

## SEMANTİK ARAMA İLE MULTIMEDYA VERİLERİN ANLAMLANDIRILMASI İÇİN ÖN İŞLEMLER

Oğuzhan MENEMENCİOĞLU<sup>1</sup> İlhami M. ORAK<sup>1</sup>

<sup>1</sup> Karabük Üniversitesi, Bilgisayar Mühendisliği Bölümü, 78050, Karabük, TÜRKİYE  
omenemencioglu@karabuk.edu.tr

**Özet-** Günümüzde çok büyük miktarda multimedia verisi üretilmiştir. Ortaya çıkan veri miktarı karşısında geleneksel ve manuel görüntü işleme yöntemleri yetersiz kalmaktadır. Buna karşın otomatik açıklama üretme yaklaşımları da ele alınması ve geliştirilmesi zor yöntemlerdir. Aynı zamanda otomatik açıklama üretmek sadece görüntü işleme basamağında kalmaktadır. Sonuçta da ortaya çıkan veriye erişecek, veriyi ilişkilendirecek ve anlamlandırarak sistemlere sahip olma anlamı taşımamaktadır. Çalışmanın odak noktası bu ana fikre dayanmaktadır. Çalışma, semantik arama ile multimedia verilerin anlamlandırılması amacıyla oluşturulacak ontolojilere kaynaklık edecek verilerin üretilmesi ve işlenmesi için yapılmıştır. Bilgisayarlı görü yaklaşımlarından arka plan çıkarma algoritması kullanılarak örnek videolar işlenerek sonuçlar XML dosyasına yazılmaktadır. Öngörü başarısını artırmak için mühendislik hesaplamaları ile hatalı durumlar elimine edilmektedir. Sonuç olarak üretilen verilerden semantik arama ile anlamlandırma yapılacak bir çerçeve üretilmesi hedeflenmektedir. Hali hazırdaki ve gelecekte üretilen verileri işleyecek bir sistem tasarlanmaktadır. Böylece multimedia veri setlerinin içerdiği bilgiye etkin olarak erişilebilecektir.

**Anahtar Kelimeler-** Semantik arama; multimedia; otomatik açıklama; multimedia erişimi; arka plan çıkarma algoritması.

## PREPROCESSES FOR INTERPRETATION OF MULTIMEDIA WITH SEMANTIC SEARCH

**Abstract-** Huge amount of multimedia data is produced in nowadays. Traditional and manual methods are incapable of processing the amount of multimedia data. After all, automatic annotation generation methods are hard to address and operate. Moreover, automatic annotation generation remains only computer vision level. It does not introduce a system to relate and produce meaning for the retrieved data. This paper's focal point is based on this main idea. The paper is prepared for producing and processing the data which is a source of ontologies that created with the aim of interpretation of multimedia with semantic search. Multimedia result which is processed by one of the computer vision approach of background subtraction algorithm, is recorded in XML files and is eliminated with engineering calculations. It is aimed to build a framework to introduce meaning on result data with semantic search. The system is designed such that current and future data is processed. In this manner, it will be possible to access the knowledge effectively on multimedia datasets.

**Key Words-** Semantic search; multimedia; annotation generation; multimedia retrieval; background subtraction algorithm.

## 1. GİRİŞ (INTRODUCTION)

Artan miktarda multimedya içerik günümüzde dijital formda üretilmektedir, işlenmektedir ve saklanmaktadır. Sadece Youtube verilerine göre, her ay 6 milyar saatin üzerinde video izlenmektedir; daha da önemlisi her dakikada 100 saatlik yeni video web'e yüklenmektedir [1]. Erişim için metadata olarak adlandırılan, genellikle kullanıcı tarafından elle oluşturulan metin tabanlı kaynak tanımlaması kullanılmaktadır. Metadata kaynağın farklı özelliklerini özetleyen genel bilgidir. Metadata arama motorlarının içeriğın aranan şeyle alakalı olup olmadığına karar vermesini sağlamaktadır. Açıklama (annotation) ise içeriği anlamayı ve yorumlamayı kolaylaştırmak için video akışındaki objeleri ve sahneleri (scene) işaretleme aracıdır [2].

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

## 2. YÖNTEM (METHOD)

Otomatik açıklama çıkarma zor, ilgi çekmeye devam eden ve oldukça güncel bir konudur. Videolar bilgisayarlı görü (computer vision) teknikleri ile işlenmektedir. Elde edilen veriler üzerinde yapılan hesaplama, eleme vb. gibi ara işlemler ile gürültüler giderilmiştir. Neticede veri seti ile elde edilen sonuçlar karşılaştırılmış ve doğruluk değerleri sunulmuştur.

### 2.1. Veri Temini (Data Supply)

Semantik arama kullanarak multimedya verilerinin anlamlandırılması sağlanmak istenmektedir. Çalışma kapsamında CAVIAR açık kıyaslama (public benchmark) veri setinin videoları işlenecektir. Bu yaklaşımla sonuçları bilinen ve sorgulanabilir veriler üzerinde işlem yaparak çalışmanın sonuçlarının doğruluğunu kıyaslamak amaçlanmaktadır. Veri setindeki her bir videonun çerçeveleri üzerinde objeler yakalanmaya çalışılmıştır.

### 2.2. Multimedya İşleme (Multimedia Processing)

Multimedya açıklama üretme işlemi manüel ve/veya otomatik yapılabilen bir işlemdir. En doğru sonuçlar henüz manüel işleme sonucu üretilebilmektedir. Fakat girişte bahsedilen veri miktarı karşısında manüel yöntemle multimedyaadan üretildiği hızlarda yararlanmak idealinden uzaklaşıldığı için bu yöntem kabul edilemez olarak değerlendirilebilir. Otomatik açıklama üretme işlemi için ise çeşitli yaklaşımlar mevcuttur [4].

Otomatik multimedya açıklama üretmek için video özetleme yaklaşımı, çekim (shot) tabanlı çerçeveler üzerinde hareket tabanlı yöntem kullanılmıştır [4]. Hareket eden nesnelere piksel farkı ve akış hesaplaması ile otomatik olarak tespit ve zaman içindeki hareketleri takip edilmiştir.

#### 2.2.1. Arka Plan Çıkarma Algoritması (Background Subtraction Algorithm)

Arka plan çıkarma algoritması bir video dizisindeki hareketli objeleri, örneğin bir yolda hareket eden arabalar gibi, bulmak için geliştirilmiş eski bir tekniktir. Fikir "Mevcut andaki resim

zamana göre ortalama arka plan resminden çıkarılırsa durağan olmayan objeler kalır.” şeklindedir. Mevcut andaki resmin her bir pikselinin sınıflandırma işlemi basit bir yaklaşım olduğu halde yavaş hareket eden objeler için başarısızdır ve gölge ile hareket eden objeleri ayıramamaktadır.

t Zamanında (x, y) pikseli için I(x, y, t) anlık piksel değeri olduğunda, arka plan resmi en basit formunda uzun vadede ortalama resim olarak Eş. 1 şeklinde tanımlanır.

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

Aynı zamanda aşamalı olarak Eş. 2 şeklinde de hesaplanabilir [5].

$$B(x, y, t) = \frac{(t-1)}{t} B(x, y, t-1) + \frac{1}{t} I(x, y, t) \quad (2)$$

### 2.2.2. Gauss Karışım Modeli (Gaussian Mixture Model)

Tek gauss modeli Wren ve arkadaşları tarafından sunulmuştur. Ancak piksel değerleri kompleks dağılımlara sahip olduğundan daha ayrıntılı modellere ihtiyaç duyulmuştur. Bu nedenle Friedman ve Russell tarafından arka plan çıkarma için Gauss karışım modeli önerilmiştir [6]. Çerçeve sayısı n olan ve her çerçevesindeki her bir pikselin bağımsız istatistikî işlem gördüğü ve gözlemlenen yoğunluğunun kaydedildiğini varsayalım. Bu durumda K Gauss karışımı ile t zamanda gözlenen  $x_t$  piksel yoğunluğunun olasılığı Eş. 3 ile modellenir.

$$\Pr(x_t) = \sum_{j=1}^K \frac{\omega_j}{(2\pi)^{\frac{d}{2}} |\Sigma_j|^{\frac{1}{2}}} e^{-\frac{1}{2}(x_t - \mu_j)^T \Sigma_j^{-1} (x_t - \mu_j)} \quad (3)$$

Burada  $\omega_j$  pikselin karışım modelinin j'inci dağılımına atanmış ağırlık,  $\mu_j$  ortalama ve  $\Sigma_j$  kovaryans matrisidir. Kolaylık için kovaryans matrisi yerine  $\Sigma_j = \sigma_j^2 I$  kullanılmaktadır. Arka plan modeli belirlenirken her bir piksel tek tek ele alınır. K dağılımları büyük olan arka planların daha istikrarlı olacağı düşüncesiyle  $\omega_j/\sigma_j^2$  temelinde sıralanır. Daha sonra T arka plan sürecinden elde edilmiş sürülen izin kesrinin tahmini olduğunda Eş. 4'deki ilk B dağılımı seçilir.

$$B = \arg \min_b \left( \frac{\sum_{j=1}^b \omega_j}{\sum_{j=1}^K \omega_j} > T \right) \quad (4)$$

Modeli güncellemek için her piksel değeri,  $x_t$ , bir eşleşme buluncaya kadar var olan dağılımlara karşı kontrol edilir. Eşleşme bir dağılımın 2 standart sapmasının içinde bir piksel olarak tanımlanmıştır. Eğer gelen değer ile hiçbir modelin dağılımı eşleşmezse, en az olasılığa sahip dağılım çıkarılır ve başlangıç olarak yüksek varyans ve düşük öncelik ağırlıklı ortalama değere sahip bir dağılım ile yer değiştirilir.

$\alpha$  öğrenme değeri,  $M_{j,t}$  ise eşleşen dağılım için 1 ve kalan dağılımlar için 0 olduğunda; t zamandaki her bir dağılım için dağılım ağırlığı Eş. 5 ile ayarlanır.

$$\omega_{j,t} = (1 - \alpha)\omega_{j,t-1} + \alpha(M_{j,t}) \quad (5)$$

Dağılımın parametrelerinin değiştiği hızı  $1/\alpha$  tanımlar.  $\omega_{j,t}$  Nedensel düşük geçişli filtrelenmiş (eşiklenmiş) sonraki olasılıktır.

$\mu$  ve  $\sigma$  parametreleri eşlenmemiş dağılımlar için aynen kalır. Eş. 6 yukarıda bahsedildiği üzere nedensel düşük geçişli filtrenin tipiyle aynı oluşu durumunda; eşleşen dağılımın parametreleri, j, Eş. 7 ve Eş. 8 şeklinde güncellenir.

$$\rho = \alpha * \Pr(x_t | \mu_{j,t-1}, \sigma_{j,t-1}) \quad (6)$$

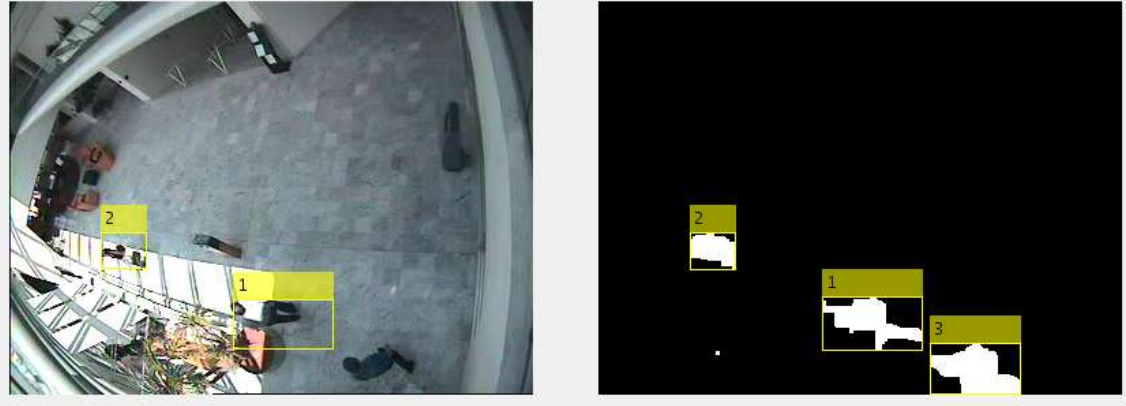
$$\mu_{j,t} = (1 - \rho)\mu_{j,t-1} + \rho x_t \quad (7)$$

$$\sigma_{j,t}^2 = (1 - \rho)\sigma_{j,t-1}^2 + \rho(x_t - \mu_{j,t})^T (x_t - \mu_{j,t}) \quad (8)$$

Bu büyük değişiklikler meydana geldiğinde hem ortalama hem de varyansın aydınlatma değişikliklerini takip etmesini fakat arka plan dağılımını bozulmadan korumayı sağlar [7].

### 2.2.3. Değerlendirme (Evaluation)

Hareket eden nesnelere piksel farkı ve akış hesaplaması ile otomatik olarak tespit edilmiş ve ayrıca zaman içindeki hareketleri takip edilmiştir. Şekil 1 obje yakalama ve takip sürecini göstermektedir. Hareket tahmininde Kalman filtresi kullanılmıştır.

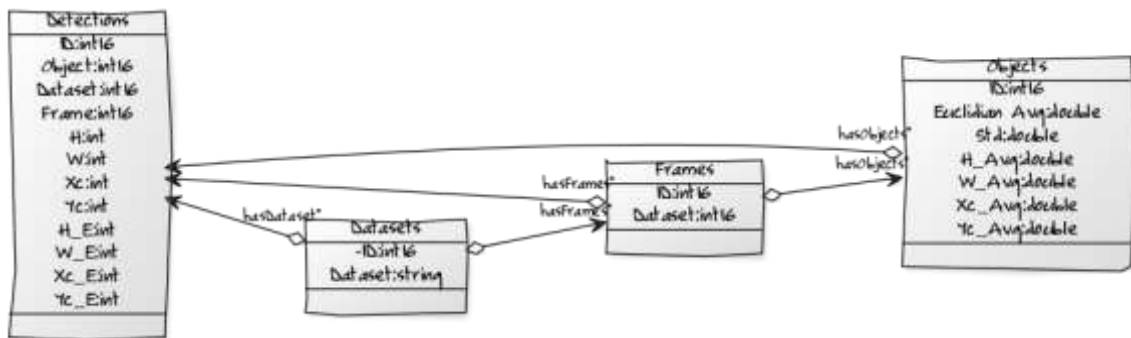


Şekil 1. Objeye yakalama ve takibi (Object detection and tracking)

Yöntemin aynı objenin zaman içindeki hareketlerinin yakalanmasında kullandığı ilişki sadece hareket tabanlıdır. Sadece hareket kullanılması bazı kısıtlara neden olmaktadır. Objenin durması ve tekrar hareket etmesi durumunda ilişki kopmaktadır. Üretilen veriler üzerinde semantik arama yapılarak anlamlar çıkarılacağı için aynı objenin hareketleri çok önemli hale gelmektedir. Bu kısıt önlemek için geliştirilen yaklaşımlar ilerleyen kısımda açıklanmıştır.

### 2.3. Veri İşleme (Data Processing)

Görüntü işleme neticesinde elde edilen veriler XML dosyasına kaydedilmektedir. Görüntülerden elde edilen verileri okumak, işlemek, süzmek, kopan ilişkileri tekrar birleştirmek ve işlenmiş verilerden ontolojileri üretmek için yaklaşımlar geliştirilmiştir. Veri setinden ve görüntü işlemeden sonra kaydedilen dosyalardan veri okuması yapılabilmektedir. Hesaplama işlemi tetiklendiğinde, sırasıyla önce ilişkisel veri tabanı yaklaşımı ile her bir Dataset, çerçeve ve obje detayları Şekil 2’de sunulan UML diyagramına göre tablolara kopyalanır. Daha sonra ilişkisel veri tabanı yaklaşımı ile ele alınmış veriler “Detection” tablosuna kaydedilir. Okunan ve üretilen veriler kullanıcının dikkatine sunulmak için DataGridView nesnelere



yazılmaktadır.

Şekil 2. UML diyagramı (UML diagram)

Videodan toplam kaç çerçevede obje yakalandığı ve yakalanan objelerin kaç çerçeve boyunca yakalanmaya devam ettiği istatistikî bilgisi hesaplanmaktadır ve kullanıcıya dönüt olarak

sunulmaktadır. Bir yandan da her bir yakalama için (Detections tablosunun her bir kaydı için) bir önceki yakalama ile arasındaki Öklid uzaklığı farkı Eş. 9 ile hesaplanarak tabloya kaydedilir.

$$d_{H_i} = \sqrt{h_i - h_{i-1}} \quad (9)$$

Benzer şekilde  $d_{W_i}$ ,  $d_{X_{c_i}}$  ve  $d_{Y_{c_i}}$  de hesaplanır.  $n$  Adet çerçevede toplam  $m$  adet obje yakalandığı ve her bir objenin toplam yakalanma sayısının  $k$  olduğu durumda uzaklıkların ortalaması Eş. 10 ile hesaplanır.

$$\mu_{d_{H_j}} = \prod_{j=1}^m (\sum_{i=1}^n d_{H_i}) / k_j \quad (10)$$

$\mu_{d_{W_j}}$ ,  $\mu_{d_{X_{c_j}}}$  ve  $\mu_{d_{Y_{c_j}}}$  değerleri de benzer şekilde hesaplanır. Objeleri temsil etmek için koordinat kullanılmıştır. Her bir objenin  $x_c$  ve  $y_c$  koordinat değerleri ile bir önceki kayıt ile arasındaki Öklid uzaklığı Eş. 11 ile hesaplanır.

$$d_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (11)$$

Objeye bu uzaklıkların ortalaması  $\mu_d$  ile temsil edilmektedir ve Eş. 12 ile hesaplanır.

$$\mu_{d_j} = \prod_{j=1}^m (\sum_{i=1}^n d_i) / k_j \quad (12)$$

Standart sapma da Eş. 13 ile hesaplanır.

$$\sigma_{d_j} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \mu_j)^2} \quad (13)$$

### 2.3.1. Hız Tabanlı Objeye Eliminasyonu (Speed Based Elimination of Object)

Veri setinde örneğin Browse1 videosunda 4 adet obje olduğu bilinmektedir. Ayrıca yukarıda bahsedildiği üzere kısıtlar nedeniyle objelerin durması ve yeniden hareket etmesi ayrıca kamera görüş alanından çıkıp tekrar görüntüye girmeleri nedeniyle çalışmamız tarafından yeni bir obje olarak yakalanmaktadır. Bu noktada objeye eliminasyonu yapmak için bir yaklaşım önerilmektedir.

Objeye tablosundaki her objeye numarası ile bir  $O$  kümesi oluşturulur.  $O$  Kümesinin her elemanı için objenin yakalandığı çerçeve numarası eklenerek  $F_r$  kümesi oluşturulur. Örneğin  $F_1$  kümesi 1 ID numaralı objenin yakalandığı bütün çerçeve numaralarını barındırır. Burada aynı çerçevede yakalanan farklı objelerin birleştirilmesini önlemek için bir ceza puanı  $pen_r = 10000$  başlatılarak (Çalışmada temsil edilmeyen objeleri de kapsamak için başlangıç değeri verilmiştir.). Eş. 14 formülü ile aynı çerçevede olan farklı objeler hesaplanır.

$$pen_r = \prod_{r=1}^{|O|} \prod_{i=1}^n \begin{cases} pen_r + 10, & (F_r \cap F_i) \neq \emptyset \\ pen_r, & (F_r \cap F_i) = \emptyset \end{cases} \quad (14)$$

Hesaplanan ceza puanı, eğer varsa,  $\mu_d$ 'ye ilave edilir. Her bir nokta diğer noktalar ile Eş. 15'de gösterildiği gibi karşılaştırılarak durum tespit edilir.

$$\prod_{r=1}^{|O|-1} \min_j \left( \prod_{j=r+1}^m \mu_{d_j} \right) \quad (15)$$

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

Tablo 1 ceza puanı hesaplaması sonucunda ortaya çıkan örnek değerleri ve yaklaşımımızı göstermektedir. Örneğin 1 numaralı objeye {2,3,4} numaralı objeler ile aynı çerçevede yer aldığı için bu ceza puanlarına yansımıştır. 1 numaralı objenin {7,8,11,13} numaralı objeler ile kıyaslandığında hiç ceza puanı almadığından aynı çerçevelerde yer almadığı görülmektedir. Yaklaşım gereği sondan başladığı için 13 numaralı objeye en yakın objeye olan ve hiç aynı çerçevede yakalanmayan 7 numaralı objeye ile, 11 - 8 ile, 8 - 1 ile ve 7 - 4 numaralı objeye ile birleştirilmesi önerilmiştir.

Bazı objeler yöntemimiz ile temsil edilememiştir. Bunun nedeni sadece bir çerçevede ve bir kez yakalanmalarıdır. Bir önceki yakalama ile uzaklıkları olmadığından değer sonuçlara 0 olarak

yansımıştır. Temsil edilemeyen  $ND$  kümesindeki noktalar için Eş. 16 denklemleri ile modellenen fark denklemleri uygulanır.

**Tablo .1.** Ceza puanı karşılaştırması (Compare of penalty)

1	2	3	4	5	6	7	8	9	10	11	12	
1												
2	850.0424											
3	972.5254	842.5678										
4	660.0544	660.012	662.5798									
5	10000	10000	10000	10000								
6	10000	10000	10000	10000	10000							
7	0.6313	0.5889	3.1567	0.5769	10000	10000						
8	0.1114	0.1538	2.414	0.1658	10000	10000	0.7427					
9	10000	10000	10000	10000	10000	10000	10000	10000				
10	10000	10000	10000	10000	10000	10000	10000	10000	10000			
11	0.4151	0.4575	2.1103	0.4695	10000	10000	1.0464	0.3037	10000	10000		
12	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10010	
13	1.4472	1.4048	3.9726	1.3928	10000	10000	0.8159	1.5586	10000	10000	281.8623	10000

$$dist_{p,j} = \prod_{p=1}^{|ND|} \prod_{j=1}^m \begin{cases} |\mu_{xc_p} - \mu_{xc_j}| & \mu_{d_p} \neq 0 \\ \mu_{d_p} & \mu_{d_p} = 0 \end{cases} \quad (16)$$

Sonrasında  $ND$  kümesinin her bir elemanı için en düşük fark Eş. 17 formülü ile bulunur.

$$\prod_{p=1}^{|ND|} \min_j (\prod_{j=1}^m dist_{p,j}) \quad (17)$$

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

Bu hesaplamaların ardından eliminasyon tamamlanmaktadır. Örnek de veri setinde olan 4 obje elde edildiği görülmektedir. Eliminasyon yapıldıktan sonra "Objects" tablosundaki  $\mu_d$ , standart sapma,  $\mu_{d_H}$  vb sonuçlar da yeni obje sayısına ve son değerlere göre güncellenmektedir. Sonuçlar kullanıcıya dönüt olarak yansıtılmaktadır.

### 2.3.2. Tahmin Tabanlı Objeye Eliminasyonu (Estimation Based Elimination of Object)

Hız tabanlı obje eliminasyonunun hata oranının yüksek olacağı varsayımından hareketle ikinci bir yaklaşım geliştirilmiştir. Yaklaşım duran objenin aynı konumdan hareket edeceği ve kamera görüş alanından çıkan objenin aynı konumdan görüş alanına geri döneceği varsayımına dayanmaktadır.

Objeye tablosundaki her obje numarası ile bir  $O$  kümesi oluşturulur.  $O$  kümesinin her elemanı için objenin yakalandığı çerçeve numarası eklenerek  $F_r$  kümesi oluşturulur.  $O$  kümesinin her elemanı için  $\min_r F_r$  ve  $\max_r F_r$ ; her bir objenin ilk ve son tespit edildiği çerçeveler belirlenir. Eğer  $\min_{r+1} F_{r+1} < \max_r F_r$  ise,  $r$  ve  $r+1$  objelerinin aynı çerçevede kesiştiği anlamına gelir. Aynı çerçevede yakalanan farklı objeler birleştirilemeyeceği için yok sayılır. Bunun haricindeki kesişmeyen her bir obje sondan başlayarak ( $|O|, |O| - 1, \dots, 1$ ) diğer objeler ile Eş. 18 ile hesaplanır ve Eş. 19 ile karşılaştırılıp en kısa mesafe seçilir.

$$d_r = \prod_{r=|O|}^1 \prod_{s=r-1}^2 \sqrt{(x_r - x_s)^2 + (y_r - y_s)^2} \quad (18)$$

$$\prod_{r=|O|}^2 \min_s (\prod_{s=r-1}^1 d_s) \quad (19)$$

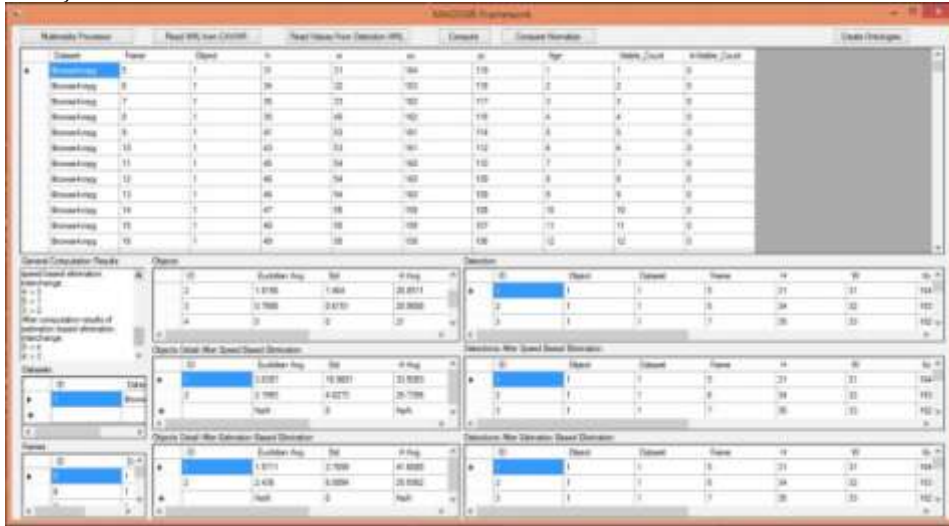
Böylece en yakın iki objenin, sondan başlayarak başa doğru, birleştirilmesi önerilmiş olur.

### 2.3.3. Geliştirilen Program (Developed Software)

Önceki bölümlerde üretilen yaklaşımları kullanan ve uygulayan bir yazılım geliştirilmiştir. Geliştirme ortamı olarak Microsoft Visual Studio :NET kullanılmıştır. Yazılım C# dilinde kodlanmıştır. Taşınabilirlik ve etkinliği sağlamak için veri tabanı yerine XML dosyalar kullanılmıştır. Yazılımın ara yüzünden sadece bir görüntü Şekil 3’de sunulmuştur. Elde edilen bulgular ve sonuçlar sonraki bölümlerde sunularak ele alınmıştır.

## 3. BULGULAR (FINDINGS)

CAVIAR veri setinde bulunan videolardan 12 tanesi üzerinde inceleme ve işleme yapılmıştır. İşlenen videolar Tablo 2’de belirtilmiştir. Veri setinde 12 videoda toplam çerçeve sayısı 16.858 adettir. Kullandığımız yöntem gereği sadece hareket eden objeler ve hareketleri yakalandığından CAVIAR veri setindeki hareket eden objeleri barındıran çerçeveler ile kıyaslama yapılmıştır. Veri setine göre hareket eden obje barındıran çerçeve sayısı ise 7.276 adettir. Böylece hareketli obje barındıran çerçevelerin toplam çerçevelere oranı %43,16 olarak tespit edilmiştir.



Şekil 3. Geliştirilen program (Developed software)

Tablo 2. İşlenen videolar (Processed videos)

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

Toplamda ve genel var-yok analizinde 7.276 çerçeveden yaklaşımımız 6.122 adet çerçevede obje yakalamış, çalışmanın ön doğruluk değeri çerçeve bazında %84,14 olarak ortaya çıkmıştır.

### 3.1. Süreç Verileri (Process Data)

İstatistikî hesaplamalar sonucu her bir video için sonuçlar kullanıcıya bildirim olarak sağlanmaktadır. Browse1 videosu için bir örneği Şekil 4’de sunulmuştur.

Total Frame Count -> 672

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

Şekil 4. İstatistikî sonuç örneği (Statistical results example)

Önceki bölümde bahsedilen hesaplama süreçleri sonrasında elde edilen Browse1 videosunun obje detayları bir örnek olarak Tablo 3’de sunulmuştur. Hız tabanlı yaklaşımın obje yakınlığını belirlemek için karşılaştırdığı ceza puanı matris örneği Tablo 1’de sunulmuştur.

Tablo 3. Tespit edilen obje detay örneği (Example of detected object details).

ID	Euclidian Avg	Std	H Avg	W Avg	Xc Avg	Yc Avg
1	1.8548	12.5767	20.5347	48.8614	248.2376	262.3465
2	1.8124	6.4476	29.9529	43.2706	76.1059	169.4
3	4.3802	7.2761	40.1719	56.2292	202.3281	179.5156
4	1.8004	7.7235	23	51.5152	177.7121	225.5606
5	0	0	21	35	65	136
6	0	0	20	35	65	135
7	1.2235	0.8727	38.8347	38.9153	208.7839	91.4915
8	1.9662	1.8577	45.4348	44.8571	276.1304	141.3106
9	0	0	43	23	299	70
10	0	0	43	23	299	70
11	2.2699	2.0012	43.2	46.4267	239.4133	72.5733
12	0	0	17	41	92	207
13	0.4076	0.8128	16.0714	41.9286	91.8571	209.1071

### 3.2. Üretilen Veriler (Produced Data)

Hız tabanlı obje eleme yaklaşımı ile tahmin tabanlı obje eleme yaklaşımının karşılaştırılması Tablo 4’de sunulmuştur. Örneğin orijinal veri setinde Browse1 videosunda 4 obje belirtilmiştir. Yöntemimiz ile yakalanan 13 obje her iki yaklaşım ile işlendiğinde ve birbiriyle aynı olan objeler elendiğinde geriye toplam 4 obje kalmıştır. Ancak obje eliminasyon sonuçları farklılaşmaktadır. Yaklaşımların ürettiği sonuçlar %50 örtüşürken, %50 objede ise farklılaşmaktadır. İşlediğimiz tüm videolar için karşılaştırmalı obje eliminasyon sonuçları Tablo 4’de sunulmuştur.



**Tablo 4.** Eliminasyon karşılaştırma (Compare of Elimination)

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

Tablo 5’de ve Tablo 6’de sırasıyla hız tabanlı yaklaşım ve tahmin tabanlı yaklaşım ile elde edilen objelerin detayları sunulmuştur. Tablo 7’da ise veri setindeki objelerin detayları sunulmuştur.

**Tablo 5.** Hız tabanlı eliminasyon sonuçları (Speed based elimination results)

Dataset	Id	Euclidian avg	Std	H avg	W avg	Xc avg	Yc avg
Browse_WhileWaiting1	1	1.8076	7.1184	25.2011	53.8122	136.1958	188.0952
Browse_WhileWaiting2	1	5.6709	17.5323	17.9722	45.9537	171.8889	258.5278
Browse_WhileWaiting2	2	2.8016	17.6699	28.4425	43.6626	114.8386	107.6601
Browse1	1	2.5313	11.1266	37.5074	46.2684	259.8319	161.7434
Browse1	2	3.7734	16.9778	29.7356	43.0805	75.8506	168.6207
Browse1	3	5.5491	17.4578	40.1719	56.2292	202.3281	179.5156
Browse1	4	3.0236	18.7688	33.6858	41.6888	192.3444	128.5227
Browse2	1	2.1375	12.6461	38.1556	38.2353	228.5218	106.685
Browse3	1	20.9589	47.8804	34.3347	43.3247	197.1175	128.2789
Browse4	1	3.0397	16.9881	33.5093	48.2453	170.6584	178.8571
Browse4	2	2.1993	4.8273	25.7356	51.8966	122.6264	174.7759
Rest_FallOnFloor	2	2.5764	14.5427	38.2306	42.4854	204.6359	133.1505
Rest_FallOnFloor	3	4.4642	9.6208	54.344	60.656	134.088	78.608
Rest_SlumpOnFloor	1	4.1703	12.5779	39.7576	63.2929	184.798	188.9798
Rest_SlumpOnFloor	2	7.0502	25.2375	19.9625	44.6375	118.5375	192.825
Rest_SlumpOnFloor	3	4.7565	19.6053	25.8333	45.2619	162.2857	182.7976
Rest_SlumpOnFloor	4	2.4001	12.7735	35.8953	42.1172	216.7609	124.9891
Rest_WiggleOnFloor	1	2.9321	6.287	39.863	50.8852	196.5889	162.6852
Rest_WiggleOnFloor	2	3.0645	11.252	35.9055	44.4073	182.4727	106.0873
Walk1	1	2.6769	1.8284	34.506	46.4518	215.4096	182.4337
Walk1	3	3.3118	21.8238	27.2135	39.8652	104.3071	152.3146
Walk2	1	4.2342	28.134	35.0625	35.975	131.9375	128.9
Walk2	2	2.7684	10.2751	28.0301	46.3871	162.2602	164.2452
Walk3	1	2.4861	3.9963	35.9877	42.2408	180.2681	99.0561

**Tablo 6.** Tahmin tabanlı eliminasyon sonuçları (Estimation based elimination results)

Dataset	Id	Euclidian avg	Std	H avg	W avg	Xc avg	Yc avg
Browse_WhileWaiting1	1	1.8076	7.1184	25.2011	53.8122	136.1958	188.0952
Browse_WhileWaiting2	1	4.6094	12.4072	17.1258	48.8742	165.5157	263.6478
Browse_WhileWaiting2	2	1.7109	11.8789	30.3101	42.0391	109.5419	83.8939
Browse1	1	1.8548	12.5767	20.5347	48.8614	248.2376	262.3465
Browse1	2	18.5937	51.1272	38.7458	42.1983	206.0119	119.7593
Browse1	3	5.2588	13.9004	40.1719	56.2292	202.3281	179.5156
Browse1	4	4.7692	25.0927	23.0000	51.5152	177.7121	225.5606
Browse2	1	2.1375	12.6461	38.1556	38.2353	228.5218	106.6850
Browse3	1	20.9589	47.8804	34.3347	43.3247	197.1175	128.2789
Browse4	1	1.9711	3.7699	41.6688	47.7375	215.0125	160.2500
Browse4	2	2.4360	6.8894	25.5982	50.3780	124.6637	185.6042
Rest_FallOnFloor	1	4.5456	15.5278	41.9814	46.7151	188.2142	120.4544
Rest_SlumpOnFloor	1	4.1703	12.5779	39.7576	63.2929	184.7980	188.9798
Rest_SlumpOnFloor	2	7.0502	25.2375	19.9625	44.6375	118.5375	192.8250
Rest_SlumpOnFloor	3	4.7565	19.6053	25.8333	45.2619	162.2857	182.7976
Rest_SlumpOnFloor	4	2.4001	12.7735	35.8953	42.1172	216.7609	124.9891
Rest_WiggleOnFloor	1	2.9321	6.2870	39.8630	50.8852	196.5889	162.6852
Rest_WiggleOnFloor	2	3.0645	11.2520	35.9055	44.4073	182.4727	106.0873
Walk1	1	2.2679	4.1328	30.4360	46.2965	174.5233	165.2965
Walk1	2	1.7846	10.1861	28.3596	27.2921	40.1348	158.3146
Walk2	1	4.2342	28.1340	35.0625	35.9750	131.9375	128.9000
Walk2	2	2.7684	10.2751	28.0301	46.3871	162.2602	164.2452
Walk3	1	2.4861	3.9963	35.9877	42.2408	180.2681	99.0561

**Tablo 7.** Veri seti obje detayları (Benchmark object details)

Dataset	Id	Euclidian avg	Std	H avg	W avg	Xc avg	Yc avg
Browse_WhileWaiting1	0	0.8591	0.7916	18.7484	23.6774	32.7548	39.7161
Browse_WhileWaiting1	1	0.3375	0.7097	19.7941	36.8752	105.1591	259.5008
Browse_WhileWaiting1	2	0.8607	0.9237	23.9784	44.3098	119.4863	181.2922
Browse_WhileWaiting2	0	0.2493	0.6302	29.5749	34.7098	138.6223	89.0251
Browse1	0	1.7665	1.5616	37.4108	47.6929	243.1162	212.0788
Browse1	1	1.0599	1.0809	18.6352	36.8893	80.1393	174.5328
Browse1	2	0.1391	0.4435	17.1026	34.2186	87.6079	207.0642
Browse1	3	1.0816	1.0818	35.5346	32.9633	219.4556	127.33
Browse2	0	0.5714	0.7473	16.75	20.25	77.6429	66.0714
Browse2	1	2.1278	1.8722	37.4478	47.313	226.2913	194.3522
Browse2	2	0.3727	0.5963	18.7949	22.2436	76.6923	67
Browse2	3	0.684	0.9316	38.1373	28.8307	241.467	83.5561
Browse3	0	1.4175	1.5491	35.99	44.37	228.895	183.015
Browse3	1	0.7752	1.0046	26.6373	30.4707	146.8873	96.9568
Browse3	2	0.8915	1.0898	11.1111	10.7778	7.2222	74.1728
Browse3	3	0.7934	1.225	10.8472	12.7361	8.1528	70.8056
Browse3	4	1	0.7071	6	19.3333	177.6667	283.6667
Browse4	0	1.8941	1.6064	45.7171	49.7303	226.4342	164.0658
Browse4	1	0.8189	1.0347	20.7969	42.9672	98.3777	183.4956
Browse4	2	1.0769	1.2096	19.6568	36.9742	89.5978	181.8118
Rest_FallOnFloor	0	0.6276	0.6799	29.7288	31.0452	146.3503	49.5876
Rest_FallOnFloor	1	0.788	1.1508	28.563	34.5793	239.0447	196.3923
Rest_FallOnFloor	2	0.789	1.0051	36.8482	38.222	186.6338	103.296
Rest_SlumpOnFloor	0	0.7814	0.9515	18.985	31.635	63.795	144.9
Rest_SlumpOnFloor	1	1.3788	1.6088	35.1728	45.0266	223.3256	181.0266
Rest_SlumpOnFloor	2	1.7881	1.2467	24.4286	40.6104	195.5065	239.3377
Rest_SlumpOnFloor	5	0.8776	1.2102	33.2249	36.0957	219.6061	118.4625
Rest_WiggleOnFloor	0	0.9021	1.4201	21.4	27.3109	279.8605	241.6504
Rest_WiggleOnFloor	1	1.5426	1.2175	30.9319	35.6553	175.3404	113.0851
Rest_WiggleOnFloor	2	0.4119	0.6216	31.572	40.2825	184.5091	79.8
Walk1	0	2.3576	1.7184	35.4847	48.2331	231.5583	195.0736
Walk1	1	1.1703	1.1724	23.1522	35.2029	137.8913	142.1812
Walk1	4	0.1654	0.4258	16.3287	33.2552	49.5962	178.5017
Walk1	5	0.1917	0.4546	16.0255	29.3193	49.719	163.7044
Walk2	0	1.7872	1.2595	22.1957	35.2391	215.5	260.5
Walk2	1	0.0936	0.3062	28.4171	26.855	116.3517	52.0626
Walk2	2	0.0492	0.2253	24.5896	27.7412	98.3194	63.3441
Walk2	3	2.4691	1.625	30.6442	44.8773	226.3681	217.1534
Walk2	4	1.1477	1.2938	20.4029	31.45	107.8676	128.7441
Walk2	5	0.7714	0.8165	17.9843	28.4392	75.7216	110.0667
Walk3	0	2.3741	1.6316	32.0983	45.7168	241.4277	202.3295
Walk3	1	0.8508	0.9351	8.5	23.5098	33.8039	193.451
Walk3	2	1.1613	0.9793	24.5096	31.0955	135.914	71.5382
Walk3	3	1.2235	0.989	27.4806	33.0424	128.3816	87.2862
Walk3	4	2.0066	1.3563	38.485	27.7605	270.1916	118.6826

### 3.3. Yaklaşımların Karşılaştırması (Comparison of Approaches)

Hız tabanlı yaklaşım ile tahmin tabanlı eliminasyon sonuçları karşılaştırıldığında, iki yaklaşımın bir birine çok yakın sonuçlar ürettiği görülmüştür. Elde ettiğimiz mevcut sonuçlara göre birbirlerine herhangi bir üstün yanı tespit edilememiştir. Bu değerlendirme birden çok kameraya sahip bir ağda sonuçlar karşılaştırılınca tekrar yapılmak üzere ötelenmiştir.

### 3.4. Deneysel Çalışma (Empirical Study)

12 videonun her biri için yakalanan hareketli obje sayısı var-yok analizinde CAVIAR veri setinde yakalanmış nesnelerin çalışmamızda yanlışlıkla yakalanmama oranı (yalancı negatiflik – false negative) 0,71 olarak hesaplanmaktadır. Böylece hassaslık değeri %29 olarak hesaplanmaktadır.

Her bir videonun her bir çerçevesi için var-yok analizi yapmak hesaplama maliyeti açısından anlamlı değildir. Değerlendirme açısından sadece Browse1 videosuna uygulanmıştır. Sonuçlara göre veri setinde yakalanmış nesnelerin çalışmamızda yanlışlıkla yakalanmama oranı (yalancı negatiflik – false negative) 0,45 olarak hesaplanmıştır. Hassaslık değeri %55 olarak ortaya çıkmaktadır.

### 3.5. Geçmiş Çalışmalarla Karşılaştırma (Comparison With Recent Studies)

Genelde değerlendirme metriği olarak hassasiyet (precision) ve yeniden çağırma (recall) kullanılır. Hassasiyet doğru alınmış eşleşmelerin toplam alınmış eşleşmelere oranı iken, yeniden çağırma doğru alınmış eşleşmelerin toplam kesin referans oluşumlara oranıdır. Hassasiyet erişim kümesindeki doğru eşleşmelerin yüzdesi, yeniden çağırma ise yapı içerisinde gerçekleşen bütün örneklerden alınan doğru eşleşmelerin yüzdesini ifade eder [8]. Ortalama hassasiyet ise hem sınıflandırma hem de obje yakalama için kullanılan ve hassasiyeti ifade etmekte kullanılan bir yöntemdir [9]. Bazı durumlarda yeniden çağırma sonuçları ani düşüşler gösterdiğinden [9], çalışmamızda hesaplama maliyetinden de kaçınarak yeniden çağırma hesaplanmamıştır.

**Tablo 8.** Bezer çalışmaların başarı durumu

	Ortalama Hassasiyet (%)	Minimum Has. (%)	Maksimum Has. (%)
<b>Yıldırım</b> [10]		62.50	94.23
<b>Sudha</b> [11]		50	75
<b>Amir</b> [8]		21	23
<b>You</b> [12]		50.2	94.8
<b>Zhang</b> [13]		42	92
<b>Everingham</b> [9]	22.1		

Çalışmaların uygulandıkları domainler genelde farklıdır, çoğunda birden çoktur. Her bir yaklaşımın kullandığı yöntem de farklıdır. Burada videodan semantik içerik çıkarılması kapsamında bir çerçevenin kabul edilebilir olması için yakalaması gereken aralığı göstermek için bir karşılaştırma yapılmıştır. Sonuçları Tablo 8’de sunulmuştur.

## 4. SONUÇ VE TARTIŞMA (CONCLUSION AND DISCUSSION)

Sistemin geliştirilmesinin asıl amacı işlenen multimedya verilerinin modellenerek ontolojiler oluşturulması ve ontolojiler üzerinde SPARQL sorgulamalar ile anlamlandırma yapılmasıdır. Çalışmanın geliştirmeyi hedeflediğimiz çerçeveye, anlamlandırma basamağında, kaynak teşkil edecek doğruluk ve hassasiyet oranını yakalaması amaçlanmıştır.

Obje sayısı ile yapılan ölçmenin yetersiz olduğu düşünülmektedir. Bu değerlendirmeye göre hassaslık beklenenden düşük çıkmıştır. Yine de benzer çalışmalarda %40 civarlarında sonuçların kabul edilebilir olduğu dikkate değerdir. Ancak sayı yerine obje detaylarına göre bir değerlendirme de maliyet-etkin görünmemektedir.

Çalışmada her iki yaklaşımda elde ettiğimiz verilerin standart sapma değerlerinin çok yüksek olduğu görülmüştür. Doğruluğu ve hassasiyeti artırmak için bazı verilerin eliminasyonu yapılabilir. Gürültü niteliğinde olan yakalamaların temizlenmesi ile daha hassas veriler elde edilmesi mümkün kılınabilir. Bunun için Chauvenet ve benzeri yöntemler kullanılabilir.

Geliştirdiğimiz yaklaşım ve prototip ile multimedya üzerinde:

- Ekrandan veya bir noktadan geçişler sayılabilmektedir
- Hareketli obje tanıma ve takibi yapılabilmektedir
- Kapalı sistem bir video ortamında, hareket eden objelerin hız ve koordinata göre tanıma ve takibi sürdürülebilmektedir

## 5. KAYNAKLAR (REFERENCES)

- [1] "Statistics," *YouTube*. Available: <http://www.youtube.com/yt/press/statistics.html>. [Accessed: 12-May-2015].
- [2] Gang Zhai, G. C. Fox, M. Pierce, Wenjun Wu, and H. Bulut, "eSports: Collaborative and Synchronous Video Annotation System in Grid Computing Environment," in *Seventh IEEE International Symposium on Multimedia (ISM'05)*, 2005, vol. 2005, pp. 95–103.
- [3] R. Troncy, H. Benoit, and S. Simon, "Introduction," in *Multimedia Semantics*, R. Troncy, H. Benoit, and S. Simon, Eds. West Sussex: Wiley, 2011, pp. 1–5.
- [4] O. Menemencioglu and İ. M. Orak, "A Review on Semantic Text and Multimedia Retrieval and Recent Trends," *International Journal of Multimedia Data Engineering and Management*, vol. 6, no. 1, pp. 54–74, Jan. 2015.
- [5] N. Friedman and S. Russell, "Image Segmentation in Video Sequences: A Probabilistic Approach," *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pp. 175–181, 1997.
- [6] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 2, no. 2, 2004.
- [7] W. E. L. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231)*, pp. 22–29, 1998.
- [8] A. Amir, S. Basu, G. Iyengar, C. Y. Lin, M. Naphade, J. R. Smith, S. Srinivasan, and B. Tseng, "A multi-modal system for the retrieval of semantic video events," *Computer Vision and Image Understanding*, vol. 96, no. 2 SPEC. ISS., pp. 216–236, 2004.
- [9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.
- [10] Y. Yildirim, A. Yazici, and S. Member, "in Videos Using a Fuzzy Ontology and Rule-Based Model," vol. 25, no. 1, pp. 47–61, 2013.
- [11] D. Sudha and J. Priyadarshini, "Reducing Semantic Gap in Video Retrieval with Fusion : A survey," *Procedia - Procedia Computer Science*, vol. 50, pp. 496–502, 2015.
- [12] J. You, G. Liu, and A. Perkis, "A semantic framework for video genre classification and event analysis," *Signal Processing: Image Communication*, vol. 25, no. 4, pp. 287–302, 2010.
- [13] Y. Zhang, W. Qu, and D. Wang, "Action-scene Model for Human Action Recognition from Videos," *AASRI Procedia*, vol. 6, no. 0, pp. 111–117, 2014.

## Teşekkür (Acknowledgments)

Bu çalışma KBÜ Araştırma Projeleri Koordinatörlüğü tarafından KBÜ-BAP-13/2-AA-057 numaralı proje ile desteklenmiştir.