- Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M. M., Hicks, S. L. and Torr, P. H. S. 2016. Struck: Structured Output Tracking with Kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2015.2509974>
- He, Y., Gao, C., Sang, N., Qu, Z. and Han, J. 2017a. Graph coloring based surveillance video synopsis. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2016.11.011>
- He, Y., Qu, Z., Gao, C. and Sang, N. 2017b. Fast Online Video Synopsis Based on Potential Collision Graph. *IEEE Signal Processing Letters*. <https://doi.org/10.1109/LSP.2016.2633374>
- Heikkilä, M. and Pietikäinen, M. 2006. A texture-based method for modeling the background and detecting moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2006.68>
- Henriques, J.F., Caseiro, R., Martins, P. and Batista, J. 2012. Exploiting the circulant structure of tracking-by-detection with kernels. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-642-33765-9_50
- Hofmann, M., Tiefenbacher, P. and Rigoll, G. 2012. Background segmentation with feedback: The pixel-based adaptive segmenter. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*.

<https://doi.org/10.1109/CVPRW.2012.6238925>

- Hong, Z., Chen, Z., Wang, C., Mei, X., Prokhorov, D. and Tao, D. 2015. MUlti-Store Tracker (MUSTer): A cognitive psychology inspired approach to object tracking. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/CVPR.2015.7298675>
- Hoshen, Y. and Peleg, S. 2015. Live video synopsis for multiple cameras. In Proceedings - International Conference on Image Processing, ICIP. <https://doi.org/10.1109/ICIP.2015.7350790>
- Hsia, C.H., Tsung-Cheng Wu, Chiang, J.S. and Hsieh, C.F. 2011. VLSI architecture design of moving objects detection using adaptive least-mean-square scheme. In 2011 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS) (pp. 1–6). IEEE. <https://doi.org/10.1109/ISPACS.2011.6146193>
- Hsia, C.H., Chiang, J.S. and Hsieh, C.F. 2016. Low-complexity range tree for video synopsis system. Multimedia Tools and Applications. <https://doi.org/10.1007/s11042-015-2714-2>
- Huang, C.R., Chung, P.C.J., Yang, D.K., Chen, H.C. and Huang, G.J. 2014. Maximum *a Posteriori* Probability Estimation for Online Surveillance Video Synopsis. IEEE Transactions on Circuits and Systems for Video Technology. <https://doi.org/10.1109/TCSVT.2014.2308603>
- Huang, T. and Russell, S. 1997. Object identification in a Bayesian context. In IJCAI International Joint Conference on Artificial Intelligence.
- Irani, M., Anandan, P., Bergen, J., Kumar, R. and Hsu, S. 1996. Efficient representations of video sequences and their applications. Signal Processing: Image Communication. [https://doi.org/10.1016/0923-5965\(95\)00055-0](https://doi.org/10.1016/0923-5965(95)00055-0)
- Javed, Rasheed, Shafique. and Shah. 2003. Tracking across multiple cameras with disjoint views. In Proceedings Ninth IEEE International Conference on Computer Vision. <https://doi.org/10.1109/ICCV.2003.1238451>
- Jin, J., Liu, F., Gan, Z. and Cui, Z. 2016. Online video synopsis method through simple tube projection strategy. In 2016 8th International Conference on Wireless Communications and Signal Processing, WCSP 2016. <https://doi.org/10.1109/WCSP.2016.7752708>
- KaewTraKulPong, P. and Bowden, R. 2002. An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection. In Video-Based Surveillance Systems. https://doi.org/10.1007/978-1-4615-0913-4_11
- Kalal, Z., Mikolajczyk, K. and Matas, J. 2012. Tracking-learning-detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. <https://doi.org/10.1109/TPAMI.2011.239>

- Kalman, R.E. 1960. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*. <https://doi.org/10.1115/1.3662552>
- Kanhere, N.K. and Birchfield, S.T. 2010. A taxonomy and analysis of camera calibration methods for traffic monitoring applications. *IEEE Transactions on Intelligent Transportation Systems*. <https://doi.org/10.1109/TITS.2010.2045500>
- Kasamwattanakrote, S., Cooharajanane, N., Satoh, S. and Lipikorn, R. 2010. Real Time Tunnel Based Video Summarization Using Direct Shift Collision Detection. In *ADVANCES IN MULTIMEDIA INFORMATION PROCESSING-PCM 2010*, PT I.
- Kim, K., Chalidabhongse, T.H., Harwood, D. and Davis, L. 2005. Real-time foreground-background segmentation using codebook model. *Real-Time Imaging*. <https://doi.org/10.1016/j.rti.2004.12.004>
- Kim, K. and Davis, L.S. 2006. Multi-camera Tracking and Segmentation of Occluded People on Ground Plane Using Search-Guided Particle Filtering. *ECCV*. <https://doi.org/10.1007/11744078>
- Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P. 1983. Optimization by simulated annealing. *Science* (New York, N.Y.), 220(4598), 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Kolmogorov, V., and Zabih, R. (2004). What Energy Functions Can Be Minimized via Graph Cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2004.1262177>
- Kristan, M., Matas, J., Leonardis, A., Felsberg, M., Cehovin, L., Fernandez, G. ... Pflugfelder, R. 2015. The Visual Object Tracking VOT2015 Challenge Results. Retrieved from https://www.cv-foundation.org/openaccess/content_iccv_2015_workshops/w14/html/Kristan_The_Visual_Object_ICCV_2015_paper.html
- Li, X., Wang, Z. and Lu, X. 2016. Surveillance video synopsis via scaling down objects. *IEEE Transactions on Image Processing*. <https://doi.org/10.1109/TIP.2015.2507942>
- Li, Y. and Zhu, J. 2015. A Scale Adaptive Kernel Correlation Filter Tracker with Feature Integration (pp. 254–265). Springer, Cham. https://doi.org/10.1007/978-3-319-16181-5_18
- Liao, S., Zhao, G., Kellokumpu, V., Pietikäinen, M. and Li, S.Z. 2010. Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. <https://doi.org/10.1109/CVPR.2010.5539817>
- Liao, W. and Wang, S. 2017. Compressed-domain Video Synopsis via 3D Graph Cut and Blank Frame Deletion. *MMW*. <https://doi.org/10.1145/3126686.3126778>

- Lin, L., Lin, W., Xiao, W. and Huang, S. 2017. An optimized video synopsis algorithm and its distributed processing model. *Soft Computing*. <https://doi.org/10.1007/s00500-015-1823-1>
- Lin, W., Zhang, Y., Lu, J., Zhou, B., Wang, J. and Zhou, Y. 2015. Summarizing surveillance videos with local-patch-learning-based abnormality detection, blob sequence optimization, and type-based synopsis. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2014.12.044>
- Lu, M., Wang, Y. and Pan, G. 2013. Generating fluent tubes in video synopsis. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*. <https://doi.org/10.1109/ICASSP.2013.6638063>
- Mahapatra, A., Mishra, T.K., Sa, P.K. and Majhi, B. 2014. Human recognition system for outdoor videos using Hidden Markov model. *AEU - International Journal of Electronics and Communications*. <https://doi.org/10.1016/j.aeue.2013.08.011>
- Mahapatra, A., Sa, P.K., Majhi, B. and Padhy, S. 2016. MVS: A multi-view video synopsis framework. *Signal Processing: Image Communication*. <https://doi.org/10.1016/j.image.2016.01.002>
- Mazzon, R. and Cavallaro, A. 2013. Multi-camera tracking using a Multi-Goal Social Force Model. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2011.09.038>
- Mitchell, M. 1996. An introduction to genetic algorithms. *Computers and Mathematics with Applications*. [https://doi.org/10.1016/S0898-1221\(96\)90227-8](https://doi.org/10.1016/S0898-1221(96)90227-8)
- Munaro, M., Basso, F. and Menegatti, E. 2016. OpenPTrack: Open source multi-camera calibration and people tracking for RGB-D camera networks. *Robotics and Autonomous Systems*. <https://doi.org/10.1016/j.robot.2015.10.004>
- Nie, Y., Xiao, C., Sun, H. and Li, P. 2013. Compact video synopsis via global spatiotemporal optimization. *IEEE Transactions on Visualization and Computer Graphics*. <https://doi.org/10.1109/TVCG.2012.176>
- Papenberg, N., Bruhn, A., Brox, T., Didas, S. and Weickert, J. 2006. Highly accurate optic flow computation with theoretically justified warping. In *International Journal of Computer Vision*. <https://doi.org/10.1007/s11263-005-3960-y>
- Pasula, H., Russell, S., Ostland, M. and Ritov, Y. 1999. Tracking many objects with many sensors. In *IJCAI International Joint Conference on Artificial Intelligence*. <https://doi.org/10.1.1.34.7997>
- Pérez, P., Gangnet, M. and Blake, A. 2003. Poisson image editing. *ACM Transactions on Graphics*. <https://doi.org/10.1145/882262.882269>
- Pérez, P., Hue, C., Vermaak, J. and Gangnet, M. 2002. Color-Based Probabilistic Tracking (pp. 661–675). Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-47969-4_44

- Press, W., Teukolsky, S., Vetterling, W., Flannery, B., Ziegler, E., Press, W. ... Vetterling, W. 1992. Numerical Recipes in C. Cambridge University Press. <https://doi.org/10.2307/1269484>
- Previtali, F., Bloisi, D.D. and Iocchi, L. 2017. A distributed approach for real-time multi-camera multiple object tracking. *Machine Vision and Applications*, 28(3–4), 421–430. <https://doi.org/10.1007/s00138-017-0827-5>
- Pritch, Y., Ratovitch, S., Hendel, A. and Peleg, S. 2009. Clustered synopsis of surveillance video. In 6th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009. <https://doi.org/10.1109/AVSS.2009.53>
- Pritch, Y., Rav-Acha, A., Gutman, A. and Peleg, S. 2007. Webcam synopsis: Peeking around the world. In Proceedings of the IEEE International Conference on Computer Vision. <https://doi.org/10.1109/ICCV.2007.4408934>
- Pritch, Y., Rav-Acha, A. and Peleg, S. 2008. Nonchronological video synopsis and indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/TPAMI.2008.29>
- Rakotomamonjy, A., Bach, F., Canu, S. and Grandvalet, Y. 2008. SimpleMKL. *Journal of Machine Learning Research*.
- Rav-Acha, A., Pritch, Y., Pattern, S. P.-C. V. and, and 2006, undefined. (n.d.). Making a long video short: Dynamic video synopsis. *Ieeexplore.Ieee.Org*. Retrieved from <https://ieeexplore.ieee.org/abstract/document/1640790/>
- Rodriguez, M. 2010. CRAM: Compact representation of actions in movies. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. <https://doi.org/10.1109/CVPR.2010.5540030>
- Rodriguez, M.D., Ahmed, J. and Shah, M. 2008. Action MACH: A spatio-temporal maximum average correlation height filter for action recognition. In 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR. <https://doi.org/10.1109/CVPR.2008.4587727>
- Rui Zhong, Ruimin Hu, Zhongyuan Wang, and Shizheng Wang. 2014. Fast Synopsis for Moving Objects Using Compressed Video. *IEEE Signal Processing Letters*. <https://doi.org/10.1109/LSP.2014.2317754>
- Schoepflin, T.N. and Dailey, D.J. 2007. Algorithms for calibrating roadside traffic cameras and estimating mean vehicle speed. In IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. <https://doi.org/10.1109/ITSC.2007.4357806>
- Schwartz, W.R. and Davis, L.S. 2009. Learning discriminative appearance-based models using partial least squares. In Proceedings of SIBGRAPI 2009 - 22nd Brazilian Symposium on Computer Graphics and Image Processing. <https://doi.org/10.1109/SIBGRAPI.2009.42>

- Smith, M. A. and Kanade, T. 1998. Video skimming and characterization through the combination of image and language understanding. In Proceedings - 1998 IEEE International Workshop on Content-Based Access of Image and Video Database, CAIVD 1998. <https://doi.org/10.1109/CAIVD.1998.646034>
- Stauffer, C. 2003. Estimating Tracking Sources and Sinks. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. <https://doi.org/10.1109/CVPRW.2003.10036>
- Stauffer, C. and Grimson, W. 2000. Learning Patterns of Activity Using Real-Time Tracking. *Ieee Transactions on Pattern Analysis and Machine Intelligence*. <https://doi.org/10.1109/34.868677>
- Stauffer, C. and Grimson, W.E.L. 1999. Adaptive background mixture models for real-time tracking. Proceedings 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Cat No PR00149. <https://doi.org/10.1109/CVPR.1999.784637>
- Straw, A.D., Branson, K., Neumann, T.R. and Dickinson, M.H. 2011. Multi-camera real-time three-dimensional tracking of multiple flying animals. *Journal of the Royal Society Interface*. <https://doi.org/10.1098/rsif.2010.0230>
- Sun, J., Zhang, W., Tang, X. and Shum, H.Y. 2006. Background cut. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/11744047_48
- Taj, M., Maggio, E. and Cavallaro, A. 2006. Multi-feature Graph-Based Object Tracking. In *Multimodal Technologies for Perception of Humans* (pp. 190–199). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-69568-4_15
- Tao Yang, Li, S.Z., Quan Pan, and Jing. Li. 2005. Real-Time Multiple Objects Tracking with Occlusion Handling in Dynamic Scenes. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). <https://doi.org/10.1109/CVPR.2005.292>
- Tian, Y., Zheng, H., Chen, Q., Wang, D. and Lin, R. 2016. Surveillance video synopsis generation method via keeping important relationship among objects. *IET Computer Vision*, 10(8), 868–872. <https://doi.org/10.1049/iet-cvi.2016.0128>
- Tong Yao, Maosen Xiao, Caiwen Ma, Chao Shen, and Peng Li. 2014. Object based video synopsis. In 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA) (pp. 1138–1141). IEEE. <https://doi.org/10.1109/WARTIA.2014.6976479>
- Truong, B.T. and Venkatesh, S. 2007. Video Abstraction: A Systematic Review and Classificatio. *ACM Transactions on Multimedia Computing, Communications, and Applications*. <https://doi.org/10.1145/1198302.1198305>
- Vlachos, M., Kollios, G. and Gunopulos, D. 2002. Discovering similar

- multidimensional trajectories. Proceedings - International Conference on Data Engineering. <https://doi.org/10.1109/ICDE.2002.994784>
- Vojir, T., Noskova, J. and Matas, J. 2014. Robust scale-adaptive mean-shift for tracking. Pattern Recognition Letters. <https://doi.org/10.1016/j.patrec.2014.03.025>
- Vural, U. and Akgul, Y.S. 2009. Eye-gaze based real-time surveillance video synopsis. Pattern Recognition Letters. <https://doi.org/10.1016/j.patrec.2009.03.002>
- Wang, S., Xu, W., Wang, C. and Wang, B. 2013a. A Framework for Surveillance Video Fast Browsing Based on Object Flags. In The Era of Interactive Media (pp. 411–421). New York, NY: Springer New York. https://doi.org/10.1007/978-1-4614-3501-3_34
- Wang, S.Z., Wang, Z.Y. and Hu, R.M. 2013b. Surveillance video synopsis in the compressed domain for fast video browsing. Journal of Visual Communication and Image Representation. <https://doi.org/10.1016/j.jvcir.2013.10.001>
- Wang, X., Doretto, G., Sebastian, T., Rittscher, J. and Tu, P. 2007. Shape and appearance context modeling. In Proceedings of the IEEE International Conference on Computer Vision. <https://doi.org/10.1109/ICCV.2007.4409019>
- Wang, Y., Jodoin, P.M., Porikli, F., Konrad, J., Benezeth, Y. and Ishwar, P. 2014. CDnet 2014: An Expanded Change Detection Benchmark Dataset. Retrieved from https://www.cv-foundation.org/openaccess/content_cvpr_workshops_2014/W12/html/Wang_CDnet_2014_An_2014_CVPR_paper.html
- Welch, G. and Bishop, G. 2001. An Introduction to the Kalman Filter. In SIGGRAPH 2001, Los Angeles, CA, August 12-17. <https://doi.org/10.1.1.117.6808>
- Whitley, D. 1994. A genetic algorithm tutorial. Statistics and Computing. <https://doi.org/10.1007/BF00175354>
- Wu, Y., Lim, J. and Yang, M.H. 2015. Object tracking benchmark. IEEE Transactions on Pattern Analysis and Machine Intelligence. <https://doi.org/10.1109/TPAMI.2014.2388226>
- Xu, L., Liu, H., Yan, X., Liao, S. and Zhang, X. 2015. Optimization method for trajectory combination in surveillance video synopsis based on genetic algorithm. Journal of Ambient Intelligence and Humanized Computing. <https://doi.org/10.1007/s12652-015-0278-7>
- Xu, M., Xu, M., Li, S. Z., Li, S. Z., Li, B., Li, B., ... Xiang, S. 2008. A Set Theoretical Method for Video Synopsis. Science And Technology. <https://doi.org/10.1145/1460096.1460156>
- Yildiz, A., Ozgur, A. and Akgul, Y. S. 2008. Fast non-linear video synopsis. In 2008 23rd International Symposium on Computer and Information Sciences (pp. 1–6). IEEE. <https://doi.org/10.1109/ISCIS.2008.4717951>

- Zhang, P., Wang, L., Huang, W., Xie, L. and Chen, G. 2014. Multiple pedestrian tracking based on couple-states Markov chain with semantic topic learning for video surveillance. *Soft Computing*. <https://doi.org/10.1007/s00500-014-1375-9>
- Zhu, J., Liao, S. and Li, S.Z. 2016a. Multicamera Joint Video Synopsis. *IEEE Transactions on Circuits and Systems for Video Technology*. <https://doi.org/10.1109/TCSVT.2015.2430692>
- Zhu, X., Liu, J., Wang, J. and Lu, H. 2014. Key observation selection-based effective video synopsis for camera network. *Machine Vision and Applications*. <https://doi.org/10.1007/s00138-013-0519-8>
- Zhu, X., Loy, C.C. and Gong, S. 2013. Video synopsis by heterogeneous multi-source correlation. In *Proceedings of the IEEE International Conference on Computer Vision*. <https://doi.org/10.1109/ICCV.2013.17>
- Zhu, X., Loy, C.C. and Gong, S. 2016b. Learning from Multiple Sources for Video Summarisation. *International Journal of Computer Vision*. <https://doi.org/10.1007/s11263-015-0864-3>
- Zivkovic, Z. 2004. Improved adaptive Gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition*. <https://doi.org/10.1109/ICPR.2004.1333992>

EKLER

EK 1 Video sinopsiste kullanılan yöntemler.....	127
EK 2 Korelasyon filtresinin elde edilmesi	133

EK 1 Video Sinopsiste Kullanılan Yöntemler

Çizelge 1 Video sinopsiste kullanılan yöntemler

Çalışma	Nesne algılama	Nesne takibi	Optimizasyon	Arka plan oluşturma	Harmanlama
Rav-Acha vd. (2006)	Pixel difference with temporal median (Rav-Acha vd. 2006)	NA	Simulated annealing (Kirkpatrick vd. 1983)	Temporal median (Rav-Acha vd. 2006)	Stitching (Irani vd. 1996)
Pritch vd. (2007)	Background cut (Sun vd. 2006)	NA	Simulated annealing (Kirkpatrick vd. 1983)	Temporal median (Pritch vd. 2007)	Poisson image editing (Patrick Pérez vd. 2003)
Xu vd. (2008)	GMM (Stauffer & Grimson, 2000)	Multiple object tracking (Yang vd. 2005)	Mean shift (M. Xu vd. 2008)	NA	NA
Yildiz vd. (2008)*	Gradient magnitude (Yildiz vd. 2008)	NA	Dynamic programming (Yildiz vd. 2008)	NA	NA
Pritch vd. (2008)	Background cut (Sun vd. 2006)	Tube generation (Pritch vd. 2008)	Greedy optimization (Pritch vd. 2008)	Temporal median (Pritch vd. 2008)	Poisson image editing (Patrick Pérez vd. 2003)
Pritch vd. (2009)	Background cut (Sun vd. 2006)	Tube generation (Pritch vd. 2008)	Packing cost (Pritch vd. 2009)	NA	NA

Çizelge 1 Video sinopsiste kullanılan yöntemler (devam)

Vural ve Akgul (2009)*	Gradient magnitude (Vural & Akgul, 2009)	NA	Dynamic programming (Vural & Akgul, 2009)	NA	NA
Rodriguez (2010)	Optical flow (Papenberg vd. 2006)	Clifford worm extraction (M. Rodriguez, 2010)	Simulated annealing (Kirkpatrick vd. 1983)	NA	NA
Feng vd. (2010)*	Background cut (Sun vd. 2006)	Multiple object tracking (Yang vd. 2005)	Simulated annealing (Kirkpatrick vd. 1983)	OPBS (Feng vd. 2010)	Poisson image editing (Patrick Pérez vd. 2003)
Kasamwattananarote vd. (2010)	HOG for human detection (Dalal & Triggs, 2005)	Euclidean distance (Kasamwattananarote vd. 2010)	Film map generation (Kasamwattananarote vd. 2010)	Manually created (Kasamwattananarote vd. 2010)	JIT renderer (Kasamwattananarote vd. 2010)
Feng vd. (2012)*	Background cut (Sun vd. 2006)	Sticky tracking (Feng vd. 2012)	Online tube filling (Greedy & Roulette wheel selection (Mitchell, 1996))	OPBS (Feng vd. 2010)	Poisson image editing (Patrick Pérez vd. 2003)
Nie vd. (2013)	Pixel difference with temporal median (Nie vd. 2013)	Tracking with shadow detection (KaewTraKul Pong & Bowden, 2002)	Alpha-beta swap graph-cut (Boykov vd. 2001)	Multilevel patch relocation (Nie vd. 2013)	Poisson image editing (Patrick Pérez vd. 2003)
Wang vd. (2013b)	Quadtree (Finkel & Bentley, 1974)	NA	Simulated annealing (Kirkpatrick vd. 1983)	OPBS (Feng vd. 2010)	NA

Çizelge 1 Video sinopsiste kullanılan yöntemler (devam)

Lu vd. (2013)	Texture GMM (Improved GMM (Zivkovic, 2004) and LBP (Heikkilä & Pietikäinen , 2006))	Particle filter (P. Pérez vd. 2002)	NA	NA	NA
Wang vd. (2013a)	Motion vector based LBP (Wang vd. 2013a)	Sticky tracking (Feng 2012)	Simulated annealing (Kirkpatrick vd. 1983)	OPBS (Feng vd. 2010)	NA
Zhong vd. (2014)	3D graph- cut using GMM (Zhong vd. 2014)	3D graph-cut (Zhong vd. 2014)	Simulated annealing (Kirkpatrick vd. 1983)	NA	NA
Zhu vd. (2014)**	Backgroun d cut (Sun vd. 2006)	Multi-feature graph-based object tracking (Taj vd. 2006)	Simulated annealing (Kirkpatrick vd. 1983)	Temporal median (Xiaobin Zhu vd. 2014)	Poisson image editing (Patrick Pérez vd. 2003)
Huang vd. (2014)*	Hierarchica l backgroun d modeling (Chen vd. 2007)	Using proposed synopsis table (C.-R. Huang vd. 2014)	Synopsis table (C.-R. Huang vd. 2014)	NA	NA
Yao vd. (2014)	Frame difference (Yao vd. 2014)	Kalman filter (Kalman, 1960)	Genetic algorithm (Whitley, 1994)	Frame averaging (Yao vd. 2014)	NA

Çizelge 1 Video sinopsiste kullanılan yöntemler (devam)

Fu vd. (2014)*	GMM (Stauffer & Grimson, 2000) and Min-cut (Kolmogorov & Zabih, 2004))	Modified sticky tracking (Fu vd. 2014)	Online tube filling (Greedy and Roulette wheel selection (Mitchell, 1996))	NA	NA
Chou vd. (2015)	GMM (Stauffer & Grimson, 2000)	Tracking (Chou vd. 2015)	Trajectory clustering (Chou vd. 2015)	NA	NA
Hoshen ve Peleg (2015)* **	Adaptive background modeling (Stauffer & Grimson, 2000) ve HOG for human detection (Dalal & Triggs, 2005)	Tube generation (Pritch vd. 2008)	Fast greedy approach (Hoshen & Peleg, 2015)	NA	NA
Xu vd. (2015)	NA	NA	Genetic algorithm (Whitley, 1994)	NA	NA
Lin vd. (2015)	Abnormal activity detection (W. Lin vd. 2015)	Blob sequence optimization (W. Lin vd. 2015)	Abnormality-type-based video synopsis (W. Lin vd. 2015)	NA	NA
Li vd. (2016)	GMM (Stauffer & Grimson, 2000)	Kalman filter (Kalman, 1960)	Simulated annealing (Kirkpatrick vd. 1983)	Temporal median (X. Li vd. 2016)	NA

Çizelge 1 Video sinopsiste kullanılan yöntemler (devam)

Hsia vd. (2016)	GMM (Stauffer & Grimson, 2000)	LMS tracking (C.-H. Hsia vd. 2011)	Low-complexity range tree (W. Lin vd. 2015)	NA	NA
Zhu vd. (2016a)**	SILTP (S. Liao, Zhao, Kellokumpu, Pietikäinen, & Li, 2010)	Sticky tracking (Feng 2012)	Simple greedy approach (Zhu vd. 2016a)	Frame averaging (Zhu vd. 2016a)	Poisson image editing (Patrick Pérez vd. 2003)
Mahapatra vd. (2016)**	Human detection (Mahapatra, Mishra, Sa, & Majhi, 2014)	Clustered track extraction (Mahapatra vd. 2016)	Table-driven approach, contradictory binary graph coloring (Mahapatra vd. 2016) and simulated annealing (Kirkpatrick vd. 1983)	NA	NA
Jin vd. (2016)*	R-CNN (Girshick, 2015)	Chi-square distance (Jin vd. 2016)	Simple tube generation (Jin vd. 2016)	Frame averaging (Jin vd. 2016)	Poisson image editing (Patrick Pérez vd. 2003)
Tian vd. (2016)	GMM (Stauffer & Grimson, 2000)	Kalman filter (Kalman, 1960)	Genetic algorithm (Whitley, 1994)	NA	NA

Çizelge 1 Video sinopsiste kullanılan yöntemler (devam)

Li vd. (2016)	Temporal median & Aggregated channel features detection (Dollar, Appel, Belongie, & Perona, 2014)	Kalman filter (Kalman, 1960)	Simulated annealing (Kirkpatrick vd. 1983)	Temporal median (X. Li vd. 2016)	Poisson image editing (Patrick Pérez vd. 2003)
Lin vd. (2017)	GMM (Stauffer & Grimson, 2000)	Multiple pedestrian tracking (Zhang vd. 2014)	Huiyan (L. Lin vd. 2017)	Temporal median (L. Lin vd. 2017)	Color median for image seam (L. Lin vd. 2017)
He vd. (2017a)	ViBe (Barnich & Van Droogenbroeck, 2011)	NN-based tracking (Choeychuen vd. 2006)	Graph coloring approach (He vd. 2017a)	Frame averaging (He vd. 2017a)	Poisson image editing (Patrick Pérez vd. 2003)
He vd. (2017b) *	ViBe (Barnich & Van Droogenbroeck, 2011)	NN-based tracking (Choeychuen vd. 2006)	Potential collision graph (He vd. 2017b)	Frame averaging (He vd. 2017b)	Poisson image editing (Patrick Pérez vd. 2003)
Liao ve Wang (2017)	Motion vectors (W. Liao & Wang, 2017)	VBF Deletion on 3D Graph (Liao ve Wang 2017)	Simulated annealing (Kirkpatrick vd. 1983)	PBS (Feng vd. 2012)	NA

* çevrimiçi yaklaşımları temsil etmektedir.

** çok-kamerallı yaklaşımları temsil etmektedir.

NA ilgili çalışmada belirtilmeyen yöntemleri temsil etmektedir.

EK 2 Korelasyon Filtresinin Elde Edilmesi

Eğitim girdilerini (F_i) beklenen çıktılara (G_i) ilintileyen filtreyi bulmak için güncel çıktı ($F_i \odot H^*$) ile beklenen çıktısı (G_i) arasındaki hatanın kareler toplamını minimuma indirgeyen korelasyon filtresi (H) aşağıdaki gibi hesaplanmaktadır (Denklem 1).

$$H = \min_H \sum_i |F_i \odot H^* - G_i|^2 \quad (1)$$

Denklem 1'deki minimizasyon problemi, korelasyon filtresi H 'nin her bir elemanı için ayrı çözülmektedir. Fourier düzlemde tüm işlemler eleman bazlı yapıldığı için bu dönüşüm kolayca gerçekleştirilmektedir. Bu durumda denklem 1, H 'nin elemanları ω ve ν ile gösterilecek şekilde denklem 2'deki gibi tekrar yazılmaktadır.

$$H_{\omega\nu} = \min_{H_{\omega\nu}} \sum_{i\omega\nu} |F_{i\omega\nu} H_{\omega\nu}^* - G_{i\omega\nu}|^2 \quad (2)$$

H bağımsız bir değişken olarak tutulmak üzere denklem 2, $H_{\omega\nu}$ ve $H_{\omega\nu}^*$ dikkate alınarak tekrar yazıldığında denklem 3 elde edilmektedir. Bu denklem sistemi adım adım çözüldüğünde (Denklem 4 ve 5) denklem 6 elde edilmektedir.

$$0 = \frac{\partial}{\partial H_{\omega\nu}^*} \sum_{i\omega\nu} |F_{i\omega\nu} H_{\omega\nu}^* - G_{i\omega\nu}|^2 \quad (3)$$

$$0 = \frac{\partial}{\partial H_{\omega\nu}^*} \sum_i (F_{i\omega\nu} H_{\omega\nu}^* - G_{i\omega\nu})(F_{i\omega\nu} H_{\omega\nu}^* - G_{i\omega\nu})^* \quad (4)$$

$$0 = \frac{\partial}{\partial H_{\omega\nu}^*} \sum_i [(F_{i\omega\nu} H_{\omega\nu}^*)(F_{i\omega\nu} H_{\omega\nu}^*)^* - (F_{i\omega\nu} H_{\omega\nu}^*) G_{i\omega\nu}^* - G_{i\omega\nu} (F_{i\omega\nu} H_{\omega\nu}^*)^* + G_{i\omega\nu} G_{i\omega\nu}^*] \quad (5)$$

$$0 = \frac{\partial}{\partial H_{\omega\nu}^*} \sum_i F_{i\omega\nu} F_{i\omega\nu}^* H_{\omega\nu} H_{\omega\nu}^* - F_{i\omega\nu} G_{i\omega\nu}^* H_{\omega\nu}^* - F_{i\omega\nu}^* G_{i\omega\nu} H_{\omega\nu} + G_{i\omega\nu} G_{i\omega\nu}^* \quad (6)$$

Denklem 6'nın türevi alındığında aşağıdaki hali almaktadır.

$$0 = \sum_i F_{i\omega\nu} F_{i\omega\nu}^* H_{\omega\nu} - F_{i\omega\nu} G_{i\omega\nu}^* \quad (7)$$

Bu noktada toplamları dağıtıp denklemi $H_{\omega\nu}$ için çözdüğümüzde denklem 8 elde edilmektedir.

$$H_{\omega\nu} = \frac{\sum_i F_{i\omega\nu} G_{i\omega\nu}^*}{\sum_i F_{i\omega\nu} F_{i\omega\nu}^*} \quad (8)$$

Sonuç olarak denklem eleman bazlı notasyondan başlangıçtaki girdi bazlı haline dönüştürüldüğünde aşağıdaki gibi olmaktadır:

$$H = \frac{\sum_i F_i \odot G_i^*}{\sum_i F_i \odot F_i^*}, \quad H^* = \frac{\sum_i G_i \odot F_i^*}{\sum_i F_i \odot F_i^*} \quad (9)$$

Denklem 9'de elde edilen son hali ile korelasyon filtresinin pay kısmı girdi ile beklenen çıktı arasındaki korelasyondan, payda ise girdinin enerji spektrumundan oluşmaktadır.

ÖZGEÇMİŞ

Adı Soyadı : Kemal Batuhan BAŞKURT
Doğum Yeri : Osmaniye
Doğum Tarihi : 23.07.1987
Medeni Hali : Evli
Yabancı Dili : İngilizce, İspanyolca

Eğitim Durumu (Kurum ve Yıl)

Lise : Osmaniye Anadolu Lisesi (2005)
Lisans : Mersin Üniversitesi Mühendislik Fakültesi
Bilgisayar Mühendisliği (2009)
Yüksek Lisans : Zaragoza Üniversitesi Mühendislik ve Mimarlık Fakültesi
Sistem ve Bilişim Mühendisliği (01/2010 - 09/2011)
Doktora : Ankara Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği
Anabilim Dalı (02/2014 – 02/2019)

Çalıştığı Kurum/Kurumlar ve Yıl

Nervia Consultores, 05/2010-09/2010
Infodif, 08/2012-03/2017
Ayonix, 04/2017-08/2017
Ekin Teknoloji, 11/2017-Devam ediyor

Yayınlar (SCI)

Baskurt, K.B. and Samet, R., 2018. "Long-term multi object tracking using alternative correlation filters." Turkish Journal of Electrical Engineering & Computer Sciences 26.5.

Baskurt, K.B. and Samet, R., 2019. "Video Synopsis: A Survey" Computer Vision and Image Understanding.

Hakemli Dergiler

Samet, R., Ghulam, S.S., **Baskurt, K.B.**, 2016. "An Efficient Pose Tolerant Face Recognition Approach." Transactions on Computational Science XXVI. Springer, Berlin, Heidelberg, 161-172.

Ulusal Kongre Sunum

Başkurt, K.B. ve Samet, R., 2017. "Robust object tracking using alternative correlation filters" Signal Processing and Communications Applications Conference (SIU), 2017 25th. IEEE.

Uluslararası Kongre Sunum

Baskurt, K.B. ve Samet, R., 2017. “Improved adaptive background subtraction method using pixel-based segmenter” in Computer Graphics, Visualization and Computer Vision (WSCG), 25th International Conference in Central Europe on. WSCG, 2017, pp. 41–46.

Baskurt, K.B. ve Samet, R., 2018. “Online video synopsis based on grid map representation” in COIA-2018: The 6th International Conference on Control and Optimization with Industrial Applications, Baku, Azerbaijan.

